# 5. Stereo

## Computer Vision

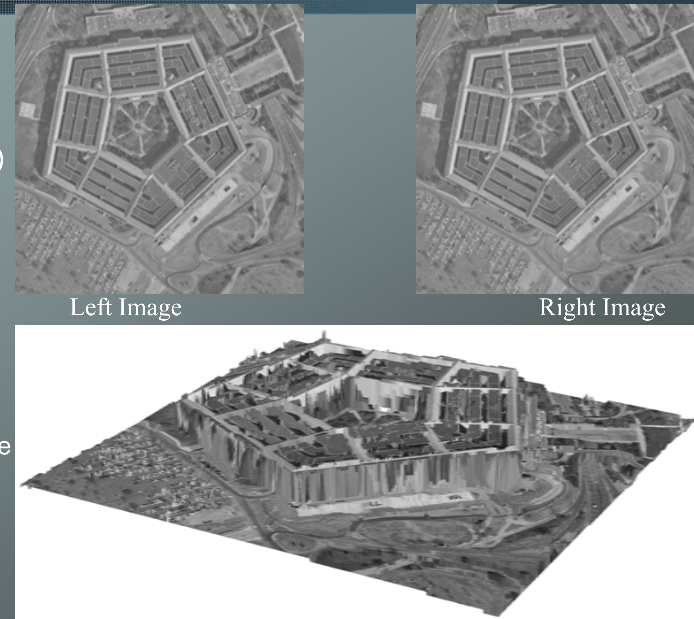
**Zoltan Kato**

http://www.inf.u-szeged.hu/~kato/



# Binocular Stereo

- A way of getting depth (3D) information about a scene from two 2D views (images) of the scene
- Used by humans
- Computational stereo vision
  - Programming machines to do stereo vision
  - Studied extensively in the past 25 years
  - Difficult; still being researched








# Three geometric questions

1. **Correspondence geometry:** Given an image point **x** in the first view, how does this constrain the position of the corresponding point **x'** in the second image?
2. **Camera geometry (motion):** Given a set of corresponding image points $\{x_i \leftrightarrow x'_i\}, i=1,\dots,n$, what are the camera matrixes **P** and **P'** for the two views?
3. **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?

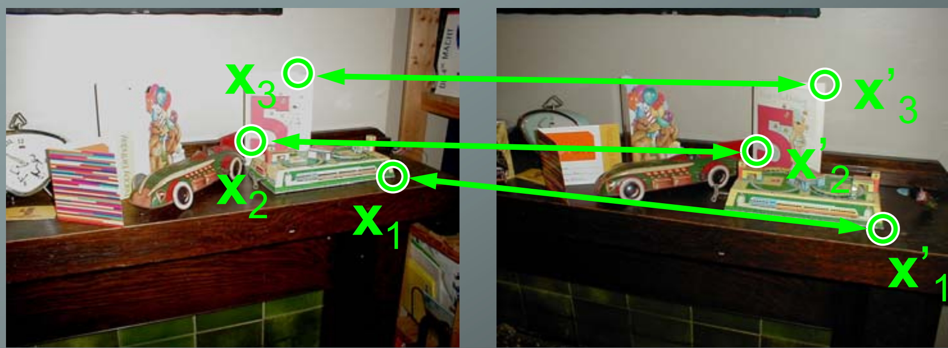




# Mapping Points between Images

- What is the relationship between the images **x**, **x'** of the scene point **X** in two views?
- Intuitively, it depends on:
  - The rigid transformation (motion) between cameras (derivable from the camera matrices **P**, **P'**)
  - The scene structure (i.e., the depth of **X**)
    - Parallax: Closer points appear to move more

# Example: Two-View Geometry




courtesy of F. Dellaert

Is there a transformation relating the points $\mathbf{x}_i$ to $\mathbf{x}'_i$ ?



# Slides adopted from

## CS 395/495-26: Spring 2004

## IBMR:

## 2-D Projective Geometry

## --Introduction--

Jack Tumblin
jet@cs.northwestern.edu

# 2-D Homogeneous Coordinates

**WHAT?!** Why $x_3$? Why 'default' value of 1?

- Look at lines in $\mathbf{R}^2$ :
  - 'line' == all (x,y) points where $ax + by + c = 0$
  - scale by 'k' →→ no change: $kax + kby + kc = 0$
- Using '$x_3$' for points **UNIFIES** notation:
  - line is a 3-vector named $\mathbf{l}$
  - now point (x,y) is a 3-vector too, named $\mathbf{x}$

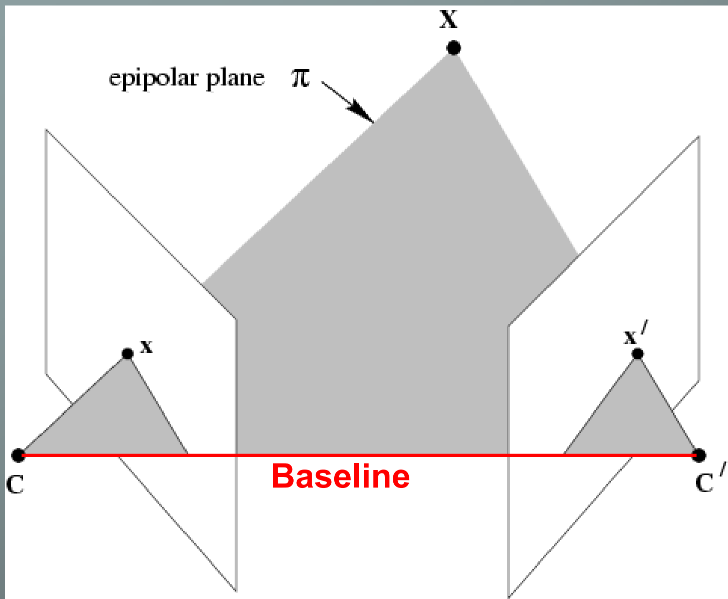
$$ax + by + c = 0 \qquad \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \mathbf{0} \qquad \mathbf{x}^T \cdot \mathbf{l} = \mathbf{0}$$

# 2D Homogeneous Coordinates

## Important Properties 1 (see book for details)

- **3 coordinates, but only 2 degrees of freedom**
  (only 2 ratios $(x_1 / x_3)$, $(x_2 / x_3)$ can change)
- **DUALITY: points, lines are interchangeable**
  - Line Intersections = point: $\mathbf{l}_1 \times \mathbf{l}_2 = \mathbf{x}$
  - Point 'Intersections' = line: $\mathbf{x}_1 \times \mathbf{x}_2 = \mathbf{l}$ (a 3D cross-product)
  - Projective theorem for lines ←→ theorem for points!

# Epipolar geometry



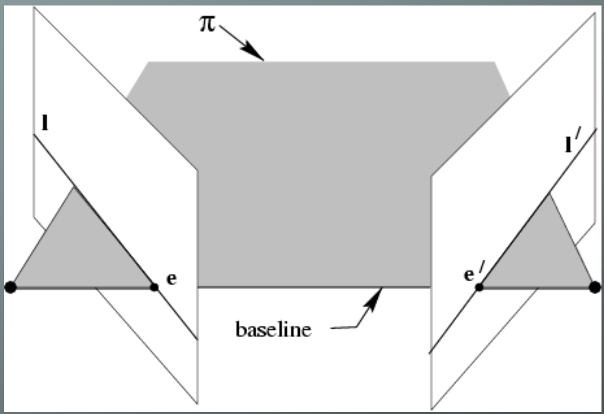

- The fundamental constraint in stereo
- ***Baseline***: Line joining camera centers **C**, **C'**
- **C**,**C'**,**x**,**x'** and **X** are coplanar





# Epipolar lines

- **Epipolar lines l, l'**:
  - Intersection of epipolar plane π with image planes
  - The image in one view of the other camera's projection ray.
- **Epipoles e, e'**:
  - Where baseline intersects image planes
  - The image in one view of the other camera center.
  - Intersection of the epipolar lines
  - Vanishing point of camera motion direction

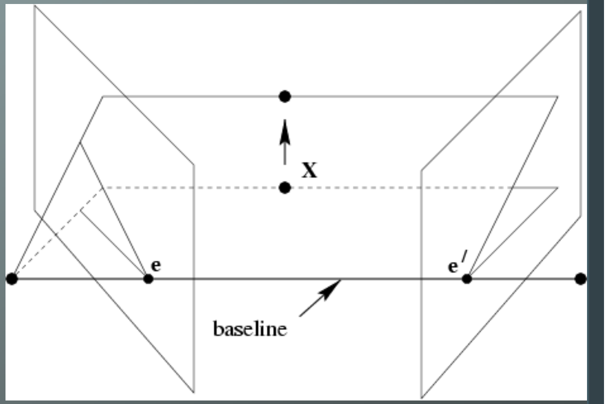







# Epipolar pencil

- As position of **X** varies, epipolar planes "rotate" about the baseline
  - This set of planes is called the **epipolar pencil**
- Epipolar lines "radiate" from epipole—this is the **pencil of epipolar lines**

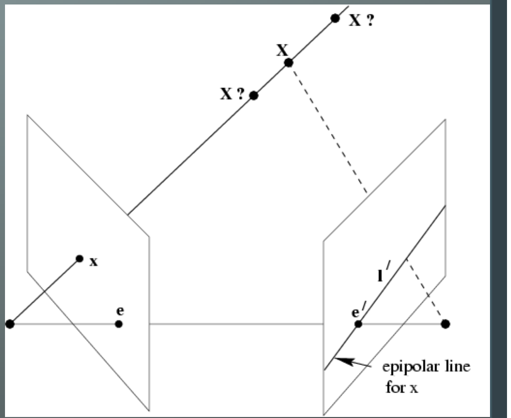




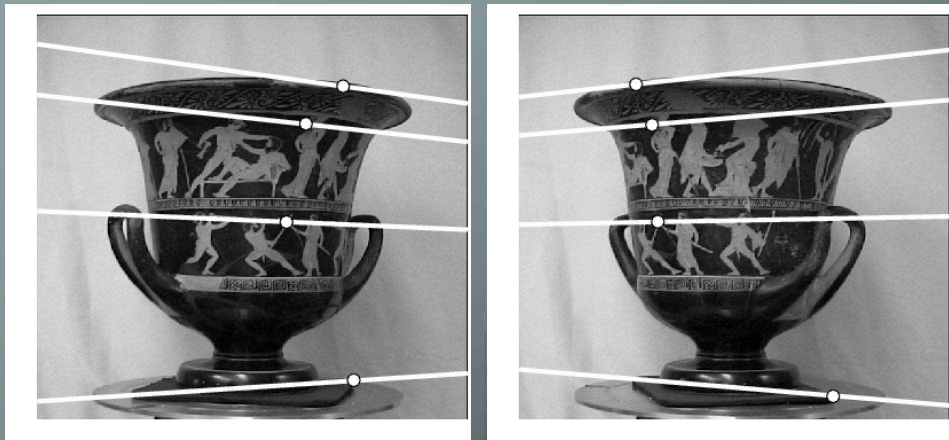



# Epipolar constraint

- Camera center **C** and image point **x** define a ray in 3D space that projects to the epipolar line **l'** in the other view (since it's on the epipolar plane)
- 3D point **X** is on this ray ➔ image of **X** in other view **x'** must be on **l'**.
- In other words, the epipolar geometry defines a mapping **x ➔ l'** of points in one image to lines in the other

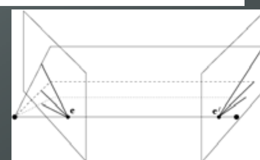


## Example: Epipolar Lines for Converging Cameras



Left view

Right view

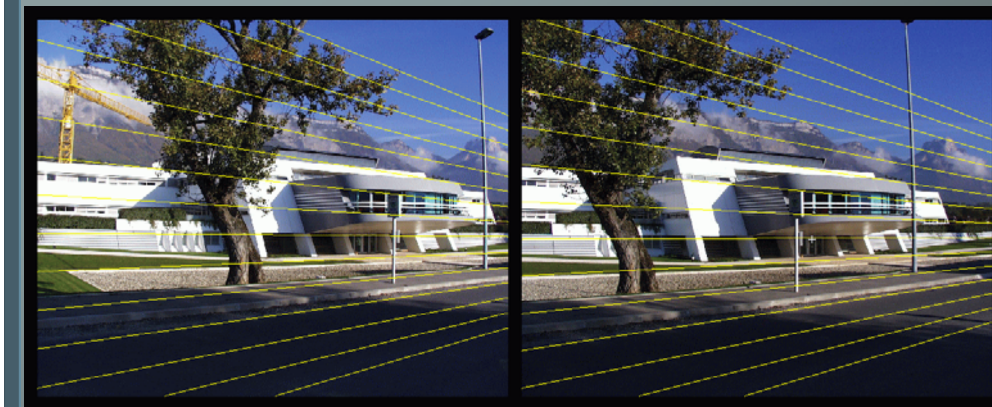


- Intersection of epipolar lines = Epipole !
- Indicates location of other camera center





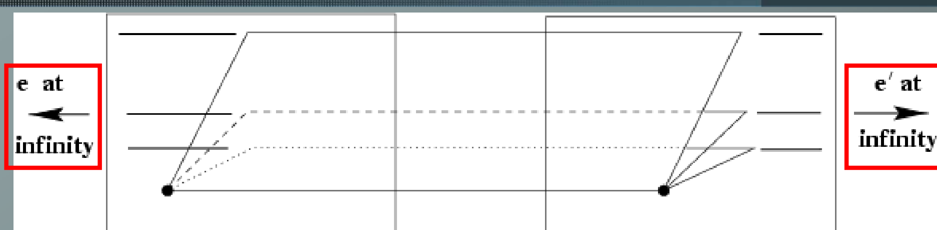
## Example: Epipolar Lines for Translating Cameras







## Special case: aligned image planes

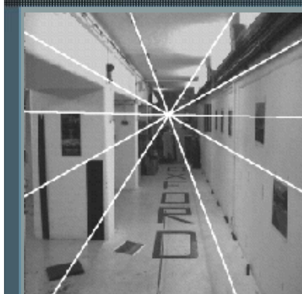


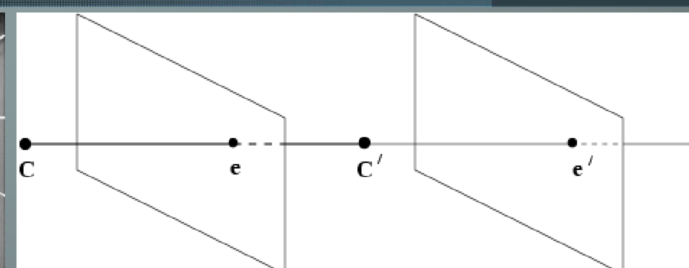

- epipolar lines are parallel
- epipolar lines correspond to rows in the image
- epipoles in both images are at infinity along the x axis.



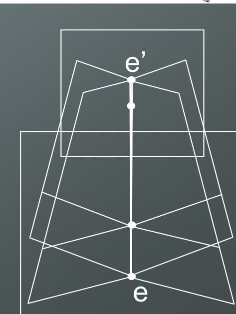


## Special Case: Translation along Optical Axis






- Epipoles coincide at ***focus of expansion***
- Not the same (in general) as vanishing point of scene lines



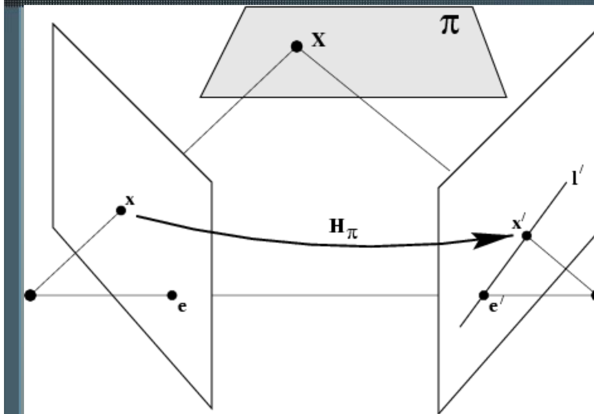
# The Fundamental Matrix (F)

- Mapping a point in one image to epipolar line in other image **x→l'** is expressed algebraically by the **Fundamental Matrix F**
- Write this as **l'=Fx**
- **F** is
  - 3 x 3
  - rank 2 (not invertible, in contrast to homographies)
  - 7 DOF (homogeneity and rank constraint -2 DOF)





# Fundamental Matrix




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

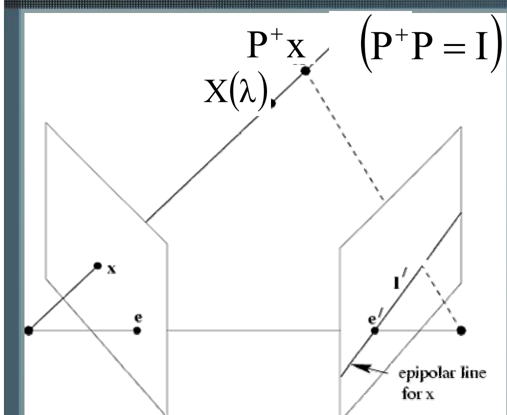
Geometric derivation:

- **F** is a mapping from 2D (plane) to 1D (line) family
  - ➔ **F** is 3X3 but rank 2





# Fundamental Matrix




$$X(\lambda) = P^+x + \lambda C$$
$$l = \underbrace{P'C}_{e'} \times \underbrace{P'P^+x}_{x'}$$
$$F = [e']_\times P'P^+$$

Algebraic derivation:

- Doesn't work for **C=C'** ➔ **F=0**





# Fundamental Matrix

$F = [e']_\times P' P^+$ But what's this? A NEW TRICK:

- Cross Product written as matrix multiply (Zisserman pg. 554)

$$a \times b = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{bmatrix} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = [a]_\times \cdot b$$

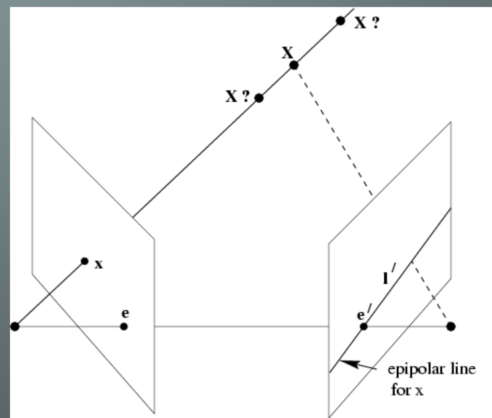
a 'skew symmetric' matrix

- So write: $a \times b = -b \times a = [a]_\times \cdot b = (a^T \cdot [b]_\times)^T$

# Correspondence condition & F

- Since **x'** is on **l'**, by the point-on-line definition we know that **$x'^T l' = 0$**
- Combined with **l'=Fx**, we can thus relate corresponding points in the camera pair **(P,P')** to each other by

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




  - ➔ the fundamental matrix satisfies the above condition for any pair of corresponding points **x↔x'** in the two images
  - The fundamental matrix of **(P',P)** is the transpose **$F^T$**





# Fundamental matrix summary

- **F** is the unique 3X3 rank 2 matrix that satisfies **$x'^T F x = 0$** for all **x↔x'**
  1. **Transpose:** if **F** is fundamental matrix for **(P,P')** , then **$F^T$** is fundamental matrix for **(P',P)**
  2. **Epipolar lines:** **l'=Fx** & **$l = F^T x'$**
  3. **Epipoles:** on all epipolar lines, ➔ **∀x: $e'^T F x = 0$**, ➔ **$e'^T F = 0$**, similarly **Fe=0**
  4. **F** has 7 DOF , i.e. 3X3 - 1(homogeneous) -1(rank2)
  5. **F** is a correlation, projective mapping from a point **x** to a line **l'=Fx** (not a proper correlation, i.e. not invertible)
  6. **F** is unaffected by any proj. transforms done on **BOTH** cameras
     - **(PH, P'H)** has same **F** matrix as **(P, P')** for any full-rank **H**
     - ➔ **F** measures camera **P** vs. Camera **P'** only, no matter where you put them





# The Essential Matrix E

- If the calibration matrix **K** is known
  - ➔ **$\underline{x} = K^{-1}x = [R|t]X$** ***normalized coordinates***
  - ➔ $K^{-1}P = [R|t]$ ***normalized camera matrix***
- Consider a pair of normalized cameras **P=[I|0]** and **P'=[R|t]**.
  - The Fundamental matrix correspondig to them is called the ***Essential Matrix*** **$E = [t]_x R = R[R^T t]_x$**
  - It is defined by **$\underline{x}'^T E \underline{x} = 0$**
  - Relationship between **E** and **F**:

$$E = K'^T F K$$

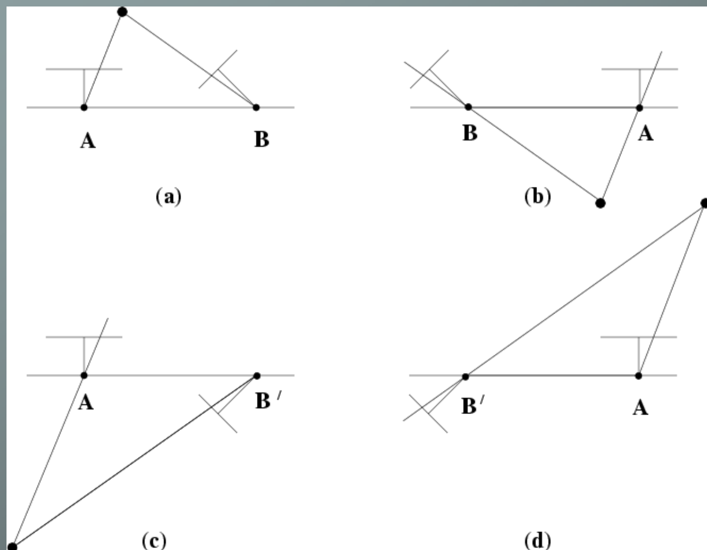




# Properties of Essential Matrix E

- Has 5 DOF (3 for **R** and 2 for **t** up to scale)
  - First two singular values are equal
  - The third is 0
  - **$E = U \mathrm{diag}(1,1,0) V^T$**
- Allows computation of camera matrices **P**, **P'**
  - up to a scale and
  - a four-fold ambiguity

# Four possible reconstructions from E




(only one solution where points is in front of both cameras)





# Computing F

- Compute **F** from $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$

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

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

- separate known from unknown:

$$\underbrace{[x'x, x'y, x', y'x, y'y, y', x, y, 1]}_{\text{data}}\underbrace{[f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33}]^{\mathrm{T}}}_{\text{unknows(linear)}} = 0$$

$$\begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix}\mathbf{f} = \mathbf{0}$$

$$\mathbf{Af} = \mathbf{0}$$





# The singularity constraint

$\mathrm{e}'^{\mathrm{T}}\mathrm{F} = 0$ $\quad$ $\mathrm{Fe} = 0$ $\quad$ $\det\mathrm{F} = 0$ $\quad$ $\mathrm{rank}\,\mathrm{F} = 2$

SVD from linearly computed F matrix (rank 3)

$$\mathrm{F} = \mathrm{U}\begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{bmatrix}\mathrm{V}^{\mathrm{T}} = \mathrm{U}_1\sigma_1\mathrm{V}_1^{\mathrm{T}} + \mathrm{U}_2\sigma_2\mathrm{V}_2^{\mathrm{T}} + \mathrm{U}_3\sigma_3\mathrm{V}_3^{\mathrm{T}}$$

Compute closest rank-2 approximation $\quad \min\|\mathrm{F} - \mathrm{F}'\|_F$

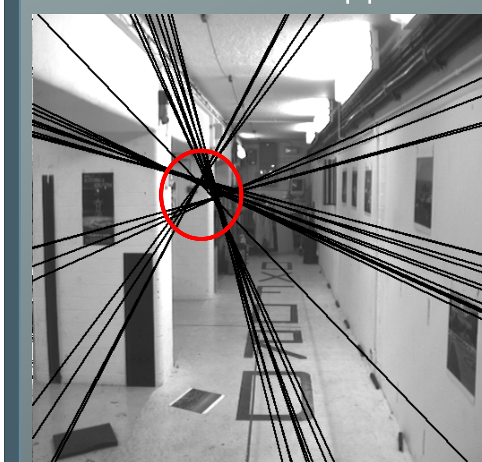
$$\mathrm{F}' = \mathrm{U}\begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & 0 \end{bmatrix}\mathrm{V}^{\mathrm{T}} = \mathrm{U}_1\sigma_1\mathrm{V}_1^{\mathrm{T}} + \mathrm{U}_2\sigma_2\mathrm{V}_2^{\mathrm{T}}$$



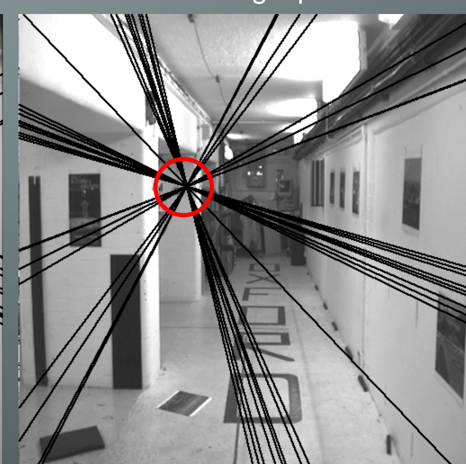


# Example

- The importance of the singularity constraint
  - Guarantees that epipolar lines intersect in one single point



Without **rank F=2** constraint



Constraint enforced

# How many correspondences?

$$\mathbf{Af} = \mathbf{0}$$

- When **A** has rank 8
  - ➔ possible to solve for **f** up to scale
  - ➔ need 8 point correspondences
- When **A** has rank > 8
  - Use LSE:
    - Minimize **||Af||** subject to **||f||=1** (SVD)
  - At least 8 point correspondences
- However **F** has 7 DOF
  - **rank A = 7** is still OK
  - ➔ possible to solve with 7 point correspondences
  - **AND** by making use of the singularity constraint





# 7 point correspondences

- The solution is a 2D space:
  - one parameter family of solutions
  - not automatically rank 2

$$\mathbf{F} = \mathbf{F}_1 + \lambda \mathbf{F}_2$$

$$\mathbf{Af} = \mathbf{0}$$

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

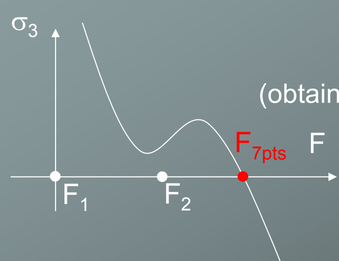
$$\mathrm{A} = \mathrm{U}_{7x7} \mathrm{diag}(\sigma_1, \ldots, \sigma_7, 0, 0) \mathrm{V}_{9x9}^{\mathrm{T}} \quad \Rightarrow \quad \mathrm{A}[\mathrm{V}_8 \mathrm{V}_9] = 0_{9x2}$$

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





# 7 point correspondences




(obtain 1 or 3 solutions)

- Impose rank 2 constraint ➔ Cubic equation:

$$\det(\mathrm{F}_1 + \lambda \mathrm{F}_2) = a_3 \lambda^3 + a_2 \lambda^2 + a_1 \lambda + a_0 = 0$$

$$\det(\mathrm{F}_1 + \lambda \mathrm{F}_2) = \det \mathrm{F}_2 \det(\mathrm{F}_2^{-1} \mathrm{F}_1 + \lambda \mathrm{I}) = 0$$

- Compute $\lambda$ as eigenvalues of $\mathbf{F}_2^{-1}\mathbf{F}_1$
  - only real solutions are potential solutions





# 8 point correspondences

- LSE solution
  - 8 equations but usually **rank A = 9** in case of real (noisy) data

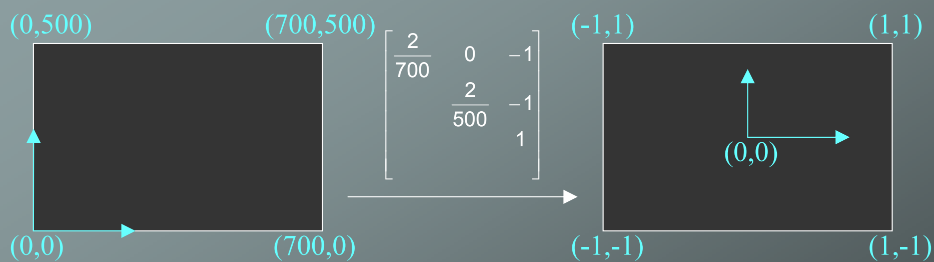
$$\begin{bmatrix} x_1 x_1' & y_1 x_1' & x_1' & x_1 y_1' & y_1 y_1' & y_1' & x_1 & y_1 & 1 \\ x_2 x_2' & y_2 x_2' & x_2' & x_2 y_2' & y_2 y_2' & y_2' & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n x_n' & y_n x_n' & x_n' & x_n y_n' & y_n y_n' &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

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

⚠ Orders of magnitude difference between columns of data matrix ➔ LSE yields poor results

# Normalized 8 point algorithm

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




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

- Given n>=8 point correspondences
  1. Normalization: **Tx** and **T'x'**
  2. Find **F''**
     a) **F'** = Singular vector of smallest singular value from **SVD(A)**
     b) Enforce rank 2 constraint using **SVD(F')** ➔ **F''**
  3. Denormalization: **F=T'ᵀF''T**

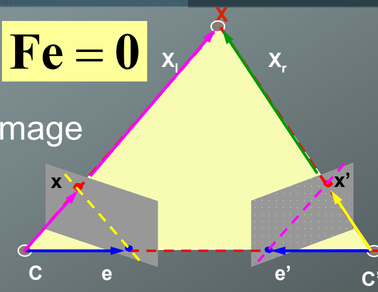




# Locating the Epipoles

$$\mathbf{x}'^{\mathrm{T}}\mathbf{F}\mathbf{x} = 0 \Rightarrow \mathbf{x}'^{\mathrm{T}}\mathbf{F}\mathbf{e} = 0 \Rightarrow \mathbf{F}\mathbf{e} = 0$$

**e** lies on all epipolar lines of the left image




- Input: Fundamental Matrix **F**
  - Find the SVD of **F**: $\mathbf{F} = \mathbf{U}\mathbf{D}\mathbf{V}^T$
  - The epipole **e** is the column of **V** corresponding to the null singular value (as shown above)
  - The epipole **e'** is the column of **U** corresponding to the null singular value (similar treatment as for **e**)
- Output: Epipole **e** and **e'**