Assignment 2 EEN020 Computer Vision

Erik Norlin

14 December 2023

Calibrated vs. Uncalibrated Reconstruction

Theoretical Exercise 1

If we have the image projection x, a 3D reconstruction X with camera P, and another 3D reconstruction $\tilde{X} = TX$ with camera $\tilde{P}$ where T is any projective transformation, we have that

$$\lambda x = PX = PT^{-1}TX = \tilde{P}\tilde{X}$$

thus, we can get the same image projections for different 3D reconstructions which implies that there is an ambiguity in the 3D reconstruction for uncalibrated cameras.

Computer Exercise 1

Plotting the 3D points and the cameras we can see in Figure 1 that the physical properties do not look completely realistic. The corner looks like an obstuse angle and not 90 degrees. The walls look also very tall and the narrow. One can identify contours of the window and the depth of it, but overall the 3D reconstruction does not look too great.

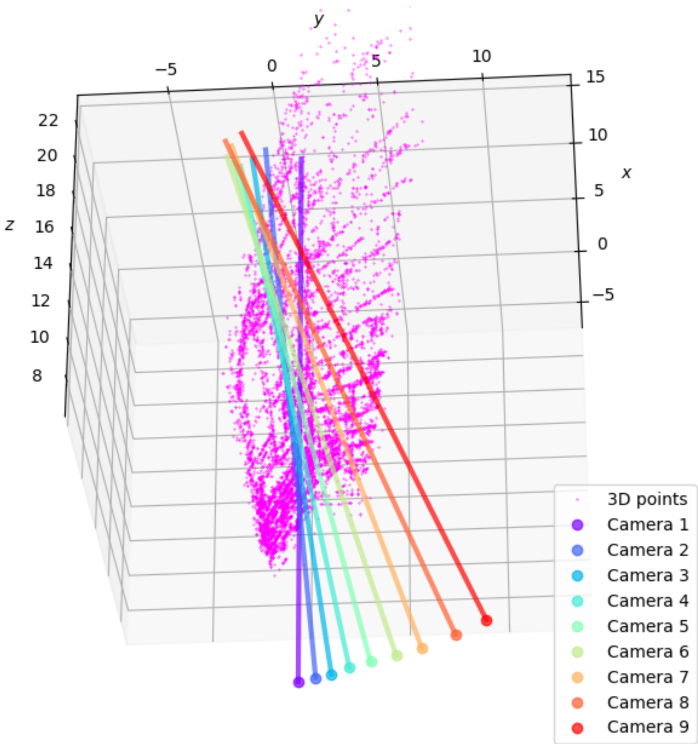

Figure 1: 3D plot of cameras and 3D points from `compEx1data.mat`.

Figure 2 shows a plot of the image, the projected points, and the image points of the first view where we can see that the projection points fall close to the corresponding image points.

Figure 2: Plot of the first view with its corresponding image, the projected points, and the image points.

Transforming the 3D points with T_1 we can see in Figure 3 that this 3D reconstruction does not look reasonable. The two walls are almost completely flat with each other where there is supposed to be a corner, and the tall narrow walls are now short and wide. This does not seem like an accurate representation of the 3D structure.

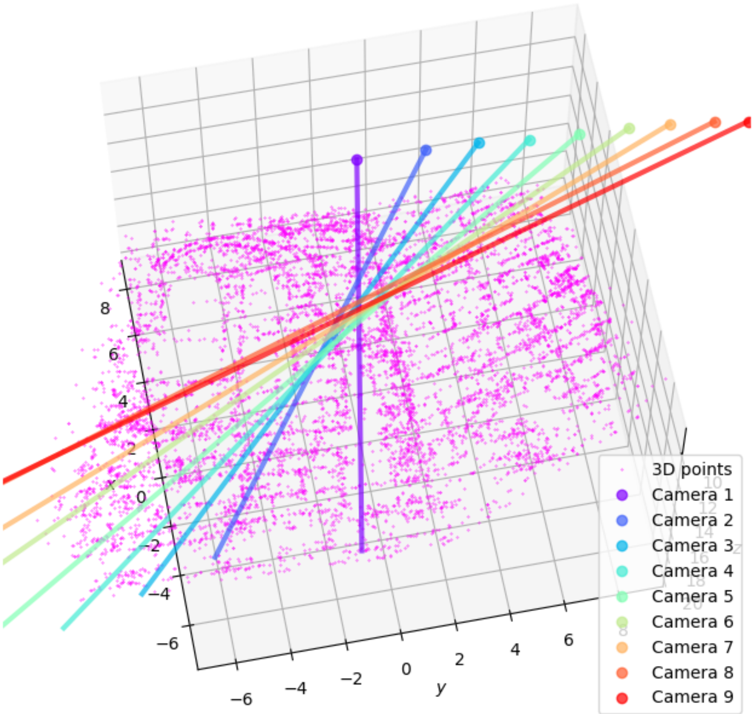

Figure 3: Plot of the 3D reconstruction T_1X.

Now transforming the 3D points with T_2 we can see in Figure 4 that this 3D reconstruction looks reasonable. The two walls are perpendicular as expected, the height and width of the walls seem more proportional. This 3D reconstruction is the most realistic one of so far.

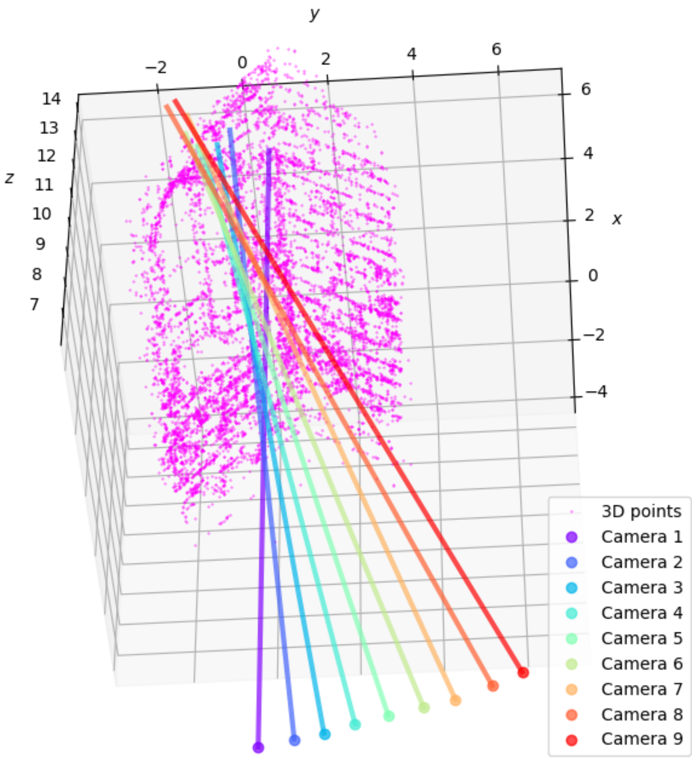

Figure 4: Plot of the 3D reconstruction T_2X.

Projecting these new 3D reconstructions into the same view we can see in Figure 5 that both projections align very closely with the image points, supporting the fact that there is an ambiguity in uncalibrated reconstruction because both 3D reconstructions gave if not identical image projection.

Figure 5: Projected 3D points T_1X to the left, projected 3D points T_2X to the right along with original image points.

Theoretical Exercise 2

If we know the calibration matrix K for camera P and normalise x, i.e., $\tilde{x} = K^{-1}x$, we get that

$$\lambda\tilde{x} = [R|t]X = [R|t]T^{-1}TX$$

Where the first 3×3 is a rotation which implies that T^{-1} must be defined such that $[R|t]T^{-1}$ also is a rotation. Before normalisation the rotation was multiplied with K which does not necessarily make the first 3×3 a rotation. Therefore, we cannot get the same projective ambiguity after having normalised. However, there is still some ambiguity left as can be seen in the equation that we still have $T^{-1}T$. The reconstruction can still be rescaled, rotated etc. without changing the image projections.

Camera Calibration

Theoretical Exercise 3

$$K = \begin{pmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{pmatrix}$$

Taking the inverse of K we get

$$K = \left(\begin{array}{ccc|ccc} f & 0 & x_0 & 1 & 0 & 0 \\ 0 & f & y_0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right) = \left(\begin{array}{ccc|ccc} f & 0 & x_0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1/f & -y_0/f \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right) =$$

$$\left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1/f & 0 & -x_0/f \\ 0 & 1 & 0 & 0 & 1/f & -y_0/f \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right) \Rightarrow K^{-1} = \begin{pmatrix} 1/f & 0 & -x_0/f \\ 0 & 1/f & -y_0/f \\ 0 & 0 & 1 \end{pmatrix}$$

and K^{-1} can indeed be factorized into

$$K^{-1} = \begin{pmatrix} 1/f & 0 & 0 \\ 0 & 1/f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1 \end{pmatrix}$$

because the expression yields K^{-1}. Let A be the left matrix and B be the right matrix. The geometric interpretation of A is that it scales an image point in sensor coordinates to image coordinates, and B translates the origin of the sensor coordinate system to the the principal point in the image coordinate system. Together they transform an image point in sensor coordinates to image coordinates.

When transforming an image with K^{-1} the image in sensor coordinates is transformed to image coordinates, i.e., the image is normalised. In this transformation, the sensor coordinates

$$\begin{cases} x_s = fx_i = fx_c/z_c \\ y_s = fy_i = fy_c/z_c \\ z_s = fz_i = fz_c/z_c \end{cases}$$

is scaled by $1/f$ which removes the f from the image coordinate, hence transforming the sensor coordinates to image coordinates. The coordinates are also translated with the principal point $(x_0,\, y_0)^T$. Since the sensor coordinate system is in the upper left corner (inverse of x_s and y_s) the origin of the coordinate system is translated to the principal point, i.e., the center of the image. A point with a distance of f to the principal point, say $(x_s,\, y_s)^T = (fx_0+f,\, fy_0)^T$, is transformed to image coordinates then the point $(x_i,\, y_i)^T = (x_0 + 1,\, y_0)^T$ is obtained. It can be concluded from this that any point with a distance of f to the principal point will be one unit length away from it in image coordinates. Thus, for any given focal length f, transforming an image with its camera's corresponding K^{-1} *normalises* the image.

We want to normalise the points

$$x_1 = \begin{pmatrix} 0 \\ 300 \end{pmatrix}, \text{ and } x_2 = \begin{pmatrix} 800 \\ 300 \end{pmatrix}$$

with

$$K = \begin{pmatrix} 400 & 0 & 400 \\ 0 & 400 & 300 \\ 0 & 0 & 1 \end{pmatrix}$$

We get that

$$K^{-1}x_1 = \begin{pmatrix} 400 & 0 & 400 \\ 0 & 400 & 300 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 300 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} = \tilde{x}_1$$

$$K^{-1}x_2 = \begin{pmatrix} 400 & 0 & 400 \\ 0 & 400 & 300 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 800 \\ 300 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \tilde{x}_2$$

The angle of ray between these points is

$$\theta = \arccos(\frac{\tilde{x}_1^T \tilde{x}_2}{||\tilde{x}_1||||\tilde{x}_2||}) = \arccos(0) = \pi/2$$

Thus, 90 degrees.

Now we show that the camera $K[R|t]$ and the corresponding normalised version $[R|t]$ have the same camera center and principal axis. For any camera P we that the camera center is $C = -M^{-1}P_4$ where $M = P_{:,1:3}$ and P_4 the fourth column of P. For $P = K[R|t]$ we have that

$$C = -(KR)^{-1}Kt = -R^{-1}K^{-1}Kt = -R^{-1}t$$

and for $P = [R|t]$ we have the camera center

$$C = -R^{-1}t$$

which shows that we get the same camera center for both cases. The principal axis of a camera is $P_{3,1:3}$. For $P = K[R|t]$ we have that the principal axis a is $a = R^T k_3$ where $k_3 = (0{,}0{,}1)^T$ is the third row of K. We get that $a = R^T k_3 = r_3$ where r_3 is the third row of R. For $P = [R|t]$ we have that the principal axis a is r_3, which shows that the principal axis in both cases are the same.

RQ Factorization and Computation of K

Theoretical Exercise 4

We have that

$$K = \begin{pmatrix} a & b & c \\ 0 & d & e \\ 0 & 0 & f \end{pmatrix}, KR = \begin{pmatrix} aR_1^T & bR_2^T & cR_3^T \\ 0 & dR_2^T & eR_3^T \\ 0 & 0 & fR_3^T \end{pmatrix}, R = \begin{pmatrix} R_1^T \\ R_2^T \\ R_3^T \end{pmatrix}$$

and

$$P = \begin{pmatrix} 2400 & 0 & 800\sqrt{2} & 4000 \\ 700\sqrt{2} & 2800 & -700\sqrt{2} & 4900 \\ 1/\sqrt{2} & 0 & -1/\sqrt{2} & 3 \end{pmatrix}$$

We solve K by solving row for row starting from the bottom up of K. We begin by solving f and R_3. We use the fact that $||R_k|| = 1$. We have that $||P_{3,1:3}|| = 1$ so $||fR_k|| = 1 \Rightarrow f = 1$

$$P_{3,1:3} = (KR)_{3,1:3} \Rightarrow \begin{pmatrix} 1/\sqrt{2} \\ 0 \\ -1/\sqrt{2} \end{pmatrix} = fR_3 \Rightarrow R_3 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

For the second row of K we solve for R_2, d and e. We utilize the fact that the dot product of two perpendicular vectors is zero.

$$R_2 \perp R_3 \Rightarrow R_3^T R_2 = 0 \Rightarrow \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}^T R_2 = 0 \Rightarrow R_{21} = 0,\ R_{23} = 0$$

$$P_{2,1:3} = \begin{pmatrix} 700 \\ 2800 \\ -700\sqrt{2} \end{pmatrix} = dR_2 + eR_3$$

We get the following equation system

$$\Rightarrow \begin{cases} dR_{21} + e/\sqrt{2} = 700\sqrt{2} \\ dR_{22} = 2800 \\ dR_{23} - e/\sqrt{2} = -700\sqrt{2} \end{cases}$$

Now using the fact that $||R_2|| = 1$

$$\Rightarrow e = 1400$$

$$||R_2|| = \begin{pmatrix} 0 \\ R_{22} \\ 0 \end{pmatrix}^T \begin{pmatrix} 0 \\ R_{22} \\ 0 \end{pmatrix} = 1 \Rightarrow R_{22} = 1 \Rightarrow R_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \Rightarrow d = 2800$$

For the first row of K we get the following equation system

$$P_{1,1:3} = \begin{pmatrix} 2400 \\ 0 \\ 800\sqrt{2} \end{pmatrix} = aR_1 + bR_2 + cR_3$$

$$\Rightarrow \begin{cases} aR_{11} + c/\sqrt{2} = 2400\sqrt{2} \\ aR_{12} + b = 0 \\ aR_{13} - c/\sqrt{2} = 800\sqrt{2} \end{cases}$$

$$R_1 \perp R_2 \Rightarrow R_1^T R_2 = R_{12} = 0$$

$$R_1 \perp R_3 \Rightarrow R_1^T R_3 = R_{11} - R_{13} = 0 \Rightarrow R_{11} = R_{13}$$

$$||R_1|| = \begin{pmatrix} R_{11} \\ 0 \\ R_{11} \end{pmatrix}^T \begin{pmatrix} R_{11} \\ 0 \\ R_{11} \end{pmatrix} = \sqrt{2} R_{11} = 1 \Rightarrow R_{11} = R_{13} = 1/\sqrt{2} \Rightarrow R_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

We can then simplify the equation system to

$$\Rightarrow \begin{cases} a/\sqrt{2} + c/\sqrt{2} = 2400\sqrt{2} \\ b = 0 \\ a/\sqrt{2} - c/\sqrt{2} = 800\sqrt{2} \end{cases}$$

$$\Rightarrow \begin{cases} a = 1600 + c \\ 1600 + 2c = 4800 \end{cases}$$

$$\Rightarrow a = 3200,\ c = 1600$$

Thus, we get

$$K = \begin{pmatrix} \gamma f_l & s f_l & x_0 \\ 0 & f_l & y_0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ 0 & d & e \\ 0 & 0 & f \end{pmatrix} = \begin{pmatrix} 3200 & 0 & 1600 \\ 0 & 2800 & 1400 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R = \begin{pmatrix} R_1^T \\ R_2^T \\ R_3^T \end{pmatrix} = \begin{pmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 0 & 1 & 0 \\ 1/\sqrt{2} & 0 & -1/\sqrt{2} \end{pmatrix}$$

and

Focal length $f_l = 2800$
Skew $s = 0$
Aspect ratio $\gamma = 8/7$
Principal point $(x_0,y_0)^T = (1600,1400)^T$

Direct Linear Transformation DLT

Theoretical Exercise 5

$$\min_v ||Mv||^2 = \min_v v^T M^T M v$$

Unconstrained convex optimization problem so we set gradient w.r.t. v to 0 to find min.

$$M^T M v = 0$$

Solution is $v = 0$.

We have that $M = U\Sigma V^T$ and that U and V are orthonormal which gives us that $U^T U = I$ and $V^T V = V V^T = I$. This gives us

$$||Mv||^2 = (U\Sigma V^T v)^T U\Sigma V^T v = v^T V \Sigma^T U^T U \Sigma V^T v = v^T V \Sigma^T \Sigma V^T v = ||\Sigma V^T v||^2$$

and

$$||V^T v|| = \sqrt{v^T V V^T v} = \sqrt{v^T v}$$

Since $||v||^2 = v^T v = 1$, $\sqrt{v^T v}$ must also be $1 \Rightarrow ||V^T v|| = 1$ if $||v||^2 = 1$.

Now let $\tilde{v} = V^T v$ then we have that

$$\min_{||v||^2=1} ||Mv||^2 = \min_{||V^T v||^2=1} ||\Sigma V^T v||^2 = \min_{||\tilde{v}||^2=1} ||\Sigma \tilde{v}||^2 =$$

Thus, we get the same optimization problem and the same solution if we let v be transformed. We always get two solutions because we have that $v^T v = 1$ then we also have that $(-v)^T(-v) = 1$.

If we lagrangian relax the constraint we get

$$\min_{||v||^2} v^T MTMv - \lambda(v^T v - 1)$$

We get the solution by setting the gradient w.r.t. to v to 0 and solve for v. So we get

$$M^T Mv = \lambda v$$

with $||v|| = 1$. We can see that the solutions are eigenvectors of $M^T M$ where λ are the corresponding eigenvalues. To get the minimised value of this we want to take the eigenvector corresponding to the smallest eigenvalue of $M^T M$. From before we have that

$$M^T M = V\Sigma^2 V^T$$

where Σ^2 is a diagonal matrix containing the eigenvalues of $M^T M$ and V containing the corresponding eigenvectors of $M^T M$. Since the values of Σ^2 decreases along the diagonal, the last element in Σ^2 is the smallest eigenvalue, and the corresponding eigenvector to this is the last column in V. When M does not have full rank, i.e., rank$(M) < n$ then the smallest eigenvalue in Σ^2 will be zero, choosing the corresponding eigenvector yields an exact zero-solution and will therefore be a solution to $\min_{||v||^2=1} ||Mv||^2$, i.e., $Mv = 0$. For the case of rank$(M) \geq n$ the last element of Σ^2 will not be zero, hence there are no eigenvectors that yield the solution $Mv = 0$.

Theoretical Exercise 6

We have that $\tilde{x} \sim Nx \sim \tilde{P}X$ and $x \sim PX$. So we get

$$N^{-1}\tilde{x} \sim N^{-1}Nx \sim N^{-1}\tilde{P}x = PX$$

So we compute the camera to the original problem as following

$$\Rightarrow P = N^{-1}\tilde{P}$$

Computer Exercise 2

View 1

We can see in Figure 6 that the normalised points center around (0,0).

Table 1: Mean and standard deviation of view 1.

	mean	std.
x	1014.85	193.94
y	893.04	195.82

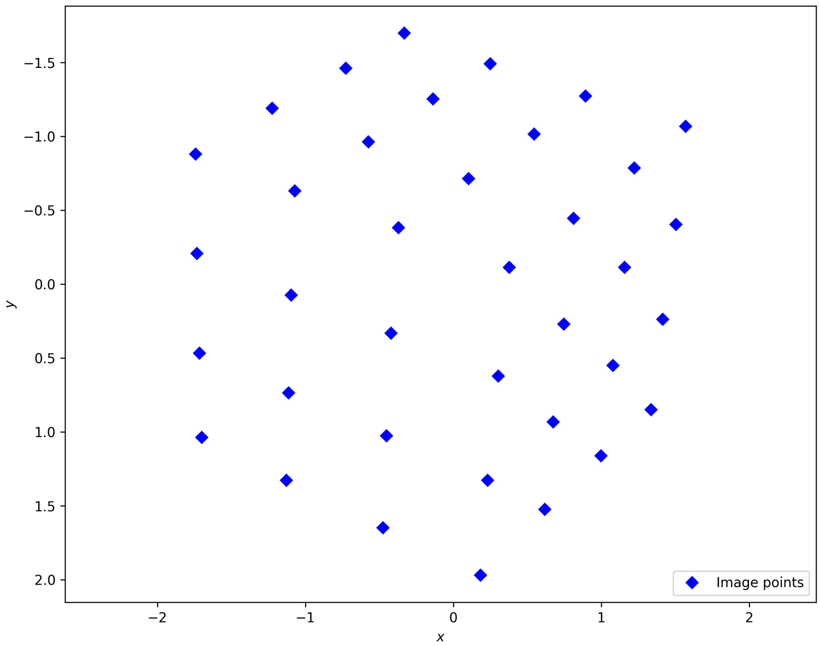

Figure 6: Normalised image points of view 1.

When verifying the mean and std. we get mean approximately 0 and std. 1.

When estimating the cameras with DLT, the smallest singular value we get when computing the SVD of M is approximately 0.051. $||Mv||$ is approximately 0.051, equal to the smallest singular value.

Figure 7 shows that the projected 3D points on to the image aligns with the original image points. They do not align exactly but it is still a satisfactory result.

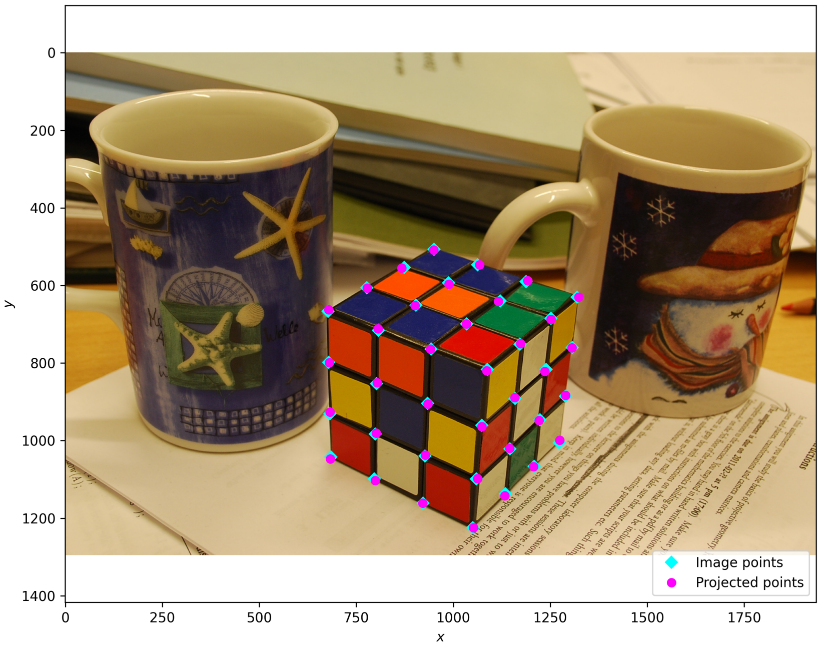

Figure 7: Image points and projected points from the estimated camera in view 1.

View 2

Table 2: Mean and standard deviation of view 2.

	mean	std.
x	930.97	195.71
y	795.19	196.72

We can see in Figure 8 that the normalised points center around (0,0).

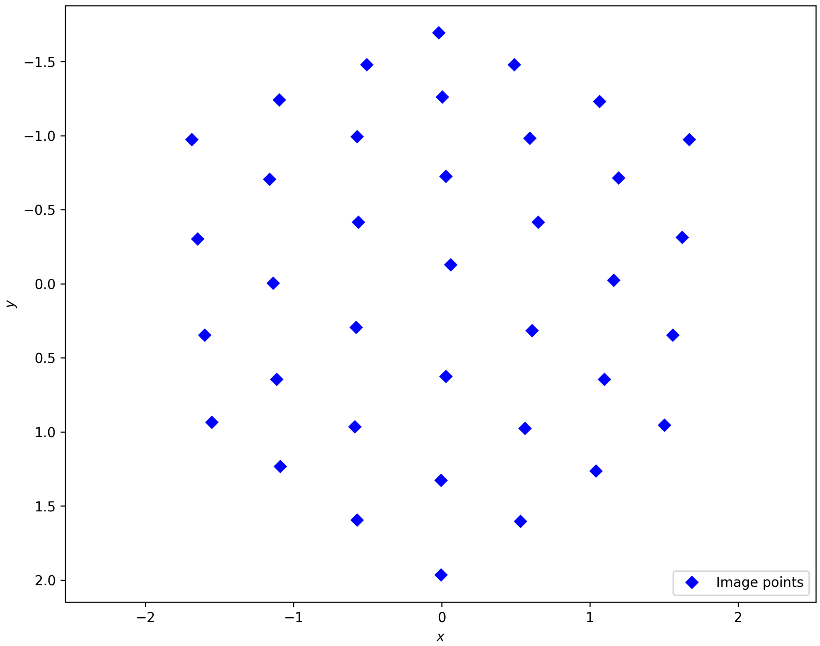

Figure 8: Normalised image points of view 2.

When verifying the mean and std. we get mean approximately 0 and std. 1.

When estimating the cameras with DLT, the smallest singular value we get when computing the SVD of M is approximately 0.044. $||Mv||$ is approximately 0.044, equal to the smallest singular value.

Figure 9 shows that the projected 3D points on to the image aligns with the original image points. They do not align exactly but it is still a reasonable result.

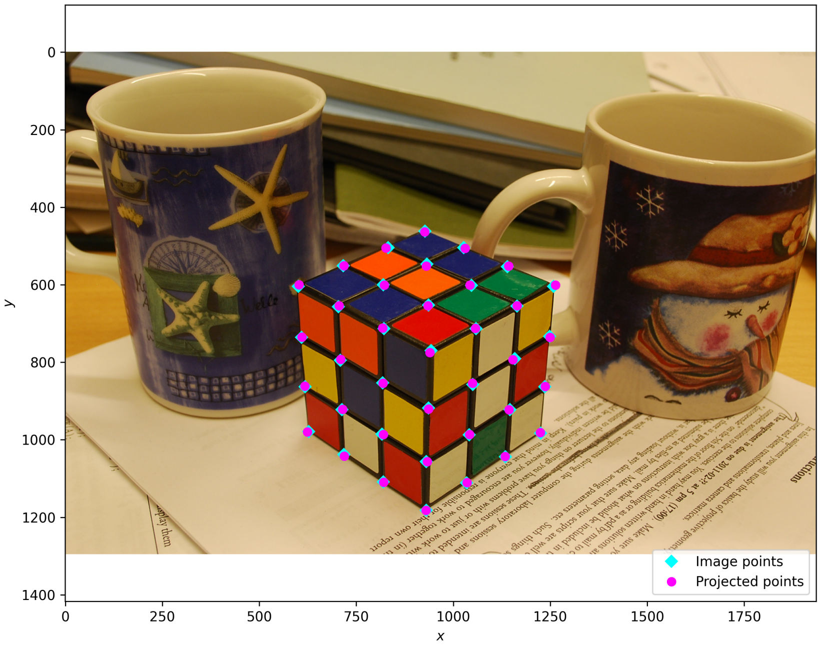

Figure 9: Image points and projected points from the estimated camera in view 2.

3D Reconstruction

Figure 10 shows a 3D view from above and in front of the cube. The cameras look reasonable because camera 1 looks at the cube from above and from the left, and camera 2 looks at the cube from above and from the right as expected.

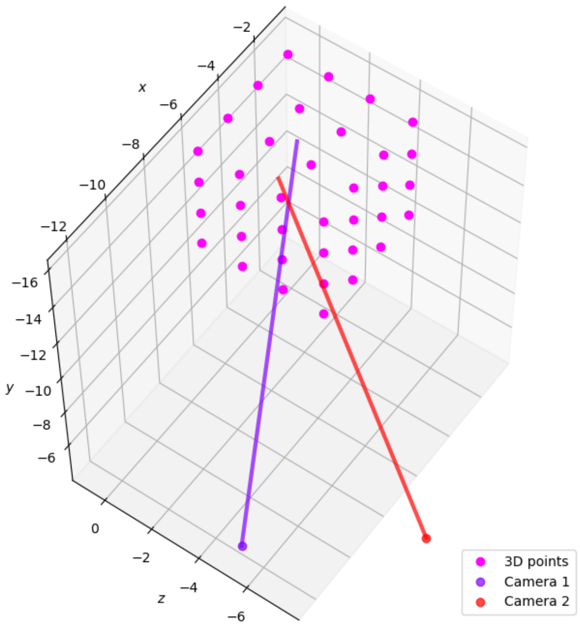

Figure 10: 3D points and cameras from estimated camera matrices. View from above and in front of the cube.

Inner Parameters

The estimated inner parameters of the first camera are

$$K \approx \begin{pmatrix} 2422 & -7 & 980 \\ 0 & 2420 & 694 \\ 0 & 0 & 1 \end{pmatrix}$$

The parameters are not completely true because when estimating the parameters we get numerical approximations. However, we can get close to the true K by RQ-decomposition; any square matrix $A \in \mathbb{R}^{n\times n}$ can always be decomposed to RQ where $R \in \mathbb{R}^{n\times n}$ is an upper diagonal matrix and $Q \in \mathbb{R}^{n\times n}$ is an orthonormal matrix. Since we have that $P = [KR|Kt] \in \mathbb{R}^{3\times 4}$ where $K \in \mathbb{R}^{3\times 3}$ is an upper diagonal matrix and $R \in \mathbb{R}^{3\times 3}$ is an orthonormal matrix, we can construct a submatrix of P as $P' = KR \in \mathbb{R}^{3\times 3}$ and retrieve K by RQ-decomposing P'. Thus, we get close to the true inner parameters, and this is what we have done in the exercise.

There is no ambiguity here as in exercise 1 because in this exercise we normalize the cameras with N^{-1} (instead of K^{-1} which serves similar purpose). With this normalization we get rid of some ambiguity as demonstrated earlier.

Computer Exercise 3

Nothing here as by the instructions.

Computer Exercise 4

Figure 11 shows SIFT points and projected points when the 3D reconstruction was computed from unnormalised points and cameras, whereas Figure 12 shows the SIFT points and projected points when the 3D reconstruction was computed from normalised points and cameras. Both cases show similar result. It is worth noting however that points with larger pixel error than 3 were removed in both cases, which could explain why there is not a noticeably better result for the normalised case which could otherwise be expected.

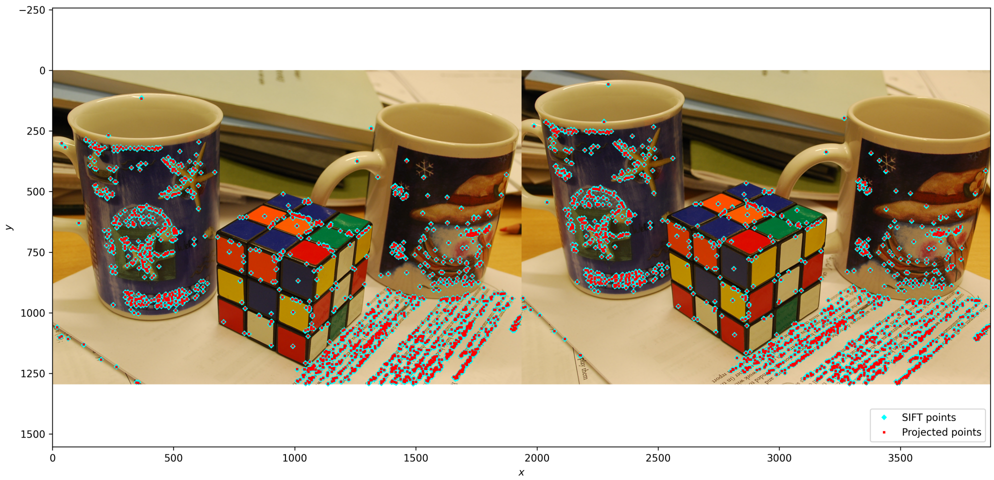

Figure 11: SIFT points and projected points in both images (unnormalised case). SIFT points are cyan and projected points are red.

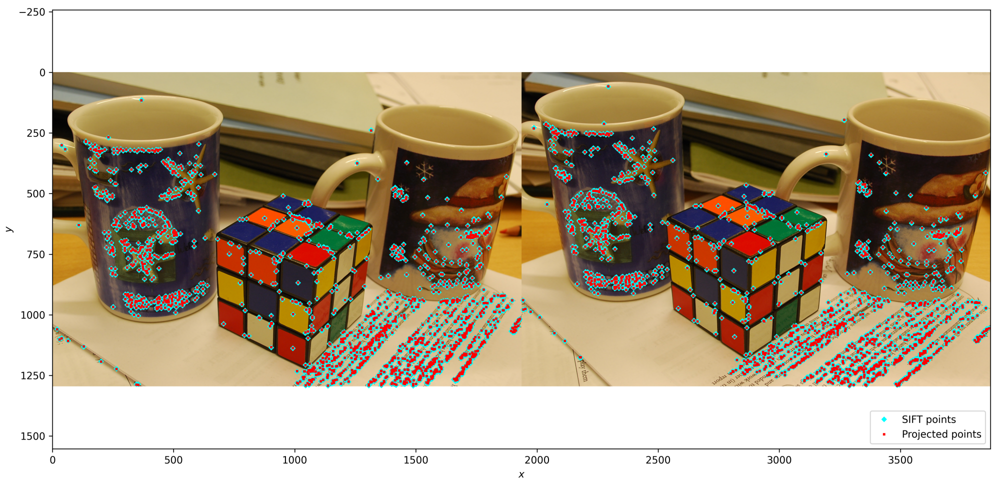

Figure 12: SIFT points and projected points in both images (normalised case). SIFT points are cyan and projected points are red.

Figure 13 and Figure 14 show the 3D reconstruction from triangulation using DLT. 3D objects are clearly distinguishable here; one can see contours of the two cups, and the text on the paper. The scale of the cube also seems accurate compared to the cups and paper around it.

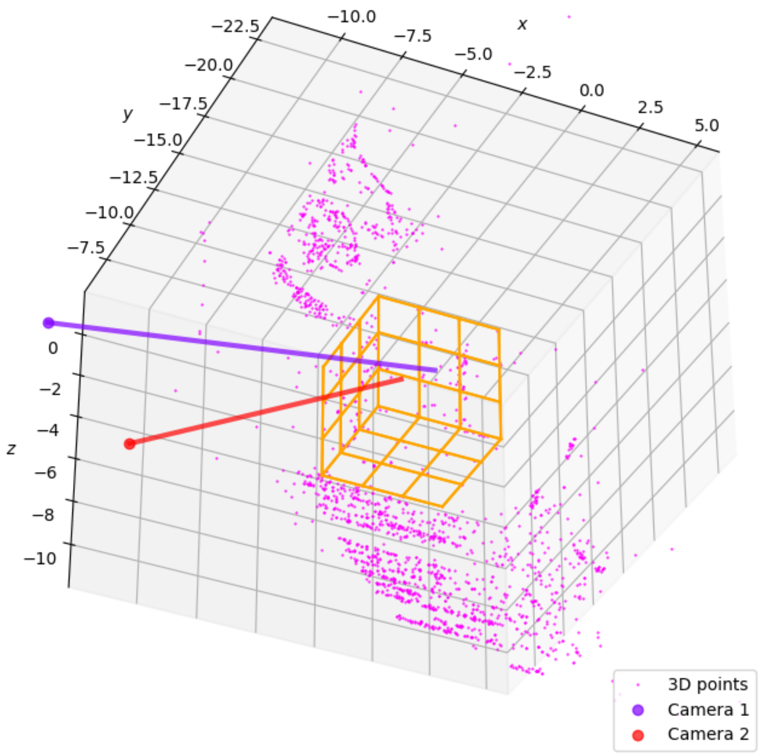

Figure 13: 3D reconstruction from triangulation using DLT. View from the right and in front of the cube.

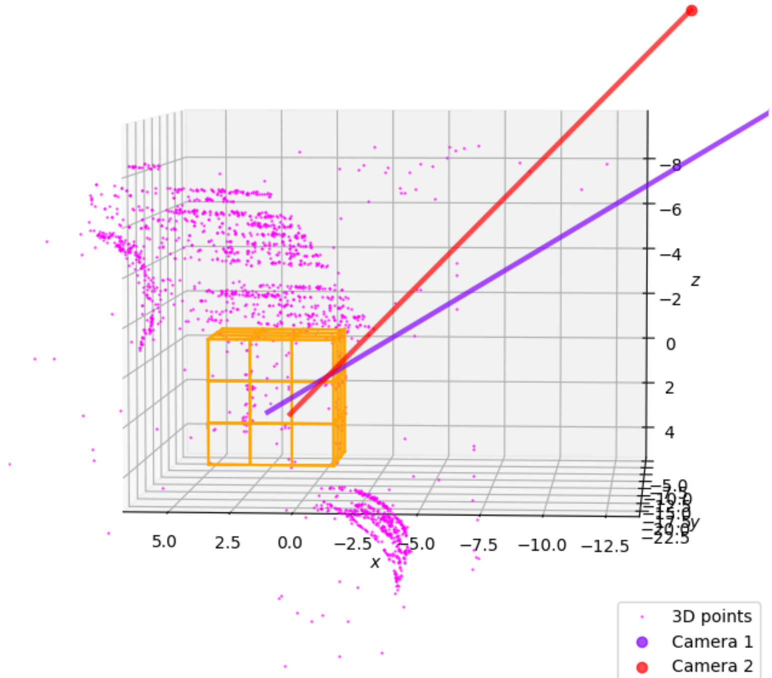

Figure 14: 3D reconstruction from triangulation using DLT. View from above the cube.

The average pixel error were different depending how many SIFT points that were considered. To properly reconstruct the 3D objects, all SIFT points were considered and those with greater pixel error than 3 were removed. For both the normalised and the unnormalised case, it followed that 6412 SIFT points were filtered down to 1926 points. Considering only the normalised case, the average pixel error before removing any points were 187 pixels for the first image, and 116 pixels for the second image which shows that most SIFT points were nonsense, which explains why so many points were removed. The average pixel error after removal were 1 pixel for both images. An interesting observation with unclear explanation was that there were some 3D points that were outliers, far away from the scene despite normalisation and removal of points.