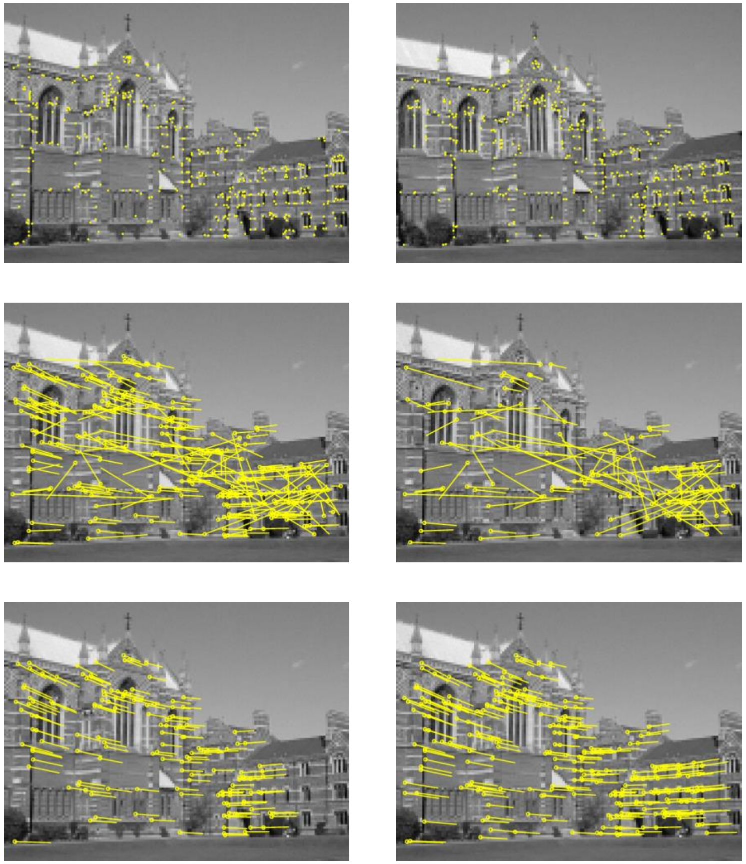
Estimation of Transformations

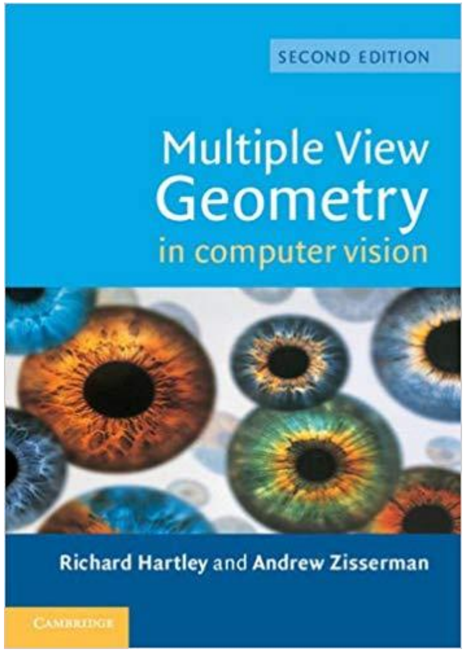
簡韶逸 Shao-Yi Chien
Department of Electrical Engineering
National Taiwan University

Fall 2018

Outline

- Estimation – 2D Projective Transformation





[Slides credit: Marc Pollefeys]

Parameter Estimation

- 2D homography

  Given a set of (x_i, x_i'), compute H ($x_i' = Hx_i$)
- 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:

$$\mathrm{x'} = \mathrm{Hx} \qquad \lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2 independent equations / point
8 degrees of freedom

4x2≥8

Approximate Solutions

- Minimal solution
  - 4 points yield an exact solution for H
- More points
  - Robust estimation algorithms, such as RANSAC
  - 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)

$$\mathrm{x}_i' = \mathrm{H}\mathrm{x}_i \rightarrow \mathrm{x}_i' \times \mathrm{H}\mathrm{x}_i = 0$$

$$\mathrm{x}_i' = \left(x_i', y_i', w_i'\right)^{\mathsf{T}} \quad \mathrm{H}\mathrm{x}_i = \begin{pmatrix} \mathrm{h}^{1^{\mathsf{T}}}\mathrm{x}_i \\ \mathrm{h}^{2^{\mathsf{T}}}\mathrm{x}_i \\ \mathrm{h}^{3^{\mathsf{T}}}\mathrm{x}_i \end{pmatrix}$$

$$\mathrm{x}_i' \times \mathrm{H}\mathrm{x}_i = \begin{pmatrix} y_i'\mathrm{h}^{3^{\mathsf{T}}}\mathrm{x}_i - w_i'\mathrm{h}^{2^{\mathsf{T}}}\mathrm{x}_i \\ w_i'\mathrm{h}^{1^{\mathsf{T}}}\mathrm{x}_i - x_i'\mathrm{h}^{3^{\mathsf{T}}}\mathrm{x}_i \\ x_i'\mathrm{h}^{2^{\mathsf{T}}}\mathrm{x}_i - y_i'\mathrm{h}^{1^{\mathsf{T}}}\mathrm{x}_i \end{pmatrix}$$

$$\begin{bmatrix} 0^{\mathsf{T}} & -w_i'\mathrm{x}_i^{\mathsf{T}} & y_i'\mathrm{x}_i^{\mathsf{T}} \\ w_i'\mathrm{x}_i^{\mathsf{T}} & 0^{\mathsf{T}} & -x_i'\mathrm{x}_i^{\mathsf{T}} \\ -y_i'\mathrm{x}_i^{\mathsf{T}} & x_i'\mathrm{x}_i^{\mathsf{T}} & 0^{\mathsf{T}} \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

$$\mathrm{A}_i\mathrm{h} = 0$$

Direct Linear Transformation (DLT)

- Equations are linear in h

$$\mathrm{A}_i \mathrm{h} = 0$$

- Only 2 out of 3 are linearly independent (indeed, 2 eq/pt)

$$\begin{bmatrix} 0^\top & -w'_i \mathrm{x}_i^\top & y'_i \mathrm{x}_i^\top \\ w'_i \mathrm{x}_i^\top & 0^\top & -x'_i \mathrm{x}_i^\top \\ -y'_i \mathrm{x}_i^\top & x'_i \mathrm{x}_i^\top & 0^\top \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

Direct Linear Transformation (DLT)

- Equations are linear in h

$$\mathrm{A}_i \mathrm{h} = 0$$

- Only 2 out of 3 are linearly independent (indeed, 2 eq/pt)

$$\begin{bmatrix} 0^{\mathsf{T}} & -w_i' \mathrm{x}_i^{\mathsf{T}} & y_i' \mathrm{x}_i^{\mathsf{T}} \\ w_i' \mathrm{x}_i^{\mathsf{T}} & 0^{\mathsf{T}} & -x_i' \mathrm{x}_i^{\mathsf{T}} \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

- Holds for any homogeneous representation, e.g. $(x_i', y_i', 1)$

Direct Linear Transformation (DLT)

- Solving for H

$$\begin{bmatrix} \mathrm{A}_1 \\ \mathrm{A}_2 \\ \mathrm{A}_3 \\ \mathrm{A}_4 \end{bmatrix} \mathrm{h} = 0 \qquad \mathrm{Ah} = 0$$

size A is 8x9 or 12x9, but rank 8

Trivial solution is $\mathrm{h}=0_9^{\mathsf{T}}$ is not interesting

1-D null-space yields solution of interest

pick for example the one with $\|\mathrm{h}\| = 1$

Direct Linear Transformation (DLT)

- Over-determined solution

$$\mathrm{Ah} = 0 \qquad \begin{bmatrix} \mathrm{A}_1 \\ \mathrm{A}_2 \\ \vdots \\ \mathrm{A}_n \end{bmatrix} \mathrm{h} = 0$$

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

Find approximate solution

- Additional constraint needed to avoid 0, e.g. $\|\mathrm{h}\| = 1$
- $\mathrm{Ah} = 0$ not possible, so minimize $\|\mathrm{Ah}\|$

DLT Algorithm

<u>Objective</u>

Given n≥4 2D to 2D point correspondences $\{x_i \leftrightarrow x_i'\}$, determine the 2D homography matrix H such that $x_i'=Hx_i$

<u>Algorithm</u>

(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 2*n*x9 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 $\mathrm{h}_j=1$, e.g. $\mathrm{h}_9=1$, and solve for 8-vector $\tilde{\mathrm{h}}$

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

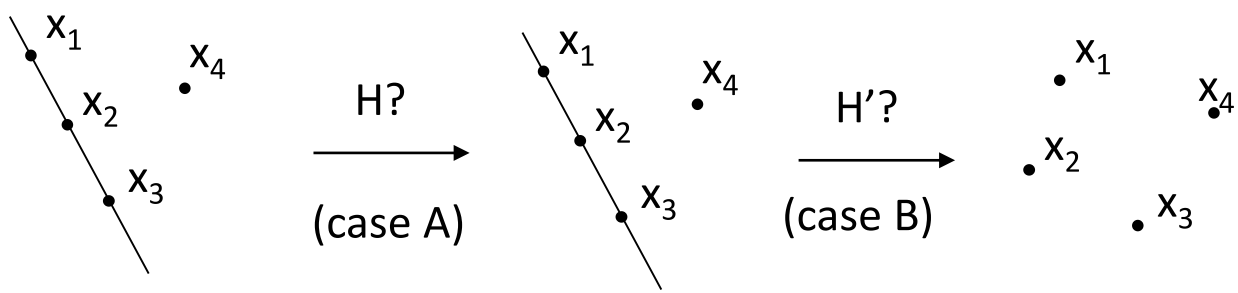
Solve using Gaussian elimination (4 points) or using linear least-squares (more than 4 points)

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

$$\mathrm{l}_\infty^\mathsf{T} \mathrm{H} \mathrm{x}_0 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathrm{H} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

Degenerate Configurations



Constraints: $\mathrm{x}_i' \times \mathrm{H}\mathrm{x}_i = 0$ $\quad i$=1,2,3,4

Define: $\mathrm{H}^* = \mathrm{x}_4' \mathrm{l}^\mathsf{T}$

Then, $\mathrm{H}^* \mathrm{x}_i = \mathrm{x}_4' \left(\mathrm{l}^\mathsf{T} \mathrm{x}_i\right) = 0, \; i = 1,2,3$

$\mathrm{H}^* \mathrm{x}_4 = \mathrm{x}_4' \left(\mathrm{l}^\mathsf{T} \mathrm{x}_4\right) = k\mathrm{x}_4'$

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 $\alpha\mathrm{H}^* + \beta\mathrm{H}$ (case A)
(2-D null-space in stead of 1-D null-space)

Solutions from Lines

2D homographies from 2D lines

$$\mathrm{l}'_i = \mathrm{H}^{\mathsf{T}} \mathrm{l}_i \qquad \mathrm{Ah} = 0$$

Minimum of 4 lines

3D Homographies (15 dof)

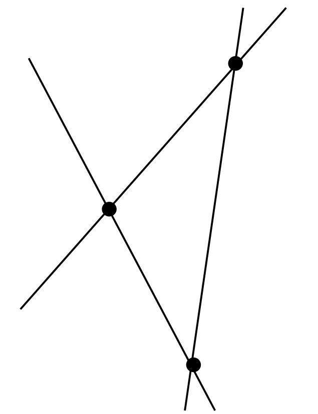
Minimum of 5 points or 5 planes

2D affinities (6 dof)

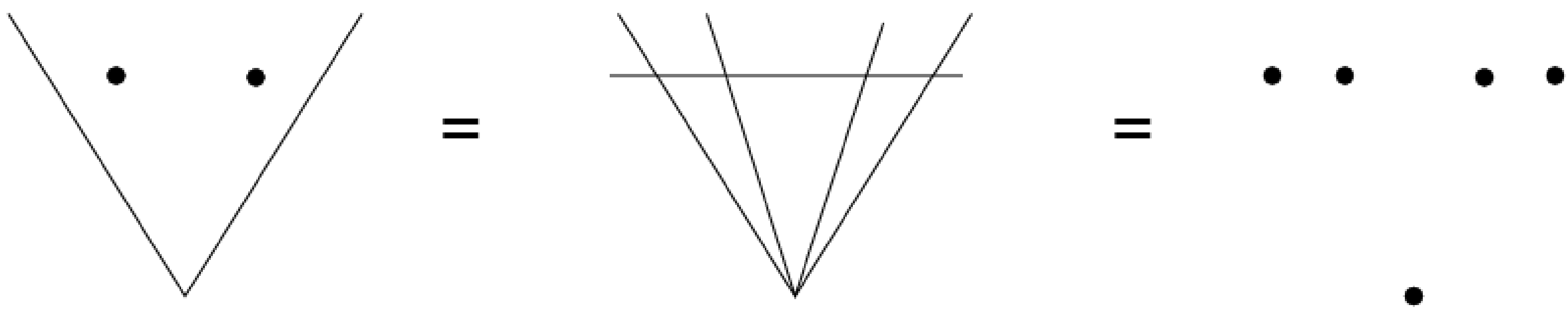
Minimum of 3 points or lines

Conic provides 5 constraints

Solutions from Mixed Type



- 2D homography
  - cannot be determined uniquely from the correspondence of 2 points and 2 line
  - can from 3 points and 1 line or 1 point and 3 lines



Cost Functions

- Algebraic distance
- Geometric distance
- Reprojection error

- Comparison
- Geometric interpretation
- Sampson error

Algebraic Distance

DLT minimizes $\|\mathrm{Ah}\|$

$e = \mathrm{Ah}$ residual vector

e_i partial vector for each ($x_i \leftrightarrow x_i'$)

algebraic error vector

$$d_{\mathrm{alg}}(\mathrm{x}_i', \mathrm{H}\mathrm{x}_i)^2 = \|e_i\|^2 = \left\| \begin{bmatrix} 0^{\mathsf{T}} & -w_i'\mathrm{x}_i^{\mathsf{T}} & -y_i'\mathrm{x}_i^{\mathsf{T}} \\ -w_i'\mathrm{x}_i^{\mathsf{T}} & 0^{\mathsf{T}} & -x_i'\mathrm{x}_i^{\mathsf{T}} \end{bmatrix} \mathrm{h} \right\|^2$$

algebraic distance

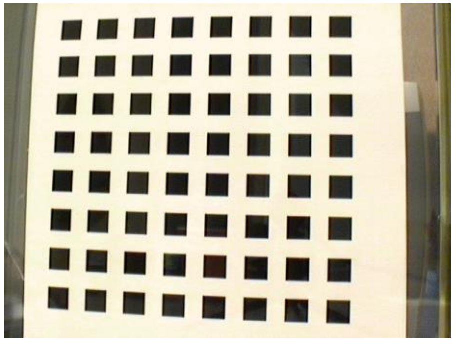
$$d_{\mathrm{alg}}(\mathrm{x}_1, \mathrm{x}_2)^2 = a_1^2 + a_2^2 \text{ where } \mathrm{a} = (a_1, a_2, a_3)^{\mathrm{T}} = \mathrm{x}_1 \times \mathrm{x}_2$$

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

Not geometrically/statistically meaningfull, but given good normalization it works fine and is very fast (use for initialization for non-linear minimization)

Geometric Distance

x measured coordinates
$\hat{\mathrm{x}}$ estimated coordinates
$\bar{\mathrm{x}}$ true coordinates
$d(.,.)$ Euclidean distance (in image)



e.g. calibration pattern

Error in one image

$$\hat{\mathrm{H}} = \underset{\mathrm{H}}{\operatorname{argmin}} \sum_i d\left(\mathrm{x}'_i, \mathrm{H}\bar{\mathrm{x}}_i\right)^2$$

Symmetric transfer error

$$\hat{\mathrm{H}} = \underset{\mathrm{H}}{\operatorname{argmin}} \sum_i d\left(\mathrm{x}_i, \mathrm{H}^{-1}\mathrm{x}'_i\right)^2 + d\left(\mathrm{x}'_i, \mathrm{H}\mathrm{x}_i\right)^2$$

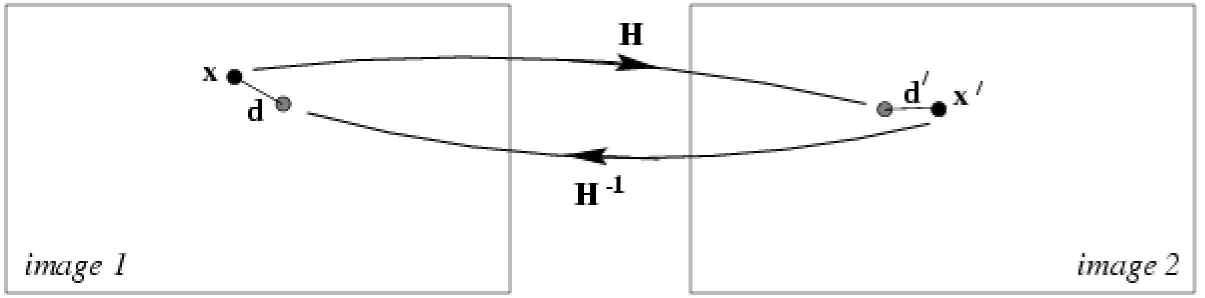
Reprojection error

$$\left(\hat{\mathrm{H}}, \hat{\mathrm{x}}_i, \hat{\mathrm{x}}'_i\right) = \underset{\mathrm{H}, \hat{\mathrm{x}}_i, \hat{\mathrm{x}}'_i}{\operatorname{argmin}} \sum_i d\left(\mathrm{x}_i, \hat{\mathrm{x}}_i\right)^2 + d\left(\mathrm{x}'_i, \hat{\mathrm{x}}'_i\right)^2$$

$$\text{subject to } \hat{\mathrm{x}}'_i = \hat{\mathrm{H}}\hat{\mathrm{x}}_i$$

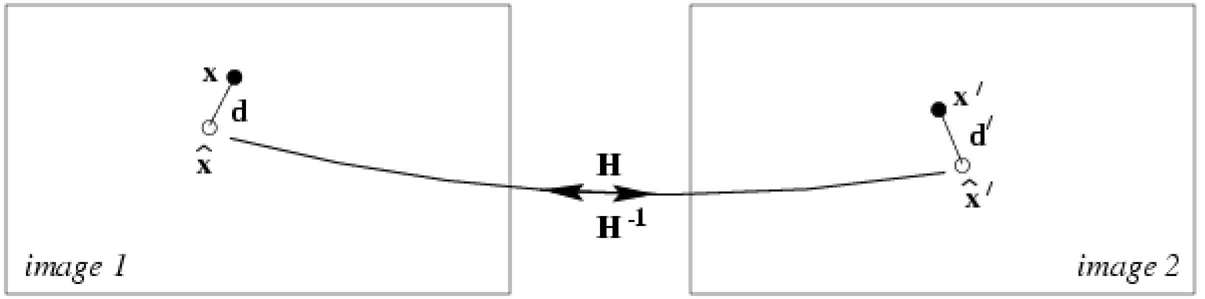
Symmetric Transfer Error v.s. Reprojection Error

Symmetric Transfer Error



$$d\left(\mathrm{x}, \mathrm{H}^{-1}\mathrm{x}'\right)^2 + d\left(\mathrm{x}', \mathrm{H}\mathrm{x}\right)^2$$

Reprojection Error



$$d\left(\mathrm{x}, \hat{\mathrm{x}}\right)^2 + d\left(\mathrm{x}', \hat{\mathrm{x}}'\right)^2$$

Comparison of Geometric and Algebraic Distances

Error in one image

$$\mathrm{x}'_i = \left(x'_i, y'_i, w'_i\right)^{\mathsf{T}} \quad \hat{\mathrm{x}}'_i = \left(\hat{x}'_i, \hat{y}'_i, \hat{w}'_i\right)^{\mathsf{T}} = \mathrm{H}\overline{\mathrm{x}}$$

$$\begin{bmatrix} 0^{\mathsf{T}} & -w'_i\mathrm{x}_i^{\mathsf{T}} & y'_i\mathrm{x}_i^{\mathsf{T}} \\ w'_i\mathrm{x}_i^{\mathsf{T}} & 0^{\mathsf{T}} & -x'_i\mathrm{x}_i^{\mathsf{T}} \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} \qquad \mathrm{A}_i\mathrm{h} = e_i = \begin{pmatrix} y'_i\hat{w}'_i - w'_i\hat{y}'_i \\ w'_i\hat{x}'_i - x'_i\hat{w}'_i \end{pmatrix}$$

$$d_{\mathrm{alg}}\left(\mathrm{x}'_{\mathrm{i}}, \hat{\mathrm{x}}'_{\mathrm{i}}\right)^2 = \left(y'_i\hat{w}'_i - w'_i\hat{y}'_i\right)^2 + \left(w'_i\hat{x}'_i - x'_i\hat{w}'_i\right)^2$$

$$d\left(\mathrm{x}'_{\mathrm{i}}, \hat{\mathrm{x}}'_{\mathrm{i}}\right)^2 = \left(\left(y'_i / w'_i - \hat{y}'_i / \hat{w}'_i\right)^2 + \left(\hat{x}'_i / \hat{w}'_i - x'_i / w'_i\right)^2\right)^{1/2}$$

$$= d_{\mathrm{alg}}\left(\mathrm{x}'_{\mathrm{i}}, \hat{\mathrm{x}}'_{\mathrm{i}}\right) / w'_i\hat{w}'_i$$ these two distance metrics are related, but not identical

$w'_i = 1$ typical, but not $\hat{w}'_i = \mathrm{h}_3\mathrm{x}_{\mathrm{i}}$ except for affinities

➔ For affinities, DLT can minimize geometric distance

Geometric Interpretation of Reprojection Error

Estimating homography ~ fit surface ν_{H} to points $\mathrm{X}=(x,y,x',y')^{\mathsf{T}}$ in $\mathbb{R}^4$

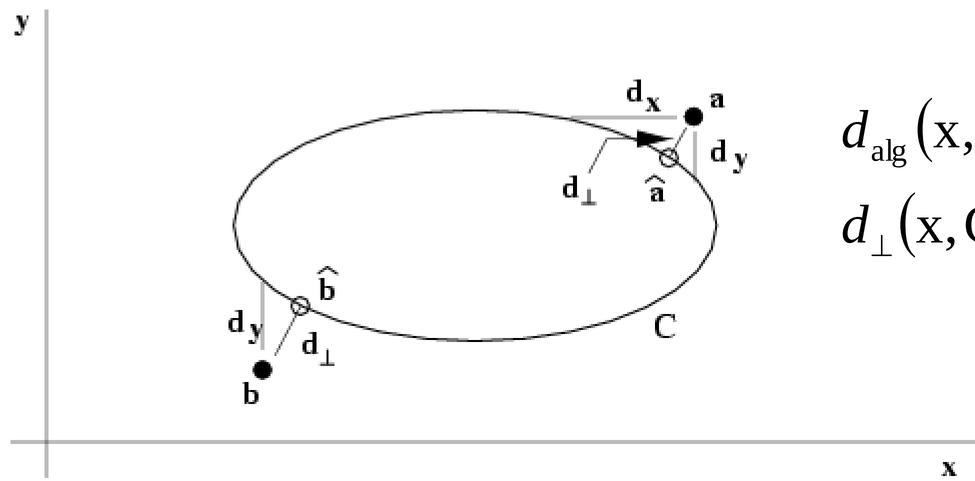
$\mathrm{x}_i' \times \mathrm{H}\mathrm{x}_i = 0$ represents 2 quadrics in $\mathbb{R}^4$ (quadratic in X)

$$\left\|\mathrm{X}_{\mathrm{i}} - \hat{\mathrm{X}}_{\mathrm{i}}\right\|^2 = \left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2 + \left(x_i' - \hat{x}_i'\right)^2 + \left(y_i' - \hat{y}_i'\right)^2$$

$$= d\left(\mathrm{x}_i, \hat{\mathrm{x}}_i\right)^2 + d\left(\mathrm{x}_i', \hat{\mathrm{x}}_i'\right)^2$$

$$d\left(\mathrm{x}_i, \hat{\mathrm{x}}_i\right)^2 + d\left(\mathrm{x}_i', \hat{\mathrm{x}}_i'\right)^2 = d_{\perp}\left(\mathrm{X}_i, \nu_{\mathrm{H}}\right)^2$$

Analog to conic fitting



$$d_{\mathrm{alg}}\left(\mathrm{x}, \mathrm{C}\right)^2 = \mathrm{x}^{\mathsf{T}}\mathrm{C}\mathrm{x}$$

$$d_{\perp}\left(\mathrm{x}, \mathrm{C}\right)^2$$

Sampson Error

between algebraic and geometric error

Vector $\hat{\mathrm{X}}$ that minimizes the geometric error $\left\|\mathrm{X}-\hat{\mathrm{X}}\right\|^2$ is the closest point on the variety ν_{H} to the measurement X

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

$$\mathrm{Ah} = C_{\mathrm{H}}(\mathrm{X}) = 0$$

$$C_{\mathrm{H}}(\mathrm{X}+\delta_{\mathrm{X}}) = C_{\mathrm{H}}(\mathrm{X}) + \frac{\partial C_{\mathrm{H}}}{\partial \mathrm{X}}\delta_{\mathrm{X}} \qquad \delta_{\mathrm{X}} = \hat{\mathrm{X}} - \mathrm{X} \qquad C_{\mathrm{H}}(\hat{\mathrm{X}}) = 0$$

$$C_{\mathrm{H}}(\mathrm{X}) + \frac{\partial C_{\mathrm{H}}}{\partial \mathrm{X}}\delta_{\mathrm{X}} = 0 \qquad \mathrm{J}\delta_{\mathrm{X}} = -e \quad \text{with } \mathrm{J} = \frac{\partial C_{\mathrm{H}}}{\partial \mathrm{X}}$$

Find the vector δ_{X} that minimizes $\left\|\delta_{\mathrm{X}}\right\|$ subject to $\mathrm{J}\delta_{\mathrm{X}} = -e$

Sampson Error

Find the vector δ_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

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

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

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

$$\Rightarrow \delta_X = -J^T\left(JJ^T\right)^{-1} e$$

$$\hat{X} = X + \delta_X \qquad \|\delta_X\|^2 = \delta_X^T\delta_X = e^T\left(JJ^T\right)^{-1} e$$

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 ν_H to the measurement X

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

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

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

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

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

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

Sampson Approximation

$$\|\delta_{\mathrm{X}}\|^2 = e^{\mathrm{T}}\left(\mathrm{J}\mathrm{J}^{\mathrm{T}}\right)^{-1} e$$

A few points

(i) For a 2D homography $\mathrm{X}=(\mathrm{x},\mathrm{y},\mathrm{x}',\mathrm{y}')$

(ii) $e = C_{\mathrm{H}}(\mathrm{X})$ is the algebraic error vector

(iii) $\mathrm{J} = \dfrac{\partial C_{\mathrm{H}}}{\partial \mathrm{X}}$ is a 2x4 matrix, e.g. $J_{11} = \partial\left(-w_i'\mathrm{x}_i{}^{\mathrm{T}}\mathrm{h}^2 + y_i'\mathrm{x}_i{}^{\mathrm{T}}\mathrm{h}^3\right)/\partial x = -w_i' h_{21} + y_i' h_{31}$

(iv) Similar to algebraic error $\|e\|^2 = e^{\mathrm{T}} e$
in fact, same as Mahalanobis distance $\|e\|^2_{\mathrm{J}\mathrm{J}^{\mathrm{T}}}$

(v) Sampson error independent of linear reparametrization (cancels out in between e and J)

(vi) Must be summed for all points $\sum e^{\mathrm{T}}\left(\mathrm{J}\mathrm{J}^{\mathrm{T}}\right)^{-1} e$

(vii) Close to geometric error, but much fewer unknowns

Statistical Cost Function and Maximum Likelihood Estimation

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

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

Error in one image

$$\Pr(\{\mathrm{x}'_\mathrm{i}\} \mid \mathrm{H}) = \prod_i \frac{1}{2\pi\sigma^2} e^{-d(\mathrm{x}'_\mathrm{i},\mathrm{H}\bar{\mathrm{x}}_i)^2/(2\sigma^2)}$$

$$\log \Pr(\{\mathrm{x}'_\mathrm{i}\} \mid \mathrm{H}) = -\frac{1}{2\sigma^2} \sum d(\mathrm{x}'_\mathrm{i}, \mathrm{H}\bar{\mathrm{x}}_i)^2 + \text{constant}$$

Maximum Likelihood Estimate

$$\sum d(\mathrm{x}'_\mathrm{i}, \mathrm{H}\bar{\mathrm{x}}_i)^2$$

Equivalent to minimizing the geometric error function

Statistical Cost Function and Maximum Likelihood Estimation

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

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

Error in both images

$$\Pr(\{\mathrm{x}'_\mathrm{i}\} \mid \mathrm{H}) = \prod_i \frac{1}{2\pi\sigma^2} e^{-\left(d(\mathrm{x}_\mathrm{i},\bar{\mathrm{x}}_i)^2 + d(\mathrm{x}'_\mathrm{i},\mathrm{H}\bar{\mathrm{x}}_i)^2\right)/(2\sigma^2)}$$

Maximum Likelihood Estimate

$$\sum d(\mathrm{x}_\mathrm{i},\hat{\mathrm{x}}_i)^2 + d(\mathrm{x}'_\mathrm{i},\hat{\mathrm{x}}'_i)^2$$

Equivalent to minimizing the reprojection error function

Mahalanobis Distance

- General Gaussian case

Measurement X with covariance matrix Σ

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

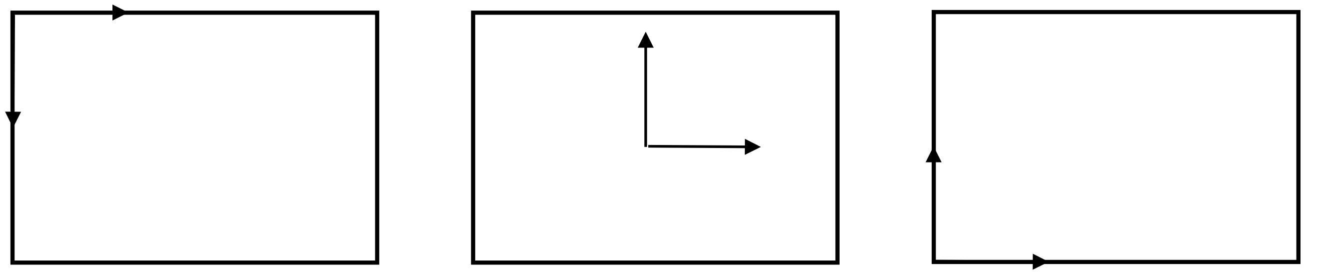
Error in two images (independent)

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

Varying covariances

$$\sum_{i}\left\|\mathrm{X}_{i}-\overline{\mathrm{X}}_{i}\right\|_{\Sigma_{i}}^{2}+\left\|\mathrm{X}_{i}'-\overline{\mathrm{X}}_{i}'\right\|_{\Sigma_{i}'}^{2}$$

Invariance to Transforms ?



$$x' = Hx$$

$$\tilde{x} = Tx$$

$$\tilde{x}' = T'x'$$

$$H \overset{?}{=} T'^{-1}\tilde{H}T$$

$$\tilde{x}' = \tilde{H}\tilde{x}$$

$$T'x' = \tilde{H}Tx$$

$$x' = T'^{-1}\tilde{H}Tx$$

will result change?

for which algorithms? for which transformations?

Non-invariance of DLT

Given $\mathrm{x_i} \leftrightarrow \mathrm{x'_i}$ and H computed by DLT,
and $\tilde{\mathrm{x}}_\mathrm{i} = \mathrm{T}\mathrm{x}_\mathrm{i}, \tilde{\mathrm{x}}'_\mathrm{i} = \mathrm{T}'\mathrm{x}'_\mathrm{i}$

Does the DLT algorithm applied to $\tilde{\mathrm{x}}_\mathrm{i} \leftrightarrow \tilde{\mathrm{x}}'_\mathrm{i}$
yield $\tilde{\mathrm{H}} = \mathrm{T}'\mathrm{H}\mathrm{T}^{-1}$*?*

Non-invariance of DLT

Effect of change of coordinates on algebraic error

$$\tilde{e}_i = \tilde{\mathrm{x}}'_i \times \tilde{\mathrm{H}}\tilde{\mathrm{x}}_i = \mathrm{T}'\mathrm{x}'_i \times \left(\mathrm{T}'\mathrm{H}\mathrm{T}^{-1}\right)\mathrm{T}\mathrm{x}_i = \mathrm{T}'^{*}\left(\mathrm{x}'_i \times \mathrm{H}\mathrm{x}_i\right) = \mathrm{T}'^{*} e_i$$

for similarities

$$\mathrm{T}' = \begin{bmatrix} s\mathrm{R} & \mathrm{t} \\ 0 & 1 \end{bmatrix} \qquad \mathrm{T}'^{*} = s\begin{bmatrix} \mathrm{R} & 0 \\ -\mathrm{t}^{\mathrm{T}}\mathrm{R} & s \end{bmatrix}$$

(T*: cofactor matrix)

so $\left\|\tilde{\mathrm{A}}_{\mathrm{i}}\tilde{\mathrm{h}}\right\| = \left\|\left(\tilde{e}_1, \tilde{e}_2\right)^{\mathsf{T}}\right\| = \left\|s\mathrm{R}\left(e_1, e_2\right)^{\mathsf{T}}\right\| = s\left\|\mathrm{A}_{\mathrm{i}}\mathrm{h}\right\|$

$$d_{\mathrm{alg}}\left(\mathrm{x}'_i, \mathrm{H}\mathrm{x}_i\right) = s\,d_{\mathrm{alg}}\left(\tilde{\mathrm{x}}'_i, \tilde{\mathrm{H}}\tilde{\mathrm{x}}_i\right)$$

Non-invariance of DLT

Given $\mathrm{x}_\mathrm{i} \leftrightarrow \mathrm{x}'_\mathrm{i}$ and H computed by DLT,
and $\tilde{\mathrm{x}}_\mathrm{i} = \mathrm{T}\mathrm{x}_\mathrm{i}, \tilde{\mathrm{x}}'_\mathrm{i} = \mathrm{T}'\mathrm{x}'_\mathrm{i}$

Does the DLT algorithm applied to $\tilde{\mathrm{x}}_\mathrm{i} \leftrightarrow \tilde{\mathrm{x}}'_\mathrm{i}$ yield $\tilde{\mathrm{H}} = \mathrm{T}'\mathrm{H}\mathrm{T}^{-1}$?

$$\text{minimize} \sum_i d_{\mathrm{alg}}\left(\mathrm{x}'_i, \mathrm{H}\mathrm{x}_i\right)^2 \text{subject to } \|\mathrm{H}\| = 1$$

$$\Leftrightarrow \text{minimize} \sum_i d_{\mathrm{alg}}\left(\tilde{\mathrm{x}}'_i, \tilde{\mathrm{H}}\tilde{\mathrm{x}}_i\right)^2 \text{subject to } \|\mathrm{H}\| = 1$$

$$\not\Leftrightarrow \text{minimize} \sum_i d_{\mathrm{alg}}\left(\tilde{\mathrm{x}}'_i, \tilde{\mathrm{H}}\tilde{\mathrm{x}}_i\right)^2 \text{subject to } \|\tilde{\mathrm{H}}\| = 1$$

Invariance of Geometric Error

Given $\mathrm{x_i} \leftrightarrow \mathrm{x'_i}$ and H,

and $\tilde{\mathrm{x}}_\mathrm{i} \leftrightarrow \tilde{\mathrm{x}}'_\mathrm{i}$, $\tilde{\mathrm{x}}_\mathrm{i} = \mathrm{T}\mathrm{x}_\mathrm{i}, \tilde{\mathrm{x}}'_\mathrm{i} = \mathrm{T'}\mathrm{x}'_\mathrm{i}$, $\widetilde{\mathrm{H}} = \mathrm{T'}\mathrm{H}\mathrm{T}^{-1}$

Assume $\mathrm{T'}$ is a similarity transformations

$$d\left(\tilde{\mathrm{x}}'_i, \tilde{\mathrm{H}}\tilde{\mathrm{x}}_i\right) = d\left(\mathrm{T'x'}_i, \mathrm{T'HT^{-1}Tx}_i\right) = d\left(\mathrm{T'x'}_i, \mathrm{T'Hx}_i\right)$$
$$= sd\left(\mathrm{x}'_i, \mathrm{Hx}_i\right)$$

Normalizing Transformations

- Since DLT is not invariant,

  what is a good choice of coordinates?

  e.g. Isotropic scaling

  - Translate centroid to origin
  - Scale to a $\sqrt{2}$ average distance to the origin
  - Independently on both images

Or

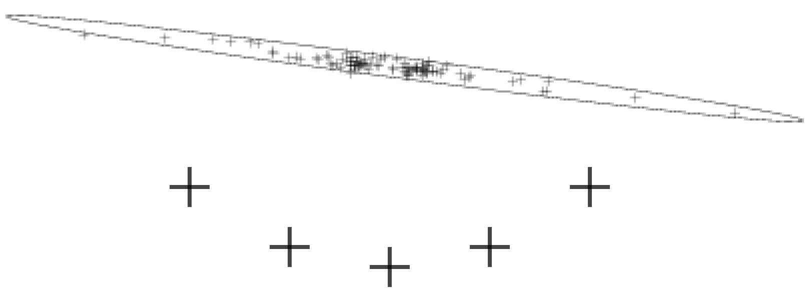
$$\mathrm{T}_{\mathrm{norm}} = \begin{bmatrix} w+h & 0 & w/2 \\ 0 & w+h & h/2 \\ 0 & 0 & 1 \end{bmatrix}^{-1}$$

Importance of Normalization

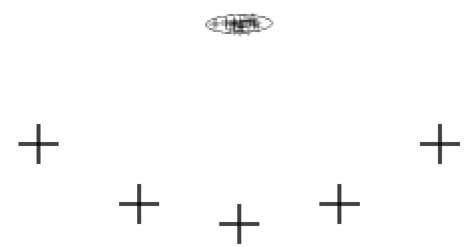
$$\begin{bmatrix} 0 & 0 & 0 & -x'_i & -y'_i & -1 & y'_i x_i & y'_i y_i & y'_i \\ x_i & y_i & 1 & 0 & 0 & 0 & -x'_i x_i & -x'_i y_i & -x'_i \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

$$\begin{matrix} \sim 10^2 & \sim 10^2 & 1 & \sim 10^2 & \sim 10^2 & 1 & \sim 10^4 & \sim 10^4 & \sim 10^2 \end{matrix}$$

orders of magnitude difference!



Without normalization



With normalization

Normalized DLT Algorithm

Objective

Given n≥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) Normalize points $\tilde{x}_i = T_{norm} x_i, \tilde{x}_i' = T_{norm}' x_i'$

(ii) Apply DLT algorithm to $\tilde{x}_i \leftrightarrow \tilde{x}_i'$,

(iii) Denormalize solution $H = T_{norm}'^{-1} \tilde{H} T_{norm}$

Employ this algorithm instead of the original DLT algorithm!

- More accurate
- Invariant to arbitrary choices of the scale and coordinate origin

Normalization is also called **pre-conditioning**

Iterative Minimization Methods

Required to minimize geometric error

(i) Often slower than DLT

(ii) Require initialization

(iii) No guaranteed convergence, local minima

(iv) Stopping criterion required

Therefore, careful implementation required:

(i) Cost function

(ii) Parameterization (minimal or not)

(iii) Cost function (parameters)

(iv) Initialization

(v) Iterations

Parameterization

Parameters should cover complete space and allow efficient estimation of cost

- Minimal or over-parameterized? e.g. 8 or 9

  (minimal often more complex, also cost surface)

  (good algorithms can deal with over-parameterization)

  (sometimes also local parameterization)

- Parametrization can also be used to restrict transformation to particular class, e.g. affine

Function Specifications

(i) Measurement vector $X \in \mathbb{R}^{N}$ with covariance Σ

(ii) Set of parameters represented by vector $P \in \mathbb{R}^{N}$

(iii) Mapping $f: \mathbb{R}^{M} \rightarrow \mathbb{R}^{N}$. Range of mapping is surface S representing allowable measurements

(iv) Cost function: squared Mahalanobis distance

$$\left\| \mathrm{X} - f\left(\mathrm{P}\right) \right\|_{\Sigma}^{2} = \left(\mathrm{X} - f\left(\mathrm{P}\right)\right)^{\mathrm{T}} \Sigma^{-1} \left(\mathrm{X} - f\left(\mathrm{P}\right)\right)$$

Goal is to achieve $f\left(\mathrm{P}\right) = \mathrm{X}$, or get as close as possible in terms of Mahalanobis distance

Error in one image

$$\sum d\left(\mathrm{x}_{\mathrm{i}}', \mathrm{H}\bar{\mathrm{x}}_i\right)^2$$

$$f : \mathrm{h} \rightarrow \left(\mathrm{Hx}_1, \mathrm{Hx}_2, \ldots, \mathrm{Hx}_n\right)$$

$$\left\|\mathrm{X} - f\left(\mathrm{h}\right)\right\|$$

X composed of 2n inhomogeneous coordinates of the points x_i'

Symmetric transfer error

$$\sum_i d\left(\mathrm{x}_i, \mathrm{H}^{-1}\mathrm{x}_i'\right)^2 + d\left(\mathrm{x}_i', \mathrm{Hx}_i\right)^2$$

$$f : \mathrm{h} \rightarrow \left(\mathrm{H}^{-1}\mathrm{x}_1', \mathrm{H}^{-1}\mathrm{x}_2', \ldots, \mathrm{H}^{-1}\mathrm{x}_n', \mathrm{Hx}_1, \mathrm{Hx}_2, \ldots, \mathrm{Hx}_n\right)$$

$$\left\|\mathrm{X} - f\left(\mathrm{h}\right)\right\|$$

X composed of 4n-vector inhomogeneous coordinates of the pointsx_i and x_i'

Reprojection error

$$\sum d\left(\mathrm{x}_{\mathrm{i}}, \hat{\mathrm{x}}_i\right)^2 + d\left(\mathrm{x}_{\mathrm{i}}', \hat{\mathrm{x}}_i'\right)^2$$

$$f : (\mathbf{h}, \hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_n) \mapsto (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_1', \ldots, \hat{\mathbf{x}}_n, \hat{\mathbf{x}}_n')$$

$$\left\|\mathrm{X} - f\left(\mathrm{h}\right)\right\|$$

X composed of 4n-vector

Initialization

- Typically, use linear solution
- If outliers, use robust algorithm

- Alternative, sample parameter space

Iteration Methods

Many algorithms exist

- Newton’s method
- Levenberg-Marquardt

- Powell’s method
- Simplex method

Levenberg-Marquardt Algorithm

For a mapping function *f* with parameter vector $\mathbf{p} \in \mathcal{R}^m$

To an estimated measurement vector $\hat{\mathbf{x}} = f(\mathbf{p}),\ \hat{\mathbf{x}} \in \mathcal{R}^n$

We want to find p that can minimize $\epsilon^T \epsilon$, where $\epsilon = \mathbf{x} - \hat{\mathbf{x}}$

f(**p**) can be approximated as $f(\mathbf{p} + \delta_{\mathbf{p}}) \approx f(\mathbf{p}) + \mathbf{J}\delta_{\mathbf{p}}$

with small $||\delta_{\mathbf{p}}||$ and **J**= $\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}$

➔Find $\delta_{\mathbf{p}}$ to minimize

$$||\mathbf{x} - f(\mathbf{p} + \delta_{\mathbf{p}})|| \approx ||\mathbf{x} - f(\mathbf{p}) - \mathbf{J}\delta_{\mathbf{p}}|| = ||\epsilon - \mathbf{J}\delta_{\mathbf{p}}||$$

Levenberg-Marquardt Algorithm

Find $\delta_{\mathbf{p}}$ to minimize

$$||\mathbf{x} - f(\mathbf{p} + \delta_{\mathbf{p}})|| \approx ||\mathbf{x} - f(\mathbf{p}) \; - \; \mathbf{J}\delta_{\mathbf{p}}|| = ||\epsilon - \mathbf{J}\delta_{\mathbf{p}}||$$

The least-square solution: $\underbrace{\mathbf{J}^T\mathbf{J}}_{\text{Hessian}}\delta_{\mathbf{p}} = \mathbf{J}^T\epsilon$

Augmented normal equation (with damping term μ):

$$\mathbf{N}\delta_{\mathbf{p}} = \mathbf{J}^T\epsilon \qquad \mathbf{N}_{ii} \; = \; \mu + \left[\mathbf{J}^T\mathbf{J}\right]_{ii}$$

Gold Standard Algorithm

<u>Objective</u>

Given n≥4 2D to 2D point correspondences $\{x_i \leftrightarrow x_i'\}$, determine the Maximum Likelyhood Estimation of H

(this also implies computing optimal $x_i' = Hx_i$)

<u>Algorithm</u>

(i) **Initialization:** compute an initial estimate using normalized DLT or RANSAC

(ii) **Geometric minimization of -Either Sampson error:**

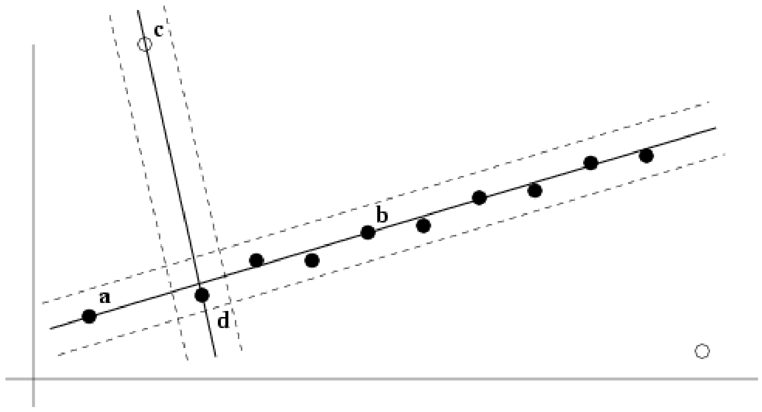
- Minimize the Sampson error
- Minimize using Levenberg-Marquardt over 9 entries of h

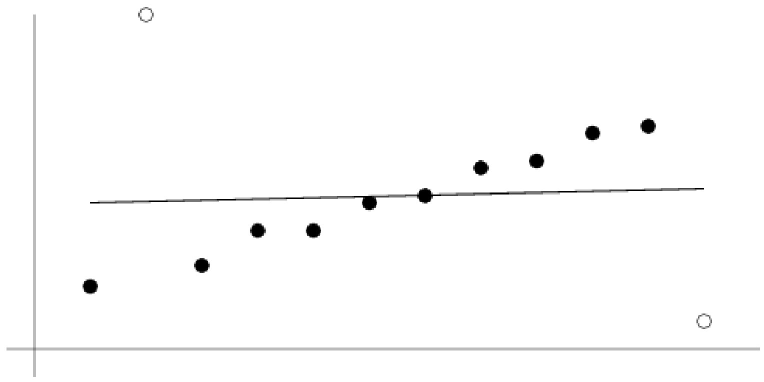
or Gold Standard error:

- compute initial estimate for optimal $\{x_i\}$
- minimize cost $\sum d(\mathrm{x}_i, \hat{\mathrm{x}}_i)^2 + d(\mathrm{x}_i', \hat{\mathrm{x}}_i')^2$ over $\{H, x_1, x_2, \ldots, x_n\}$
- if many points, use sparse method

Robust Estimation

- What if set of matches contains gross outliers?





RANSAC: RANdom SAmple Consensus

Objective

Robust fit of model to data set S which contains outliers

Algorithm

(i) Randomly select a sample of *s* data points from S and instantiate the model from this subset.

(ii) Determine the set of data points S_i which are within a distance threshold *t* of the model. The set S_i is the consensus set of samples and defines the inliers of S.

(iii) If the subset of S_i is greater than some threshold *T*, re-estimate the model using all the points in S_i and terminate

(iv) If the size of S_i is less than *T*, select a new subset and repeat the above.

(v) After *N* trials the largest consensus set S_i is selected, and the model is re-estimated using all the points in the subset S_i

Distance Threshold

Choose t so probability for inlier is α (e.g. 0.95)

- Often empirically
- Zero-mean Gaussian noise σ then $d_\perp^2$ follows χ_m^2 distribution with m=codimension of model

(dimension+codimension=dimension space)

Codimension	Model	t^2
1	Line (l), Fundamental matrix (F)	$3.84\sigma^2$
2	Homography (H), Camera Matrix (P)	$5.99\sigma^2$
3	Trifocal tensor (T)	$7.81\sigma^2$

How Many Samples?

Choose *N* so that, with probability *p*, at least one random sample is free from outliers. e.g. *p*=0.99

$$\left(1-\left(1-e\right)^{s}\right)^{N}=1-p$$

$$N=\log\left(1-p\right)/\log\left(1-\left(1-e\right)^{s}\right)$$

	proportion of outliers e						
s	5%	10%	20%	25%	30%	40%	50%
2	2	3	5	6	7	11	17
3	3	4	7	9	11	19	35
4	3	5	9	13	17	34	72
5	4	6	12	17	26	57	146
6	4	7	16	24	37	97	293
7	4	8	20	33	54	163	588
8	5	9	26	44	78	272	1177

Acceptable Consensus Set?

- Typically, terminate when inlier ratio reaches expected ratio of inliers

$$T = (1 - e)n$$

Adaptively Determining the Number of Samples

e is often unknown a priori, so pick worst case, e.g. 50%, and adapt if more inliers are found, e.g. 80% would yield e=0.2

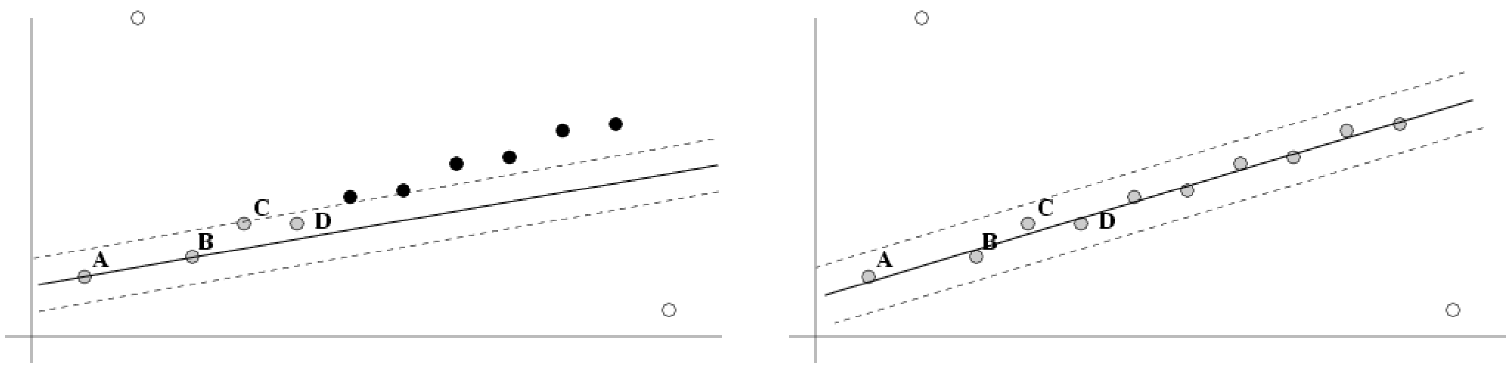
- $N=\infty$, *sample_count* =0
- While N >*sample_count* repeat
  - Choose a sample and count the number of inliers
  - Set e=1-(number of inliers)/(total number of points)
  - Recompute N from e
  - Increment the *sample_count* by 1

$$\left(N = \log(1-p)/\log\left(1-(1-e)^s\right)\right)$$

- Terminate

Robust Maximum Likelyhood Estimation

Previous MLE algorithm considers fixed set of inliers



Better, robust cost function (reclassifies)

$$\mathcal{R} = \sum_{\mathrm{i}} \rho\left(d_{\perp i}\right) \text{with}\, \rho(e) = \begin{cases} e^2 & e^2 < t^2 \text{ inlier} \\ t^2 & e^2 > t^2 \text{ outlier} \end{cases}$$

Other Robust Algorithms

- RANSAC maximizes number of inliers
- LMedS minimizes median error

Automatic Computation of H

<u>Objective</u>

Compute homography between two images

<u>Algorithm</u>

(i) **Interest points:** Compute interest points in each image

(ii) **Putative correspondences:** Compute a set of interest point matches based on some similarity measure

(iii) **RANSAC robust estimation:** Repeat for *N* samples

(a) Select 4 correspondences and compute H

(b) Calculate the distance $d_\perp$ for each putative match

(c) Compute the number of inliers consistent with H ($d_\perp < t$)

Choose H with most inliers

(iv) **Optimal estimation:** re-estimate H from all inliers by minimizing ML cost function with Levenberg-Marquardt

(v) Guided matching: Determine more matches using prediction by computed H

Optionally iterate last two steps until convergence

Determine Putative Correspondences

- Compare interest points

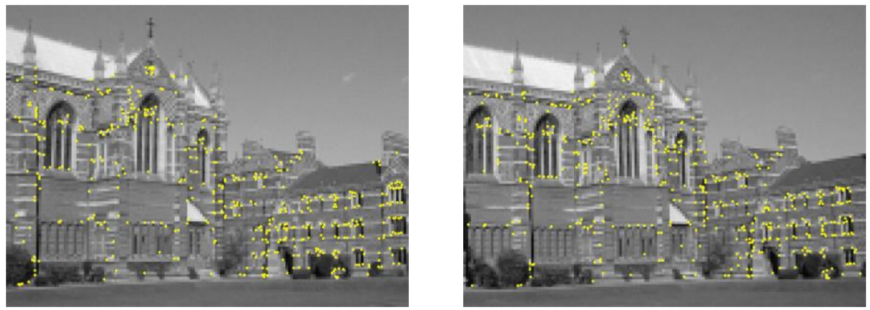
  Similarity measure:

  - SAD, SSD, ZNCC on small neighborhood

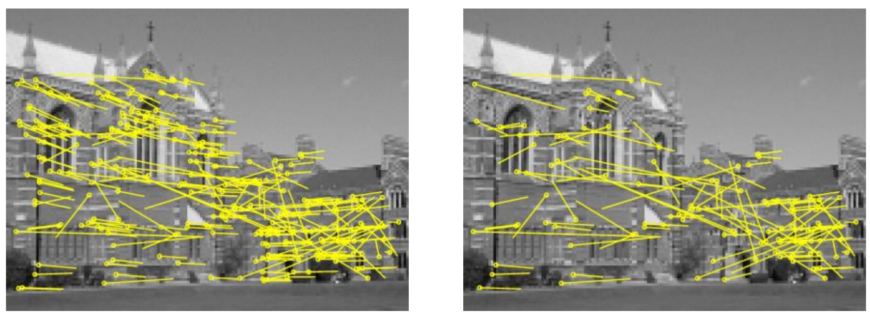
- If motion is limited, only consider interest points with similar coordinates

- More advanced approaches exist, based on invariance…
  - Such as SIFT

Example: robust computation

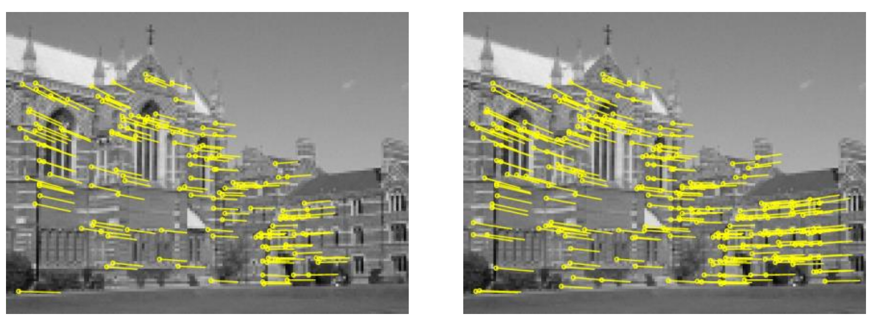


Interest points (500/image)



Putative correspondences (268)

Outliers (117)



Inliers (151)

Final inliers (262)