Assignment 4 - Computer Vision (EEN020)

Luca Modica

December 14, 2023

1 Robust Epipolar Geometry and Two-View Reconstruction

Theoretical Exercise 1

To compute the essential matrix for the 2 calibrated cameras $P_1 = \begin{bmatrix} R_1 & t_1 \end{bmatrix}$ and $P_2 = \begin{bmatrix} R_2 & t_2 \end{bmatrix}$ we first need to convert the pair of cameras into a canonical pair. In other words, we need to find a specific value such that the pair of cameras will have the following values:

$$P_1 = \begin{pmatrix} I & 0 \end{pmatrix}, P_2 = \begin{pmatrix} R & t \end{pmatrix}.$$

In this case, the specific value can be $\begin{pmatrix} R_1^T & -R_1^T t_1 \\ 0^T & 1 \end{pmatrix}$. This since:

$$\begin{pmatrix} R_1 & t_1 \end{pmatrix} \begin{pmatrix} R_1^T & -R_1^T t_1 \\ 0^T & 1 \end{pmatrix} = \begin{pmatrix} I & 0 \end{pmatrix},$$
$$\begin{pmatrix} R_2 & t_2 \end{pmatrix} \begin{pmatrix} R_1^T & -R_1^T t_1 \\ 0^T & 1 \end{pmatrix} = \begin{pmatrix} R_2 R_1^T & t_2 - R_2 R_1^T t_1 \end{pmatrix},$$

where $R = R_2 R_1^T$ and $t = t_2 - R_2 R_1^T t_1$. Now that we have the pair of cameras in our canonical form, the essential matrix is computed as follows:

$$E = [t]_\times R = [t_2 - R_2 R_1^T t_1]_\times (R_2 R_1^T).$$

Theoretical Exercise 2

1. Suppose we have an essential matrix that related to 2 sets of points ($x_1^T E x_2 = 0$), where:
   - x_1 are image point of the camera $P_1 = \begin{pmatrix} I & 0 \end{pmatrix}$,
   - x_2 are image point of the camera $P_2 = \begin{pmatrix} R & t \end{pmatrix}$.

   Since, in this case, the essential matrix can be computed as $E = [t]_\times R$, the d.o.f. of E are 5. This is because:
   - R has 3 d.o.f.;
   - $[t]_\times$ has 3 d.o.f.;
   - the scale is arbitrary, so -1 d.o.f .

   So, in total, we can count $3 + 3 - 1 = 6$ d.o.f. .
2. The minimal number of point correspondences that you need to determine E is 5: this because the essential matrix can be estimated by first computing the fundamental matrix F using the 5-point algorithm and then converted to E with the calibration cameras of the 2 cameras involved. To determine E using an 8-point solver, instead, we need at least 8 point correspondences.
3. - s corresponds to the d.o.f of the point solver used. Since in this case the 8-point solver is used, $s = 8$.
   - The proportion of inliers, given the data in the question (proportion of incorrect correspondences = 25%), can be computed as follows: $\epsilon = 1 - 0.25 = 0.75$.
   - To finally how many iterations of RANSAC with an eight point solver do we need to find an outlier free sample set with 99% probability ($\alpha = 0.99$), we can follow this formula:

   $$T \geq \lceil \frac{log(1-\alpha)}{log(1-\epsilon^s)} \rceil = \lceil \frac{log(1-0.99)}{log(1-0.75^8)} \rceil = 44 \text{ iterations.}$$

Computer Exercise 1

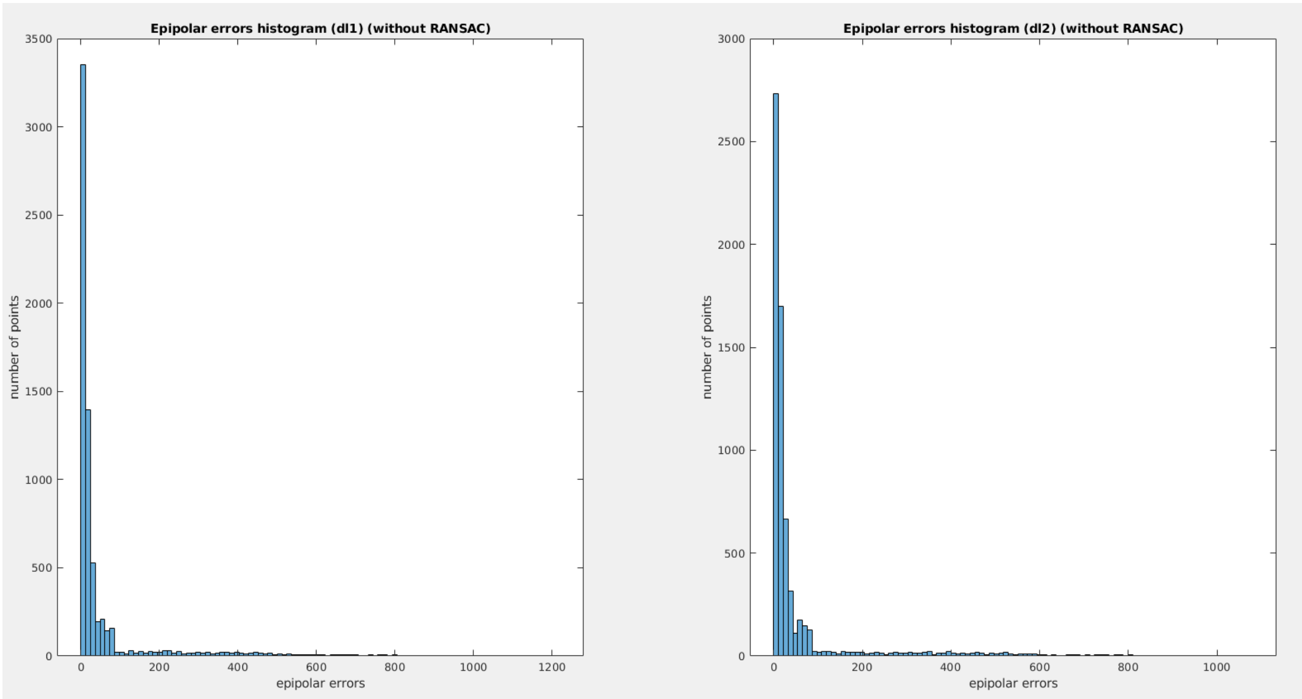

Figure 1: Histogram of the epipolar errors, before applying RANSAC. The RMS obtained of the distance between the images points and the corresponding epipolar lines is 155.9583.

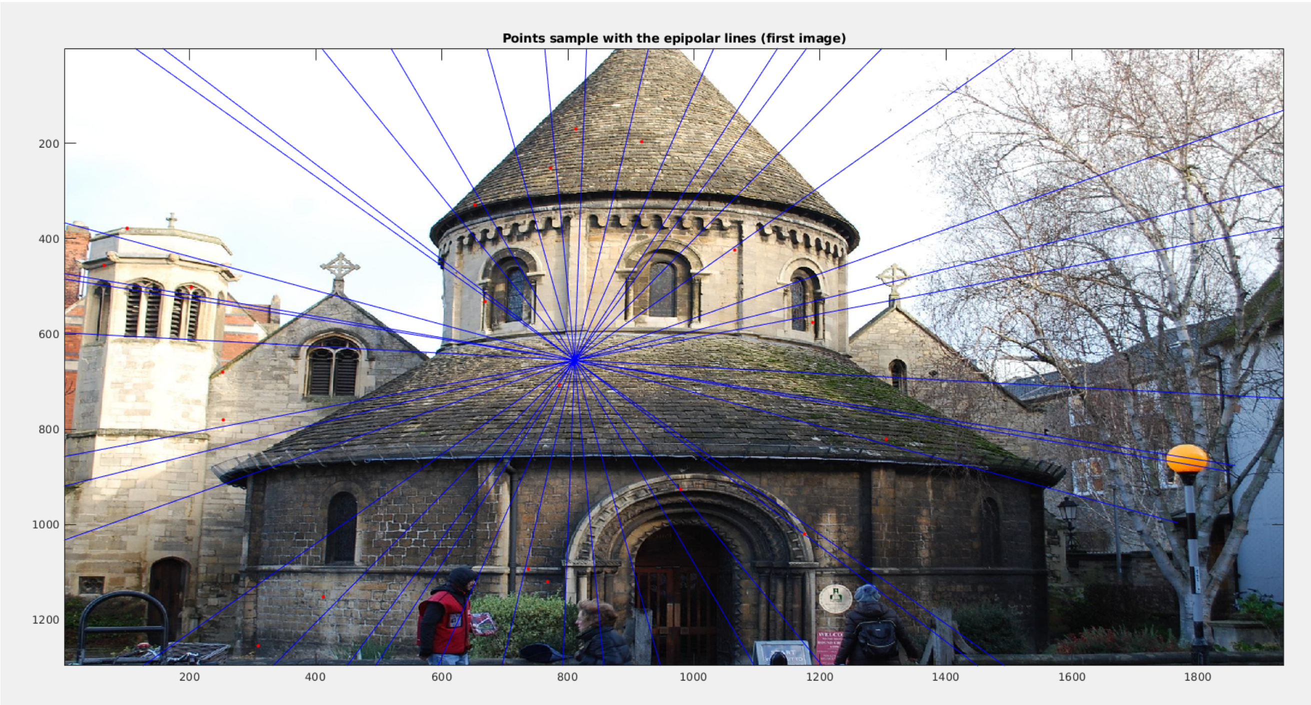

Figure 2: Epipolar lines and related image points in the first image.

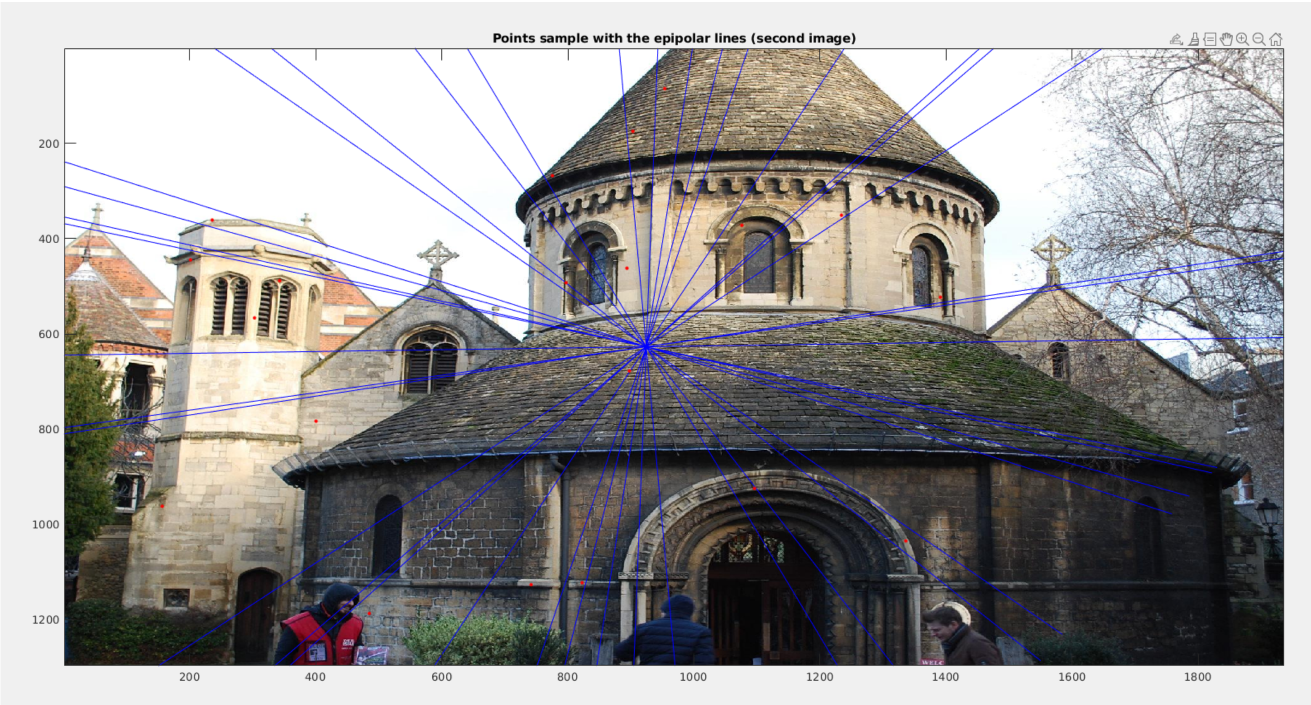

Figure 3: Epipolar lines and related image points in the second image.

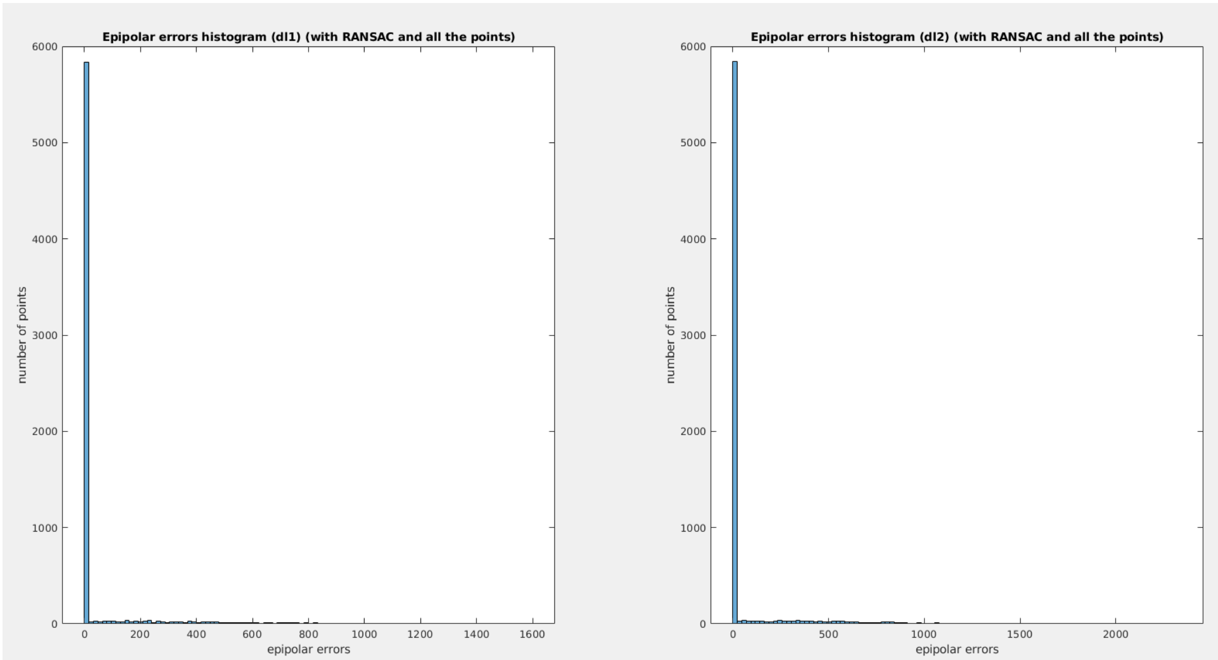

Figure 4: Histogram of the epipolar errors, after applying RANSAC. The number of inliers got in this case is 5816. The RMS obtained this time is higher, since the model took into consideration only the inliers; its value is 208.9571.

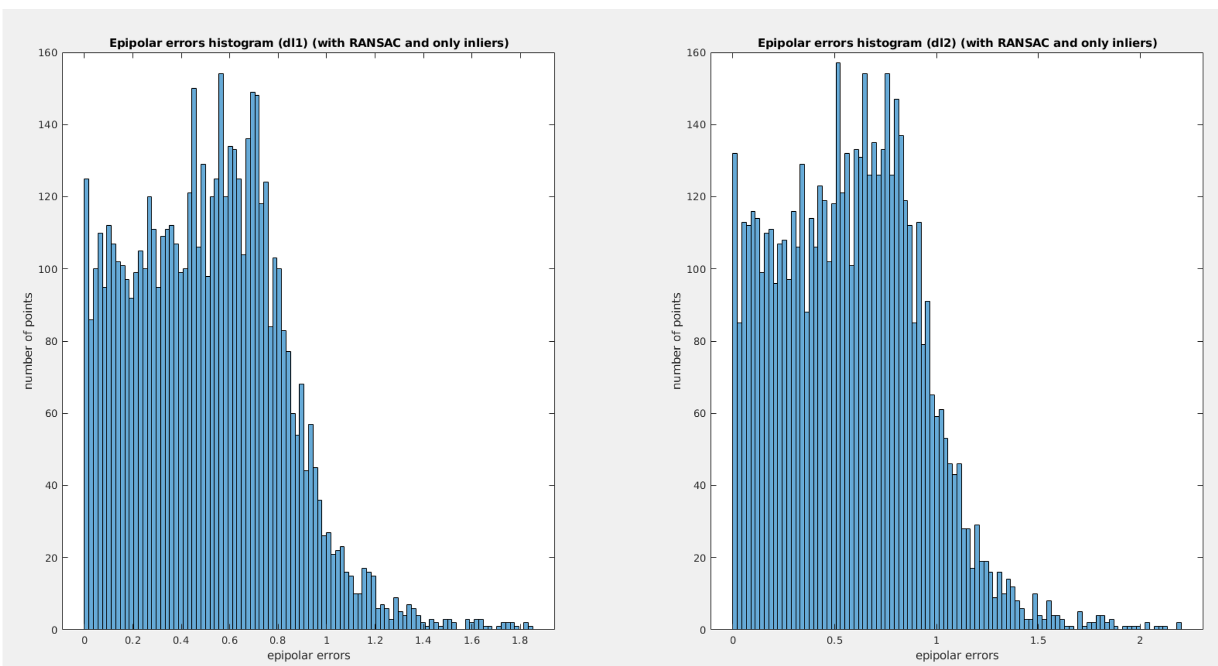

Figure 5: Histogram of the epipolar errors only considering the inliers, after applying RANSAC. The RMS obtained is 0.5831. As we can also notice from the much lower value of the RMS, the use of RANSAC brings to a better estiamte of the essential matrix.

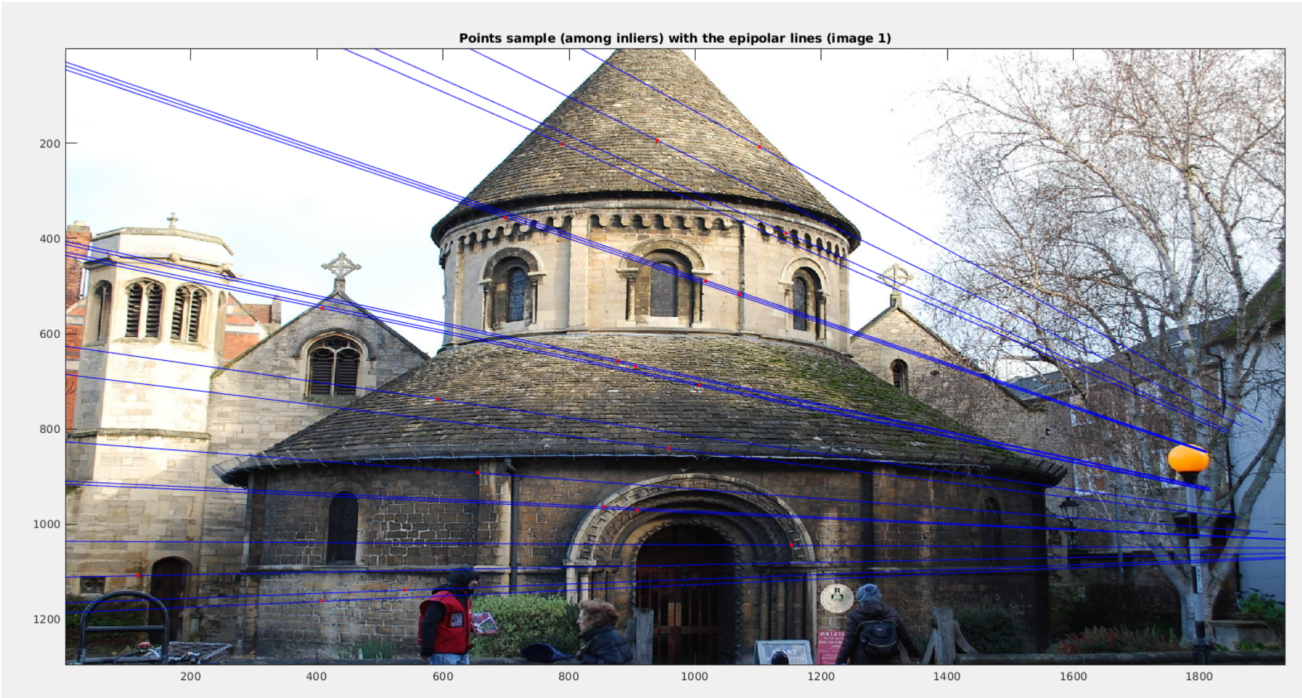

Figure 6: Inliers and related epipolar lines in the first image.

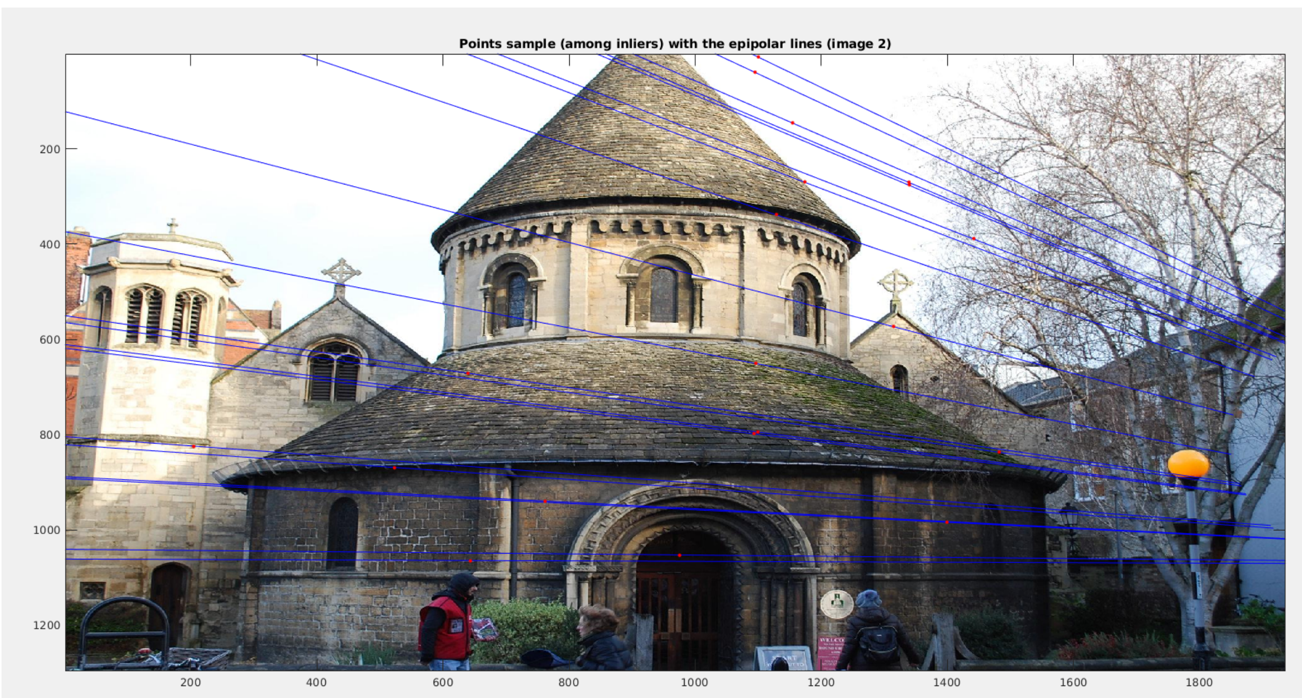

Figure 7: Inliers and related epipolar lines in the second image.

Computer Exercise 2

- Number of features found in image 1 and image 2, respectively: 39561 and 38775.
- Matches found between the 2 images: 2604.
- Number of inliers found after applying RANSAC: 1636.

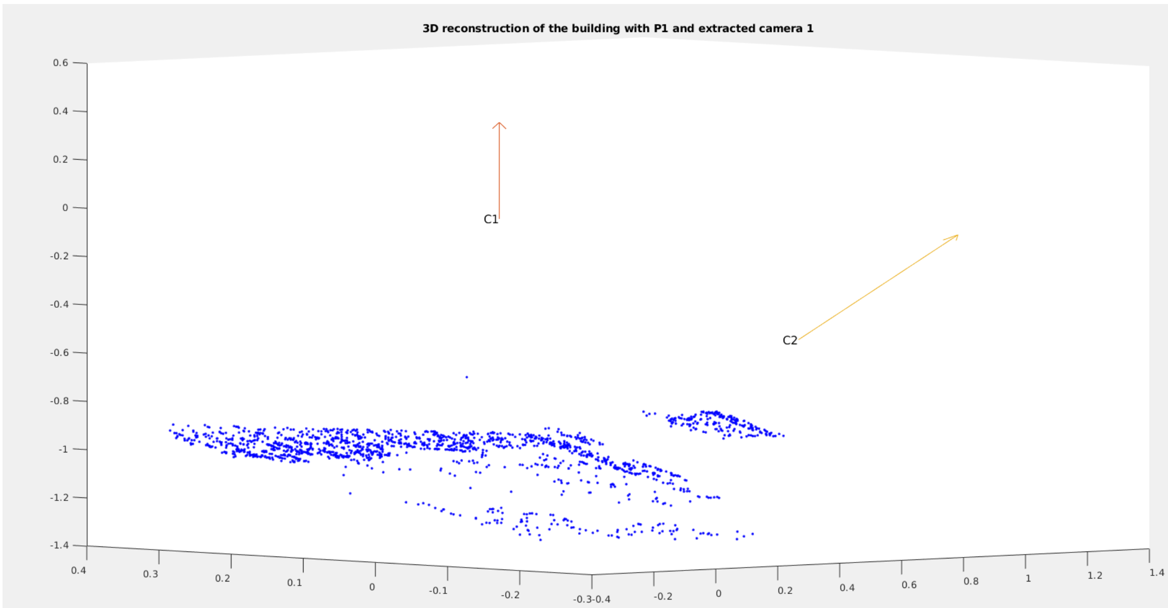

Figure 8: 3D reconstruction with the first possible camera.

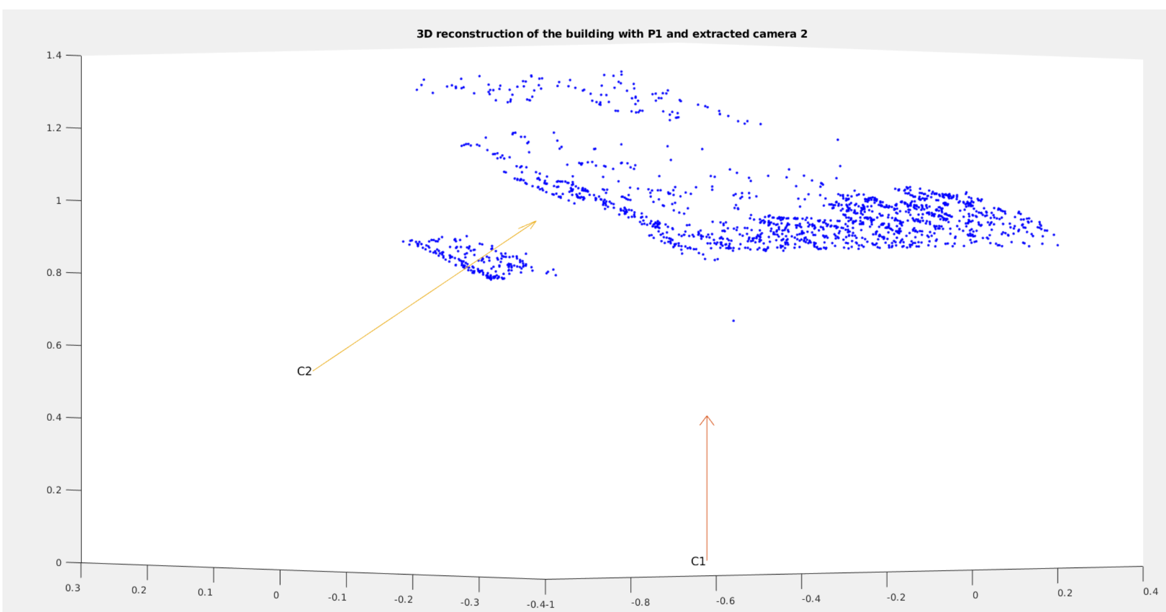

Figure 9: 3D reconstruction with the second possible camera.

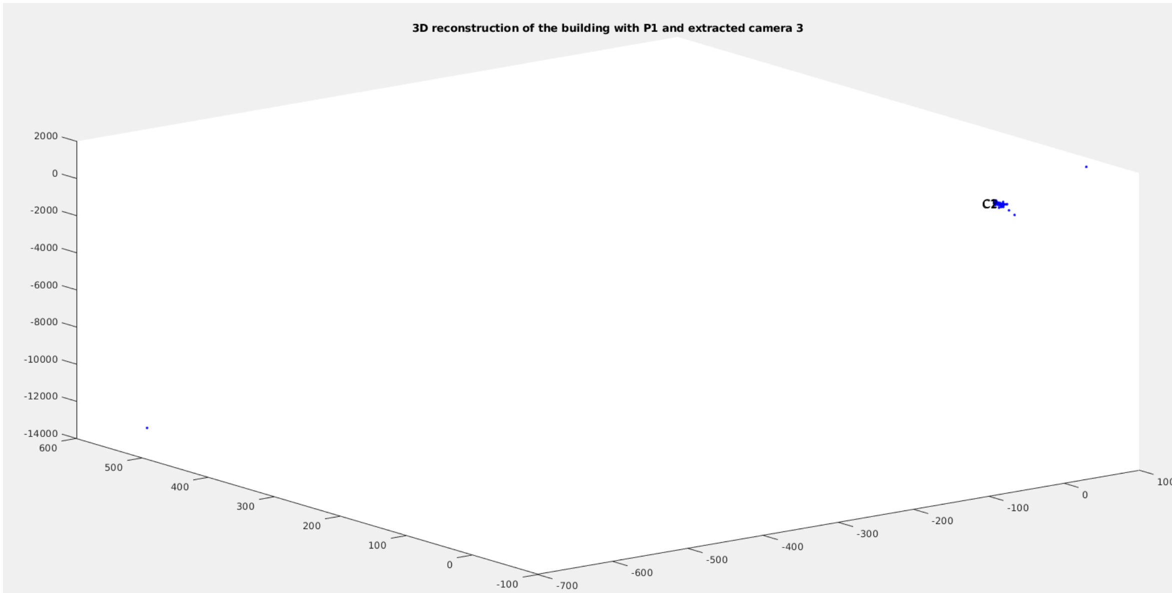

Figure 10: 3D reconstruction with the third possible camera.

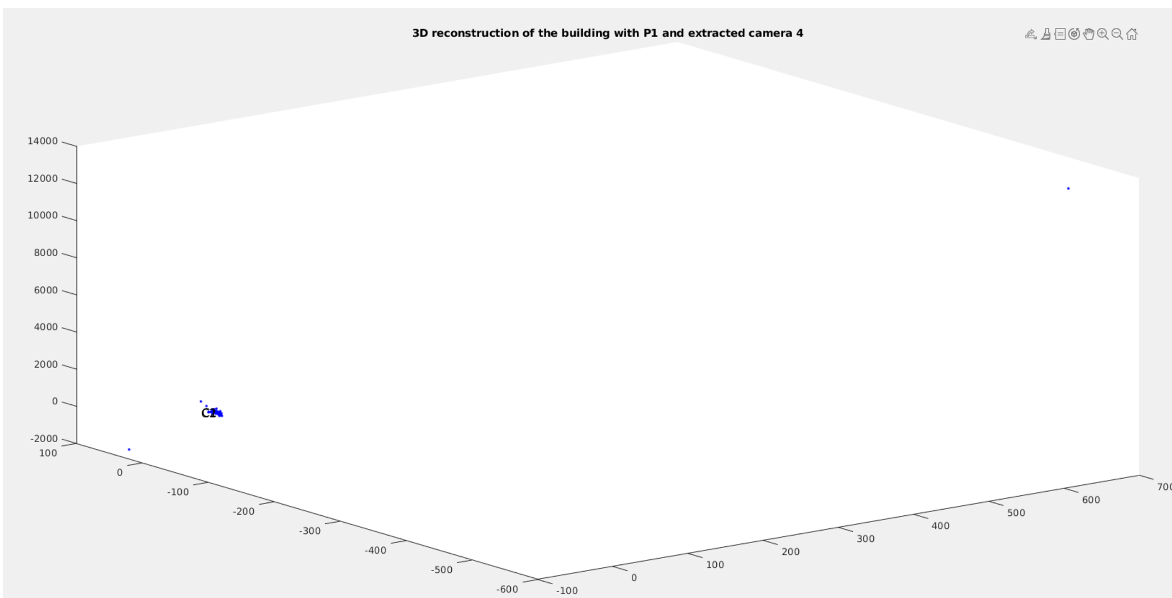

Figure 11: 3D reconstruction with the fourth possible camera.

Among the 4 possible cameras found, the second camera seems the most correct for the highest number of inliers in front P_1 and the candidate P_2.

2 Levenberg-Marquardt for Structure from Motion Problems

Theoretical Exercise 3

Part a)

WE HAVE:

$$r_i(X_j) = \left(r_{i,1}(X_j),\ r_{i,2}(X_j)\right) = \left(x_{ij,1} - \frac{P_i^1 X_j}{P_i^3 X_j},\ x_{ij,2} - \frac{P_i^2 X_j}{P_i^3 X_j}\right)$$

$$J_i(X_j) = \begin{bmatrix} \frac{(P_i^1 X_j)}{(P_i^3 X_j)^2} P_i^3 - \frac{1}{P_i^3 X_j} P_i^1 \\ \\ \frac{(P_i^2 X_j)}{(P_i^3 X_j)^2} P_i^3 - \frac{1}{P_i^3 X_j} P_i^2 \end{bmatrix}$$

TO SIMPLIFY THE CALCULATION, WE WILL USE THE THE INTERMEDIATE REPRESENTATION:

$$\boxed{Z_{ij}^k = P_i^k X_j}$$

WE CAN THEN SAY THAT:

$$J_i(X_j) = \left(\frac{\partial r_{i,1}}{\partial X_j},\ \frac{\partial r_{i,2}}{\partial X_j}\right)^T$$

$$= \left(\frac{\partial r_{i,1}}{\partial Z_{ij}^k} \frac{\partial Z_{ij}^k}{\partial X_j},\ \frac{\partial r_{i,2}}{\partial Z_{ij}^k} \frac{\partial Z_{ij}^k}{\partial X_j}\right)^T \quad \text{FOR } k = 1,2,3$$

USING THE RELATION $\frac{\partial (Ax)}{\partial x} = A$, WE CAN COMPUTE:

$$\frac{\partial Z_{ij}^k}{\partial X_j} = \frac{\partial (P_i^k X_j)}{\partial X_j} = \underbrace{\boxed{P_i^k}}_{1\times 4 \text{ MATRIX}} \quad \text{FOR } k = 1,2,3$$

$$\frac{\partial r_{i,1}}{\partial Z^1_{ij}} = \frac{\partial}{\partial Z^1_{ij}}\left(x_{ij,1} - \frac{Z^1_{ij}}{Z^3_{ij}}\right) = -\frac{1}{Z^3_{ij}} = -\frac{1}{P_i^3 X_j}$$

$$\frac{\partial r_{i,1}}{\partial Z^2_{ij}} = 0 \qquad \frac{\partial r_{i,1}}{\partial Z^3_{ij}} = \frac{\partial}{\partial Z^3_{ij}}\left(x_{ij,1} - \frac{Z^1_{ij}}{Z^3_{ij}}\right) = \frac{Z^1_{ij}}{(Z^3_{ij})^2} = \frac{P_i^1 X_j}{(P_i^3 X_j)^2}$$

KEEPING IN MIND THAT:

$$\frac{\partial r_{i,1}}{\partial Z_{ij}} \rightarrow 1 \times 4 \text{ VECTOR} \qquad \frac{\partial Z_{ij}}{\partial X_j} \rightarrow 3 \times 4 \text{ VECTOR}$$

WE CAN CAN KEEP COMPUTING:

$$\frac{\partial r_{i,1}}{\partial Z_{ij}} \frac{\partial Z_{ij}}{\partial X_j} = \overbrace{\left(\frac{\partial r_{i,1}}{\partial Z^1_{ij}} + \frac{\partial r_{i,1}}{\partial Z^2_{ij}} + \frac{\partial r_{i,1}}{\partial Z^3_{ij}}\right)}^{1 \times 4 \text{ VECTOR}} \overbrace{\frac{\partial Z_{ij}}{\partial X_j}}^{3 \times 4 \text{ VECTOR}}$$

$$= \left(-\frac{1}{P_i^3 X_j} P_i^1 + 0 \cdot P_i^2 + \frac{P_i^1 X_j}{(P_i^3 X_j)^2} P_i^3\right)$$

$$= \boxed{\frac{(P_i^1 X_j)}{(P_i^3 X_j)^2} P^3 - \frac{1}{P^3 X_j} P_i^1} \longrightarrow$$

1×4 VECTOR AND 1ST ROW OF $J_i(X_j)$

THE COMPUTATION ABOVE CAN BE APPLIED FOR THE SECOND ROW OF $J_i(X_j)$, CONSIDERING $r_{i,2}(X_j)$:

$$\frac{\partial Z^k_{ij}}{\partial X_j} = P_i^k \quad , \quad k = 1, 2, 3 \quad ,$$

$$\frac{\partial r_{i,2}}{\partial Z^1_{ij}} = 0 \quad , \quad \frac{\partial r_{i,2}}{\partial Z^2_{ij}} = -\frac{1}{P_i^3 X_j} \quad , \quad \frac{\partial r_{i,2}}{\partial Z^3_{ij}} = \frac{P_i^2 X_j}{(P_i^3 X_j)^2}$$

$$\frac{\partial r_{i,1}}{\partial Z_{ij}} \frac{\partial Z_{ij}}{\partial X_j} = \boxed{\frac{P_i^2 X_j}{(P_i^3 X_j)^2} P_i^3 - \frac{1}{P_i^3 X_j} P_i^2} \longrightarrow$$

1×4 VECTOR AND 2ND ROW OF $J_i(X_j)$

WITH THESE CALCULATION WE PROVED THAT

$J_i(X_j)$, WITH THE PROVIDED STRUCTURE, IS A 2×4 MATRIX.

Computer Exercise 3

- Total reprojection error before running LM: $2.2354e^4$.
- Total reprojection error after running LM: $2.1566e^4$.
- Median reprojection error before running LM: 11.6599.
- Median reprojection error after running LM: 11.1979.

What we can observe from the values related to the reprojection errors is that the Levenberg-Marquardt method perform a slight refinement of the 3D points, since the errors appear to be close to each other. Moreover this is confrmed by the following 3D plot, where both the 3D points before and after LM are visualized:

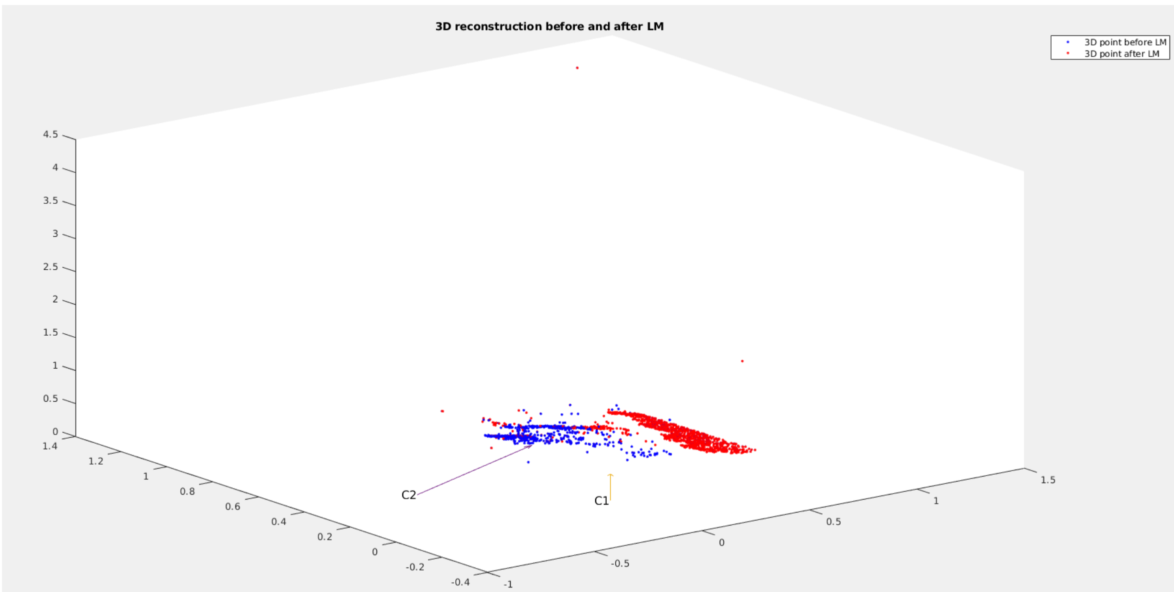

Figure 12: 3D points before and after LM. As we can notice, the 2 sets of points appear to be almost aligned to each other.

Computer Exercise 4

1. with standard deviation of 0 for both the 3D and 2D points, no noise will be added . Thus, the results are the same as the values found in the computer exercise 3.
2. $\sigma_X = 0.1$, $\sigma_x = 0$
   - Total reprojection error before running LM: $9.0065e^8$.
   - Total reprojection error before running LM: $2.1566e^4$.
   - Median reprojection error before running LM: $3.1804e^5$.
   - Median reprojection error before running LM: 11.1979.
3. $\sigma_X = 0$, $\sigma_x = 3$
   - Total reprojection error before running LM: $9.2547e^4$.
   - Total reprojection error before running LM: $3.8706e^4$.

- Median reprojection error before running LM: 39.9743.
- Median reprojection error before running LM: 11.40583.

4. $\sigma_X = 0.1$, $\sigma_x = 3$
    - Total reprojection error before running LM: $8.9995e^8$.
    - Total reprojection error before running LM: $3.8434e^4$.
    - Median reprojection error before running LM: $3.2100e^5$.
    - Median reprojection error before running LM: 11.3446.

As we notice from the numerical results, adding noise to the 3D points or / and to the 2D points let the reprojection error vary more from an initial situation to apply the LM method. In particular, we had the highest gap both in the total and the median reprojection error when the Gaussian noise is only applied to the 3D points, with a standard deviation of $\sigma_X = 0.1m$.

To better see the results explained above, in the below lineplots it will be shown the trend of the total and median reprojection error (note that the datapoints are shown in the same order as the bullet list above).

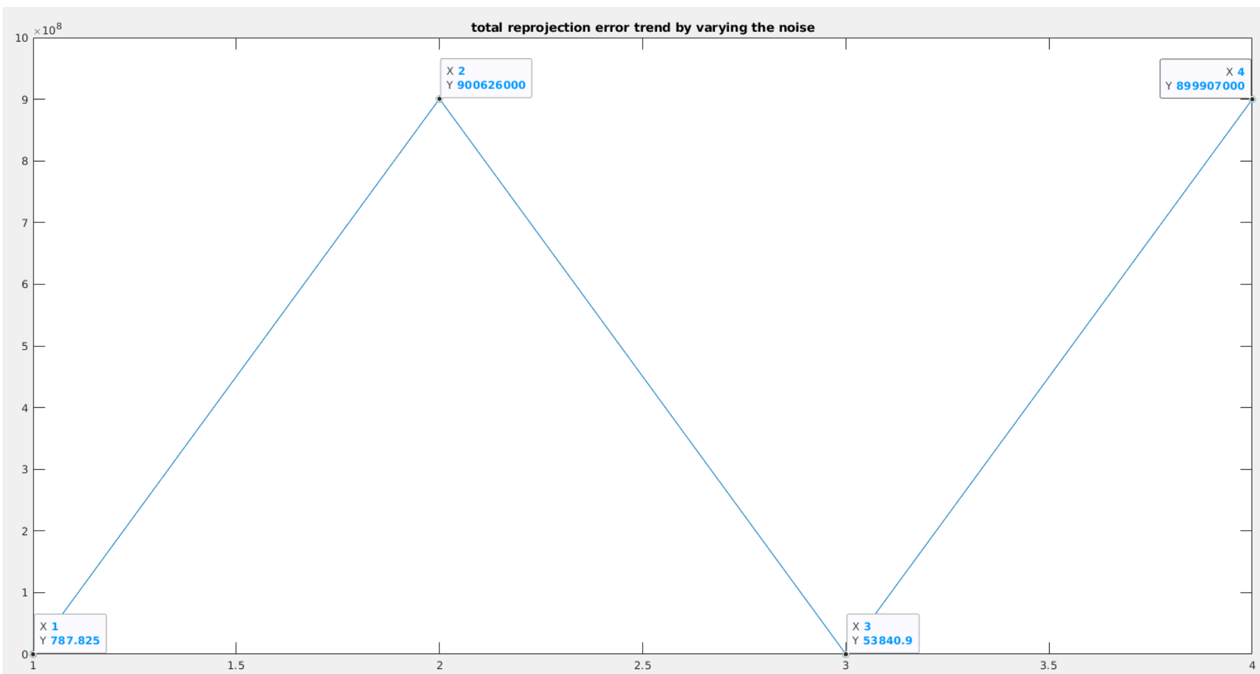

Figure 13: Total reprojection error trend, by varying the Gaussian noise.

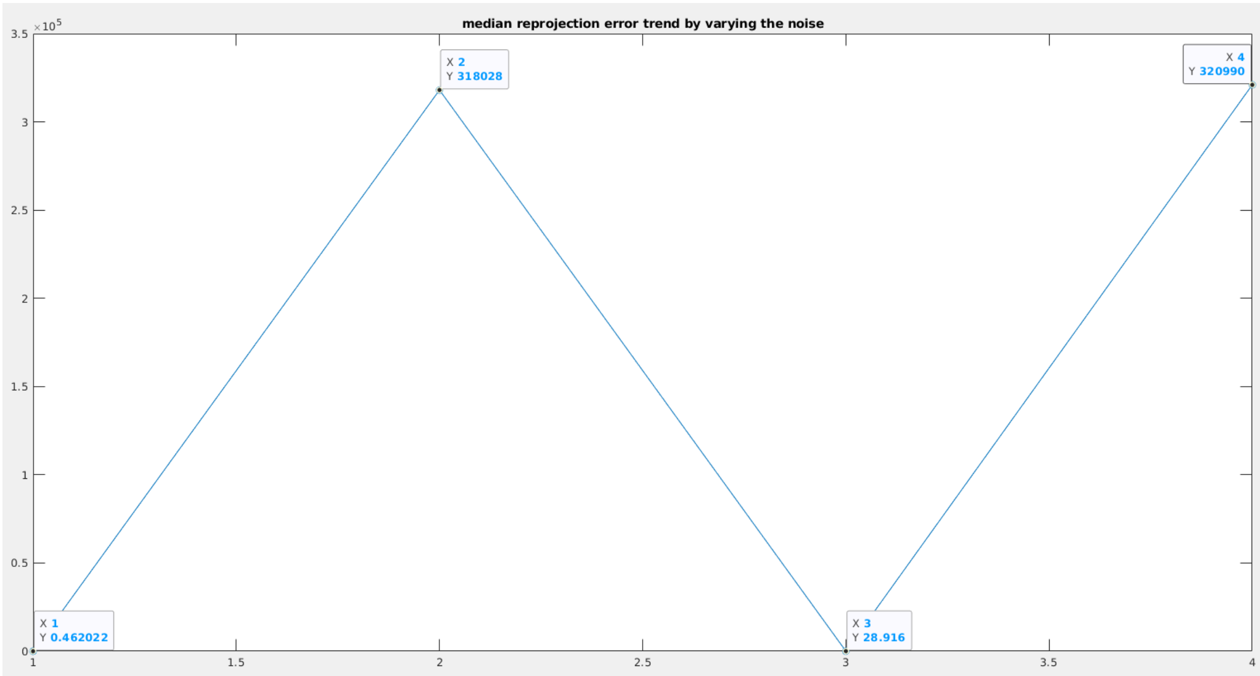

Figure 14: Median reprojection error trend, by varying the Gaussian noise.

Theoretical exercise 4

THEORETICAL 4

$$F(v) = \|r(v)\|^2$$

DIRECTION d IS A DESCENT DIRECTION OF A FUNCTION $F: \mathbb{R}^n \to \mathbb{R}$ AT THE POINT v IF:

$$\boxed{\nabla F(v)^T d < 0}$$

IN THIS CASE, WE HAVE A MATRIX $\boxed{M}$ THAT IS POSITIVE DEFINITE AND A DIRECTION d SUCH THAT:

$$d = -M \nabla F(v)$$

WE NEED TO PROVE THAT IF M IS POSITIVE DEFINITE, THEN d IS A DESCENT DIRECTION.

FOR THE DEFINITION OF POSITIVE DEFINITE FOR M, WE HAVE:

$$\boxed{w^T M w > 0}, \quad \text{FOR ANY } w \text{ SUCH THAT } \|w\| \neq 0$$

SETTING $w = \nabla F(v)$, WE CAN WRITE:

$$\nabla F(v)^T M \nabla F(v) > 0$$

$$\downarrow$$

$$\boxed{-\nabla F(v)^T M \nabla F(v) < 0}$$

WE KNOW THAT THE DIRECTION $d = -M \nabla F(v)$, SO:

$$\boxed{\nabla F(v)^T d < 0} \rightarrow d \text{ IS A DESCENT DIRECTION FOR } \boxed{F} \text{ AT } \boxed{v}$$

NOW, WE NEED TO PROVE THAT THE UPDATE CHOSEN IN LEVENBERG-MARQUARDT IN EQUATION (40) IS IN A DESCENT DIRECTION FOR THE $F(v)$ IN EQUATION (9).

$$\nabla F(v) = \nabla\left(\|r(v)\|^2\right) = \boxed{2J(v)^T r(v)}, \text{ SINCE } F(v) = r(v)^T r(v)$$

AND IT'S POSSIBLE TO USE THE CHAIN RULE.

FOR THIS PROOF WE WILL IGNORE "2" IN THE GRADIENT, SINCE IT DOES NOT AFFECT THE DIRECTION BUT JUST THE MAGNITUDE.

- $J(v)^T J(v)$ IS POSITIVE SEMIDEFINITE FOR BEING A GRAM-MATRIX. IN OTHER WORDS:

$$w^T\left(J(v)^T J(v)\right) w \geq 0 \quad , \quad \|w\| \neq 0$$

- BY ADDING THE POSITIVE AMOUNT μI (SINCE $\mu > 0$), THE MATRIX $\boxed{J(v)^T J(v) + \mu I}$ WILL BECOME POSITIVE DEFINITE. WE CAN ALSO SAY THAT ITS INVERSE IS POSITIVE DEFINITE.

$$\left(J(v)^T J(v) + \mu I\right)^{-1} \succ 0$$

SETTING $M = \left(J(v)^T J(v) + \mu I\right)^{-1}$ AND USING THE PROOF FROM THE PREVIOUS QUESTION, WE CAN CONFIRM THAT:

$$d = -\left(J(v)^T J(v) + \mu I\right)^{-1} J(v)^{-1} r(v) = -M \nabla F(v),$$

WHICH IS THE UPDATE CHOOSEN IN LEVENBERG-MARQUARDT, IS A DESCENT DIRECTION FOR $F(v) = \|r(v)\|^2$.