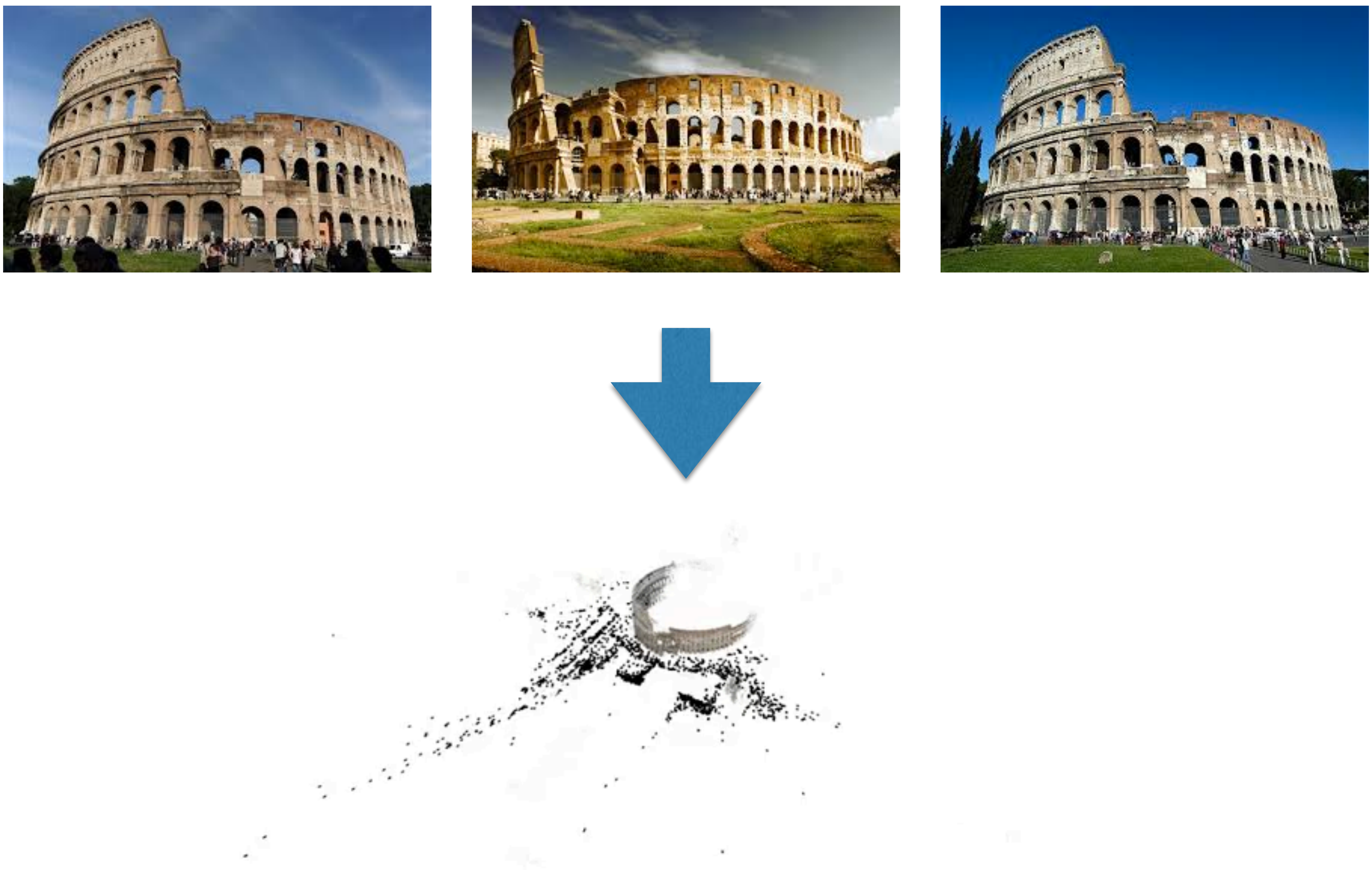
Two-view geometry

Multi-view geometry



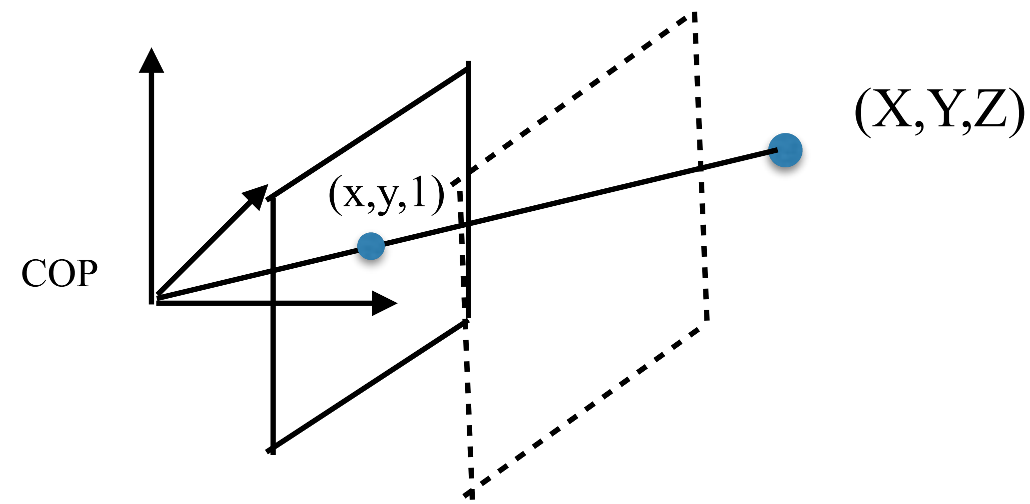
Three questions:

(i) **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?

(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?

Recall perspective projection

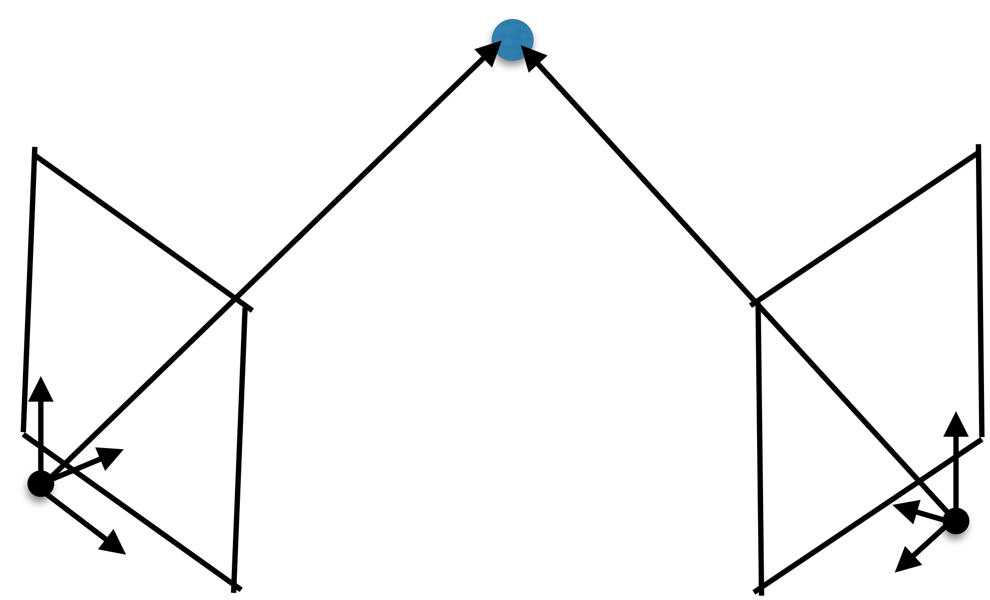


Assume *calibrated* camera with known intrinsics (f = 1)

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Goal: let's build intuition with pictures rather than math

Two-view (stereo)

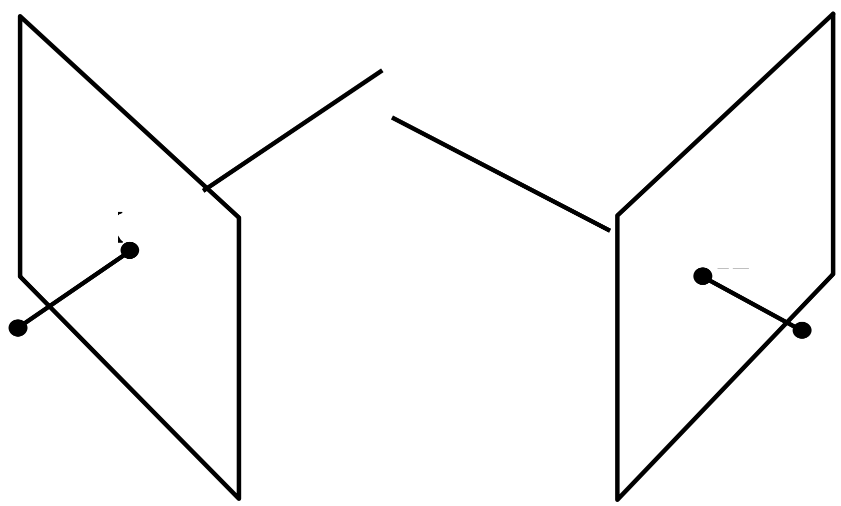


Given 2 corresponding points in 2 cameras, see where the cast rays meet

Much of basics can be derived from this picture

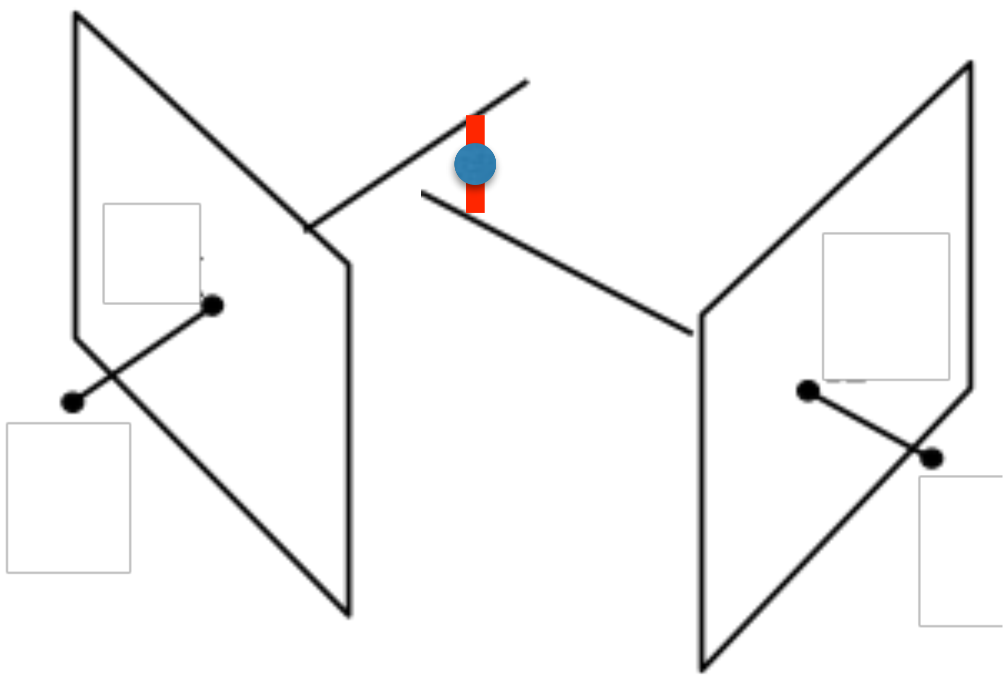
Goal: let's get the geometric intuition before the math

An annoying "detail"

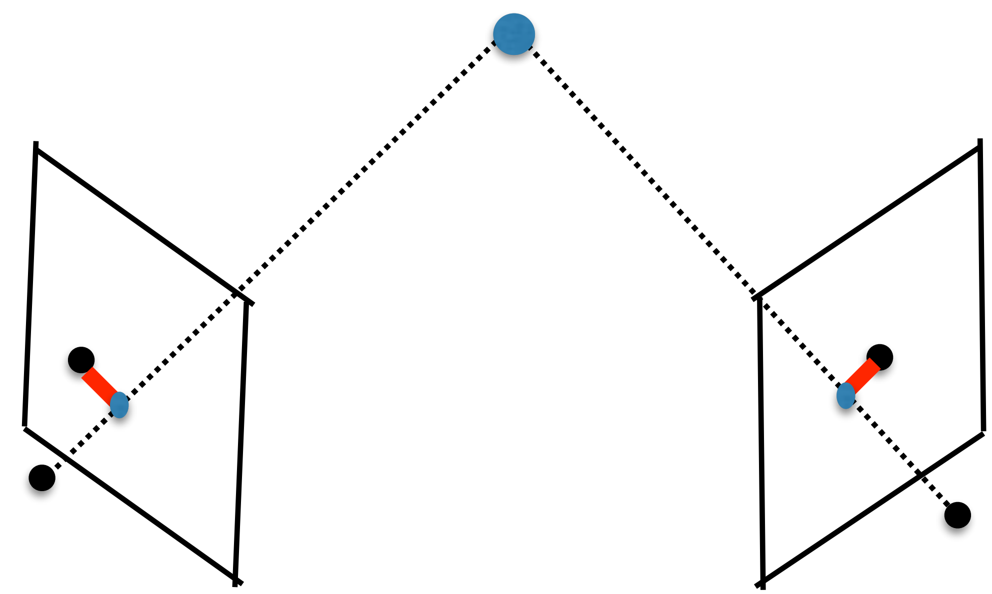


What to do when ray's don't intersect?

Possible solns



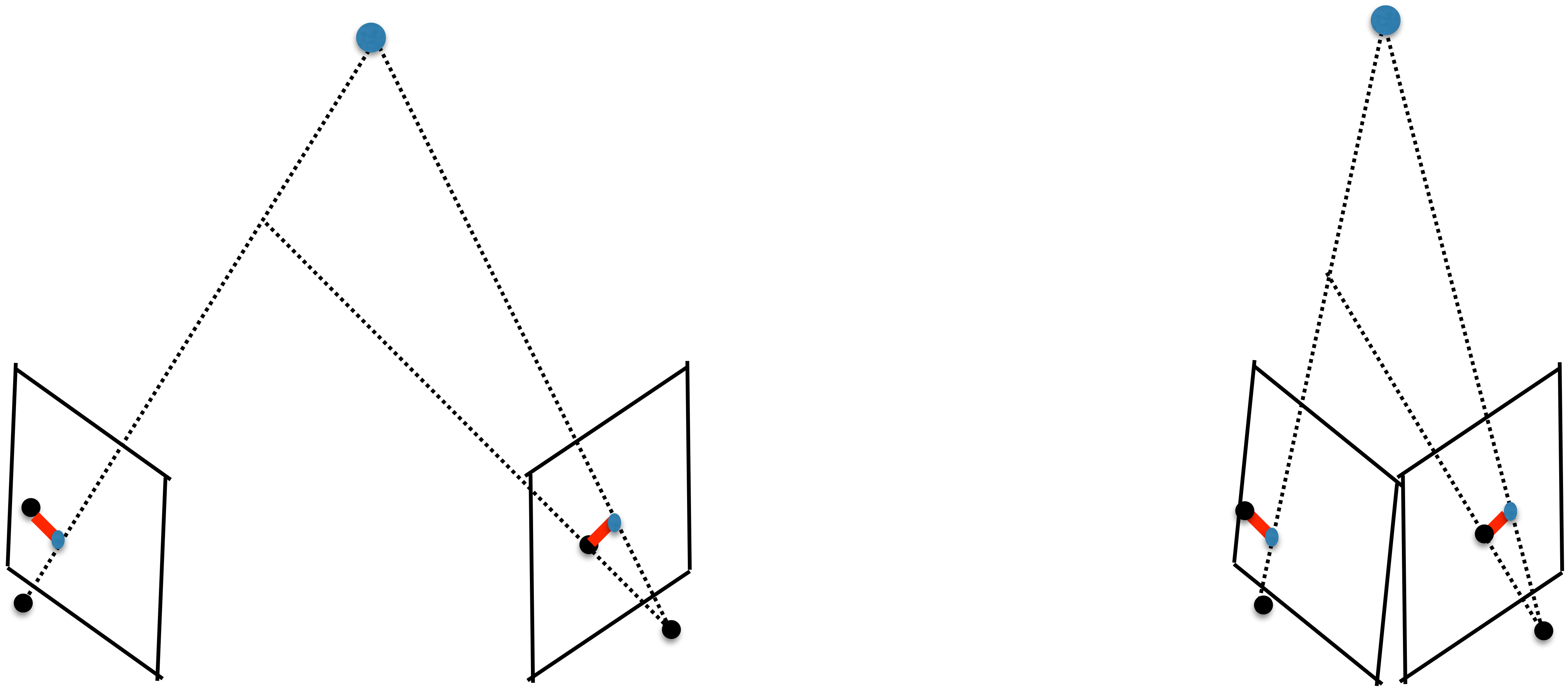
Find 3D point closest to 2 rays



Find 3D point with low reprojection error

Which makes sense?

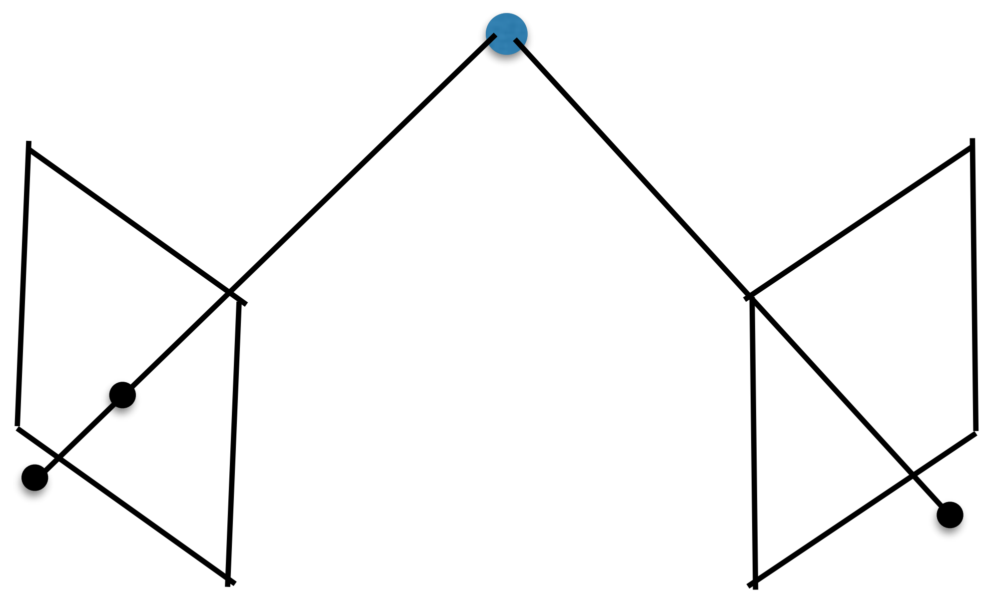
Numerical stability



Which setup produces noisier estimates of depth (as a function of image noise)?

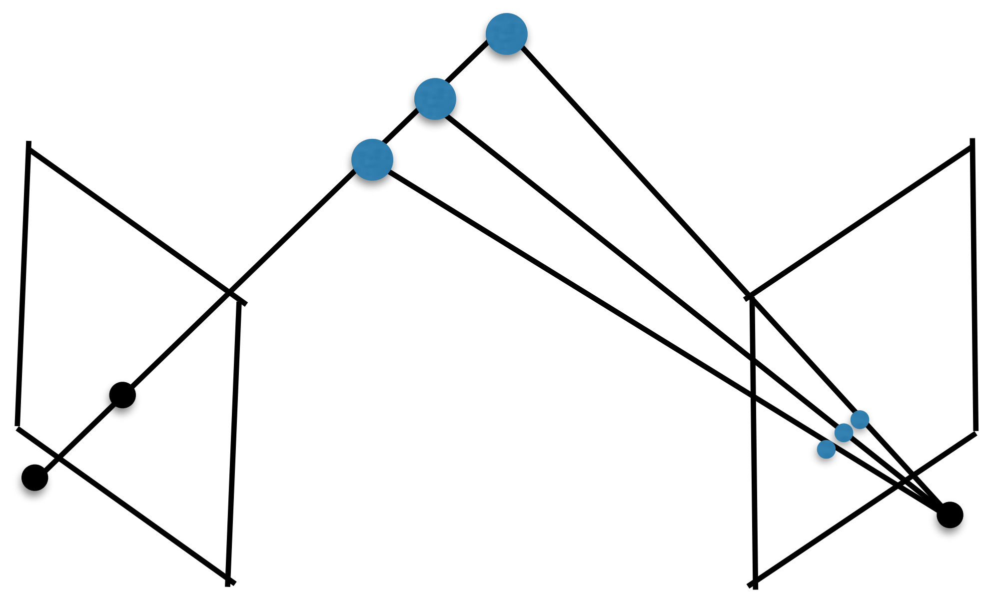
Which setup will produce points that are easier to match?

Questions



Given a point in left view, what is the set of points it could project to in right view?

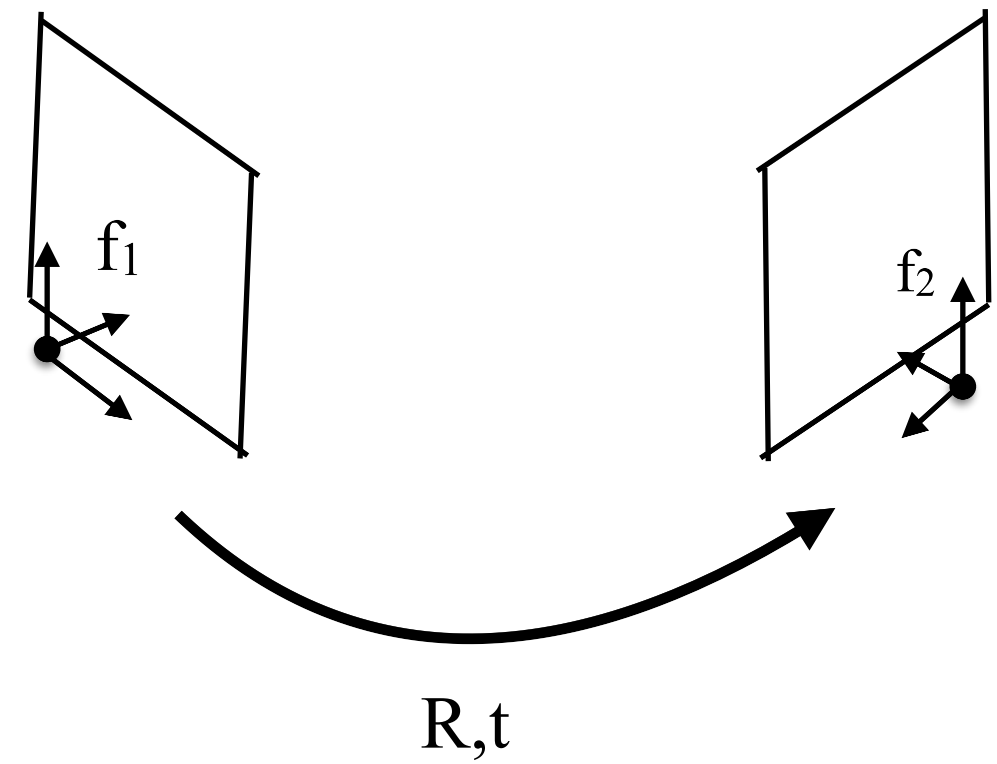
Questions



Given a point in left view, what is the set of points it could project to in right view?

Implies that *for known camera geometry,* we need search for correspondence only over 1D line

Epipolar geometry

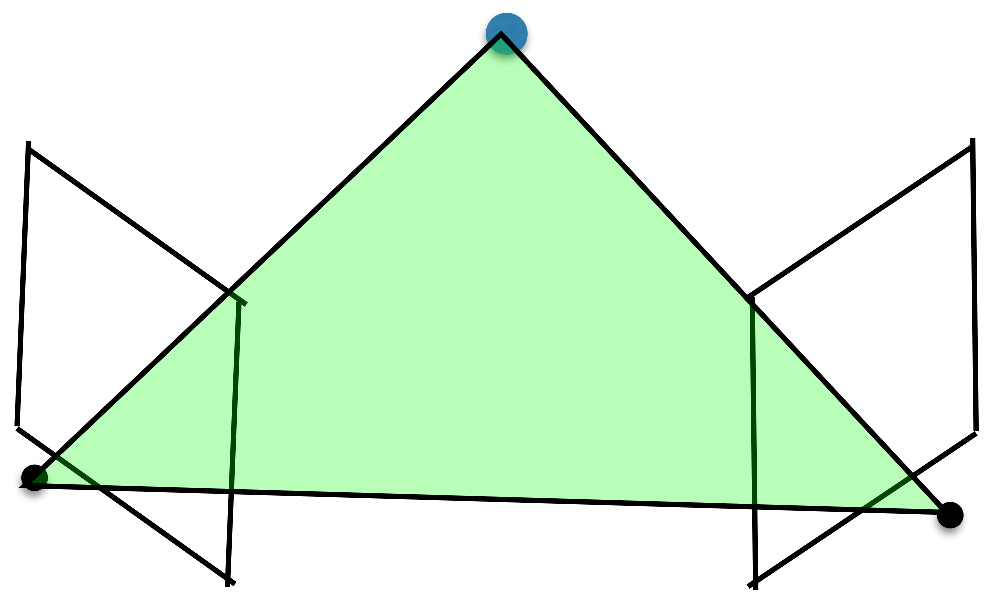


Epipolar geometry describes the set of candidate correspondences across 2 views as a function of camera extrinsics (R,t) and intrinsics (f)

Epipolar geometry is *not* a function of the 3D scene

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

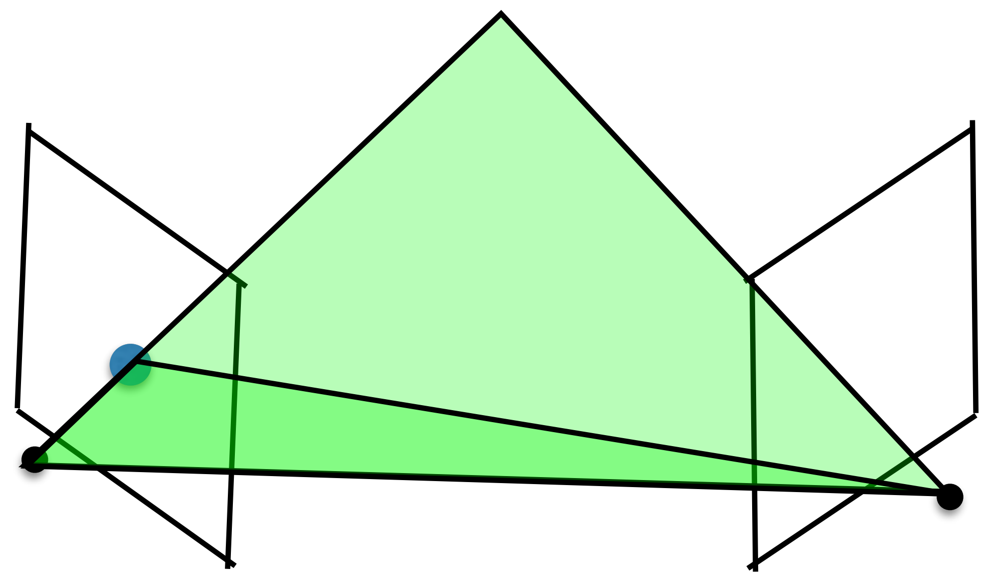
Definitions



Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

…but didn't we just state that epipolar geometry doesn't depend on the 3D scene?

Definitions

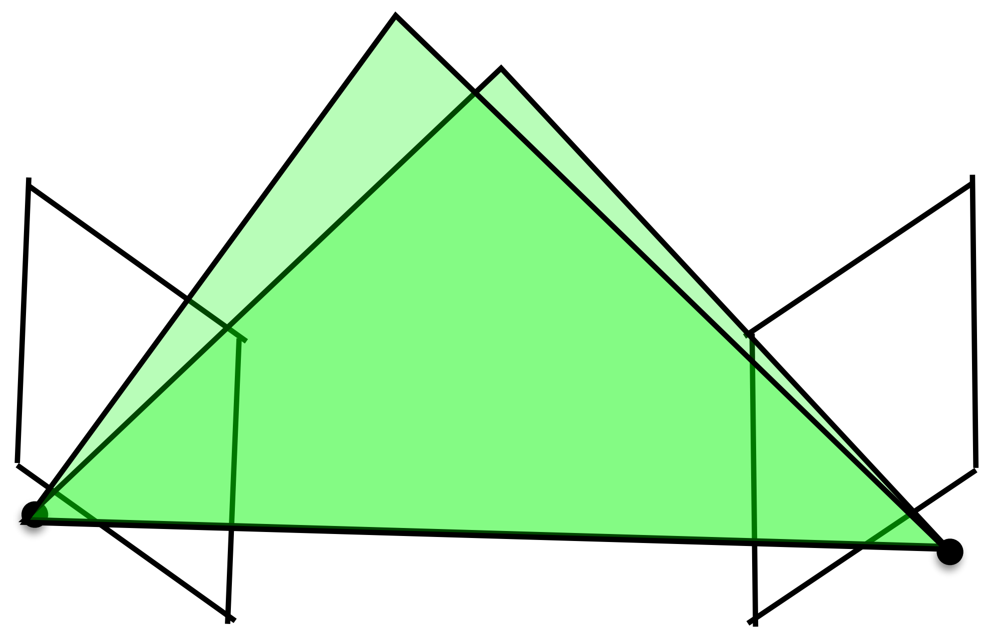


Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

(formally defined by 2 camera centers any 1 point in either image plane;
for convenience, we'll draw the triangle connecting to 3D point)

How does epipolar plane change when we double distance between two cameras?

Definitions

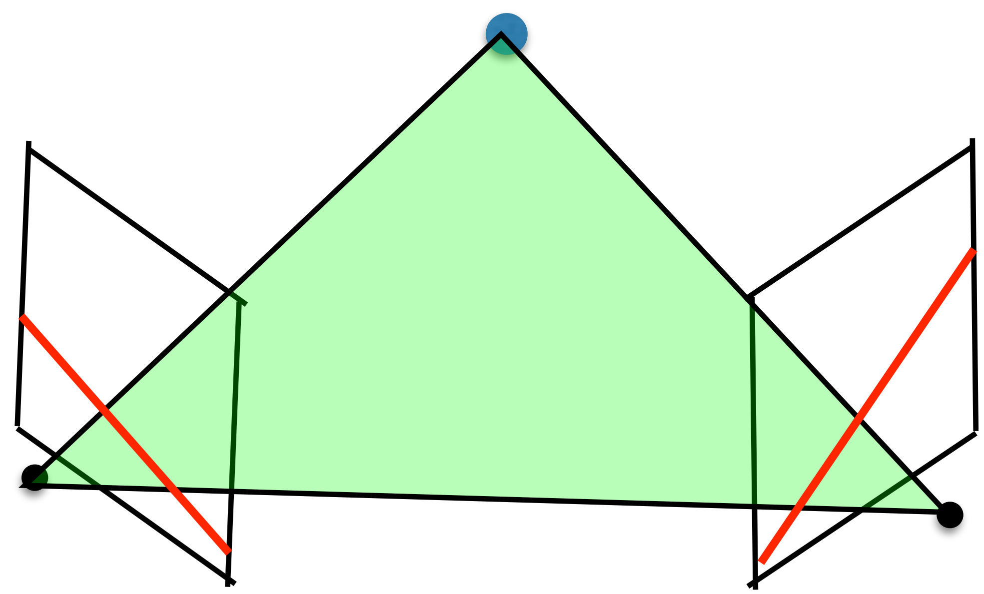


Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

How large is the *family* of epipolar planes?

We’ll show a neat way to index this family in a few slides

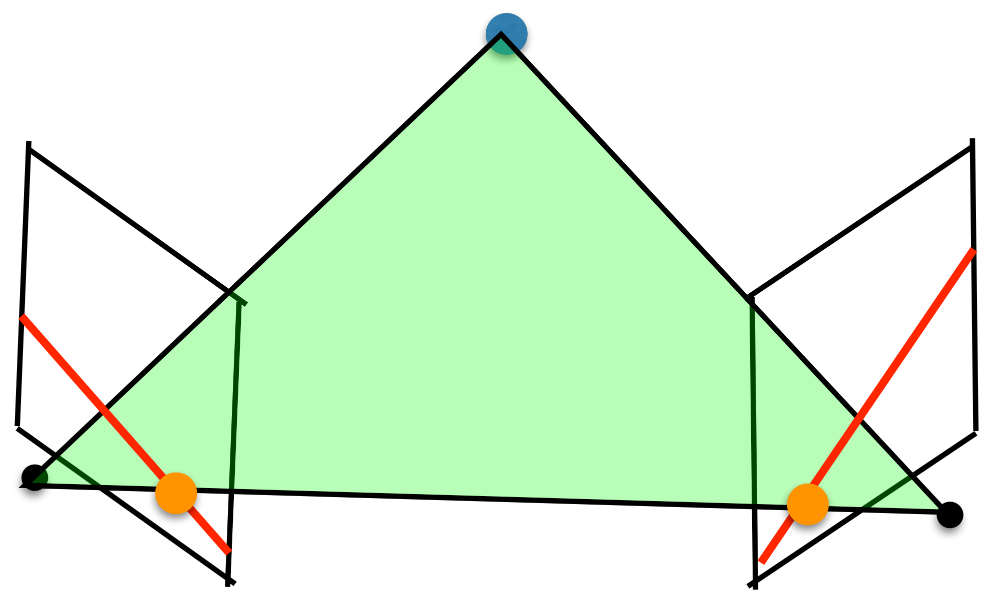
Definitions



Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

Epipolar lines: intersection of epipolar plane and image planes (red)

Definitions



Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

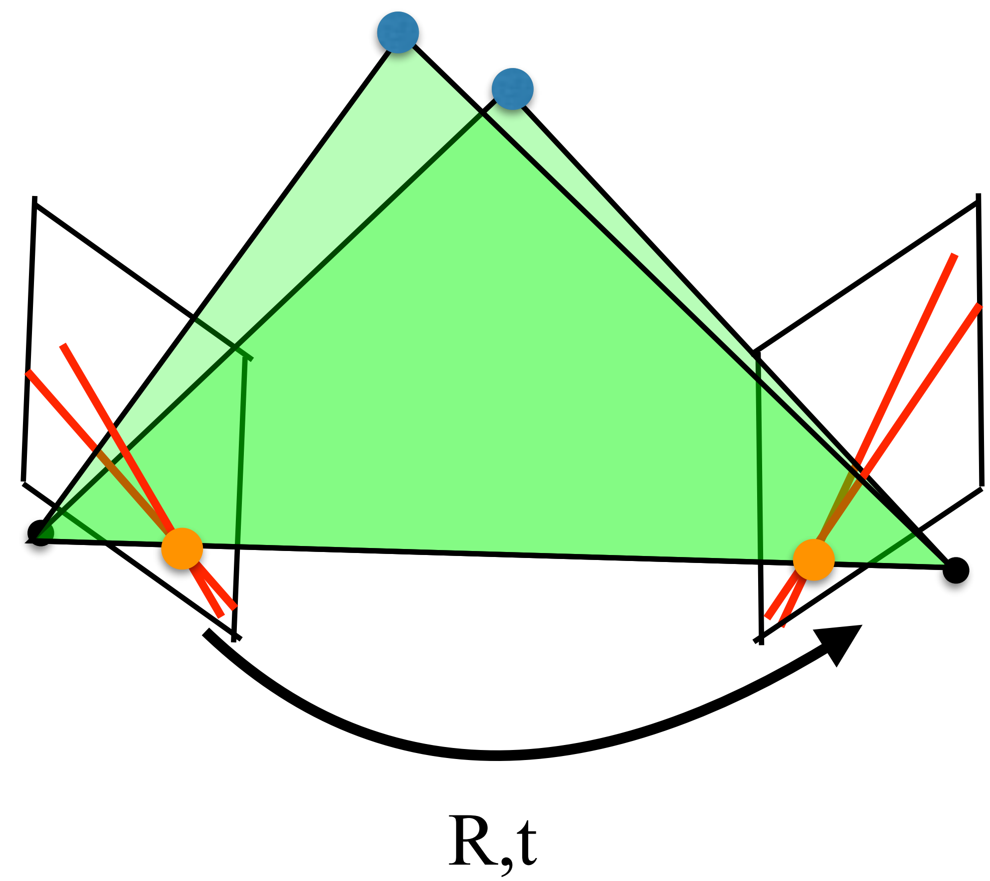
(also defined by 2 camera centers any 1 points in either image plane)

Epipolar lines: intersection of epipolar plane and image planes (red)

Epipoles: projection of camera center 1 in camera 2 (& vice versa) (orange)

Definitions

What happens if we scale translation vector?



Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

(also defined by 2 camera centers any 1 points in either image plane)

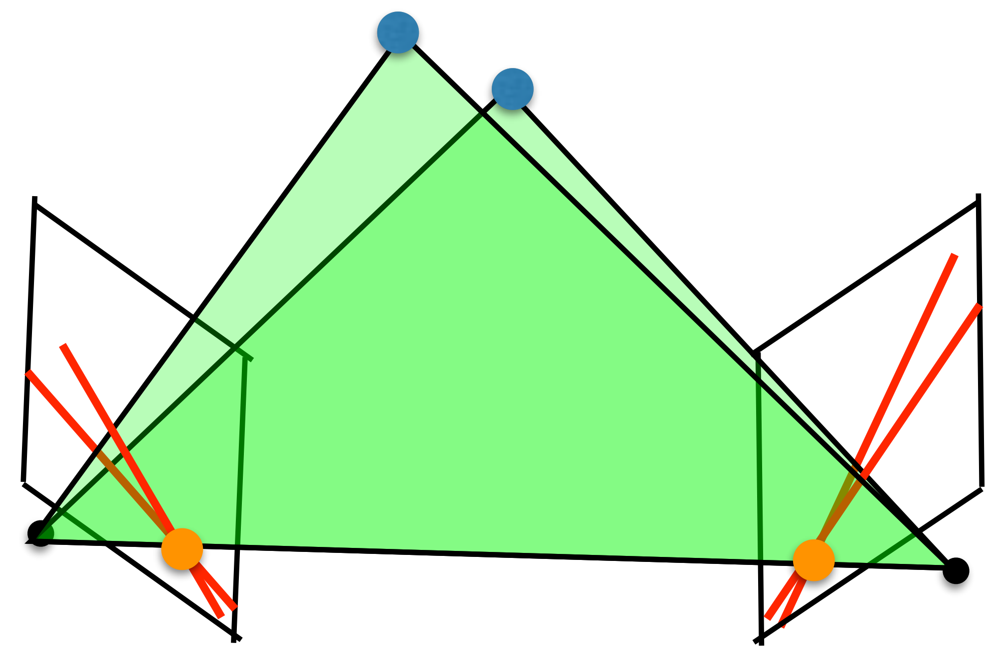
Epipolar lines: intersection of epipolar plane and image planes (red)

Epipoles: projection of camera center 1 in camera 2 (& vice versa) (orange)

(set of all epipolar lines intersect at the epipoles)

(imples set of all epipolar planes can be indexed by 2D angle)

Definitions



Epipolar plane: plane defined by 2 camera centers & candidate 3D point (green)

(also defined by 2 camera centers any 1 points in either image plane)

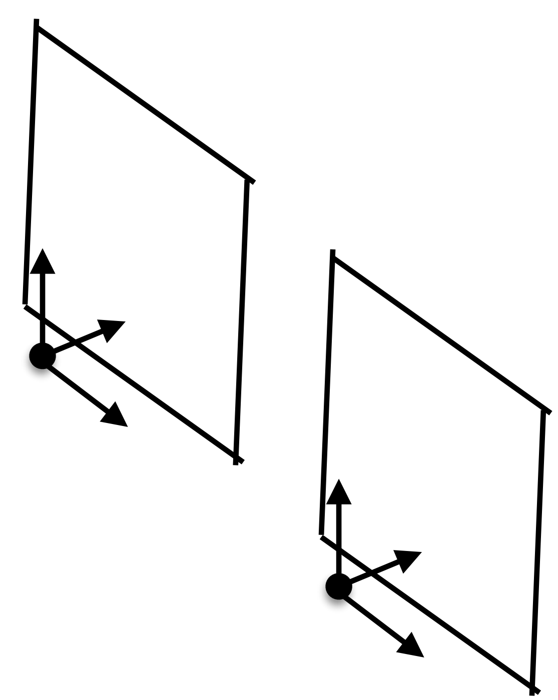
Epipolar lines: intersection of epipolar plane and image planes (red)

Epipoles: projection of camera center 1 in camera 2 (& vice versa) (orange)

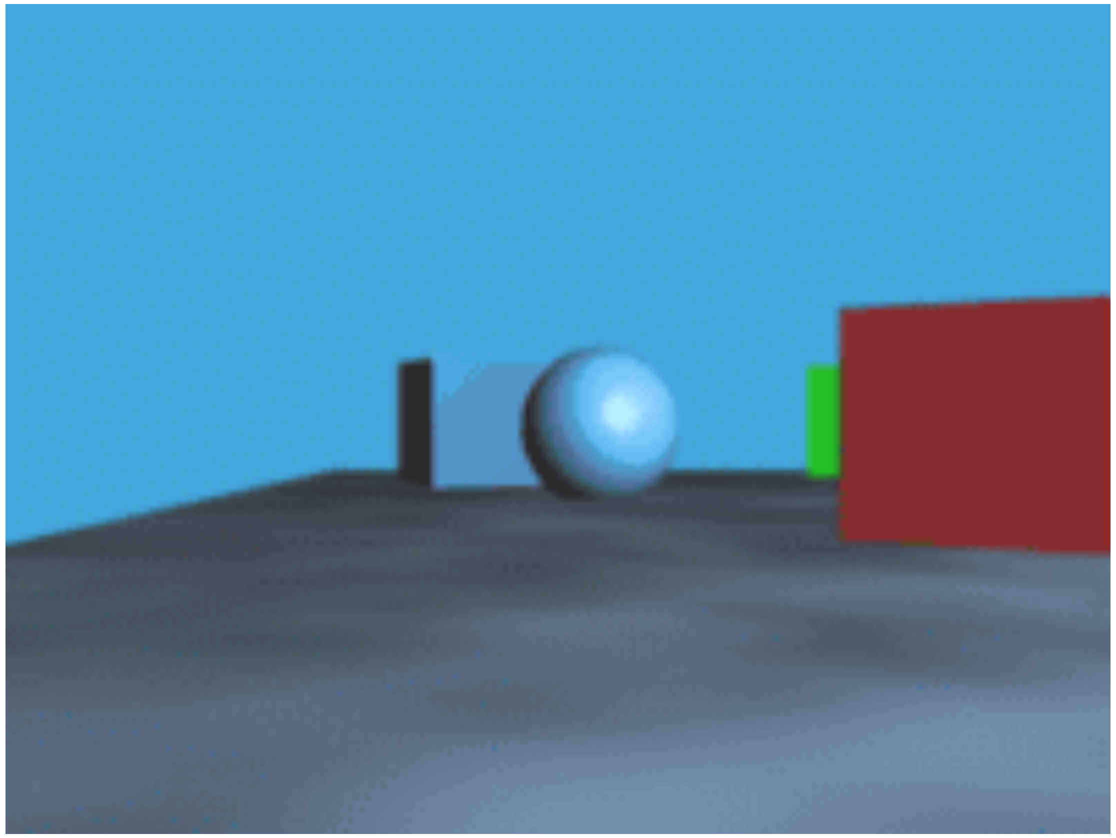
(set of all epipolar lines intersect at the epipoles)

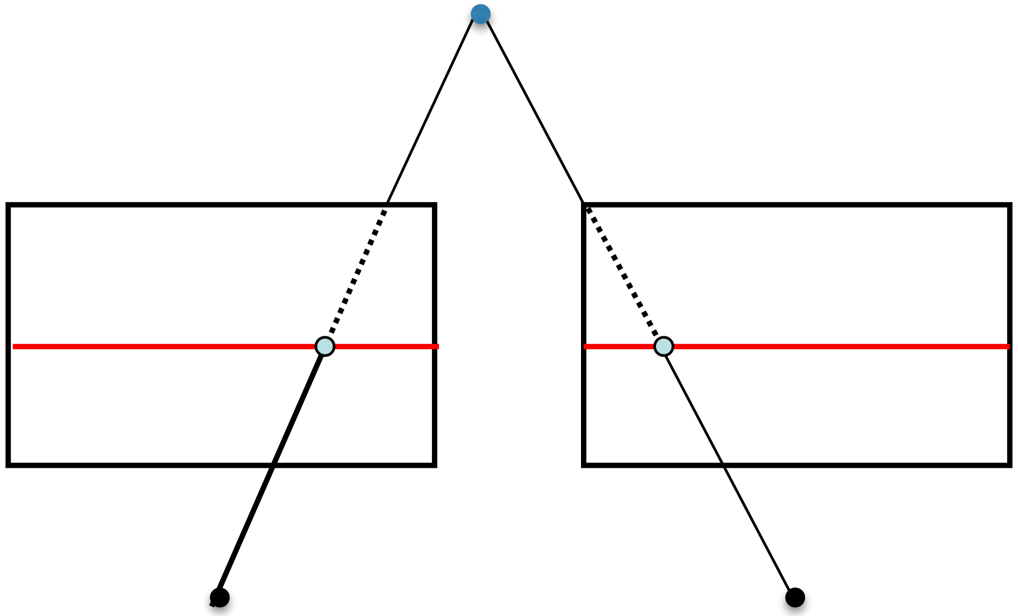
Special case (I)

Parallel, offset cameras



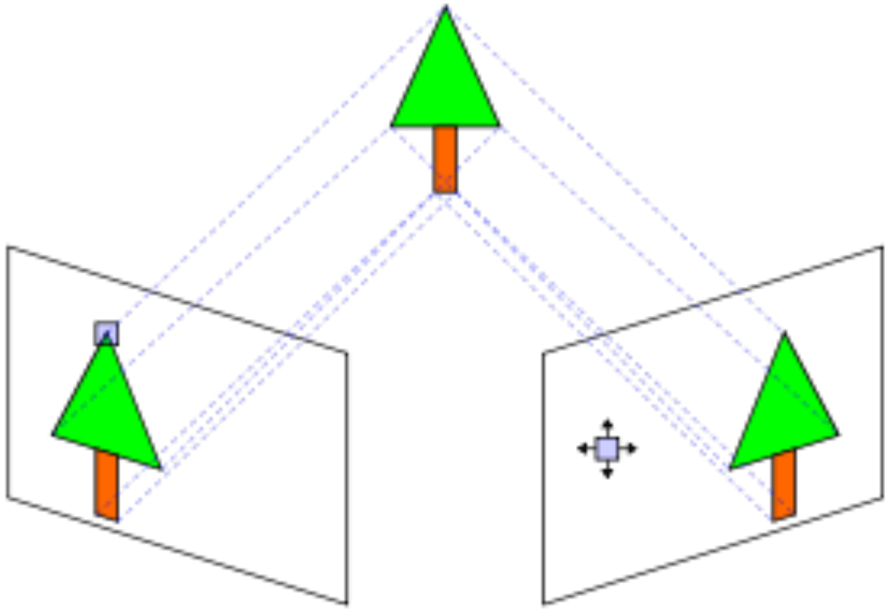
What would epipolar lines look like?



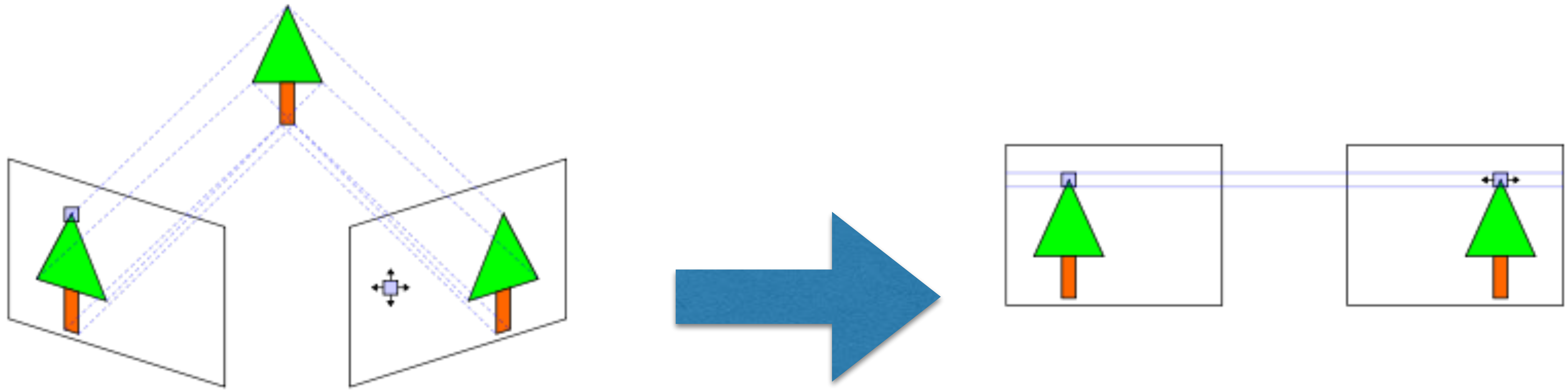


Epipolar lines don’t intersect (are parallel)
Epipoles are at infinity (derive by rotating image planes)

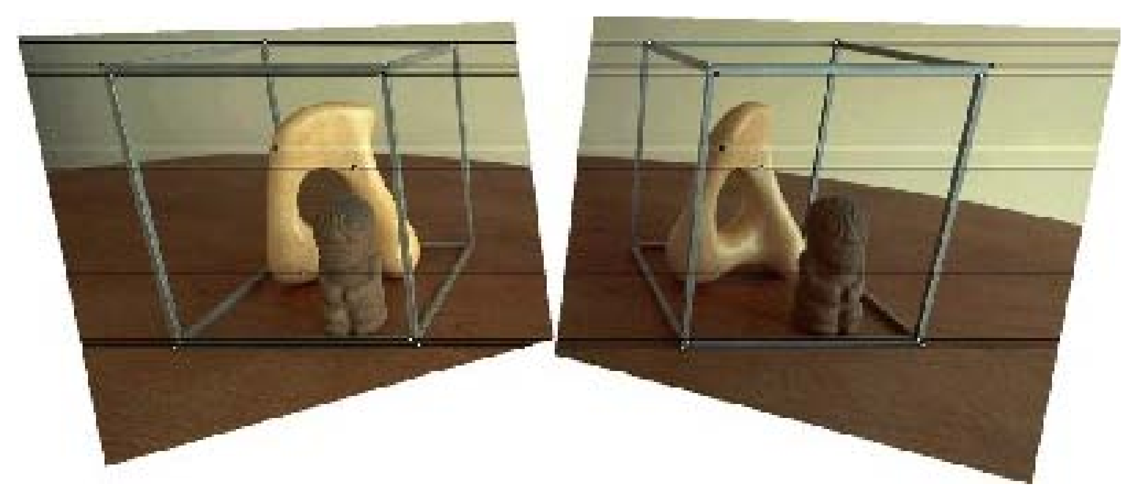
What happens to epipolar lines when we double the distance between two camera views?



Stereo Pair



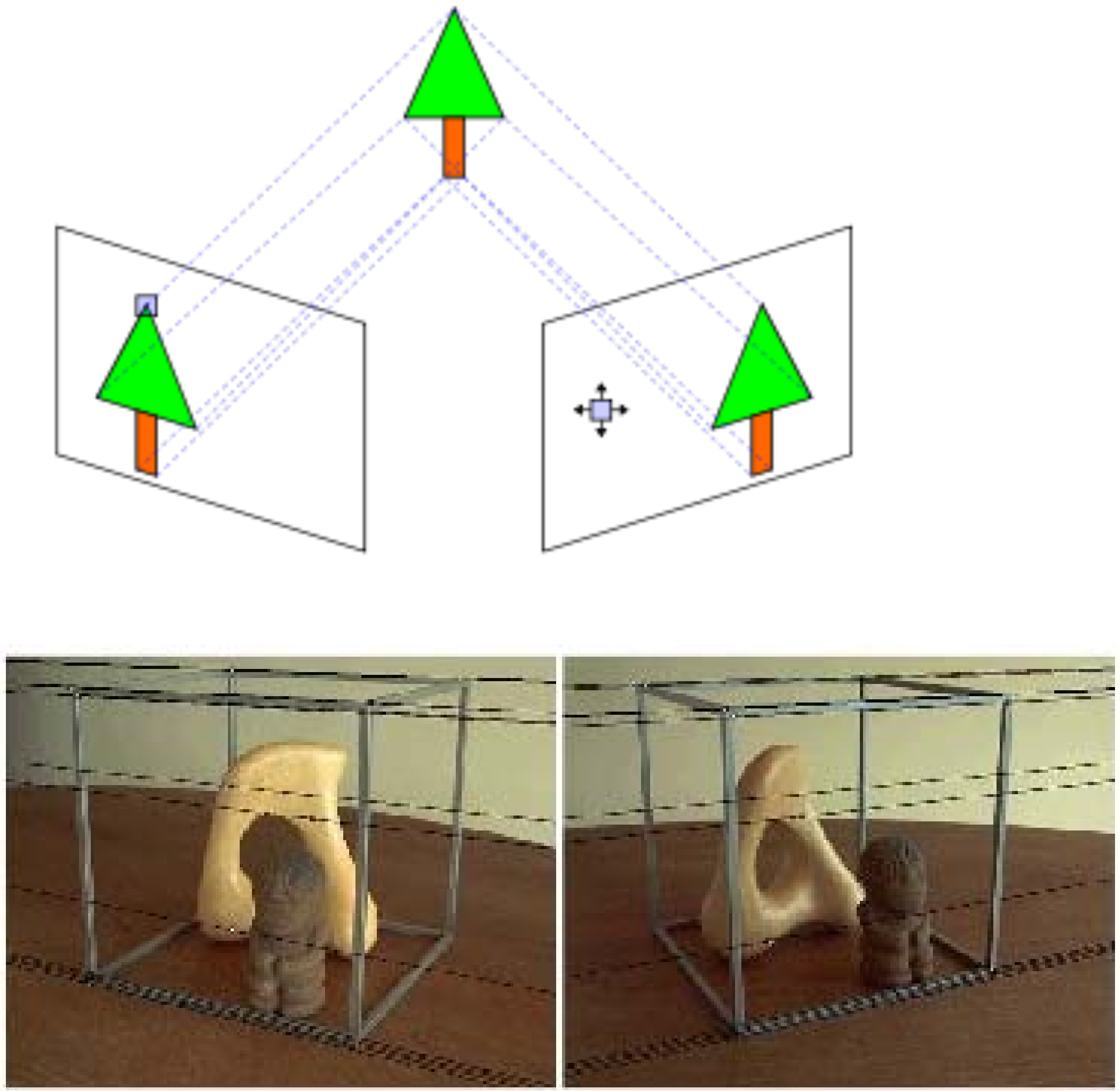
Stereo Pair



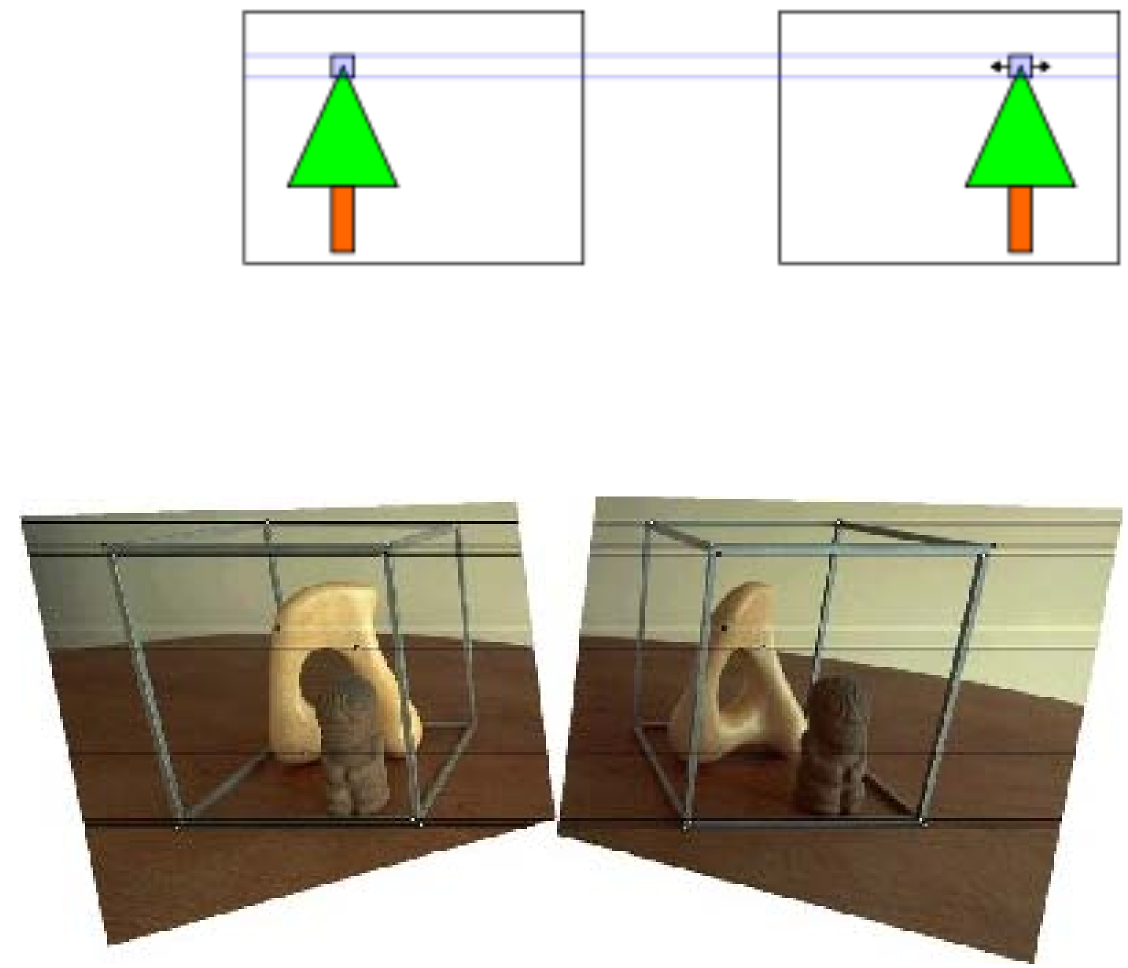
Rectified Stereo Pair

Rotate image plane about fixed camera centers

Aside: what kind of transformation is this?



Stereo Pair



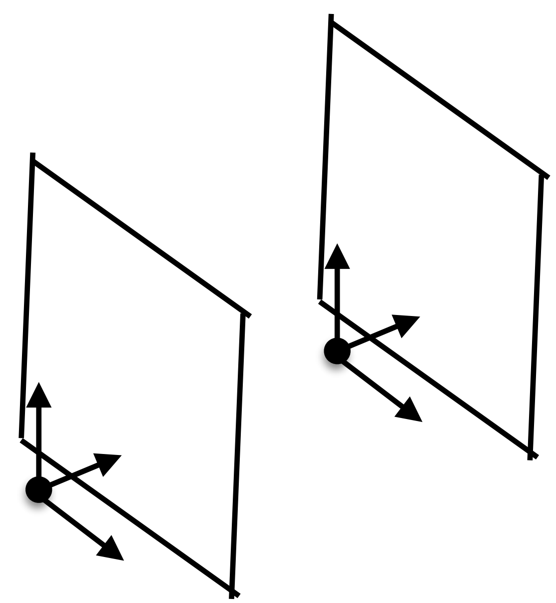
Rectified Stereo Pair

Question: do the epipolar lines depend on scene structure, cameras, or both?

Epipolar geometry is purely determined by camera extrinsics and camera instrinics

Special case (II)

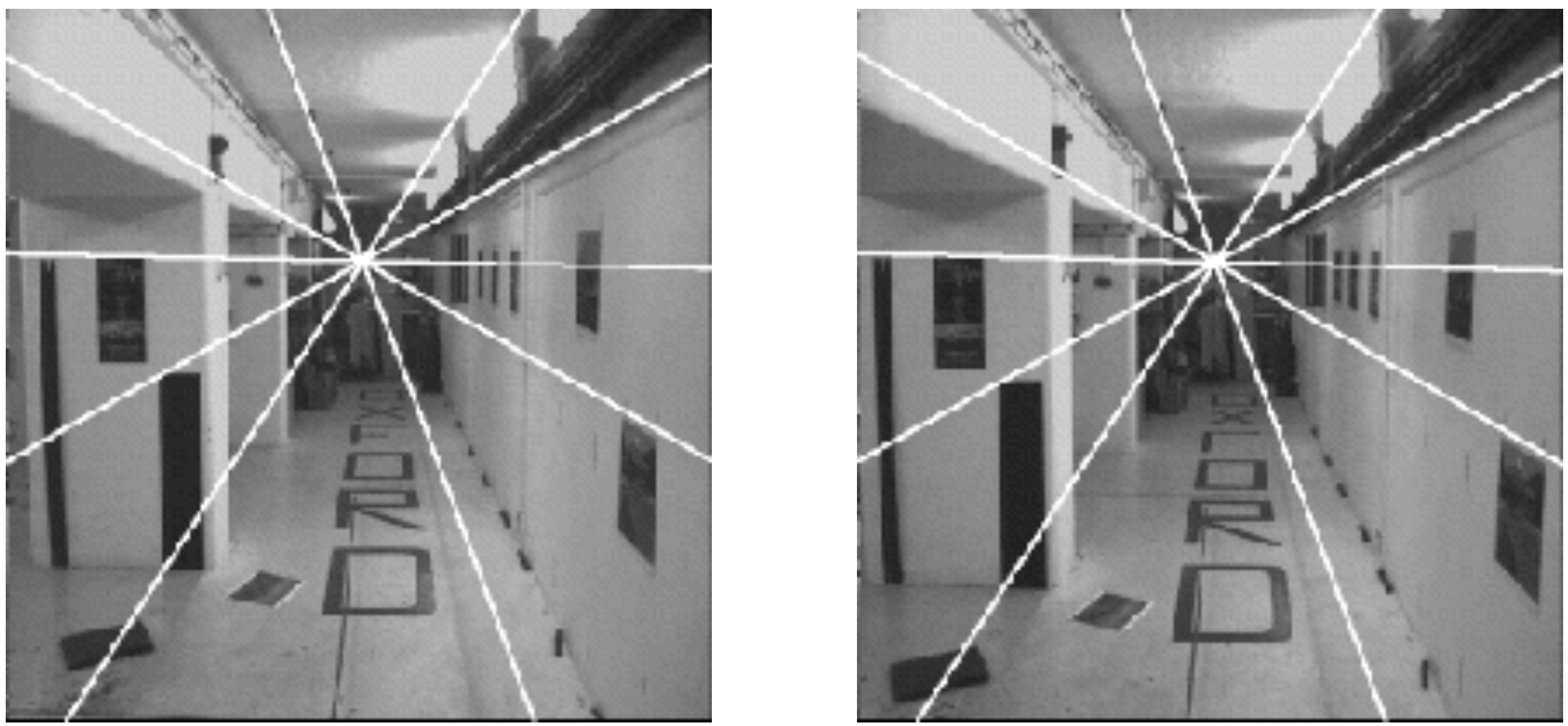
Forward camera motion



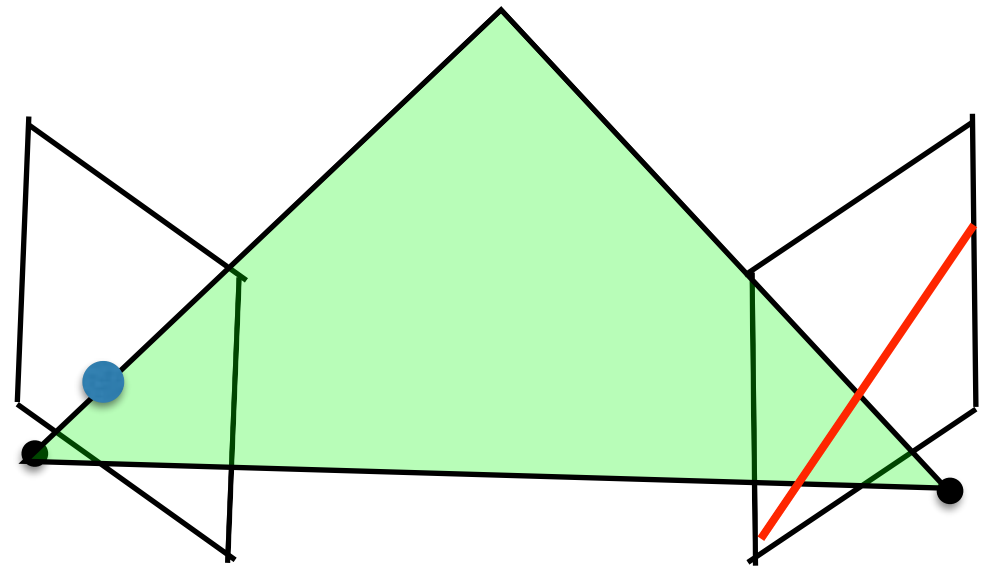
What would epipolar lines look like?

Special case (II)

Forward camera motion



Mathematical formulation



Goal: given point in left image, we want to compute the equation of the line on the right image

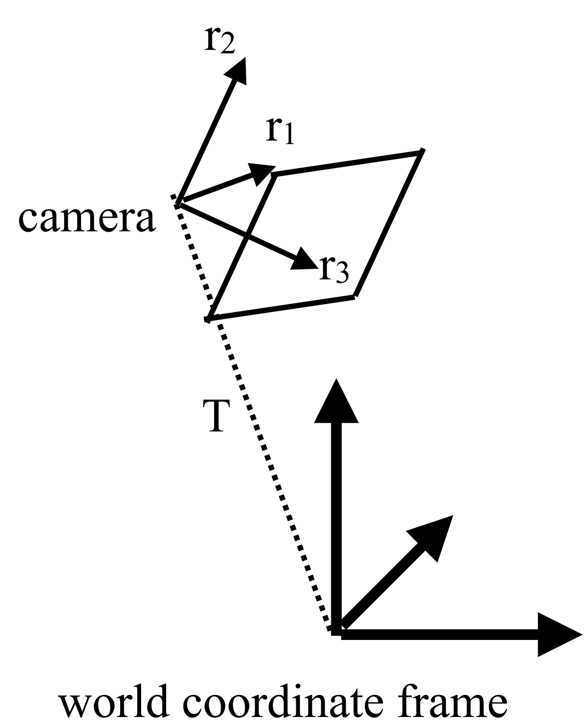
Recall: camera projection

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Camera instrinsic matrix K
(can include skew & non-square pixel size)

Camera extrinsics
(rotation and translation)

3D point in
world coordinates



Recall notation

[Using Matlab's rows x columns]

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$= K_{3\times 3} \begin{bmatrix} R_{3\times 3} & T_{3\times 1} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$= M_{3\times 4} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

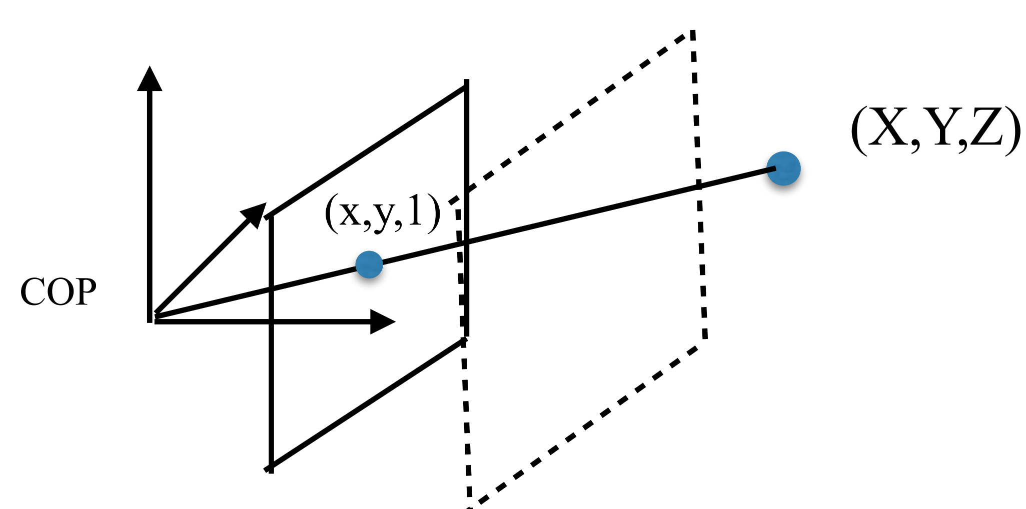
Normalized coordinates

Assume camera intrinsics K = Identity (focal length of 1)

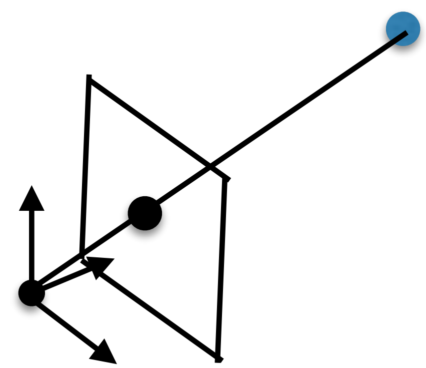
$$\begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

If K is known, compute warped image

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K^{-1} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



Normalized coordinates

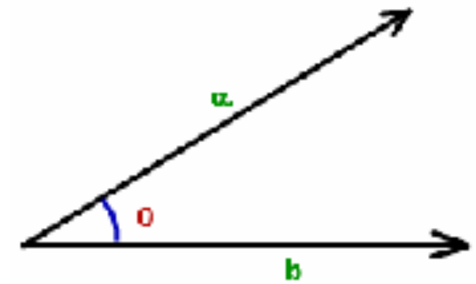


$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

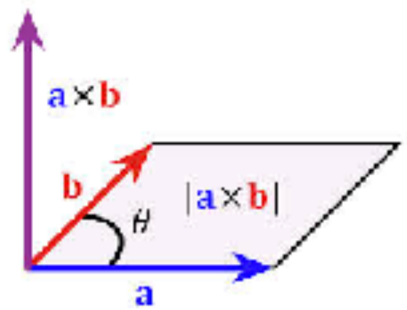
$$\lambda \mathbf{x} = \mathbf{X}$$

Recall

Dot product: $\mathbf{a} \cdot \mathbf{b} = ||\mathbf{a}|| \ \ ||\mathbf{b}|| cos\theta$



Cross product: $\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \sin\theta \ \mathbf{n}$

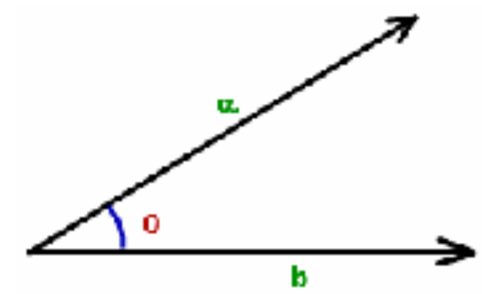


Cross product matrix: $\mathbf{a} \times \mathbf{b} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \equiv \hat{\mathbf{a}}\mathbf{b}$

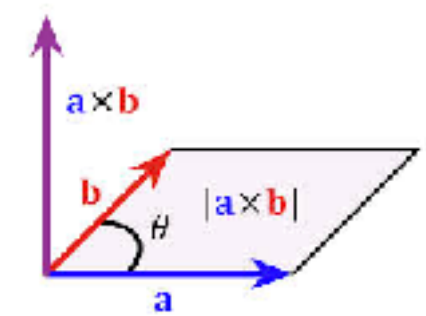
Important property (skew symmetric): $\hat{\mathbf{a}}^T = -\hat{\mathbf{a}}$

Recall

Dot product: $\mathbf{a} \cdot \mathbf{b} = ||\mathbf{a}|| \;\; ||\mathbf{b}||cos\theta$

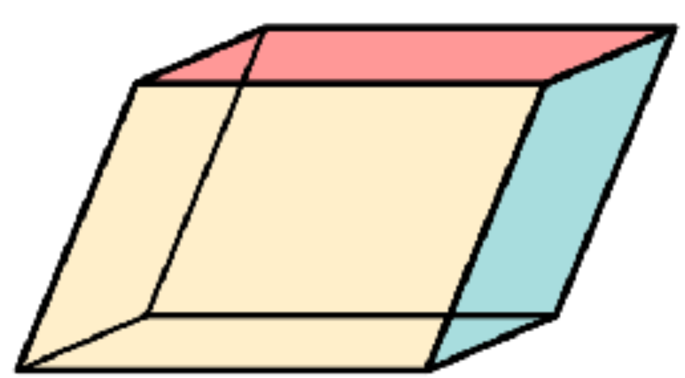


Cross product: $\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \sin\theta \; \mathbf{n}$

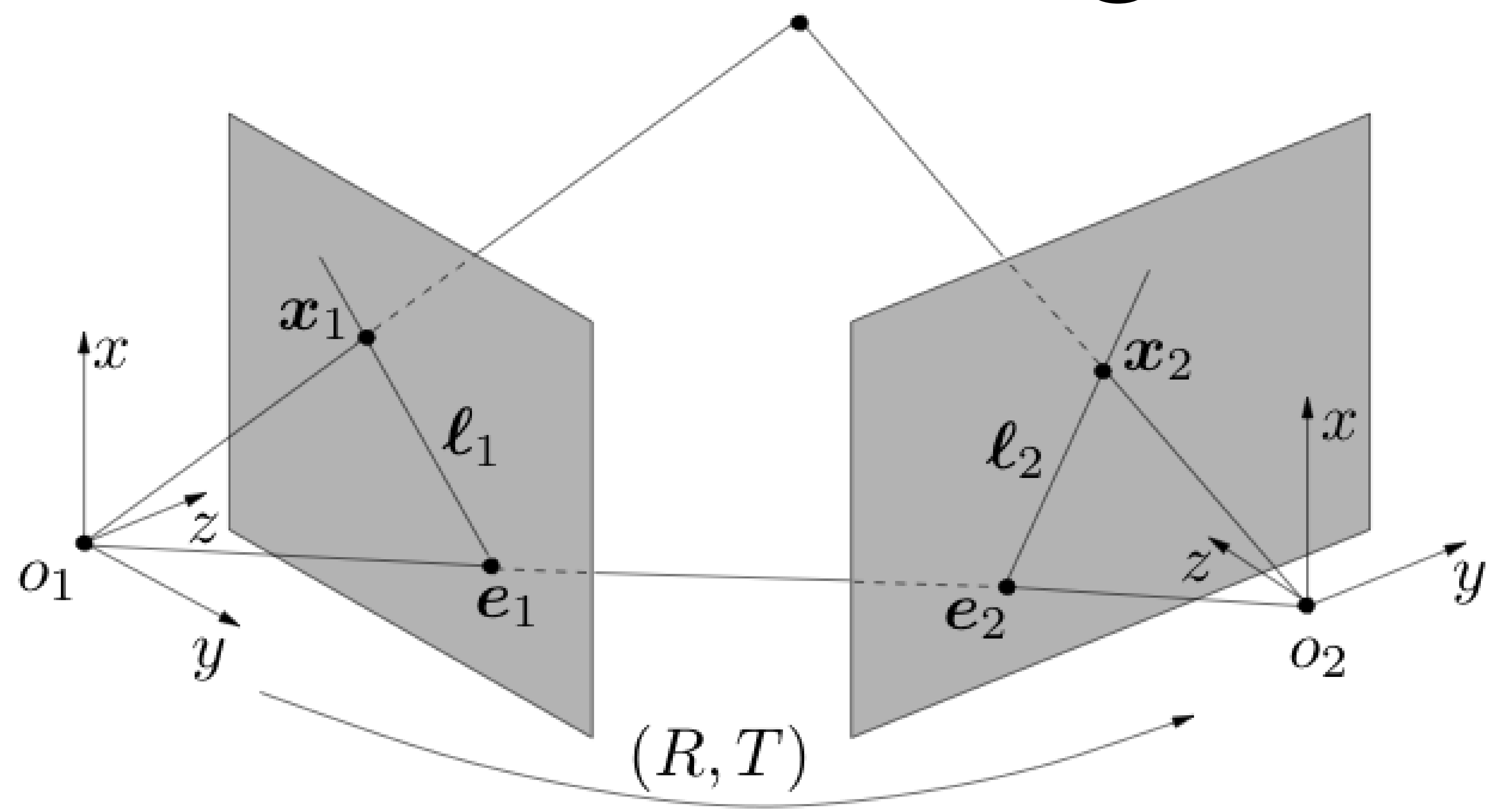


Cross product matrix: $\mathbf{a} \times \mathbf{b} = \hat{\mathbf{a}}\mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$

$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$ = volume of parallelpiped
= 0 for coplanar vectors



Calibrated 2-view geometry

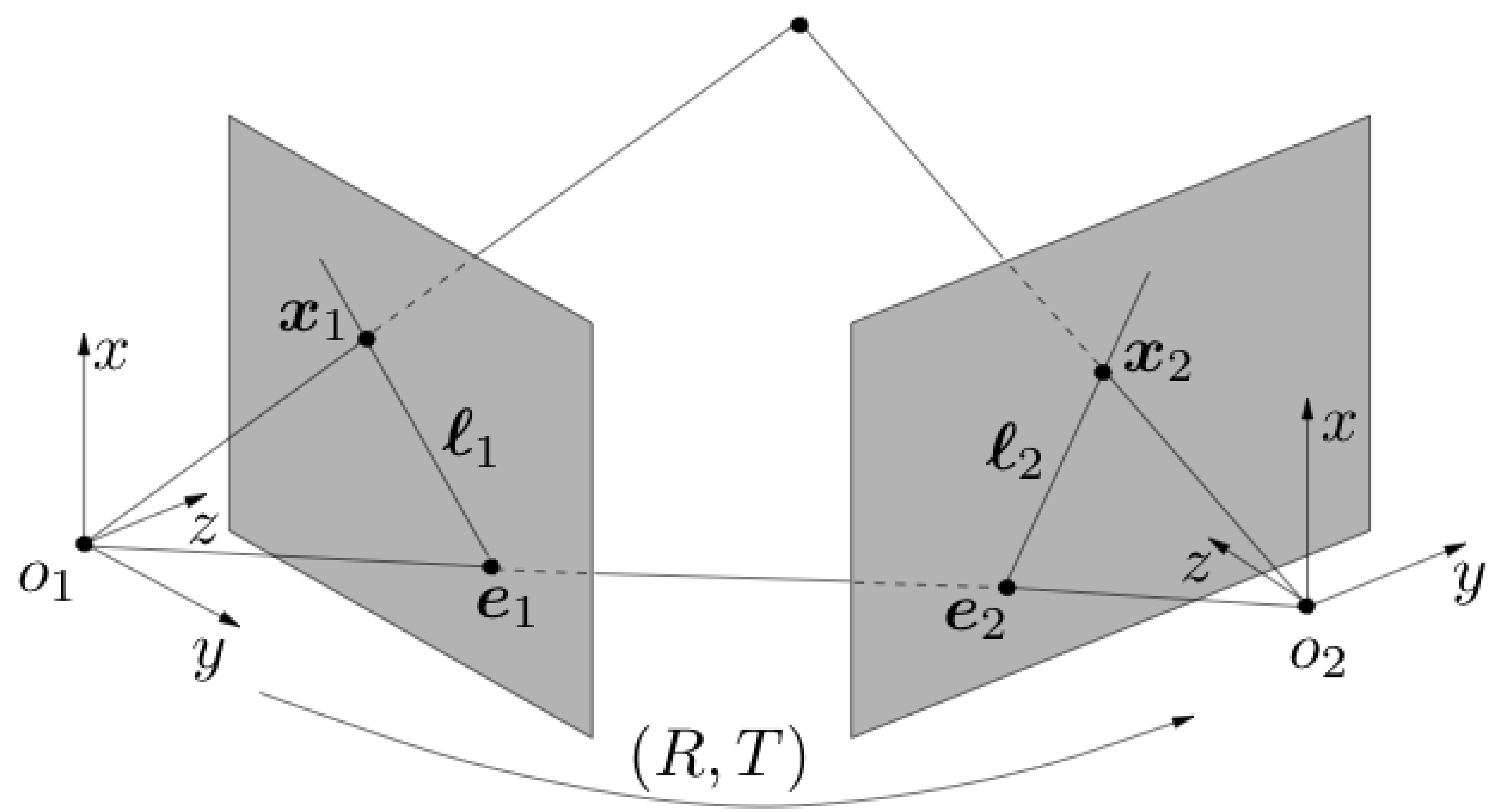


$$\boxed{\boldsymbol{X}_2 = R\boldsymbol{X}_1 + \boldsymbol{T}}$$

$\mathbf{X}_1$: postion of 3D point in camera 1's coordinate system

$\mathbf{X}_2$: postion of 3D point in camera 2's coordinate system

Calibrated 2-view geometry



$$\boxed{\boldsymbol{X}_2 = R\boldsymbol{X}_1 + \boldsymbol{T}}$$

$$\boldsymbol{X}_1 = \lambda_1 \boldsymbol{x}_1, \quad \boldsymbol{X}_2 = \lambda_2 \boldsymbol{x}_2$$

Epipolar geometry

$$\boxed{\boldsymbol{X}_2 = R\boldsymbol{X}_1 + \boldsymbol{T}}$$

$$\boldsymbol{X}_1 = \lambda_1 \boldsymbol{x}_1, \quad \boldsymbol{X}_2 = \lambda_2 \boldsymbol{x}_2$$

$$\lambda_2 \boldsymbol{x}_2 = R\lambda_1 \boldsymbol{x}_1 + \boldsymbol{T}$$

(On board)

1. Take (left) cross product of both sides with T
2. Take (left) dot product of both sides with $\mathbf{x}_2$

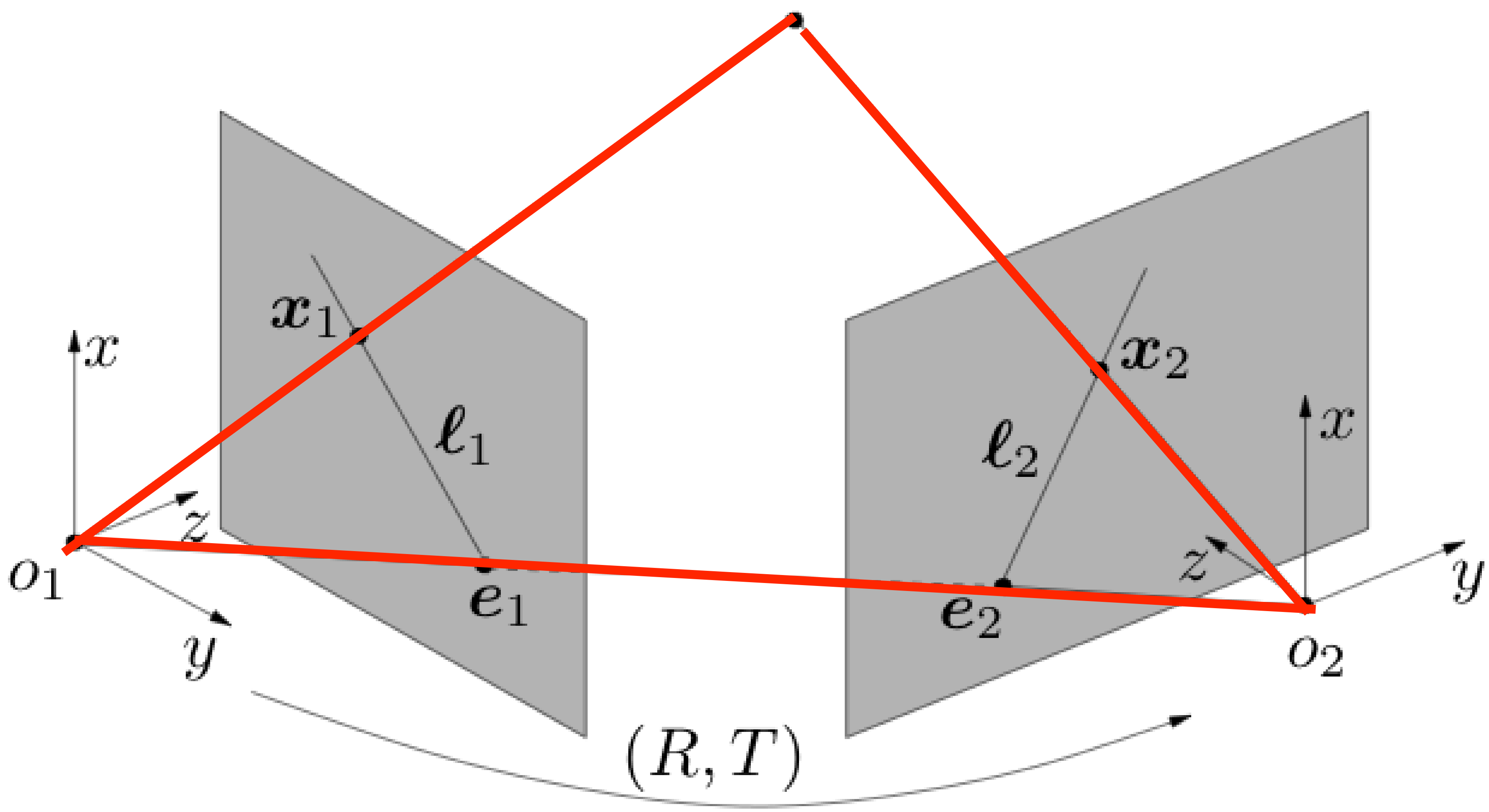
Epipolar geometry

$$\lambda_2 \widehat{T} \boldsymbol{x}_2 = \widehat{T} R \lambda_1 \boldsymbol{x}_1 + \underbrace{\widehat{T} T}_{=\mathbf{0}}$$

$$\lambda_2 \underbrace{\boldsymbol{x}_2^\top \widehat{T} \boldsymbol{x}_2}_{=0} = \boldsymbol{x}_2^\top \widehat{T} R \lambda_1 \boldsymbol{x}_1$$

$$\boldsymbol{x}_2^\top \widehat{T} R \boldsymbol{x}_1 = 0$$

Epipolar geometry



$$\boldsymbol{x}_2^\top \widehat{T} R \boldsymbol{x}_1 = 0$$

$$\mathbf{x}_2 \cdot (\mathbf{T} \times R\mathbf{x}_1) = 0$$

Simply the coplanar constraint applied to 3 vectors from camera 2's coordinate system

Epipolar geometry

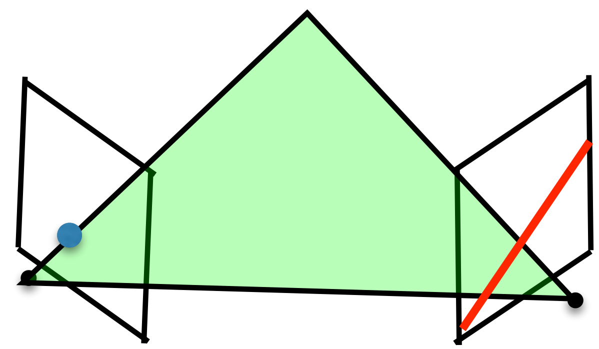
$$\boldsymbol{x}_2^\top \widehat{T} R \boldsymbol{x}_1 = 0$$

$$\boxed{\boldsymbol{x}_2^\top E \boldsymbol{x}_1 = 0}$$

E is known as the *essential* matrix

Essential matrix

$$\boxed{\boldsymbol{x}_2^\top E \boldsymbol{x}_1 = 0}$$



$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = E \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

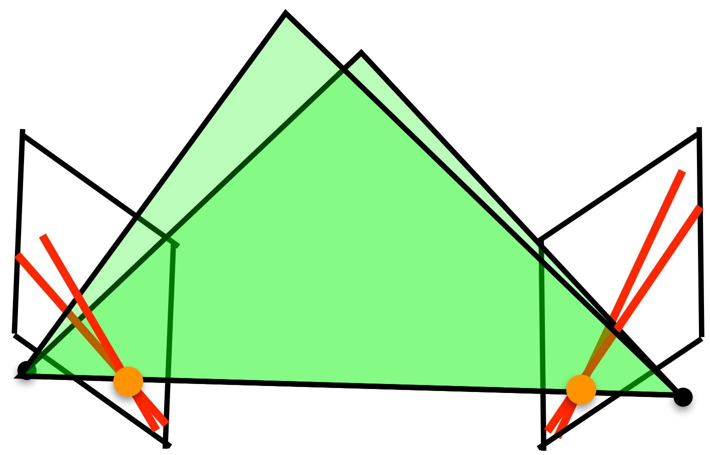
$$ax_2 + by_2 + c = 0$$

Maps a (x1,y1) point from left image to line in right image (and vice versa)

But how is this different from a Homography (also a 3X3 matrix)?

Epipoles

$$\boxed{\boldsymbol{x}_2^\top E \boldsymbol{x}_1 = 0}$$



We'll write epipolar lines as 3-vectors: $\mathbf{l_2} = E\mathbf{x_1}$

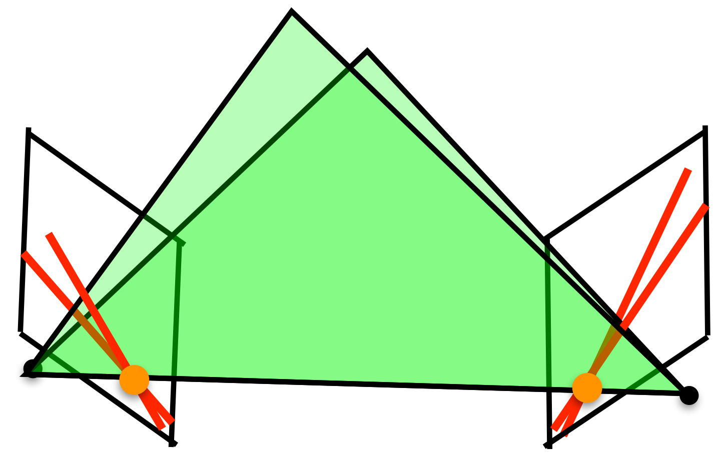
Note that all epipolar lines in an image plane intersect at the epipole. Equivalently, the epipole has a distance of zero from every epipolar line: $\boldsymbol{e}_2^\top \boldsymbol{l}_2 = 0, \forall \boldsymbol{x}_1$, and similarly $\boldsymbol{e}_1^\top \boldsymbol{l}_1 = 0, \forall \boldsymbol{x}_2$.

For this to hold true, $\boldsymbol{e}_2^\top E$ and $E\boldsymbol{e}_1$ must be zero vectors, i.e.,

$$\boldsymbol{e}_2^\top E = \mathbf{0}, \qquad E\boldsymbol{e}_1 = \mathbf{0}$$

Thus $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are vectors in the right and left null space of E, respectively, i.e., the left and right singular vectors of E with singular value 0.

Uncalibrated case



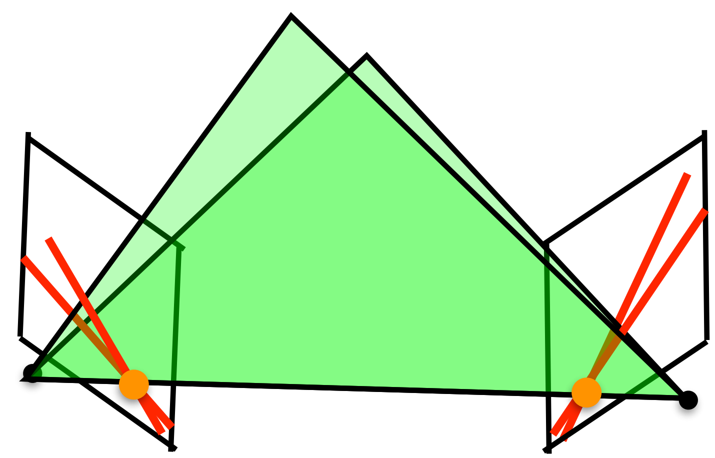
$$\lambda_1 \mathbf{x_1} = K_1 \mathbf{X}_1$$
$$\lambda_2 \mathbf{x_2} = K_2 \mathbf{X}_2$$

$$\boxed{\boldsymbol{X}_2 = R\boldsymbol{X}_1 + \boldsymbol{T}}$$
$$\boldsymbol{x}_2^\top \widehat{T} R \boldsymbol{x}_1 = 0$$

$$\mathbf{x_2}^T {K_2}^{-T} \hat{T} R K_1^{-1} \mathbf{x_1} = 0$$
$$\mathbf{x_2}^T F \mathbf{x_1} = 0$$

Fundamental matrix

(Faugeras and Luong, 1992)



$$E = \hat{T}R$$

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

Next lecture

- 2-view geometry
- essential matrix, fundamental matrix
- properties
- estimation

Next lecture

- Properties of F,E
- Estimating F,E from correspondences
- Recovering T,R,K from estimated F,E