

# Algorithm Evaluation and Error Analysis

Nassir Navab

based on a course given at UNC by Marc Pollefeys & the book "Multiple View Geometry" by Hartley & Zisserman

23 November 2005

chair for computer aided medical procedures

department of computer science | technische universität münchen

# Outline of the Computer Vision Lecture

## projective geometry and transformations

- homogeneous coordinates
- 2D projective geometry
- 3D projective geometry
- 1D projective geometry
- homographies

## estimation

- DLT algorithm
- cost functions: algebraic, geometric, reprojection, sampson error, statistical and MLE, Mahalanobis

## algorithm evaluation and error analysis

- statistics, error propagation

## camera geometry

- camera models
- camera calibration

## advanced estimation

- bundle adjustment
- ICP



## single-view geometry

### two-view geometry

- epipolar geometry / fundamental matrix / essential matrix
- 3D reconstruction of cameras and structure

## three-view geometry

- trifocal tensor
- computation of the trifocal tensor

## N-view geometry

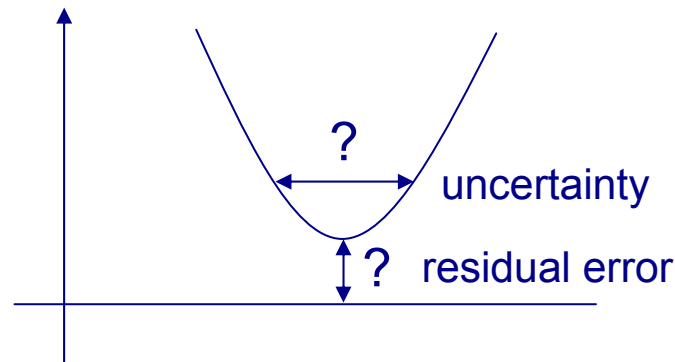
- N-linearities and multiple view tensors
- N-view computational methods
- factorisation
- structure from motion

# Outline :: Algorithm Evaluation & Error Estimation

- Motivation
- Estimation :: Reminder
- Bounds on Performance
- Optimal Estimators (MLE)
- Covariance fo the estimated transformation
  - Forward propagation of covariance
  - Non-linear propagation of covariance
  - Backward propagation of covariance
- Over-parameterization
- Monte Carlo estimation of covariance

# Motivation for Algorithm Evaluation & Error Analysis

- Goal: accessing and quantifying results of estimation algorithms
- How accurate is the algorithm?
  - Bounds on performance
  - Covariance estimation
    - Uncertainty (covariance) vs. Residual error



# Estimation :: Reminder

- Maximum Likelihood Estimation

$$\sum_i \left\| \mathbf{X}_i - \bar{\mathbf{X}}_i \right\|_{\Sigma_i}^2 + \left\| \mathbf{X}'_i - \bar{\mathbf{X}}'_i \right\|_{\Sigma'_i}^2$$

- DLT not invariant  $\Rightarrow$  normalization
- Geometric minimization invariant
- Iterative minimization
  - Cost function
  - Parameterization
  - Initialization
  - Minimization algorithm

# Estimation :: Reminder

## Objective

Compute automatically homography  $H$  between two images

## Algorithm

- **Interest points:** Compute interest points in each image
- **Putative correspondences:** Compute a set of interest point matches based on some similarity measure
- **RANSAC robust estimation:** Repeat for  $N$  samples
  - Select 4 correspondences and compute  $H$
  - Calculate the distance  $d_{\perp}$  for each putative match
  - Compute the number of inliers consistent with  $H$  ( $d_{\perp} < t$ )
  - Choose  $H$  with most inliers
- **Optimal estimation:** re-estimate  $H$  from all inliers by minimizing ML cost function with Levenberg-Marquardt
- **Guided matching:** Determine more matches using prediction by computed  $H$
- Optionally iterate last two steps until convergence

# Bounds on performance

- Test on real or synthetic data
  - $x$  measured coordinates
  - $\hat{x}, \hat{H}$  estimated quantities
  - $\bar{x}, \bar{H}$  true coordinates
- i. Generate synthetic correspondences  $\bar{x}'_i \leftrightarrow \bar{x}_i$
- ii. Add Gaussian noise, yielding  $x'_i \leftrightarrow x_i$
- iii. Estimate  $x'_i \leftrightarrow x_i$  from  $\{\hat{x}'_i, \hat{x}_i\}$  using algorithm; maybe also
- iv. Verify how well  $x'_i = \hat{H}x_i$  or  $\bar{x}'_i = \hat{H}\bar{x}_i$
- v. Repeat many times (different noise, same  $\sigma$ )

- Error in one image

- Noise is added only to the second image

$$\mathbf{x}_i \leftrightarrow \bar{\mathbf{x}}_i \quad \mathbf{x}'_i = \bar{\mathbf{x}}'_i + G(0, \sigma^2)$$

- Estimate  $\hat{\mathbf{H}}$ , then

$$e_{res} = \left( \frac{1}{2n} \sum_{i=1}^n d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2 \right)^{1/2}$$

- Note: residual error  $\neq$  absolute measure of quality of  $\hat{\mathbf{H}}$ 
      - e.g. estimation from four points yields  $\epsilon_{res} = 0$
      - more points better results, but  $\epsilon_{res}$  will increase

- Error in both images

$$\mathbf{x}'_i \leftrightarrow \mathbf{x}_i \quad \mathbf{x}_i = \bar{\mathbf{x}}_i + G(0, \sigma^2) \quad \mathbf{x}'_i = \bar{\mathbf{x}}'_i + G(0, \sigma^2)$$

- Estimate  $\hat{\mathbf{H}}, \hat{\mathbf{x}}_i, \hat{\mathbf{x}}'_i$  such that  $\hat{\mathbf{x}}'_i = \hat{\mathbf{H}}\hat{\mathbf{x}}_i$ , then

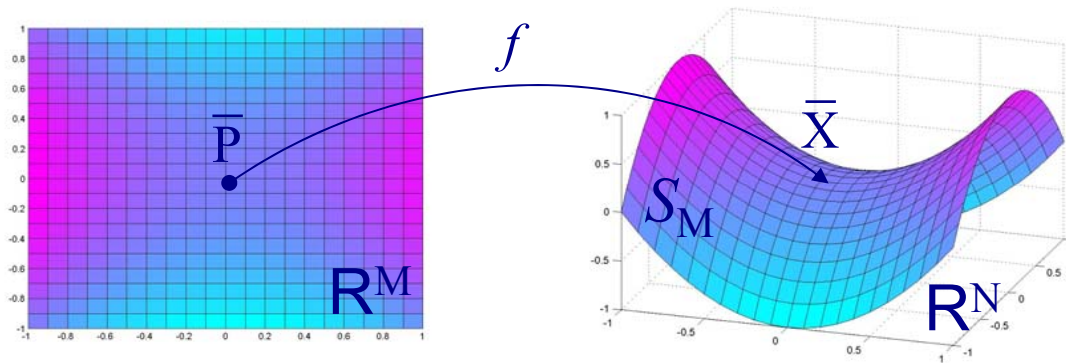
$$e_{res} = \frac{1}{\sqrt{4n}} \left( \sum_{i=1}^n d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + \sum_{i=1}^n d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2 \right)^{1/2}$$

# Optimal estimators (MLE)

- Estimate expected residual error of MLE
- Other algorithms can then be judged to this standard

$f: \mathbb{R}^M \rightarrow \mathbb{R}^N$  (parameter space to measurement space)

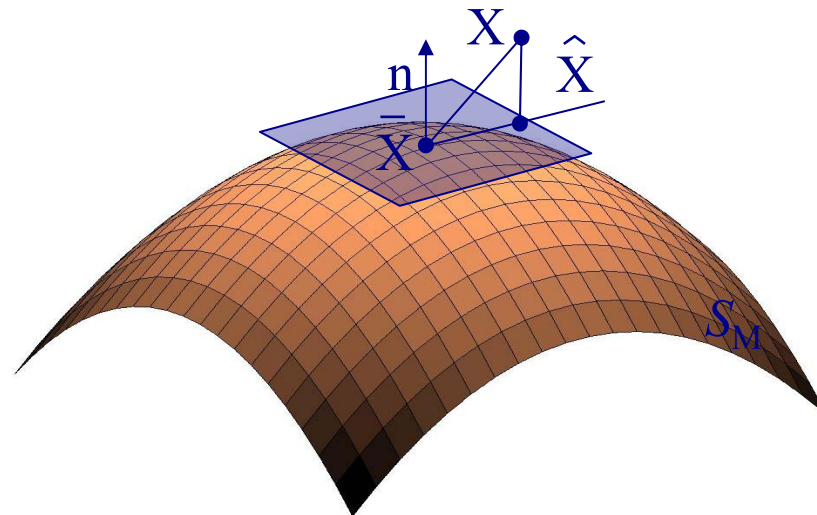
$$\bar{X} \in \mathbb{R}^N \quad \bar{P} \in \mathbb{R}^M \quad f(\bar{P}) = \bar{X}$$



- dimension of submanifold  $S_M = \#(\text{essential parameters})$

# Optimal estimators (MLE)

- $X = G(\bar{X}, N\sigma^2)$
- $\hat{X} = MLE(X)$
- Assume  $S_M$  locally planar around  $\bar{X}$
- Projection of isotropic Gaussian distribution on  $R_N$  with total variance  $N\sigma^2$  onto a subspace of dimension  $s$  is an isotropic Gaussian distribution with total variance  $s\sigma^2$

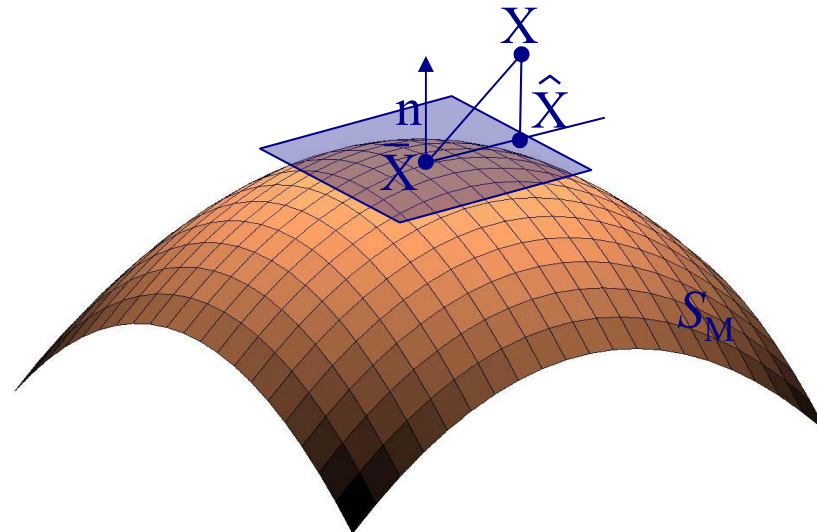


- $N$  measurements (independent Gaussian noise  $\sigma^2$ ) modeled with  $d$  essential parameters (use  $s=d$  and  $s=(N-d)$ )
- i. RMS (root-mean-squared) residual error for ML estimator

$$e_{res} = E \left[ \left\| \hat{\mathbf{X}} - \mathbf{X} \right\|^2 / N \right]^{1/2} = \sigma (1 - d / N)^{1/2}$$

- ii. RMS estimation error for ML estimator

$$e_{est} = E \left[ \left\| \hat{\mathbf{X}} - \bar{\mathbf{X}} \right\|^2 / N \right]^{1/2} = \sigma (d / N)^{1/2}$$



- Error in one image

$$d = 8, N = 2n$$

$$e_{res} = \sigma \sqrt{\frac{N-d}{N}}$$

$$e_{res} = \sigma \sqrt{\frac{2n-8}{2n}}$$

$$e_{est} = \sigma \sqrt{\frac{d}{N}}$$

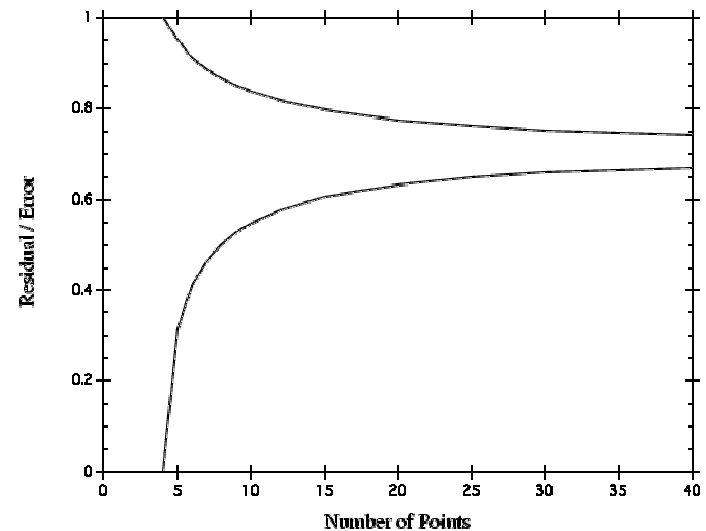
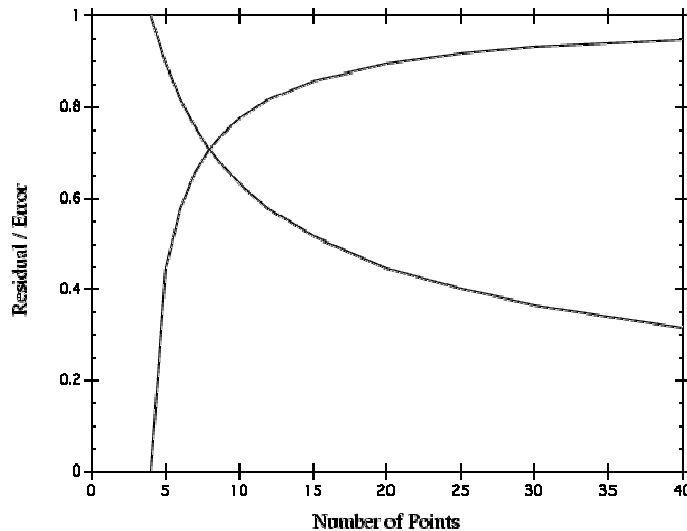
$$e_{est} = \sigma \sqrt{\frac{4}{n}}$$

- Error in two images

$$d = 8 + 2n, N = 4n$$

$$e_{res} = \sigma \sqrt{\frac{n-4}{2n}}$$

$$e_{est} = \sigma \sqrt{\frac{n+4}{2n}}$$



# Covariance of Estimated Model

- Previous question: how close is the error to smallest possible error?
- Independent of point configuration
  
- Real question: how close is estimated model to real model?
- Dependent on point configuration  
(e.g. 4 points close to a line)

## Forward propagation of covariance

Let  $\mathbf{v}$  be a random vector in  $\mathbb{R}^M$  with mean  $\bar{\mathbf{v}}$  and covariance matrix  $\Sigma$ , and suppose that  $f: \mathbb{R}^M \rightarrow \mathbb{R}^N$  is an affine mapping defined by  $f(\mathbf{v}) = f(\bar{\mathbf{v}}) + A(\mathbf{v} - \bar{\mathbf{v}})$ . Then  $f(\mathbf{v})$  is a random variable with mean  $f(\bar{\mathbf{v}})$  and covariance matrix  $A\Sigma A^T$ .

Note: does not assume  $A$  is a square matrix

# Example

- Given two random variables (zero-mean,  $\sigma = 1/2$ ):

$$x = G(0,1^2), y = G(0,2^2)$$

- What is the mean and standard deviation of

$$x' = f(x, y) = 3x + 2y - 7$$

- Mean:

$$\bar{x}' = f(0,0) = -7$$

- Variance:

$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} \quad A = [3 \quad 2]$$

$$\sigma^2 = A\Sigma A^T = [3 \quad 2] \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix} = 25 \Rightarrow \sigma(x') = 5$$

# Example

- Given two random variables (zero-mean,  $\sigma = 1/2$ ):

$$x = G(0, 1^2), y = G(0, 2^2)$$

- What are mean and standard deviation of

$$x' = 3x + 2y$$

$$y' = 3x - 2y$$

- Result:

$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} \quad \mathbf{A} = \begin{bmatrix} 3 & 2 \\ 3 & -2 \end{bmatrix}$$

$$\mathbf{A}\Sigma\mathbf{A}^T = \begin{bmatrix} 25 & -7 \\ -7 & 25 \end{bmatrix} \Rightarrow \sigma = \sqrt{25} = 5, \quad E[x'y'] = -7$$

## Non-linear propagation of covariance

$$f(\mathbf{v}) \approx f(\bar{\mathbf{v}}) + \mathbf{J}(\mathbf{v} - \bar{\mathbf{v}}) \quad J_{ij} = \frac{\partial f_i}{\partial v_j}$$

Let  $\mathbf{v}$  be a random vector in  $\mathbb{R}^M$  with mean  $\bar{\mathbf{v}}$  and covariance matrix  $\Sigma$ , and suppose that  $f: \mathbb{R}^M \rightarrow \mathbb{R}^N$  differentiable in the neighborhood of  $\bar{\mathbf{v}}$ . Then, up to a first order approximation,  $f(\mathbf{v})$  is a random variable with mean  $f(\bar{\mathbf{v}})$  and covariance matrix  $\mathbf{J}\Sigma\mathbf{J}^\top$ , where  $\mathbf{J}$  is the Jacobian matrix evaluated at  $\bar{\mathbf{v}}$

Note: good approximation if  $f$  close to linear within variability of  $\mathbf{v}$

# Example

- Given a Gaussian random vector (zero-mean,  $\sigma = \text{diag}(1, 4)$ )

$$\mathbf{x} = (x, y)^T = G\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}\right)$$

- What is the true mean and standard deviation of

$$x' = f(x, y) = x^2 + 3x - 2y + 5$$

according to

$$\bar{x}' = \int \int_{-\infty}^{\infty} P(x, y) f(x, y) dx dy$$

$$\sigma^2 = \int \int_{-\infty}^{\infty} P(x, y) (f(x, y) - \bar{x}')^2 dx dy$$

where

$$P(x, y) = \frac{1}{8\pi} e^{-(x^2 + y^2/4)/2\sigma^2}$$

# Example

- One obtains

$$\bar{x}' = 5 + \sigma^2$$

$$\sigma_{x'}^2 = 25\sigma^2 + 2\sigma^4$$

and with the previous approximation, noting that  $J = \begin{bmatrix} 3 & -2 \end{bmatrix}$

$$\bar{x}' = 5$$

$$\sigma_{x'}^2 = \sigma^2 \begin{bmatrix} 3 & -2 \end{bmatrix} \begin{bmatrix} 1 & \\ & 4 \end{bmatrix} \begin{bmatrix} 3 \\ -2 \end{bmatrix} = 25\sigma^2$$

# Example

- Given independent zero-mean Gaussian random variables  $x$  and  $y$  and the function

$$f(x, y) = ax^2 + bxy + cy^2 + dx + ey + f$$

one may compute

$$\text{mean} = a\sigma_x^2 + c\sigma_y^2 + f$$

$$\text{variance} = 2a^2\sigma_x^4 + b\sigma_x^2\sigma_y^2 + 2c^2\sigma_y^2 + d\sigma_x^2 + e\sigma_y^2$$

which are close to the estimated

$$\text{est. mean} = f$$

$$\text{est. variance} = d\sigma_x^2 + e\sigma_y^2$$

if  $\sigma_x$  and  $\sigma_y$  are small.

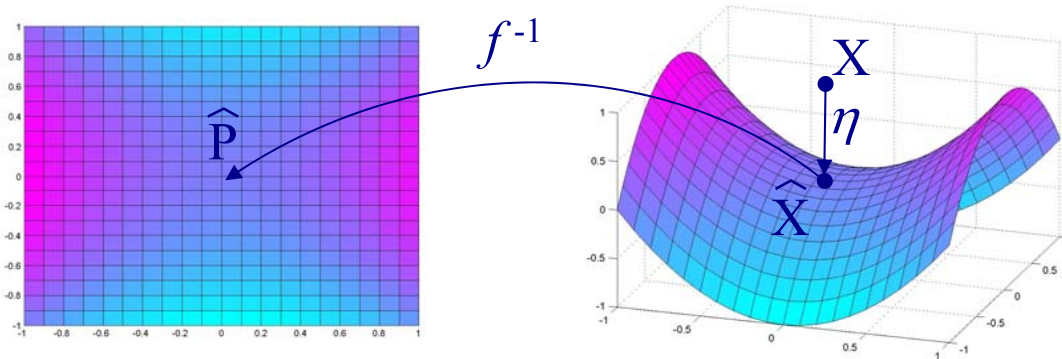
# Backward propagation of covariance

$$f: \mathbb{R}^M \rightarrow \mathbb{R}^N$$

$$\hat{X} \in \mathbb{R}^N \quad \hat{P} \in \mathbb{R}^M$$

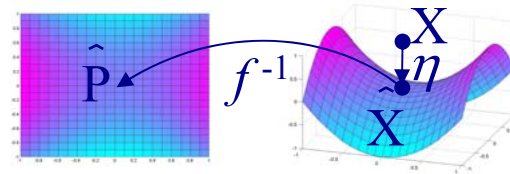
$$\eta(\hat{X}) = X \quad f^{-1}(\hat{X}) = \hat{P}$$

$$f^{-1} \circ \eta(X) = \hat{P}$$



# Backward propagation of covariance

$$f^{-1} \circ \eta(\mathbf{X}) = \hat{\mathbf{P}}$$



- assume  $f$  is affine

$$f(\mathbf{P}) = f(\bar{\mathbf{P}}) + \mathbf{J}(\mathbf{P} - \bar{\mathbf{P}}) \quad f(\bar{\mathbf{P}}) = \bar{\mathbf{X}}$$

- what about  $f^{-1} \circ \eta$ ?

- minimize:  $\|\mathbf{X} - \hat{\mathbf{X}}\|_{\Sigma} = \|\mathbf{X} - f(\hat{\mathbf{P}})\|_{\Sigma} = \|(\mathbf{X} - \bar{\mathbf{X}}) - \mathbf{J}(\hat{\mathbf{P}} - \bar{\mathbf{P}})\|_{\Sigma}$

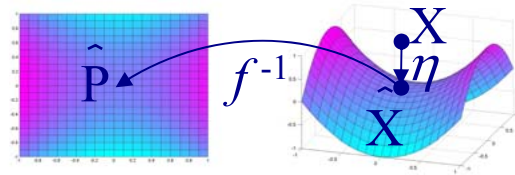
- solution:  $(\hat{\mathbf{P}} - \bar{\mathbf{P}}) = (\mathbf{J}^T \Sigma^{-1} \mathbf{J})^{-1} \mathbf{J}^T \Sigma^{-1} (\mathbf{X} - \bar{\mathbf{X}})$

$$\bar{\mathbf{P}} = f^{-1}(\bar{\mathbf{X}}) \quad \hat{\mathbf{P}} = f^{-1}(\hat{\mathbf{X}})$$

$$f^{-1} \circ \eta(\mathbf{X}) = \hat{\mathbf{P}} = (\mathbf{J}^T \Sigma^{-1} \mathbf{J})^{-1} \mathbf{J}^T \Sigma^{-1} (\mathbf{X} - \bar{\mathbf{X}}) + f^{-1} \circ \eta(\bar{\mathbf{X}})$$

# Backward propagation of covariance

$$f^{-1} \circ \eta(\mathbf{X}) = \hat{\mathbf{P}}$$



$$\mathbf{A} = \left( \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J} \right)^{-1} \mathbf{J}^T \boldsymbol{\Sigma}^{-1}$$

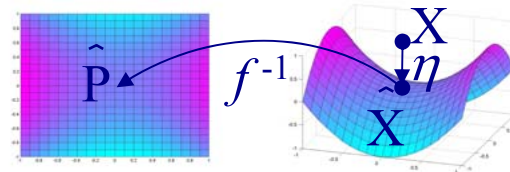
$$\boldsymbol{\Sigma}_{f^{-1} \circ \eta} = \left[ \left( \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J} \right)^{-1} \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \right] \boldsymbol{\Sigma} \left[ \left( \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J} \right)^{-1} \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \right]^T$$

$$= \left( \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J} \right)^{-1} \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\Sigma} \boldsymbol{\Sigma}^{-T} \mathbf{J} \left( \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J} \right)^{-T}$$

$$= \left( \mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J} \right)^{-1}$$

# Backward propagation of covariance

$$f^{-1} \circ \eta(\mathbf{X}) = \hat{\mathbf{P}}$$

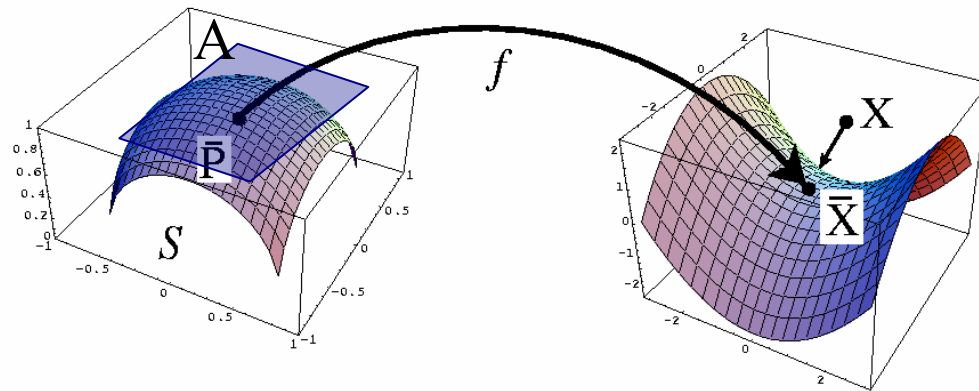


If  $f$  is affine, then  $\Sigma_{\mathbf{P}} = \left( \mathbf{J}^T \Sigma_{\mathbf{X}}^{-1} \mathbf{J} \right)^{-1}$

non-linear case, obtain first order approximations by using Jacobian

# Over-parameterization

- In this case  $f$  is not one-to-one and  $\text{rank } J < M$   
 so  $\Sigma_P = (J^T \Sigma_X^{-1} J)^{-1}$  can not hold
  - e.g. scale ambiguity implies infinite variance!
- However, if constraints are imposed, then ok.



$$\Sigma_P = (J^T \Sigma_X^{-1} J)^{+A} = A (A^T J^T \Sigma_X^{-1} J A)^{-1} A^T$$

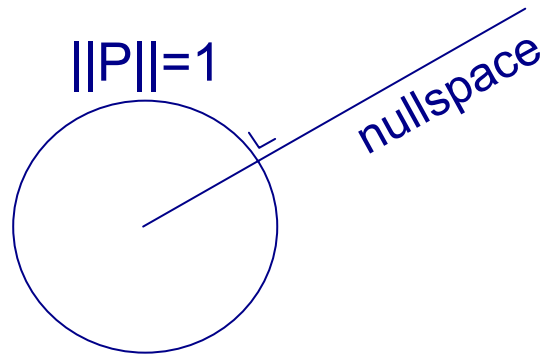
- Invert  $d \times d$  in stead of  $M \times M$

# Over-parameterization

- When constraint surface is locally orthogonal to the null space of  $J$

$$\Sigma_P = \left( J^T \Sigma_x^{-1} J \right)^+ \quad (\text{pseudo-inverse})$$

- e.g. usual constraint  $\|P\|=1$



## Example: error in one image

- i. Estimate the transformation  $\hat{\mathbf{H}}$  from the data
- ii. Compute Jacobian  $\mathbf{J}_f = \partial \mathbf{X}' / \partial \mathbf{h}$ , evaluated at  $\hat{\mathbf{h}}$
- iii. The covariance matrix of the estimated  $\hat{\mathbf{h}}$  is given by

$$\Sigma_{\mathbf{h}} = \left( \mathbf{J}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{J} \right)^+$$

$$\mathbf{J}_i = \partial \mathbf{x}'_i / \partial \mathbf{h} = \frac{1}{w'_i} \begin{bmatrix} \tilde{\mathbf{x}}_i^T & 0 & -x'_i \tilde{\mathbf{x}}_i^T \\ 0 & \tilde{\mathbf{x}}_i^T & -y'_i \tilde{\mathbf{x}}_i^T \end{bmatrix}$$

$$\Sigma_{\mathbf{h}} = \left( \mathbf{J}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{J} \right)^+ = \left( \sum_i \mathbf{J}_i^T \Sigma_i^{-1} \mathbf{J}_i \right)^+$$

## Example: error in both images

- Given

$$\mathbf{J} = [\mathbf{A} \mid \mathbf{B}]$$

- separate in homography and point parameters

$$\mathbf{J}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{J} = \begin{bmatrix} \mathbf{A}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{A} & \mathbf{A}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{B} \\ \mathbf{B}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{A} & \mathbf{B}^T \Sigma_{\mathbf{x}'}^{-1} \mathbf{B} \end{bmatrix}$$

# Using covariance matrix in point transfer

- Error in one image

$$\Sigma_{x'} = J_h \Sigma_h J_h^T$$

- Error in two images

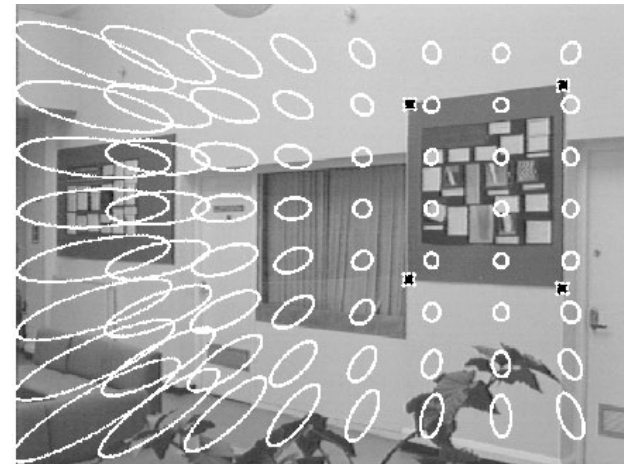
$$\Sigma_{x'} = J_h \Sigma_h J_h^T + J_x \Sigma_x J_x^T$$

(if  $h$  and  $x$  independent, i.e. new points)

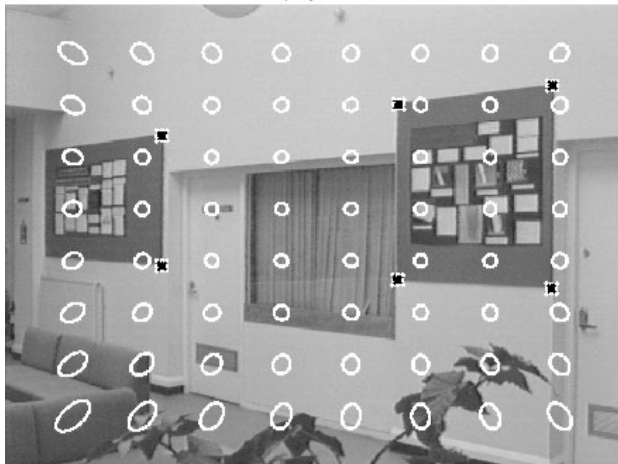
# Example:



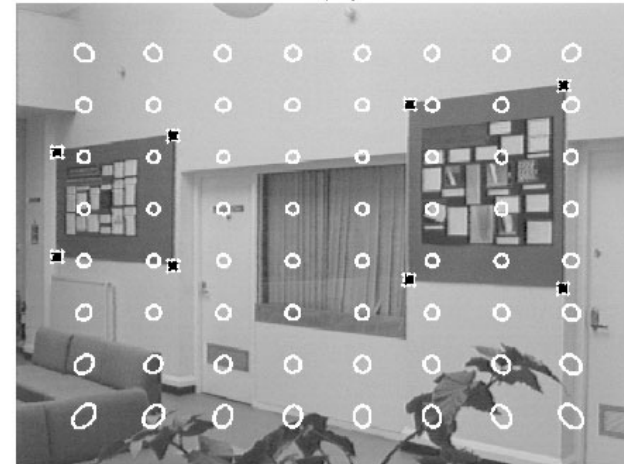
(a)



(b)



(c)



(d)

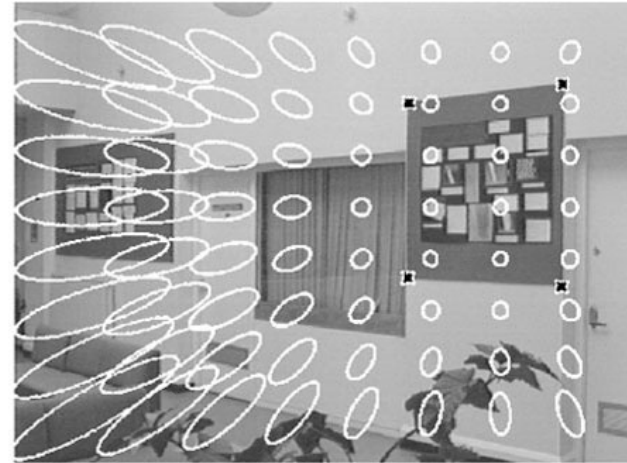
$\sigma=1$  pixel  $\Sigma=0.5$ cm

(Crimisi'97)

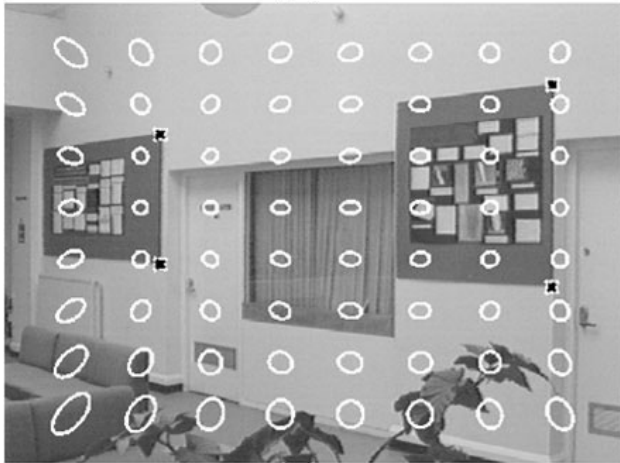
# Example:



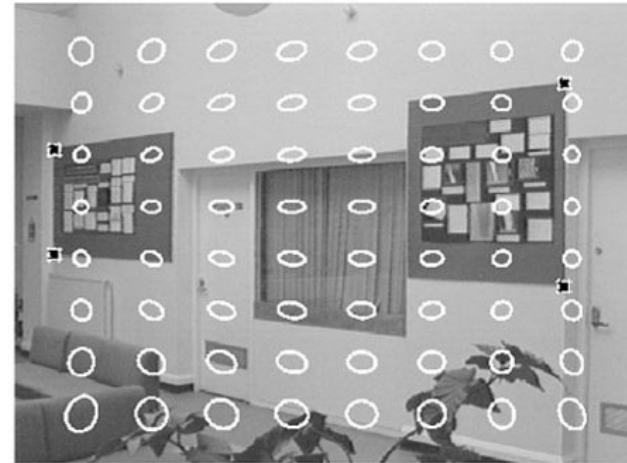
(a)



(b)



(c)



(d)

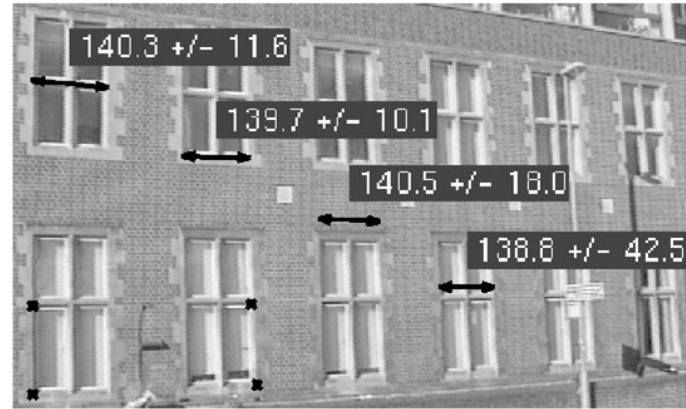
$\sigma=1$  pixel  $\Sigma=0.5$ cm

(Crimisi'97)

# Example:



(a)



(b)



(c)

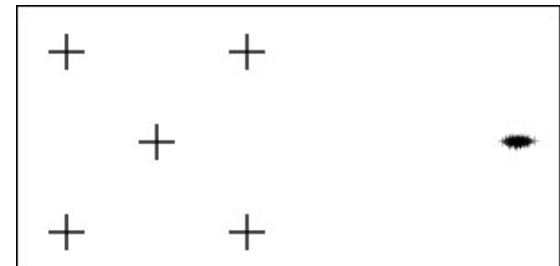
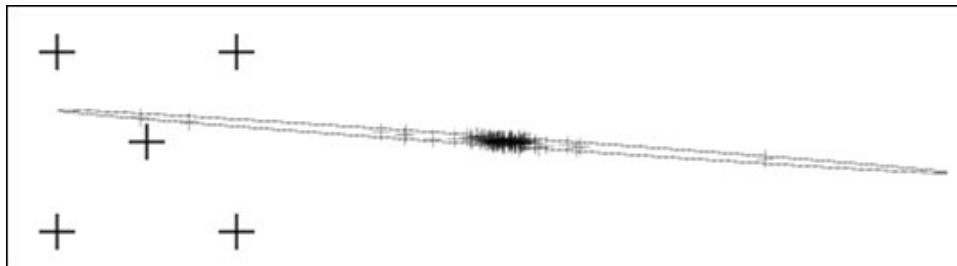


(d)

(Crimisi'97)

# Monte Carlo estimation of covariance

- To be used when previous assumptions do not hold (e.g. non-flat within variance) or too complicated to compute.
- Simple and general, but expensive
- Generate samples according to assumed noise distribution, carry out computations, observe distribution of result



# Next Class :: Camera Models

