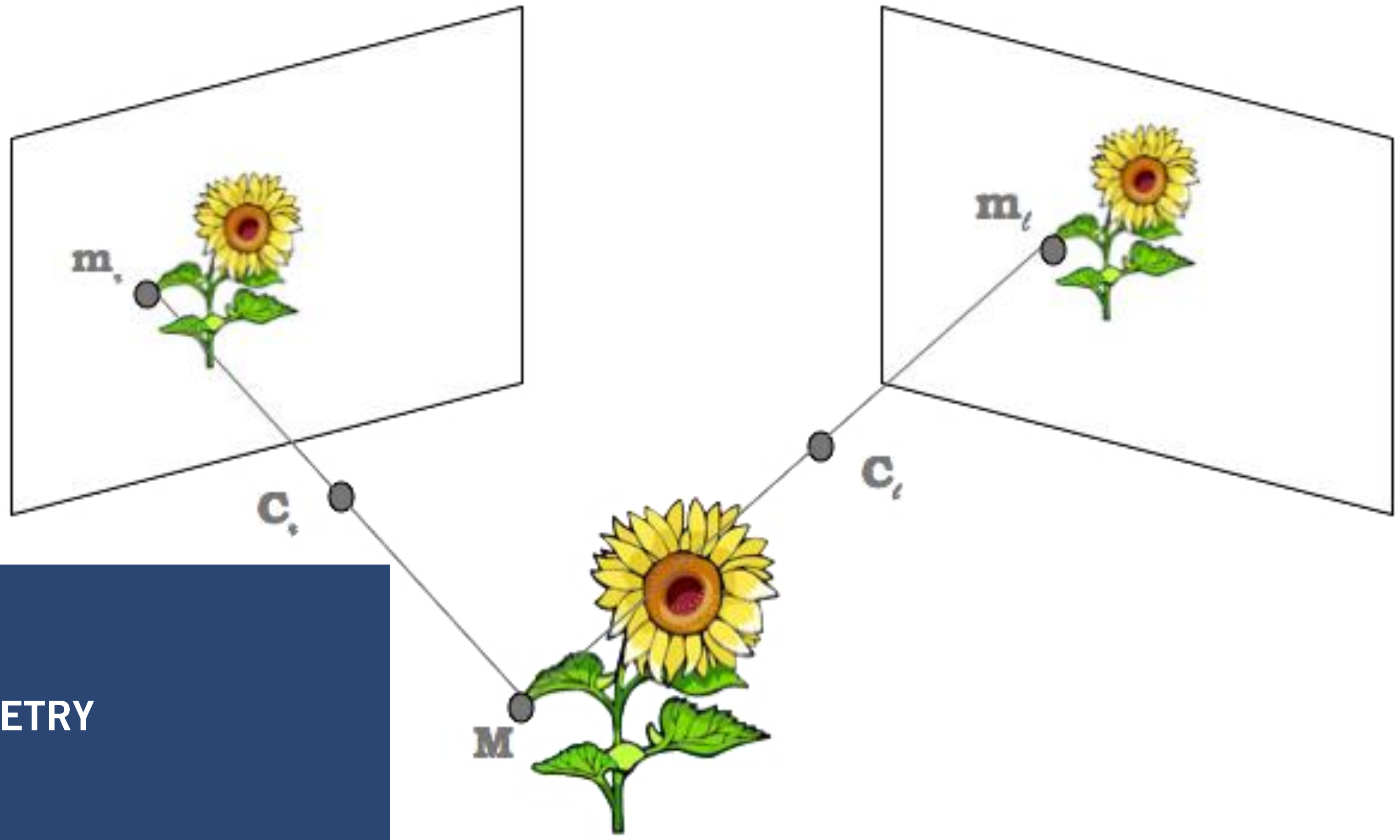
COMPUTER VISION
LECTURE 16-17 – TWO-VIEW GEOMETRY



Prof. Dr. Francesco Maurelli
2019-10-23
2019-10-29

Courtesy of Ioannis Gkioulekas, CMU

Imaging

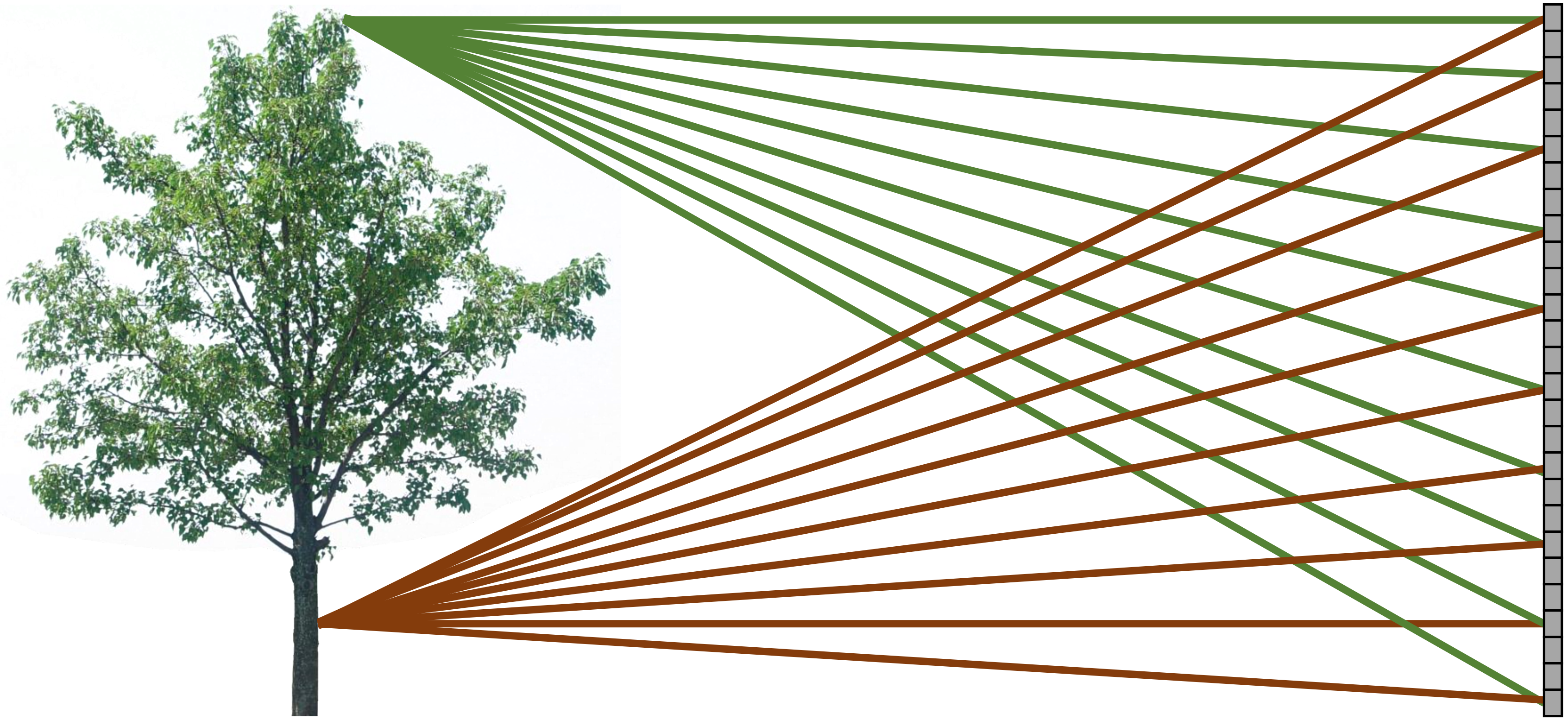
real-world
object

digital sensor
(CCD or CMOS)

What would an image taken like this look like?

Bare-sensor imaging

real-world
object



digital sensor
(CCD or CMOS)

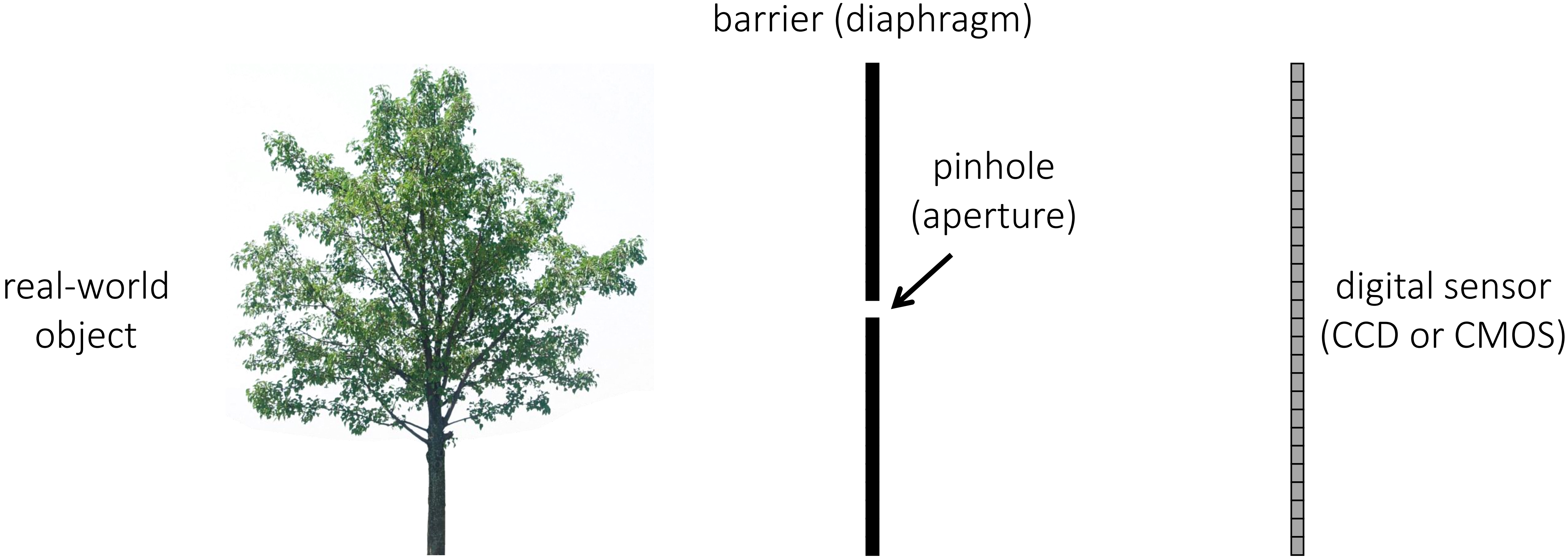
All scene points contribute to all sensor pixels

What does the
image on the
sensor look like?

Bare-sensor imaging

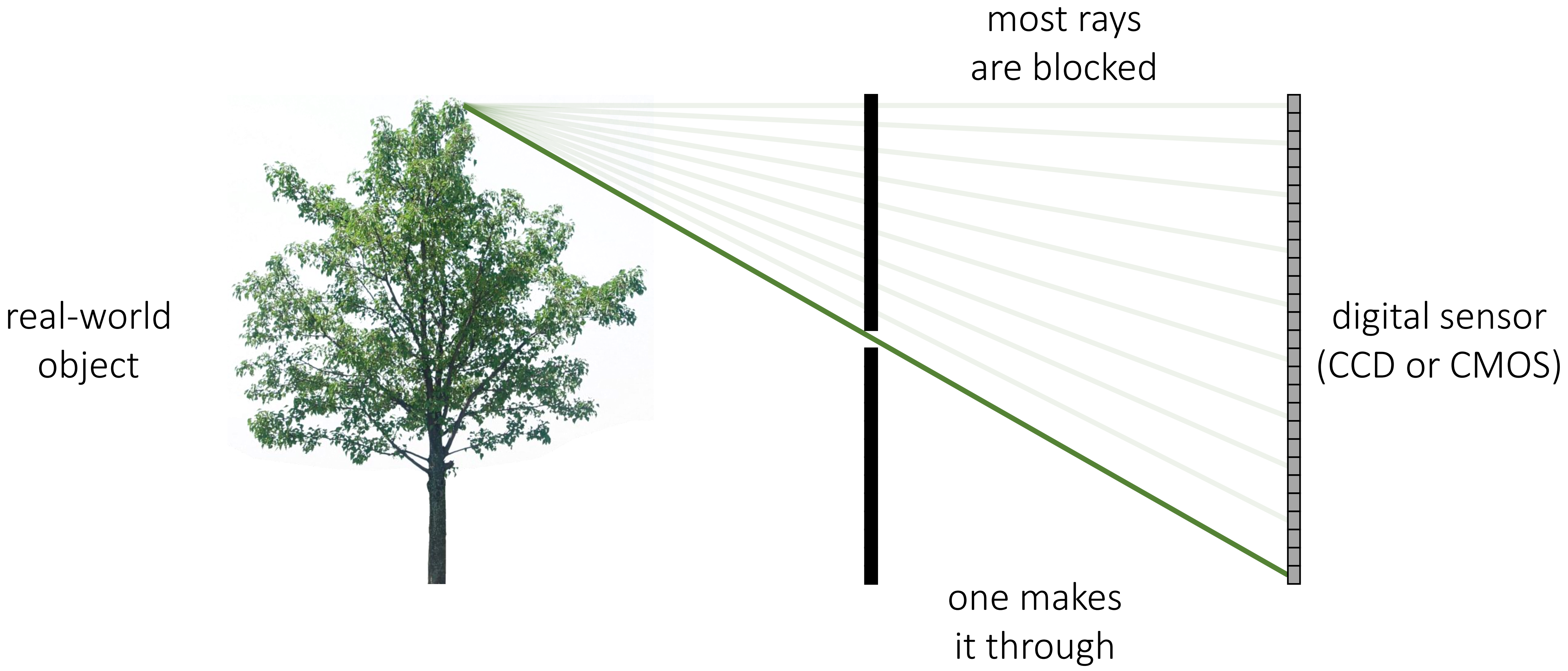
All scene points contribute to all sensor pixels

Let’s add something to this scene

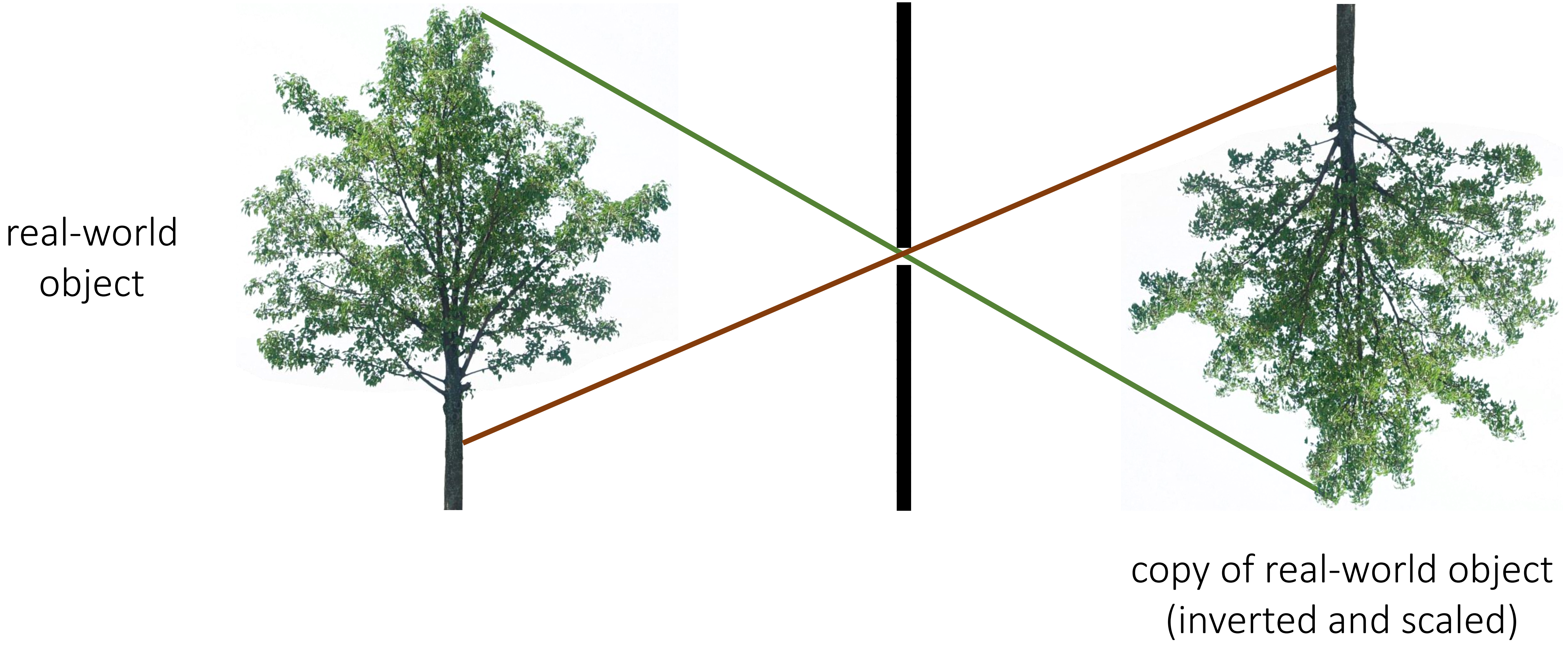


What would an image taken like this look like?

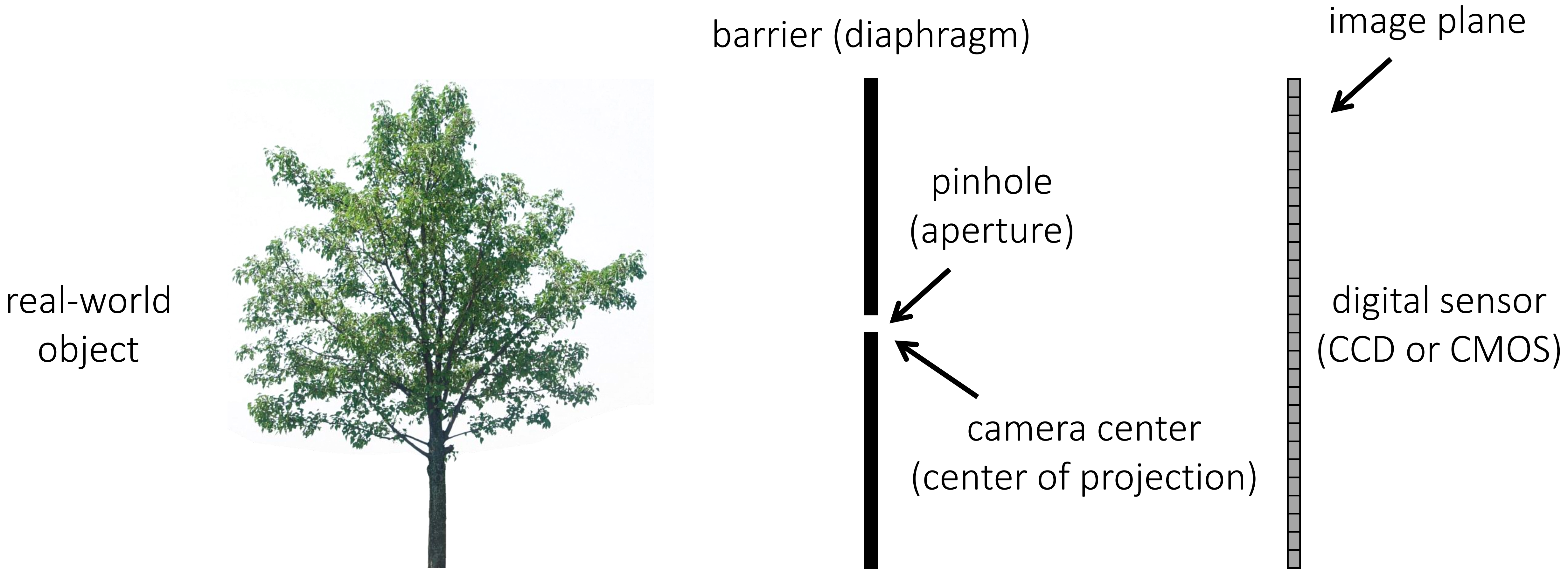
Pinhole imaging



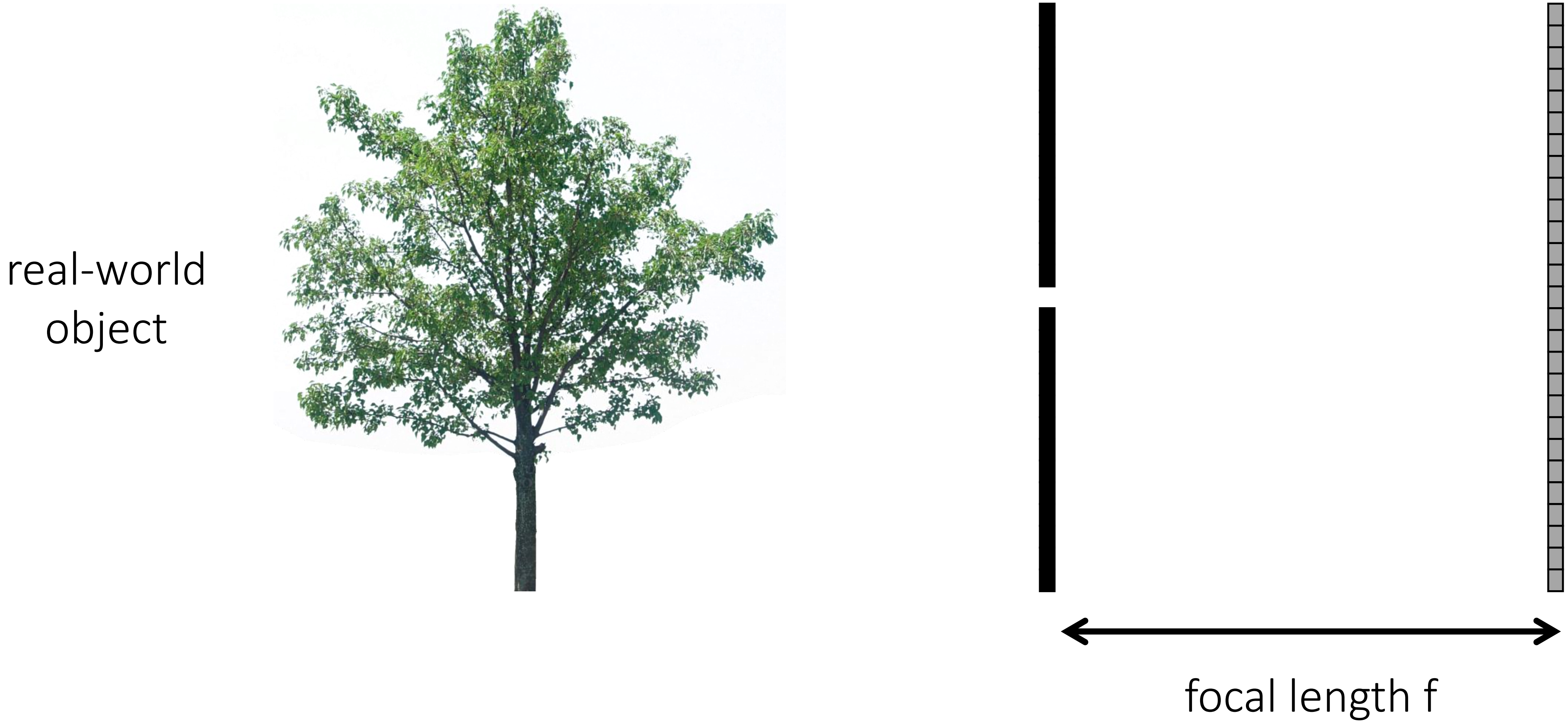
Pinhole imaging



Pinhole camera terms

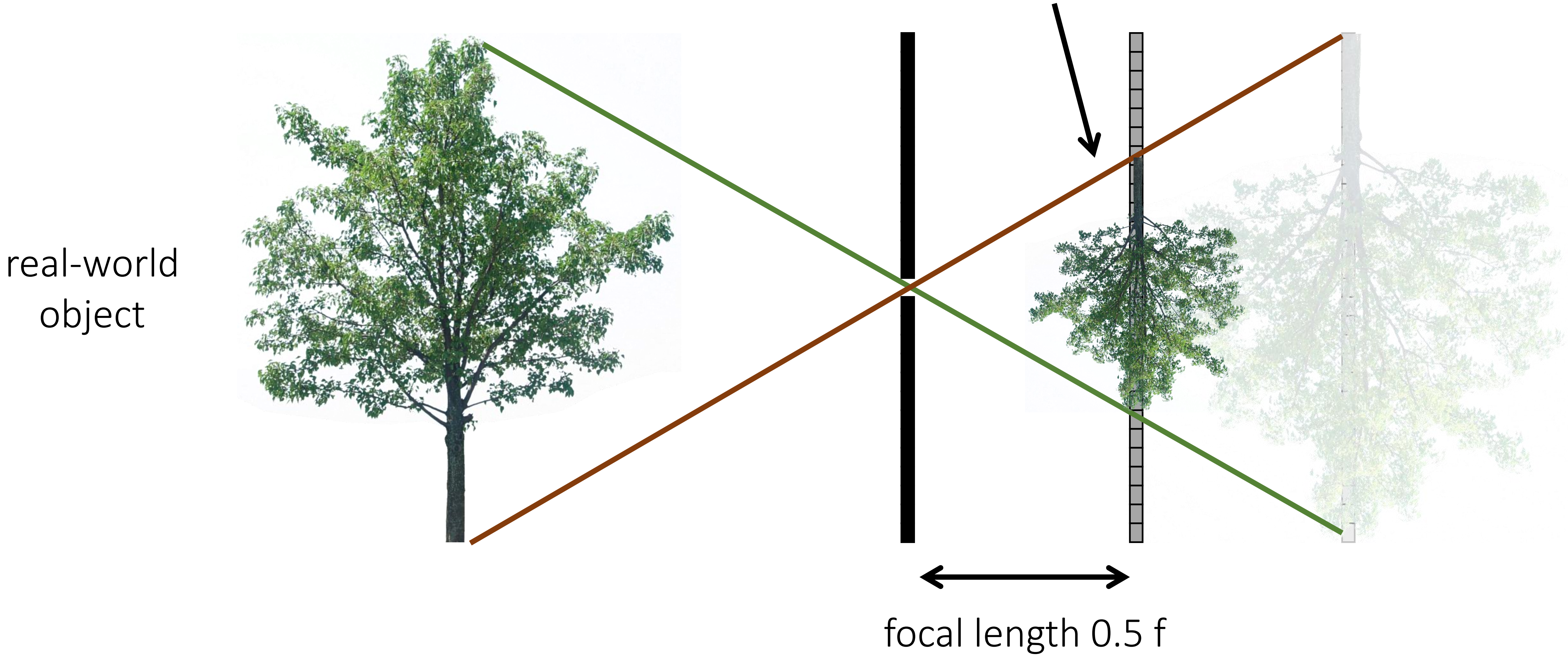


Focal length



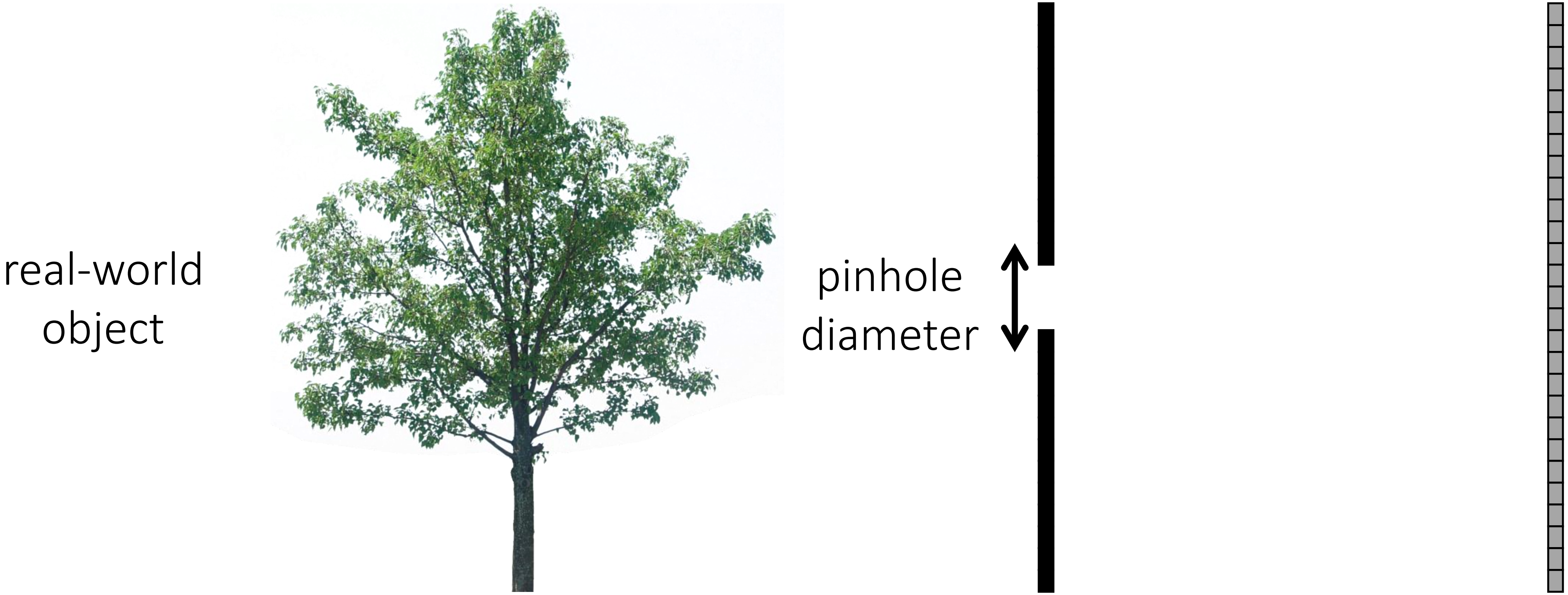
Focal length

What happens as we change the focal length?



Pinhole size

What happens as we change the pinhole diameter?

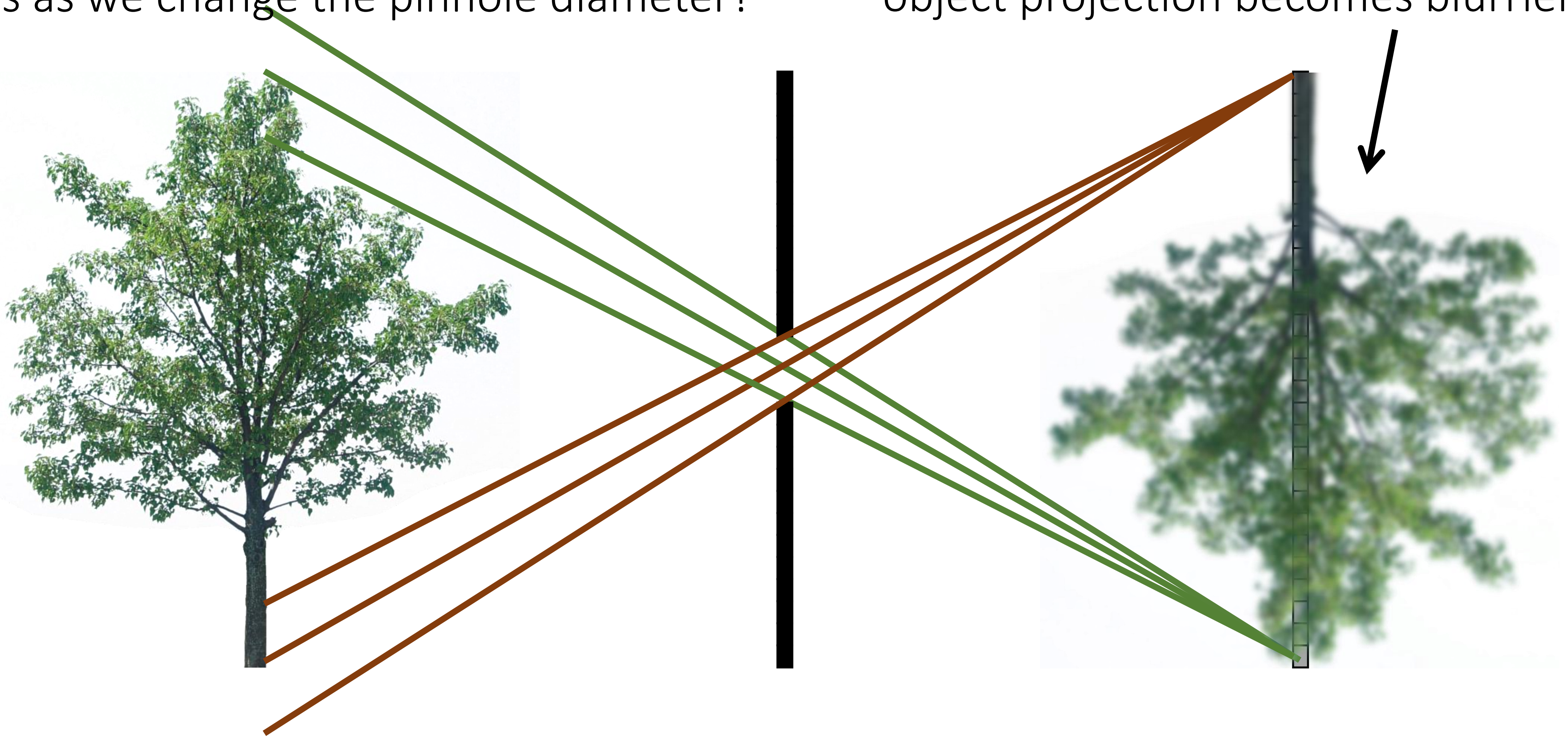


Pinhole size

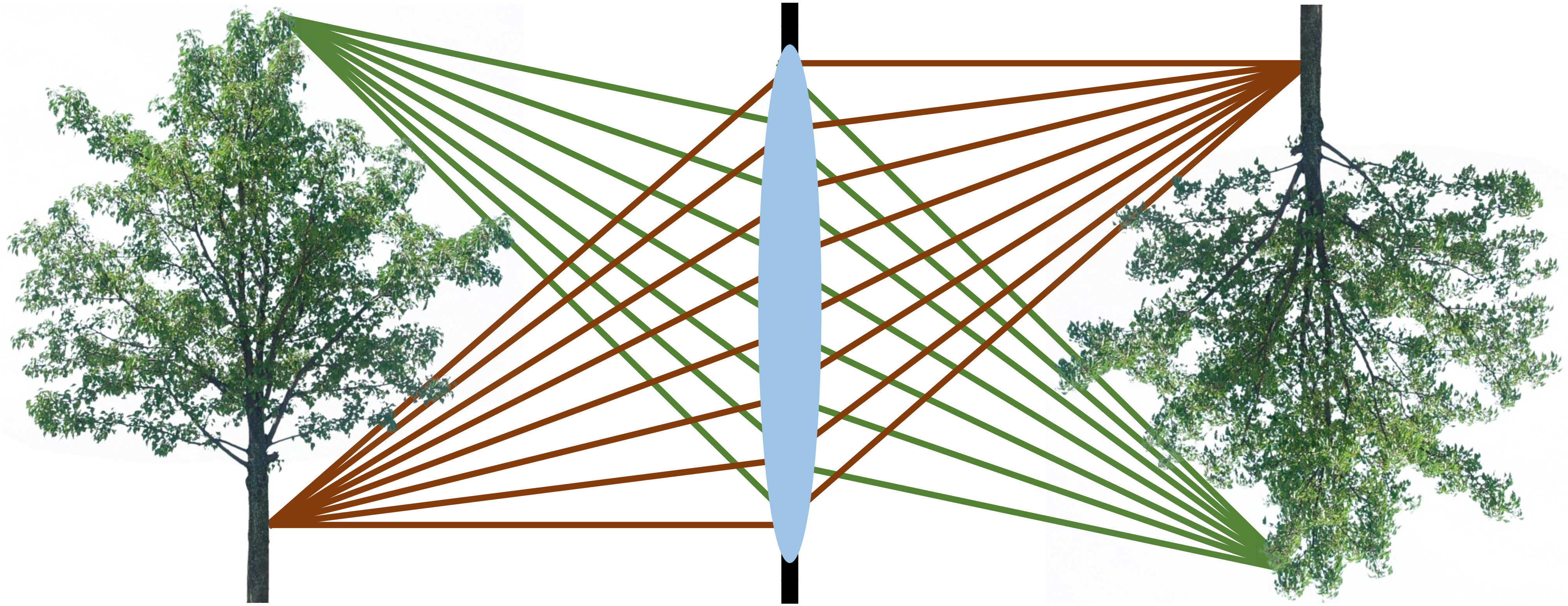
What happens as we change the pinhole diameter?

object projection becomes blurrier

real-world object



In practice



Lenses map “bundles” of rays from points on the scene to the sensor.

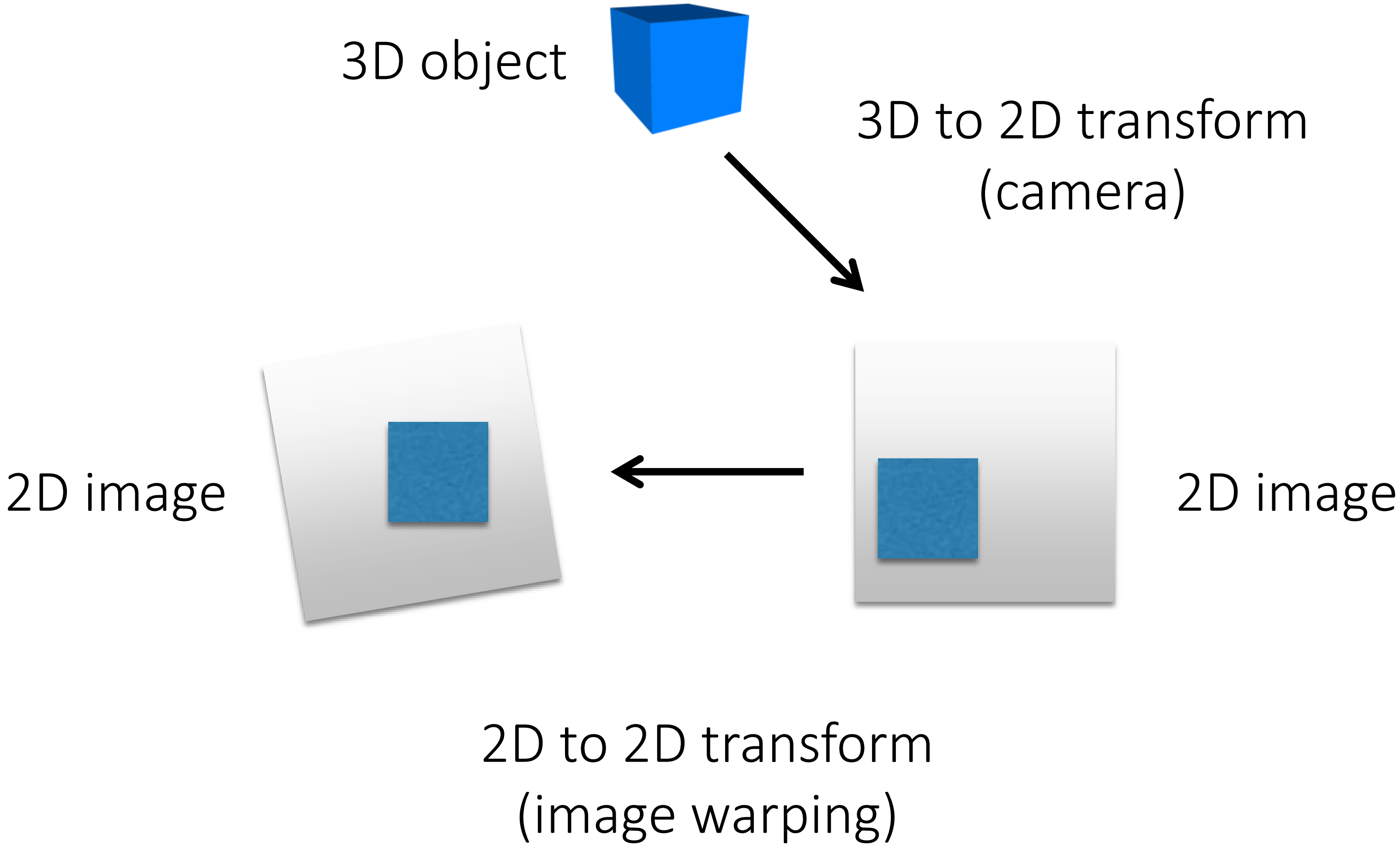
The camera as a coordinate transformation

A camera is a mapping from:

the 3D world

to:

a 2D image



The camera as a coordinate transformation

A camera is a mapping from:

the 3D world

to:

a 2D image

homogeneous coordinates

$$\boldsymbol{x} = \mathbf{P}\mathbf{X}$$

2D image point

camera matrix

3D world point

What are the dimensions of each variable?

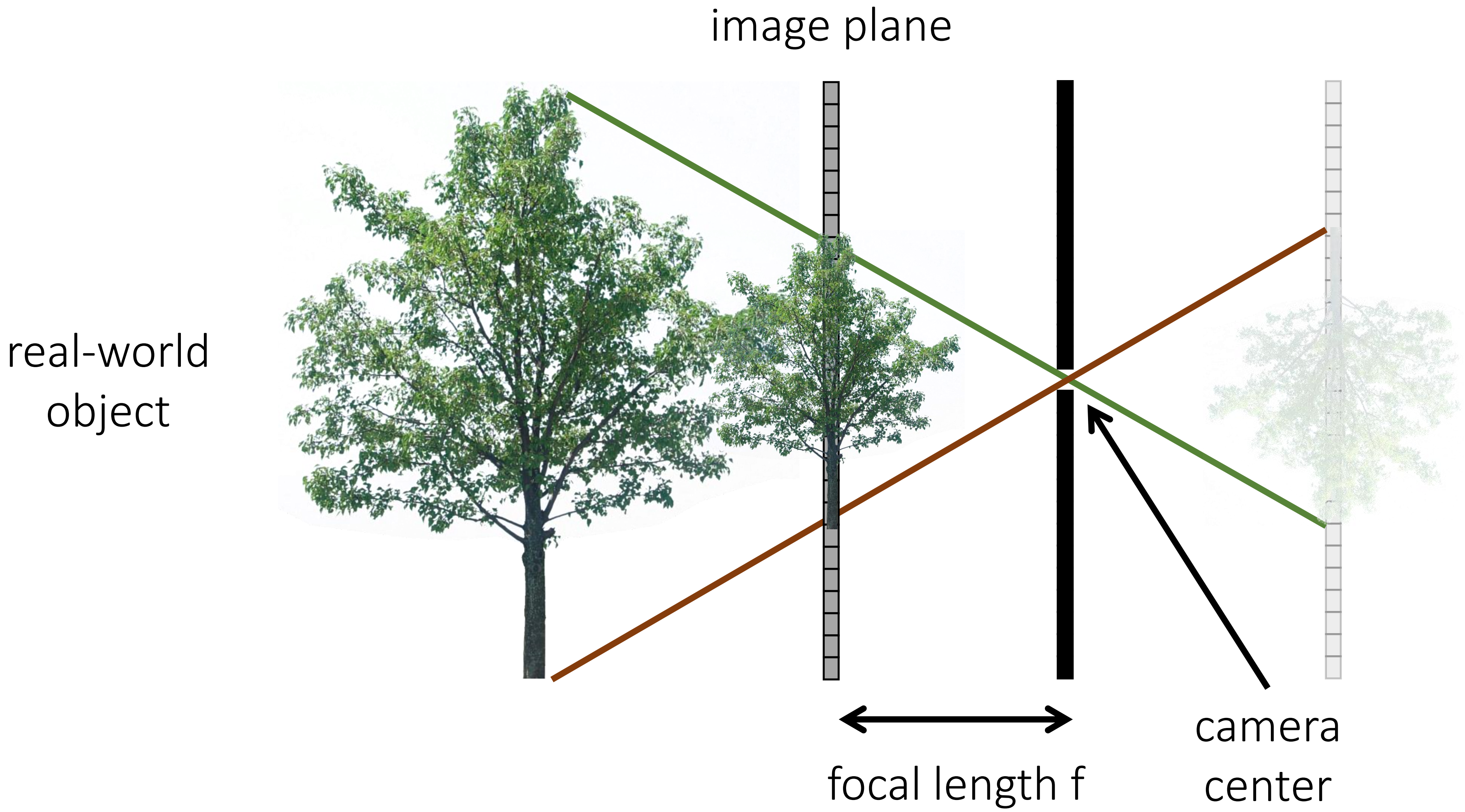
The camera as a coordinate transformation

$$\boldsymbol{x} = \mathbf{PX}$$

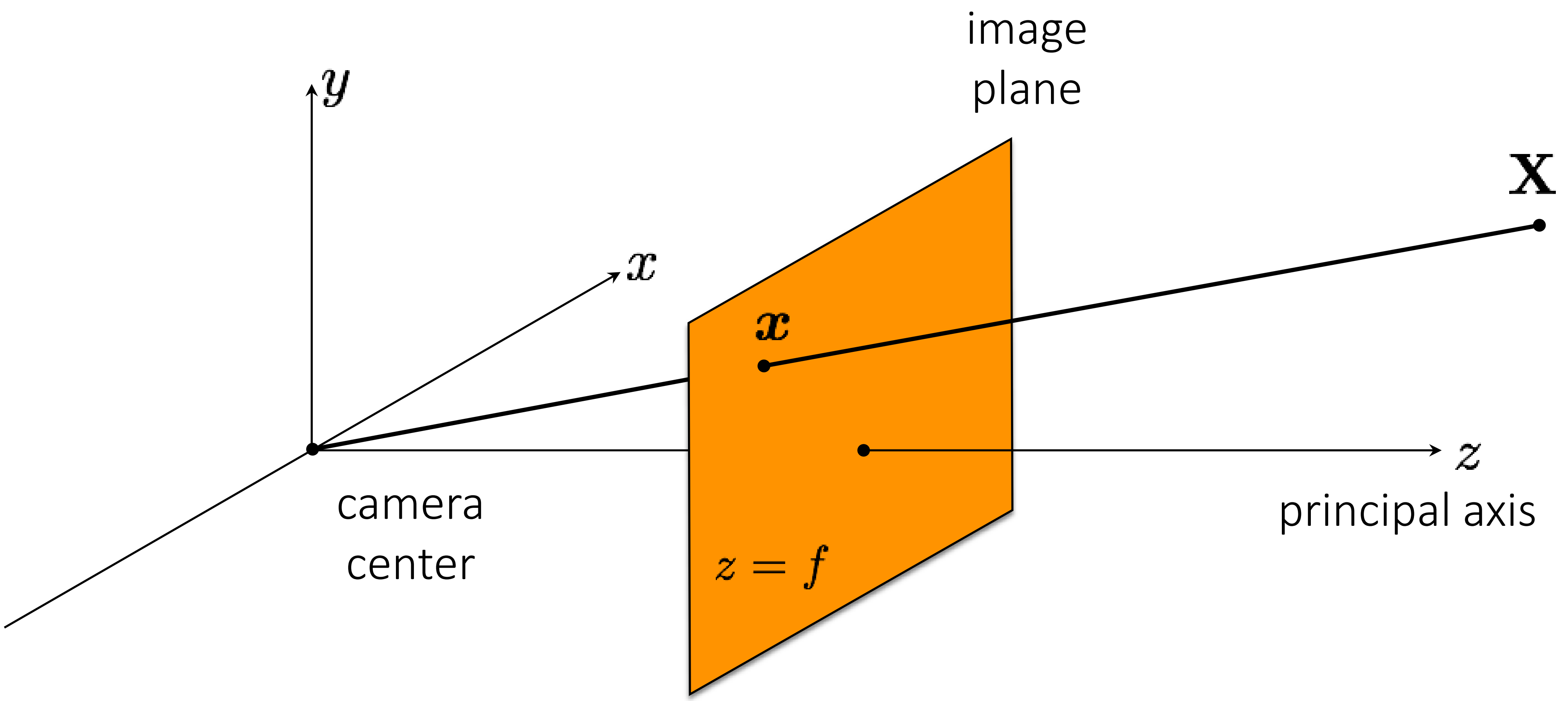
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

homogeneous image coordinates 3 x 1	camera matrix 3 x 4	homogeneous world coordinates 4 x 1

The (rearranged) pinhole camera

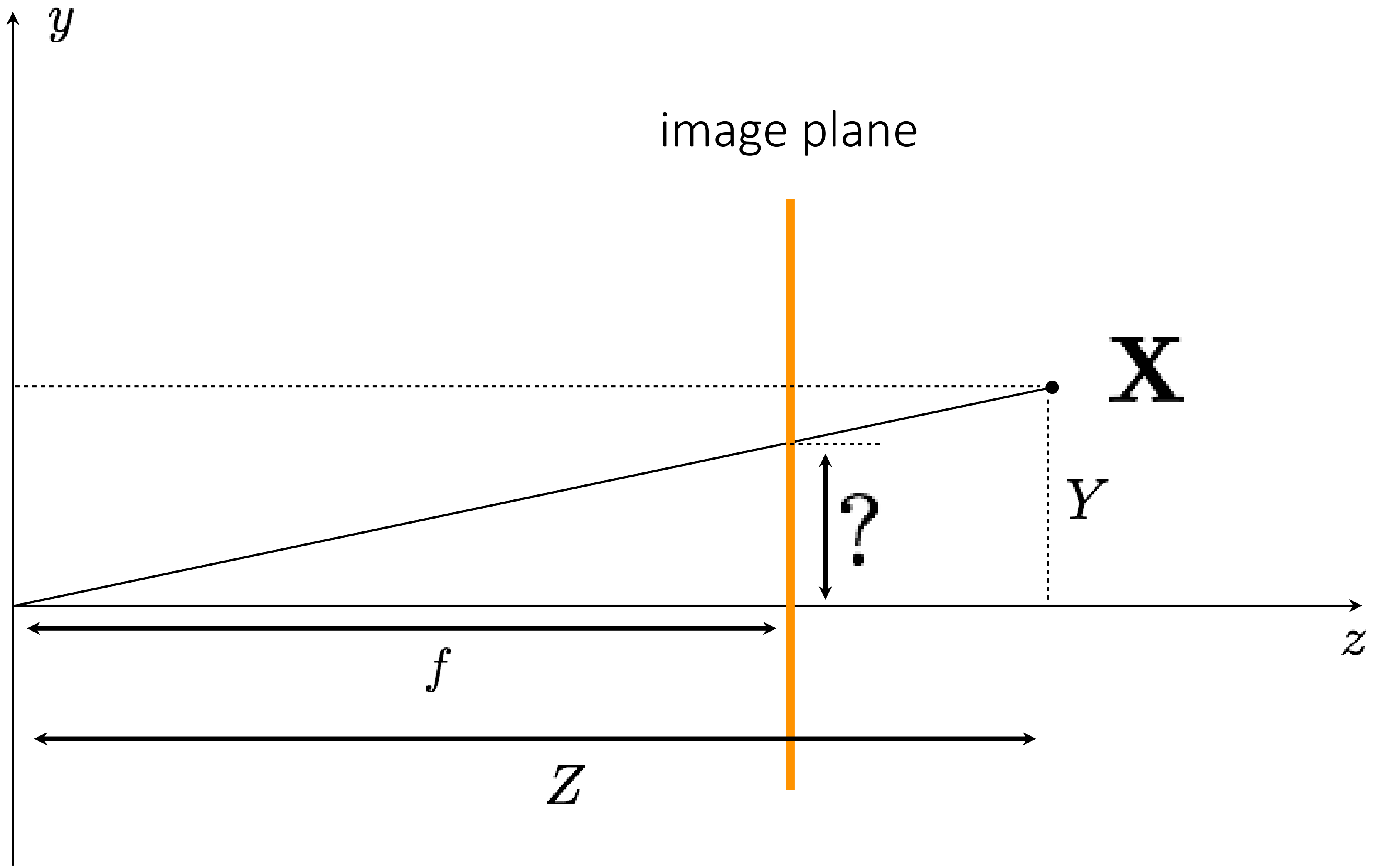


The (rearranged) pinhole camera



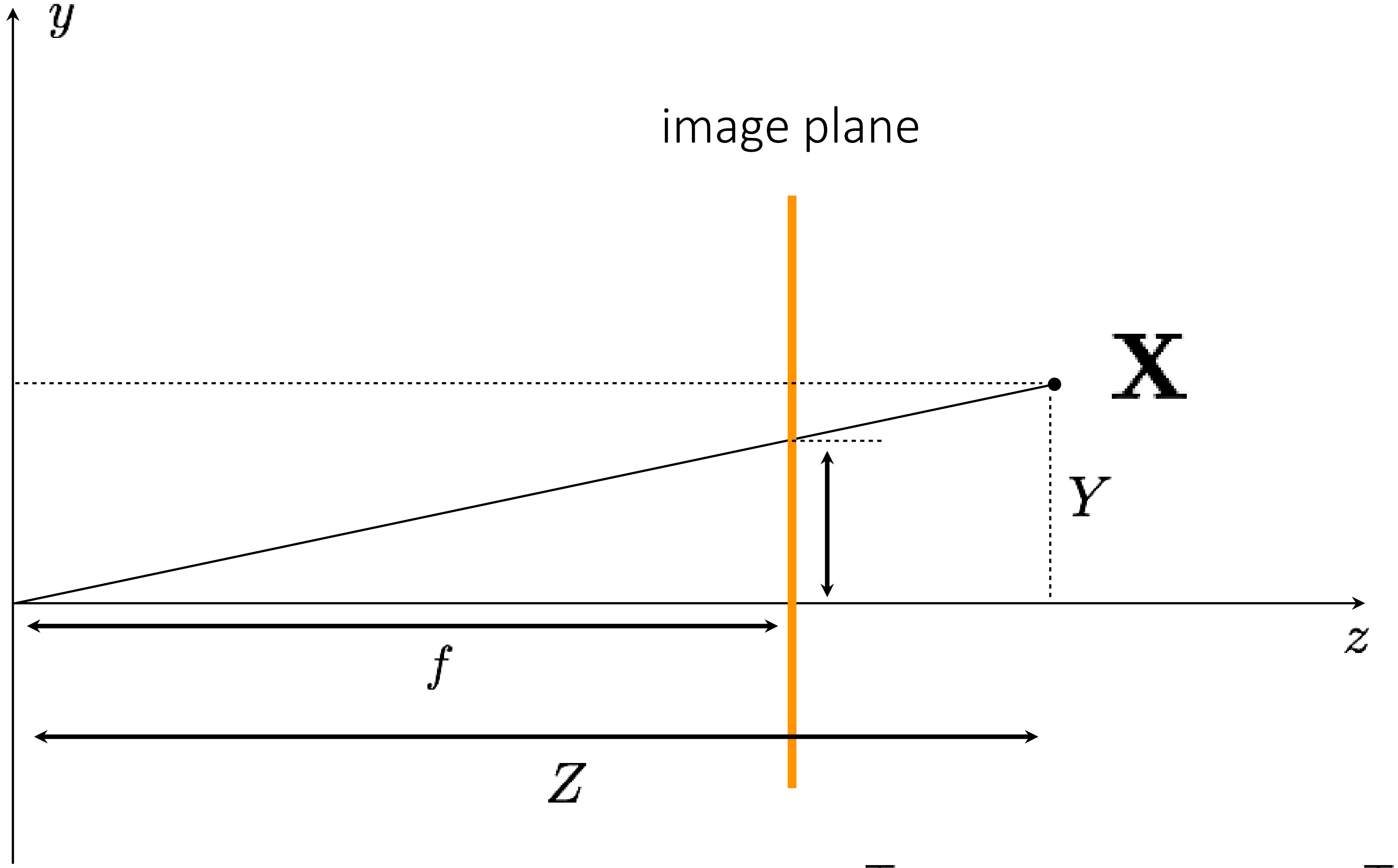
What is the equation for image coordinate **x** in terms of **X**?

The 2D view of the (rearranged) pinhole camera



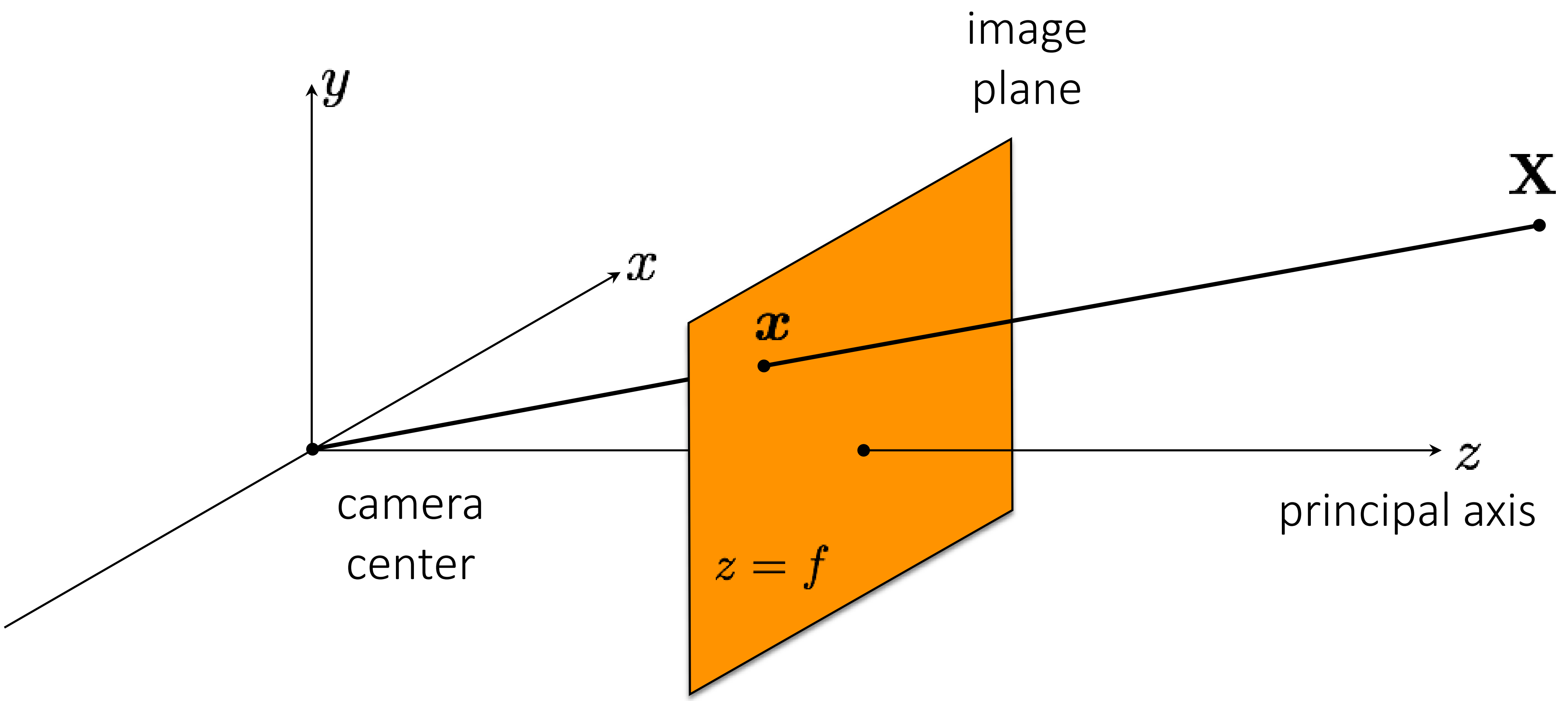
What is the equation for image coordinate **x** in terms of **X**?

The 2D view of the (rearranged) pinhole camera



$$[X \quad Y \quad Z]^\top \mapsto [fX/Z \quad fY/Z]^\top$$

The (rearranged) pinhole camera



What is the camera matrix **P** for a pinhole camera?

$$\boldsymbol{x} = \mathbf{P}\mathbf{X}$$

The pinhole camera matrix

Relationship from similar triangles:

$$[X \quad Y \quad Z]^\top \mapsto [fX/Z \quad fY/Z]^\top$$

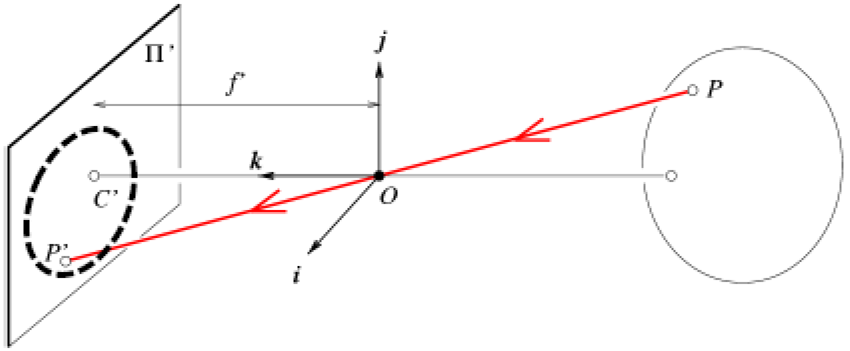
General camera model:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

What does the pinhole camera projection look like?

$$\mathbf{P} = \begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix}$$

Relating a real-world point to a point on the camera



In homogeneous coordinates:

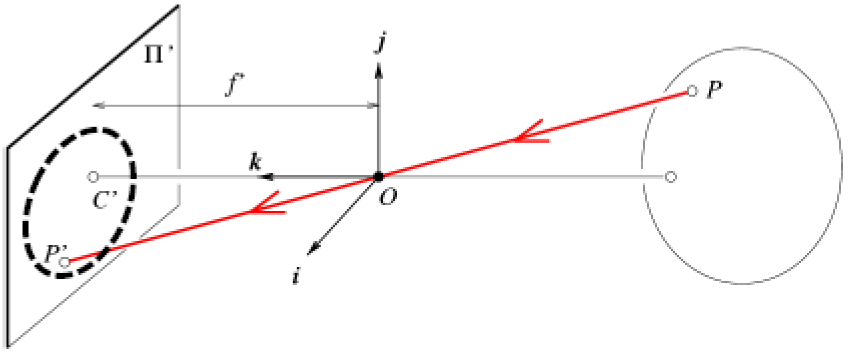
$$P' = \begin{bmatrix} f\,x \\ f\,y \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathrm{M}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

ideal world

Intrinsic Assumptions
- Unit aspect ratio
- Optical center at (0,0)
- No skew

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

Relating a real-world point to a point on the camera



In homogeneous coordinates:

$$P' = \begin{bmatrix} f\,x \\ f\,y \\ z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = K\begin{bmatrix} I & 0 \end{bmatrix} P$$

K

Intrinsic Assumptions

- Unit aspect ratio
- Optical center at (0,0)
- No skew

Extrinsic Assumptions

- No rotation
- Camera at (0,0,0)

Remove assumption: known optical center

Intrinsic Assumptions

- ~~Optical center at (0,0)~~
- Optical center at (u_0, v_0)
- Square pixels
- No skew

Extrinsic Assumptions

- No rotation
- Camera at (0,0,0)

$$P' = K\begin{bmatrix} I & 0 \end{bmatrix} P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Remove assumption: square pixels

Intrinsic Assumptions
- Optical center at (u_0, v_0)
- ~~Square pixels~~
- Rectangular pixels
- No skew

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

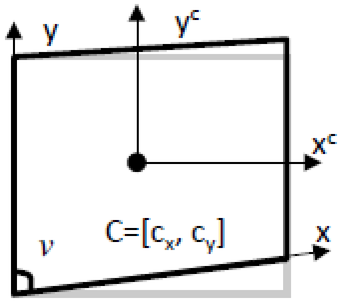
$$P' = K\begin{bmatrix} I & 0 \end{bmatrix} P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & u_0 & 0 \\ 0 & \beta & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Remove assumption: non-skewed pixels

Intrinsic Assumptions
- Optical center at (u_0, v_0)
- Rectangular pixels
- ~~No skew~~
- Small skew

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

$$P' = K\begin{bmatrix} I & 0 \end{bmatrix}P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \boxed{\begin{matrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{matrix}} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$



Remove assumption: non-skewed pixels

Intrinsic Assumptions

- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

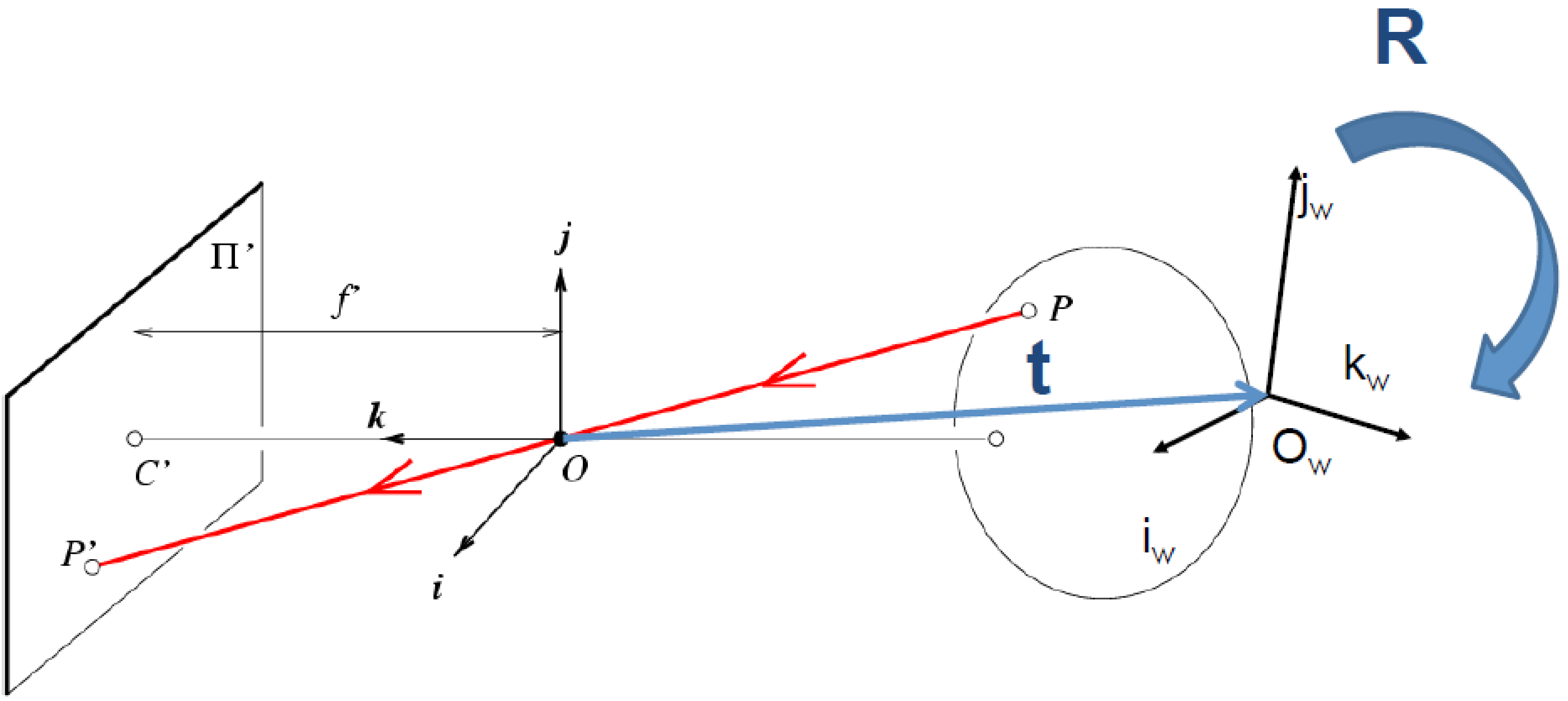
Extrinsic Assumptions

- No rotation
- Camera at (0,0,0)

Intrinsic parameters

$$P' = K\begin{bmatrix} I & 0 \end{bmatrix} P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 & 0 \\ 0 & \beta & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Real world camera: Translate + Rotate



Remove assumption: allow translation

Intrinsic Assumptions

- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

Extrinsic Assumptions

- No rotation
- Camera at ~~(0,0,0)~~ -> (t_x, t_y, t_z)

$$P' = K\begin{bmatrix} I & \vec{t} \end{bmatrix} P \quad \Rightarrow \quad w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

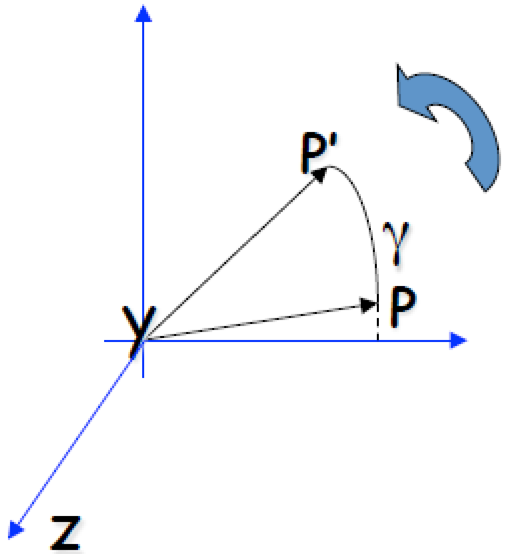
Remove assumption: allow rotation

Intrinsic Assumptions

- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

Extrinsic Assumptions

- ~~No~~ rotation
- Camera at (t_x, t_y, t_z)



Rotation around the coordinate axes, counter-clockwise

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Remove assumption: allow rotation

Intrinsic Assumptions
- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

Extrinsic Assumptions
- ~~No~~ rotation
- Camera at (t_x, t_y, t_z)

$$P' = K\begin{bmatrix} R & \bar{t} \end{bmatrix} P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

A generic projection matrix

Intrinsic Assumptions

- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

Extrinsic Assumptions

- Allow rotation
- Camera at (t_x, t_y, t_z)

$$P' = K\begin{bmatrix} R & \overline{t} \end{bmatrix} P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

A generic projection matrix

Intrinsic Assumptions

- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

Extrinsic Assumptions

- Allow rotation
- Camera at (t_x, t_y, t_z)

$$P' = K\begin{bmatrix} R & \bar{t} \end{bmatrix} P \quad \Rightarrow \quad w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Degrees of freedom??

A generic projection matrix

Intrinsic Assumptions

- Optical center at (u_0, v_0)
- Rectangular pixels
- Small skew

Extrinsic Assumptions

- Allow rotation
- Camera at (t_x,t_y,t_z)

$$P' = K\begin{bmatrix} R & \bar{t} \end{bmatrix} P \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{5} \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}}_{6} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Degrees of freedom??

Pose estimation

3D Pose Estimation

(Resectioning, Geometric Calibration, Perspective n-Point)

Given a set of matched points

$$\underbrace{\{\mathbf{X}_i}_{\text{point in 3D space}}, \underbrace{\boldsymbol{x}_i\}}_{\text{point in the image}}$$

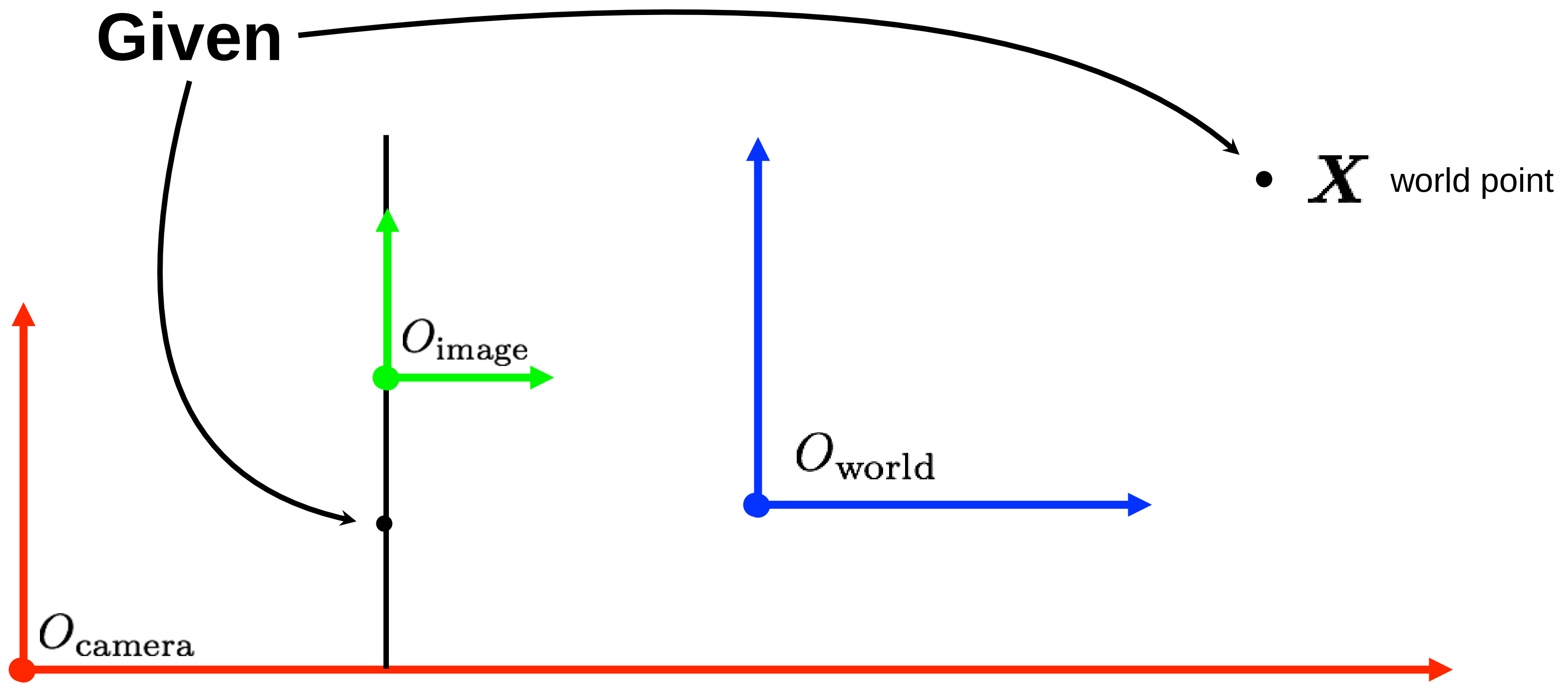
and camera model

$$\boldsymbol{x} = \underbrace{\boldsymbol{f}}_{\text{projection model}}(\mathbf{X}; \underbrace{\boldsymbol{p}}_{\text{parameters}}) = \underbrace{\mathbf{P}}_{\text{Camera matrix}}\mathbf{X}$$

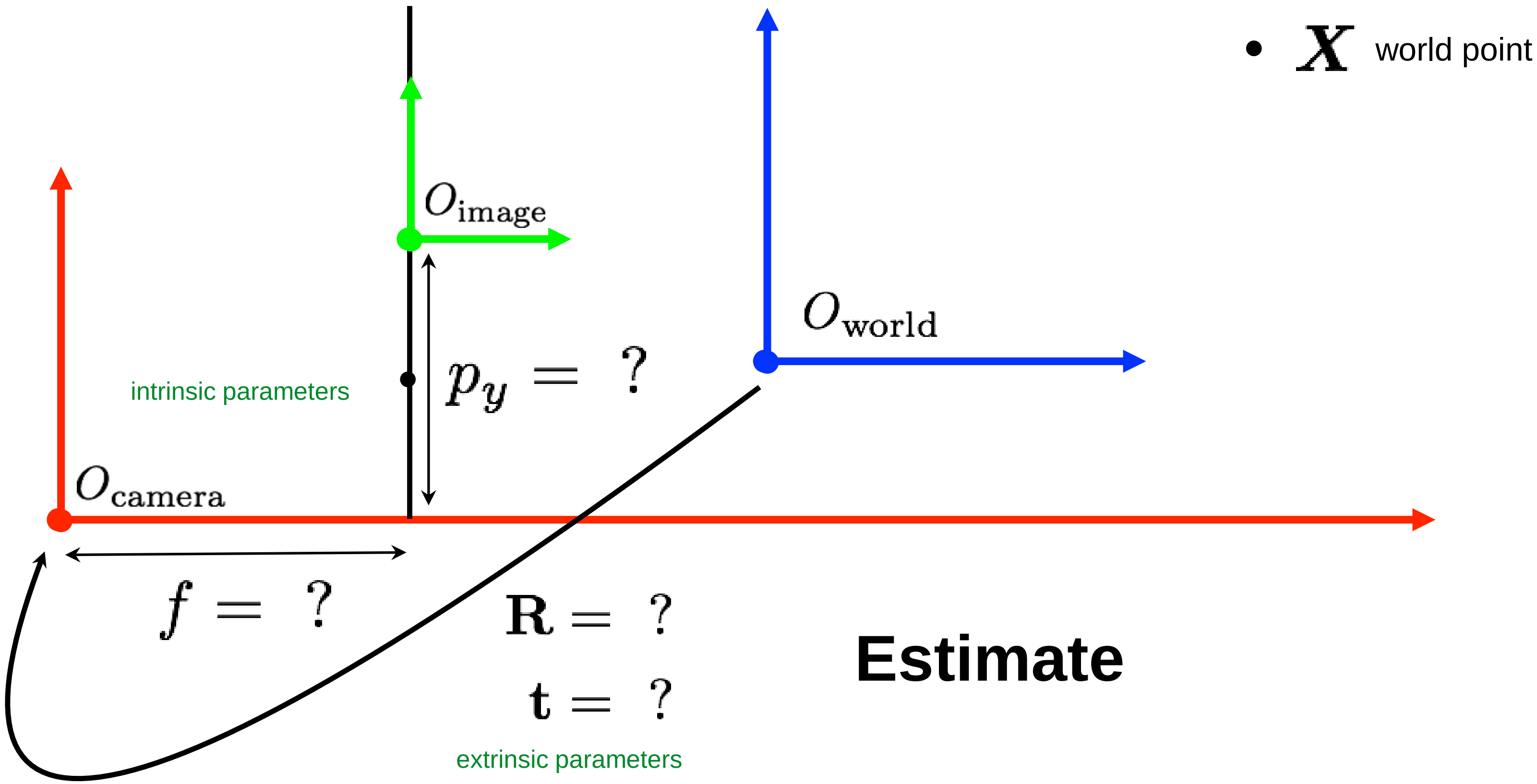
Find the (pose) estimate of

$$\mathbf{P}$$

What is Pose Estimation?



What is Pose Estimation?



Mapping between 3D point and image points

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

What are the unknowns?

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} - & \boldsymbol{p}_1^\top & - \\ - & \boldsymbol{p}_2^\top & - \\ - & \boldsymbol{p}_3^\top & - \end{bmatrix} \begin{bmatrix} | \\ \boldsymbol{X} \\ | \end{bmatrix}$$

Heterogeneous coordinates

$$x' = \frac{\boldsymbol{p}_1^\top \boldsymbol{X}}{\boldsymbol{p}_3^\top \boldsymbol{X}} \qquad y' = \frac{\boldsymbol{p}_2^\top \boldsymbol{X}}{\boldsymbol{p}_3^\top \boldsymbol{X}}$$

(non-linear correlation between coordinates)

How can we make these relations linear?

How can we make these relations linear?

$$x' = \frac{\boldsymbol{p}_1^\top \boldsymbol{X}}{\boldsymbol{p}_3^\top \boldsymbol{X}} \qquad y' = \frac{\boldsymbol{p}_2^\top \boldsymbol{X}}{\boldsymbol{p}_3^\top \boldsymbol{X}}$$

Make them linear with algebraic manipulation…

$$\boldsymbol{p}_2^\top \boldsymbol{X} - \boldsymbol{p}_3^\top \boldsymbol{X} y' = 0$$

$$\boldsymbol{p}_1^\top \boldsymbol{X} - \boldsymbol{p}_3^\top \boldsymbol{X} x' = 0$$

Now you can setup a system of linear equations with multiple point correspondences

$$\boldsymbol{p}_2^\top \boldsymbol{X} - \boldsymbol{p}_3^\top \boldsymbol{X} y' = 0$$

$$\boldsymbol{p}_1^\top \boldsymbol{X} - \boldsymbol{p}_3^\top \boldsymbol{X} x' = 0$$

In matrix form …
$$\begin{bmatrix} \boldsymbol{X}^\top & \boldsymbol{0} & -x'\boldsymbol{X}^\top \\ \boldsymbol{0} & \boldsymbol{X}^\top & -y'\boldsymbol{X}^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \boldsymbol{p}_3 \end{bmatrix} = \boldsymbol{0}$$

For N points …
$$\begin{bmatrix} \boldsymbol{X}_1^\top & \boldsymbol{0} & -x'\boldsymbol{X}_1^\top \\ \boldsymbol{0} & \boldsymbol{X}_1^\top & -y'\boldsymbol{X}_1^\top \\ \vdots & \vdots & \vdots \\ \boldsymbol{X}_N^\top & \boldsymbol{0} & -x'\boldsymbol{X}_N^\top \\ \boldsymbol{0} & \boldsymbol{X}_N^\top & -y'\boldsymbol{X}_N^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \boldsymbol{p}_3 \end{bmatrix} = \boldsymbol{0}$$

Solve for camera matrix by

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}}{\arg\min} \, ||\mathbf{A}\boldsymbol{x}||^2 \text{ subject to } ||\boldsymbol{x}||^2 = 1$$

$$\mathbf{A} = \begin{bmatrix} \boldsymbol{X}_1^\top & \mathbf{0} & -x'\boldsymbol{X}_1^\top \\ \mathbf{0} & \boldsymbol{X}_1^\top & -y'\boldsymbol{X}_1^\top \\ \vdots & \vdots & \vdots \\ \boldsymbol{X}_N^\top & \mathbf{0} & -x'\boldsymbol{X}_N^\top \\ \mathbf{0} & \boldsymbol{X}_N^\top & -y'\boldsymbol{X}_N^\top \end{bmatrix} \qquad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \boldsymbol{p}_3 \end{bmatrix}$$

SVD!

Solve for camera matrix by

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|\mathbf{A}\boldsymbol{x}\|^2 \text{ subject to } \|\boldsymbol{x}\|^2 = 1$$

$$\mathbf{A} = \begin{bmatrix} \boldsymbol{X}_1^\top & \mathbf{0} & -x'\boldsymbol{X}_1^\top \\ \mathbf{0} & \boldsymbol{X}_1^\top & -y'\boldsymbol{X}_1^\top \\ \vdots & \vdots & \vdots \\ \boldsymbol{X}_N^\top & \mathbf{0} & -x'\boldsymbol{X}_N^\top \\ \mathbf{0} & \boldsymbol{X}_N^\top & -y'\boldsymbol{X}_N^\top \end{bmatrix} \qquad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \boldsymbol{p}_3 \end{bmatrix}$$

Solution **x** is the column of **V** corresponding to smallest singular value of

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$$

Solve for camera matrix by

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|\mathbf{A}\boldsymbol{x}\|^2 \text{ subject to } \|\boldsymbol{x}\|^2 = 1$$

$$\mathbf{A} = \begin{bmatrix} \boldsymbol{X}_1^\top & \mathbf{0} & -x'\boldsymbol{X}_1^\top \\ \mathbf{0} & \boldsymbol{X}_1^\top & -y'\boldsymbol{X}_1^\top \\ \vdots & \vdots & \vdots \\ \boldsymbol{X}_N^\top & \mathbf{0} & -x'\boldsymbol{X}_N^\top \\ \mathbf{0} & \boldsymbol{X}_N^\top & -y'\boldsymbol{X}_N^\top \end{bmatrix} \qquad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \boldsymbol{p}_3 \end{bmatrix}$$

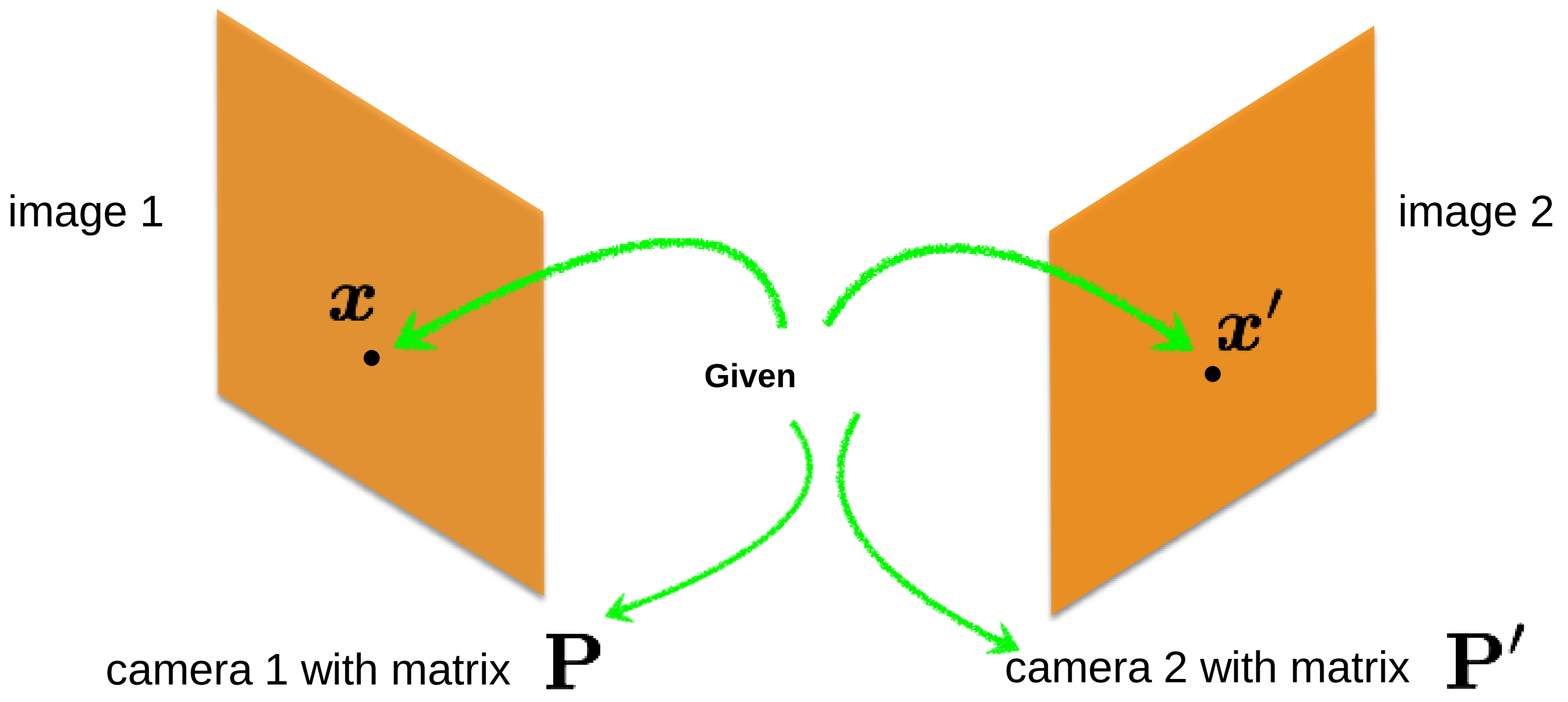
Equivalently, solution **x** is the Eigenvector corresponding to smallest Eigenvalue of $\mathbf{A}^\top\mathbf{A}$

Other topics for today

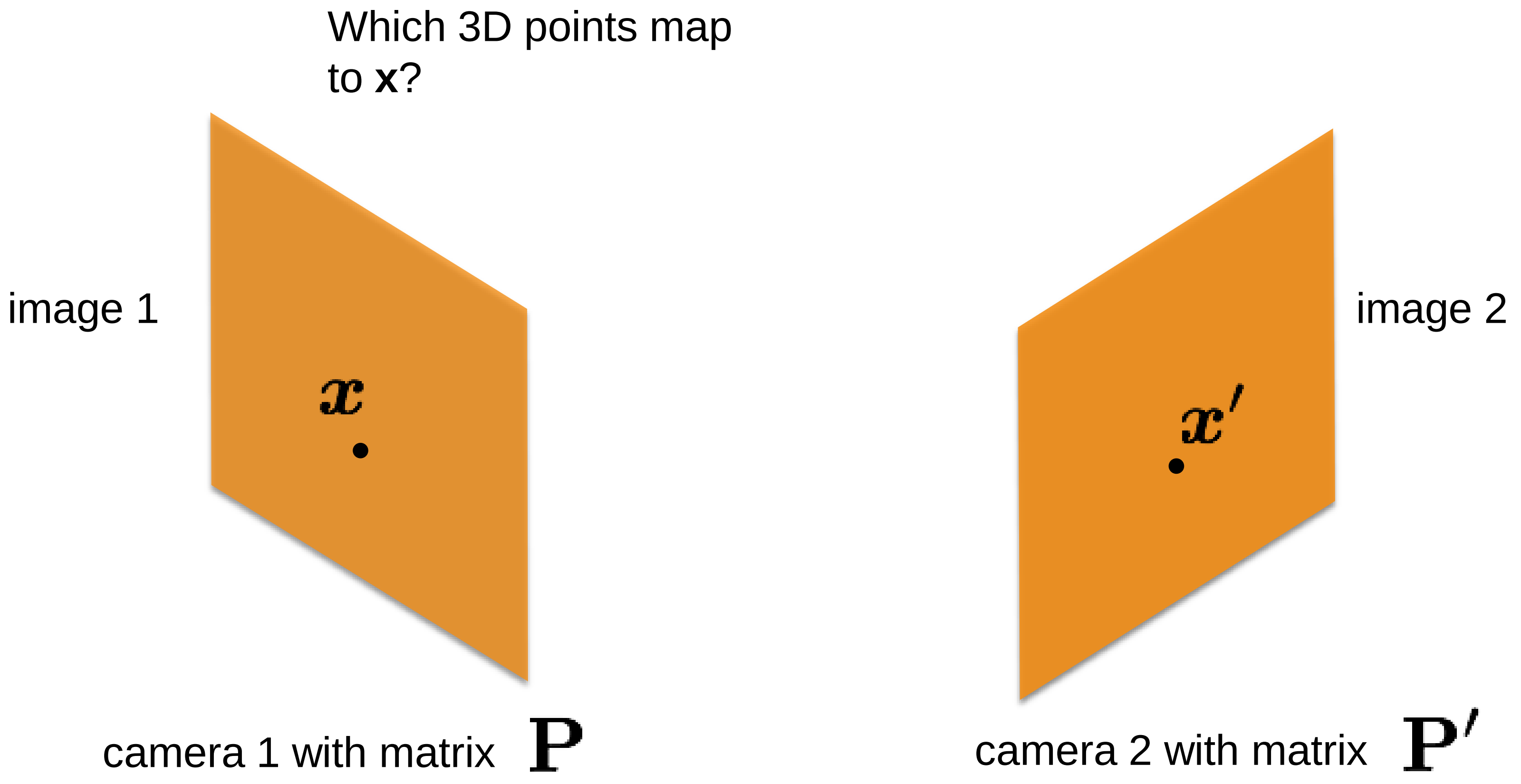
- Triangulation.
- Epipolar geometry.
- Essential matrix.
- Fundamental matrix.
- 8-point algorithm.

Triangulation

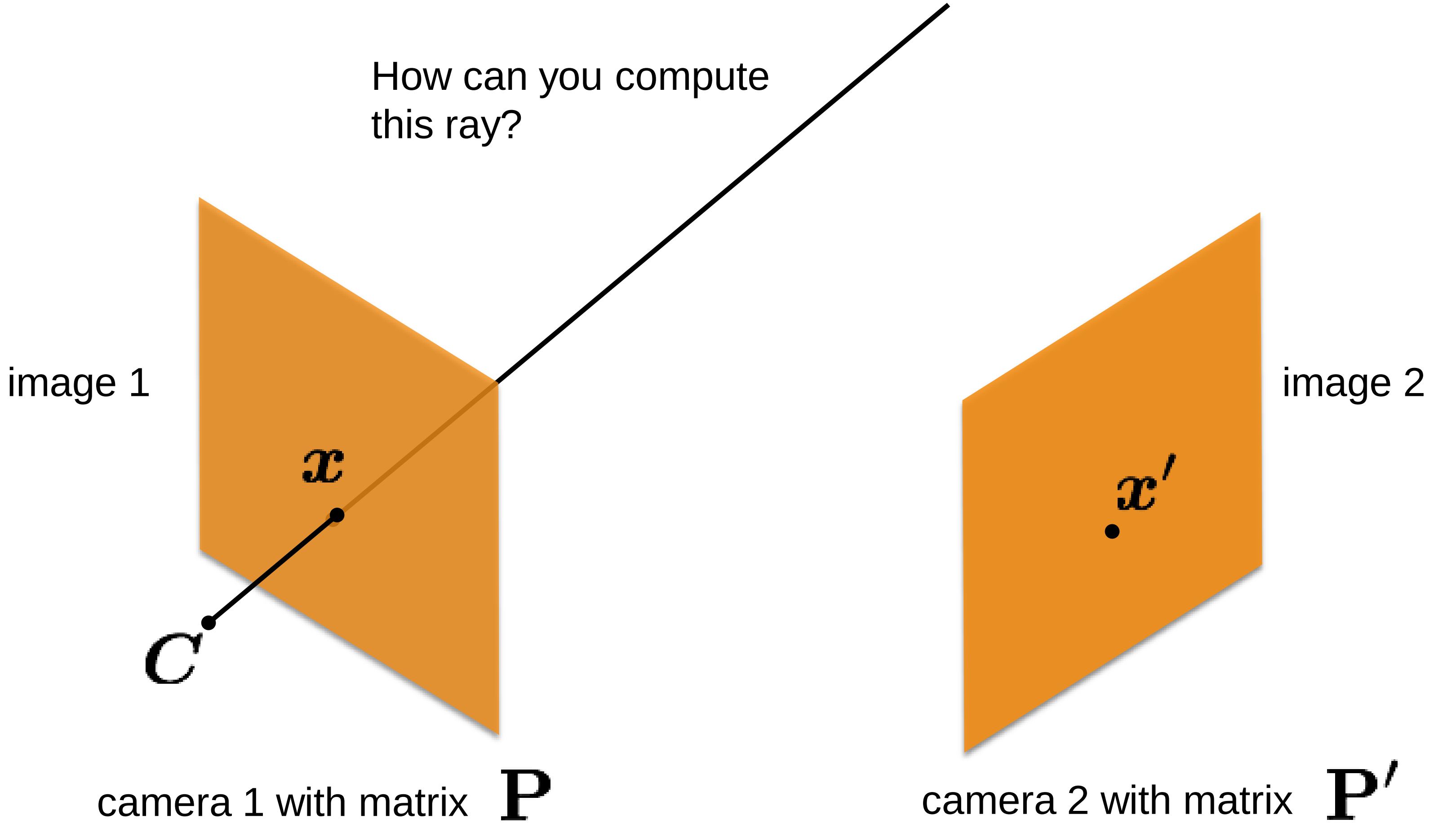
Triangulation



Triangulation



Triangulation

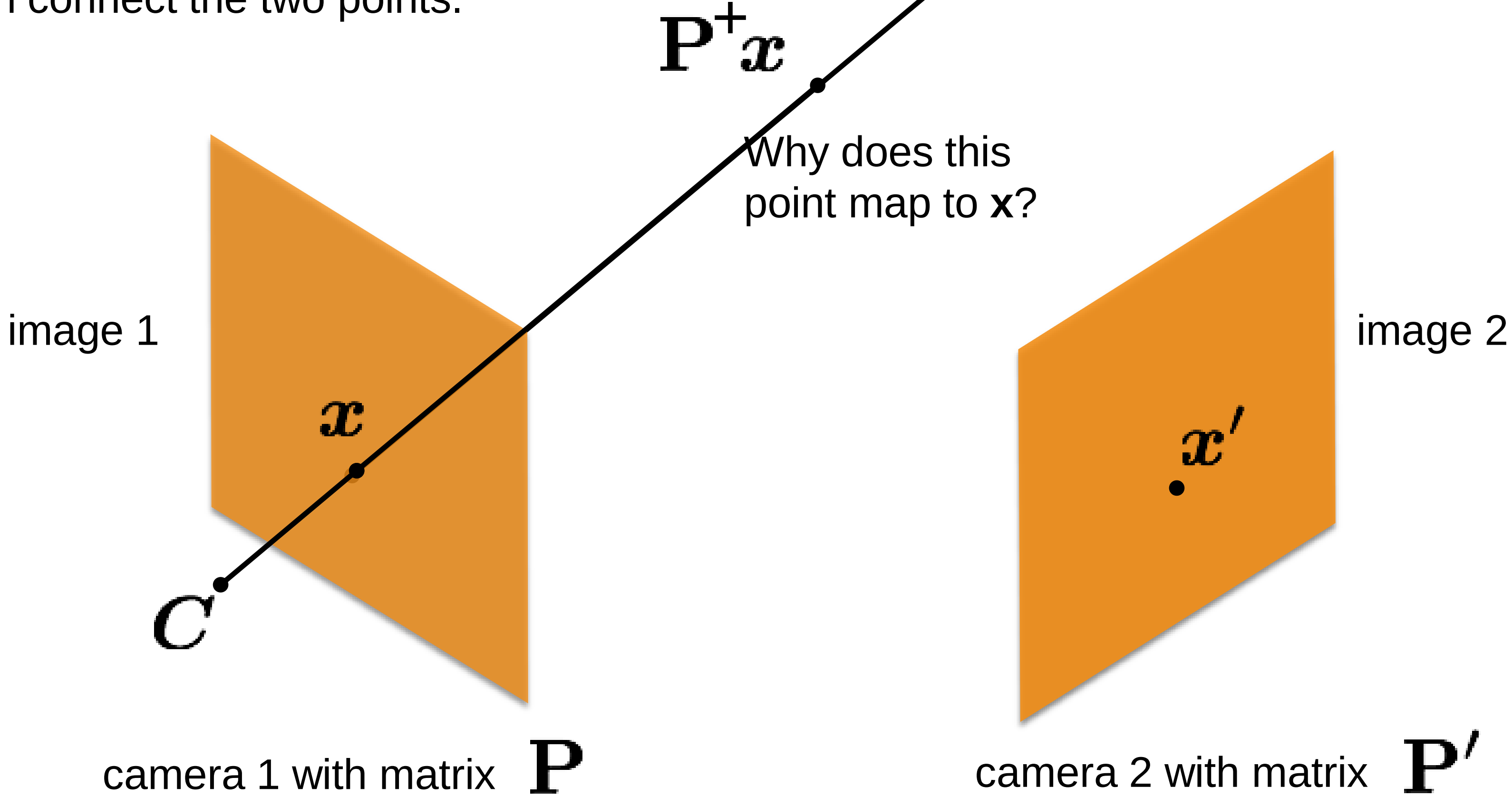


Triangulation

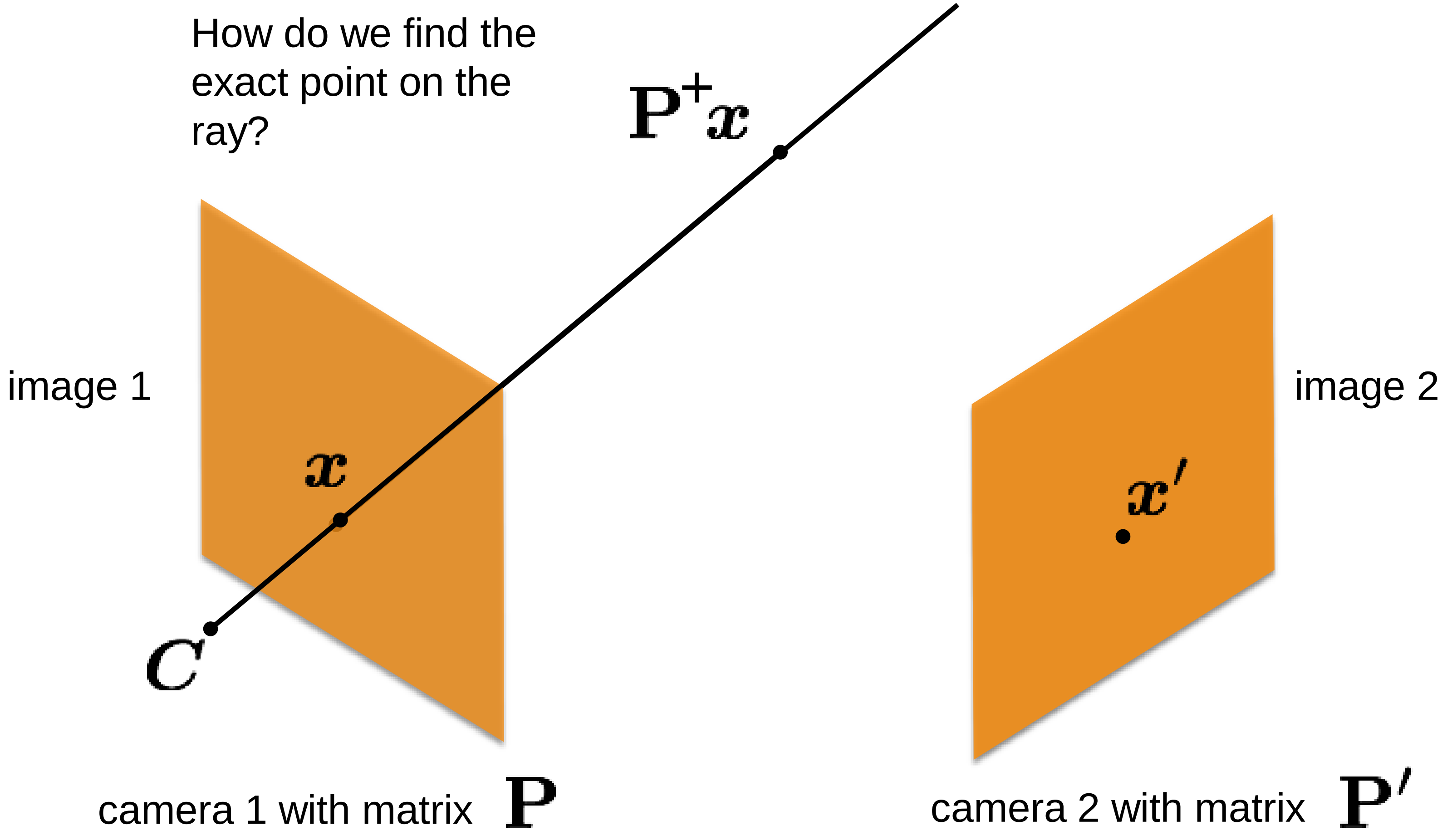
Create two points on the ray:

1) find the camera center; and
2) apply the pseudo-inverse of **P** on **x**.

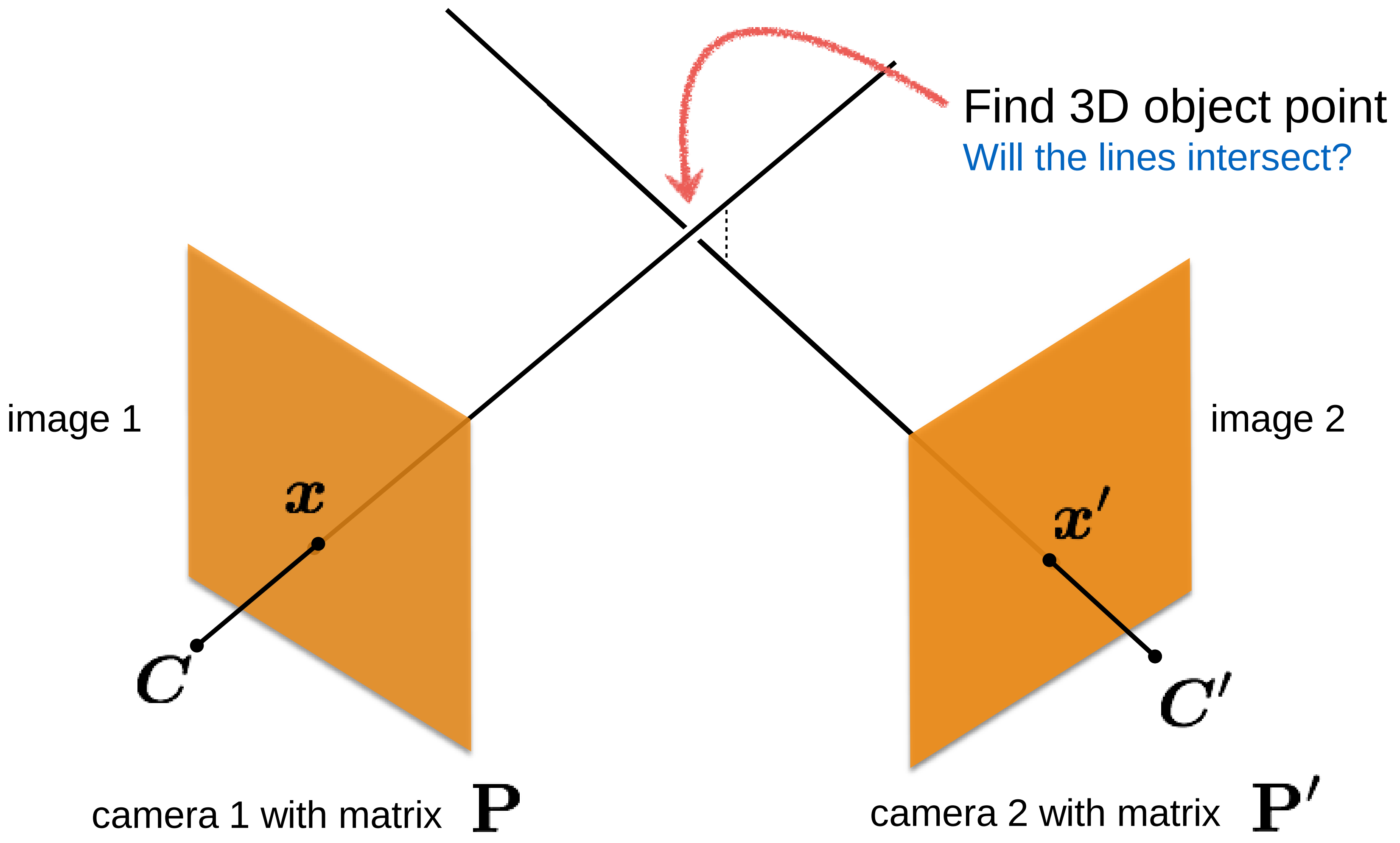
Then connect the two points.



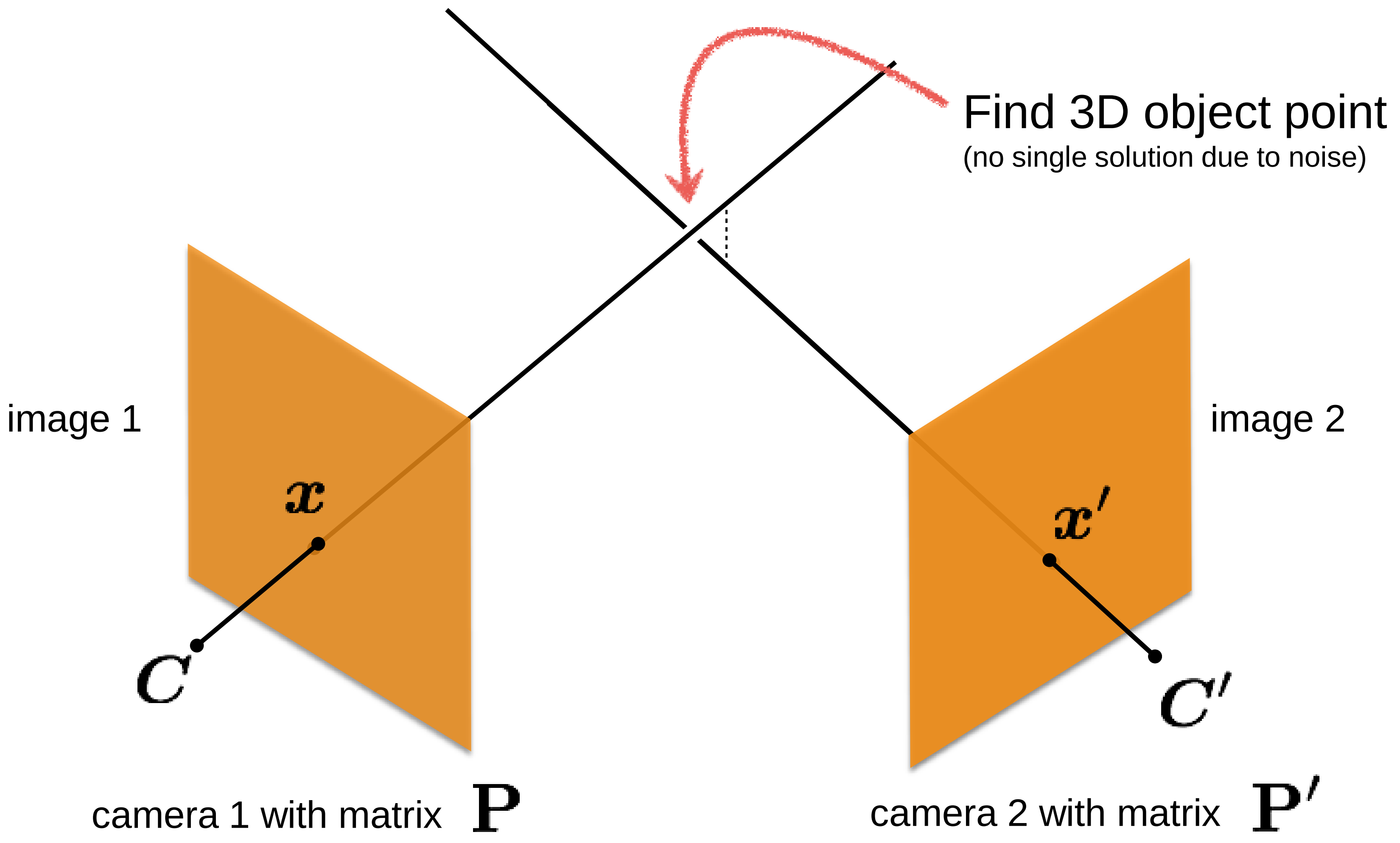
Triangulation



Triangulation



Triangulation



Triangulation

Given a set of (noisy) matched points

$$\{\boldsymbol{x}_i, \boldsymbol{x}'_i\}$$

and camera matrices

$$\mathbf{P}, \mathbf{P}'$$

Estimate the 3D point

$$\mathbf{X}$$

$$\mathbf{x} = \mathbf{P}\boldsymbol{X}$$

known known

Can we compute **X** from a single correspondence **x**?

$$\mathbf{x} = \mathbf{P}\boldsymbol{X}$$

known known

Can we compute X from two *correspondences x and x'?*

$$\underset{\text{known}}{\mathbf{x}} = \underset{\text{known}}{\mathbf{P}}\boldsymbol{X}$$

Can we compute X from two *correspondences x and x'?*

Yes, if perfect measurements

$$\mathbf{x} = \mathbf{P}\boldsymbol{X}$$

known known

Can we compute X from two *correspondences x and x'?*

Yes, if perfect measurements

There will not be a point that satisfies both constraints because the measurements are usually noisy

$$\mathbf{x}' = \mathbf{P}'\boldsymbol{X} \qquad \mathbf{x} = \mathbf{P}\boldsymbol{X}$$

Need to find the **best fit**

$$\mathbf{x} = \mathbf{P}\boldsymbol{X}$$

(homogeneous coordinate)

Also, this is a similarity relation because it involves homogeneous coordinates

$$\mathbf{x} = \alpha\mathbf{P}\boldsymbol{X}$$

(homorogeneous coordinate)

Same ray direction but differs by a scale factor

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

How do we solve for unknowns in a similarity relation?

$$\mathbf{x} = \mathbf{P}\boldsymbol{X}$$

(homogeneous
coordinate)

Also, this is a similarity relation because it involves homogeneous coordinates

$$\mathbf{x} = \alpha\mathbf{P}\boldsymbol{X}$$

(homorogeneous
coordinate)

Same ray direction but differs by a scale factor

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

How do we solve for unknowns in a similarity relation?

Remove scale factor, convert to linear system and solve with [redacted]

$$\mathbf{x} = \mathbf{P}\boldsymbol{X}$$

(homogeneous coordinate)

Also, this is a similarity relation because it involves homogeneous coordinates

$$\mathbf{x} = \alpha\mathbf{P}\boldsymbol{X}$$

(homorogeneous coordinate)

Same ray direction but differs by a scale factor

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

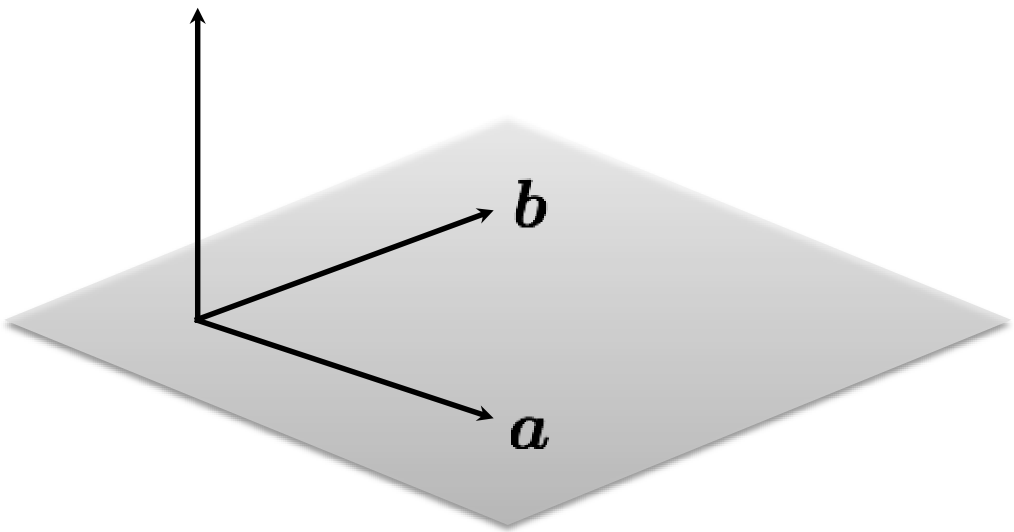
How do we solve for unknowns in a similarity relation?

Remove scale factor, convert to linear system and solve with SVD

Recall: Cross Product

Vector (cross) product
takes two vectors and returns a vector perpendicular to both

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



$$\boldsymbol{a} \times \boldsymbol{b} = \begin{bmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{bmatrix}$$

cross product of two vectors in the same direction is zero

$$\boldsymbol{a} \times \boldsymbol{a} = 0$$

remember this!!!

$$\boldsymbol{c} \cdot \boldsymbol{a} = 0 \qquad \boldsymbol{c} \cdot \boldsymbol{b} = 0$$

$$\mathbf{x} = \alpha \mathbf{P} \boldsymbol{X}$$

Same direction but differs by a scale factor

$$\mathbf{x} \times \mathbf{P} \boldsymbol{X} = \mathbf{0}$$

Cross product of two vectors of same direction is zero
(this equality removes the scale factor)

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} \text{---} & \boldsymbol{p}_1^\top & \text{---} \\ \text{---} & \boldsymbol{p}_2^\top & \text{---} \\ \text{---} & \boldsymbol{p}_3^\top & \text{---} \end{bmatrix} \begin{bmatrix} | \\ \boldsymbol{X} \\ | \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} \boldsymbol{p}_1^\top \boldsymbol{X} \\ \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_3^\top \boldsymbol{X} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} \text{---} & \boldsymbol{p}_1^\top & \text{---} \\ \text{---} & \boldsymbol{p}_2^\top & \text{---} \\ \text{---} & \boldsymbol{p}_3^\top & \text{---} \end{bmatrix} \begin{bmatrix} | \\ \boldsymbol{X} \\ | \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} \boldsymbol{p}_1^\top \boldsymbol{X} \\ \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_3^\top \boldsymbol{X} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \times \begin{bmatrix} \boldsymbol{p}_1^\top \boldsymbol{X} \\ \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_3^\top \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} y\boldsymbol{p}_3^\top \boldsymbol{X} - \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_1^\top \boldsymbol{X} - x\boldsymbol{p}_3^\top \boldsymbol{X} \\ x\boldsymbol{p}_2^\top \boldsymbol{X} - y\boldsymbol{p}_1^\top \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Using the fact that the cross product should be zero

$$\mathbf{x} \times \mathbf{P}\boldsymbol{X} = \mathbf{0}$$

$$\left[\begin{array}{c} x \\ y \\ 1 \end{array}\right] \times \left[\begin{array}{c} \boldsymbol{p}_1^\top \boldsymbol{X} \\ \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_3^\top \boldsymbol{X} \end{array}\right] = \left[\begin{array}{c} y\boldsymbol{p}_3^\top \boldsymbol{X} - \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_1^\top \boldsymbol{X} - x\boldsymbol{p}_3^\top \boldsymbol{X} \\ x\boldsymbol{p}_2^\top \boldsymbol{X} - y\boldsymbol{p}_1^\top \boldsymbol{X} \end{array}\right] = \left[\begin{array}{c} 0 \\ 0 \\ 0 \end{array}\right]$$

Third line is a linear combination of the first and second lines.
(x times the first line plus y times the second line)

One 2D to 3D point correspondence give you [redacted] equations

Using the fact that the cross product should be zero

$$\mathbf{x} \times \mathbf{P}\boldsymbol{X} = \mathbf{0}$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \times \begin{bmatrix} \boldsymbol{p}_1^\top \boldsymbol{X} \\ \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_3^\top \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} y\boldsymbol{p}_3^\top \boldsymbol{X} - \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_1^\top \boldsymbol{X} - x\boldsymbol{p}_3^\top \boldsymbol{X} \\ x\boldsymbol{p}_2^\top \boldsymbol{X} - y\boldsymbol{p}_1^\top \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Third line is a linear combination of the first and second lines.
(x times the first line plus y times the second line)

One 2D to 3D point correspondence give you 2 equations

$$\begin{bmatrix} y\boldsymbol{p}_3^\top \boldsymbol{X} - \boldsymbol{p}_2^\top \boldsymbol{X} \\ \boldsymbol{p}_1^\top \boldsymbol{X} - x\boldsymbol{p}_3^\top \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} y\boldsymbol{p}_3^\top - \boldsymbol{p}_2^\top \\ \boldsymbol{p}_1^\top - x\boldsymbol{p}_3^\top \end{bmatrix} \boldsymbol{X} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\mathbf{A}_i \boldsymbol{X} = \mathbf{0}$$

Now we can make a system of linear equations
(two lines for each 2D point correspondence)

Concatenate the 2D points from both images

$$\begin{bmatrix} y\boldsymbol{p}_3^\top - \boldsymbol{p}_2^\top \\ \boldsymbol{p}_1^\top - x\boldsymbol{p}_3^\top \\ y'\boldsymbol{p}_3'^\top - \boldsymbol{p}_2'^\top \\ \boldsymbol{p}_1'^\top - x'\boldsymbol{p}_3'^\top \end{bmatrix} \boldsymbol{X} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

sanity check! dimensions?

$$\mathbf{A}\boldsymbol{X} = \mathbf{0}$$

How do we solve homogeneous linear system?

Concatenate the 2D points from both images

$$\begin{bmatrix} y\boldsymbol{p}_3^\top - \boldsymbol{p}_2^\top \\ \boldsymbol{p}_1^\top - x\boldsymbol{p}_3^\top \\ y'\boldsymbol{p}_3'^\top - \boldsymbol{p}_2'^\top \\ \boldsymbol{p}_1'^\top - x'\boldsymbol{p}_3'^\top \end{bmatrix} \boldsymbol{X} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{A}\boldsymbol{X} = \mathbf{0}$$

How do we solve homogeneous linear system?

S V D !

Recall: Total least squares

(**Warning:** change of notation. x is a vector of parameters!)

$$E_{\mathrm{TLS}} = \sum_i (\boldsymbol{a}_i \boldsymbol{x})^2$$

$$= \|\mathbf{A}\boldsymbol{x}\|^2 \qquad \text{(matrix form)}$$

$$\|\boldsymbol{x}\|^2 = 1 \qquad \text{constraint}$$

minimize $\|\mathbf{A}\boldsymbol{x}\|^2$

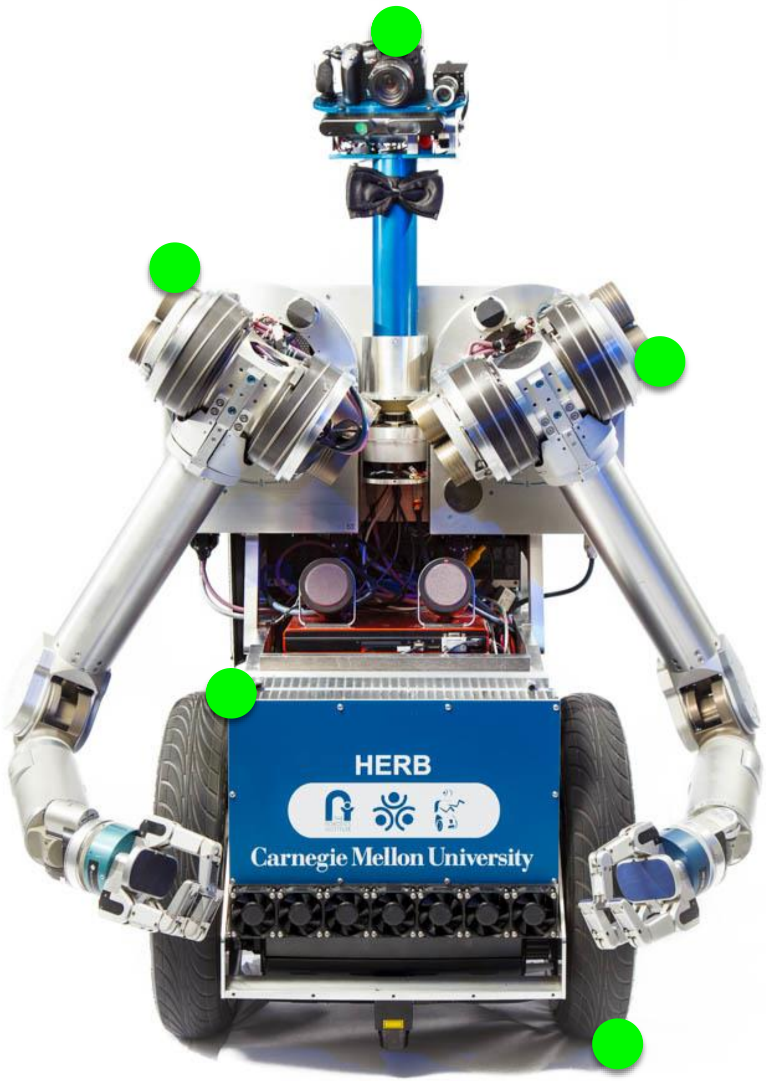
subject to $\|\boldsymbol{x}\|^2 = 1$

$\Rightarrow$ minimize $\dfrac{\|\mathbf{A}\boldsymbol{x}\|^2}{\|\boldsymbol{x}\|^2}$

(Rayleigh quotient)

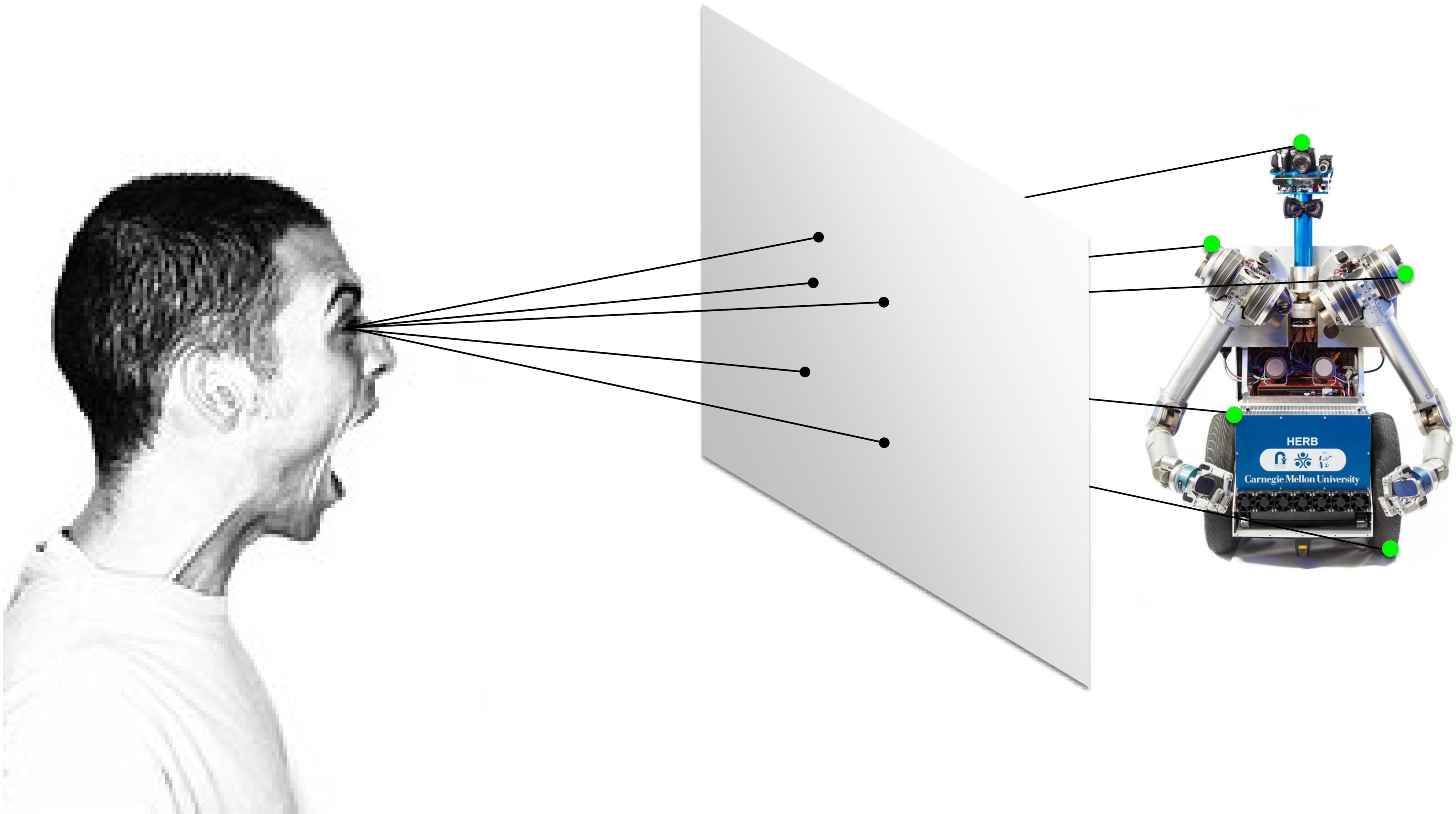
Solution is the eigenvector
corresponding to smallest eigenvalue of $\mathbf{A}^\top \mathbf{A}$

Epipolar geometry

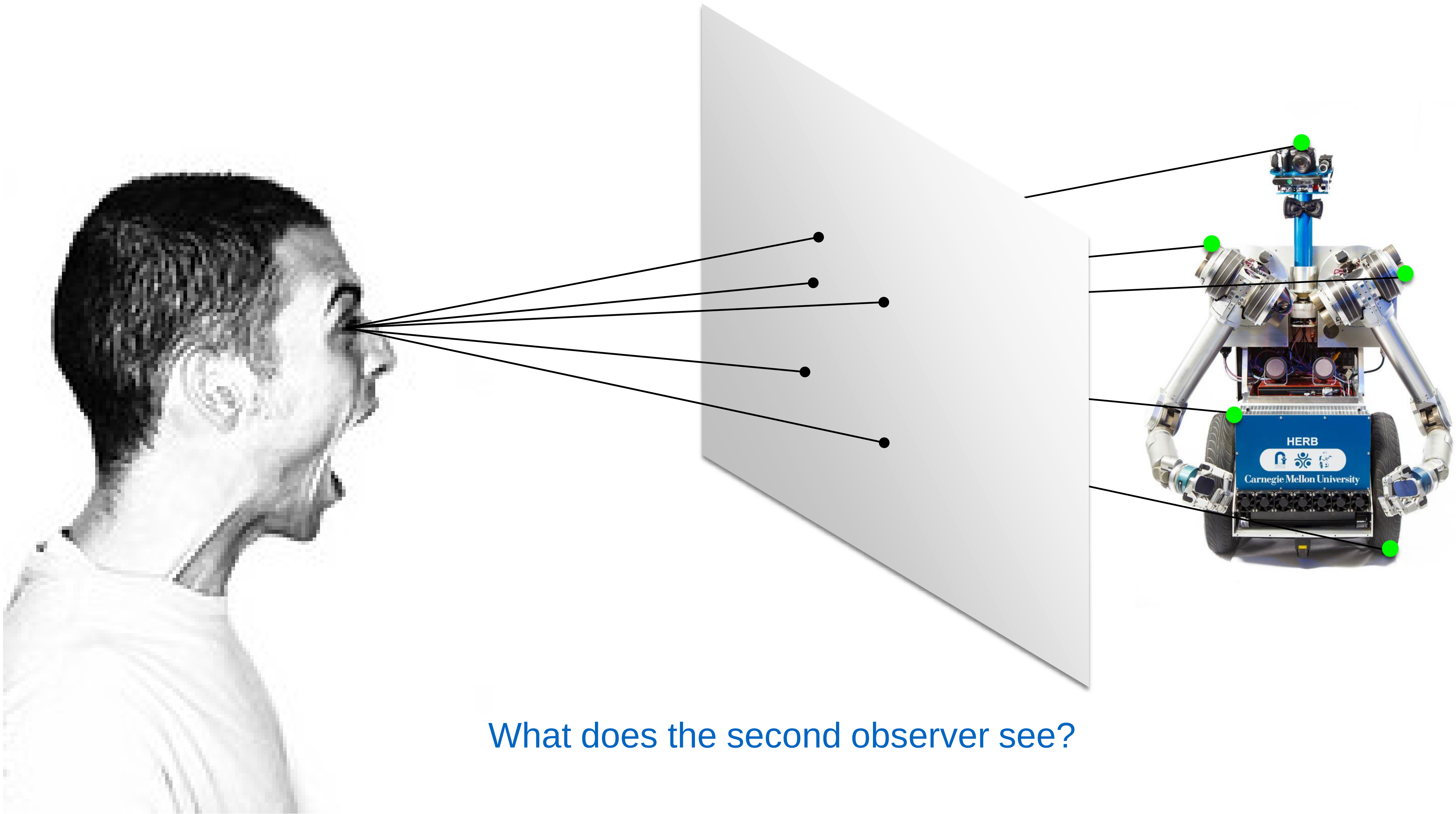


Tie tiny threads on HERB and pin them to your eyeball

What would it look like?

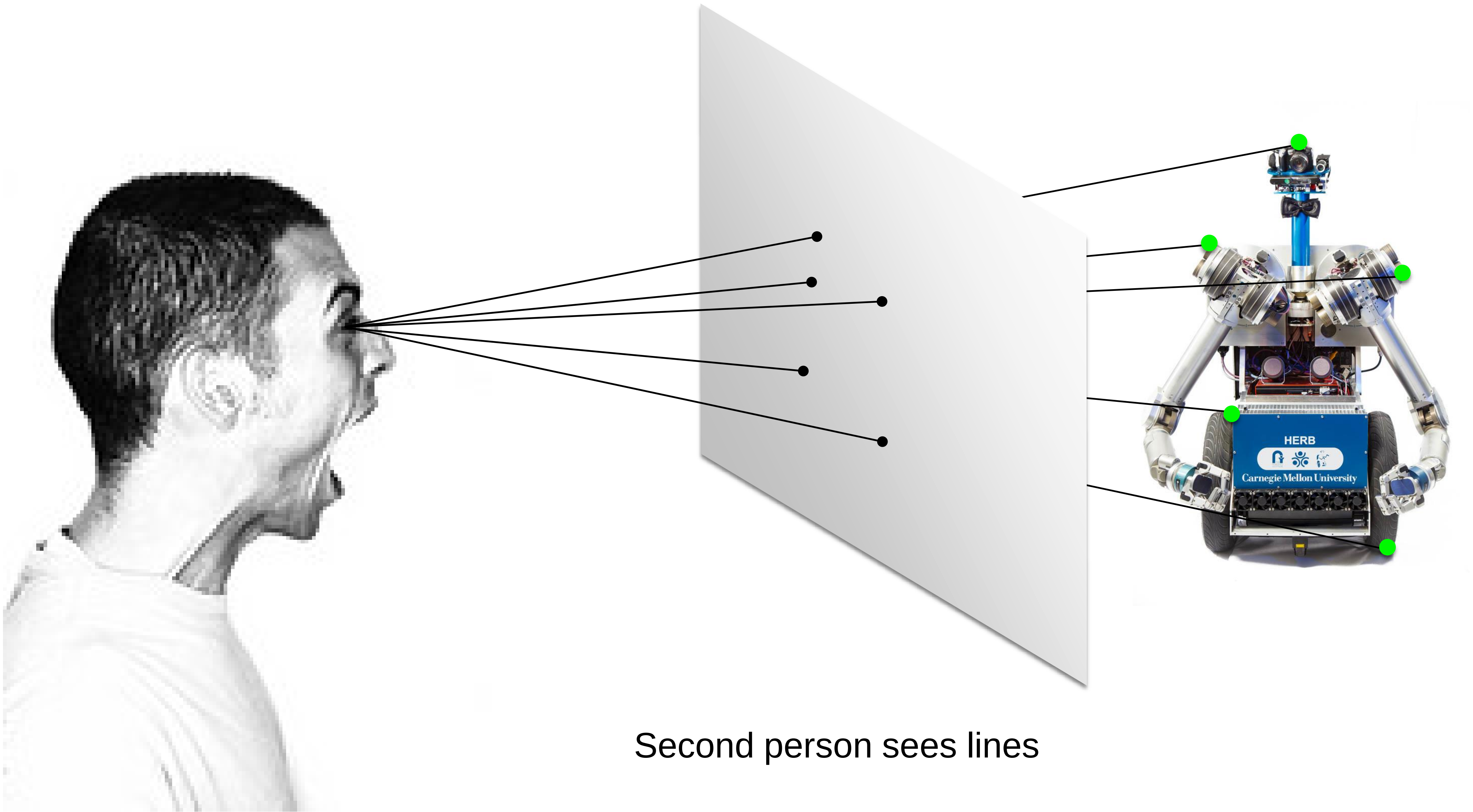

HERB
Carnegie Mellon University

You see points on HERB



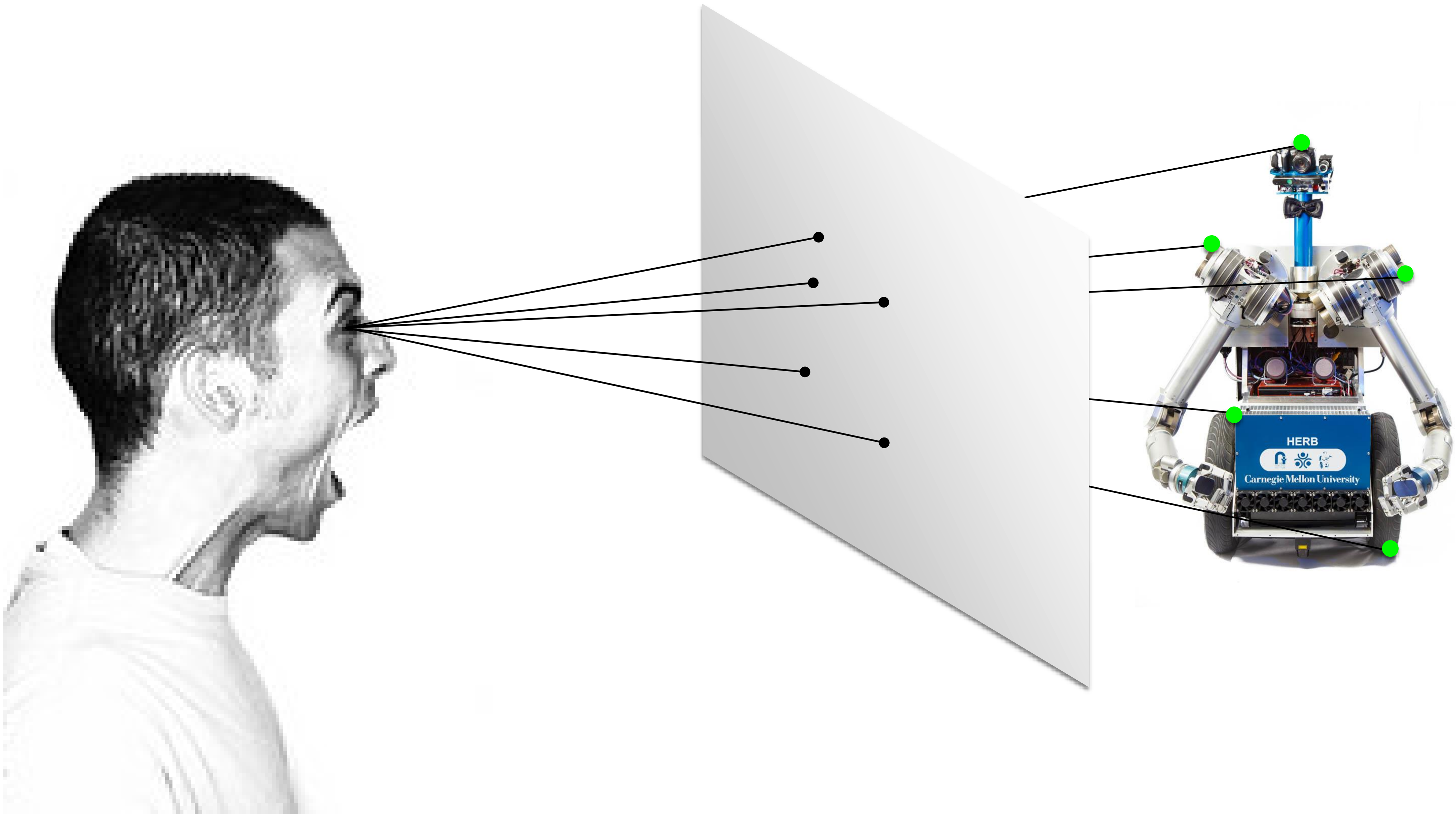
What does the second observer see?

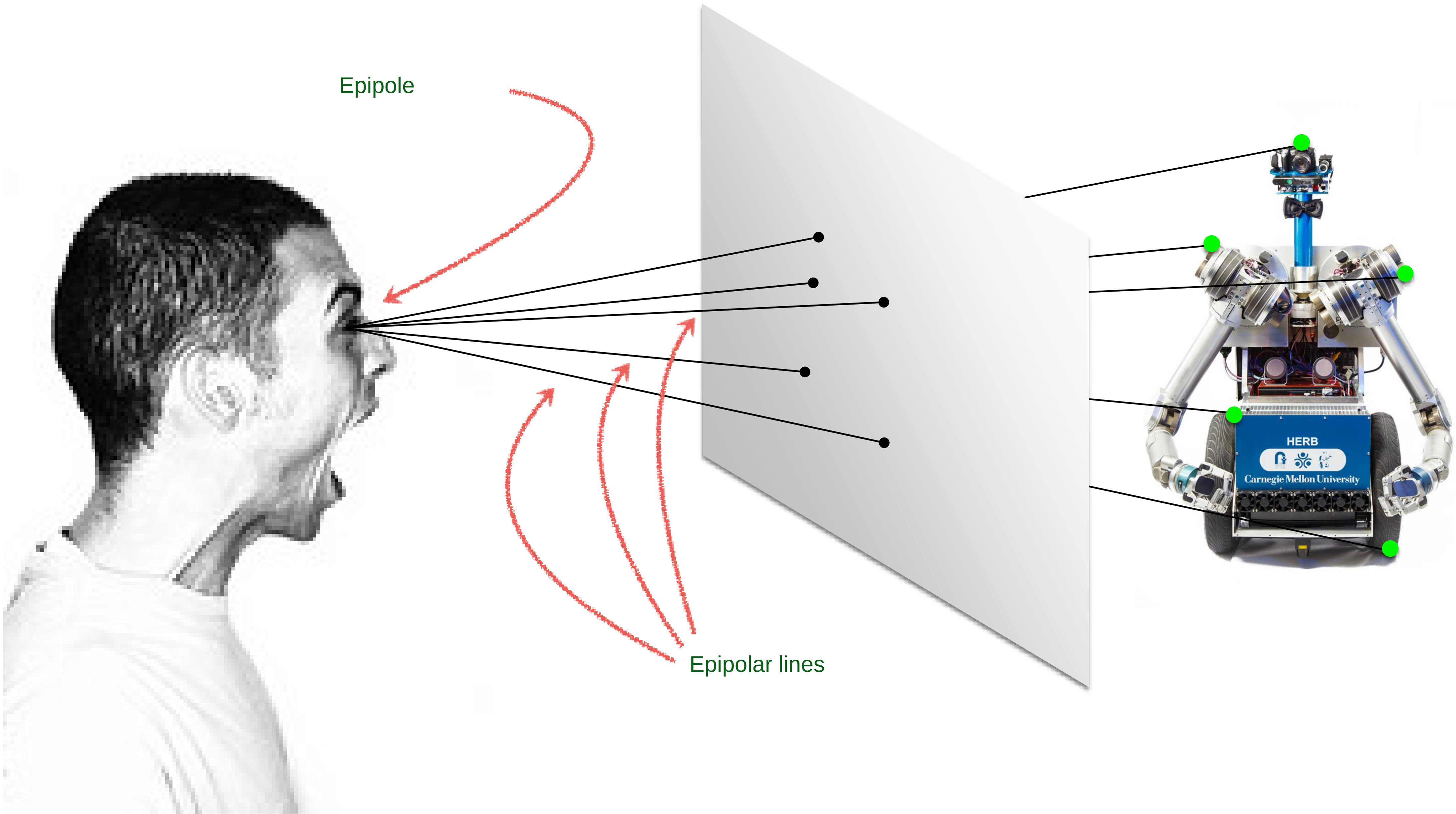
You see points on HERB



Second person sees lines

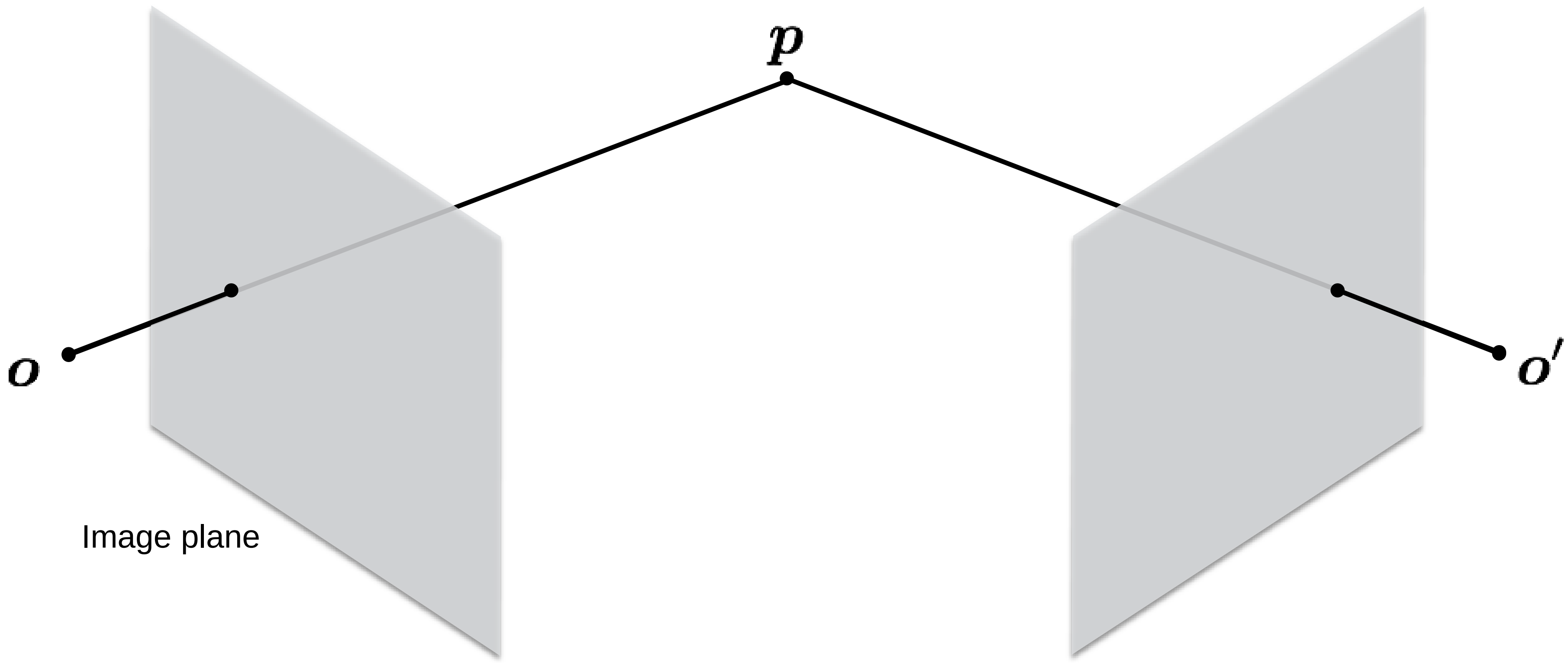
This is Epipolar Geometry



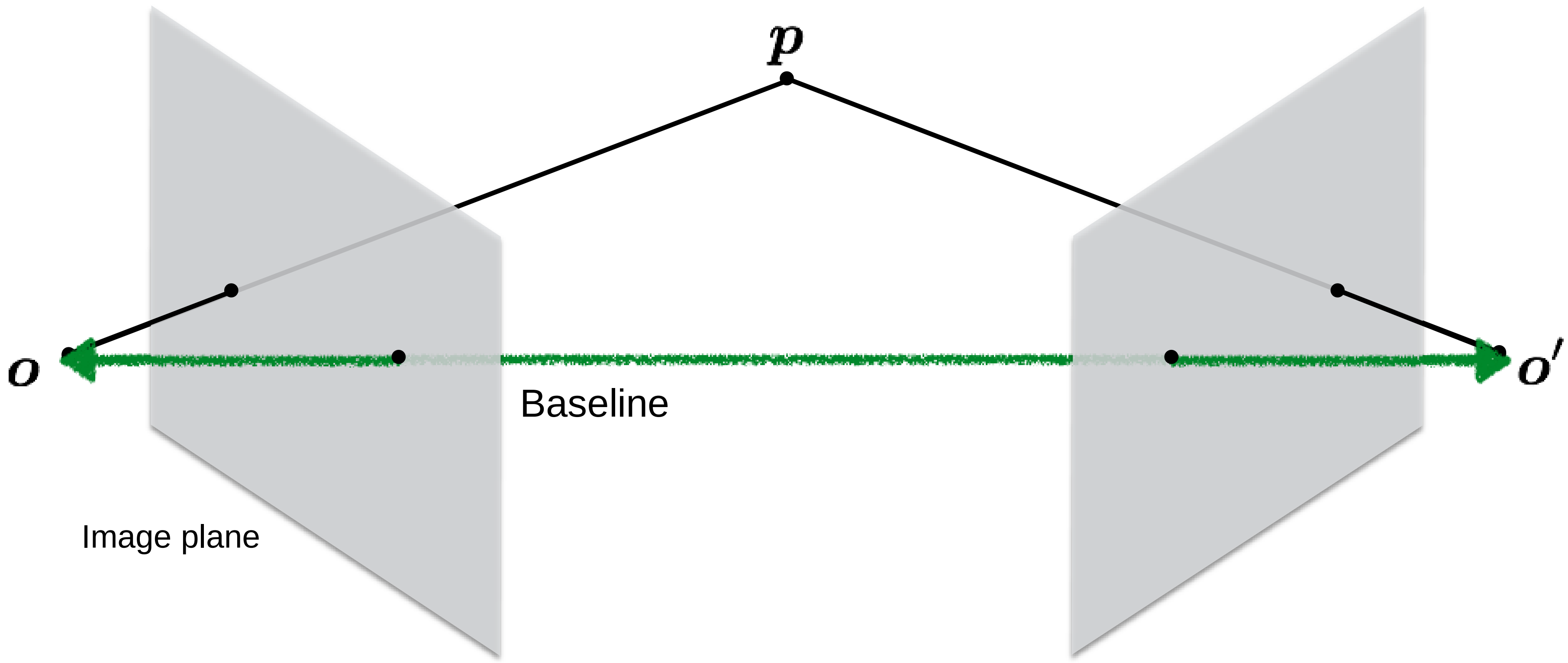


Epipole
Epipolar lines
HERB
Carnegie Mellon University

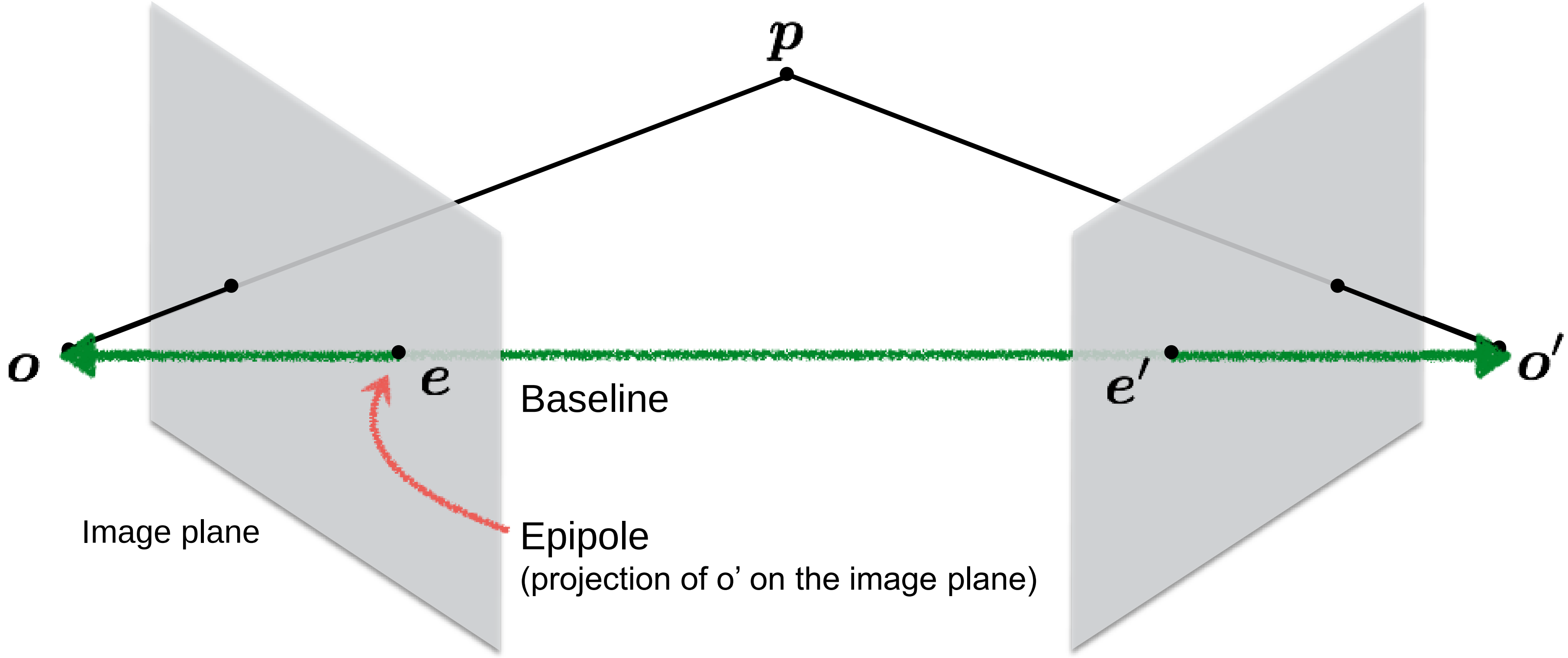
Epipolar geometry



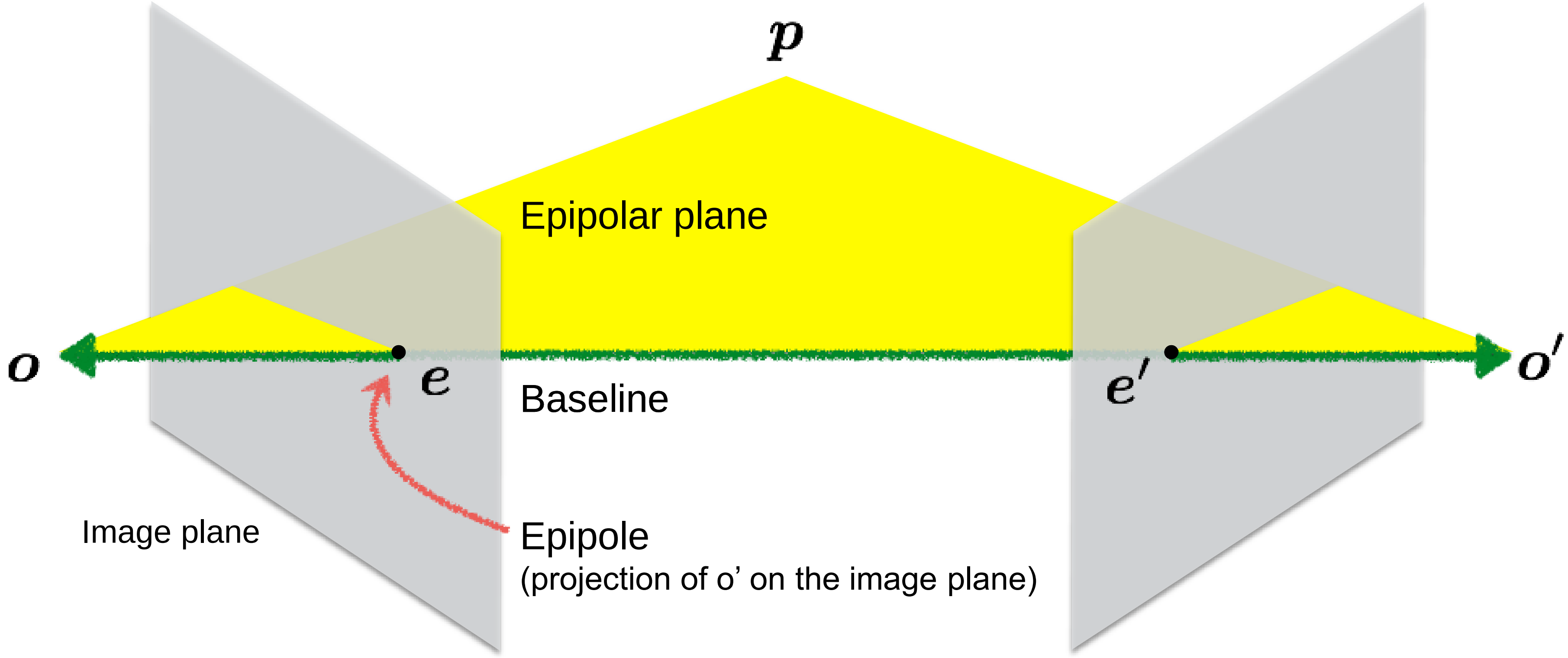
Epipolar geometry



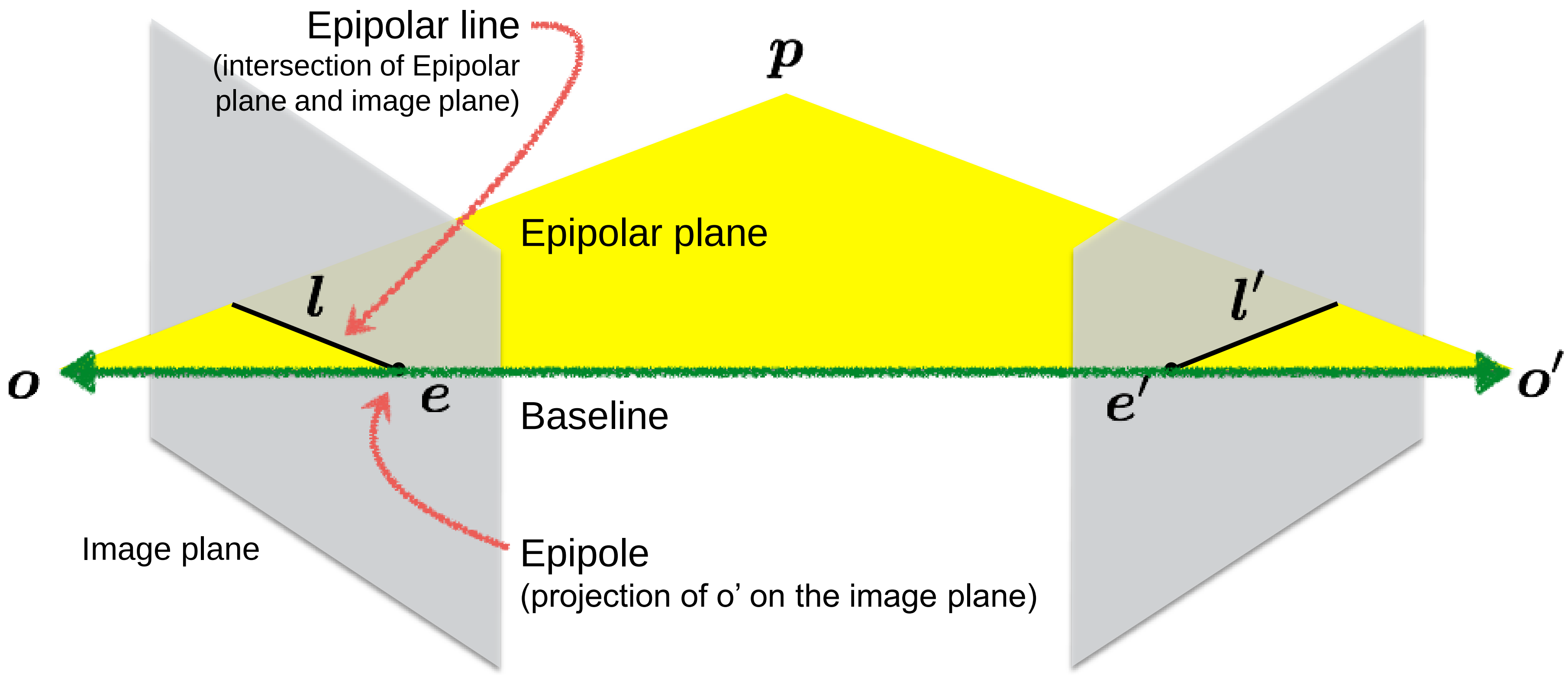
Epipolar geometry



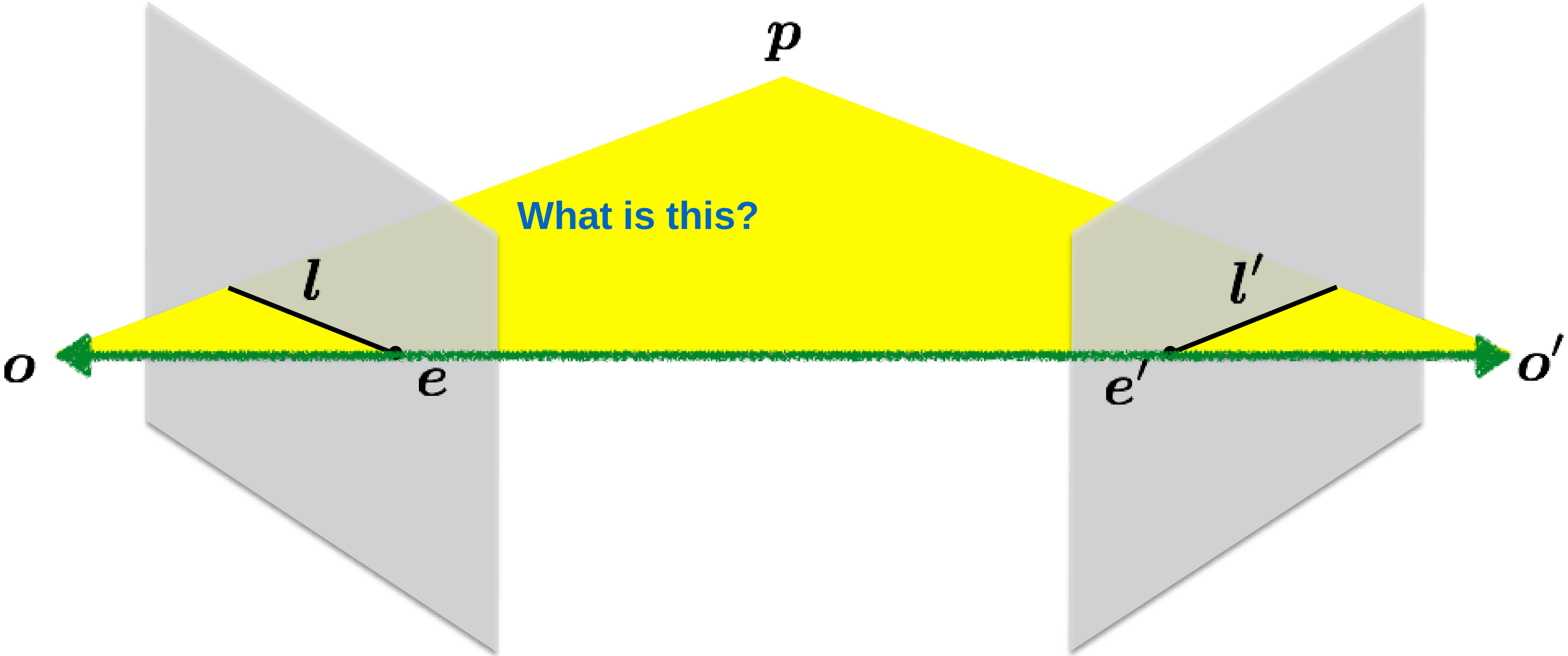
Epipolar geometry



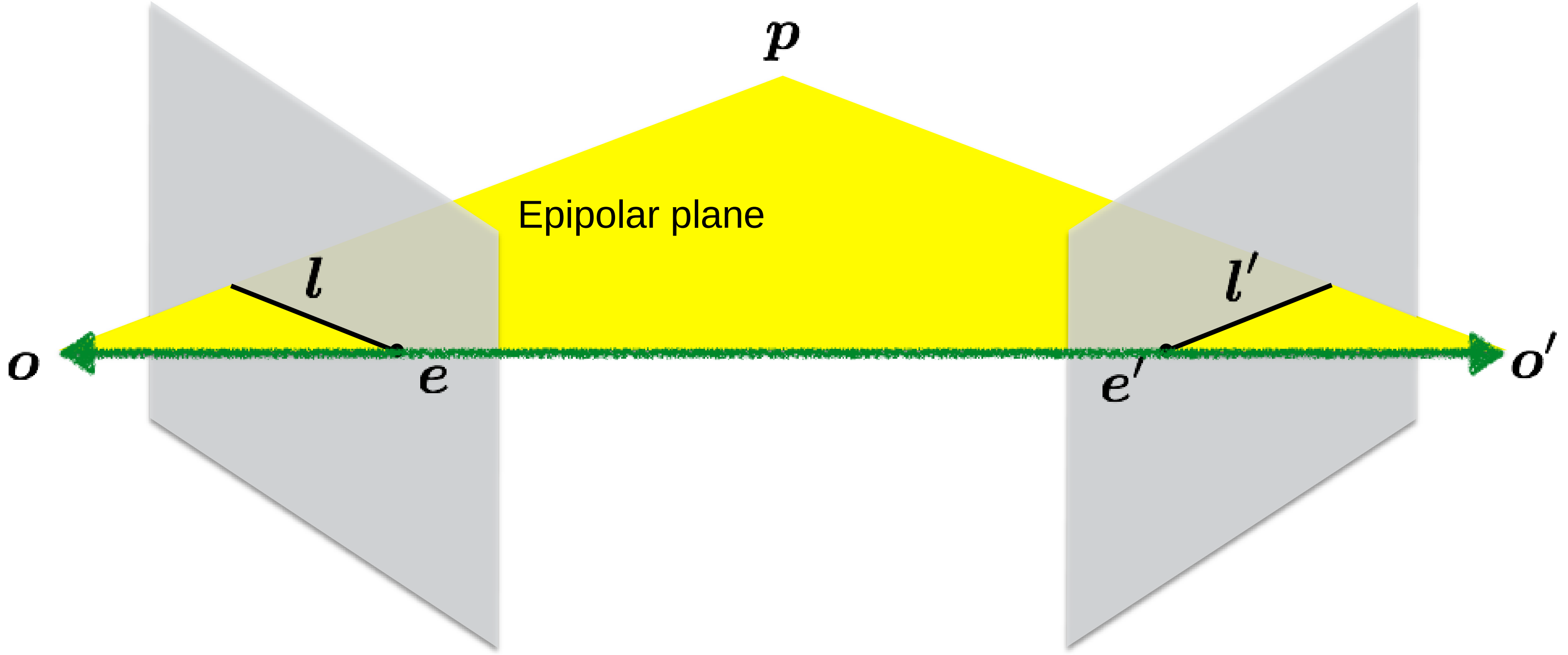
Epipolar geometry



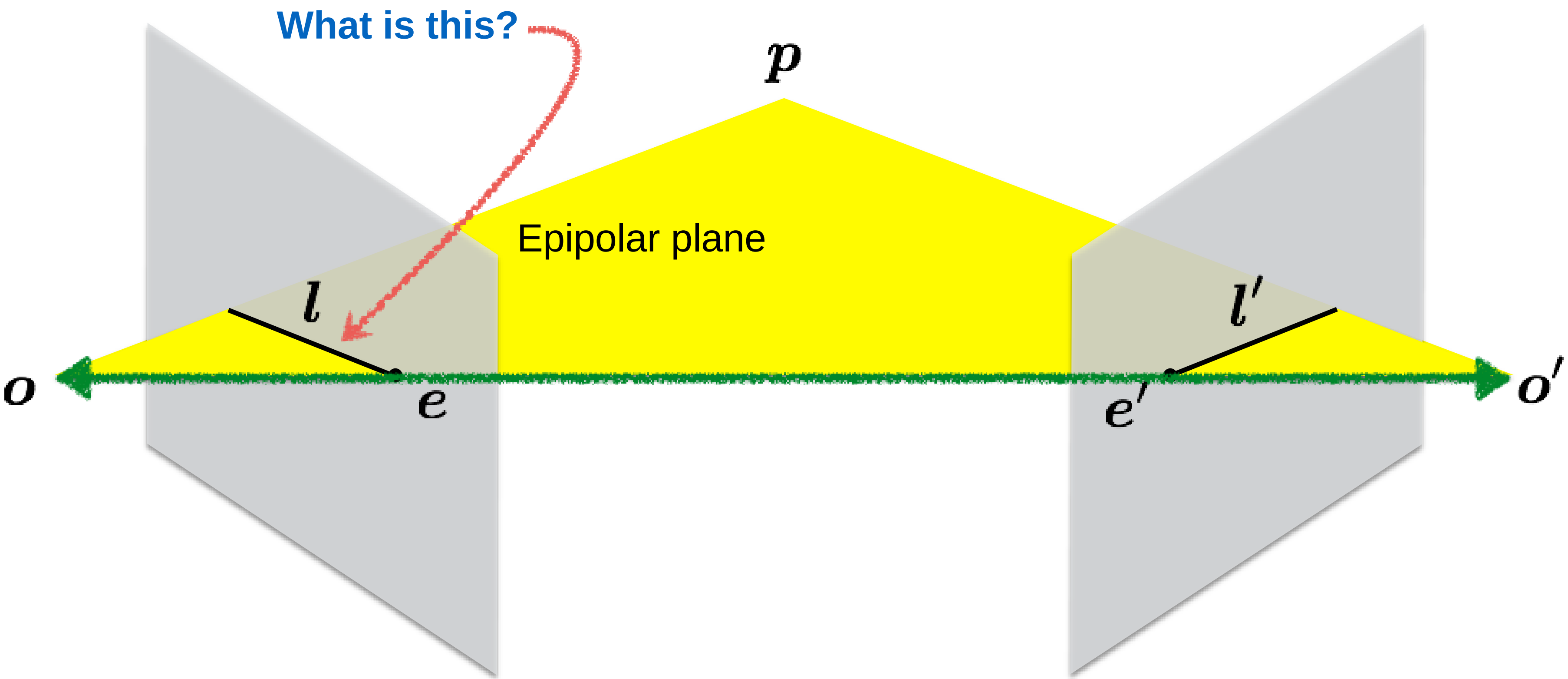
Quiz



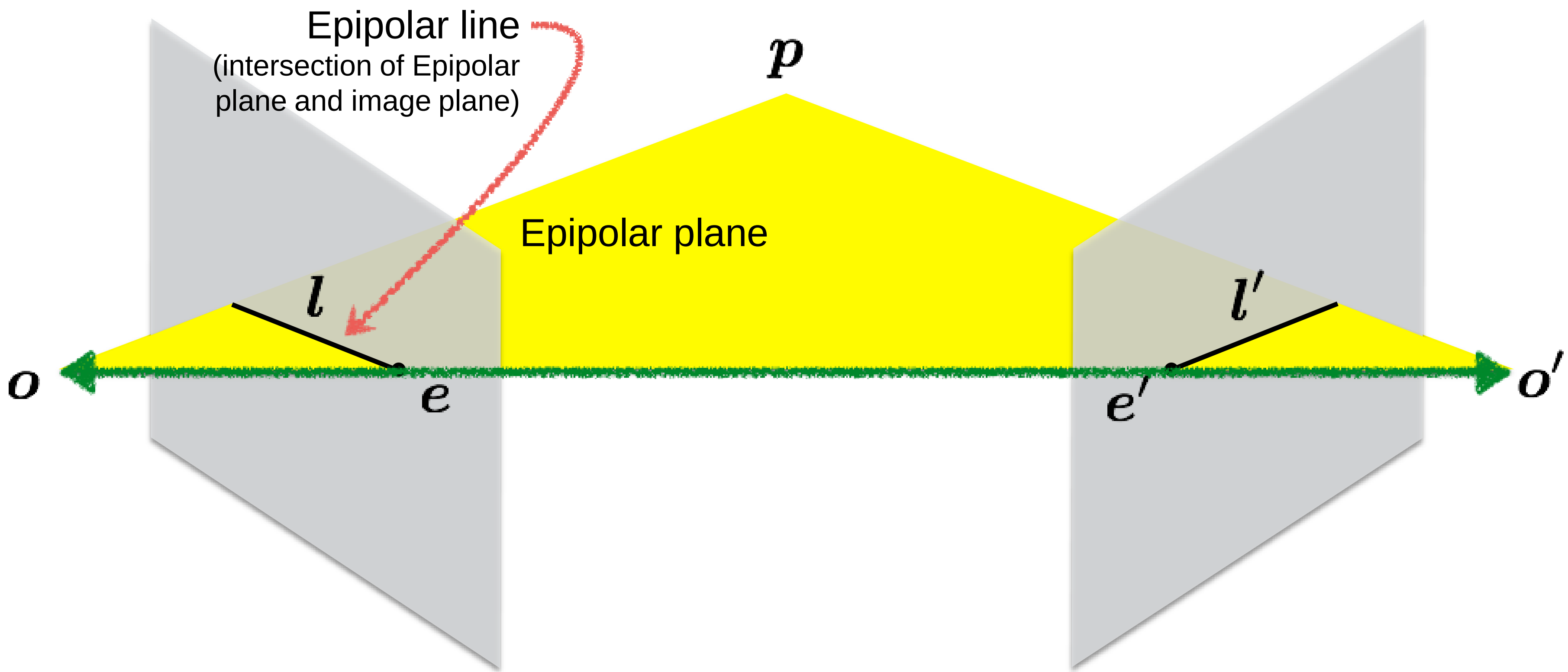
Quiz



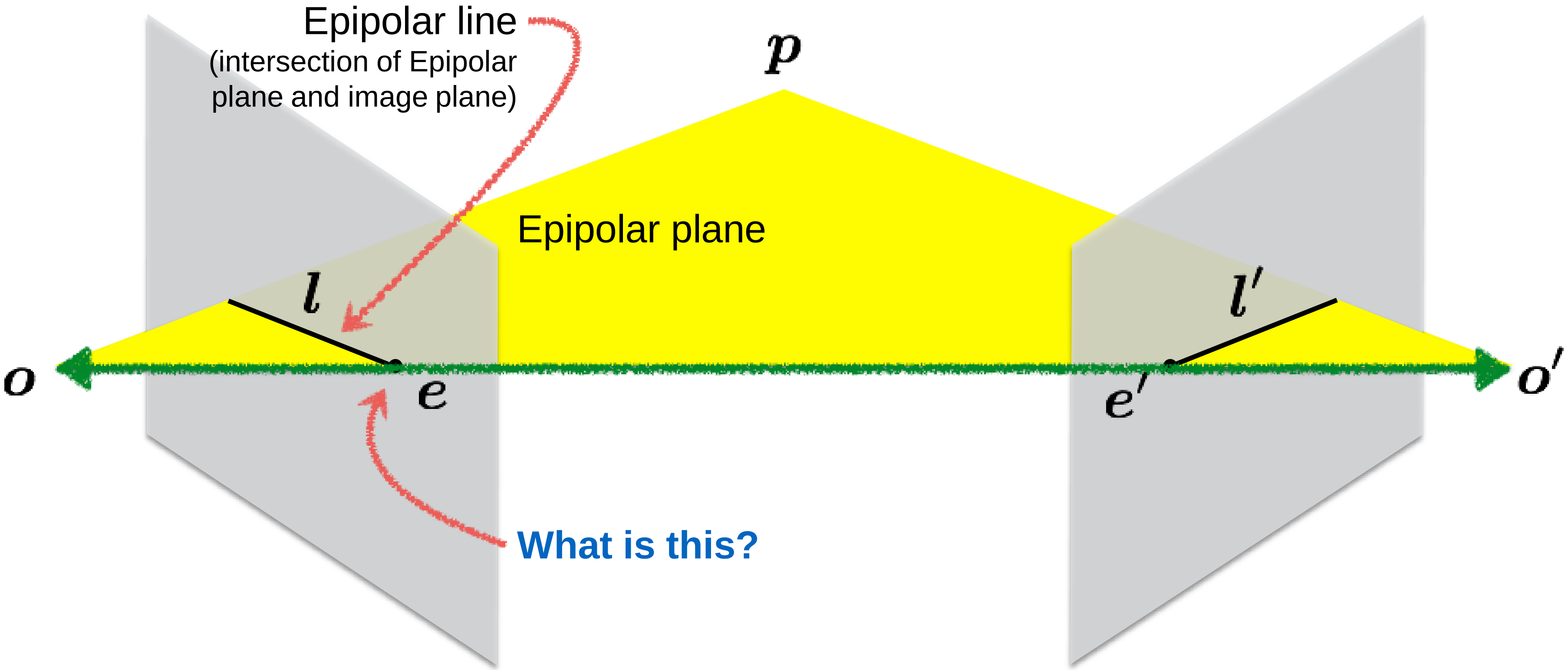
Quiz



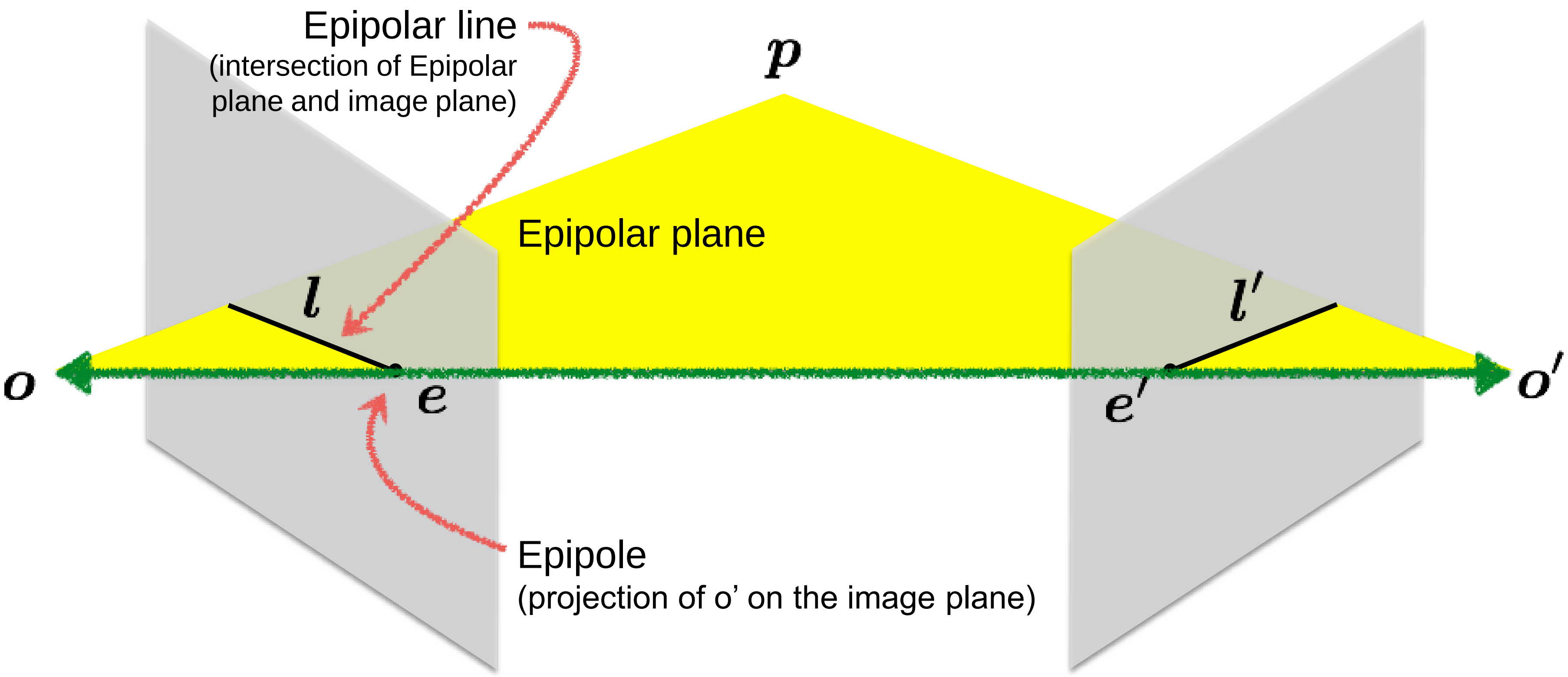
Quiz


Epipolar line
(intersection of Epipolar plane and image plane)
$\boldsymbol{p}$
Epipolar plane
$\boldsymbol{l}$
$\boldsymbol{l}'$
$\boldsymbol{o}$
$\boldsymbol{e}$
$\boldsymbol{e}'$
$\boldsymbol{o}'$

Quiz

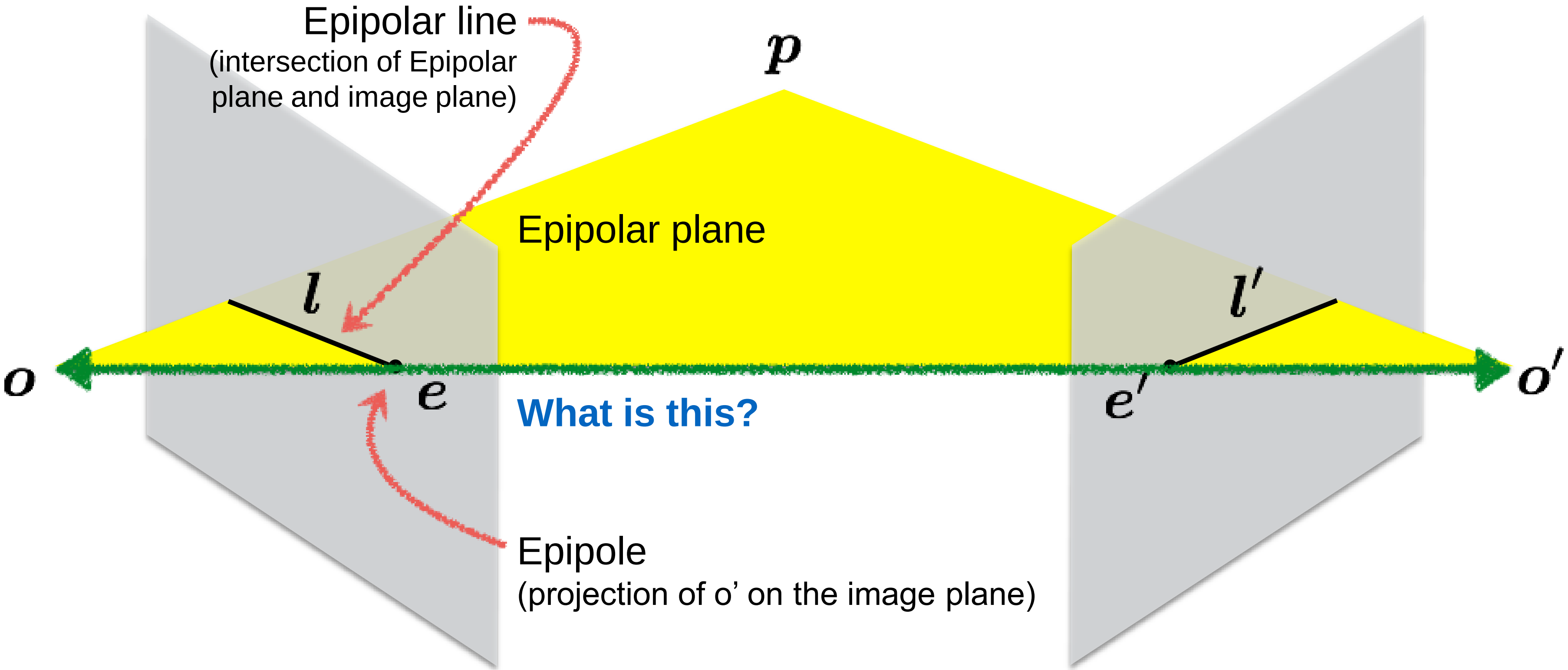

Epipolar line
(intersection of Epipolar
plane and image plane)
p
Epipolar plane
l
l'
o
e
e'
o'
What is this?

Quiz



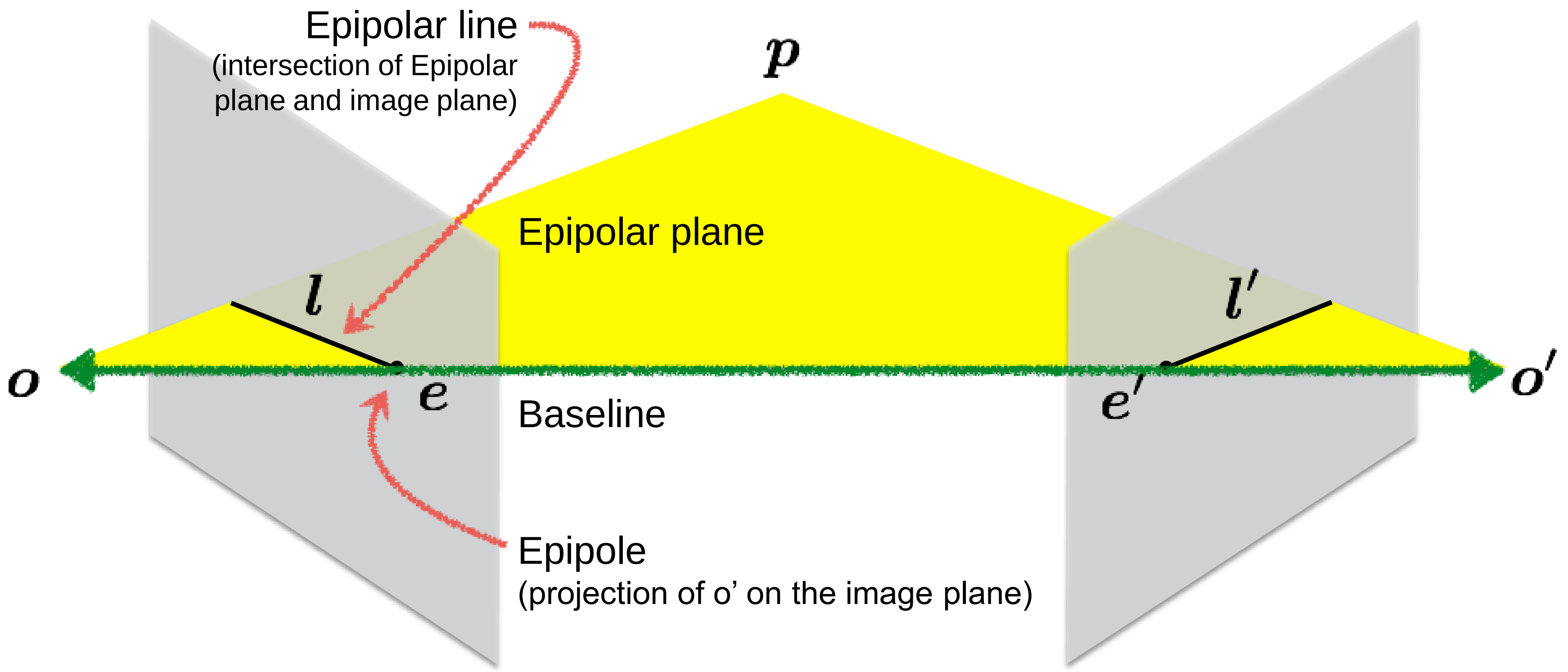
Epipolar line
(intersection of Epipolar plane and image plane)
p
Epipolar plane
l
l'
o
e
e'
o'
Epipole
(projection of o' on the image plane)

Quiz

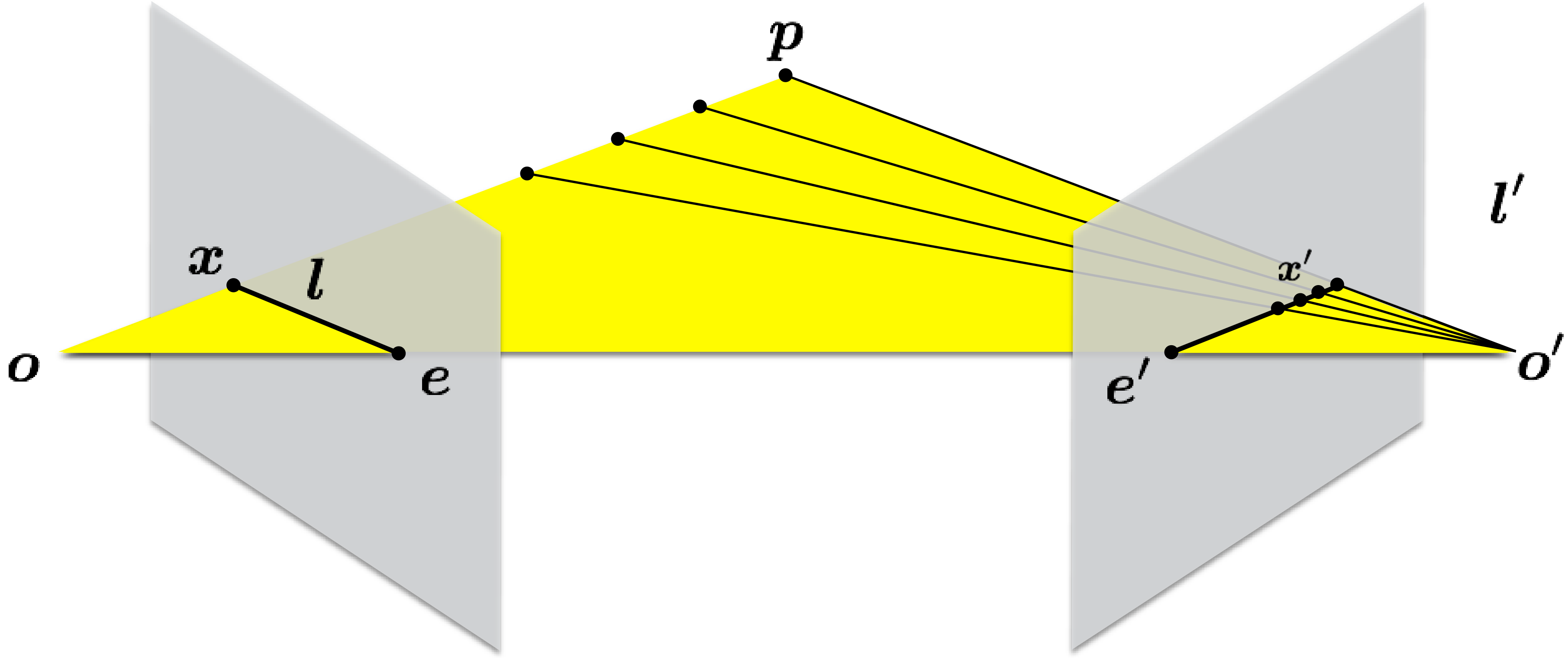


Epipolar line
(intersection of Epipolar plane and image plane)
p
Epipolar plane
l
l'
o
e
e'
o'
What is this?
Epipole
(projection of o' on the image plane)

Quiz

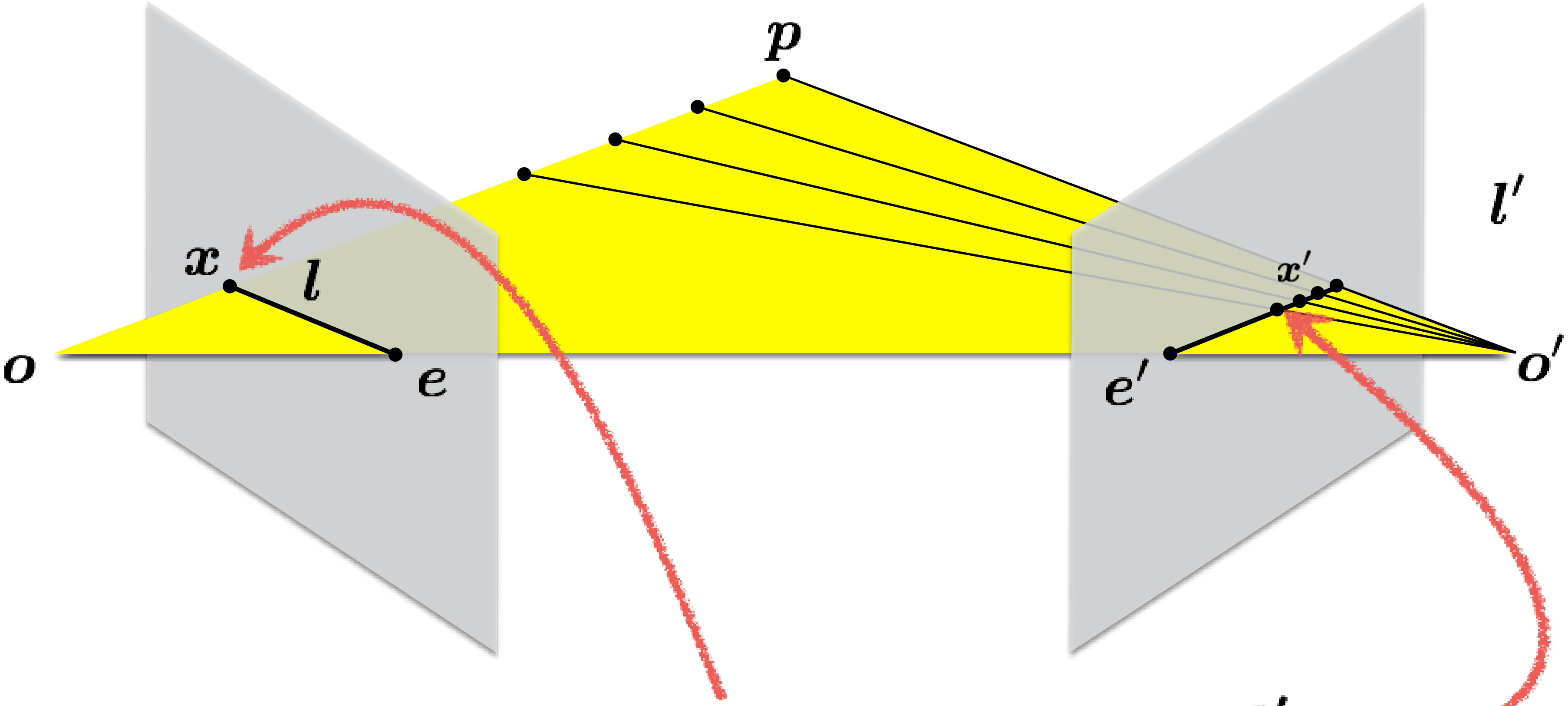

Epipolar line
(intersection of Epipolar plane and image plane)
p
Epipolar plane
l
l'
o
e
e'
o'
Baseline
Epipole
(projection of o’ on the image plane)

Epipolar constraint

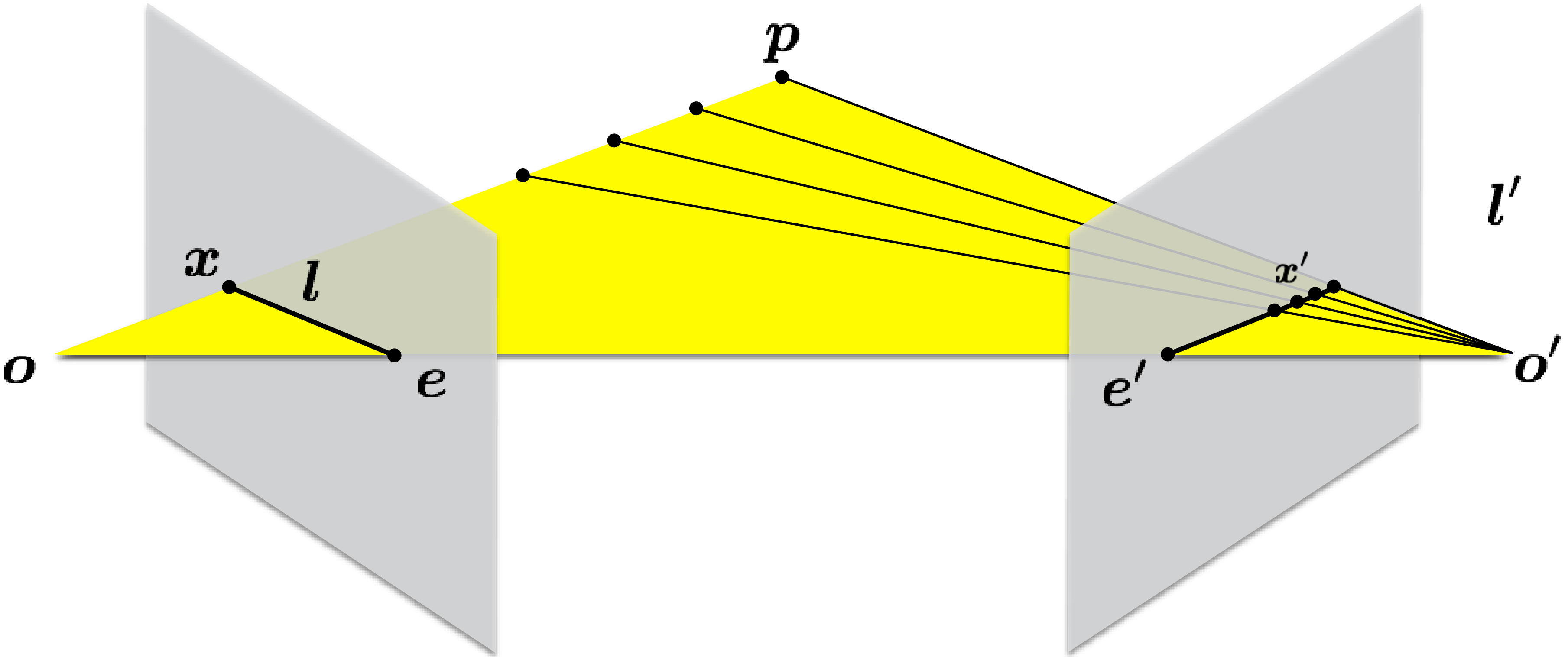


Potential matches for x lie on the epipolar line l'

Epipolar constraint



Potential matches for x lie on the epipolar line l'



The point **x** (left image) maps to a ____________ in the right image

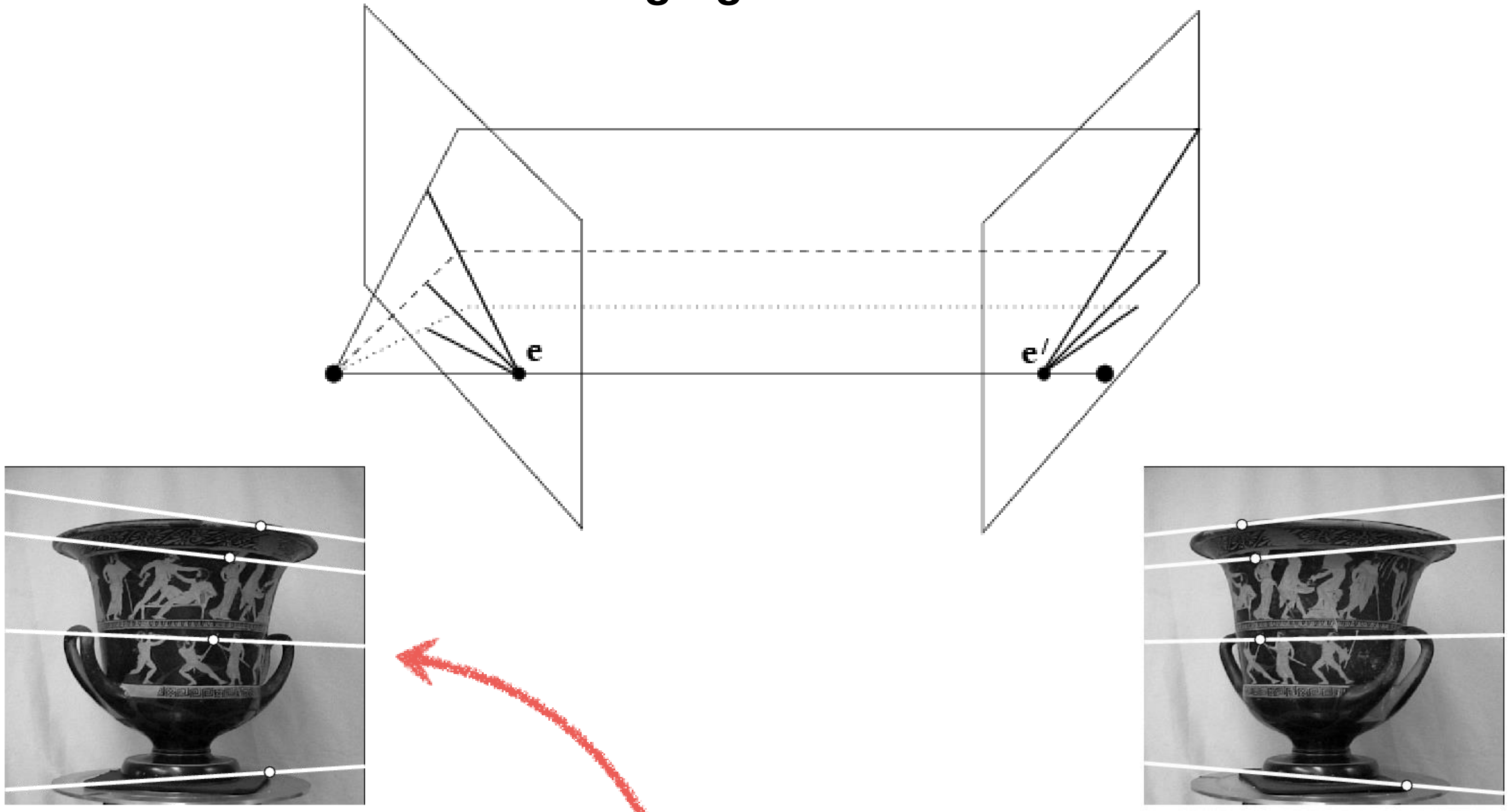
The baseline connects the ____________ and ____________

An epipolar line (left image) maps to a ___________ in the right image

An epipole **e** is a projection of the _______________ on the image plane

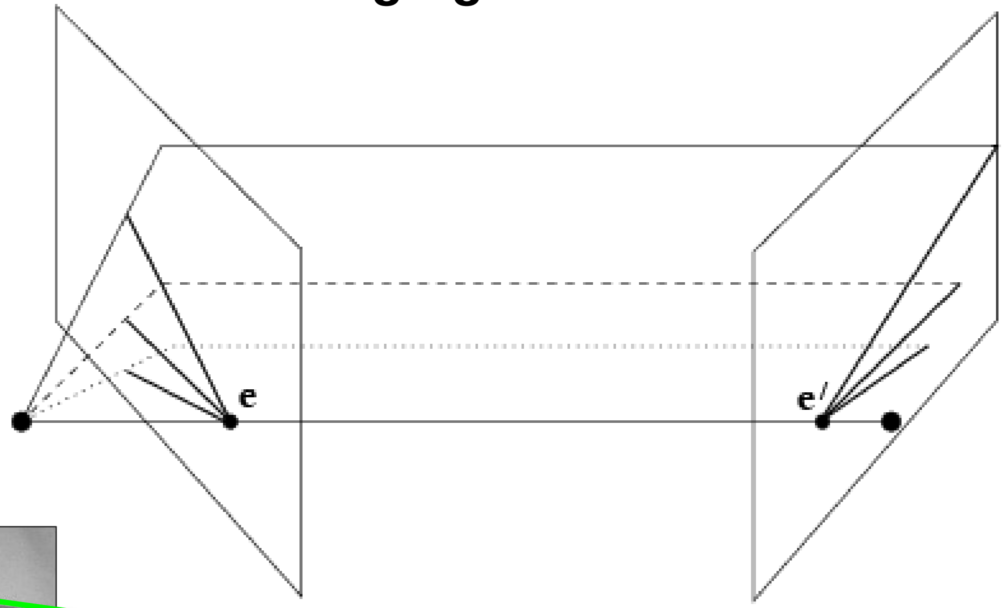
All epipolar lines in an image intersect at the _______________

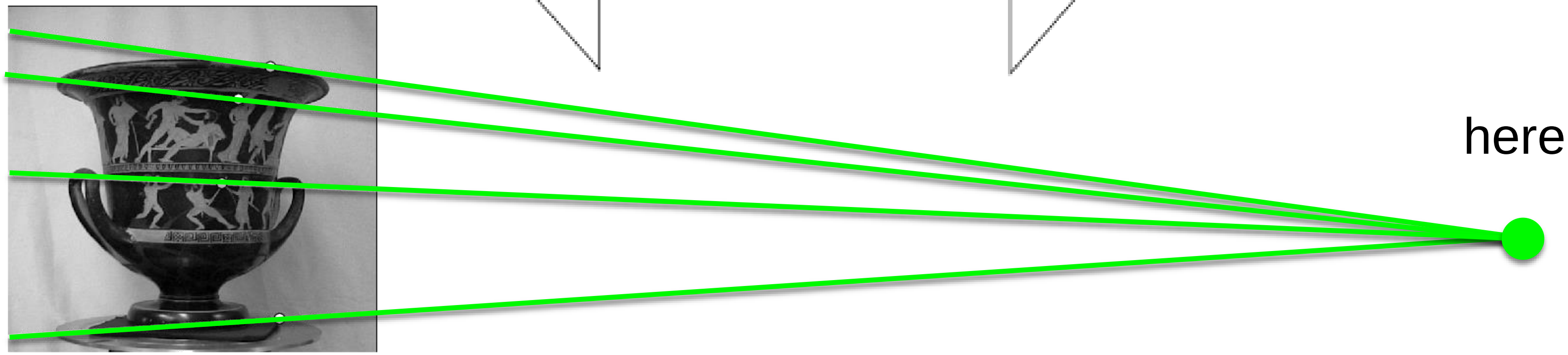
Converging cameras



Where is the epipole in this image?

Converging cameras



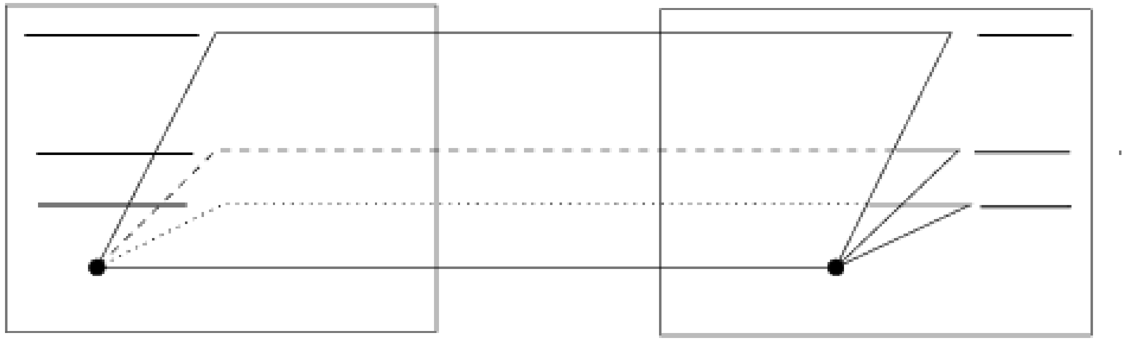


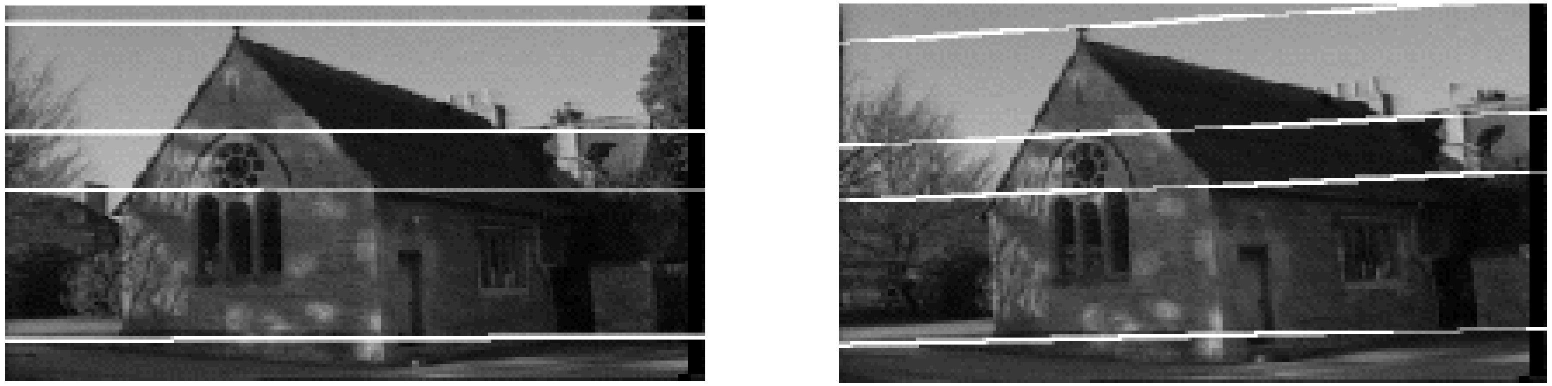
here!

Where is the epipole in this image?

It's not always in the image

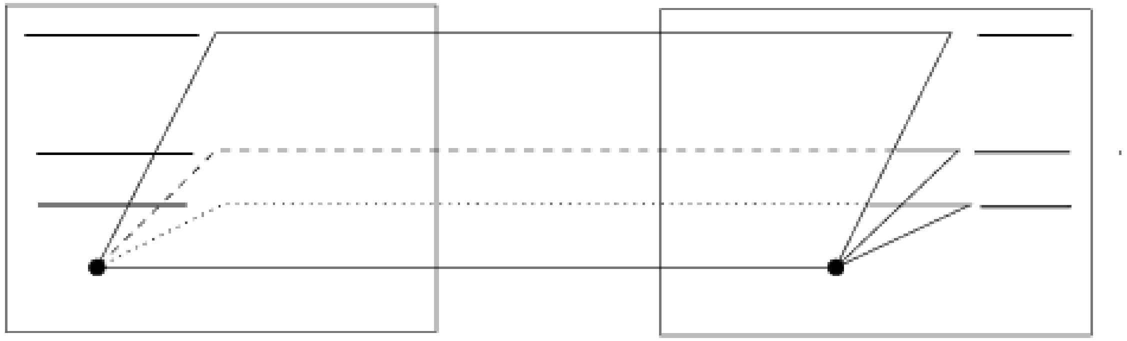
Parallel cameras

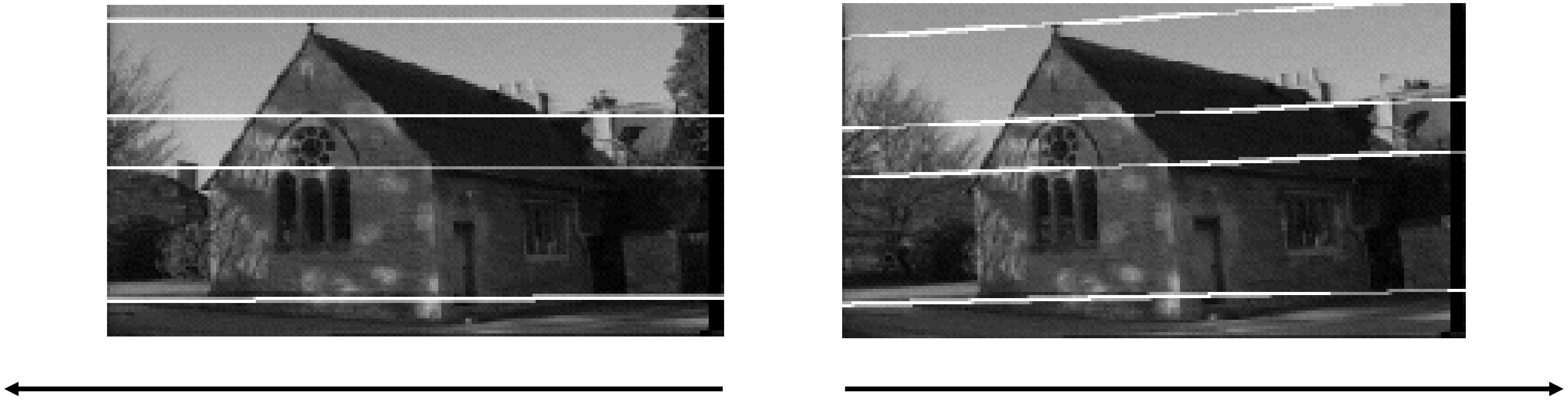




Where is the epipole?

Parallel cameras





epipole at infinity

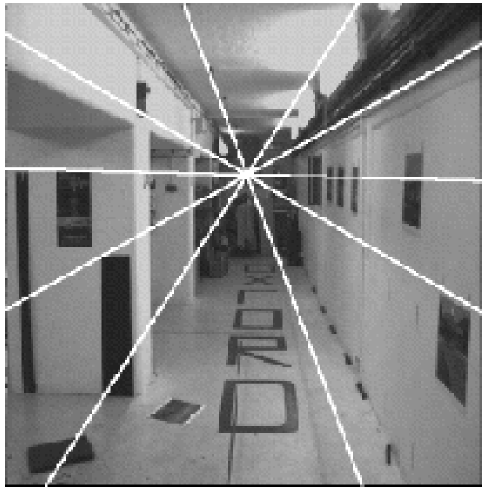
Forward-moving cameras

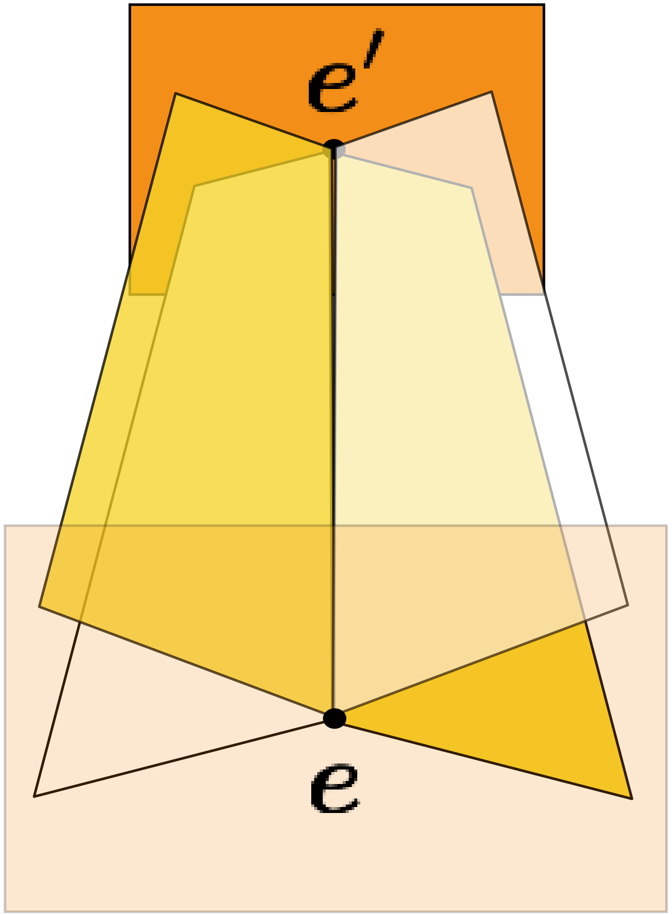
Forward-moving cameras

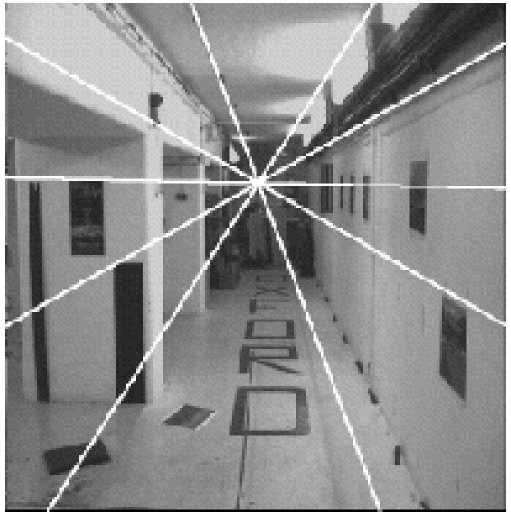
Where is the epipole?

What do the epipolar lines look like?

Epipole has same coordinates in both images.
Points move along lines radiating from "Focus of expansion"

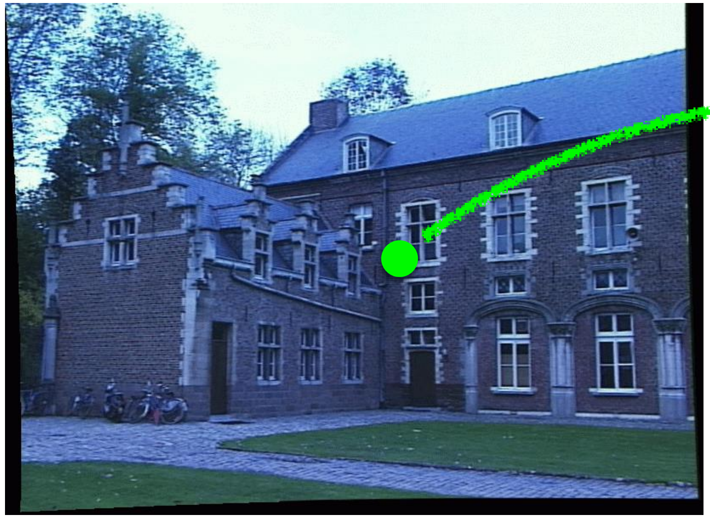




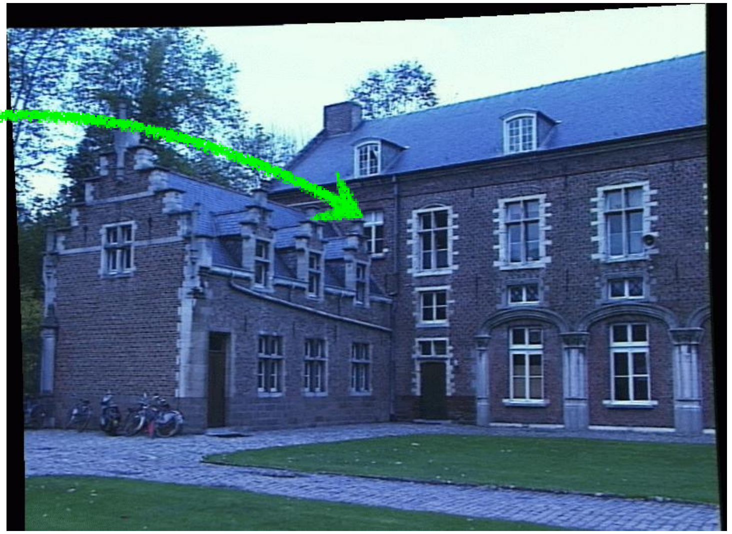


The epipolar constraint is an important concept for stereo vision

Task: Match point in left image to point in right image



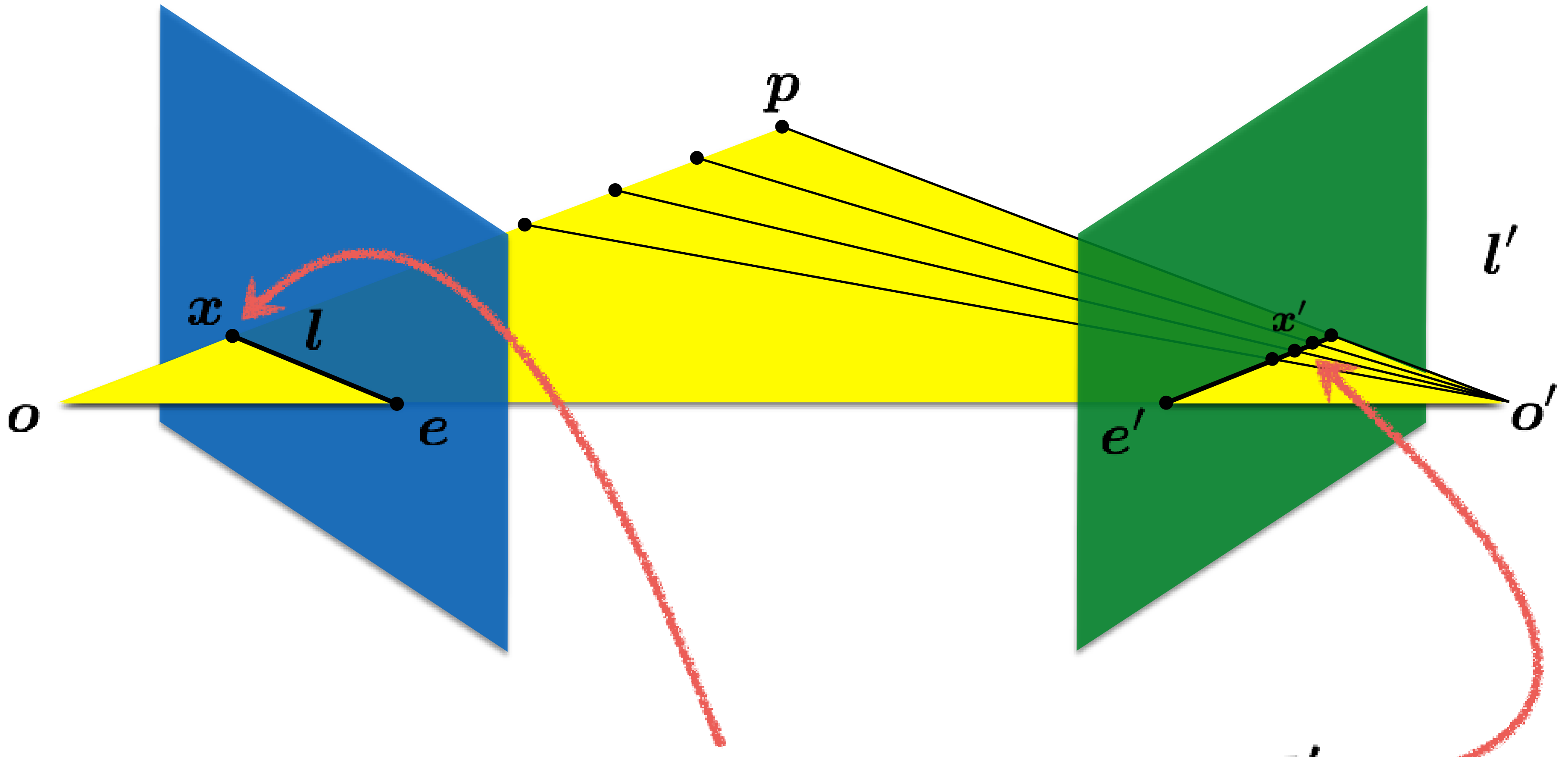
Left image



Right image

How would you do it?

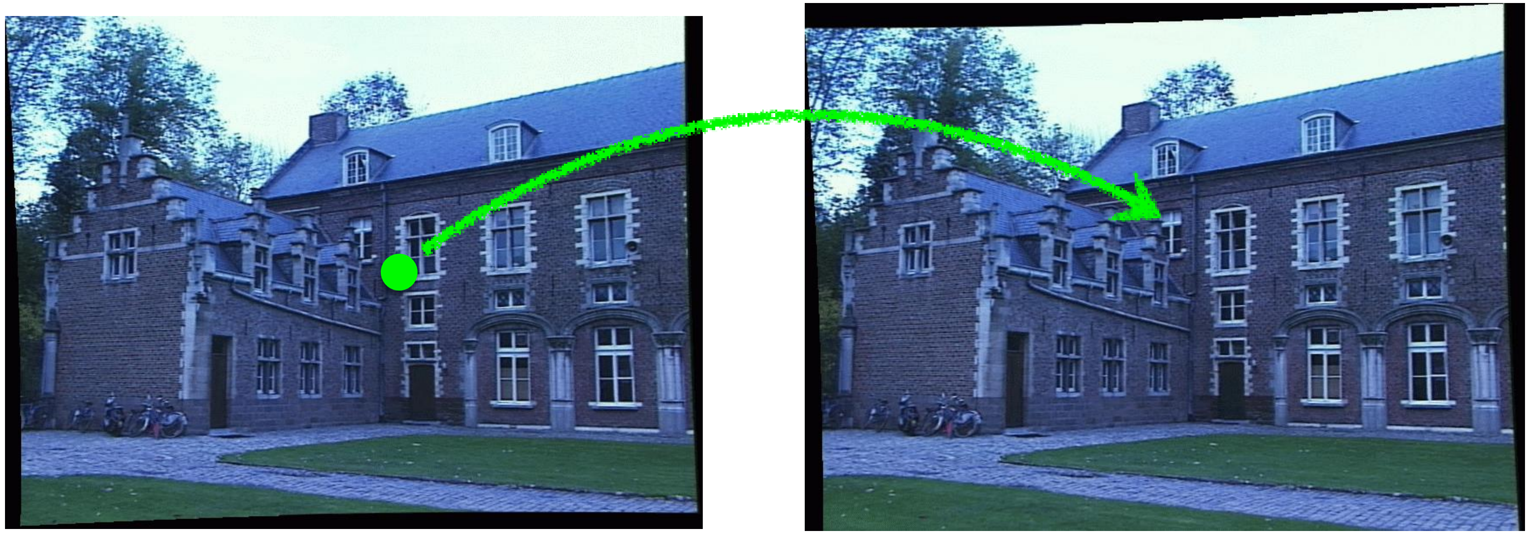
Recall:Epipolar constraint



Potential matches for x lie on the epipolar line l'

The epipolar constraint is an important concept for stereo vision

Task: Match point in left image to point in right image



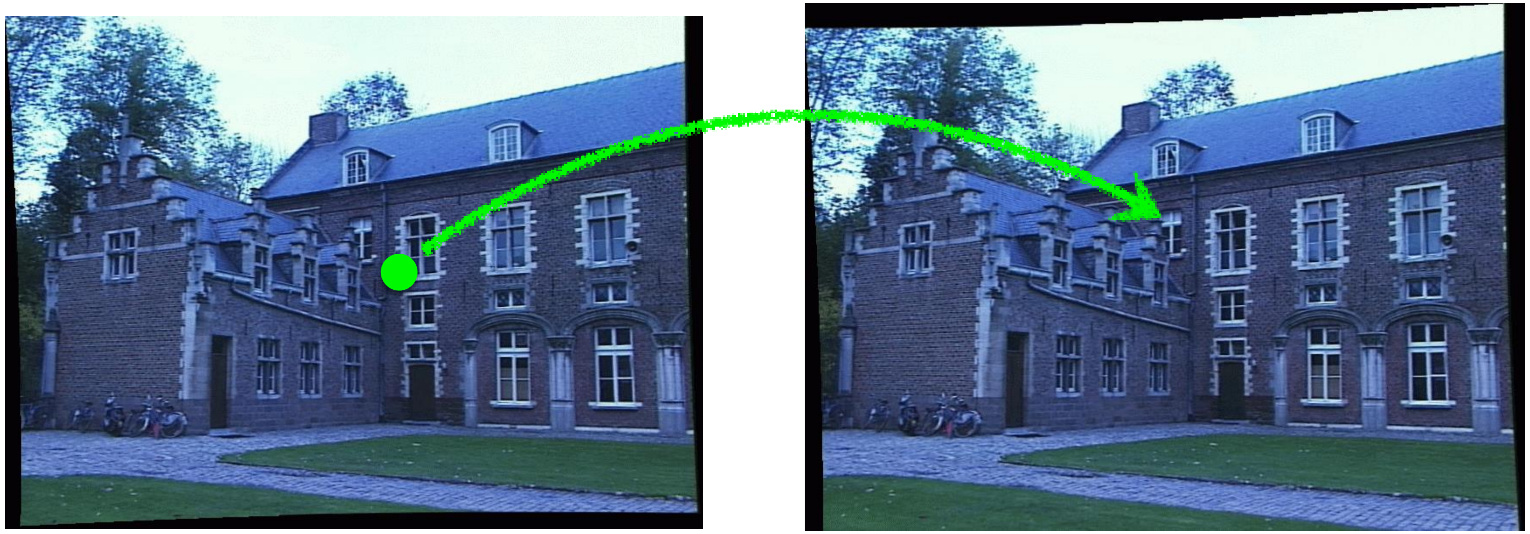
Left image   Right image

Want to avoid search over entire image

Epipolar constraint reduces search to a single line

The epipolar constraint is an important concept for stereo vision

Task: Match point in left image to point in right image



Left image

Right image

Want to avoid search over entire image

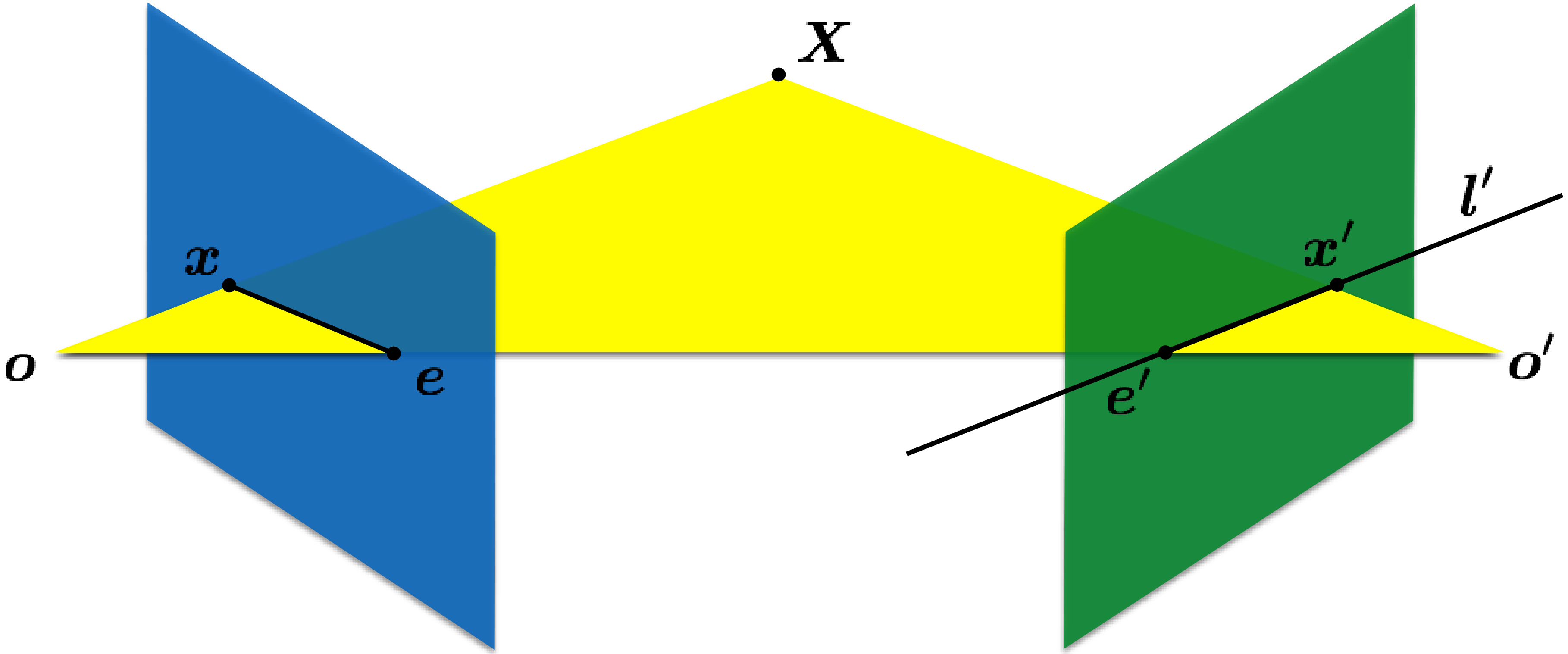
Epipolar constraint reduces search to a single line

How do you compute the epipolar line?

The essential matrix

Given a point in one image,
multiplying by the **essential matrix** will tell us
the **epipolar line** in the second view.

$$\boxed{\mathbf{E}\boldsymbol{x} = \boldsymbol{l}'}$$



Motivation

The Essential Matrix is a 3 x 3 matrix that encodes **epipolar geometry**

Given a point in one image, multiplying by the **essential matrix** will tell us the **epipolar line** in the second view.

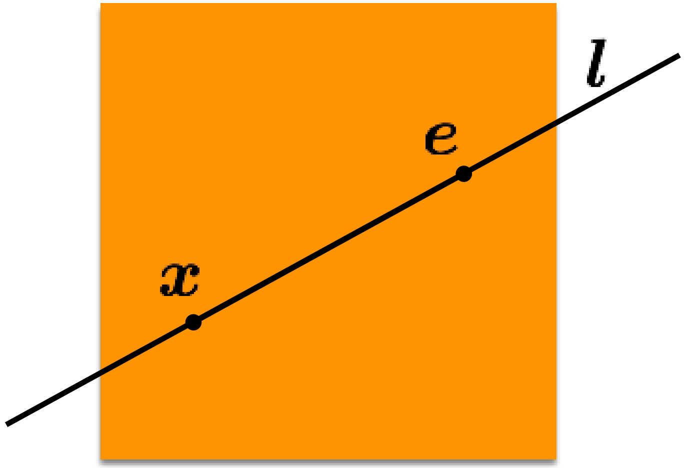
Representing the …

Epipolar Line

$$ax + by + c = 0$$

in vector form

$$\boldsymbol{l} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$



If the point $\boldsymbol{x}$ is on the epipolar line $\boldsymbol{l}$ then

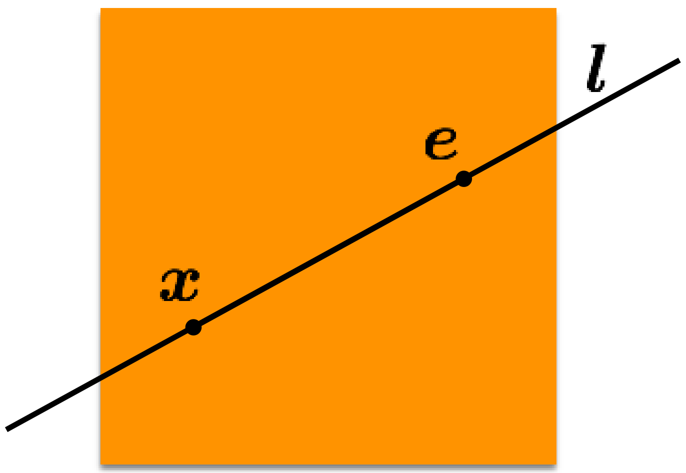
$$\boldsymbol{x}^\top \boldsymbol{l} = \ ?$$

Epipolar Line

$$ax + by + c = 0$$

in vector form

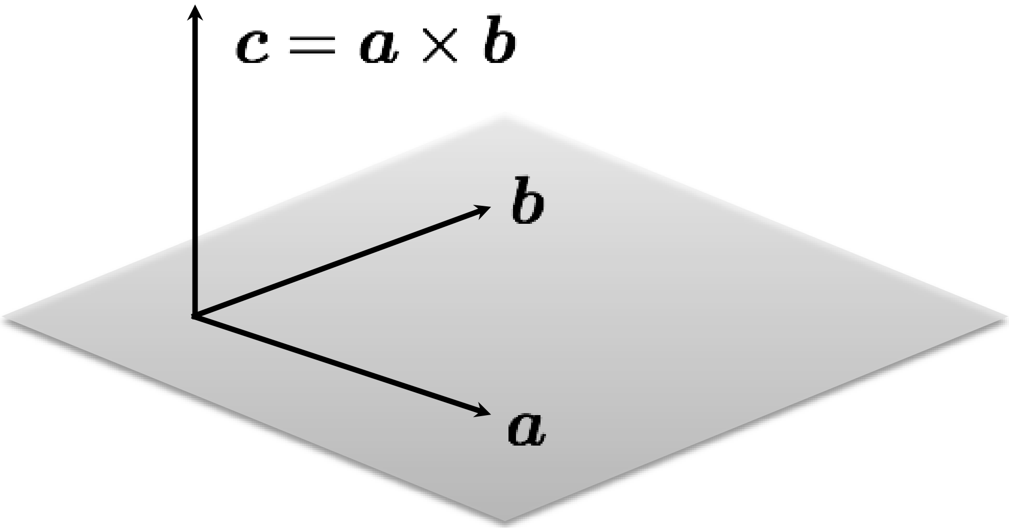
$$\boldsymbol{l} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$



If the point $\boldsymbol{x}$ is on the epipolar line $\boldsymbol{l}$ then

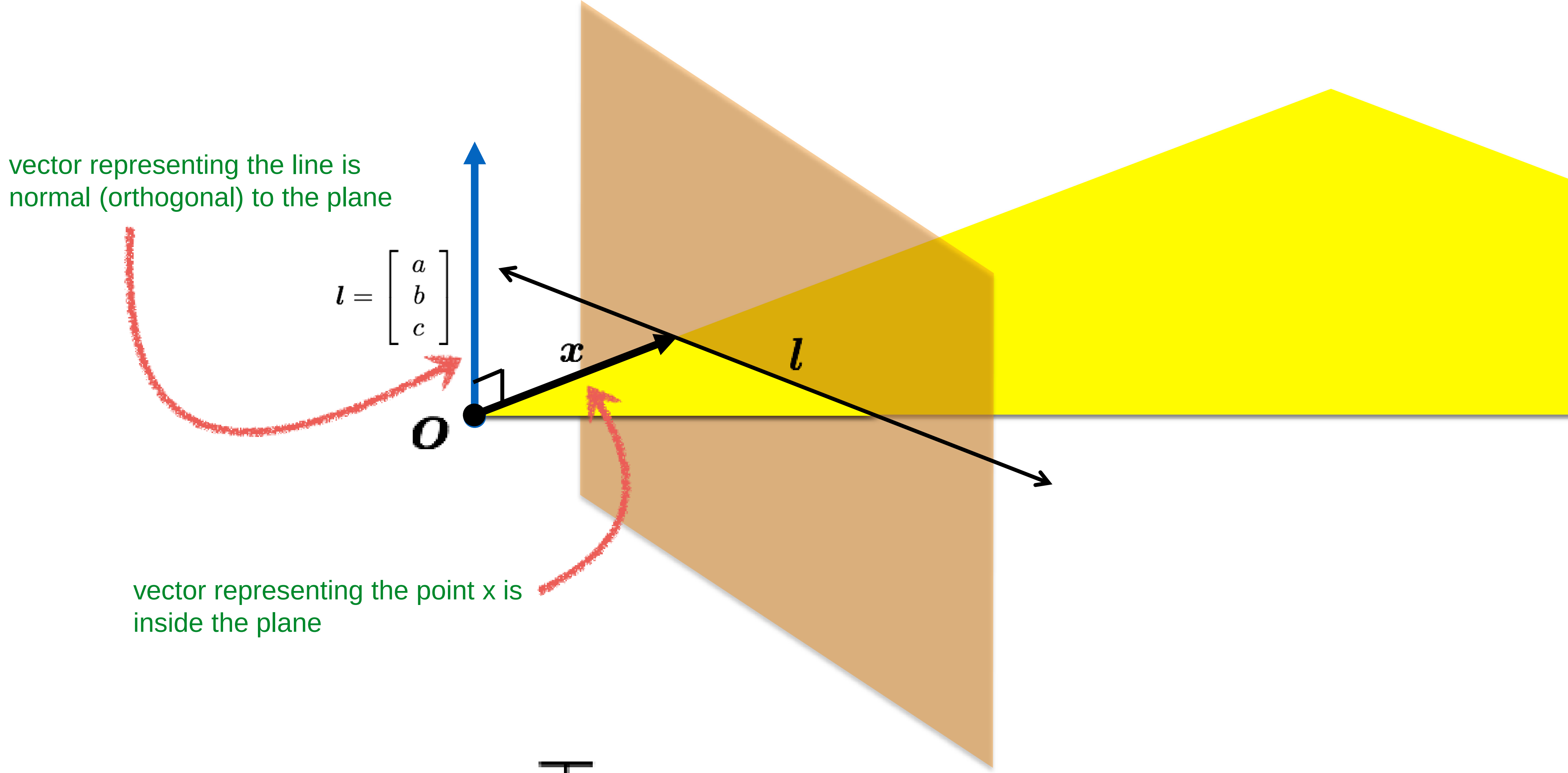
$$\boldsymbol{x}^\top \boldsymbol{l} = 0$$

Recall: Dot Product



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

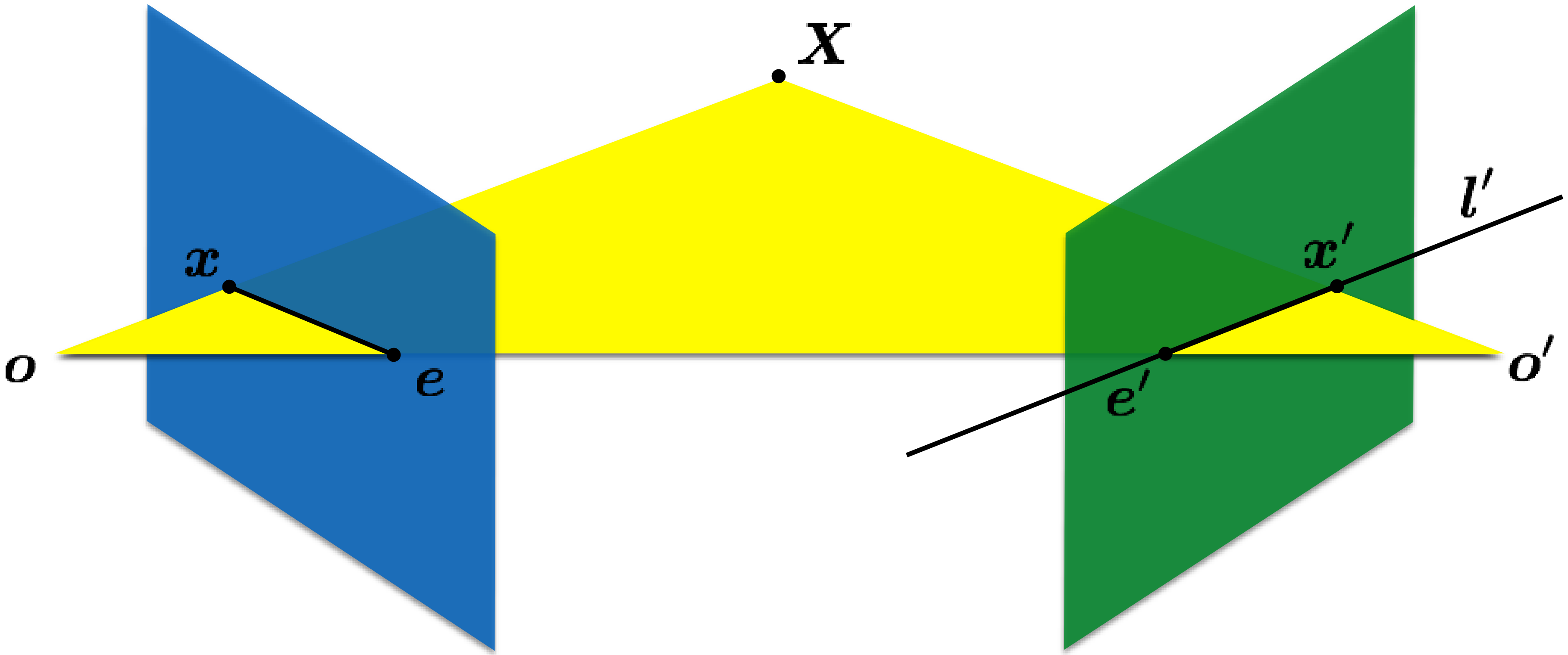
dot product of two orthogonal vectors is zero



Therefore: $\boldsymbol{x}^\top \boldsymbol{l} = 0$

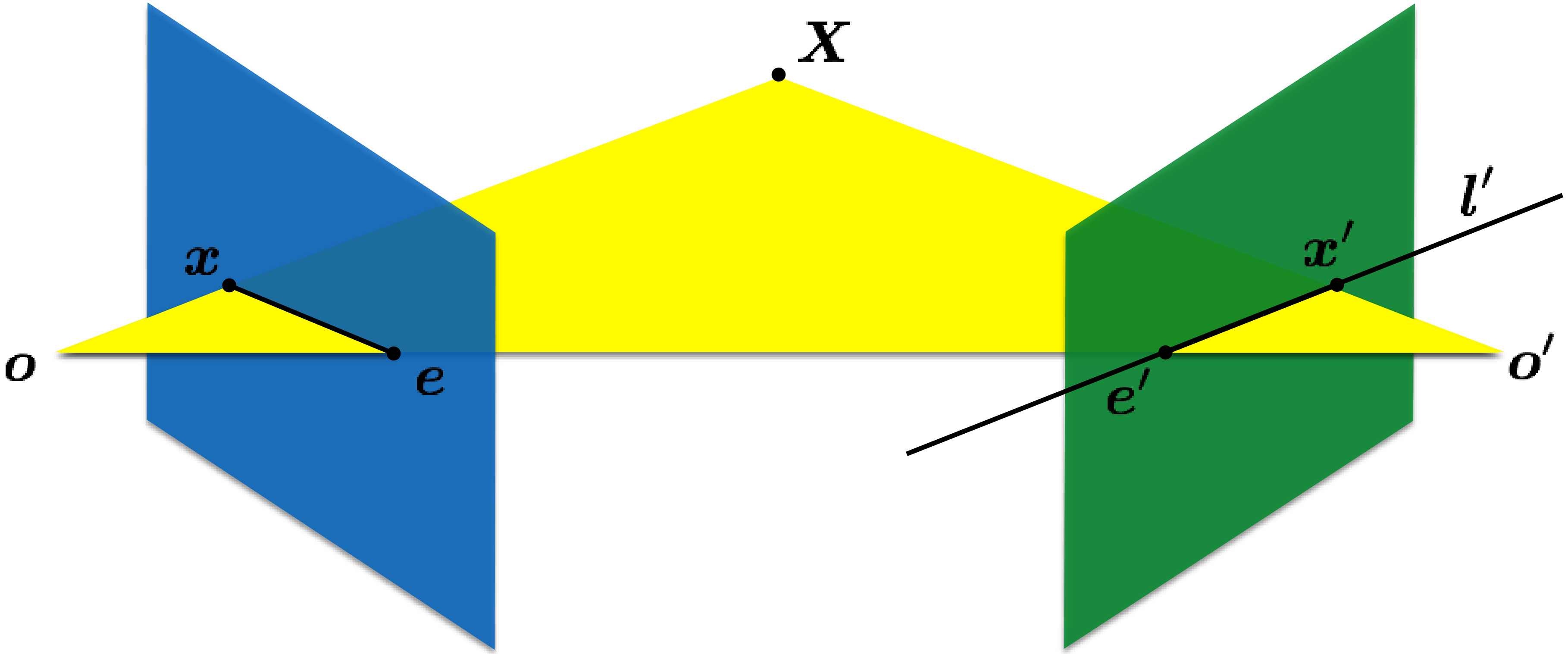
So if $\boldsymbol{x}^\top \boldsymbol{l} = 0$ and $\mathbf{E}\boldsymbol{x} = \boldsymbol{l}'$ then

$$\boldsymbol{x}'^\top \mathbf{E}\boldsymbol{x} = \ ?$$



So if $\boldsymbol{x}^\top \boldsymbol{l} = 0$ and $\mathbf{E}\boldsymbol{x} = \boldsymbol{l}'$ then

$$\boxed{\boldsymbol{x}'^\top \mathbf{E}\boldsymbol{x} = 0}$$



Essential Matrix vs Homography

What's the difference between the essential matrix and a homography?

Essential Matrix vs Homography

What's the difference between the essential matrix and a homography?

They are both 3 x 3 matrices but …

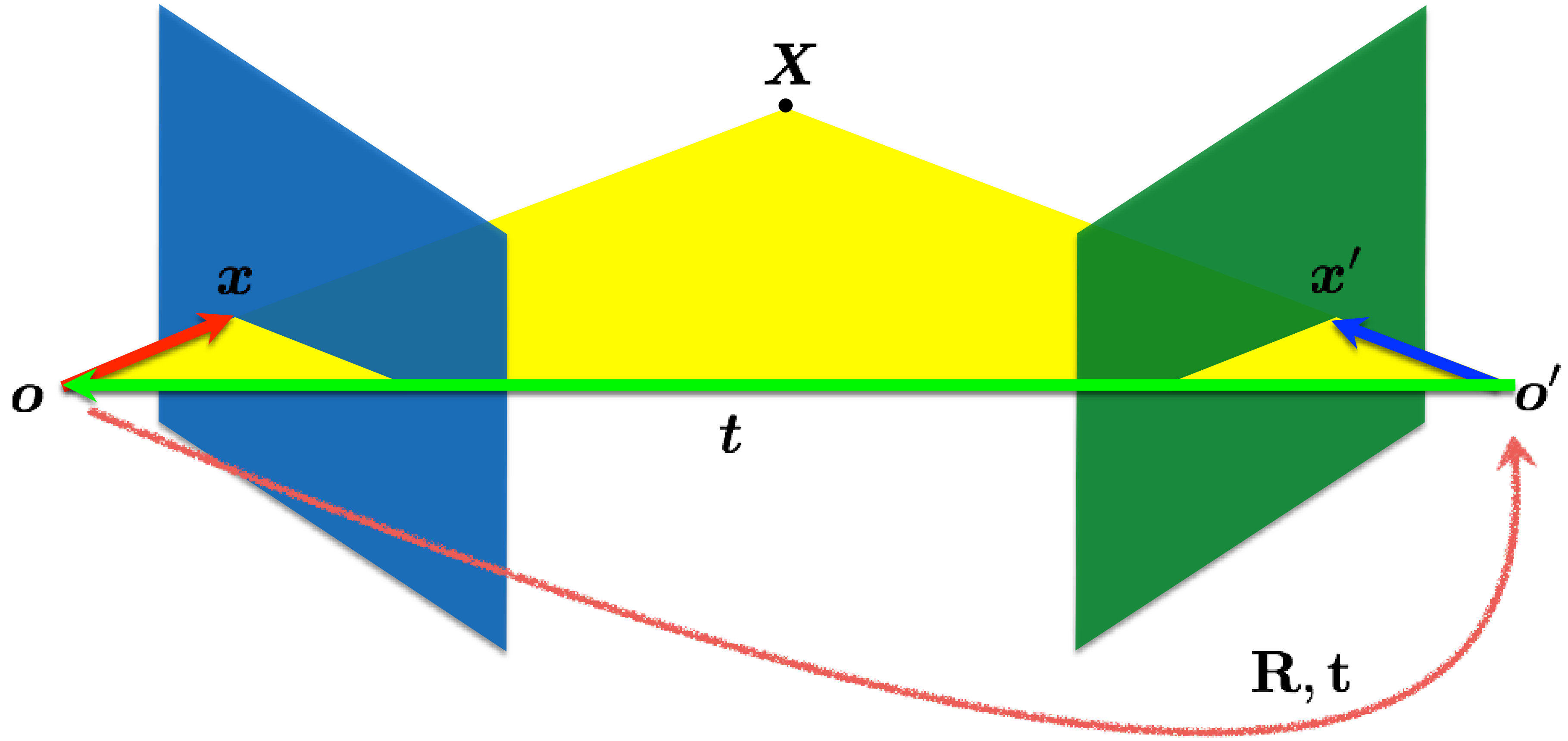
$$\boldsymbol{l}' = \mathbf{E}\boldsymbol{x}$$

Essential matrix maps a **point** to a **line**

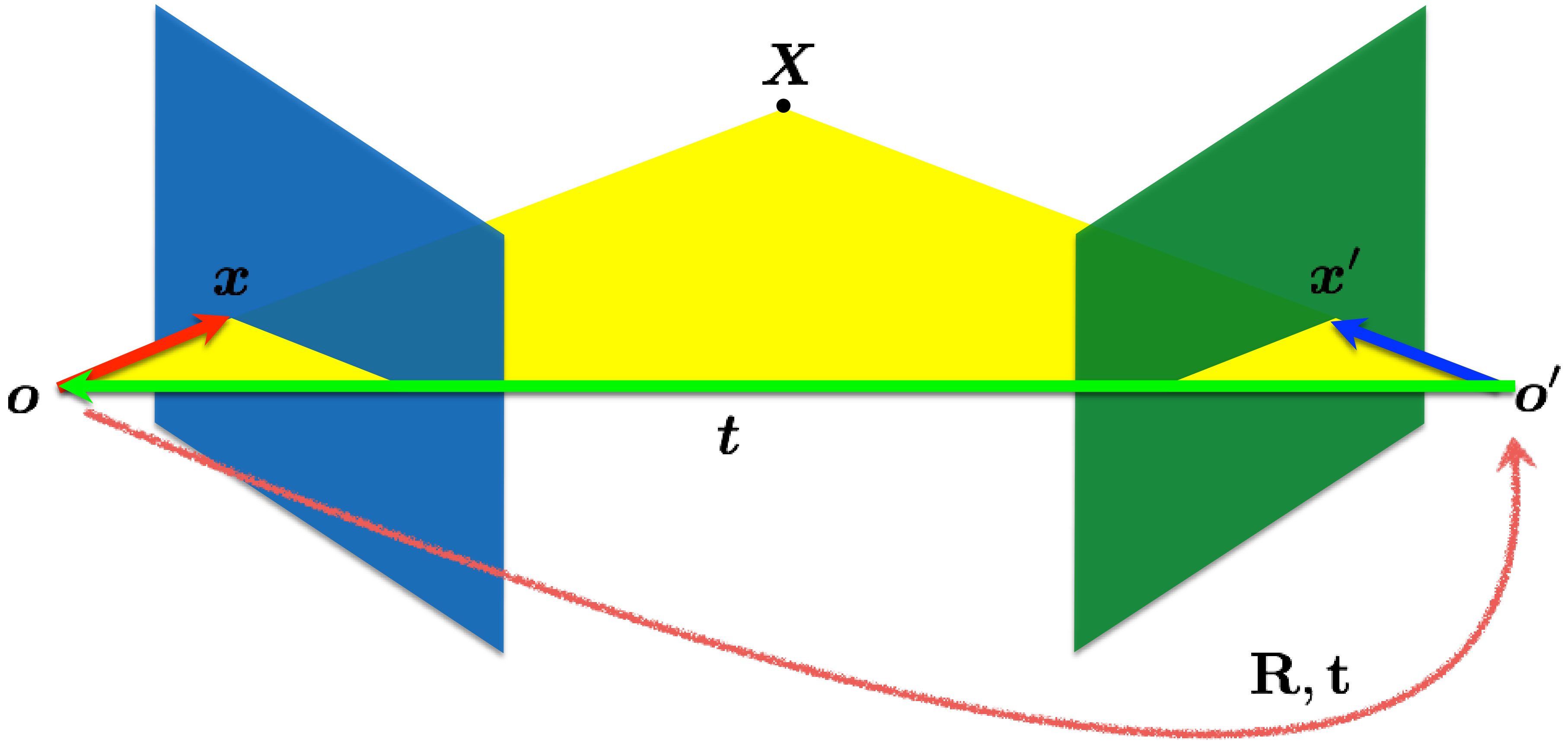
$$\boldsymbol{x}' = \mathbf{H}\boldsymbol{x}$$

Homography maps a **point** to a **point**

Where does the Essential matrix come from?

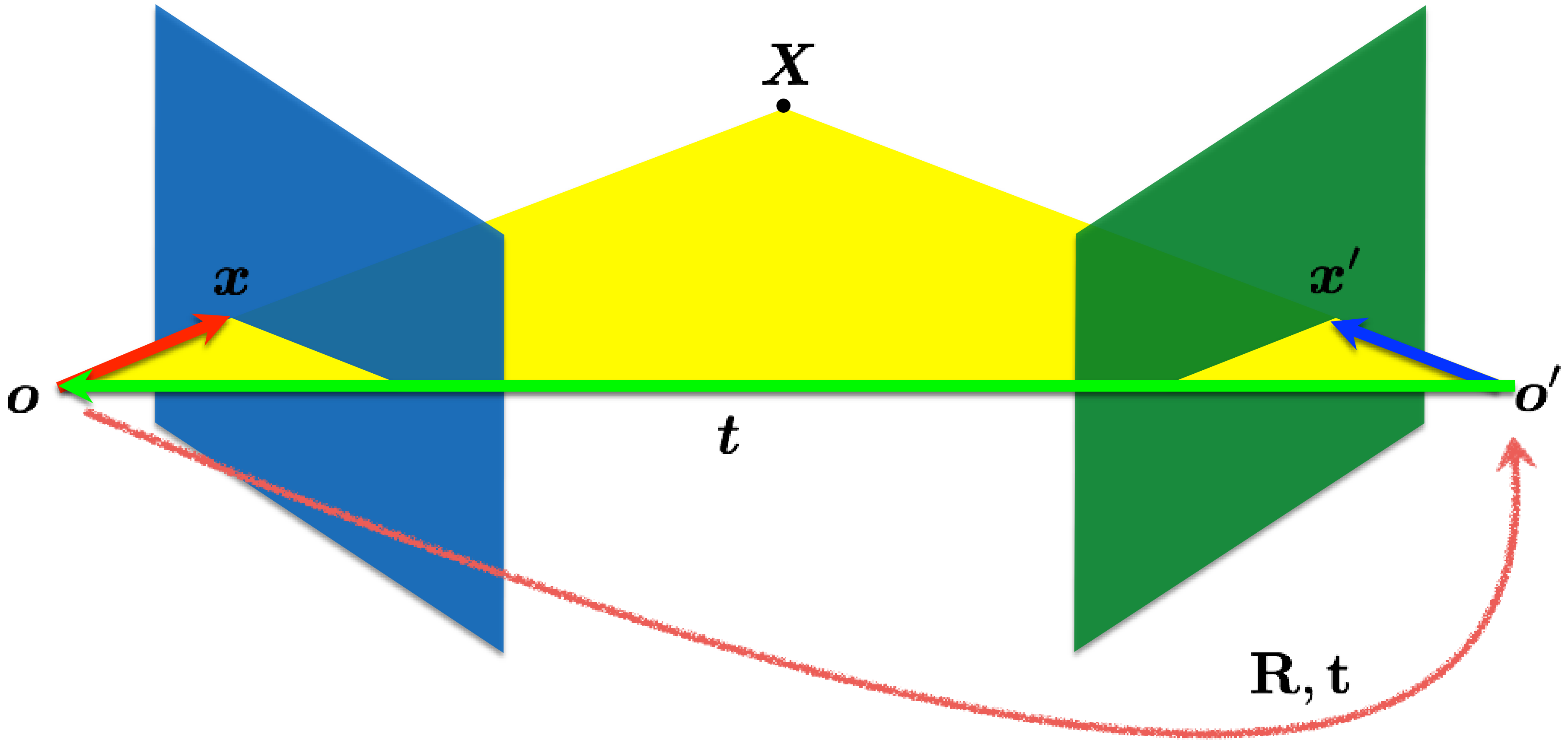


$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$



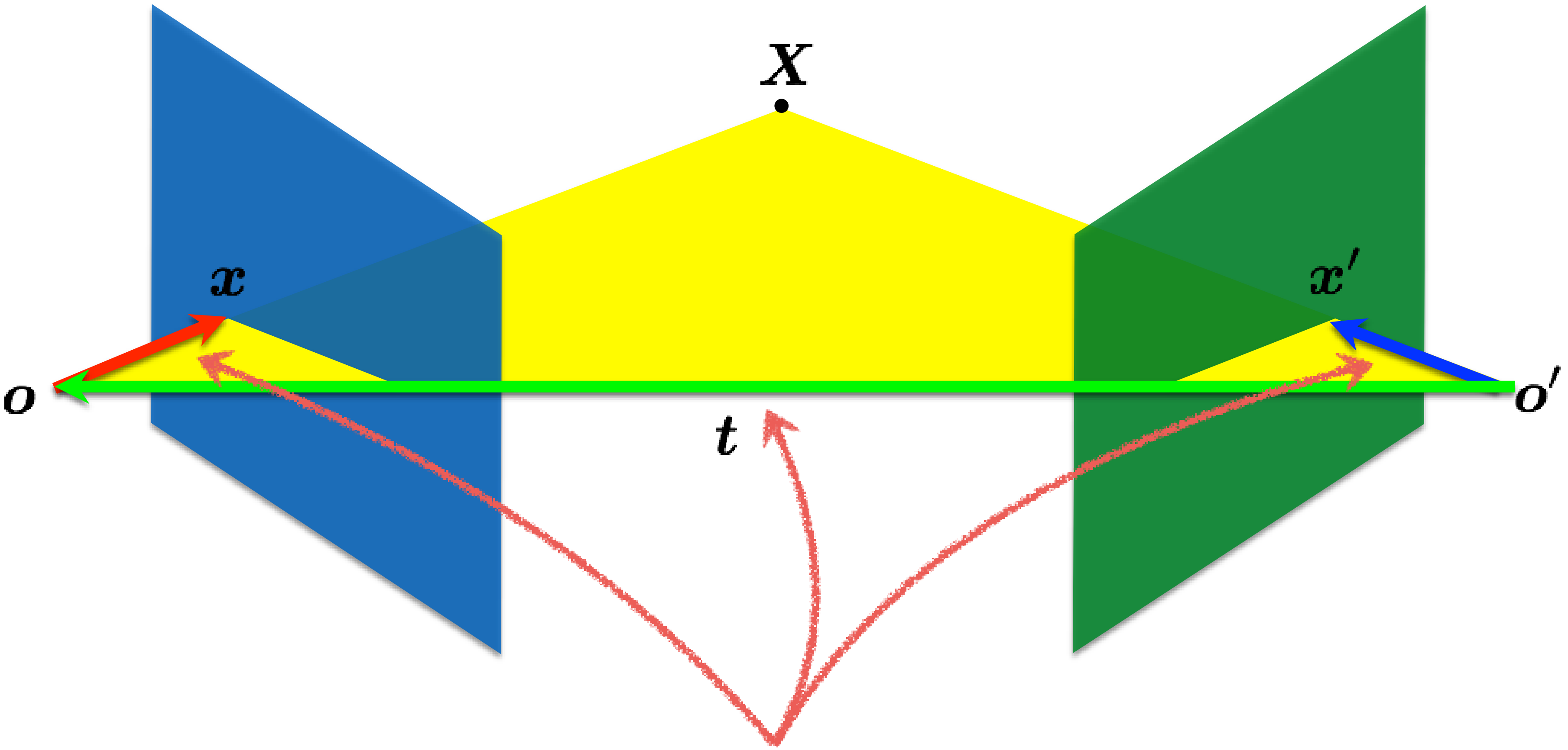
$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$

Does this look familiar?



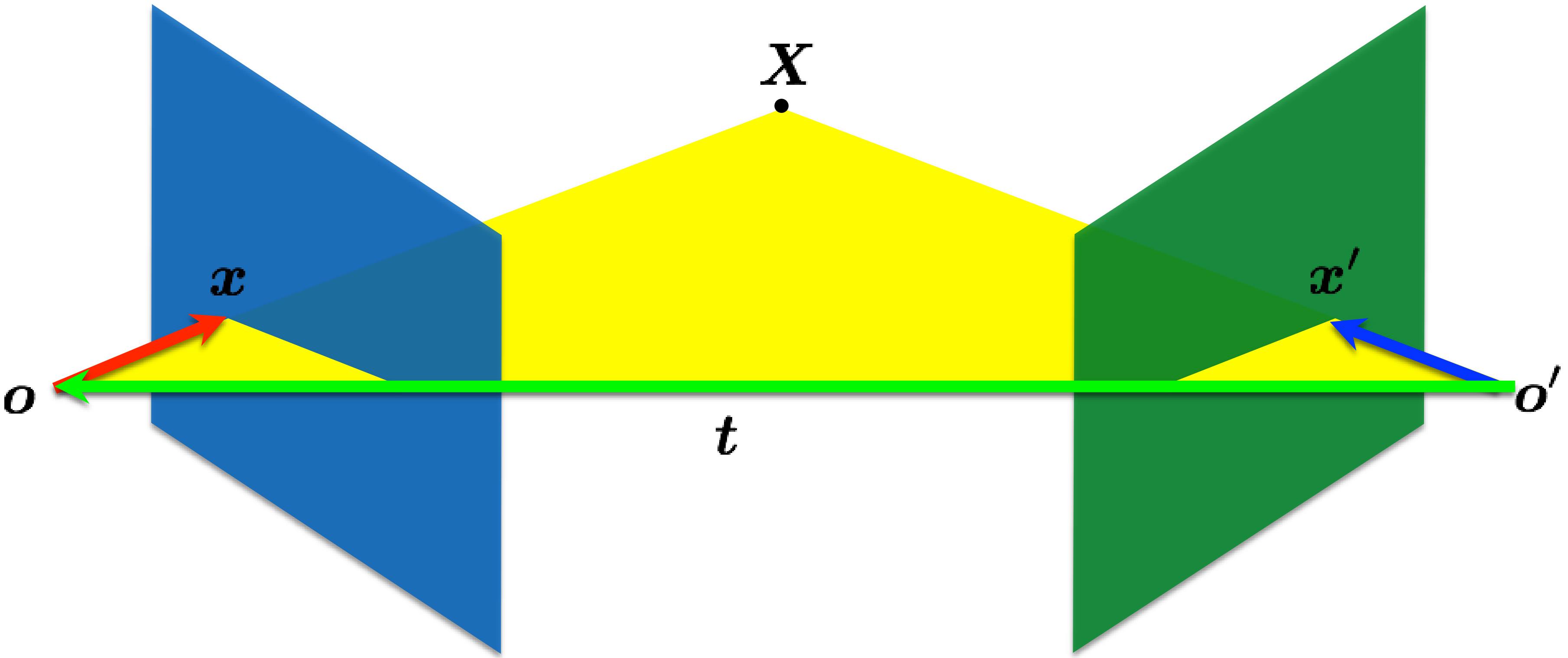
$$x' = \mathbf{R}(x - t)$$

Camera-camera transform just like **world-camera** transform



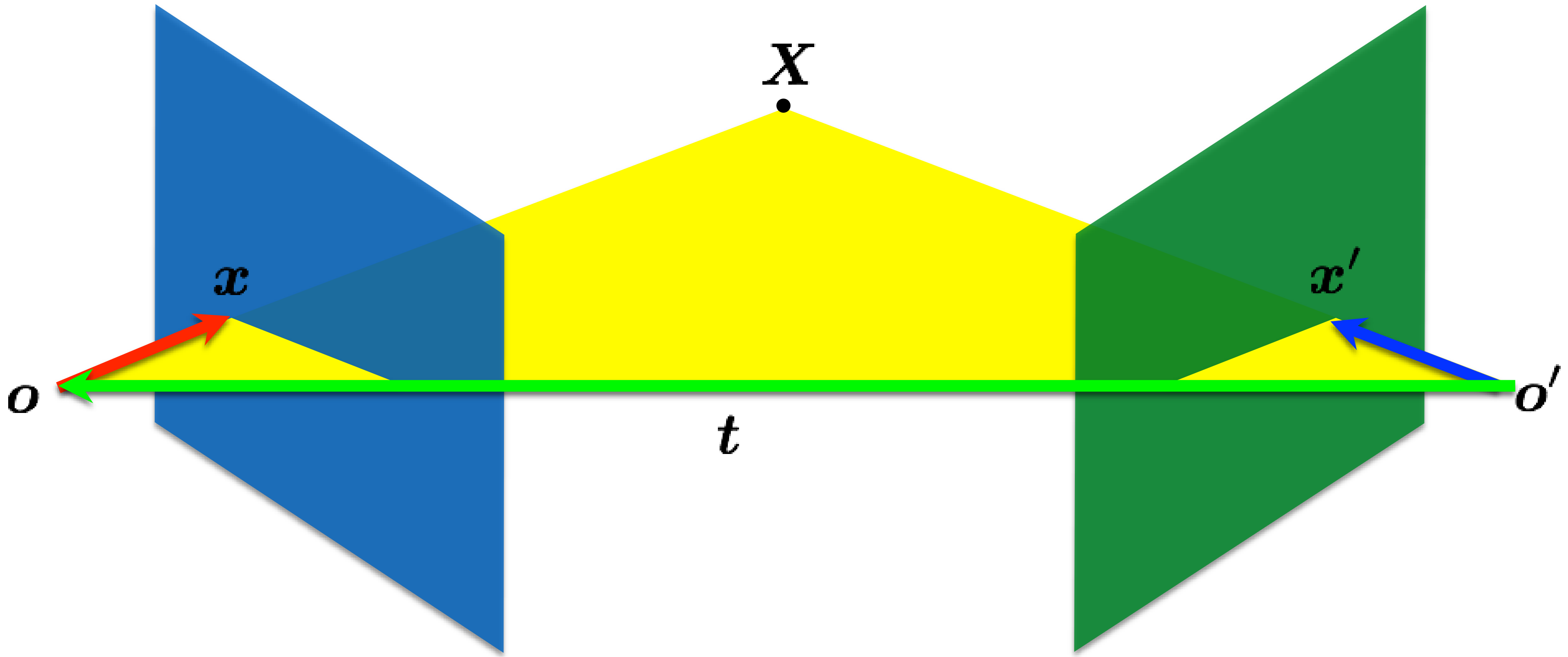
These three vectors are coplanar

$$\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{x}'$$



If these three vectors $\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{x}'$ are coplanar, then

$$\boldsymbol{x}^\top(\boldsymbol{t} \times \boldsymbol{x}) = \ ?$$

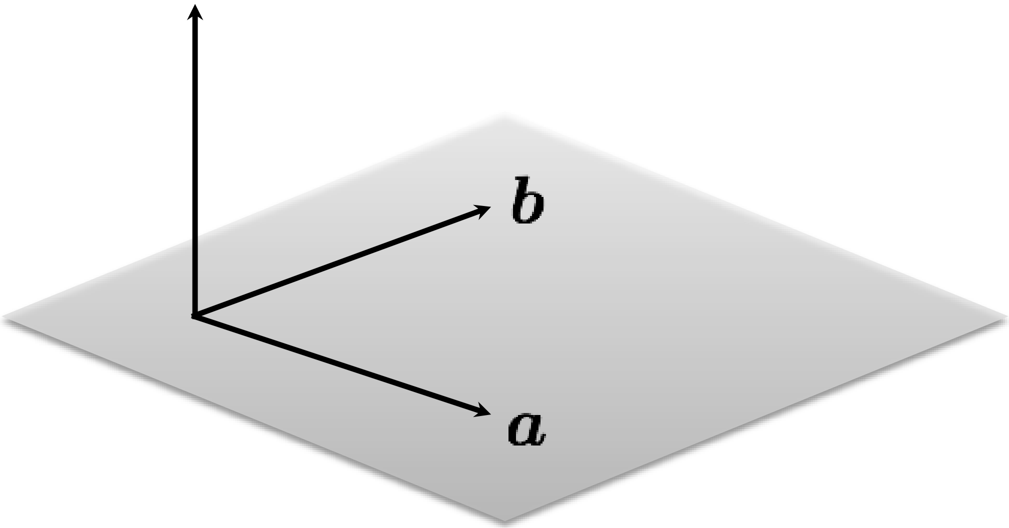


$$\boldsymbol{x}^\top(\boldsymbol{t} \times \boldsymbol{x}) = 0$$

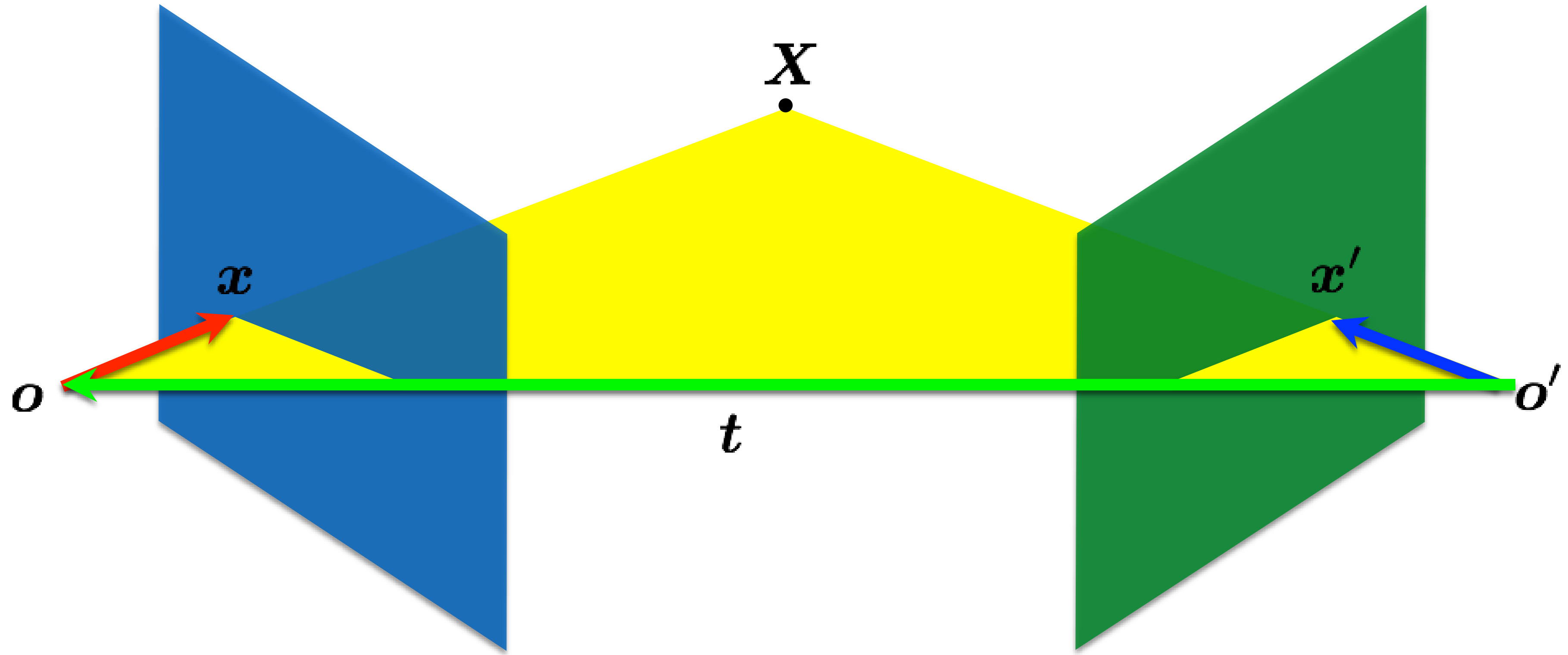
Recall: Cross Product

Vector (cross) product
takes two vectors and returns a vector perpendicular to both

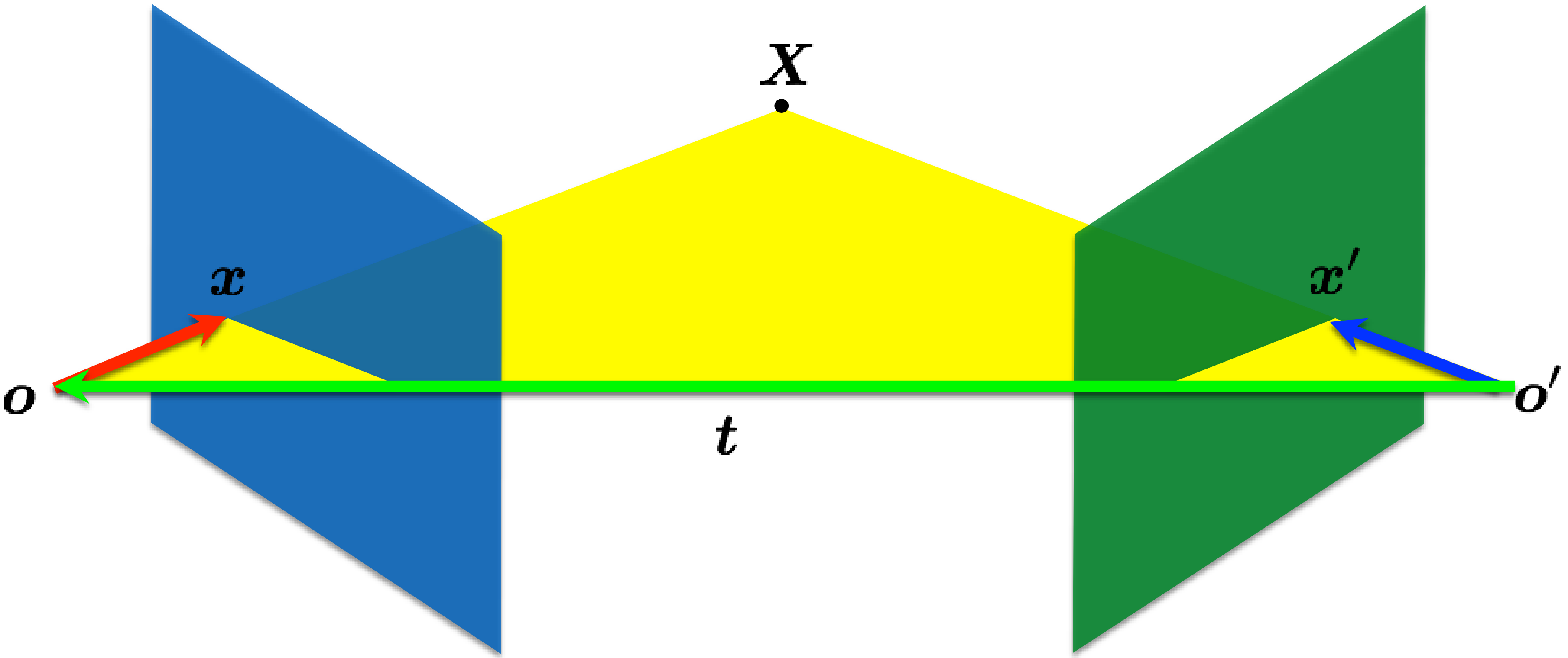
$$\boldsymbol{c} = \boldsymbol{a} \times \boldsymbol{b}$$



$$\boldsymbol{c} \cdot \boldsymbol{a} = 0 \qquad \boldsymbol{c} \cdot \boldsymbol{b} = 0$$



$$(\boldsymbol{x}-\boldsymbol{t})^\top(\boldsymbol{t}\times\boldsymbol{x}) = \ ?$$



$$(\boldsymbol{x} - \boldsymbol{t})^\top (\boldsymbol{t} \times \boldsymbol{x}) = 0$$

putting it together

rigid motion

$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$

coplanarity

$$(\boldsymbol{x} - \boldsymbol{t})^\top (\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})(\boldsymbol{t} \times \boldsymbol{x}) = 0$$

Cross product

$$\boldsymbol{a} \times \boldsymbol{b} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix}$$

Can also be written as a matrix multiplication

$$\boldsymbol{a} \times \boldsymbol{b} = [\boldsymbol{a}]_\times \boldsymbol{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Skew symmetric

putting it together

rigid motion

$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$

coplanarity

$$(\boldsymbol{x} - \boldsymbol{t})^\top (\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})(\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})([\mathbf{t}_\times]\boldsymbol{x}) = 0$$

putting it together

rigid motion

$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$

coplanarity

$$(\boldsymbol{x} - \boldsymbol{t})^\top (\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})(\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})([\mathbf{t}_\times]\boldsymbol{x}) = 0$$

$$\boldsymbol{x}'^\top (\mathbf{R}[\mathbf{t}_\times])\boldsymbol{x} = 0$$

putting it together

rigid motion

$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$

coplanarity

$$(\boldsymbol{x} - \boldsymbol{t})^\top (\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})(\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})([\mathbf{t}_\times]\boldsymbol{x}) = 0$$

$$\boldsymbol{x}'^\top (\mathbf{R}[\mathbf{t}_\times])\boldsymbol{x} = 0$$

$$\boldsymbol{x}'^\top \mathbf{E}\boldsymbol{x} = 0$$

putting it together

rigid motion

$$\boldsymbol{x}' = \mathbf{R}(\boldsymbol{x} - \boldsymbol{t})$$

coplanarity

$$(\boldsymbol{x} - \boldsymbol{t})^\top (\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})(\boldsymbol{t} \times \boldsymbol{x}) = 0$$

$$(\boldsymbol{x}'^\top \mathbf{R})([\mathbf{t}_\times]\boldsymbol{x}) = 0$$

$$\boldsymbol{x}'^\top (\mathbf{R}[\mathbf{t}_\times])\boldsymbol{x} = 0$$

$$\boxed{\boldsymbol{x}'^\top \mathbf{E}\boldsymbol{x} = 0}$$

Essential Matrix
[Longuet-Higgins 1981]

properties of the E matrix

Longuet-Higgins equation $$\boldsymbol{x}'^{\top}\mathbf{E}\boldsymbol{x} = 0$$

(points in normalized coordinates)

properties of the E matrix

Longuet-Higgins equation

$$\boldsymbol{x}'^{\top}\mathbf{E}\boldsymbol{x} = 0$$

Epipolar lines

$$\boldsymbol{x}^{\top}\boldsymbol{l} = 0 \qquad \boldsymbol{x}'^{\top}\boldsymbol{l}' = 0$$

$$\boldsymbol{l}' = \mathbf{E}\boldsymbol{x} \qquad \boldsymbol{l} = \mathbf{E}^{T}\boldsymbol{x}'$$

(points in normalized coordinates)

properties of the E matrix

Longuet-Higgins equation

$$\boldsymbol{x}'^\top \mathbf{E}\boldsymbol{x} = 0$$

Epipolar lines

$$\boldsymbol{x}^\top \boldsymbol{l} = 0 \qquad \boldsymbol{x}'^\top \boldsymbol{l}' = 0$$

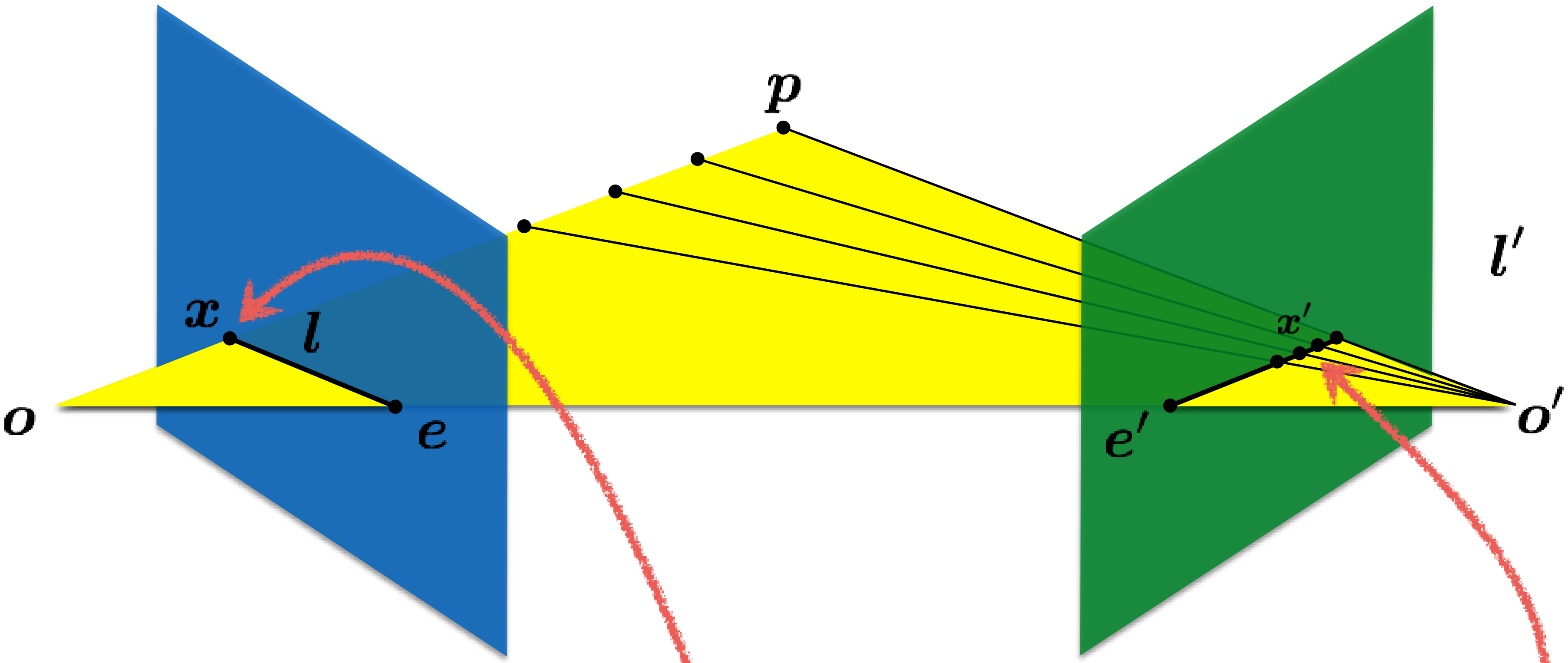
$$\boldsymbol{l}' = \mathbf{E}\boldsymbol{x} \qquad \boldsymbol{l} = \mathbf{E}^T \boldsymbol{x}'$$

Epipoles

$$\boldsymbol{e}'^\top \mathbf{E} = \mathbf{0} \qquad \mathbf{E}\boldsymbol{e} = \mathbf{0}$$

points in normalized <u>camera</u> coordinates…

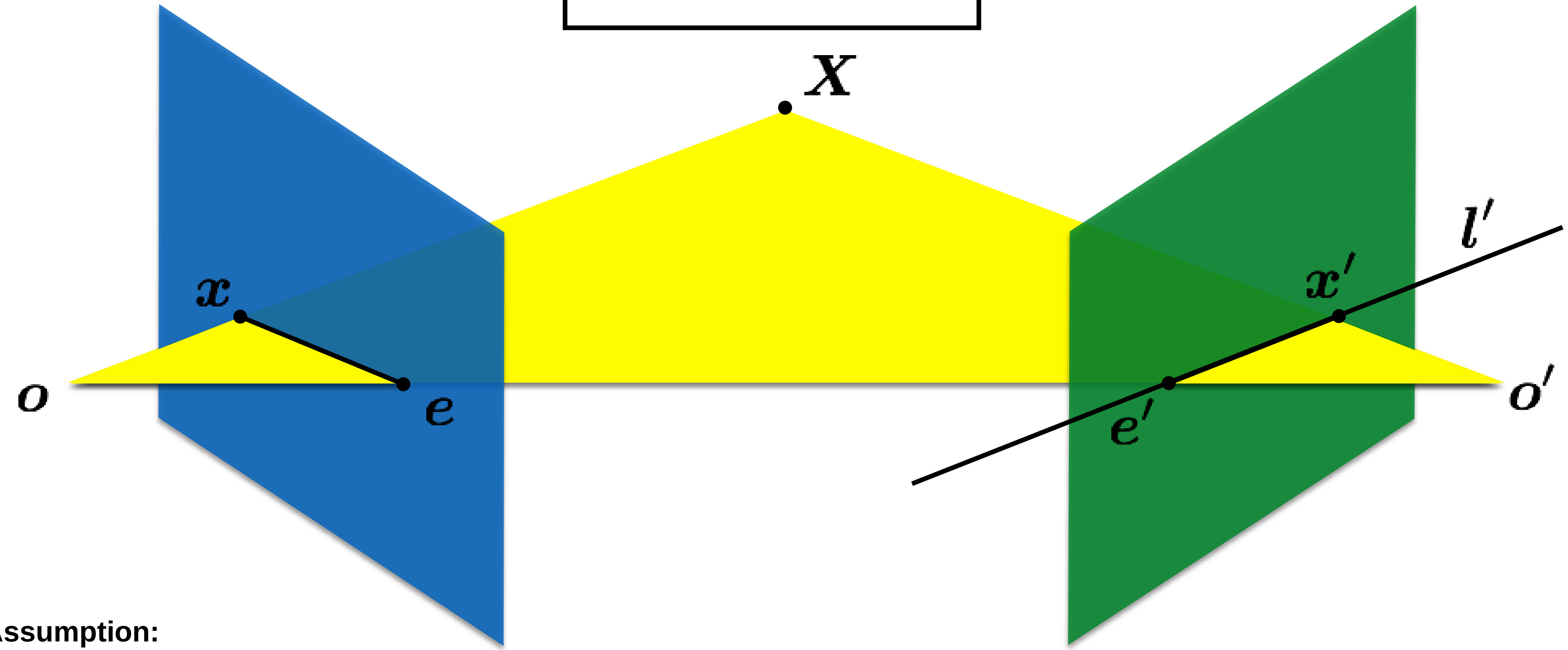
Recall: Epipolar constraint



Potential matches for x lie on the epipolar line l'

Given a point in one image,
multiplying by the **essential matrix** will tell us
the **epipolar line** in the second view.

$$\boxed{\mathbf{E}\boldsymbol{x} = \boldsymbol{l}'}$$



Assumption:
points aligned to camera coordinate axis (calibrated camera)

How do you generalize to uncalibrated cameras?

The fundamental matrix

The Fundamental matrix is a **generalization** of the Essential matrix, where the assumption of calibrated cameras is removed

$$\hat{\boldsymbol{x}}'^{\top}\mathbf{E}\hat{\boldsymbol{x}} = 0$$

The Essential matrix operates on image points expressed in
normalized coordinates
(points have been aligned (normalized) to camera coordinates)

$$\hat{\boldsymbol{x}}' = \mathbf{K}^{-1}\boldsymbol{x}'$$

$$\hat{\boldsymbol{x}} = \mathbf{K}^{-1}\boldsymbol{x}$$

camera point

image point

$$\hat{\boldsymbol{x}}'^{\top}\mathbf{E}\hat{\boldsymbol{x}} = 0$$

The Essential matrix operates on image points expressed in
normalized coordinates
(points have been aligned (normalized) to camera coordinates)

$$\hat{\boldsymbol{x}}' = \mathbf{K}^{-1}\boldsymbol{x}'$$

$$\hat{\boldsymbol{x}} = \mathbf{K}^{-1}\boldsymbol{x}$$

camera point — image point

Writing out the epipolar constraint in terms of image coordinates

$$\boldsymbol{x}'^{\top}\mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1}\boldsymbol{x} = 0$$

$$\boldsymbol{x}'^{\top}(\mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1})\boldsymbol{x} = 0$$

$$\boldsymbol{x}'^{\top}\mathbf{F}\boldsymbol{x} = 0$$

Same equation works in image coordinates!

$$\boldsymbol{x}'^{\top}\mathbf{F}\boldsymbol{x} = 0$$

it maps pixels to epipolar lines

properties of the ~~E~~ F matrix

Longuet-Higgins equation	$\boldsymbol{x}'^\top \mathbf{F} \boldsymbol{x} = 0$	
Epipolar lines	$\boldsymbol{x}^\top \boldsymbol{l} = 0$	$\boldsymbol{x}'^\top \boldsymbol{l}' = 0$
	$\boldsymbol{l}' = \mathbf{F} \boldsymbol{x}$	$\boldsymbol{l} = \mathbf{F}^T \boldsymbol{x}'$
Epipoles	$\boldsymbol{e}'^\top \mathbf{F} = \mathbf{0}$	$\mathbf{F} \boldsymbol{e} = \mathbf{0}$

(points in **image** coordinates)

Breaking down the fundamental matrix

$$\mathbf{F} = \mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1}$$

$$\mathbf{F} = \mathbf{K}'^{-\top}[\mathbf{t}_{\times}]\mathbf{R}\mathbf{K}^{-1}$$

Depends on both intrinsic and extrinsic parameters

Breaking down the fundamental matrix

$$\mathbf{F} = \mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1}$$

$$\mathbf{F} = \mathbf{K}'^{-\top}[\mathbf{t}_{\times}]\mathbf{R}\mathbf{K}^{-1}$$

Depends on both intrinsic and extrinsic parameters

How would you solve for F?

$$\boldsymbol{x}'^{\top}_m \mathbf{F} \boldsymbol{x}_m = 0$$

The 8-point algorithm

Assume you have M matched *image* points

$$\{\boldsymbol{x}_m, \boldsymbol{x}'_m\} \qquad m = 1, \ldots, M$$

Each correspondence should satisfy

$$\boldsymbol{x}'^{\top}_m \mathbf{F} \boldsymbol{x}_m = 0$$

How would you solve for the 3 x 3 **F** matrix?

Assume you have M matched *image* points

$$\{\boldsymbol{x}_m, \boldsymbol{x}'_m\} \qquad m = 1, \ldots, M$$

Each correspondence should satisfy

$$\boldsymbol{x}'^{\top}_m \mathbf{F} \boldsymbol{x}_m = 0$$

How would you solve for the 3 x 3 **F** matrix?

S V D

Assume you have M matched *image* points

$$\{\boldsymbol{x}_m, \boldsymbol{x}'_m\} \qquad m = 1, \ldots, M$$

Each correspondence should satisfy

$$\boldsymbol{x}'^{\top}_m \mathbf{F} \boldsymbol{x}_m = 0$$

How would you solve for the 3 x 3 **F** matrix?

Set up a homogeneous linear system with 9 unknowns

$$\boldsymbol{x}_m'^{\top}\mathbf{F}\boldsymbol{x}_m = 0$$

$$\begin{bmatrix} x_m' & y_m' & 1 \end{bmatrix} \begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = 0$$

How many equation do you get from one correspondence?

$$\begin{bmatrix} x'_m & y'_m & 1 \end{bmatrix} \begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = 0$$

ONE correspondence gives you ONE equation

$$\begin{aligned} x_m x'_m f_1 + x_m y'_m f_2 + x_m f_3 + \\ y_m x'_m f_4 + y_m y'_m f_5 + y_m f_6 + \\ x'_m f_7 + y'_m f_8 + f_9 = 0 \end{aligned}$$

$$\begin{bmatrix} x'_m & y'_m & 1 \end{bmatrix} \begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = 0$$

Set up a homogeneous linear system with 9 unknowns

$$\begin{bmatrix} x_1x'_1 & x_1y'_1 & x_1 & y_1x'_1 & y_1y'_1 & y_1 & x'_1 & y'_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_Mx'_M & x_My'_M & x_M & y_Mx'_M & y_My'_M & y_M & x'_M & y'_M & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \end{bmatrix} = \mathbf{0}$$

How many equations do you need?

Each point pair (according to epipolar constraint) contributes only one <u>scalar</u> equation

$$\boldsymbol{x}_m'^{\top} \mathbf{F} \boldsymbol{x}_m = 0$$

Note: This is different from the Homography estimation where each point pair contributes 2 equations.

We need at least 8 points

Hence, the 8 point algorithm!

How do you solve a homogeneous linear system?

$$\mathbf{A}\boldsymbol{X} = \mathbf{0}$$

How do you solve a homogeneous linear system?

$$\mathbf{A}\mathbf{X} = \mathbf{0}$$

Total Least Squares

minimize $\|\mathbf{A}\boldsymbol{x}\|^2$

subject to $\|\boldsymbol{x}\|^2 = 1$

How do you solve a homogeneous linear system?

$$\mathbf{A}\mathbf{X} = \mathbf{0}$$

Total Least Squares

minimize $\|\mathbf{A}\boldsymbol{x}\|^2$

subject to $\|\boldsymbol{x}\|^2 = 1$

S V D !

Eight-Point Algorithm

0. (Normalize points)
1. Construct the M x 9 matrix **A**
2. Find the SVD of **A**
3. Entries of **F** are the elements of column of **V** corresponding to the least singular value
4. (Enforce rank 2 constraint on F)
5. (Un-normalize F)

Eight-Point Algorithm

0. (Normalize points)
1. Construct the M x 9 matrix **A**
2. Find the SVD of **A**
3. Entries of **F** are the elements of column of **V** corresponding to the least singular value
4. (Enforce rank 2 constraint on F)
5. (Un-normalize F)

See Hartley-Zisserman for why we do this

Eight-Point Algorithm

0. (Normalize points)
1. Construct the M x 9 matrix **A**
2. Find the SVD of **A**
3. Entries of **F** are the elements of column of **V** corresponding to the least singular value
4. (Enforce rank 2 constraint on F)
5. (Un-normalize F)

How do we do this?

Eight-Point Algorithm

0. (Normalize points)
1. Construct the M x 9 matrix **A**
2. Find the SVD of **A**
3. Entries of **F** are the elements of column of **V** corresponding to the least singular value
4. (Enforce rank 2 constraint on F)
5. (Un-normalize F)

How do we do this?

S V D !

Enforcing rank constraints

Problem: Given a matrix **F**, find the matrix **F'** of rank k that is closest to **F**,

$$\min_{\substack{F' \\ \mathrm{rank}(F')=k}} \quad ||F - F'||^2$$

Solution: Compute the singular value decomposition of **F**,

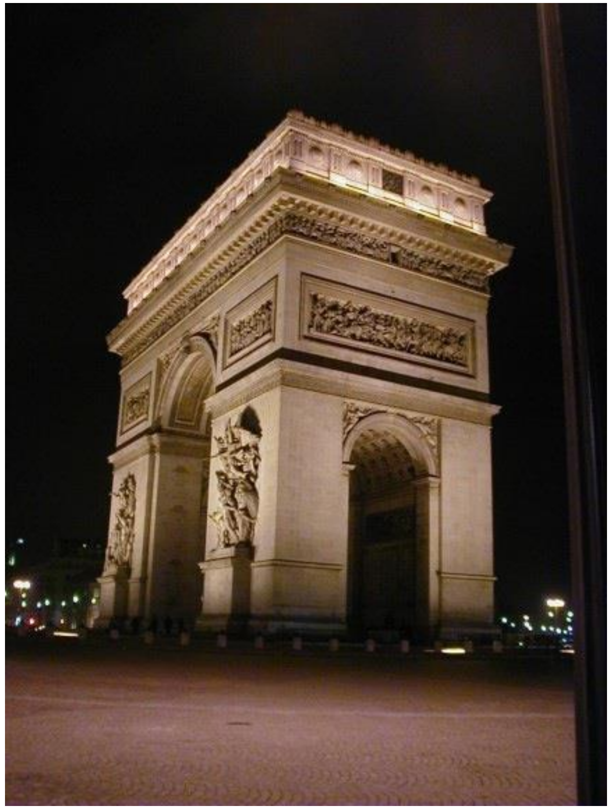
$$F = U\Sigma V^T$$

Form a matrix **Σ'** by replacing all but the k largest singular values in **Σ** with 0.

Then the problem solution is the matrix **F'** formed as,

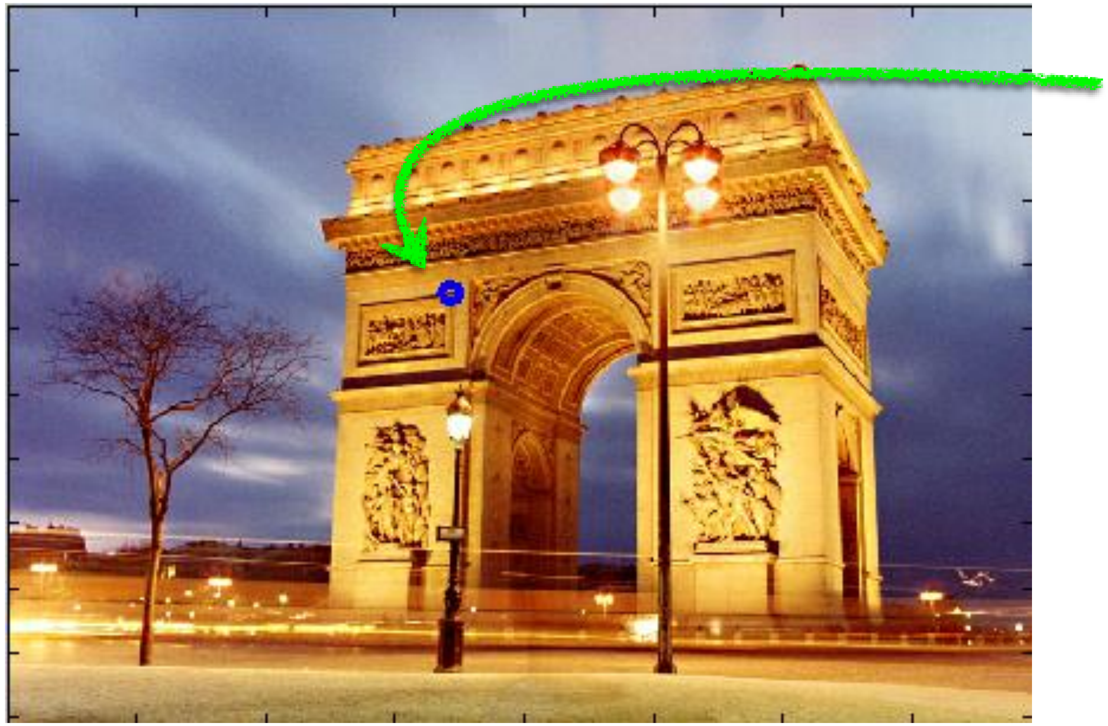
$$F' = U\Sigma' V^T$$

Example



epipolar lines

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



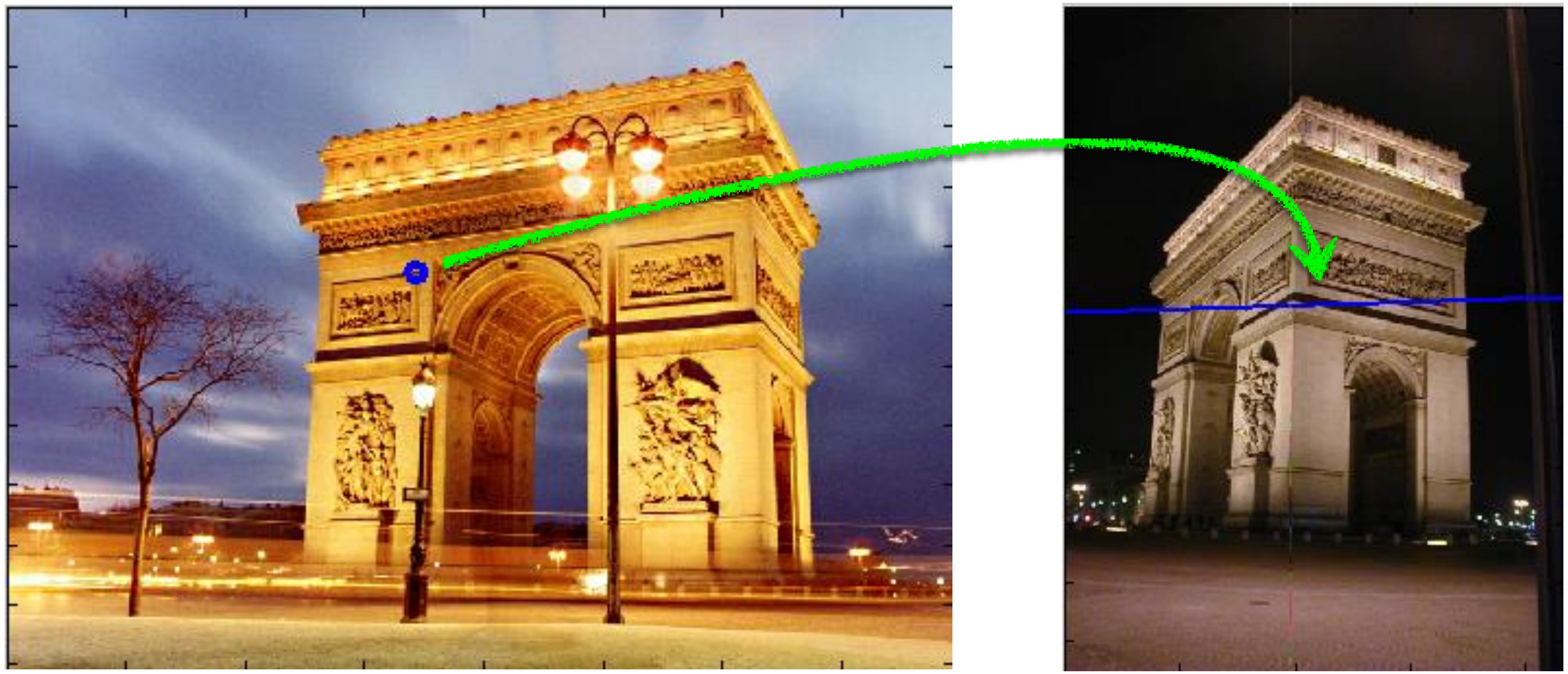
$$\boldsymbol{x} = \begin{bmatrix} 343.53 \\ 221.70 \\ 1.0 \end{bmatrix}$$

$$\boldsymbol{l}' = \mathbf{F}\boldsymbol{x}$$

$$= \begin{bmatrix} 0.0295 \\ 0.9996 \\ -265.1531 \end{bmatrix}$$

$$l' = \mathbf{F}x$$

$$= \begin{bmatrix} 0.0295 \\ 0.9996 \\ -265.1531 \end{bmatrix}$$



Where is the epipole?

How would you compute it?

$$\mathbf{F}\boldsymbol{e} = \mathbf{0}$$

The epipole is in the right null space of **F**

How would you solve for the epipole?

(hint: this is a homogeneous linear system)

$$\mathbf{F}\boldsymbol{e} = \mathbf{0}$$

The epipole is in the right null space of **F**

How would you solve for the epipole?

(hint: this is a homogeneous linear system)

S V D !

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

eigenvectors

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

eigenvalue

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

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

eigenvectors

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

eigenvalue

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

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

eigenvectors

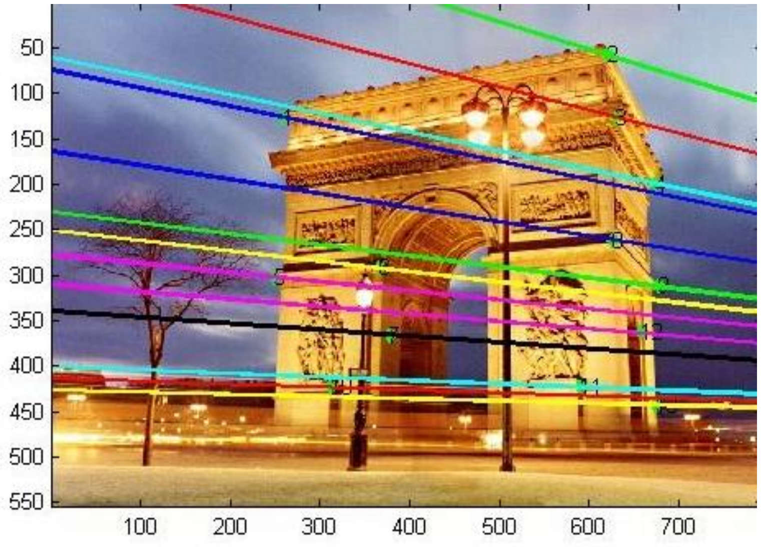
```
u =
   -0.0013    0.2586   -0.9660
    0.0029   -0.9660   -0.2586
    1.0000    0.0032   -0.0005
```

eigenvalue

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

Eigenvector associated with smallest eigenvalue

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



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

eigenvectors

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

eigenvalue

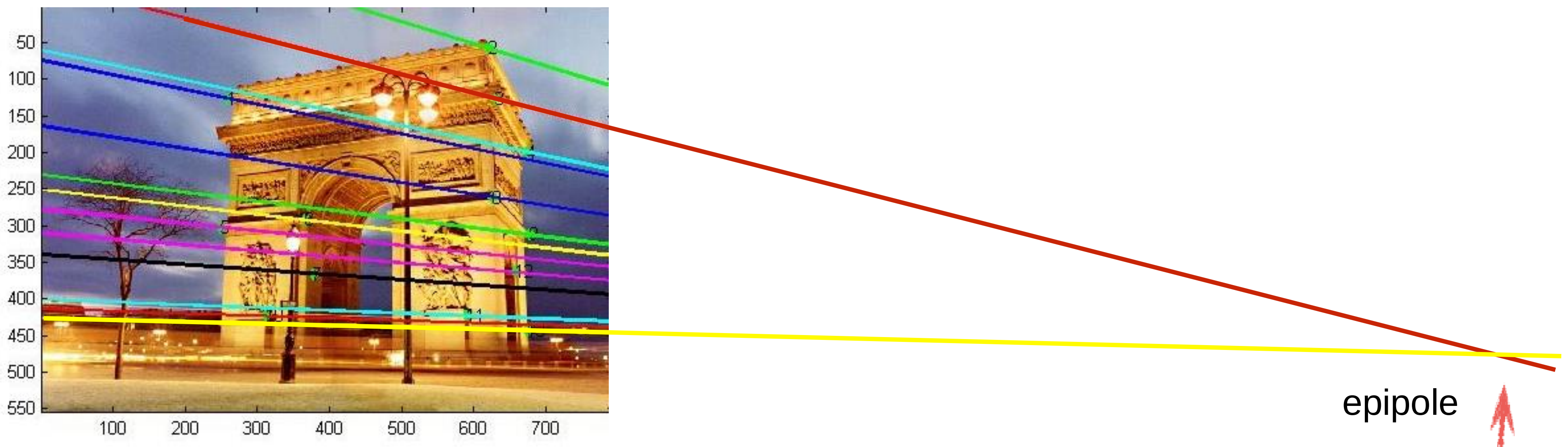
```
d = 1.0e8*
    -1.0000          0          0
          0    -0.0000          0
          0          0    -0.0000
```

Eigenvector associated with smallest eigenvalue

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

Epipole projected to image coordinates

```
>> uu / uu(3)
(1861.02     498.21     1.0)
```



Epipole projected to image coordinates

```
>> uu / uu(3)
(1861.02     498.21     1.0)
```

References

Basic reading:

- Szeliski textbook, Sections 7.1, 7.2, 11.1.
- Hartley and Zisserman, Chapters 9, 11, 12.