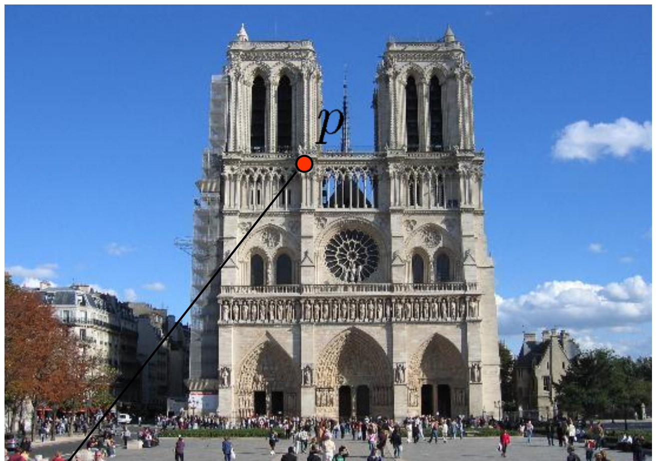
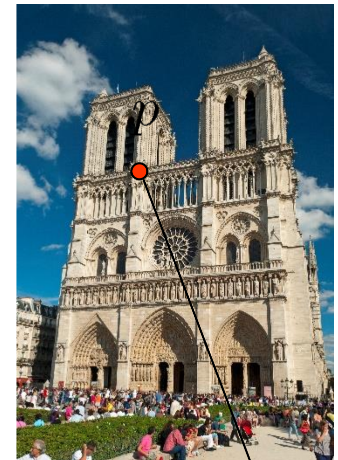
CS 395T
Lecture 7: Two-View Geometry

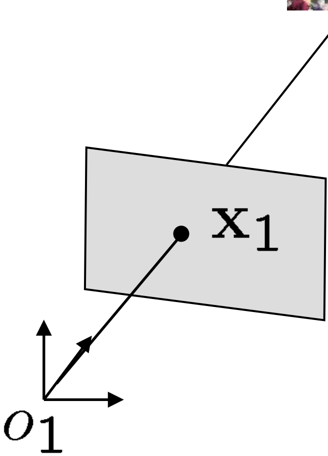
Qixing Huang

September 24th 2018

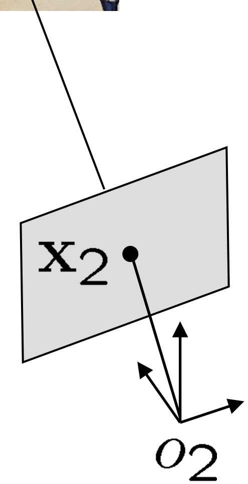
The problem







Given two views of the scene recover the unknown camera displacement and 3D scene structure



https://www.tripadvisor.com/Attraction_Review-g187147-d188679-Reviews-Notre_Dame_Cathedral-Paris_Ile_de_France.html

https://commons.wikimedia.org/wiki/File:Notre_Dame_de_Paris_Cathédrale_Notre-Dame_de_Paris_(6094168584).jpg

Pinhole camera model-review

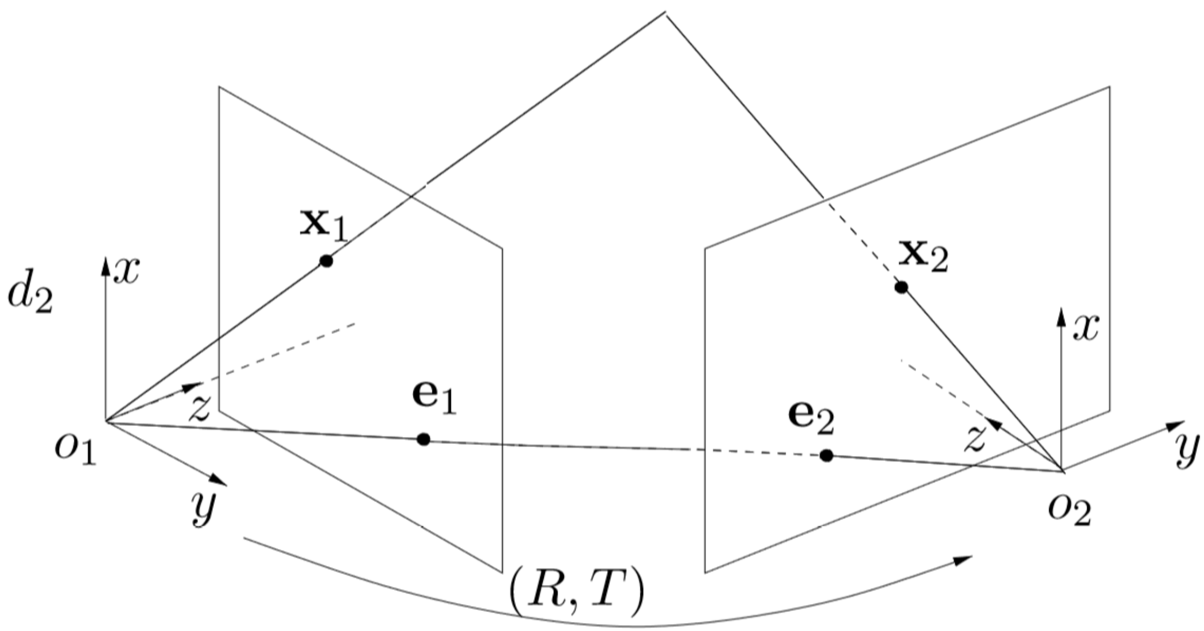
- 3D points $\mathbf{X} = [X, Y, Z, W]^T \in \Re^4, \quad (W = 1)$
- Image points $\mathbf{x} = [x, y, z]^T \in \Re^3, \quad (z = 1)$
- Perspective projection $\lambda \mathbf{x} = \mathbf{X}$

$$\lambda = Z \;\; x = \frac{X}{Z} \;\; y = \frac{Y}{Z}$$

- Rigid body motion $\Pi = [R, T] \in \Re^{3\times 4}$
- Rigid body motion + Projective projection

$$\lambda \mathbf{x} = \Pi \mathbf{X} = [R, T]\mathbf{X}$$

Two views

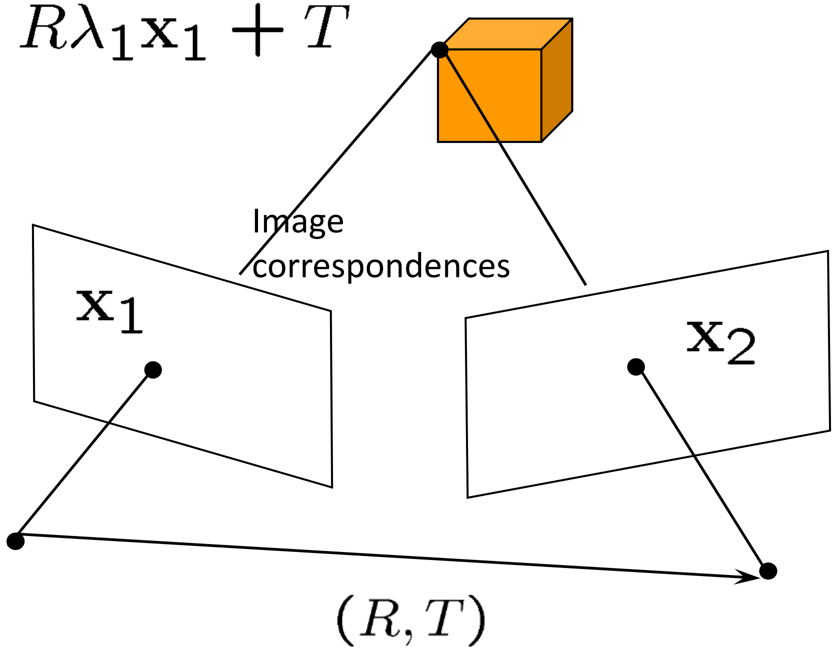


$$\lambda_2 \mathbf{x}_2 = R\lambda_1 \mathbf{x}_1 + T$$

Think about how you would solve this problem

Epipolar geometry

$$\lambda_2 \mathbf{x}_2 = R\lambda_1 \mathbf{x}_1 + T$$



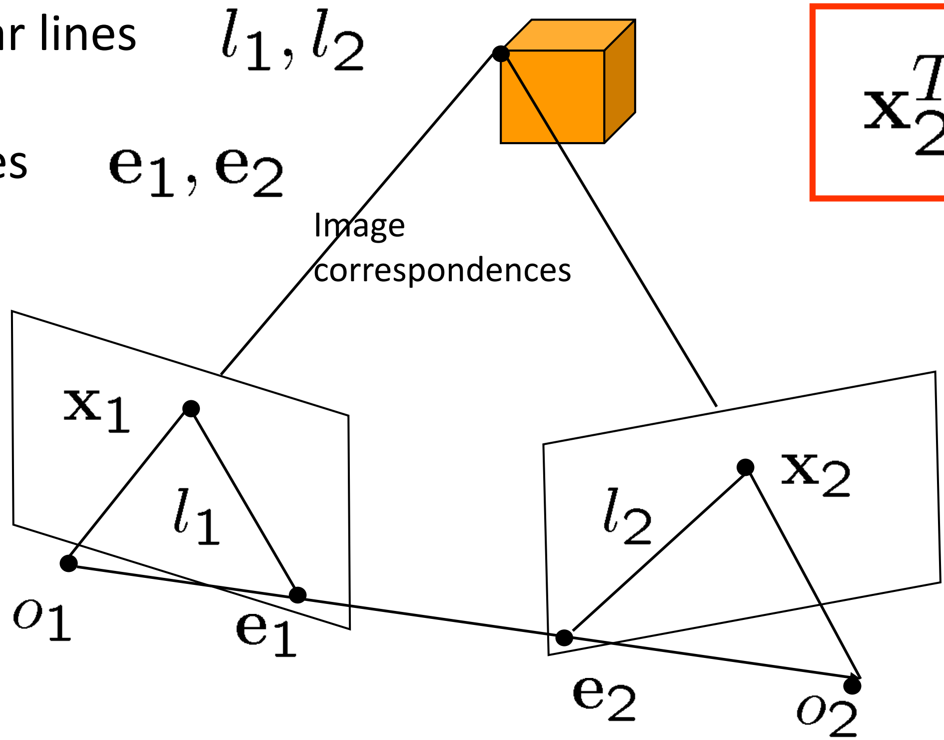
- Multiply both sides by the cross product of T [Longuet-Higgins ’81]:

$$\mathbf{x}_2^T \underbrace{\hat{T} R}_{E} \mathbf{x}_1 = 0$$

- Essential matrix $E = \hat{T} R$

Epipolar geometry

- Epipolar lines l_1, l_2
- Epipoles $\mathbf{e}_1, \mathbf{e}_2$



$$\boxed{\mathbf{x}_2^T E \mathbf{x}_1 = 0}$$

$$E = \hat{T} R$$

Properties (pay attention to geometric interpretations):

$$l_1 \sim E^T \mathbf{x}_2 \qquad l_i^T \mathbf{x}_i = 0 \qquad l_2 \sim E \mathbf{x}_1$$

$$E \mathbf{e}_1 = 0 \qquad l_i^T \mathbf{e}_i = 0 \qquad \mathbf{e}_2 E^T = 0$$

Characterization of the Essential Matrix

$$\mathbf{x}_2^T \hat{T} R \mathbf{x}_1 = 0$$

- Essential matrix $E = \hat{T} R$ Special 3x3 matrix

$$\mathbf{x}_2^T \begin{bmatrix} e_1 & e_2 & e_2 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \mathbf{x}_1 = 0$$

Theorem 5.1 (Characterization of the essential matrix). *A nonzero matrix $E \in \mathbb{R}^{3\times3}$ is an essential matrix* if and only if *E has a singular value decomposition (SVD): $E = U\Sigma V^T$ with*

$$\Sigma = diag\{\sigma, \sigma, 0\}$$

for some $\sigma \in \mathbb{R}_+$ and $U, V \in SO(3)$.

Characterization of the Essential Matrix

- Space of all Essential Matrices is 5 dimensional
  - 3 Degrees of Freedom - Rotation
  - 2 Degrees of Freedom – Translation (up to scale!)

- Decompose essential matrix into *R, T*

$$\mathbf{x}_2^T \hat{T} R \mathbf{x}_1 = 0$$

- Given feature correspondences, a straightforward approach is to find such Rotation and Translation that the epipolar error is minimized – nonlinear optimization

$$min_E \sum_{j=1}^{n} \mathbf{x}_2^{jT} E \mathbf{x}_1^j$$

Pose recovery from the Essential Matrix

Essential matrix $$E = \hat{T}R$$

Theorem 5.2 (Pose recovery from the essential matrix). *There exist exactly two relative poses* (R, T) *with* $R \in SO(3)$ *and* $T \in \mathbb{R}^3$ *corresponding to a non-zero essential matrix* $E = U\Sigma V^T$

$$\boxed{\begin{aligned} (\widehat{T}_1, R_1) = &\quad (UR_Z(+\tfrac{\pi}{2})\Sigma U^T, UR_Z^T(+\tfrac{\pi}{2})V^T), \\ (\widehat{T}_2, R_2) = &\quad (UR_Z(-\tfrac{\pi}{2})\Sigma U^T, UR_Z^T(-\tfrac{\pi}{2})V^T). \end{aligned}}$$

$$R_Z\left(+\frac{\pi}{2}\right) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Estimating essential matrix

- The eight-point linear constraint
  - Essential vector

$$E = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \longrightarrow \mathbf{e} = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9]^T \in \mathbb{R}^9$$

  - Vectorized correspondence

$$\mathbf{x}_1 = [x_1, y_1, z_1]^T \in \mathbb{R}^3 \text{ and } \mathbf{x}_2 = [x_2, y_2, z_2]^T \in \mathbb{R}^3$$

$$\downarrow$$

$$\mathbf{a} = [x_2x_1, x_2y_1, x_2z_1, y_2x_1, y_2y_1, y_2z_1, z_2x_1, z_2y_1, z_2z_1]^T \in \mathbb{R}^9$$

  - Linear constraint

$$\mathbf{a}^T\mathbf{e} = 0$$

Estimating essential matrix

- The eight-point linear constraint
  - Multiple correspondences

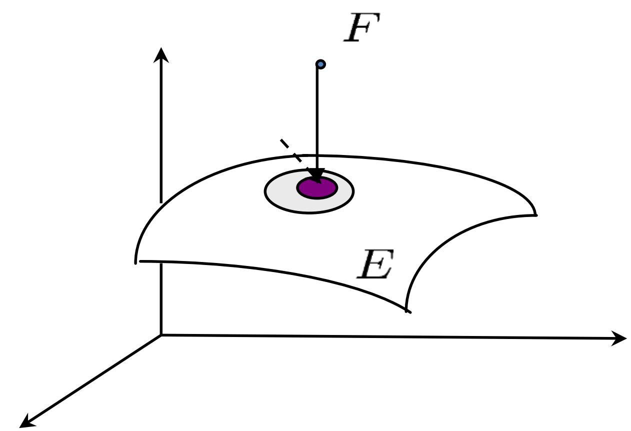
$$A\mathbf{e} = 0.$$

  - More than 8 ideal correspondences
  - Due to noise, choose the eigenvector of $A^T A$ that correspondences to the smallest eigenvalue:

$$\min_{\mathbf{e}} \frac{||A\mathbf{e}||^2}{||\mathbf{e}||^2}$$

Projection to the space of essential matrices

Theorem 5.3 (Projection onto the essential space). *Given a real matrix $F \in \mathbb{R}^{3\times 3}$ with a SVD: $F = U\,diag\{\lambda_1, \lambda_2, \lambda_3\}V^T$ with $U, V \in SO(3)$, $\lambda_1 \geq \lambda_2 \geq \lambda_3$, then the essential matrix $E \in \mathcal{E}$ which minimizes the error $\|E - F\|_f^2$ is given by $E = U\,diag\{\sigma, \sigma, 0\}V^T$ with $\sigma = (\lambda_1 + \lambda_2)/2$. The subscript f indicates the Frobenius norm.*



There is a general theorem that is widely used in low-rank matrix recovery

$$\min_{X,\mathrm{rank}(X)=r} \|A - X\|_{\mathcal{F}}^2$$

$$X = U_r \Sigma_r V_r^T$$

The eight-point method

Algorithm 5.1 (The eight-point algorithm). *For a given set of image correspondences* $(\mathbf{x}_1^j, \mathbf{x}_2^j)$, $j = 1, \ldots, n$ $(n \geq 8)$, *this algorithm finds* $(R, T) \in SE(3)$ *which solves*

$$\mathbf{x}_2^{jT} \widehat{T} R \mathbf{x}_1^j = 0, \quad j = 1, \ldots, n.$$

1. **Compute a first approximation of the essential matrix**
   Construct the $A \in \mathbb{R}^{n \times 9}$ *from correspondences* $\mathbf{x}_1^j$ *and* $\mathbf{x}_2^j$ *as in (6.21), namely.*

   $$\mathbf{a}^j = [x_2^j x_1^j, x_2^j y_1^j, x_2^j z_1^j, y_2^j x_1^j, y_2^j y_1^j, y_2^j z_1^j, z_2^j x_1^j, z_2^j y_1^j, z_2^j z_1^j]^T \in \mathbb{R}^9.$$

   Find the vector $\mathbf{e} \in \mathbb{R}^9$ *of unit length such that* $\|A\mathbf{e}\|$ *is minimized as follows: compute the SVD* $A = U_A \Sigma_A V_A^T$ *and define* $\mathbf{e}$ *to be the* 9^{th} *column of* V_A*. Rearrange the 9 elements of* $\mathbf{e}$ *into a square* 3×3 *matrix* E *as in (5.8). Note that this matrix will in general* not *be an essential matrix.*

The eight-point method

2. **Project onto the essential space**
   Compute the Singular Value Decomposition of the matrix E recovered from data to be

$$E = U\,diag\{\sigma_1, \sigma_2, \sigma_3\}V^T$$

   where $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq 0$ and $U, V \in SO(3)$. In general, since E may not be an essential matrix, $\sigma_1 \neq \sigma_2$ and $\sigma_3 > 0$. Compute its projection onto the essential space as $U\Sigma V^T$, where $\Sigma = diag\{1, 1, 0\}$.

3. **Recover displacement from the essential matrix**
   Define the diagonal matrix Σ to be Extract R and T from the essential matrix as follows:

$$R = UR_Z^T\left(\pm\frac{\pi}{2}\right)V^T, \quad \widehat{T} = UR_Z\left(\pm\frac{\pi}{2}\right)\Sigma U^T.$$

3D structure recovery

$$\lambda_2 \mathbf{x}_2 = R\lambda_1 \mathbf{x}_1 + \gamma T$$

- Eliminate one of the scale's

$$\lambda_1^j \widehat{\mathbf{x}_2^j} R \mathbf{x}_1^j + \gamma \widehat{\mathbf{x}_2^j} T = 0, \quad j = 1, 2, \ldots, n$$

- Solve LLSE problem

$$M^j \bar{\lambda}^j \doteq \left[\widehat{\mathbf{x}_2^j} R \mathbf{x}_1^j, \ \widehat{\mathbf{x}_2^j} T \right] \begin{bmatrix} \lambda_1^j \\ \gamma \end{bmatrix} = 0$$

If the configuration is non-critical, the Euclidean structure of then points and motion of the camera can be reconstructed up to a universal scale.

Issues to take care of

- Infinitesimal viewpoint change

- Enough parallax

$$E = 0 \Leftrightarrow T = 0$$

- Positive depth constraint
  - Four potential solutions, i.e., E and –E, and each of leads to two solutions

- General position requirement

Explicit formulation of uncertainty

Constrained optimization

$$\hat{R}, \hat{T} = \arg\min \sum_{j=1}^{n} \sum_{i=1}^{2} \|w_i^j\|_2^2$$

subject to

$$\begin{cases} \tilde{\mathbf{x}}_i^j = \mathbf{x}_i^j + {w_i}^j, \quad i = 1, 2, j = 1, \ldots, n \\ \mathbf{x}_2^{jT} \widehat{T} R \mathbf{x}_1^j = 0, \quad j = 1, \ldots, n \\ \mathbf{x}_1^{jT} e_3 = 1, \quad j = 1, \ldots, n \\ \mathbf{x}_2^{jT} e_3 = 1, \quad j = 1, \ldots, n \\ R \in SO(3) \\ T \in \mathbb{S}^2 \end{cases}$$

We will talk about how to solve this later