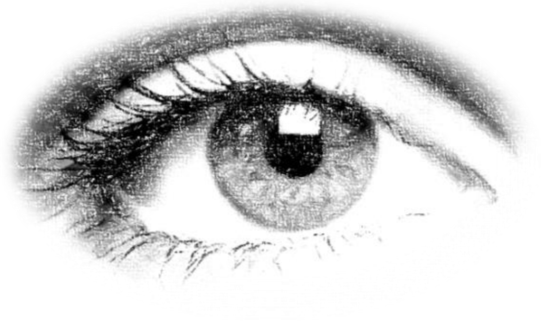


Two-View Geometry: Computation of the Fundamental Matrix F

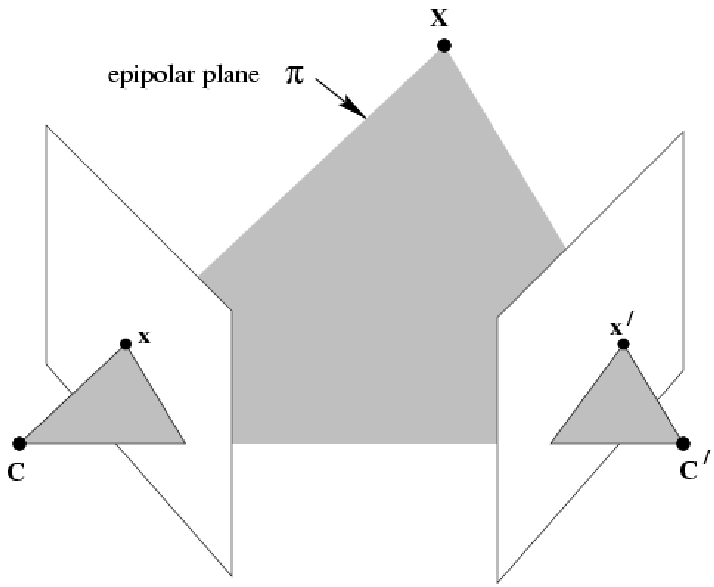
簡韶逸 Shao-Yi Chien

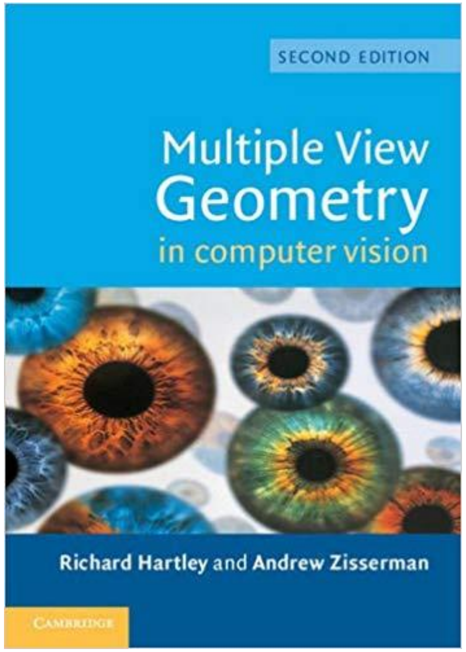
Department of Electrical Engineering

National Taiwan University

Outline

- Computation of the fundamental matrix F





[Slides credit: Marc Pollefeys]

Epipolar Geometry: Basic Equation

$$\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\mathrm{x} = 0$$

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0$$

separate known from unknown

$$[x'x, x'y, x', y'x, y'y, y', x, y, 1][f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33}]^{\mathrm{T}} = 0$$

(data) (unknowns)

(linear)

$$\begin{bmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_n x_n & x'_n y_n & x'_n & y'_n x_n & y'_n y_n & y'_n & x_n & y_n & 1 \end{bmatrix} \mathrm{f} = 0$$

$$\mathrm{Af} = 0$$

The Singularity Constraint

$$e'^{T} F = 0 \qquad Fe = 0 \qquad \det F = 0 \qquad \text{rank } F = 2$$

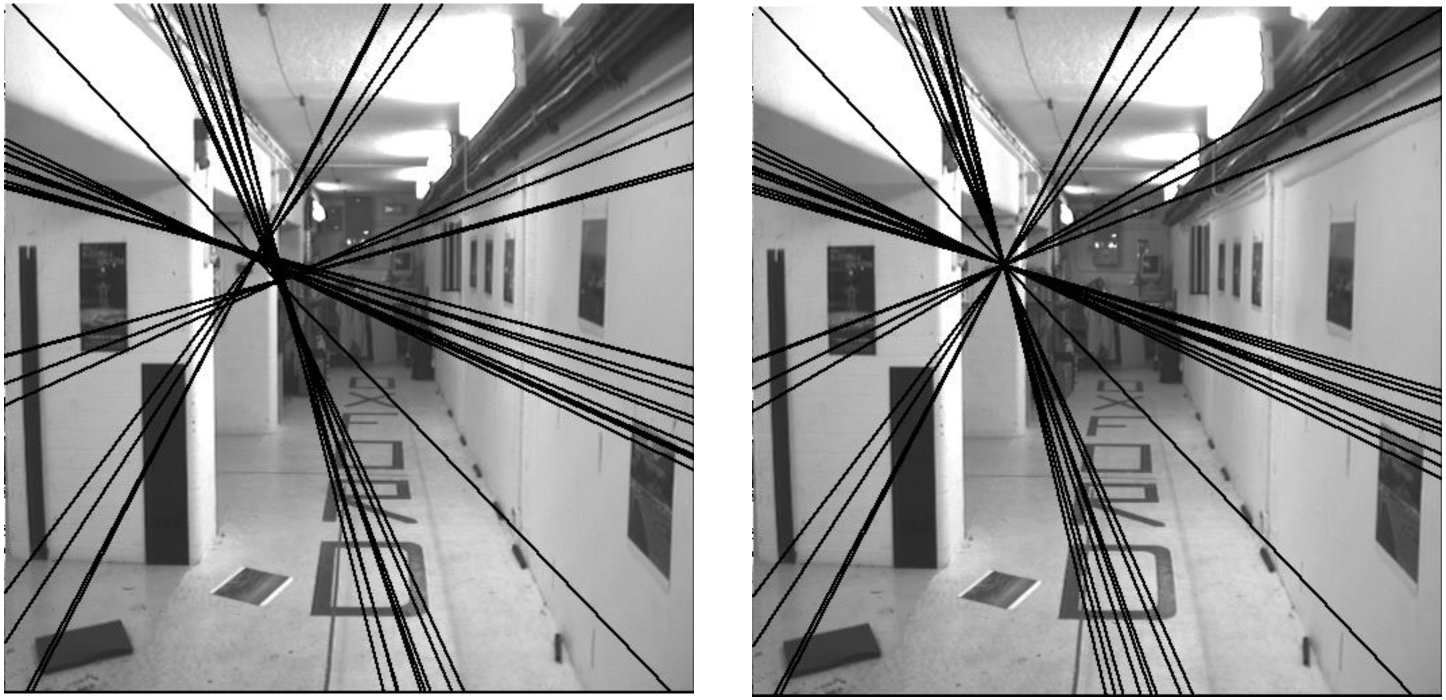
SVD from linearly computed F matrix (rank 3)

$$F = U \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{bmatrix} V^T = U_1 \sigma_1 V_1^T + U_2 \sigma_2 V_2^T + U_3 \sigma_3 V_3^T$$

Compute closest rank-2 approximation $\qquad \min \left\| F - F' \right\|_F$

$$F' = U \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & 0 \end{bmatrix} V^T = U_1 \sigma_1 V_1^T + U_2 \sigma_2 V_2^T$$

Enforcing singularity!



Effect of enforcing singularity

The Minimum Case – 7 Point Correspondences

$$\begin{bmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_7 x_7 & x'_7 y_7 & x'_7 & y'_7 x_7 & y'_7 y_7 & y'_7 & x_7 & y_7 & 1 \end{bmatrix} \mathrm{f} = 0$$

$$\mathrm{A} = \mathrm{U}_{7\mathrm{x}7} \operatorname{diag}(\sigma_1, \ldots, \sigma_7, 0, 0) \mathrm{V}_{9\mathrm{x}9}{}^{\mathrm{T}}$$

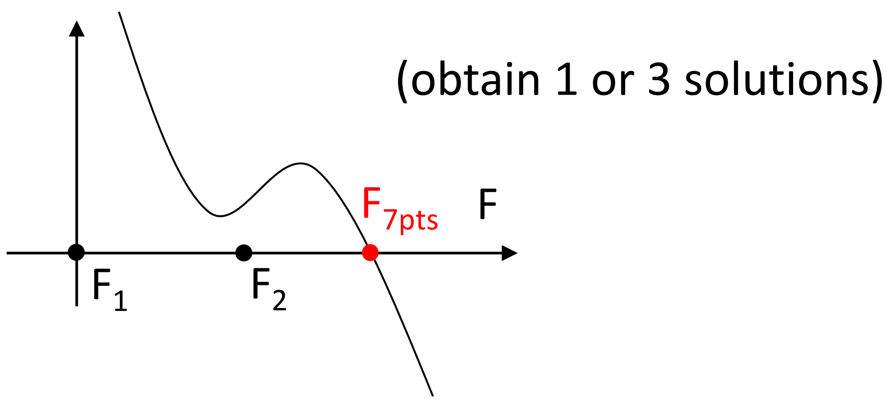
$$\Rightarrow \quad \mathrm{A}[\mathrm{V}_8 \mathrm{V}_9] = 0_{7\times 2} \qquad \left(\text{e.g.} \mathrm{V}^{\mathrm{T}} \mathrm{V}_8 = [000000010]^{\mathrm{T}}\right)$$

$$\mathrm{x}_i{}^{\mathrm{T}} (\mathrm{F}_1 + \lambda \mathrm{F}_2) \mathrm{x}_i = 0, \forall i = 1 \ldots 7$$

one parameter family of solutions

but $F_1+\lambda F_2$ not automatically rank 2

The Minimum Case – Impose Rank 2



$$\det(F_1 + \lambda F_2) = a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0$$

(cubic equation)

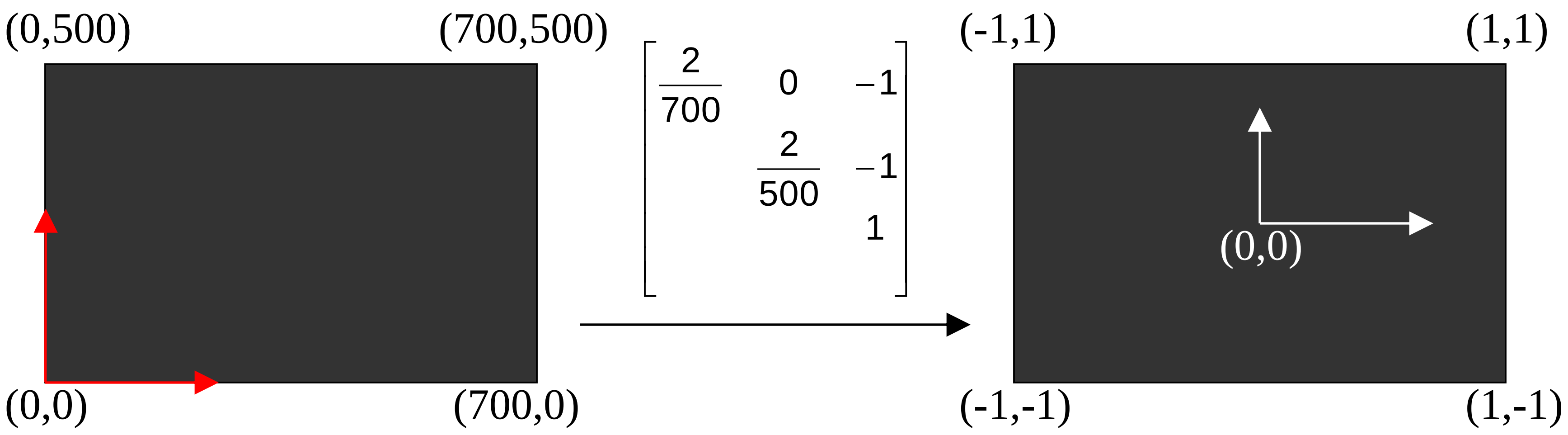
NOT Normalized 8-point Algorithm

$$
\begin{array}{c}
\begin{bmatrix}
x_1x_1' & y_1x_1' & x_1' & x_1y_1' & y_1y_1' & y_1' & x_1 & y_1 & 1 \\
x_2x_2' & y_2x_2' & x_2' & x_2y_2' & y_2y_2' & y_2' & x_2 & y_2 & 1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_nx_n' & y_nx_n' & x_n' & x_ny_n' & y_ny_n' & y_n' & x_n & y_n & 1
\end{bmatrix} \\
\begin{array}{ccccccccc}
\sim 10000 & \sim 10000 & \sim 100 & \sim 10000 & \sim 10000 & \sim 100 & \sim 100 & \sim 100 & 1
\end{array}
\end{array}
\begin{bmatrix}
f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33}
\end{bmatrix} = 0
$$

Orders of magnitude difference
Between column of data matrix
$\rightarrow$ least-squares yields poor results

The Normalized 8-point Algorithm

Transform image to ~[-1,1]x[-1,1]

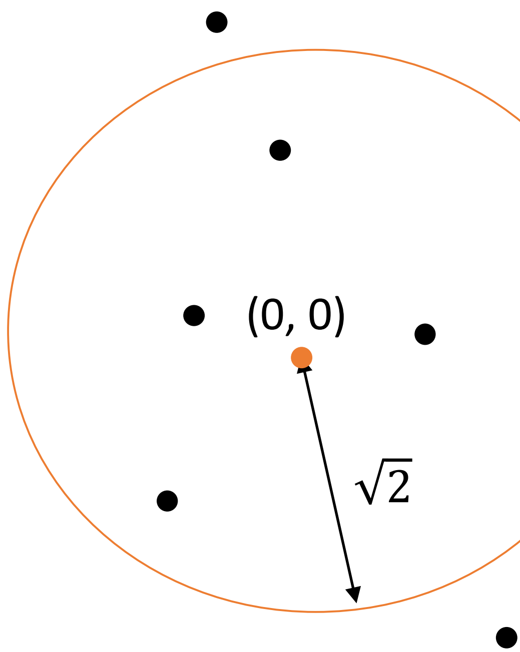


$$\begin{bmatrix} \frac{2}{700} & 0 & -1 \\ & \frac{2}{500} & -1 \\ & & 1 \end{bmatrix}$$

Least squares yields good results (Hartley, PAMI´97)

The Normalized 8-point Algorithm

Or nomalization with the previous method:



The Normalized 8-point Algorithm

Objective

Given $n \geq 8$ image point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}$, determine the fundamental matrix $\mathtt{F}$ such that $\mathbf{x}'^{\mathsf{T}}_i \mathtt{F} \mathbf{x}_i = 0$.

Algorithm

(i) **Normalization:** Transform the image coordinates according to $\hat{\mathbf{x}}_i = \mathtt{T}\mathbf{x}_i$ and $\hat{\mathbf{x}}'_i = \mathtt{T}'\mathbf{x}'_i$, where $\mathtt{T}$ and $\mathtt{T}'$ are normalizing transformations consisting of a translation and scaling.

(ii) Find the fundamental matrix $\hat{\mathtt{F}}'$ corresponding to the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$ by

  (a) **Linear solution:** Determine $\hat{\mathtt{F}}$ from the singular vector corresponding to the smallest singular value of $\hat{\mathtt{A}}$, where $\hat{\mathtt{A}}$ is composed from the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$ as defined in (11.3).

  (b) **Constraint enforcement:** Replace $\hat{\mathtt{F}}$ by $\hat{\mathtt{F}}'$ such that $\det \hat{\mathtt{F}}' = 0$ using the SVD (see section 11.1.1).

(iii) **Denormalization:** Set $\mathtt{F} = \mathtt{T}'^{\mathsf{T}} \hat{\mathtt{F}}' \mathtt{T}$. Matrix $\mathtt{F}$ is the fundamental matrix corresponding to the original data $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$.

Geometric Distance

- Gold standard
- Sampson error
- Symmetric epipolar distance

Gold Standard

Maximum Likelihood Estimation (= least-squares for Gaussian noise)

$$\sum_i d\left(\mathrm{x}_i, \hat{\mathrm{x}}_i\right)^2 + d\left(\mathrm{x}'_i, \hat{\mathrm{x}}'_i\right)^2 \quad \text{subject to } \hat{\mathrm{x}}'^{\mathrm{T}} \mathrm{F} \hat{\mathrm{x}} = 0$$

Initialize: normalized 8-point, (P,P‘) from F, reconstruct X_i

Parameterize:

$$\mathrm{P} = [\mathrm{I} \mid 0], \mathrm{P}' = [\mathrm{M} \mid \mathrm{t}], \mathrm{X}_i$$

$$\hat{\mathrm{x}}_i = \mathrm{P}\mathrm{X}_i, \hat{\mathrm{x}}'_i = \mathrm{P}'\mathrm{X}_i$$

(overparametrized)

Minimize cost using Levenberg-Marquardt
(preferably sparse LM, see book)

Objective

Given $n \geq 8$ image point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}$, determine the Maximum Likelihood estimate $\hat{\mathtt{F}}$ of the fundamental matrix.

The MLE involves also solving for a set of subsidiary point correspondences $\{\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i\}$, such that $\hat{\mathbf{x}}'^{\mathsf{T}}_i \hat{\mathtt{F}} \hat{\mathbf{x}}_i = 0$, and which minimizes

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2.$$

Algorithm (Expensive Method)

(i) Compute an initial rank 2 estimate of $\hat{\mathtt{F}}$ using a linear algorithm such as algorithm 11.1.

(ii) Compute an initial estimate of the subsidiary variables $\{\hat{\mathbf{x}}_i, \hat{\mathbf{x}}'_i\}$ as follows:

- (a) Choose camera matrices $\mathtt{P} = [\mathtt{I} \mid \mathbf{0}]$ and $\mathtt{P}' = [[\mathbf{e}']_\times \hat{\mathtt{F}} \mid \mathbf{e}']$, where $\mathbf{e}'$ is obtained from $\hat{\mathtt{F}}$.
- (b) From the correspondence $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ and $\hat{\mathtt{F}}$ determine an estimate of $\widehat{\mathbf{X}}_i$ using the triangulation method of chapter 12.
- (c) The correspondence consistent with $\hat{\mathtt{F}}$ is obtained as $\hat{\mathbf{x}}_i = \mathtt{P}\widehat{\mathbf{X}}_i,\ \hat{\mathbf{x}}'_i = \mathtt{P}'\widehat{\mathbf{X}}_i$.

(iii) Minimize the cost

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2$$

over $\hat{\mathtt{F}}$ and $\widehat{\mathbf{X}}_i,\ i = 1, \ldots, n$. The cost is minimized using the Levenberg–Marquardt algorithm over $3n + 12$ variables: $3n$ for the n 3D points $\widehat{\mathbf{X}}_i$, and 12 for the camera matrix $\mathtt{P}' = [\mathtt{M} \mid \mathbf{t}]$, with $\hat{\mathtt{F}} = [\mathbf{t}]_\times \mathtt{M}$, and $\hat{\mathbf{x}}_i = \mathtt{P}\widehat{\mathbf{X}}_i,\ \hat{\mathbf{x}}'_i = \mathtt{P}'\widehat{\mathbf{X}}_i$.

First-order Geometric Error (Sampson Error)

$$\sum e^{\mathrm{T}} \left(\mathrm{JJ}^{\mathrm{T}}\right)^{-1} e \qquad \sum \frac{e^{\mathrm{T}} e}{\mathrm{JJ}^{\mathrm{T}}}$$

(one eq./point
$\Rightarrow$ JJ^{T} scalar)

$$e = \sum \mathrm{x}'^{\mathrm{T}}\, \mathrm{F}\mathrm{x} = 0$$

$$\mathrm{JJ}^{\mathrm{T}} = \left(\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\right)_1^2 + \left(\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\right)_2^2 + \left(\mathrm{F}\mathrm{x}\right)_1^2 + \left(\mathrm{F}\mathrm{x}\right)_2^2$$

where $(Fx_i)_j^2$ represents the square of the j-th entry of the vector Fx_i

$$\sum \frac{e^{\mathrm{T}} e}{\mathrm{JJ}^{\mathrm{T}}} \qquad \sum \frac{\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\mathrm{x}}{\left(\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\right)_1^2 + \left(\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\right)_2^2 + \left(\mathrm{F}\mathrm{x}\right)_1^2 + \left(\mathrm{F}\mathrm{x}\right)_2^2}$$

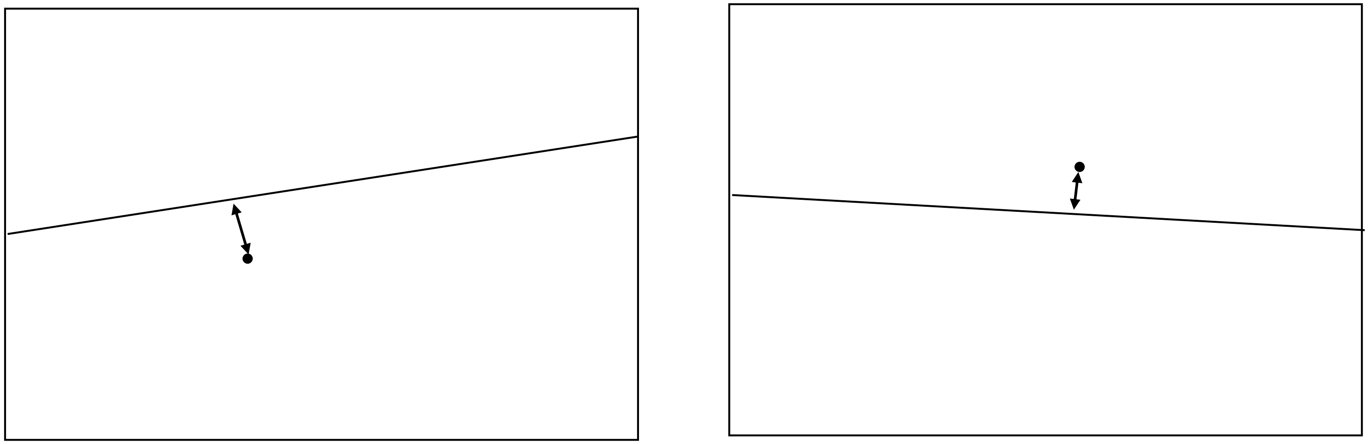
(problem if some x is located at epipole)

<u>advantage</u>: no subsidiary variables required

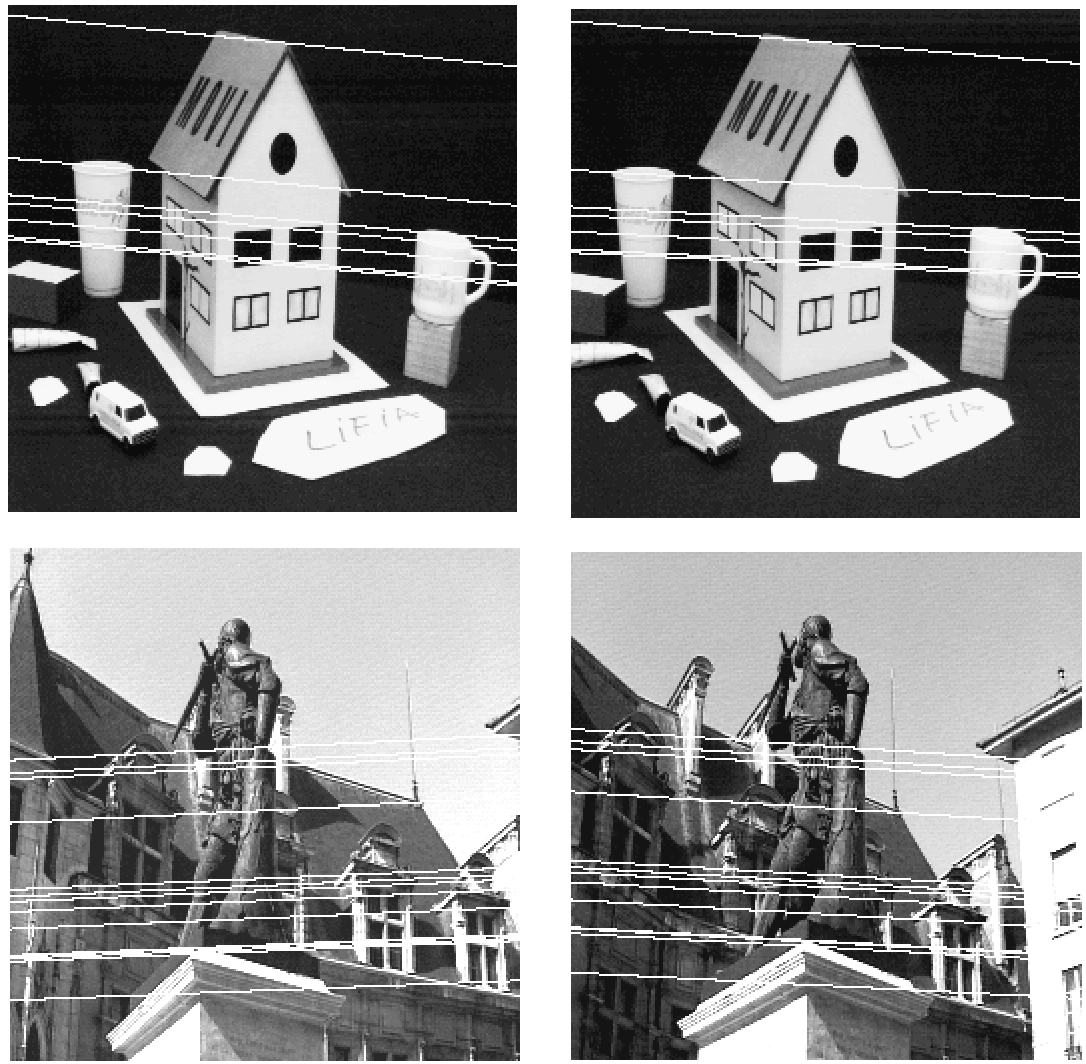
Symmetric Epipolar Error

$$\sum_i d\left(\mathrm{x}'_i, F\mathrm{x}_i\right)^2 + d\left(\mathrm{x}_i, \mathrm{F}^{\mathrm{T}}\mathrm{x}'_i\right)^2$$

$$= \sum \mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\mathrm{x}\left(\frac{1}{\left(\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\right)_1^2 + \left(\mathrm{x}'^{\mathrm{T}}\,\mathrm{F}\right)_2^2} + \frac{1}{\left(\mathrm{F}\mathrm{x}\right)_1^2 + \left(\mathrm{F}\mathrm{x}\right)_2^2}\right)$$

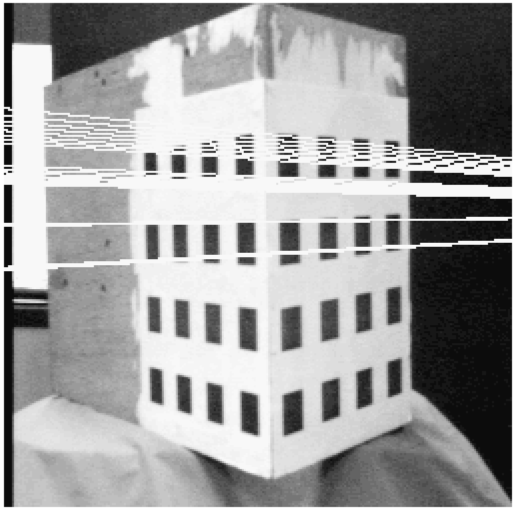
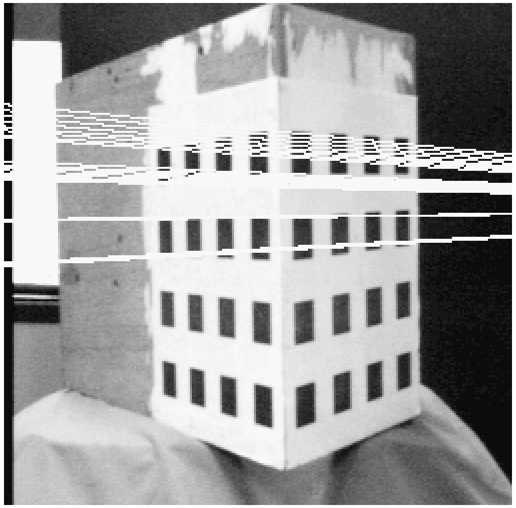


Some experiments:

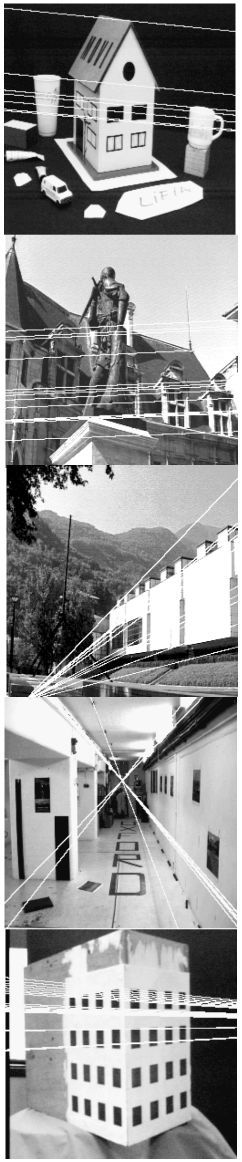


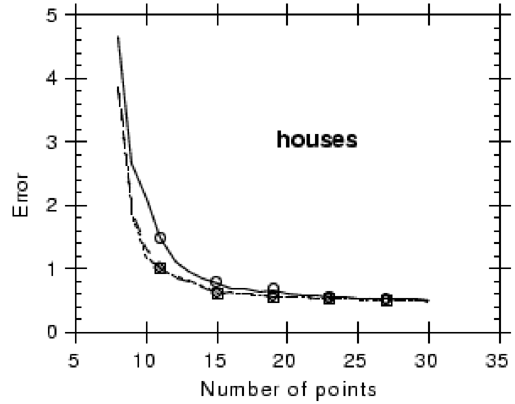
Some experiments:

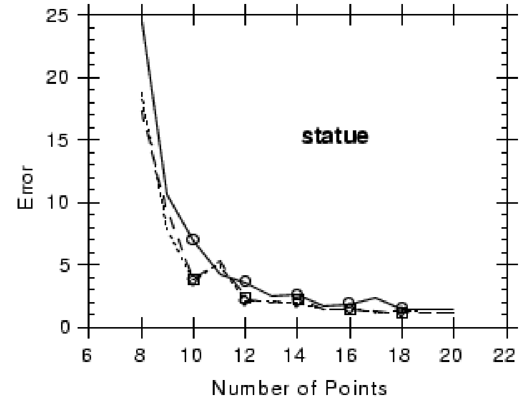
Some experiments:

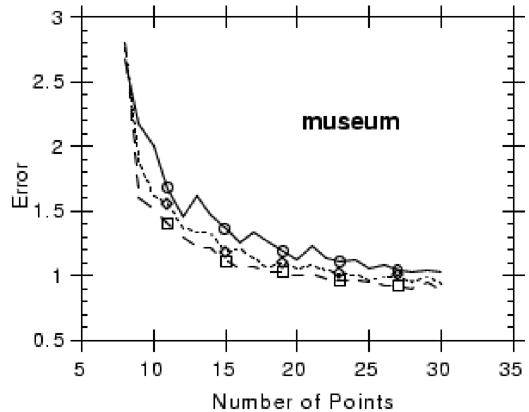



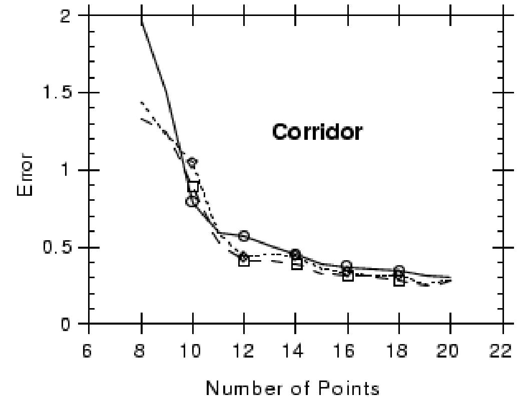
Some experiments:

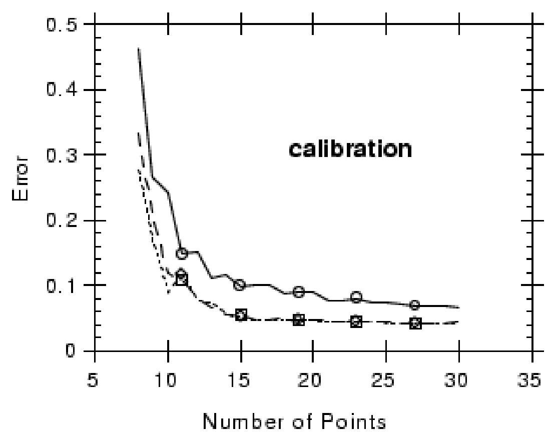












Residual error:

$$\sum_i d\left(\mathrm{x}'_i, F\mathrm{x}_i\right)^2 + d\left(\mathrm{x}_i, \mathrm{F}^{\mathrm{T}}\mathrm{x}'_i\right)^2$$

(for all points)

Recommendations

- Do not use unnormalized algorithms
- Quick and easy to implement: 8-point normalized
- Better: enforce rank-2 constraint during minimization
- Best: Maximum Likelihood Estimation

Automatic Computation of F

1. Interest points
2. Putative correspondences
3. RANSAC
4. Non-linear re-estimation of F
5. Guided matching

(repeat 4 and 5 until stable)

Automatic Computation of F

Objective Compute the fundamental matrix between two images.

Algorithm

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

(ii) **Putative correspondences:** Compute a set of interest point matches based on proximity and similarity of their intensity neighbourhood.

(iii) **RANSAC robust estimation:** Repeat for N samples, where N is determined adaptively as in algorithm 4.5(p121):

  (a) Select a random sample of 7 correspondences and compute the fundamental matrix $\mathtt{F}$ as described in section 11.1.2. There will be one or three real solutions.

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

  (c) Compute the number of inliers consistent with $\mathtt{F}$ by the number of correspondences for which $d_\perp < t$ pixels.

  (d) If there are three real solutions for $\mathtt{F}$ the number of inliers is computed for each solution, and the solution with most inliers retained.

Choose the $\mathtt{F}$ with the largest number of inliers. In the case of ties choose the solution that has the lowest standard deviation of inliers.

(iv) **Non-linear estimation:** re-estimate $\mathtt{F}$ from all correspondences classified as inliers by minimizing a cost function, e.g. (11.6), using the Levenberg–Marquardt algorithm of section A6.2(p600).

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2$$

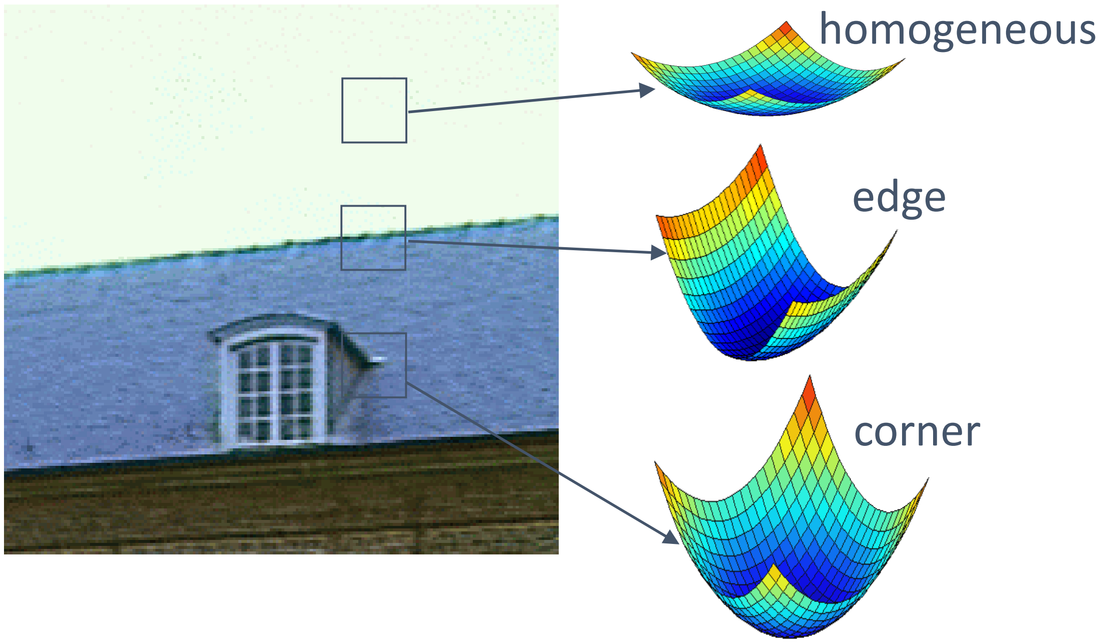
(v) **Guided matching:** Further interest point correspondences are now determined using the estimated $\mathtt{F}$ to define a search strip about the epipolar line.

The last two steps can be iterated until the number of correspondences is stable.

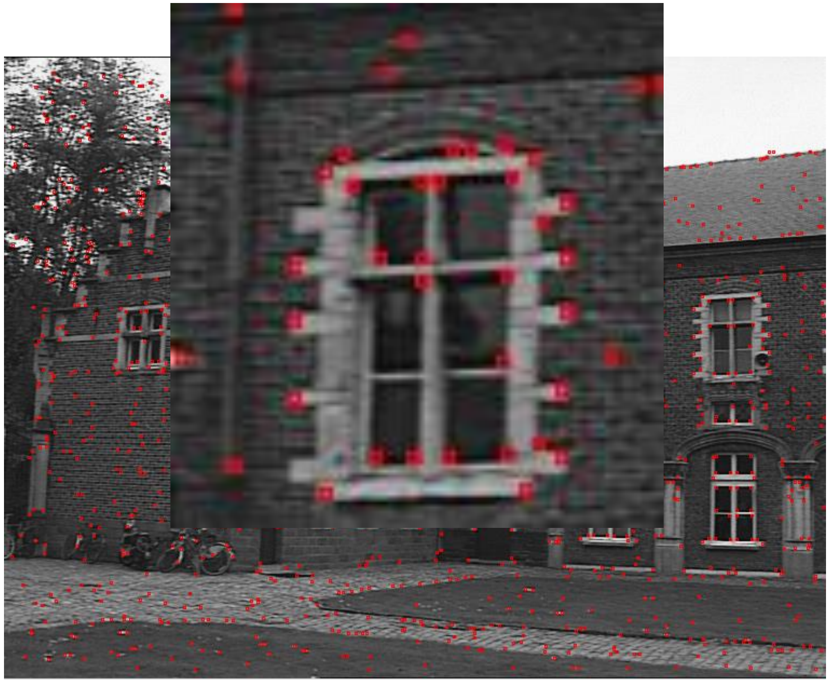
Interest Points

(e.g.Harris&Stephens´88; Shi&Tomasi´94)

Find points that differ as much as possible
from all neighboring points



Interest Points

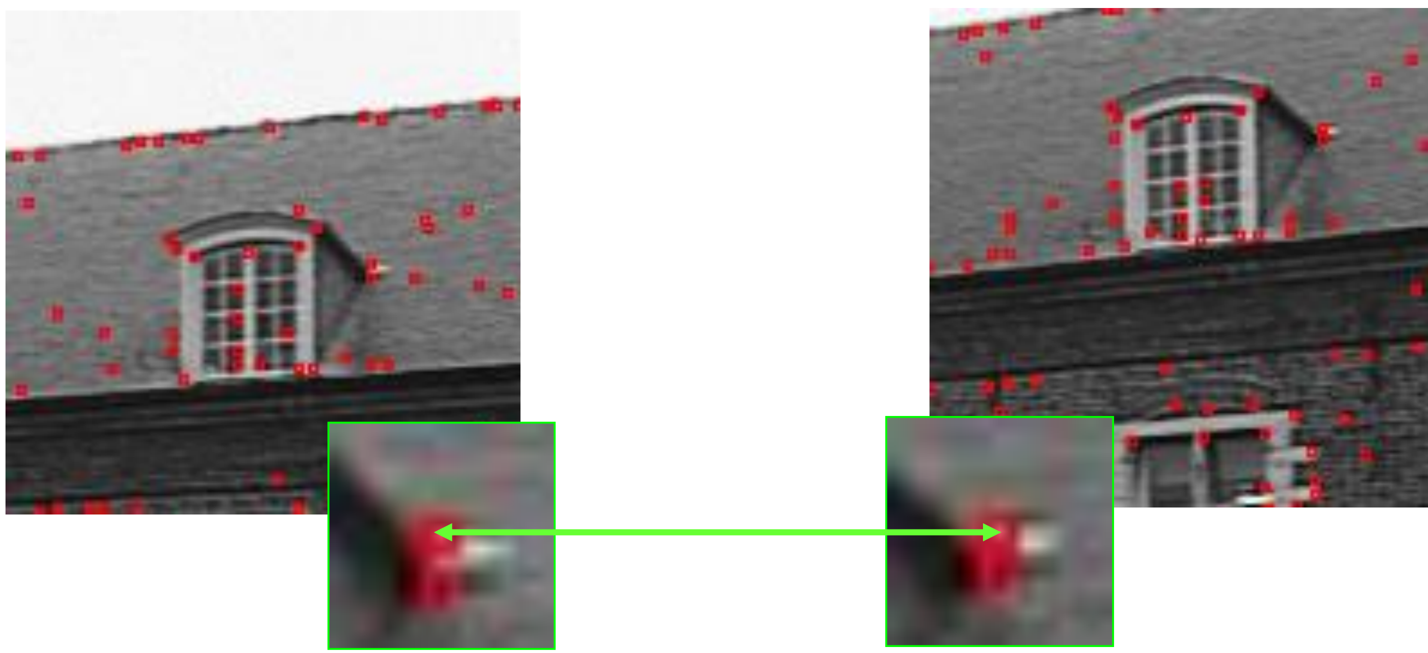


Select strongest features **(e.g. 1000/image)**

Interest Points

Evaluate NCC for all features with similar coordinates

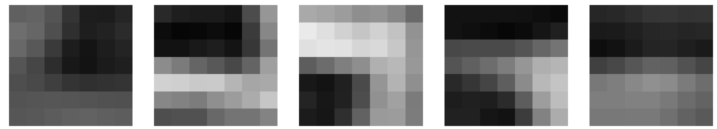
$$\text{e.g.}\ (x?\ y') \in \left[x - \tfrac{w}{10}, x + \tfrac{w}{10}\right] \times \left[y - \tfrac{h}{10}, y + \tfrac{h}{10}\right]$$

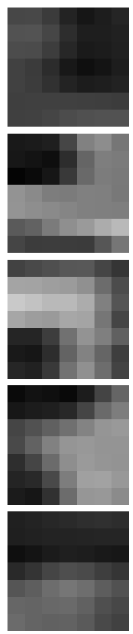


Keep mutual best matches

Still many wrong matches!

Interest Points





0.96	-0.40	-0.16	-0.39	0.19
-0.05	**0.75**	-0.47	0.51	0.72
-0.18	-0.39	**0.73**	0.15	-0.75
-0.27	0.49	0.16	**0.79**	0.21
0.08	0.50	-0.45	0.28	**0.99**

Gives satisfying results
for small image motions

RANSAC

Step 1. Extract features

Step 2. Compute a set of potential matches

Step 3. do

Step 3.1 select minimal sample (i.e. 7 matches)

Step 3.2 compute solution(s) for F

(generate hypothesis)

Step 3.3 determine inliers (verify hypothesis)

until $\Gamma(\#inliers,\#samples)<95\%$

Step 4. Compute F based on all inliers

Step 5. Look for additional matches by guided matching

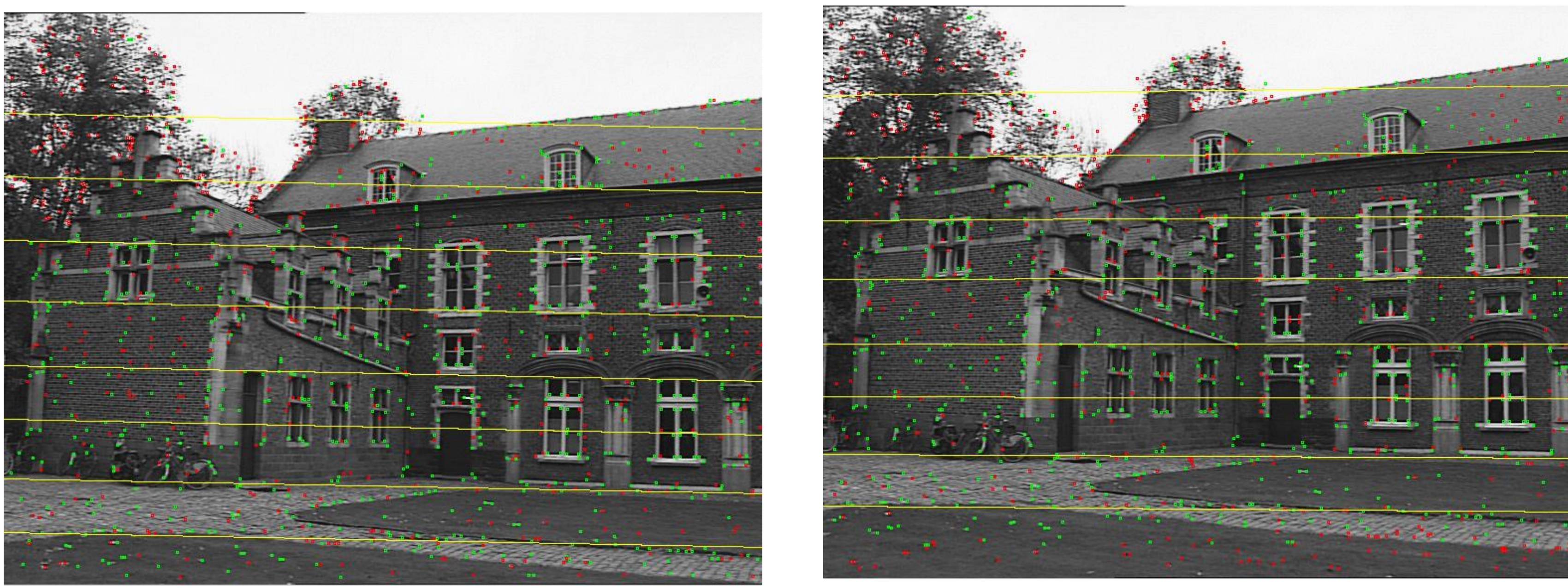
Step 6. Refine F based on all correct matches

RANSAC

- Why choose 7-point algorithm instead of 8-point algorithm?
  - A rank 2 matrix is produced without enforcement
  - The number of samples that must be tried in order to ensure a high probability of the no outliers is exponential in the size of the sample set
- Distance measure
  - Reprojection error $$\sum_i d\left(\mathrm{x}_i, \hat{\mathrm{x}}_i\right)^2 + d\left(\mathrm{x}'_i, \hat{\mathrm{x}}'_i\right)^2$$
  - Sampson approximation $$\sum \frac{\mathrm{x}'^{\mathrm{T}}\mathrm{F}\mathrm{x}}{\left(\mathrm{x}'^{\mathrm{T}}\mathrm{F}\right)_1^2 + \left(\mathrm{x}'^{\mathrm{T}}\mathrm{F}\right)_2^2 + \left(\mathrm{F}\mathrm{x}\right)_1^2 + \left(\mathrm{F}\mathrm{x}\right)_2^2}$$

Guided Matching

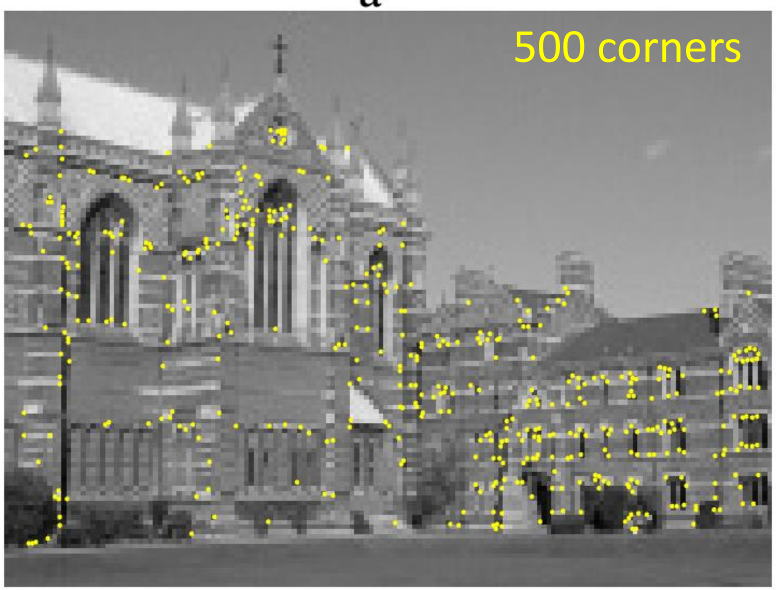
restrict search range to neighborhood of epipolar line
 (±1.5 pixels)
relax disparity restriction (along epipolar line)



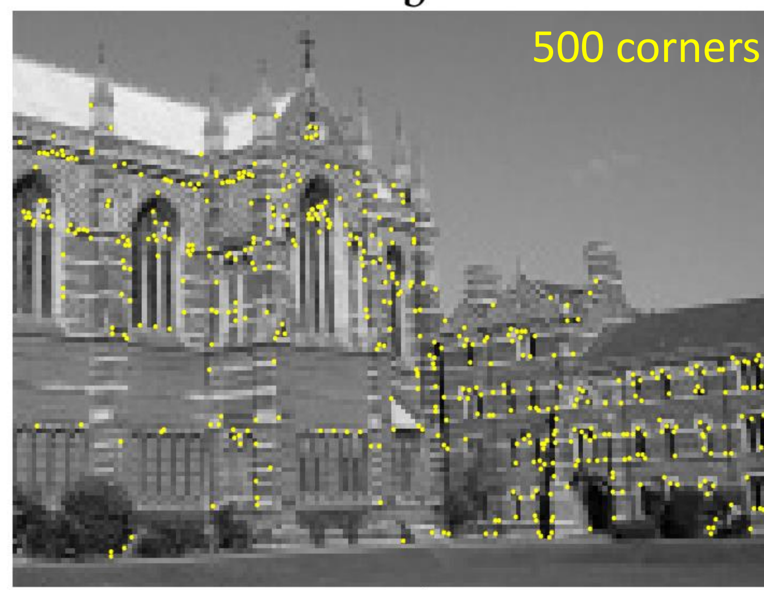
geometric relations between two views is fully described by recovered 3x3 matrix F

a

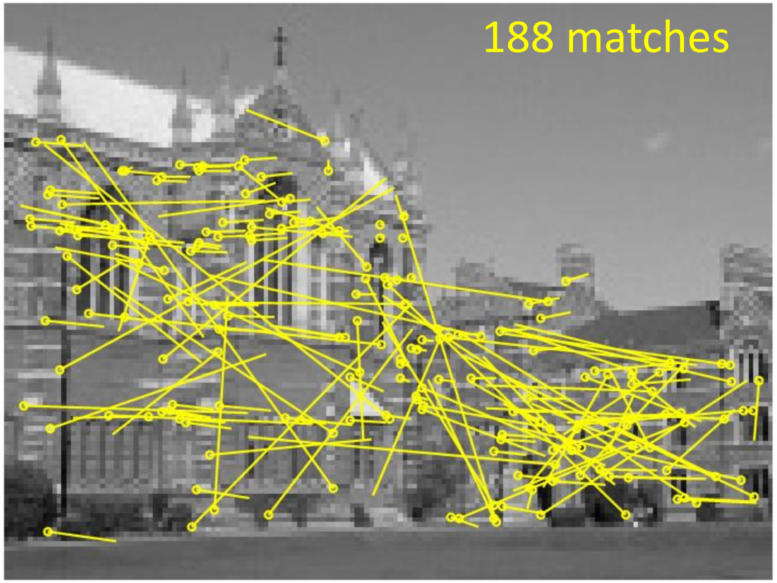
b



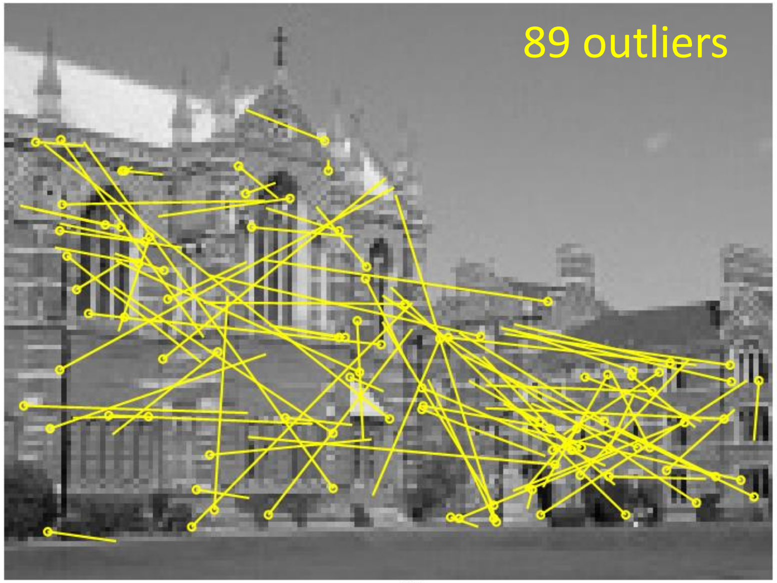
c



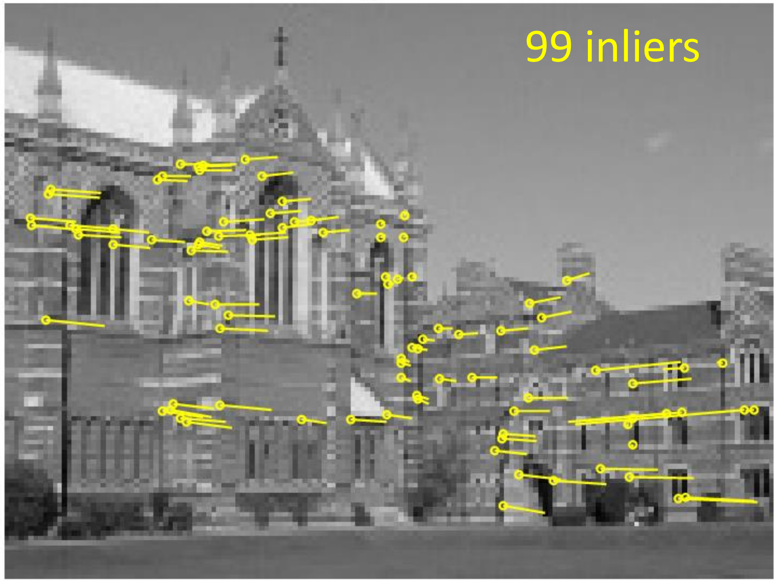
d


188 matches

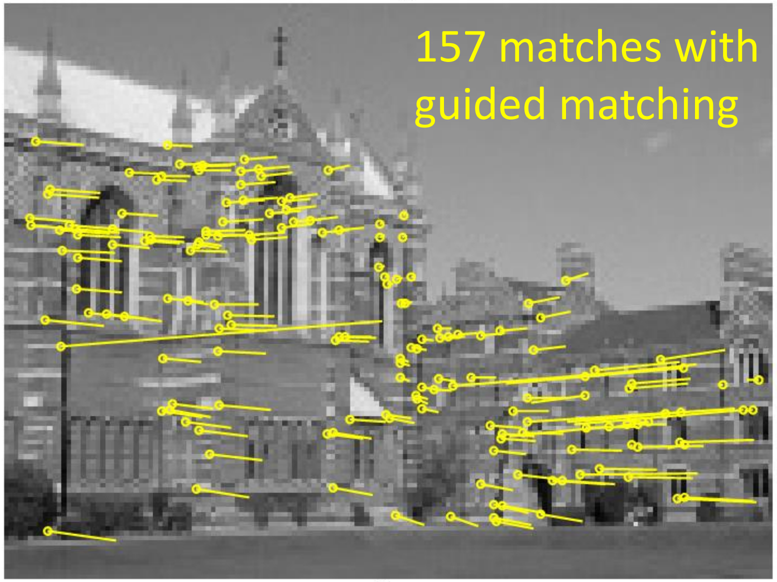
e


89 outliers

f


99 inliers

g


157 matches with
guided matching

h

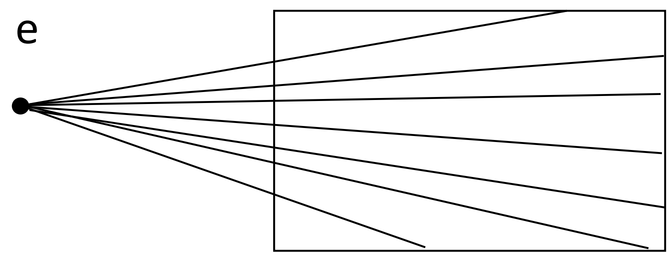
Degenerate Cases

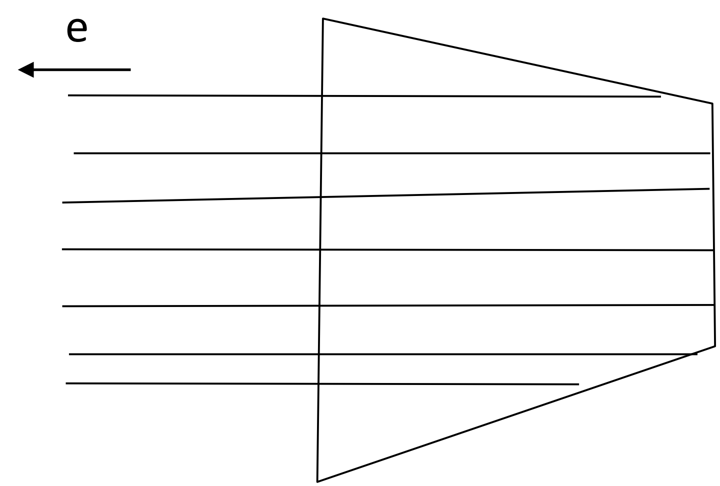
- Degenerate cases
  - Planar scene
  - Pure rotation
- No unique solution
  - Remaining DOF filled by noise
  - Use simpler model (e.g. homography)
- Model selection (Torr et al., ICCV´98, Kanatani, Akaike)
  - Compare H and F according to expected residual error (compensate for model complexity)

Image Pair Rectification

simplify stereo matching
by warping the images

Apply projective transformation so that epipolar lines
correspond to horizontal scanlines





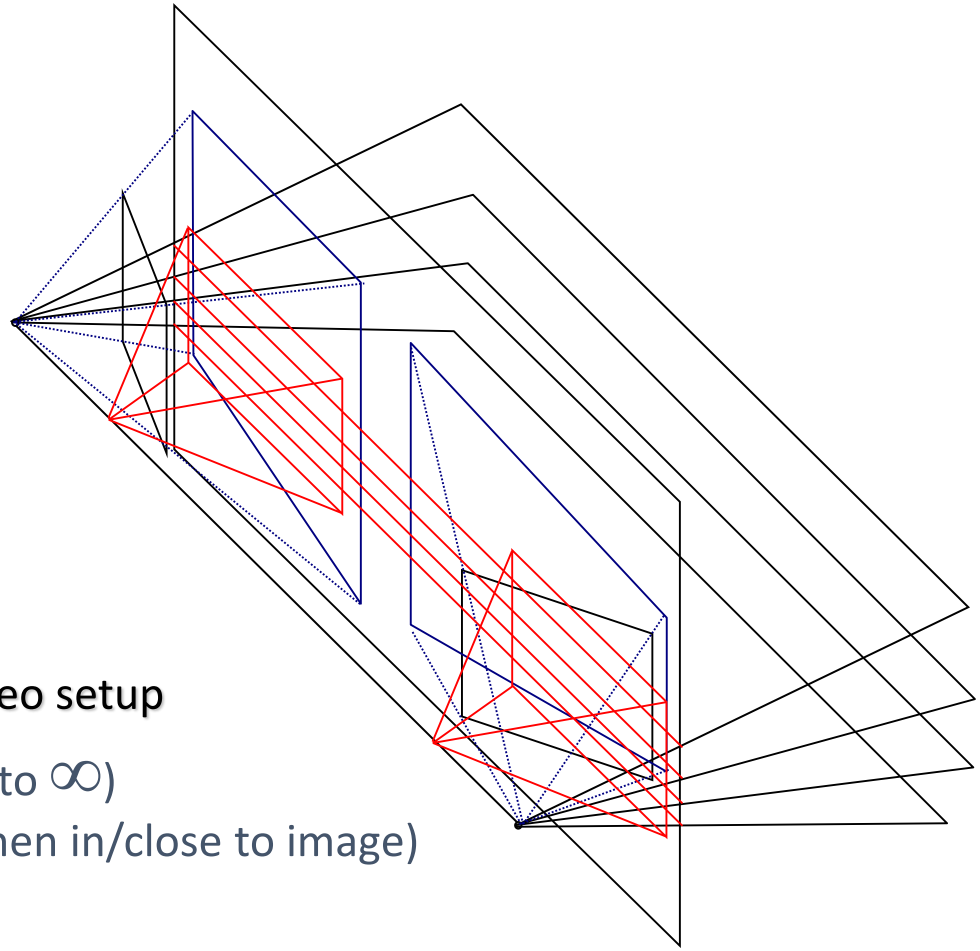
map epipole e to (1,0,0)

try to minimize image distortion

problem when epipole in (or close to) the image

Planar Rectification

(standard approach)



Bring two views
to standard stereo setup

(moves epipole to ∞)
(not possible when in/close to image)

Rectification

- Two steps:
  - Mapping the epipolar to infinity
    - Finding a projective transformation H of an image that maps the epipole to a point at infinity
    - Avoid distortion: better to have rigid transformation, to first-order the neighborhood of x_0 may undergo rotation and translation only
  - Matching transformation
    - Match the epiplolar lines
    - Find a match pair that

$$H^{-T}l = H'^{-T}l'$$

Mapping the Epipolar to Infinity

- For example, given epipole $e=(f, 0, 1)^T$
- A good transform is

$$\mathtt{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1/f & 0 & 1 \end{bmatrix}$$

$$(\hat{x}, \hat{y}, 1)^\mathsf{T} = (x, y, 1 - x/f)^\mathsf{T} = (x(1 + x/f + \ldots), y(1 + x/f + \ldots), 1)^\mathsf{T}$$

$$\frac{\partial(\hat{x}, \hat{y})}{\partial(x, y)} = \begin{bmatrix} 1 + 2x/f & 0 \\ y/f & 1 + x/f \end{bmatrix}$$

=I if x=y=0

- For an arbitrary x_0 and epipole **e**
  - H=GRT: R: rotate to x-axis, T: translate to $(f, 0, 1)^T$

Matching Transformation

- Target: to minimize $\sum_i d(\mathtt{H}\mathbf{x}_i, \mathtt{H}'\mathbf{x}'_i)^2$

Corollary 11.4. *Let J and J' be images with fundamental matrix $\mathtt{F} = [\mathbf{e}']_\times \mathtt{M}$, and let $\mathtt{H}'$ be a projective transformation of J' mapping the epipole $\mathbf{e}'$ to the infinite point $(1, 0, 0)^\mathsf{T}$. A transformation $\mathtt{H}$ of J matches $\mathtt{H}'$ if and only if $\mathtt{H}$ is of the form $\mathtt{H} = \mathtt{H}_\mathrm{A}\mathtt{H}_0$, where $\mathtt{H}_0 = \mathtt{H}'\mathtt{M}$ and $\mathtt{H}_\mathrm{A}$ is an affine transformation of the form (11.20).*

$$\mathtt{H}_\mathrm{A} = \begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- To solve a, b, c, minimize

$$\sum_i (a\hat{x}_i + b\hat{y}_i + c - \hat{x}'_i)^2 + (\hat{y}_i - \hat{y}'_i)^2$$

Algorithm Outline

(i) Identify a seed set of image-to-image matches $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ between the two images. Seven points at least are needed, though more are preferable. It is possible to find such matches by automatic means.

(ii) Compute the fundamental matrix $\mathtt{F}$ and find the epipoles $\mathbf{e}$ and $\mathbf{e}'$ in the two images.

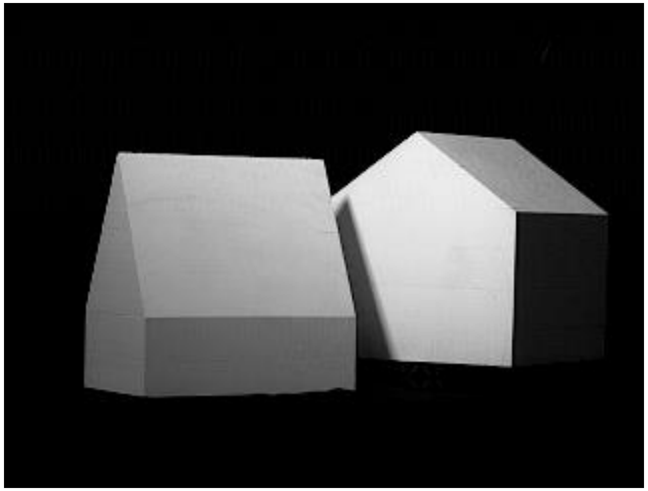
(iii) Select a projective transformation $\mathtt{H}'$ that maps the epipole $\mathbf{e}'$ to the point at infinity, $(1, 0, 0)^\mathsf{T}$. The method of section 11.12.1 gives good results.

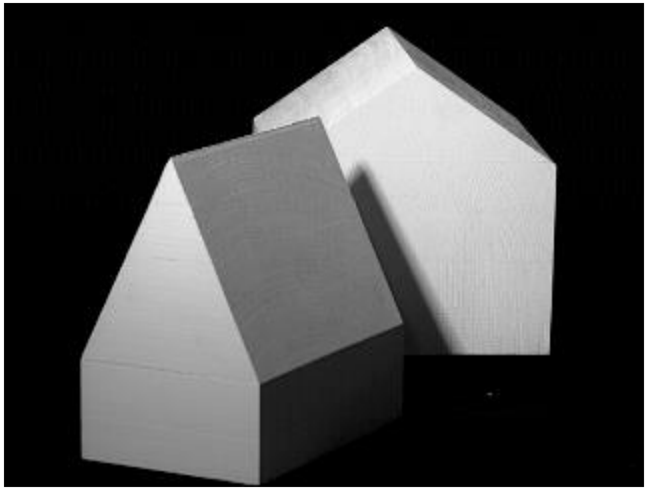
(iv) Find the matching projective transformation $\mathtt{H}$ that minimizes the least-squares distance

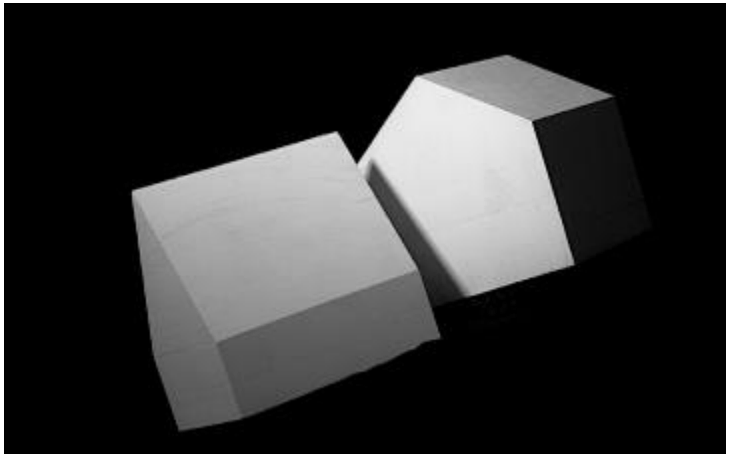
$$\sum_i d(\mathtt{H}\mathbf{x}_i, \mathtt{H}'\mathbf{x}'_i). \tag{11.22}$$

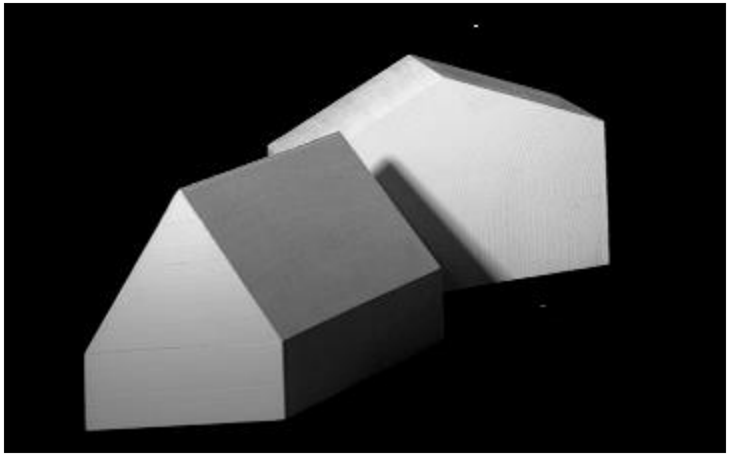
The method used is a linear method described in section 11.12.2.

(v) Resample the first image according to the projective transformation $\mathtt{H}$ and the second image according to the projective transformation $\mathtt{H}'$.












Polar Rectification

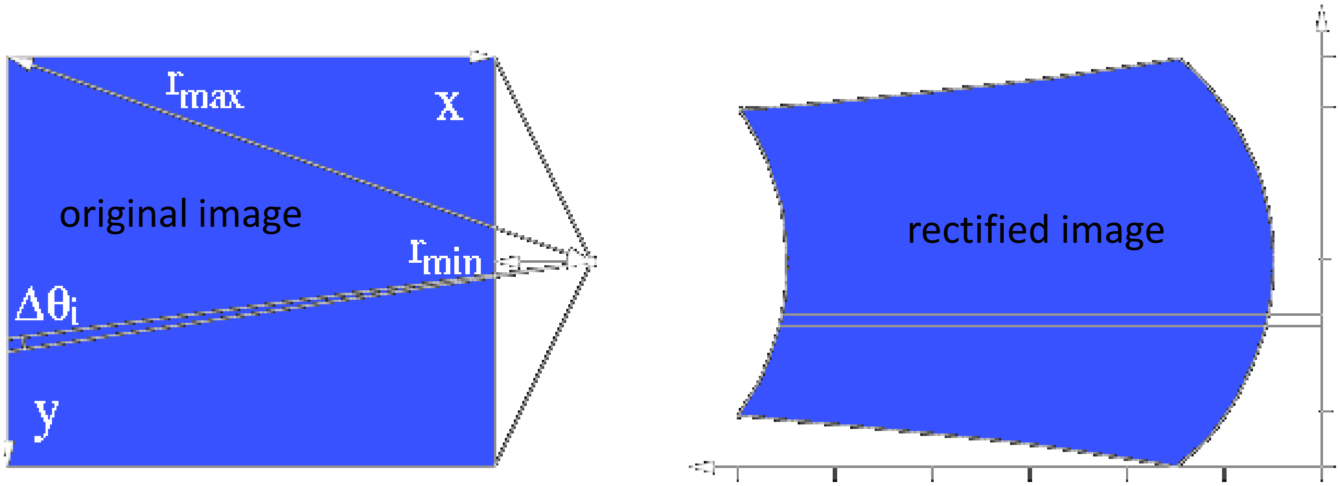
(Pollefeys et al. ICCV'99)

Polar re-parameterization around epipoles

Requires only (oriented) epipolar geometry

Preserve length of epipolar lines

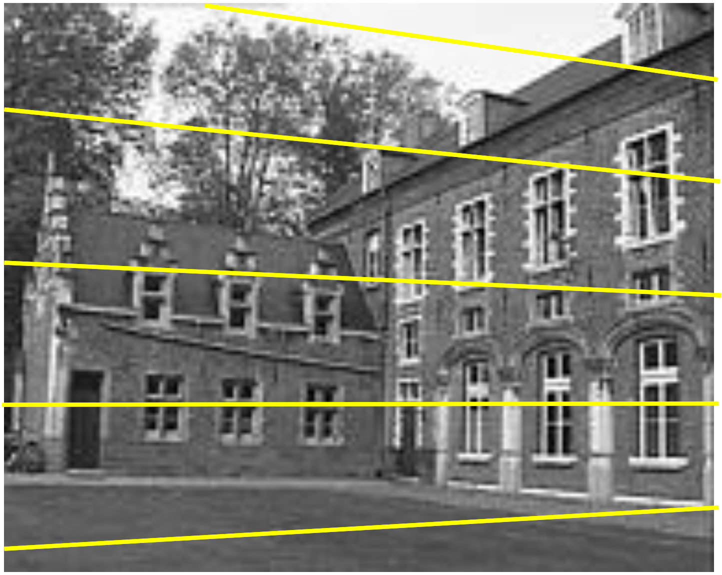
Choose $\Delta\theta$ so that no pixels are compressed

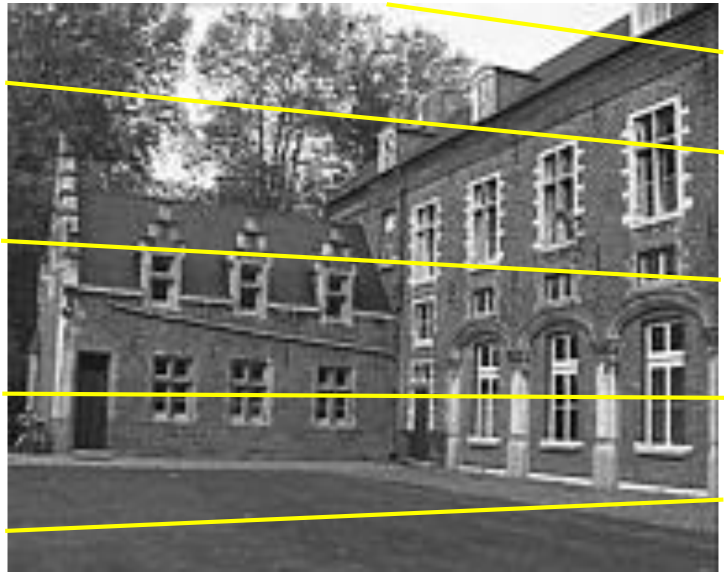


Works for all relative motions

Guarantees minimal image size

Polar Rectification: Example





Polar Rectification: Example