Multiple View Geometry

Marc Pollefeys

Schedule (tentative)

#	date	topic
1	Sep.17	Introduction and geometry
2	Sep.24	Camera models and calibration
3	Oct.1	Invariant features
4	Oct.8	**Multiple-view geometry**
5	Oct.15	Model fitting (RANSAC, EM, …)
6	Oct.22	Stereo Matching
7	Oct.29	Structure from motion
8	Nov.5	Segmentation
9	Nov.12	Shape from X (silhouettes, …)
10	Nov.19	Optical flow
11	Nov.26	Tracking (Kalman, particle filter)
12	Dec.3	Object category recognition
13	Dec.10	Specific object recognition
14	Dec.17	Research overview

Two-view geometry

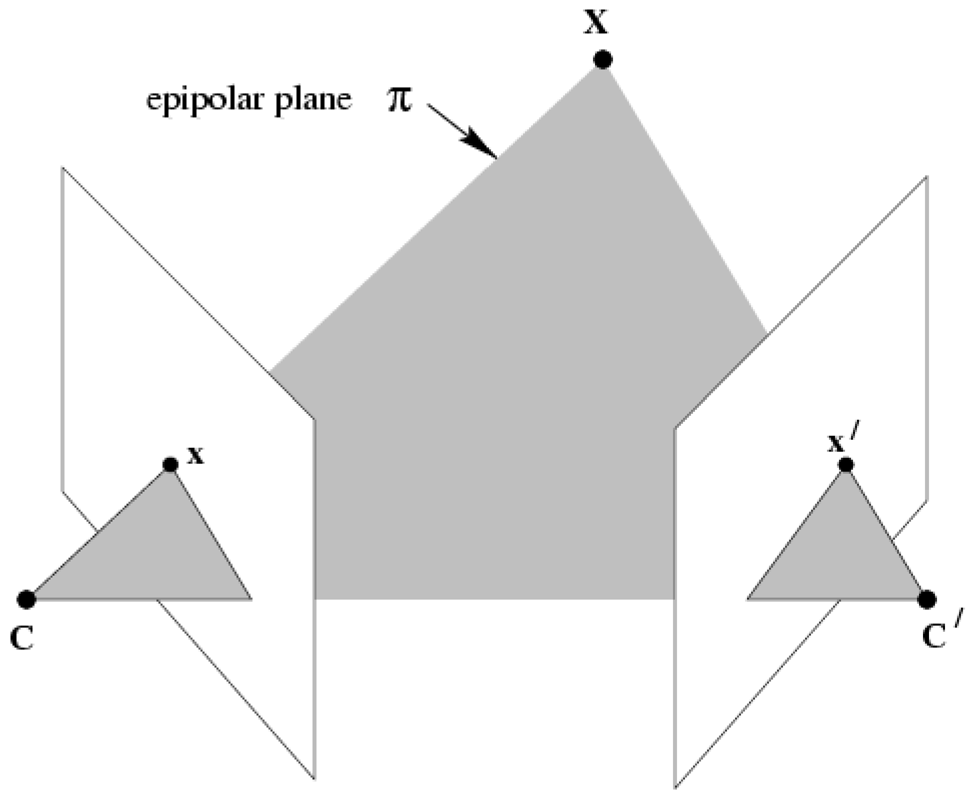
Three questions:

(i) Correspondence geometry: Given an image point $\mathbf{x}$ in the first image, how does this constrain the position of the corresponding point $\mathbf{x}'$ in the second image?

(ii) Camera geometry (motion): Given a set of corresponding image points $\{x_i \leftrightarrow x'_i\}$, $i=1,\ldots,n$, 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?

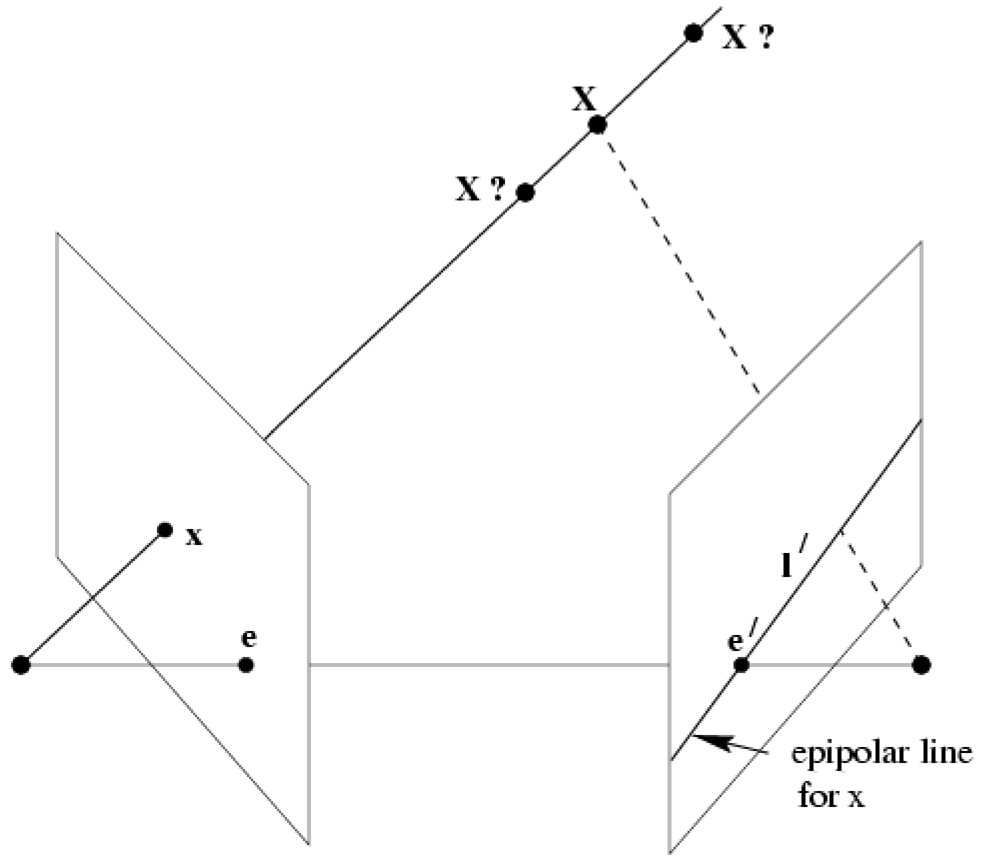
The epipolar geometry



C,C',x,x' and X are coplanar

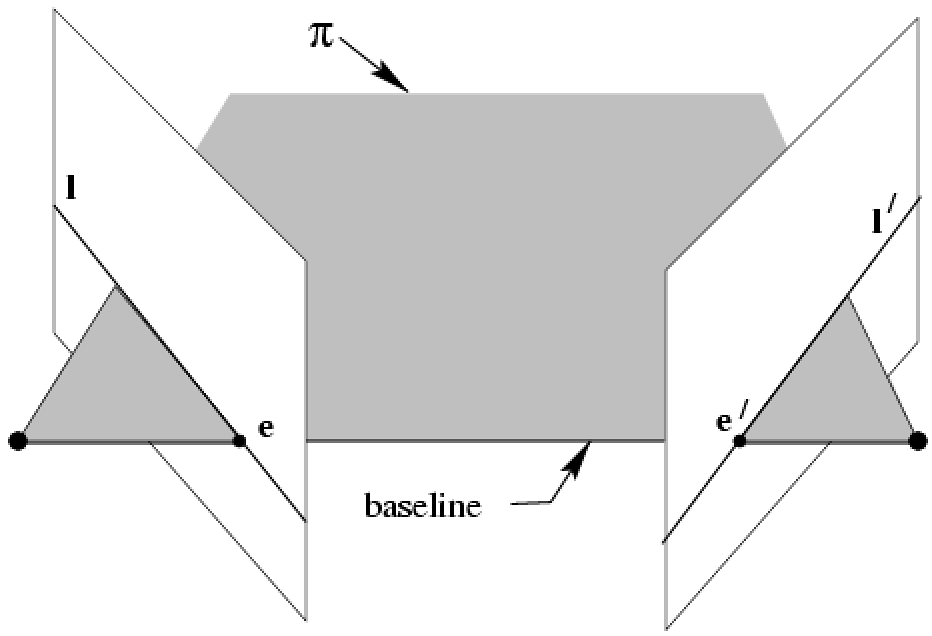
Computer Vision

The epipolar geometry



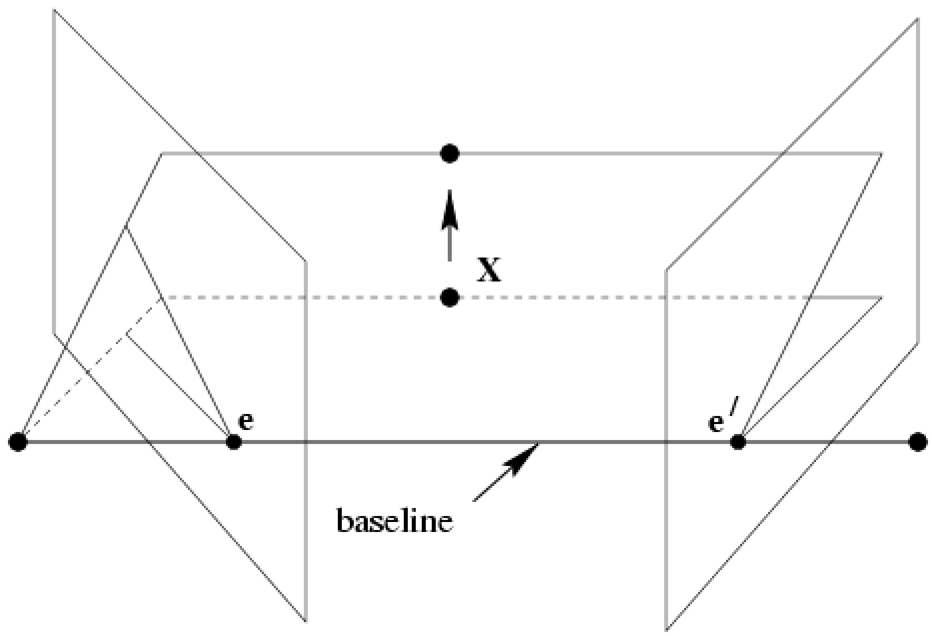
What if only C,C’,x are known?

The epipolar geometry



All points on π project on l and l'

The epipolar geometry

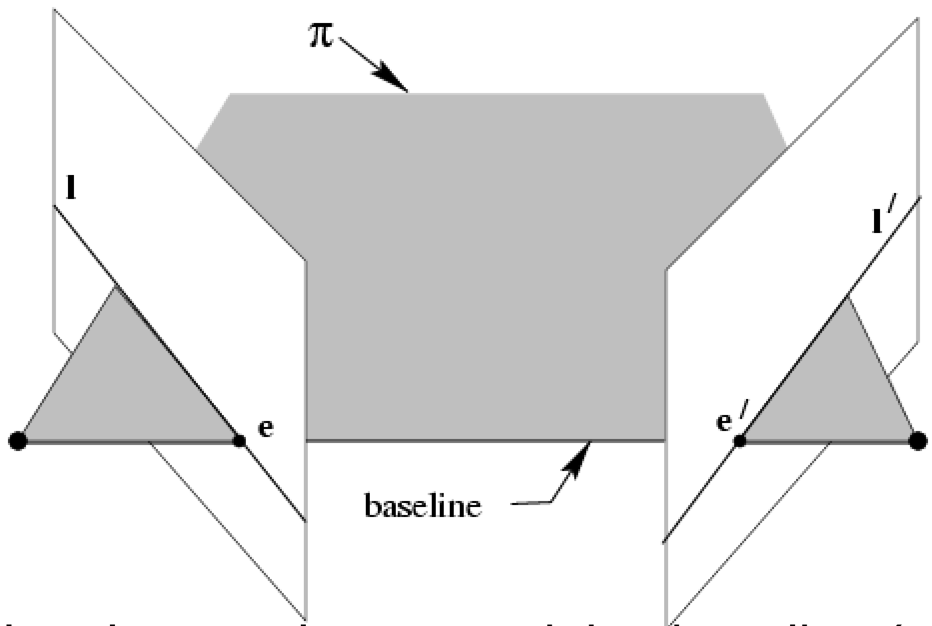


Family of planes π and lines l and l'
Intersection in e and e'

Computer Vision

The epipolar geometry

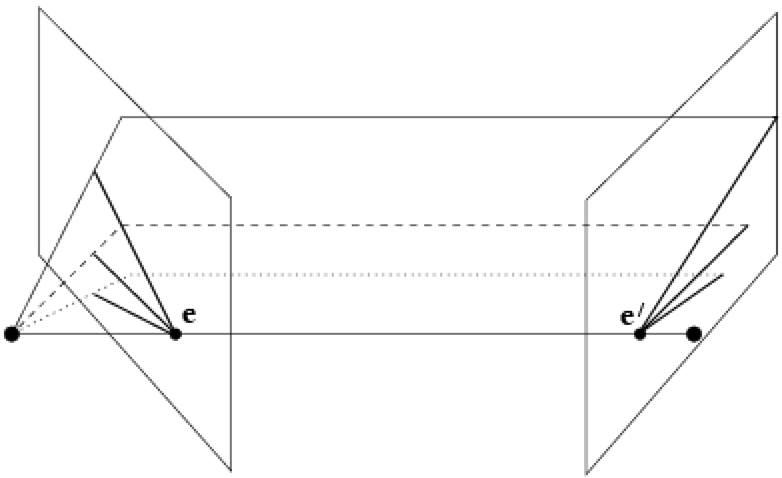
epipoles e,e'
= intersection of baseline with image plane
= projection of projection center in other image
= vanishing point of camera motion direction

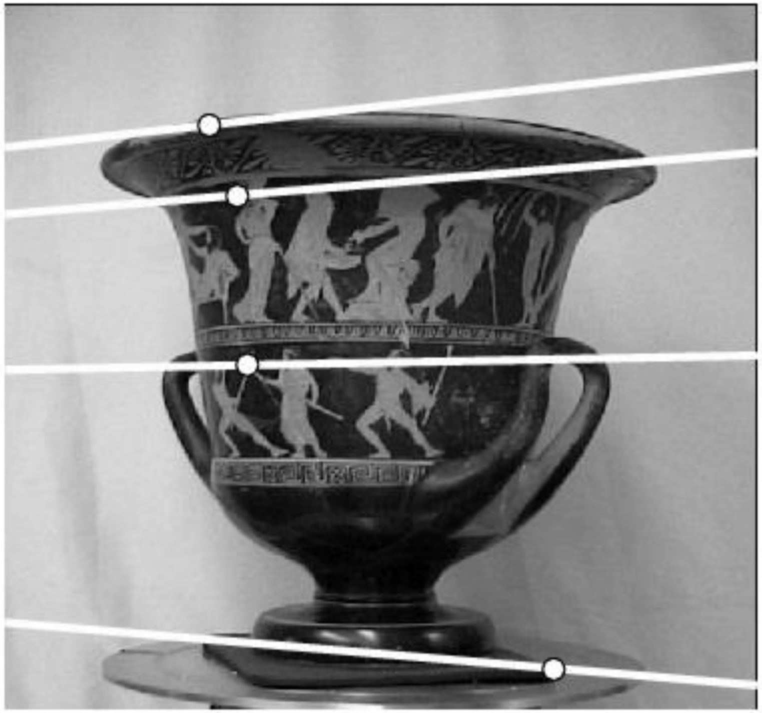


an epipolar plane = plane containing baseline (1-D family)

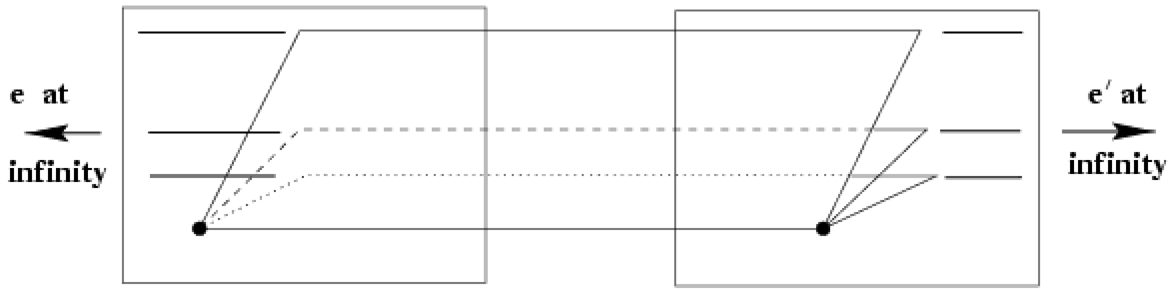
an epipolar line = intersection of epipolar plane with image
(always come in corresponding pairs)

Example: converging cameras



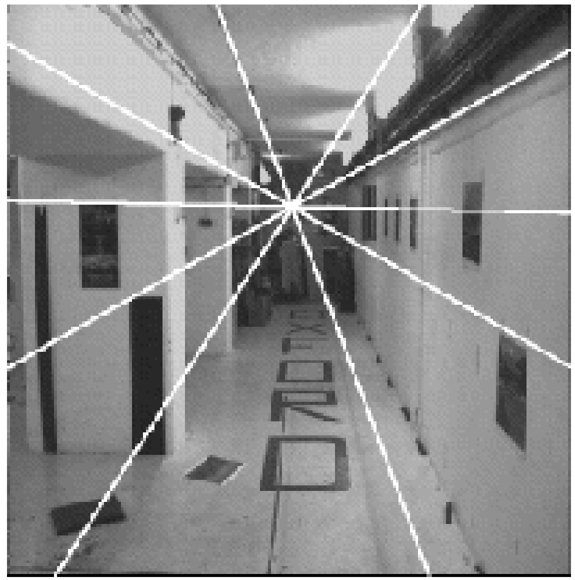


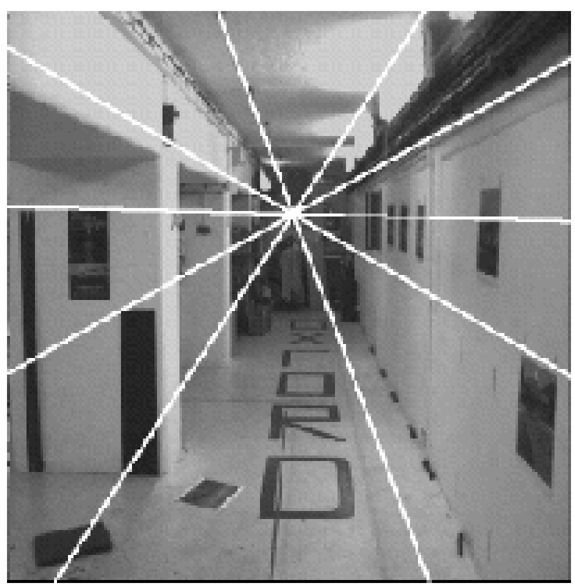
Example: motion parallel with image plane

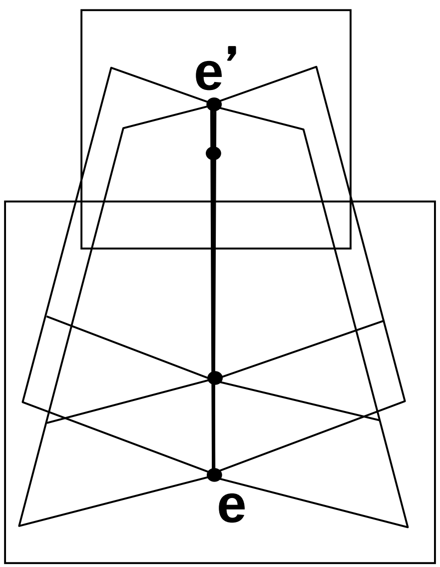


(simple for stereo → rectification)

Example: forward motion







The fundamental matrix F

algebraic representation of epipolar geometry

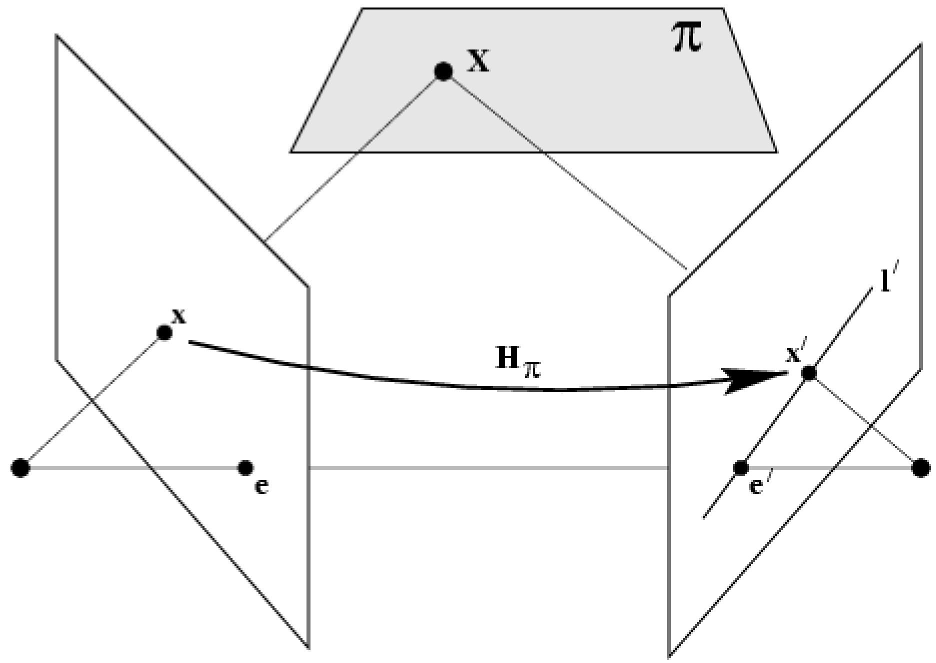
$$\mathrm{x} \mapsto \mathrm{l'}$$

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

Computer Vision

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)

Computer Vision

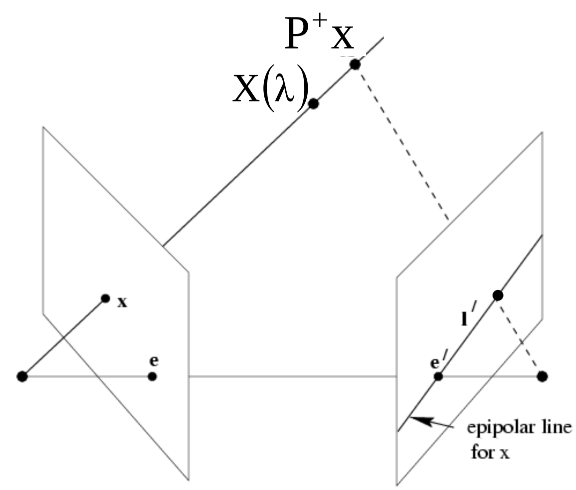
The fundamental matrix F

algebraic derivation

$$X(\lambda) = P^+ x + \lambda C \qquad \left(PP^+ = I\right)$$

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

$$F = [e']_\times P'P^+$$



(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

$$x'^T F x = 0 \qquad \left(x'^T l' = 0\right)$$

The fundamental matrix F

F is the unique 3x3 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: $l' = Fx$ & $l = F^T x'$

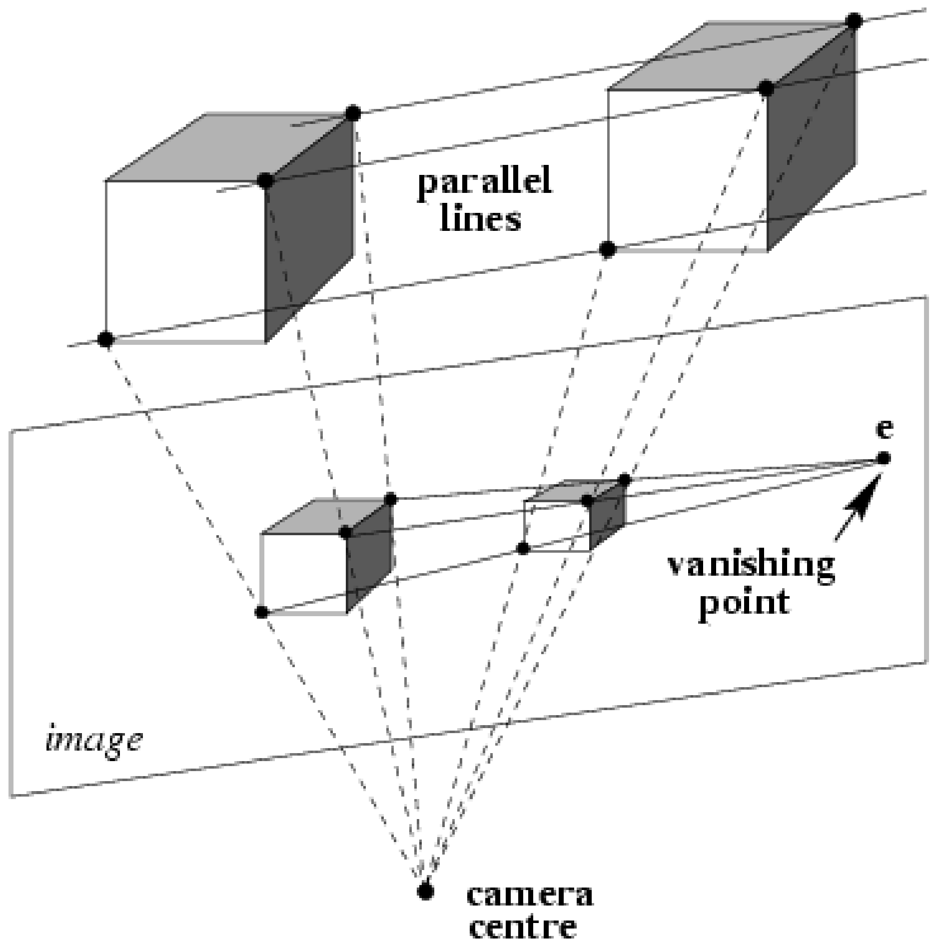
(iii) Epipoles: on all epipolar lines, thus $e'^T F x = 0, \forall x$ $\Rightarrow e'^T F = 0$, similarly $Fe = 0$

(iv) F has 7 d.o.f. , i.e. 3x3-1(homogeneous)-1(rank2)

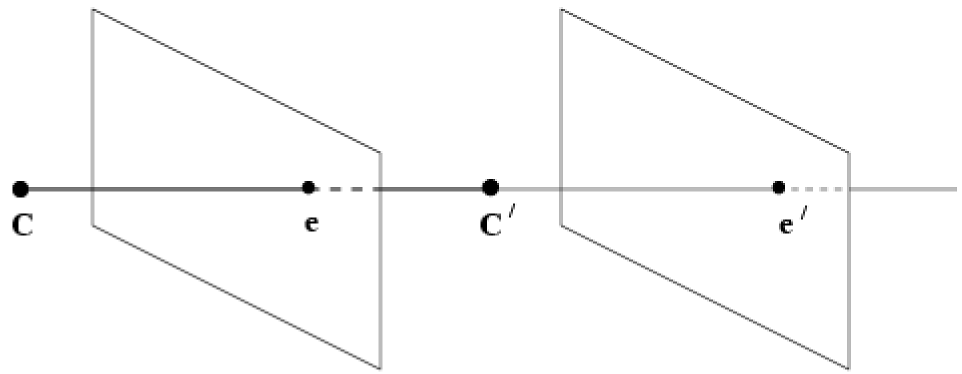
(v) F is a correlation, projective mapping from a point x to a line $l' = Fx$ (not a proper correlation, i.e. not invertible)

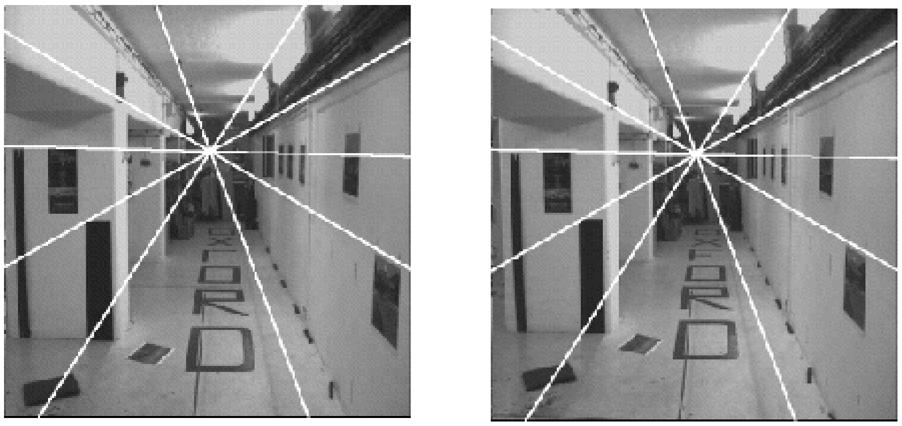
Computer Vision

Fundamental matrix for pure translation



Fundamental matrix for pure translation





Fundamental matrix for pure translation

General motion

$$F = [e']_\times P' P^+$$

Pure translation

$$P = K[I \mid 0] \qquad P^+ = \begin{bmatrix} K^{-1} \\ 0 \end{bmatrix}$$

$$P' = K[I \mid t]$$

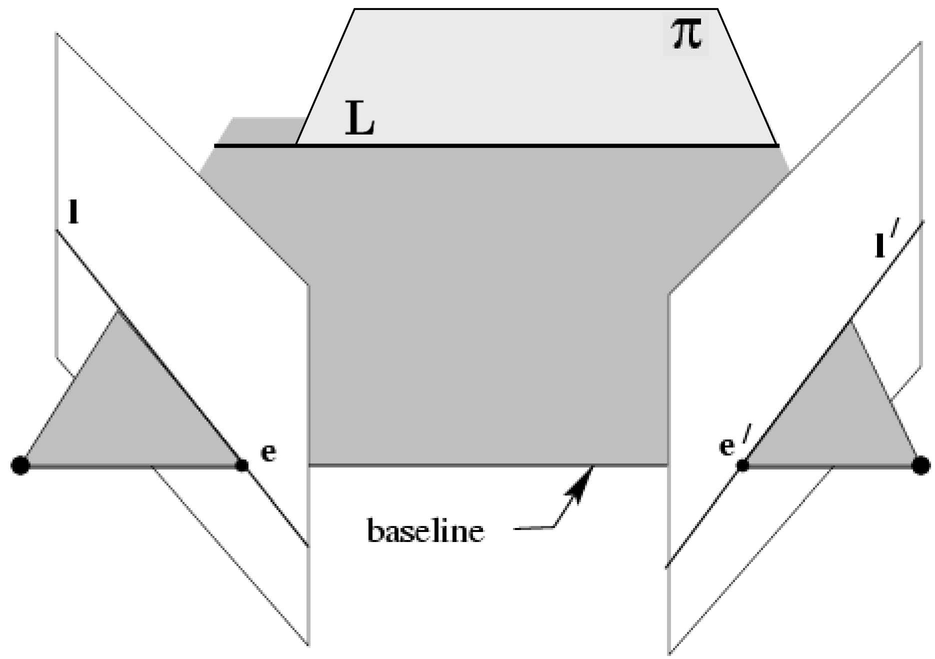
$$F = [e']_\times = \begin{bmatrix} 0 & e'_z & -e'_y \\ -e'_z & 0 & e'_x \\ e'_y & -e'_x & 0 \end{bmatrix}$$

for pure translation F only has 2 degrees of freedom

Computer Vision

The fundamental matrix F

relation to homographies

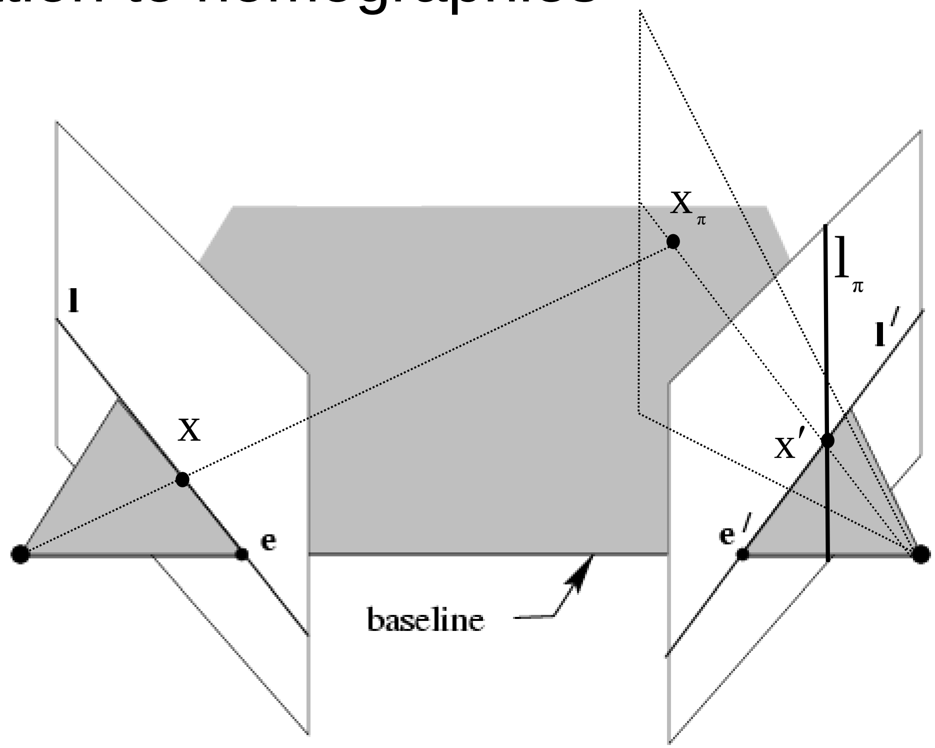


$$[e']_\times H_\pi = F \qquad l' = H_\pi^{-T} l \qquad e' = H_\pi e$$

valid for all plane homographies

The fundamental matrix F

relation to homographies



$$x' = H_\pi x = [l_\pi]_\times F x$$

requires $l_\pi^T e' \neq 0$

$$\text{e.g. } H = [e']_\times F$$

$$\left(e'^T e' \neq 0\right)$$

Projective transformation and invariance

Derivation based purely on projective concepts

$$\hat{x} = Hx,\ \hat{x}' = H'x' \Rightarrow \hat{F} = H'^{-T} F H^{-1}$$

F invariant to transformations of projective 3-space

$$x = PX = (PH)(H^{-1}X) = \hat{P}\hat{X}$$
$$x' = P'X = (P'H)(H^{-1}X) = \hat{P}'\hat{X}$$

$(P, P') \mapsto F$ **unique**

$F \mapsto (P, P')$ **not unique**

canonical form

$$P = [I \mid 0] \qquad P' = [M \mid m]$$

$$F = [m]_\times M \qquad \left(F = [e']_\times P'P^+\right)$$

Projective ambiguity of cameras given F

previous slide: at least projective ambiguity
this slide: not more!

Show that if F is same for (P,P’) and ($\tilde{P}$,$\tilde{P}$’),
there exists a projective transformation H so that
$\tilde{P}$=HP and $\tilde{P}$’=HP’

$$P=[I\,|\,0] \quad P'=[A\,|\,a]$$
$$\tilde{P}=[I\,|\,0] \quad \tilde{P}'=[\tilde{A}\,|\,\tilde{a}]$$
$$F=[a]_\times A=[\tilde{a}]_\times \tilde{A}$$

lemma: $\tilde{a}=ka$ and $\tilde{A}=k^{-1}\left(A+av^T\right)$

$$aF=a[a]_\times A=0=\tilde{a}F \overset{\text{rank 2}}{\Longrightarrow} \tilde{a}=ka$$
$$[a]_\times A=[\tilde{a}]_\times \tilde{A} \Rightarrow [a]_\times\left(k\tilde{A}-A\right)=0 \Rightarrow \left(k\tilde{A}-A\right)=av^T$$

$$H=\begin{bmatrix} k^{-1}I & 0 \\ k^{-1}v^T & k \end{bmatrix} \quad P'H=[A\,|\,a]\begin{bmatrix} k^{-1}I & 0 \\ k^{-1}v^T & k \end{bmatrix}$$

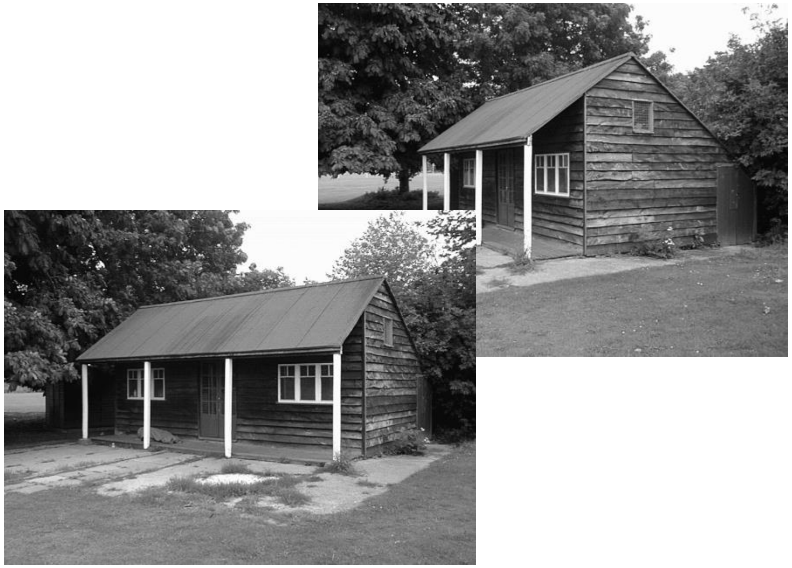
(22-15=7, ok)

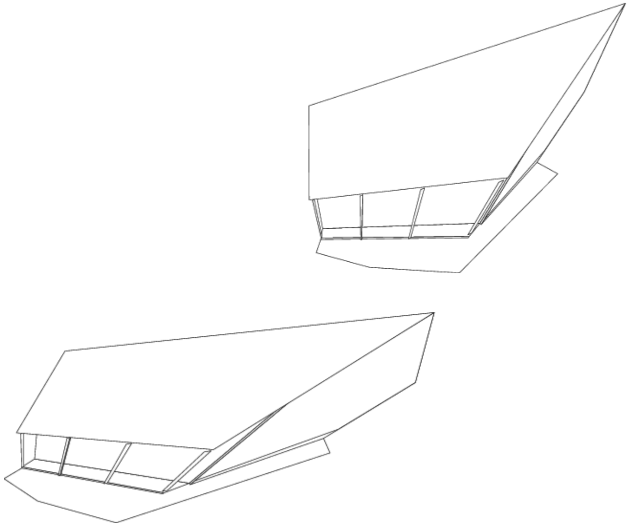
$$=[k^{-1}\left(A-av^T\right)|\,ka]=\tilde{P}'$$

The projective reconstruction theorem

If a set of point correspondences in two views determine the fundamental matrix uniquely, then the scene and cameras may be reconstructed from these correspondences alone, and any two such reconstructions from these correspondences are projectively equivalent

allows reconstruction from pair of uncalibrated images!





Canonical cameras given F

Possible choice:

$$P = [I \,|\, 0] \quad P' = [[e']_\times F \,|\, e']$$

$$F = [e']_\times P' P^+ = [e']_\times [[e']_\times F \,|\, e'] \begin{bmatrix} I \\ 0 \end{bmatrix}$$

$$\left([e']_\times [e']_\times = e'.e'^T - \left(e'^T.e'\right) I \right)$$

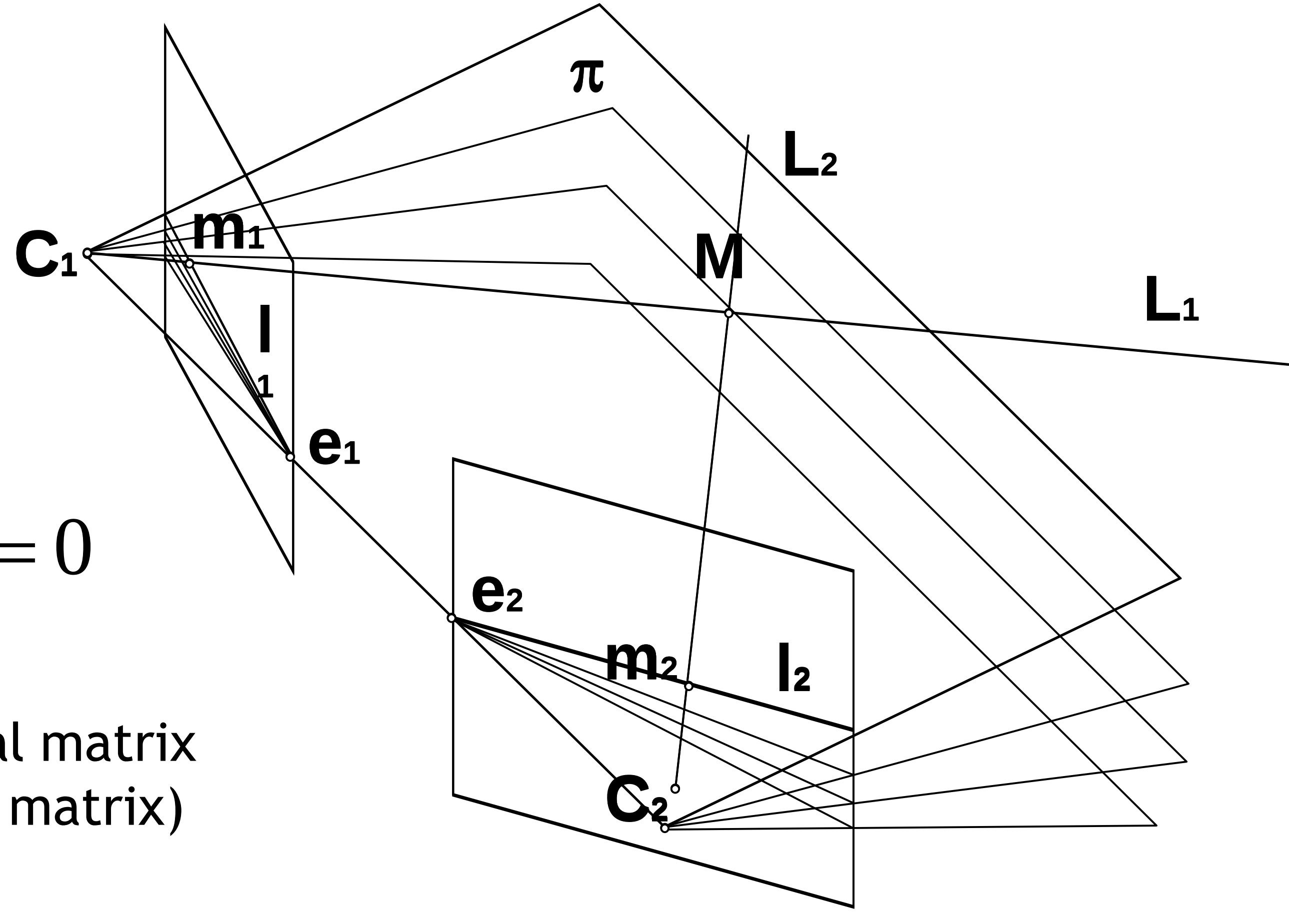
$$= \left(e'.e'^T - \left(e'^T.e'\right)\right) F = \lambda F$$

Canonical representation:

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

Epipolar geometry

Underlying structure in set of matches for rigid scenes



$$\overbrace{m_2^{T}\mathbf{F}}^{l_1^{T}}\, m_1 = 0$$

Fundamental matrix (3x3 rank 2 matrix)

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

Canonical representation:

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

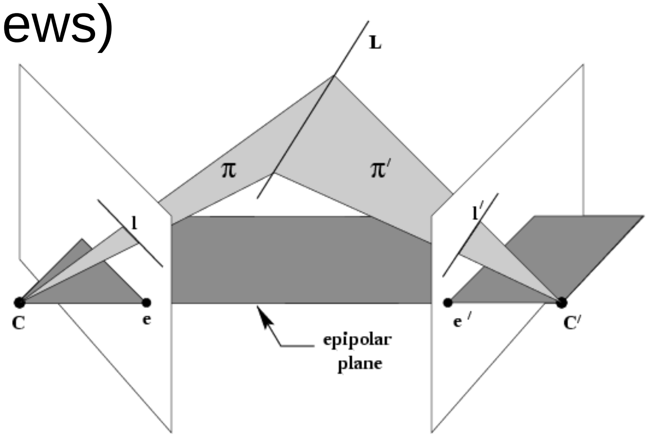
Epipolar geometry?

courtesy Frank Dellaert

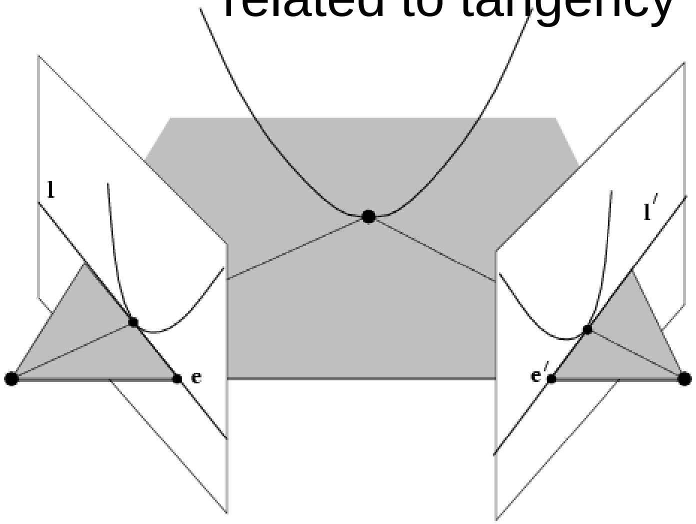
Computer Vision

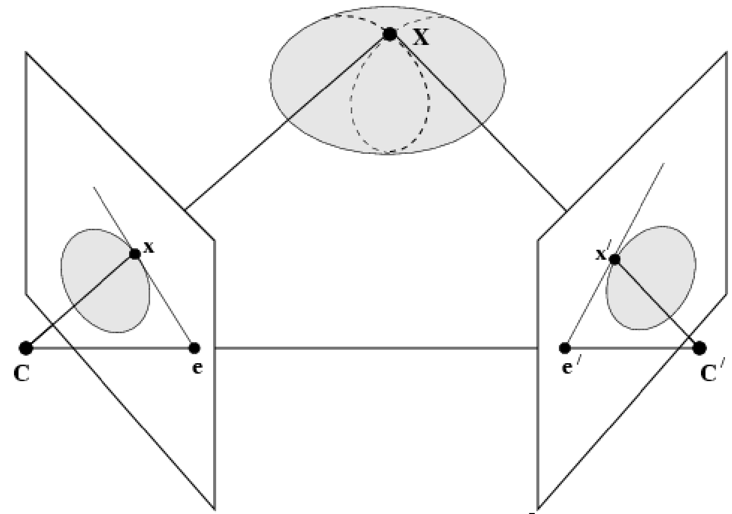
Other entities besides points?

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



Curves and surfaces yield some constraints
related to tangency





(e.g. Sinha et al. CVPR' 04)

Computation of F (and E)

- Linear (8-point)
- Minimal (7-point)
- Non-linear refinement (MLE, …)

- Calibrated 5-point
- Calibrated + know vertical 3-point

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)

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

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

Problem with eight-point algorithm

250906.36	183269.57	921.81	200931.10	146766.13	738.21	272.19	198.81	1.00
2692.28	131633.03	176.27	6196.73	302975.59	405.71	15.27	746.79	1.00
416374.23	871684.30	935.47	408110.89	854384.92	916.90	445.10	931.81	1.00
191183.60	171759.40	410.27	416435.62	374125.90	893.65	465.99	418.65	1.00
48988.86	30401.76	57.89	298604.57	185309.58	352.87	846.22	525.15	1.00
164786.04	546559.67	813.17	1998.37	6628.15	9.86	202.65	672.14	1.00
116407.01	2727.75	138.89	169941.27	3982.21	202.77	838.12	19.64	1.00
135384.58	75411.13	198.72	411350.03	229127.78	603.79	681.28	379.48	1.00

$$\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

linear least-squares:
unit norm vector F yielding smallest residual

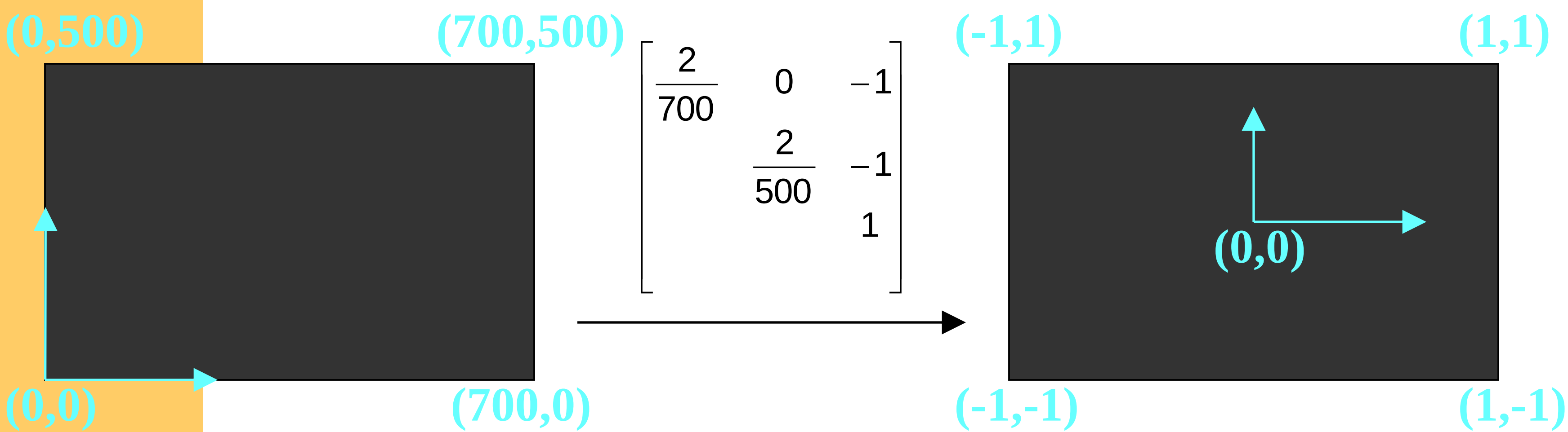
What happens when there is noise?

The Normalized Eight-Point Algorithm (Hartley, 1995)

- Center the image data at the origin, and scale it so the mean squared distance between the origin and the data points is 2 pixels: $q_i = T p_i$, $q_i' = T' p_i'$.
- Use the eight-point algorithm to compute $\mathcal{F}$ from the points q_i and q_i' .
- Enforce the rank-2 constraint.
- Output $T^T \mathcal{F} T'$.

Simplified normalized 8-point algorithm

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



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

normalized least squares yields good results

the singularity constraint

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

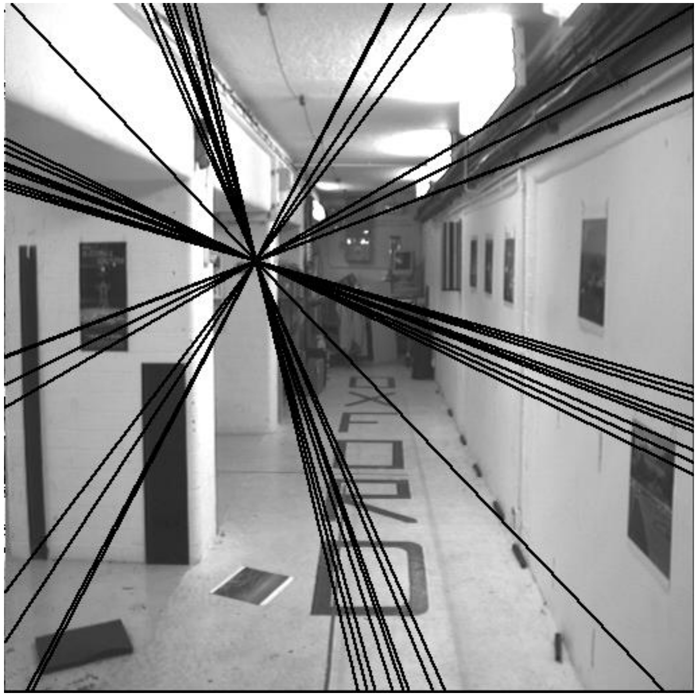
SVD from linearly computed F matrix (rank 3)

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

Compute closest rank-2 approximation $\min \|F - F'\|_{F}$

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

Computer Vision



the minimum case – 7 point correspondences

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

$$\mathrm{A} = \mathrm{U}_{7x7} \mathrm{diag}(\sigma_1, \ldots, \sigma_7, 0, 0) \mathrm{V}_{9x9}{}^{\mathrm{T}}$$

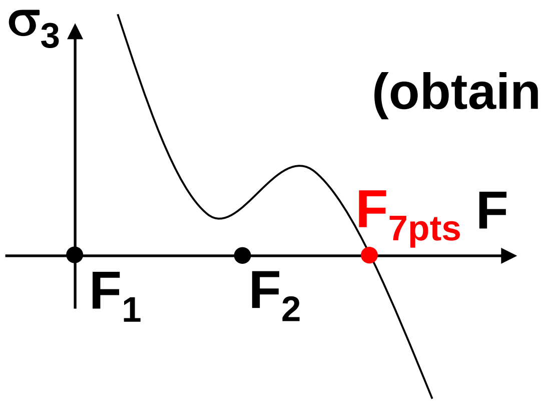
$$\Rightarrow \mathrm{A}[\mathrm{V}_8 \mathrm{V}_9] = 0_{9x2} \qquad \left(\mathrm{e.g.}\, \mathrm{V}^{\mathrm{T}} \mathrm{V}_8 = [000000010]^{\mathrm{T}}\right)$$

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

one parameter family of solutions

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

the minimum case – impose rank 2



(obtain 1 or 3 solutions)

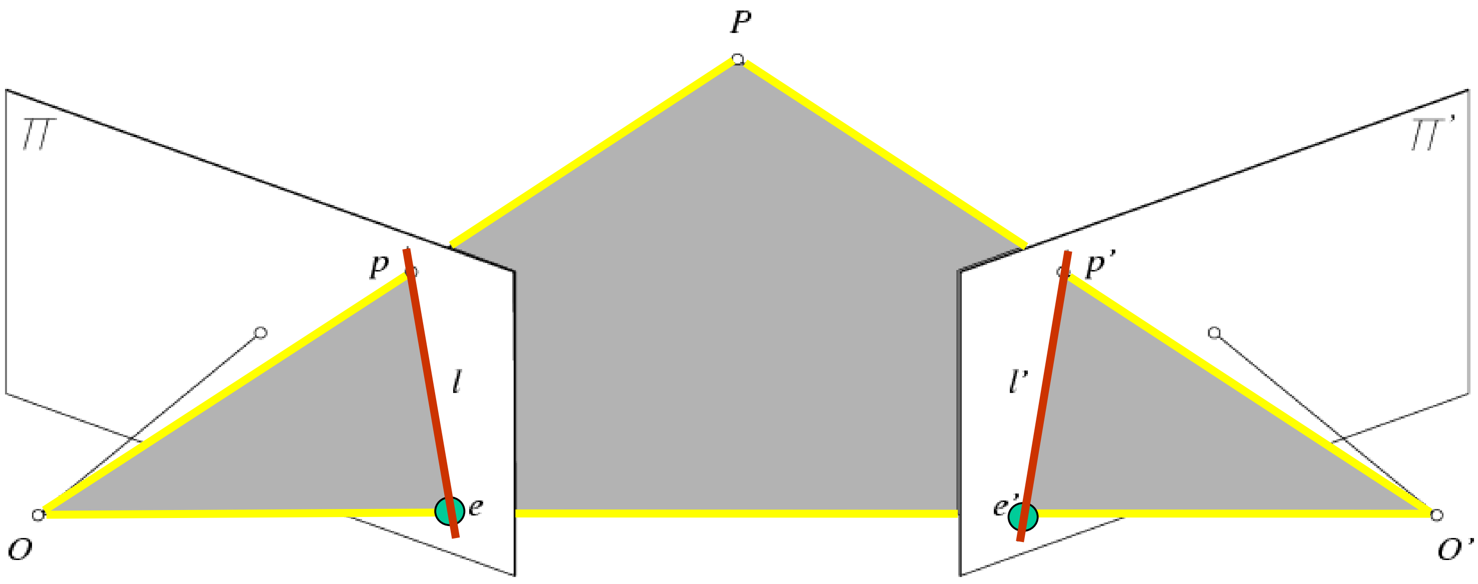
$$\det(F_1 + \lambda F_2) = a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0$$ **(cubic equation)**

$$\det(F_1 + \lambda F_2) = \det F_2 \det(F_2^{-1}F_1 + \lambda I) = 0$$ $\left(\det(AB) = \det(A).\det(B)\right)$

Compute possible λ as eigenvalues of $F_2^{-1}F_1$

(only real solutions are potential solutions)

Epipolar Constraint: Calibrated Case



$$\overrightarrow{Op} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p'}] = 0 \quad \Rightarrow \quad \boldsymbol{p} \cdot [\boldsymbol{t} \times (\mathcal{R}\boldsymbol{p}')] = 0 \quad \text{with} \begin{cases} \boldsymbol{p} = (u, v, 1)^T \\ \boldsymbol{p}' = (u', v', 1)^T \\ \mathcal{M} = (\text{Id} \quad \boldsymbol{0}) \\ \mathcal{M}' = (\mathcal{R}^T, -\mathcal{R}^T\boldsymbol{t}) \end{cases}$$

$$\boldsymbol{p}^T \mathcal{E} \boldsymbol{p}' = 0 \quad \text{with} \quad \mathcal{E} = [\boldsymbol{t}_\times]\mathcal{R}$$

Essential Matrix
(Longuet-Higgins, 1981)

Properties of the Essential Matrix

$$\boxed{\boldsymbol{p}^T \mathcal{E} \boldsymbol{p}' = 0 \quad \text{with} \quad \mathcal{E} = [\boldsymbol{t}_\times]\mathcal{R}}$$

- $\mathcal{E}$ p' is the epipolar line associated with p'.
- $\mathcal{E}^T$p is the epipolar line associated with p.
- $\mathcal{E}$ e' =0 and $\mathcal{E}^T$e=0.
- $\mathcal{E}$ is singular.
- $\mathcal{E}$ has two equal non-zero singular values (Huang and Faugeras, 1989).

5-point relative motion

(Nister, CVPR03)

- Linear equations for 5 points

$$\begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ x'_2x_2 & x'_2y_2 & x'_2 & y'_2x_2 & y'_2y_2 & y'_2 & x_2 & y_2 & 1 \\ x'_3x_3 & x'_3y_3 & x'_3 & y'_3x_3 & y'_3y_3 & y'_3 & x_3 & y_3 & 1 \\ x'_4x_4 & x'_4y_4 & x'_4 & y'_4x_4 & y'_4y_4 & y'_4 & x_4 & y_4 & 1 \\ x'_5x_5 & x'_5y_5 & x'_5 & y'_5x_5 & y'_5y_5 & y'_5 & x_5 & y_5 & 1 \end{bmatrix} \begin{bmatrix} E_{11} \\ E_{12} \\ E_{13} \\ E_{21} \\ E_{22} \\ E_{23} \\ E_{31} \\ E_{32} \\ E_{33} \end{bmatrix}$$

- Linear solution space

$E = xX + yY + zZ + wW$ scale does not matter, choose $w = 1$

- Non-linear constraints

$$\left.\begin{aligned} &\det E = 0 \\ &EE^{T}E - \frac{1}{2}\mathit{trace}\left(EE^{T}\right)E = 0 \end{aligned}\right\} \text{10 cubic polynomials}$$

5-point relative motion

(Nister, CVPR03)

- Perform Gauss-Jordan elimination on polynomials [n] represents polynomial of degree n in z

	A	x^3	y^3	x^2y	xy^2	x^2z	x^2	y^2z	y^2	xyz	xy	x	y	1
	$\langle a\rangle$	1	.	.	.	.	.	.	.	.	.	[2]	[2]	[3]
	$\langle b\rangle$		1	.	.	.	.	.	.	.	.	[2]	[2]	[3]
	$\langle c\rangle$			1	.	.	.	.	.	.	.	[2]	[2]	[3]
	$\langle d\rangle$				1	.	.	.	.	.	.	[2]	[2]	[3]
$\langle k\rangle$	$\langle e\rangle$					1						[2]	[2]	[3]
	-z $\langle f\rangle$						1					[2]	[2]	[3]
$\langle l\rangle$	$\langle g\rangle$							1				[2]	[2]	[3]
	-z $\langle h\rangle$								1			[2]	[2]	[3]
$\langle m\rangle$	$\langle i\rangle$									1		[2]	[2]	[3]
	-z $\langle j\rangle$										1	[2]	[2]	[3]

$$\langle k\rangle \equiv \langle e\rangle - z\langle f\rangle$$
$$\langle l\rangle \equiv \langle g\rangle - z\langle h\rangle$$
$$\langle m\rangle \equiv \langle i\rangle - z\langle j\rangle$$

B	x	y	1
$\langle k\rangle$	[3]	[3]	[4]
$\langle l\rangle$	[3]	[3]	[4]
$\langle m\rangle$	[3]	[3]	[4]

$$\boxed{\langle n\rangle \equiv det(B)}$$

Minimal relative pose with know vertical

Fraundorfer, Tanskanen and Pollefeys, ECCV2010

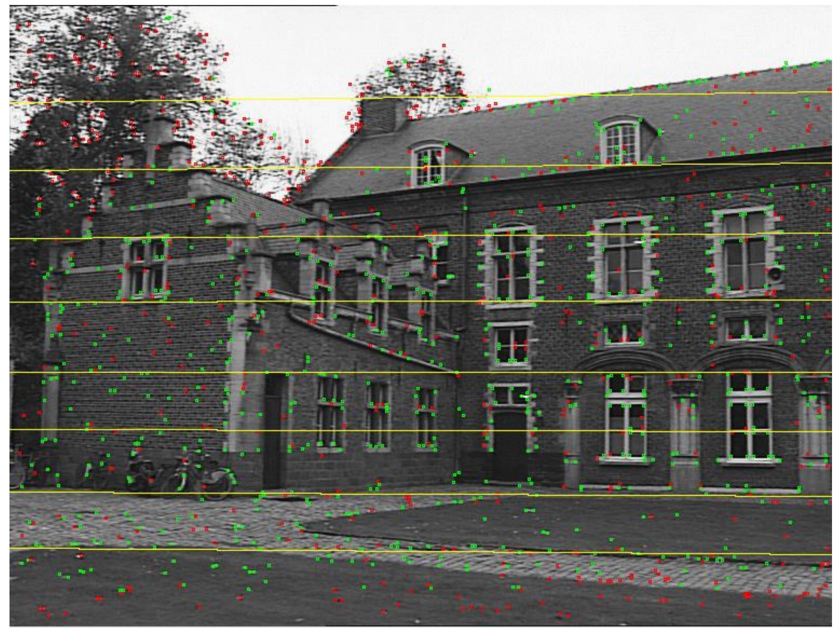
Vertical direction can often be estimated
- inertial sensor
- vanishing point

$$E = \begin{bmatrix} t_z \sin(y) & -t_z \cos(y) & t_y \\ t_z \cos(y) & t_z \sin(y) & -t_x \\ -t_y \cos(y) - t_x \sin(y) & t_x \cos(y) - t_y \sin(y) & 0 \end{bmatrix}$$

5 linear unknowns → linear 5 point algorithm

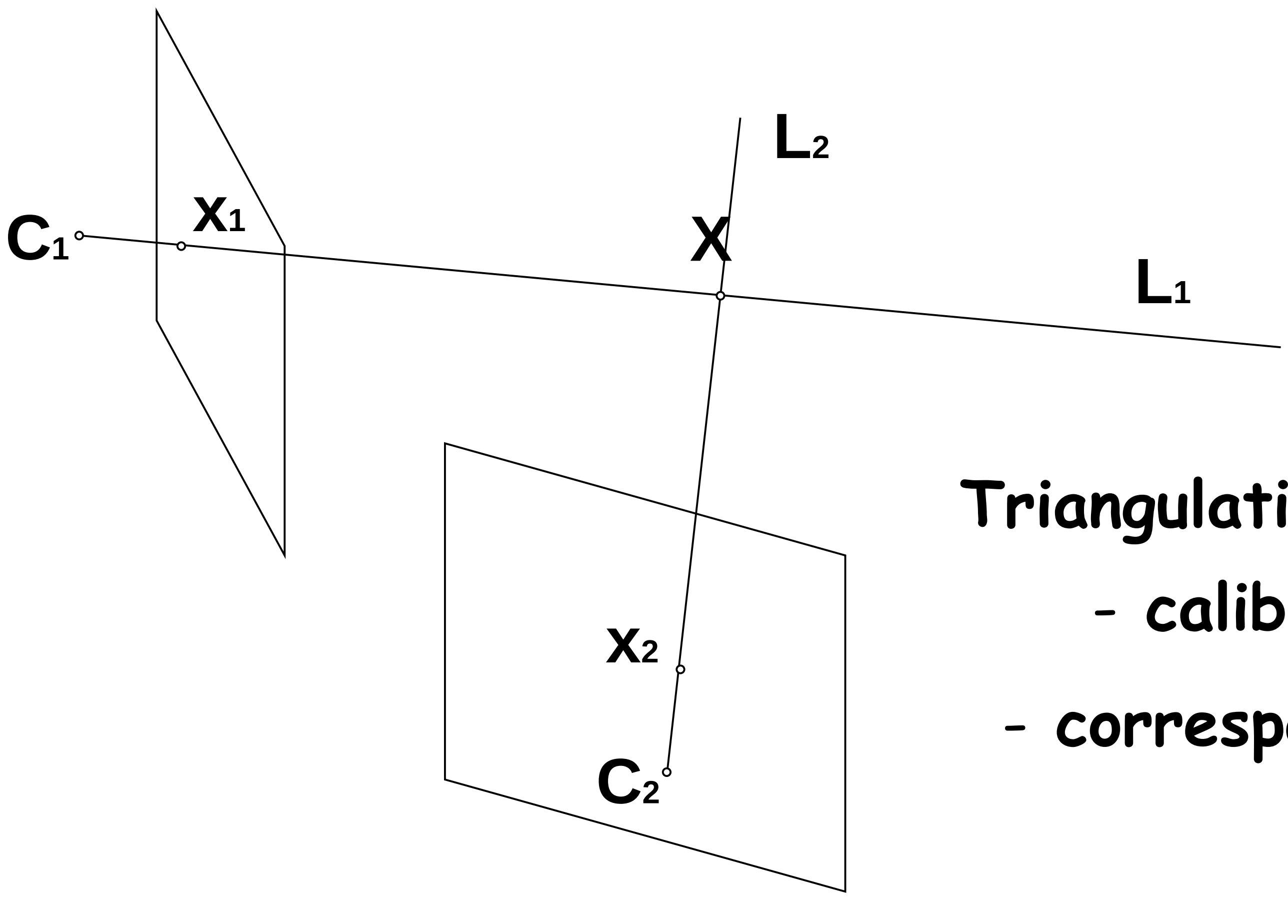
3 unknowns → quartic 3 point algorithm

two-view geometry



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

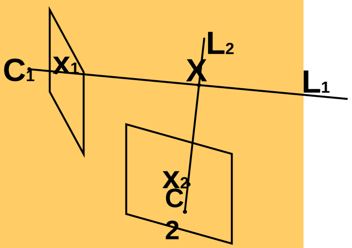
Triangulation



Triangulation

- calibration
- correspondences

Triangulation



- Backprojection

$$\lambda \mathrm{x} = \mathbf{P}\mathrm{X}$$

$$\begin{bmatrix} \lambda x \\ \lambda y \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathrm{P}_1 \\ \mathrm{P}_2 \\ \mathrm{P}_3 \end{bmatrix} \mathrm{X}$$

$$\begin{aligned} \mathrm{P}_3 \mathrm{X} x &= \mathrm{P}_1 \mathrm{X} \\ \mathrm{P}_3 \mathrm{X} y &= \mathrm{P}_2 \mathrm{X} \end{aligned}$$

$$\begin{bmatrix} \mathrm{P}_3 x - \mathrm{P}_1 \\ \mathrm{P}_3 y - \mathrm{P}_2 \end{bmatrix} \mathrm{X} = 0$$

- Triangulation

$$\begin{bmatrix} \mathrm{P}_3 x - \mathrm{P}_1 \\ \mathrm{P}_3 y - \mathrm{P}_2 \\ \mathrm{P}'_3 x' - \mathrm{P}'_1 \\ \mathrm{P}'_3 y' - \mathrm{P}'_2 \end{bmatrix} \mathrm{X} = 0$$

$$\begin{bmatrix} \frac{1}{\mathrm{P}_3 \tilde{\mathrm{X}}} \begin{pmatrix} \mathrm{P}_3 x - \mathrm{P}_1 \\ \mathrm{P}_3 y - \mathrm{P}_2 \end{pmatrix} \\ \frac{1}{\mathrm{P}'_3 \tilde{\mathrm{X}}} \begin{pmatrix} \mathrm{P}'_3 x - \mathrm{P}'_1 \\ \mathrm{P}'_3 y - \mathrm{P}'_2 \end{pmatrix} \end{bmatrix} \mathrm{X} = 0$$

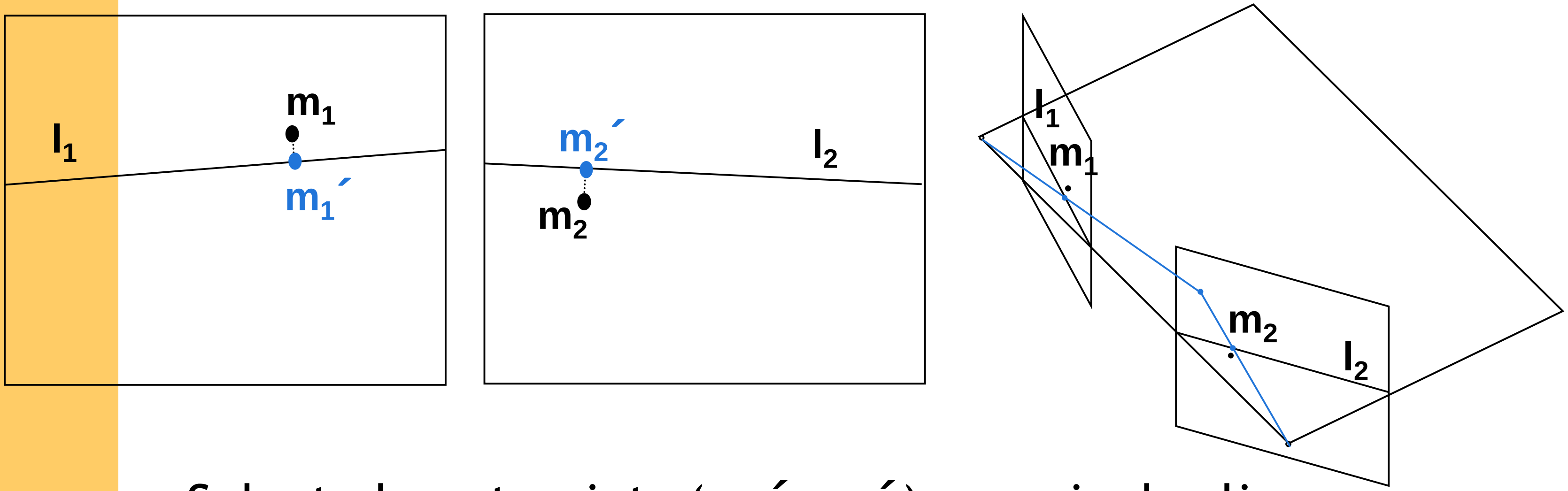
Iterative least-squares

- Maximum Likelihood Triangulation

$$\arg\min_{\mathrm{X}} \sum_i \left(\mathrm{x}_i - \lambda^{-1} \mathbf{P}_i \mathrm{X} \right)^2$$

Optimal 3D point in epipolar plane

- Given an epipolar plane, find best 3D point for (m_1,m_2)



Select closest points $(m_1´,m_2´)$ on epipolar lines

Obtain 3D point through exact triangulation

Guarantees minimal reprojection error (given this epipolar plane)

Non-iterative optimal solution

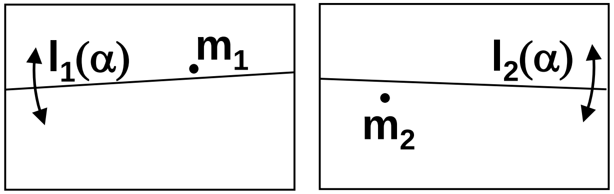
- Reconstruct matches in projective frame by minimizing the reprojection error

$$D(\mathbf{m}_1,\mathbf{P}_1\mathbf{M})^2 + D(\mathbf{m}_2,\mathbf{P}_2\mathbf{M})^2 \quad \textbf{3DOF}$$

- Non-iterative method **(Hartley and Sturm, CVIU´97)**

  Determine the epipolar plane for reconstruction

$$D(\mathbf{m}_1,\mathbf{l}_1(\alpha))^2 + D(\mathbf{m}_2,\mathbf{l}_2(\alpha))^2 \quad \textbf{(polynomial of degree 6)}$$



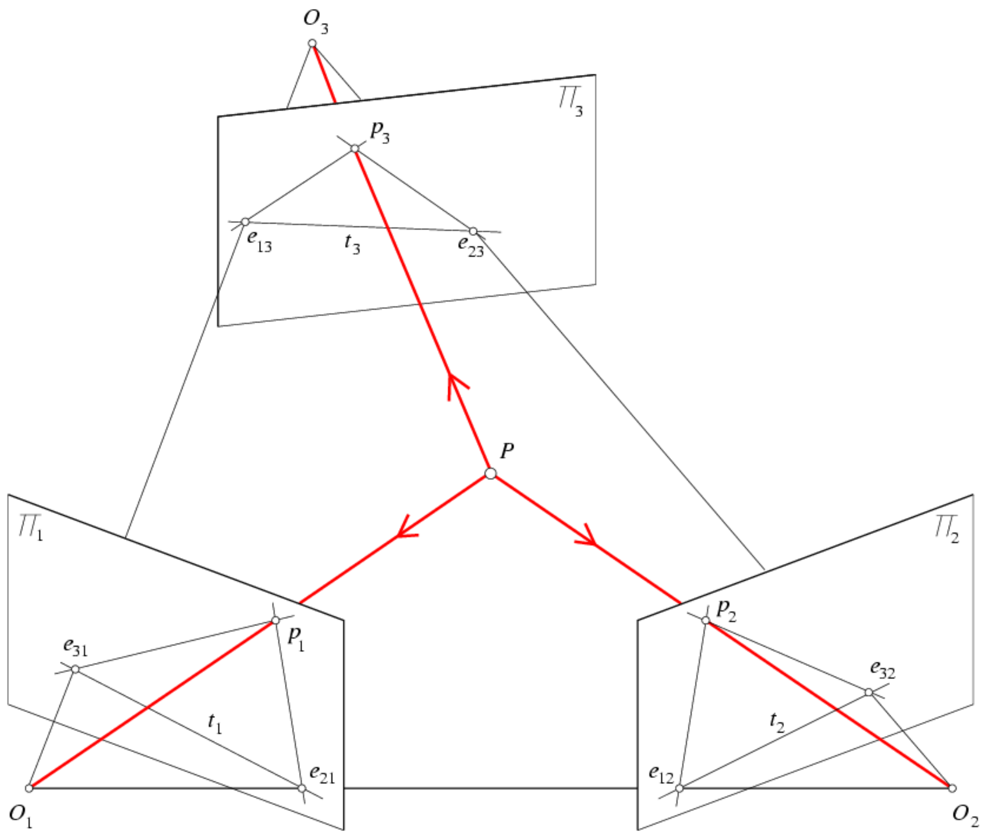
1DOF

Reconstruct optimal point from selected epipolar plane

Note: only works for two views

Computer Vision

Trinocular Epipolar Constraints

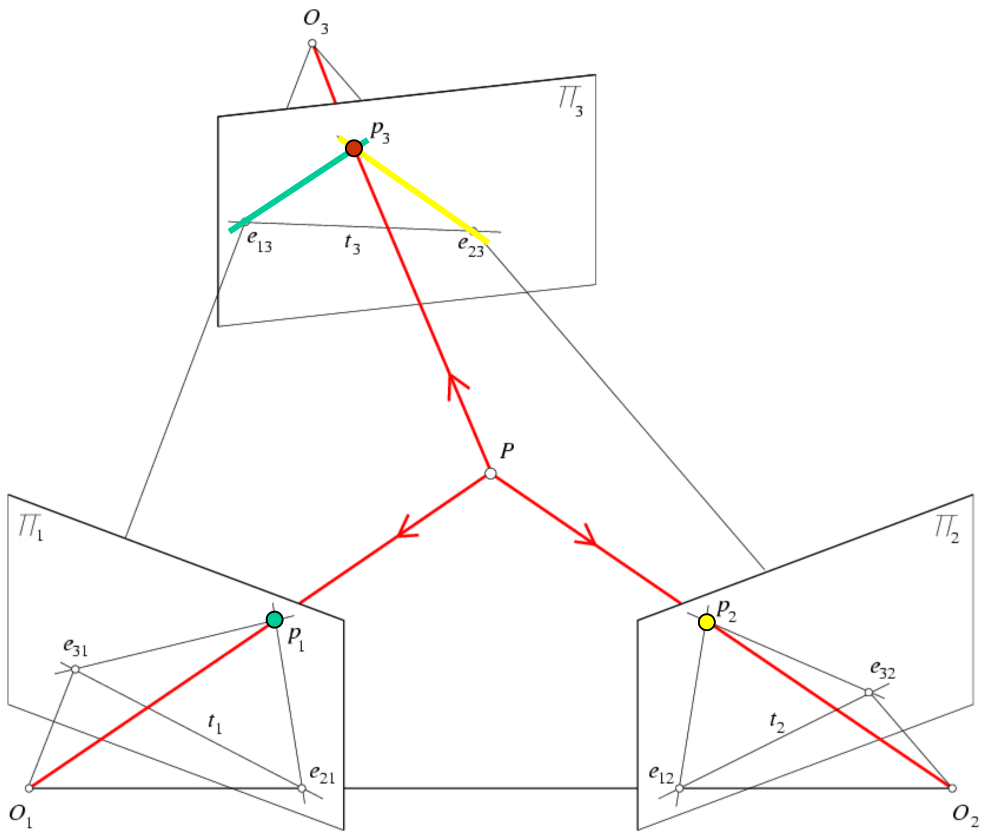


$$\begin{cases} \boldsymbol{p}_1^T \mathcal{E}_{12} \boldsymbol{p}_2 = 0 \\ \boldsymbol{p}_2^T \mathcal{E}_{23} \boldsymbol{p}_3 = 0 \\ \boldsymbol{p}_3^T \mathcal{E}_{31} \boldsymbol{p}_1 = 0 \end{cases}$$

These constraints are not independent!

$$\boldsymbol{e}_{31}^T \mathcal{E}_{12} \boldsymbol{e}_{32} = \boldsymbol{e}_{12}^T \mathcal{E}_{23} \boldsymbol{e}_{13} = \boldsymbol{e}_{23}^T \mathcal{E}_{31} \boldsymbol{e}_{21} = 0$$

Trinocular Epipolar Constraints: Transfer

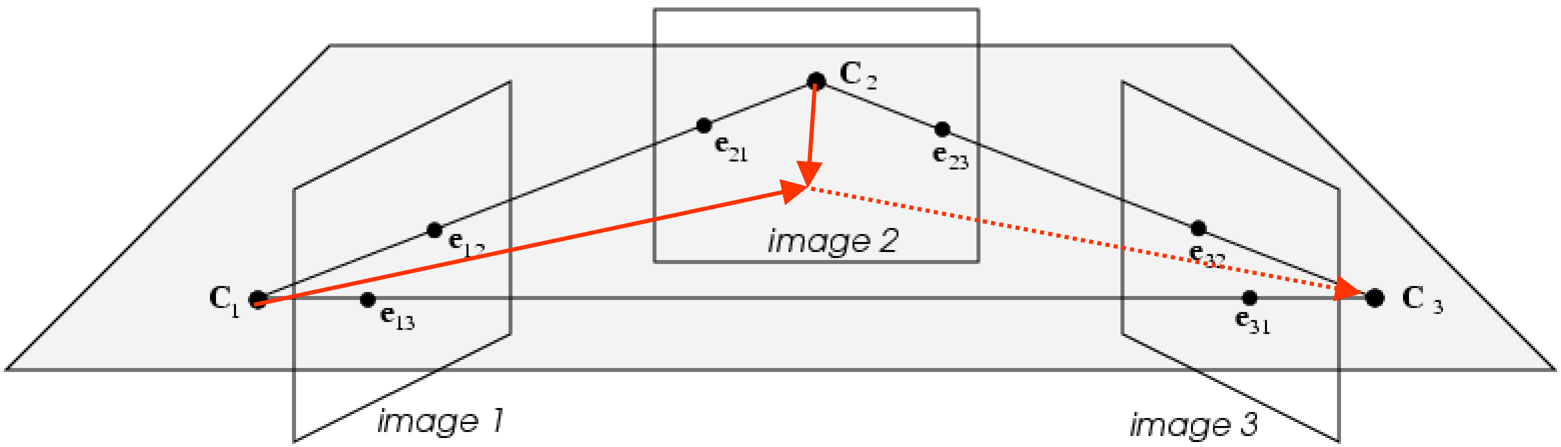


$$
\begin{cases}
\boldsymbol{p}_1^T \mathcal{E}_{12} \boldsymbol{p}_2 = 0 \\
\boldsymbol{p}_2^T \mathcal{E}_{23} \boldsymbol{p}_3 = 0 \\
\boldsymbol{p}_3^T \mathcal{E}_{31} \boldsymbol{p}_1 = 0
\end{cases}
$$

Given p_1 and p_2, p_3 can be computed as the solution of linear equations.

Trinocular Epipolar Constraints: Transfer

- problem for epipolar transfer in trifocal plane!



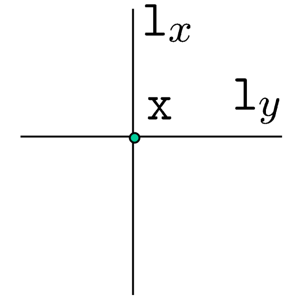
There must be more to trifocal geometry…

image from Hartley and Zisserman

Backprojection

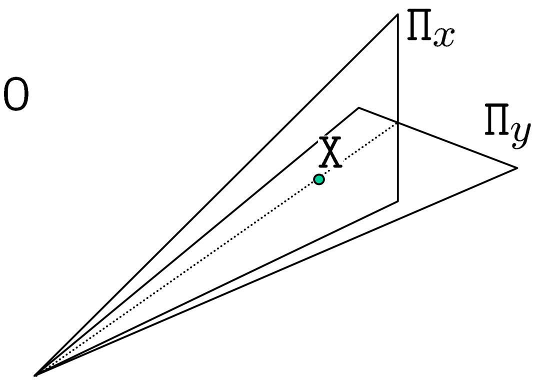
- Represent point as intersection of row and column

$$\mathtt{x} = \mathtt{l}_x \times \mathtt{l}_y \text{ with } \mathtt{l}_x = \begin{bmatrix} -1 \\ 0 \\ x \end{bmatrix}, \mathtt{l}_y = \begin{bmatrix} 0 \\ -1 \\ y \end{bmatrix}$$



$$\Pi = \mathtt{P}^\top \mathtt{l}$$

$$\begin{bmatrix} \Pi_x^\top \\ \Pi_y^\top \end{bmatrix} \mathtt{X} = 0 \qquad \begin{bmatrix} \mathtt{l}_x^\top \mathtt{P} \\ \mathtt{l}_y^\top \mathtt{P} \end{bmatrix} \mathtt{X} = 0$$



- Condition for solution?

$$\det \begin{bmatrix} \mathtt{l}_x^\top \mathtt{P} \\ \mathtt{l}_y^\top \mathtt{P} \\ \mathtt{l}_{x'}^\top \mathtt{P}' \\ \mathtt{l}_{y'}^\top \mathtt{P}' \end{bmatrix} = 0$$

Useful presentation for deriving and understanding multiple view geometry (notice 3D planes are linear in 2D point coordinates)

Multi-view geometry

$$\det \begin{bmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{bmatrix} = 0$$

(intersection constraint)

$$\begin{vmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix} = \begin{vmatrix} P_1 \\ P_2 - yP_3 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix} - x \begin{vmatrix} P_3 \\ P_2 - yP_3 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix}$$

(multi-linearity of determinants)

$$= \begin{vmatrix} P_1 \\ P_2 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix} - x \begin{vmatrix} P_3 \\ P_2 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix} - y \begin{vmatrix} P_1 \\ P_3 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix} - xy \cancel{\begin{vmatrix} P_3 \\ P_3 \\ P'_1 - x'P'_3 \\ P'_2 - y'P'_3 \end{vmatrix}}$$

$$= \cdots$$

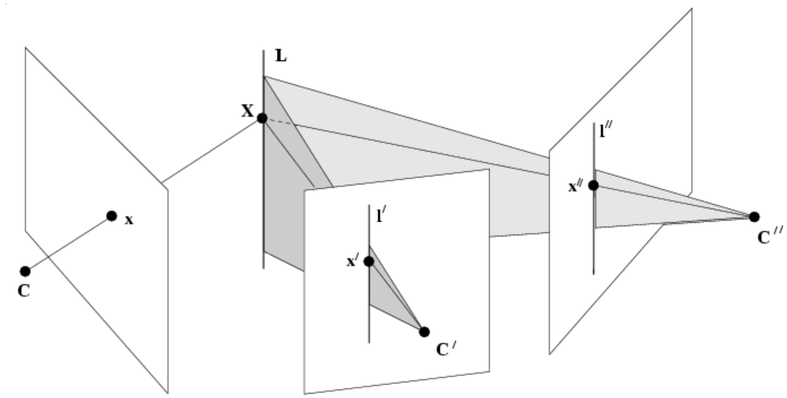
$$= axx' + byx' + cx' + dxy' + eyy' + fy' + gx + hy + i = 0$$

(= epipolar constraint!)

(counting argument: 11x2-15=7)

Multi-view geometry

$$\det \begin{bmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ P'_1 - x'P'_3 \\ P''_1 - x''P''_3 \end{bmatrix} = 0$$



$$\det \begin{bmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ l'_1P'_1 + l'_2P'_2 + l'_3P'_3 \\ l''_1P''_1 + l''_2P''_2 + l''_3P''_3 \end{bmatrix} = 0$$

(multi-linearity of determinants)

$$\begin{vmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ l'_1P'_1 + l'_2P'_2 + l'_3P'_3 \\ l''_1P''_1 + l''_2P''_2 + l''_3P''_3 \end{vmatrix} = l'_1 \begin{vmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ P'_1 \\ l''_1P''_1 + l''_2P''_2 + l''_3P''_3 \end{vmatrix} + l'_2 \begin{vmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ P'_2 \\ l''_1P''_1 + l''_2P''_2 + l''_3P''_3 \end{vmatrix} + l'_3 \begin{vmatrix} P_1 - xP_3 \\ P_2 - yP_3 \\ P'_3 \\ l''_1P''_1 + l''_2P''_2 + l''_3P''_3 \end{vmatrix}$$

$$= \cdots$$

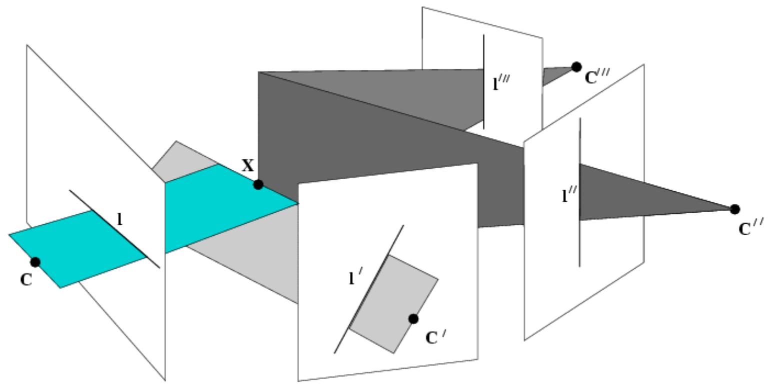
$$= axl'_1l''_1 + byl'_1l''_1 + cl'_1l''_1 + dxl'_2l''_1 + \cdots$$ (3x3x3=27 coefficients)

(= trifocal constraint!)

(counting argument: 11x3-15=18)

Multi-view geometry

$$\det\begin{bmatrix} \mathrm{P}_1 - x\mathrm{P}_3 \\ \mathrm{P}'_1 - x'\mathrm{P}'_3 \\ \mathrm{P}''_1 - x''\mathrm{P}''_3 \\ \mathrm{P}'''_1 - x'''\mathrm{P}'''_3 \end{bmatrix} = 0$$



$$\det\begin{bmatrix} l_1\mathrm{P}_1 + l_2\mathrm{P}_2 + l_3\mathrm{P}_3 \\ l'_1\mathrm{P}'_1 + l'_2\mathrm{P}'_2 + l'_3\mathrm{P}'_3 \\ l''_1\mathrm{P}''_1 + l''_2\mathrm{P}''_2 + l''_3\mathrm{P}''_3 \\ l'''_1\mathrm{P}'''_1 + l'''_2\mathrm{P}'''_2 + l'''_3\mathrm{P}'''_3 \end{bmatrix} = 0$$

(multi-linearity of determinants)

$$= a l_1 l'_1 l''_1 l'''_1 + b l_2 l'_1 l''_1 l'''_1 + c l_3 l'_1 l''_1 l'''_1 + \cdots$$

(3x3x3x3=81 coefficients)

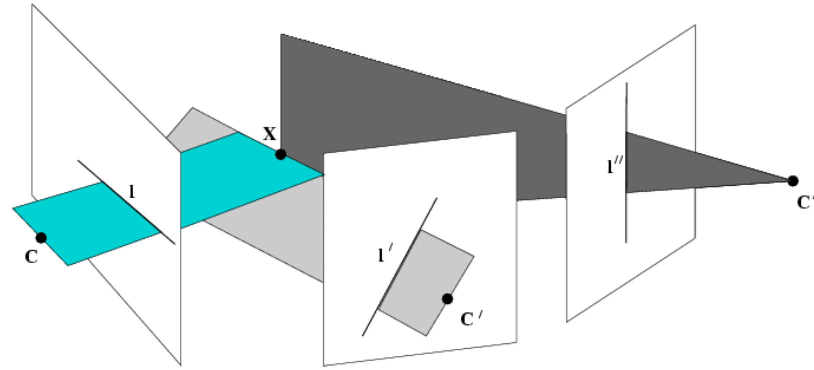
(= quadrifocal constraint!)

(counting argument: 11x4-15=29)

from perspective to omnidirectional cameras

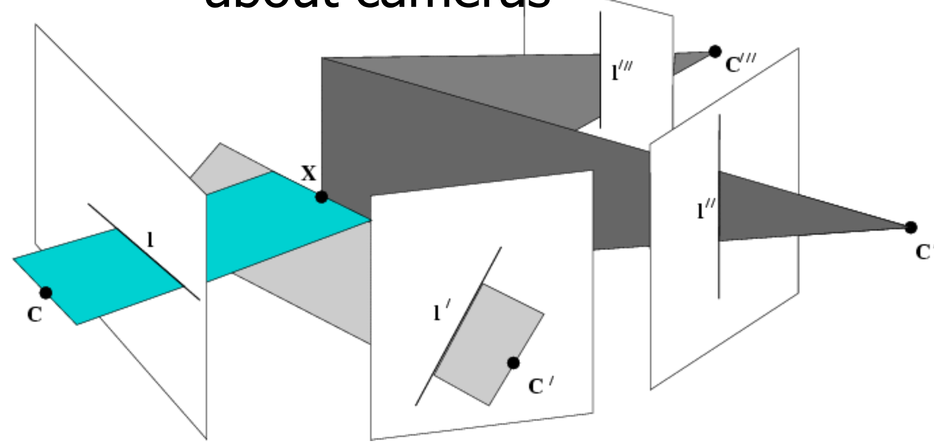
perspective camera
(2 constraints / feature)

3 constraints allow to reconstruct 3D point



more constraints also tell something about cameras

radial camera (uncalibrated)
(1 constraints / feature)



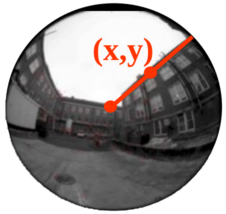
multilinear constraints known as epipolar, trifocal and quadrifocal constraints

Quadrifocal constraint

$$\lambda \mathbf{l} \quad = \quad \varepsilon \mathbf{P} \mathbf{X} \quad \text{with}\, \varepsilon = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} \varepsilon \mathbf{P} & \mathbf{l} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \varepsilon \mathbf{P}' & \mathbf{0} & \mathbf{l}' & \mathbf{0} & \mathbf{0} \\ \varepsilon \mathbf{P}'' & \mathbf{0} & \mathbf{0} & \mathbf{l}'' & \mathbf{0} \\ \varepsilon \mathbf{P}''' & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{l}''' \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ -\lambda \\ -\lambda' \\ -\lambda'' \\ -\lambda''' \end{bmatrix} = \mathbf{0}$$

$$\mathbf{l}_i \mathbf{l}'_j \mathbf{l}''_k \mathbf{l}'''_l \mathbf{Q}^{ijkl} = 0$$



Radial quadrifocal tensor

$$l = \begin{pmatrix} y \\ -x \end{pmatrix}$$

- Linearly compute radial quadrifocal tensor Q^{ijkl} from 15 pts in 4 views

$$l_i l_j l_k l_l Q^{ijkl} = 0$$ (2x2x2x2 tensor)

- Reconstruct 3D scene and use it for calibration

Not easy for real data, hard to avoid degenerate cases (e.g. 3 optical axes intersect in single point). However, degenerate case leads to simpler 3 view algorithm for pure rotation

- Radial trifocal tensor T^{ijk} from 7 points in 3 views

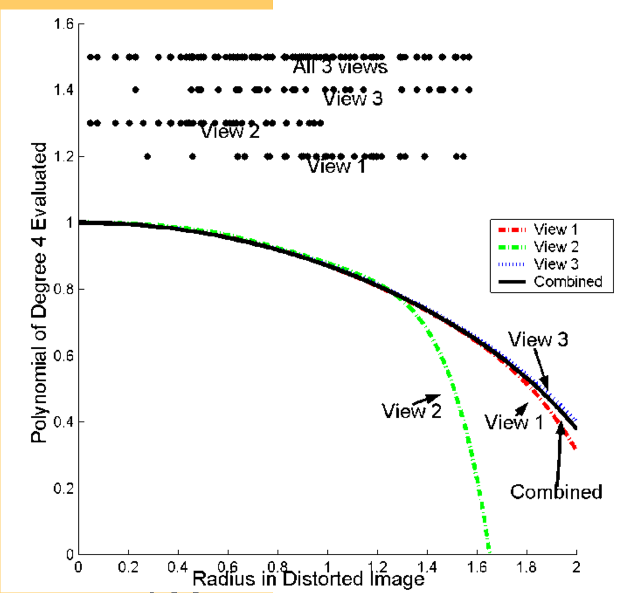
$$l_i l_j l_k T^{ijk} = 0$$ (2x2x2 tensor)

- Reconstruct 2D panorama and use it for calibration

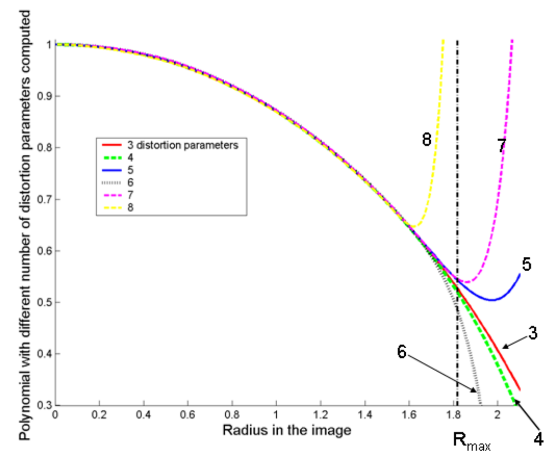
Dealing with Wide FOV Camera

(Thirthala and Pollefeys CVPR05)

- Two-step linear approach to compute radial distortion
- Estimates distortion polynomial of arbitrary



undistorted image



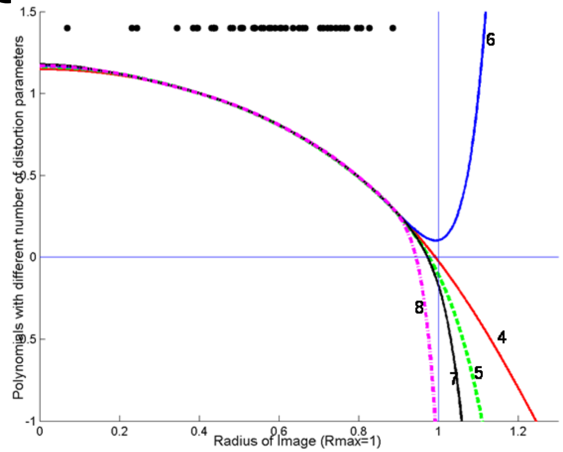
estimated distortion

(4-8 coefficients)

Dealing with Wide FOV Camera

(Thirthala and Pollefeys CVPR05)

- Two-step linear approach to compute radial distortion
- Estimates distortion polynomial of arbitrary degree



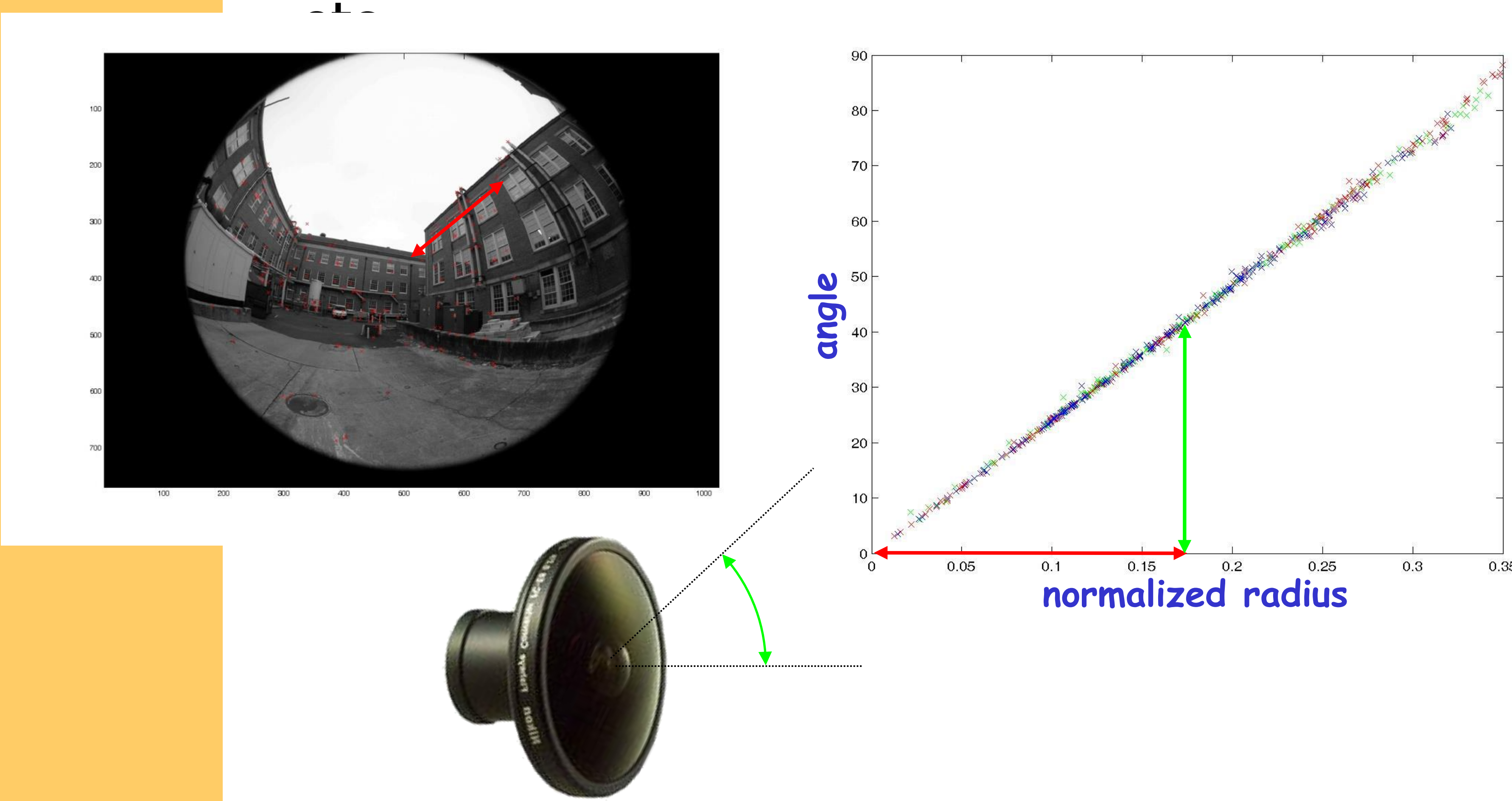
estimated distortion
(4-8 coefficients)

unfolded cubemap

Non-parametric distortion calibration

(Thirthala and Pollefeys ICCV05)

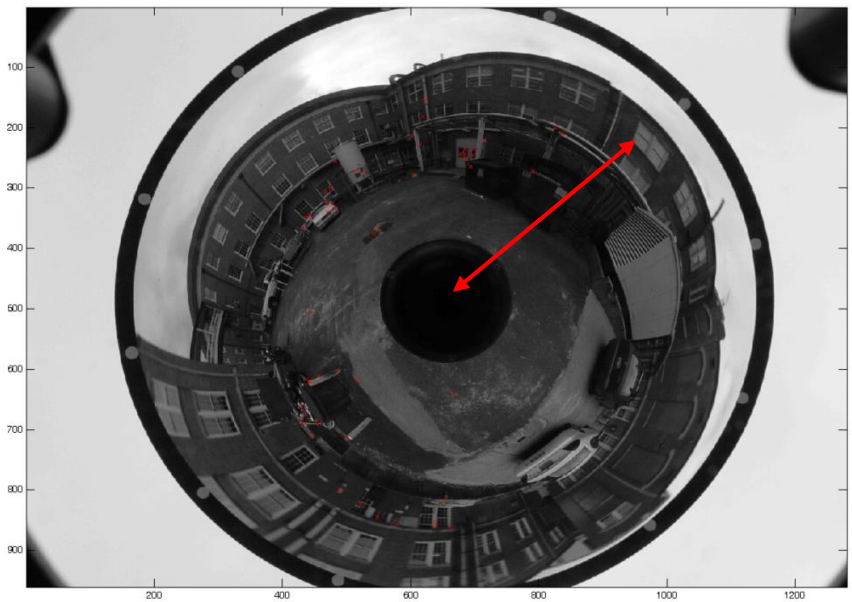
- Models fish-eye lenses, cata-dioptric systems, etc.

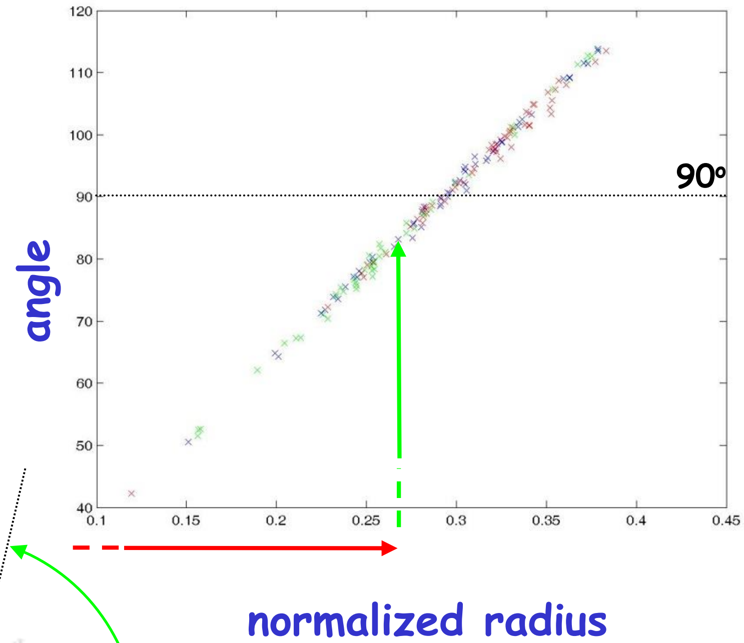


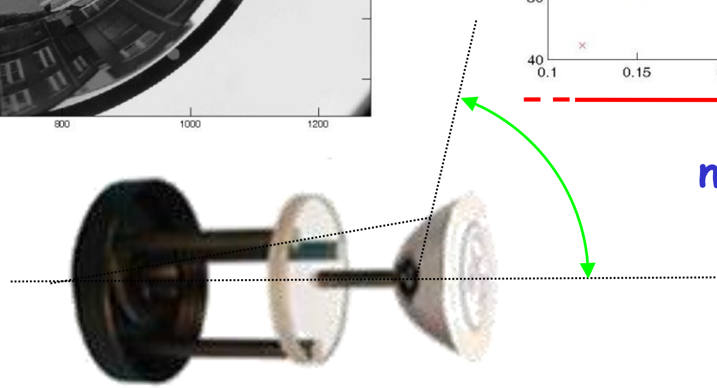
Non-parametric distortion calibration

(Thirthala and Pollefeys ICCV05)

- Models fish-eye lenses, cata-dioptric

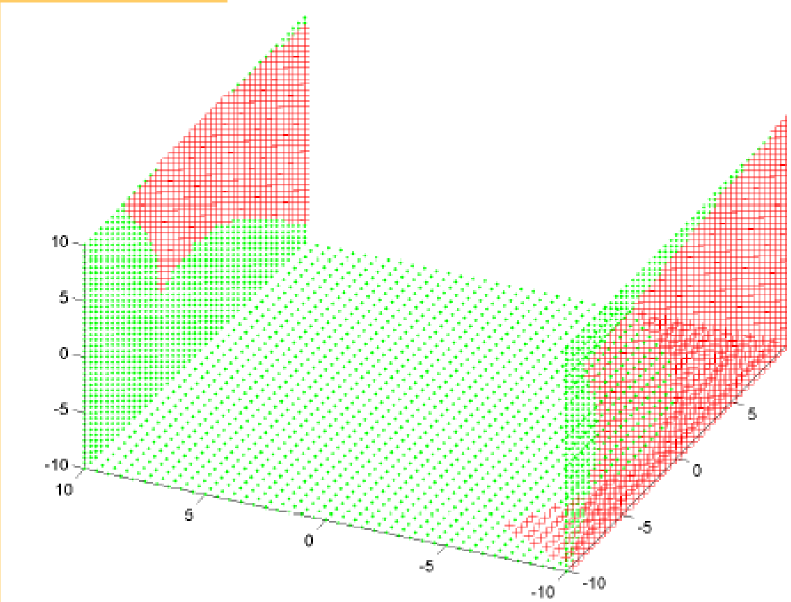


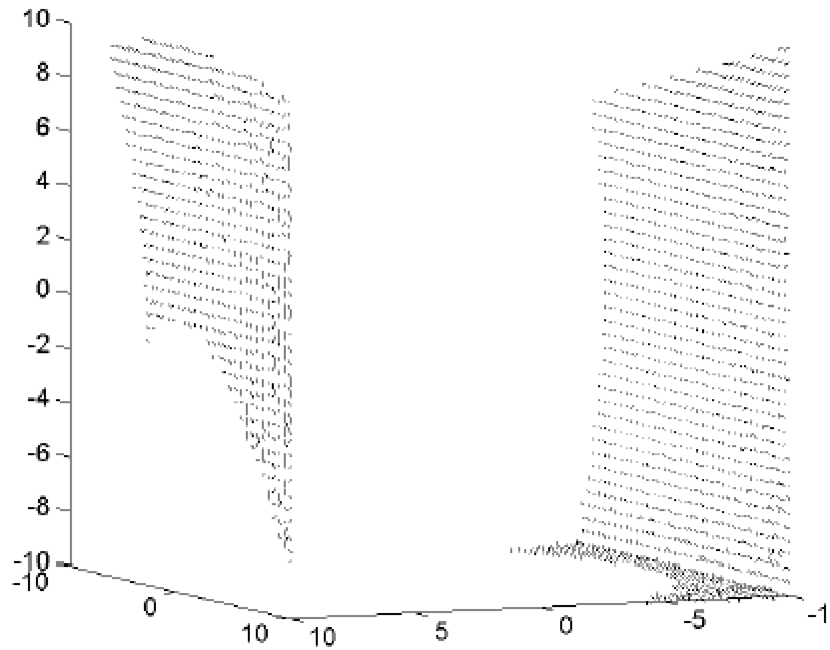




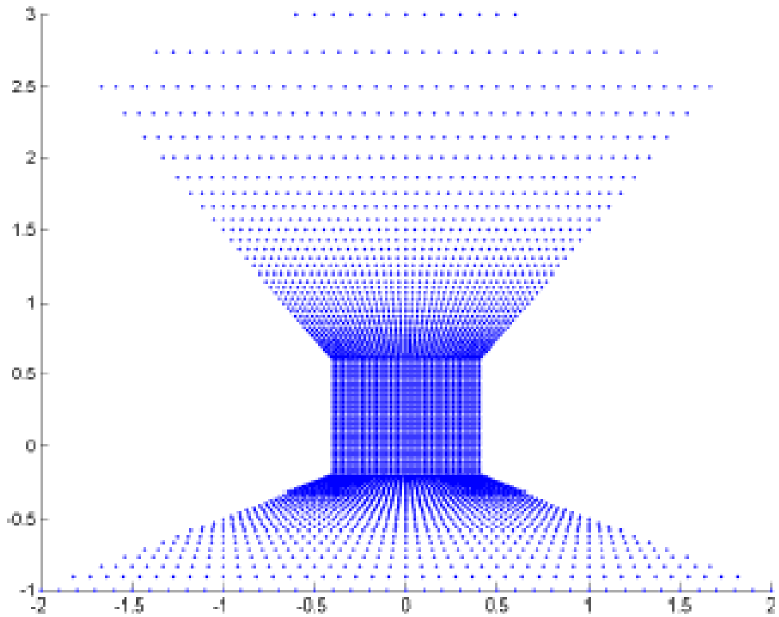
Synthetic quadrifocal tensor example

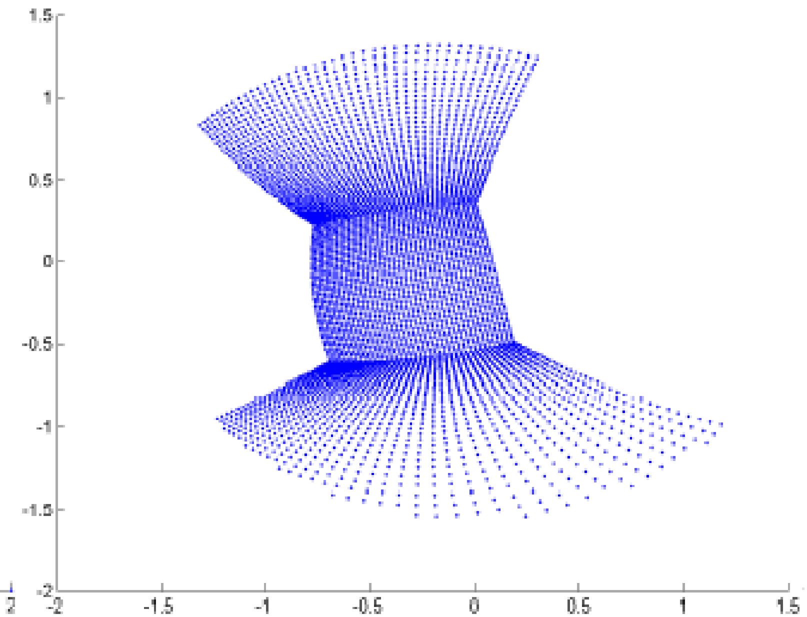
- Perspective
- Fish-eye
- Spherical mirror
- Hyperbolic mirror





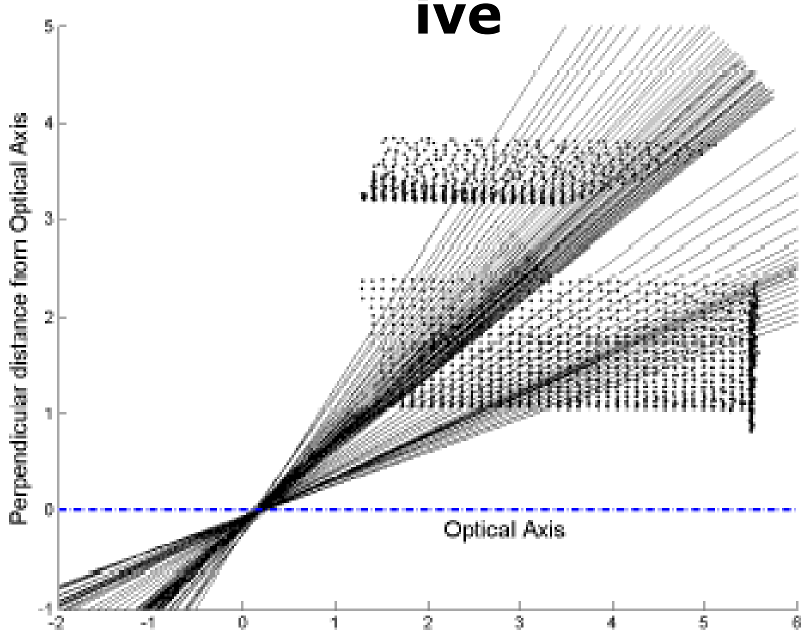
$$\begin{bmatrix} \mathbf{P}^T\mathbf{l} & \mathbf{P}'^T\mathbf{l}' & \mathbf{P}''^T\mathbf{l}'' & \mathbf{P}'''^T\mathbf{l}''' \end{bmatrix}^T$$

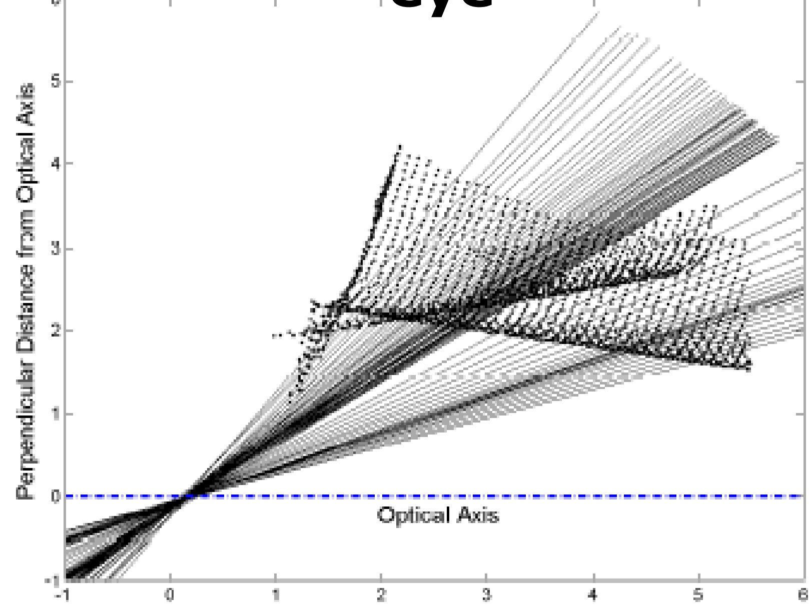


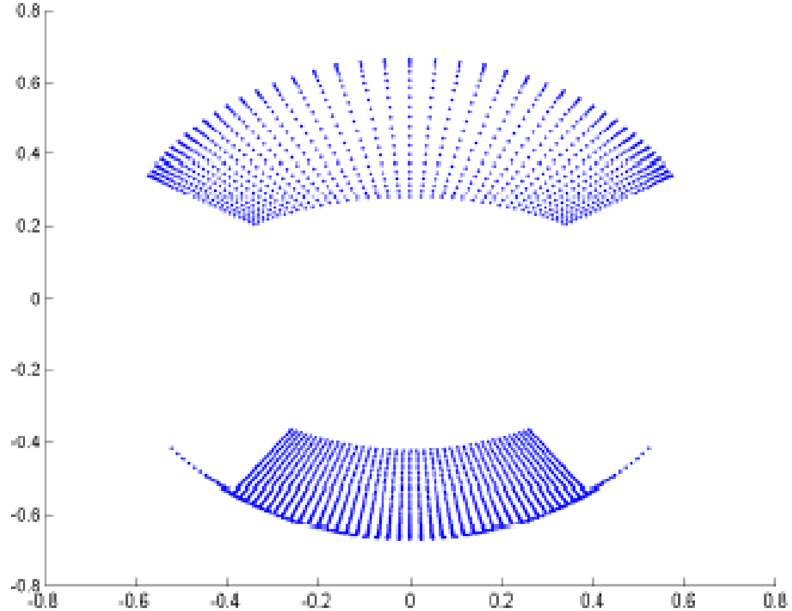


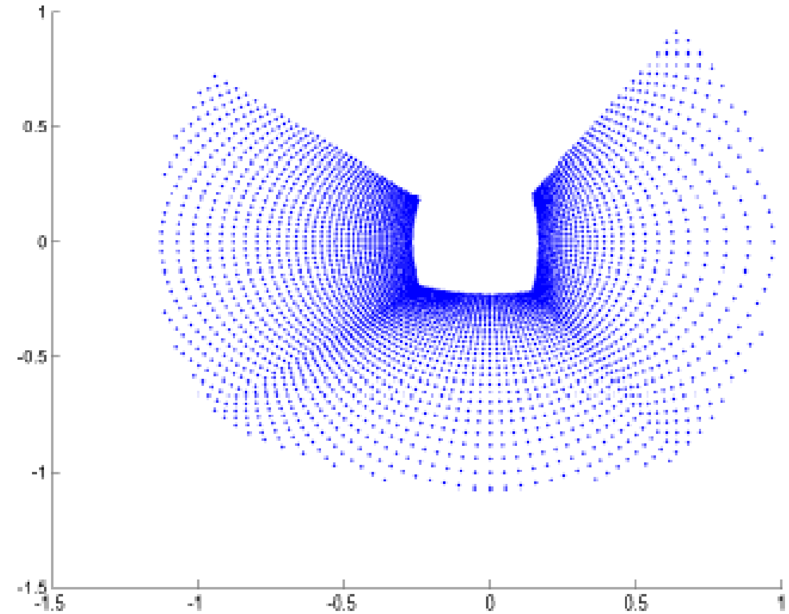
Perspective

Fish-eye



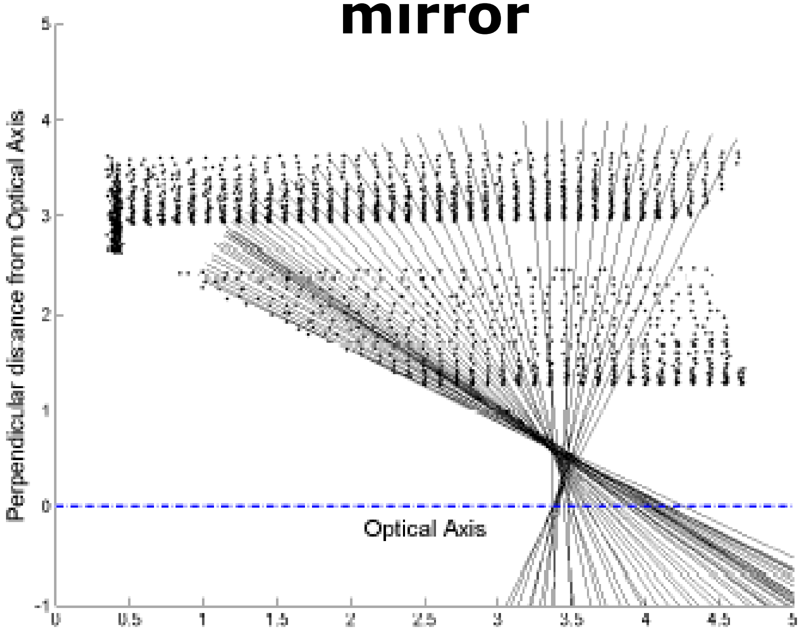


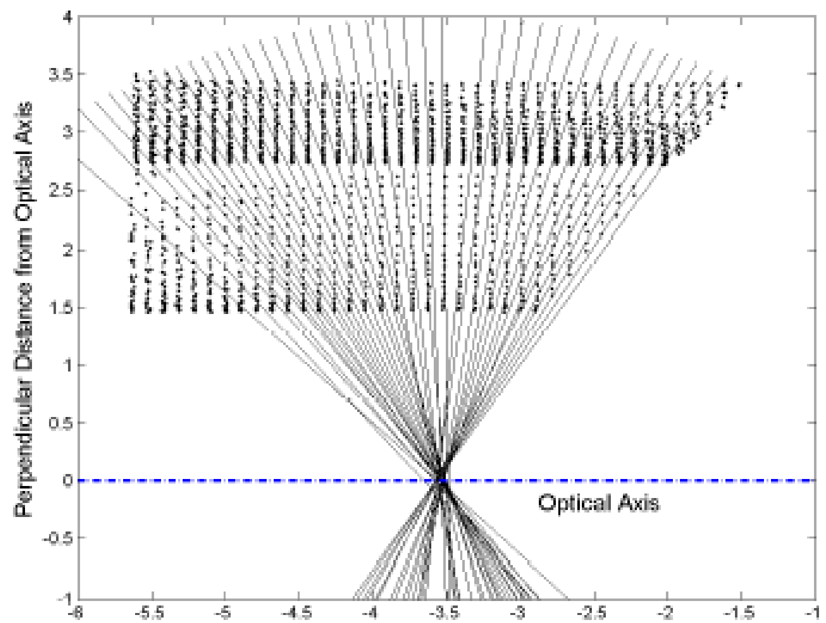




Spherical mirror

Hyperbolic mirror





Next week

Model fitting (RANSAC, EM,...)