Image Analysis and Computer Vision
Homework Report

Lorenzo Bardelli 10831941
lorenzo1.bardelli@mail.polimi.it

Prof. Vincenzo Caglioti

Contents

1 Problem Formulation

Using the features visible in the image scene:

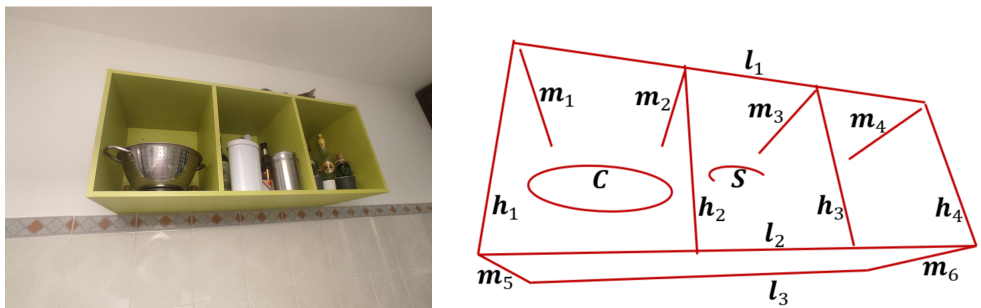

Figure 1: Photo of the cabinet and highlighted features

That is,

- the images of parallel lines along X: l_1, l_2, l_3.
- the images of parallel lines along Y: $m_1, m_2, m_3, m_4, m_5, m_6$.
- the images of parallel lines along Z: h_1, h_2, h_3, h_4.
- the image of the circumference C.
- the image of the unknown planar curve S.

And using the given data:

- cabinet width along X: $l = 1$.
- height of the planar curve S: $h/2$.
- zero-skew camera.

We need to find

1. the vanishing line l'_∞ of the horizontal plane.
2. the Euclidean rectification mapping H_R for the horizontal plane and the depth m of the parallelepiped.
3. the calibration matrix K.
4. the height h of the parallelepiped.
5. the X-Y coordinates of a dozen points on S.
6. the location of the camera in world coordinates.

2 Feature Extraction

2.1 Line detection

To automatically detect useful lines for reconstruction, the image is analyzed using the Canny edge detection algorithm and two distinct Hough transforms.
Each one tailored to specific angles for the height, depth and width lines in the image.

2.1.1 Canny edge detection

The image converted to gray-scale yields the following edges:

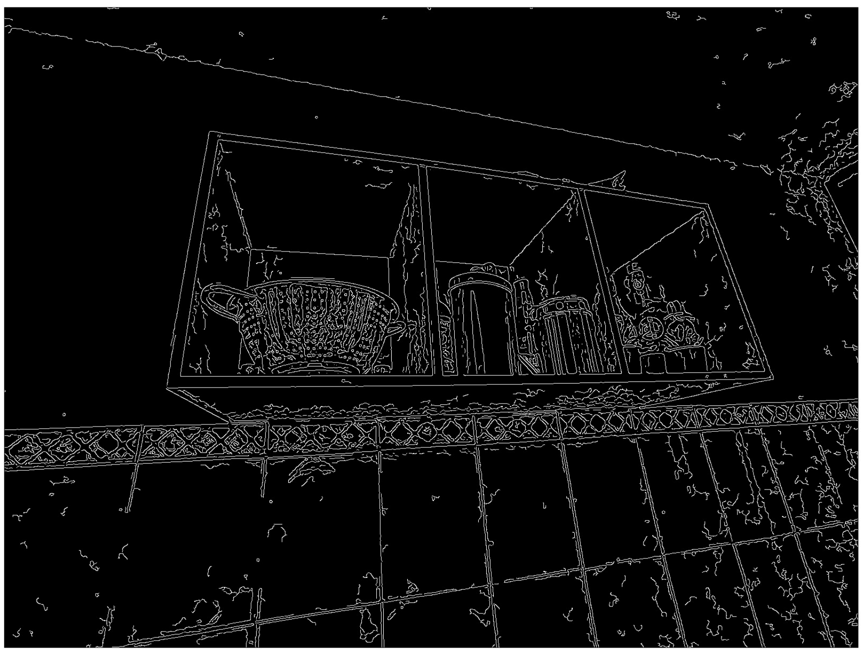

Figure 2: Canny edges from original image

Unfortunately, the image shows a lot of noise in the shaded areas.
To improve edge detection, the parameters of the Hough transform need to be tuned.

2.1.2 Hough transform

Main parameters used in MATLAB:

θ range	range of values for rotation angle
ρ spacing	resolution of values of distance from the origin
threshold	used to filter peaks

Table 1: Transform parameters

To increase robustness to noise the parameters of the transform are different for two cases:
The green lines have been computed using a θ ranging from -45° to 45° to search for vertical and diagonal lines.
The red lines have been computed using a θ ranging from -90° to -80° to search for horizontal lines.
Also, for the horizontal lines, the spacing of the bins has been increased to 2:
By doing this, close lines may be grouped together making the detected line position less accurate but the length is increased, ignoring discontinuities.

Figure 3: Hough lines

The lines indicated by the arrows will be used as an estimate for the images of the height, length and depth given their good approximation of the feature.
The line used to estimate length is significantly longer, as a consequence the estimate for height and depth after rectification will be shorter in both cases but still realistic.

2.1.3 Manual extraction

Other than selecting the best lines, manual intervention is used to draw the image of the circle C and the dozen points on the S curve:

Figure 4: Manual drawings

3 Proposed Solution

3.1 The vanishing line l'_∞ of the horizontal plane

To find l'_∞ we first need to find the vanishing points of the horizontal plane v_x, v_y. These points can be calculated by taking the intersection of any pair of parallel lines along the X and Y direction, in homogeneous coordinates.

$$l_i \longrightarrow l_1^T x = \begin{bmatrix} a & b & c \end{bmatrix}_i^T x = 0$$

for instance:

$$v_x = RNS[l_1^T, l_2^T] = l_1 \times l_2$$
$$v_y = RNS[m_1^T, m_2^T] = m_1 \times m_2$$

after calculating the vanishing points, the vanishing line is simply the line passing through them:

$$l'_\infty = RNS[v_x^T, v_y^T] = v_x \times v_y$$

in the same way we will also find the vanishing point for the vertical plane and the other line at infinity.

Experimental results:

$$v_x = \begin{bmatrix} 376,98 & 694,0554 & 1 \end{bmatrix}^T, v_y = \begin{bmatrix} 587,3763 & 884,4571 & 1 \end{bmatrix}^T, v_z = \begin{bmatrix} 668,0408 & -1369,4 & 1 \end{bmatrix}^T$$
$$l'_{\infty,1} = \begin{bmatrix} -0,00006 & -0,0011 & 1 \end{bmatrix}^T, l'_{\infty,2} = \begin{bmatrix} -0,00036 & -0,00055 & 1 \end{bmatrix}^T$$

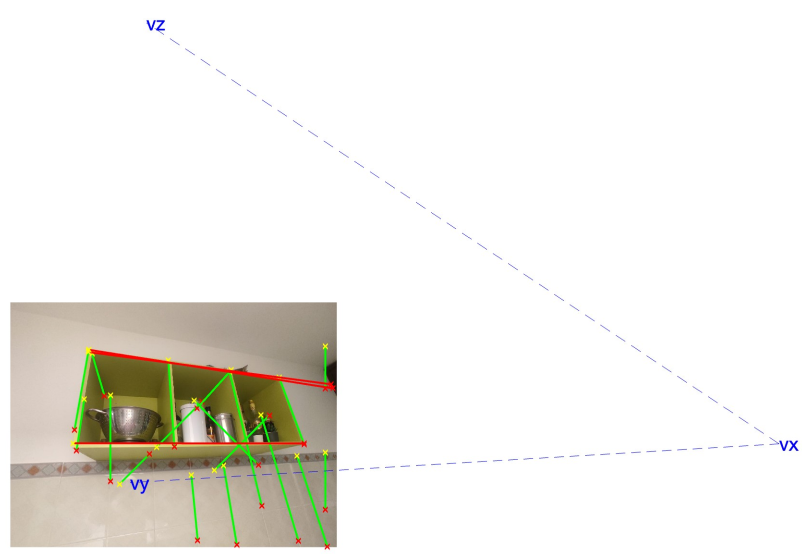

Figure 5: Vanishing points and lines

3.2 Euclidean rectification mapping H_R and depth m

We can compute an Homography using the image of the circle and the image of the horizontal line at infinity, following a stratified geometric approach to avoid numerical noise.

Step 1: affine reconstruction
Recall the following theorem
Theorem: l_∞ is invariant under affine transformations.

The affine reconstruction matrix is obtained mapping the image of the line at infinity to its canonical coordinates.

$$H_A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & {l'_\infty}^T & \end{bmatrix}$$

Step 2: shape reconstruction
From the image of the ellipse get the parameters for the center, semi-axes and rotation angle. Convert them to algebraic coefficients to build the conic matrix:

$$C' = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

Normalize and apply H_A to get the affine reconstructed circle Q.

$$Q = H_A^{-T} C H_A^{-1}$$

To build an Affinity from Q to a circle, extract the usual parameters (rotation angle α and semi-axes a, b), build the rotation matrix U and the scaling matrix S to rotate the ellipse and impose equal length (1) to both semi-axes.

$$U = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}, S = \begin{bmatrix} 1 & 0 \\ 0 & a/b \end{bmatrix}$$

The affinity matrix is then computed as a rotation, scaling and inverse rotation.

$$A = USU^T$$

The final metric rectification homography is simply the composition of the two steps:

$$H_R = AH_A$$

The Euclidean reconstructed image is

$$Rect = H_R Image$$

Finally, to obtain the depth m of the parallelepiped we can simply compute the ratio of lengths (invariant with respect to Euclidean transformations), using rectified lines and given real length $l = 1$:

$$\frac{m}{l} = \frac{m'}{l'} \rightarrow m = \frac{m'}{l'} l \rightarrow m = \frac{m'}{l'}$$

Experimental results:

$$H_R = \begin{bmatrix} 1,4273 & -0.6249 & 0 \\ -0,6249 & 1,9140 & 0 \\ -0,0001 & -0.0011 & 1 \end{bmatrix}, m = 0,2946$$

As stated before m is underestimated due to a longer image for the corresponding line.

Metric rectification of XY plane.

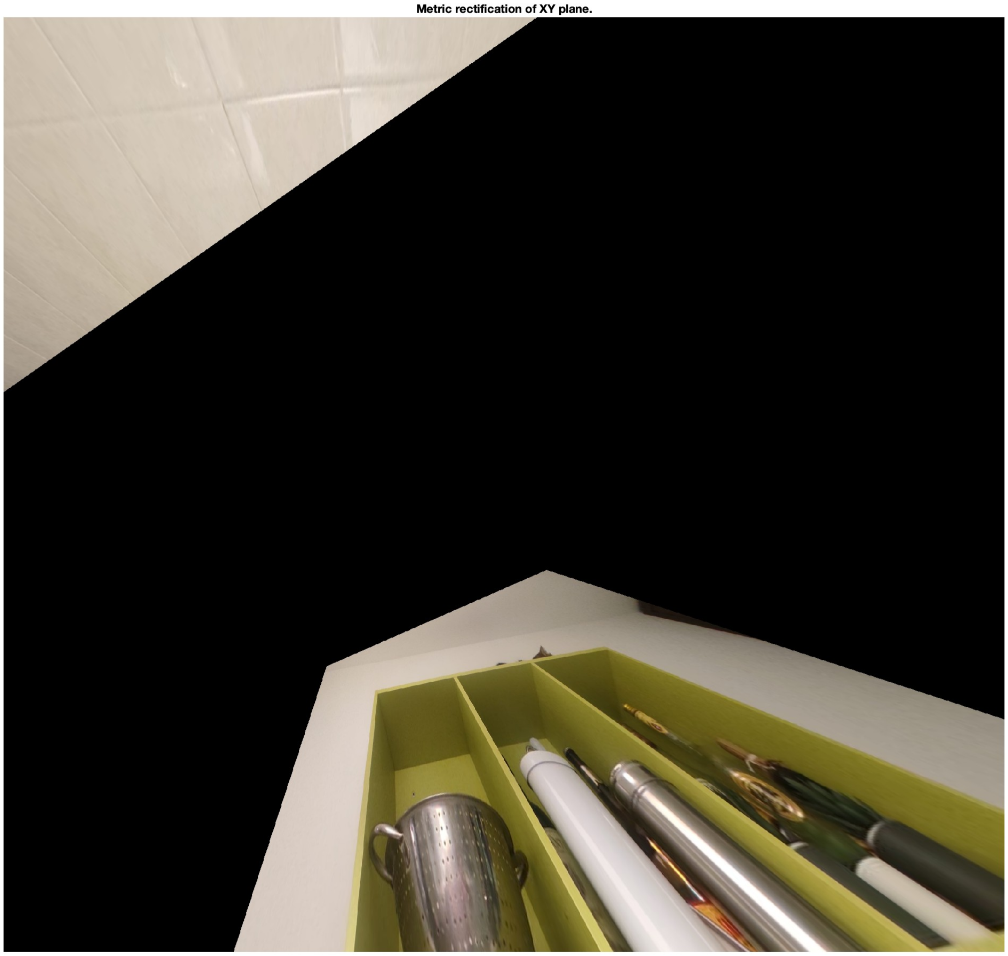

Figure 6: Metric rectification of the horizontal (upper) face

3.3 Calibration matrix K

To calculate the calibration matrix K for a zero-skew camera

$$K = \begin{bmatrix} f_x & 0 & U_0 \\ 0 & f_y & V_0 \\ 0 & 0 & 1 \end{bmatrix}$$

we recall that:

$$\omega = (KK^T)^{-1} = \begin{bmatrix} a^2 & 0 & -U_0 a^2 \\ 0 & 1 & -V_0 \\ -U_0 a^2 & -V_0 & f_y^2 + a^2 U_0^2 + V_0^2 \end{bmatrix} \text{ (4 unknowns)}$$

We can use the previously calculated homography H_R and the vanishing point v_z, orthogonal to the rectified face.
First we take the inverse of the rectifying homography:

$$H_R\text{(from image to plane)} \longrightarrow H = H_R^{-1} = \begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix} \text{(from plane to image)}$$

Note that

$$H = K \begin{bmatrix} r_{\pi 1} & r_{\pi 2} & o_\pi \end{bmatrix} \rightarrow \begin{bmatrix} r_{\pi 1} & r_{\pi 2} & o_\pi \end{bmatrix} = K^{-1} H = \begin{bmatrix} K^{-1}\alpha & K^{-1}\beta & K^{-1}\gamma \end{bmatrix}$$

In other words the vectors $K^{-1}\alpha, K^{-1}\beta$ are orthogonal and have the same module, yielding 2 constraints:

$$\begin{cases} \alpha^T K^{-T} K^{-1} \beta = 0 \\ \alpha^T K^{-T} K^{-1} \alpha - \beta^T K^{-T} K^{-1} \beta = 0 \end{cases} = \begin{cases} \alpha^T \omega \beta = 0 \\ \alpha^T \omega \alpha - \beta^T \omega \beta = 0 \end{cases} \text{ (2 equations)}$$

We then impose the orthogonality relation to v_x

$$\begin{cases} v_x^T \omega \alpha = 0 \\ v_y^T \omega \beta = 0 \end{cases} \text{ (2 equations)}$$

After solving for ω apply Cholesky factorization to isolate K

$$\omega = (KK^T)^{-1} \longrightarrow \omega^{-1} = KK^T$$

Experimental results:

$$\omega = \begin{bmatrix} 1,047 & 0 & -815,665 \\ 0 & 1 & -573,196 \\ -815,665 & -573,196 & 1546389 \end{bmatrix}$$

$$K = \begin{bmatrix} 1,4132 & 0,5425 & 0,0009 \\ 0 & 1,1268 & 0,0004 \\ 0 & 0 & 0,0008 \end{bmatrix}$$

(not normalized, not zero-skew)

by manually reconstructing K using individual parameters of ω we get a correct zero-skew calibration matrix

$$K_{man} = \begin{bmatrix} 780 & 0 & 779,05 \\ 0 & 763,146 & 573,196 \\ 0 & 0 & 1 \end{bmatrix}$$

3.4 Height h of the parallelepiped

To compute the euclidean rectification for the frontal face we use the following theorem
Theorem: circular points are invariant under Euclidean transformation.

from ω find the image of the absolute conic, then extract the circular points intersecting it with the image of the line at infinity for the vertical plane.

$$\begin{cases} \omega_{1,1}x^2 + 2\omega_{1,2}xy + \omega_{2,2}y^2 + 2\omega_{1,3}x + 2\omega_{2,3}y + \omega_{3,3} = 0 \\ {l'_{\infty,2}}^1 x + {l'_{\infty,2}}^2 y + {l'_{\infty,2}}^3 = 0 \end{cases}$$

getting the image of the circular points I', J'.
Using them, calculate the image of their dual conic.

$$C^{*\prime}_{\infty} = I'J'^T + J'I'^T$$

From it we can extract the homography with singular value decomposition

$$SVD(C^{*\prime}_{\infty}) = UDV^T = H_{SR}^{-1} C^*_{\infty} H_{SR}^{-T}$$

$$H_{SR} = (UD^{1/2})^{-1}$$

After that by knowing that the rectifying homography preserves length ratios we can compute the height (same procedure used for m)

$$\frac{h}{l} = \frac{h'}{l'} \rightarrow h = \frac{h'}{l'} l \rightarrow h = \frac{h'}{l'}$$

Experimental results:

$$H_{SR} = \begin{bmatrix} -0,9977 & -0,0683 & -0,0003 \\ 0,1272 & -1,8585 & 0,0011 \\ -0,0004 & 0,0006 & 1 \end{bmatrix}, h = 0,4918$$

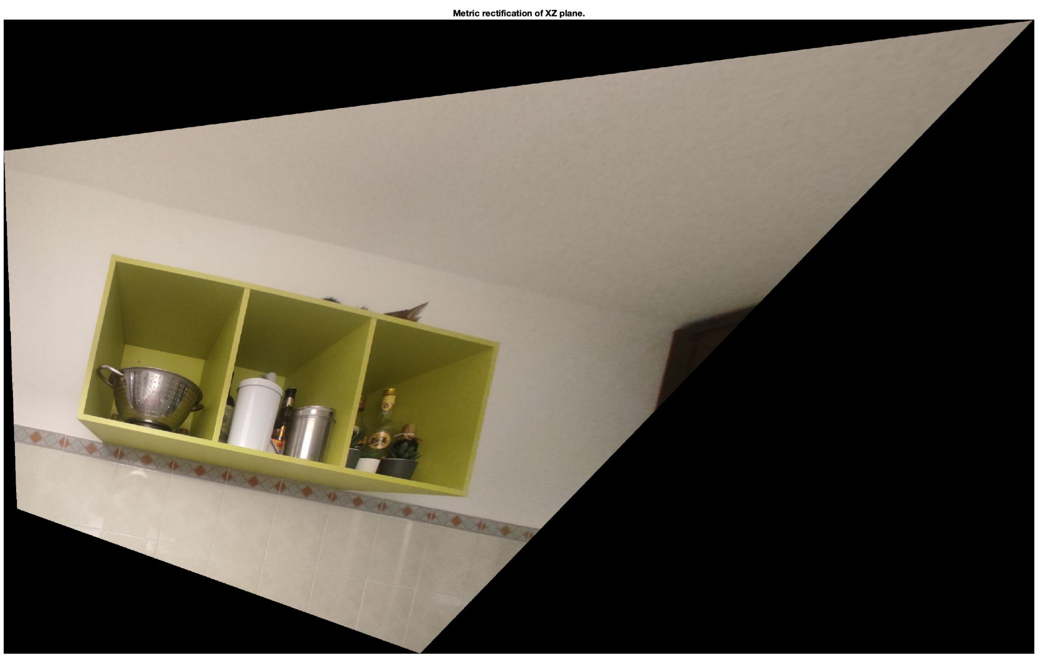

Figure 7: Metric rectification of the frontal face

3.5 Coordinates of points on S

We leverage the previously computed homography H_R for the horizontal plane.
The (x, y) coordinates of a point p_s on S can be computed applying the rectifying homography to the image of that point:

$$p_s = H_{SR} * p'_s$$

We repeat this calculation for a dozen points.

Experimental results:
The obtained point coordinates in the image plane trace a correct image of the rectified curve:

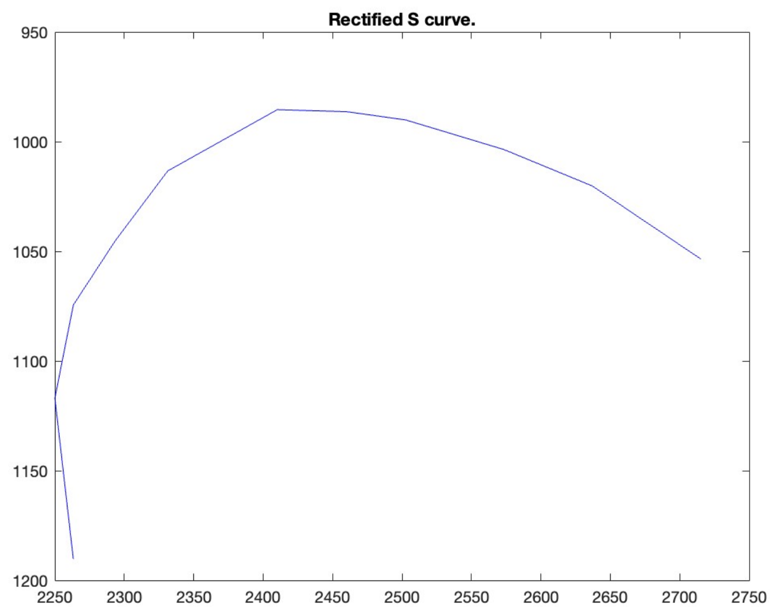

Figure 8: Rectified S curve

3.6 Camera localization

Having calibrated the camera, use K to calculate the extrinsic parameters of the camera (rotation R and translation t).
The rotation of the horizontal plane with respect to the camera is

$$R_\pi = \begin{bmatrix} r_1 & r_2 & o_\pi \end{bmatrix} = K^{-1} H_r^{-1}$$

Since r_3 is orthogonal to r_1 and r_2

$$r_3 = r_1 \times r_2$$

We build the rotation matrix R and translation vector t as follows

$$R = \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}, t = \begin{bmatrix} o_\pi \end{bmatrix}$$

We finally get the camera relative position

$$C = -R^T t$$

and the projection matrix

$$P = K \begin{bmatrix} R|t \end{bmatrix}$$

Experimental results:

$$R = \begin{bmatrix} 0,9006 & -0,4308 & -0,0569 \\ 0,1100 & 0,3527 & -0,9293 \\ 0,4204 & 0,8307 & 0,3650 \end{bmatrix}, t = \begin{bmatrix} -0,9977 \\ -0,7511 \\ 1 \end{bmatrix}$$

$$P = \begin{bmatrix} 1,3329 & -0,4167 & -0,5842 & -1,8165 \\ 0,1241 & 0,3978 & -1,0470 & -0,8460 \\ 0,0003 & 0,0007 & 0,0003 & 0,0008 \end{bmatrix}$$

$$C = \begin{bmatrix} 0,5608 \\ -0,9956 \\ -1,1198 \end{bmatrix}$$

4 Final reconstructed scene

Using computed data about camera postition, rotation and object dimensions we can plot an estimate of the original 3D scene.
By doing so we find that it is coherent with the observed perspective of the original image:

- The camera is below the height of the lower plane of the cabinet.
- The camera is decentered to the left.
- The camera is facing slighlty upwards and to the right

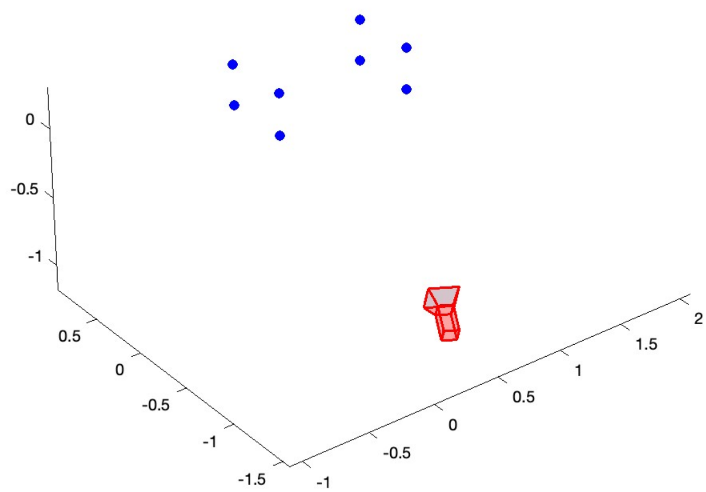

Figure 9: Reconstructed 3D scene