Parameter estimation
class 5

Multiple View Geometry
Comp 290-089
Marc Pollefeys
Content

• **Background:** Projective geometry (2D, 3D), Parameter estimation, Algorithm evaluation.
• **Single View:** Camera model, Calibration, Single View Geometry.
• **Two Views:** Epipolar Geometry, 3D reconstruction, Computing F, Computing structure, Plane and homographies.
• **Three Views:** Trifocal Tensor, Computing T.
• **More Views:** N-Linearities, Multiple view reconstruction, Bundle adjustment, auto-calibration, Dynamic SfM, Cheirality, Duality
<table>
<thead>
<tr>
<th>Jan. 7, 9</th>
<th>Intro &amp; motivation</th>
<th>Projective 2D Geometry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan. 14, 16</td>
<td>(no class)</td>
<td>Projective 2D Geometry</td>
</tr>
<tr>
<td>Jan. 21, 23</td>
<td>Projective 3D Geometry</td>
<td>(no class)</td>
</tr>
<tr>
<td>Jan. 28, 30</td>
<td>Parameter Estimation</td>
<td>Parameter Estimation</td>
</tr>
<tr>
<td>Feb. 4, 6</td>
<td>Algorithm Evaluation</td>
<td>Camera Models</td>
</tr>
<tr>
<td>Feb. 11, 13</td>
<td>Camera Calibration</td>
<td>Single View Geometry</td>
</tr>
<tr>
<td>Feb. 18, 20</td>
<td>Epipolar Geometry</td>
<td>3D reconstruction</td>
</tr>
<tr>
<td>Mar. 4, 6</td>
<td>Planes &amp; Homographies</td>
<td>Trifocal Tensor</td>
</tr>
<tr>
<td>Mar. 18, 20</td>
<td>Three View Reconstruction</td>
<td>Multiple View Geometry</td>
</tr>
<tr>
<td>Mar. 25, 27</td>
<td>MultipleView Reconstruction</td>
<td>Bundle adjustment</td>
</tr>
<tr>
<td>Apr. 1, 3</td>
<td>Auto-Calibration</td>
<td>Papers</td>
</tr>
<tr>
<td>Apr. 8, 10</td>
<td>Dynamic SfM</td>
<td>Papers</td>
</tr>
<tr>
<td>Apr. 15, 17</td>
<td>Cheirality</td>
<td>Papers</td>
</tr>
<tr>
<td>Apr. 22, 24</td>
<td>Duality</td>
<td>Project Demos</td>
</tr>
</tbody>
</table>
Projective 3D Geometry

- Points, lines, planes and quadrics
- Transformations
- $\Pi_\infty$, $\omega_\infty$ and $\Omega$
Singular Value Decomposition

\[ A = U \Sigma V^T \]

Homogeneous least-squares

\[ \min \|AX\| \text{ subject to } \|X\| = 1 \quad \text{solution } X = V_n \]
Parameter estimation

• **2D homography**
  Given a set of \((x_i,x_i')\), compute \(H (x_i' = H x_i)\)

• **3D to 2D camera projection**
  Given a set of \((X_i,x_i)\), compute \(P (x_i = PX_i)\)

• **Fundamental matrix**
  Given a set of \((x_i,x_i')\), compute \(F (x_i'^t F x_i = 0)\)

• **Trifocal tensor**
  Given a set of \((x_i,x_i',x_i'')\), compute \(T\)
Number of measurements required

- At least as many independent equations as degrees of freedom required
- Example: \[ x' = Hx \]

\[
\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}
\]

2 independent equations / point
8 degrees of freedom

4x2 \geq 8
Approximate solutions

- Minimal solution
  4 points yield an exact solution for H
- More points
  - No exact solution, because measurements are inexact ("noise")
  - Search for "best" according to some cost function
  - Algebraic or geometric/statistical cost
Gold Standard algorithm

- Cost function that is optimal for some assumptions
- Computational algorithm that minimizes it is called “Gold Standard” algorithm
- Other algorithms can then be compared to it
Direct Linear Transformation (DLT)

\[ \mathbf{x}_{ii}^{\prime\prime} \propto \mathbf{Hx}_i = 0 \]

\[ \mathbf{x}_i^{\prime} = (x_i^{\prime}, y_i^{\prime}, w_i^{\prime})^T \]

\[ \mathbf{Hx}_i = \begin{pmatrix} h_1^T x_i \\ h_2^T x_i \\ h_3^T x_i \end{pmatrix} \]

\[ \mathbf{x}_i^{\prime} \times \mathbf{Hx}_i = \begin{pmatrix} y_i^{\prime} h_3^T x_i - w_i^{\prime} h_2^T x_i \\ w_i^{\prime} h_1^T x_i - x_i^{\prime} h_3^T x_i \\ x_i^{\prime} h_2^T x_i - y_i^{\prime} h_1^T x_i \end{pmatrix} \]

\[
\begin{bmatrix}
0^T & -w_i^{\prime}x_i^T & y_i^{\prime}x_i^T \\
-w_i^{\prime}x_i^T & 0^T & -x_i^{\prime}x_i^T \\
-y_i^{\prime}x_i^T & x_i^{\prime}x_i^T & 0^T
\end{bmatrix}
\begin{pmatrix}
h_1^T \\
h_2^T \\
h_3^T
\end{pmatrix} = 0
\]

\[ A_i \mathbf{h} = 0 \]
Direct Linear Transformation (DLT)

- Equations are linear in $h$
  \[ A_i h = 0 \]
- Only 2 out of 3 are linearly independent (indeed, 2 eq/pt)
  \[
  \begin{bmatrix}
  0^T & -w'_ix'_i^T & y'_ix'_i^T \\
  0^T & -w'_ix'_i^T & y'_ix'_i^T \\
  w'_ix'_i^T & 0^T & -x'_ix'_i^T \\
  w'_ix'_i^T & 0^T & -x'_ix'_i^T \\
  y'_ix'_i & x'_ix'_i^T & 0^T
  \end{bmatrix}
  \begin{bmatrix}
  h^1 \\
  h^2 \\
  h^3
  \end{bmatrix} = 0
\]
  (only drop third row if $A_i^3 \neq 0$
- Holds for any homogeneous representation, e.g. $(x'_i, y'_i, 1)$
Direct Linear Transformation (DLT)

- Solving for $H$

\[
\begin{bmatrix}
A_1 \\
A_2 \\
A_3 \\
A_4
\end{bmatrix}h = 0
\]

Size $A$ is 8x9 or 12x9, but rank 8

Trivial solution is $h=0^T_9$ is not interesting
1-D null-space yields solution of interest
pick for example the one with $\|h\| = 1$
Direct Linear Transformation (DLT)

- Over-determined solution

$$\begin{bmatrix}
A_1 \\
A_2 \\
\vdots \\
A_n
\end{bmatrix} h = 0$$

No exact solution because of inexact measurement i.e. “noise”

Find approximate solution
- Additional constraint needed to avoid 0, e.g. $\|h\| = 1$
- $Ah = 0$ not possible, so minimize $\|Ah\|$
DLT algorithm

Objective
Given \( n \geq 4 \) 2D to 2D point correspondences \( \{x_i \leftrightarrow x'_i\}\), determine the 2D homography matrix \( H \) such that \( x'_i = Hx_i \).

Algorithm
(i) For each correspondence \( x_i \leftrightarrow x'_i \) compute \( A_i \). Usually only two first rows needed.
(ii) Assemble \( n \) 2x9 matrices \( A_i \) into a single 2nx9 matrix \( A \)
(iii) Obtain SVD of \( A \). Solution for \( h \) is last column of \( V \)
(iv) Determine \( H \) from \( h \)
Inhomogeneous solution

Since $h$ can only be computed up to scale, pick $h_j = 1$, e.g. $h_9 = 1$, and solve for 8-vector $\sim h$

$$
\begin{bmatrix}
0 & 0 & 0 & -x_i w_i' & -y_i w_i' & -w_i w_i' & x_i y_i' & y_i y_i' \\
x_i w_i' & y_i w_i' & w_i w_i' & 0 & 0 & 0 & x_i y_i' & y_i y_i'
\end{bmatrix}
\sim h = \begin{pmatrix}
-w_i y_i' \\
w_i x_i'
\end{pmatrix}
$$

Solve using Gaussian elimination (4 points) or using linear least-squares (more than 4 points)

However, if $h_9 = 0$ this approach fails
also poor results if $h_9$ close to zero
Therefore, not recommended

Note $h_9 = H_{33} = 0$ if origin is mapped to infinity

$$
1^T H x_0 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} H \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0
$$
Degenerate configurations

Constraints: \( x'_i \times H x_i = 0 \quad i=1,2,3,4 \)

Define: \( H^* = x'_4 l^T \)

Then,
\[
H^* x_i = x'_4 (l^T x_i) = 0, \quad i = 1,2,3
\]
\[
H^* x_4 = x'_4 (l^T x_4) = kx'_4
\]

\( H^* \) is rank-1 matrix and thus not a homography

If \( H^* \) is unique solution, then no homography mapping \( x_i \rightarrow x'_i \) (case B)
If further solution \( H \) exist, then also \( \alpha H^* + \beta H \) (case A)
(2-D null-space in stead of 1-D null-space)
Solutions from lines, etc.

2D homographies from 2D lines

\[ l_i' = H^T l_i \quad \text{Ah} = 0 \]

Minimum of 4 lines

3D Homographies (15 dof)

Minimum of 5 points or 5 planes

2D affinities (6 dof)

Minimum of 3 points or lines

Conic provides 5 constraints

Mixed configurations?
Cost functions

- Algebraic distance
- Geometric distance
- Reproduction error
- Comparison
- Geometric interpretation
- Sampson error
Algebraic distance

DLT minimizes $\|Ah\|

e = Ah \quad \text{residual vector}

e_i \quad \text{partial vector for each } (x_i \leftrightarrow x_i')

algebraic error vector

$$d_{\text{alg}}(x'_i, Hx_i)^2 = \|e_i\|^2 = \left\| \begin{bmatrix} 0^T & -w'_ix_i^T & -y'_ix_i^T \\ -w'_ix_i^T & 0^T & -x'_ix_i^T \end{bmatrix} h \right\|_2^2$$

algebraic distance

$$d_{\text{alg}}(x_1, x_2)^2 = a_1^2 + a_2^2 \quad \text{where } a = (a_1, a_2, a_3)^T = x_1 \times x_2$$

$$\sum_i d_{\text{alg}}(x'_i, Hx_i)^2 = \sum_i \|e_i\|^2 = \|Ah\|^2 = \|e\|^2$$

Not geometrically/statistically meaningfull, but given good normalization it works fine and is very fast (use for initialization)
Geometric distance

\( \mathbf{X} \) measured coordinates
\( \hat{\mathbf{X}} \) estimated coordinates
\( \overline{\mathbf{X}} \) true coordinates

\( d(.,.) \) Euclidean distance (in image)

Error in one image

\[
\hat{H} = \arg\min_H \sum_i d(x_i', H\overline{x}_i)^2
\]

e.g. calibration pattern

Symmetric transfer error

\[
\hat{H} = \arg\min_H \sum_i d(x_i, H^{-1}x_i')^2 + d
\]

Reprojection error

\[
(\hat{H}, \hat{x}_i, \hat{x}_i') = \arg\min_{H, \hat{x}_i, \hat{x}_i'} \sum_i d(x_i, \hat{x}_i)^2 + d(x_i', \hat{x}_i')^2
\]

subject to \( \hat{x}_i' = \hat{H}\hat{x}_i \)
Reprojection error

\[ d(x, H^{-1}x')^2 + d(x', Hx)^2 \]

\[ d(x, \hat{x})^2 + d(x', \hat{x}')^2 \]
Comparison of geometric and algebraic distances

Error in one image

\[ x'_i = (x'_i, y'_i, w'_i)^T \quad \hat{x}'_i = (\hat{x}'_i, \hat{y}'_i, \hat{w}'_i)^T = H\bar{x} \]

\[
\begin{bmatrix}
 0^T \\
w'_i \hat{x}'_i \\
\end{bmatrix} = \begin{bmatrix}
 0^T \\
 w'_i \hat{x}'_i \\
\end{bmatrix}
= e_i \begin{bmatrix}
 0^T \\
 \begin{pmatrix}
 1 & 0 & 0 \\
 w'_i \hat{x}'_i \\
 w'_i \hat{y}'_i \\
\end{pmatrix}
\end{bmatrix}
\]

\[
d_{\text{alg}}(x'_i, \hat{x}'_i)^2 = (y'_i \hat{w}'_i - w'_i \hat{y}'_i)^2 + (w'_i \hat{x}'_i - x'_i \hat{w}'_i)^2
\]

\[
d(x'_i, \hat{x}'_i)^2 = \left(\left(\frac{y'_i}{w'_i} - \frac{\hat{y}'_i}{\hat{w}'_i}\right)^2 + \left(\frac{\hat{x}'_i}{\hat{w}'_i} - \frac{x'_i}{w'_i}\right)^2\right)^{1/2}
\]

\[ = d_{\text{alg}}(x'_i, \hat{x}'_i) / w'_i \hat{w}'_i \]

\[ w'_i = 1 \text{ typical, but not } \hat{w}'_i = h_3 x_i, \text{ except for affinities} \]

For affinities DLT can minimize geometric distance

Possibility for iterative algorithm
Geometric interpretation of reprojection error

Estimating homography~fit surface \( \nu_H \) to points \( X=(x,y,x',y')^T \) in \( \mathbb{R}^4 \).

\[ x'_i \times H x_i = 0 \] represents 2 quadrics in \( \mathbb{R}^4 \) (quadratic in \( X \))

\[ \left\| X_i - \hat{X}_i \right\|^2 = (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (x'_i - \hat{x}'_i)^2 + (y'_i - \hat{y}'_i)^2 \]

\[ = d(x_i, \hat{x}_i)^2 + d(x'_i, \hat{x}'_i)^2 \]

\[ d(x_i, \hat{x}_i)^2 + d(x'_i, \hat{x}'_i)^2 = d_\perp(X_i, \nu_H)^2 \]

Analog to conic fitting

\[ d_{\text{alg}}(x, C)^2 = x^T C x \]

\[ d_\perp(x, C)^2 \]
Sampson error

between algebraic and geometric error

Vector $\hat{X}$ that minimizes the geometric error $\|X - \hat{X}\|^2$ is the closest point on the variety $\mathcal{V}_H$ to the measurement $X$

Sampson error: 1st order approximation of $\hat{X}$

$$Ah = C_H(X) = 0$$

$$C_H(X + \delta_X) = C_H(X) + \frac{\partial C_H}{\partial X} \delta_X \quad \delta_X = \hat{X} - X \quad C_H(\hat{X}) = 0$$

$$C_H(X) + \frac{\partial C_H}{\partial X} \delta_X = 0 \quad J\delta_X = -e \quad \text{with } J = \frac{\partial C_H}{\partial X}$$

Find the vector $\delta_X$ that minimizes $\|\delta_X\|$ subject to $J\delta_X = -e$
Find the vector $\delta_x$ that minimizes $\|\delta_x\|$ subject to $J\delta_x = -e$

Use Lagrange multipliers:

minimize $\delta_x^T \delta_x - 2\lambda (J\delta_x + e) = 0$

derivatives $2\delta_x - 2\lambda^T J = 0^T \quad \Rightarrow \quad \delta_x = J^T \lambda$

$2(J\delta_x + e) = 0 \quad \Rightarrow \quad JJ^T \lambda + e = 0$

$\Rightarrow \lambda = -(JJ^T)^{-1} e$

$\Rightarrow \delta_x = -J^T(JJ^T)^{-1} e$

$\hat{X} = X + \delta_x \quad \|\delta_x\|^2 = \delta_x^T \delta_x = e^T(JJ^T)^{-1} e$
Sampson error

between algebraic and geometric error

Vector $\hat{X}$ that minimizes the geometric error $\|X - \hat{X}\|^2$ is the closest point on the variety $\mathcal{V}_H$ to the measurement $X$.

Sampson error: 1st order approximation of $\hat{X}$

$$Ah = C_H(X) = 0$$

$$C_H(X + \delta_X) = C_H(X) + \frac{\partial C_H}{\partial X} \delta_X \quad \delta_X = \hat{X} - X \quad C_H(\hat{X}) = 0$$

$$C_H(X) + \frac{\partial C_H}{\partial X} \delta_X = 0 \quad J\delta_X = -e$$

Find the vector $\delta_X$ that minimizes $\|\delta_X\|$ subject to $J\delta_X = -e$

$$\|\delta_X\|^2 = \delta_X^T \delta_X = e^T(JJ^T)^{-1} e \quad \text{(Sampson error)}$$
Sampson approximation

\[ \| \delta_X \|^2 = e^T (J J^T)^{-1} e \]

A few points

(i) For a 2D homography \( X=(x,y,x',y') \)
(ii) \( e = C_H(X) \) is the algebraic error vector
(iii) \( J = \frac{\partial C_H}{\partial X} \) is a 2x4 matrix, e.g. \( J_{11} = \frac{\partial}{\partial x} \left( -w'_i x_i^T h^2 + y'_i x_i^T h^3 \right) / \partial x = -w'_i h_{21} + y'_i h_{31} \)
(iv) Similar to algebraic error \( \| e \|^2 = e^T e \) in fact, same as Mahalanobis distance \( \| e \|^2_{JJ^T} \)
(v) Sampson error independent of linear reparametrization (cancels out in between \( e \) and \( J \))
(vi) Must be summed for all points \( \sum e^T (J J^T)^{-1} e \)
(vii) Close to geometric error, but much fewer unknowns
Statistical cost function and Maximum Likelihood Estimation

- Optimal cost function related to noise model
- Assume zero-mean isotropic Gaussian noise (assume outliers removed)

\[
\Pr(x) = \frac{1}{2\pi\sigma^2} e^{-d(x,\bar{x})^2/(2\sigma^2)}
\]

Error in one image

\[
\Pr(\{x_i'\} \mid H) = \prod_i \frac{1}{2\pi\sigma^2} e^{-d(x_i',H\bar{x}_i)^2/(2\sigma^2)}
\]

\[
\log \Pr(\{x_i'\} \mid H) = -\frac{1}{2\sigma^2} \sum d(x_i', H\bar{x}_i)^2 + \text{constant}
\]

Maximum Likelihood Estimate

\[
\sum d(x_i', H\bar{x}_i)^2
\]
Statistical cost function and Maximum Likelihood Estimation

- Optimal cost function related to noise model
- Assume zero-mean isotropic Gaussian noise (assume outliers removed)

\[ \Pr(x) = \frac{1}{2\pi\sigma^2} e^{-d(x, \bar{x})^2/(2\sigma^2)} \]

Error in both images

\[ \Pr(\{x_i\}' | H) = \prod_i \frac{1}{2\pi\sigma^2} e^{-d(x_i, \bar{x}_i)^2 + d(x_i', \bar{x}_i)^2)/(2\sigma^2)} \]

Maximum Likelihood Estimate

\[ \sum d(x_i, \hat{x}_i)^2 + d(x_i', \hat{x}_i')^2 \]
Mahalanobis distance

- General Gaussian case

Measurement $X$ with covariance matrix $\Sigma$

$$\|X - \overline{X}\|_\Sigma^2 = (X - \overline{X})^T \Sigma^{-1} (X - \overline{X})$$

Error in two images (independent)

$$\|X - \overline{X}\|_\Sigma^2 + \|X' - \overline{X}'\|_{\Sigma'}^2$$

Varying covariances

$$\sum_i \|X_i - \overline{X}_i\|_{\Sigma_i}^2 + \|X'_i - \overline{X}'_i\|_{\Sigma'_i}^2$$
Next class:
Parameter estimation (continued)

Transformation invariance and normalization
Iterative minimization
Robust estimation
Upcoming assignment

- Take two or more photographs taken from a single viewpoint
- Compute panorama
- Use different measures DLT, MLE

- Use Matlab
- Due Feb. 13