Lecture 25: Structure from Motion

Given a set of flow fields or displacement vectors from a moving camera over time, determine:
- the sequence of camera poses
- the 3D structure of the scene

SFM “Killer App”

Match Move
Track a set of feature points through a movie sequence
Deduce where the cameras are and the 3D locations of the points that were tracked
Render synthetic objects with respect to the deduced 3D geometry of the scene / cameras

Factorization


Goal: combine point correspondence information from multiple points over multiple frames to solve for scene structure and camera motion (structure from motion)
Approach: numerically stable approach based on using SVD to “factor” matrix of observed point positions.
Historical significance: until that time, most SFM work dealt with minimal configurations, and noise-free data. Factorization was one of the first “practical SFM algorithms”

Recall: World to Camera Transform

\[
P_C = R \left( P_W - C \right)
\]

\[
\begin{bmatrix}
P_C^x \\
P_C^y \\
P_C^z \\
1
\end{bmatrix}
= \begin{bmatrix}
1 & 0 & 0 & -c_x \\
0 & 1 & 0 & -c_y \\
0 & 0 & 1 & -c_z \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
P_W^x \\
P_W^y \\
P_W^z \\
1
\end{bmatrix}
\]

\[
P_C = M_{ext} \cdot P_W
\]
**Perspective Projection**

\[ x = \frac{fX}{Z} \]
\[ y = \frac{fY}{Z} \]

- Non-linear equations
- Any point on the ray OP has image p !

**Simplification: Weak Perspective**

\[ x = \frac{fX}{Z_o} \]
\[ y = \frac{fY}{Z_o} \]

Weak perspective = Parallel projection (parallel lines remain parallel) + Scaling to simulate change in size due to object distance.

**Perspective Matrix Equation**

\[
\begin{bmatrix}
    x' \\
    y' \\
    z'
\end{bmatrix} = \begin{bmatrix}
    f & 0 & 0 \\
    0 & f & 0 \\
    0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
    X \\
    Y \\
    Z
\end{bmatrix}
\]

Using homogeneous coordinates:

\[ x = \frac{x'}{z'} \]
\[ y = \frac{y'}{z'} \]

**Simpler: Orthographic Projection**

\[ x = X \]
\[ y = Y \]

Pure parallel projection. Highly simplified case where we even ignore the scaling due to distance.

**Weak Perspective Approximation**

Using homogeneous coordinates:

\[
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix} = \begin{bmatrix}
    f/Z_o & 0 & 0 \\
    0 & f/Z_o & 0 \\
    0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
    X \\
    Y \\
    Z
\end{bmatrix}
\]
Let's Consider Orthographic

Using homogeneous coordinates:

\[
\begin{align*}
x &= X \\
y &= Y
\end{align*}
\]

\[
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
X \\
Y \\
Z \\
1
\end{bmatrix}
\]

Combine with External Params

\[
\begin{bmatrix}
x \\
y
\end{bmatrix} = \begin{bmatrix}
r_{11} & r_{12} & r_{13} & 0 \\
r_{21} & r_{22} & r_{23} & 0 \\
r_{31} & r_{32} & r_{33} & 0 \\
0 & 0 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
p^{w}_x \\
p^{w}_y \\
p^{w}_z \\
1
\end{bmatrix}
\]

Orthographic: Algebraic Equation

\[
\begin{bmatrix}
x \\
y
\end{bmatrix} = \begin{bmatrix}
r_{11} & r_{12} & r_{13} & 0 \\
r_{21} & r_{22} & r_{23} & 0 \\
r_{31} & r_{32} & r_{33} & 0 \\
0 & 0 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
p^{w}_x - c_i \\
p^{w}_y - c_i \\
p^{w}_z - c_i \\
1
\end{bmatrix}
\]

Multiple Points, Multiple Frames

Notation (attack of the killer subscripts)

N points \( P_1, P_2, \ldots, P_j, \ldots, P_N \)

\[
\begin{align*}
x &= i^T ( P - T ) \\
y &= j^T ( P - T )
\end{align*}
\]

Factorization Approach

N points \( P_1, P_2, \ldots, P_j, \ldots, P_N \)

(We want to recover these)

\[
\begin{align*}
x_{ij} &= i^T ( P_j - T_i ) \\
y_{ij} &= j^T ( P_j - T_i )
\end{align*}
\]

Note that absolute position of the set of points is something that cannot be uniquely recovered, so…

First Trick: set the origin of the world coordinate system to be the center of pass of the N points!

\[
\frac{1}{N} \sum_{i=1}^{N} P_i = 0
\]
Factorization Approach

Centroid at 0:
\[ x_i = i^T (P_j - T_i) \]
\[ y_j = j^T (P_j - T_i) \]
\[ \frac{1}{N} \sum_{i=1}^{N} P_i = 0 \]

Implication:
\[ \bar{x}_i = \frac{1}{N} \sum_{i=1}^{N} i^T (P_j - T_i) = \frac{1}{N} i^T P_j - \frac{1}{N} \sum_{i=1}^{N} T_i = 0 - T_i^T T_i \]

Note: this is the center of mass of x coordinates in frame t

What have we accomplished so far?
1) Removed unknown camera locations from equations.
2) More importantly, we can now write everything as a big matrix equation...

Form a matrix of centered image points.
\[ 2F \times N \]

All N points in one frame

Tracking one point through all F frames

Form a matrix of centered image points:
\[ 2F \times 3 \]
\[ P_i \quad \underline{P_o} \]

Second Trick: subtract off the center of mass of the 2D points in each frame. (Centering)
\[ \bar{x}_i = i^T (P_j - T_i) \]
\[ \bar{y}_j = j^T (P_j - T_i) \]
\[ \bar{y}_i = y_i - \bar{y}_i = j^T P_i \]
**Factorization Approach**

\[
W = M S
\]

- **Centered measurement matrix**
- **“Motion” (camera rotation)**
- **Structure (3D scene points)**

**Rank Theorem:**
The 2FxN centered observation matrix has at most rank 3.

**Proof:**
Trivial, using the properties:
- rank of mxn matrix is at most min(m,n)
- rank of A*B is at most min(rank(A),rank(B))

**Rank of a Matrix**

What is rank of a matrix, anyways?

- Number of columns (rows) that are linearly independent.
- If matrix A is treated as a linear map, it is the intrinsic dimension of the space that is mapped into.

This matrix would have rank 1

**Factorization Rank Theorem**

**Importance of rank theorem:**
- Shows that video data is highly redundant
- Precisely quantifies the redundancy
- Suggests an algorithm for solving SFM

**Form SVD of measurement matrix W**

\[
W = U D V^T
\]

- Diagonal matrix with eigenvalues sorted in decreasing order:
  - \( d_{11} \geq d_{22} \geq d_{33} \geq \ldots \)

**Another useful rank property:**
Rank of a matrix is equal to the number of nonzero eigenvalues.
- \( d_{11}, d_{22}, d_{33} \) are only nonzero eigenvalues (the rest are 0).
**Factorization Approach**

\[ 2 \times F \times N = 2 \times F \times 2 \times F \times N \times N \times N \]

Eigenvalues in decreasing order

Rank theorem says:

These 3 are nonzero

In practice, due to noise, there may be more than 3 nonzero eigenvalues, but rank theorem tells us to ignore all but the largest three.

\[ 3 \times 3 \]

\[ W = U' \ D' \ V'^T \]

\[ W = M \ S \]

**Annoying Details**

\[ W = (U' \ D'^{1/2}) (D'^{1/2} V'^T) \]

\[ W = M \ S \]

Problems:

1) This is not a unique decomposition.
   eg: \((M \ Q) (Q^{-1} \ S) = M \ Q \ Q^{-1} \ S = M \ S\)

2) \(i^T, j^T\) pairs (rows of \(M\)) are not necessarily orthogonal

**Solving the Annoying Details**

Solution to both problems:

Solve for \(Q\) such that appropriate rows of \(M\) satisfy

\[\begin{align*}
    i^T Q Q^T i & = 1 \\
    i^T Q Q^T j & = 0 \\
    j^T Q Q^T j & = 1
\end{align*}\]

unit vectors

orthogonal

3N equations in 9 unknowns

But these are nonlinear equations

linearize and iterate

(see Exercise 8.8 in book for Newton’s method)

(alternative approach is to use Cholesky decomposition – outside our scope)
Factorization Summary

Assumptions
- orthographic camera
- N non-coplanar points tracking in \( F \geq 3 \) frames

Form the centered measurement matrix \( W = [X ; Y] \)
- where \( x_{ij} = x_i - m_{xj} \)
- where \( y_{ij} = y_i - m_{yj} \)
- \( m_x \) and \( m_y \) are mean of points in frame \( i \)
- \( j \) ranges over set of points

Rank theorem: The centered measurement matrix has a rank of at most 3

Factorization Algorithm

1) Form the centered measurement matrix \( W \) from \( N \) points tracked over \( F \) frames.
2) Compute SVD of \( W = U D V^T \)
   - \( U \) is \( 2Fx2F \)
   - \( D \) is \( 2FxN \)
   - \( V^T \) is \( NxN \)
3) Take largest 3 eigenvalues, and form
   - \( D' = 3x3 \) diagonal matrix of largest eigenvalues
   - \( U' = 2Fx3 \) matrix of corresponding column vectors from \( U \)
   - \( V'T = 3xN \) matrix of corresponding row vectors from \( V^T \)
4) Define
   \( M = U' D'^{1/2} \) and \( S = D'^{1/2} V'T \)
5) Solve for \( Q \) that makes appropriate rows of \( M \) orthogonal
6) Final solution is
   \( M^* = M Q \) and \( S^* = Q^{-1} S \)

Sample Results