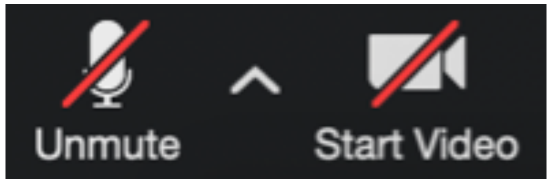
AEROSP 584 - Navigation and Guidance: From Perception to Control



Lectures start at 10:30am EST

Vasileios Tzoumas

Lecture 18

COLLEGE OF ENGINEERING
AEROSPACE ENGINEERING
UNIVERSITY OF MICHIGAN

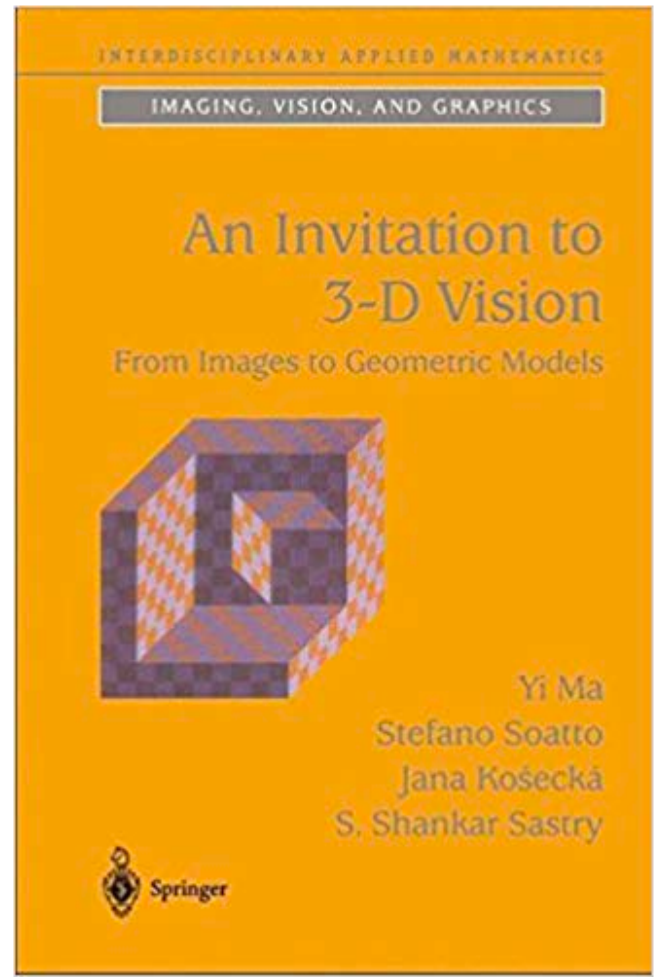
To ask questions:

Raise Hand or Chat

Based on slides made by Luca Carlone @ AEROASTRO MIT

Today

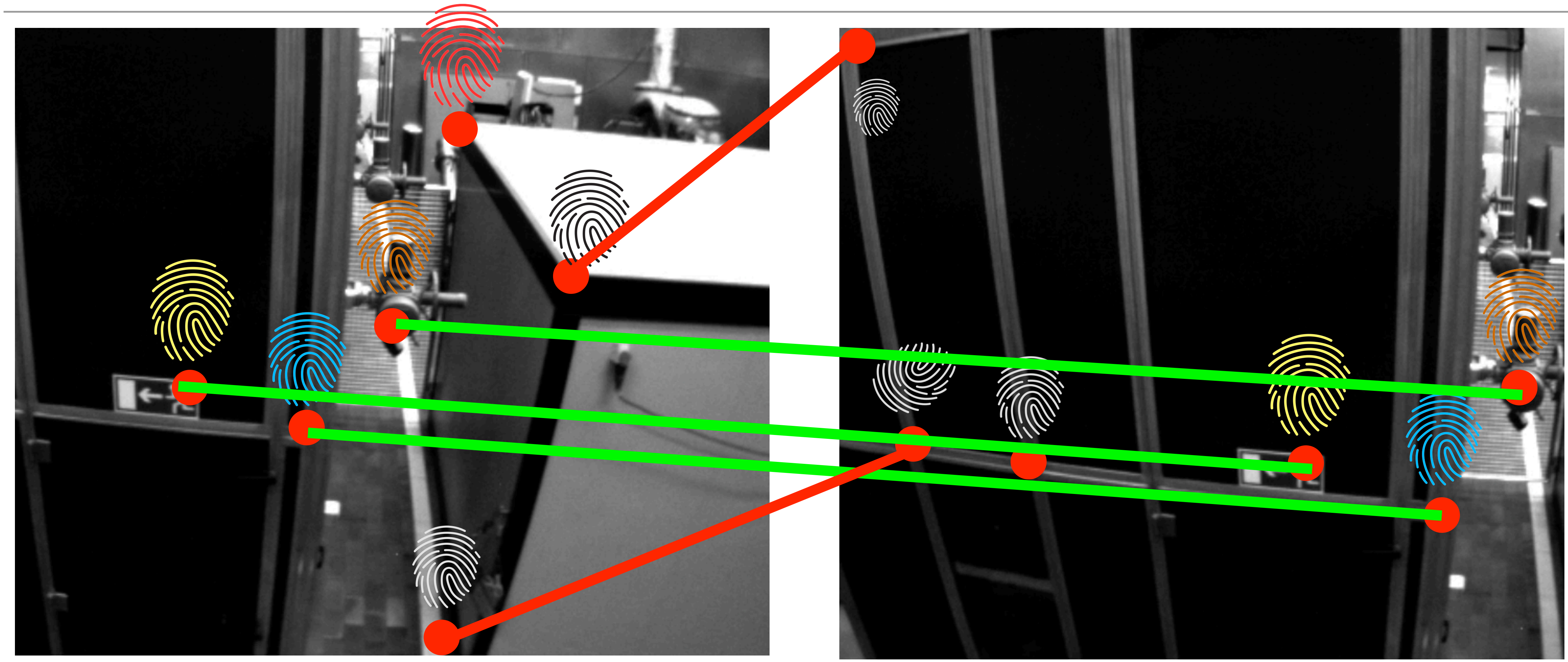
- 2-view geometry



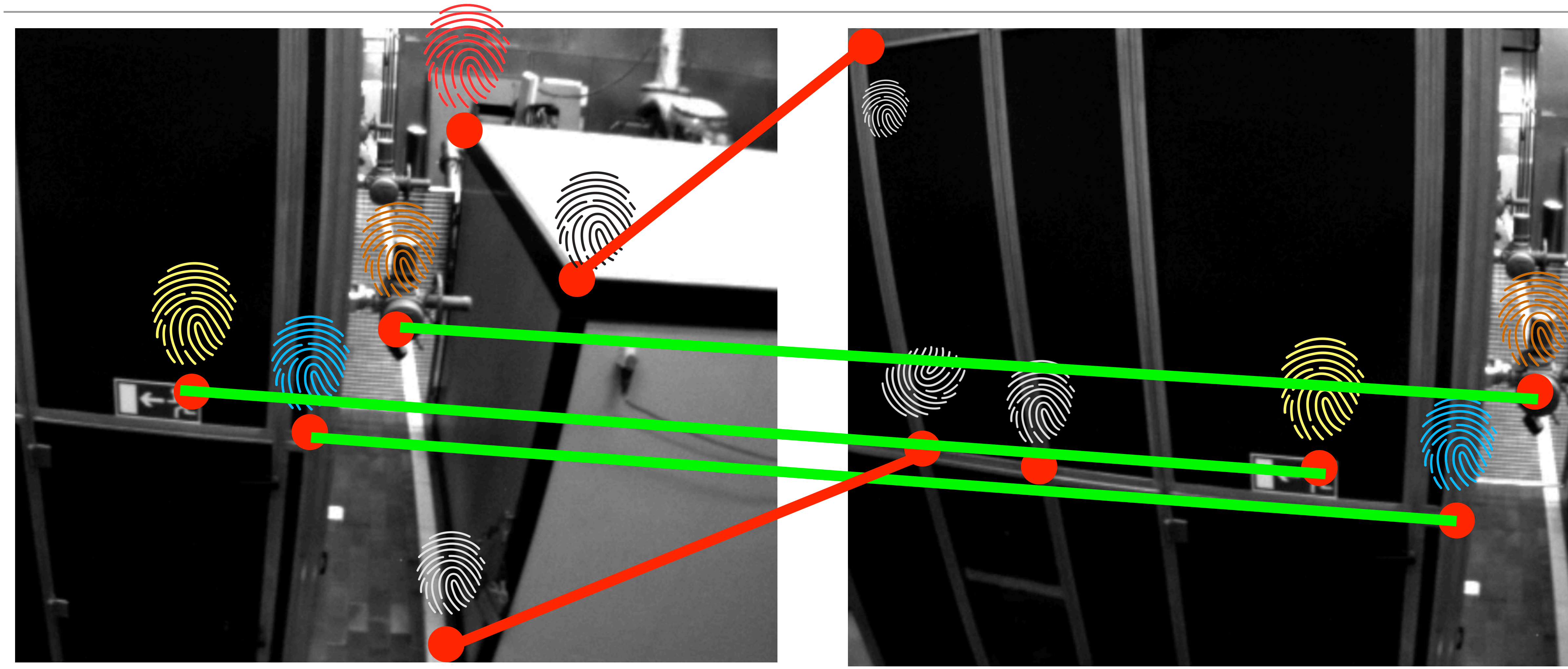
Chapter 5

Reconstruction from Two Calibrated Views

Recap: Point Correspondences

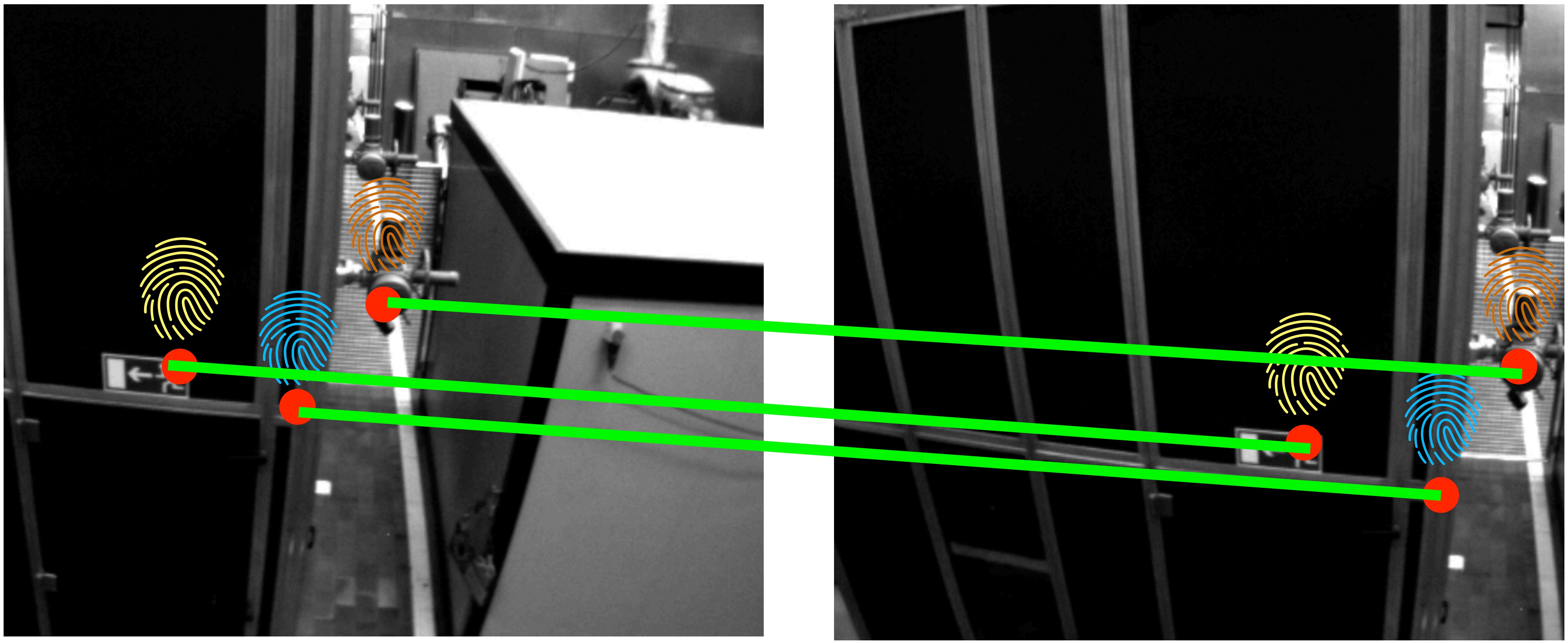


2-view Geometry



Question: can we estimate the motion of the camera between $\boldsymbol{I_1}$ and $\boldsymbol{I_2}$ using pixel correspondences?

2-view Geometry

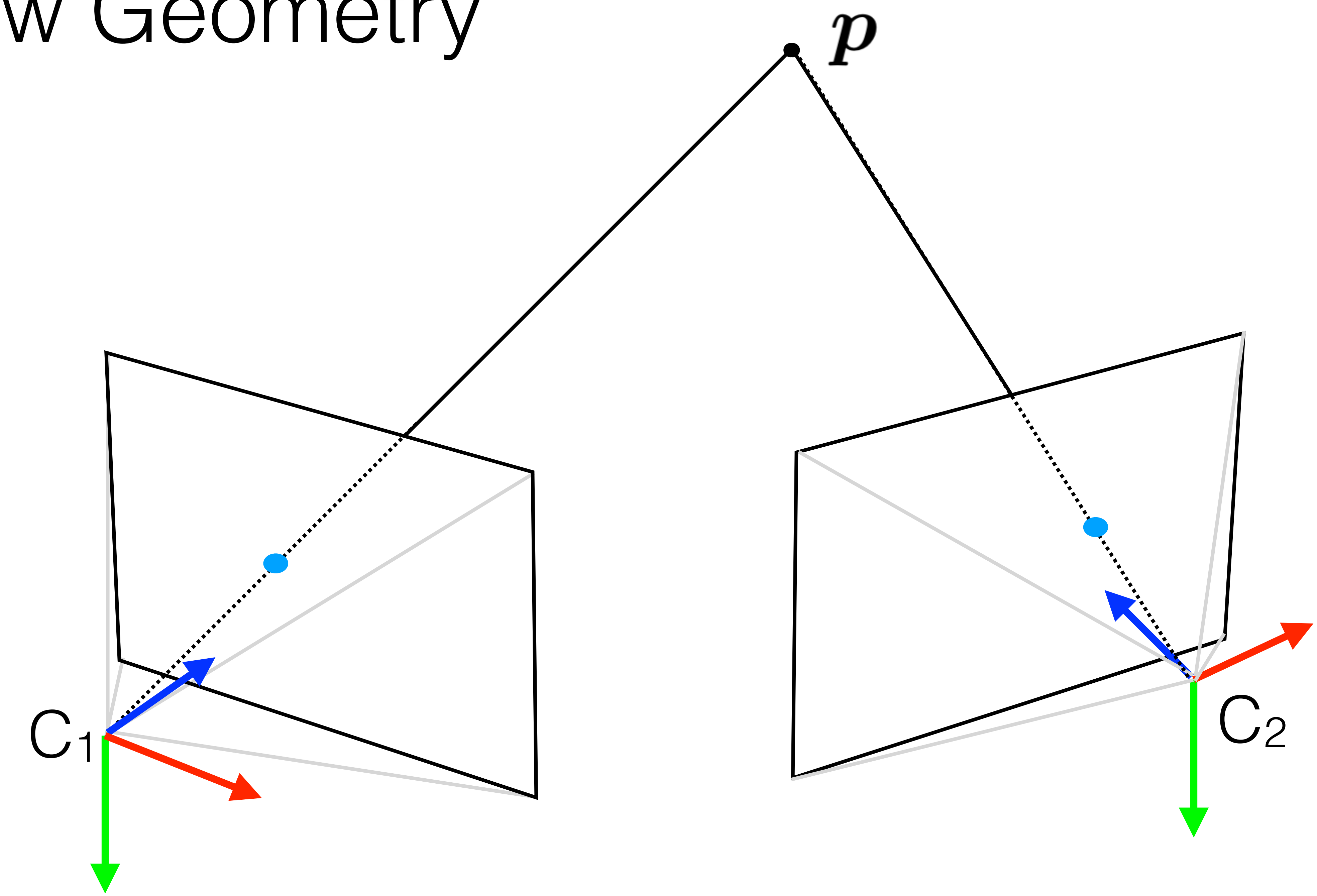


Question: can we estimate the motion of the camera between $\boldsymbol{I_1}$ and $\boldsymbol{I_2}$ using pixel correspondences?

Today's assumptions:

- no wrong correspondences (outliers)
- 3D point is not moving
- camera calibration is known

2-view Geometry



$$p_z^{c_1}\ \tilde{\boldsymbol{x}}_1 = \boldsymbol{K}_1\left[\boldsymbol{R}_{\mathrm{w}}^{\mathrm{c}_1}\ \boldsymbol{t}_{\mathrm{w}}^{\mathrm{c}_1}\right]\tilde{\boldsymbol{p}}^{\mathrm{w}}$$

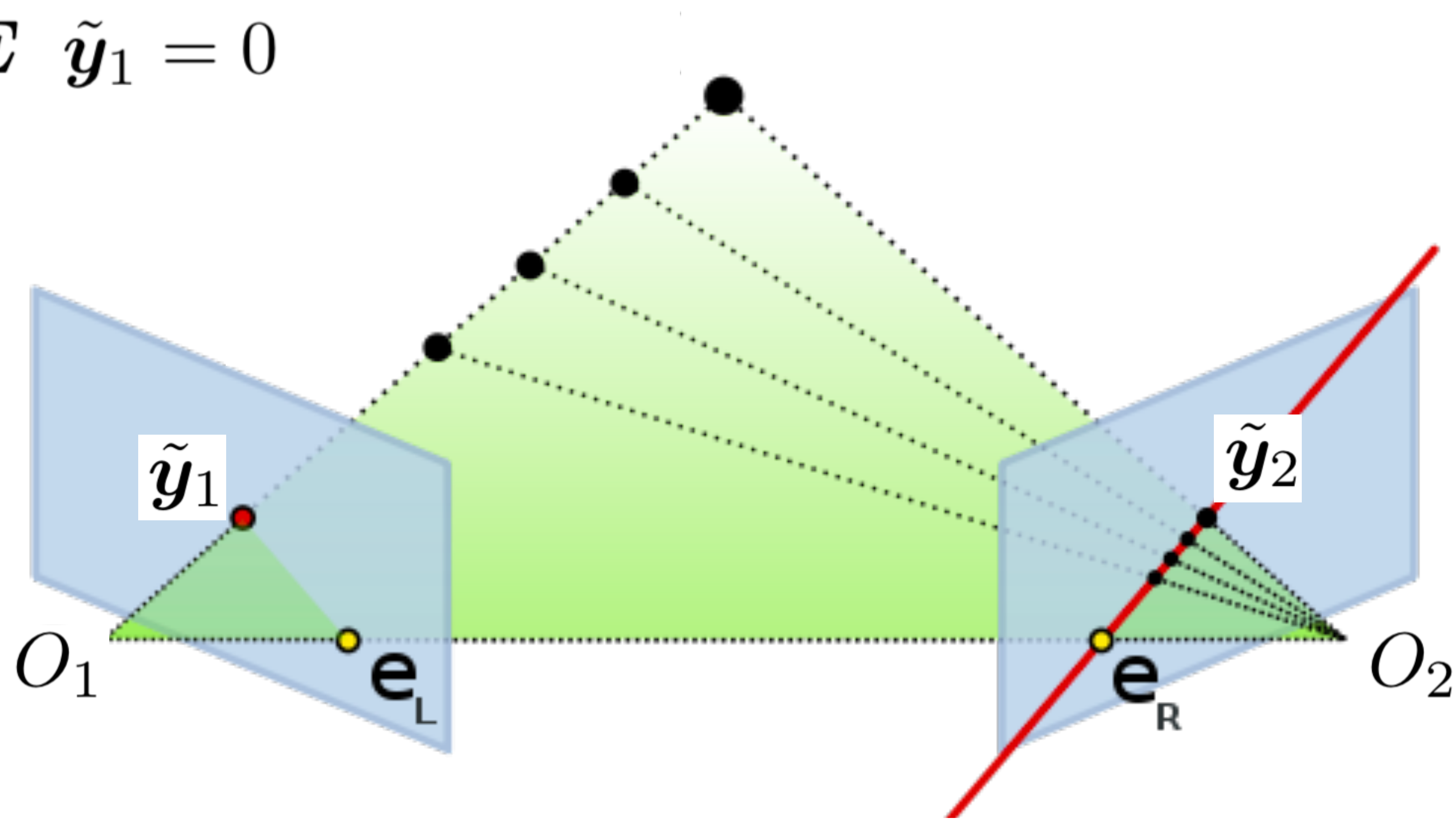
$$p_z^{c_2}\ \tilde{\boldsymbol{x}}_2 = \boldsymbol{K}_2\left[\boldsymbol{R}_{\mathrm{w}}^{\mathrm{c}_2}\ \boldsymbol{t}_{\mathrm{w}}^{\mathrm{c}_2}\right]\tilde{\boldsymbol{p}}^{\mathrm{w}}$$

$$\boldsymbol{K}_1 = \begin{bmatrix} s_{x_1} f_1 & s_{\theta_1} f_1 & o_{x_1} \\ 0 & s_{y_1} f_1 & o_{y_1} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{K}_2 = \begin{bmatrix} s_{x_2} f_2 & s_{\theta_2} f_2 & o_{x_2} \\ 0 & s_{y_2} f_2 & o_{y_2} \\ 0 & 0 & 1 \end{bmatrix}$$

Epipolar Geometry

$$\tilde{\boldsymbol{y}}_2^\top \ \boldsymbol{E} \ \tilde{\boldsymbol{y}}_1 = 0$$

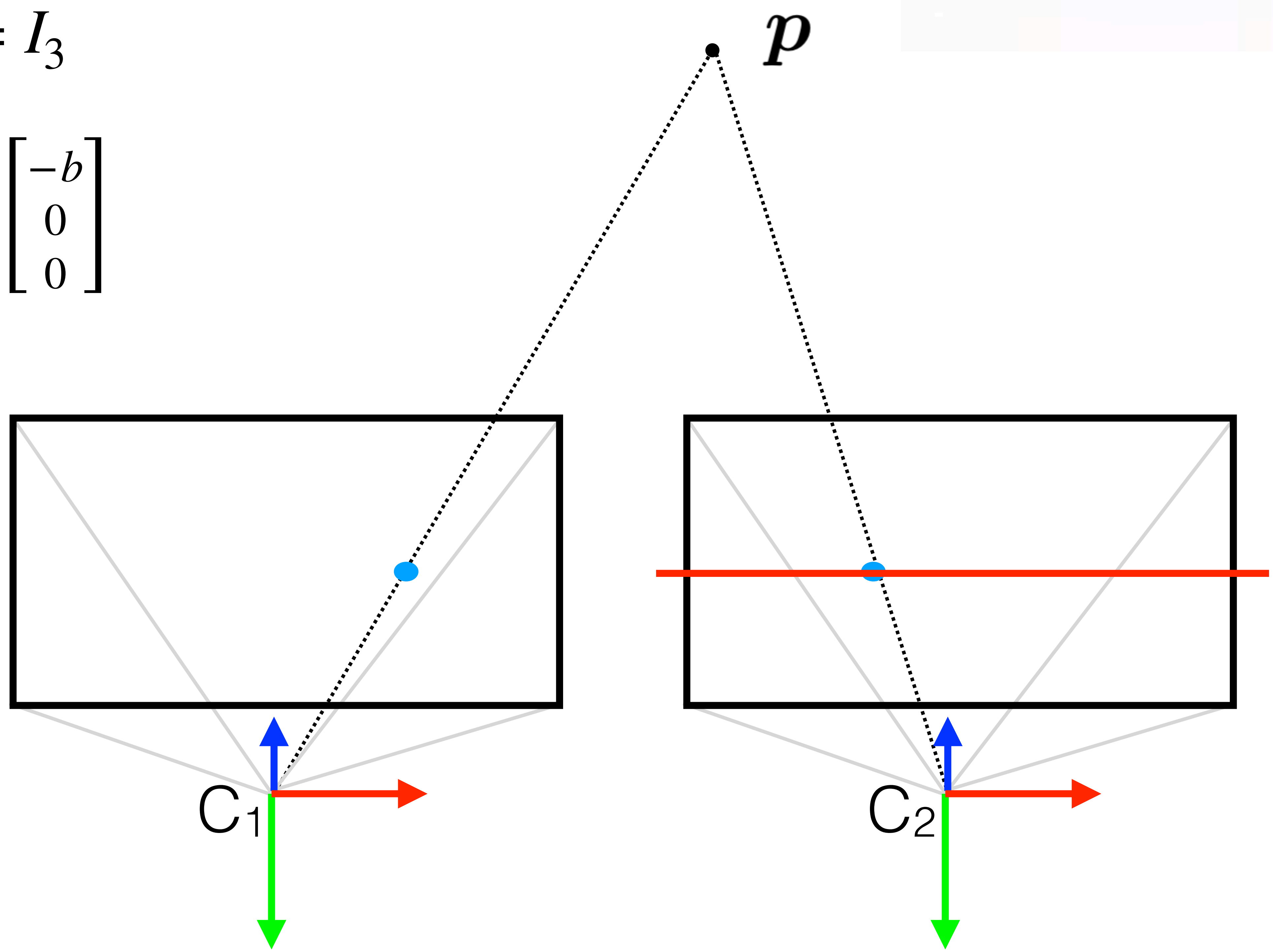


epipolar plane epipolar line **$\mathbf{e}_L$, $\mathbf{e}_R$:** epipoles

Example: **Stereo Camera**

$$R_{C_1}^{C_2} = I_3$$

$$t_{C_1}^{C_2} = \begin{bmatrix} -b \\ 0 \\ 0 \end{bmatrix}$$



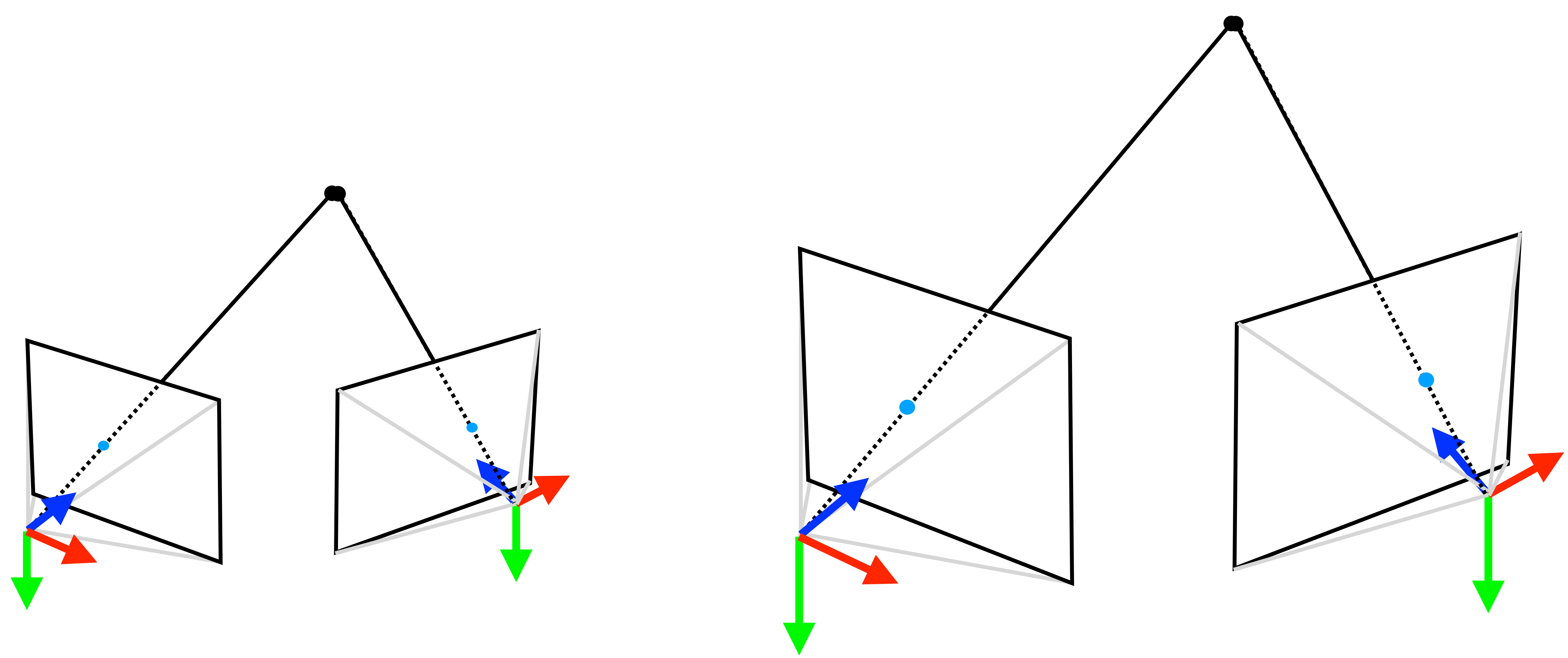
also: easy to triangulate points given geometry

Estimating Poses from Correspondences

Given *N* calibrated pixel correspondences:

$$(\tilde{\boldsymbol{y}}_{1,k}, \tilde{\boldsymbol{y}}_{2,k}) \text{ for } k = 1, \ldots, N$$

compute the relative pose between the cameras



Can we estimate the scale of the translation (baseline)?

Estimating Poses from Correspondences

Given *N* calibrated pixel correspondences:

$$(\tilde{\boldsymbol{y}}_{1,k}, \tilde{\boldsymbol{y}}_{2,k}) \text{ for } k = 1, \ldots, N$$

1. leverage the epipolar constraints to estimate the essential matrix ***E***

$$\tilde{\boldsymbol{y}}_{2,k}^{\mathsf{T}} \, \boldsymbol{E} \, \tilde{\boldsymbol{y}}_{1,k} = 0$$

2. Retrieve the rotation and translation (up to scale) from the ***E***

$$\boldsymbol{E} = [\boldsymbol{t}]_{\times} \boldsymbol{R}$$

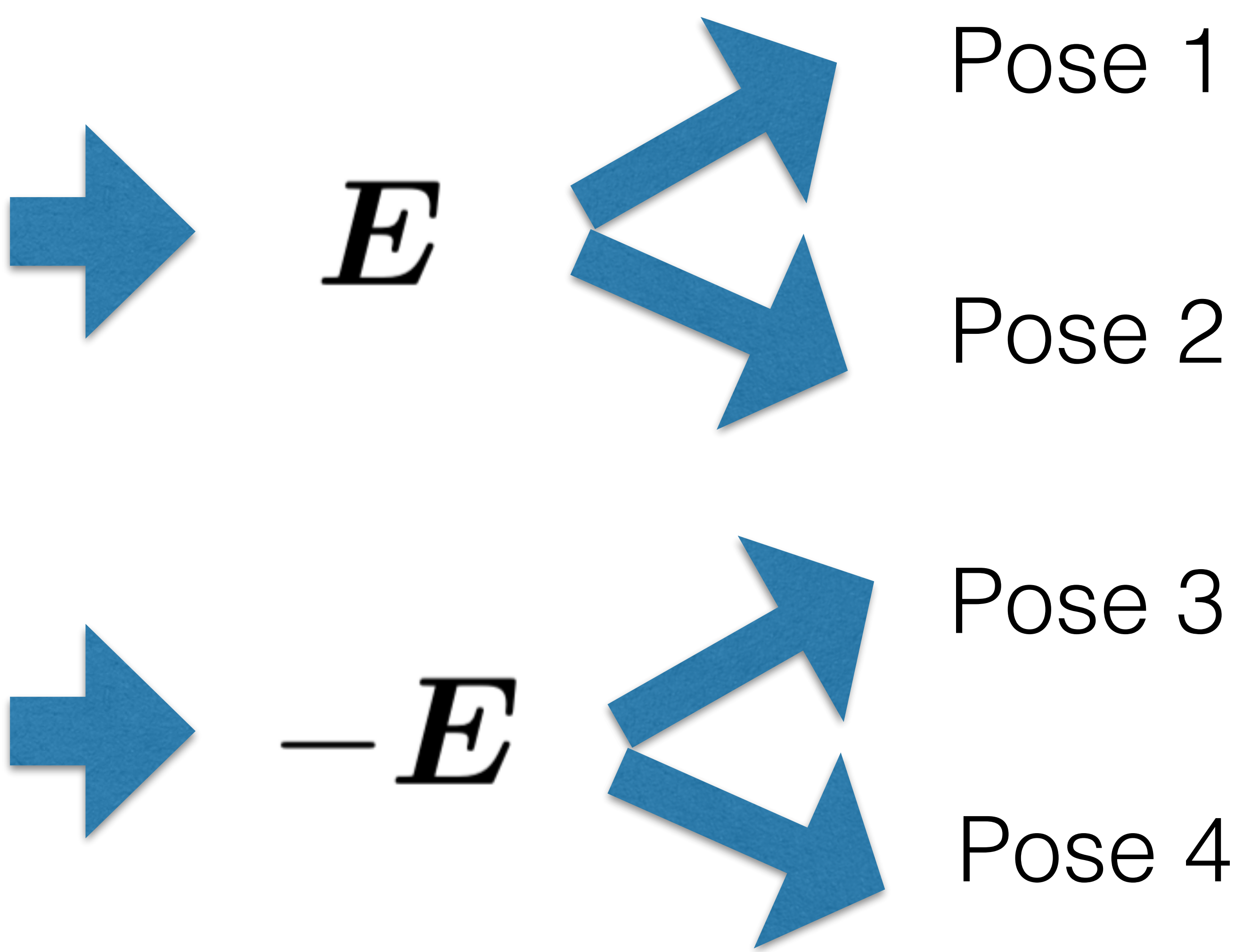
Retrieving Pose from Essential Matrix

Theorem 1 (Pose recovery from essential matrix, Thm 5.7 in [1]). *There exist exactly two relative poses* $(\boldsymbol{R}, \boldsymbol{t})$ *with* $\boldsymbol{R} \in \mathrm{SO}(3)$ *and* $\boldsymbol{t} \in \mathbb{R}^3$ *corresponding to a nonzero essential matrix* $\boldsymbol{E}$ *(i.e., such that* $\boldsymbol{E} = [\mathbf{t}]_\times \boldsymbol{R}$*):*

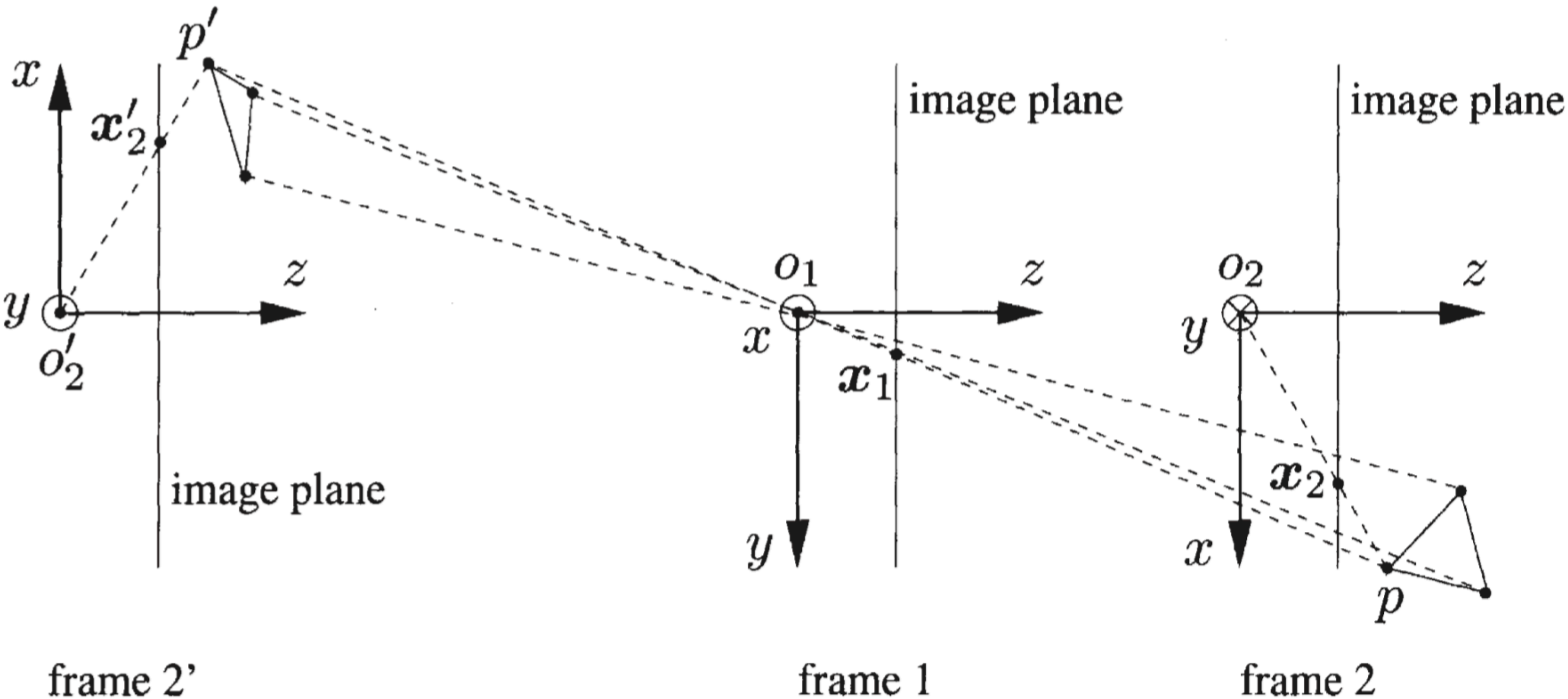
$$\boldsymbol{t}_1 = \boldsymbol{U}\boldsymbol{R}_z(+\pi/2)\boldsymbol{\Sigma}\boldsymbol{U}^\mathsf{T} \qquad \boldsymbol{R}_1 = \boldsymbol{U}\boldsymbol{R}_z(+\pi/2)\boldsymbol{V}^\mathsf{T} \tag{13.19}$$

$$\boldsymbol{t}_2 = \boldsymbol{U}\boldsymbol{R}_z(-\pi/2)\boldsymbol{\Sigma}\boldsymbol{U}^\mathsf{T} \qquad \boldsymbol{R}_2 = \boldsymbol{U}\boldsymbol{R}_z(-\pi/2)\boldsymbol{V}^\mathsf{T} \tag{13.20}$$

where $\boldsymbol{E} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\mathsf{T}$ *is the singular value decomposition of the matrix* $\boldsymbol{E}$*, and* $\boldsymbol{R}_z(+\pi/2)$ *is an elementary rotation around the z-axis of an angle* $\pi/2$*.*



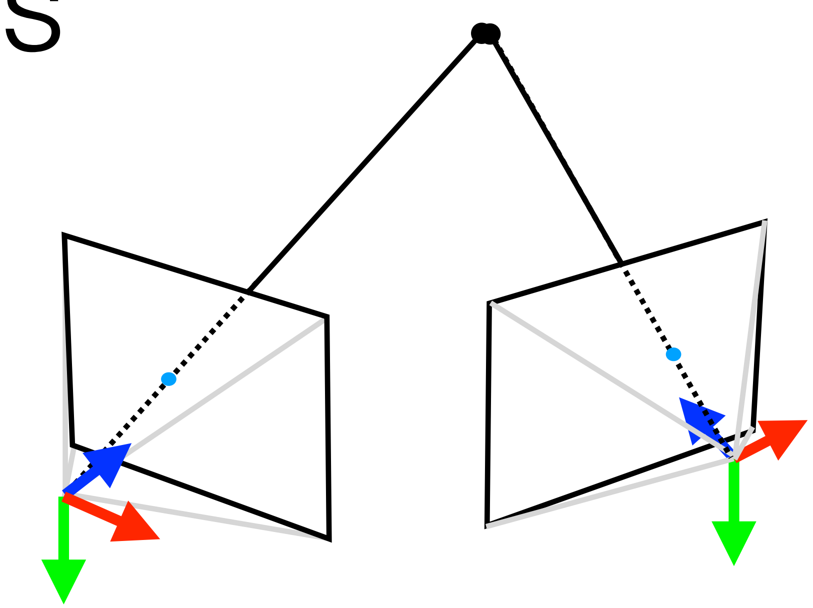
Cheirality constraints



Points must be in front of the cameras!

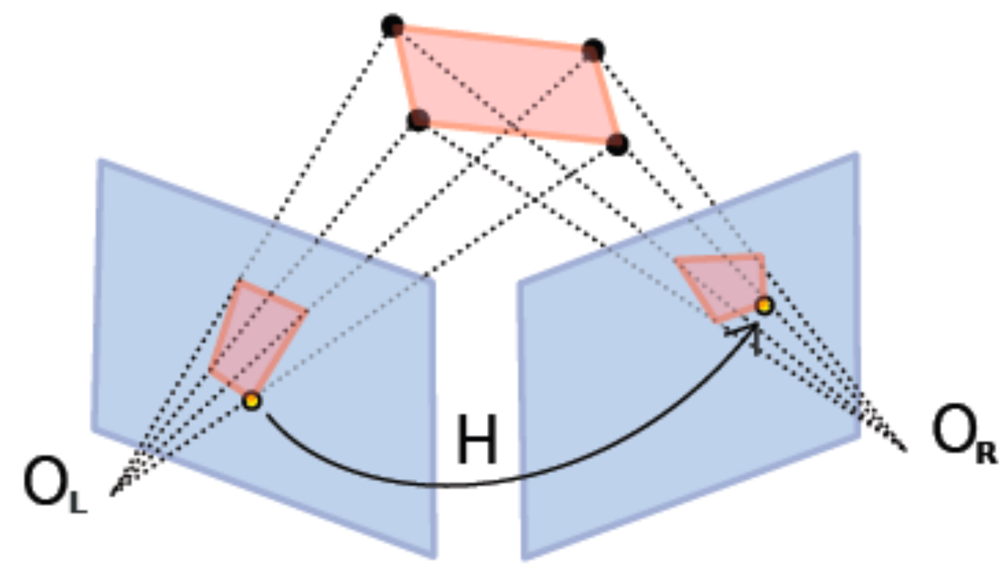
8-point method: Limitations

Number of correspondences: do we really need 8 points?



$$\boldsymbol{E} = [\boldsymbol{t}]_{\times} \boldsymbol{R}$$

Scene structures: there are certain configurations of 3D points that make the algorithm fail

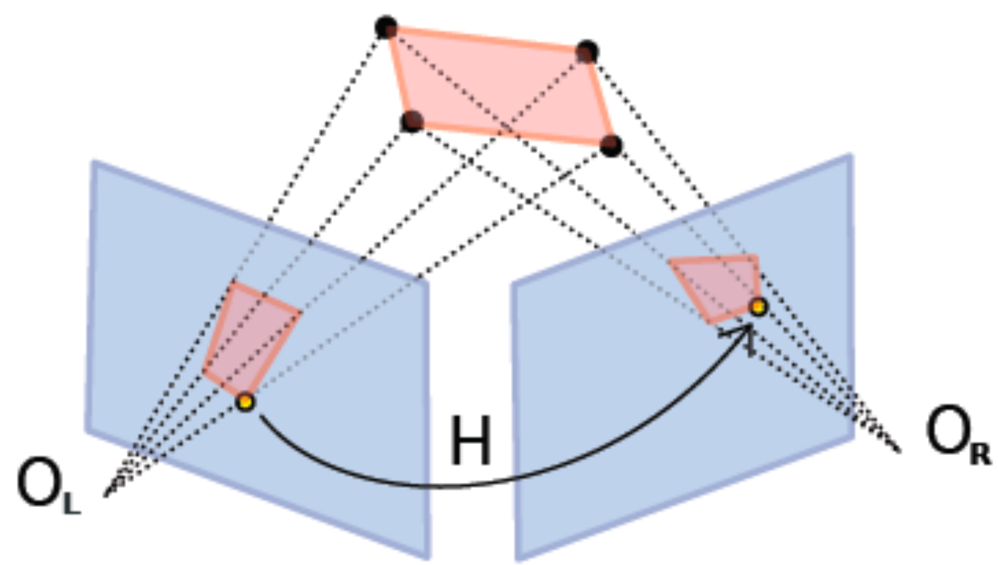


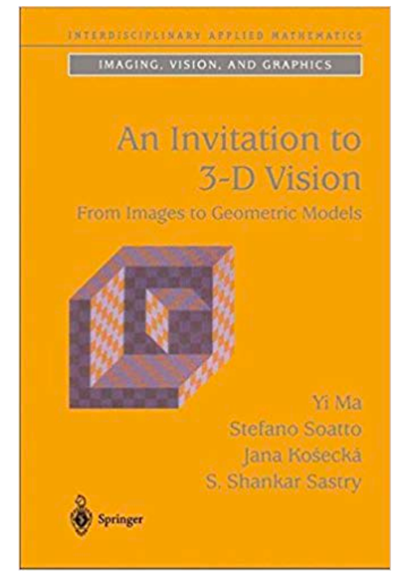
Parallax: what if ***t* = 0**?

Other Matrices in 2-view Geometry

Homography matrix **H**

$$\lambda_2 \boldsymbol{x}_2 = H \lambda_1 \boldsymbol{x}_1$$

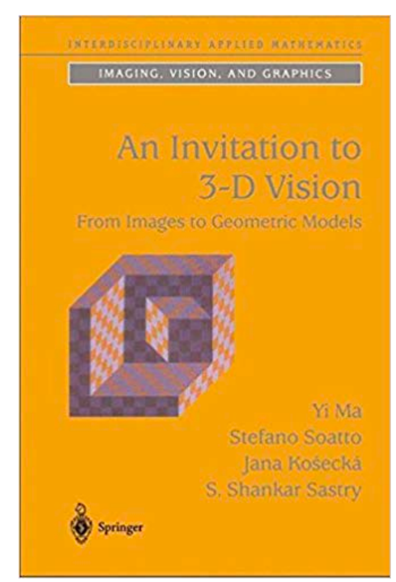




Section 5.3

Fundamental matrix **F**

$$\boldsymbol{F} = \boldsymbol{K}_2^{-\top} \ [\boldsymbol{t}]_\times \boldsymbol{R} \ \boldsymbol{K}_1^{-1}$$



Chapter 6


youtube.com/brusspup