Estimation- 2D Projective Transformations

Prof. Kyoung Mu Lee
Dept. of ECE, Seoul National University

SVD (Singular Value Decomposition)

- Any real *mxn* matrix ***A*** can be decomposed uniquely by

$$\boxed{\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m}\mathbf{D}_{m\times n}\mathbf{V}^{\mathsf{T}}_{n\times n}} \qquad m \geq n$$

where

$$\mathbf{U} = \left[\boxed{\mathbf{u}_1, \ldots \mathbf{u}_n} \cdots \mathbf{u}_m\right]$$

$$\mathbf{V} = \left[\mathbf{v}_1, \ldots \mathbf{v}_n\right]$$

orthonormal column vectors

$$\mathbf{U}^{\mathsf{T}}\mathbf{U} = \mathbf{I}_{m\times m}$$

$$\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}_{n\times n}$$

$$\mathbf{D} = \begin{bmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

$$d_1 \geq d_2 \geq \cdots \geq d_n \geq 0$$

SVD (Singular Value Decomposition)

- If we consider only the non-zero terms, we can write it by

$$\boxed{\mathbf{A}_{m\times n} = \mathbf{U}_{m\times n}\mathbf{D}_{n\times n}\mathbf{V}^{\mathsf{T}}_{n\times n}} \qquad m \geq n$$

- Properties:

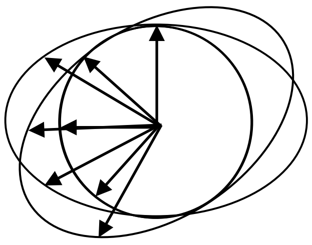
- $$\mathbf{AA}^T = \mathbf{UDV}^T\mathbf{VDU}^T = \mathbf{UD}^2\mathbf{U}^T$$

$$\Rightarrow \begin{cases} \mathbf{AA}^T\mathbf{u}_i = d_i^{\,2}\mathbf{u}_i \\ \mathbf{A}^T\mathbf{Av}_i = d_i^{\,2}\mathbf{v}_i \end{cases}$$

- $$\mathbf{A} = \mathbf{UDV}^T = \mathbf{u}_1 d_1 \mathbf{v}_1^{\mathsf{T}} + \mathbf{u}_2 d_2 \mathbf{v}_2^{\mathsf{T}} + \cdots + \mathbf{u}_n d_n \mathbf{v}_n^{\mathsf{T}}$$

$$= d_1\mathbf{u}_1\mathbf{v}_1^{\mathsf{T}} + d_2\mathbf{u}_2\mathbf{v}_2^{\mathsf{T}} + \cdots + d_n\mathbf{u}_n\mathbf{v}_n^{\mathsf{T}} = \sum_{i=1}^{n} d_i\mathbf{u}_i\mathbf{v}_i^{\mathsf{T}}$$

SVD (Singular Value Decomposition)



- $$\mathbf{Ax} = \mathbf{UDV}^{\mathsf{T}}\mathbf{x}$$

  $$\left\|\mathbf{Ax}\right\|^2 = \mathbf{x}^T\mathbf{A}^T\mathbf{Ax} = \mathbf{x}^T\mathbf{VD}^2\mathbf{V}^T\mathbf{x} = \left\|\mathbf{DV}^T\mathbf{x}\right\|^2$$

  $$\Rightarrow \left\|\mathbf{Ax}\right\| \text{ is only affected by } \mathbf{V} \text{ and } \mathbf{D}$$

- For square, non-singular matrices,

  $$\mathbf{A}^{-1} = \mathbf{VD}^{-1}\mathbf{U}^T \qquad \mathbf{D}^{-1} = diag\left(1/d_1, \cdots, 1/d_n\right)$$

- Closest rank r approximation

  $$\tilde{\mathbf{A}} = \mathbf{U}\tilde{\mathbf{D}}\mathbf{V}^{\mathsf{T}} \qquad \tilde{\mathbf{D}} = diag\left(d_1, d_2, \cdots, d_r, 0, \cdots, 0\right)$$

  $$= d_1\mathbf{u}_1\mathbf{v}_1^{\mathsf{T}} + d_2\mathbf{u}_2\mathbf{v}_2^{\mathsf{T}} + \cdots + d_r\mathbf{u}_r\mathbf{v}_r^{\mathsf{T}}$$

SVD (Singular Value Decomposition)

- Solution of linear equations: $\mathbf{A}_{m\times n}\mathbf{x} = \mathbf{b}$
- Least squares solution: $\boxed{\text{Find } \mathbf{x} \text{ that minimizes } \|\mathbf{A}\mathbf{x}-\mathbf{b}\|}$

i) if rank $\mathbf{A}_{mxn} = n$ *(full column rank)*

$$\text{since } \|\mathbf{A}\mathbf{x}-\mathbf{b}\| = \|\mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{x}-\mathbf{b}\| = \|\mathbf{D}\mathbf{V}^T\mathbf{x}-\mathbf{U}^T\mathbf{b}\|$$

$$\Rightarrow \min\|\mathbf{D}\mathbf{y}-\mathbf{b}'\|, \quad \mathbf{y} = \mathbf{V}^T\mathbf{x}, \mathbf{b}' = \mathbf{U}^T\mathbf{b}$$

$$\Rightarrow \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \\ \hline & & 0 & \\ & & & \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} b_1' \\ b_2' \\ \vdots \\ b_n' \\ \hline b_{n+1}' \\ \vdots \\ b_m' \end{bmatrix}$$

$$\overset{\min}{\Rightarrow} y_i = \frac{b_i'}{d_i}, i = 1, \cdots n$$

And $\mathbf{x} = \mathbf{V}\mathbf{y}$

(unique solution)

SVD (Singular Value Decomposition)

ii) if rank $\mathbf{A}_{mxn} = r < n$ *(not full column rank)*

$$\mathbf{x} = \mathbf{V}\mathbf{y} + \underbrace{\lambda_{r+1}\mathbf{v}_{r+1} + \ldots + \lambda_n \mathbf{v}_n}_{\text{homogeneous solution}}$$

- Pseudo inverse

$$\mathbf{A}_{m\times n}\mathbf{x} = \mathbf{b} \rightarrow \mathbf{x} = \mathbf{A}^{+}\mathbf{b}$$

$$\mathbf{A}^{+} = \mathbf{V}\mathbf{D}^{+}\mathbf{U}^{\mathsf{T}} \quad \mathbf{D}^{+} = diag\left(d_1^{-1}, d_2^{-1}, \cdots, d_r^{-1},\ 0\ \ , \cdots,\ 0\ \right)$$

SVD (Singular Value Decomposition)

- Solution of homogeneous equations $\mathbf{A}_{m\times n}\mathbf{x} = 0$
- Least squares solution:

$$\boxed{\text{Find } \mathbf{x} \text{ that minimizes } \|\mathbf{A}\mathbf{x}\| \text{ subject to } \|\mathbf{x}\| = 1}$$

$$\text{since } \|\mathbf{A}\mathbf{x}\| = \|\mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{x}\| = \|\mathbf{D}\mathbf{V}^T\mathbf{x}\|, \text{ and } \|\mathbf{x}\| = \|\mathbf{V}^T\mathbf{x}\|$$

$$\Rightarrow \min\|\mathbf{D}\mathbf{y}\| \text{ subject to } \|\mathbf{y}\| = 1,\ \mathbf{y} = \mathbf{V}^T\mathbf{x}$$

i) if rank $\mathbf{A}_{mxn} = n$ (full col. rank)

$$\mathbf{y} = (0,0,\ldots,0,1)^{\mathrm{T}}$$

$$\Rightarrow \mathbf{x} = \mathbf{V}\mathbf{y} = \mathbf{v}_n \quad \textit{(the last e-vector)}$$

ii) if rank $\boldsymbol{A}_{mxn} = r < n$ ($\mathbf{x}$ = null-vector of $\mathbf{A}$)

$$\mathbf{y} = (0,0,\ldots,\lambda_{r+1},\ldots,\lambda_n)^T$$

$$\Rightarrow \mathbf{x} = \mathbf{V}\mathbf{y} = \lambda_{r+1}\mathbf{v}_{r+1} + \ldots + \lambda_n\mathbf{v}_n$$

- Another interpretation

$$\mathbf{Ax} = \mathbf{UDV}^T\mathbf{x} = d_1\mathbf{u}_1\mathbf{v}_1^{\mathsf{T}}\mathbf{x} + d_2\mathbf{u}_2\mathbf{v}_2^{\mathsf{T}}\mathbf{x} + \cdots + d_n\mathbf{u}_n\mathbf{v}_n^{\mathsf{T}}\mathbf{x}$$

✓ Which $\mathbf{x}$ makes $\|\mathbf{Ax}\|$ to be minimum subject to $\|\mathbf{x}\| = 1$?

i) if $\quad d_i \neq 0, i = 1, \ldots, n$

$$\Rightarrow \mathbf{x} = \mathbf{v}_n, \text{ (the last e-vector)}$$

$$\Rightarrow \text{the minimum is } d_n$$

ii) if $\quad d_i = 0, i = r+1, \ldots, n$

$$\Rightarrow \mathbf{x} = \mathbf{Vy} = \lambda_{r+1}\mathbf{v}_{r+1} + \ldots + \lambda_n\mathbf{v}_n,$$

(the span of the singular e-vectors, or null space of $\mathbf{A}$)

$$\Rightarrow \mathbf{Ax} = 0$$

2D Homography Estimation

- To estimate the parameters of a 2D homography, we need

  # of independent equations $\geq$ degrees of freedom

- Example:

  To estimate **H**: $\mathbf{x}' = \mathbf{H}\mathbf{x}$

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

✓ **H** has 8 DOF

✓ 2 independent equations / a point correspondence ($\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$)

✓ 4x2≥8

✓ So, at least 4 point correspondences are required

Approximate solutions

- Minimal solution
  - ✓ 4 points $\Rightarrow$ an exact solution for **H**
- More points
  - ✓ No exact solution, because measurements are inexact (“noise”)
  - ✓ Find the *optimal solution* 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

The Direct Linear Transformation (DLT) algorithm

- Given a set of (noisy) point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$, how to compute $\mathbf{H}$ robustly for $\lambda \mathbf{x}'_i = \mathbf{H}\mathbf{x}_i$?
- Note that $\mathbf{H}$ must satisfy $\boxed{\mathbf{x}'_i \times \mathbf{H}\mathbf{x}_i = 0}$
- Let $\mathbf{H}\mathbf{x}_i = \begin{pmatrix} \mathbf{h}^{1^\mathrm{T}}\mathbf{x}_i \\ \mathbf{h}^{2^\mathrm{T}}\mathbf{x}_i \\ \mathbf{h}^{3^\mathrm{T}}\mathbf{x}_i \end{pmatrix}$ and $\mathbf{x}'_i = (x'_i, y'_i, 1)$, $\mathbf{x}_i = (x_i, y_i, 1)$
- Then

$$\mathbf{x}'_i \times \mathbf{H}\mathbf{x}_i = \begin{bmatrix} y'_i \mathbf{h}^{3T}\mathbf{x}_i - \mathbf{h}^{2T}\mathbf{x}_i \\ \mathbf{h}^{1T}\mathbf{x}_i - x'_i \mathbf{h}^{3T}\mathbf{x}_i \\ x'_i \mathbf{h}^{2T}\mathbf{x}_i - y'_i \mathbf{h}^{1T}\mathbf{x}_i \end{bmatrix} = \mathbf{0}$$

$$\Rightarrow \begin{bmatrix} \mathbf{0}^T & -\mathbf{x}_i^T & y'_i \mathbf{x}_i^T \\ \mathbf{x}_i^T & \mathbf{0}^T & -x'_i \mathbf{x}_i^T \\ -y'_i \mathbf{x}_i^T & x'_i \mathbf{x}_i^T & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{bmatrix} = \mathbf{0} \quad \Rightarrow \quad \boxed{\mathbf{A}_i \mathbf{h} = 0}$$

The DLT algorithm (cont.)

- Since 3rd row is redundant, we have

$$\Rightarrow \begin{bmatrix} \mathbf{0}^T & -\mathbf{x}_i^T & y_i'\mathbf{x}_i^T \\ \mathbf{x}_i^T & \mathbf{0}^T & -x_i'\mathbf{x}_i^T \end{bmatrix} \begin{bmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{bmatrix} = \mathbf{0}$$

$$\Rightarrow \quad \mathbf{A}_i\mathbf{h} = \mathbf{0} \qquad \boxed{\mathbf{A}_i\mathbf{h} = \mathbf{e}_i \quad \text{for noisy measurement}}$$

$\Rightarrow$ Two lin. independent Eqs. for each point correspondence.

- Thus, 4 point correspondences are enough.
- For more than 4 point correspondences

$\Rightarrow$ (Over-determined System)

The DLT algorithm (cont.)

- 4 point case:
  - ✓ Solve the exact solution **H** satisfying

$$\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \mathbf{A}_3 \\ \mathbf{A}_4 \end{bmatrix} \mathbf{h} = \mathbf{0} \quad \Rightarrow \quad \mathbf{A}\mathbf{h} = \mathbf{0}$$

  - ✓ The size of **A** is *8*x*9* or *12*x*9* (rank is *8)*
  - ✓ 1-D null-space **h** is the nontrivial solutions
  - ✓ Choose the one with $\|\mathbf{h}\| = 1$

The DLT algorithm (cont.)

- More points case:
  - ✓ Over-determined solution satisfying

$$\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_n \end{bmatrix} \mathbf{h} = \mathbf{0} \quad \Rightarrow \quad \mathbf{Ah} = \mathbf{0}$$

  - ✓ No exact solution due to the "noise"
  - ✓ So, find the approximate solution

$$\mathbf{h}^* = \min_{\mathbf{h}} \|\mathbf{Ah}\|$$

  - ✓ Additional constraint needed for nontrivial solution e.g. $\|\mathbf{h}\| = 1$

The DLT algorithm (cont.)

Objective

Given $n \geq 4$ 2D to 2D point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}_i'\}$, determine the 2D homography matrix $\mathbf{H}$ such that $\mathbf{x}_i' = \mathbf{H}\mathbf{x}_i$

Algorithm

(i) For each correspondence $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ compute $\mathbf{A}_i$. Usually only two first rows needed.

(ii) Assemble n 2x9 matrices $\mathbf{A}_i$ into a single $2n$x9 matrix $\mathbf{A}$

(iii) Obtain SVD of $\mathbf{A}$. Solution for $\mathbf{h}$ is last column of $\mathbf{V}$

(iv) Determine $\mathbf{H}$ from $\mathbf{h}$

Cost functions

- Which cost function can we employ?

- Algebraic distance
- Geometric distance
  - ✓ Transfer error, Symmetric transfer error, Reprojection error
- Sampson error

Algebraic distance error

- Residual (algebraic error) vector: $\mathbf{e}_i = \mathbf{A}_i \mathbf{h}$
  - ✓ Error associated with each correspondence $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$

$$d_{\text{alg}}(\mathbf{x}'_i, \mathbf{H}\mathbf{x}_i)^2 = \|\mathbf{e}_i\|^2 = \left\| \left[\begin{bmatrix} \mathbf{0}^T & -\mathbf{x}_i^T & y'_i \mathbf{x}_i^T \\ \mathbf{x}_i^T & \mathbf{0}^T & -x'_i \mathbf{x}_i^T \end{bmatrix} \right] \mathbf{h} \right\|^2$$

algebraic distance

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

- The total algebraic distance error:

$$\boxed{\sum_i d_{\text{alg}}(\mathbf{x}'_i, \mathbf{H}\mathbf{x}_i)^2 = \sum_i \|\mathbf{e}_i\|^2 = \|\mathbf{A}\mathbf{h}\|^2 = \|\mathbf{e}\|^2}$$

- Has no geometrical and statistical meaning
- However, with good normalization it works satisfactory, so it can be used for initialization

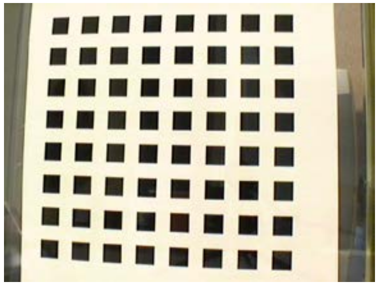
Geometric distance error (Symmetric transfer error)

- Transfer error (error in one (second) image)

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

$\mathbf{x}$: measured value
$\hat{\mathbf{x}}$: estimated value
$\bar{\mathbf{x}}$: true value

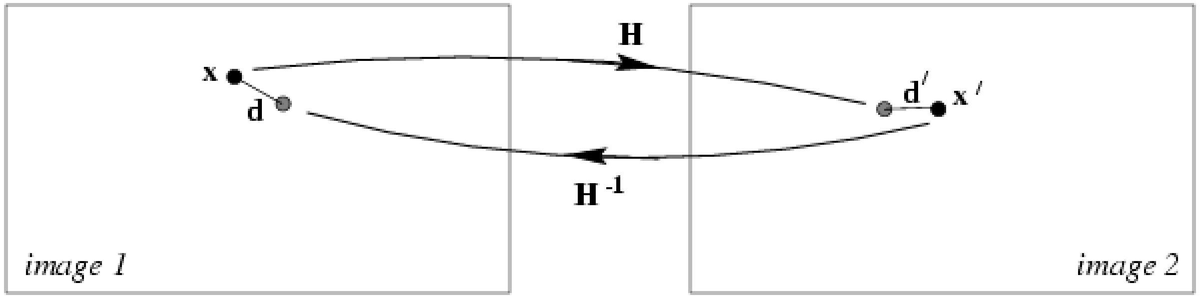
$d(\mathbf{x}, \mathbf{y})$: Euclidean distance



e.g. calibration pattern

- Symmetric transfer error (error in both images)

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



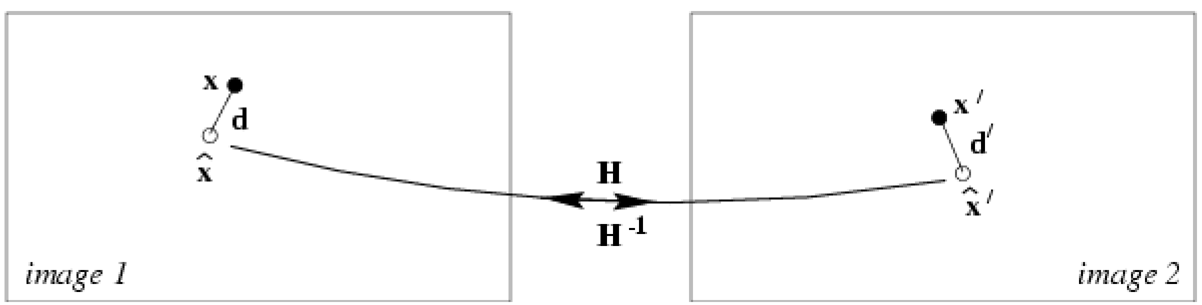
$$d\left(\mathbf{x}, \mathbf{H}^{-1}\mathbf{x}'\right)^2 + d\left(\mathbf{x}', \mathbf{H}\mathbf{x}\right)^2$$

Geometric distance error (Reprojection error)

- Reprojection error
  - ✓ Find $\hat{\mathbf{H}}$ and pairs of perfectly matched points $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ simultaneously

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

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



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

Maximum Likelihood estimation (MLE)

- Optimal estimator in statistical sense
- Assume *iid* mean-zero isotropic Gaussian measurement error (outlier removed):

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

- Error in one image:

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

✓ MLE: $\text{Maximize} \log \Pr(\{\mathbf{x}'_i\} \mid \mathbf{H}) \Rightarrow \text{Minimize} \sum_i d(\mathbf{x}'_i, \mathbf{H}\bar{\mathbf{x}}_i)^2$

- Error in both images

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

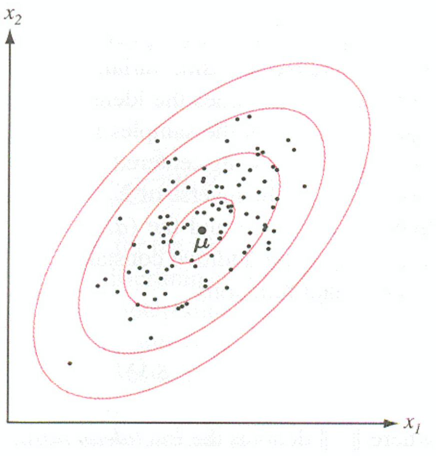
✓ MLE: minimize $\sum d(\mathbf{x}_{\mathrm{i}}, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_{\mathrm{i}}, \hat{\mathbf{x}}'_i)^2$, with $\hat{\mathbf{x}}'_i = \hat{\mathbf{H}}\hat{\mathbf{x}}_i$

Mahalanobis distance

- Consider non-isotropic Gaussian model
- Density normalized distance.

✓ Measurement $\mathbf{X}$ with covariance matrix $\boldsymbol{\Sigma}$

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



✓ **E**rror in two images (independent)

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

✓ Varying covariances

$$\sum_{i}\left\|\mathbf{X}_i-\overline{\mathbf{X}}_i\right\|_{\Sigma_i}^{2}+\left\|\mathbf{X}_i'-\overline{\mathbf{X}}_i'\right\|_{\Sigma_i'}^{2}$$

Normalizing transformations

- How to make DLT invariant to the choice of coordinates?
- Data normalization: **(Essential step in the DLT algorithm)**

✓ Translate so that the centroid to be the origin
✓ Scale so that the average distance to the origin to be $\sqrt{2}$
✓ Perform independently on both images

  - **Improves accuracy**
  - **Invariant to scale and coordinate system**
- Or apply

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

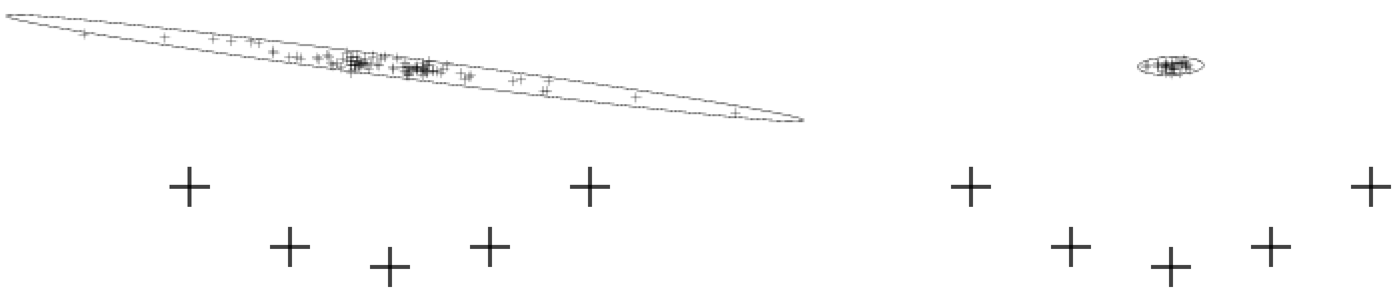
w,*h*: width and height of the image

Importance of normalization

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

$\sim 10^2$ $\sim 10^2$ 1 $\sim 10^2$ $\sim 10^2$ 1 $\sim 10^4$ $\sim 10^4$ $\sim 10^2$

Big difference in orders of magnitude $\longrightarrow$ unstable



Normalized DLT algorithm

Objective

Given $n \geq 4$ 2D to 2D point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}_i'\}$, determine the 2D homography matrix $\mathbf{H}$ such that $\mathbf{x}_i' = \mathbf{H}\mathbf{x}_i$

Algorithm

(i) Normalize points $\tilde{\mathbf{x}}_i = \mathbf{T}_{\mathrm{norm}} \mathbf{x}_i, \; \tilde{\mathbf{x}}_i' = \mathbf{T}_{\mathrm{norm}}' \mathbf{x}_i'$

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

(iii) Denormalize solution $\mathbf{H} = \mathbf{T}'^{-1}_{\mathrm{norm}} \tilde{\mathbf{H}} \mathbf{T}_{\mathrm{norm}}$

- What about the rotation?
- Don't we need normalize the rotation effect? Why?

$$\|\tilde{\mathbf{H}}\| = \|\mathbf{R}'\mathbf{H}\mathbf{R}^{-1}\| = \|\mathbf{H}\|$$