Estimation of Essential Matrix, Nonlinear

Computer Vision II
CSE 252B
Lecture 15
Announcements

- Assignment 4 is due Mar 6, 11:59 PM
- Assignment 5 will be released Mar 6
  - Due Mar 20, 11:59 PM
Two view geometry
Two view geometry

• Imaging a 3D scene
Calibrated camera

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

where

\[
\begin{align*}
\alpha_x &= fm_x & \text{Focal length in } x \text{ direction in terms of pixel dimensions} \\
\alpha_y &= fm_y & \text{Focal length in } y \text{ direction in terms of pixel dimensions} \\
s &= \text{Skew} \\
(x_0, y_0)^T &= (pxm_x, pym_y)^T & \text{Coordinates of principal point in terms of pixel dimensions}
\end{align*}
\]

where \( m_x \) and \( m_y \) are number of pixels per unit distance 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

\[ \hat{x} = \hat{P}X \]
Image coordinate frames

- Pixel coordinate frame
- Principal point
- Normalized coordinate frame
Camera projection

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

$$
\mathbf{x} = \mathbf{P} \mathbf{X} \\
\mathbf{x} = K[R | t] \mathbf{X} \\
K^{-1} \mathbf{x} = [R | t] \mathbf{X} \\
\hat{\mathbf{x}} = \hat{\mathbf{P}} \mathbf{X}
$$

where

Camera projection matrix $\mathbf{P} = K[R | t] = K\hat{\mathbf{P}}$

Normalized camera projection matrix $\hat{\mathbf{P}} = [R | t] = K^{-1} \mathbf{P}$

Image point in pixel coordinates $\mathbf{x} = K[R | t] \mathbf{X} = K\hat{\mathbf{x}}$

Image point in normalized coordinates $\hat{\mathbf{x}} = [R | t] \mathbf{X} = K^{-1} \mathbf{x}$
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

\[ \hat{\ell}' = E\hat{x} \text{ and } \hat{x}'^T \hat{\ell}' = 0 \]
\[ \hat{x}'^T E\hat{x} = 0 \quad \text{Epipolar constraint} \]

\[ \hat{\ell}' = E\hat{x} \] \hspace{1cm} \text{The epipolar line corresponding to the point } \hat{x} \]
\[ \hat{\ell} = E^T \hat{x}' \] \hspace{1cm} \text{The epipolar line corresponding to the point } \hat{x}' \]
\[ E\hat{e} = 0 \] \hspace{1cm} \text{The epipole } \hat{e} \text{ is the null space of } E \]
\[ \hat{e}'^T E = 0^T \] \hspace{1cm} \text{The epipole } \hat{e}' \text{ is the left null space of } E \]
\[ E^T \hat{e}' = 0 \] \hspace{1cm} \text{The epipole } \hat{e}' \text{ is the null space of } E^T \]

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

\[ [R \mid t]H^{-1}_E = [I \mid 0] \text{ and } [R' \mid t']H^{-1}_E = [R'R^T \mid t' - R'R^T t], \text{ where } H^{-1}_E = \begin{bmatrix} R^T & -R^T t \\ 0^T & 1 \end{bmatrix} \]

Then

\[ E = [t' - R'R^T t] \times R'R^T \]

The essential matrix \( E \) from canonical cameras \( P = [I \mid 0] \) and \( P' = [R \mid t] \)

\[ E = [t] \times R \]

CSE 252B, Winter 2024
Essential matrix

• Error in image point measurements
Estimation of essential matrix

\[ \hat{x}_i = \hat{P}X_i \forall i \] and \[ \hat{x}'_i = \hat{P}'X_i \forall i, \] set \[ \hat{P} = [I \mid 0] \] and solve for \( \hat{P}' \)

- Given point correspondences \( \hat{x}_i \leftrightarrow \hat{x}'_i \)
- Minimize geometric error \[ \sum_i \left( d(\hat{x}_i, \hat{P}X_i)^2 + d(\hat{x}'_i, \hat{P}'X_i)^2 \right) \]
- 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

\[ \hat{x}' E \hat{x} = 0 \quad \text{Epipolar constraint} \]
Linear estimation of essential matrix using the direct linear transformation (DLT) algorithm

Given \( n \geq 8 \) point correspondences \( \hat{x}_i \leftrightarrow \hat{x}'_i \)

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

\[
\begin{align*}
\hat{x}'_i^T E \hat{x}_i &= 0 \forall i, \text{ solve for } E \\
(\hat{x}'_i^T \otimes \hat{x}_i^T) e &= 0 \forall i, \text{ solve for } e = \text{vec}(E^T) \\
\begin{bmatrix}
\hat{x}'_1^T \otimes \hat{x}_1^T \\
\hat{x}'_2^T \otimes \hat{x}_2^T \\
\vdots \\
\hat{x}'_n^T \otimes \hat{x}_n^T 
\end{bmatrix} e &= 0 \\
A e &= 0, \text{ where } A = \\
\begin{bmatrix}
\hat{x}'_1^T \otimes \hat{x}_1^T \\
\hat{x}'_2^T \otimes \hat{x}_2^T \\
\vdots \\
\hat{x}'_n^T \otimes \hat{x}_n^T 
\end{bmatrix}
\end{align*}
\]

Then, enforce constraint (using SVD)

\[
E = U \Sigma V^T, \text{ where } \Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3) \\
E = U \Sigma' V^T, \text{ where } \Sigma' = \text{diag}(1, 1, 0)
\]
Decompose essential matrix

\[ \hat{P} = [I \mid 0] \text{ and } \hat{P}' = [R \mid t] \]

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

(a) \hspace{2cm} (b) \hspace{2cm} (c) \hspace{2cm} (d)
Decompose essential matrix

\( \hat{P} = [I \mid 0] \) and \( \hat{P}' = [R \mid t] \)

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

\[ E = UDV^\top \]  singular value decomposition, where \( D = \text{diag}(1, 1, 0) \) to scale

Two possible choices of second camera rotation matrix \( R \)

\[ R_1 = UZV^\top \] or \( R_2 = UZ^\top V^\top \), where \( Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \)

Ensure \( R_1 \) and \( R_2 \) are special orthogonal. If \( \det(R_1) = -1 \), then negate \( R_1 \). If \( \det(R_2) = -1 \), then negate \( R_2 \).

Two possible choices of second camera translation vector \( t \) (to positive scale)

\[ t_1 = u_3 \] or \( t_2 = -u_3 \), where \( u_3 \) is the last column of \( U \)

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

\[ \hat{P}'_1 = [R_1 \mid t_1], \hat{P}'_2 = [R_1 \mid t_2], \hat{P}'_3 = [R_2 \mid t_1], \text{ or } \hat{P}'_4 = [R_2 \mid t_2] \]

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

- Error in image point measurements

mapped to line $E^\top \hat{x}'$

measured point $\hat{x}$

projected point $\hat{P}X$

mapped to line $E\hat{x}$

measured point $\hat{x}'$

projected point $\hat{P}'X$
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 $E$ and corresponding points $\hat{x} \leftrightarrow \hat{x}'$, determine the “optimal” corrected corresponding points $\tilde{x} \leftrightarrow \tilde{x}'$ that exactly satisfy $\hat{x}_i^T E \hat{x}_i = 0$ as follows.

Determine the essential matrix $E_s$ in a special form.

First, $E_s = T'^{-T} E T^{-1}$, where

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

are the transformations that translate the points to the origin. Then, calculate its associated epipoles $\tilde{e}$ (i.e., the null space of $E_s$) and $\tilde{e}'$ (i.e., the null space of $E_s'^T$).

Then, scale $\tilde{e} = (\hat{e}_1, \hat{e}_2, \hat{e}_3)^T$ and $\tilde{e}' = (\hat{e}'_1, \hat{e}'_2, \hat{e}'_3)^T$ such that $\hat{e}_1^2 + \hat{e}_2^2 = 1$ and $\hat{e}'_1^2 + \hat{e}'_2^2 = 1$, i.e.,

$$\tilde{e} = \frac{1}{\sqrt{\hat{e}_1^2 + \hat{e}_2^2}} \hat{e} \quad \text{and} \quad \tilde{e}' = \frac{1}{\sqrt{\hat{e}'_1^2 + \hat{e}'_2^2}} \hat{e}'$$
“Optimal” correction

Last, $E_s = R'E_sR^T = R'T''^{-T}ET^{-1}R^T$, where

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

are the transformations that rotate the scaled epipoles to the $\hat{x}$-axis such that $R\hat{e} = (1, 0, \hat{e}_3)^T$ and $R'\hat{e}' = (1, 0, \hat{e}'_3)^T$. Note $E_s(1, 0, \hat{e}_3)^T = 0$ and $(1, 0, \hat{e}'_3)E_s = 0^T$, so $E_s$ has the form

$$E_s = \begin{bmatrix} ff'd & -f'c & -f'd \\ -fb & a & b \\ -fd & c & d \end{bmatrix}$$

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

$$\hat{\mathbf{e}}_s(t) = \begin{bmatrix} 0 \\ t \\ 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0 \\ f \end{bmatrix} = \begin{bmatrix} tf \\ 1 \\ -t \end{bmatrix} \quad \text{and} \quad \hat{\mathbf{e}}'_s(t) = \mathbf{E}_s \begin{bmatrix} 0 \\ t \\ 1 \end{bmatrix} = \begin{bmatrix} -f'(ct + d) \\ at + b \\ ct + d \end{bmatrix}$$

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

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

\[ s(t) = \frac{t^2}{1 + f^2t^2} + \frac{(ct + d)^2}{(at + b)^2 + f^2c^2(ct + d)^2} \]

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

\[ g(t) = t((at + b)^2 + f^2c^2(ct + d)^2)^2 - (ad - bc)(1 + f^2t^2)^2(at + b)(ct + d) = 0, \]
solve for \( t \). The 6 roots of \( g(t) \) are the maxima and minima of \( s(t) \). Note the minimum may also be at \( t = \infty \).

\[ \lim_{t \to \infty} s(t) = \frac{1}{f^2} + \frac{c^2}{a^2 + f^2c^2} \]

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 \( \hat{x}_s \) and \( \hat{x}'_s \) as the closest points on the transformed lines \( \hat{\ell}_s(t_{\min}) \) and \( \hat{\ell}'_s(t_{\min}) \), noting \( \hat{\ell}_s(t \to \infty) = (f, 0, -1)^T \) and \( \hat{\ell}'_s(t \to \infty) = (-f', a, c)^T \), to the origin and inverse transform the points back to their original coordinates.

\[ \hat{x} = T^{-1}R^T \hat{x}_s \text{ and } \hat{x}' = T'^{-1}R'^T \hat{x}'_s \]
Line orthogonal to hyperplane and through point

Hyperplane $\pi = (n^T, d)^T$, where $n$ is normal vector
Point $X'$ on line orthogonal to $\pi$ and through $X$ is $X$ translated by $n$

$$X' = \begin{bmatrix} 1 & n \\ 0^T & 1 \end{bmatrix} X$$

Line orthogonal to $\pi$ and through $X$ is join of $X$ and $X'$
Point on hyperplane closest to point

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

$$\mathbf{X}(\lambda) = \lambda \mathbf{X} + (1 - \lambda) \mathbf{X}'$$ and

$$\pi^T \mathbf{X}(\lambda) = 0,$$

solve for $\lambda$

$$\pi^T(\lambda \mathbf{X} + (1 - \lambda) \mathbf{X}') = 0$$

$$\lambda \pi^T \mathbf{X} + \pi^T \mathbf{X}' - \lambda \pi^T \mathbf{X}' = 0$$

$$\lambda (\pi^T \mathbf{X} - \pi^T \mathbf{X}') = -\pi^T \mathbf{X}'$$

$$\lambda = -\frac{\pi^T \mathbf{X}'}{\pi^T \mathbf{X} - \pi^T \mathbf{X}'}$$

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

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

• Corrected image points exactly satisfy epipolar constraint, so 3D lines intersect
Essential matrix
Triangulation

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

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

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

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

Backproject point \( \hat{x} \) to 3D line defined by two 3D points

\[
\hat{P}C = 0, \text{ where camera center } C \text{ is the null space of } \hat{P}
\]

\[
X = \hat{P}^+\hat{x}, \text{ where } \hat{P}^+ = \hat{P}^T(\hat{P}\hat{P}^T)^{-1}
\]

Intersection of 3D line and 3D plane

\[
X_\pi = \begin{bmatrix}
X_2(bY_1 + cZ_1 + dT_1) - X_1(bY_2 + cZ_2 + dT_2) \\
Y_2(aX_1 + cZ_1 + dT_1) - Y_1(aX_2 + cZ_2 + dT_2) \\
Z_2(aX_1 + bY_1 + dT_1) - Z_1(aX_2 + bY_2 + dT_2) \\
T_2(aX_1 + bY_1 + cZ_1) - T_1(aX_2 + bY_2 + cZ_2)
\end{bmatrix}, \text{ where } X_1 = C \text{ and } X_2 = X
\]
Triangulation

• Canonical cameras

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

$$X_\pi = \begin{bmatrix} \frac{d\hat{x}}{d\hat{x}} \\ \frac{d\hat{y}}{d\hat{y}} \\ \frac{d\hat{w}}{d\hat{w}} \\ -(a\hat{x} + b\hat{y} + c\hat{w}) \end{bmatrix} = \begin{bmatrix} d\hat{x} \\ -n^T\hat{x} \end{bmatrix}$$
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 $\mathbf{e}^\top \Sigma_{\mathbf{X}}^{-1} \mathbf{e}$, where $\mathbf{e} = \mathbf{X} - \hat{\mathbf{X}}$
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 \([\mathbf{\omega}]_\times\) corresponding to the rotation matrix \(\mathbf{R}\) is the matrix logarithm of \(\mathbf{R}\).

\[
[\mathbf{\omega}]_\times = \ln(\mathbf{R})
\]

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

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

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

where

$$(\mathbf{R} - \mathbf{I})\mathbf{v} = \mathbf{0},$$

solve for $\mathbf{v}$

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

$$\theta = \tan^{-1} \left( \frac{\sin(\theta)}{\cos(\theta)} \right),$$

where

$$\cos(\theta) = \frac{\text{Tr}(\mathbf{R}) - 1}{2}$$

and

$$\sin(\theta) = \frac{\mathbf{v}^\top \hat{\mathbf{v}}}{2},$$

where

$$\hat{\mathbf{v}} = \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$

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

$$\mathbf{\omega} = \left(1 - \frac{2\pi}{\|\mathbf{\omega}\|} \left\lfloor \frac{\|\mathbf{\omega}\| - \pi}{2\pi} \right\rfloor \right) \mathbf{\omega}$$

For very small rotations

Very small rotation if null space is more than 1 column

$$\mathbf{\omega} = \frac{1}{2} \hat{\mathbf{v}}$$
Angle-axis representation

• Angle-axis representation to rotation matrix

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

$$R = \exp([\omega]_\times)$$

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

where $\theta = ||\omega||$

For very small rotations

$$R = I + [\omega]_\times$$

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 \( H_{\mathbf{v}(\mathbf{x})} \) (i.e., \( H_{\mathbf{v}(\mathbf{x})}\mathbf{x} = (1, 0, \ldots, 0)^\top = e_1 \)), where

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

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

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

\[
H_{\mathbf{v}(\mathbf{x})} = I - 2 \frac{\mathbf{v}(\mathbf{x})\mathbf{v}(\mathbf{x})^\top}{\mathbf{v}(\mathbf{x})^\top \mathbf{v}(\mathbf{x})} \in \mathbb{R}^{n \times n}
\]

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

\[
\frac{\partial f(\mathbf{y})}{\partial \mathbf{y}} = \begin{bmatrix}
  0^\top \\
  I
\end{bmatrix} \in \mathbb{R}^{n \times (n-1)}
\]
Parameterization of the $n$-sphere

A mapping (unique for the half-sphere, which is fine since the sign of transformed $x$ does not change) that satisfies these criteria is

$$f(y) = (\cos(\|y\|), \text{sinc}(\|y\|)y^T)^T$$

A composite of the mapping and the transformation gives the parameterization of the $n$-sphere

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

with associated Jacobian

$$\frac{\partial x}{\partial y} = \frac{\partial x}{\partial f(y)} \frac{\partial f(y)}{\partial y}$$
$$\frac{\partial x}{\partial y} = H_v(x) \begin{bmatrix} 0^T \\ I \end{bmatrix}$$

(i.e., $\partial x/\partial y$ is $H_v(x)$ with the first column omitted)
Parameterization of a homogeneous vector

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

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

then, if $\|v\| > \pi$, normalized by

$$v = \left(1 - \frac{2\pi}{\|v\|} \left\lfloor \frac{\|v\| - \pi}{2\pi} \right\rfloor \right) v$$

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

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

Unit vector with nonnegative first element

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

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

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

where

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

and

$$
\frac{\partial \mathbf{b}}{\partial \mathbf{v}} = \begin{cases} \frac{1}{2} \mathbf{I} & \text{if } \|\mathbf{v}\| = 0 \\ \frac{1}{\sin \left( \frac{\|\mathbf{v}\|}{2} \right)} \left[ \mathbf{I} + \frac{1}{4\|\mathbf{v}\|} \frac{d \sin \left( \frac{\|\mathbf{v}\|}{2} \right)}{d \|\mathbf{v}\|} \mathbf{v} \right] \mathbf{v}^\top & \text{otherwise} \end{cases}
$$
Parameterization of a homogeneous vector

- Sinc function

The sinc function

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

The derivative is given by

\[
\frac{d\text{sinc}(x)}{dx} = \begin{cases} 
  0 & \text{if } x = 0 \\
  \frac{\cos(x)}{x} - \frac{\sin(x)}{x^2} & \text{otherwise}
\end{cases}
\]
Two view geometry

- Imaging a 3D scene

![Diagram of two views and corresponding points](image-url)
Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

Parameter vector\( (3 + 3 + 3n) \times 1 \)
\[
\begin{pmatrix}
\omega^T, t^T, \hat{X}_1^T, \hat{X}_2^T, \ldots, \hat{X}_n^T
\end{pmatrix}^T
\]

Mapping
\[
\begin{align*}
\hat{x}_i &= \hat{P}\hat{X}_i = [I \mid 0]\hat{X}_i \quad \forall i \\
\hat{x}'_i &= \hat{P}'\hat{X}_i = \exp([\omega']_x) | t' \\
&= \begin{bmatrix}
\hat{X}_i \\
\hat{Y}_i \\
\hat{Z}_i \\
\hat{T}_i
\end{bmatrix} = \exp([\omega']_x) \\
&= \begin{bmatrix}
\hat{X}_i \\
\hat{Y}_i \\
\hat{Z}_i \\
\hat{T}_i
\end{bmatrix} + \hat{T}_i t' \quad \forall i
\end{align*}
\]

Measurement vector\( 4n \times 1 \)
\[
\begin{pmatrix}
\hat{x}_1^T, \hat{x}_2^T, \ldots, \hat{x}_n^T, \hat{x}'_1^T, \hat{x}'_2^T, \ldots, \hat{x}'_n^T
\end{pmatrix}^T
\]

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

Cost
\[
\sum_i \left( \epsilon_i^T \Sigma_{\hat{x}_i}^{-1} \epsilon_i + \epsilon'_i^T \Sigma_{\hat{x}'_i}^{-1} \epsilon'_i \right), \text{ where } \epsilon_i = \hat{x}_i - \hat{x}_i \text{ and } \epsilon'_i = \hat{x}'_i - \hat{x}'_i
\]

Always use the parameterized homogeneous vector, not the input homogeneous vector or matrix.
Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

- **Jacobian**

\[
A'_i = \frac{\partial \hat{x}'_i}{\partial (\hat{\omega}'^T, \hat{t}'^T)} = \begin{bmatrix} \frac{\partial \hat{x}'_i}{\partial \hat{\omega}'} & \frac{\partial \hat{x}'_i}{\partial \hat{t}'} \end{bmatrix}
\]

where

\[
\frac{\partial \hat{x}'_i}{\partial \hat{t}'} = \frac{\partial \hat{x}'_i}{\partial \hat{t}'} H_{v(t')^T} \begin{bmatrix} 0^T \\ I \end{bmatrix}
\]

\[
B_i = \frac{\partial \hat{x}_i}{\partial \hat{X}_i} = \frac{\partial \hat{x}_i}{\partial \bar{X}_i} \frac{\partial \bar{X}_i}{\partial \hat{X}_i}
\]

\[
B'_i = \frac{\partial \hat{x}'_i}{\partial \hat{X}_i} = \frac{\partial \hat{x}'_i}{\partial \bar{X}_i} \frac{\partial \bar{X}_i}{\partial \hat{X}_i}
\]
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 $\mathbf{\hat{x}}$ under the normalized camera projection matrix $\hat{\mathbf{P}} = [\exp([\omega]_\times) | \mathbf{t}] = [\mathbf{R} | \mathbf{t}]$, where $\mathbf{R} = \exp([\omega]_\times)$, by

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

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

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

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

where $\mathbf{v'} = \exp([\omega]_\times)\mathbf{v} = \mathbf{R}\mathbf{v}$. Dehomogenizing the 2D point results in the mapping $\mathbf{X} \mapsto \mathbf{\hat{x}}$. 
Projection of a point under the normalized camera projection matrix

For this mapping

\[
\frac{\partial \hat{x}}{\partial \omega} = \frac{\partial \hat{x}}{\partial v'} \frac{\partial v'}{\partial \omega}
\]

where

\[
\frac{\partial \hat{x}}{\partial v'} = \begin{bmatrix} 1/\hat{w} & 0 & -\hat{x}/\hat{w} \\ 0 & 1/\hat{w} & -\hat{y}/\hat{w} \end{bmatrix}
\]

where $\hat{w} = v'_3 + t_3T$, $\frac{\partial v'}{\partial \omega} = \exp([\omega]_x) = R$,

\[
\frac{\partial \hat{x}}{\partial t} = \begin{bmatrix} T/\hat{w} & 0 & -\hat{x}T/\hat{w} \\ 0 & T/\hat{w} & -\hat{y}T/\hat{w} \end{bmatrix}
\]

where $\hat{w} = r^{3T}v + t_3T$, and

\[
\frac{\partial \hat{x}}{\partial X} = \frac{1}{\hat{w}} \begin{bmatrix} \hat{p}^{1T} - \hat{x}\hat{p}^{3T} \\ \hat{p}^{2T} - \hat{y}\hat{p}^{3T} \end{bmatrix}
\]

where $\hat{w} = \hat{p}^{3T}X$ and $\hat{p}^{iT}$ is the $i$th row of $\hat{P}$.
Angle-axis representation, rotate vector

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

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

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

where $\theta = ||\omega||$. For this rotation

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

where

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

and

$$
\frac{d\theta}{d\omega} = \frac{1}{\theta} \omega^\top \quad \text{and} \quad \frac{\partial \mathbf{v}'}{\partial \mathbf{v}} = \exp([\omega]_\times)
$$
Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

• Normal equations matrix

\[ U' = \sum_i A_i'^T \Sigma_{\hat{X}_i}^{-1} A_i' \quad 5 \times 5 \]

\[ V_i = B_i'^T \Sigma_{\hat{X}_i}^{-1} B_i + B_i'^T \Sigma_{\hat{X}_i'}^{-1} B_i' \quad 3 \times 3 \]

\[ W_i' = A_i'^T \Sigma_{\hat{X}_i}^{-1} B_i' \quad 5 \times 3 \]
Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

• Normal equations vector

\[
\left( \epsilon_{a'}, \epsilon_{b1}', \epsilon_{b2}', \ldots, \epsilon_{bn}' \right)^T
\]

where

\[
\epsilon_{a'} = \sum_i A_i'^T \Sigma_{\hat{x}_i}^{-1} \epsilon_i' \quad 5 \times 1
\]

\[
\epsilon_{b_i} = B_i'^T \Sigma_{\hat{x}_i}^{-1} \epsilon_i + B_i'^T \Sigma_{\hat{x}_i}' \epsilon_i' \quad 3 \times 1
\]

where \( \epsilon_i = \hat{x}_i - \hat{x}_i \) and \( \epsilon_i' = \hat{x}_i' - \hat{x}_i' \)
Nonlinear estimation of essential matrix using the Levenberg-Marquardt algorithm

Augmented normal equations

\[ s' = u'^* - \sum_i w'_i v'^*_{ii} w'_i^T \quad 5 \times 5 \]
\[ e' = \epsilon_{a'} - \sum_i w'_i v'^*_{ii} \epsilon_{b_i} \quad 5 \times 1 \]
\[ s' \delta_{a'} = e', \text{ solve for } \delta_{a'} \quad 5 \times 1 \]
\[ \delta_{b_i} = v'^*_{ii}^{-1} (\epsilon_{b_i} - w'_i^T \delta_{a'}) \quad 3 \times 1 \]

where \( u'^* = u' + \lambda I \) and \( v'_i = v_i + \lambda I \)

Candidate parameter vector

\[ \delta_{a'} = (\delta_{\omega'}^T, \delta_{\hat{\nu}'}^T)^T \]
\[ \hat{\omega}' = \hat{\omega}' + \delta_{\omega'} \]
\[ \hat{\nu}'_0 = H_{v'(\hat{\nu}')} f(\delta_{\hat{\nu}'}) \]
\[ \hat{X}_{i_0} = \hat{X}_i + \delta_{b_i} \]

Parameterization of \( n \)-sphere
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

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

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

Covariance propagation

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

If unknown, then assume to be identity
Up to 3D similarity transformation
3D geometric transformations

• Similarity

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

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

\[
\tilde{X}' = sR\tilde{X} + t, \text{ where } R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\
r_{21} & r_{22} & r_{23} \\
r_{31} & r_{32} & r_{33} \end{bmatrix} \in \text{SO}(3) \text{ and } t = (t_1, t_2, t_3)^T
\]

\[
\tilde{X}' = \begin{bmatrix} sR \\ t \end{bmatrix} \begin{bmatrix} \tilde{X} \\ 1 \end{bmatrix}
\]

\[
X' = \begin{bmatrix} sR \\ t \end{bmatrix} \begin{bmatrix} 0^T \\ 1 \end{bmatrix} X
\]

\[
X' = HSX, \text{ where } H_S = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix}
\]
3D similarity transformation
3D similarity transformation

Given \( n \geq 3 \) point correspondences \( \tilde{X}_i \leftrightarrow \tilde{X}'_i \)
\[ \tilde{X}'_i = sR\tilde{X}_i + t \forall i, \] solve for \( s, R, \) and \( t \)

1. Calculate the mean vector \( \mu_{\tilde{X}} \) (i.e., centroid) and total unbiased variance \( \sigma^2_{\tilde{X}} \) of the first set of 3D points

2. Calculate the mean vector \( \mu_{\tilde{X}'} \) (i.e., centroid) of the second set of 3D points

3. Compute the matrix

\[ S = \frac{1}{n-1} \sum_i \left( \tilde{X}'_i - \mu_{\tilde{X}'} \right) \left( \tilde{X}_i - \mu_{\tilde{X}} \right)^\top \]

and its singular value decomposition \( S = U\Sigma V^\top \), where \( \Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3) \) and \( \sigma = (\sigma_1, \sigma_2, \sigma_3)^\top \)
3D similarity transformation

4. The rotation matrix

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

such that \(\det(R) = +1\)

5. The scale

\[
s = \begin{cases} 
\frac{\sigma^T(1,1,-1)^T}{\sigma^2_x} & \text{if } \det(U) \det(V) < 0 \\
\frac{\text{Tr}(\Sigma)}{\sigma^2_x} & \text{otherwise}
\end{cases}
\]

such that \(\det(R) = +1\)

6. The translation vector \(t = \mu_{x'} - sR\mu_x\)
Next lecture

• Three view geometry
• Reading
  – Chapters 15 and 16