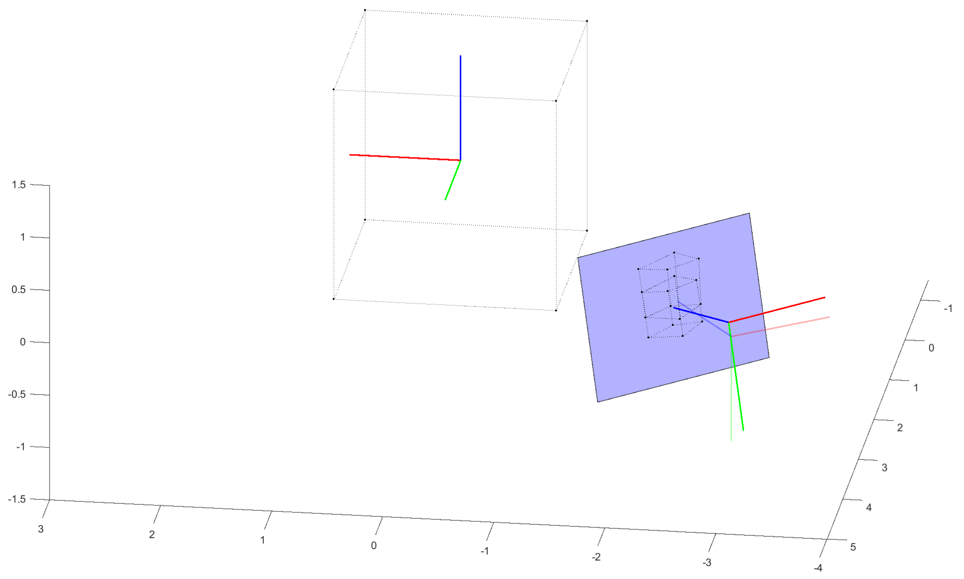
UiO : Department of Technology Systems
University of Oslo

Estimating pose by minimizing reprojection error

Trym Vegard Haavardsholm



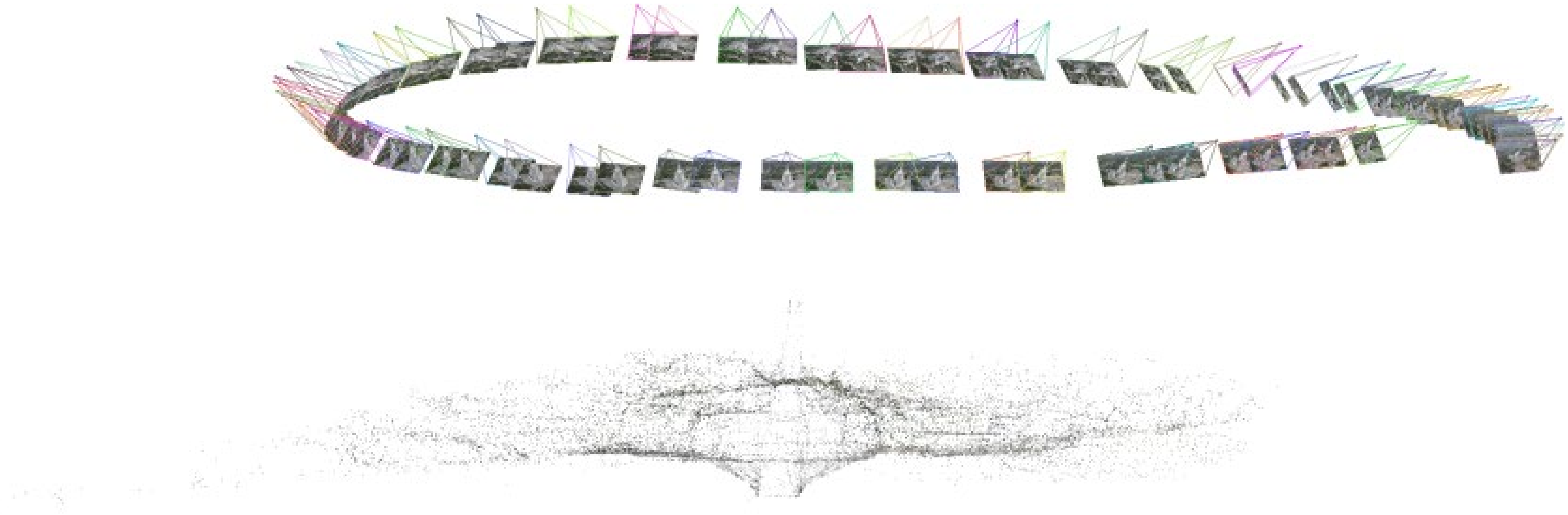
TEK5030

Bundle adjustment

Bundle Adjustment (BA)

Estimating the imaging geometry
based on minimizing reprojection error

- Motion-only BA
- Structure-only BA
- Full BA

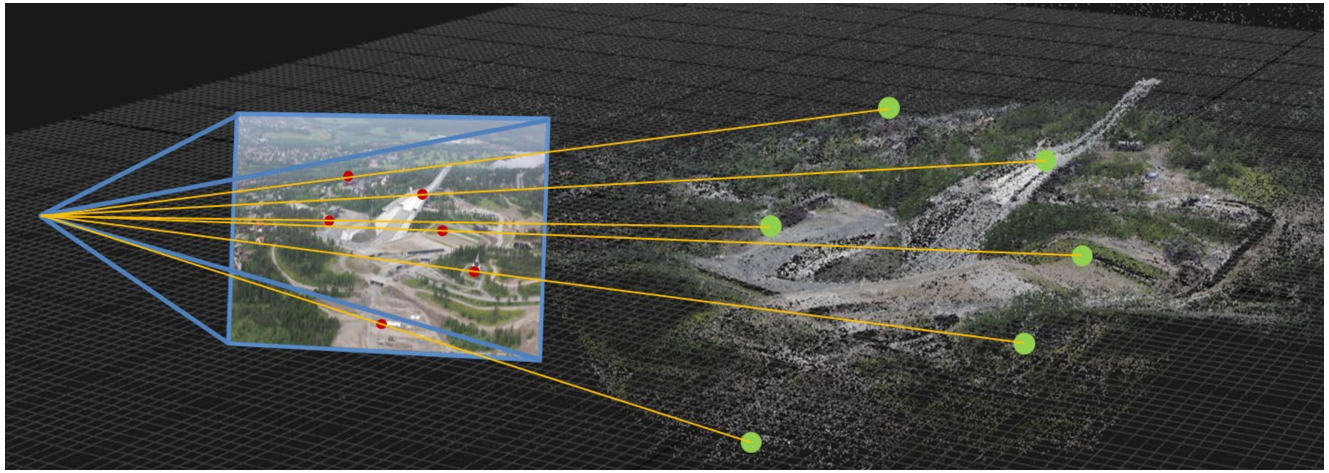


Pose estimation by minimizing reprojection error

Minimize **geometric error** over the **camera pose** given **known structure**

This is also sometimes called **Motion-Only Bundle Adjustment**

$$\mathbf{T}_{wc}^{*} = \underset{\mathbf{T}_{wc}}{\operatorname{argmin}} \sum_{j} \left\| \pi(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_{j}^{w}) - \mathbf{u}_{j} \right\|^{2}$$



Pose estimation by minimizing reprojection error

Given:

- World points $\mathbf{x}_j^w$

Measurements:

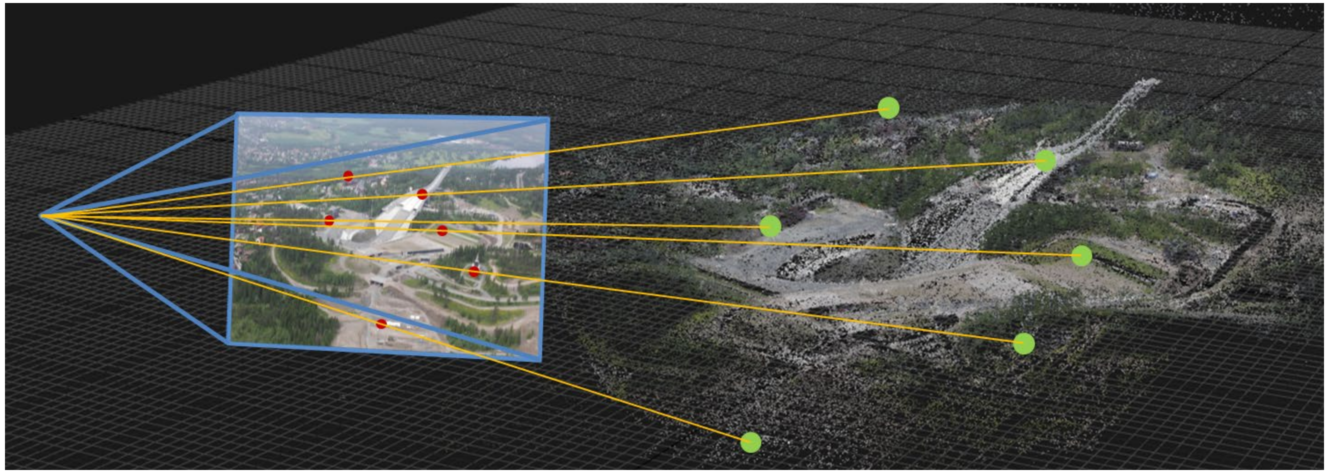
- Correspondences $\mathbf{u}_j \leftrightarrow \mathbf{x}_j^w$ with measurement noise $\mathbf{\Sigma}_j$

State we wish to estimate:

- Camera pose $\mathbf{T}_{wc}$

Initial estimate:

- Motion model
- Pose from homography
- PnP (P3P, EPnP, …)

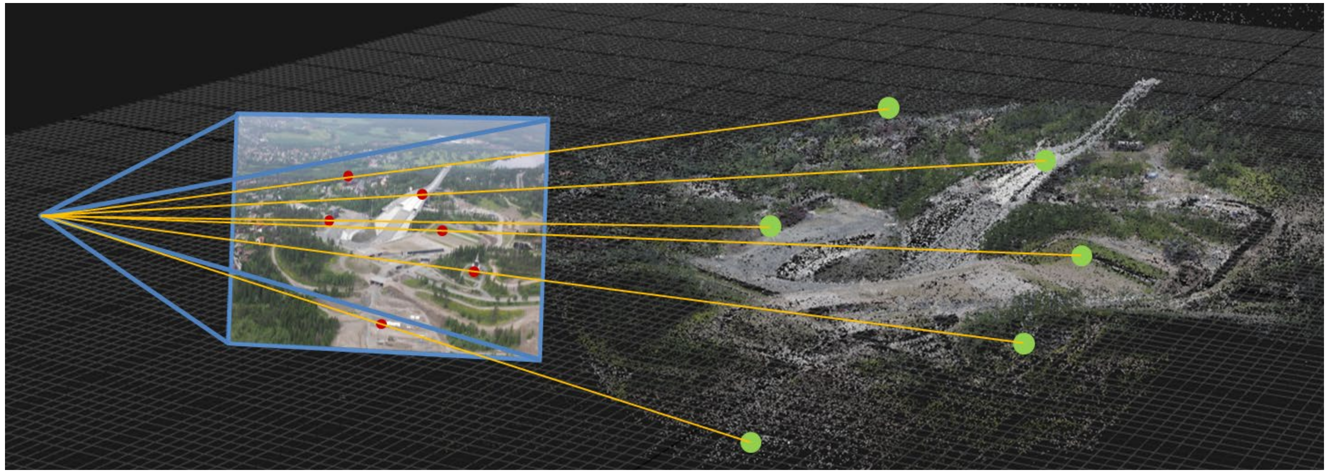


n-Point Pose Problem (P*n*P)

- Typically fast non-iterative methods
- Minimal in number of points, perfect for RANSAC!
- Accuracy can be comparable to iterative methods
- Good as initial estimates

Examples

- P3P, EPnP
- P4Pf
  - Estimate pose and focal length
- P6P
  - Estimates **P** with DLT
- R6P
  - Estimate pose with rolling shutter



Applying the MAP framework

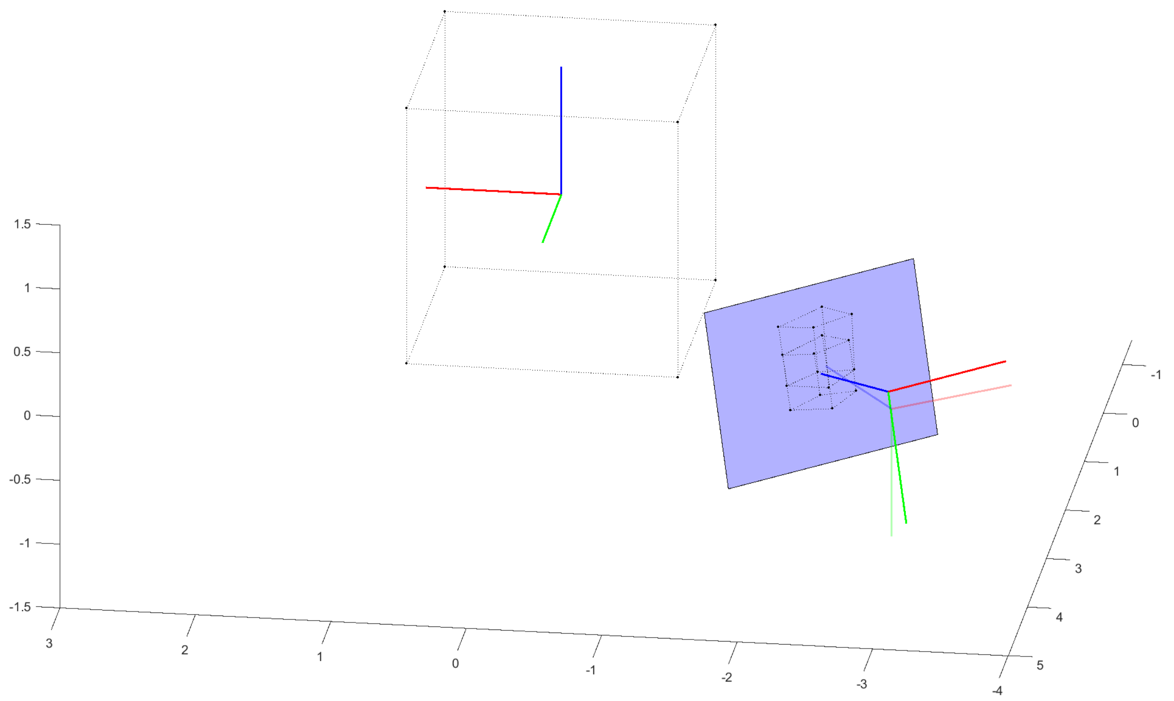
For simplicity,
we pre-calibrate to normalized image coordinates (and propagate the noise)

This gives us the measurement prediction function

$$h_j(\mathbf{T}_{wc}) = \pi_n(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_j^w)$$

and measurement error function

$$e_j(\mathbf{T}_{wc}) = \pi_n(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_j^w) - \mathbf{x}_{n\,j}$$



Applying the MAP framework

$$\mathbf{x}_n = \pi_n(\mathbf{x}^c) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \frac{1}{z^c} \mathbf{x}^c = \begin{bmatrix} x^c / z^c \\ y^c / z^c \end{bmatrix}$$

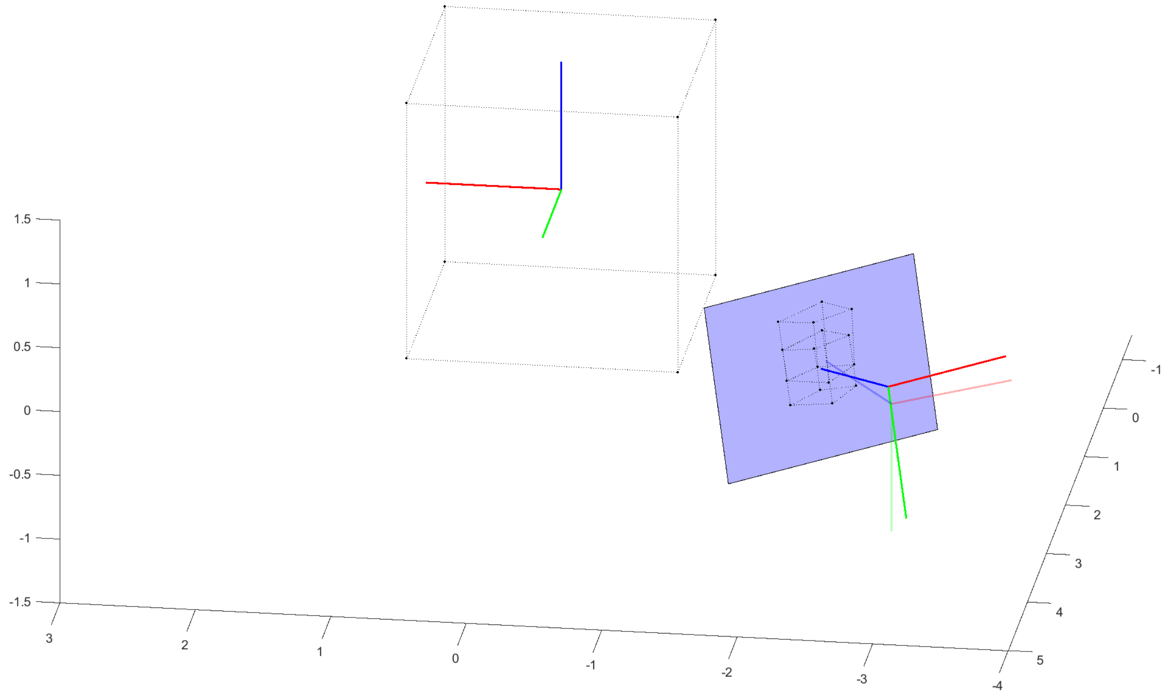
For simplicity,
we pre-calibrate to normalized image coordinates (and propagate the noise)

This gives us the measurement prediction function

$$h_j(\mathbf{T}_{wc}) = \pi_n(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_j^w)$$

and measurement error function

$$e_j(\mathbf{T}_{wc}) = \pi_n(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_j^w) - \mathbf{x}_{n\,j}$$



Applying the MAP framework

$$\mathbf{x}_n = \pi_n(\mathbf{x}^c) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \frac{1}{z^c}\mathbf{x}^c = \begin{bmatrix} x^c/z^c \\ y^c/z^c \end{bmatrix}$$

The measurement Jacobian is given by

$$\begin{aligned}
\mathbf{J}^h_{\mathbf{T}_{wc}} &= \mathbf{J}^{\pi_n(\mathbf{T}_{wc}^{-1}\cdot\mathbf{x}^w)}_{\mathbf{T}_{wc}^{-1}\cdot\mathbf{x}^w} \mathbf{J}^{\mathbf{T}_{wc}^{-1}\cdot\mathbf{x}^w}_{\mathbf{T}_{wc}^{-1}} \mathbf{J}^{\mathbf{T}_{wc}^{-1}}_{\mathbf{T}_{wc}} \\
&= \mathbf{J}^{\pi_n(\mathbf{x}^c)}_{\mathbf{x}^c} \mathbf{J}^{\mathbf{T}_{wc}^{-1}\cdot\mathbf{x}^w}_{\mathbf{T}_{wc}^{-1}} \mathbf{J}^{\mathbf{T}_{wc}^{-1}}_{\mathbf{T}_{wc}} \\
&= \frac{1}{z^c}\begin{bmatrix} 1 & 0 & -x^c/z^c \\ 0 & 1 & -y^c/z^c \end{bmatrix} \begin{bmatrix} \mathbf{R}_{wc}^\top & -\mathbf{R}_{wc}^\top[\mathbf{x}^w]_\times \end{bmatrix} \cdot - \begin{bmatrix} \mathbf{R}_{wc} & [\mathbf{t}^w_{wc}]_\times \mathbf{R}_{wc} \\ \mathbf{0} & \mathbf{R}_{wc} \end{bmatrix} \\
&= d \begin{bmatrix} 1 & 0 & -x_n \\ 0 & 1 & -y_n \end{bmatrix} \begin{bmatrix} -\mathbf{I} & [\mathbf{x}^c]_\times \end{bmatrix} \\
&= \begin{bmatrix} -d & 0 & dx_n & x_n y_n & -1 - x_n^2 & y_n \\ 0 & -d & dy_n & 1 + y_n^2 & -x_n y_n & -x_n \end{bmatrix},
\end{aligned}$$

Applying the MAP framework

This results in the linearized weighted least squares problem

$$\begin{aligned}\boldsymbol{\xi}^* &= \arg\min_{\boldsymbol{\xi}} \sum_{j=1}^{n} \left\| \mathbf{A}_j \boldsymbol{\xi} - \mathbf{b}_j \right\|^2 \\ &= \arg\min_{\boldsymbol{\xi}} \left\| \mathbf{A} \boldsymbol{\xi} - \mathbf{b} \right\|^2 ,\end{aligned}$$

where

$$\begin{aligned}\mathbf{A}_j &= \boldsymbol{\Sigma}_{n\,j}^{-1/2} \mathbf{J}_{\mathbf{T}_{wc}}^{h_j} \\ \mathbf{b}_j &= \boldsymbol{\Sigma}_{n\,j}^{-1/2} (\mathbf{x}_{n\,j} - h_j(\mathbf{T}_{wc})),\end{aligned} \qquad \mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_n \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_n \end{bmatrix} .$$

Applying the MAP framework

For an example with *three points*,
the measurement Jacobian $\mathbf{A}$ and the prediction error $\mathbf{b}$ are

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \mathbf{A}_3 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \end{bmatrix}$$

Applying the MAP framework

The solution can be found by solving the normal equations

$$\left(\mathbf{A}^T\mathbf{A}\right)\boldsymbol{\xi}^* = \mathbf{A}^T\mathbf{b}$$

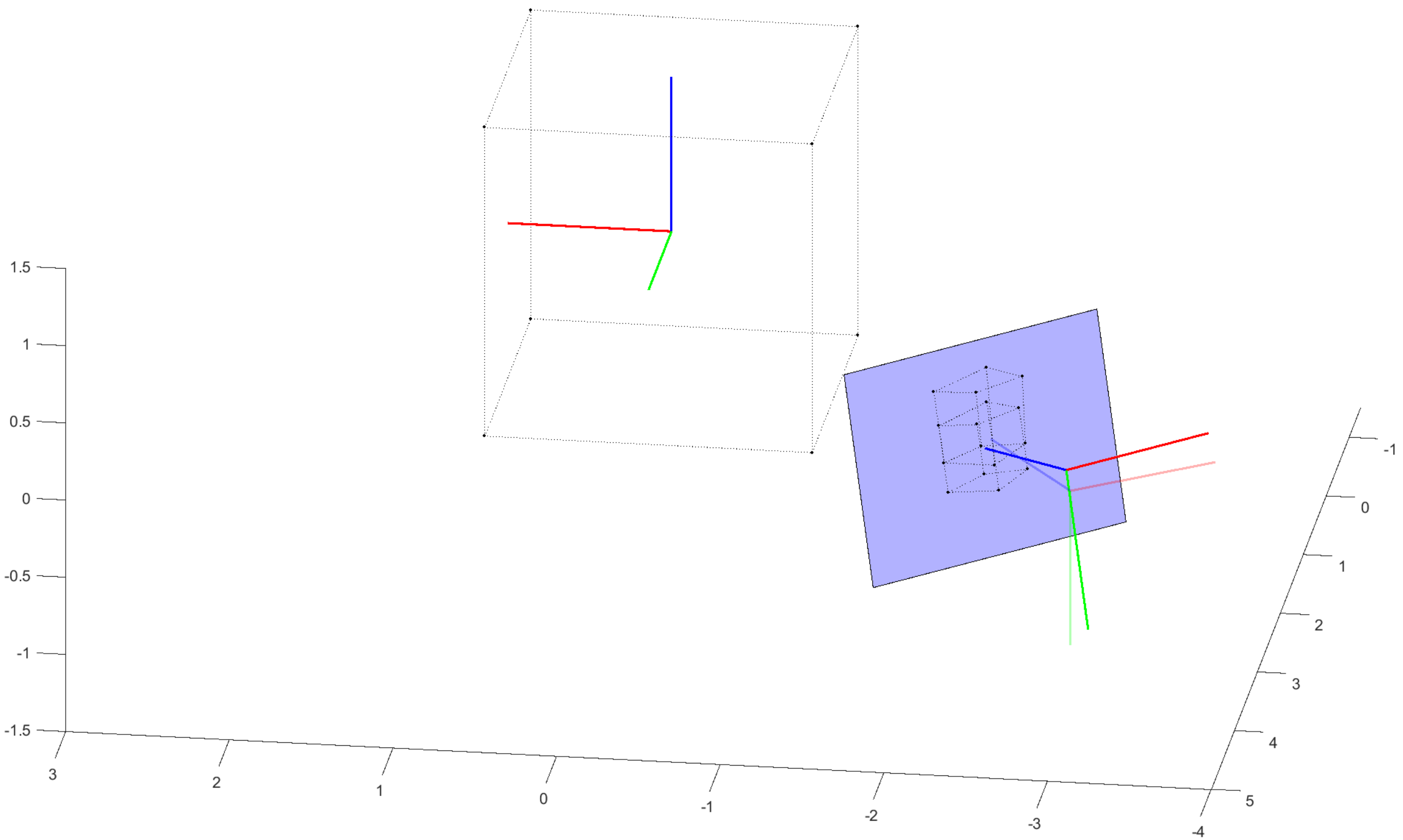
Choose a suitable inital estimate $\underline{\hat{\mathcal{X}}}^0$

$$\mathbf{A}, \mathbf{b} \leftarrow \text{Linearize at } \underline{\hat{\mathcal{X}}}^t$$

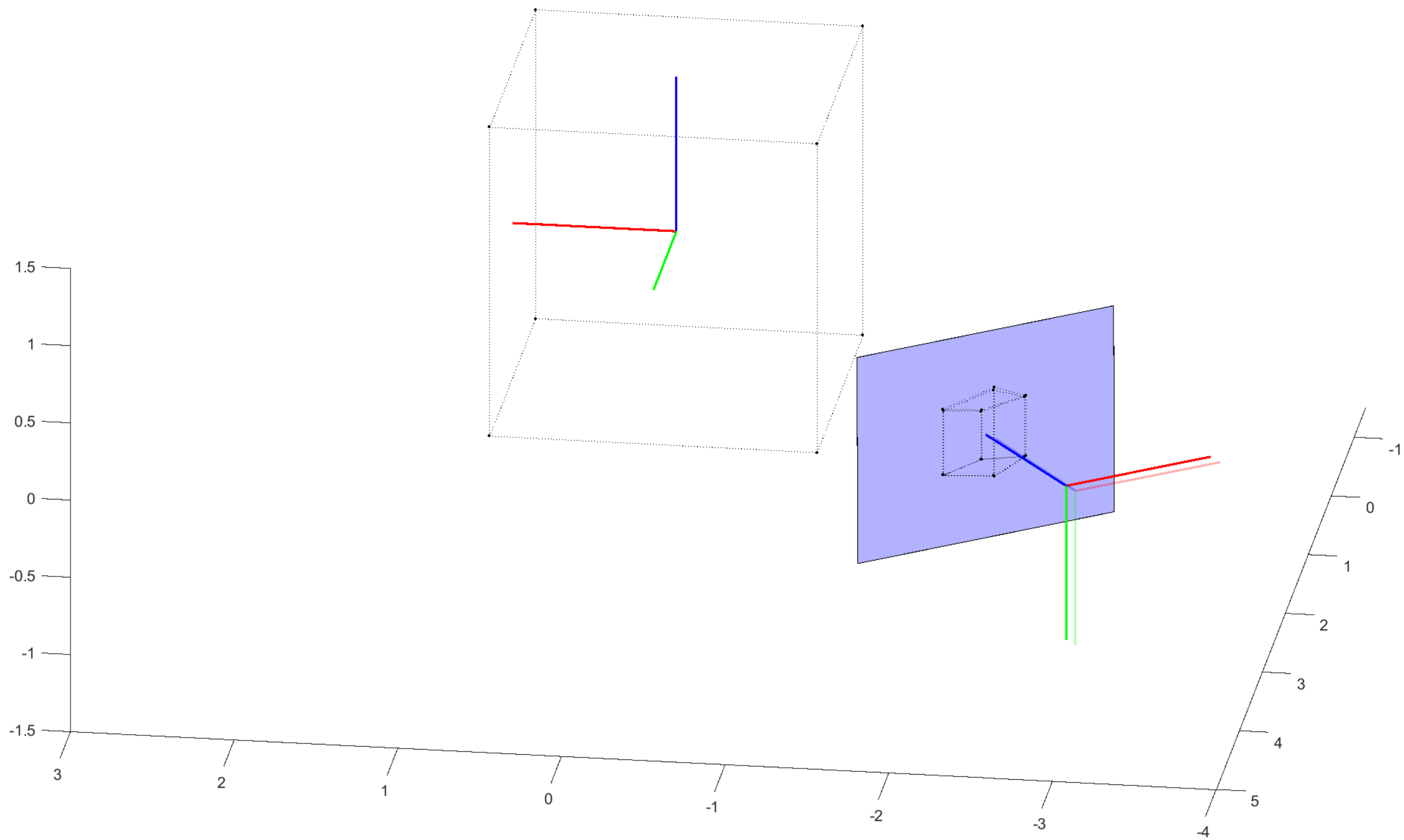
$$\underline{\boldsymbol{\tau}}^* \leftarrow \mathit{S}\text{olve } \underset{\underline{\boldsymbol{\tau}}}{\operatorname{argmin}} \left\| \mathbf{A}\underline{\boldsymbol{\tau}} - \mathbf{b} \right\|^2$$

$$\underline{\hat{\mathcal{X}}}^{t+1} \leftarrow \underline{\hat{\mathcal{X}}}^t \oplus \underline{\boldsymbol{\tau}}^*$$

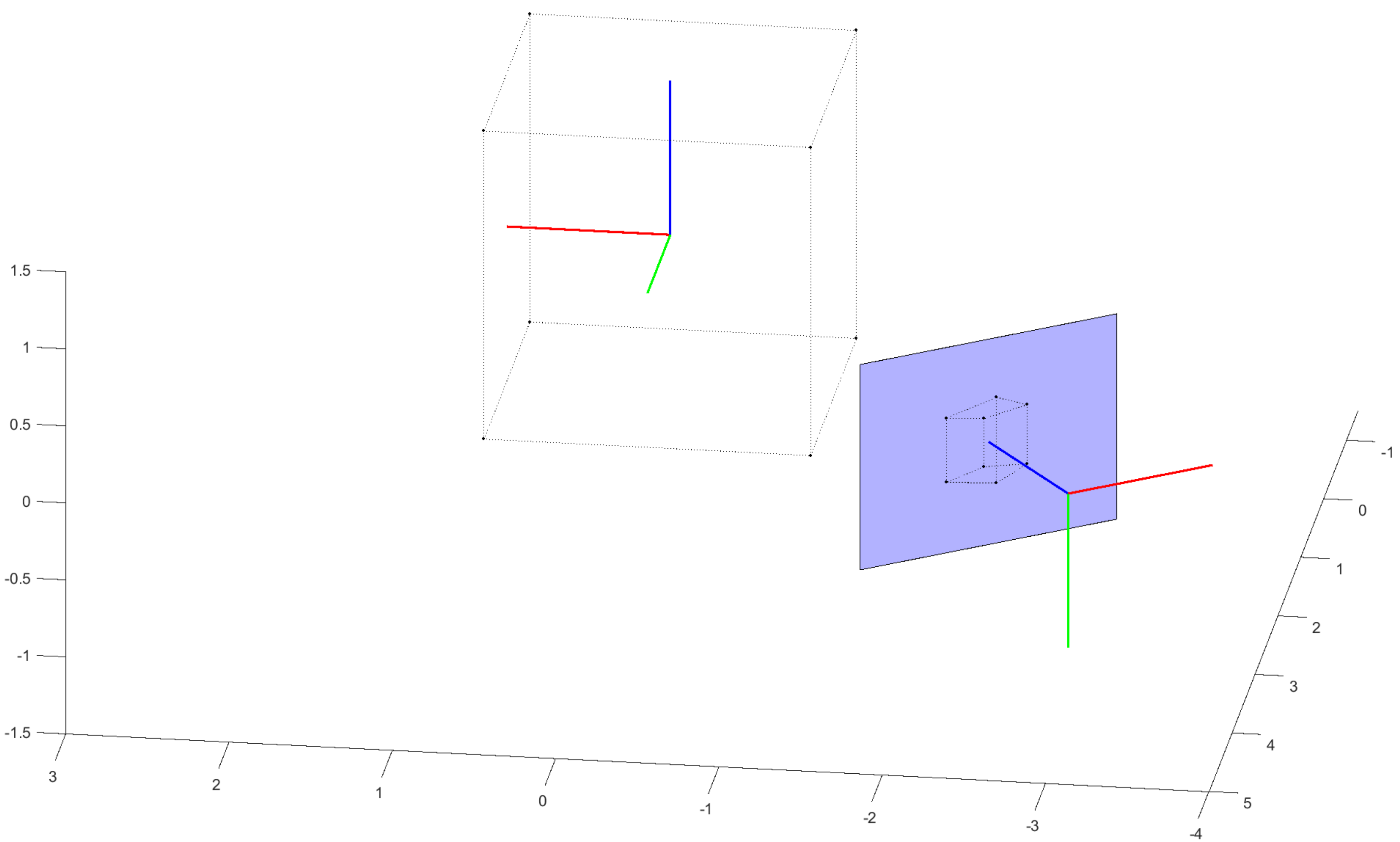
Example



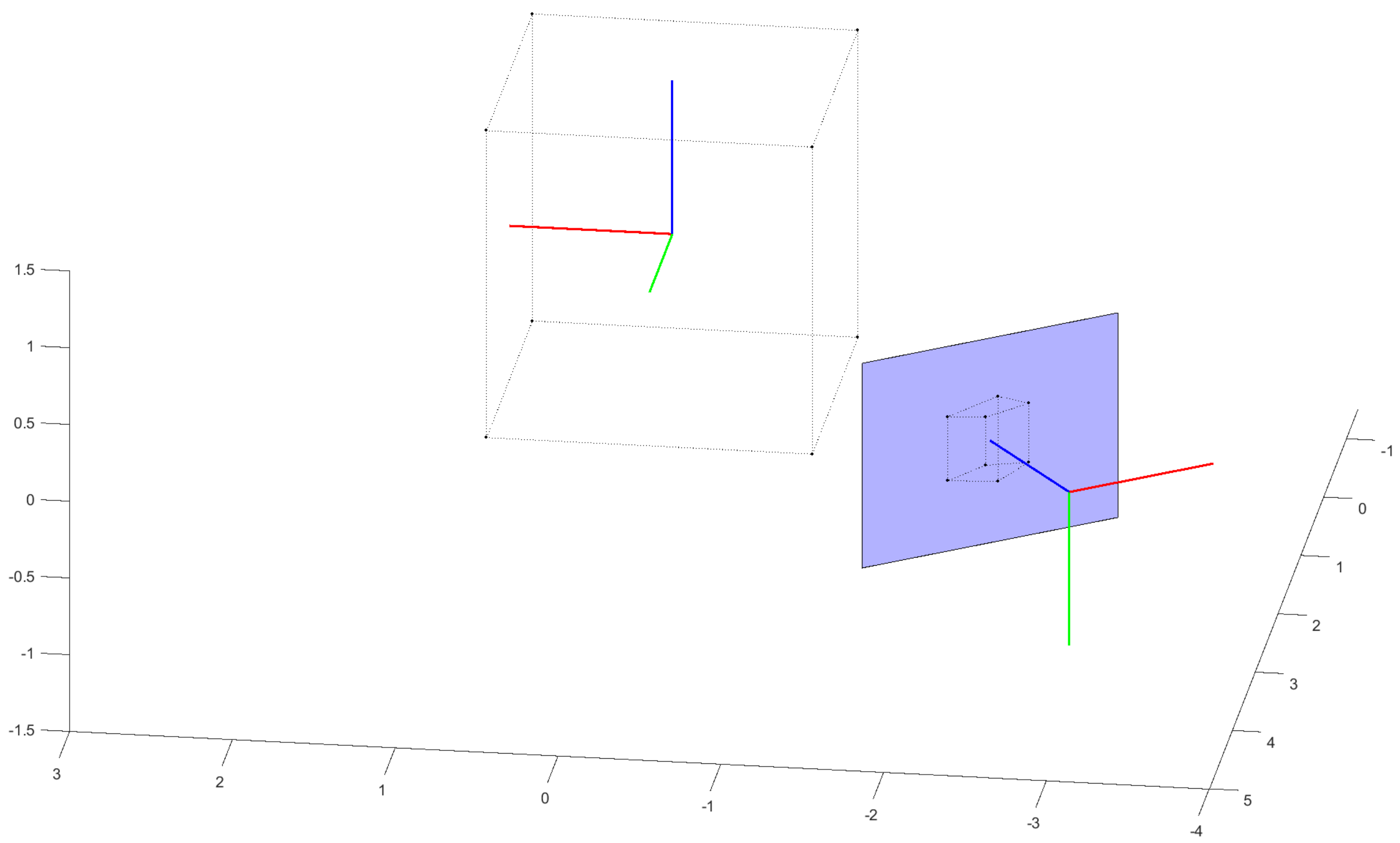
Example



Example



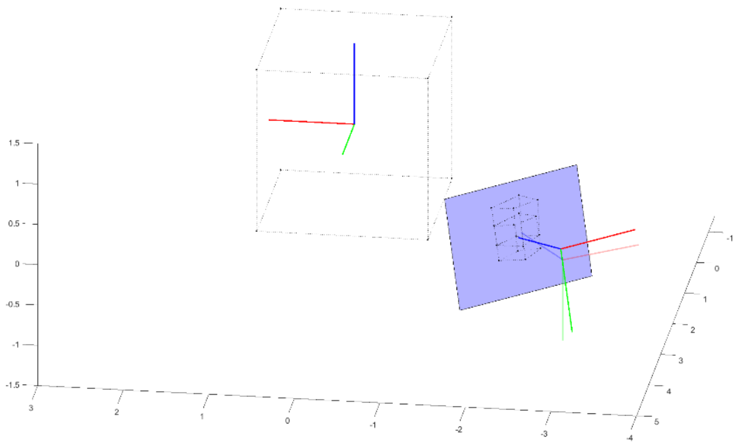
Example



Summary

- Nonlinear MAP estimation of states such as poses

- Pose estimation relative to known 3D points
  - Pose from homography
  - PnP
  - Minimising reprojection error



Choose a suitable inital estimate $\underline{\hat{\mathcal{X}}}^0$

$$\mathbf{A}, \mathbf{b} \leftarrow \text{Linearize at } \underline{\hat{\mathcal{X}}}^t$$

$$\underline{\boldsymbol{\tau}}^* \leftarrow \text{Solve } \underset{\underline{\boldsymbol{\tau}}}{\operatorname{argmin}} \left\| \mathbf{A}\underline{\boldsymbol{\tau}} - \mathbf{b} \right\|^2$$

$$\underline{\hat{\mathcal{X}}}^{t+1} \leftarrow \underline{\hat{\mathcal{X}}}^t \oplus \underline{\boldsymbol{\tau}}^*$$

Supplementary material

- The compendium!
  - Let me know if you would like to go through the derivations in greater detail!

- Python implementation of the bundle adjustment examples:
  - https://github.com/ttk21/lab_05