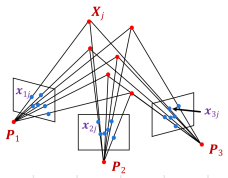
Lec 8. structure from motion

① problem formulation:

- Given: m images of n fixed 3D points such that (ignoring visibility)

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

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



— 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 remain exactly the same:

$$x \cong PX = \left(\frac{1}{k}P\right)(kX)$$

- Without a reference measurement, it is impossible to recover the absolute scale of the scene!
- In general, if we transform the scene using a transformation Q and apply the inverse transformation to the camera matrices, then the image observations do not change:

$$x \cong PX = (PQ^{-1})(QX)$$

→ we can reconstruct it into a projective ambiguity Q: a general full-rank 4x4 matrix

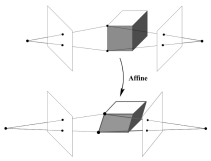
1) affine ambiguity

- If we impose parallelism constraints, we can get a reconstruction up to an *affine* ambiguity:

$$x \cong PX = (PQ_A^{-1})(Q_A X)$$

3×3 full-rank matrix; 3×1 translation vector

$$Q_A = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix}$$



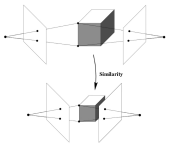
2) Similarity ambiguity

- A reconstruction that obeys orthogonality constraints on camera parameters and/or scene

$$x \cong PX = (PQ_S^{-1})(Q_S X)$$

3×3 rotation matrix; 3×1 translation vector

$$Q_S = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix}$$



② Affine structure from motion

1) general affine projection: $P = \begin{bmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix}$

~ in nonhomogenous cord.: $\begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} = AX + t$

Given m images of n fixed 3D points such that:

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

problem use $m \cdot n$ correspondences x_{ij} to estimate m projection matrix A_i and translation vector t_i, and n points X_j

$$x_{ij} = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix} \cdot \begin{bmatrix} x_j \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix} Q^{-1} Q \begin{pmatrix} x_j \\ 1 \end{pmatrix}$$

Affine structure from motion pipline:

① normalize for each view (image) → codes:

- First, center the data by subtracting the centroid of the image points in each view:

$$\hat{x}_{ij} = x_{ij} - \frac{1}{n}\sum_{k=1}^{n} x_{ik} \longrightarrow \hat{x}_{ij} = A_i \hat{X}_j$$

$$= A_i X_j + t_i - \frac{1}{n}\sum_{k=1}^{n}(A_i X_k + t_i)$$

$$= A_i\left(X_j - \frac{1}{n}\sum_{k=1}^{n} X_k\right)$$

$$= A_i \hat{X}_j$$

so there is no translation vector t here

② $D = MS \longrightarrow X = A \cdot X$

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

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

M: cameras ($2m \times 3$); S: points ($3 \times n$)

S → $\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$ ($3 \times 1 \times n$)

$2m \times n$

- What must be the rank of the measurement matrix $D = MS$?

m camera, n point; $\hat{x}_{11} = \begin{pmatrix} \hat{x}_{11} \\ \hat{y}_{11} \end{pmatrix}$

$2m \times 3$ → $P = \begin{bmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix}$ ($m \times 2 \times 3$)

Factorizing the measurment matrix (分解)

given D, how to figure out M, S ?

method: SVD

first step

- Keep top 3 singular values:
- This is the closest approximation of D with a rank-3 matrix in terms of Frobenius norm

$$D_{2m\times n} = U_{3\ 2m\times 3} \times \Sigma_{3\ 3\times 3} \times V_3^T{}_{3\times n}$$

- What to do about Σ_3?
- One solution: $M = U_3\Sigma_3^{\frac{1}{2}}, S = \Sigma_3^{\frac{1}{2}}V_3^T$

This is a good way to eliminate noise "rank Theorem for Noisy Measurment"

second step

$$D_{2m\times n} = M_{2m\times 3} \times Q_{3\times 3} \times Q^{-1}_{3\times 3} \times S_{3\times n}$$

We can estimate Q to give the camera matrices in M desirable properties, like orthographic projection

result

$$D = \begin{bmatrix} A_1 Q \\ A_2 Q \\ \vdots \\ A_m Q \end{bmatrix} \begin{bmatrix} Q^{-1}X_1 & Q^{-1}X_2 & \cdots & Q^{-1}X_n \end{bmatrix}$$

$2m \times 3$; $3 \times n$

- One possible solution:

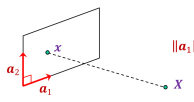
$$D_{2m\times n} = M_{2m\times 3} \times S_{3\times n}$$

$M = U_3\Sigma_3^{\frac{1}{2}}$, $S = \Sigma_3^{\frac{1}{2}}V_3^T$

- Are there other s

To be specific: that each camera matrix A_iQ represent orthographic projection

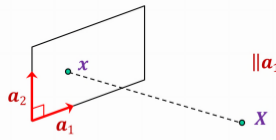
- Let a_1 and a_2 be the rows of a 2×3 orthographic projection matrix. Then



$$a_1 \cdot a_2 = 0$$
$$\|a_1\|^2 = \|a_2\|^2 = 1$$

- This translates into $3m$ constraints on the 9 entries of Q:
$$(A_iQ)(A_iQ)^T = A_i(QQ^T)A_i^T = I_{2\times2}, \qquad i = 1, \ldots, m$$
  - Are the constraints linear?
  - First, solve for $L = QQ^T$
  - Recover Q from L by Cholesky decomposition
  - Update M to MQ, S to $Q^{-1}S$

- Let a_1 and a_2 be the rows of a 2×3 orthographic projection matrix. Then



$$a_1 \cdot a_2 = 0$$
$$\|a_1\|^2 = \|a_2\|^2 = 1$$

to make sure

- This translates into $3m$ constraints on the 9 entries of Q:

$$(A_i Q)(A_i Q)^T = A_i (Q Q^T) A_i^T = I_{2\times 2}, \qquad i = 1, \ldots, m$$

  - Are the constraints linear?
  - First, solve for $L = QQ^T$
  - Recover Q from L by Cholesky decomposition
  - Update M to MQ, S to $Q^{-1}S$

algorithm:

1. Compute the singular-value decomposition $\tilde{\mathbf{W}} = \mathbf{O}_1 \Sigma \mathbf{O}_2$.
2. Define $\hat{\mathbf{R}} = \mathbf{O}_1'(\Sigma')^{1/2}$ and $\hat{\mathbf{S}} = (\Sigma')^{1/2}\mathbf{O}_2'$, , where the primes refer to the block partitioning defined in (13).
3. Compute the matrix $\mathbf{Q}$ in equations (15) by imposing the metric constraints (equations (16)).
4. Compute the rotation matrix $\mathbf{R}$ and the shape matrix $\mathbf{S}$ as $\mathbf{R} = \hat{\mathbf{R}}\mathbf{Q}$ and $\mathbf{S} = \mathbf{Q}^{-1}\hat{\mathbf{S}}$.
5. If desired, align the first camera reference system with the world reference system by forming the products $\mathbf{R}\mathbf{R}_0$ and $\mathbf{R}_0^T\mathbf{S}$, where the orthonormal matrix $\mathbf{R}_0 = [\mathbf{i}_1\ \mathbf{j}_1\ \mathbf{k}_1]$ rotates the first camera reference system into the identity matrix.

how to deal with Q?

Now $\hat{W} = O_1' \Sigma' O_2'$

init, $\hat{R} = O_1' \cdot [\Sigma']^{\frac{1}{2}}$ $\hat{S} = [\Sigma']^{\frac{1}{2}} \cdot O_2'$

$\hat{W} = \hat{R}\cdot\hat{S} = \hat{R}\cdot Q \cdot Q^{-1}\cdot\hat{S}$ → shape matrix

↳ rotation matrix

A_i (2×3) A_i^T (3×2)

Q (3×3) Q^T (3×3)

A (2m×3) $Q\cdot Q^T$ · A^T (3×2m) = I (2m×2m)

(3×3)

$\therefore L = A^{-1}\cdot I \cdot A^{-T}$

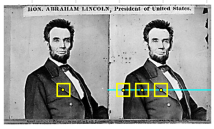
$Q^{-1}\cdot S$

(3×3) (3×2or5)

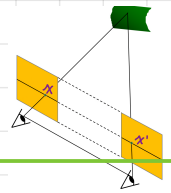
Lec 19 stero vision

— problem formulation: given stereo pair (assumed calibrated)
figure out dense depth map

① Basic stereo matching algorithm



- For each pixel in the first image
  - Find corresponding epipolar line in the right image ①
  - Examine all pixels on the epipolar line and pick the best match ②
  - Triangulate the matches to get depth information ③
- Simplest case: epipolar lines are corresponding scanlines
  - When does this happen?



- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at the same height
- Focal lengths are the same
- Then epipolar lines fall along horizontal scan lines of the images

→ Triangulation: not using $X = P \cdot X$ (since we don't know P)

$x'^T \cdot E \cdot x = 0$, $E = [t_\times]R$ and $R = I$ $t = (t, 0, 0)$

$\therefore E = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t \\ 0 & t & 0 \end{bmatrix}$

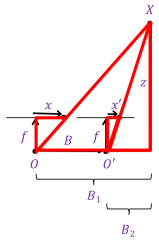
$(u', v', 1) \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t \\ 0 & t & 0 \end{bmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = 0$

$\therefore (u', v', 1) \begin{pmatrix} 0 \\ -t \\ tv \end{pmatrix} = 0$

$-tv + tv' = 0$

$v = v'$ y-coord. is same

Depth from disparity



$\frac{x}{f} = \frac{B_1}{z}$ $\frac{x'}{f} = \frac{B_2}{z}$

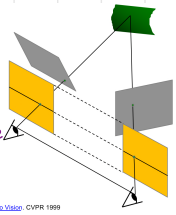
$\frac{x - x'}{f} = \frac{B_1 - B_2}{z}$

$x - x' = \frac{fB}{z}$

$z = \frac{fB}{x - x'}$

- If the image planes are not parallel, we can find homographies to project each view onto a common plane parallel to the baseline

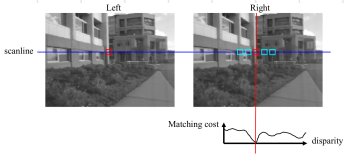
only change the rotation, don't change camera center



C. Loop and Z. Zhang. Computing Rectifying Homographies for Stereo Vision. CVPR 1999

matching:

1) first method: Local stero matching algorithm



- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

- Smaller window:
  - \+ More detail
  - – More noise
- Larger window:
  - \+ Smoother disparity maps
  - – Less detail

2) Beyond local stereo matching

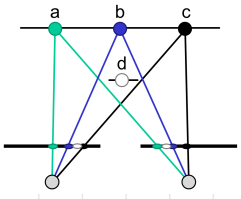
Non-local constraint:

① Uniqueness

Each point in one image should match at most one point in the other image

② ordering

Corresponding points should appear in the same order
Is ordering always preserved in real life?



if have no depth

a d b c a b d c

③ smothness