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

\[
(X, Y, Z) \xrightarrow{\text{Projection}} (x, y)
\]

- The equations of perspective projections are given by

\[
x = f \frac{X}{Z} \quad y = f \frac{Y}{Z}
\]

\(f\) 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 $\mathbf{x} = (x, y)$ is referred to as the inhomogeneous representation of the point $\mathbf{x}$.

- The homogeneous representation of a point $\mathbf{x}$ is given by $\mathbf{x} = (x, y, 1)$. In fact, the homogeneous representation of a point maps it to an entire class of set of points:
  
  $$(x, y) \leftrightarrow (\lambda x, \lambda y, \lambda), \quad \forall \lambda \neq 0$$

  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

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

• 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

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

• In $\mathbb{R}^2$, all pairs of lines intersect except for the ones that are parallel.

• In $\mathbb{P}^2$, all pairs of lines intersect, and parallel lines intersect in points of infinity and these points have the form $(x, y, 0)^T$.

• Consider the two lines given by

$$l_1 = (a_1, b_1, c_1)^T$$
$$l_2 = (a_2, b_2, c_2)^T$$

• The intersection of these two lines is given by

$$x = l_1 \times l_2$$
Intersection of parallel lines

- Given a line $l_1 = (a, b, c)^\top$, a line parallel to it is given by $l_2 = (a, b, c')^\top$

- The intersection is now given by

$$l_1 \times l_2 = (bc' - cb, ac - ac', 0)^\top$$
$$= (c - c')(b, -a, 0)^\top$$
$$\sim (b, -a, 0)^\top$$
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 \( l = x_1 \times x_2 \)

• The definition of points at infinity leads us to the definition of the line at infinity \( l_\infty \)

• Consider two points at infinity \( x_1 = (x_1, y_1, 0)^T \) and \( x_2 = (x_2, y_2, 0)^T \)

• The line passing through these two points is given by

\[
l_\infty = x_1 \times x_2 = (0, 0, x_1 y_2 - y_1 x_2)^2 \sim (0, 0, 1)^T
\]
A model for $\mathbb{P}^2$ in $\mathbb{R}^3$
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.

$$
\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K \Pi_0 g \begin{bmatrix} R \\ T \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}
$$

- $\lambda$: Scalar factor
- $K$: Intrinsic parameters
- $\Pi_0$: Canonical projection matrix
- $g$: Extrinsic parameters
The problem

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

\[ \lambda_2 x_2 = R\lambda_1 x_1 + T \]

- Multiply both sides by the cross product of \( T \) [Longuet-Higgins ‘81]:

\[ x_2^T \hat{E} R x_1 = 0 \]

- Essential matrix

\[ E = \hat{T} R \]
Mathematical Derivation

\[ \lambda_2 x_2 = R \lambda_1 x_1 + T \]

\[ T \times (\lambda_2 x_2) = T \times (R \lambda_1 x_1 + T) \]

\[ \lambda_2 T \times x_2 = \lambda_1 (T \times R) x_1 + T \times T \]

\[ \lambda_2 T \times x_2 = \lambda_1 (T \times R) x_1 \]

\[ (T \times R) x_1 \text{ is perpendicular to } x_2 \]

\[ x_2^T (T \times R) x_1 = 0 \]
Epipolar geometry

- Epipolar lines $l_1, l_2$
- Epipoles $e_1, e_2$

Properties (pay attention to geometric interpretations):

\[
\begin{align*}
    l_1 & \sim E^T x_2 \\
    l_i^T x_i & = 0 \\
    l_2 & \sim E x_1 \\
    E e_1 & = 0 \\
    l_i^T e_i & = 0 \\
    e_2 E^T & = 0
\end{align*}
\]
Singular-Value Decomposition

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

$$U = \begin{bmatrix} \mathbf{u}_1 & \cdots & \mathbf{u}_m \end{bmatrix} \in \mathbb{R}^{m \times m}$$

$$V = \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_n \end{bmatrix} \in \mathbb{R}^{n \times n}$$

such that

$$U^TAV = \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_p) \in \mathbb{R}^{m \times n}$$

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 $\sigma_r$ is the smallest nonzero singular value of $A$, then

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

Check Wikipedia for more details
Characterization of the Essential Matrix

\[ x_2^T \hat{T} R x_1 = 0 \]

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

\[ x_2^T \begin{bmatrix} e_1 & e_2 & e_2 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} x_1 = 0 \]

**Theorem 5.1 (Characterization of the essential matrix).** A non-zero 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

\[ \Sigma = \text{diag}\{\sigma, \sigma, 0\} \]

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 $R, T$

$$x^T_2 \hat{T} Rx_1 = 0$$

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

$$\min_E \sum_{j=1}^{n} x^T_2 E x_1^j$$
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 \(E = U\Sigma V^T\)

\[
\begin{align*}
(\hat{T}_1, R_1) &= (UR_Z(+\frac{\pi}{2})\Sigma U^T, UR_Z^T(+\frac{\pi}{2})V^T), \\
(\hat{T}_2, R_2) &= (UR_Z(-\frac{\pi}{2})\Sigma U^T, UR_Z^T(-\frac{\pi}{2})V^T).
\end{align*}
\]

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
    \[ E = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \rightarrow \ e = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9]^T \in \mathbb{R}^9 \]
  – Vectorized correspondence
    \[ \begin{align*}
    \mathbf{x}_1 &= [x_1, y_1, z_1]^T \in \mathbb{R}^3 \text{ and } \mathbf{x}_2 &= [x_2, y_2, z_2]^T \in \mathbb{R}^3 \\
    \mathbf{a} &= [x_2x_1, x_2y_1, x_2z_1, y_2x_1, y_2y_1, y_2z_1, z_2x_1, z_2y_1, z_2z_1]^T \in \mathbb{R}^9
    \end{align*} \]
  – 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 corresponds to the smallest eigenvalue:
    \[
    \min_e \frac{\|Ae\|^2}{\|e\|^2}
    \]
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: $F = U \text{diag}\{\lambda_1, \lambda_2, \lambda_3\}V^T$ 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 $E = U \text{diag}\{\sigma, \sigma, 0\}V^T$ 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

$$
\min_{X, \text{rank}(X)=r} \|A - X\|_F^2
$$

$$
X = U_r \Sigma_r V_r^T
$$
The eight-point method

Algorithm 5.1 (The eight-point algorithm). For a given set of image correspondences $(x_i^j, x_2^j), j = 1, \ldots, n \ (n \geq 8)$, this algorithm finds $(R, T) \in SE(3)$ which solves

$$x_2^j T \hat{T} R x_1^j = 0, \ j = 1, \ldots, n.$$

1. Compute a first approximation of the essential matrix

Construct the $A \in \mathbb{R}^{n \times 9}$ from correspondences $x_1^j$ and $x_2^j$ as in (6.21), namely.

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

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

   Compute the Singular Value Decomposition of the matrix $E$ recovered from data to be
   \[
   E = U \text{diag}\{\sigma_1, \sigma_2, \sigma_3\} V^T
   \]

   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 $\Sigma = \text{diag}\{1, 1, 0\}$.

3. **Recover displacement from the essential matrix**

   Define the diagonal matrix $\Sigma$ to be $\hat{E}$ and $T$ from the essential matrix as follows:
   \[
   R = U R_Z^T \left( \pm \frac{\pi}{2} \right) V^T, \quad \hat{T} = U R_Z \left( \pm \frac{\pi}{2} \right) \Sigma U^T.
   \]
Camera Calibration
Uncalibrated Camera – Intrinsic Parameters are unknown

\[
x' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = Kx = \begin{bmatrix} f s_x & f s_\theta & o_x \\ 0 & f s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}
\]

Linear transformation

\( K \)

Calibrated coordinates

Pixel coordinates

\((0, 0)\)
Overview

• Calibration with a rig (Checkboard for example)

• Uncalibrated epipolar geometry

• Ambiguities in image formation

• Stratified reconstruction
Uncalibrated Camera Using Homogeneous Coordinates

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

Last Lecture:
- Image plane coordinates \( x = [x, y, 1]^T \)
- Camera extrinsic parameters \( g = (R, T) \)
- Perspective projection \( \lambda x = [R, T]X \)

This Lecture:
- Pixel coordinates \( x' = Kx \)
- Projection matrix \( \lambda x' = \pi X = [KR, KT]X \)
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 $X$

  $$\lambda x' = [K R, K T] X \quad \Rightarrow \quad \lambda x' = \pi X$$

  $$\lambda \begin{bmatrix} x^i \\ y^i \\ 1 \end{bmatrix} = \begin{bmatrix} \pi^T_1 \\ \pi^T_2 \\ \pi^T_3 \end{bmatrix} \begin{bmatrix} X^i \\ Y^i \\ Z^i \\ 1 \end{bmatrix}$$

- Eliminate unknown scales

  $$x^i(\pi^T_3 X) = \pi^T_1 X, \quad y^i(\pi^T_3 X) = \pi^T_2 X$$
Calibration with a Rig

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

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

\[ \min ||M\Pi^s||^2 \text{ subject to } ||\Pi^s||^2 = 1 \]

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

\[ \lambda_2 K x_2 = K R \lambda_1 x_1 + K T \]
\[ \lambda_2 x'_2 = K R K^{-1} \lambda_1 x'_1 + T' \]

- Epipolar constraint
  \[ x'_2^T K^{-T} \hat{T} R K^{-1} x'_1 = 0 \]
- 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

\[ x'^T_2 F x'_1 = 0 \]

- Epipolar lines \( l_1, l_2 \)
- Epipoles \( e_1, e_2 \)

\[
\begin{align*}
  l_1 &\sim F^T x'_2 \\
  Fe_1 &\equiv 0 \\
  l_i^T x'_i &\equiv 0 \\
  l_i^T e_i &\equiv 0 \\
  l_2 &\sim F x'_1 \\
  e_2^T F &\equiv 0
\end{align*}
\]
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

\[
\Sigma = \text{diag}\{\sigma_1, \sigma_2, 0\}
\]

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

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

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

$$a = [x_1 x_2, x_1 y_2, x_1 z_2, y_1 x_2, y_1 y_2, y_1 z_2, z_1 x_2, z_1 y_2, z_1 z_2]^T$$

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

- Rewrite $a^T F^s = 0$

$$\min_{F^s} \|A F^s\|^2 \quad s.t. \quad \|F^s\|^2 = 1$$
Two view linear algorithm – 8-point algorithm

- Solve the LLSE problem:

\[
\min_F \sum_{j=1}^n (\mathbf{x}_{2,j}'^T F \mathbf{x}_{1,j}')^2 \quad \text{s.t.} \quad \|F\|_F^2 = 1
\]

- Solution eigenvector associated with smallest eigenvalue of \(A^TA\)

- Compute SVD of \(F\) recovered from data

\[
F = U \Sigma V^T \\
\Sigma = diag(\sigma_1, \sigma_2, \sigma_3)
\]

- Project onto the essential manifold:

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

- \(F\) 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
Projective structure from motion

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

$$z_{ij} \, x_{ij} = P_i \, X_j, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n$$

Problem: estimate $m$ projection matrices $P_i$ and $n$ 3D points $X_j$ from the $mn$ correspondences $x_{ij}$
Projective structure from motion

- Given: $m$ images of $n$ fixed 3D points
  \[ z_{ij} x_{ij} = P_i X_j, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n \]

- Problem: estimate $m$ projection matrices $P_i$ and $n$ 3D points $X_j$ from the $mn$ correspondences $x_{ij}$

- With no calibration info, cameras and points can only be recovered up to a 4x4 projective transformation $Q$:
  \[ X \rightarrow QX, \quad P \rightarrow PQ^{-1} \]

- We can solve for structure and motion when
  \[ 2mn \geq 11m + 3n - 15 \]

- 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

\[ E(P, X) = \sum_{i=1}^{m} \sum_{j=1}^{n} D(x_{ij}, P_i X_j)^2 \]
Large-scale structure from motion

Photo Tourism
Input: Point correspondences
Feature detection

Describe features using SIFT [Lowe, IJCV 2004]
Feature matching

Match features between each pair of images
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 connectivity graph

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

\[ \Pi_1 X_1 \sim p_{11} \]

\[ \text{minimize } g(R, T, X) \]

[non-linear least squares]

Camera 1
\[ R_1, t_1 \]

Camera 2
\[ R_2, t_2 \]

Camera 3
\[ R_3, t_3 \]
Global structure from motion

- Minimize sum of squared reprojection errors:

\[ g(\mathbf{X}, \mathbf{R}, \mathbf{T}) = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} \cdot \left\| \mathbf{P}(\mathbf{x}_i, \mathbf{R}_j, \mathbf{t}_j) - \begin{bmatrix} u_{i,j} \\ v_{i,j} \end{bmatrix} \right\|^2 \]

- 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*,
Initialization: Incremental structure from motion
Incremental structure from motion
Final reconstruction
Next lecture

Stereo Reconstruction