

EEN020 Computer Vision
Assignment 4
Model Fitting and Local Optimization

Vasiliki Kostara

December 2023

Robust Epipolar Geometry and Two-View Reconstruction

Theoretical Exercise 1

If $P_1 = [R_1 \ t_1]$ and $P_2 = [R_2 \ t_2]$, then we can apply the Euclidean transformation:

$$H = \begin{pmatrix} R_1^T & -R_1^T t_1 \\ \mathbf{0} & 1 \end{pmatrix}$$

on each camera. Then, we have:

$$P_1 H = [R_1 \ t_1] \begin{pmatrix} R_1^T & -R_1^T t_1 \\ \mathbf{0} & 1 \end{pmatrix} = [R_1 R_1^T \ -R_1 R_1^T t_1 + t_1] = [I \ \mathbf{0}]$$

and

$$P_2 H = [R_2 \ t_2] \begin{pmatrix} R_1^T & -R_1^T t_1 \\ \mathbf{0} & 1 \end{pmatrix} = [R_2 R_1^T \ -R_2 R_1^T t_1 + t_2] = [R \ \mathbf{T}]$$

With the new camera pair being $P_1 = [I \ \mathbf{0}]$ and $P_2 = [R \ \mathbf{T}]$, the essential matrix is:

$$E = [-R_2 R_1^T t_1 + t_2] \times R_2 R_1^T = [\mathbf{T}] \times R$$

where $R = R_2 R_1^T$ and $\mathbf{T} = -R_2 R_1^T t_1 + t_2$.

Theoretical Exercise 2

The essential matrix is defined as $E = [\mathbf{t}]_\times R$, where $[\mathbf{t}]_\times$ is an antisymmetric matrix and R is a rotation matrix. In this case, $[\mathbf{t}]_\times$ is rank-deficient and of dimension 3. However, an antisymmetric matrix has at least rank equal to two, therefore $\text{rank}([\mathbf{t}]_\times) = 2$. Then, the rotation matrix has three degrees of freedom, one for rotation in each dimension, therefore $\text{rank}(R) = 3$. Consequently, the rank of E is given by:

$$\text{rank}(E) = \text{rank}([\mathbf{t}]_\times) + \text{rank}(R) = 2 + 3 = 5$$

The minimal number of point correspondences needed to determine E is its rank, i.e. 5. Moreover, as the name suggests, the point correspondences required by the eight point solver to determine E are *at least* 8.

The probability of an outlier is 25%:

$$1 - \epsilon = 0.25 \Leftrightarrow \epsilon = 0.75$$

The probability to find an outlier free sample set is 99%:

$$\alpha = 0.99$$

For the eight point algorithm ($s = 8$) the number of iterations T is greater or equal to the following fraction rounded up to the nearest integer:

$$T \geq \left\lceil \frac{\log(1 - \alpha)}{\log(1 - \epsilon^s)} \right\rceil = \left\lceil \frac{\log(1 - 0.99)}{\log(1 - 0.75^8)} \right\rceil \approx \lceil 43.66 \rceil = 44$$

Therefore, the number of iterations is:

$$T \geq 44$$

Computer Exercise 1

Before RANSAC

The RMS distance between the inliers and the epipolar lines is

$$e_{RMS} = 155.96$$

This value is quite large, which shows that computing the essential matrix by using all the points is unreliable.

By observing Figure 1, the epipolar errors are quite large. Even though most lie closer to zero, they reach very large values.

Figures 2 and 3 show that the randomly chosen points do not always overlap with the corresponding epipolar lines. Also, the epipolar lines do not look reasonable, as they intersect within the images.

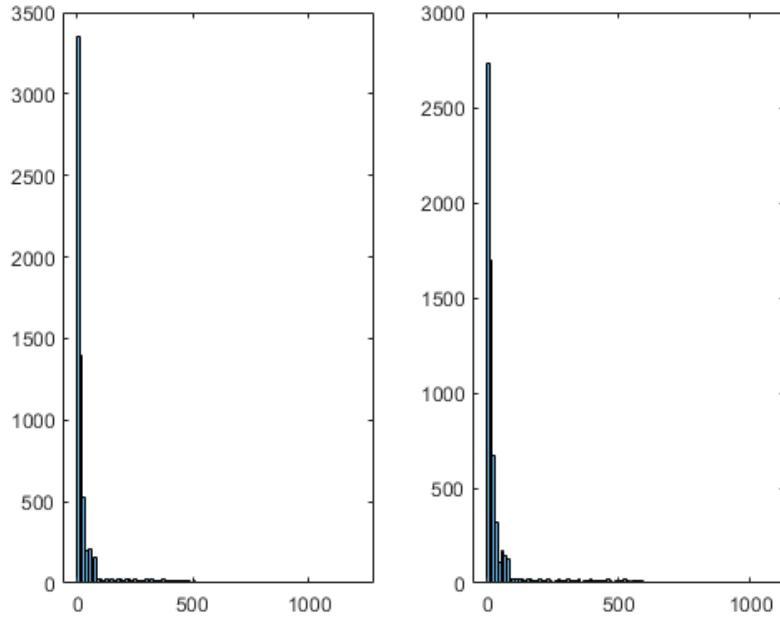


Figure 1: Histograms of epipolar errors for image 1 (left) and image 2 (right) by using all points.

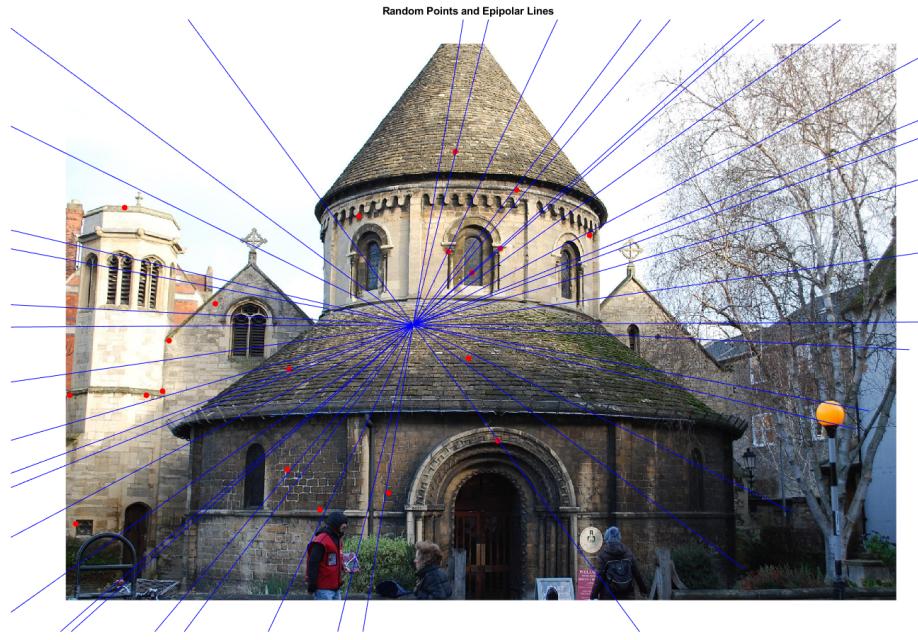


Figure 2: 20 random points and corresponding epipolar lines for image 1.

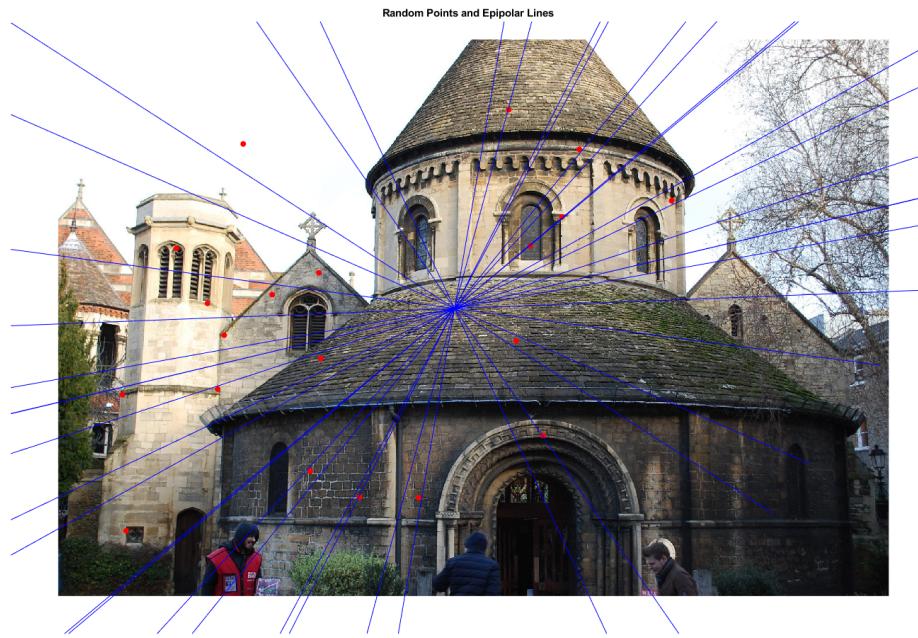


Figure 3: 20 random points and corresponding epipolar lines for image 1.

After RANSAC

For each run with the RANSAC method, the results are slightly different, because the 8 points are chosen randomly. For one of the runs, the results are presented below.

The total number of inliers is 5253.

The RMS distance between the inliers and the epipolar lines is

$$e_{RMS} = 0.0867$$

This value is much smaller than what we had before RANSAC, which indicates the success of the implementation.

By observing Figure 4, the epipolar errors are significantly closer to zero. They can be further optimized by lowering the inlier pixel threshold.

Furthermore, Figures 5 and 6 show that all the randomly chosen inliers belong to the corresponding epipolar lines. In contrast with the previous images, the epipolar lines look corrected and reasonable.

In conclusion, the better estimate of the essential matrix is with RANSAC, because we set a threshold between the points and the corresponding epipolar lines.

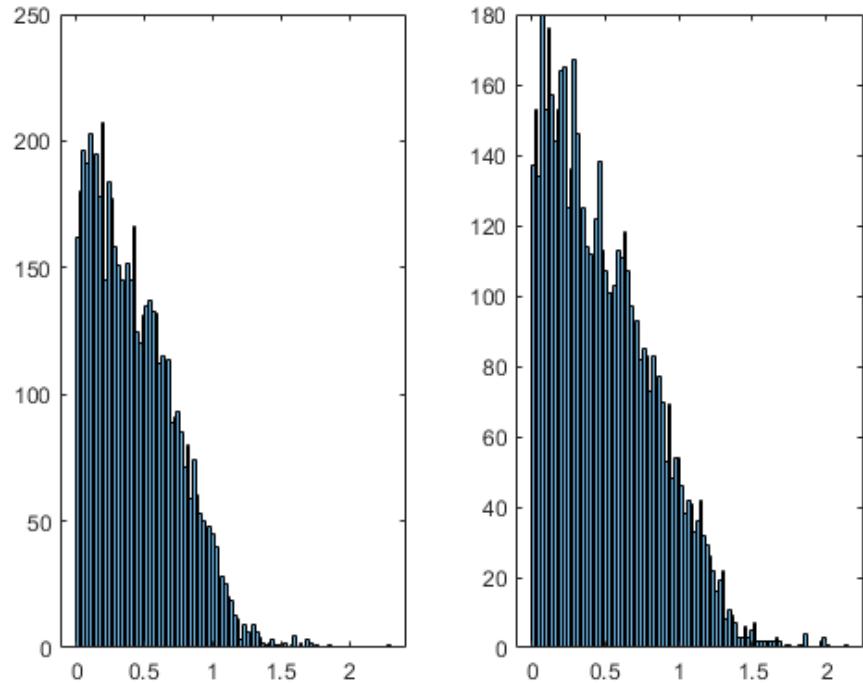


Figure 4: Histograms of epipolar errors for image 1 (left) and image 2 (right) by using only inliers.

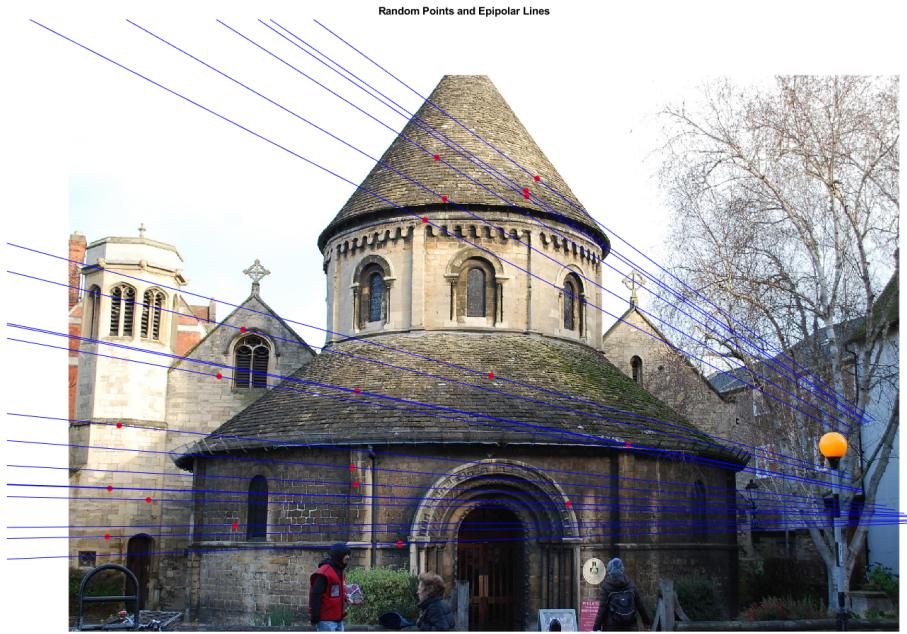


Figure 5: 20 random inliers and corresponding epipolar lines for image 1.

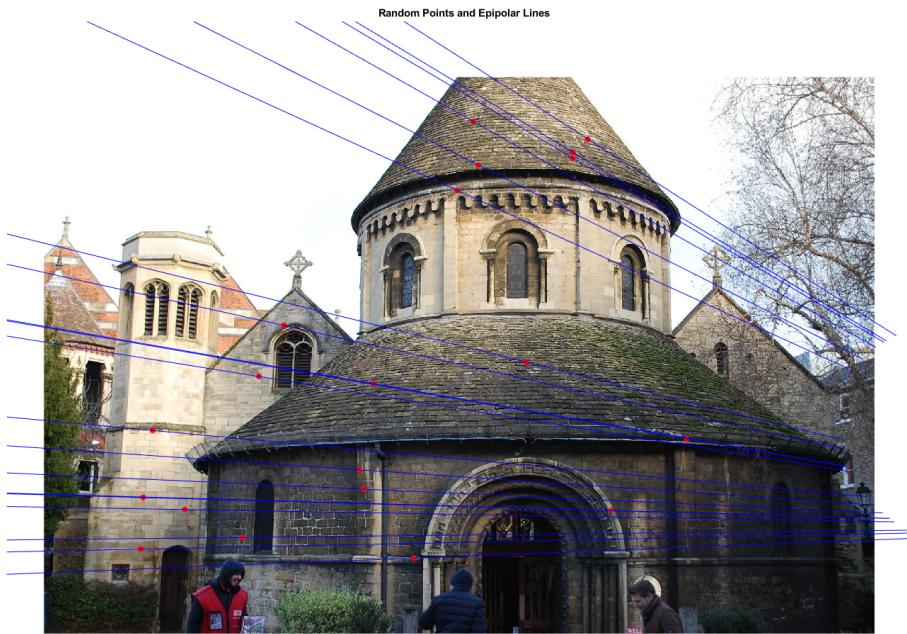


Figure 6: 20 random inliers and corresponding epipolar lines for image 2.

Computer Exercise 2

The number of SIFT points found is 2604.

By using RANSAC, the results are different every time due to randomness.
Below are the results for one of these runs.

The number of inliers is 1598.

The essential matrix is:

$$E = \begin{pmatrix} -0.1097 & 5.9953 & -0.7562 \\ 9.3106 & 0.8452 & 14.3668 \\ -0.3644 & -16.0159 & 1.0 \end{pmatrix}$$

By observing the 3D reconstructions of the inliers in Figure 7, we see that the camera with all the inliers in front is given by:

$$P_2 = [UWV^T \ u_3] = \begin{pmatrix} 0.5941 & -0.0900 & -0.7994 & 0.9358 \\ 0.0384 & 0.9958 & -0.0835 & 0.0248 \\ 0.8035 & 0.0189 & 0.5950 & 0.3516 \end{pmatrix}$$

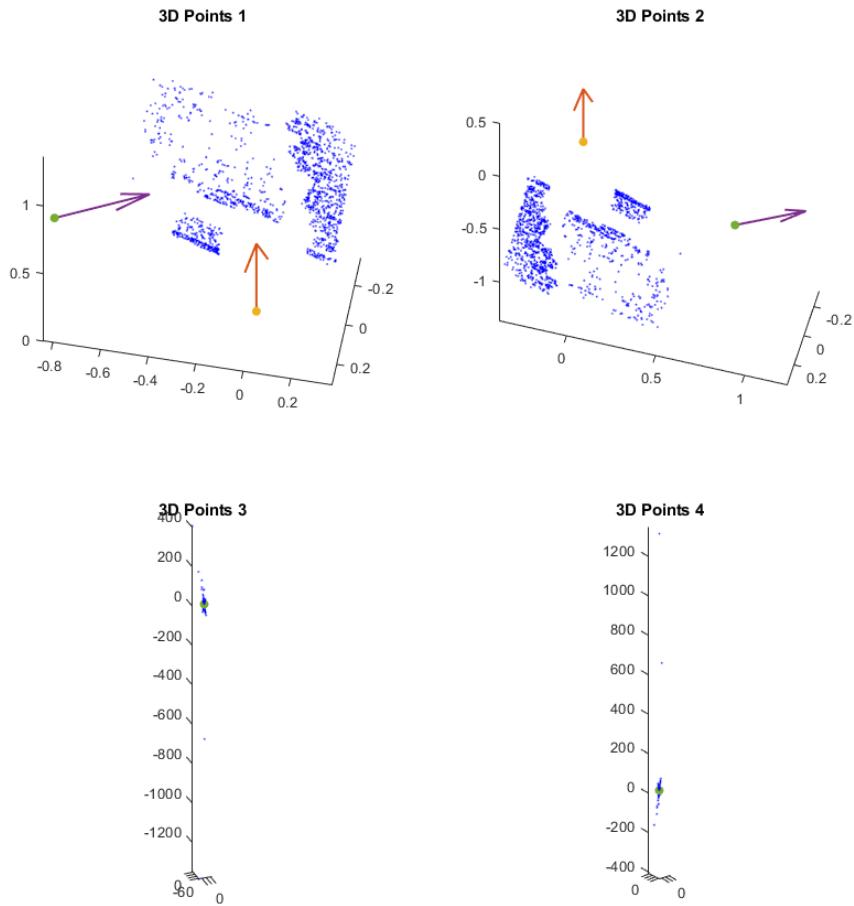


Figure 7: 3D reconstructions of the inliers for the four possible cameras P_2 :

1. $[UWV^T \quad -u_3]$, upper left
2. $[UWV^T \quad -u_3]$, upper right
3. $[UW^T V^T \quad u_3]$, lower left
4. $[UW^T V^T \quad -u_3]$, lower right

+

Levenberg-Marquardt for Structure from Motion Problems

Theoretical Exercise 3

a)

Let $\mathbf{r}_i(\mathbf{X}_j) = \begin{bmatrix} r_{i,1}(\mathbf{X}_j) \\ r_{i,2}(\mathbf{X}_j) \end{bmatrix} = \begin{bmatrix} x_{ij,1} - \frac{P_i^1 \mathbf{X}_j}{P_i^3 \mathbf{X}_j} \\ x_{ij,2} - \frac{P_i^2 \mathbf{X}_j}{P_i^3 \mathbf{X}_j} \end{bmatrix}$. Then, the Jacobian of $\mathbf{r}_i(\mathbf{X}_j)$

will be:

$$\begin{aligned} \mathbf{J}_i(\mathbf{X}_j) &= \begin{bmatrix} \frac{\partial r_{i,1}}{\partial \mathbf{X}_j} \\ \frac{\partial r_{i,2}}{\partial \mathbf{X}_j} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial \mathbf{X}_j} \left(x_{ij,1} - \frac{P_i^1 \mathbf{X}_j}{P_i^3 \mathbf{X}_j} \right) \\ \frac{\partial}{\partial \mathbf{X}_j} \left(x_{ij,2} - \frac{P_i^2 \mathbf{X}_j}{P_i^3 \mathbf{X}_j} \right) \end{bmatrix} = \\ &= \begin{bmatrix} -P_i^1 (P_i^3 \mathbf{X}_j)^{-1} + P_i^1 \mathbf{X}_j \frac{P_i^3}{(P_i^3 \mathbf{X}_j)^2} \\ -P_i^2 (P_i^3 \mathbf{X}_j)^{-1} + P_i^2 \mathbf{X}_j \frac{P_i^3}{(P_i^3 \mathbf{X}_j)^2} \end{bmatrix} = \\ &= \begin{bmatrix} \frac{P_i^1 \mathbf{X}_j}{(P_i^3 \mathbf{X}_j)^2} P_i^3 - \frac{1}{P_i^3 \mathbf{X}_j} P_i^1 \\ \frac{P_i^2 \mathbf{X}_j}{(P_i^3 \mathbf{X}_j)^2} P_i^3 - \frac{1}{P_i^3 \mathbf{X}_j} P_i^2 \end{bmatrix} \end{aligned}$$

b)

First, we consider that $\mathbf{P}_i^l \in \mathbb{R}^{1 \times 4}$ and $\mathbf{X}_j \in \mathbb{R}^{4 \times 1}$, due to their definitions. This means that each operation $\mathbf{P}_i^l \mathbf{X}_j$ results in a scalar, $i = 1, \dots, m$ and $j = \text{constant}$. In the end, we will consider the case $j = 1, \dots, n$.

$$\text{If } \mathbf{r}_i(\mathbf{X}_j) = \begin{bmatrix} r_{i,1}(\mathbf{X}_j) \\ r_{i,2}(\mathbf{X}_j) \end{bmatrix} = \begin{bmatrix} x_{ij,1} - \frac{P_i^1 \mathbf{X}_j}{P_i^3 \mathbf{X}_j} \\ x_{ij,2} - \frac{P_i^2 \mathbf{X}_j}{P_i^3 \mathbf{X}_j} \end{bmatrix}, \text{ then } \mathbf{r} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_m \end{bmatrix} = \begin{bmatrix} r_{1,1}(\mathbf{X}_j) \\ \vdots \\ r_{m,1}(\mathbf{X}_j) \\ r_{m,2}(\mathbf{X}_j) \end{bmatrix}.$$

This shows that $\mathbf{r} \in \mathbb{R}^{2m \times 1}$.

Continuing in a similar manner, the operations in each row vector component i of the Jacobian result in $\mathbf{J}_i \in \mathbb{R}^{2 \times 4}$ and the full Jacobian is

$$J = \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_m \end{bmatrix} = \begin{bmatrix} [\mathbf{J}_1^{(1)}(\mathbf{X}_j)] \\ [\mathbf{J}_1^{(2)}(\mathbf{X}_j)] \\ \vdots \\ [\mathbf{J}_m^{(1)}(\mathbf{X}_j)] \\ [\mathbf{J}_m^{(2)}(\mathbf{X}_j)] \end{bmatrix}, \text{ which means that } J \in \mathbb{R}^{2m \times 4}. \quad 8$$

Lastly, for $j = 1, \dots, n$ the dimensions become:

$$\mathbf{r} \in \mathbb{R}^{2m \times n} \quad \text{and} \quad J \in \mathbb{R}^{2m \times 4n}$$

because we have

$$r = \begin{bmatrix} [r_{1,1}(\mathbf{X}_1)] & \cdots & [r_{1,1}(\mathbf{X}_n)] \\ [r_{1,2}(\mathbf{X}_1)] & \ddots & [r_{1,2}(\mathbf{X}_n)] \\ \vdots & \ddots & \vdots \\ [r_{m,1}(\mathbf{X}_1)] & \cdots & [r_{m,1}(\mathbf{X}_n)] \\ [r_{m,2}(\mathbf{X}_1)] & \cdots & [r_{m,2}(\mathbf{X}_n)] \end{bmatrix} \quad \text{and} \quad J = \begin{bmatrix} [\mathbf{J}_1^{(1)}(\mathbf{X}_1)] & \cdots & [\mathbf{J}_1^{(1)}(\mathbf{X}_n)] \\ [\mathbf{J}_1^{(2)}(\mathbf{X}_1)] & \ddots & [\mathbf{J}_1^{(2)}(\mathbf{X}_n)] \\ \vdots & \ddots & \vdots \\ [\mathbf{J}_m^{(1)}(\mathbf{X}_1)] & \cdots & [\mathbf{J}_m^{(1)}(\mathbf{X}_n)] \\ [\mathbf{J}_m^{(2)}(\mathbf{X}_1)] & \cdots & [\mathbf{J}_m^{(2)}(\mathbf{X}_n)] \end{bmatrix}$$



Computer Exercise 3

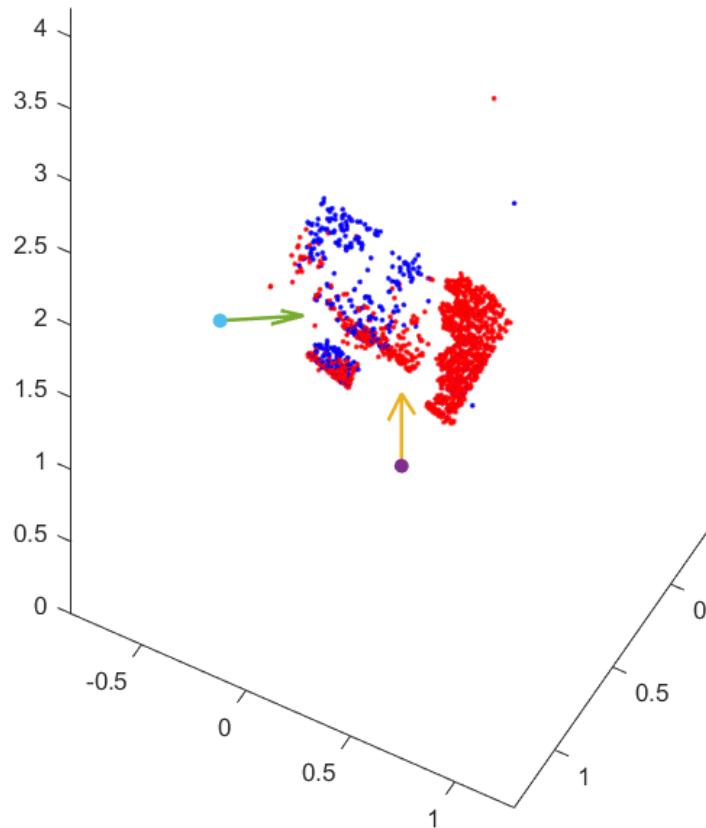


Figure 8: 3D reconstructions before (blue) and after (red) optimization.

<i>reprojection error</i>	total	median
initial 3D points	$2.2354 \cdot 10^4$	11.6599
optimized 3D points	$2.1566 \cdot 10^4$	11.1979

Table 1

By looking at Table 1, we can conclude that after the LM optimization the reprojection error, both total and median, is reduced, however not significantly.

+

Computer Exercise 4

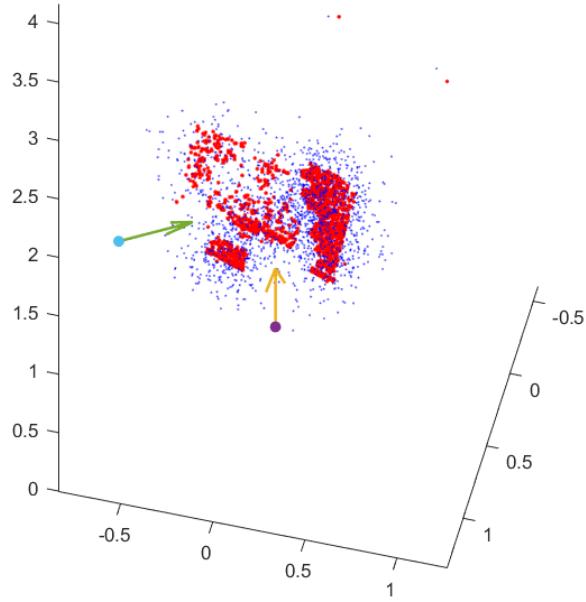


Figure 9: 3D reconstructions with noise before (blue) and after (red) optimization.

<i>reprojection error</i>	total	median
initial 3D points with noise	$8.9392 \cdot 10^8$	$33.044 \cdot 10^4$
optimized 3D points	$3.6797 \cdot 10^4$	10.6015

Table 2

By looking at Table 2, we can conclude that introducing noise to the 3D model increases the total and median of the initial model substantially. However, the reprojection errors for the LM optimized points are comparable to those in Table 1.

0.1 Theoretical Exercise 4

With a matrix $M \succ 0$ and the vector $\nabla F(v)$, where F is a function $F : \mathbb{R}^N \rightarrow \mathbb{R}$:

$$\begin{aligned}\nabla F(v)^T M \nabla F(v) > 0 \Leftrightarrow \\ \nabla F(v)^T (-M \nabla F(v)) < 0\end{aligned}$$

This shows that $-M \nabla F(v)$ is a descent direction of F .

When using Levenberg-Marquardt, we add μI with $\mu > 0$ to $J^T J$ such that $J^T J + \mu I \succ 0$. When a matrix is positive definite, it is invertible and the inverse matrix is also positive definite, i.e. $(J^T J + \mu I)^{-1} \succ 0$.

For the function $F = \|\mathbf{r}(v)\|^2$ we have

$$\nabla F(v) = 2J(v)^T \mathbf{r}$$

which shows that

$$\delta v = -(J(v)^T J(v) + \mu I)^{-1} J(v)^T \mathbf{r} = -\frac{1}{2} (J(v)^T J(v) + \mu I)^{-1} \nabla F(v)$$

Respectively, for $(J(v)^T J(v) + \mu I)^{-1} \succ 0$ we have:

$$\begin{aligned}\nabla F(v)^T (J(v)^T J(v) + \mu I)^{-1} \nabla F(v) > 0 \Leftrightarrow \\ \nabla F(v)^T \left(-\frac{1}{2} (J(v)^T J(v) + \mu I)^{-1} \nabla F(v) \right) < 0 \Leftrightarrow \\ \nabla F(v)^T \delta v < 0\end{aligned}$$

which proves that the update δv is a descent direction of $F(v)$.