

3D Computer Vision II

Two View Geometry

Part 2 – Fundamental Matrix Computation

Nassir Navab

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

November 19, 2009

Outline – Two-View Geometry

- Epipolar Geometry
- Fundamental Matrix Computation
- 3D Reconstruction



Reminder – Three Questions

- (i) **Correspondence geometry:** Given an image point x in the first image, how does this constrain the position of the corresponding point x' in the second image?
- (ii) **Camera geometry (motion):** Given a set of corresponding image points $\{x_i \leftrightarrow x'_i\}$, $i=1, \dots, n$, what are the cameras P and P' for the two views?
- (iii) **Scene geometry (structure):** Given corresponding image points $x_i \leftrightarrow x'_i$ and cameras P, P' , what is the position of (their pre-image) X in space?

The Fundamental Matrix F

Algebraic representation of Epipolar Geometry:

$$\mathbf{X} \mapsto \mathbf{l}'$$

We will see that mapping is (singular) correlation (i.e. projective mapping from points to lines) represented by the fundamental matrix F.

Points From Lines and Vice-versa

- Intersections of lines

The intersection of two lines l and l' is $x = l \times l' = [l]_x l' = -[l']_x l$

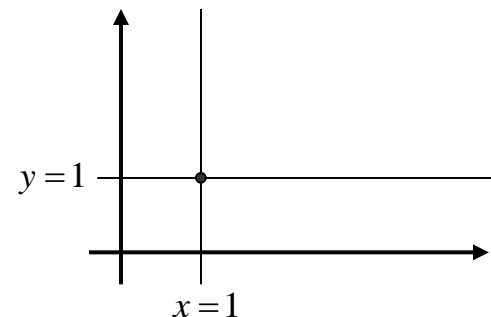
- Line joining two points

The line through two points x and x' is $l = x \times x' = [x]_x x' = -[x']_x x$

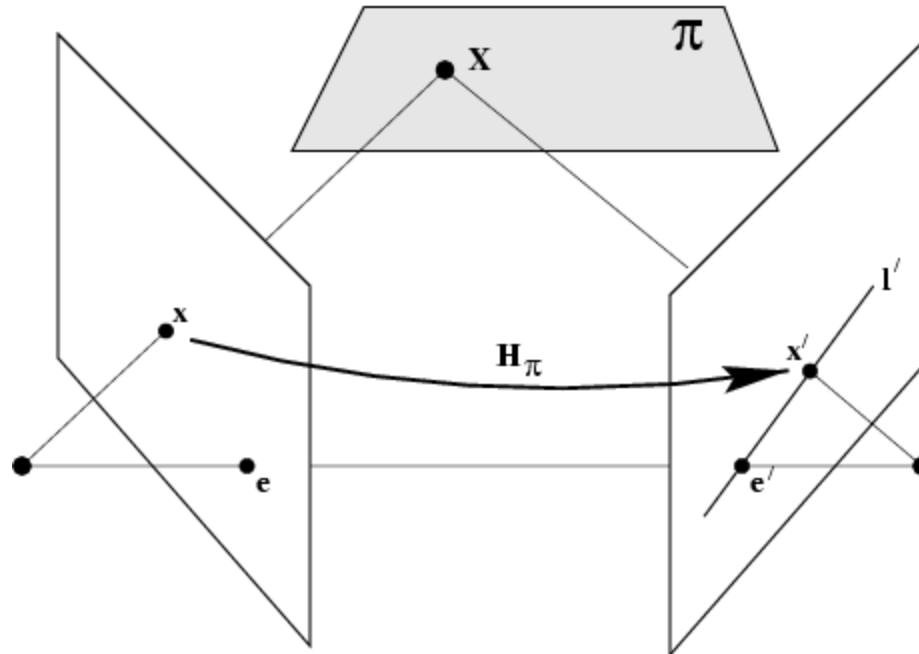
- Anti-symmetric Matrix

$$[l]_x = \begin{bmatrix} 0 & -l_3 & l_2 \\ l_3 & 0 & -l_1 \\ -l_2 & l_1 & 0 \end{bmatrix}$$

Example



The Fundamental Matrix F – Geometric Derivation



$$x' = H_{\pi} x$$

$$l' = e' \times x' = [e']_{\times} H_{\pi} x = Fx$$

mapping from 2-D to 1-D family (rank 2)

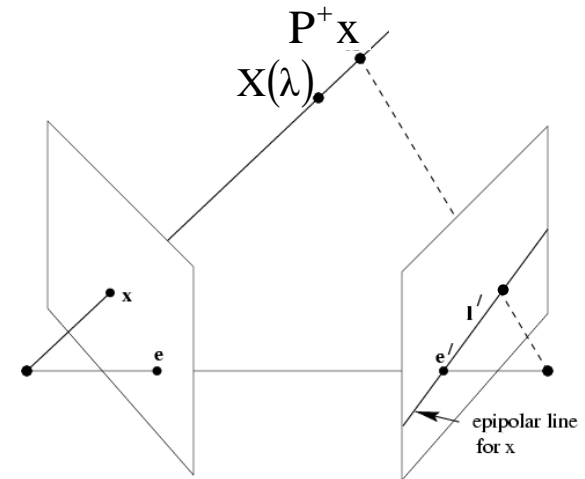
The Fundamental Matrix F

Algebraic derivation:

$$X(\lambda) = (1 - \lambda)C + \lambda P^+ x \quad (P^+ P = I)$$

$$1 = P' C \times P' P^+ x$$

$$F = [e']_x P' P^+$$



(note: doesn't work for $C=C' \Rightarrow F=0$)

The Fundamental Matrix F

Correspondence condition:

The fundamental matrix satisfies the condition that for any pair of corresponding points $x \leftrightarrow x'$ in the two images:

$$x'^T F x = 0 \quad (x'^T l' = 0)$$

The Fundamental Matrix F

F is the unique 3×3 rank 2 matrix that satisfies $x'^T F x = 0$ for all $x \leftrightarrow x'$.

- (i) **Transpose:** if F is fundamental matrix for (P, P') , then F^T is fundamental matrix for (P', P)
- (ii) **Epipolar lines:** $l' = Fx$ & $l = F^T x'$
- (iii) **Epipoles:** on all epipolar lines, thus $e'^T F x = 0, \forall x \Rightarrow e'^T F = 0$, similarly $F e = 0$
- (iv) F has 7 d.o.f. , i.e. $3 \times 3 - 1$ (homogeneous) - 1 (rank 2)
- (v) F is a correlation, projective mapping from a point x to a line $l' = Fx$ (not a proper correlation, i.e. not invertible)

Epipolar Geometry

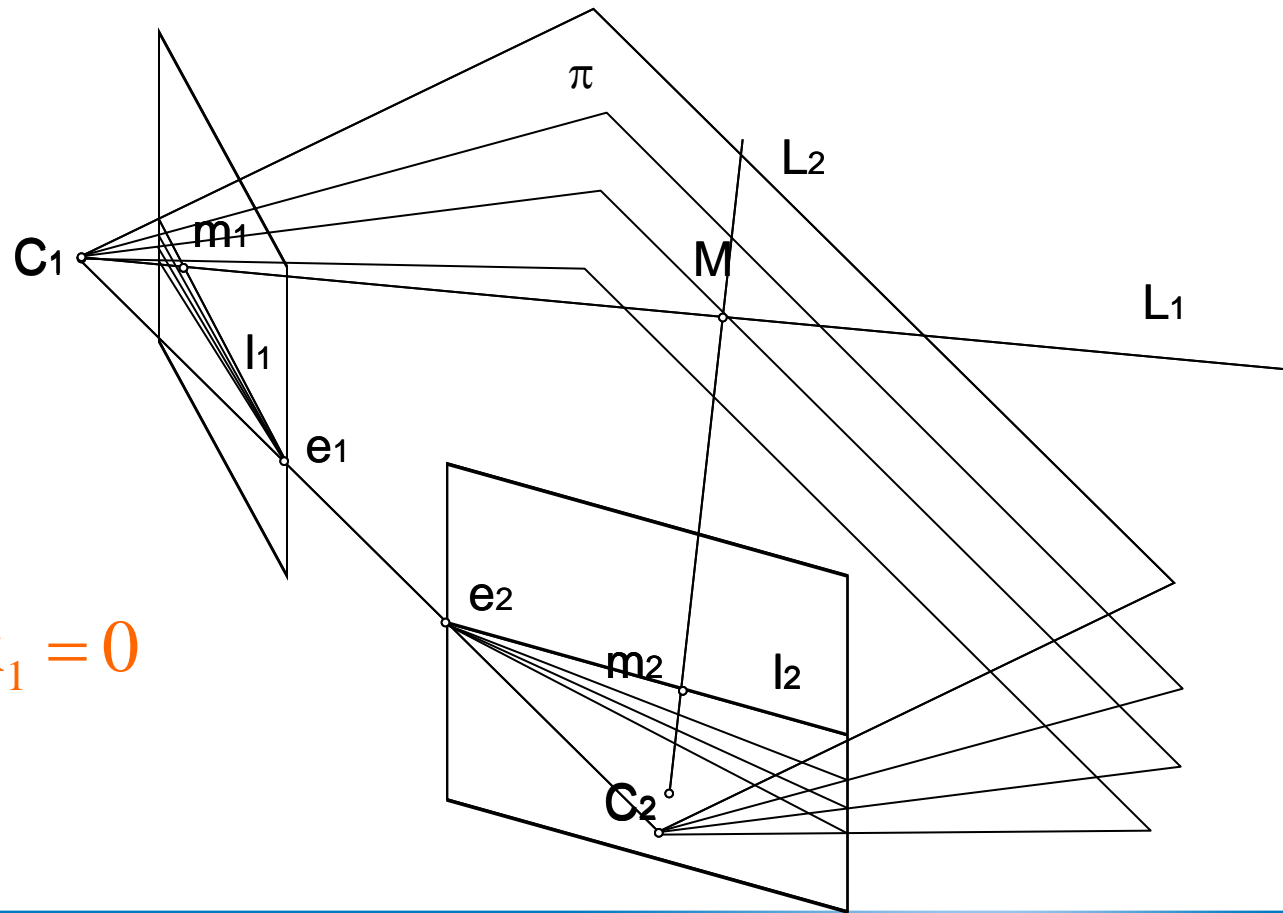
$$l_1 = [e_1]_x x_1$$

$$l_2 = P_2^{+T} P_1^T l_1$$

$$\pi = P_1^T l_1$$

$$x_2^T l_2 = 0$$

$$x_2^T P_2^{+T} P_1^T [e_1]_x x_1 = 0$$



Epipolar Geometry

Canonical representation:

$$P = [I | 0] \quad P' = [[e']_{\times} F + e' v^T | \lambda e']$$

1. Computable from corresponding points
2. Simplifies matching
3. Allows to detect wrong matches
4. Related to calibration

Epipolar Geometry – Basic Equations

$$\mathbf{x}'^T \mathbf{F} \mathbf{X} = 0$$

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

... separate knowns from unknowns:

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

(data)

(unknowns)

(linear)

Epipolar Geometry – Basic Equations

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

$$A\mathbf{f} = \mathbf{0}$$

The Singularity Constraint

$$e'^T F = 0$$

$$Fe = 0$$

$$\det F = 0$$

$$\text{rank } F = 2$$

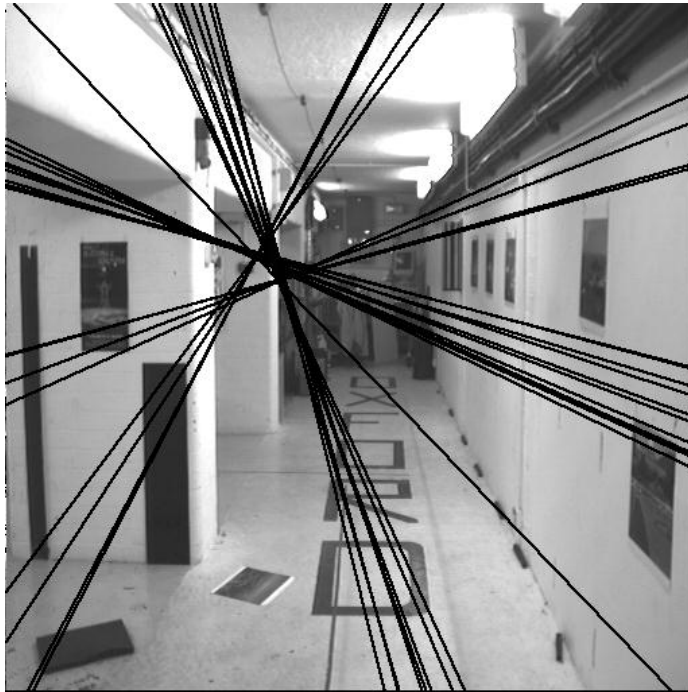
SVD from linearly computed F matrix (rank 3)

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

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

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

The Singularity Constraint



Non singular \mathbf{F}



Forcing singularity using the SVD method

The Minimum Case – 7 Point Correspondences

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

$$\mathbf{A} = \mathbf{U}_{7 \times 9} \text{diag}(\sigma_1, \dots, \sigma_7, 0, 0) \mathbf{V}_{9 \times 9}^T$$

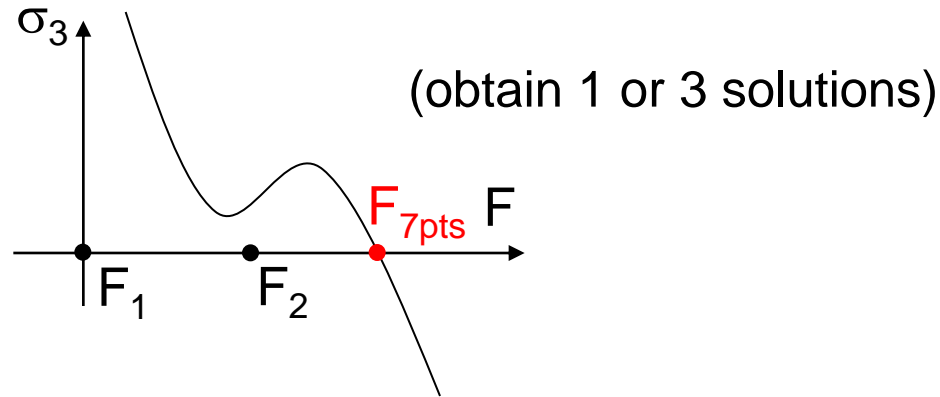
$$\Rightarrow \mathbf{A}[\mathbf{V}_8 \mathbf{V}_9] = \mathbf{0}_{9 \times 2}$$

$$\mathbf{x}'_i{}^T (\mathbf{F}_1 + \lambda \mathbf{F}_2) \mathbf{x}_i = 0, \forall i = 1 \dots 7$$

One parameter family of solutions

But $\mathbf{F}_1 + \lambda \mathbf{F}_2$ not automatically rank 2

The Minimum Case – Impose Rank 2



$$\det(F_1 + \lambda F_2) = a_3 \lambda^3 + a_2 \lambda^2 + a_1 \lambda + a_0 = 0 \quad (\text{cubic equation})$$

One or three solutions (only real solutions are acceptable)


The 8-Point Algorithm

$$\begin{bmatrix}
 x_1 x_1' & y_1 x_1' & x_1' & x_1 y_1' & y_1 y_1' & y_1' & x_1 & y_1 & 1 \\
 x_2 x_2' & y_2 x_2' & x_2' & x_2 y_2' & y_2 y_2' & y_2' & x_2 & y_2 & 1 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 x_n x_n' & y_n x_n' & x_n' & x_n y_n' & y_n y_n' & y_n' & x_n & y_n & 1
 \end{bmatrix}
 \begin{bmatrix}
 f_{11} \\
 f_{12} \\
 f_{13} \\
 f_{21} \\
 f_{22} \\
 f_{23} \\
 f_{31} \\
 f_{32} \\
 f_{33}
 \end{bmatrix}
 = 0$$

The **Un**normalized 8-Point Algorithm

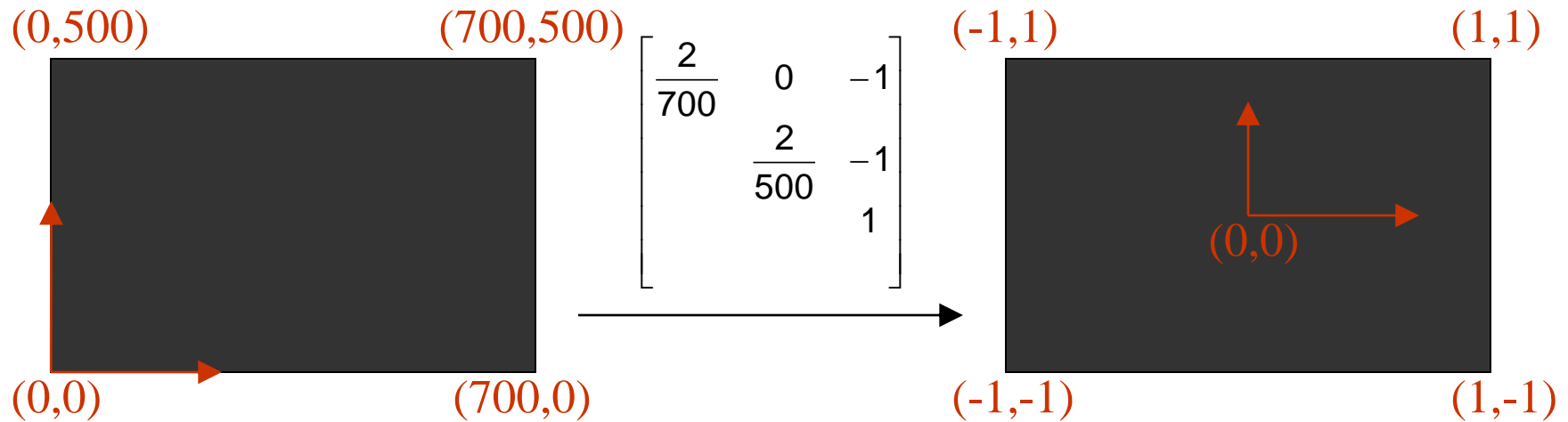
$$\begin{bmatrix}
 x_1 x_1' & y_1 x_1' & x_1' & x_1 y_1' & y_1 y_1' & y_1' & x_1 & y_1 & 1 \\
 x_2 x_2' & y_2 x_2' & x_2' & x_2 y_2' & y_2 y_2' & y_2' & x_2 & y_2 & 1 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 x_n x_n' & y_n x_n' & x_n' & x_n y_n' & y_n y_n' & y_n' & x_n & y_n & 1
 \end{bmatrix}
 \begin{bmatrix}
 f_{11} \\
 f_{12} \\
 f_{13} \\
 f_{21} \\
 f_{22} \\
 f_{23} \\
 f_{31} \\
 f_{32} \\
 f_{33}
 \end{bmatrix}
 = 0$$

~ 10000 ~ 10000 ~ 100 ~ 10000 ~ 10000 ~ 100 ~ 100 ~ 100 1


Orders of magnitude difference between columns of data matrix
 → least-squares yields poor results

The Normalized 8-Point Algorithm

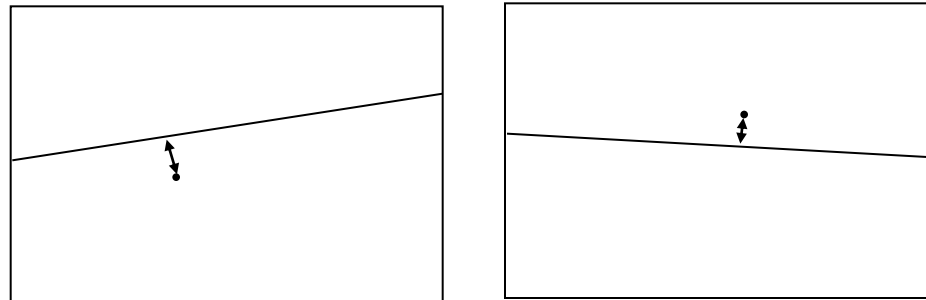
Transform image to $\sim[-1,1] \times [-1,1]$



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

Symmetric Epipolar Error

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



Gold Standard

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

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

Initialize: normalized 8-point, $(\mathbf{P}, \mathbf{P}')$ from \mathbf{F} , reconstruct \mathbf{X}_i

Parameterize:

$$\mathbf{P} = [\mathbf{I} \mid \mathbf{0}], \mathbf{P}' = [\mathbf{M} \mid \mathbf{t}], \mathbf{X}_i \quad (\text{overparametrized})$$

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

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

Gold Standard

Alternative, minimal parametrization (with $a=1$)

$$\mathbf{F} = \begin{bmatrix} a & b & -ax - by \\ c & d & -cx - dy \\ -ax' - cy' & -bx' - dy' & F_{33} \end{bmatrix}$$

where $F_{33} = (ax + by)x' + (cx + dy)y'$ (note $(x,y,1)$ and $(x',y',1)$ are epipoles)

problems:

- $a=0$ → pick largest of a,b,c,d to fix
- epipole at infinity → pick largest of x,y,w and of x',y',w'

$4 \times 3 \times 3 = 36$ parametrizations!

reparametrize at every iteration, to be sure

First-order Geometric Error (Sampson Error)

$$\sum e^T (JJ^T)^{-1} e \quad \sum \frac{e^T e}{JJ^T} \quad (\text{one eq./point} \Rightarrow JJ^T \text{ scalar})$$

$$e = \sum x'^T Fx = 0 \quad \frac{\partial e}{\partial x_i} = x'^T F \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$JJ^T = (x'^T F)_1^2 + (x'^T F)_2^2 + (Fx)_1^2 + (Fx)_2^2$$

$$\sum \frac{e^T e}{JJ^T} \quad \sum \frac{(x'^T Fx)^2}{(x'^T F)_1^2 + (x'^T F)_2^2 + (Fx)_1^2 + (Fx)_2^2}$$

(problem if some x is located at epipole)

advantage: no subsidiary variables required

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
(minimal parameterization, sparse implementation)

Special Case

Enforce constraints for optimal results:

- Pure translation (2dof):

$$F = [e']_x$$

- Planar motion (6dof),
- Calibrated case (5dof)

The Envelope of Epipolar Lines

What happens to an epipolar line if there is noise?



Monte Carlo



3D Computer Vision II - Two View Geometry

$n=10$

$n=15$

$n=25$

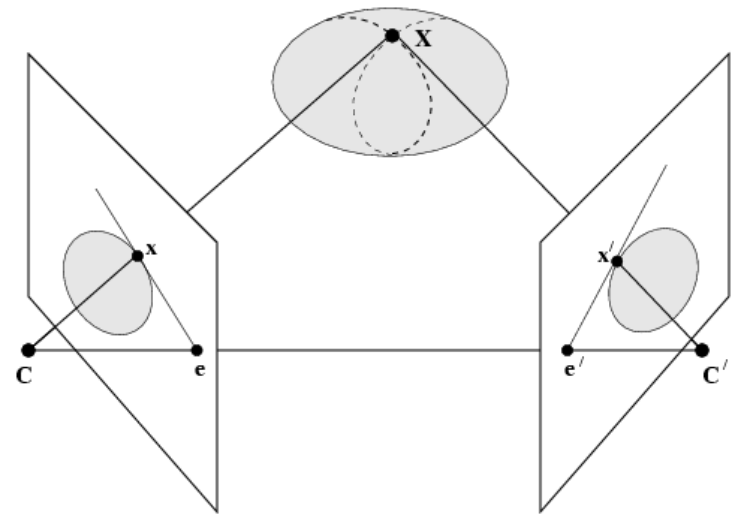
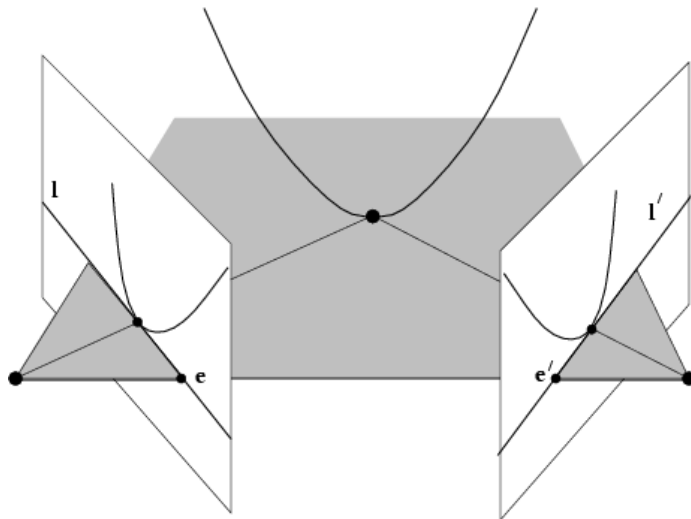
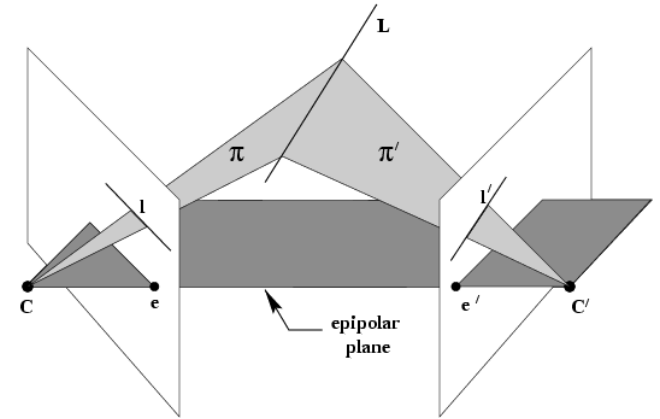
$n=50$

34

Other Entities

Lines give no constraint for two view geometry
(but will for three and more views)

Curves and surfaces yield some constraints related to tangency



Automatic Computation of F

- (i) Interest points
- (ii) Putative correspondences
- (iii) RANSAC
- (iv) Non-linear re-estimation of F
- (v) Guided matching

Repeat (iv) and (v) until stable

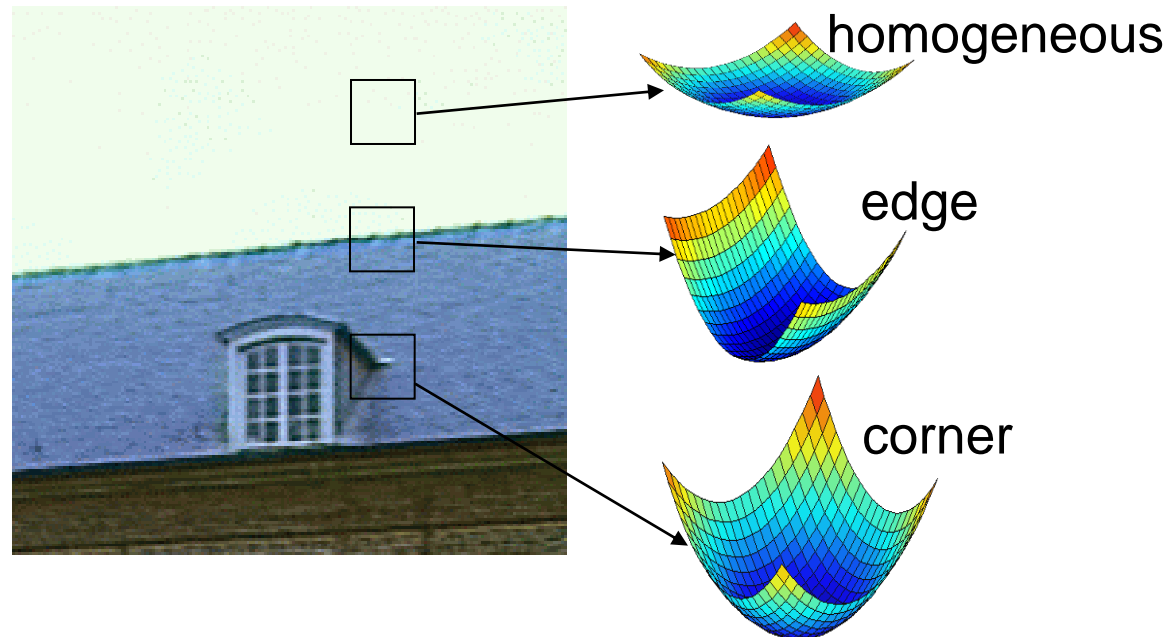
Feature Points

- Extract feature points to relate images
- Required properties:
 - Well-defined
(i.e. neighboring points should all be different)
 - Stable across views
(i.e. same 3D point should be extracted as feature for neighboring viewpoints)

Feature Points

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

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



Feature Points

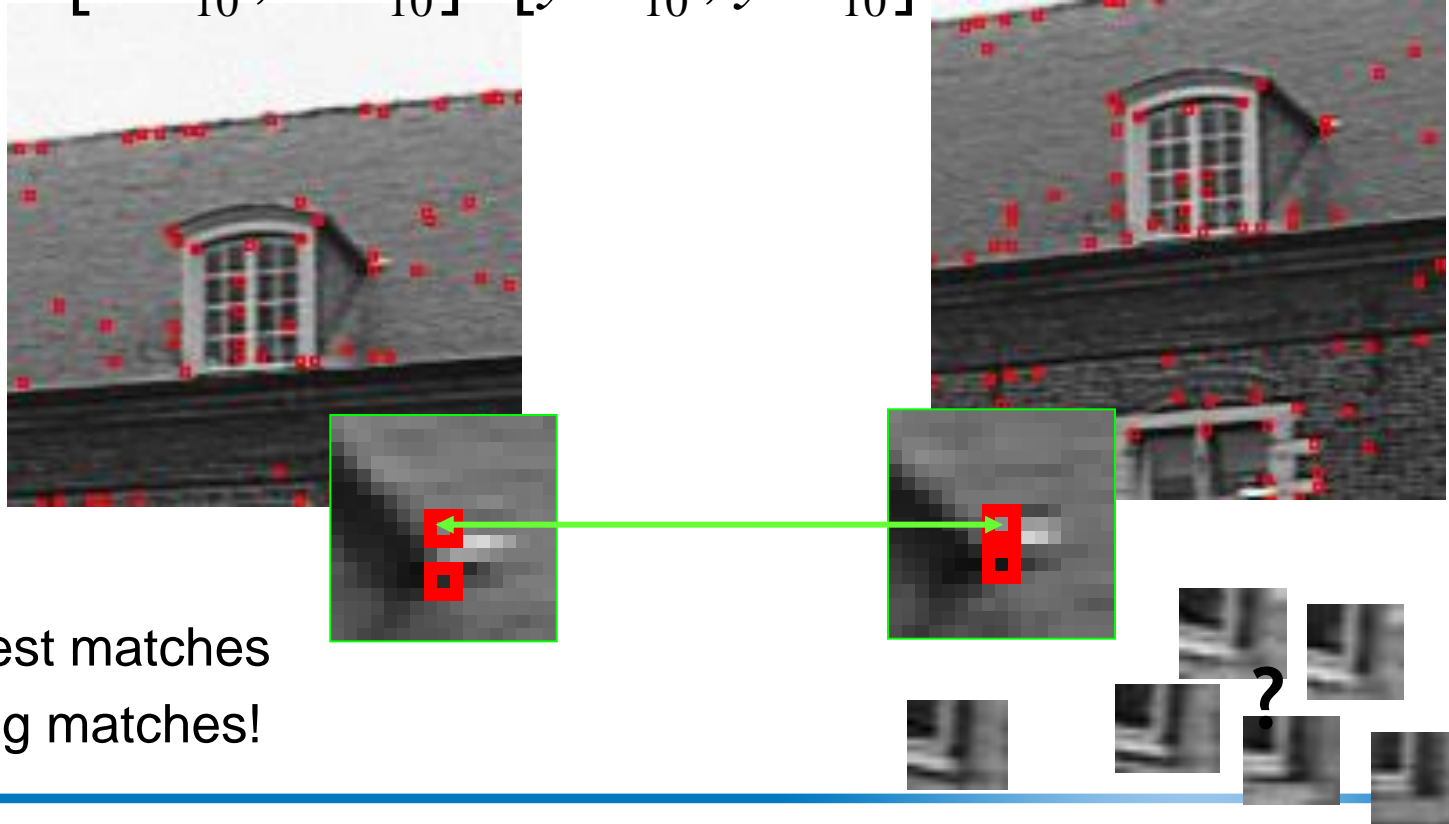


Select strongest features (**e.g. 1000/image**)

Feature Matching

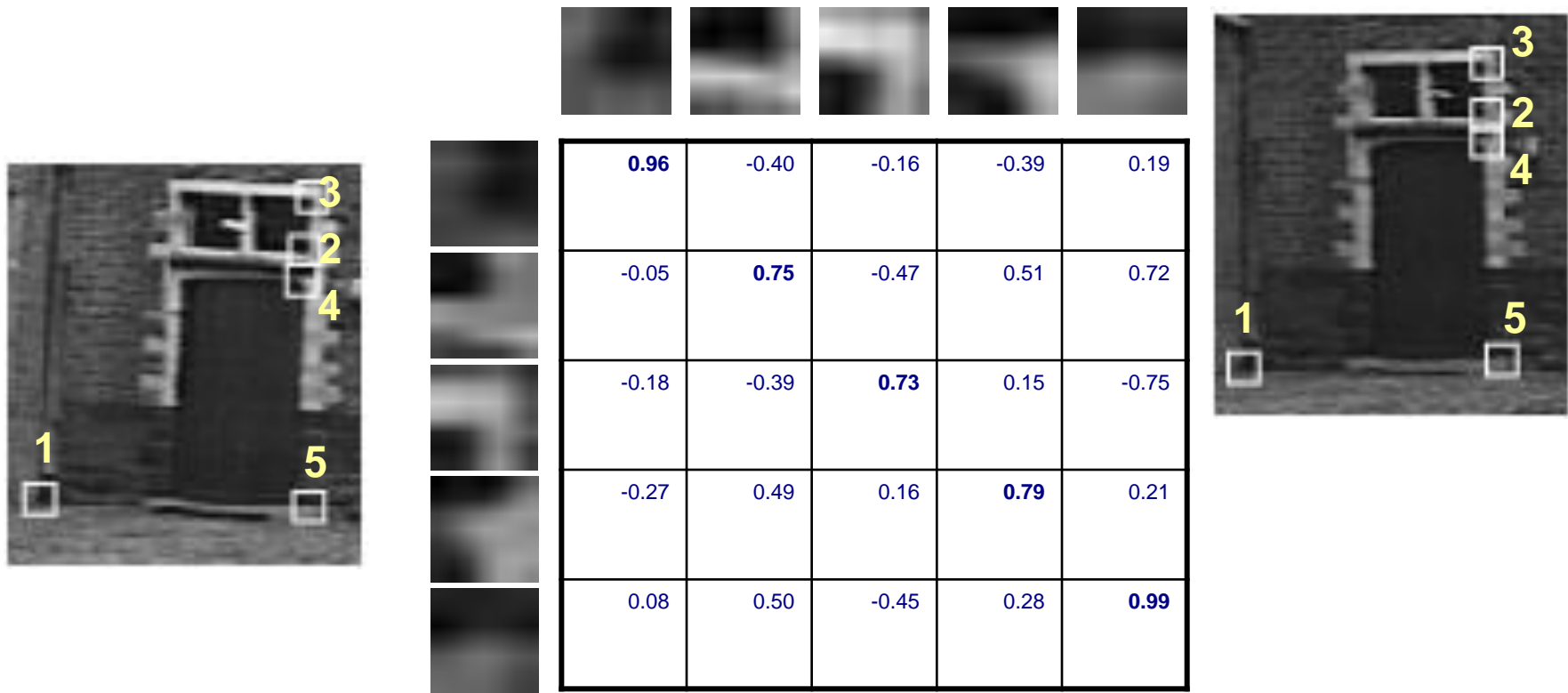
Evaluate NCC for all features with similar coordinates

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



Keep mutual best matches
Still many wrong matches!

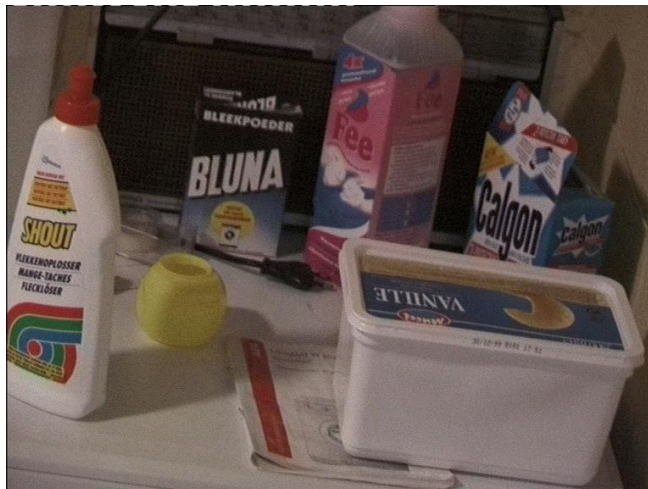
Feature Example



Gives satisfying results
for small image motions

Wide-baseline Matching

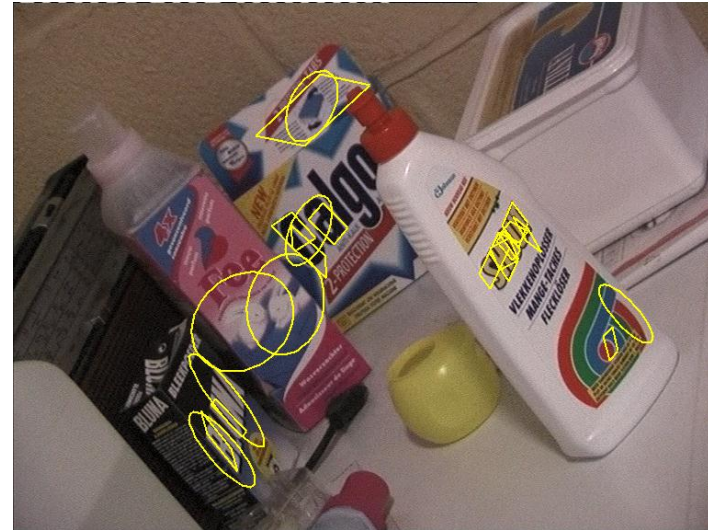
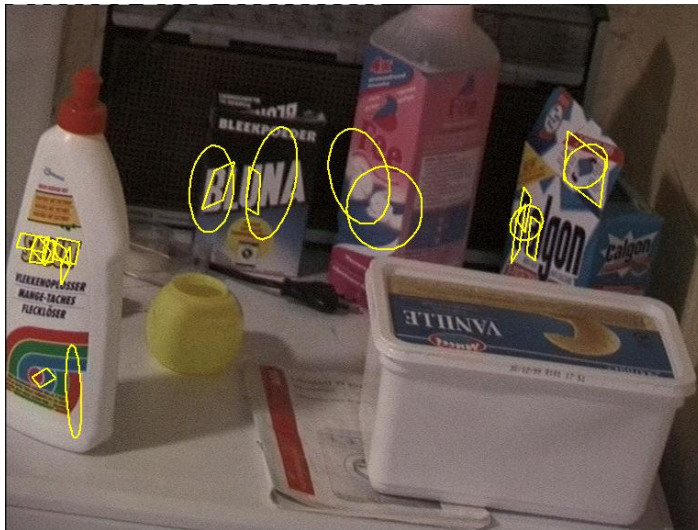
- Requirement to cope with larger variations between images
 - Translation, rotation, scaling
 - Foreshortening
 - Non-diffuse reflections
 - Illumination
- } geometric transformations
- } photometric changes



Wide-baseline Matching

(Tuytelaars and Van Gool BMVC 2000)

Wide baseline matching for two different region types



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

Step 3.3 determine inliers

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

} generate hypothesis
 }
 } verify hypothesis

Step 4. Compute F based on all inliers

Step 5. Look for additional matches

Step 6. Refine F based on all correct matches

$$\Gamma = 1 - \left(1 - \left(\frac{\#inliers}{\#matches}\right)^7\right)^{\#samples}$$

#inliers	90%	80%	70%	60%	50%
#samples	5	13	35	106	382

Finding More Matches



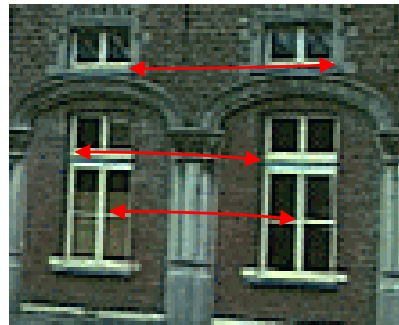
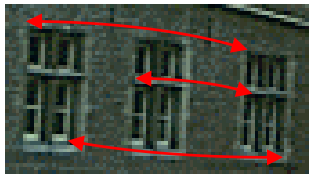
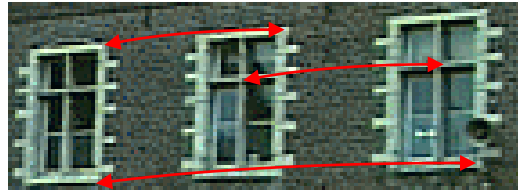
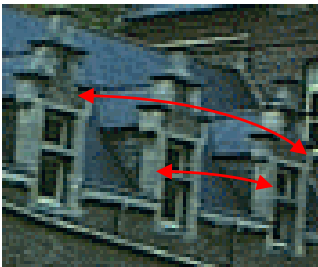
Restrict search range to neighborhood of epipolar line
(± 1.5 pixels)
Relax disparity restriction (along epipolar line)

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)

More Problems

- Absence of sufficient features (no texture)
- Repeated structure ambiguity



- Robust matcher also finds support for wrong hypothesis
- Solution: detect repetition
(Schaffalitzky and Zisserman, BMVC '98)

Two-view Geometry

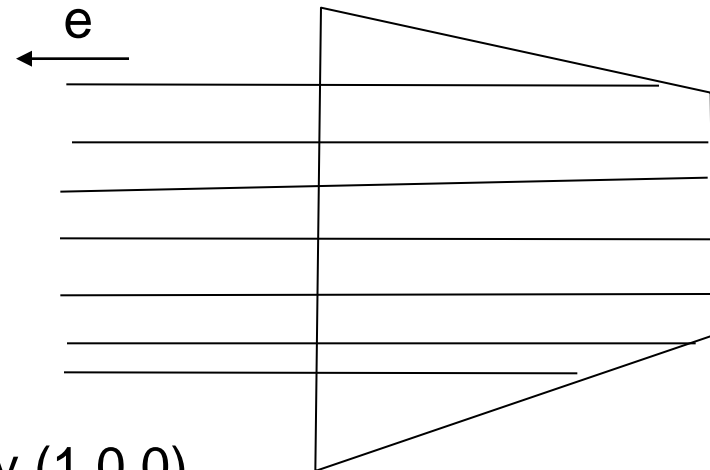
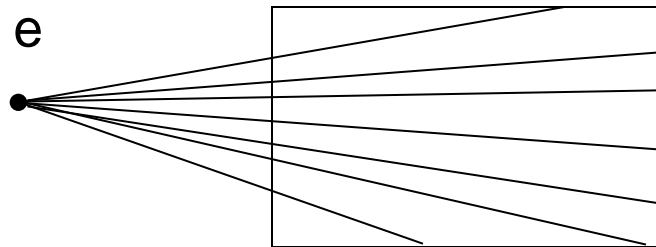


Geometric relations between two views are fully described by recovered 3×3 matrix \mathbf{F} .

Image Rectification

Simplify stereo matching by warping the images.

Apply projective transformation so that epipolar lines correspond to horizontal scanlines



$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = H\mathbf{e}$$

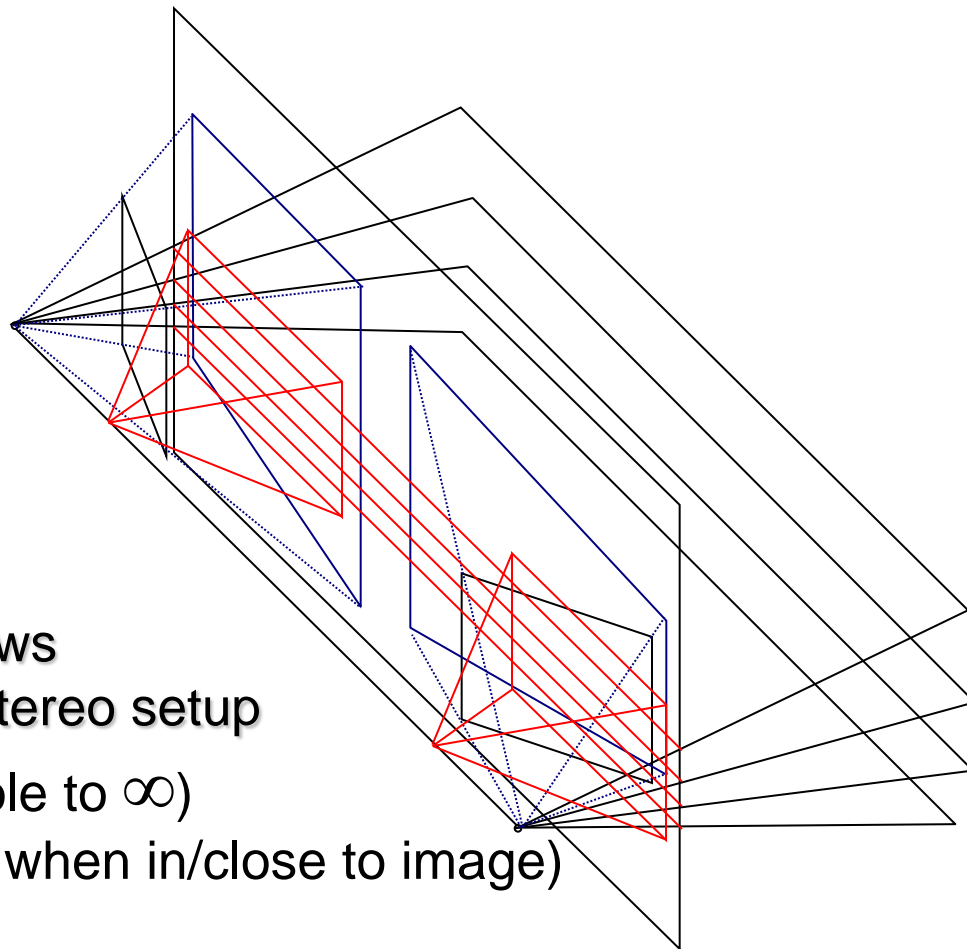
map epipole \mathbf{e} to infinity $(1,0,0)$

try to minimize image distortion

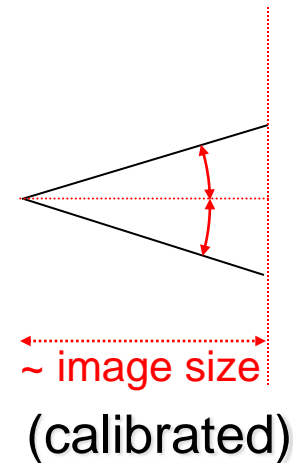
problem when epipole in (or close to) the image

Planar Rectification

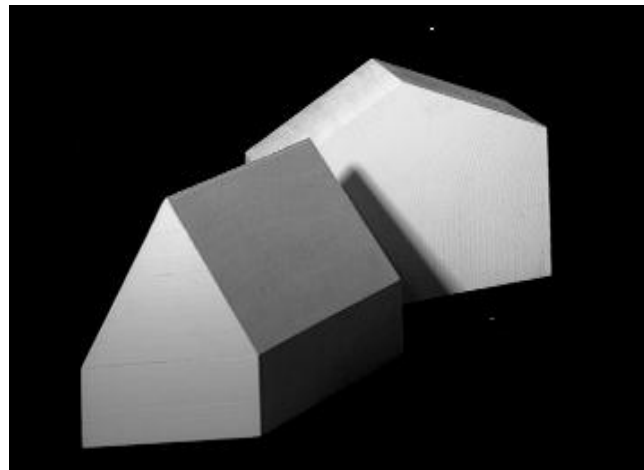
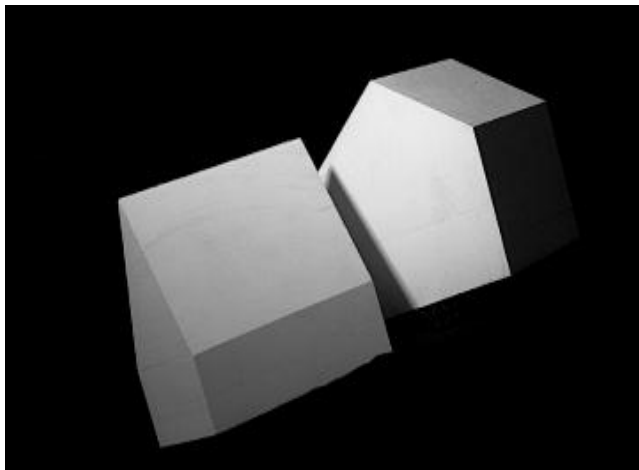
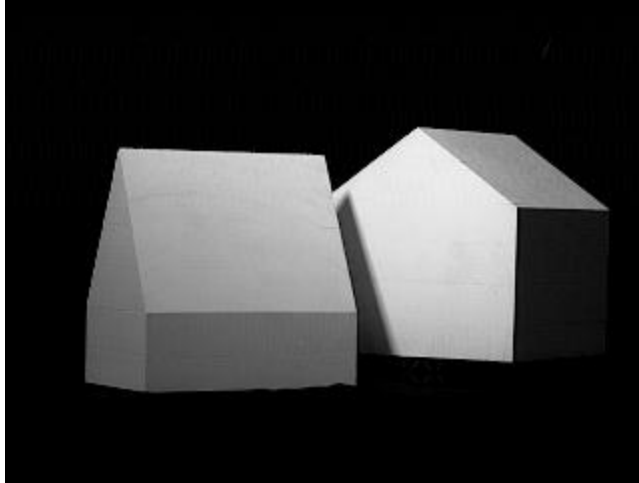
(standard approach)



Bring two views
to standard stereo setup
(moves epipole to ∞)
(not possible when in/close to image)



Distortion minimization
(uncalibrated)





Disparity Map

image $I(x,y)$



Disparity map $D(x,y)$

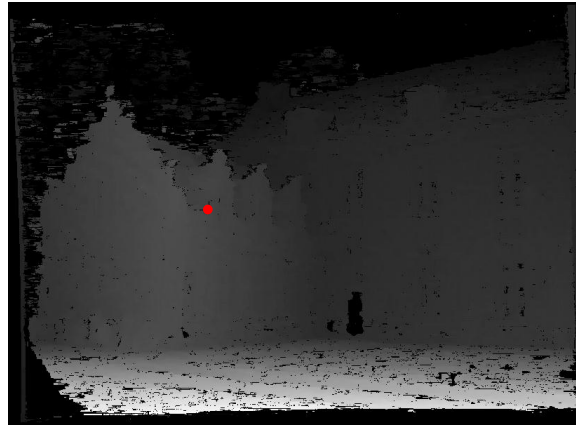


image $I'(x',y')$



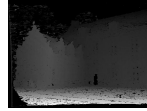
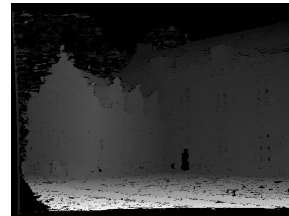
$$(x',y')=(x+D(x,y),y)$$

Hierarchical Stereo Matching

Allows faster computation

Deals with large disparity ranges

Downsampling
(Gaussian pyramid)



Disparity propagation



(Falkenhagen '97; Van Meerbergen, Vergauwen, Pollefeys, VanGool IJCV'02)

Example: Reconstruct Image from Neighboring Images

