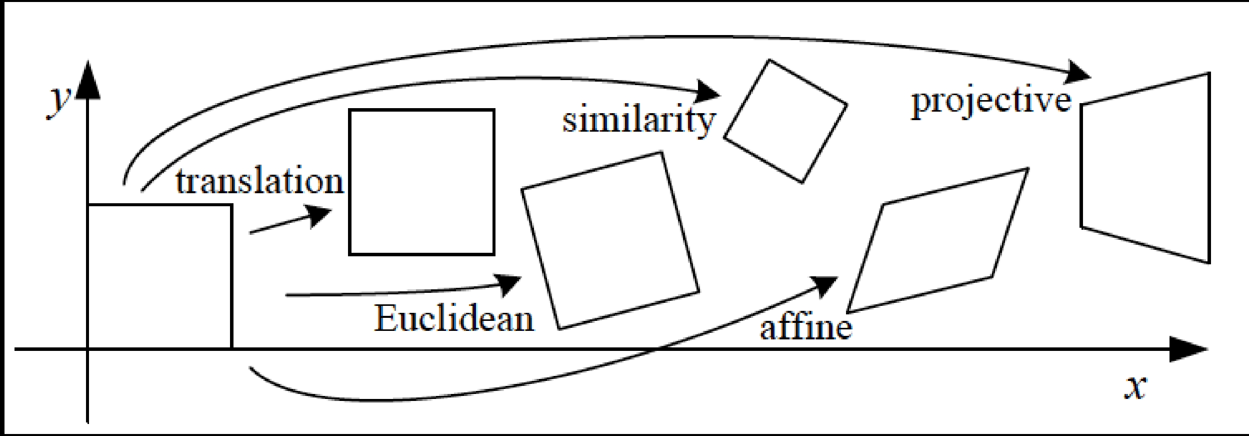
CS4495/6495
Introduction to Computer Vision

3D-L4 *Essential matrix*

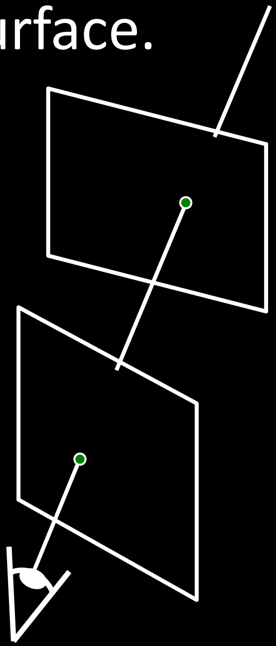
Last time

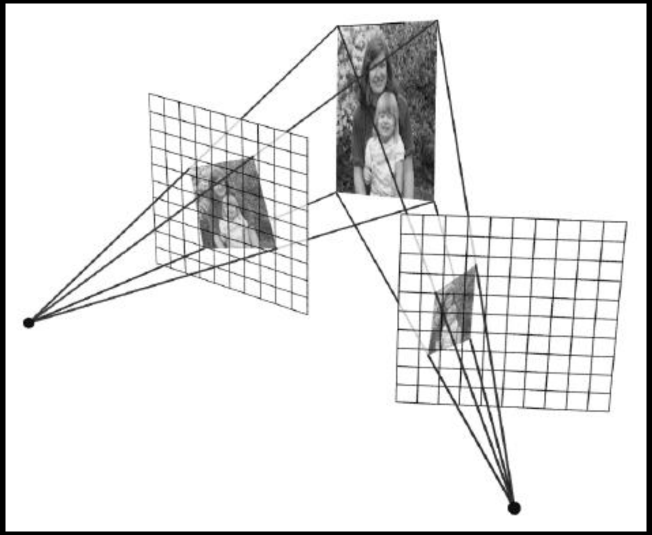
- Projective Transforms: Matrices that provide transformations including translations, rotations, similarity, affine and finally general (or perspective) projection.
- When 2D matrices are 3x3; for 3D they are 4x4.



Last time: Homographies

- Provide mapping between images (image planes) taken from same center of projection; also mapping between any images of a planar surface.

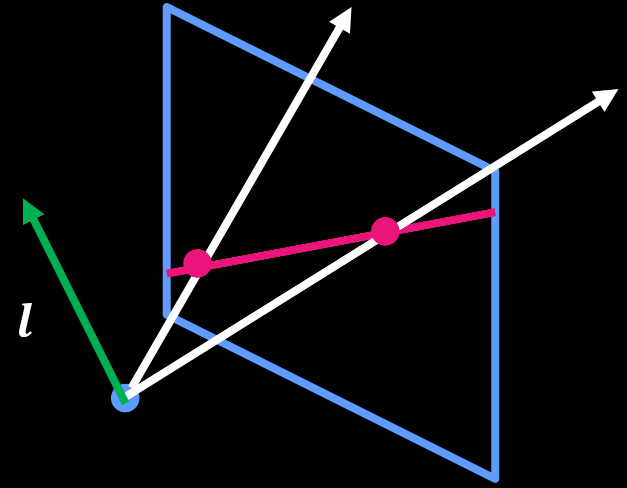




Projective lines

In Vector Notation:

$$0 = \underbrace{\begin{bmatrix} a & b & c \end{bmatrix}}_{l} \underbrace{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}_{p}$$



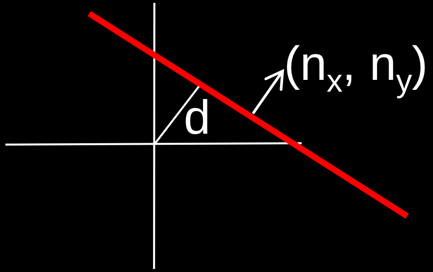
A line is also represented as a homogeneous 3-vector!

Projective Geometry: Lines and points

2D Lines: $ax + by + c = 0$

$$l = \begin{bmatrix} a & b & c \end{bmatrix} \Rightarrow \begin{bmatrix} n_x & n_y & -d \end{bmatrix}$$

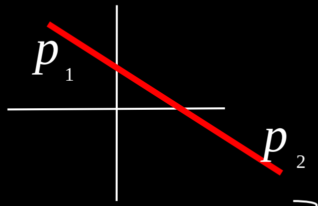
$$\begin{bmatrix} a & b & c \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0$$



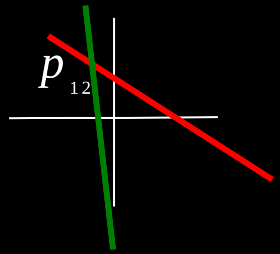
Eq of line

$$\mathbf{l}^T \mathbf{x} = 0$$

Projective Geometry: Lines and points



$$\left.\begin{array}{l} p_1 = \begin{bmatrix} x_1 & y_1 & 1 \end{bmatrix} \\ p_2 = \begin{bmatrix} x_2 & y_2 & 1 \end{bmatrix} \end{array}\right\} l = p_1 \times p_2$$

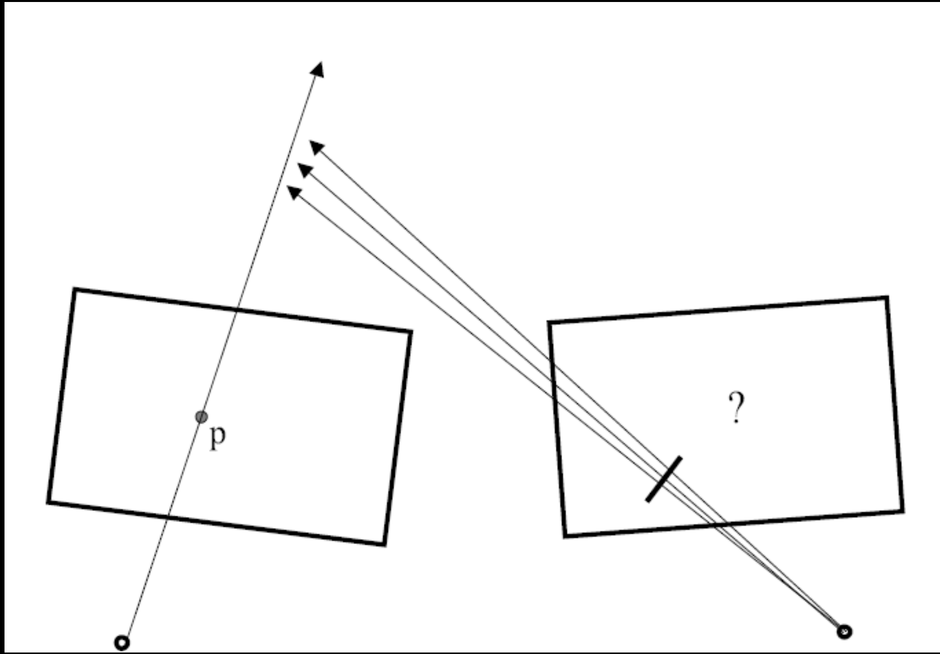


$$\left.\begin{array}{l} l_1 = \begin{bmatrix} a_1 & b_1 & c_1 \end{bmatrix} \\ l_2 = \begin{bmatrix} a_2 & b_2 & c_2 \end{bmatrix} \end{array}\right\} p_{12} = l_1 \times l_2$$

Motivating the problem: Stereo

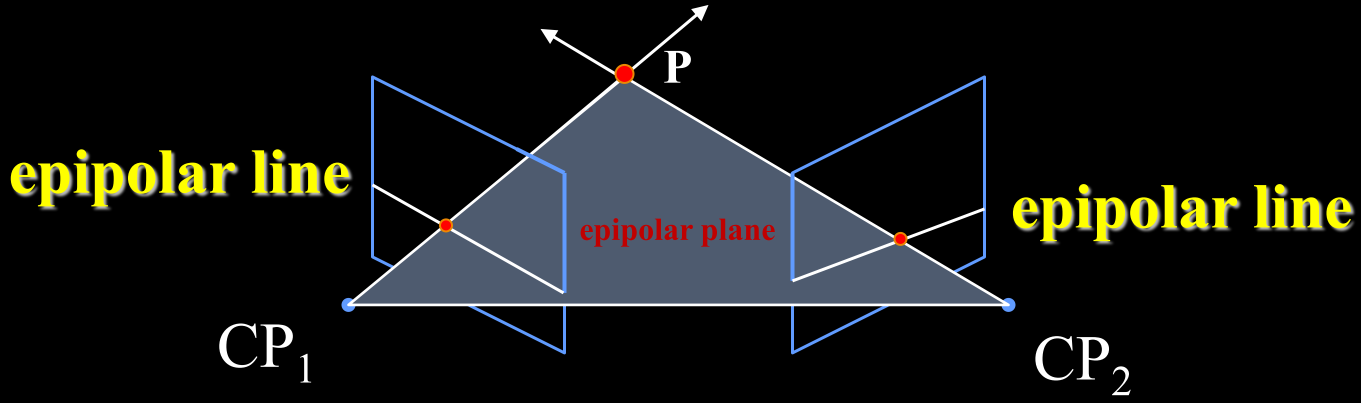
- Given two views of a scene (the two cameras not necessarily having optical axes) what is the relationship between the location of a scene point in one image and its location in the other?

Motivating the problem: Stereo



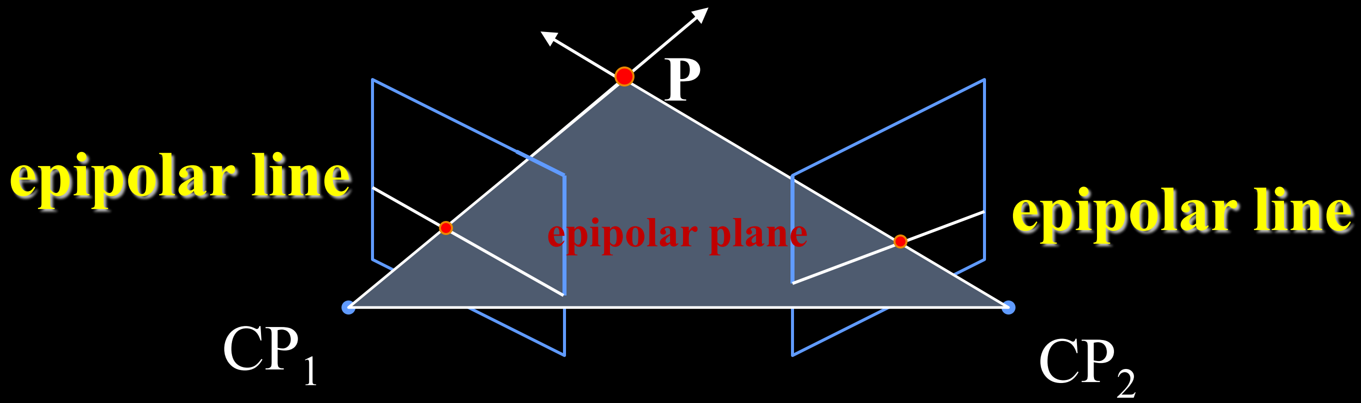
Stereo correspondence

- Find pairs of points that correspond to same scene point



Stereo correspondence

Epipolar Constraint *r*educes correspondence problem to 1D search along conjugate epipolar lines



Example: Converging cameras

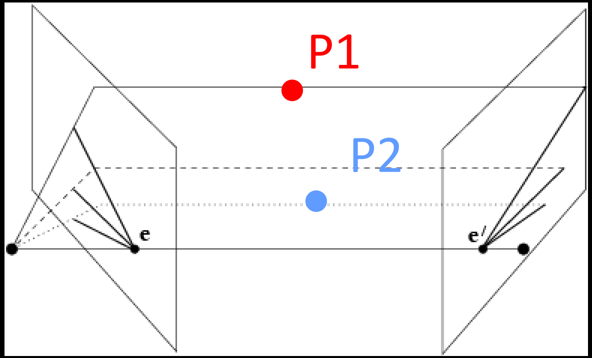
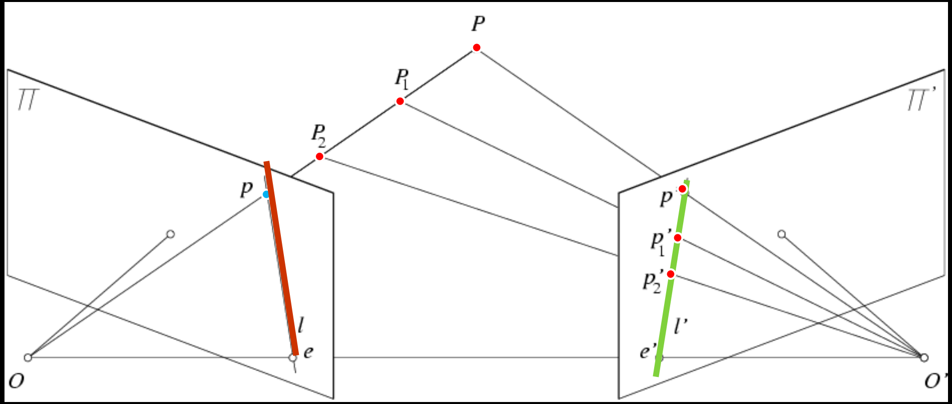


Figure from Hartley & Zisserman

Epipolar geometry: Terms

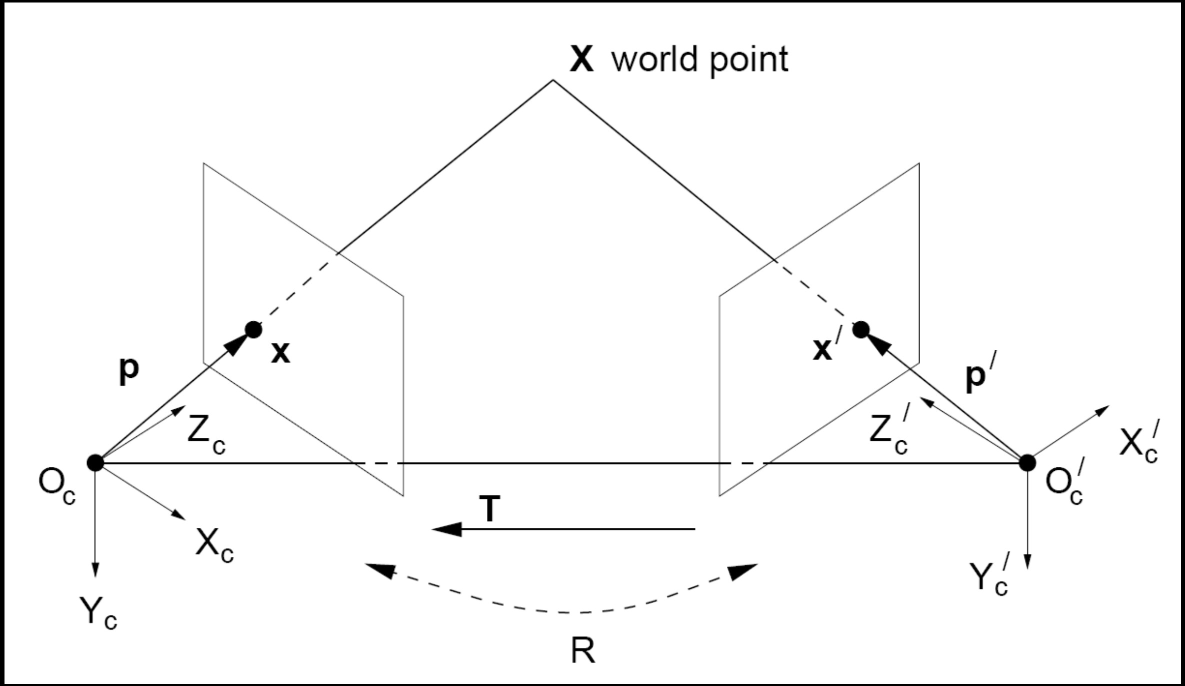
- ***Baseline***: line joining the camera centers
- *Epipolar plane*: plane containing baseline and world point
- *Epipolar line*: intersection of epipolar plane with the image plane – come in pairs
- *Epipole*: point of intersection of baseline with image plane



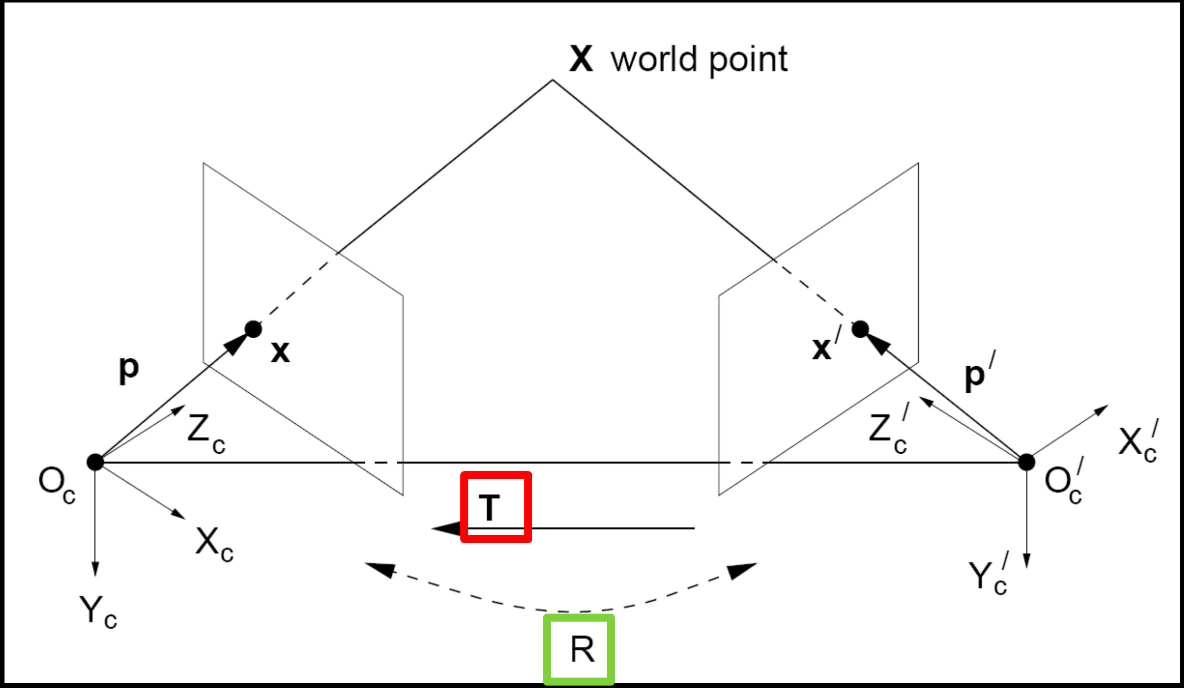
From Geometry to Algebra

- So far, we have the explanation in terms of geometry.
- Now, how do we express the epipolar constraints algebraically?

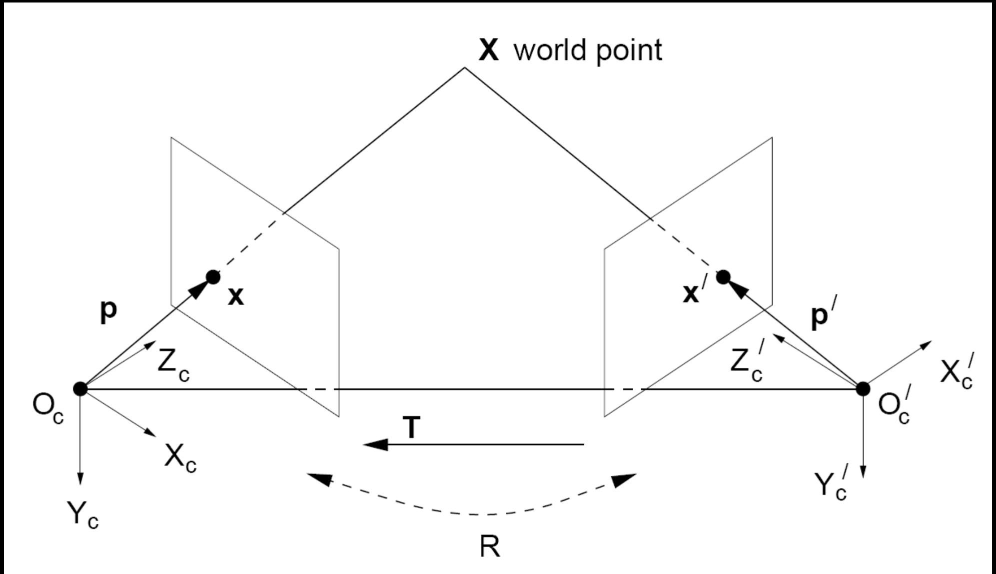
Stereo geometry, with calibrated cameras



Stereo geometry, with calibrated cameras



From geometry to algebra



$$\mathbf{X}'_c = \mathbf{R}\mathbf{X}_c + \mathbf{T}$$

Aside 1: Reminder of cross product

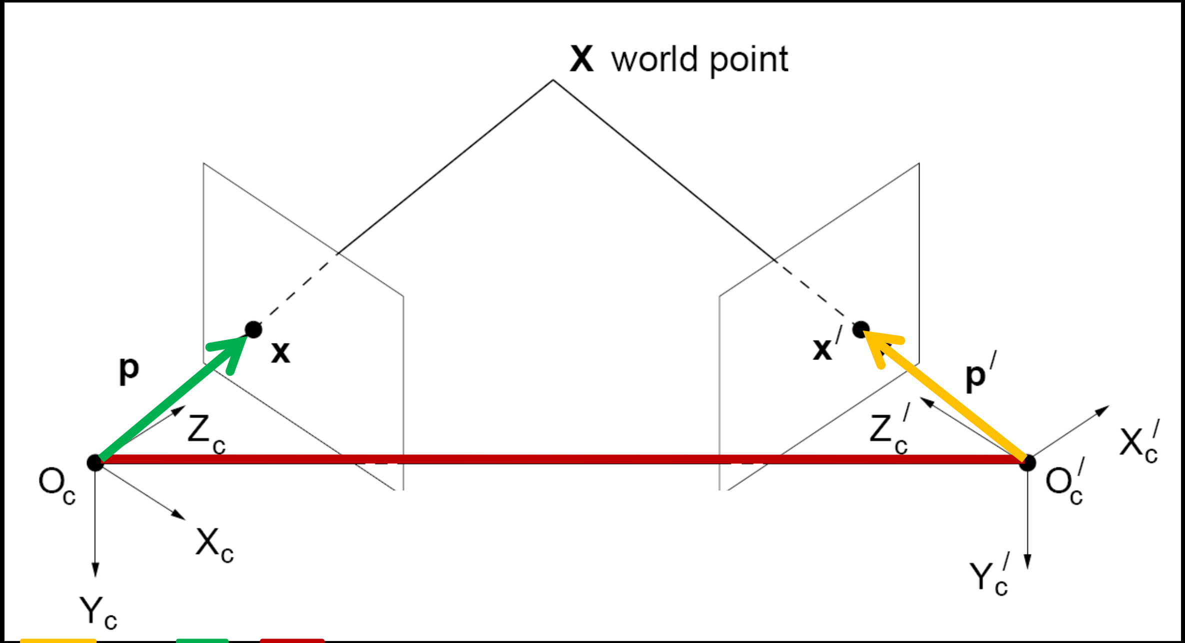
Vector cross product takes two vectors and returns a third vector that's perpendicular to both inputs.

$$\vec{a} \times \vec{b} = \vec{c}$$

Here c is perpendicular to both a and b, i.e. the dot product = 0.

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

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



$$\boxed{\mathbf{X}'} = \mathbf{R}\,\boxed{\mathbf{X}} + \boxed{\mathbf{T}}$$

$$\underbrace{\mathbf{T} \times \mathbf{X}'}_{\text{Normal to the plane}} = \mathbf{T} \times \mathbf{R}\,\mathbf{X}$$

$$\mathbf{X}' \cdot (\mathbf{T} \times \mathbf{X}') = \mathbf{X}' \cdot (\mathbf{T} \times \mathbf{R}\,\mathbf{X})$$

$$0 = \mathbf{X}' \cdot (\mathbf{T} \times \mathbf{R}\,\mathbf{X})$$

Aside 2: Matrix form of cross product

$$\vec{a} \times \vec{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \vec{c}$$

Can be expressed as a matrix multiplication!!!

Aside 2: Matrix form of cross product

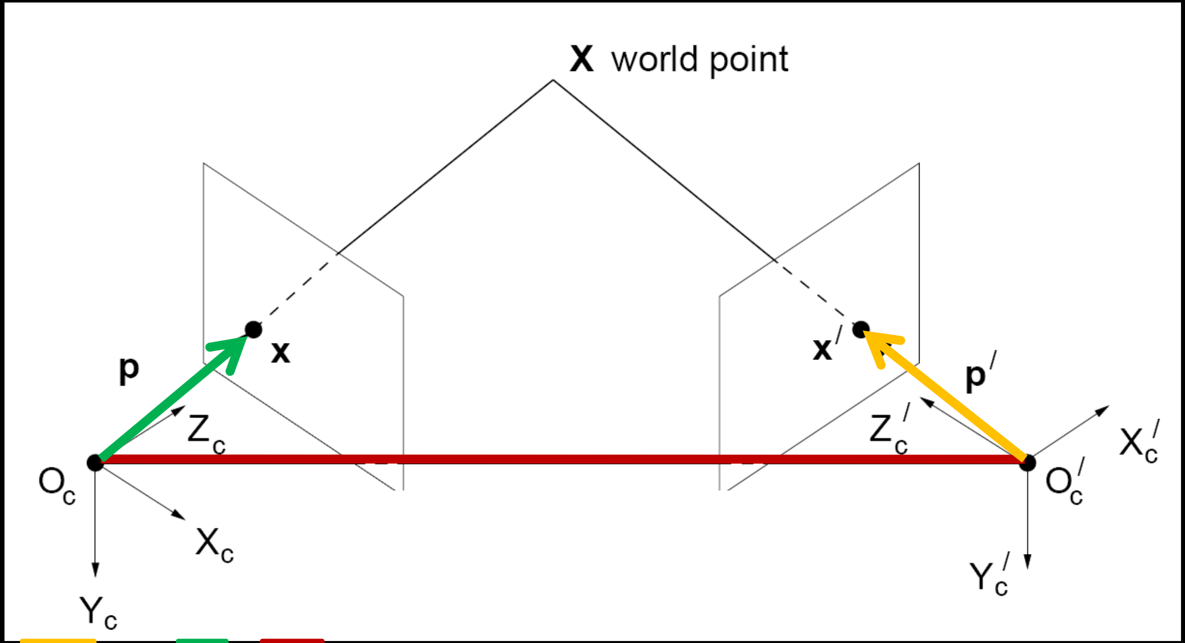
Can define a cross product matrix operation:

$$[a_x] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

Notation:

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

Has rank 2!



$$\boxed{X'} = R\,\boxed{X} + \boxed{T}$$

$$\underbrace{T \times X'}_{\text{Normal to the plane}} = T \times R\,X$$

$$X' \cdot \left(T \times X' \right) = X' \cdot \left(T \times R\,X \right)$$

$$\boxed{0 = X' \cdot \left(T \times R\,X \right)}$$

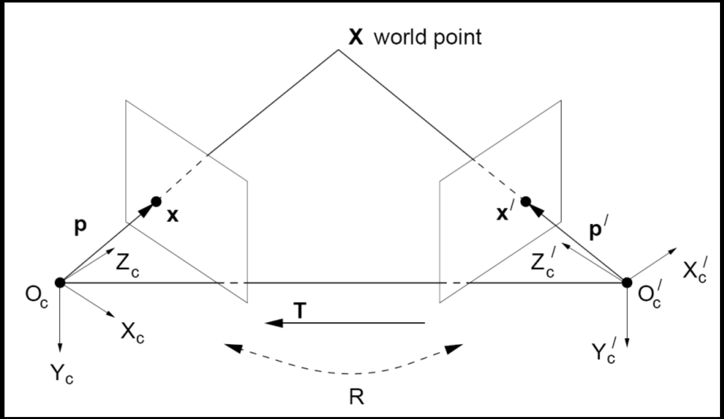
Essential matrix

$$\boxed{\mathbf{X}' \cdot (\mathbf{T} \times \mathbf{R}\mathbf{X}) = 0}$$

$$\mathbf{X}' \cdot ([\mathrm{T}_x]\mathbf{R}\mathbf{X}) = 0$$

$$\text{Let} \quad \mathbf{E} = [\mathrm{T}_x]\mathbf{R}$$

$$\boxed{\mathbf{X}'^{T}\mathbf{E}\mathbf{X} = 0}$$



E is called the "essential matrix".

Quiz

- That’s fine for some converged cameras. But what if the image planes are parallel. What happens?

a) That is a degenerate case. You’ll see in a bit.

b) That’s fine. R is just the identity and the math works.

c) I have no idea.

Quiz – answer

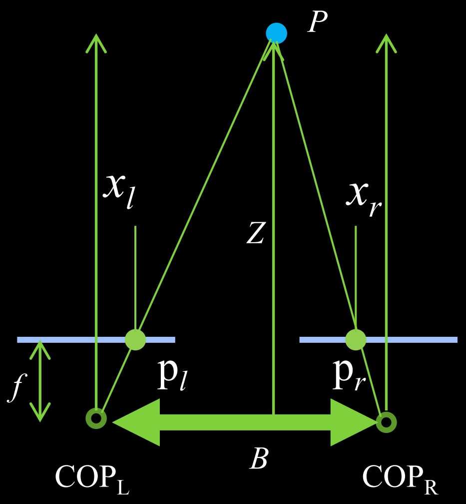
- That's fine for some converged cameras. But what if the image planes are parallel. What happens?

a) That is a degenerate case. You'll see in a bit.

b) That's fine. R is just the identity and the math works.

c) I have no idea.

Essential matrix example: parallel cameras

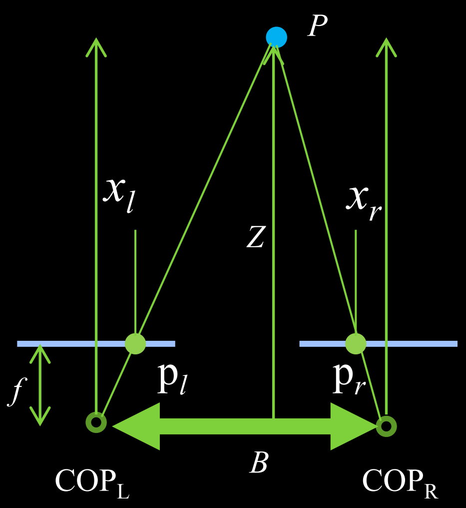


$$\mathbf{R} = \mathbf{I}$$

$$\mathbf{T} = [-B, 0, 0]^{\mathrm{T}}$$

$$\mathbf{E} = [\mathrm{T}_{\mathbf{x}}]\mathbf{R} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & B \\ 0 & -B & 0 \end{pmatrix}$$

Essential matrix example: parallel cameras



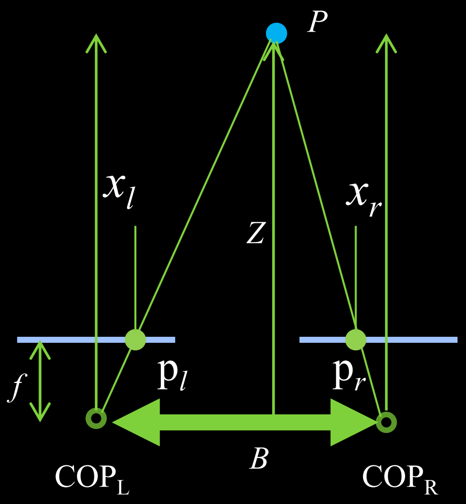
$$\mathbf{p}'^{T}\mathbf{E}\,\mathbf{p} = 0$$

$$\mathbf{p} = [X, Y, Z] = [\frac{Zx}{f}, \frac{Zy}{f}, Z]$$

$$\mathbf{p}' = [X', Y', Z] = [\frac{Zx'}{f}, \frac{Zy'}{f}, Z]$$

$$\begin{bmatrix} x' & y' & f \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & B \\ 0 & -B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ f \end{bmatrix} = 0$$

Essential matrix example: parallel cameras



$$\begin{bmatrix} x' & y' & f \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & B \\ 0 & -B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ f \end{bmatrix} = 0$$

$$\begin{bmatrix} x' & y' & f \end{bmatrix} \begin{bmatrix} 0 \\ Bf \\ -By \end{bmatrix} = 0$$

$$Bfy' = Bfy \Rightarrow \boxed{\mathbf{y' = y}}$$

Given a known point (x,y) in the original image, this is a *line* in the (x',y') image.