Last time: Stereo

- Human stereopsis & stereograms
- Epipolar geometry and the epipolar constraint
  - Case example with parallel optical axes
  - General case with calibrated cameras
- Correspondence search
- The Essential and the Fundamental Matrix
- Multi-view stereo
Today: SFM

• SFM problem statement
• Factorization
• Projective SFM
Structure from motion
Multiple-view geometry questions

- **Scene geometry (structure):** Given 2D point matches in two or more images, where are the corresponding points in 3D?

- **Correspondence (stereo matching):** Given a point in just one image, how does it constrain the position of the corresponding point in another image?

- **Camera geometry (motion):** Given a set of corresponding points in two or more images, what are the camera matrices for these views?
Structure from motion

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

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

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

• If we scale the entire scene by some factor $k$ and, at the same time, scale the camera matrices by the factor of $1/k$, the projections of the scene points in the image remain exactly the same:

\[
x = PX = \left( \frac{1}{k} P \right) (k X)
\]

It is impossible to recover the absolute scale of the scene!
Structure from motion ambiguity

- If we scale the entire scene by some factor $k$ and, at the same time, scale the camera matrices by the factor of $1/k$, the projections of the scene points in the image remain exactly the same.

- More generally: if we transform the scene using a transformation $Q$ and apply the inverse transformation to the camera matrices, then the images do not change.

$$x = PX = (PQ^{-1})(QX)$$
Projective ambiguity

\[ x = PX = \left( PQ_P^{-1} \right) \left( Q_P X \right) \]
Projective ambiguity

Lazebnik
Affine ambiguity

\[ x = PX = \left( PQ_A^{-1} \right) \left( Q_A X \right) \]
Affine ambiguity
Similarity ambiguity

\[ x = PX = \left( PQ_S^{-1} \right) \left( Q_S X \right) \]

Lazebnik
Similarity ambiguity

Lazebnik
Hierarchy of 3D transformations

- **Projective**
  - 15dof
  - \[
    \begin{bmatrix}
    A & t \\
    v^T & v
    \end{bmatrix}
  \]
  - Preserves intersection and tangency

- **Affine**
  - 12dof
  - \[
    \begin{bmatrix}
    A & t \\
    0^T & 1
    \end{bmatrix}
  \]
  - Preserves parallelism, volume ratios

- **Similarity**
  - 7dof
  - \[
    \begin{bmatrix}
    s R & t \\
    0^T & 1
    \end{bmatrix}
  \]
  - Preserves angles, ratios of length

- **Euclidean**
  - 6dof
  - \[
    \begin{bmatrix}
    R & t \\
    0^T & 1
    \end{bmatrix}
  \]
  - Preserves angles, lengths

- With no constraints on the camera calibration matrix or on the scene, we get a **projective** reconstruction.
- Need additional information to **upgrade** the reconstruction to affine, similarity, or Euclidean.
Structure from motion

• Let’s start with *affine cameras* (the math is easier)
Recall: Orthographic Projection

Special case of perspective projection

• Distance from center of projection to image plane is infinite

• Projection matrix:

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z \\
1 \\
\end{bmatrix}
= \begin{bmatrix}
x \\
y \\
1 \\
\end{bmatrix} \Rightarrow (x, y)
Affine cameras

Orthographic Projection

Parallel Projection

Lazebnik
Affine cameras

- A general affine camera combines the effects of an affine transformation of the 3D space, orthographic projection, and an affine transformation of the image:

\[
P = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
a_{11} & a_{12} & a_{13} & b_1 \\
a_{21} & a_{22} & a_{23} & b_2 \\
0 & 0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
A & b \\
0 & 1
\end{bmatrix}
\]

- Affine projection is a linear mapping + translation in inhomogeneous coordinates

\[
x = \begin{pmatrix}
x \\
y
\end{pmatrix}
= \begin{bmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23}
\end{bmatrix}
\begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix}
+ \begin{pmatrix}
b_1 \\
b_2
\end{pmatrix}
= AX + b
\]
Affine structure from motion

• Given: $m$ images of $n$ fixed 3D points:
  \[ x_{ij} = A_i X_j + b_i, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n \]

• Problem: use the $mn$ correspondences $x_{ij}$ to estimate $m$ projection matrices $A_i$ and translation vectors $b_i$, and $n$ points $X_j$

• The reconstruction is defined up to an arbitrary affine transformation $Q$ (12 degrees of freedom):
  \[
  \begin{bmatrix}
  A & b \\
  0 & 1
  \end{bmatrix}
  \rightarrow
  \begin{bmatrix}
  A & b \\
  0 & 1
  \end{bmatrix}Q^{-1}, \quad \begin{pmatrix} X \\ 1 \end{pmatrix} \rightarrow Q\begin{pmatrix} X \\ 1 \end{pmatrix}
  \]

• We have $2mn$ knowns and $8m + 3n$ unknowns (minus 12 dof for affine ambiguity)

• Thus, we must have $2mn \geq 8m + 3n - 12$

• For two views, we need four point correspondences
Affine structure from motion

- Centering: subtract the centroid of the image points

\[
\hat{x}_{ij} = x_{ij} - \frac{1}{n} \sum_{k=1}^{n} x_{ik} = A_i X_j + b_i - \frac{1}{n} \sum_{k=1}^{n} (A_i X_k + b_i)
\]

\[
= A_i \left( X_j - \frac{1}{n} \sum_{k=1}^{n} X_k \right) = A_i \hat{X}_j
\]

- For simplicity, assume that the origin of the world coordinate system is at the centroid of the 3D points
- After centering, each normalized point \( x_{ij} \) is related to the 3D point \( X_i \) by

\[
\hat{x}_{ij} = A_i X_j
\]
Affine structure from motion

- Let's create a $2m \times n$ data (measurement) matrix:

$$D = \begin{bmatrix}
\hat{X}_{11} & \hat{X}_{12} & \cdots & \hat{X}_{1n} \\
\hat{X}_{21} & \hat{X}_{22} & \cdots & \hat{X}_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\hat{X}_{m1} & \hat{X}_{m2} & \cdots & \hat{X}_{mn}
\end{bmatrix}$$

Affine structure from motion

Let’s create a $2m \times n$ data (measurement) matrix:

$$D = \begin{bmatrix}
\hat{x}_{11} & \hat{x}_{12} & \cdots & \hat{x}_{1n} \\
\hat{x}_{21} & \hat{x}_{22} & \cdots & \hat{x}_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\hat{x}_{m1} & \hat{x}_{m2} & \cdots & \hat{x}_{mn}
\end{bmatrix} = \begin{bmatrix}
A_1 \\
A_2 \\
\vdots \\
A_m
\end{bmatrix}\begin{bmatrix}
X_1 \\
X_2 \\
\vdots \\
X_n
\end{bmatrix}$$

points (3 × n)
cameras (2m × 3)

The measurement matrix $D = MS$ must have rank 3!

Factorizing the measurement matrix

\[ \text{Measurements} = \text{Motion} \times \text{Shape} \]

\[ D = MS \]

Source: M. Hebert
Factorizing the measurement matrix

- Singular value decomposition of $D$:

\[
\begin{array}{c}
\begin{array}{c}
\text{D} = \\
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\text{U} = \\
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\text{W} \\
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\text{V}^T \\
\end{array}
\end{array}
\end{array}
\]

Lazebnik

Source: M. Hebert
Factorizing the measurement matrix

- Singular value decomposition of D:

\[ D = U W V^T \]

To reduce to rank 3, we just need to set all the singular values to 0 except for the first 3.
Factorizing the measurement matrix

• Obtaining a factorization from SVD:

\[ 2m \times U_3 \times W_3 \times V_3^T \]
Factorizing the measurement matrix

- Obtaining a factorization from SVD:

\[
D = U_3 W_3^{1/2} S = W_3^{1/2} V_3^T
\]

This decomposition minimizes \( |D-MS|^2 \)

Source: M. Hebert
Affine ambiguity

- The decomposition is not unique. We get the same $D$ by using any $3\times3$ matrix $C$ and applying the transformations $M \rightarrow MC$, $S \rightarrow C^{-1}S$
- That is because we have only an affine transformation and we have not enforced any Euclidean constraints (like forcing the image axes to be perpendicular, for example)
Eliminating the affine ambiguity

- Orthographic: image axes are perpendicular and scale is 1

- This translates into $3m$ equations in $L = CC^T$:
  \[ A_i L A_i^T = \text{Id}, \quad i = 1, \ldots, m \]

- Solve for $L$
- Recover $C$ from $L$ by Cholesky decomposition: $L = CC^T$
- Update $M$ and $S$: $M = MC$, $S = C^{-1}S$

Source: M. Hebert
Algorithm summary

- Given: $m$ images and $n$ features $x_{ij}$
- For each image $i$, center the feature coordinates
- Construct a $2m \times n$ measurement matrix $D$:
  - Column $j$ contains the projection of point $j$ in all views
  - Row $i$ contains one coordinate of the projections of all the $n$ points in image $i$
- Factorize $D$:
  - Compute SVD: $D = U W V^T$
  - Create $U_3$ by taking the first 3 columns of $U$
  - Create $V_3$ by taking the first 3 columns of $V$
  - Create $W_3$ by taking the upper left $3 \times 3$ block of $W$
- Create the motion and shape matrices:
  - $M = U_3 W_3^{\frac{1}{2}}$ and $S = W_3^{\frac{1}{2}} V_3^T$ (or $M = U_3$ and $S = W_3 V_3^T$)
- Eliminate affine ambiguity

Source: M. Hebert
Reconstruction results

Dealing with missing data

- So far, we have assumed that all points are visible in all views.
- In reality, the measurement matrix typically looks something like this: