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:

$$\mathbf{x} = \mathbf{PX} = \left(\frac{1}{k}\mathbf{P}\right)(k\mathbf{X})$$

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.

\[ x = PX = (PQ^{-1})(QX) \]
Reconstruction ambiguity: Similarity

\[ x = PX = \left( PQ^{-1}_S \right) (Q_SX) \]
Reconstruction ambiguity: Affine

\[ X = PX = \left( PQ_A^{-1} \right) \left( Q_A X \right) \]
Reconstruction ambiguity: Projective

\[ x = PX = \left( PQ_P^{-1} \right) \left( Q_P X \right) \]
Projective ambiguity
From projective to affine
From affine to similarity
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
  - \[ \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} \]
  - 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:

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

Orthographic Projection

Parallel Projection
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:

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

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

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

Projection of world origin
Affine structure from motion

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

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

• 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 \( X_j \) by

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

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

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

Cameras $(2m)$

Points $(n)$

Affine structure from motion

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

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

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

Factorizing the measurement matrix

\[ D = MS \]

Source: M. Hebert
Factorizing the measurement matrix

- Singular value decomposition of D:

\[ D = U S V^T \]

\[ D = U_3 W_3 V_3^T \]

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.

Source: M. Hebert
Factorizing the measurement matrix

- Obtaining a factorization from SVD:

\[ D = U_3 \times W_3 \times V_3^T \]
Factorizing the measurement matrix

- Obtaining a factorization from SVD:

\[
\begin{align*}
2m & \quad D \\
& \quad = \\
& \quad U_3 \times 3 W_3 \times 3 V_3^T
\end{align*}
\]

Possible decomposition:
\[
M = U_3 W_3^{1/2} \quad S = W_3^{1/2} V_3^T
\]

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 axis to be perpendicular, for example)

Source: M. Hebert
Eliminating the affine ambiguity

- Orthographic: image axes are perpendicular and scale is 1

This translates into $3m$ equations in $L = CC^T$:

$$A_i L A_i^T = Id, \quad i = 1, \ldots, m$$

- 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:
  - $M = U_3 W_3^{1/2}$ and $S = W_3^{1/2} V_3^T$ (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:
Dealing with missing data

- Possible solution: decompose matrix into dense sub-blocks, factorize each sub-block, and fuse the results
  - Finding dense maximal sub-blocks of the matrix is NP-complete (equivalent to finding maximal cliques in a graph)

- Incremental bilinear refinement

(1) Perform factorization on a dense sub-block
(2) Solve for a new 3D point visible by at least two known cameras (linear least squares)
(3) Solve for a new camera that sees at least three known 3D points (linear least squares)

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

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

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

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

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

- For two cameras, at least 7 points are needed
Projective SFM: Two-camera case

- Compute fundamental matrix $F$ between the two views
- First camera matrix: $[I|0]Q^{-1}$
- Second camera matrix: $[A|b]Q^{-1}$
- Let $\tilde{X} = QX$
- Then $z'x' = A[I|0]\tilde{X} + b = zAx + b$
  
  $z'x' \times b = zAx \times b$

  $(z'x' \times b) \cdot x' = (zAx \times b) \cdot x'$

  $x'^T [b_x] Ax = 0$

$F = [b_x] A \quad b: \text{epipole} \ (F^Tb = 0), \quad A = -[b_x]F$

F&P sec. 13.3.1
Projective factorization

\[
D = \begin{bmatrix}
    z_{11}x_{11} & z_{12}x_{12} & \cdots & z_{1n}x_{1n} \\
    z_{21}x_{21} & z_{22}x_{22} & \cdots & z_{2n}x_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    z_{m1}x_{m1} & z_{m2}x_{m2} & \cdots & z_{mn}x_{mn}
\end{bmatrix} = \begin{bmatrix}
P_1 \\
P_2 \\
\vdots \\
P_m
\end{bmatrix} \begin{bmatrix}
X_1 & X_2 & \cdots & X_n
\end{bmatrix}
\]

points \((4 \times n)\)
cameras \((3m \times 4)\)

\[D = MS\] has rank 4

- If we knew the depths \(z\), we could factorize \(D\) to estimate \(M\) and \(S\)
- If we knew \(M\) and \(S\), we could solve for \(z\)
- Solution: iterative approach (alternate between above two steps)
Sequential structure from motion

- Initialize motion from two images using fundamental matrix
- Initialize structure
- 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
• 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
• 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

\[ E(P, X) = \sum_{i=1}^{m} \sum_{j=1}^{n} D(x_{ij}, P_i X_j)^2 \]
Self-calibration

- Self-calibration (auto-calibration) is the process of determining intrinsic camera parameters directly from uncalibrated images.
- For example, when the images are acquired by a single moving camera, we can use the constraint that the intrinsic parameter matrix remains fixed for all the images.
  - Compute initial projective reconstruction and find 3D projective transformation matrix $Q$ such that all camera matrices are in the form $P_i = K [R_i | t_i]$
- Can use constraints on the form of the calibration matrix: zero skew.
Summary: Structure from motion

• Ambiguity
• Affine structure from motion: factorization
• Dealing with missing data
• Projective structure from motion: two views
• Projective structure from motion: iterative factorization
• Bundle adjustment
• Self-calibration
Summary: 3D geometric vision

• Single-view geometry
  • The pinhole camera model
    – Variation: orthographic projection
  • The perspective projection matrix
  • Intrinsic parameters
  • Extrinsic parameters
  • Calibration

• Multiple-view geometry
  • Triangulation
  • The epipolar constraint
    – Essential matrix and fundamental matrix
  • Stereo
    – Binocular, multi-view
  • Structure from motion
    – Reconstruction ambiguity
    – Affine SFM
    – Projective SFM