Lecture 8.2
Triangulation

Thomas Opsahl

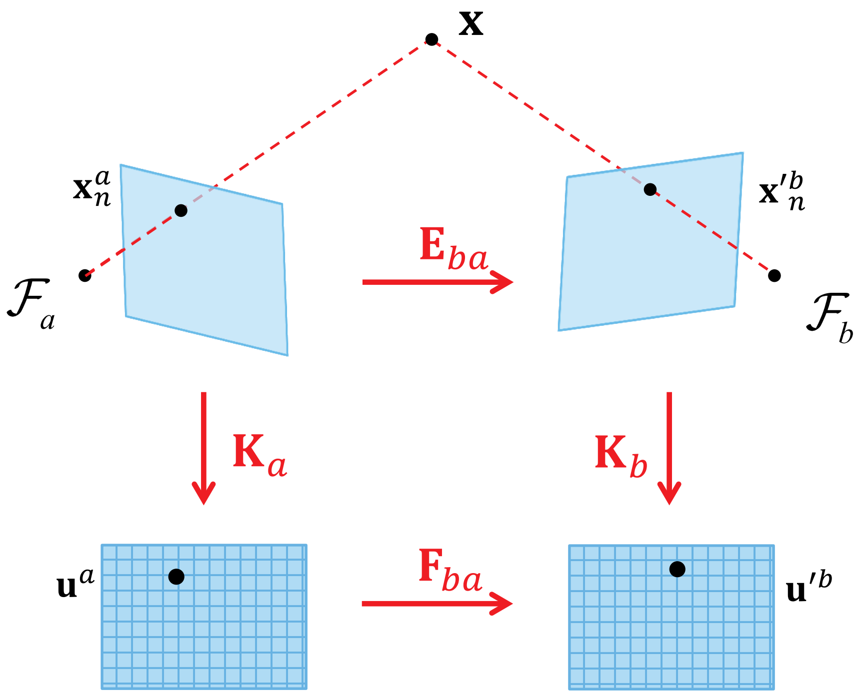
Introduction

We have seen that two perspective cameras observing the same points must satisfy the epipolar constraint

$$\left(\tilde{\mathbf{x}}_n'^b\right)^T \mathbf{E}_{ba}\tilde{\mathbf{x}}_n^a = 0 \qquad \left(\tilde{\mathbf{u}}'^b\right)^T \mathbf{F}_{ba}\tilde{\mathbf{u}}^a = 0$$

Being observed by two perspective cameras also puts a strong geometric constraint on the observed point $\mathbf{x}$

In the following we will look at how we can estimate observed 3D points $\mathbf{x}_i$ from known correspondences $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i'^b$ when we know $\mathbf{P}_a$ and $\mathbf{P}_b$



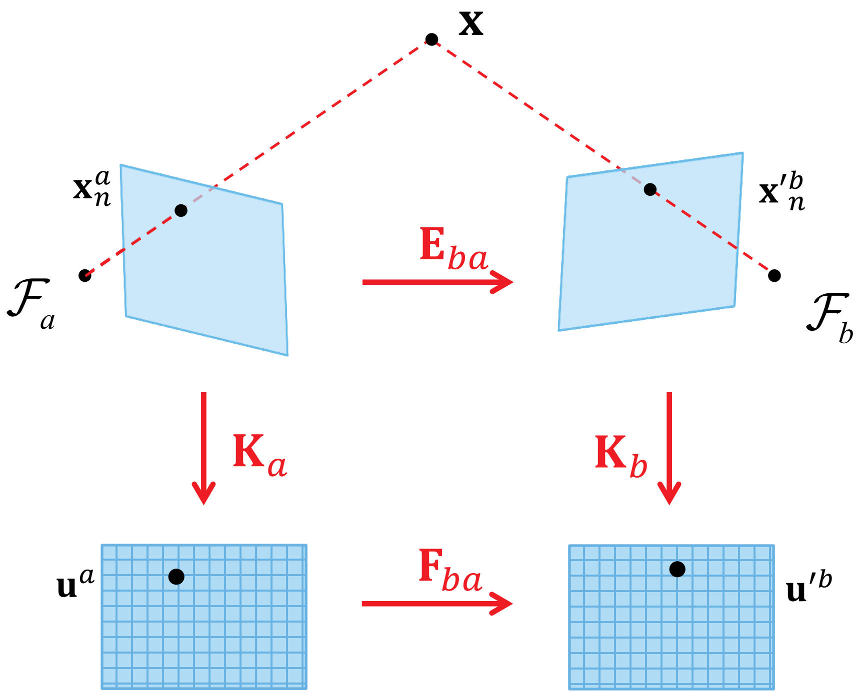
Introduction

We have seen that two perspective cameras observing the same points must satisfy the epipolar constraint

$$\left(\tilde{\mathbf{x}}_n'^b\right)^T \mathbf{E}_{ba}\tilde{\mathbf{x}}_n^a = 0 \qquad \left(\tilde{\mathbf{u}}'^b\right)^T \mathbf{F}_{ba}\tilde{\mathbf{u}}^a = 0$$

Being observed by two perspective cameras also puts a strong geometric constraint on the observed point $\mathbf{x}$

In the following we will look at how we can estimate observed 3D points $\mathbf{x}_i$ from known correspondences $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i'^b$ when we know $\mathbf{P}_a$ and $\mathbf{P}_b$



Recall that the camera projection matrix

$$\mathbf{P} = \mathbf{K}[\mathbf{R}_{cw} \quad \mathbf{t}_{cw}^c]$$

that maps

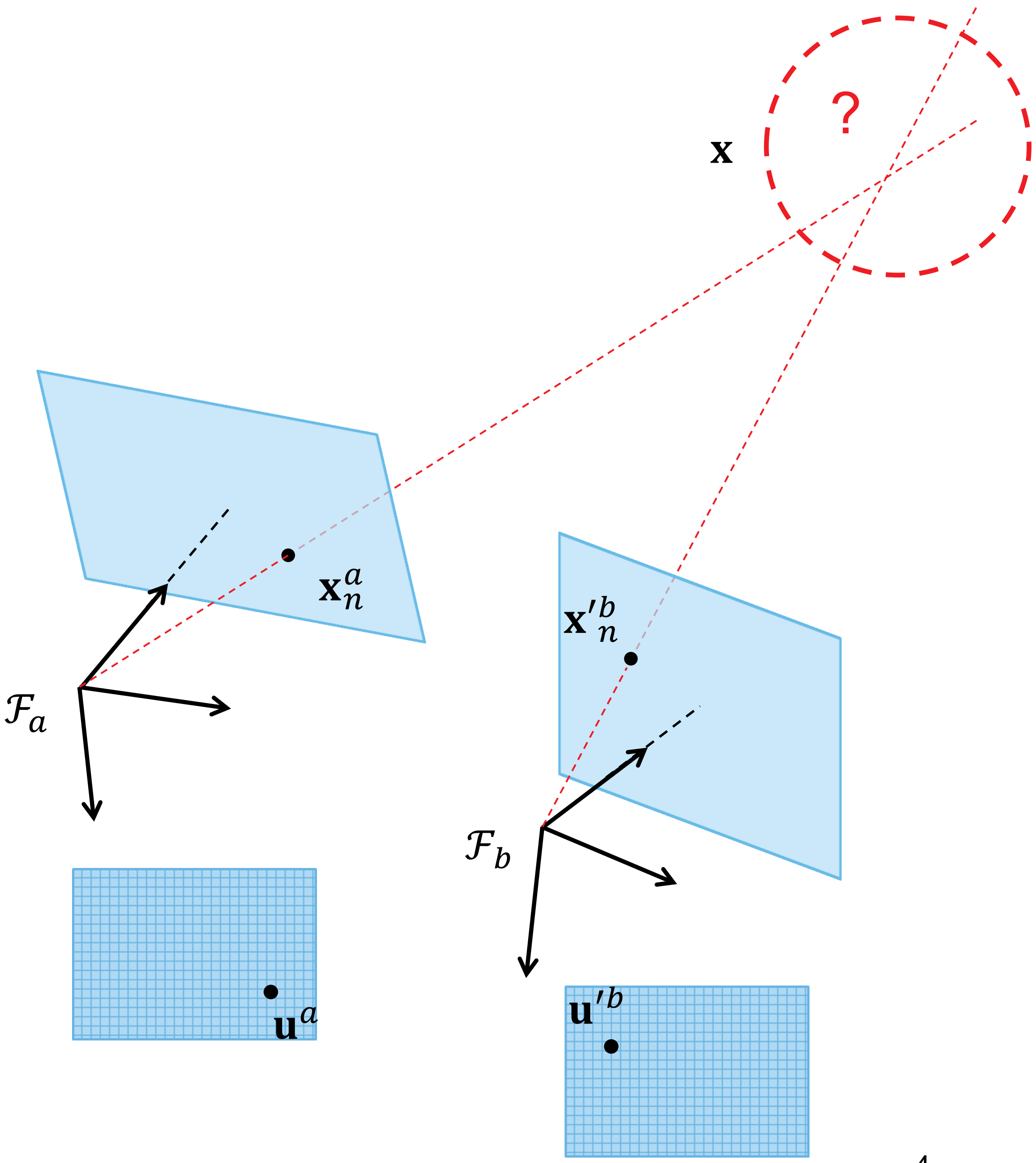
$$\mathbf{P}\tilde{\mathbf{x}}^w = \tilde{\mathbf{u}}$$

Introduction

In order to estimate the 3D point $\mathbf{x}$ it is tempting to back-project the two image points and determine where these rays intersect

However, due to inevitable errors in the positions of $\mathbf{u}^a$ and $\mathbf{u}'^b$, the two rays will not have a common point

So we need to estimate a best solution to the problem

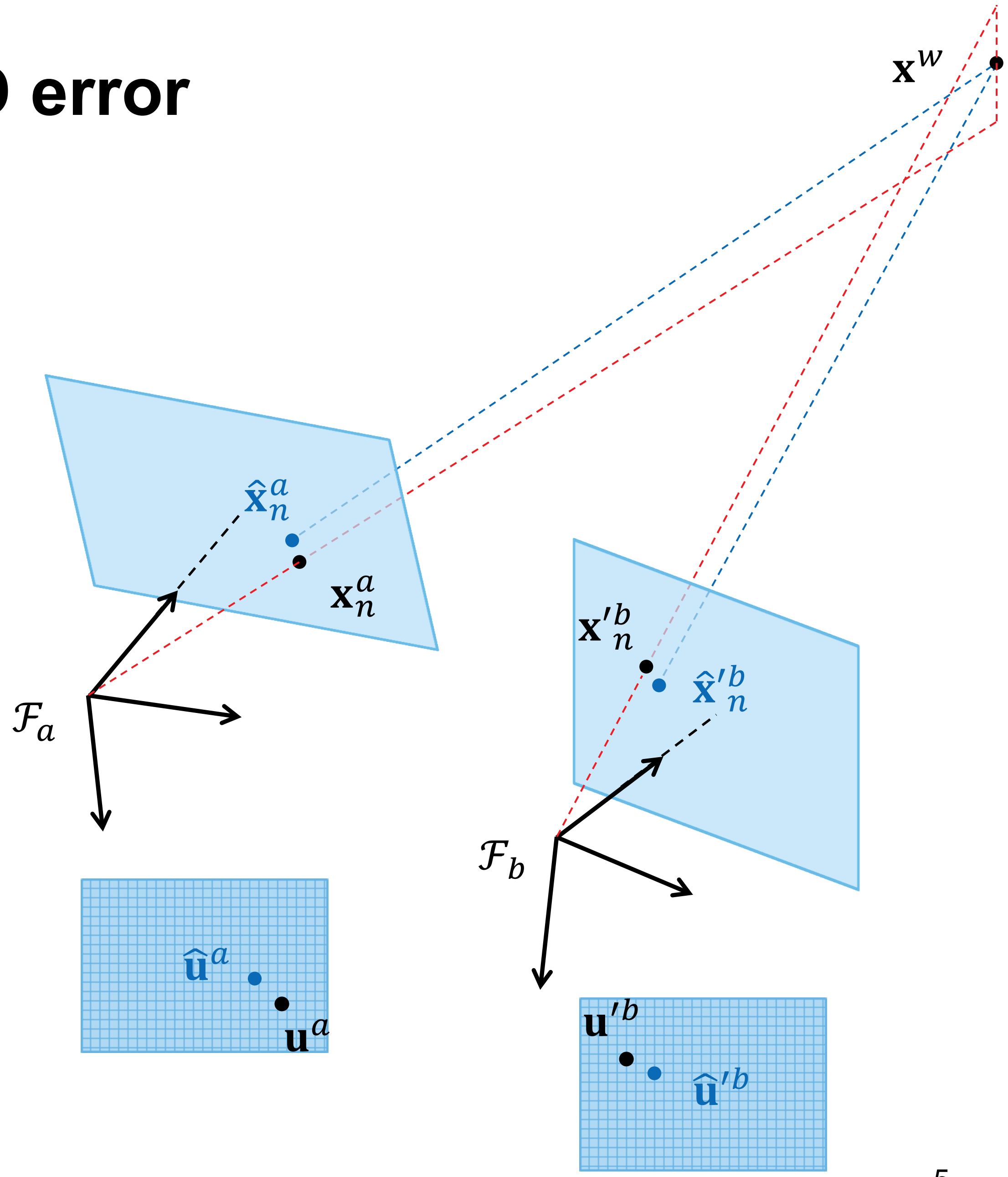


Triangulation by minimizing the 3D error

One intuitive estimate for $\mathbf{x}$ is the midpoint on the shortest line between the two back-projected rays

This estimate minimize the 3D error, but it is not recommended

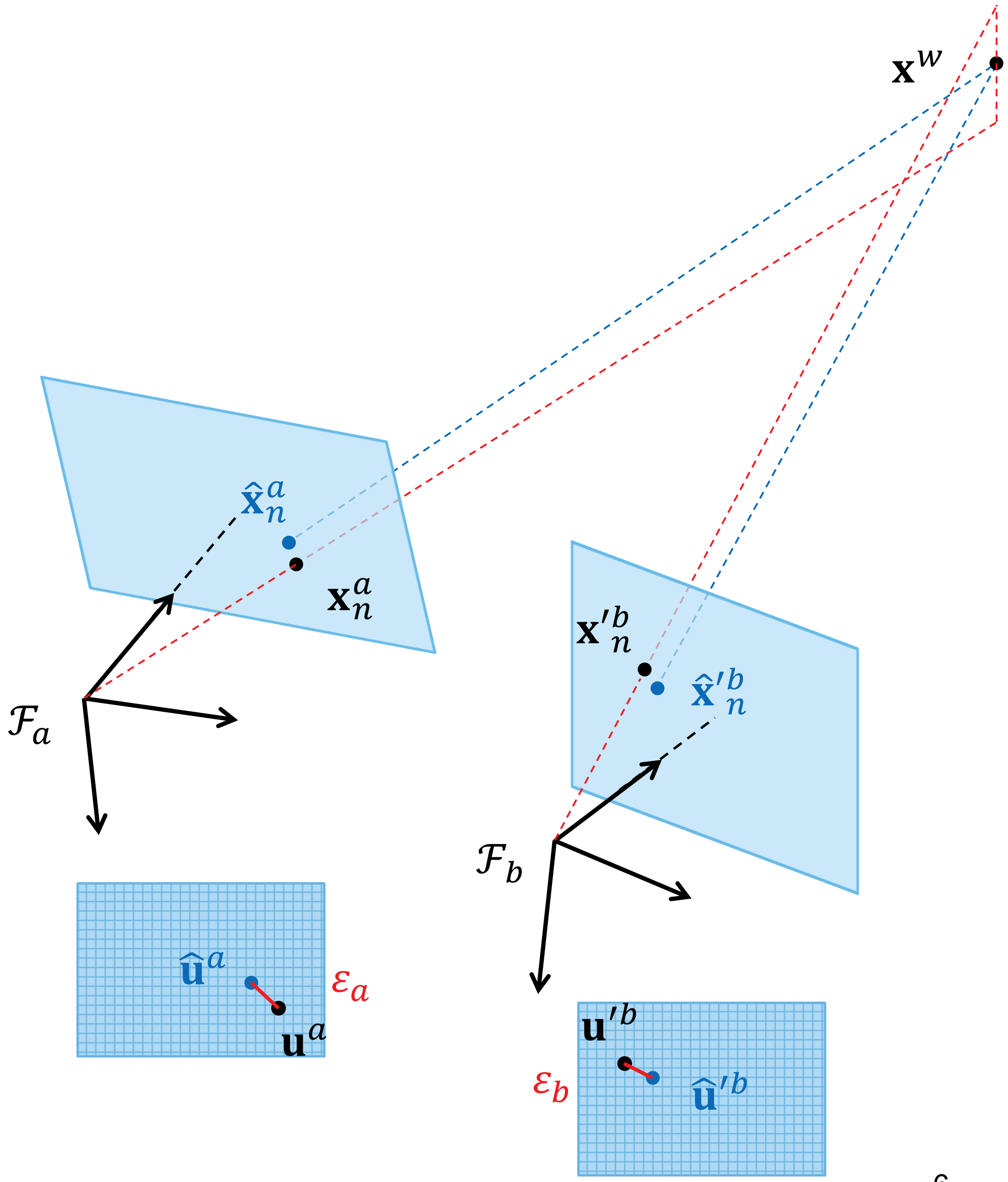
- We "measure" the positions of $\mathbf{u}^a$ and $\mathbf{u}'^b$ in the images, so this is where the errors are
- Depending on the actual position of $\mathbf{x}$, a small perturbation in $\mathbf{u}^a$ and $\mathbf{u}'^b$ can move the 3D midpoint by nothing at all, infinitely much or anything in between



Reprojection error

A much better choice is to minimize the **reprojection error**

$$\varepsilon = \varepsilon_a{}^2 + \varepsilon_b{}^2$$
$$= \left\| \pi_a\left(\mathbf{T}_{aw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^a \right\|^2 + \left\| \pi_b\left(\mathbf{T}_{bw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^b \right\|^2$$

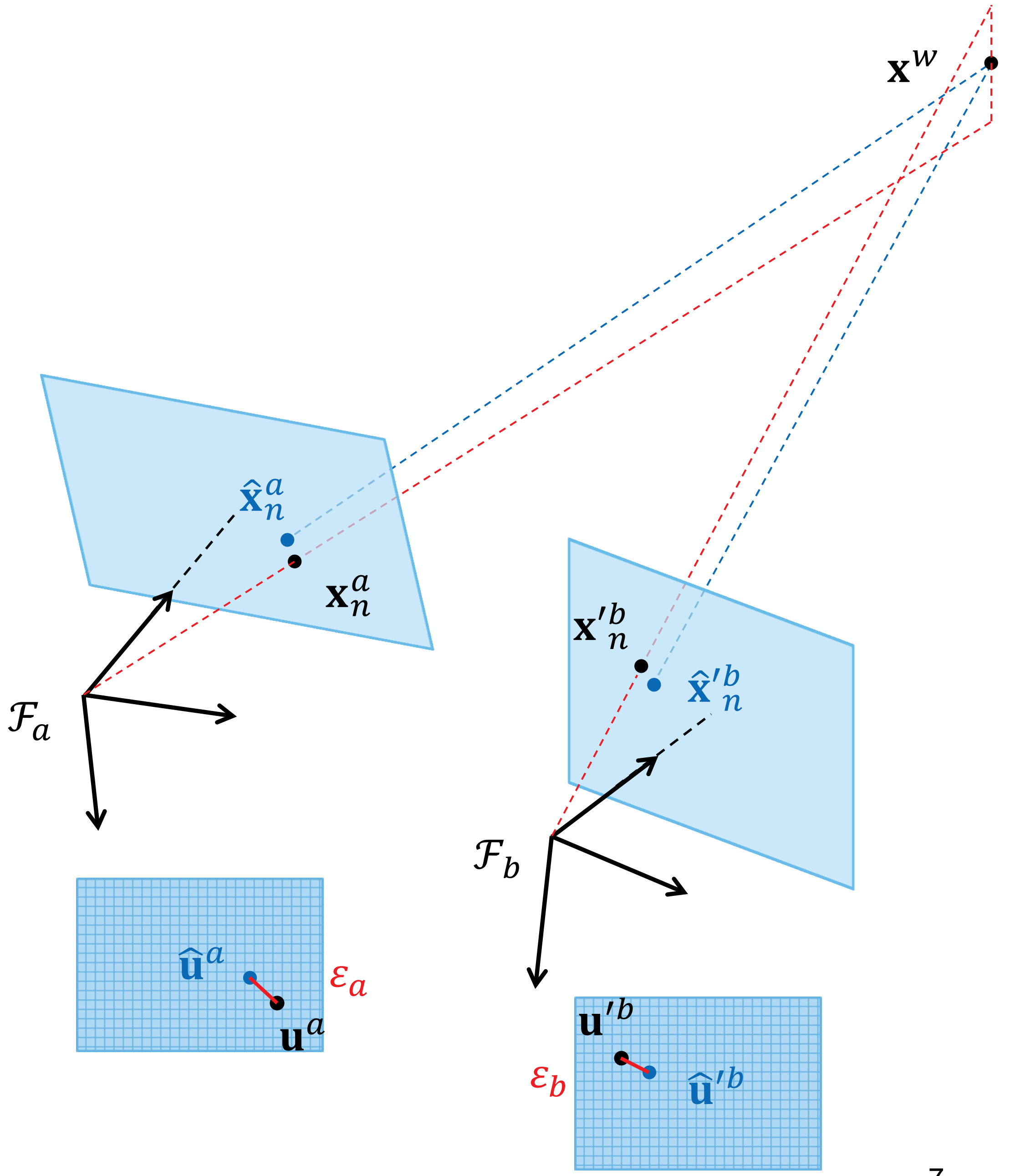


Reprojection error

A much better choice is to minimize the **reprojection error**

$$\varepsilon = \varepsilon_a{}^2 + \varepsilon_b{}^2$$
$$= \left\| \pi_a\left(\mathbf{T}_{aw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^a \right\|^2 + \left\| \pi_b\left(\mathbf{T}_{bw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^b \right\|^2$$

But this is a non-linear optimization problem, which needs an initial estimate…



Linear triangulation by minimizing the algebraic error

Assume that we know the camera projection matrices $\mathbf{P}_a$, $\mathbf{P}_b$ and a 2D correspondence $\mathbf{u}^a \leftrightarrow \mathbf{u}'^b$ for a 3D point $\mathbf{x}$

Each perspective camera model gives rise to two equations on the three entries of $\mathbf{x}$

$$\tilde{\mathbf{u}}^a = \mathbf{P}_a \tilde{\mathbf{x}}$$

$$\begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_a^{1T} \\ \mathbf{p}_a^{2T} \\ \mathbf{p}_a^{3T} \end{bmatrix} \tilde{\mathbf{x}}$$

$$\Downarrow$$

$$\begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_a^{1T} \\ \mathbf{p}_a^{2T} \\ \mathbf{p}_a^{3T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

$$\begin{bmatrix} v^a \mathbf{p}_a^{3T} - \mathbf{p}_a^{2T} \\ \mathbf{p}_a^{1T} - u^a \mathbf{p}_a^{3T} \\ u^a \mathbf{p}_a^{2T} - v^a \mathbf{p}_a^{1T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

$$\Updownarrow$$

$$\begin{bmatrix} v^a \mathbf{p}_a^{3T} - \mathbf{p}_a^{2T} \\ \mathbf{p}_a^{1T} - u^a \mathbf{p}_a^{3T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

$$\tilde{\mathbf{u}}'^b = \mathbf{P}_b \tilde{\mathbf{x}}$$

$$\begin{bmatrix} u'^b \\ v'^b \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_b^{1T} \\ \mathbf{p}_b^{2T} \\ \mathbf{p}_b^{3T} \end{bmatrix} \tilde{\mathbf{x}}$$

$$\Downarrow$$

$$\begin{bmatrix} u'^b \\ v'^b \\ 1 \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_b^{1T} \\ \mathbf{p}_b^{2T} \\ \mathbf{p}_b^{3T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

$$\begin{bmatrix} v'^b \mathbf{p}_b^{3T} - \mathbf{p}_b^{2T} \\ \mathbf{p}_b^{1T} - u'^b \mathbf{p}_b^{3T} \\ u'^b \mathbf{p}_b^{2T} - v'^b \mathbf{p}_b^{1T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

$$\Updownarrow$$

$$\begin{bmatrix} v'^b \mathbf{p}_b^{3T} - \mathbf{p}_b^{2T} \\ \mathbf{p}_b^{1T} - u'^b \mathbf{p}_b^{3T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

Linear triangulation by minimizing the algebraic error

Assume that we know the camera projection matrices $\mathbf{P}_a$, $\mathbf{P}_b$ and a 2D correspondence $\mathbf{u}^a \leftrightarrow \mathbf{u}'^b$ for a 3D point $\mathbf{x}$

$$\begin{bmatrix} v^a \mathbf{p}_a^{3T} - \mathbf{p}_a^{2T} \\ \mathbf{p}_a^{1T} - u^a \mathbf{p}_a^{3T} \\ v'^b \mathbf{p}_b^{3T} - \mathbf{p}_b^{2T} \\ \mathbf{p}_b^{1T} - u'^b \mathbf{p}_b^{3T} \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

Each perspective camera model gives rise to two equations on the three entries of $\mathbf{x}$

$$\mathbf{A}\tilde{\mathbf{x}} = \mathbf{0}$$

Combining these equations gives us an overdetermined homogenous system of linear equations that we can solve with SVD to find the 3D point $\mathbf{x}$ that minimize the **algebraic error**

$$\varepsilon = \left\| \mathbf{A}\tilde{\mathbf{x}} \right\|$$

in a linear least squares sense

Linear triangulation by minimizing the algebraic error

Assume that we know the camera projection matrices $\mathbf{P}_a$, $\mathbf{P}_b$ and a 2D correspondence $\mathbf{u}^a \leftrightarrow \mathbf{u}'^b$ for a 3D point $\mathbf{x}$

Each perspective camera model gives rise to two equations on the three entries of $\mathbf{x}$

Combining these equations gives us an overdetermined homogenous system of linear equations that we can solve with SVD to find the 3D point $\mathbf{x}$ that minimize the **algebraic error**

$$\varepsilon = \left\| \mathbf{A}\tilde{\mathbf{x}} \right\|$$

in a linear least squares sense

$$\begin{bmatrix} v^a \mathbf{p}_a^{3T} - \mathbf{p}_a^{2T} \\ \mathbf{p}_a^{1T} - u^a \mathbf{p}_a^{3T} \\ v'^b \mathbf{p}_b^{3T} - \mathbf{p}_b^{2T} \\ \mathbf{p}_b^{1T} - u'^b \mathbf{p}_b^{3T} \\ \vdots \end{bmatrix} \tilde{\mathbf{x}} = \mathbf{0}$$

$$\mathbf{A}\tilde{\mathbf{x}} = \mathbf{0}$$

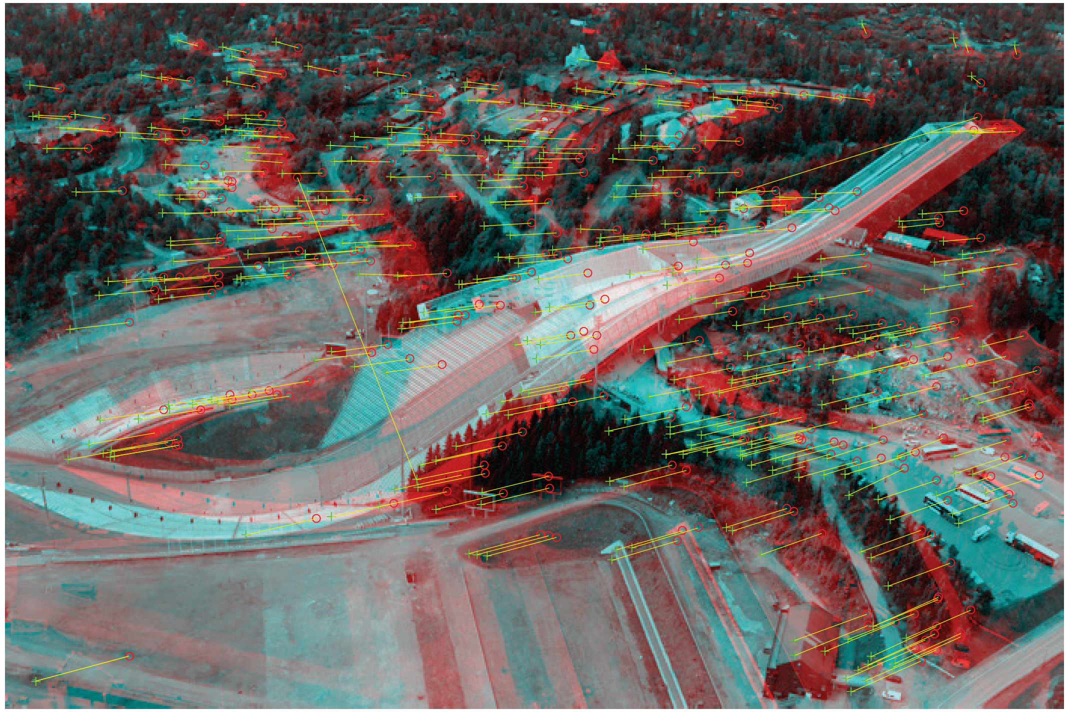
The algebraic error is not geometrically meaningful, but this approach generalizes naturally to the case when $\mathbf{x}$ is observed by more than two cameras

Just construct $\mathbf{A}$ with two rows per camera

Example

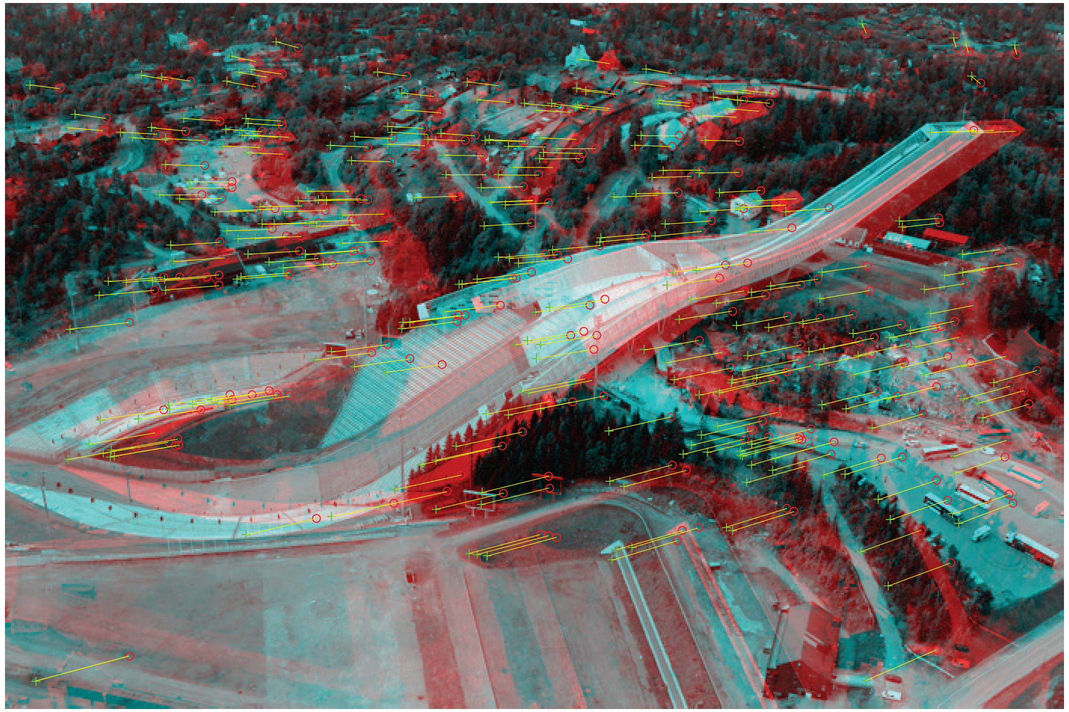
- Two views with known relative pose

Example



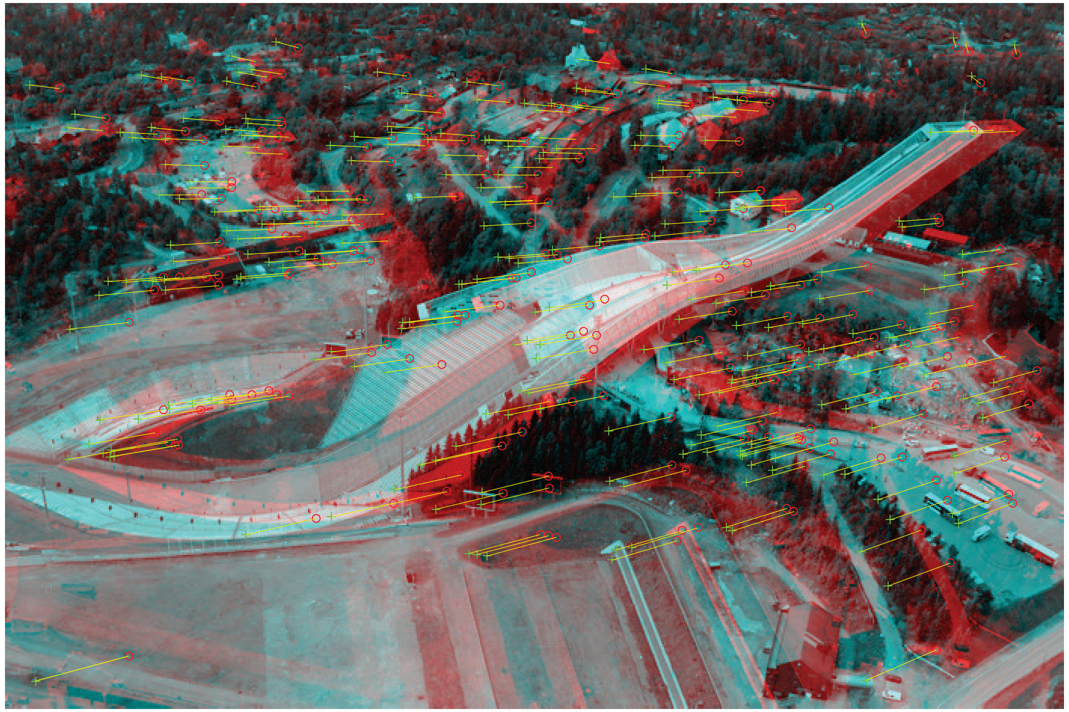
- Two views with known relative pose
- Matching feature points

Example

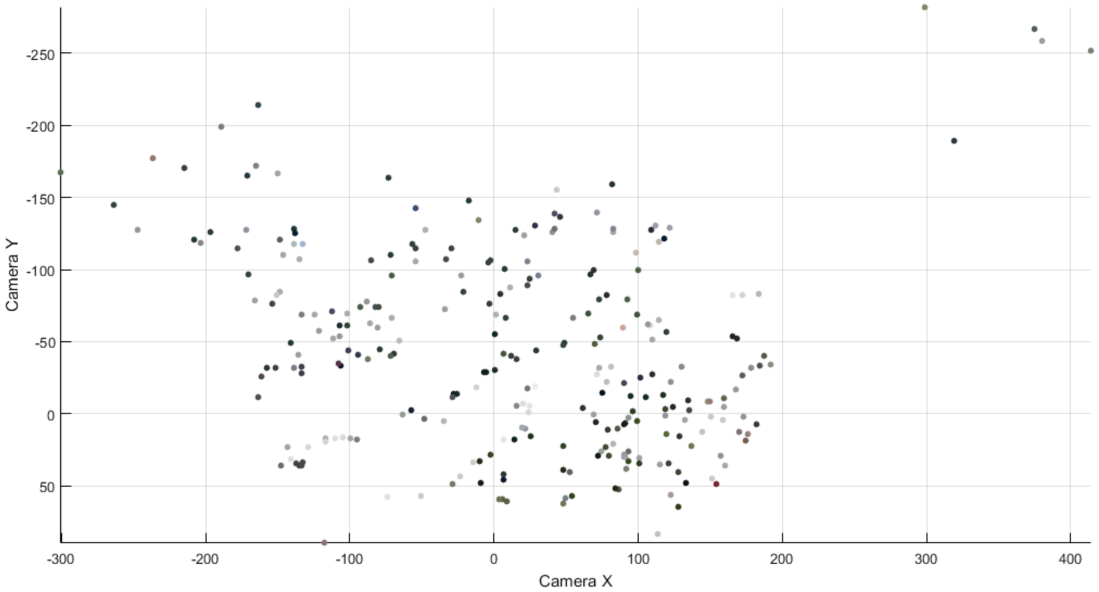


- Two views with known relative pose
- Matching feature points
- After filtering by the epipolar constraint
  - Keeping matches that are within ± 0.5 pixels of the epipolar line

Example

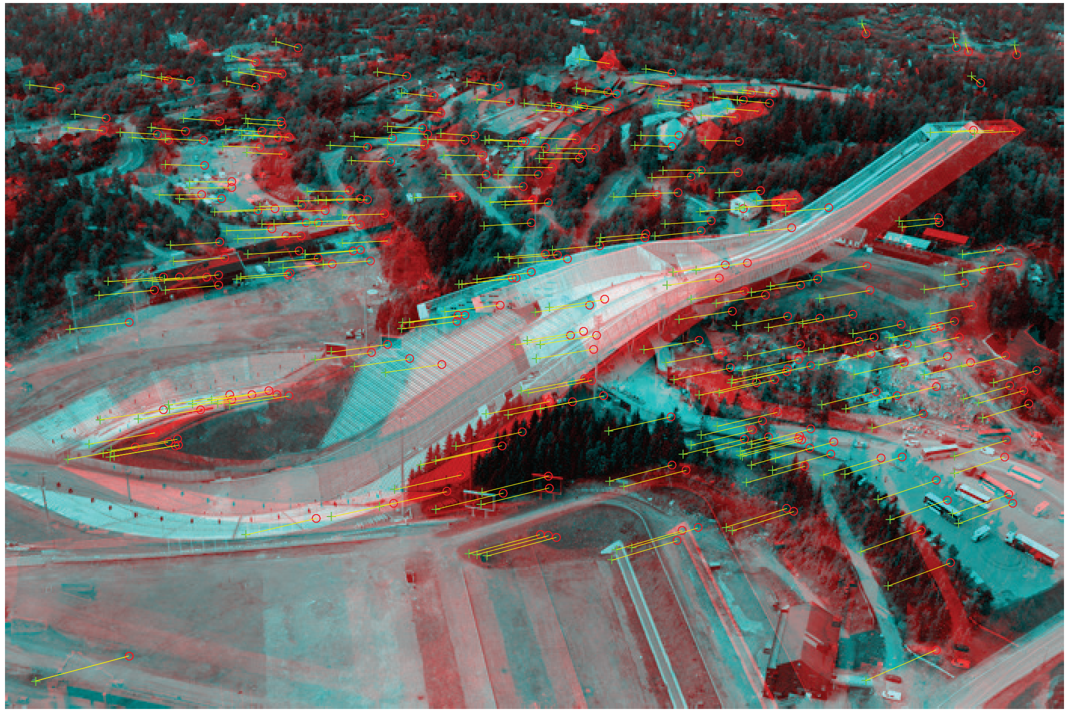


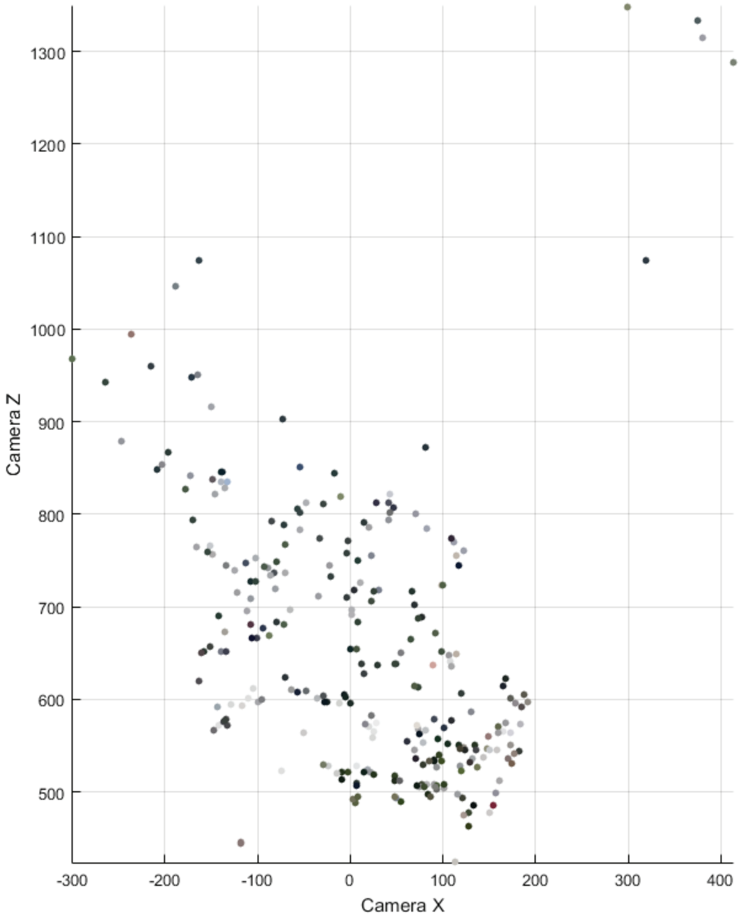
- Two views with known relative pose
- Matching feature points
- After filtering by the epipolar constraint
  - Keeping matches that are within ± 0.5 pixels of the epipolar line



- Sparse 3D reconstruction of the scene by triangulation

Example

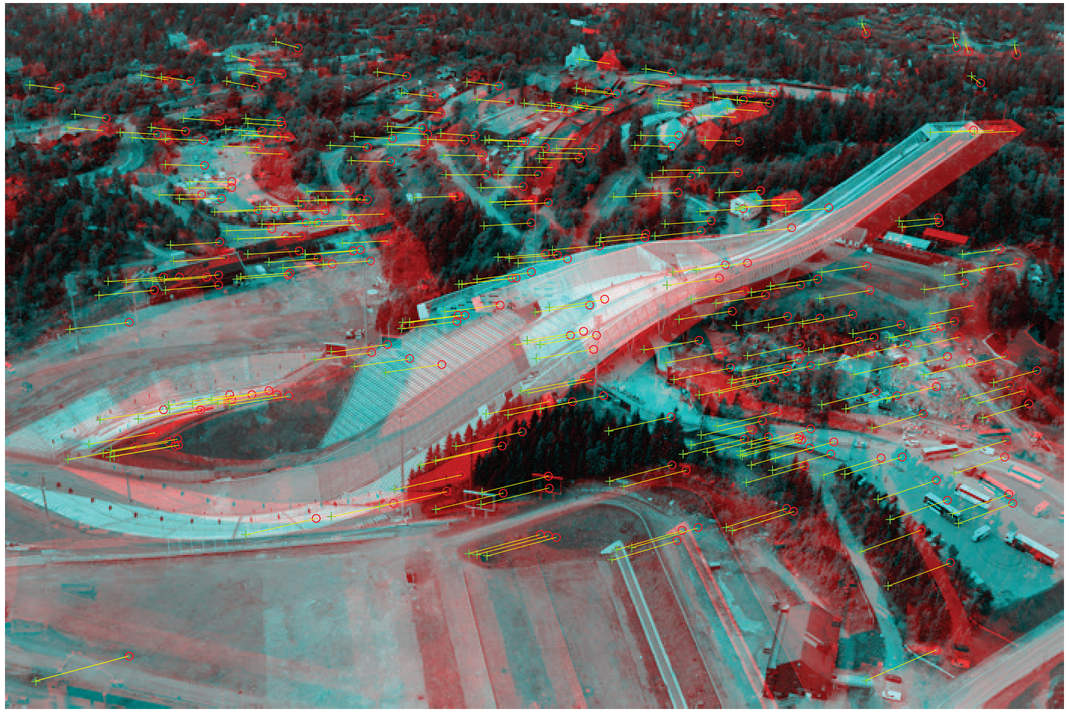


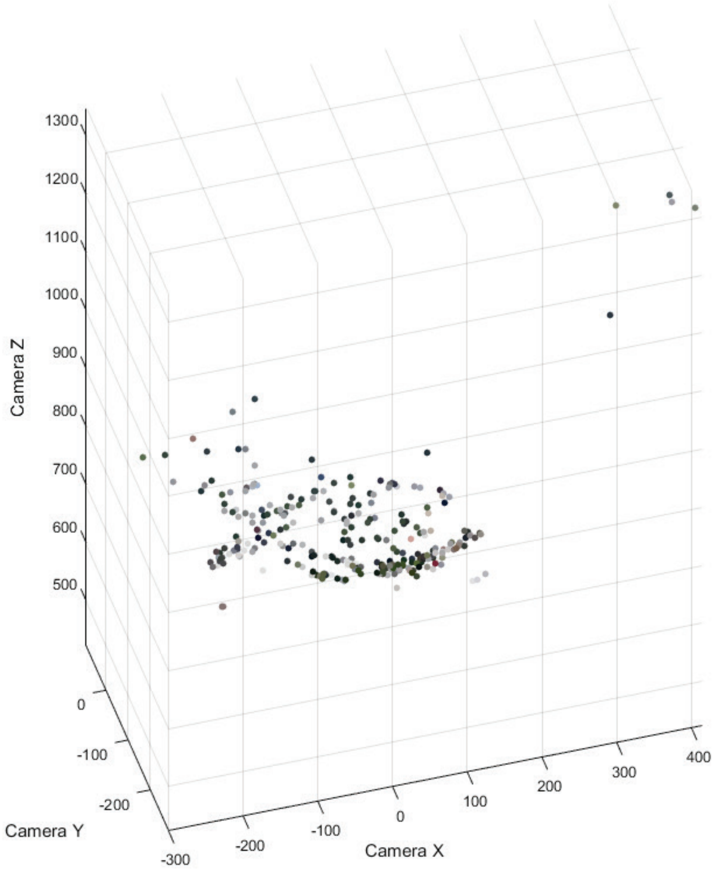


- Two views with known relative pose
- Matching feature points
- After filtering by the epipolar constraint
  - Keeping matches that are within ± 0.5 pixels of the epipolar line

- Sparse 3D reconstruction of the scene by triangulation

Example





- Two views with known relative pose
- Matching feature points
- After filtering by the epipolar constraint
  - Keeping matches that are within ± 0.5 pixels of the epipolar line

- Sparse 3D reconstruction of the scene by triangulation

Beyond linear triangulation

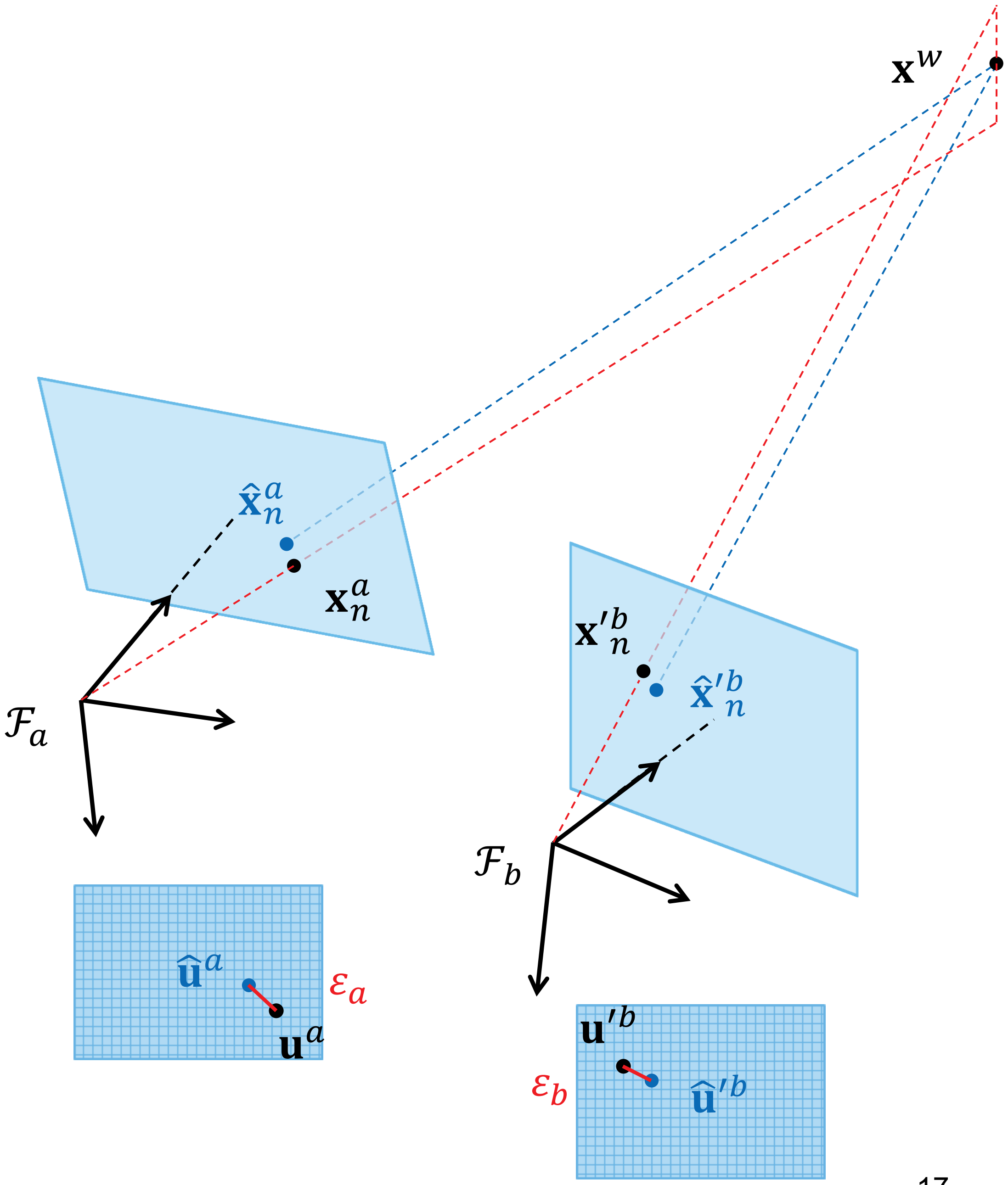
- Linear triangulation methods will typically provide decent 3D estimates that can be used as the starting point for iterative non-linear estimation methods

- To improve the estimate, one can minimize the reprojection error

$$\varepsilon = \left\| \pi_a\left(\mathbf{T}_{aw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^a \right\|^2 + \left\| \pi_b\left(\mathbf{T}_{bw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^b \right\|^2$$

  in an iterative method

  This is sometimes called **structure-only bundle adjustment**



Summary

Triangulation

Estimate a 3D point $\mathbf{x}$ for a correspondence $\mathbf{u}^a \leftrightarrow \mathbf{u}'^b$ assuming error free camera projection matrices $\mathbf{P}_a$ and $\mathbf{P}_b$

Minimal 3D error

3D midpoint, not recommended!

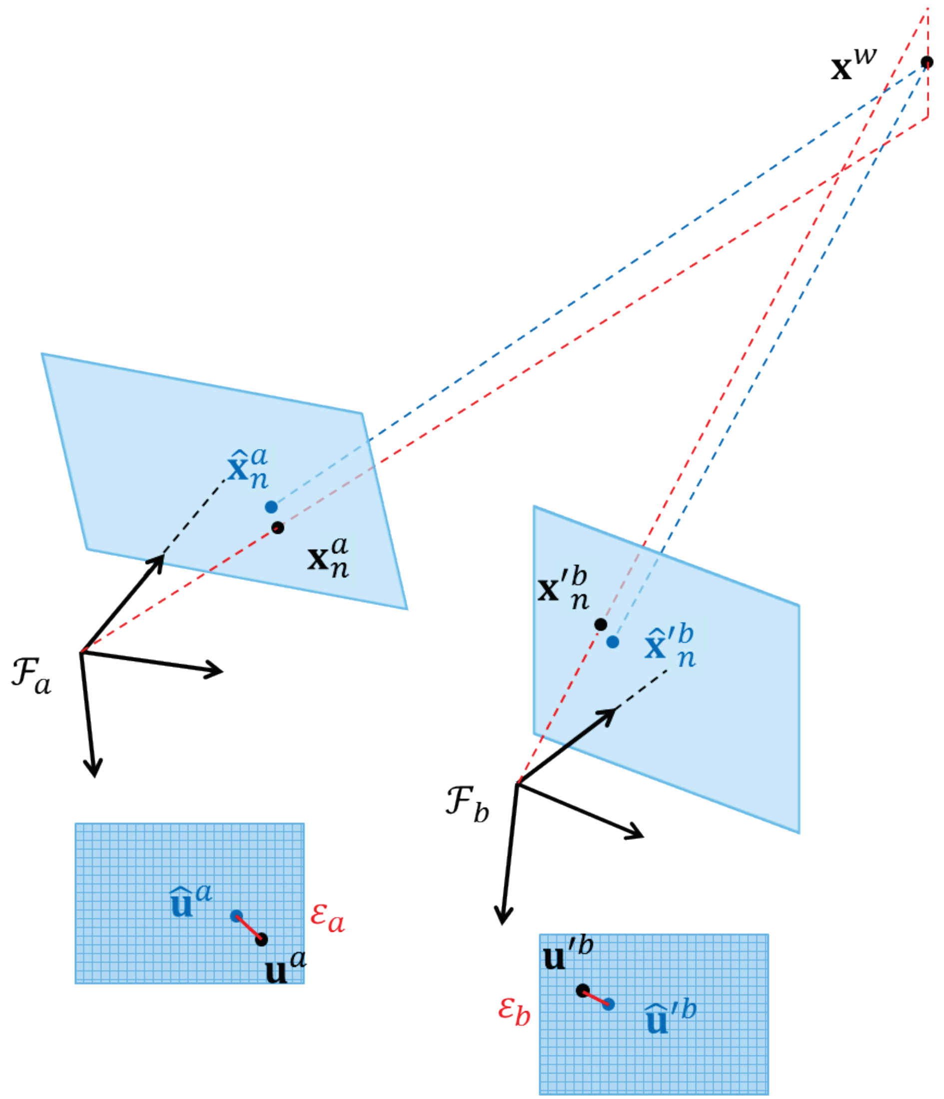
Minimal algebraic error

$$\begin{aligned}\tilde{\mathbf{u}}^a &= \mathbf{P}_a\tilde{\mathbf{x}}^w \\ \tilde{\mathbf{u}}^b &= \mathbf{P}_b\tilde{\mathbf{x}}^w\end{aligned} \longrightarrow \mathbf{A}\tilde{\mathbf{x}} = 0 \xrightarrow{\text{SVD}} \mathbf{x}$$

$$\varepsilon = \|\mathbf{A}\tilde{\mathbf{x}}\|$$

Minimal reprojection error

$$\varepsilon = \left\|\pi_a\left(\mathbf{T}_{aw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^a\right\|^2 + \left\|\pi_b\left(\mathbf{T}_{bw}\tilde{\mathbf{x}}^w\right) - \mathbf{u}^b\right\|^2$$



Further reading

- Online book by Richard Szeliski – Computer Vision: Algorithms and Applications
  http://szeliski.org/Book/drafts/SzeliskiBook_20100903_draft.pdf
  - Chapter 7.1 covers triangulation

- R. I. Hartley and P. Sturm, *Triangulation,* 1997
  https://users.cecs.anu.edu.au/~hartley/Papers/triangulation/triangulation.pdf