Multi-View Geometry Part II (Ch7 New book. Ch 10/11 old book)

Guido Gerig

CS 6320 Spring 2012

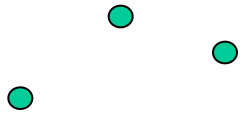
Credits: M. Shah, UCF CAP5415, lecture 23

http://www.cs.ucf.edu/courses/cap6411/cap5415/, Trevor Darrell, Berkeley, C280, Marc Pollefeys

Multi-View Geometry

Relates

Multi-View Geometry



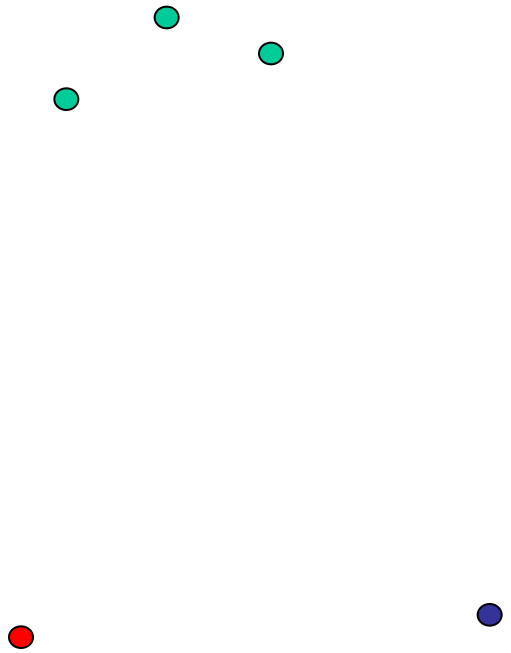
Relates

- 3D World Points

Multi-View Geometry

Relates

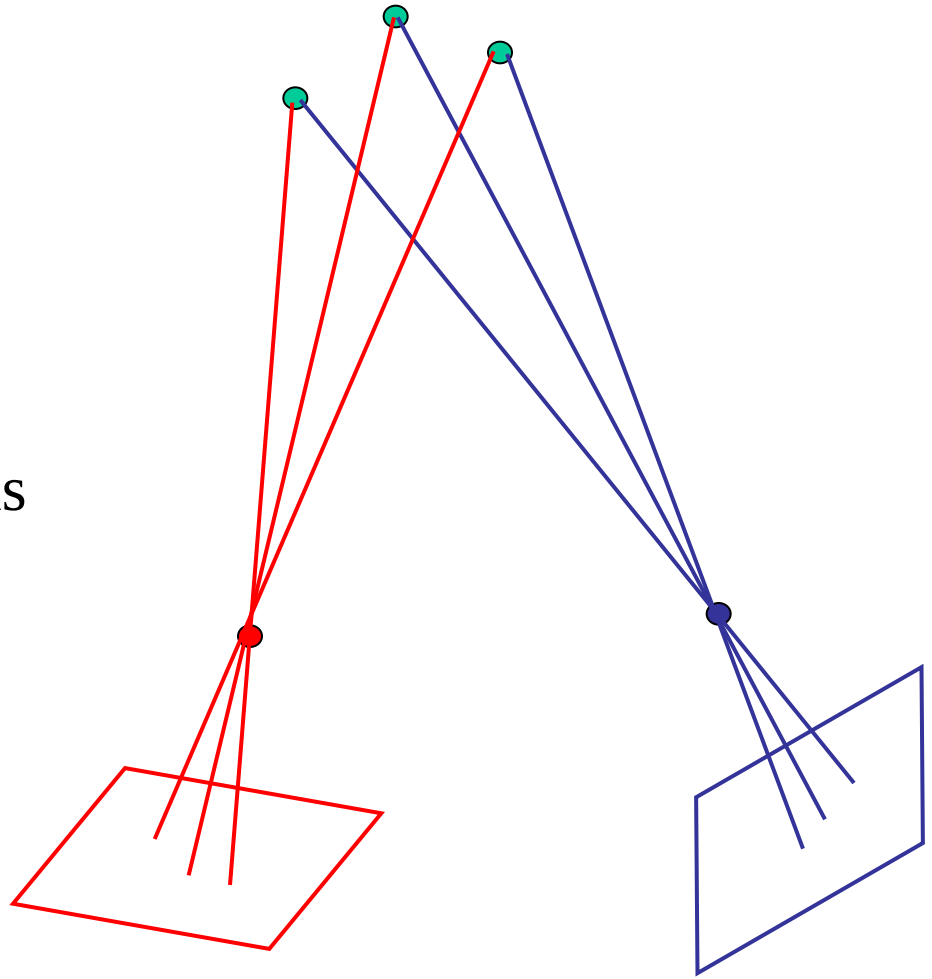
- 3D World Points
- Camera Centers



Multi-View Geometry

Relates

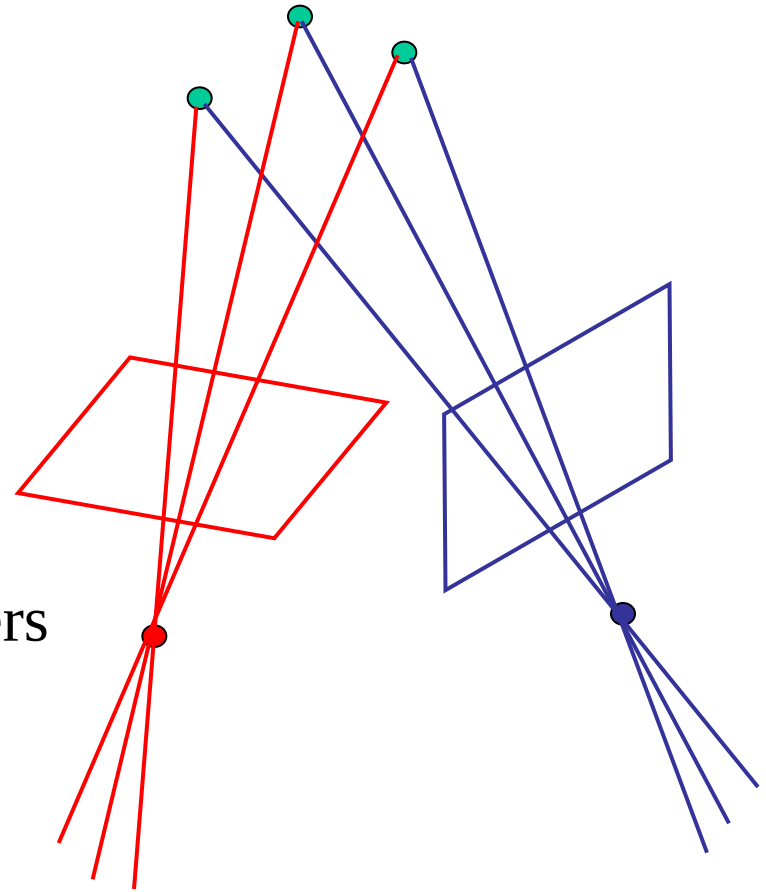
- 3D World Points
- Camera Centers
- Camera Orientations



Multi-View Geometry

Relates

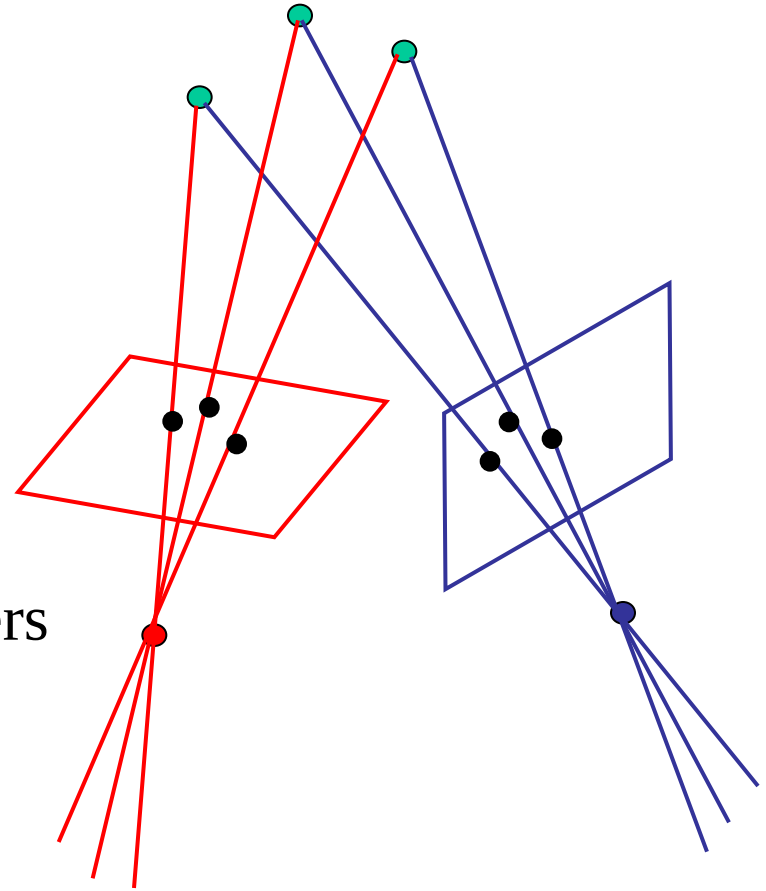
- 3D World Points
- Camera Centers
- Camera Orientations
- Camera Intrinsic Parameters



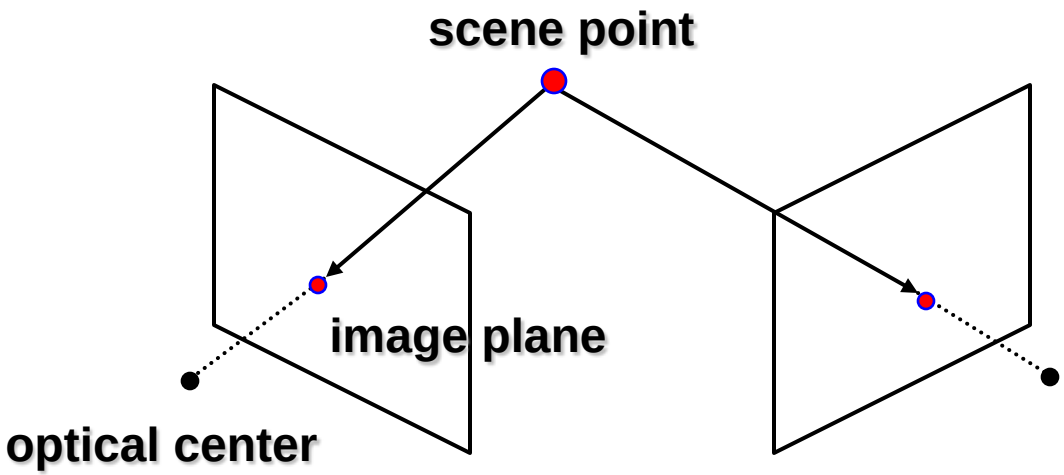
Multi-View Geometry

Relates

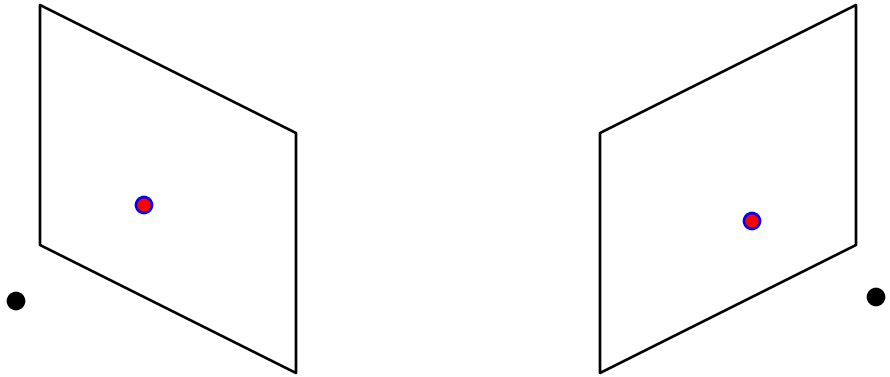
- 3D World Points
- Camera Centers
- Camera Orientations
- Camera Intrinsic Parameters
- Image Points



Stereo



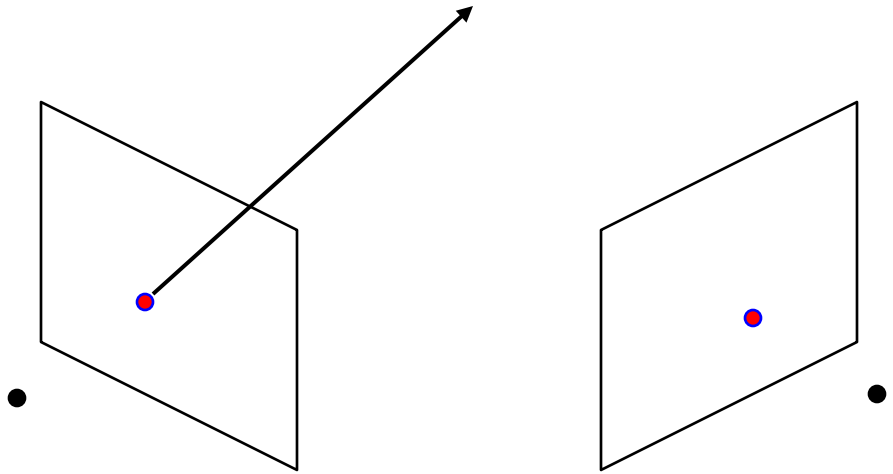
Stereo



Basic Principle: Triangulation

- Gives reconstruction as intersection of two rays

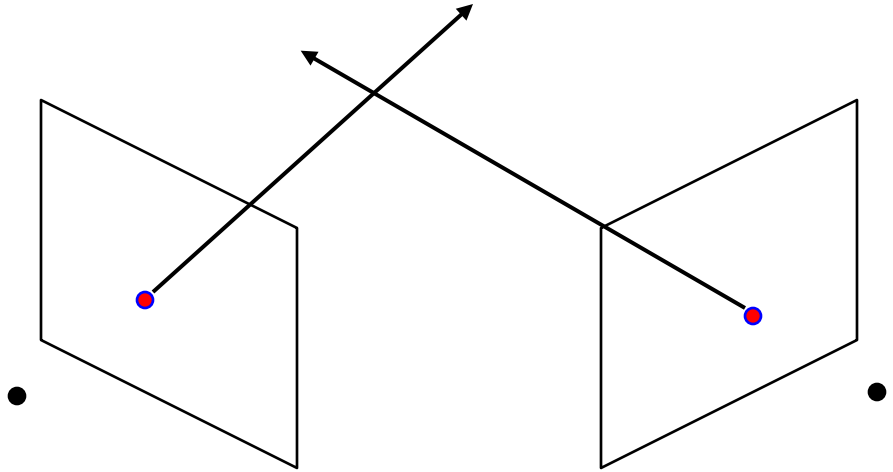
Stereo



Basic Principle: Triangulation

- Gives reconstruction as intersection of two rays

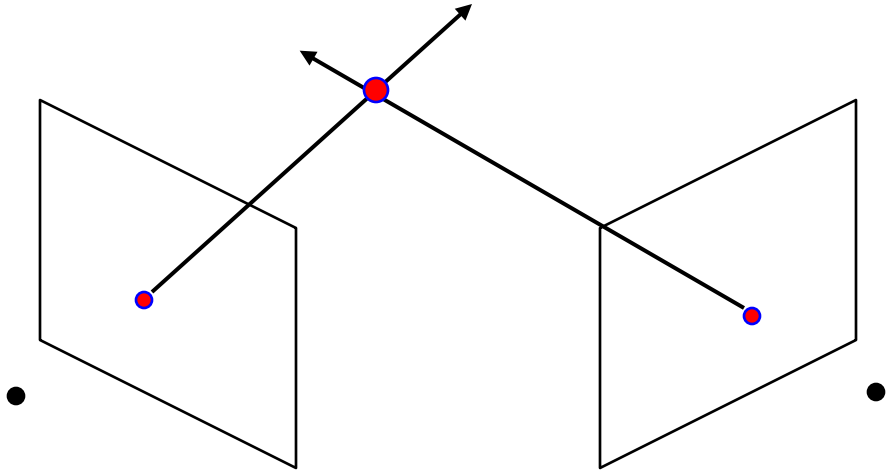
Stereo



Basic Principle: Triangulation

- Gives reconstruction as intersection of two rays

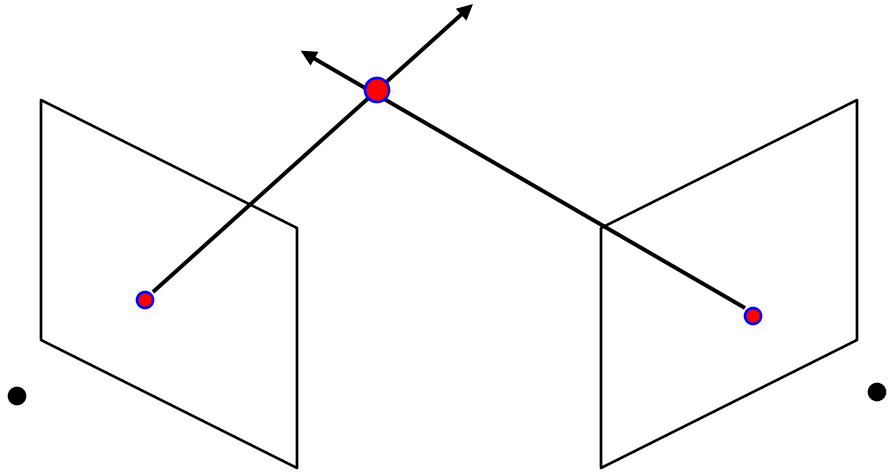
Stereo



Basic Principle: Triangulation

- Gives reconstruction as intersection of two rays

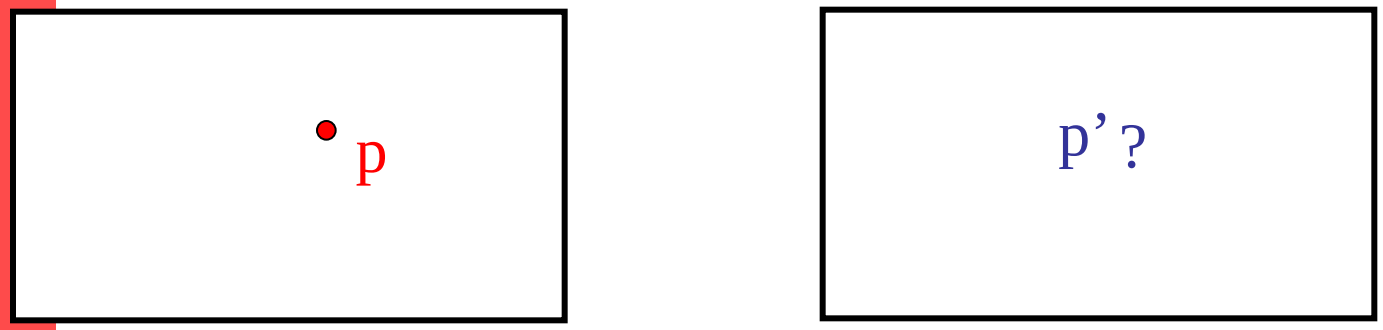
Stereo



Basic Principle: Triangulation

- Gives reconstruction as intersection of two rays
- Requires
  - calibration
  - ***point correspondence***

Stereo Constraints



Given p in left image, where can the corresponding point p’ in right image be?

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

Stereo Constraints

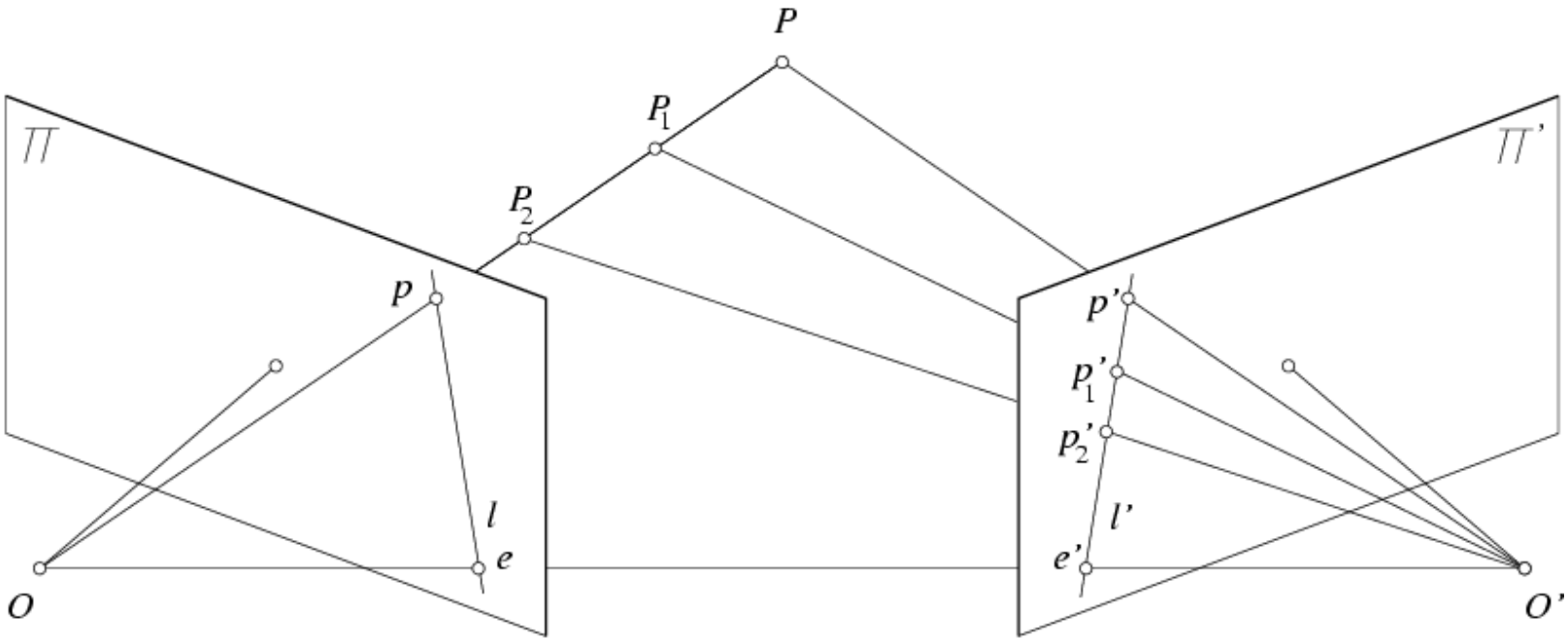
Stereo Constraints

Demo Epipolar Geometry

Java Applet

credit to:
Quang-Tuan Luong
SRI Int.
Sylvain Bougnoux

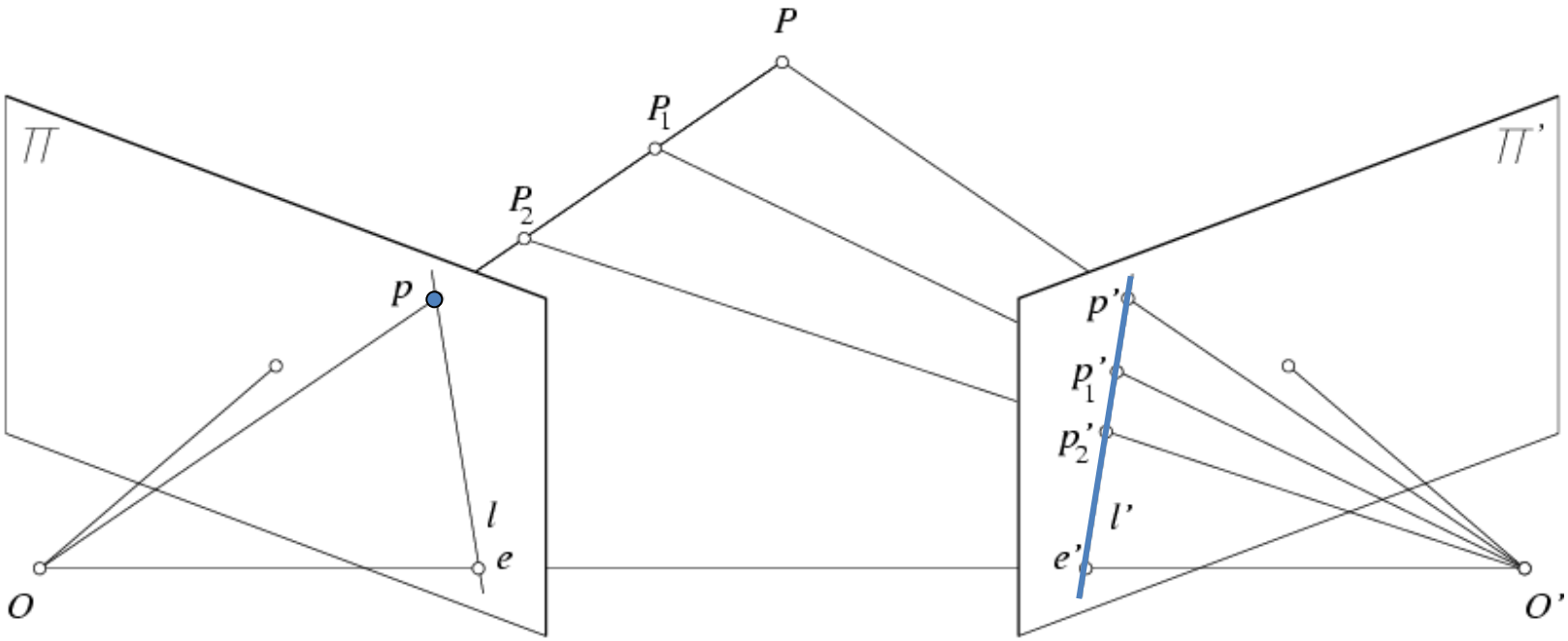
Epipolar constraint



http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Source: M. Pollefeys

Epipolar constraint

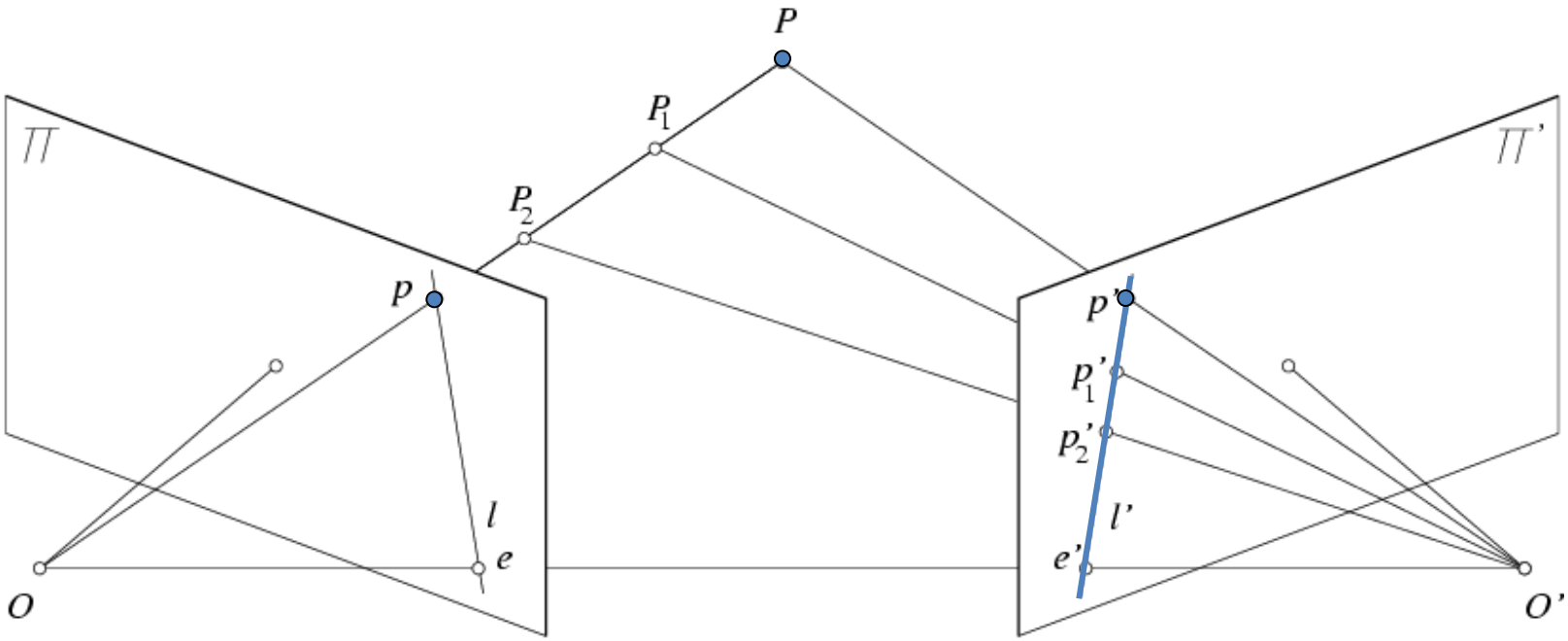


- Potential matches for *p* have to lie on the corresponding epipolar line *l'*.

http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Source: M. Pollefeys

Epipolar constraint

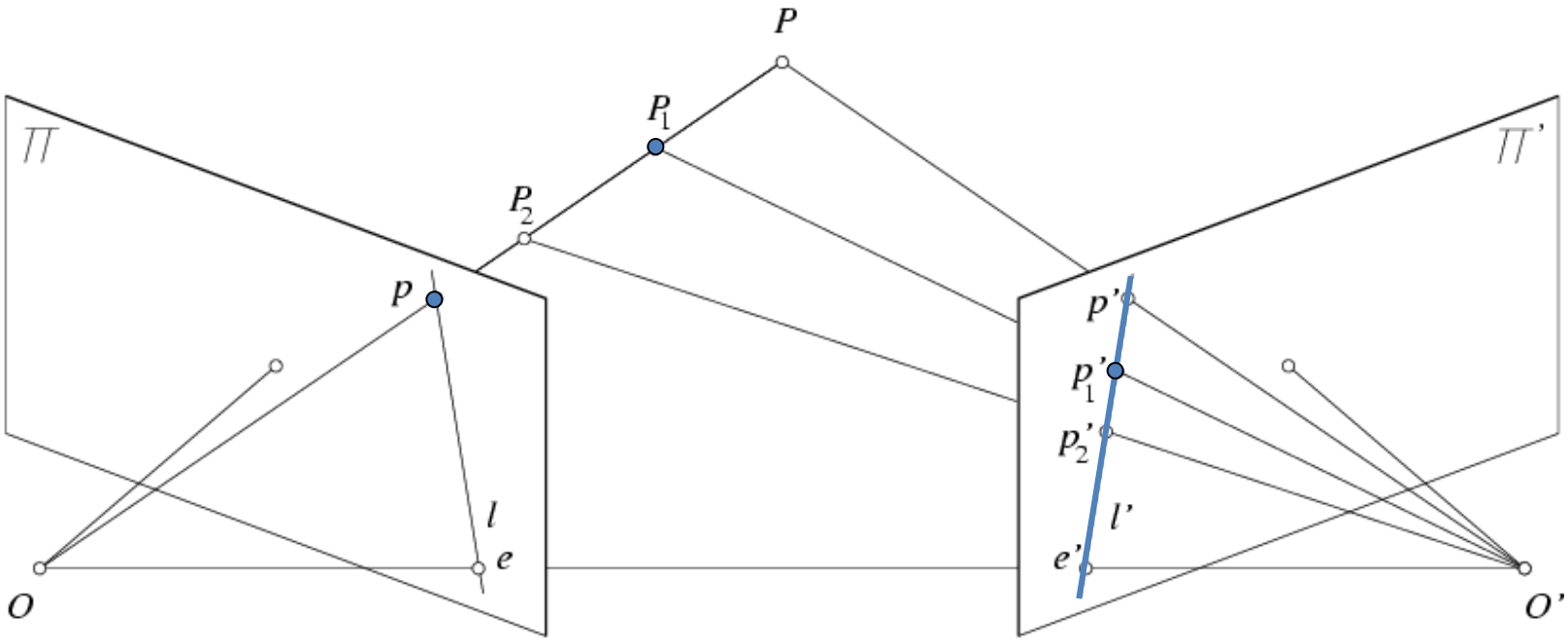


- Potential matches for *p* have to lie on the corresponding epipolar line *l'*.

http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Source: M. Pollefeys

Epipolar constraint

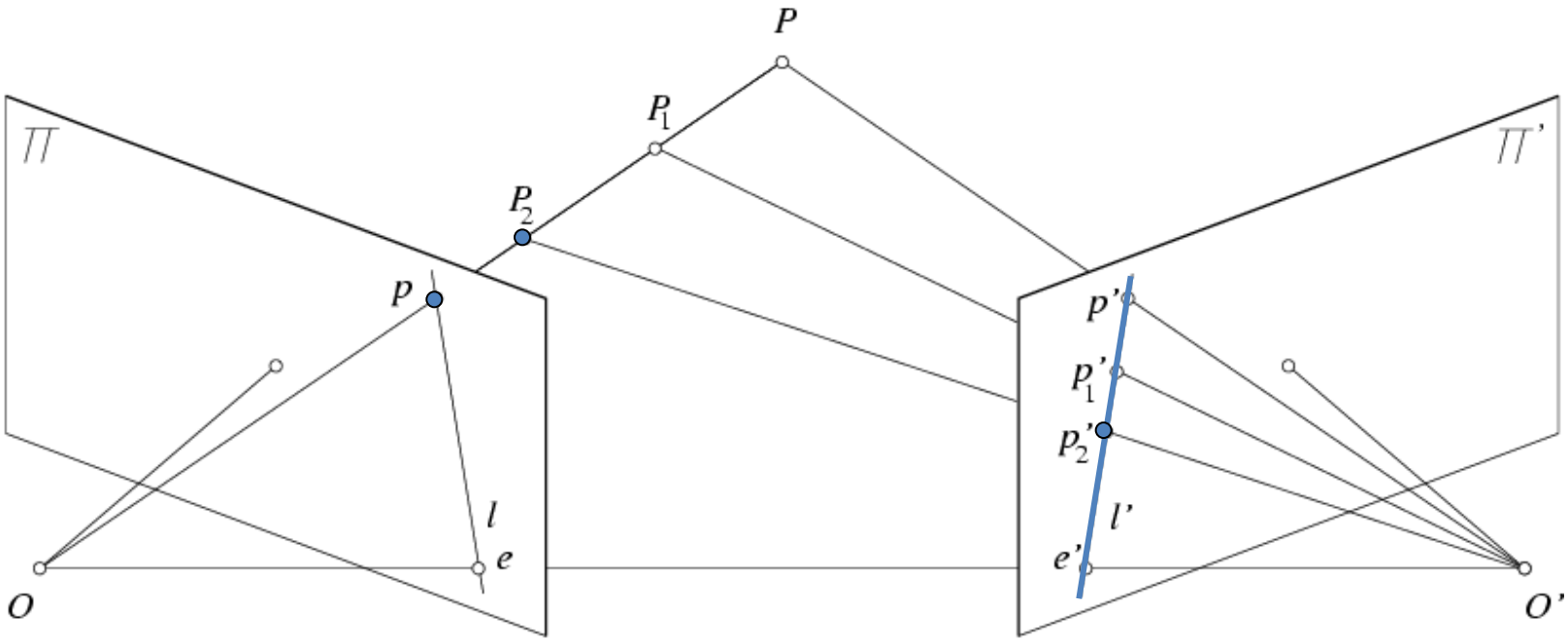


- Potential matches for *p* have to lie on the corresponding epipolar line *l'*.

http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Source: M. Pollefeys

Epipolar constraint

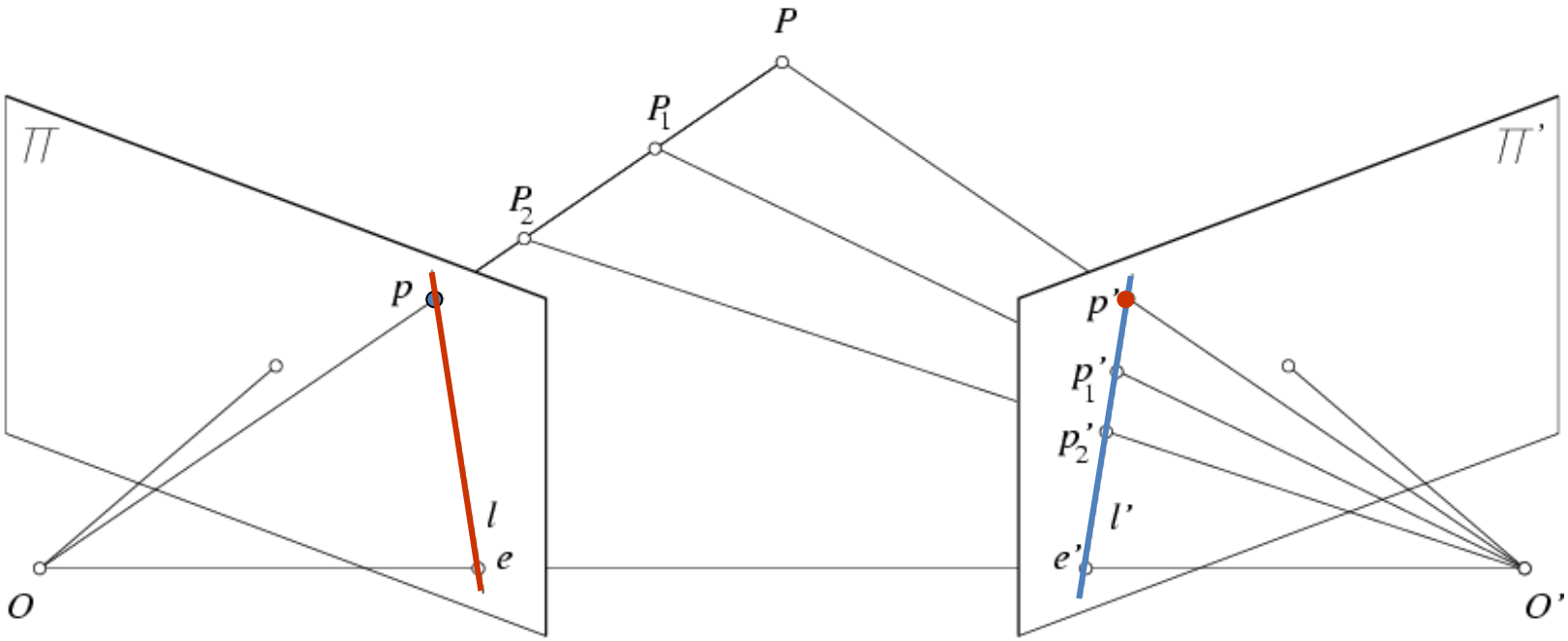


- Potential matches for *p* have to lie on the corresponding epipolar line *l'*.

http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Source: M. Pollefeys

Epipolar constraint



- Potential matches for *p* have to lie on the corresponding epipolar line *l'*.
- Potential matches for *p'* have to lie on the corresponding epipolar line *l*.

http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

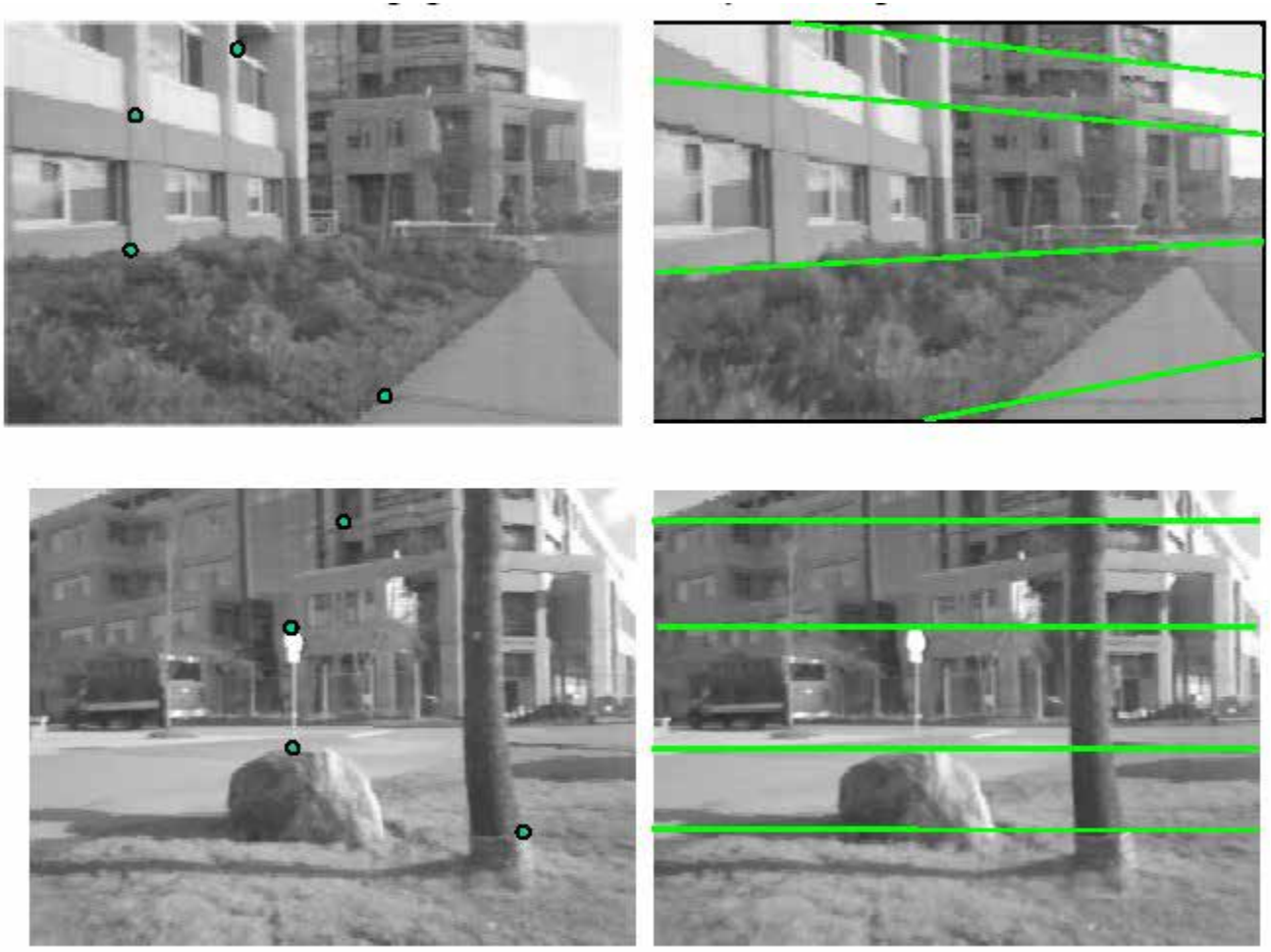
Source: M. Pollefeys

Finding Correspondences

Andrea Fusiello, CVonline

Strong constraints for searching for corresponding points!

Example



Parallel Cameras: Corresponding points on horizontal lines.

Epipolar Constraint

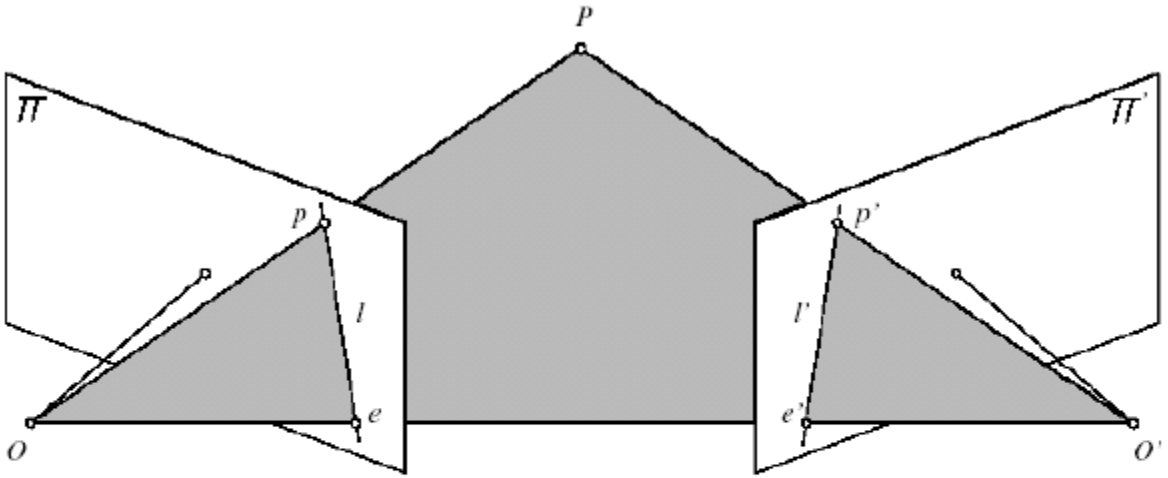


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

All epipolar lines contain epipole, the image of other camera center.

From Geometry to Algebra

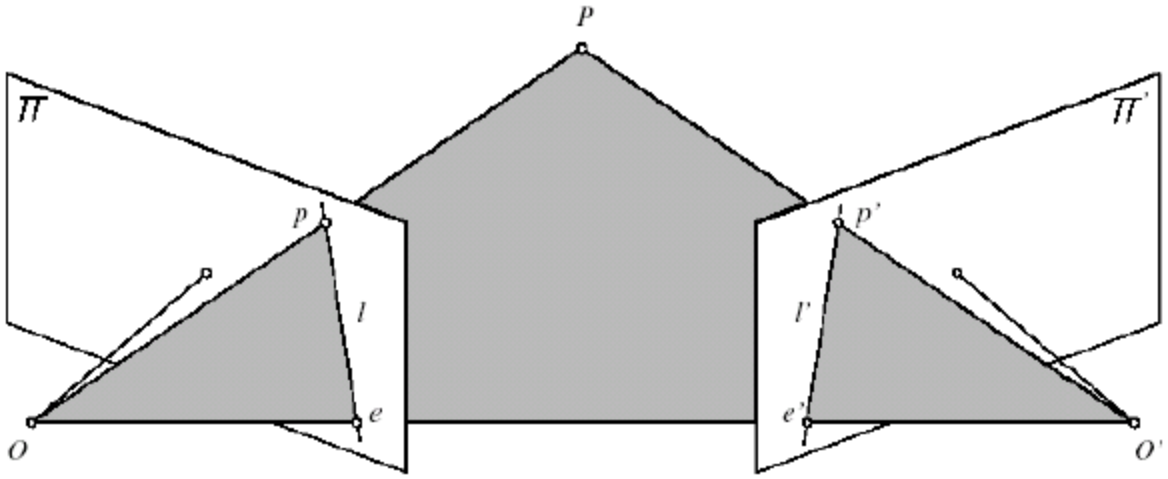


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra

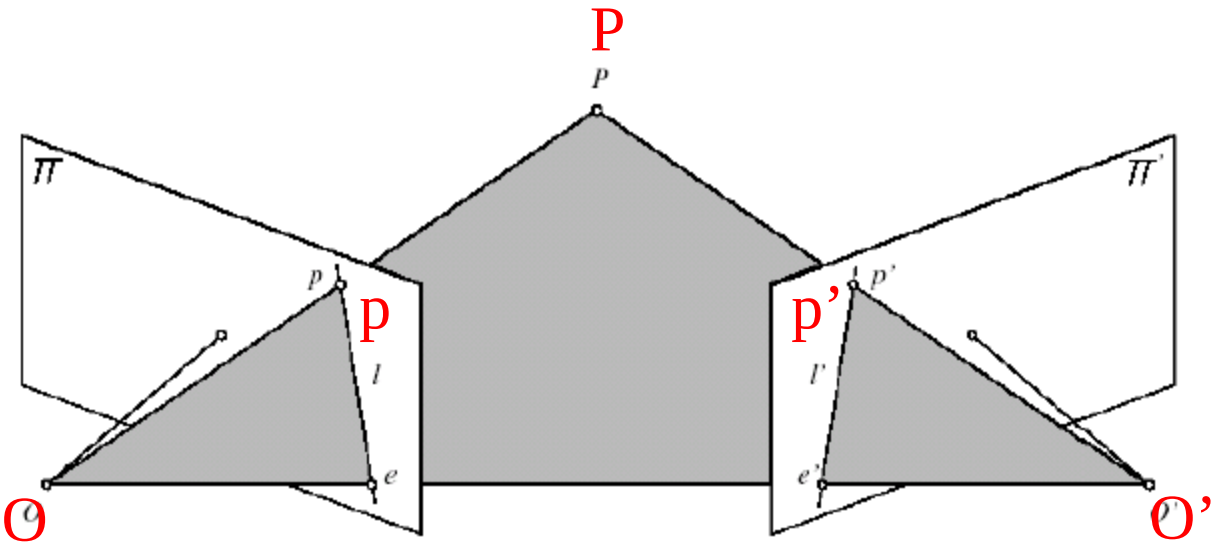


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra

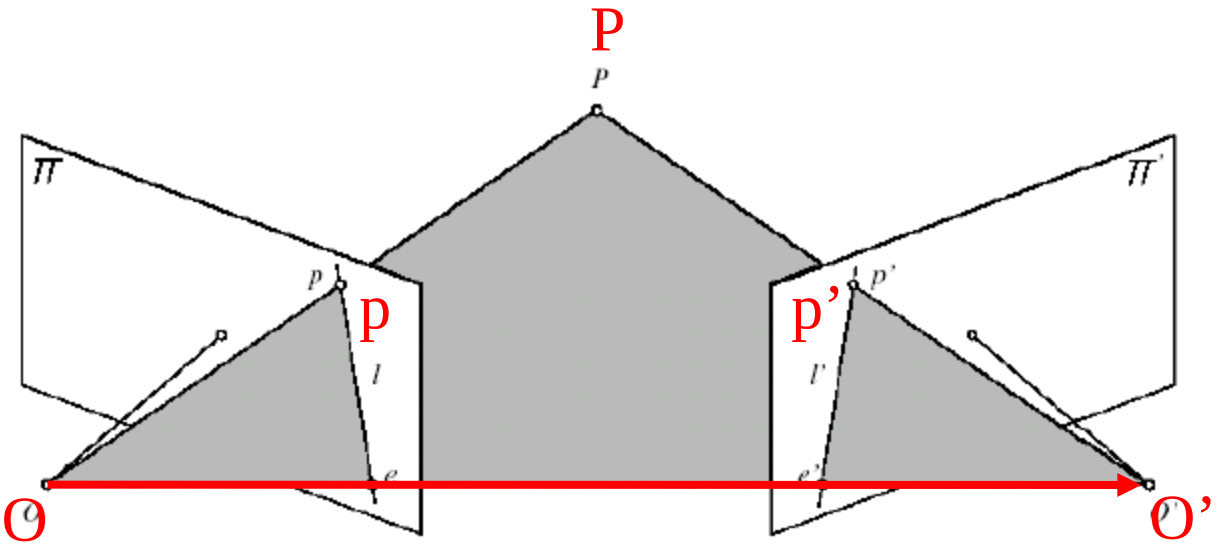


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra

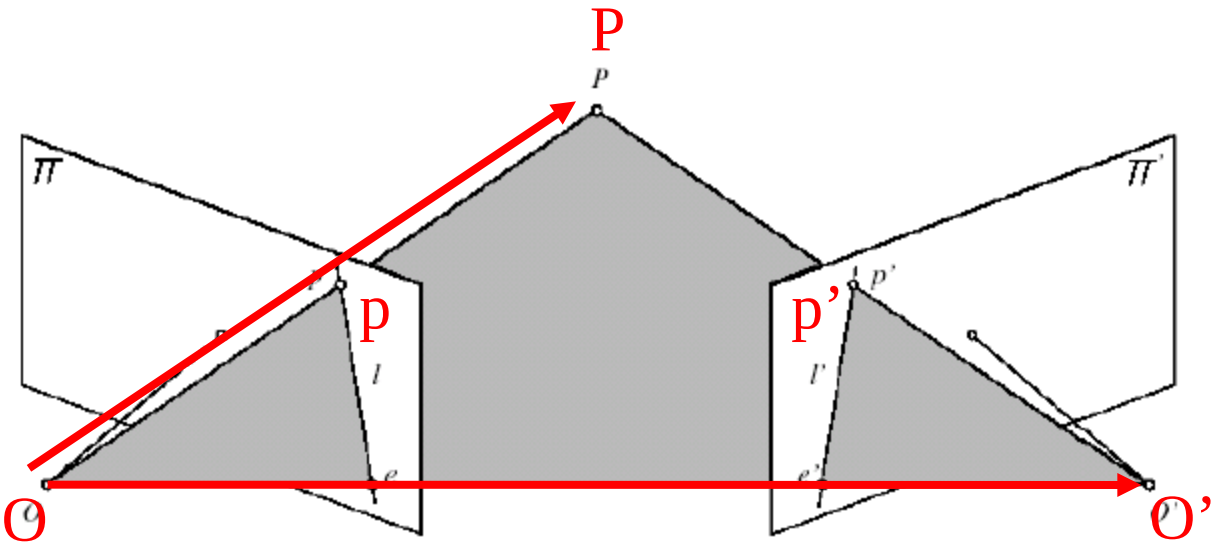


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra

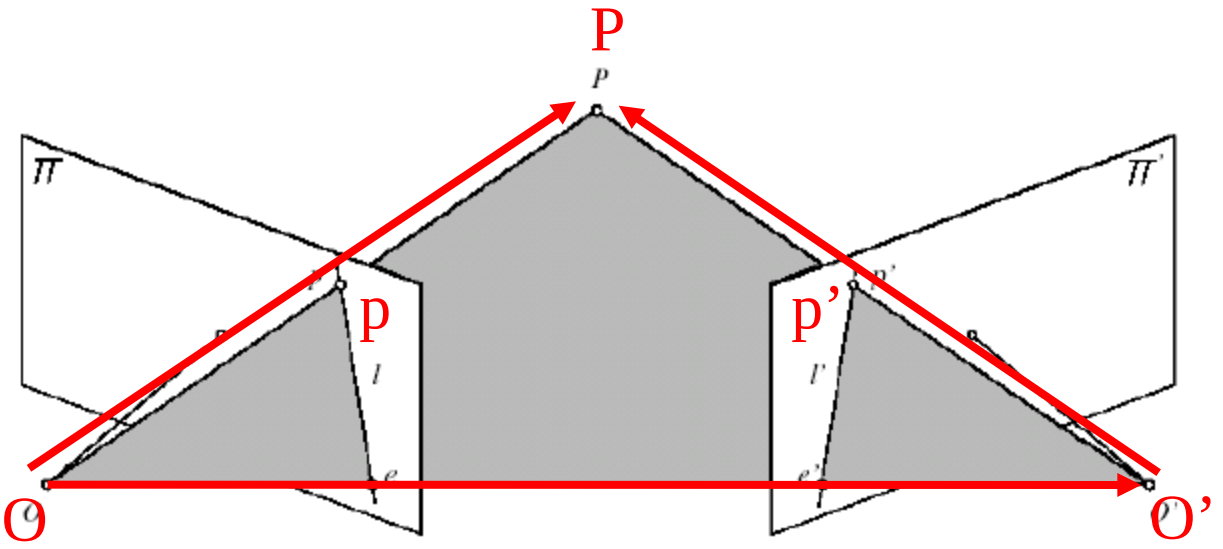


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra

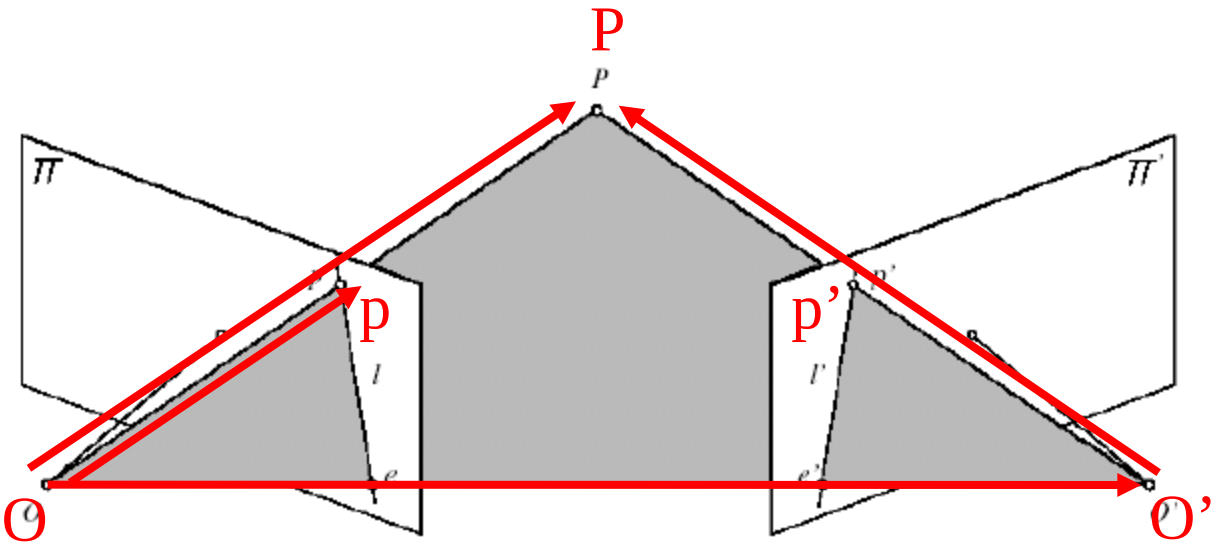


FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra

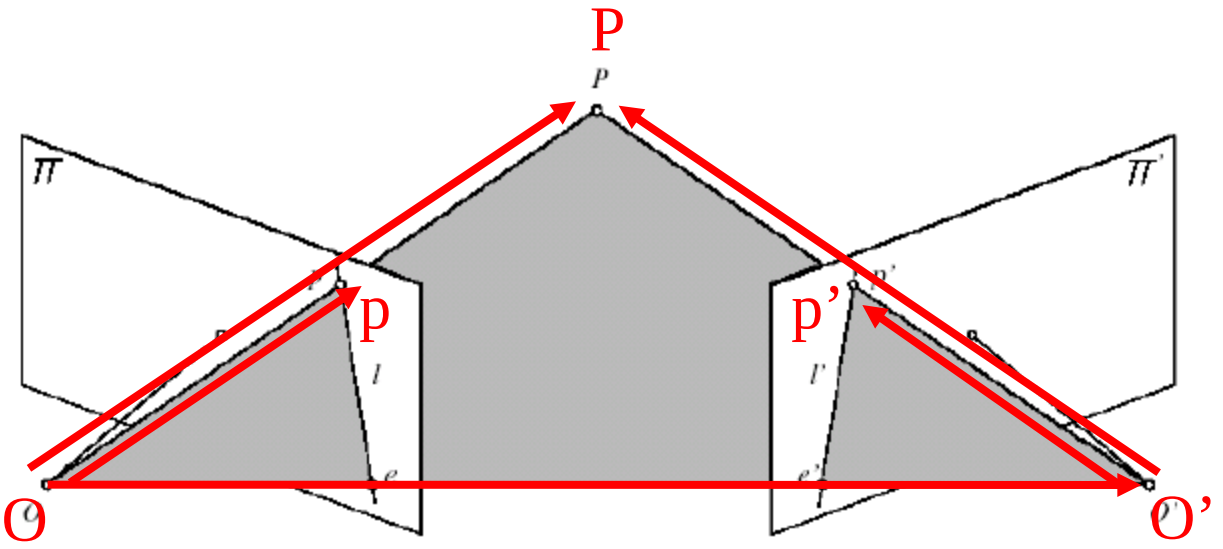
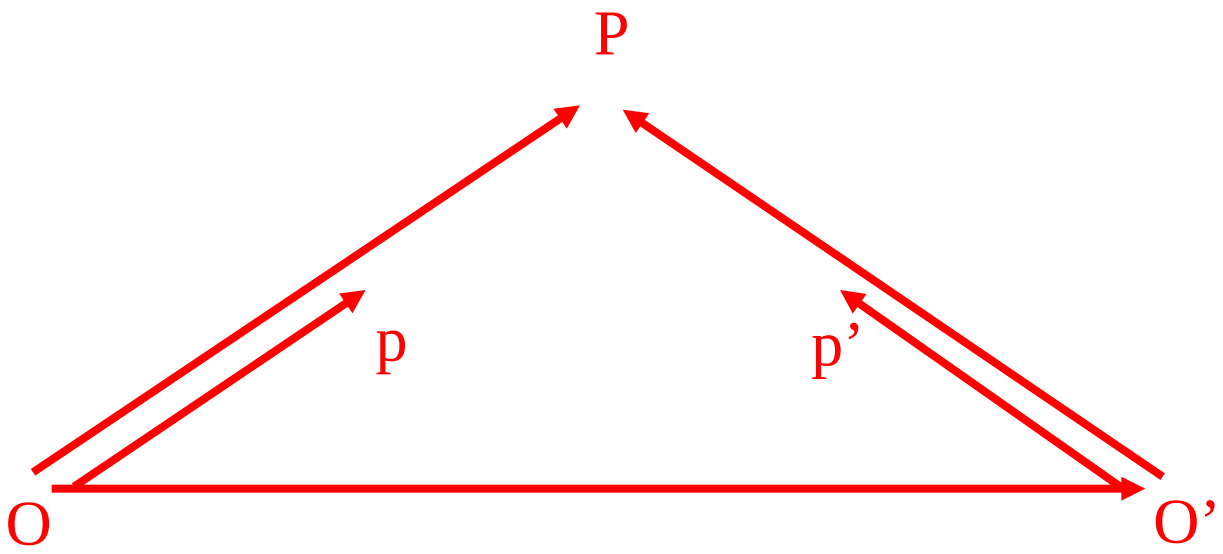


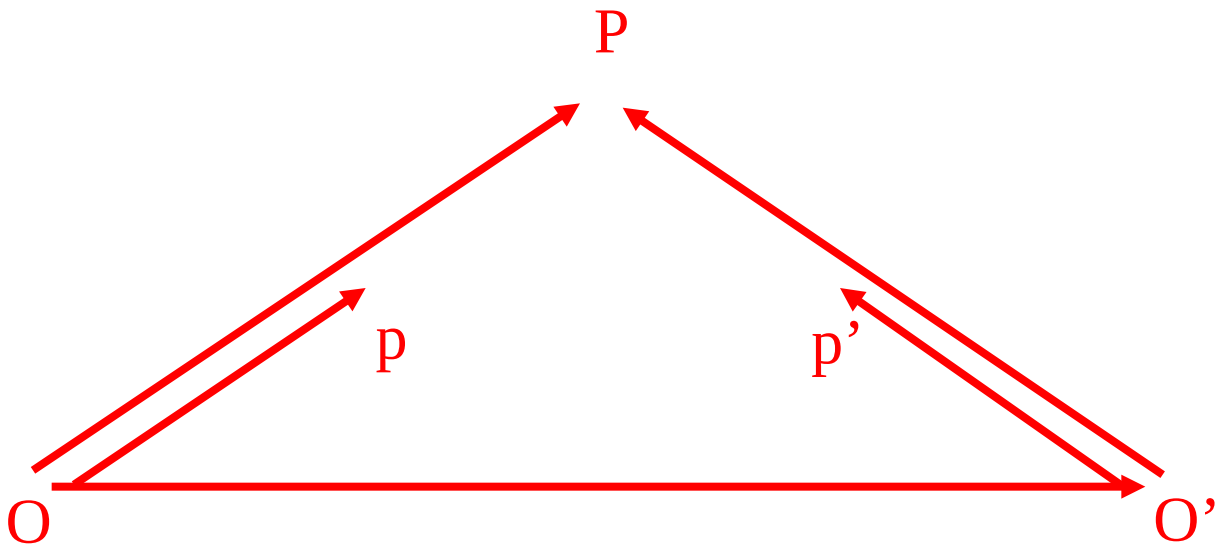
FIGURE 11.1: Epipolar geometry: the point P, the optical centers O and O' of the two cameras, and the two images p and p' of P all lie in the same plane.

From Geometry to Algebra



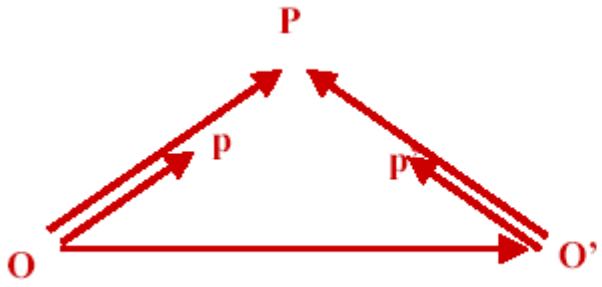
The epipolar constraint: these vectors are coplanar:

From Geometry to Algebra

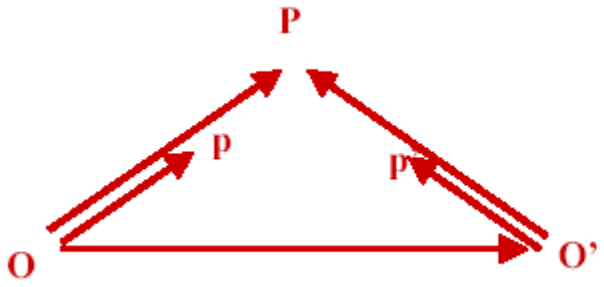


The epipolar constraint: these vectors are coplanar:

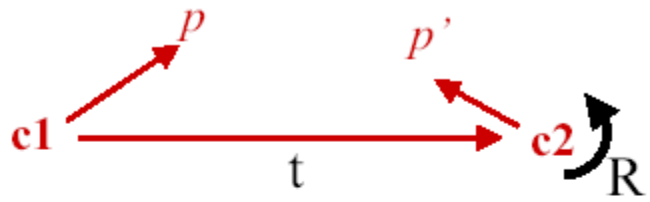
$$\overrightarrow{Op} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p'}] = 0$$



$$\overrightarrow{Op} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p'}] = 0$$

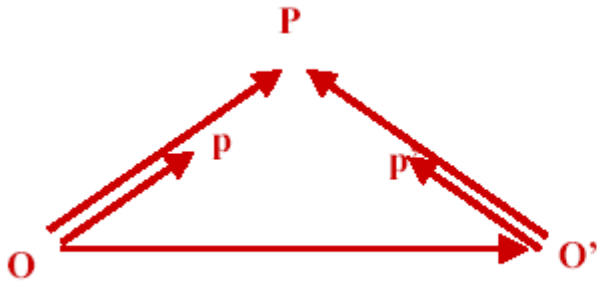


$$\overrightarrow{Op} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p'}] = 0$$

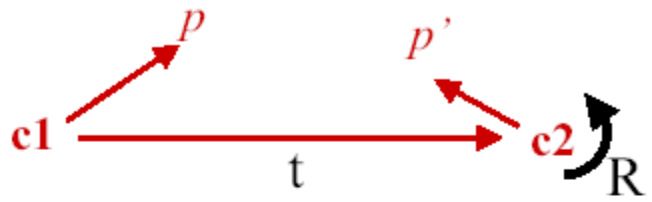


p,p' are image coordinates of P in c1 and c2…

c2 is related to c1 by rotation R and translation t



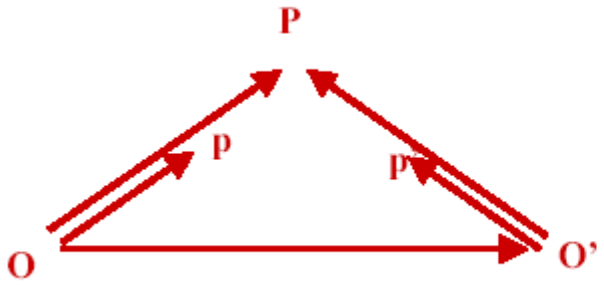
$$\overrightarrow{Op} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p'}] = 0$$



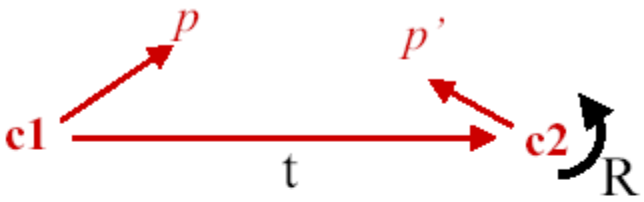
p,p' are image coordinates of P in c1 and c2…

c2 is related to c1 by rotation R and translation t

$$\boldsymbol{p} \cdot [\boldsymbol{t} \times (\mathcal{R}\boldsymbol{p}')] = 0$$



$$\overrightarrow{Op} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p'}] = 0$$



p,p' are image coordinates of P in c1 and c2…

c2 is related to c1 by rotation R and translation t

$$\boldsymbol{p} \cdot [\boldsymbol{t} \times (\mathcal{R}\boldsymbol{p}')] = 0$$

Linear Constraint:
Should be able to express as matrix multiplication.

Review: Matrix Form of Cross Product

The vector cross product also acts on two vectors and returns a third vector. Geometrically, this new vector is constructed such that its projection onto either of the two input vectors is zero.

$$\vec{a} \times \vec{b} = \begin{bmatrix} a_y b_z - a_z b_y \\ a_z b_x - a_x b_z \\ a_x b_y - a_y b_x \end{bmatrix}$$

$$\vec{a} \times \vec{b} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \vec{c} \qquad \begin{aligned} \vec{a} \cdot \vec{c} &= 0 \\ \vec{b} \cdot \vec{c} &= 0 \end{aligned}$$

Review: Matrix Form of Cross Product

$$\vec{a}\times\vec{b}=\begin{bmatrix}0 & -a_z & a_y\\ a_z & 0 & -a_x\\ -a_y & a_x & 0\end{bmatrix}\begin{bmatrix}b_x\\ b_y\\ b_z\end{bmatrix}=\vec{c}$$

$$\vec{a}\cdot\vec{c}=0$$
$$\vec{b}\cdot\vec{c}=0$$

$$[a_x]=\begin{bmatrix}0 & -a_z & a_y\\ a_z & 0 & -a_x\\ -a_y & a_x & 0\end{bmatrix}$$

$$\vec{a}\times\vec{b}=[a_x]\vec{b}$$

Matrix Form

$$\boldsymbol{p} \cdot [\boldsymbol{t} \times (\mathcal{R}\boldsymbol{p}')] = 0$$

$$\vec{a} \times \vec{b} = [a_x]\vec{b}$$

$$p^T [t_x] \Re p' = 0$$

$$\varepsilon = [t_x]\Re$$

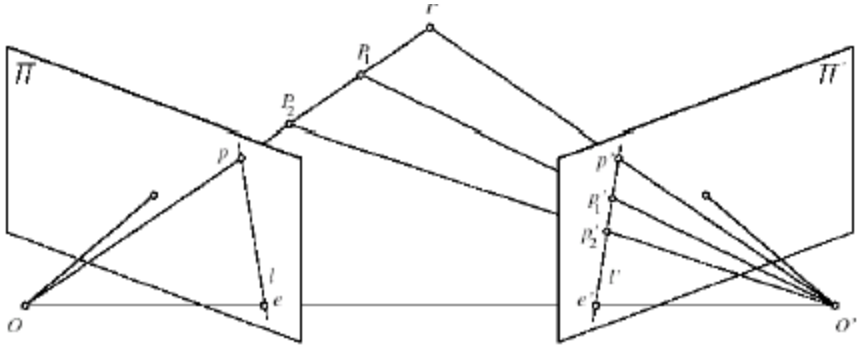
$$\boldsymbol{p}^T \mathcal{E} \boldsymbol{p}' = 0$$

The Essential Matrix

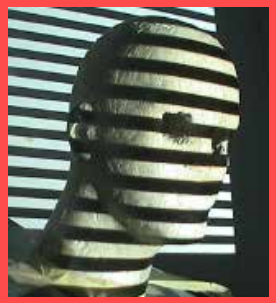
Matrix that relates image of point in one camera to a second camera, given translation and rotation.

$$\mathcal{E} = [t_x]\Re$$

$$\boldsymbol{p}^T \mathcal{E} \boldsymbol{p}' = 0$$



$$\vec{a} \times \vec{b} = [a_x]\vec{b}$$



The Essential Matrix

- Based on the Relative Geometry of the Cameras
- Assumes Cameras are calibrated (i.e., intrinsic parameters are known)
- Relates image of point in one camera to a second camera (points in camera coordinate system).
- Is defined up to scale
- 5 independent parameters

The Essential Matrix

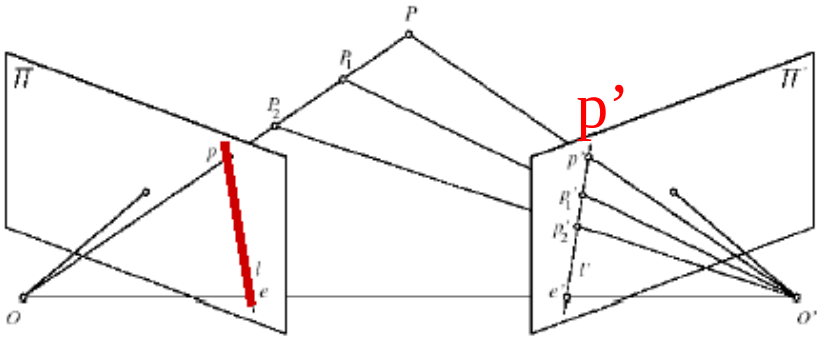
$$p^T \mathcal{E} p' = 0$$

What is $\mathcal{E}p'$?

The Essential Matrix

$\mathcal{E}p'$ is the epipolar line corresponding to p' in the left camera.

$$au + bv + c = 0$$



$$p = (u, v, 1)^T$$
$$l = (a, b, c)^T$$
$$l \cdot p = 0$$

$$\mathcal{E}p' \cdot p = 0$$

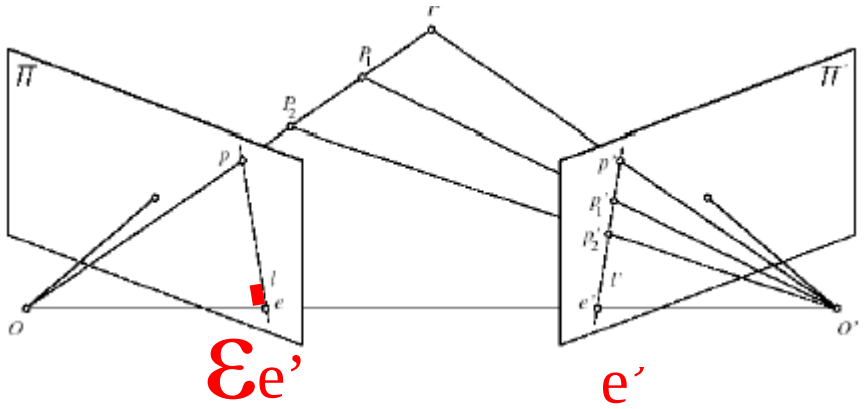
$$\mathbf{p}^T \mathcal{E} \mathbf{p}' = 0$$

Similarly $e^T p$ is the epipolar line corresponding to **p** in the right camera.

The Essential Matrix

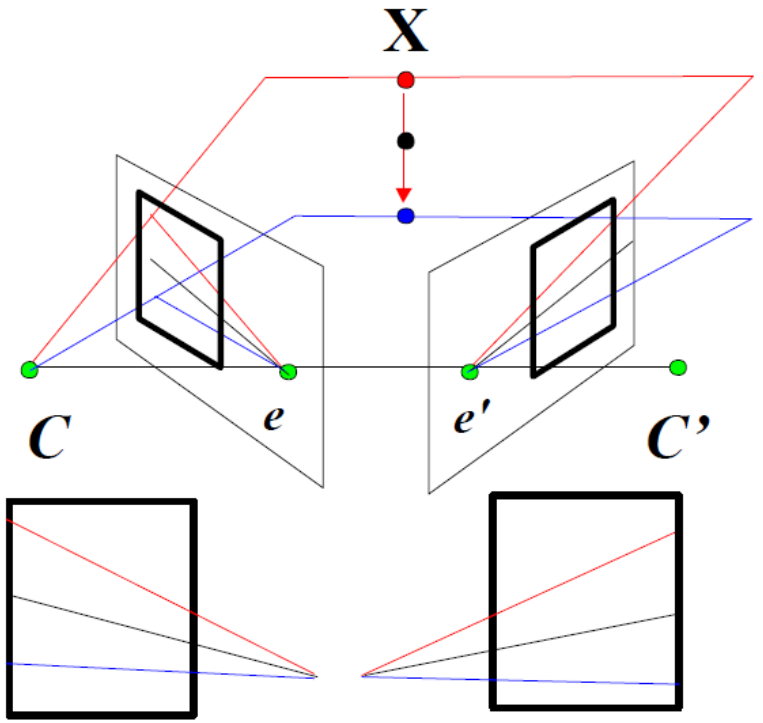
$e^T \mathcal{E} e'$?

What is $\mathcal{E} e'$?



- line $\mathcal{E}p'$ converges to epipole e
- e' (center of camera C_2 expressed in frame $C_{1)}$

Pencil of epipolar lines



- The intersection points between the base line and the image planes are called *epipoles*.
- The epipole e' in image 2 is the mapping of the camera center C.
- The epipole e in image 1 is the mapping of the camera center C'.
- Since all epipolar planes intersect both camera centers, all epipolar lines will intersect the epipoles.

Pencil of epipolar lines

image1

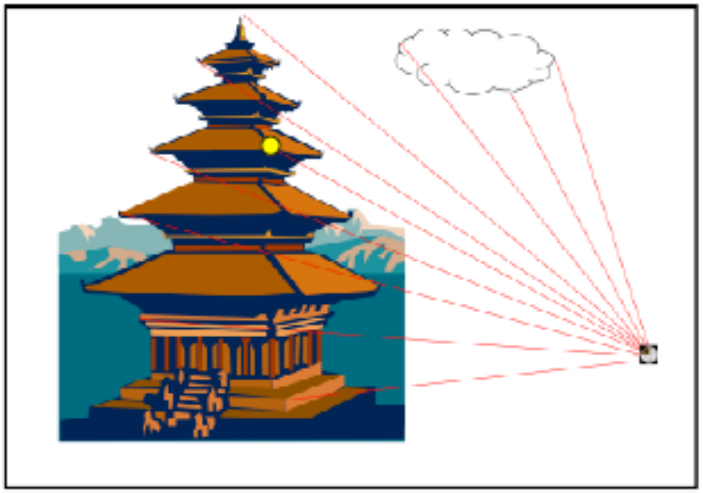
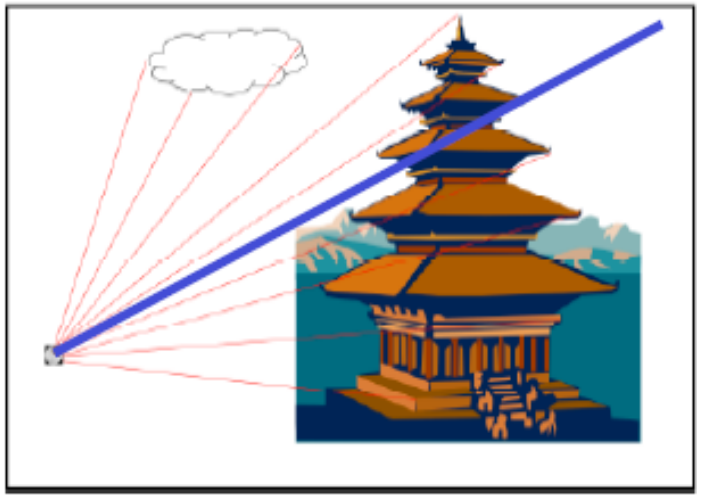


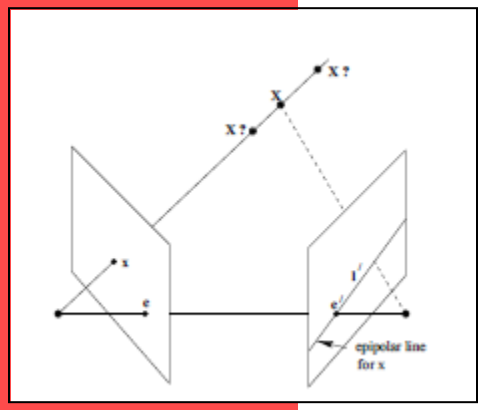
image 2



Given a point in one image, how do we determine the corresponding epipolar line to search along in the second image?

http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf

Epipoles

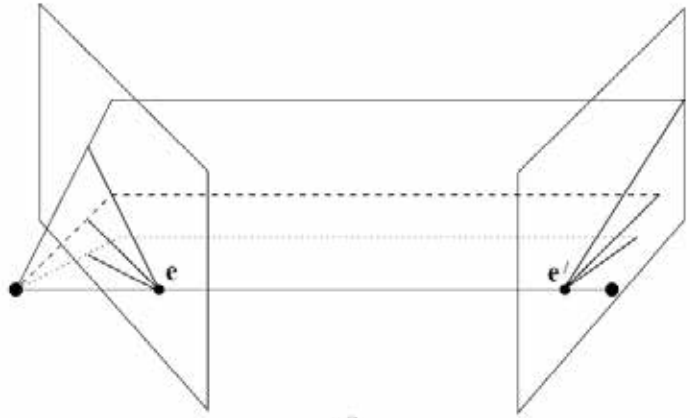


- The epipolar line $l' = \mathcal{E}x$ to each point x (except e) intersects the epipole e_0.

  Thus e' satisfies $e'^T(\mathcal{E}x) = (e'^T \mathcal{E})x = 0$ for all x.

- This implies that $e'^T \mathcal{E} = 0^T$ or $\mathcal{E}^T e' = 0$. The epipole e' is thus a null vector to $\mathcal{E}^T$ (in the left null-space of $\mathcal{E}$).

- Similarly, $\mathcal{E}e = 0$, i.e. e is a null-vector to $\mathcal{E}$ (in the right null-space of $\mathcal{E}$).



a

b c

Fig. 8.3. **Converging cameras.** *(a) Epipolar geometry for converging cameras. (b) and (c) A pair of images with superimposed corresponding points and their epipolar lines (in white). The motion between the views is a translation and rotation. In each image, the direction of the other camera may be inferred from the intersection of the pencil of epipolar lines. In this case, both epipoles lie outside of the visible image.*

MVG Hartley & Zisserman

Epipoles

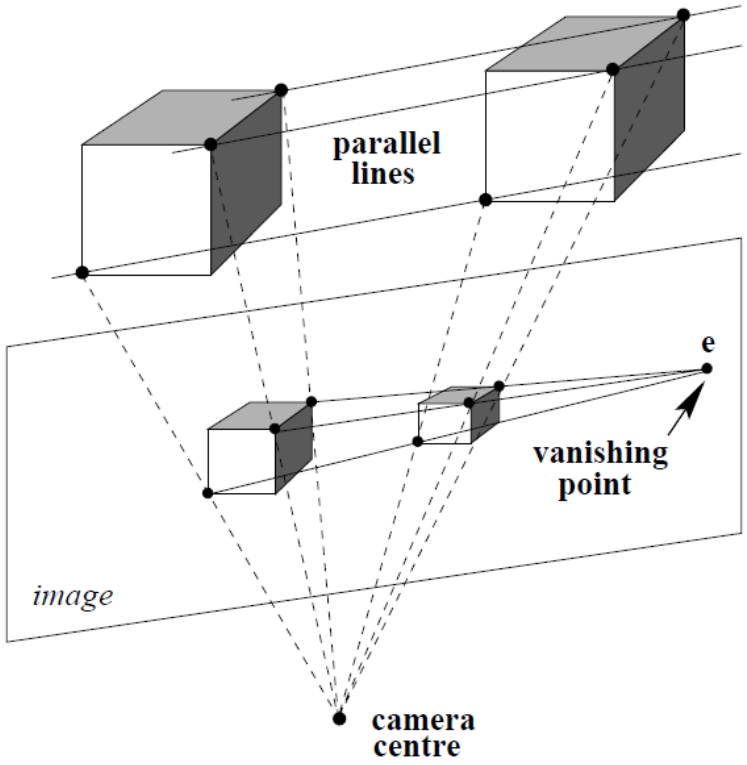
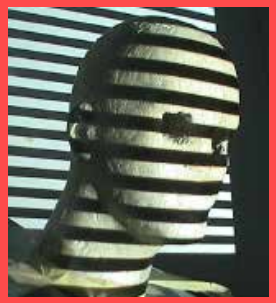


Fig. 8.7. *Under a pure translational camera motion, 3D points appear to slide along parallel rails. The images of these parallel lines intersect in a vanishing point corresponding to the translation direction. The epipole* **e** *is the vanishing point.*



The Essential Matrix

$$e^{T}\varepsilon e = e^{T}[t_{\times}]Re = 0$$

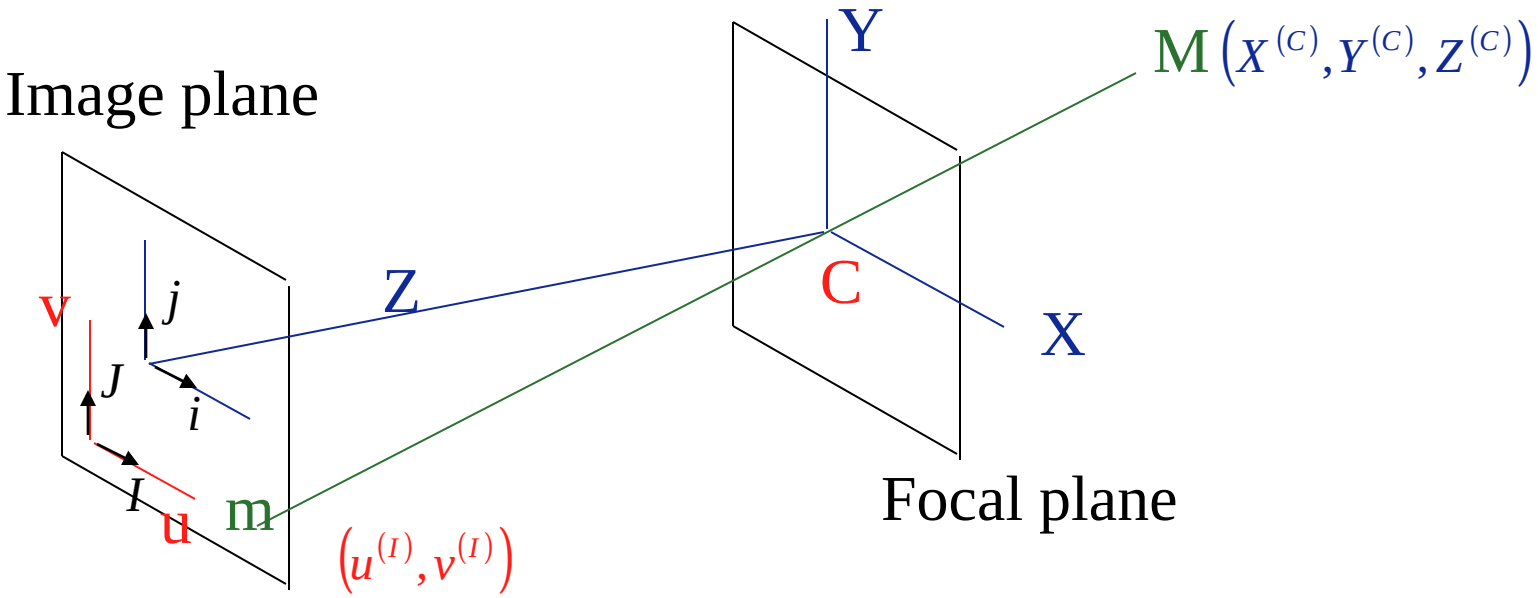
Similarly, $e^{T}\varepsilon e = R^{T}[t_{\times}]^{T}e = -R^{T}[t_{\times}]e = 0$

The essential matrix $\varepsilon = [t]\times R$ has 5 degrees of freedom; 3 rotation angles in R, 3 elements in t, but arbitrary scale.

Essential Matrix is singular with rank 2.

What if Camera Calibration is not known

Review: Intrinsic Camera Parameters



$$\begin{bmatrix} u^{(I)} \\ v^{(I)} \\ S \end{bmatrix} = \underbrace{\begin{bmatrix} -f_u & 0 & u_0 & 0 \\ 0 & -f_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{K} \begin{bmatrix} X^{(C)} \\ Y^{(C)} \\ Z^{(C)} \\ 1 \end{bmatrix}$$

$f_u = fk_u = a$

$f_v = fk_v = b$

$\theta = 90°$

Fundamental Matrix

$p^T E p' = 0$ p and p' are in camera coordinate system

If u and u' are corresponding image coordinates then we have:

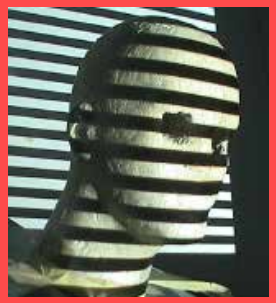
$$u = K_1 p \qquad p = K_1^{-1} u \Rightarrow p^T = \left(K_1^{-1} u\right)^T = u^T K_1^{-T}$$

$$u' = K_2 p' \Longrightarrow p' = K_2^{-1} u'$$

$$u^T K_1^{-T} E K_2^{-1} u' = 0$$

$$\Rightarrow u^T F u' = 0$$

$$F = K_1^{-T} E K_2^{-1}$$



Fundamental Matrix

$$u^T F u = 0 \qquad F = K_1^{-T} e K_2^{-1}$$

Fundamental Matrix is singular with rank 2.

The fundamental matrix F has 7 degrees of freedom: A 3 × 3 homogenous matrix has 8 degrees of freedom. The constraint rank(F) = 2 or det(F) = 0 reduces the number to 7.
In principal F has 7 parameters up to scale and can be estimated from 7 point correspondences.

Direct Simpler Method requires 8 correspondences (Olivier Faugeras, Computer Vision textbook).

Estimating Fundamental Matrix

$$u^T F u' = 0$$

The 8-point algorithm (Faugeras)

Each point correspondence can be expressed as a linear equation:

$$\begin{bmatrix} u & v & 1 \end{bmatrix} \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = 0$$

$$\begin{bmatrix} uu' & uv' & u & u' & vv' & v & u' & v' & 1 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0$$

The 8-point Algorithm

Scaling: Set F_{33} to 1 -> Solve for 8 parameters.

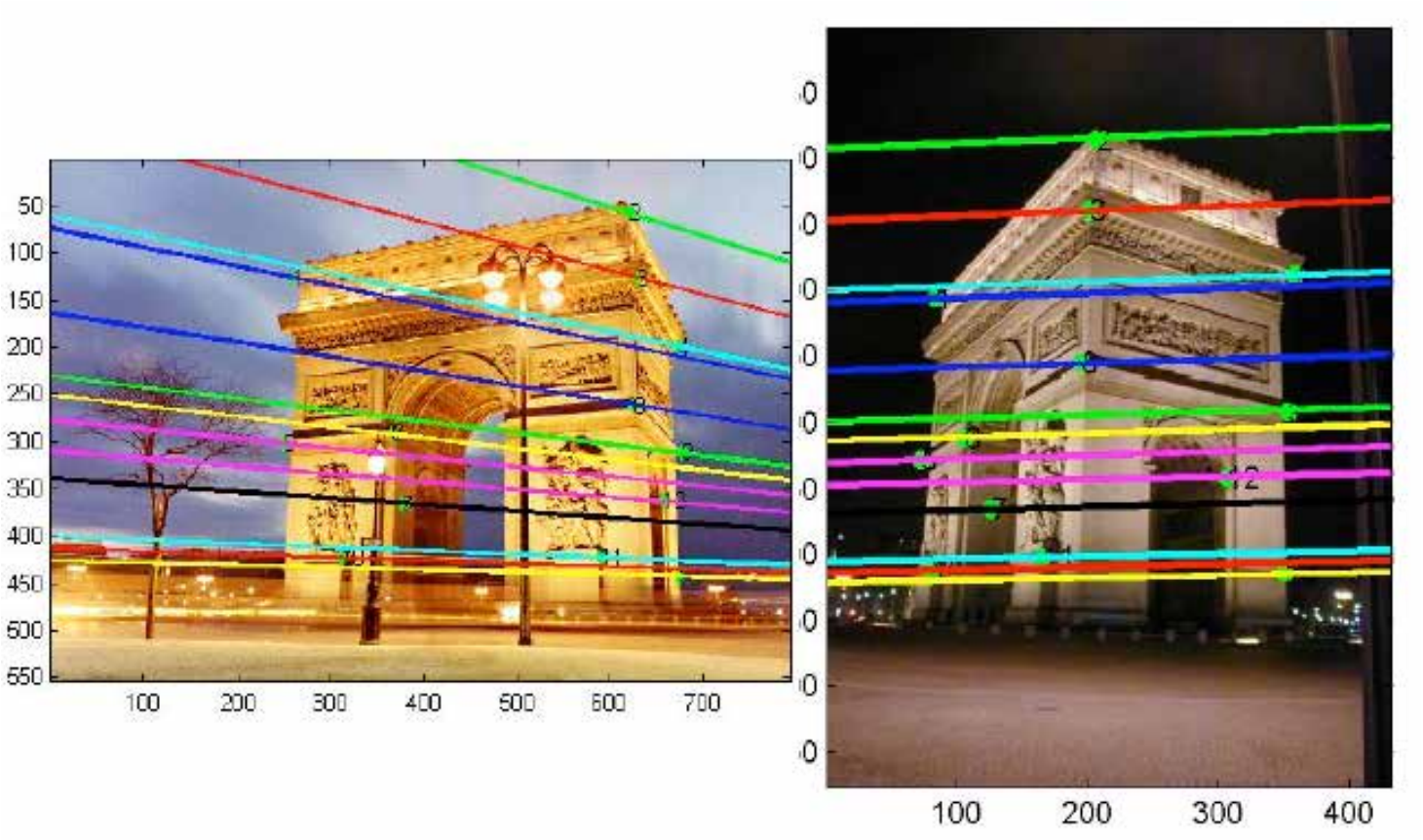
8 corresponding points, 8 equations.

$$\begin{pmatrix} u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 \\ u_2u'_2 & u_2v'_2 & u_2 & v_2u'_2 & v_2v'_2 & v_2 & u'_2 & v'_2 \\ u_3u'_3 & u_3v'_3 & u_3 & v_3u'_3 & v_3v'_3 & v_3 & u'_3 & v'_3 \\ u_4u'_4 & u_4v'_4 & u_4 & v_4u'_4 & v_4v'_4 & v_4 & u'_4 & v'_4 \\ u_5u'_5 & u_5v'_5 & u_5 & v_5u'_5 & v_5v'_5 & v_5 & u'_5 & v'_5 \\ u_6u'_6 & u_6v'_6 & u_6 & v_6u'_6 & v_6v'_6 & v_6 & u'_6 & v'_6 \\ u_7u'_7 & u_7v'_7 & u_7 & v_7u'_7 & v_7v'_7 & v_7 & u'_7 & v'_7 \\ u_8u'_8 & u_8v'_8 & u_8 & v_8u'_8 & v_8v'_8 & v_8 & u'_8 & v'_8 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \end{pmatrix} = - \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Invert and solve for $\mathcal{F}$.

(Use more points if available; find least-squares solution to minimize $\sum_{i=1}^{n} (\boldsymbol{p}_i^T \mathcal{F} \boldsymbol{p}'_i)^2$)

Example



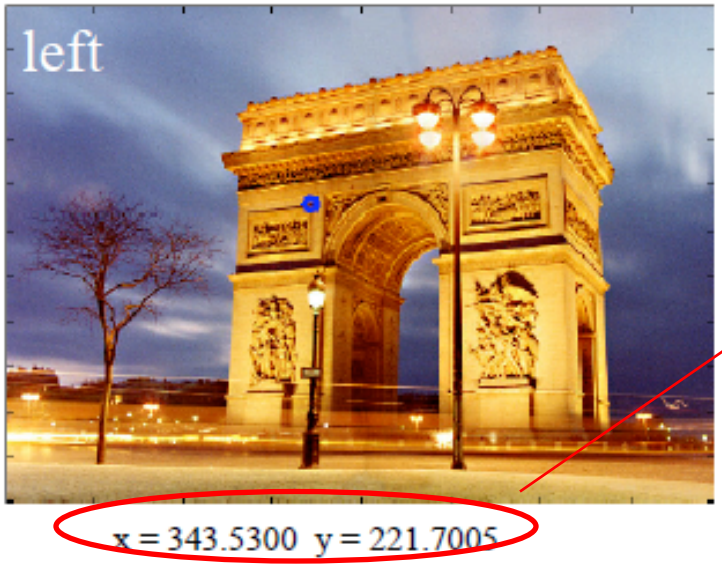
$$F = \begin{pmatrix} -0.00310695 & -0.0025646 & 2.96584 \\ -0.028094 & -0.00771621 & 56.3813 \\ 13.1905 & -29.2007 & -9999.79 \end{pmatrix}$$

http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf

Example ctd.

$u^T F u' = 0 \rightarrow Fu' = l'$, where l' is epipolar line associated to u.

$$F = \begin{pmatrix} -0.00310695 & -0.0025646 & 2.96584 \\ -0.028094 & -0.00771621 & 56.3813 \\ 13.1905 & -29.2007 & -9999.79 \end{pmatrix} \begin{pmatrix} 343.53 \\ 221.70 \\ 1.0 \end{pmatrix}$$



$$\begin{matrix} 0.0001 \\ 0.0045 \\ -1.1942 \end{matrix} \rightarrow \begin{matrix} 0.0295 \\ 0.9996 \\ -265.1531 \end{matrix}$$

normalize so sum of squares of first two terms is 1 (optional) *

***refers to normal form of line:**
rho = x cos(phi) + y sin(phi)

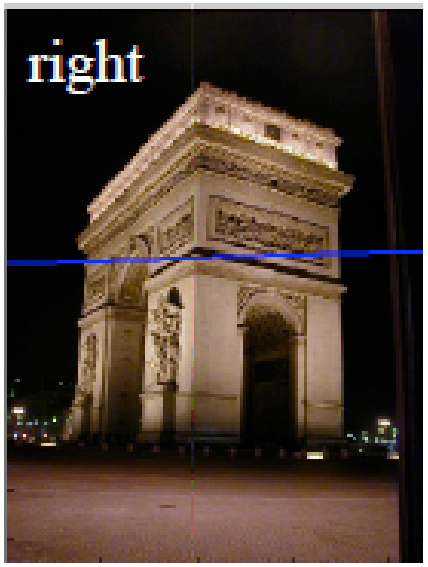
http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf

Example: Left to Right

$$F = \begin{pmatrix} -0.00310695 & -0.0025646 & 2.96584 \\ -0.028094 & -0.00771621 & 56.3813 \\ 13.1905 & -29.2007 & -9999.79 \end{pmatrix} \begin{pmatrix} 343.53 \\ 221.70 \\ 1.0 \end{pmatrix}$$

x = 343.5300 y = 221.7005

0.0295
0.9996
-265.1531

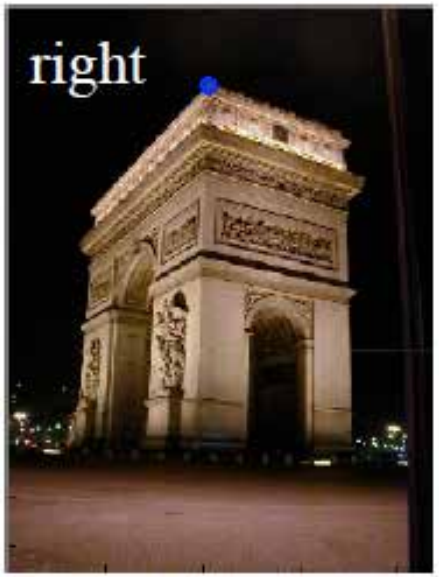


http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf

Example: Right to Left

$$(\ 205.5526 \quad 80.5 \quad 1.0) \begin{pmatrix} -0.00310695 & -0.0025646 & 2.96584 \\ -0.028094 & -0.00771621 & 56.3813 \\ 13.1905 & -29.2007 & -9999.79 \end{pmatrix}$$

$$L = (0.3211 \quad -0.9470 \quad -151.39)$$

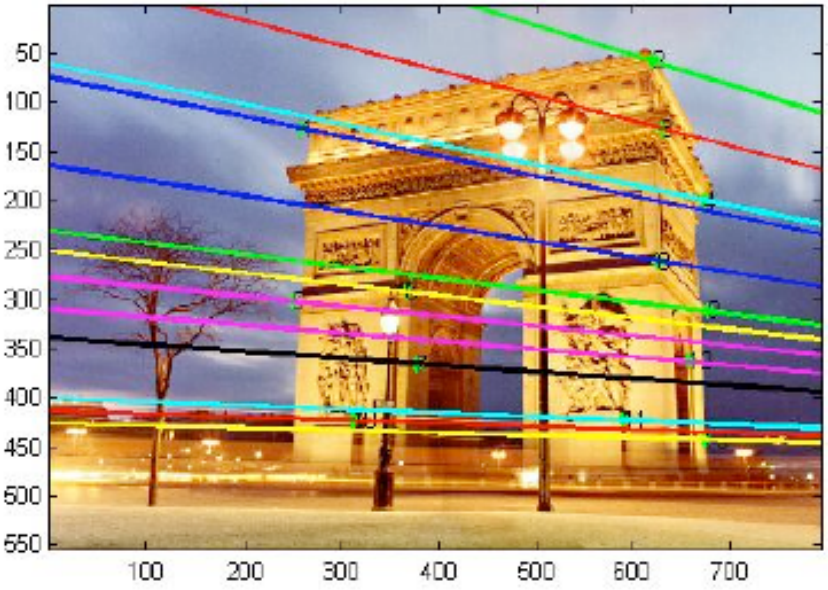


x = 205.5526 y = 80.5000

http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf

Example: Epipoles?

where is the epipole?



$$F * e_L = 0$$

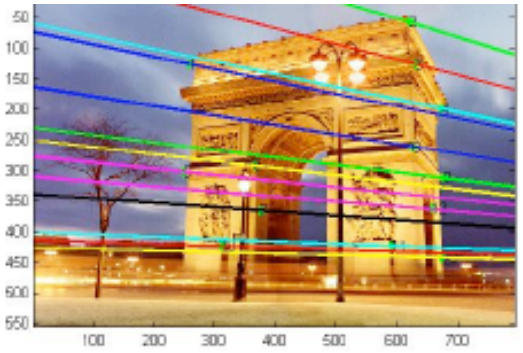
vector in the right nullspace of matrix F

However, due to noise, F may not be singular. So instead, next best thing is eigenvector associated with smallest eigenvalue of F

http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf

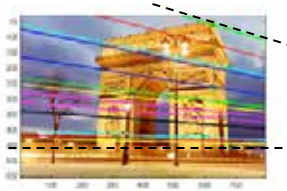
Example: Epipoles?

```
>> [u,d] = eigs(F' * F)

u =
   -0.0013    0.2586   -0.9660
    0.0029   -0.9660   -0.2586
    1.0000    0.0032   -0.0005

d = 1.0e8*
   -1.0000         0         0
         0   -0.0000         0
         0         0   -0.0000
```



eigenvector associated with smallest eigenvalue

```
>> uu = u(:,3)
uu =  (  -0.9660    -0.2586    -0.0005)

>> uu / uu(3)    : to get pixel coords
(1861.02    498.21    1.0)
```



e

http://www.cse.psu.edu/~rcollins/CSE486/lecture19_6pp.pdf