# Multiple View Geometry II

CS 534: Introduction to Computer Vision

Rutgers University

TA: Ali Elqursh

Spring 2010

## Sources

- Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)
- Medioni, G. and Kang, S. B. 2004 *Emerging Topics in Computer Vision. Prentice Hall PTR.*
- Slides for Trifocal Tensor by Marc Pollefeys

# Agenda

- Applications
- Projective Geometry
- Structure and Motion problem
  - Two Views
  - Three Views
  - N-Views

# Applications

- 3D graphical models
  - Computing the structure
- Video Augmentation
  - Must compute camera motion, no need to compute structure
  - http://www.youtube.com/watch?v=ZKw_Mp5YkaE
  - You have already done it !

# Applications

- Generating novel views from existing photographs

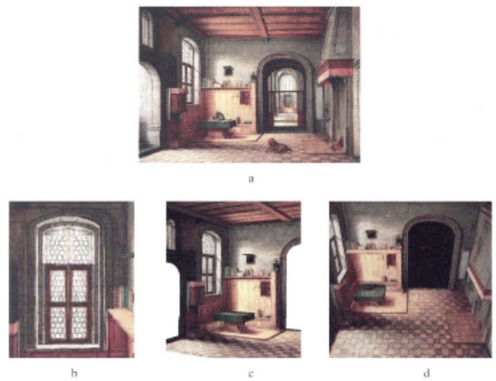



Fig. 1.2. **Single view reconstruction.** *(a) Original painting – St. Jerome in his study, 1630, Hendrick van Steenwijck (1580-1649), Joseph R. Ritman Private Collection, Amsterdam, The Netherlands. (b)(c)(d) Views of the 3D model created from the painting. Figures courtesy of Antonio Criminisi.*

# Applications

- Panoramic mosaicing

Fig. 8.9. **Planar panoramic mosaicing.** *Eight images (out of thirty) acquired by rotating a camcorder about its centre. The thirty images are registered (automatically) using planar homographies and composed into the single panoramic mosaic shown. Note the characteristic "bow tie" shape resulting from registering to an image at the middle of the sequence.*

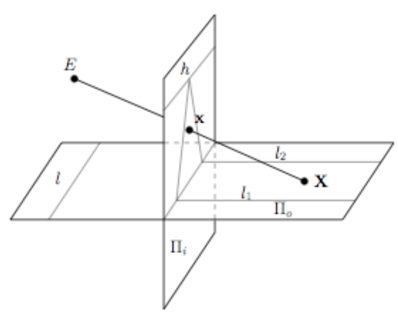
# Applications

- Measuring heights, angles


Fig. 8.21. **Height measurements using affine properties.** *(a) The original image. We wish to measure the height of the two people. (b) The image after radial distortion correction (see section 7.4(p189)). (c) The vanishing line (shown) is computed from two vanishing points corresponding to horizontal directions. The lines used to compute the vertical vanishing points are also shown. The vertical vanishing point is not shown since it lies well below the image. (d) Using the known height of the filing cabinet on the left of the image, the absolute height of the two people are measured as described in algorithm 8.1. The measured heights are within 2cm of ground truth. The computation of the uncertainty is described in [Criminisi-00].*

# Projective Geometry
# Central Perspective Transformation

- Consider mappings of points using central perspective transformation
- Points on $\Pi_i$, $\Pi_0$ can be represented a $\mathbf{R}^2$
- Problems
  - Points on l do not map to any points on $\Pi_i$
  - Points on h do not corresponds to any points on $\Pi_0$
- Solution
  - Append an extra line to $\Pi_i$ that are images of points on l ($\mathbf{R}^2$ + extraline)
  - Append an extra line to $\Pi_0$ that are points that are imaged at h ($\mathbf{R}^2$ + extraline)



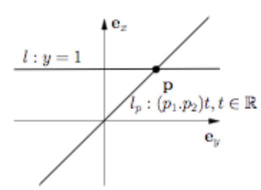
# Projective Geometry
# Projective Space

- Projective Plane **P**$^2$ $\qquad \mathbb{P}^2 = \mathbb{R}^2 \cup \{\text{ideal points}\}$ .
  - Ideal line $l_\infty$ or line at infinity
    $$l_\infty = \{\text{ideal points}\} \ .$$
  - Two different points define a line
  - Two different line intersect in a point

- Projective Line (**P**$^1$) $\qquad \mathbb{P}^1 = \mathbb{R}^1 \cup \{\text{ideal point}\}$ .
- Projective Space (**P**$^3$) $\qquad \mathbb{P}^3 = \mathbb{R}^3 \cup \{\text{ideal points}\}$ .
  - Ideal points in **P**$^3$ builds a plane
  - We call it *ideal plane* Or *plane at infinity*

# Projective Geometry
# Homogeneous coordinates

- Algebraic tool to represent entities in projective geometry
- Consider intersections of vectors through the origin and the line y=1
- We say $(p_1,p_2)$ equivalent to $(q_1,q_2)$ if $(p_1,p_2)= \lambda(q_1,q_2)$
- One-One mapping between equivalent vectors and intersection points

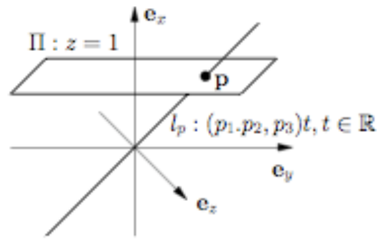



- Use vectors as representation of points on the line
- Every vectors intersects except (1,0)
- It must intersect at (1,0) , ideal point
- Call these vectors “homogeneous coordinates” of points on l

$$\mathbb{P}^1 = \{(p_1, 1) \in \mathbb{P}^1\} \cup \{(p_1, 0) \in \mathbb{P}^1\}$$

# Projective Geometry

- Generalize to $\mathbf{P}^2$
- Plane Π
- Represent each point p as intersections of $(p_1,p_2,p_3)$ with Π
- $(p_1,p_2,p_3)$ equivalent to $(q_1,q_2,q_3)$ if $(p_1,p_2,p_3) = \lambda(q_1,q_2,q_3)$




$$\mathbb{P}^2 = \{(p_1, p_2, 1) \in \mathbb{P}^2\} \cup \{(p_1, p_2, 0) \in \mathbb{P}^2\}$$

- Try to imagine the line at infinity

# Projective Geometry

- Generalize to $\mathbf{P}^3$
- We get a plane at infinity

# Projective Geometry
# Projective Transformations

- A linear transformation in homogeneous coordinates

$$\mathbf{x}' \sim H\mathbf{x}$$

- H does not have to be square
- If it is square mapping from $\mathbf{P}^n$ to $\mathbf{P}^n$ , we call it a homography
- A projective transformation can move the plane at infinity!

# Projective Geometry
# Duality

- For each statement in Projective geometry there is a dual statement

$$l = \{\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{P}^2 \mid \mathbf{x} = t_1\mathbf{p}_1 + t_2\mathbf{p}_2, \quad (t_1, t_2) \in \mathbb{P}^1\} \ .$$

$$l = \{\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{P}^2 \mid n_1x_1 + n_2x_2 + n_3x_3 = 0\}$$

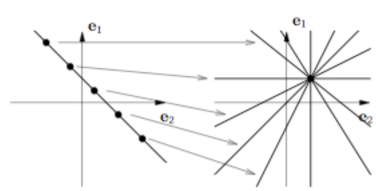



**Figure 3.5.** Duality of points and lines in $\mathbb{P}^2$.

- Given $\mathbf{n} = (n_1, n_2, n_3)$, the points $\mathbf{x} = (x_1, x_2, x_3)$ that fulfills (3.1) constitutes the line defined by $\mathbf{n}$.
- Given $\mathbf{x} = (x_1, x_2, x_3)$, the lines $\mathbf{n} = (n_1, n_2, n_3)$ that fulfills (3.1) constitutes the lines coincident by $\mathbf{x}$.

# Projective Geometry
# Conics

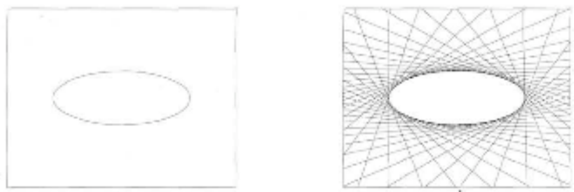
**Definition 14.** A **conic**, $c$, in $\mathbb{P}^2$ is defined as

$$c = \{\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{P}^2 \mid \mathbf{x}^T C \mathbf{x} = 0\} \ ,$$

**Theorem 4.** *The dual, $c^*$, to a conic $c : \mathbf{x}^T C \mathbf{x}$ is the set of lines*

$$\{\mathbf{l} = (l_1, l_2, l_3) \in \mathbb{P}^2 \mid \mathbf{l}^T C' \mathbf{l} = 0\} \ ,$$

*where $C' = C^{-1}$.*



a b

Fig. 2.2. *(a) Points $\mathbf{x}$ satisfying $\mathbf{x}^\mathsf{T} \mathtt{C} \mathbf{x} = 0$ lie on a point conic. (b) Lines $\mathbf{l}$ satisfying $\mathbf{l}^\mathsf{T} \mathtt{C}^* \mathbf{l} = 0$ are tangent to the point conic $\mathtt{C}$. The conic $\mathtt{C}$ is the envelope of the lines $\mathbf{l}$.*

# Projective Geometry
# Conics

**Degenerate conics.** If the matrix $\mathtt{C}$ is not of full rank, then the conic is termed degenerate. Degenerate point conics include two lines (rank 2), and a repeated line (rank 1).

**Definition 18.** The (singular, complex) conic, $\Omega$, in $\mathbb{P}^n$ defined by

$$x_1^2 + x_1^2 + \ldots + x_n^2 = 0 \quad \text{and} \quad x_{n+1} = 0$$

is called the **absolute conic**. ∎

Observe that the absolute conic is located at the plane at infinity, it contains only complex points and it is singular.

**Lemma 2.** *The dual to the absolute conic, denoted $\Omega'$, is given by the set of planes*

$$\Omega' = \{\Pi = (\Pi_1, \Pi_2, \ldots \Pi_{n+1}) \mid \Pi_1^2 + \ldots + \Pi_n^2 = 0 \ .$$

In matrix form $\Omega'$ can be written as $\Pi^T C' \Pi = 0$ with

$$C' = \begin{bmatrix} I_{n \times n} & 0_{n \times 1} \\ 0_{1 \times n} & 0 \end{bmatrix} \ .$$

# Projective Geometry
# Conics

**Lemma 2.** *The dual to the absolute conic, denoted $\Omega'$, is given by the set of planes*

$$\Omega' = \{\Pi = (\Pi_1, \Pi_2, \ldots \Pi_{n+1}) \mid \Pi_1^2 + \ldots + \Pi_n^2 = 0 \ .$$

In matrix form $\Omega'$ can be written as $\Pi^T C' \Pi = 0$ with

$$C' = \begin{bmatrix} I_{n\times n} & 0_{n\times 1} \\ 0_{1\times n} & 0 \end{bmatrix} \ .$$

**Proposition 3.** *The subgroup, $\mathcal{K}$, of projective transformations, $\mathcal{G}_P$, that preserves the absolute conic consists exactly of the projective transformations of the form (3.4), with*

$$H = \begin{bmatrix} cR_{n\times n} & t_{n\times 1} \\ 0_{1\times n} & 1 \end{bmatrix} \ ,$$

*where $0 \neq c \in \mathbb{R}$ and $R$ denote an orthogonal matrix, i.e. $RR^T = R^T R = I$.*

# Projective Geometry
# Quadrics

- In $\mathbf{P}^3$ a conic is called a quadric

**Lemma 3.** *The projection of a quadric, $\mathbf{X}^T C \mathbf{X} = 0$ (dually $\Pi^T C' \Pi = 0$, $C' = C^{-1}$), is an image conic, $\mathbf{x}^T c \mathbf{x} = 0$ (dually $l^T c' l = 0$, $c' = c^{-1}$), with $c' = PC'P^T$.*

**Proposition 6.** *The image of the absolute conic is given by the conic $\mathbf{x}^T \omega \mathbf{x} = 0$ (dually $l^T \omega' l = 0$), where $\omega' = KK^T$.*

- Proof

$$\omega' \sim P\Omega' P^T \sim KR^T \begin{bmatrix} I & -t \end{bmatrix} \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \begin{bmatrix} I \\ -t^T \end{bmatrix} RK^T = KR^T RK^T = KK^T$$

# Structure and Motion Problem

- Given:
  - sequence of images
  - corresponding feature points

$$\lambda_{ij} x_{ij} = P_i X_j$$

- Determine
  - Camera Matrices $P_i$ (Motion)
  - 3D Points $X_j$ (Structure)
- Important Results
  - Given un-calibrated image sequence, and without any assumptions, it is only possible to reconstruct the object up to an unknown projective transformation
  - Given calibrated camera then it is possible to reconstruct the scene up to an unknown similarity transformation
  - For some applications, projective reconstruction may be sufficient !

# Structure and Motion Problem

**Theorem 5.** *Given an un-calibrated image sequence with corresponding points, it is only possible to reconstruct the object up to an unknown projective transformation.*

*Proof:* Assume that $\mathbf{X}_j$ is a reconstruction of $n$ points in $m$ images, with camera matrices $P_i$ according to

$$\mathbf{x}_{ij} \sim P_i \mathbf{X}_j, \; i = 1, \ldots m, \; j = 1, \ldots n \; .$$

Then $H\mathbf{X}_j$ is also a reconstruction, with camera matrices $P_i H^{-1}$, for every non-singular $4 \times 4$ matrix $H$, since

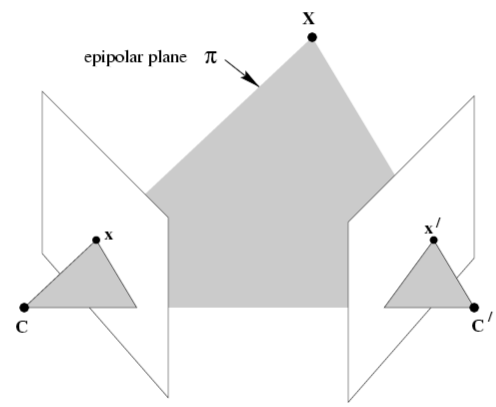
$$\mathbf{x}_{ij} \sim P_i \mathbf{X}_j \sim P_i H^{-1} H \mathbf{X}_j \sim (P_i H^{-1})(H\mathbf{X}_j) \; .$$

The transformation

$$\mathbf{X} \mapsto H\mathbf{X}$$

corresponds to all projective transformations of the object.
In the same way it can be shown that if the cameras are calibrated, then it is possible to reconstruct the scene up to an unknown similarity transformation.

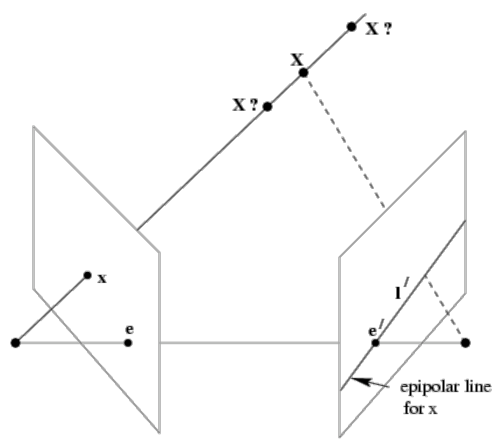
# Two Views
# Epipolar Geometry




C,C’,x,x’ and X are coplanar

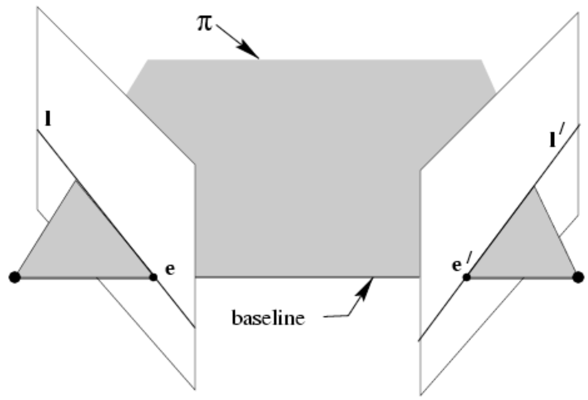


# Two Views
# Epipolar Geometry



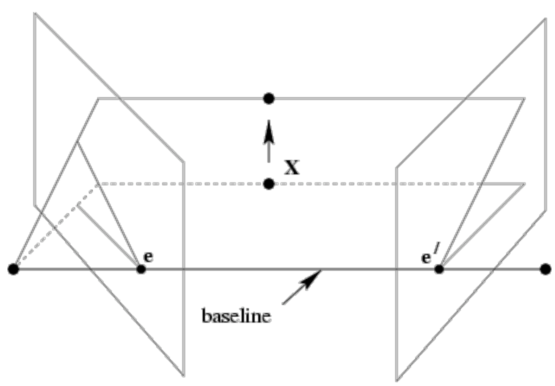

What if only C,C’,x are known?

# Two Views
# Epipolar Geometry




All points on π project on l and l'

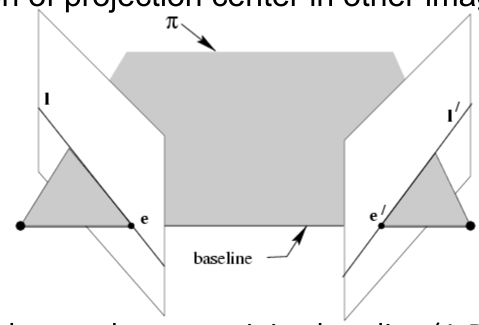
# Two Views
# Epipolar Geometry




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

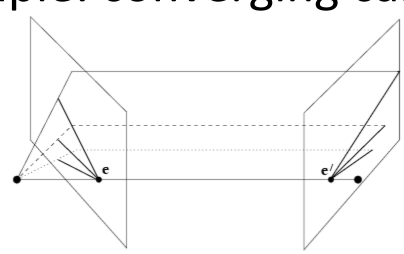
# Two Views
# Epipolar Geometry

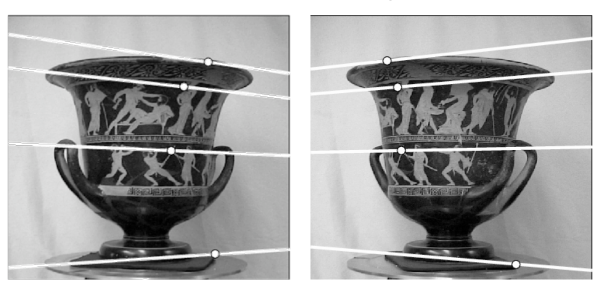
epipoles e,e'
= intersection of baseline with image plane
= projection of projection center in other image



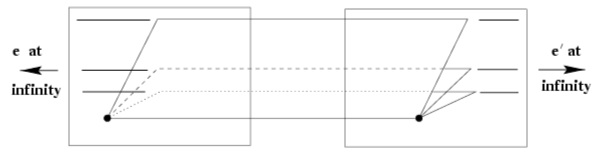

an epipolar plane = plane containing baseline (1-D family)
an epipolar line = intersection of epipolar plane with image
(always come in corresponding pairs)

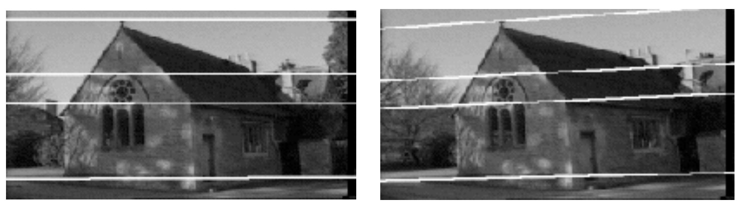
# Two Views
# Example: converging cameras



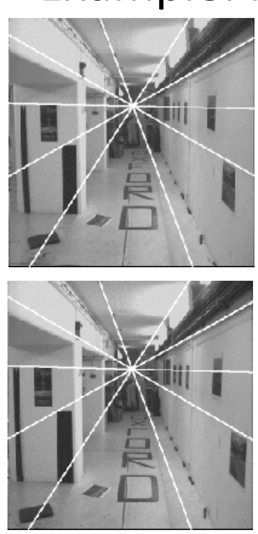



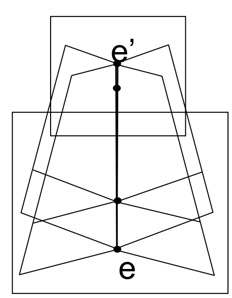
# Two Views
## Example: motion parallel with image plane



# Two Views
## Example: Forward Motion



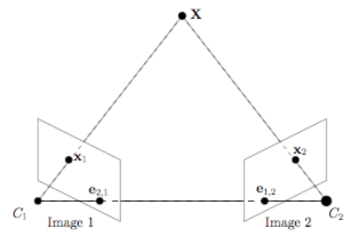


# Two Views

- Epipole $e_{i,j}$ is the projections of the camera center of camera i in image j

$$P_1 = [\, A_1 \mid b_1 \,] \quad and \quad P_2 = [\, A_2 \mid b_2 \,] \ .$$

$$\mathbf{e}_{1,2} = -A_2 A_1^{-1} b_1 + b_2$$




# Two Views
# Canonical Cameras

- Two cameras can be represented as

$$P_1 = [\, A_1 \mid b_1 \,] \quad \text{and} \quad P_2 = [\, A_2 \mid b_2 \,]$$

- It will be useful to use canonical cameras. We can always convert non-canonical cameras to canonical by using the following transformation

$A_{12}$

$$H = \begin{bmatrix} A_1^{-1} & -A_1^{-1} b_1 \\ 0 & 1 \end{bmatrix}$$

$$\bar{P}_1 = P_1 H = [\, I \mid 0 \,] \quad \bar{P}_2 = P_2 H = [\, A_2 A_1^{-1} \mid b_2 - A_2 A_1^{-1} b_1 \,] \ .$$

Notice the epipole

# Two Views
# Canonical Cameras

We can multiply again by without changing $P_1$

$$\bar{H} = \begin{bmatrix} I & 0 \\ \mathbf{v}^T & 1 \end{bmatrix}$$

$$\bar{H}\bar{P}_2 = [\, A_{12} + \mathbf{e}\mathbf{v}^T \mid \mathbf{e} \,]$$

**Definition 26.** A pair of camera matrices is said to be in **canonical form** if

$$P_1 = [\, I \mid 0 \,] \quad \text{and} \quad P_2 = [\, A_{12} + \mathbf{e}\mathbf{v}^T \mid \mathbf{e} \,] \ , \qquad (3.13)$$

where $\mathbf{v}$ denote a three-parameter ambiguity. ■

# Two Views
# Fundamental Matrix

- The fundamental matrix F is an algebraic representation of the epipolar geometry

$$\lambda_1 \mathbf{x}_1 = P_1 \mathbf{X} = [\, A_1 \mid b_1 \,]\mathbf{X}, \quad \lambda_2 \mathbf{x}_2 = P_2 \mathbf{X} = [\, A_2 \mid b_2 \,]\mathbf{X} \ .$$

$$\lambda_1 \mathbf{x}_1 = P_1 \mathbf{X} = [\, A_1 \mid b_1 \,]\mathbf{X} = A_1 \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + b_1 \quad \Rightarrow \quad \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = A_1^{-1}(\lambda_1 \mathbf{x}_1 - b_1)$$

$$\lambda_2 \mathbf{x}_2 = A_2 A_1^{-1}(\lambda_1 \mathbf{x}_1 - b_1) + b_2 = \lambda_1 A_{12} \mathbf{x}_1 + (-A_{12} b_1 - b_2)$$

$$\mathbf{x}_1^T A_{12}^T T_{\mathbf{e}} \mathbf{x}_2 = \mathbf{x}_1^T F \mathbf{x}_2 = 0$$

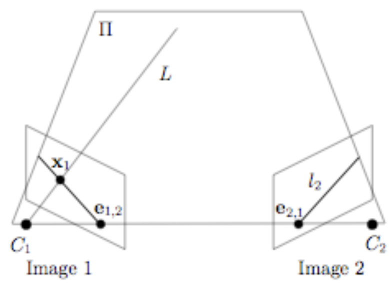
- Epipolar constraint

$$\mathbf{x}_1^T F \mathbf{x}_2 = 0$$

# Two Views
# Fundamental Matrix

- Epipolar line

**Definition 28.** The line $\mathbf{l} = F^T \mathbf{x}_1$ is called the **epipolar line** corresponding to $\mathbf{x}_1$. ■




# Two Views
# Fundamental Matrix

- Fe' = 0 (Since xFe' = 0 for all x)
- $e^T F = 0$ ( Since $e^T Fx' = 0$ for all x')
- F has Rank 2
- det F =0
- Given F we can extract the projection matrices

$$F = A_{12}^T T_{\mathbf{e}} \quad \Leftrightarrow \quad P_1 = [\, I \mid 0 \,], \quad P_2 = [\, A_{12} \mid \mathbf{e} \,] \ .$$

# Two Views
# Fundamental Matrix

Observe that

$$F = A_{12}^T T_e = (A_{12} + \mathbf{e}\mathbf{v}^T)^T T_e$$

for every vector $\mathbf{v}$, since

$$(A_{12} + \mathbf{e}\mathbf{v})^T T_e(\mathbf{x}) = A_{12}^T(\mathbf{e} \times \mathbf{x}) + \mathbf{v}\mathbf{e}^T(\mathbf{e} \times \mathbf{x}) = A_{12}^T T_e \mathbf{x} \ ,$$

since $\mathbf{e}^T(\mathbf{e} \times \mathbf{x}) = \mathbf{e} \cdot (\mathbf{e} \times \mathbf{x}) = 0$. This ambiguity corresponds to the transformation

$$\Pi P_2 = [\ A_{12} + \mathbf{e}\mathbf{v}^T \mid \mathbf{e}\ ] \ .$$

We conclude that there are three free parameters in the choice of the second camera matrix when the first is fixed to $P_1 = [\ I \mid 0\ ]$.



# Two Views
# Fundamental Matrix

- The fundamental matrix has 9 entries
- However it is a homogeneous quantity
- Has 7 dof (9 elements - det F=0 - homogeneous)
- Can be estimated from 8 point correspondences $x_i'^T F x_i = 0$

$$x'x f_{11} + x'y f_{12} + x' f_{13} + y'x f_{21} + y'y f_{22} + y' f_{23} + x f_{31} + y f_{32} + f_{33} = 0. \quad (11.2)$$

$$(x'x, x'y, x', y'x, y'y, y', x, y, 1)\,\mathbf{f} = 0.$$

$$A\mathbf{f} = \begin{bmatrix} x_1'x_1 & x_1'y_1 & x_1' & y_1'x_1 & y_1'y_1 & y_1' & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n'x_n & x_n'y_n & x_n' & y_n'x_n & y_n'y_n & y_n' & x_n & y_n & 1 \end{bmatrix} \mathbf{f} = \mathbf{0}.$$

# Two Views
# Fundamental Matrix

Here the convention is $x'^T F x = 0$ instead of $xFx' = 0$

- $F$ is a rank 2 homogeneous matrix with 7 degrees of freedom.
- **Point correspondence**: If $x$ and $x'$ are corresponding image points, then $x'^T F x = 0$.
- **Epipolar lines**:
  - $l' = Fx$ is the epipolar line corresponding to $x$.
  - $l = F^T x'$ is the epipolar line corresponding to $x'$.
- **Epipoles**:
  - $Fe = 0$.
  - $F^T e' = 0$.
- **Computation from camera matrices** $P, P'$:
  - General cameras, $F = [e']_\times P' P^+$, where $P^+$ is the pseudo-inverse of $P$, and $e' = P'C$, with $PC = 0$.
  - Canonical cameras, $P = [I \mid 0]$, $P' = [M \mid m]$, $F = [e']_\times M = M^{-T}[e]_\times$, where $e' = m$ and $e = M^{-1}m$.
  - Cameras not at infinity $P = K[I \mid 0]$, $P' = K'[R \mid t]$, $F = K'^{-T}[t]_\times R K^{-1} = [K't]_\times K' R K^{-1} = K'^{-T} R K^T [K R^T t]_\times$.

Table 9.1. *Summary of fundamental matrix properties.*



# Two Views
# Essential matrix

- Calibrated camera (K is known)
- $P_1 = K[I|0]$ , $P_2=K'[R|t]$
- $F = K'^{-1}R[t]_x K^{-1}$
- If we use normalized points then
- $E = R[t]_x$
- $x'Ex=x'R[t]_x x=0$

# Two Views
# Retrieving Projection Matrices

E is essential matrix if and only if
two singular values are equal (and third=0)

$$E = U\mathrm{diag}(1,1,0)V^T$$

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

**Result 9.19.** *For a given essential matrix* $E = U\,\mathrm{diag}(1,1,0)V^T$, *and first camera matrix* $P = [I \mid 0]$, *there are four possible choices for the second camera matrix* $P'$, *namely*

$$P' = [UWV^T \mid +u_3] \text{ or } [UWV^T \mid -u_3] \text{ or } [UW^TV^T \mid +u_3] \text{ or } [UW^TV^T \mid -u_3].$$

These four solutions can be disambiguated, Only one will yield reconstructed points in front of the two cameras



# Two Views
# Structure Computation

- We know how to compute P’s for Two views (Compute F , then compute P’s)
- We can define two new sub-problems
  - Resection: Assume that $X_j$’s (Structure) are given, we have $x_{ij}$, calculate $P_i$’s (Motion)
  - Intersection: Assume that $P_i$’s (Motion) are given, we have $x_{ij}$, calculate $X_j$’s (Structure)

# Two Views
# Intersection

- We have these constraints $\begin{cases} \lambda_1 \mathbf{x}_1 = P_1 \mathbf{X}, \\ \lambda_2 \mathbf{x}_2 = P_2 \mathbf{X}, \end{cases}$
- In matrix form $\begin{bmatrix} P_1 & \mathbf{x}_1 & \mathbf{0} \\ P_2 & \mathbf{0} & \mathbf{x}_2 \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ -\lambda_1 \\ -\lambda_2 \end{bmatrix} = \mathbf{0}$
- Can be computed linearly (6 constraints, 6 unknowns)

# N-Views
# Affine Factorization

- Assume cameras are affine
  - Write here how an affine camera looks like
- Compute both structure and motion at the same time

$$W = \begin{bmatrix} \mathbf{x}_1^1 & \mathbf{x}_2^1 & \dots & \mathbf{x}_n^1 \\ \mathbf{x}_1^2 & \mathbf{x}_2^2 & \dots & \mathbf{x}_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_1^m & \mathbf{x}_2^m & \dots & \mathbf{x}_n^m \end{bmatrix} \qquad W = \begin{bmatrix} M^1 \\ M^2 \\ \vdots \\ M^m \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \dots & \mathbf{X}_n \end{bmatrix}.$$

- With noise → equations not satisfied exactly
- Seek $W^{hat}$ that is closest to W in Frobenius norm

$$\left\| W - \hat{W} \right\|_F^2 = \sum_{ij} \left( W_{ij} - \hat{W}_{ij} \right)^2 = \sum_{ij} \left\| \mathbf{x}_j^i - \hat{\mathbf{x}}_j^i \right\|^2 = \sum_{ij} \left\| \mathbf{x}_j^i - M^i \mathbf{X}_j \right\|^2$$

# N-Views
# Affine Factorization

This can be achieved by SVD

Note $W^{hat}$ must be rank 3

W = USVT , What = $U_{2mx3}D_{3x3}V^T_{3xn}$

Remember, we can only compute Structure and Motion up to projective transformation

One solution M = $U_{2mx3}D_{3x3}$ , X=$V^T_{3xn}$

Another solution M = $U_{2mx3}$ , X =$D_{3x3}V^T_{3x3}$

# N-Views
# Projective Factorization

- In general we need to do projective factorization

$$\begin{bmatrix} \lambda_1^1 \mathbf{x}_1^1 & \lambda_2^1 \mathbf{x}_2^1 & \dots & \lambda_n^1 \mathbf{x}_n^1 \\ \lambda_1^2 \mathbf{x}_1^2 & \lambda_2^2 \mathbf{x}_2^2 & \dots & \lambda_n^m \mathbf{x}_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1^m \mathbf{x}_1^m & \lambda_2^m \mathbf{x}_2^m & \dots & \lambda_n^m \mathbf{x}_n^m \end{bmatrix} = \begin{bmatrix} \mathrm{P}^1 \\ \mathrm{P}^2 \\ \vdots \\ \mathrm{P}^m \end{bmatrix} [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n] .$$

- Need to compute λ's, P's, and X's
- Iterative process using Expectation-Maximization (EM) Algorithm
- Must choose initial values for λ's

# N-Views
# Projective Factorization

**Objective**

Given a set of $n$ image points seen in $m$ views:

$$\mathbf{x}_j^i;\ i = 1, \ldots, m,\ j = 1, \ldots, n$$

compute a projective reconstruction.

**Algorithm**

(i) Normalize the image data using isotropic scaling as in section 4.4.4($p$107).
(ii) Start with an initial estimate of the projective depths $\lambda_j^i$. This may be obtained by techniques such as an initial projective reconstruction, or else by setting all $\lambda_j^i = 1$.
(iii) Normalize the depths $\lambda_j^i$ by multiplying rows and columns by constant factors. One method is to do a pass setting the norms of all rows to 1, then a similar pass on columns.
(iv) Form the $3m \times n$ measurement matrix on the left of (18.9), find its nearest rank-4 approximation using the SVD and decompose to find the camera matrices and 3D points.
(v) **Optional iteration.** Reproject the points into each image to obtain new estimates of the depths and repeat from step (ii).

Algorithm 18.2. *Projective reconstruction through factorization.*

# N-Views
# Bundle Adjustment

- Nonlinear optimization
- Minimizes a geometric distance

$$\min_{\hat{\mathrm{P}}^i, \hat{\mathbf{X}}_j} \sum_{ij} d(\hat{\mathrm{P}}^i \hat{\mathbf{X}}_j, \mathbf{x}_j^i)^2$$

- Can be performed very efficiently by using sparse optimization
- Requires good initialization