GAMES
Structure-From-Motion

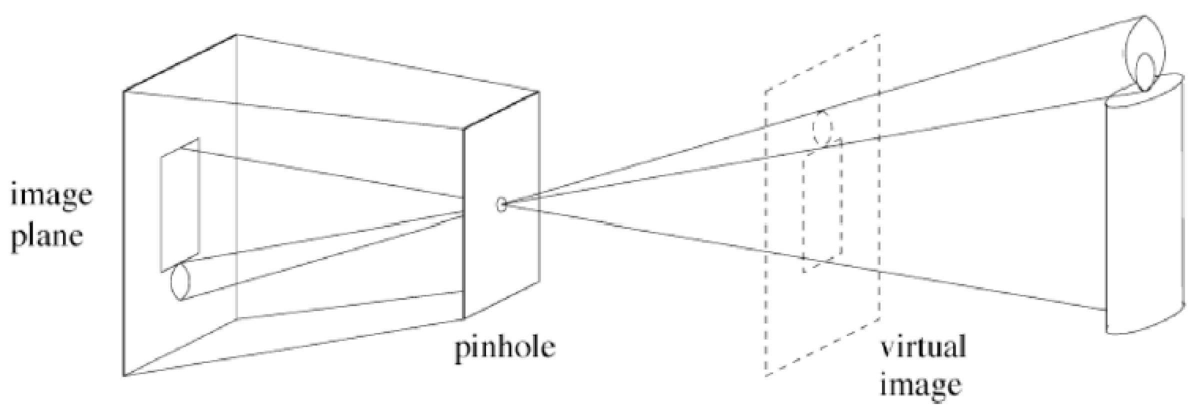
Qixing Huang

July 30th 2021

Projection and Two-View Geometry

Pinhole camera

- The dominant image formation model in computer vision
- A pinhole camera is a box in which one wall has a small hole
- Exactly one ray from each point in the scene passes through the pinhole and hits the wall opposite to it
- The inversion of the image is corrected for by considering a virtual image on the opposite side of the pinhole



Mathematical model under this idealized camera

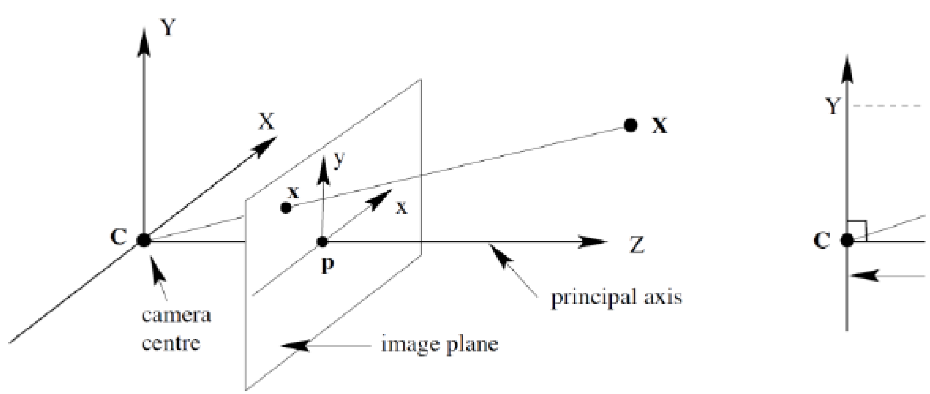
- It is clear that the camera is given by a perspective projection that maps the 3D space to a 2D plane

$$(X, Y, Z)^\top \xrightarrow{Projection} (x, y)^\top$$

- The equations of perspective projections are given by

$$x = f\frac{X}{Z} \qquad y = f\frac{Y}{Z}$$

f is the focal length of the camera, i.e., the distance between the image plane and the pinhole



Homogeneous coordinates

- The representation of the image point **x** = (x,y) is referred to as the inhomogeneous representation of the point **x**

- The homogeneous representation of a point x is given by **x** = (x, y, 1). In fact, the homogeneous representation of a point maps it to an entire class of set of points:

  $(x, y) \leftrightarrow (\lambda x, \lambda y, \lambda), \qquad \forall \lambda \neq 0$ In particular, $(x/z, y/z) \leftrightarrow (x, y, z)$

- Homogeneous coordinates encode the invariance of all points along a line and its projection

Examples

- The equation of a line *ax+by+cz = 0* can be rewritten using homogeneous coordinates

$$\boldsymbol{x}^\top \boldsymbol{l} = 0, \quad \text{where} \quad \boldsymbol{l} = (a, b, c)^\top$$

- The general conic in 3 dimensions is given by

$$ax^2 + bxy + cy^2 + dxz + eyz + fz^2 = 0$$

which can be written using 2D homogeneous coordinates as

$$\boldsymbol{x}^\top C \boldsymbol{x} = 0 \quad \text{where} \quad C = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

Points at infinity

- In R^2, all pairs of lines intersect except for the ones that are parallel

- In P^2, all pairs of lines intersect, and parallel lines intersect in points of infinity and these points have the form $(x,y,0)^\top$

- Consider the two lines given by

$$\begin{aligned} \boldsymbol{l}_1 &= (a_1, b_1, c_1)^\top \\ \boldsymbol{l}_2 &= (a_2, b_2, c_2)^\top \end{aligned}$$

- The intersection of these two lines is given by

$$\boldsymbol{x} = \boldsymbol{l}_1 \times \boldsymbol{l}_2$$

Intersection of parallel lines

- Given a line $\boldsymbol{l}_1 = (a, b, c)^\top$, a line parallel to it is given by $\boldsymbol{l}_2 = (a, b, c')^\top$

- The intersection is now given by

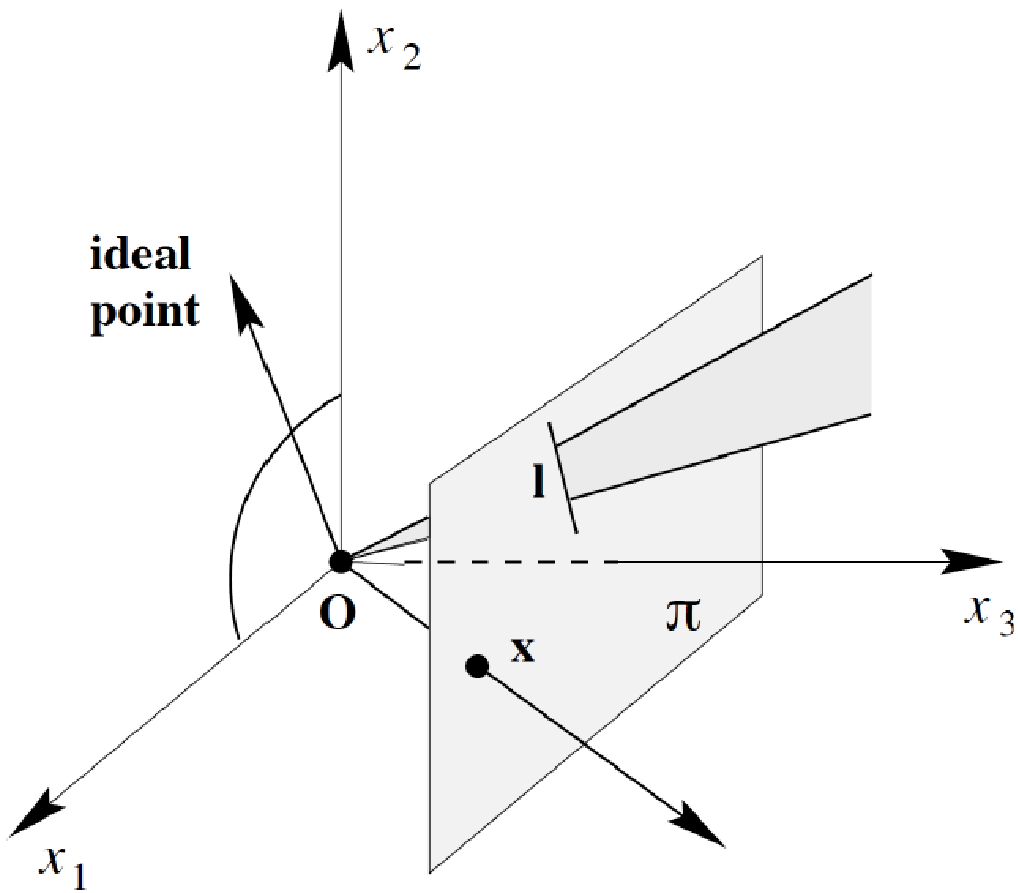
$$\begin{aligned} \boldsymbol{l}_1 \times \boldsymbol{l}_2 &= (bc' - cb, ac - ac', 0)^\top \\ &= (c - c')(b, -a, 0)^\top \\ &\sim (b, -a, 0)^\top \end{aligned}$$

Duality

- In P^2, points and lines are dual of each other
- The point of intersection of two lines is their cross product. Likewise, the line passing through any two points is given by their cross product $\boldsymbol{l} = \boldsymbol{x}_1 \times \boldsymbol{x}_2$
- The definition of points at infinity leads us to the definition of the line at infinity $\boldsymbol{l}_\infty$
- Consider two points at infinity $\boldsymbol{x}_1 = (x_1, y_1, 0)^\top$ and $\boldsymbol{x}_2 = (x_2, y_2, 0)^\top$
- The line passing through these two points is given by

$$\begin{aligned} \boldsymbol{l}_\infty &= \boldsymbol{x}_1 \times \boldsymbol{x}_2 \\ &= (0, 0, x_1 y_2 - y_1 x_2)^2 \\ &\sim (0, 0, 1)^\top \end{aligned}$$

A model for P^2 in R^3



Hartley&Zisserman

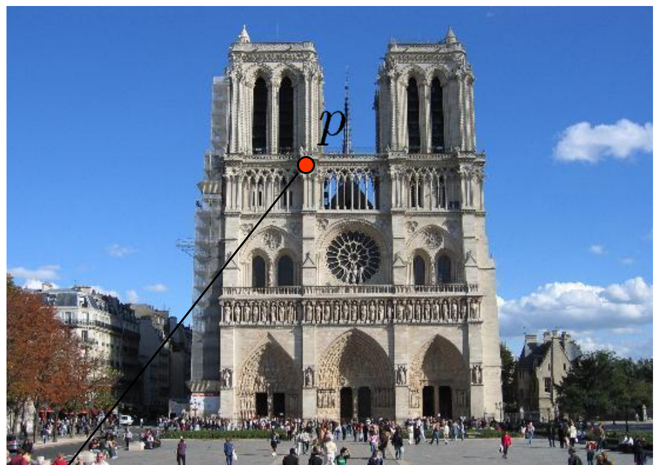
Intrinsic/Extrinsic Parameters of a Camera

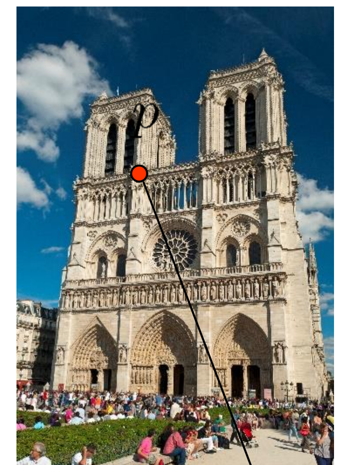
- The following equation maps the real world point X_0 in *homogeneous coordinates to its projection x' also in homogeneous coordinates*

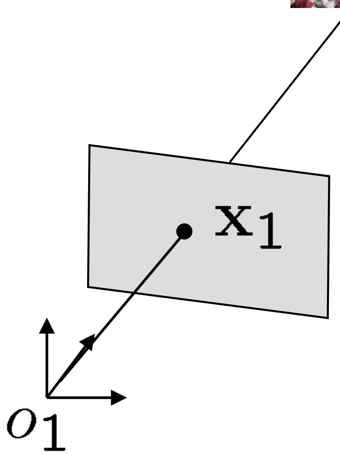
$$\underbrace{\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}}_{\boldsymbol{x}'} = \underbrace{\begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\Pi_0} \underbrace{\begin{bmatrix} R & \boldsymbol{T} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_{g} \underbrace{\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}}_{\boldsymbol{X}_0}$$

Intrinsic parameters (→ K) canonical projection matrix (→ Π_0) Extrinsic parameters (→ g)

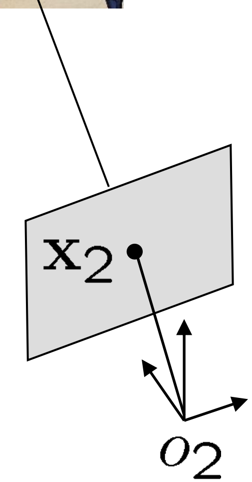
The problem







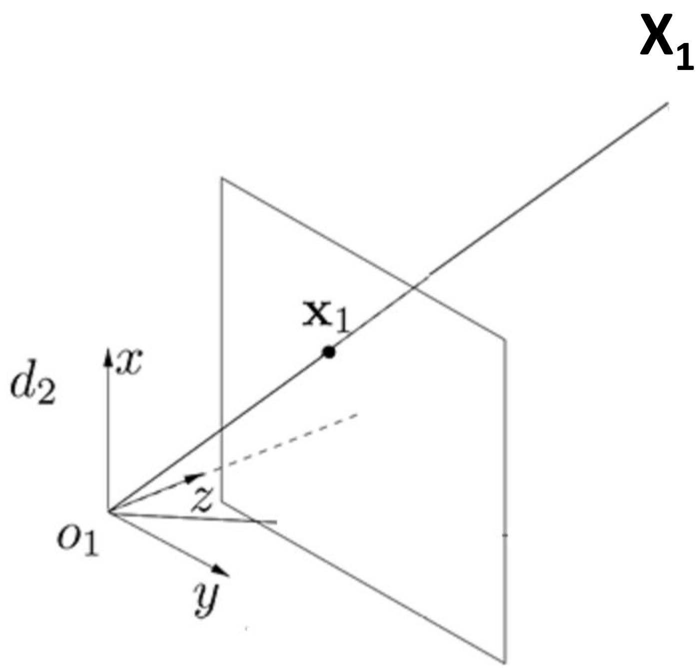
Given two views of the scene recover the unknown camera displacement and 3D scene structure



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

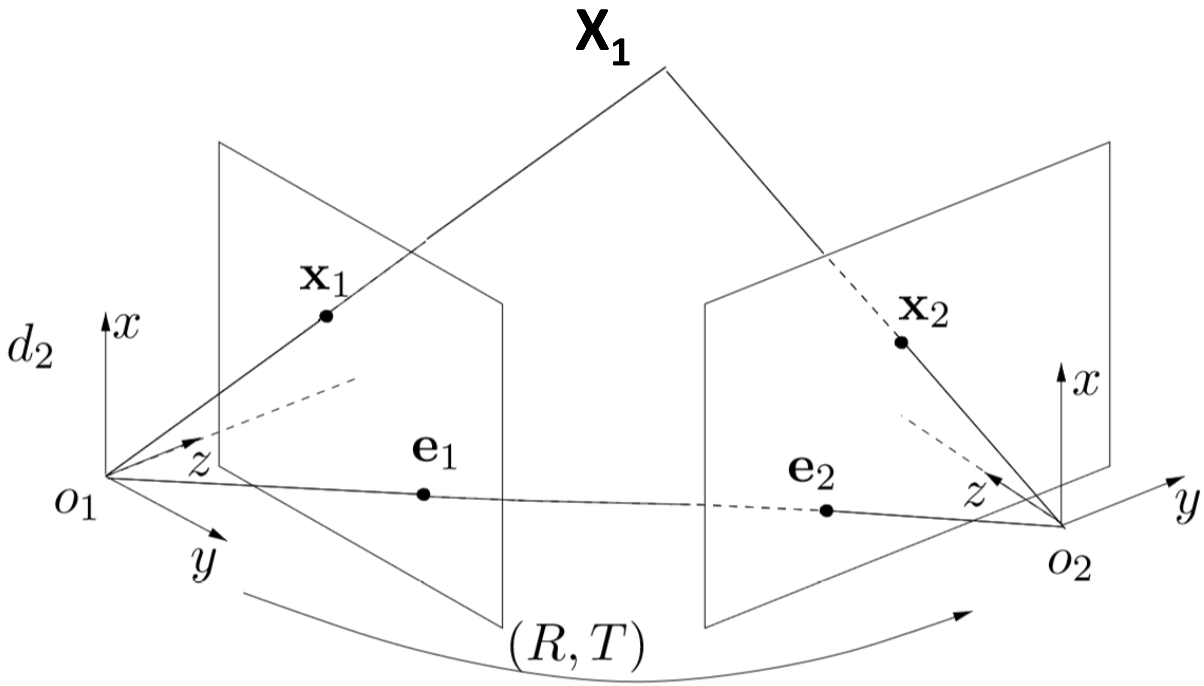
https://commons.wikimedia.org/wiki/File:Notre_Dame_de_Paris_Cathédrale_Notre-Dame_de_Paris_(6094168584).jpg

One View



$$\lambda_1 x_1 = X_1$$

Two views

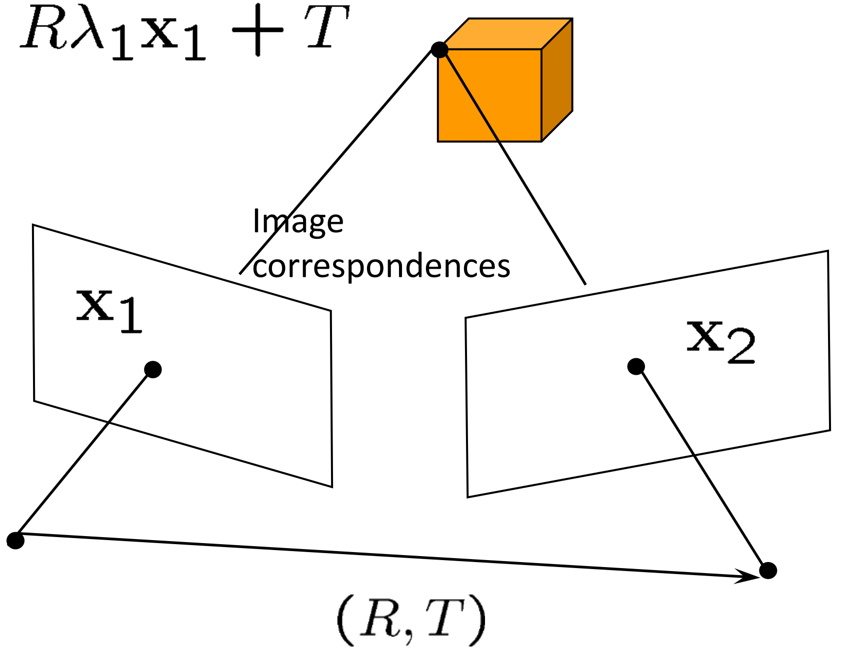


$$\lambda_2 \mathbf{x}_2 = R \lambda_1 \mathbf{x}_1 + T$$

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]:

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

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

Mathematical Derivation

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

$$T \times (\lambda_2 \boldsymbol{x}_2) = T \times (R\lambda_1 \boldsymbol{x}_1 + T)$$

$$\lambda_2 T \times \boldsymbol{x}_2 = \lambda_1 (T \times R)\boldsymbol{x}_1 + T \times T$$

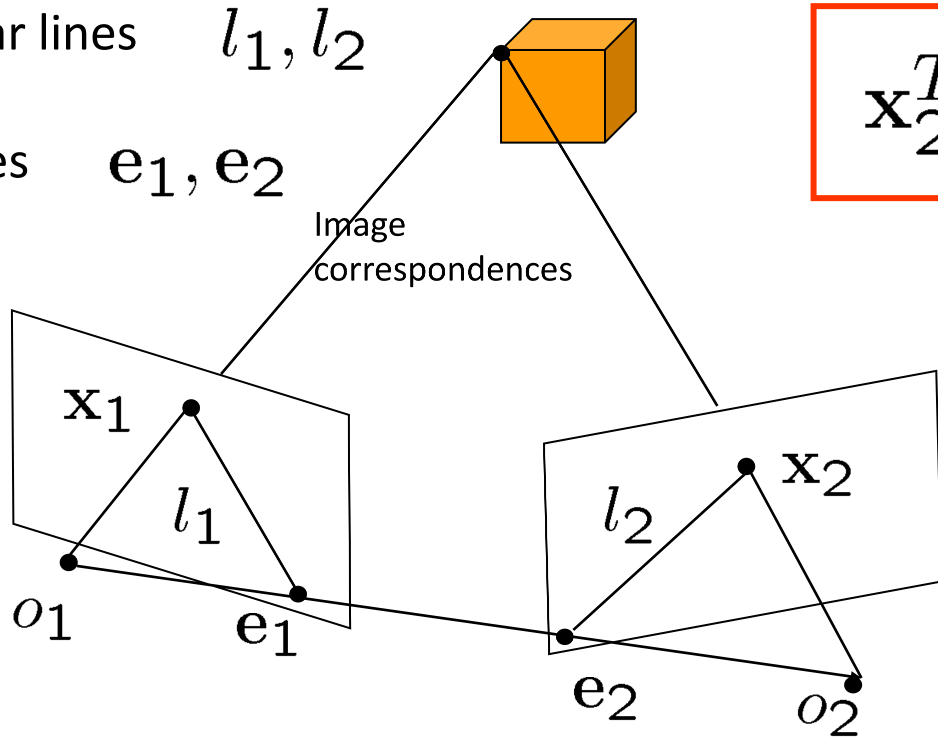
$$\lambda_2 T \times \boldsymbol{x}_2 = \lambda_1 (T \times R)\boldsymbol{x}_1$$

$(T \times R)\boldsymbol{x}_1$ is perpendicular to $\boldsymbol{x}_2$

$$\boldsymbol{x}_2^T (T \times R)\boldsymbol{x}_1 = 0$$

Epipolar geometry

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



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

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

Properties (pay attention to geometric interpretations):

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

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

Singular-Value Decomposition

Theorem *If A is a real $m \times n$ matrix then there exist orthogonal matrices*

$$U = \begin{bmatrix} \mathbf{u}_1 & \cdots & \mathbf{u}_m \end{bmatrix} \in \mathcal{R}^{m \times m}$$
$$V = \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_n \end{bmatrix} \in \mathcal{R}^{n \times n}$$

such that

$$U^T AV = \Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_p) \in \mathcal{R}^{m \times n}$$

where $p = \min(m, n)$ *and* $\sigma_1 \geq \ldots \geq \sigma_p \geq 0$. *Equivalently,*

$$A = U \Sigma V^T \ .$$

The SVD reveals a great deal about the structure of a matrix. If we define r by

$$\sigma_1 \geq \ldots \geq \sigma_r > \sigma_{r+1} = \ldots = 0 \ ,$$

that is, if σ_r is the smallest nonzero singular value of A, then

$$\mathrm{rank}(A) = r$$

Check Wikipedia for more details
https://en.wikipedia.org/wiki/Singular_value_decomposition

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 $E \in \mathbb{R}^{3\times 3}$ 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 $U, V \in SO(3)$.

[Ma et al.
Invitation to 3D Vision]
See notes for details]

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* $E = U\Sigma V^T$

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

Again, please either refer to the Invitation to 3D vision book or the course notes

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

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

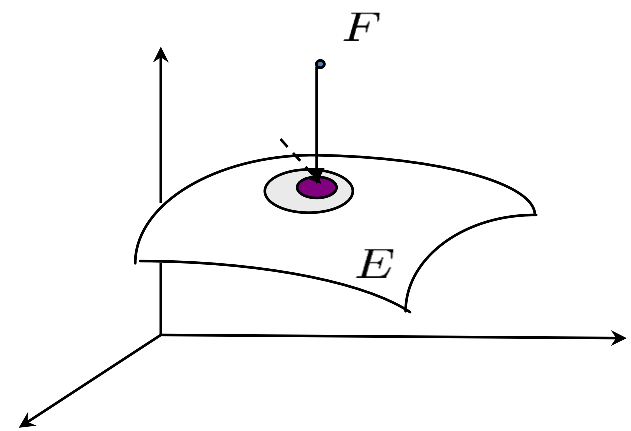
$$Ae = 0.$$

  - 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:

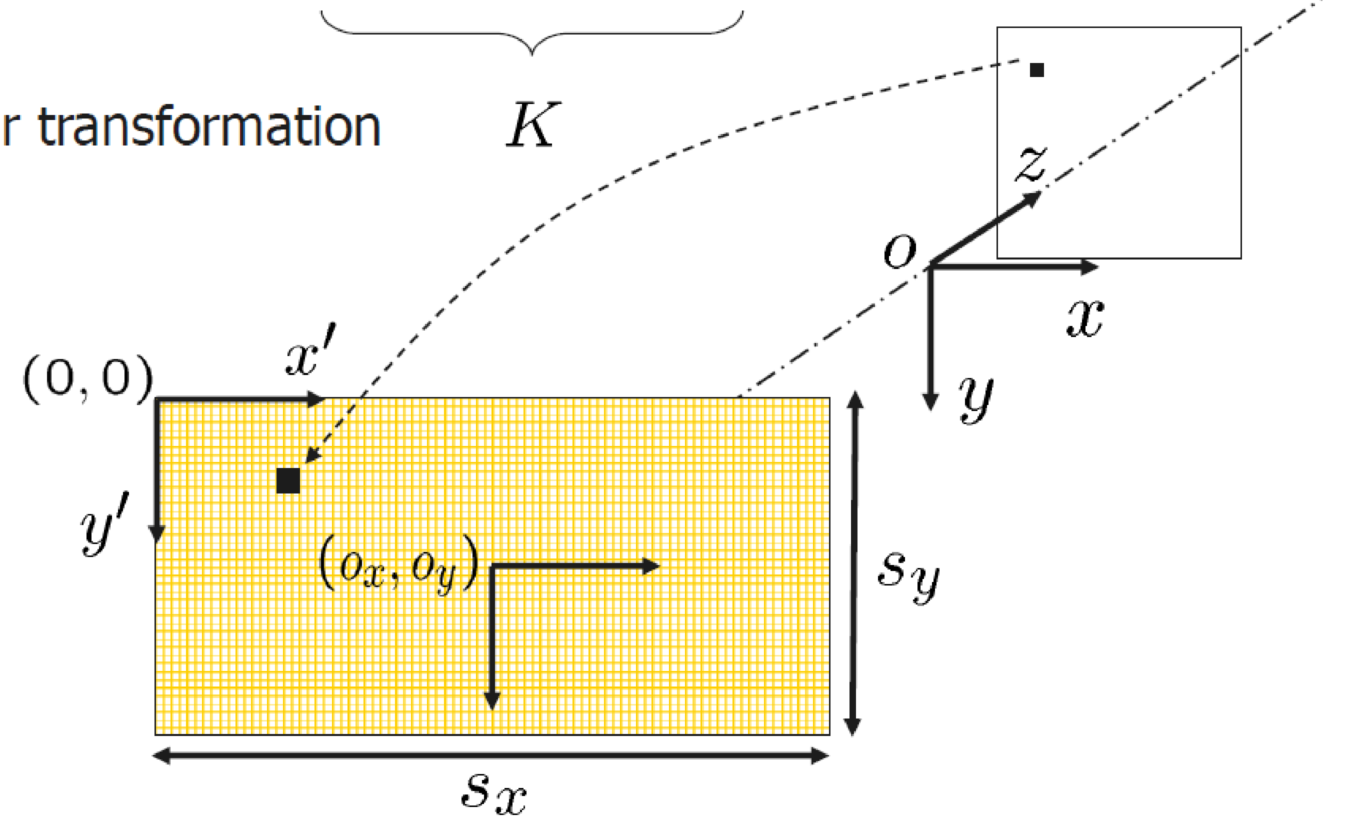
$$R = UR_Z^T\left(\pm\frac{\pi}{2}\right)V^T, \quad \widehat{T} = UR_Z\left(\pm\frac{\pi}{2}\right)\Sigma U^T.$$

Camera Calibration

Uncalibrated Camera – Intrinsic Parameters are unknown

$$\boldsymbol{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K\boldsymbol{x} = \underbrace{\begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}}_{K} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Linear transformation K



Overview

- Calibration with a rig (Checkborad for example)

- Uncalibrated epipolar geometry

- Ambiguities in image formation

- Stratified reconstruction

Uncalibrated Camera Using Homogeneous Coordinates

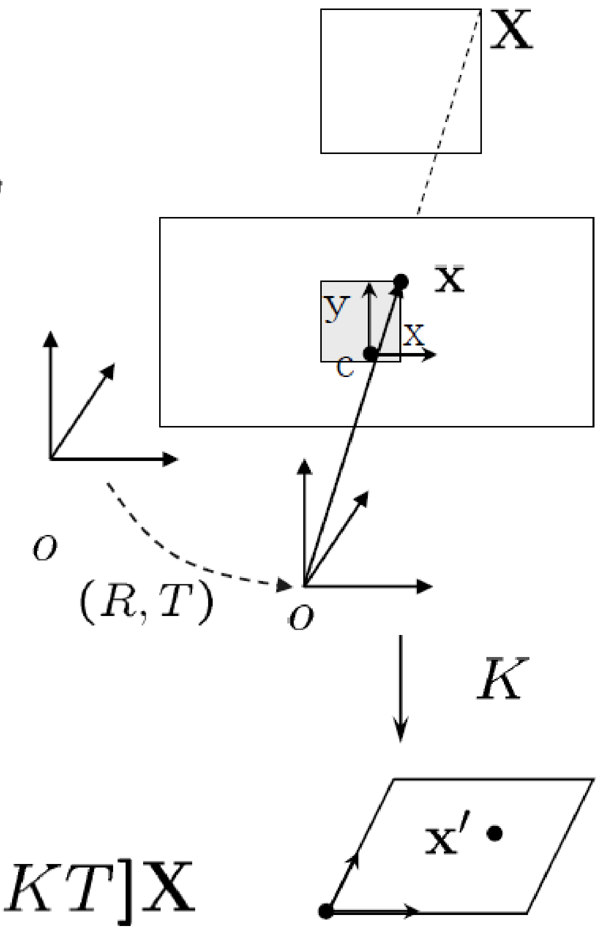
$$\mathbf{X} = [X, Y, Z, W]^T \in \mathbb{R}^4, \quad (W = 1)$$

Last Lecture:

- Image plane coordinates $\mathbf{x} = [x, y, 1]^T$
- Camera extrinsic parameters $g = (R, T)$
- Perspective projection $\lambda \mathbf{x} = [R, T]\mathbf{X}$

This Lecture:

- Pixel coordinates $\mathbf{x}' = K\mathbf{x}$
-
- Projection matrix $\lambda \mathbf{x}' = \Pi \mathbf{X} = [KR, KT]\mathbf{X}$



Calibration with a Rig

Use the fact that both 3-D and 2-D coordinates of feature points on a pre-fabricated object (e.g., a cube) are known.

Calibration with a Rig

- Given 3-D coordinates on known object $\mathbf{X}$

$$\lambda \mathbf{x}' = [KR, KT]\mathbf{X} \Rightarrow \lambda \mathbf{x}' = \Pi \mathbf{X}$$

$$\lambda \begin{bmatrix} x^i \\ y^i \\ 1 \end{bmatrix} = \begin{bmatrix} \pi_1^T \\ \pi_2^T \\ \pi_3^T \end{bmatrix} \begin{bmatrix} X^i \\ Y^i \\ Z^i \\ 1 \end{bmatrix}$$

- Eliminate unknown scales

$$\begin{aligned} x^i(\pi_3^T \mathbf{X}) &= \pi_1^T \mathbf{X}, \\ y^i(\pi_3^T \mathbf{X}) &= \pi_2^T \mathbf{X} \end{aligned}$$

Calibration with a Rig

- Recover projection matrix $\Pi = [KR, KT] = [R', T']$

$$\Pi^s = [\pi_{11}, \pi_{21}, \pi_{31}, \pi_{12}, \pi_{22}, \pi_{32}, \pi_{13}, \pi_{23}, \pi_{33}, \pi_{14}, \pi_{24}, \pi_{34}]^T$$

$$\min \|M\Pi^s\|^2 \quad \text{subject to} \quad \|\Pi^s\|^2 = 1$$

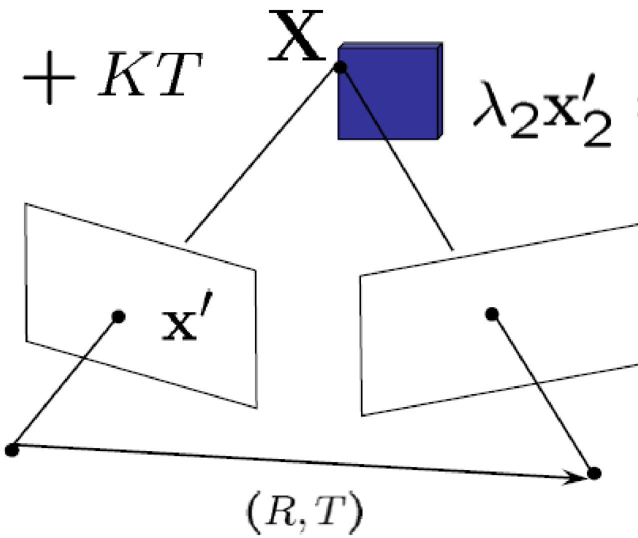
Again singular value decomposition

- Factor the KR into $R \in SO(3)$ and K using QR decomposition
- Solve for translation $T = K^{-1}T'$

Uncalibrated Epipolar Geometry
(not required)

Uncalibrated Epipolar Geometry

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



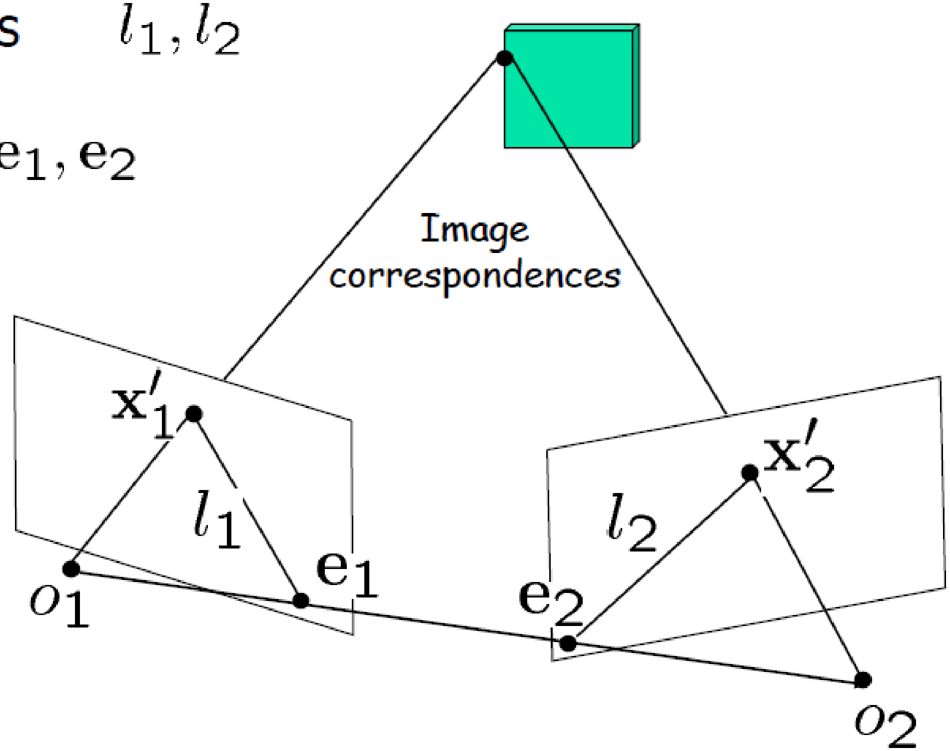
$$\lambda_2 \mathbf{x}'_2 = K R K^{-1} \lambda_1 \mathbf{x}'_1 + T'$$

- Epipolar constraint $\mathbf{x}'^{T}_{2} \underbrace{K^{-T} \hat{T} R K^{-1}} \mathbf{x}'_1 = 0$
-
- Fundamental matrix $F = K^{-T} \hat{T} R K^{-1}$
-
- Equivalent forms of $F = K^{-T} \hat{T} R K^{-1} = \hat{T}' K R K^{-1}$

Properties of the Fundamental Matrix

$$\mathbf{x}'^{T}_{2} F \mathbf{x}'_{1} = 0$$

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



$$l_1 \sim F^T \mathbf{x}'_2 \qquad l_i^T \mathbf{x}'_i = 0 \qquad l_2 \sim F \mathbf{x}'_1$$

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

Properties of the Fundamental Matrix

Remark 6.1. *Characterization of the fundamental matrix. A non-zero matrix* $F \in \mathbb{R}^{3\times 3}$ *is a fundamental matrix if* F *has a singular value decomposition (SVD):* $E = U\Sigma V^T$ *with*

$$\Sigma = diag\{\sigma_1, \sigma_2, 0\}$$

for some $\sigma_1, \sigma_2 \in \mathbb{R}_+$.

There is little structure in the matrix F except that

$$\det(F) = 0$$

Estimating Fundamental Matrix

- Find such F that the epipolar error is minimized

$$\min_F \sum_{j=1}^{n} (\boldsymbol{x}'_{2,j}{}^T F \boldsymbol{x}'_{1,j})^2 \quad s.t. \quad \|F\|_{\mathcal{F}}^2 = 1$$

- Fundamental matrix can be estimated up to scale
- Denote $\mathbf{a} = \mathbf{x}'_1 \otimes \mathbf{x}'_2$

$\mathbf{a} = [x_1x_2, x_1y_2, x_1z_2, y_1x_2, y_1y_2, y_1z_2, z_1x_2, z_1y_2, z_1z_2]^T$

$F^s = [f_1, f_4, f_7, f_2, f_5, f_8, f_3, f_6, f_9]^T$

- Rewrite $\mathbf{a}^T F^s = 0$

$$\min_{F^s} \|AF^s\|^2 \quad s.t. \quad \|F^s\|^2 = 1$$

Two view linear algorithm – 8-point algorithm

- Solve the LLSE problem:

$$\min_F \sum_{j=1}^{n} ({\boldsymbol{x}'_{2,j}}^T F \boldsymbol{x}'_{1,j})^2 \quad s.t. \quad \|F\|_{\mathcal{F}}^2 = 1$$

- Solution eigenvector associated with smallest eigenvalue of $A^T A$

- Compute SVD of F recovered from data

$$F = U \Sigma V^T \quad \Sigma = diag(\sigma_1, \sigma_2, \sigma_3)$$

- Project onto the essential manifold:

$$\Sigma' = diag(\sigma_1, \sigma_2, 0) \quad F = U \Sigma' V^T$$

- F cannot be unambiguously decomposed into pose and calibration

$$F = K^{-T} \hat{T} R K^{-1}$$

What Does *F* Tell Us?

- *F* can be inferred from point matches (eight-point algorithm)
- Cannot extract motion, structure and calibration from one fundamental matrix (two views)
- *F* allows reconstruction up to a projective transformation
- *F* encodes all the geometric information among two views when no additional information is available

Comments

- Without prior knowledge about the underlying 3D environment, one can only obtain Projective reconstruction rather than Euclidean reconstruction

- With prior knowledge about the underlying 3D environment (planar structures in particular), we can still perform Euclidean reconstruction

Multi-View Structure from Motion

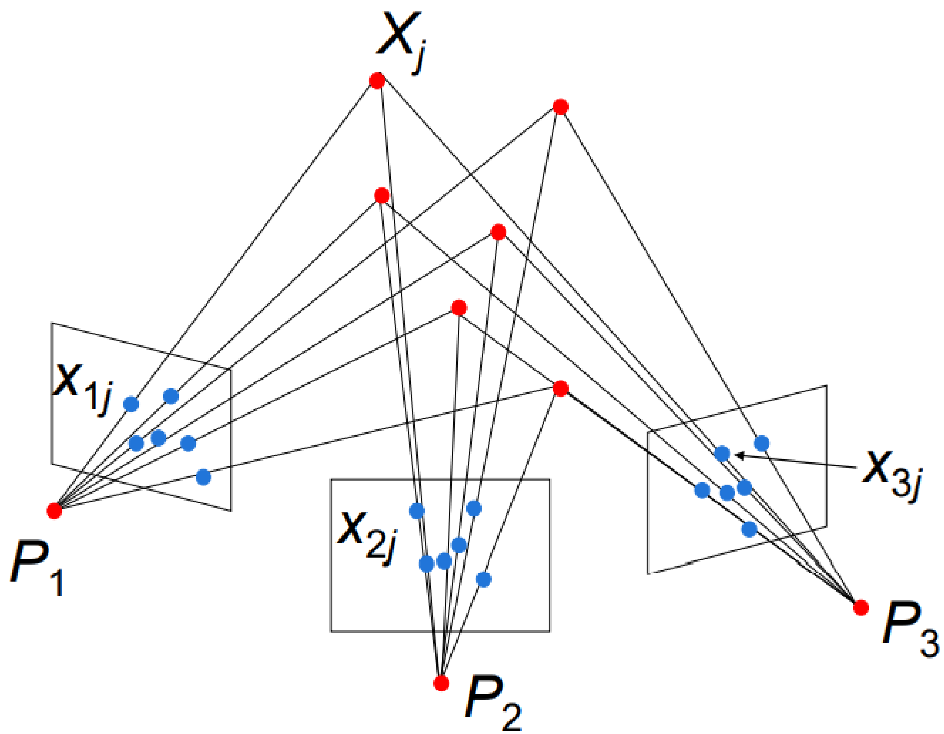
http://www.cs.cmu.edu/~16385/s18/lectures/lecture12.pdf

Projective structure from motion

Given: *m* images of *n* fixed 3D points

$$z_{ij}\,\mathbf{x}_{ij} = \mathbf{P}_i\,\mathbf{X}_j\,,\quad i = 1,\ldots,m,\quad j = 1,\ \ldots,n$$

Problem: estimate *m* projection matrices $\mathbf{P}_i$ and *n* 3D points $\mathbf{X}_j$ from the *mn* correspondences $\mathbf{x}_{ij}$



Projective structure from motion

- Given: *m* images of *n* fixed 3D points

$$z_{ij}\,\mathbf{x}_{ij} = \mathbf{P}_i\,\mathbf{X}_j\,, \quad i = 1,\dots,m, \quad j = 1,\dots,n$$

- Problem: estimate *m* projection matrices $\mathbf{P}_i$ and *n* 3D points $\mathbf{X}_j$ from the *mn* correspondences $\mathbf{x}_{ij}$
- With no calibration info, cameras and points can only be recovered up to a 4x4 projective transformation **Q**:

$$\mathbf{X} \rightarrow \mathbf{QX},\ \mathbf{P} \rightarrow \mathbf{PQ}^{-1}$$

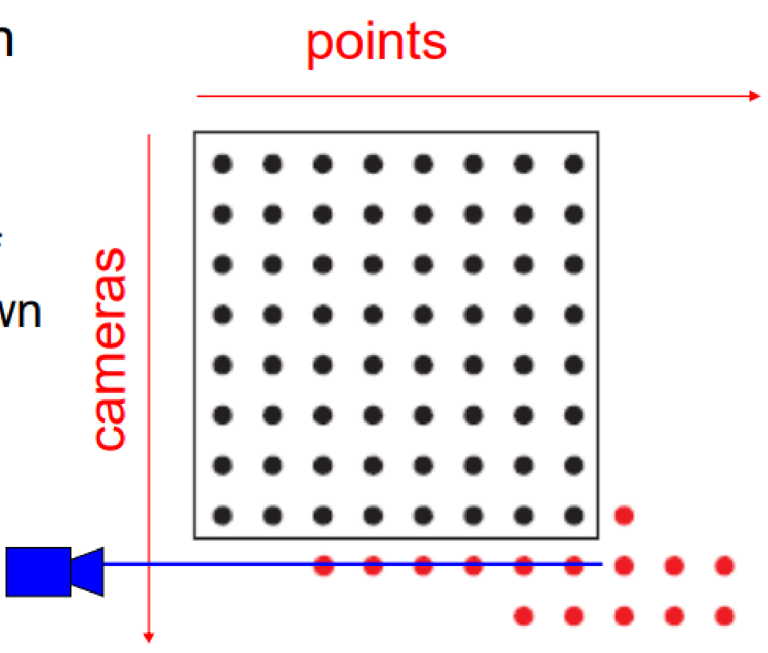
- We can solve for structure and motion when

$$2mn >= 11m + 3n - 15$$

- For two cameras, at least 7 points are needed

Sequential structure-from-motion

- Initialize motion from two images using fundamental matrix
- Initialize structure by triangulation
- For each additional view:
  - Determine projection matrix of new camera using all the known 3D points that are visible in its image – *calibration*

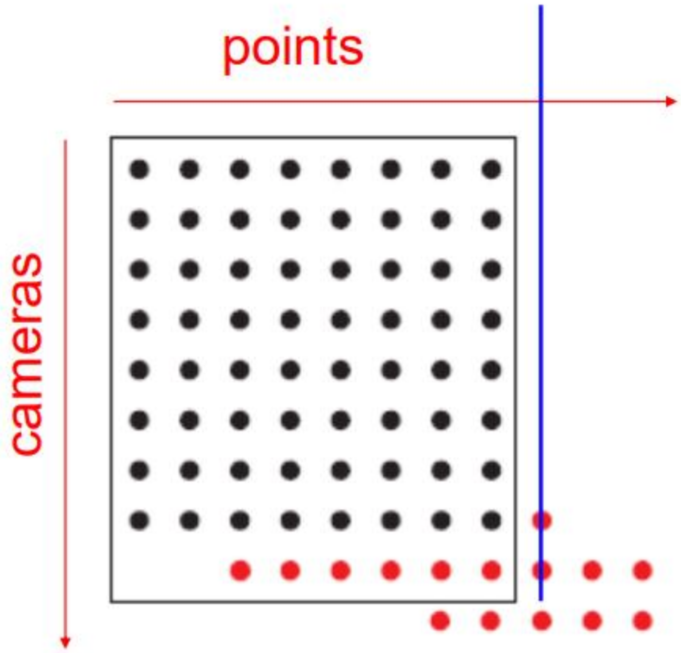


Sequential structure-from-motion

- Initialize motion from two images using fundamental matrix

- Initialize structure by triangulation

- For each additional view:
  - Determine projection matrix of new camera using all the known 3D points that are visible in its image – *calibration*
  - Refine and extend structure: compute new 3D points, re-optimize existing points that are also seen by this camera – *triangulation*



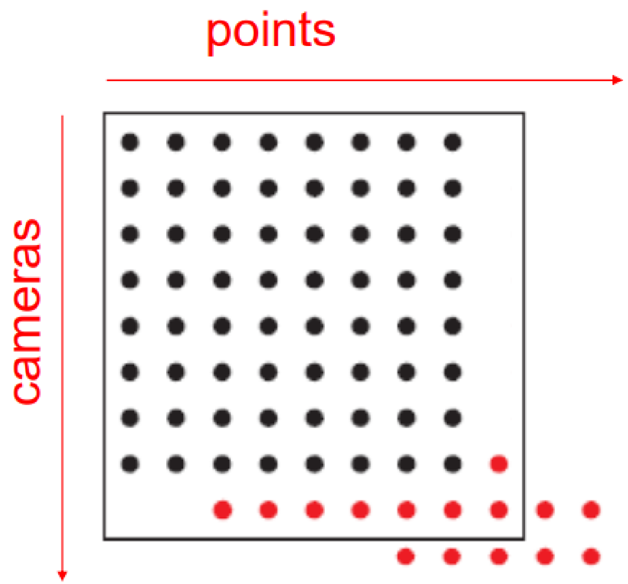
Sequential structure-from-motion

- Initialize motion from two images using fundamental matrix

- Initialize structure by triangulation

- For each additional view:
  - Determine projection matrix of new camera using all the known 3D points that are visible in its image – *calibration*

  - Refine and extend structure: compute new 3D points, re-optimize existing points that are also seen by this camera – *triangulation*

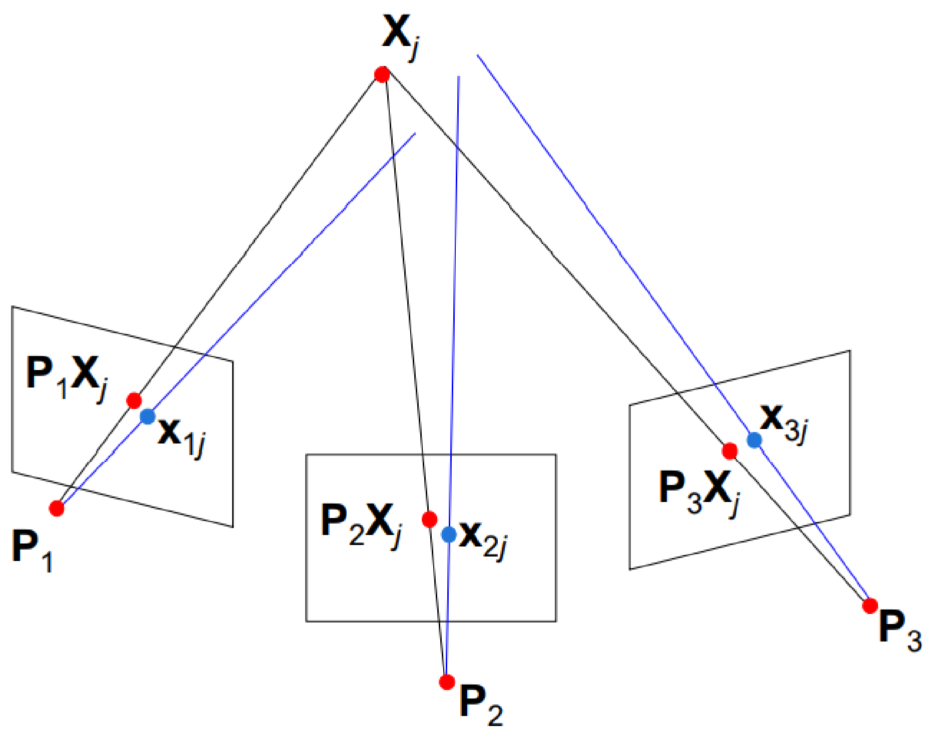


- Refine structure and motion: bundle adjustment

Bundle adjustment

- Non-linear method for refining structure and motion
- Minimizing reprojection error

$$E(\mathbf{P}, \mathbf{X}) = \sum_{i=1}^{m} \sum_{j=1}^{n} D\left(\mathbf{x}_{ij}, \mathbf{P}_i \mathbf{X}_j\right)^2$$



Large-scale structure from motion

http://www.cs.cmu.edu/~16385/s18/lectures/lecture12.pdf

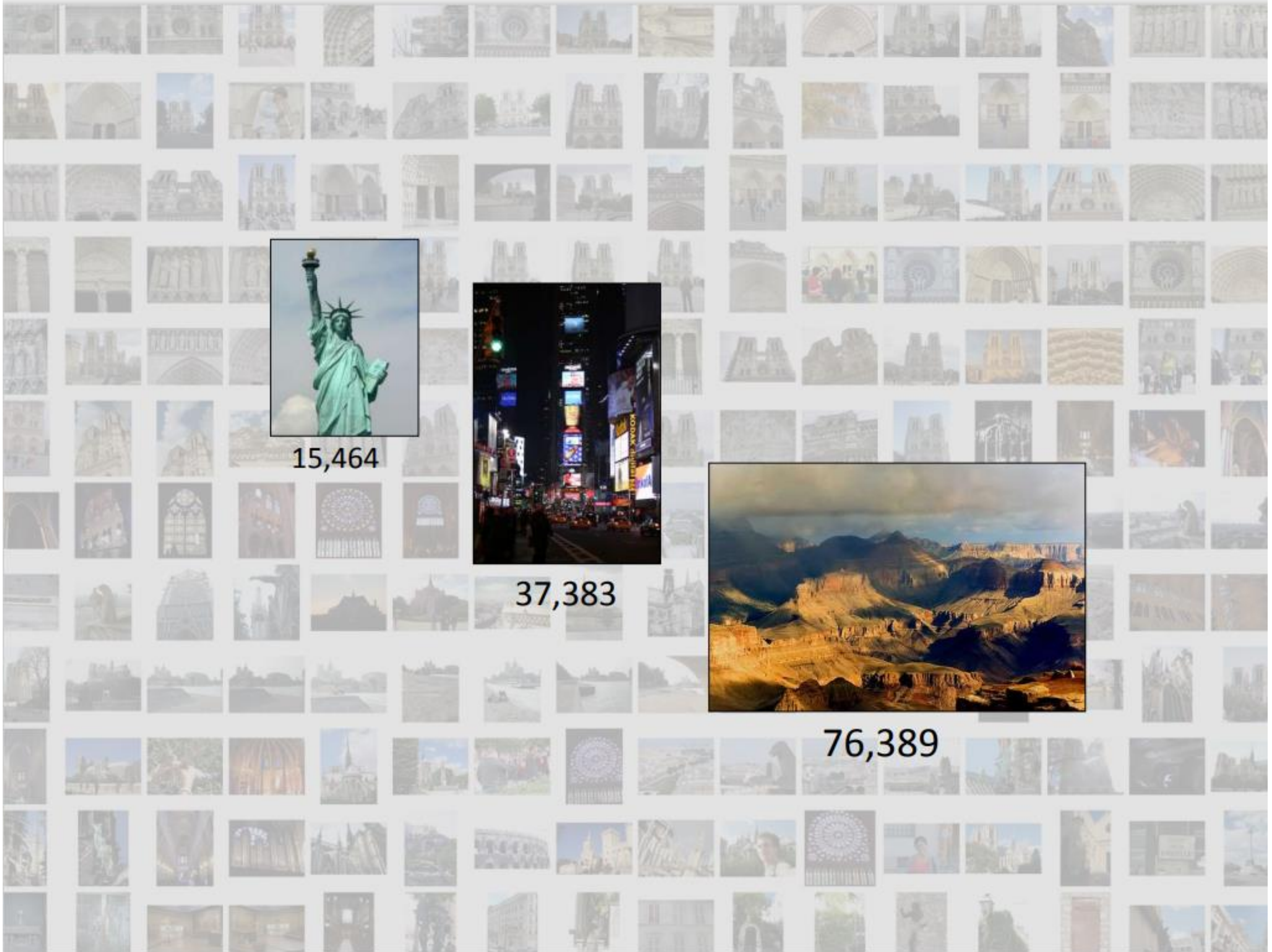
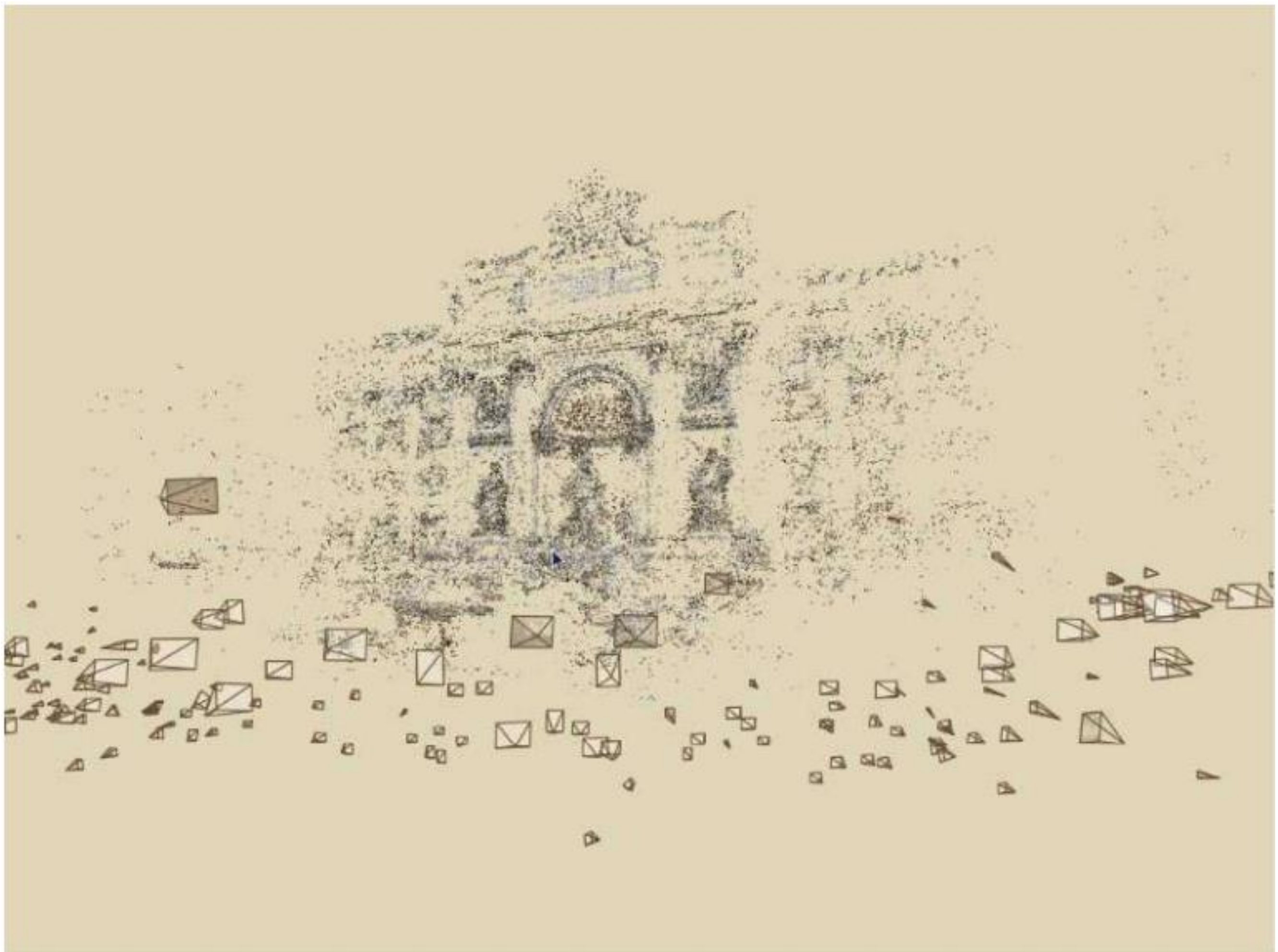
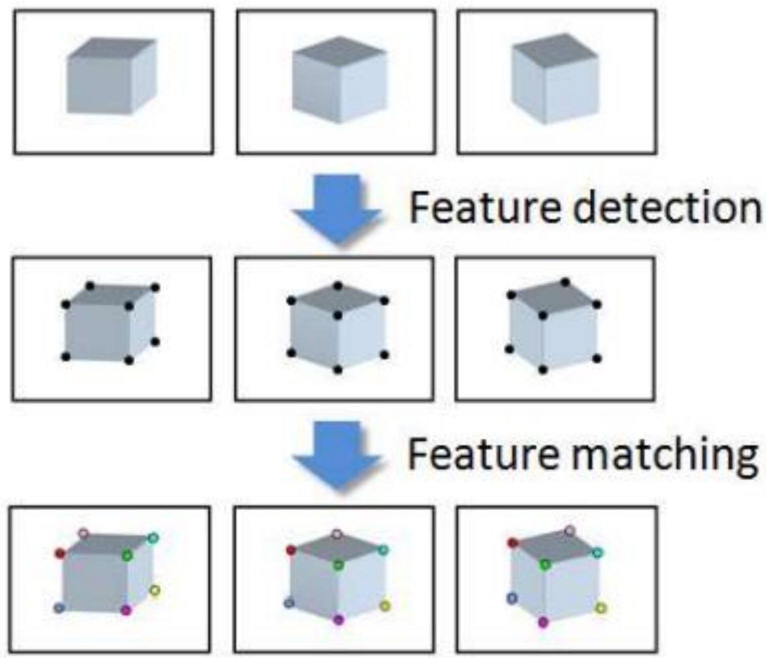

15,464
37,383
76,389

Photo Tourism



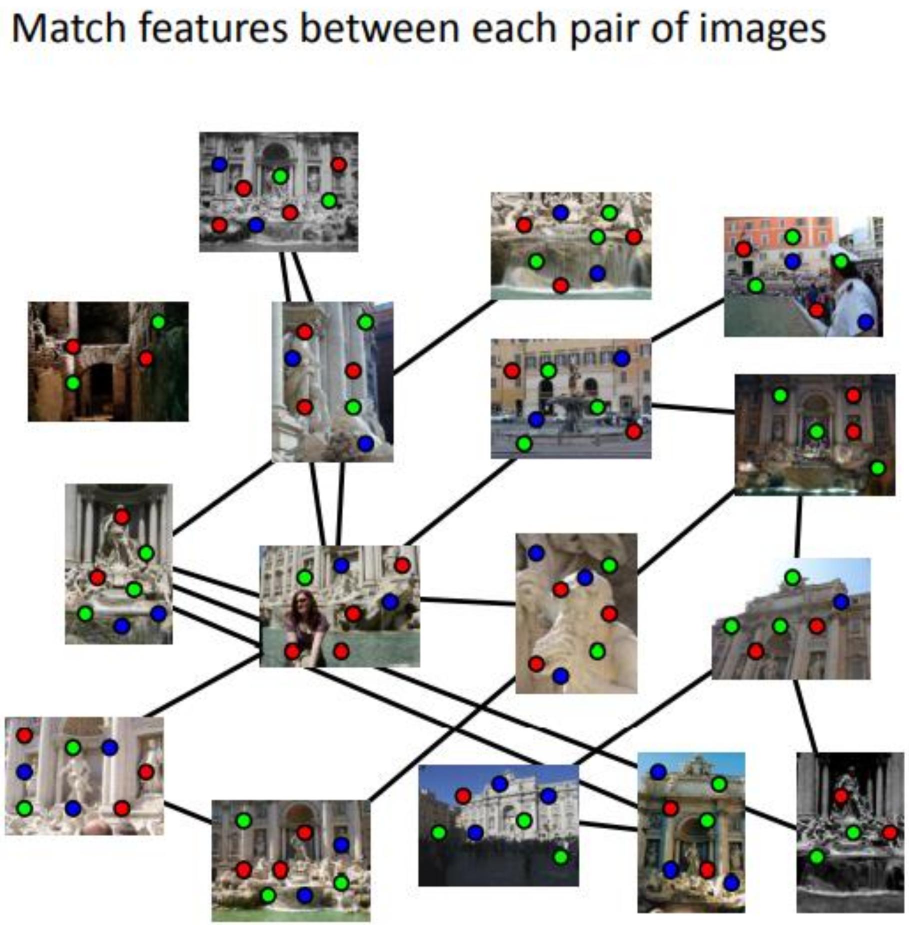
Input: Point correspondences



Feature detection

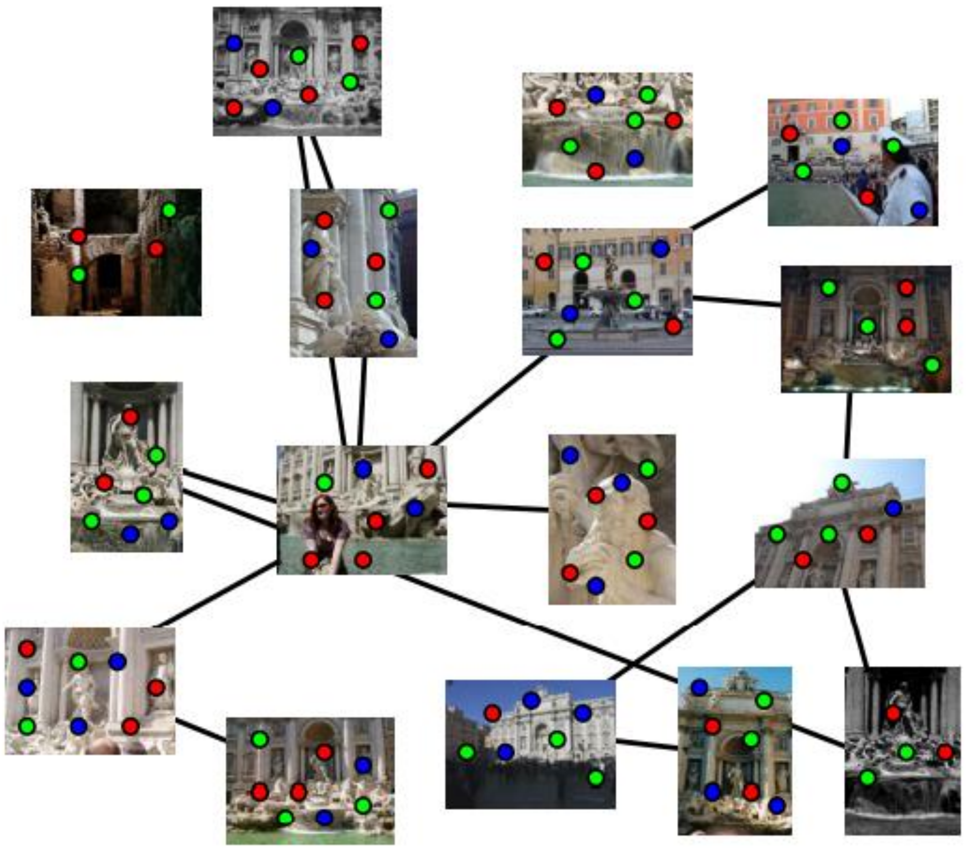
Describe features using SIFT [Lowe, IJCV 2004]

Feature matching



Feature matching

Refine matching using RANSAC to estimate fundamental matrix between each pair



Correspondence estimation

- Link up pairwise matches to form connected components of matches across several images

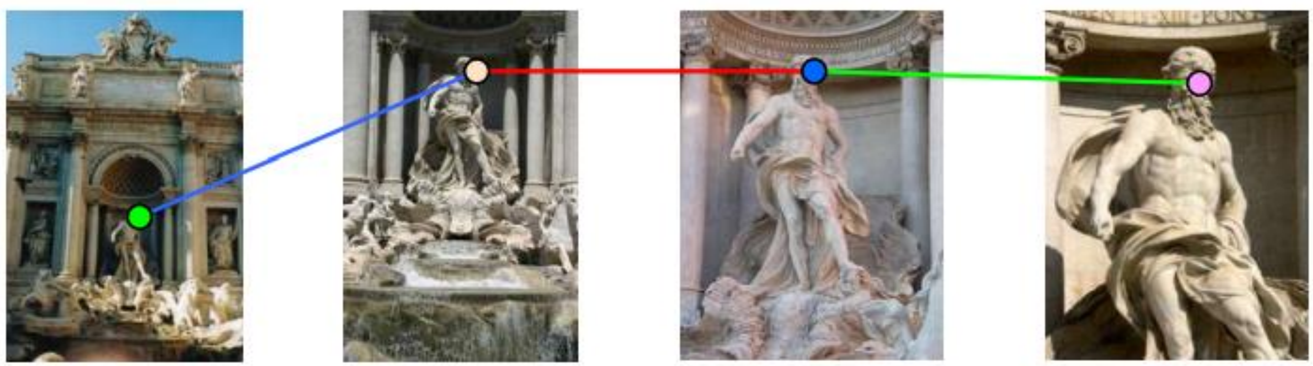
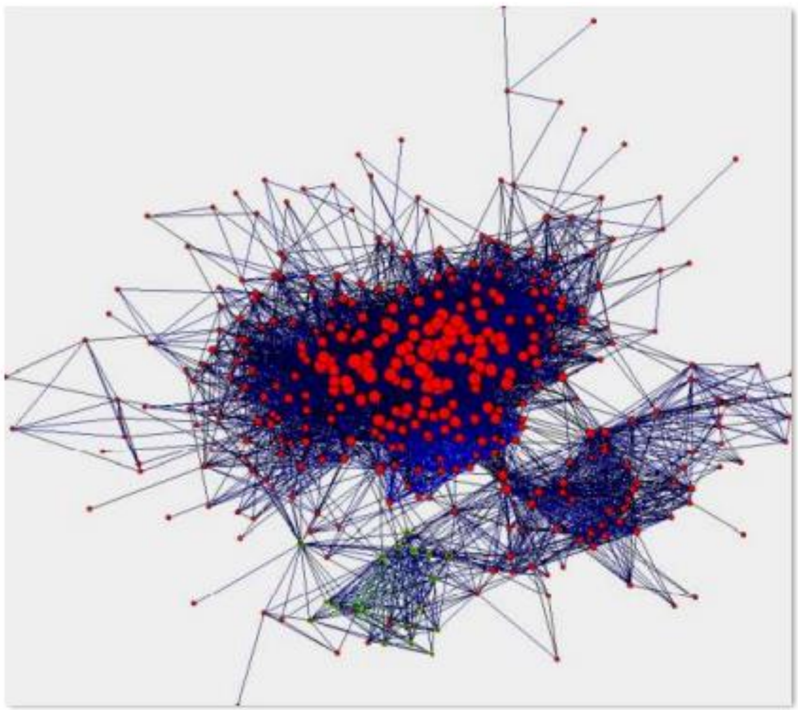


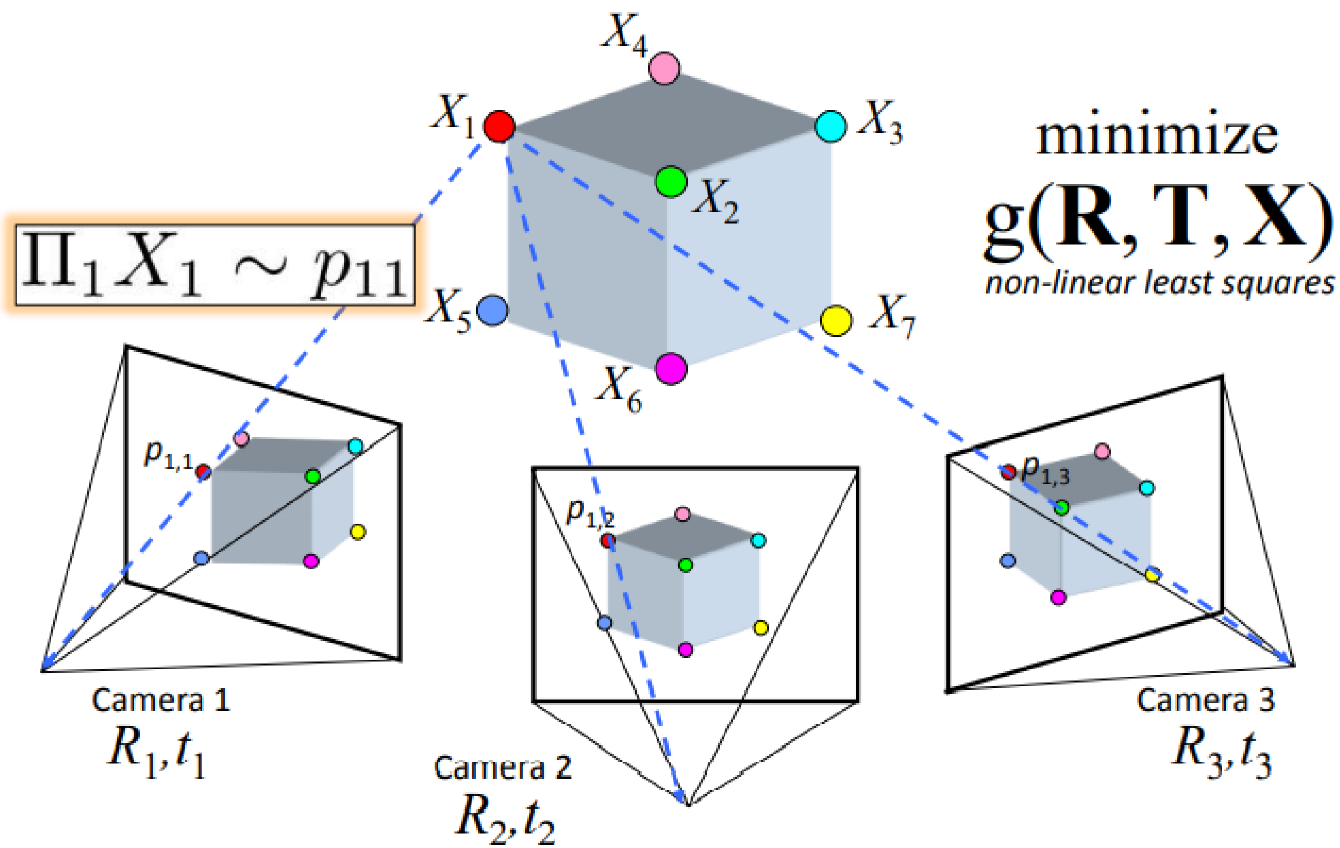
Image 1 Image 2 Image 3 Image 4

Image connectivity graph



(graph layout produced using the Graphviz toolkit: http://www.graphviz.org/)

Structure from motion



Global structure from motion

- Minimize sum of squared reprojection errors:

$$g(\mathbf{X}, \mathbf{R}, \mathbf{T}) = \sum_{i=1}^{m} \sum_{j=1}^{n} \underbrace{w_{ij}}_{\substack{\textit{indicator variable}:\\ \text{is point } i \text{ visible in image } j\text{ ?}}} \cdot \left\| \underbrace{\mathbf{P}(\mathbf{x}_i, \mathbf{R}_j, \mathbf{t}_j)}_{\substack{\textit{predicted}\\ \text{image location}}} - \underbrace{\begin{bmatrix} u_{i,j} \\ v_{i,j} \end{bmatrix}}_{\substack{\textit{observed}\\ \text{image location}}} \right\|^2$$

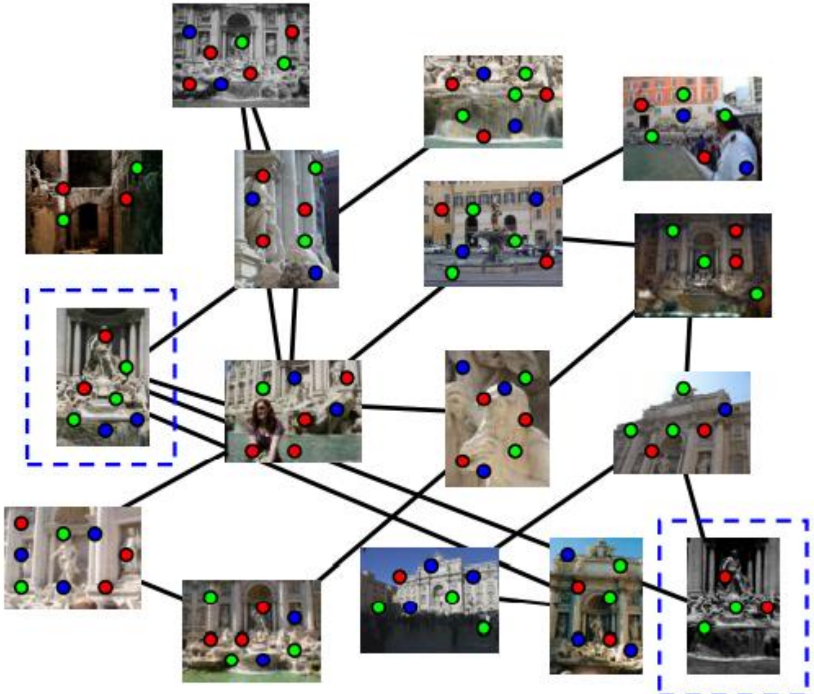
- Minimizing this function is called *bundle adjustment*
  - Optimized using non-linear least squares, e.g. Levenberg-Marquardt

Doing bundle adjustment

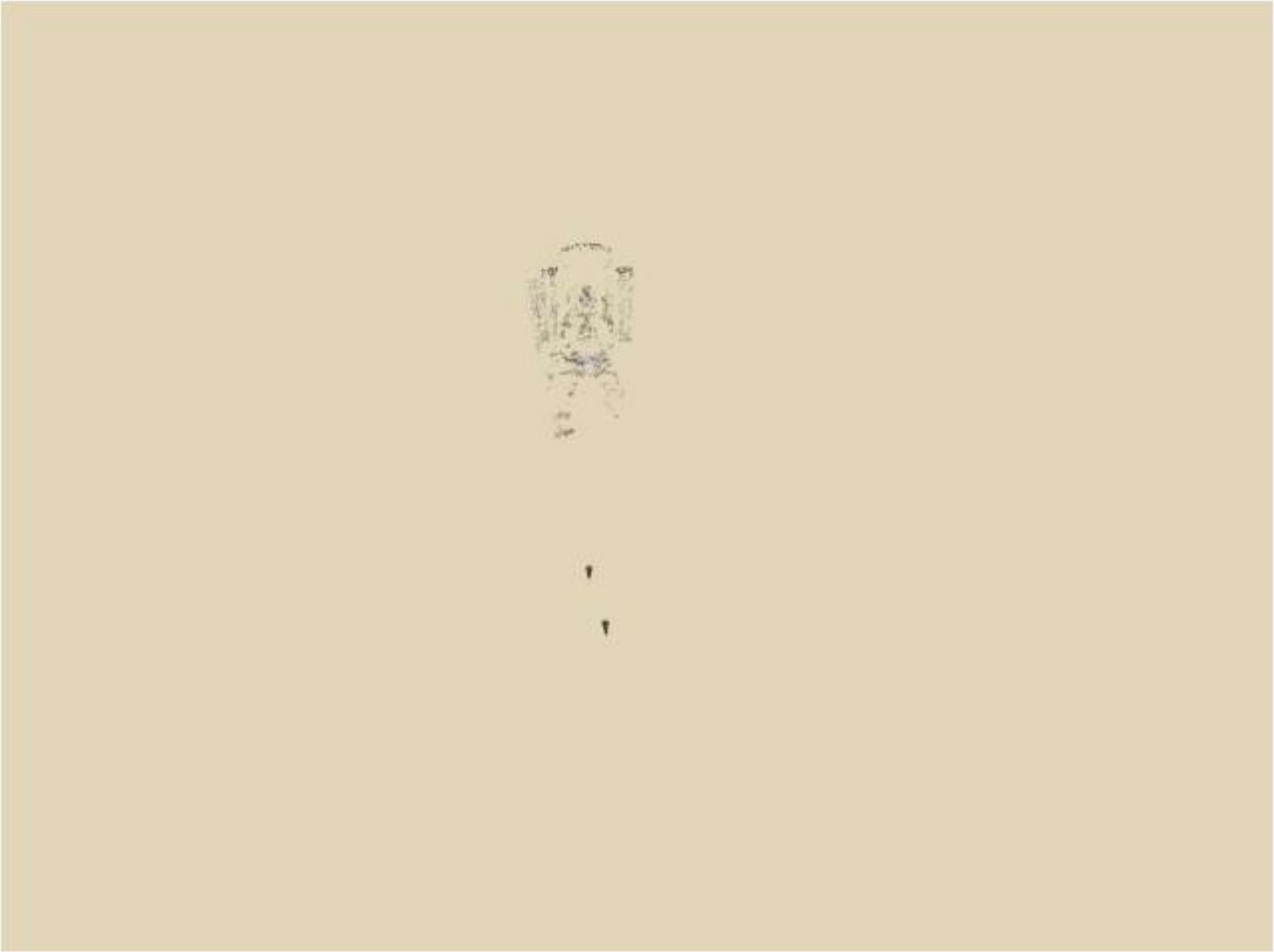
- Minimizing *g* is difficult
  - *g* is non-linear due to rotations, perspective division
  - lots of parameters: 3 for each 3D point, 6 for each camera
  - difficult to initialize
  - gauge ambiguity: error is invariant to a similarity transform (translation, rotation, uniform scale)

- Many techniques use non-linear least-squares (NLLS) optimization (*bundle adjustment*)
  - Levenberg-Marquardt is one common algorithm for NLLS
  - Lourakis, **The Design and Implementation of a Generic Sparse Bundle Adjustment Software Package Based on the Levenberg-Marquardt Algorithm,** http://www.ics.forth.gr/~lourakis/sba/
  - http://en.wikipedia.org/wiki/Levenberg-Marquardt_algorithm

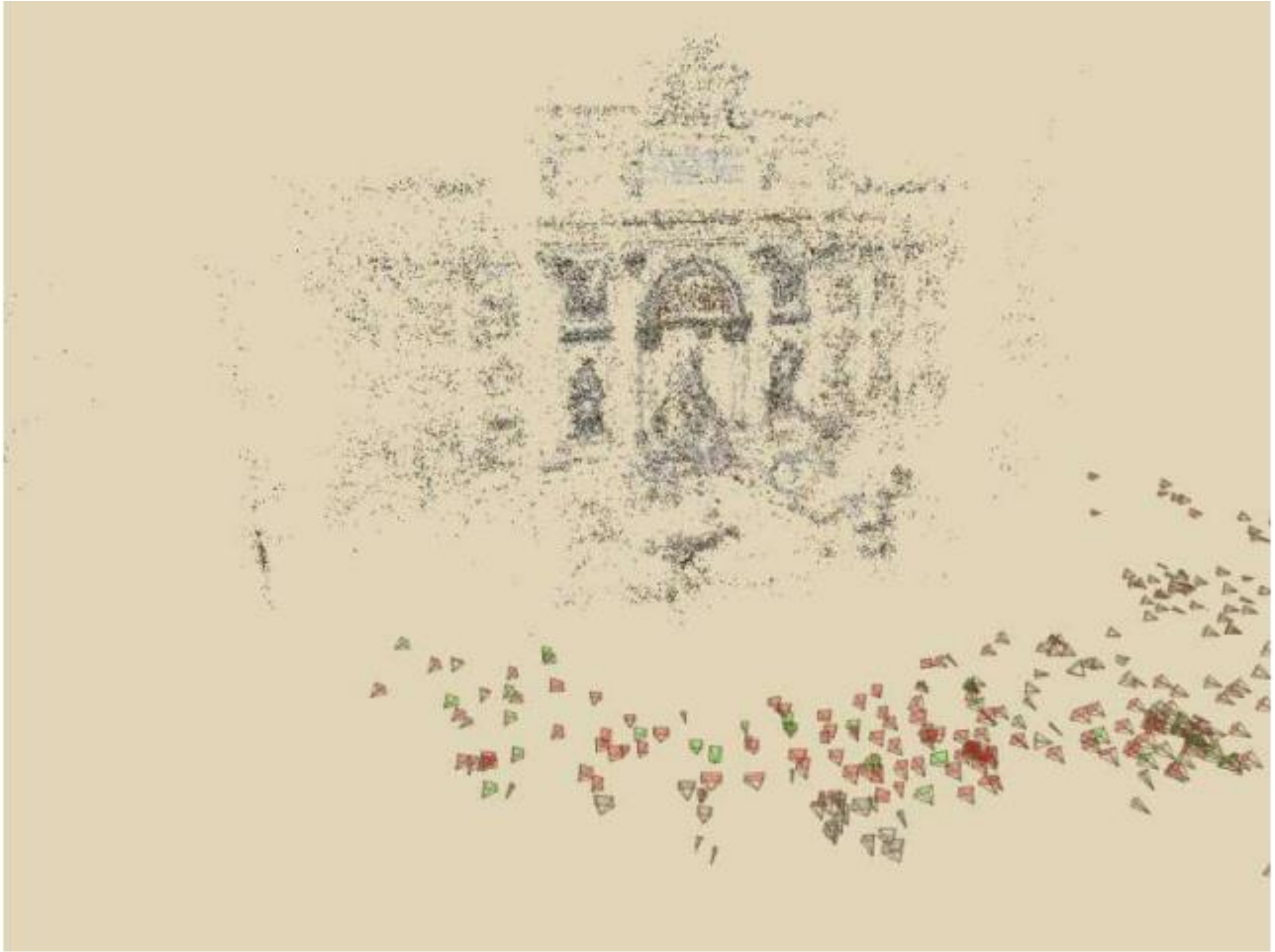
Initialization: Incremental structure from motion



Incremental structure from motion



Final reconstruction



Next lecture

Stereo Reconstruction