Estimation of Essential Matrix, Nonlinear

Computer Vision II

CSE 252B

Lecture 15

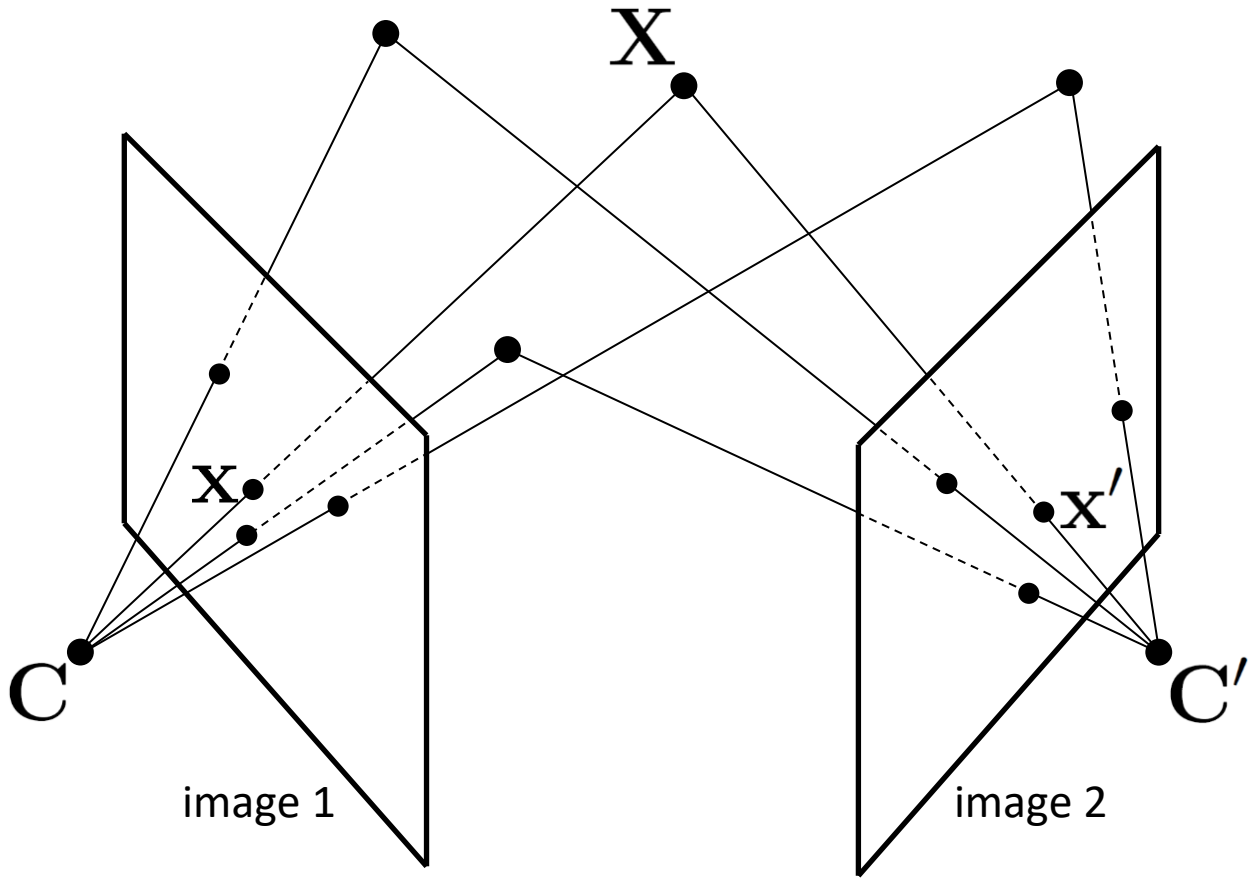
Announcements

- Assignment 4 is due Mar 4, 11:59 PM
- Assignment 5 will be released Mar 4
  - Due Mar 18, 11:59 PM

Two view geometry

Two view geometry

- Imaging a 3D scene



Calibrated camera

$$\mathtt{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

where

$\alpha_x = f m_x$	Focal length in x direction in terms of pixel dimensions
$\alpha_y = f m_y$	Focal length in y direction in terms of pixel dimensions
s	Skew
$(x_0, y_0)^\top = (p_x m_x, p_y m_y)^\top$	Coordinates of principal point in terms of pixel dimensions

where m_x and m_y are number of pixels per unit disance in x and y directions, respectively

(and lens distortion parameters)

- If camera calibration parameters are known, then use normalized camera projection matrix and image points in normalized coordinates

Normalized camera projection matrix

$$\hat{\mathbf{x}} = \hat{\mathrm{P}}\mathbf{X}$$

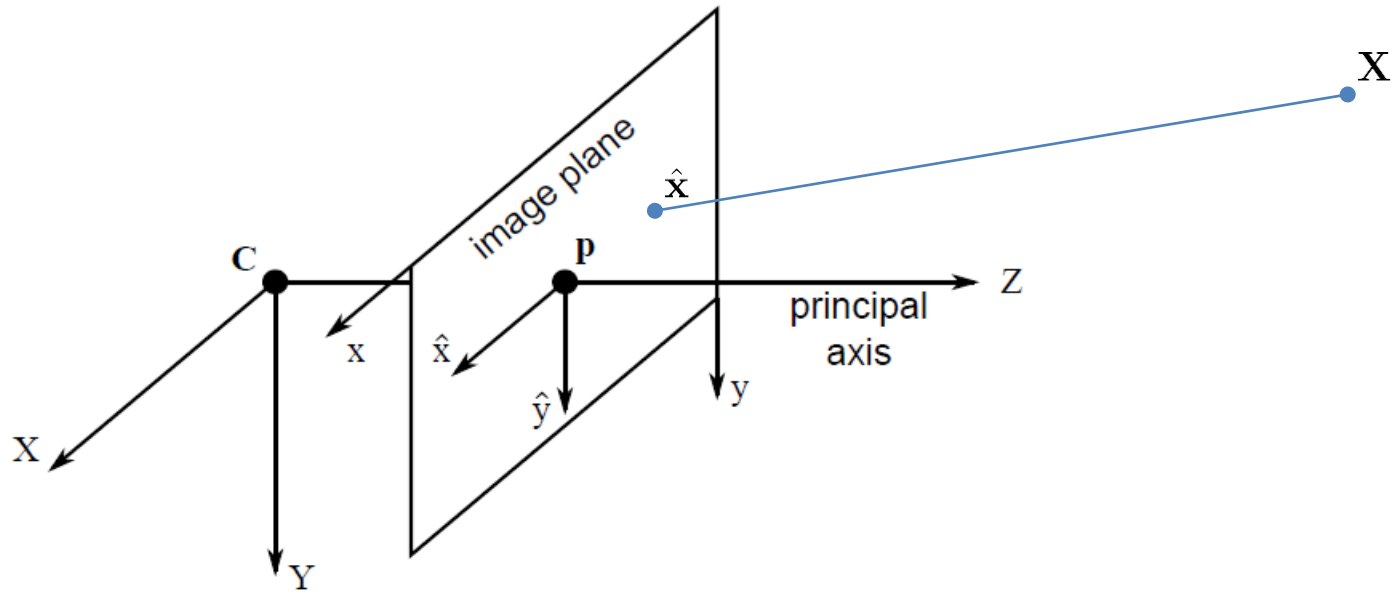
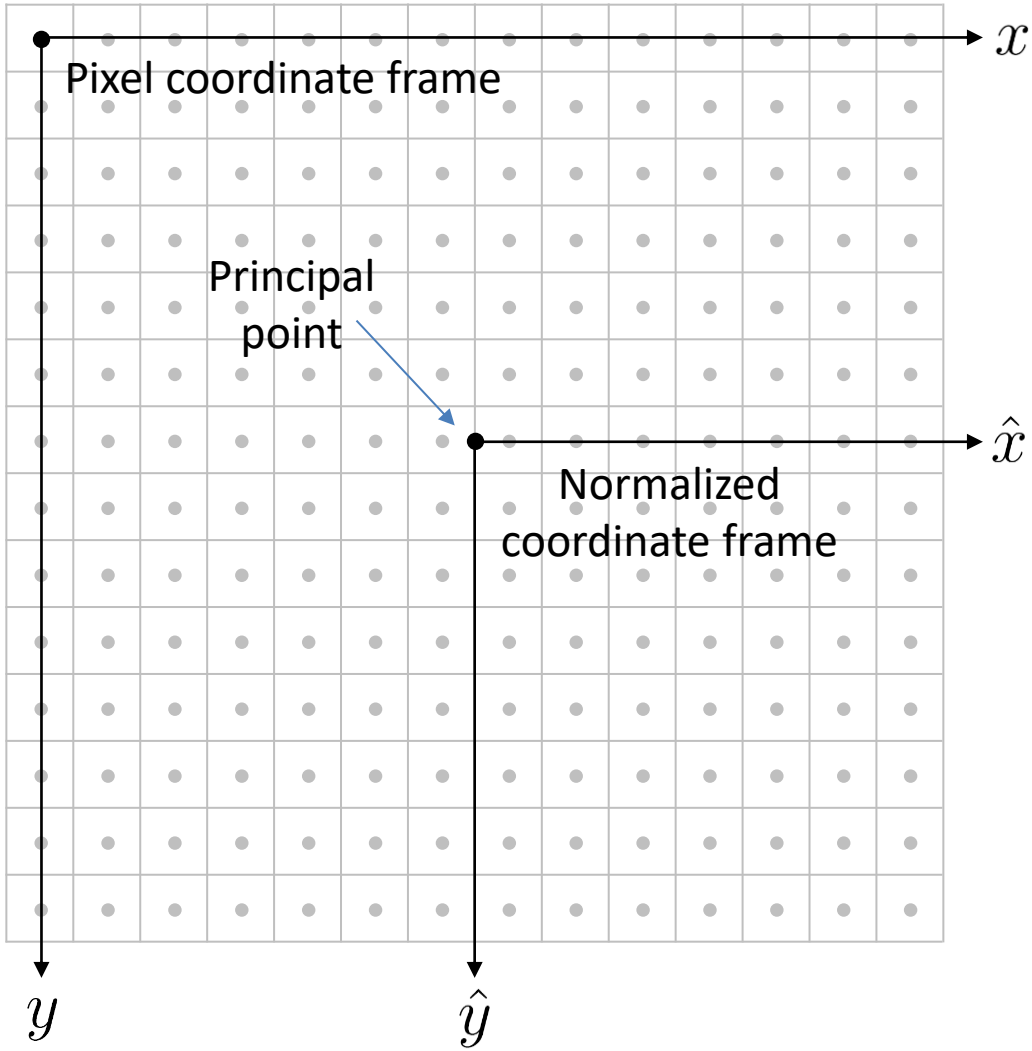


Image coordinate frames



Camera projection

Project 3D point in world coordinate frame $\mathbf{X}$ under camera projection matrix $\mathtt{P} = \mathtt{K}[\mathtt{R} \,|\, \mathbf{t}]$

$$\begin{aligned} \mathbf{x} &= \mathtt{P}\mathbf{X} \\ \mathbf{x} &= \mathtt{K}[\mathtt{R} \,|\, \mathbf{t}]\mathbf{X} \\ \mathtt{K}^{-1}\mathbf{x} &= [\mathtt{R} \,|\, \mathbf{t}]\mathbf{X} \\ \hat{\mathbf{x}} &= \hat{\mathtt{P}}\mathbf{X} \end{aligned}$$

Camera calibration matrix

where

Camera projection matrix	$\mathtt{P} = \mathtt{K}[\mathtt{R} \,|\, \mathbf{t}]$	$= \mathtt{K}\hat{\mathtt{P}}$
Normalized camera projection matrix	$\hat{\mathtt{P}} = [\mathtt{R} \,|\, \mathbf{t}]$	$= \mathtt{K}^{-1}\mathtt{P}$
Image point in pixel coordinates	$\mathbf{x} = \mathtt{K}[\mathtt{R} \,|\, \mathbf{t}]\mathbf{X}$	$= \mathtt{K}\hat{\mathbf{x}}$
Image point in normalized coordinates	$\hat{\mathbf{x}} = [\mathtt{R} \,|\, \mathbf{t}]\mathbf{X}$	$= \mathtt{K}^{-1}\mathbf{x}$

Imaging geometry models

- Single view
  - Uncalibrated: camera projection matrix
  - Calibrated: normalized camera projection matrix
- Two views
  - Rotation about the same camera center
    - Uncalibrated: 2D projective transformation matrix
    - Calibrated: 3D rotation matrix
  - Imaging a plane
    - Uncalibrated: 2D projective transformation matrix
    - Calibrated: 2D projective transformation matrix
  - Imaging a 3D scene
    - Uncalibrated: fundamental matrix
    - Calibrated: essential matrix

Essential matrix

$$\hat{\boldsymbol{\ell}}' = \mathtt{E}\hat{\mathbf{x}} \text{ and } \hat{\mathbf{x}}'^{\top}\hat{\boldsymbol{\ell}}' = 0$$

$$\hat{\mathbf{x}}'^{\top}\mathtt{E}\hat{\mathbf{x}} = 0 \qquad \text{Epipolar constraint}$$

$\hat{\boldsymbol{\ell}}' = \mathtt{E}\hat{\mathbf{x}}$	The epipolar line corresponding to the point $\hat{\mathbf{x}}$
$\hat{\boldsymbol{\ell}} = \mathtt{E}^{\top}\hat{\mathbf{x}}'$	The epipolar line corresponding to the point $\hat{\mathbf{x}}'$
$\mathtt{E}\hat{\mathbf{e}} = \mathbf{0}$	The epipole $\hat{\mathbf{e}}$ is the null space of $\mathtt{E}$
$\hat{\mathbf{e}}'^{\top}\mathtt{E} = \mathbf{0}^{\top}$	The epipole $\hat{\mathbf{e}}'$ is the left null space of $\mathtt{E}$
$\mathtt{E}^{\top}\hat{\mathbf{e}}' = \mathbf{0}$	The epipole $\hat{\mathbf{e}}'$ is the null space of $\mathtt{E}^{\top}$

The essential matrix $\mathtt{E}$ from general normalized cameras $\hat{\mathtt{P}} = [\mathtt{R} \,|\, \mathbf{t}]$ and $\hat{\mathtt{P}}' = [\mathtt{R}' \,|\, \mathbf{t}']$
Transform the cameras such that first normalized camera projection matrix is $[\mathtt{I} \,|\, \mathbf{0}]$ as follows

$$[\mathtt{R} \,|\, \mathbf{t}]\mathtt{H}_{\mathtt{E}}^{-1} = [\mathtt{I} \,|\, \mathbf{0}] \text{ and } [\mathtt{R}' \,|\, \mathbf{t}']\mathtt{H}_{\mathtt{E}}^{-1} = [\mathtt{R}'\mathtt{R}^{\top} \,|\, \mathbf{t}' - \mathtt{R}'\mathtt{R}^{\top}\mathbf{t}], \text{ where } \mathtt{H}_{\mathtt{E}}^{-1} = \begin{bmatrix} \mathtt{R}^{\top} & -\mathtt{R}^{\top}\mathbf{t} \\ \mathbf{0}^{\top} & 1 \end{bmatrix}$$

Then

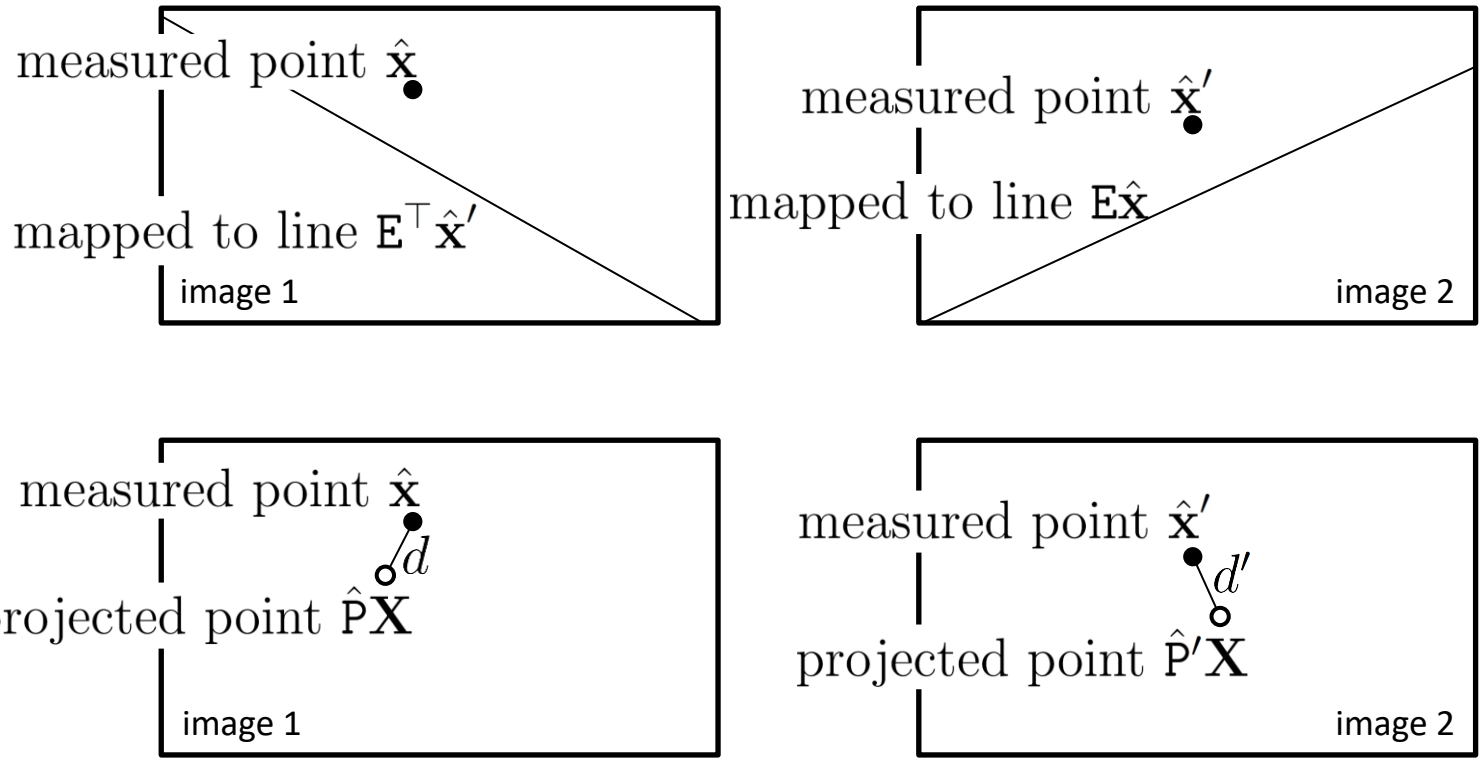
$$\mathtt{E} = [\mathbf{t}' - \mathtt{R}'\mathtt{R}^{\top}\mathbf{t}]_{\times}\mathtt{R}'\mathtt{R}^{\top}$$

The essential matrix $\mathtt{E}$ from canonical cameras $\mathtt{P} = [\mathtt{I} \,|\, \mathbf{0}]$ and $\mathtt{P}' = [\mathtt{R} \,|\, \mathbf{t}]$

$$\mathtt{E} = [\mathbf{t}]_{\times}\mathtt{R}$$

Essential matrix

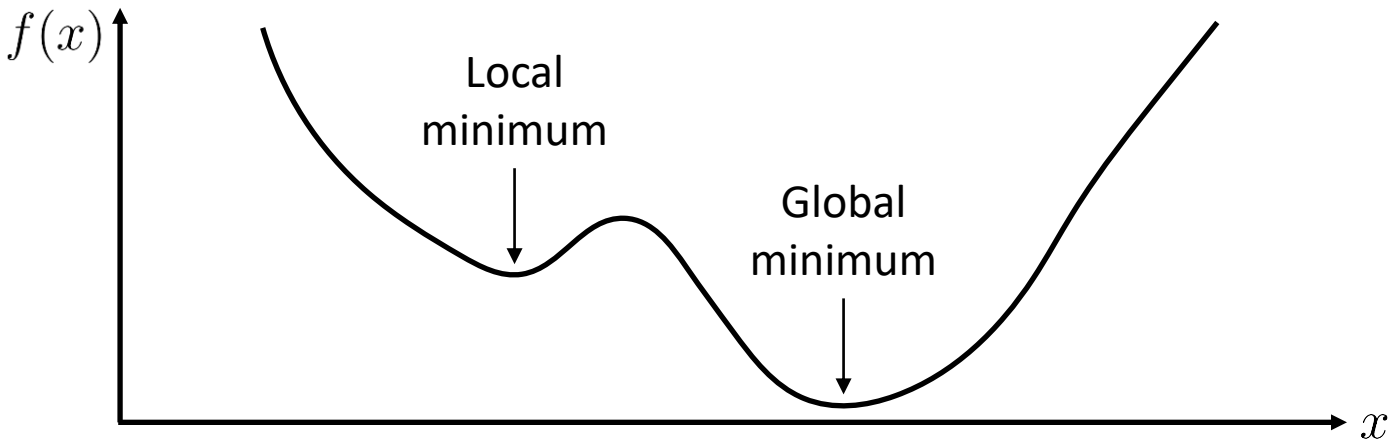
- Error in image point measurements



Estimation of essential matrix

$$\hat{\mathbf{x}}_i = \hat{\mathrm{P}}\mathbf{X}_i \,\forall\, i \text{ and } \hat{\mathbf{x}}'_i = \hat{\mathrm{P}}'\mathbf{X}_i \,\forall\, i, \text{ set } \hat{\mathrm{P}} = [\mathrm{I} \,|\, \mathbf{0}] \text{ and solve for } \hat{\mathrm{P}}'$$

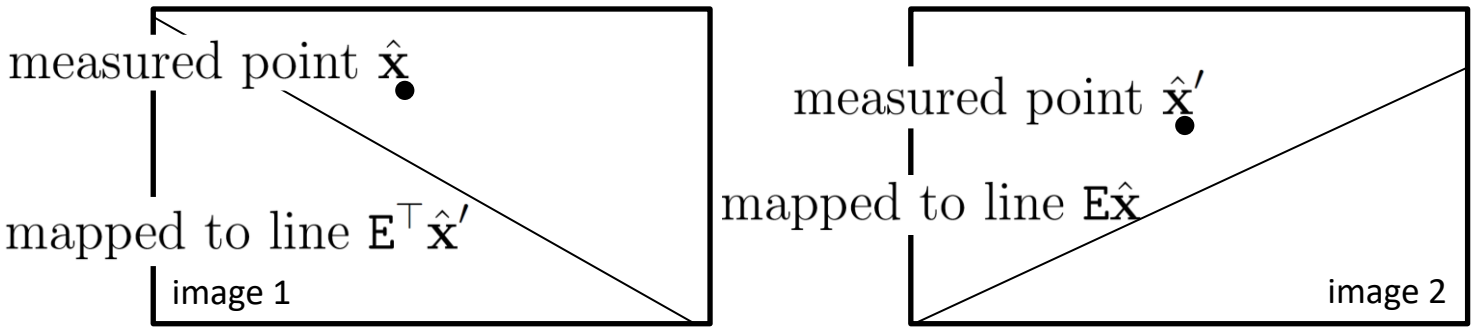
- Given point correspondences $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$
- Minimize geometric error $\sum_i \left(d(\hat{\mathbf{x}}_i, \hat{\mathrm{P}}\mathbf{X}_i)^2 + d(\hat{\mathbf{x}}'_i, \hat{\mathrm{P}}'\mathbf{X}_i)^2 \right)$
- A nonlinear mapping requires a nonlinear optimization problem solver
  - Use linear estimation for initial estimate (to mitigate converging to local optimum)
  - Iterative process to determine global optimum



Linear estimation of essential matrix using the direct linear transformation (DLT) algorithm

- Minimize algebraic error

$$\hat{\mathbf{x}}'^{\top}\mathbf{E}\hat{\mathbf{x}} = 0 \qquad \text{Epipolar constraint}$$



Linear estimation of essential matrix using the direct linear transformation (DLT) algorithm

Given $n \geq 8$ point correspondences $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$

See previous lecture estimation from 7, 6, or 5 point correspondences

$$\hat{\mathbf{x}}_i'^{\top} \mathtt{E} \hat{\mathbf{x}}_i = 0 \, \forall \, i, \text{ solve for } \mathtt{E}$$

$$\left(\hat{\mathbf{x}}_i'^{\top} \otimes \hat{\mathbf{x}}_i^{\top}\right) \mathbf{e} = 0 \, \forall \, i, \text{ solve for } \mathbf{e} = \mathrm{vec}(\mathtt{E}^{\top})$$

$$\begin{bmatrix} \hat{\mathbf{x}}_1'^{\top} \otimes \hat{\mathbf{x}}_1^{\top} \\ \hat{\mathbf{x}}_2'^{\top} \otimes \hat{\mathbf{x}}_2^{\top} \\ \vdots \\ \hat{\mathbf{x}}_n'^{\top} \otimes \hat{\mathbf{x}}_n^{\top} \end{bmatrix} \mathbf{e} = \mathbf{0}$$

$$\mathtt{A}\mathbf{e} = \mathbf{0}, \text{ where } \mathtt{A} = \begin{bmatrix} \hat{\mathbf{x}}_1'^{\top} \otimes \hat{\mathbf{x}}_1^{\top} \\ \hat{\mathbf{x}}_2'^{\top} \otimes \hat{\mathbf{x}}_2^{\top} \\ \vdots \\ \hat{\mathbf{x}}_n'^{\top} \otimes \hat{\mathbf{x}}_n^{\top} \end{bmatrix}$$

Then, enforce constraint (using SVD)

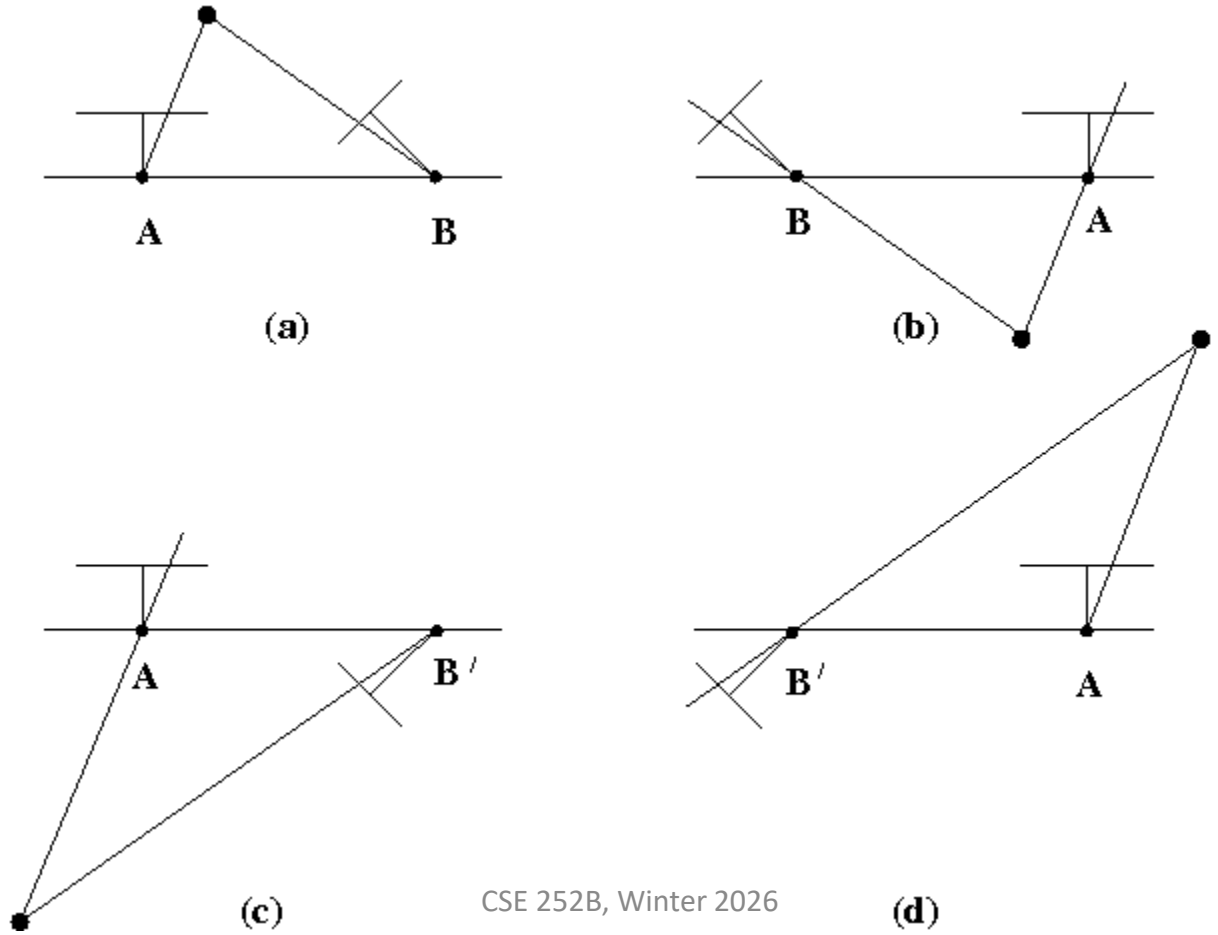
$$\mathtt{E} = \mathtt{U}\Sigma\mathtt{V}^{\top}, \text{ where } \Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \sigma_3)$$

$$\mathtt{E} = \mathtt{U}\Sigma'\mathtt{V}^{\top}, \text{ where } \Sigma' = \mathrm{diag}(1, 1, 0)$$

Decompose essential matrix

$$\hat{\mathtt{P}} = [\mathtt{I} \,|\, \mathbf{0}] \text{ and } \hat{\mathtt{P}}' = [\mathtt{R} \,|\, \mathbf{t}]$$

Four solutions, but only one where the reconstructed scene point is in front of both cameras



Decompose essential matrix

$$\hat{\mathtt{P}} = [\mathtt{I} \,|\, \mathbf{0}] \text{ and } \hat{\mathtt{P}}' = [\mathtt{R} \,|\, \mathbf{t}]$$

Four solutions, but only one where the reconstructed scene point is in front of both cameras

$$\mathtt{E} = \mathtt{U}\mathtt{D}\mathtt{V}^\top \quad \text{singular value decomposition, where } \mathtt{D} = \text{diag}(1, 1, 0) \text{ to scale}$$

Two possible choices of second camera rotation matrix $\mathtt{R}$

$$\mathtt{R}_1 = \mathtt{U}\mathtt{Z}\mathtt{V}^\top \text{ or } \mathtt{R}_2 = \mathtt{U}\mathtt{Z}^\top\mathtt{V}^\top, \text{ where } \mathtt{Z} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Ensure $\mathtt{R}_1$ and $\mathtt{R}_2$ are special orthogonal. If $\det(\mathtt{R}_1) = -1$, then negate $\mathtt{R}_1$. If $\det(\mathtt{R}_2) = -1$, then negate $\mathtt{R}_2$.

Two possible choices of second camera translation vector $\mathbf{t}$ (to positive scale)

$$\mathbf{t}_1 = \mathbf{u}_3 \text{ or } \mathbf{t}_2 = -\mathbf{u}_3, \text{ where } \mathbf{u}_3 \text{ is the last column of } \mathtt{U}$$

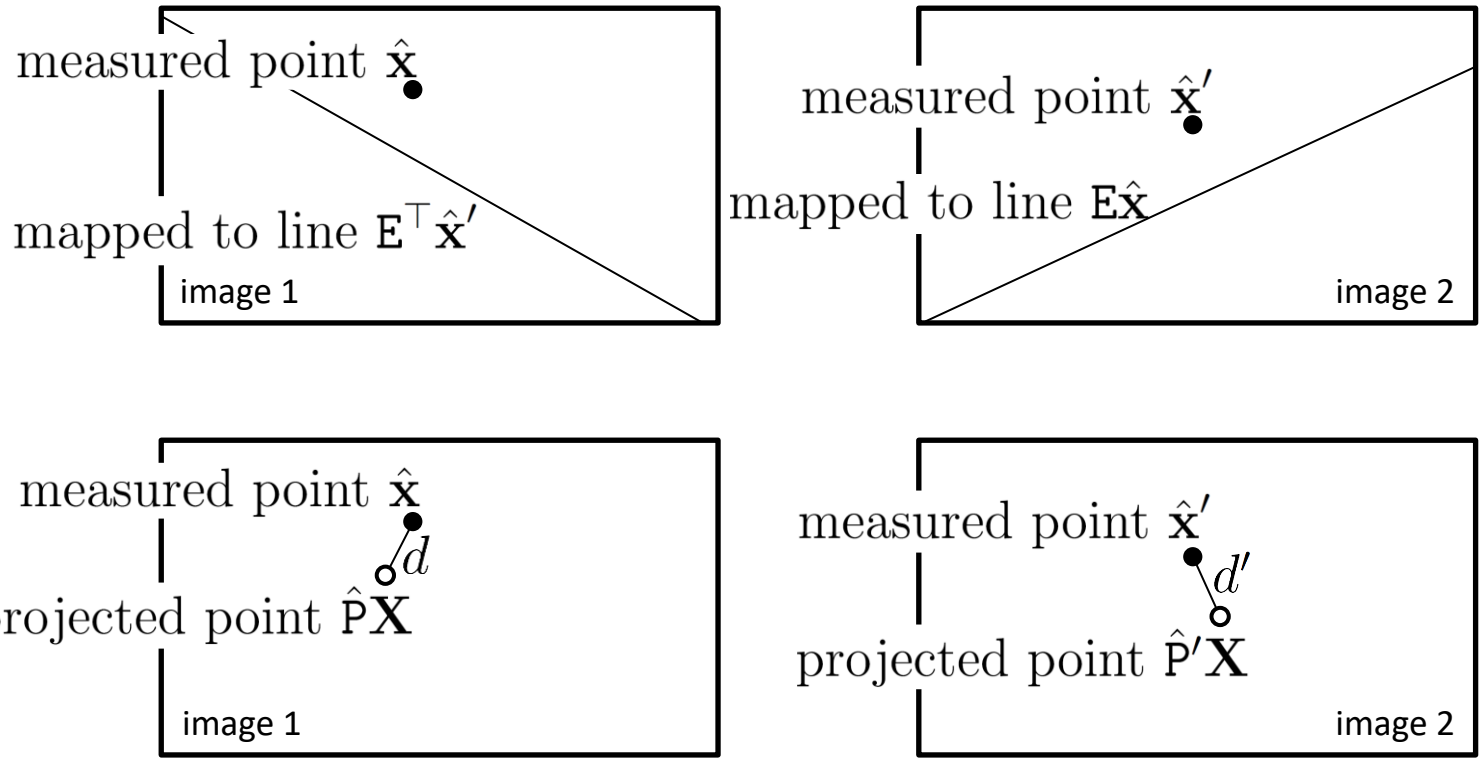
Four possible choices of the second normalized camera projection matrix $\hat{\mathtt{P}}'$

$$\hat{\mathtt{P}}'_1 = [\mathtt{R}_1 \,|\, \mathbf{t}_1], \hat{\mathtt{P}}'_2 = [\mathtt{R}_1 \,|\, \mathbf{t}_2], \hat{\mathtt{P}}'_3 = [\mathtt{R}_2 \,|\, \mathbf{t}_1], \text{ or } \hat{\mathtt{P}}'_4 = [\mathtt{R}_2 \,|\, \mathbf{t}_2]$$

For each possible choice, perform triangulation using one 2D point correspondence $\hat{\mathbf{x}} \leftrightarrow \hat{\mathbf{x}}'$ and choose the $\hat{\mathtt{P}}'$ where the reconstructed scene point is in front of both cameras

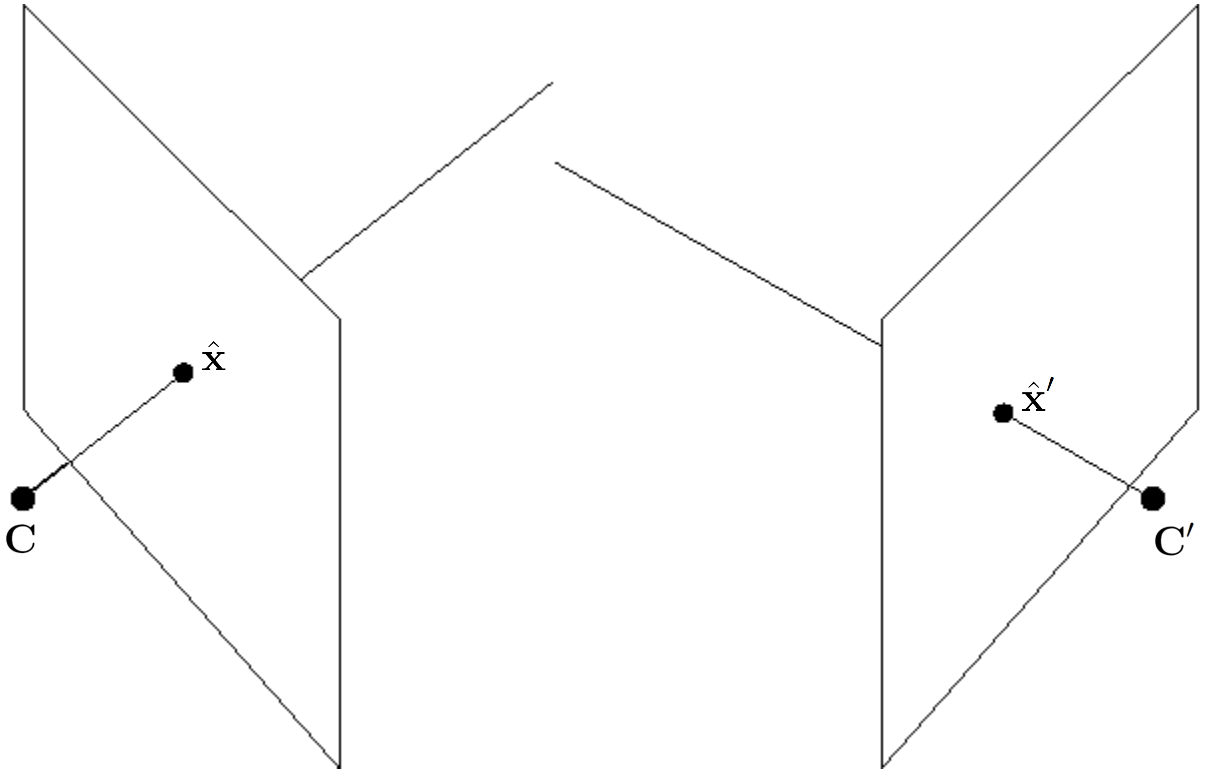
Essential matrix

- Error in image point measurements



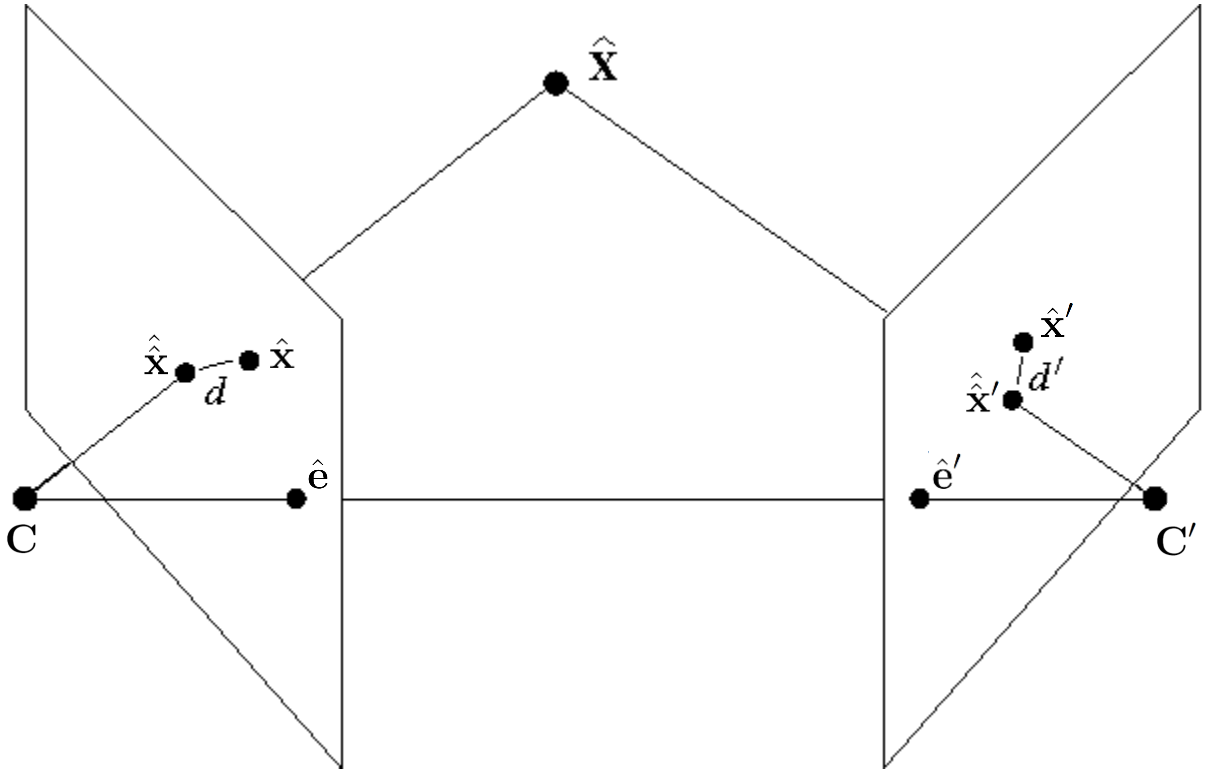
Triangulation

- In general, corresponding 2D points backprojected to 3D lines will not intersect



Triangulation

- Correct image points such that 3D lines intersect



“Optimal” correction

Given the essential matrix $\mathtt{E}$ and corresponding points $\hat{\mathbf{x}} \leftrightarrow \hat{\mathbf{x}}'$, determine the “optimal” corrected corresponding points $\hat{\hat{\mathbf{x}}} \leftrightarrow \hat{\hat{\mathbf{x}}}'$ that exactly satisfy $\hat{\hat{\mathbf{x}}}_i'^{\top} \mathtt{E} \hat{\hat{\mathbf{x}}}_i = 0$ as follows.
Determine the essential matrix $\mathtt{E}_\mathrm{s}$ in a special form.
First, $\mathtt{E}_\mathrm{s} = \mathtt{T}'^{-\top} \mathtt{E} \mathtt{T}^{-1}$, where

$$\mathtt{T} = \begin{bmatrix} \hat{w} & 0 & -\hat{x} \\ 0 & \hat{w} & -\hat{y} \\ 0 & 0 & \hat{w} \end{bmatrix} \text{ and } \mathtt{T}' = \begin{bmatrix} \hat{w}' & 0 & -\hat{x}' \\ 0 & \hat{w}' & -\hat{y}' \\ 0 & 0 & \hat{w}' \end{bmatrix}$$

are the transformations that translate the points to the origin. Then, calculate its associated epipoles $\hat{\mathbf{e}}$ (i.e., the null space of $\mathtt{E}_\mathrm{s}$) and $\hat{\mathbf{e}}'$ (i.e., the null space of $\mathtt{E}_\mathrm{s}^{\top}$).
Then, scale $\hat{\mathbf{e}} = (\hat{e}_1, \hat{e}_2, \hat{e}_3)^{\top}$ and $\hat{\mathbf{e}}' = (\hat{e}'_1, \hat{e}'_2, \hat{e}'_3)^{\top}$ such that $\hat{e}_1^2 + \hat{e}_2^2 = 1$ and $\hat{e}_1'^2 + \hat{e}_2'^2 = 1$, i.e.,

$$\hat{\mathbf{e}} = \frac{1}{\sqrt{\hat{e}_1^2 + \hat{e}_2^2}} \hat{\mathbf{e}} \text{ and } \hat{\mathbf{e}}' = \frac{1}{\sqrt{\hat{e}_1'^2 + \hat{e}_2'^2}} \hat{\mathbf{e}}'$$

“Optimal” correction

Last, $\mathtt{E}_\mathrm{s} = \mathtt{R}'\mathtt{E}_\mathrm{s}\mathtt{R}^\top = \mathtt{R}'\mathtt{T}'^{-\top}\mathtt{E}\mathtt{T}^{-1}\mathtt{R}^\top$, where

$$\mathtt{R} = \begin{bmatrix} \hat{e}_1 & \hat{e}_2 & 0 \\ -\hat{e}_2 & \hat{e}_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \mathtt{R}' = \begin{bmatrix} \hat{e}'_1 & \hat{e}'_2 & 0 \\ -\hat{e}'_2 & \hat{e}'_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

are the transformations that rotate the scaled epipoles to the $\hat{x}$-axis such that $\mathtt{R}\hat{\mathbf{e}} = (1, 0, \hat{e}_3)^\top$ and $\mathtt{R}'\hat{\mathbf{e}}' = (1, 0, \hat{e}'_3)^\top$. Note $\mathtt{E}_\mathrm{s}(1, 0, \hat{e}_3)^\top = \mathbf{0}$ and $(1, 0, \hat{e}'_3)\mathtt{E}_\mathrm{s} = \mathbf{0}^\top$, so $\mathtt{E}_\mathrm{s}$ has the form

$$\mathtt{E}_\mathrm{s} = \begin{bmatrix} ff'd & -f'c & -f'd \\ -fb & a & b \\ -fd & c & d \end{bmatrix}$$

where $f = \hat{e}_3$ and $f' = \hat{e}'_3$.

“Optimal” correction

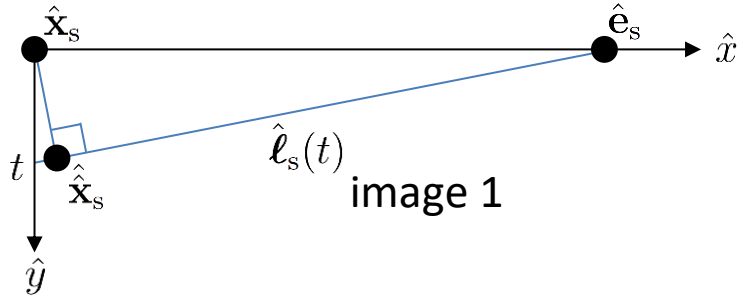
Consider a transformed epipolar line in the first image passing through (the transformed epipole $(1, 0, f)^\top$ and) a point $(0, t, 1)^\top$ on the $\hat{y}$-axis, which also maps to a transformed epipolar line in the second image, i.e.,

$$\hat{\boldsymbol{\ell}}_{\mathrm{s}}(t) = \begin{bmatrix} 0 \\ t \\ 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0 \\ f \end{bmatrix} = \begin{bmatrix} tf \\ 1 \\ -t \end{bmatrix} \text{ and } \hat{\boldsymbol{\ell}}'_{\mathrm{s}}(t) = \mathtt{E}_{\mathrm{s}} \begin{bmatrix} 0 \\ t \\ 1 \end{bmatrix} = \begin{bmatrix} -f'(ct+d) \\ at+b \\ ct+d \end{bmatrix}$$

The sum of squared distance from the origin (i.e., the transformed points) to the transformed epipolar lines is given by

image 1 image 2

$$s(t) = \frac{t^2}{1+f^2t^2} + \frac{(ct+d)^2}{(at+b)^2 + f'^2(ct+d)^2}$$



“Optimal” correction

$$s(t) = \frac{t^2}{1 + f^2 t^2} + \frac{(ct + d)^2}{(at + b)^2 + f'^2 (ct + d)^2}$$

Find the minimum of $s(t)$ by first setting its derivative to 0, which simplifies to

$$g(t) = t((at + b)^2 + f'^2 (ct + d)^2)^2 - (ad - bc)(1 + f^2 t^2)^2 (at + b)(ct + d) = 0,$$

solve for t. The 6 roots of $g(t)$ are the maxima and minima of $s(t)$. Note the minimum may also be at $t = \infty$.

$$\lim_{t \to \infty} s(t) = \frac{1}{f^2} + \frac{c^2}{a^2 + f'^2 c^2}$$

Evaluate $\lim_{t \to \infty} s(t)$ and $s(t)$ at the real roots (or the real part of the 6 roots) of $g(t)$, and select $t_{\min}$ that yields the smallest value.

Determine the transformed points $\hat{\tilde{\mathbf{x}}}_{\mathrm{s}}$ and $\hat{\tilde{\mathbf{x}}}'_{\mathrm{s}}$ as the closest points on the transformed lines $\hat{\boldsymbol{\ell}}_{\mathrm{s}}(t_{\min})$ and $\hat{\boldsymbol{\ell}}'_{\mathrm{s}}(t_{\min})$, noting $\hat{\boldsymbol{\ell}}_{\mathrm{s}}(t \to \infty) = (f, 0, -1)^\top$ and $\hat{\boldsymbol{\ell}}'_{\mathrm{s}}(t \to \infty) = (-f'c, a, c)^\top$, to the origin and inverse transform the points back to their original coordinates.

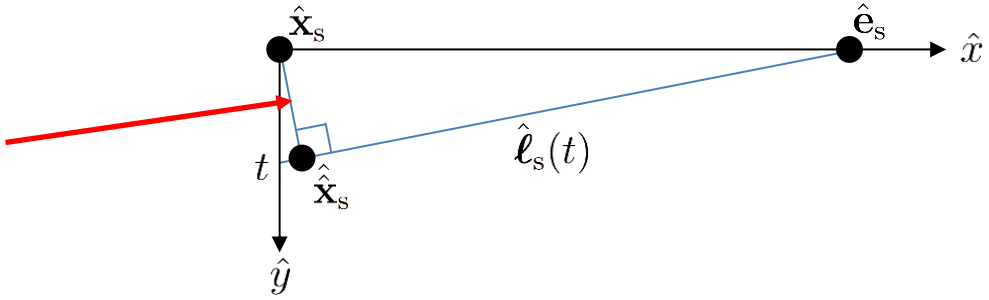
$$\hat{\tilde{\mathbf{x}}} = \mathtt{T}^{-1} \mathtt{R}^\top \hat{\tilde{\mathbf{x}}}_{\mathrm{s}} \text{ and } \hat{\tilde{\mathbf{x}}}' = \mathtt{T}'^{-1} \mathtt{R}'^\top \hat{\tilde{\mathbf{x}}}'_{\mathrm{s}}$$

Line orthogonal to hyperplane and through point

Hyperplane $\boldsymbol{\pi} = (\mathbf{n}^\top, d)^\top$, where $\mathbf{n}$ is normal vector
Point $\mathbf{X}'$ on line orthogonal to $\boldsymbol{\pi}$ and through $\mathbf{X}$ is $\mathbf{X}$ translated by $\mathbf{n}$

$$\mathbf{X}' = \begin{bmatrix} \mathtt{I} & \mathbf{n} \\ \mathbf{0}^\top & 1 \end{bmatrix} \mathbf{X}$$

Line orthogonal to $\boldsymbol{\pi}$ and through $\mathbf{X}$ is join of $\mathbf{X}$ and $\mathbf{X}'$



Point on hyperplane closest to point

Point $\mathbf{X}_{\boldsymbol{\pi}}$ on hyperplane $\boldsymbol{\pi}$ closest to point $\mathbf{X}$ is intersection of line orthogonal to $\boldsymbol{\pi}$ and through $\mathbf{X}$. Represent line as a pencil of points and solve for intersection point $\mathbf{X}_{\boldsymbol{\pi}}$.

$$\mathbf{X}(\lambda) = \lambda\mathbf{X} + (1-\lambda)\mathbf{X}' \text{ and } \quad \boldsymbol{\pi}^\top\mathbf{X}(\lambda) = 0, \text{ solve for } \lambda$$

$$\boldsymbol{\pi}^\top(\lambda\mathbf{X} + (1-\lambda)\mathbf{X}') = 0$$

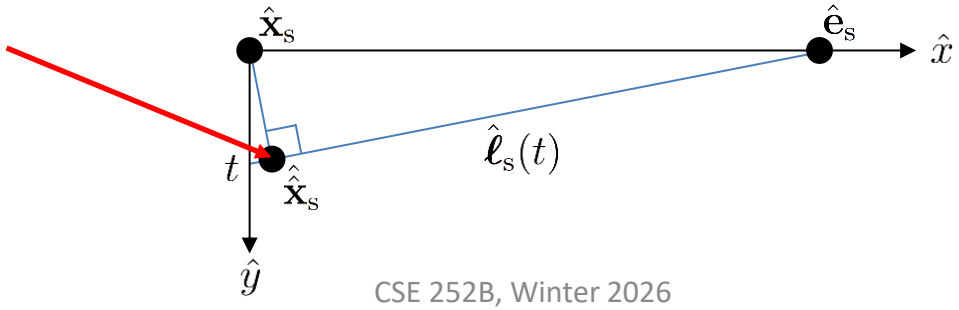
$$\lambda\boldsymbol{\pi}^\top\mathbf{X} + \boldsymbol{\pi}^\top\mathbf{X}' - \lambda\boldsymbol{\pi}^\top\mathbf{X}' = 0$$

$$\lambda(\boldsymbol{\pi}^\top\mathbf{X} - \boldsymbol{\pi}^\top\mathbf{X}') = -\boldsymbol{\pi}^\top\mathbf{X}'$$

$$\lambda = -\frac{\boldsymbol{\pi}^\top\mathbf{X}'}{\boldsymbol{\pi}^\top\mathbf{X} - \boldsymbol{\pi}^\top\mathbf{X}'}$$

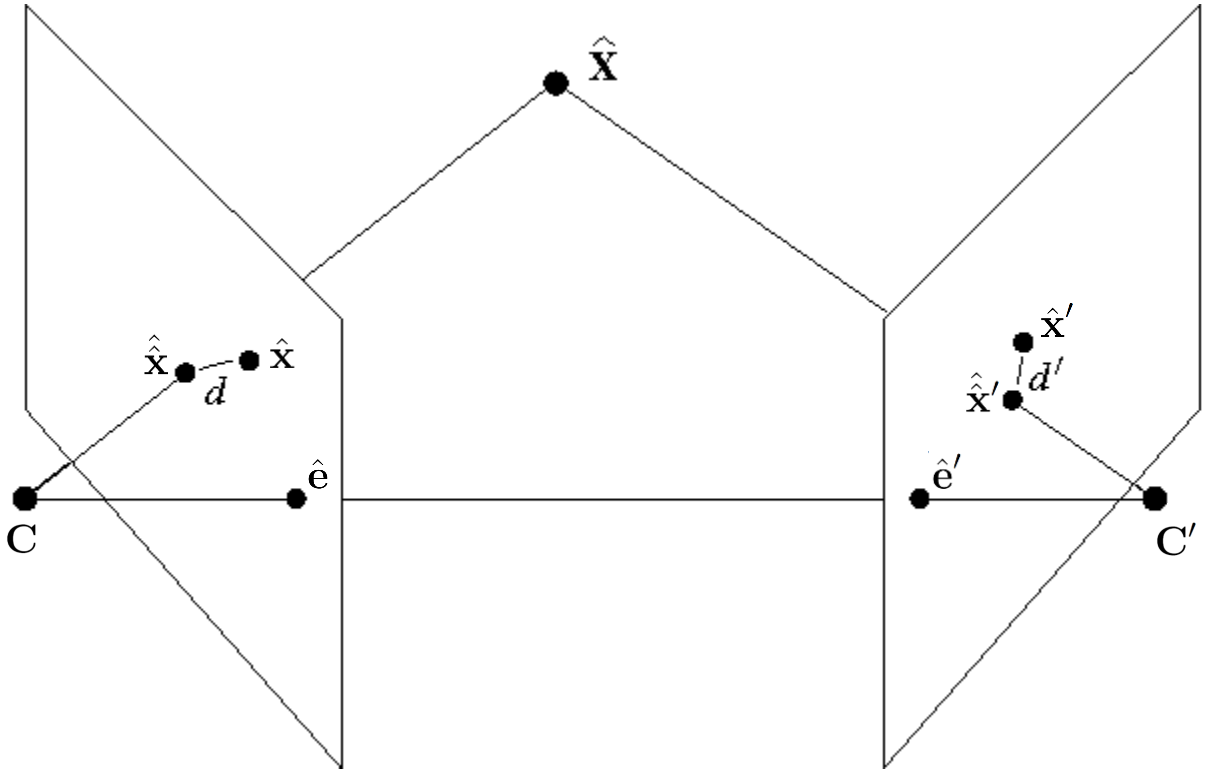
Substitute λ into $\mathbf{X}(\lambda) = \lambda\mathbf{X} + (1-\lambda)\mathbf{X}'$ to determine point on hyperplane $\boldsymbol{\pi}$ closest to point $\mathbf{X}$

$$\mathbf{X}_{\boldsymbol{\pi}} = \lambda\mathbf{X} + (1-\lambda)\mathbf{X}'$$

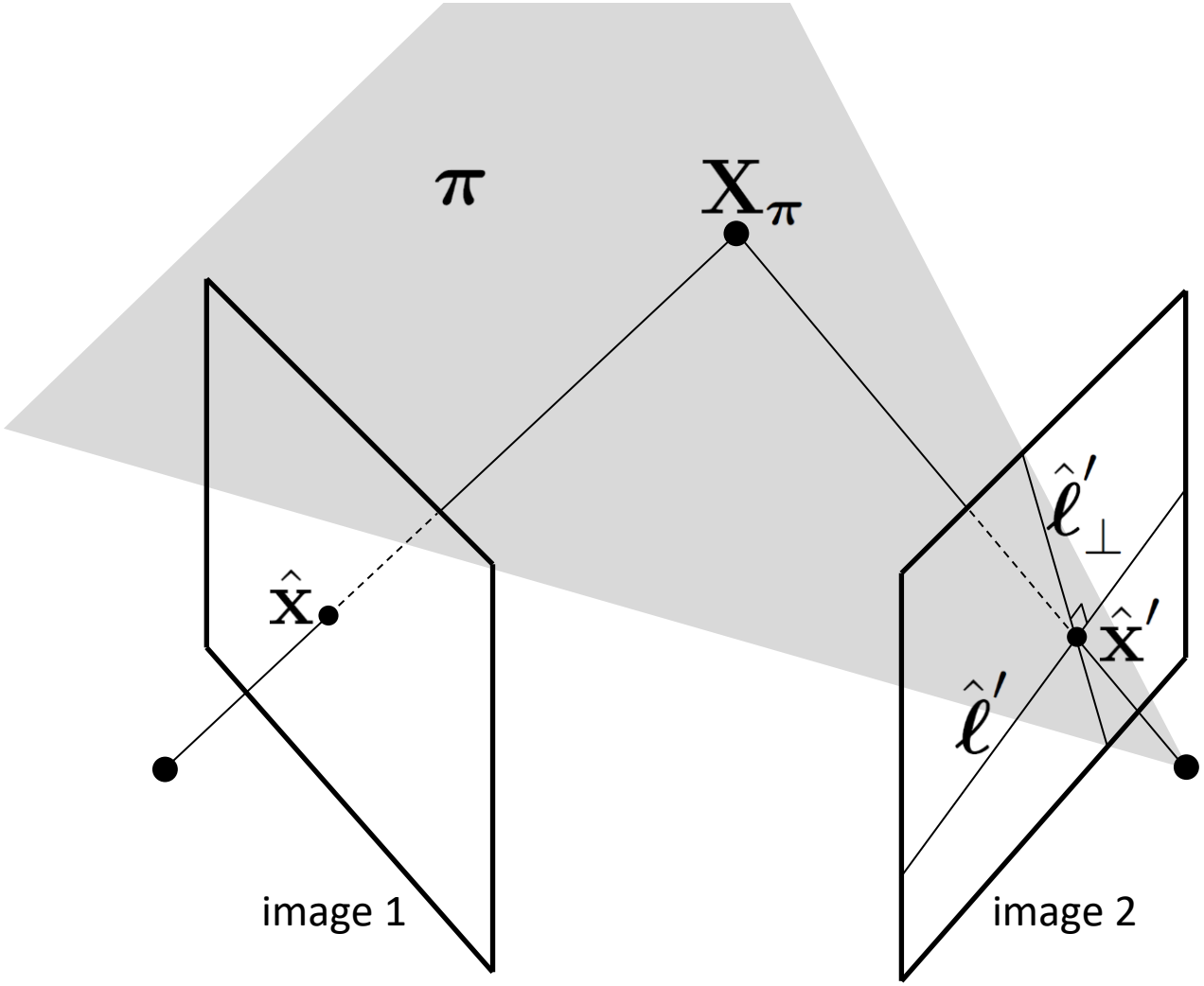


Triangulation

- Corrected image points exactly satisfy epipolar constraint, so 3D lines intersect



Essential matrix



Triangulation

Perform two view triangulation using the corresponding points $\hat{\mathbf{x}} \leftrightarrow \hat{\mathbf{x}}'$ that exactly satisfy the epipolar constraint.

Map point $\hat{\mathbf{x}} = (\hat{x}, \hat{y}, \hat{w})^\top$ to line $\hat{\boldsymbol{\ell}}' = \mathtt{E}\hat{\mathbf{x}} = (\hat{a}', \hat{b}', \hat{c}')^\top$ under the essential matrix $\mathtt{E}$

Determine line $\hat{\boldsymbol{\ell}}'_\perp = (-\hat{b}'\hat{w}', \hat{a}'\hat{w}', \hat{b}'\hat{x}' - \hat{a}'\hat{y}')^\top$ orthogonal to $\hat{\boldsymbol{\ell}}'$ and through $\hat{\mathbf{x}}' = (\hat{x}', \hat{y}', \hat{w}')^\top$

Backproject line $\hat{\boldsymbol{\ell}}'_\perp$ to 3D plane $\boldsymbol{\pi} = \hat{\mathtt{P}}'^\top \hat{\boldsymbol{\ell}}'_\perp = (a, b, c, d)^\top = (\mathbf{n}^\top, d)^\top$

Backproject point $\hat{\mathbf{x}}$ to 3D line defined by two 3D points

$\hat{\mathtt{P}}\mathbf{C} = \mathbf{0}$, where camera center $\mathbf{C}$ is the null space of $\hat{\mathtt{P}}$

$\mathbf{X} = \hat{\mathtt{P}}^+ \hat{\mathbf{x}}$, where $\hat{\mathtt{P}}^+ = \hat{\mathtt{P}}^\top (\hat{\mathtt{P}}\hat{\mathtt{P}}^\top)^{-1}$

Intersection of 3D line and 3D plane

$$\mathbf{X}_{\boldsymbol{\pi}} = \begin{bmatrix} X_2(bY_1 + cZ_1 + dT_1) - X_1(bY_2 + cZ_2 + dT_2) \\ Y_2(aX_1 + cZ_1 + dT_1) - Y_1(aX_2 + cZ_2 + dT_2) \\ Z_2(aX_1 + bY_1 + dT_1) - Z_1(aX_2 + bY_2 + dT_2) \\ T_2(aX_1 + bY_1 + cZ_1) - T_1(aX_2 + bY_2 + cZ_2) \end{bmatrix}, \text{ where } \mathbf{X}_1 = \mathbf{C} \text{ and } \mathbf{X}_2 = \mathbf{X}$$

Triangulation

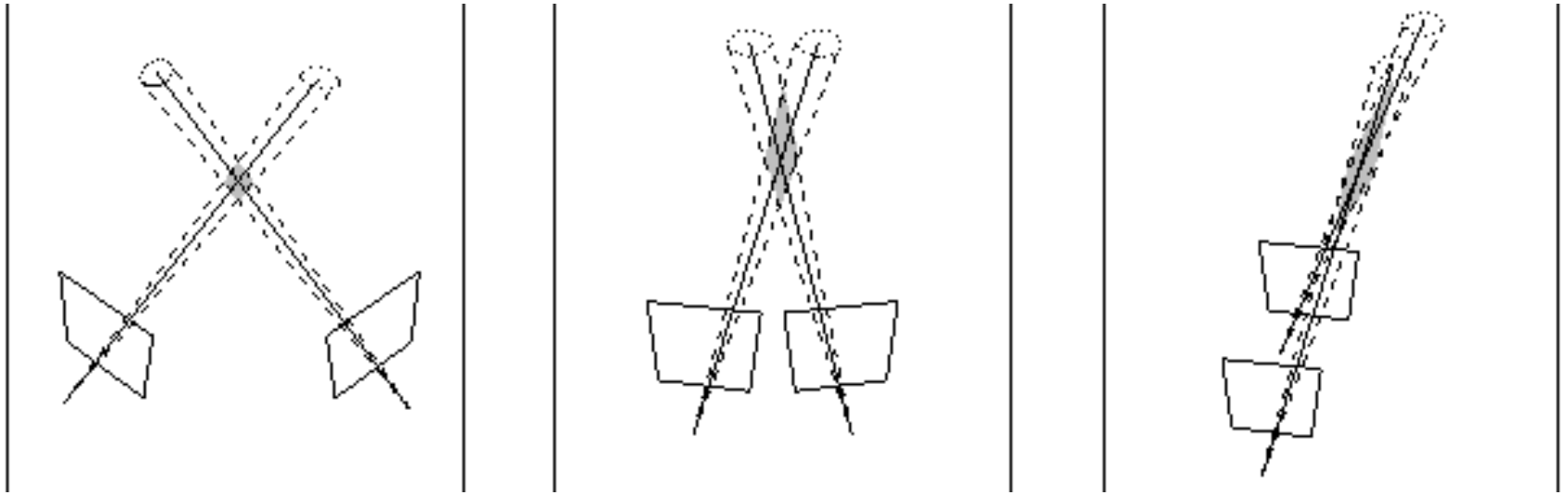
- Canonical cameras

In the case of canonical cameras, $\hat{\mathtt{P}} = [\mathtt{I} \,|\, \mathbf{0}]$ and $\hat{\mathtt{P}}^{+} = \hat{\mathtt{P}}^{\top}$, so $\mathbf{C} = (0, 0, 0, 1)^{\top}$, $\mathbf{X} = (\hat{x}, \hat{y}, \hat{w}, 0)^{\top}$, and

$$\mathbf{X}_{\boldsymbol{\pi}} = \begin{bmatrix} d\hat{x} \\ d\hat{y} \\ d\hat{w} \\ -(a\hat{x} + b\hat{y} + c\hat{w}) \end{bmatrix} = \begin{bmatrix} d\hat{\mathbf{x}} \\ -\mathbf{n}^{\top}\hat{\mathbf{x}} \end{bmatrix}$$

Triangulation uncertainty

- The more orthogonal the 3D lines, the less uncertainty



Nonlinear estimation using the Levenberg-Marquardt algorithm

Given

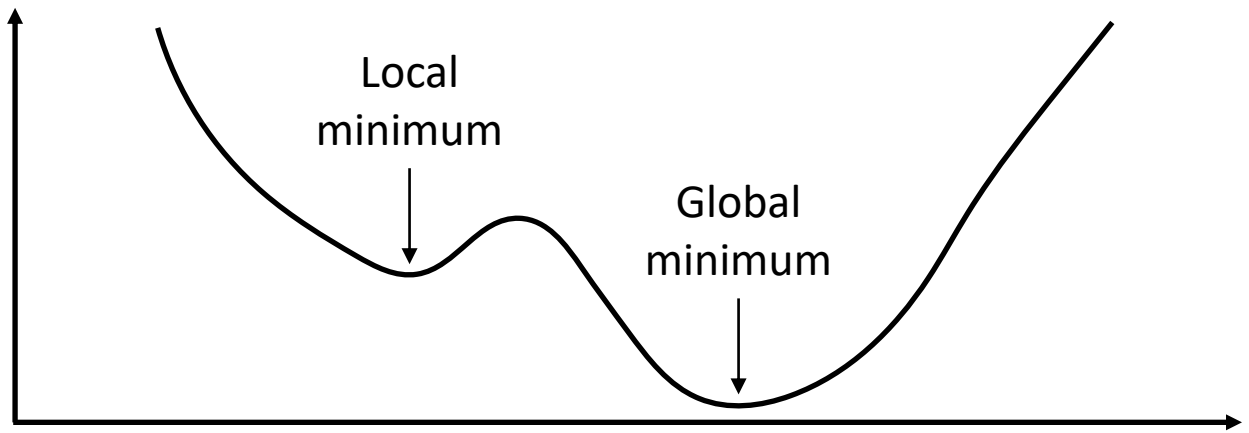
Measurement vector $\mathbf{X}$ with associated covariance matrix $\Sigma_{\mathbf{X}}$

Parameter vector $\hat{\mathbf{P}}$ (initial estimate)

Nonlinear function f, where $\hat{\mathbf{X}} = f(\hat{\mathbf{P}})$

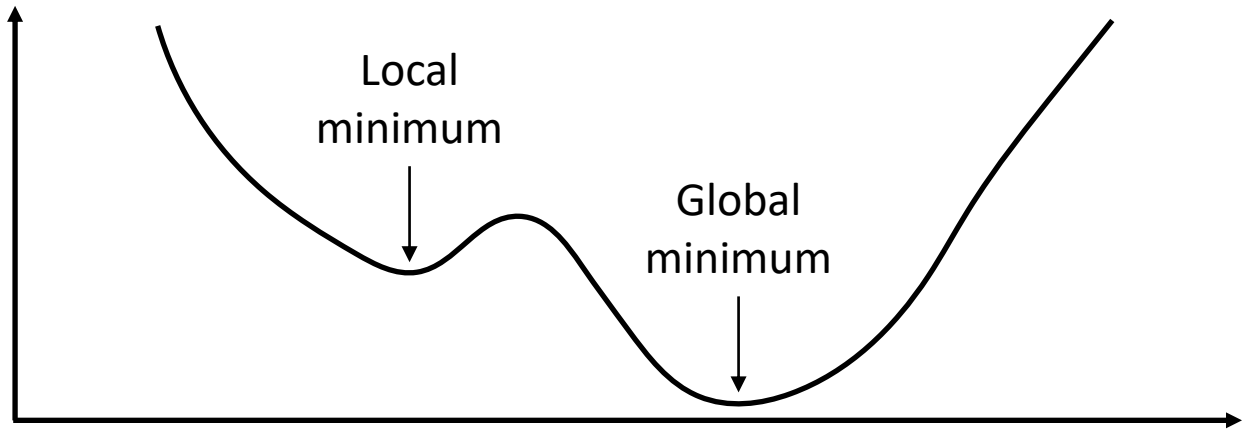
Objective

Find $\hat{\mathbf{P}}$ that minimizes $\boldsymbol{\epsilon}^\top \Sigma_{\mathbf{X}}^{-1} \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} = \mathbf{X} - \hat{\mathbf{X}}$



Parameter vector

- May get stuck in local minimum of high dimensional space
- **Use minimal parameterizations towards mitigating this**



Parameter vector

- The essential matrix has 5 degrees of freedom
- Linear estimate results in rotation matrix (3 degrees of freedom) and unit translation vector (2 degrees of freedom)
- Fix the first normalized camera projection matrix and adjust the second one such that the unit translation vector remains a unit (direction of translation) vector

Angle-axis representation

- Rotation matrix to angle-axis representation

The skew symmetric matrix form of the angle-axis representation $[\boldsymbol{\omega}]_\times$ corresponding to the rotation matrix $\mathtt{R}$ is the matrix logarithm of $\mathtt{R}$.

$$[\boldsymbol{\omega}]_\times = \ln(\mathtt{R})$$

$$\begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} = \ln\left(\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}\right)$$

Angle-axis representation

The angle-axis representation is calculated from the unit vector axis $\mathbf{v}$ and rotation angle θ about $\mathbf{v}$ as

$$\boldsymbol{\omega} = \theta \mathbf{v}$$

where

$$(\mathtt{R} - \mathtt{I})\mathbf{v} = \mathbf{0}, \text{ solve for } \mathbf{v}$$

(i.e., $\mathbf{v}$ is the null space of $(\mathtt{R} - \mathtt{I})$) and

$$\theta = \tan^{-1}\left(\frac{\sin(\theta)}{\cos(\theta)}\right), \text{ where } \cos(\theta) = \frac{\mathrm{Tr}(\mathtt{R}) - 1}{2} \text{ and } \sin(\theta) = \frac{\mathbf{v}^\top \hat{\mathbf{v}}}{2}, \text{ where } \hat{\mathbf{v}} = \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$

then, if $\|\boldsymbol{\omega}\| > \pi$, normalize by

$$\boldsymbol{\omega} = \left(1 - \frac{2\pi}{\|\boldsymbol{\omega}\|}\left\lceil \frac{\|\boldsymbol{\omega}\| - \pi}{2\pi} \right\rceil\right)\boldsymbol{\omega}$$

For very small rotations

$$\boldsymbol{\omega} = \frac{1}{2}\hat{\mathbf{v}}$$

Very small rotation if null space is more than 1 column

Angle-axis representation

- Angle-axis representation to rotation matrix

The rotation matrix $\mathtt{R}$ corresponding to the angle-axis representation $\boldsymbol{\omega}$ is calculated by

$$\mathtt{R} = \exp([\boldsymbol{\omega}]_\times)$$

$$\mathtt{R} = \cos(\theta)\mathtt{I} + \operatorname{sinc}(\theta)[\boldsymbol{\omega}]_\times + \frac{1-\cos(\theta)}{\theta^2}\boldsymbol{\omega}\boldsymbol{\omega}^\top, \text{ where } \theta = \|\boldsymbol{\omega}\|$$

For very small rotations

$$\mathtt{R} = \mathtt{I} + [\boldsymbol{\omega}]_\times$$

Very small rotation if ratio is not a number

Parameterization of the n-sphere

Let the unit vector $\mathbf{x} \in \mathbb{R}^n$ be parameterized using a local parameterization for a neighborhood of the point $\mathbf{x}$ on the unit hypersphere as follows.
First, transform $\mathbf{x}$ to lie along the first coordinate axis using a Householder matrix $\mathtt{H}_{\mathbf{v}(\mathbf{x})}$ (i.e., $\mathtt{H}_{\mathbf{v}(\mathbf{x})}\mathbf{x} = (1, 0, \ldots, 0)^\top = \mathbf{e}_1$), where

$$\mathbf{v}(\mathbf{x}) = \mathbf{x} - \|\mathbf{x}\|\mathbf{e}_1, \text{ where } \|\mathbf{x}\| = 1$$

$$\mathbf{v}(\mathbf{x}) = \mathbf{x} - \mathbf{e}_1$$

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} x_1 - 1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$\mathtt{H}_{\mathbf{v}(\mathbf{x})} = \mathtt{I} - 2\frac{\mathbf{v}(\mathbf{x})\mathbf{v}(\mathbf{x})^\top}{\mathbf{v}(\mathbf{x})^\top\mathbf{v}(\mathbf{x})} \in \mathbb{R}^{n \times n}$$

Second, define a local space $\mathbf{y} \in \mathbb{R}^{n-1}$ such that its origin is mapped to the transformed $\mathbf{x} \in \mathbb{R}^n$, i.e., $(0, \ldots, 0)^\top \in \mathbb{R}^{n-1} \mapsto (1, 0, \ldots, 0)^\top \in \mathbb{R}^n$. To simplify calculations, choose a mapping $f(\mathbf{y})$ with Jacobian

$$\frac{\partial f(\mathbf{y})}{\partial \mathbf{y}} = \begin{bmatrix} \mathbf{0}^\top \\ \mathtt{I} \end{bmatrix} \in \mathbb{R}^{n \times (n-1)}$$

Parameterization of the n-sphere

A mapping, unique for nearly the whole sphere, that satisfies these criteria is

$$f(\mathbf{y}) = (\cos(\|\mathbf{y}\|), \operatorname{sinc}(\|\mathbf{y}\|)\mathbf{y}^\top)^\top \quad \text{for } \|\mathbf{y}\| < \pi$$

The whole sphere would also include $\|\mathbf{y}\| = \pi$, which maps to the exact opposite direction $(-1, 0, \ldots, 0)^\top \in \mathbb{R}^n$ but there is a singularity at $\|\mathbf{y}\| = \pi$, as both $(1, 0, \ldots, 0)^\top \in \mathbb{R}^n$ and $(-1, 0, \ldots, 0)^\top \in \mathbb{R}^n$ inverse map to $(0, \ldots, 0)^\top \in \mathbb{R}^{n-1}$.
A composite of the mapping and the transformation gives the parameterization of the n-sphere

$$\mathbf{x} = \mathtt{H}_{\mathbf{v}(\mathbf{x})}^{-1} f(\mathbf{y}), \text{ where } \mathtt{H}_{\mathbf{v}(\mathbf{x})}^{-1} = \mathtt{H}_{\mathbf{v}(\mathbf{x})}$$
$$\mathbf{x} = \mathtt{H}_{\mathbf{v}(\mathbf{x})} f(\mathbf{y})$$

with associated Jacobian

$$\frac{\partial \mathbf{x}}{\partial \mathbf{y}} = \frac{\partial \mathbf{x}}{\partial f(\mathbf{y})} \frac{\partial f(\mathbf{y})}{\partial \mathbf{y}}$$
$$\frac{\partial \mathbf{x}}{\partial \mathbf{y}} = \mathtt{H}_{\mathbf{v}(\mathbf{x})} \begin{bmatrix} \mathbf{0}^\top \\ \mathtt{I} \end{bmatrix}$$

(i.e., $\partial \mathbf{x}/\partial \mathbf{y}$ is $\mathtt{H}_{\mathbf{v}(\mathbf{x})}$ with the first column omitted)

Parameterization of a homogeneous vector

Let the homogeneous vector $\bar{\mathbf{v}} = (a, \mathbf{b}^\top)^\top \in \mathbb{R}^n$, where $\|\bar{\mathbf{v}}\| = 1$ (i.e., $\bar{\mathbf{v}}$ is a unit vector), be parameterized as

$$\mathbf{v} = \frac{2}{\operatorname{sinc}(\cos^{-1}(a))}\mathbf{b} \in \mathbb{R}^{n-1}$$

then, if $\|\mathbf{v}\| > \pi$, normalized by

$$\mathbf{v} = \left(1 - \frac{2\pi}{\|\mathbf{v}\|}\left\lceil\frac{\|\mathbf{v}\| - \pi}{2\pi}\right\rceil\right)\mathbf{v}$$

Always use the parameterized homogeneous vector, not the input homogeneous vector or matrix

Parameterization of a homogeneous vector

The parameterized homogeneous vector $\mathbf{v}$ is deparameterized as the homogeneous vector

$$\bar{\mathbf{v}} = \left(\cos\left(\frac{\|\mathbf{v}\|}{2}\right), \frac{\operatorname{sinc}\left(\frac{\|\mathbf{v}\|}{2}\right)}{2}\mathbf{v}^\top \right)^\top \in \mathbb{R}^n$$

Unit vector with nonnegative first element

$$\bar{\mathbf{v}} = \left(a, \mathbf{b}^\top\right)^\top, \text{ where } a = \cos\left(\frac{\|\mathbf{v}\|}{2}\right) \text{ and } \mathbf{b} = \frac{\operatorname{sinc}\left(\frac{\|\mathbf{v}\|}{2}\right)}{2}\mathbf{v}$$

where $\|\bar{\mathbf{v}}\| = 1$ and a is nonnegative. For the deparameterization,

$$\frac{\partial \bar{\mathbf{v}}}{\partial \mathbf{v}} = \frac{\partial (a, \mathbf{b}^\top)}{\partial \mathbf{v}} = \begin{bmatrix} \frac{\mathrm{d}a}{\partial \mathbf{v}} \\ \frac{\partial \mathbf{b}}{\partial \mathbf{v}} \end{bmatrix} \in \mathbb{R}^{n \times (n-1)}$$

where

$$\frac{\mathrm{d}a}{\partial \mathbf{v}} = \begin{cases} \mathbf{0}^\top & \text{if } \|\mathbf{v}\| = 0 \\ -\frac{1}{2}\mathbf{b}^\top & \text{otherwise} \end{cases}$$

and

$$\frac{\partial \mathbf{b}}{\partial \mathbf{v}} = \begin{cases} \frac{1}{2}\mathtt{I} & \text{if } \|\mathbf{v}\| = 0 \\ \frac{\operatorname{sinc}\left(\frac{\|\mathbf{v}\|}{2}\right)}{2}\mathtt{I} + \frac{1}{4\|\mathbf{v}\|}\frac{\mathrm{d}\operatorname{sinc}\left(\frac{\|\mathbf{v}\|}{2}\right)}{\mathrm{d}\frac{\|\mathbf{v}\|}{2}}\mathbf{v}\mathbf{v}^\top & \text{otherwise} \end{cases}$$

Parameterization of a homogeneous vector

- Sinc function

The sinc function

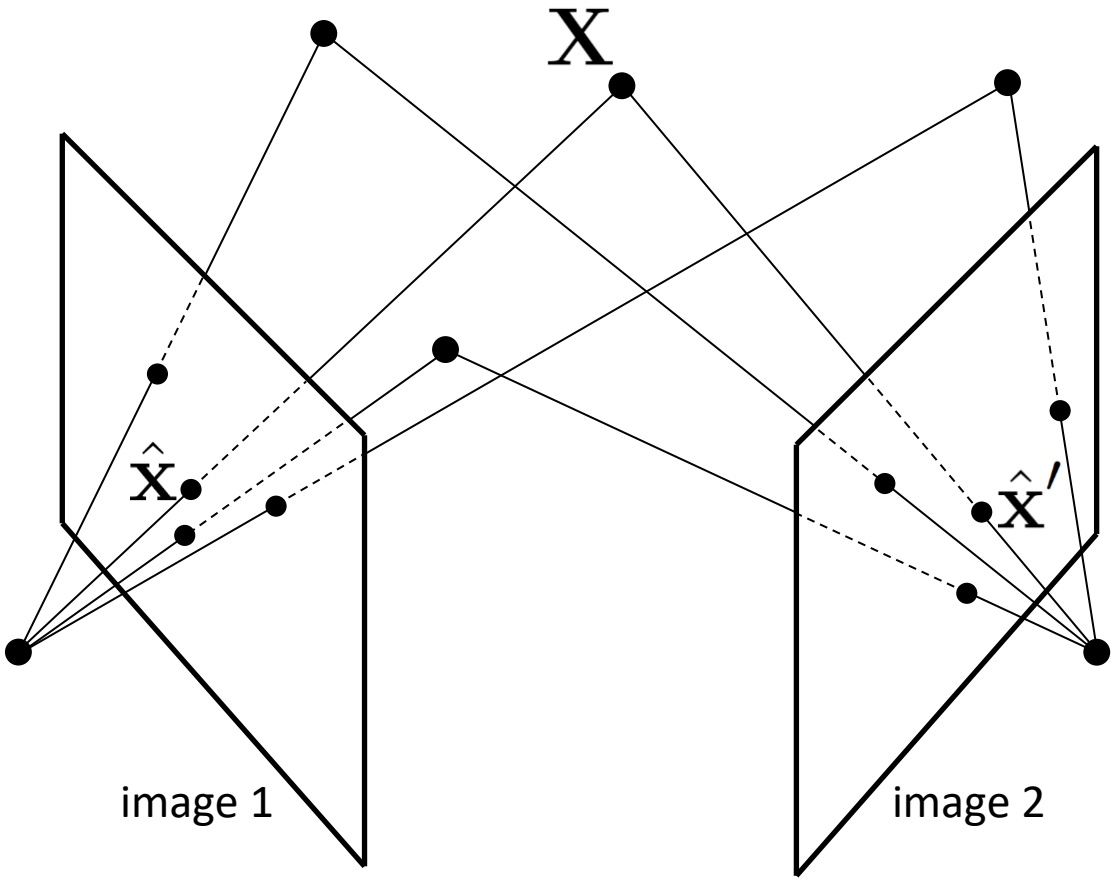
$$\operatorname{sinc}(x) = \begin{cases} 1 & \text{if } x = 0 \\ \frac{\sin(x)}{x} & \text{otherwise} \end{cases}$$

The derivative is given by

$$\frac{\mathrm{d}\operatorname{sinc}(x)}{\mathrm{d}x} = \begin{cases} 0 & \text{if } x = 0 \\ \frac{\cos(x)}{x} - \frac{\sin(x)}{x^2} & \text{otherwise} \end{cases}$$

Two view geometry

- Imaging a 3D scene



Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

Unit vector

Parameterization of homogeneous vector

Parameter vector $(3 + 3 + 3n)$ x 1

$$\left(\hat{\boldsymbol{\omega}}'^{\top}, \hat{\mathbf{t}}'^{\top}, \hat{\mathbf{X}}_1^{\top}, \hat{\mathbf{X}}_2^{\top}, \ldots, \hat{\mathbf{X}}_n^{\top}\right)^{\top}$$

Initialize scene points using two-view optimal triangulation

Always use the parameterized homogeneous vector, not the input homogeneous vector or matrix

Mapping

$$\hat{\hat{\mathbf{x}}}_i = \hat{\mathtt{P}}\hat{\mathbf{X}}_i = [\mathtt{I} \,|\, \mathbf{0}]\hat{\mathbf{X}}_i \,\forall\, i$$

$$\hat{\hat{\mathbf{x}}}'_i = \hat{\mathtt{P}}'\hat{\mathbf{X}}_i = [\exp([\hat{\boldsymbol{\omega}}']_\times) \,|\, \hat{\mathbf{t}}'] \begin{bmatrix} \hat{X}_i \\ \hat{Y}_i \\ \hat{Z}_i \\ \hat{T}_i \end{bmatrix} = \exp([\hat{\boldsymbol{\omega}}']_\times) \begin{bmatrix} \hat{X}_i \\ \hat{Y}_i \\ \hat{Z}_i \end{bmatrix} + \hat{T}_i \hat{\mathbf{t}}' \,\forall\, i$$

Measurement vector $4n$ x 1

$$\left(\hat{\hat{\mathbf{x}}}_1^{\top}, \hat{\hat{\mathbf{x}}}_2^{\top}, \ldots, \hat{\hat{\mathbf{x}}}_n^{\top}, \hat{\hat{\mathbf{x}}}_1'^{\top}, \hat{\hat{\mathbf{x}}}_2'^{\top}, \ldots, \hat{\hat{\mathbf{x}}}_n'^{\top}\right)^{\top}$$

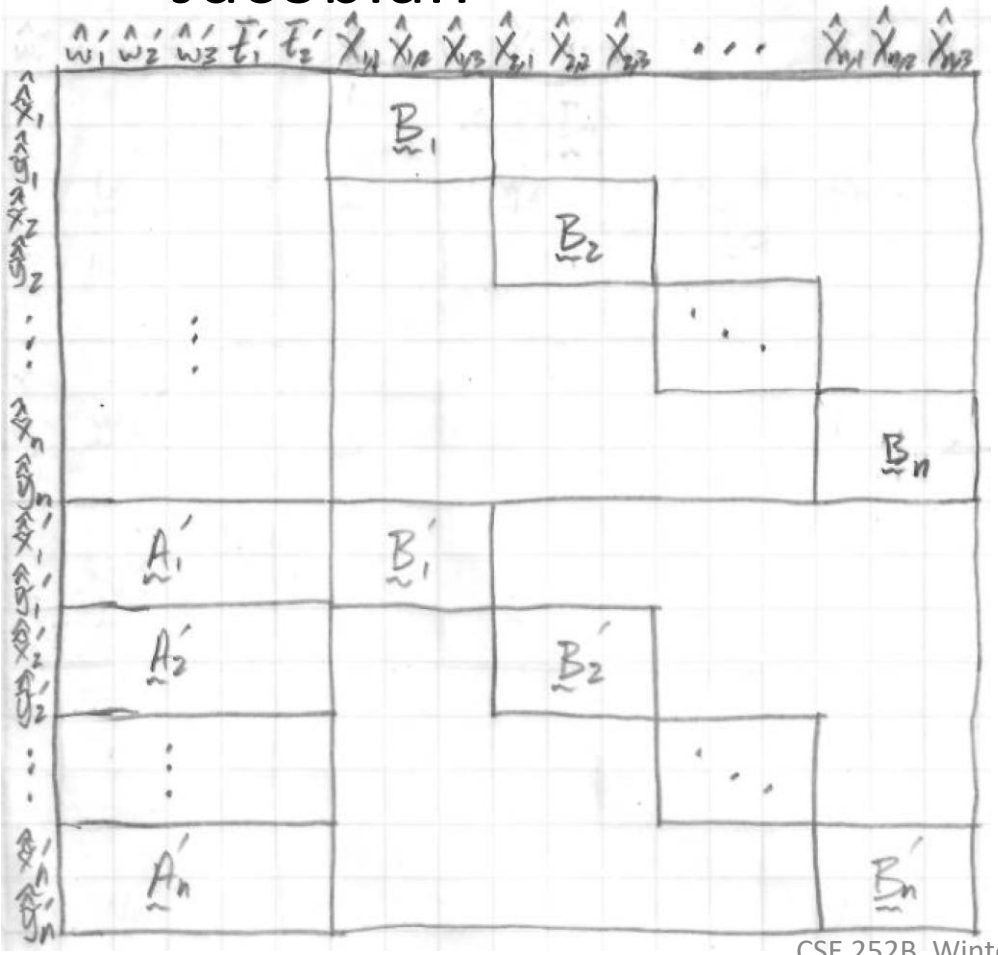
with associated covariance matrices $\Sigma_{\hat{\hat{\mathbf{x}}}_1}, \Sigma_{\hat{\hat{\mathbf{x}}}_2}, \ldots, \Sigma_{\hat{\hat{\mathbf{x}}}_n}, \Sigma_{\hat{\hat{\mathbf{x}}}'_1}, \Sigma_{\hat{\hat{\mathbf{x}}}'_2}, \ldots, \Sigma_{\hat{\hat{\mathbf{x}}}'_n}$

Cost

$$\sum_i \left(\boldsymbol{\epsilon}_i^{\top} \Sigma_{\hat{\hat{\mathbf{x}}}_i}^{-1} \boldsymbol{\epsilon}_i + \boldsymbol{\epsilon}_i'^{\top} \Sigma_{\hat{\hat{\mathbf{x}}}'_i}^{-1} \boldsymbol{\epsilon}'_i\right), \text{ where } \boldsymbol{\epsilon}_i = \hat{\hat{\mathbf{x}}}_i - \hat{\hat{\mathbf{x}}}_i \text{ and } \boldsymbol{\epsilon}'_i = \hat{\hat{\mathbf{x}}}'_i - \hat{\hat{\mathbf{x}}}'_i$$

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

- Jacobian



2 x 3 2 x 2

Parameterization of n-sphere

$$\mathbf{A}'_i = \frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial(\hat{\boldsymbol{\omega}}'^{\top}, \bar{\mathbf{t}}'^{\top})} = \left[\frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial \hat{\boldsymbol{\omega}}'} \quad \frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial \bar{\mathbf{t}}'}\right]$$

$$\text{where } \frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial \bar{\mathbf{t}}'} = \frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial \hat{\mathbf{t}}'} \mathbf{H}_{\mathbf{v}(\hat{\mathbf{t}}')} \begin{bmatrix} \mathbf{0}^{\top} \\ \mathbf{I} \end{bmatrix}$$

3 x 3 3 x 2

$$\mathbf{B}_i = \frac{\partial \hat{\bar{\mathbf{x}}}_i}{\partial \hat{\mathbf{X}}_i} = \frac{\partial \hat{\bar{\mathbf{x}}}_i}{\partial \bar{\mathbf{X}}_i} \frac{\partial \bar{\mathbf{X}}_i}{\partial \hat{\mathbf{X}}_i}$$

$$\mathbf{B}'_i = \frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial \hat{\mathbf{X}}_i} = \frac{\partial \hat{\bar{\mathbf{x}}}'_i}{\partial \bar{\mathbf{X}}_i} \frac{\partial \bar{\mathbf{X}}_i}{\partial \hat{\mathbf{X}}_i}$$

2 x 4 4 x 3

Projection of a point under the normalized camera projection matrix

The homogeneous 3D point $\mathbf{X} = (X, Y, Z, T)^\top$ is projected to the homogeneous 2D point in normalized coordinates $\hat{\mathbf{x}}$ under the normalized camera projection matrix $\hat{\mathtt{P}} = [\exp([\boldsymbol{\omega}]_\times) \,|\, \mathbf{t}] = [\mathtt{R} \,|\, \mathbf{t}]$, where $\mathtt{R} = \exp([\boldsymbol{\omega}]_\times)$, by

$$\hat{\mathbf{x}} = \hat{\mathtt{P}}\mathbf{X}$$

$$\hat{\mathbf{x}} = [\exp([\boldsymbol{\omega}]_\times) \,|\, \mathbf{t}] \begin{bmatrix} \mathbf{v} \\ T \end{bmatrix}, \text{ where } \mathbf{v} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\hat{\mathbf{x}} = \exp([\boldsymbol{\omega}]_\times)\mathbf{v} + T\mathbf{t}$$

$$\hat{\mathbf{x}} = \mathbf{v}' + T\mathbf{t}$$

where $\mathbf{v}' = \exp([\boldsymbol{\omega}]_\times)\mathbf{v} = \mathtt{R}\mathbf{v}$. Dehomogenizing the 2D point results in the mapping $\mathbf{X} \mapsto \hat{\hat{\mathbf{x}}}$.

Projection of a point under the normalized camera projection matrix

For this mapping

$$\frac{\partial \hat{\tilde{\mathbf{x}}}}{\partial \boldsymbol{\omega}} = \frac{\partial \hat{\tilde{\mathbf{x}}}}{\partial \mathbf{v}'} \frac{\partial \mathbf{v}'}{\partial \boldsymbol{\omega}} \qquad 2 \text{ x } 3$$

where

$$\frac{\partial \hat{\tilde{\mathbf{x}}}}{\partial \mathbf{v}'} = \begin{bmatrix} 1/\hat{w} & 0 & -\hat{\tilde{x}}/\hat{w} \\ 0 & 1/\hat{w} & -\hat{\tilde{y}}/\hat{w} \end{bmatrix} \qquad 2 \text{ x } 3$$

where $\hat{w} = v_3' + t_3 T$, $\frac{\partial \mathbf{v}'}{\partial \mathbf{v}} = \exp([\boldsymbol{\omega}]_\times) = \mathtt{R}$,

$$\frac{\partial \hat{\tilde{\mathbf{x}}}}{\partial \mathbf{t}} = \begin{bmatrix} T/\hat{w} & 0 & -\hat{\tilde{x}}T/\hat{w} \\ 0 & T/\hat{w} & -\hat{\tilde{y}}T/\hat{w} \end{bmatrix} \qquad 2 \text{ x } 3$$

where $\hat{w} = \mathbf{r}^{3\top}\mathbf{v} + t_3 T$, and

$$\frac{\partial \hat{\tilde{\mathbf{x}}}}{\partial \mathbf{X}} = \frac{1}{\hat{w}} \begin{bmatrix} \hat{\mathbf{p}}^{1\top} - \hat{\tilde{x}}\hat{\mathbf{p}}^{3\top} \\ \hat{\mathbf{p}}^{2\top} - \hat{\tilde{y}}\hat{\mathbf{p}}^{3\top} \end{bmatrix} \qquad 2 \text{ x } 4$$

where $\hat{w} = \hat{\mathbf{p}}^{3\top}\mathbf{X}$ and $\hat{\mathbf{p}}^{i\top}$ is the ith row of $\hat{\mathtt{P}}$.

Angle-axis representation, rotate vector

The 3-vector $\mathbf{v}$ is rotated to $\mathbf{v}'$ under the angle-axis representation $\boldsymbol{\omega}$ by

$$\mathbf{v}' = \exp([\boldsymbol{\omega}]_\times)\mathbf{v}$$

$$\mathbf{v}' = \begin{cases} \mathbf{v} + \boldsymbol{\omega} \times \mathbf{v} & \text{if } \theta \text{ is 0 or nearly 0} \\ \mathbf{v} + \operatorname{sinc}(\theta)\boldsymbol{\omega} \times \mathbf{v} + \frac{1-\cos(\theta)}{\theta^2}\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{v}) & \text{otherwise} \end{cases}$$

Very small rotation if ratio is not a number

where $\theta = \|\boldsymbol{\omega}\|$. For this rotation

$$\frac{\partial \mathbf{v}'}{\partial \boldsymbol{\omega}} = \begin{cases} [-\mathbf{v}]_\times & \text{if } \theta \text{ is 0 or nearly 0} \\ \operatorname{sinc}(\theta)[-\mathbf{v}]_\times + (\boldsymbol{\omega} \times \mathbf{v})\frac{\mathrm{d}\operatorname{sinc}(\theta)}{\mathrm{d}\theta}\frac{\mathrm{d}\theta}{\partial \boldsymbol{\omega}} & \\ + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{v})\frac{\mathrm{d}s}{\mathrm{d}\theta}\frac{\mathrm{d}\theta}{\partial \boldsymbol{\omega}} + s([\boldsymbol{\omega}]_\times[-\mathbf{v}]_\times + [-(\boldsymbol{\omega} \times \mathbf{v})]_\times) & \text{otherwise} \end{cases}$$

where

$$s = \frac{1-\cos(\theta)}{\theta^2} \text{ and } \frac{\mathrm{d}s}{\mathrm{d}\theta} = \frac{\theta\sin(\theta) - 2(1-\cos(\theta))}{\theta^3}$$

and

$$\frac{\mathrm{d}\theta}{\partial \boldsymbol{\omega}} = \frac{1}{\theta}\boldsymbol{\omega}^\top \text{ and } \frac{\partial \mathbf{v}'}{\partial \mathbf{v}} = \exp([\boldsymbol{\omega}]_\times)$$

Very small rotation if ratio is not a number

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

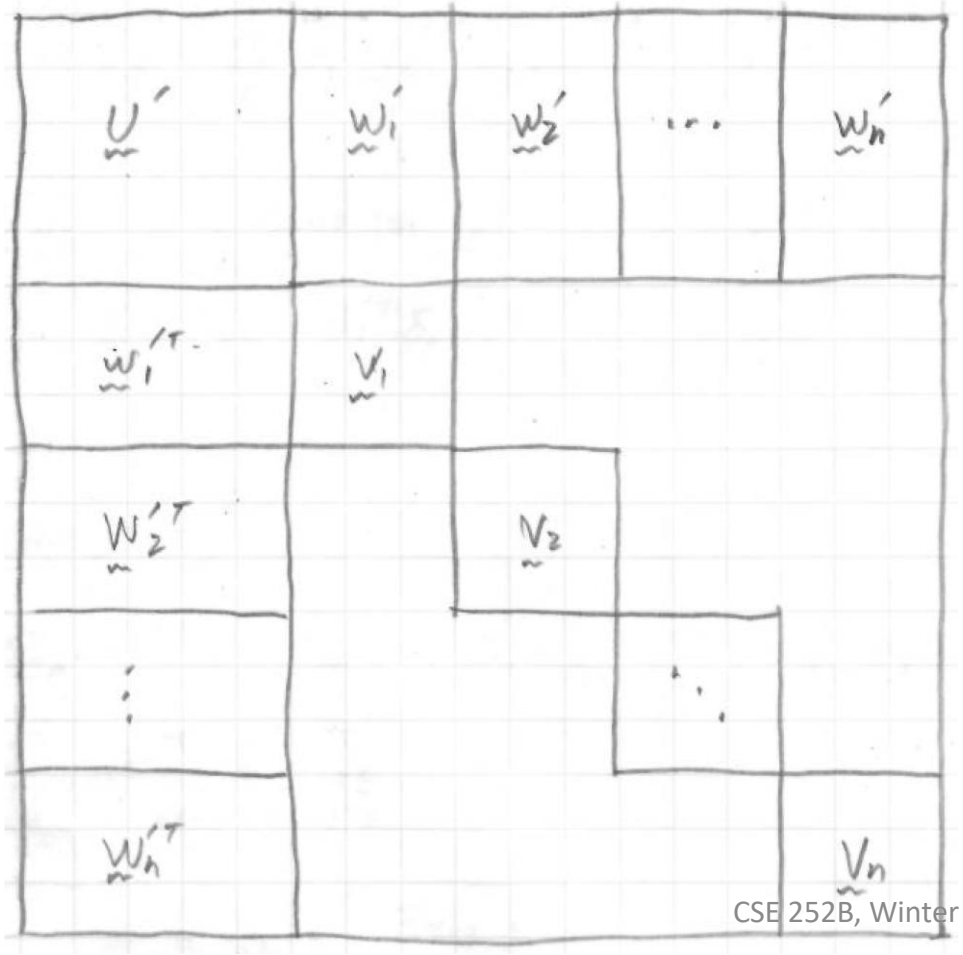
- Normal equations

$$\left(\mathtt{J}^\top \Sigma_{\mathbf{X}}^{-1} \mathtt{J}\right) \boldsymbol{\delta} = \mathtt{J}^\top \Sigma_{\mathbf{X}}^{-1} \boldsymbol{\epsilon}$$

$$\begin{bmatrix} \mathtt{U}' & \mathtt{W}' \\ \mathtt{W}'^\top & \mathtt{V} \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_{\mathbf{a}'} \\ \boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

- Normal equations matrix $\begin{bmatrix} \mathrm{U}' & \mathrm{W}' \\ \mathrm{W}'^{\top} & \mathrm{V} \end{bmatrix}$



$$\mathrm{U}' = \sum_i \mathrm{A}_i'^{\top} \Sigma_{\hat{\hat{\mathbf{x}}}_i'}^{-1} \mathrm{A}_i' \qquad 5 \times 5$$

$$\mathrm{V}_i = \mathrm{B}_i^{\top} \Sigma_{\hat{\hat{\mathbf{x}}}_i}^{-1} \mathrm{B}_i + \mathrm{B}_i'^{\top} \Sigma_{\hat{\hat{\mathbf{x}}}_i'}^{-1} \mathrm{B}_i' \qquad 3 \times 3$$

$$\mathrm{W}_i' = \mathrm{A}_i'^{\top} \Sigma_{\hat{\hat{\mathbf{x}}}_i'}^{-1} \mathrm{B}_i' \qquad 5 \times 3$$

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

- Normal equations vector $\begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$

$$\left(\boldsymbol{\epsilon}_{\mathbf{a}'}^\top, \boldsymbol{\epsilon}_{\mathbf{b}_1}^\top, \boldsymbol{\epsilon}_{\mathbf{b}_2}^\top, \ldots, \boldsymbol{\epsilon}_{\mathbf{b}_n}^\top\right)^\top$$

where

$$\boldsymbol{\epsilon}_{\mathbf{a}'} = \sum_i \mathtt{A}_i'^\top \Sigma_{\hat{\hat{\mathbf{x}}}_i'}^{-1} \boldsymbol{\epsilon}_i' \qquad 5 \text{ x } 1$$

$$\boldsymbol{\epsilon}_{\mathbf{b}_i} = \mathtt{B}_i^\top \Sigma_{\hat{\hat{\mathbf{x}}}_i}^{-1} \boldsymbol{\epsilon}_i + \mathtt{B}_i'^\top \Sigma_{\hat{\hat{\mathbf{x}}}_i'}^{-1} \boldsymbol{\epsilon}_i' \qquad 3 \text{ x } 1$$

where $\boldsymbol{\epsilon}_i = \hat{\hat{\mathbf{x}}}_i - \hat{\hat{\bar{\mathbf{x}}}}_i$ and $\boldsymbol{\epsilon}_i' = \hat{\hat{\mathbf{x}}}_i' - \hat{\hat{\bar{\mathbf{x}}}}_i'$

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

Augmented normal equations

$$\begin{bmatrix} \mathtt{U}'^{*} & \mathtt{W}' \\ \mathtt{W}'^{\top} & \mathtt{V}^{*} \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_{\mathbf{a}'} \\ \boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

where $\mathtt{U}'^{*} = \mathtt{U}' + \lambda \mathtt{I}$ and $\mathtt{V}_i^{*} = \mathtt{V}_i + \lambda \mathtt{I}$

Solve for $\left(\boldsymbol{\delta}_{\mathbf{a}'}^{\top}, \boldsymbol{\delta}_{\mathbf{b}}^{\top}\right)^{\top}$ using blockwise methods

$$\begin{bmatrix} \mathtt{I} & -\mathtt{W}'\mathtt{V}^{*-1} \\ \mathtt{0} & \mathtt{I} \end{bmatrix} \begin{bmatrix} \mathtt{U}'^{*} & \mathtt{W}' \\ \mathtt{W}'^{\top} & \mathtt{V}^{*} \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_{\mathbf{a}'} \\ \boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \mathtt{I} & -\mathtt{W}'\mathtt{V}^{*-1} \\ \mathtt{0} & \mathtt{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

$$\begin{bmatrix} \mathtt{U}'^{*} - \mathtt{W}'\mathtt{V}^{*-1}\mathtt{W}'^{\top} & \mathtt{0} \\ \mathtt{W}'^{\top} & \mathtt{V}^{*} \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_{\mathbf{a}'} \\ \boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} - \mathtt{W}'\mathtt{V}^{*-1}\boldsymbol{\epsilon}_{\mathbf{b}} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

$$\begin{bmatrix} (\mathtt{U}'^{*} - \mathtt{W}'\mathtt{V}^{*-1}\mathtt{W}'^{\top})\boldsymbol{\delta}_{\mathbf{a}'} \\ \mathtt{W}'^{\top}\boldsymbol{\delta}_{\mathbf{a}'} + \mathtt{V}^{*}\boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} - \mathtt{W}'\mathtt{V}^{*-1}\boldsymbol{\epsilon}_{\mathbf{b}} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

$$\begin{bmatrix} (\mathtt{U}'^{*} - \mathtt{W}'\mathtt{V}^{*-1}\mathtt{W}'^{\top})\boldsymbol{\delta}_{\mathbf{a}'} \\ \mathtt{W}'^{\top}\boldsymbol{\delta}_{\mathbf{a}'} + \mathtt{V}^{*}\boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} - \mathtt{W}'\mathtt{V}^{*-1}\boldsymbol{\epsilon}_{\mathbf{b}} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

1. Solve for $\boldsymbol{\delta}_{\mathbf{a}'}$

$$(\mathtt{U}'^{*} - \mathtt{W}'\mathtt{V}^{*-1}\mathtt{W}'^{\top})\boldsymbol{\delta}_{\mathbf{a}'} = \boldsymbol{\epsilon}_{\mathbf{a}'} - \mathtt{W}'\mathtt{V}^{*-1}\boldsymbol{\epsilon}_{\mathbf{b}}$$

$$\mathtt{S}'\boldsymbol{\delta}_{\mathbf{a}'} = \mathbf{e}', \text{ solve for } \boldsymbol{\delta}_{\mathbf{a}'} \quad 5 \text{ x } 1$$

where

$$\mathtt{S}' = \mathtt{U}'^{*} - \sum_i \mathtt{W}'_i\mathtt{V}_i^{*-1}\mathtt{W}_i'^{\top} \quad 5 \text{ x } 5$$

$$\mathbf{e}' = \boldsymbol{\epsilon}_{\mathbf{a}'} - \sum_i \mathtt{W}'_i\mathtt{V}_i^{*-1}\boldsymbol{\epsilon}_{\mathbf{b}_i} \quad 5 \text{ x } 1$$

Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

$$\begin{bmatrix} (\mathtt{U}'^{*} - \mathtt{W}'\mathtt{V}^{*-1}\mathtt{W}'^{\top})\boldsymbol{\delta}_{\mathbf{a}'} \\ \mathtt{W}'^{\top}\boldsymbol{\delta}_{\mathbf{a}'} + \mathtt{V}^{*}\boldsymbol{\delta}_{\mathbf{b}} \end{bmatrix} - \begin{bmatrix} \boldsymbol{\epsilon}_{\mathbf{a}'} - \mathtt{W}'\mathtt{V}^{*-1}\boldsymbol{\epsilon}_{\mathbf{b}} \\ \boldsymbol{\epsilon}_{\mathbf{b}} \end{bmatrix}$$

2. Solve for $\boldsymbol{\delta}_{\mathbf{b}}$

$$\begin{aligned} \mathtt{W}'^{\top}\boldsymbol{\delta}_{\mathbf{a}'} + \mathtt{V}^{*}\boldsymbol{\delta}_{\mathbf{b}} &= \boldsymbol{\epsilon}_{\mathbf{b}} \\ \mathtt{V}^{*}\boldsymbol{\delta}_{\mathbf{b}} &= \boldsymbol{\epsilon}_{\mathbf{b}} - \mathtt{W}'^{\top}\boldsymbol{\delta}_{\mathbf{a}'} \\ \boldsymbol{\delta}_{\mathbf{b}} &= \mathtt{V}^{*-1}\left(\boldsymbol{\epsilon}_{\mathbf{b}} - \mathtt{W}'^{\top}\boldsymbol{\delta}_{\mathbf{a}'}\right) \end{aligned}$$

where

$$\boldsymbol{\delta}_{\mathbf{b}_i} = \mathtt{V}_i^{*-1}(\boldsymbol{\epsilon}_{\mathbf{b}_i} - \mathtt{W}_i'^{\top}\boldsymbol{\delta}_{\mathbf{a}'}) \quad 3 \text{ x } 1$$

Candidate parameter vector

$$\begin{aligned} \boldsymbol{\delta}_{\mathbf{a}'} &= (\boldsymbol{\delta}_{\hat{\boldsymbol{\omega}}'}^{\top}, \boldsymbol{\delta}_{\bar{\mathbf{t}}'}^{\top})^{\top} \\ \hat{\boldsymbol{\omega}}_0' &= \hat{\boldsymbol{\omega}}' + \boldsymbol{\delta}_{\hat{\boldsymbol{\omega}}'} \\ \hat{\mathbf{t}}_0' &= \mathtt{H}_{\mathbf{v}(\hat{\mathbf{t}}')} f(\boldsymbol{\delta}_{\bar{\mathbf{t}}'}) \\ \hat{\mathbf{X}}_{i_0} &= \hat{\mathbf{X}}_i + \boldsymbol{\delta}_{\mathbf{b}_i} \end{aligned}$$

Adjustment

Parameterization of n-sphere

Normalize angle-axis representation after adjustment

Map to normalized coordinates with covariance propagation

The image point in pixel coordinates $\mathbf{x}$ is mapped to the image point in normalized coordinates $\hat{\mathbf{x}}$ by

$$\hat{\mathbf{x}} = \mathtt{K}^{-1}\mathbf{x}$$

$$\begin{bmatrix}\hat{\tilde{\mathbf{x}}}\\ 1\end{bmatrix} = \begin{bmatrix}a_{11} & a_{12} & a_{13}\\ 0 & a_{22} & a_{23}\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{x}}\\ 1\end{bmatrix}, \text{ where } \mathtt{K}^{-1} = \begin{bmatrix}a_{11} & a_{12} & a_{13}\\ 0 & a_{22} & a_{23}\\ 0 & 0 & 1\end{bmatrix}$$

$$\hat{\tilde{\mathbf{x}}} = \begin{bmatrix}a_{11} & a_{12}\\ 0 & a_{22}\end{bmatrix}\tilde{\mathbf{x}} + \begin{bmatrix}a_{13}\\ a_{23}\end{bmatrix}$$

Covariance propagation

$$\Sigma_{\hat{\tilde{\mathbf{x}}}} = \mathtt{J}\Sigma_{\tilde{\mathbf{x}}}\mathtt{J}^{\top}, \text{ where } \mathtt{J} = \frac{\partial\hat{\tilde{\mathbf{x}}}}{\partial\tilde{\mathbf{x}}} = \begin{bmatrix}a_{11} & a_{12}\\ 0 & a_{22}\end{bmatrix}$$

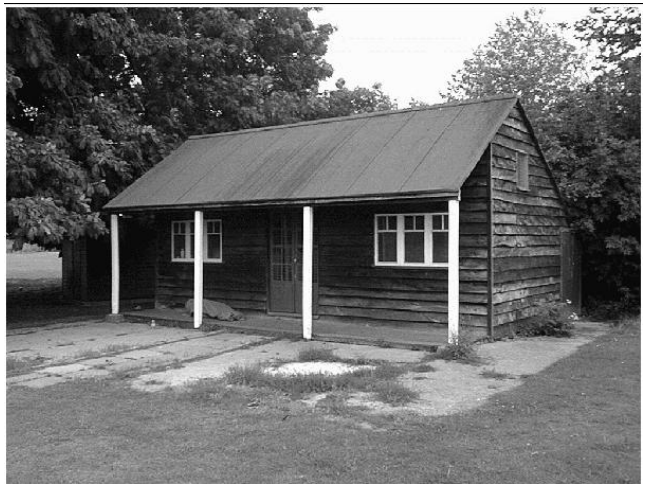
If unknown, then assume to be identity

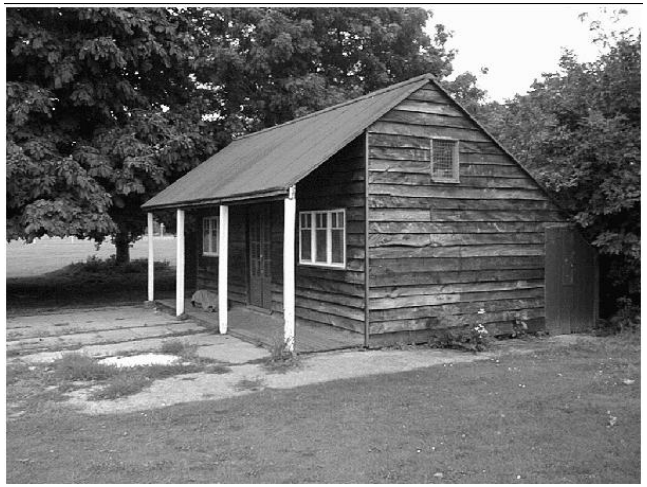
Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

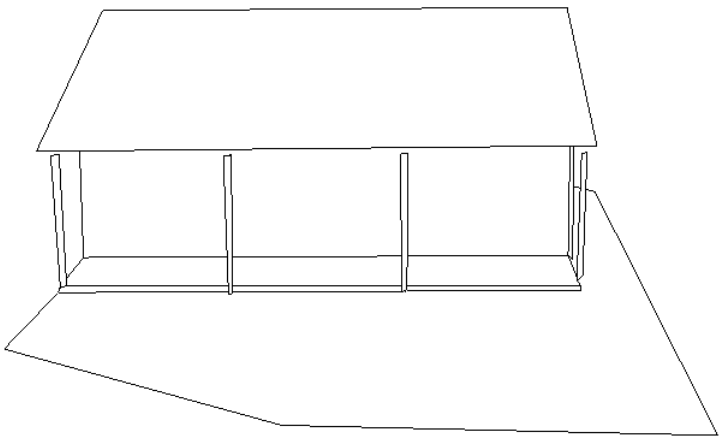
- Algorithm

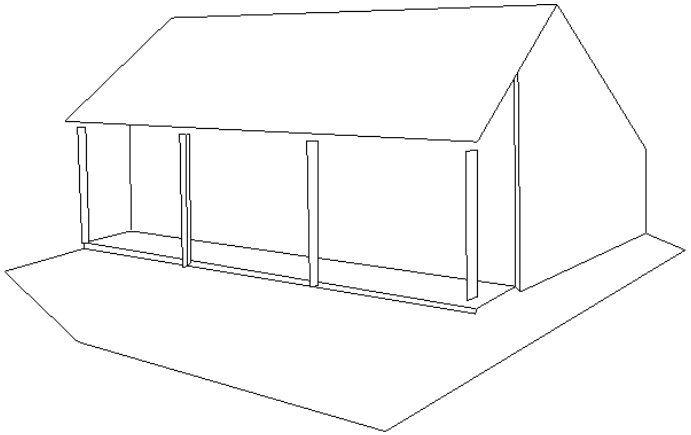
1. Initialize the scene points using two-view optimal triangulation of the corresponding image points in normalized coordinates
2. Estimate data normalized camera projection matrix $\hat{\mathbf{P}}'$ and scene points from the normalized image points

Up to 3D similarity transformation









3D geometric transformations

- Similarity

Scale, rotation, and translation (7 degrees of freedom)

$$\begin{bmatrix} \tilde{X}' \\ \tilde{Y}' \\ \tilde{Z}' \end{bmatrix} = s \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}, \text{ where } \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \in \mathrm{SO}(3)$$

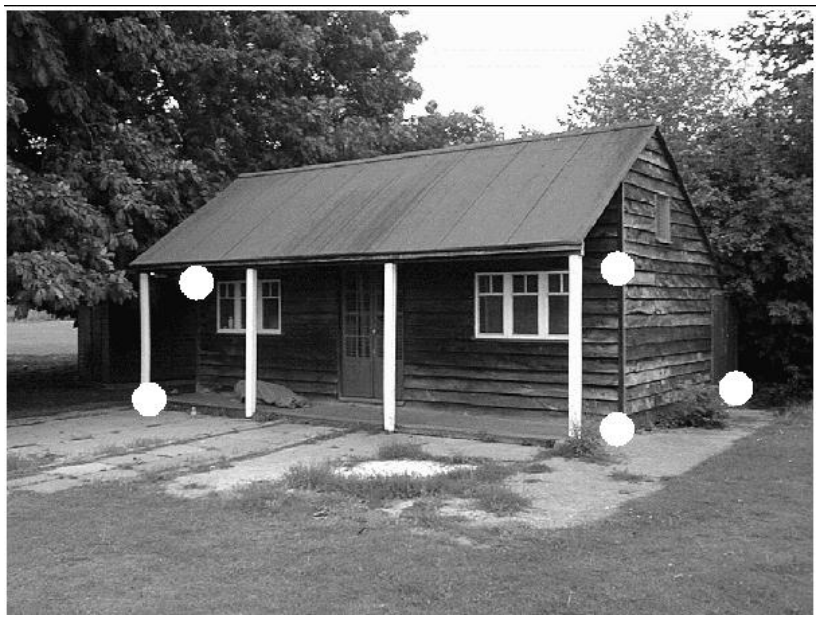
$$\tilde{\mathbf{X}}' = s\mathtt{R}\tilde{\mathbf{X}} + \mathbf{t}, \text{ where } \mathtt{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \in \mathrm{SO}(3) \text{ and } \mathbf{t} = (t_1, t_2, t_3)^\top$$

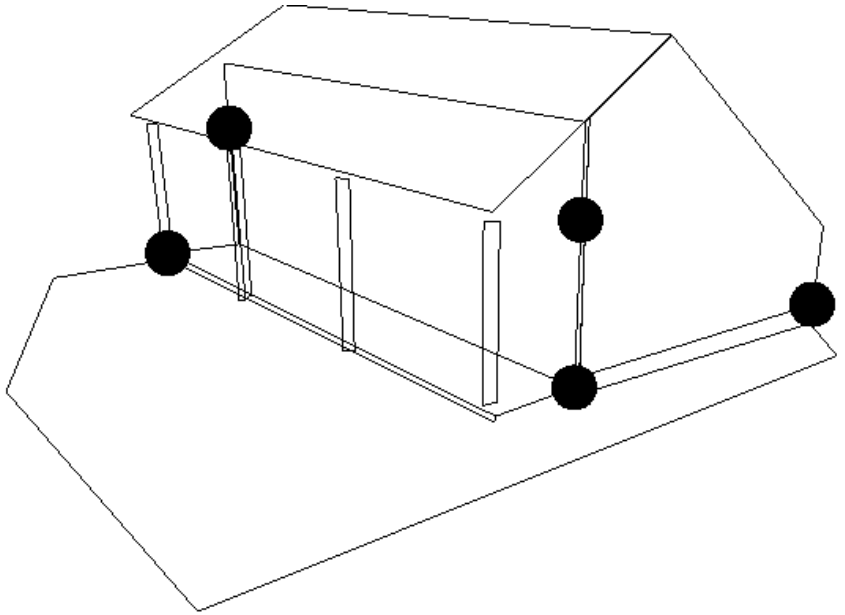
$$\tilde{\mathbf{X}}' = \begin{bmatrix} s\mathtt{R} \,|\, \mathbf{t} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{X}} \\ 1 \end{bmatrix}$$

$$\mathbf{X}' = \begin{bmatrix} s\mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \mathbf{X}$$

$$\mathbf{X}' = \mathtt{H}_\mathrm{S}\mathbf{X}, \text{ where } \mathtt{H}_\mathrm{S} = \begin{bmatrix} s\mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

3D similarity transformation





3D similarity transformation

Given $n \geq 3$ point correspondences $\tilde{\mathbf{X}}_i \leftrightarrow \tilde{\mathbf{X}}'_i$
$\tilde{\mathbf{X}}'_i = s\mathtt{R}\tilde{\mathbf{X}}_i + \mathbf{t} \,\forall\, i$, solve for s, $\mathtt{R}$, and $\mathbf{t}$

1. Calculate the mean vector $\boldsymbol{\mu}_{\tilde{\mathbf{X}}}$ (i.e., centroid) and total unbiased variance $\sigma^2_{\tilde{\mathbf{X}}}$ of the first set of 3D points
2. Calculate the mean vector $\boldsymbol{\mu}_{\tilde{\mathbf{X}}'}$ (i.e., centroid) of the second set of 3D points
3. Compute the matrix

$$\mathtt{S} = \frac{1}{n-1}\sum_i \left(\tilde{\mathbf{X}}'_i - \boldsymbol{\mu}_{\tilde{\mathbf{X}}'}\right)\left(\tilde{\mathbf{X}}_i - \boldsymbol{\mu}_{\tilde{\mathbf{X}}}\right)^\top$$

and its singular value decomposition $\mathtt{S} = \mathtt{U}\Sigma\mathtt{V}^\top$, where $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \sigma_3)$ and $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \sigma_3)^\top$

3D similarity transformation

4. The rotation matrix

$$\mathtt{R} = \begin{cases} \mathtt{U}\,\mathrm{diag}(1,1,-1)\mathtt{V}^\top & \text{if } \det(\mathtt{U})\det(\mathtt{V}) < 0 \\ \mathtt{U}\mathtt{V}^\top & \text{otherwise} \end{cases}$$

   such that $\det(\mathtt{R}) = +1$

5. The scale

$$s = \begin{cases} \frac{\boldsymbol{\sigma}^\top (1,1,-1)^\top}{\sigma^2_{\tilde{\mathbf{X}}}} & \text{if } \det(\mathtt{U})\det(\mathtt{V}) < 0 \\ \frac{\mathrm{Tr}(\Sigma)}{\sigma^2_{\tilde{\mathbf{X}}}} & \text{otherwise} \end{cases}$$

   such that $\det(\mathtt{R}) = +1$

6. The translation vector $\mathbf{t} = \boldsymbol{\mu}_{\tilde{\mathbf{X}}'} - s\mathtt{R}\boldsymbol{\mu}_{\tilde{\mathbf{X}}}$

Next lecture

- Three view geometry
- Reading
  - Chapters 15 and 16