

CSE 152: Computer Vision

Hao Su

Lecture 12: Multiview Geometry



Some slides in this lecture are courtesy to Prof. Silvio Savarese

Agenda

- Why is stereo useful?
- Epipolar constraints
- Fundamental matrix
- Estimating F

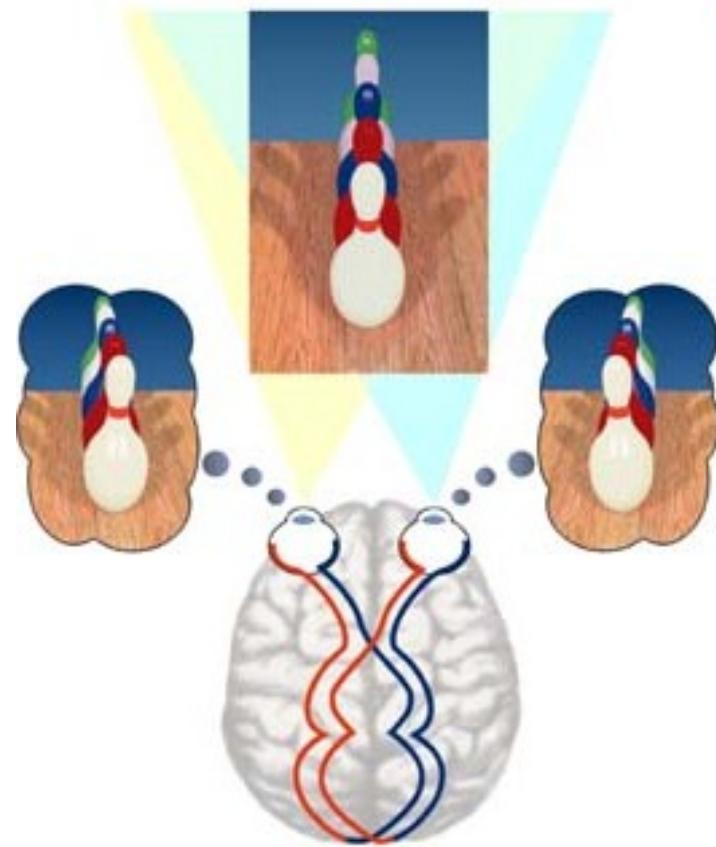
Recovering structure from a single view

Intrinsic ambiguity of the mapping from 3D to image (2D)



Courtesy slide S. Lazebnik

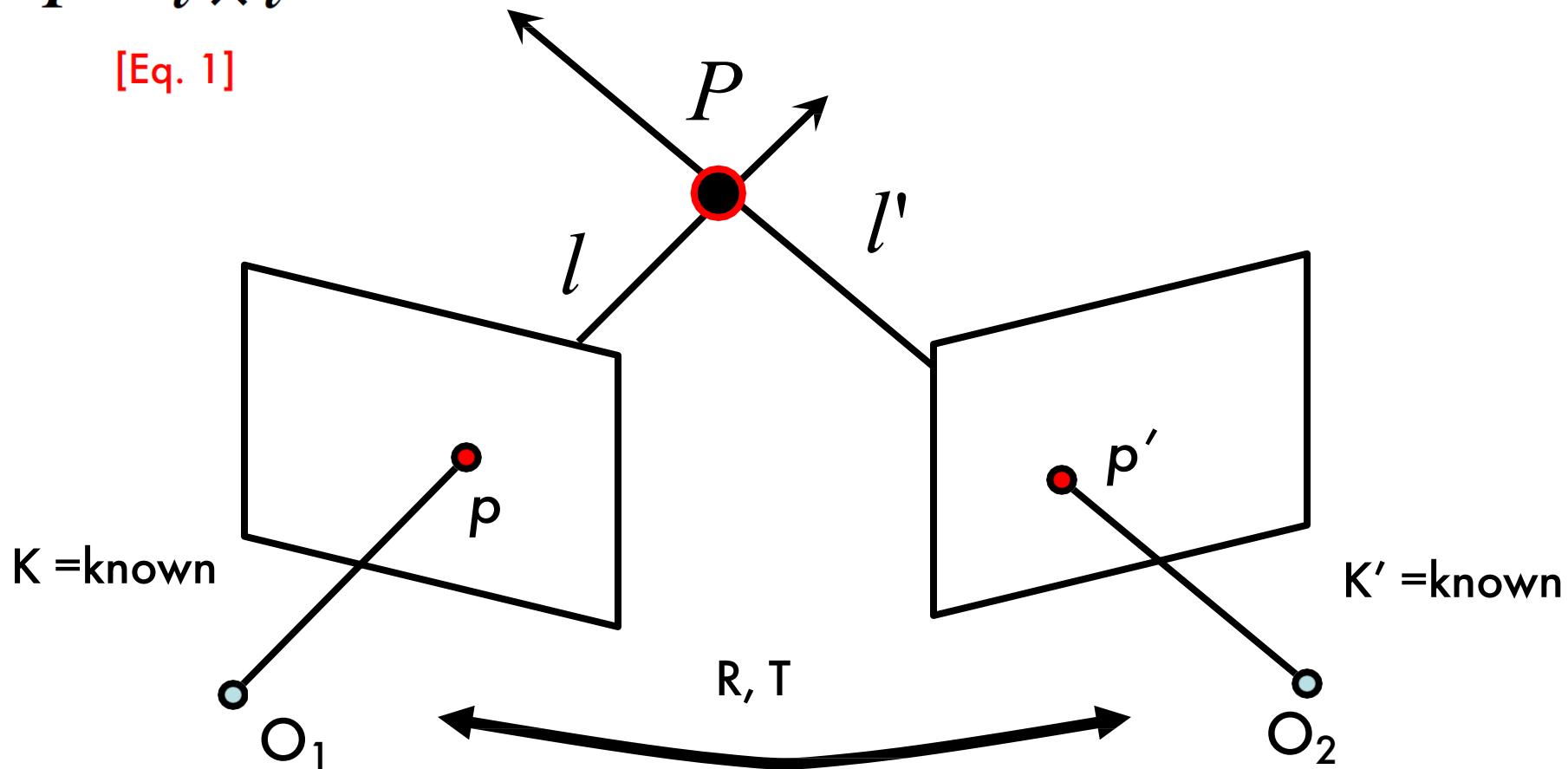
Two eyes help!



Two eyes help!

$$P = l \times l'$$

[Eq. 1]

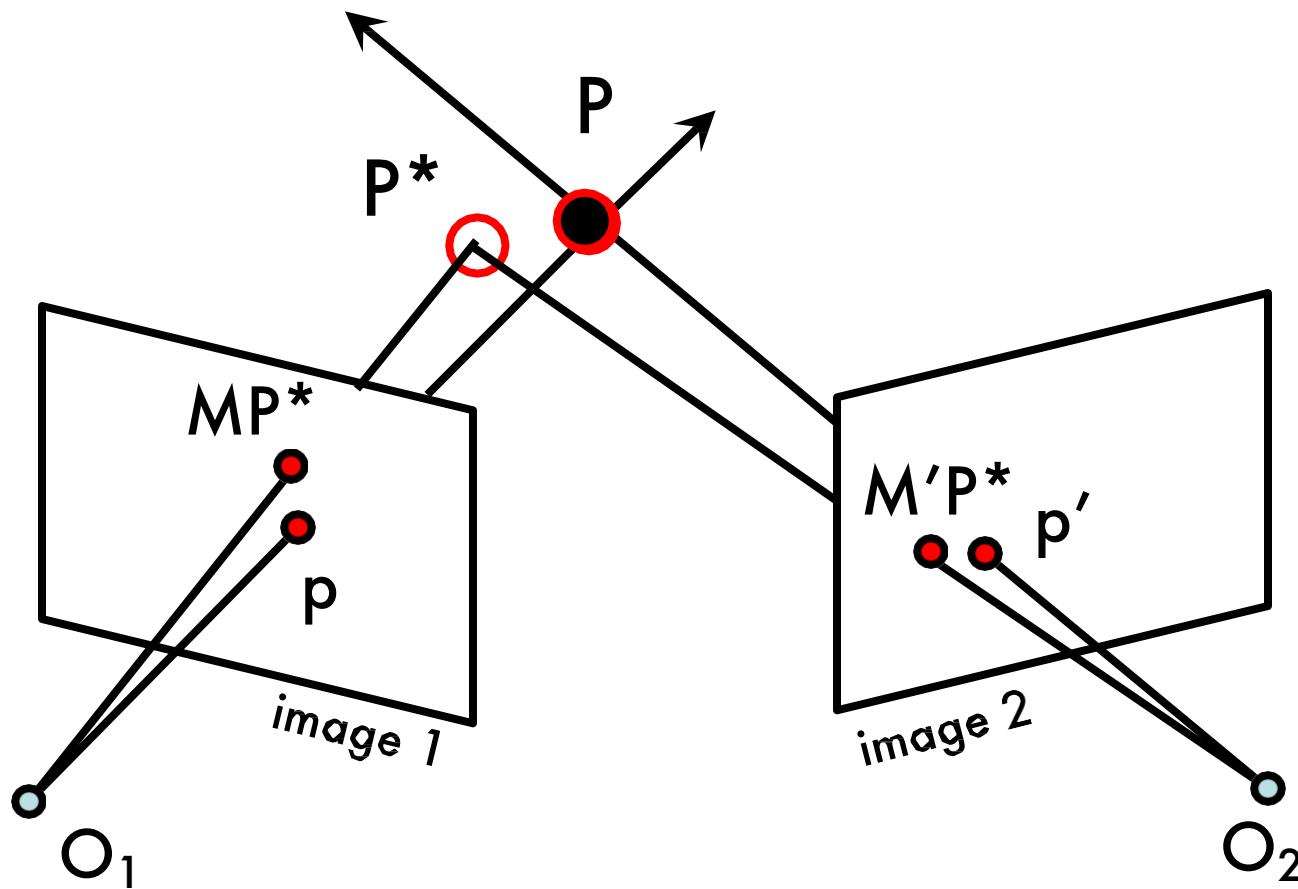


This is called **triangulation**

Triangulation

- Find P^* that minimizes

$$d(p, M P^*) + d(p', M' P^*) \quad [\text{Eq. 2}]$$



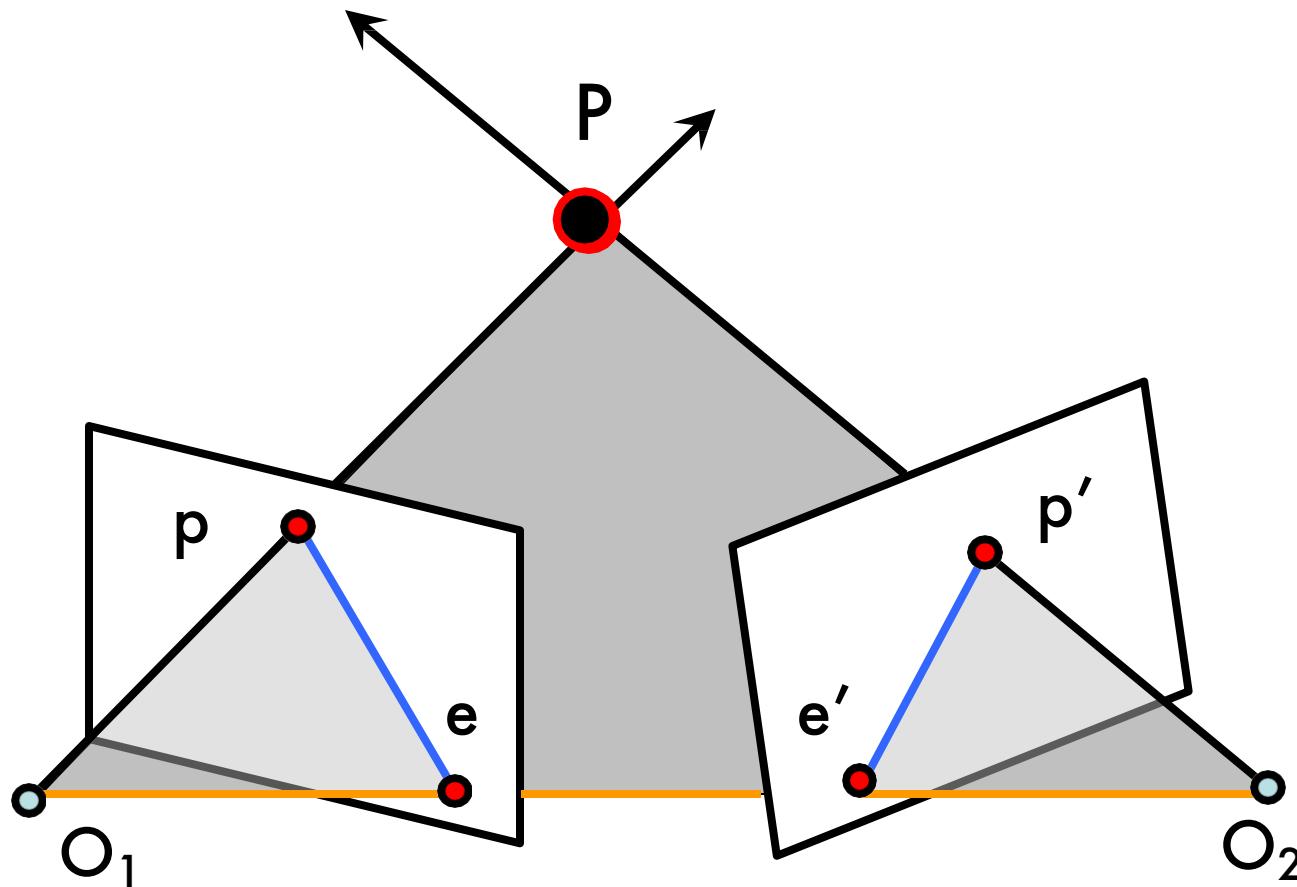
Multi (stereo)-view geometry

- **Camera geometry:** Given corresponding points in two images, find camera matrices, position and pose.
- **Scene geometry:** Find coordinates of 3D point from its projection into 2 or multiple images.
- **Correspondence:** Given a point p in one image, how can I find the corresponding point p' in another one?

Agenda

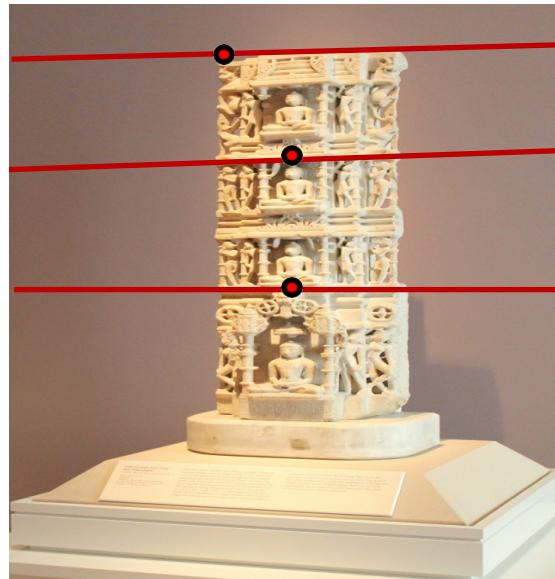
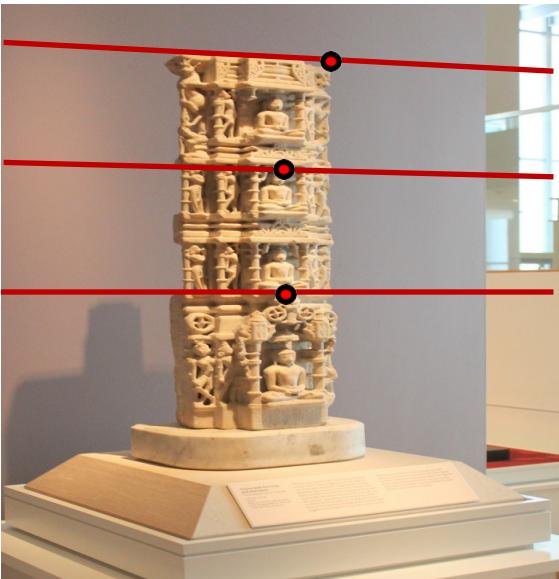
- Why is stereo useful?
- **Epipolar constraints**
- Fundamental matrix
- Estimating F

Epipolar geometry

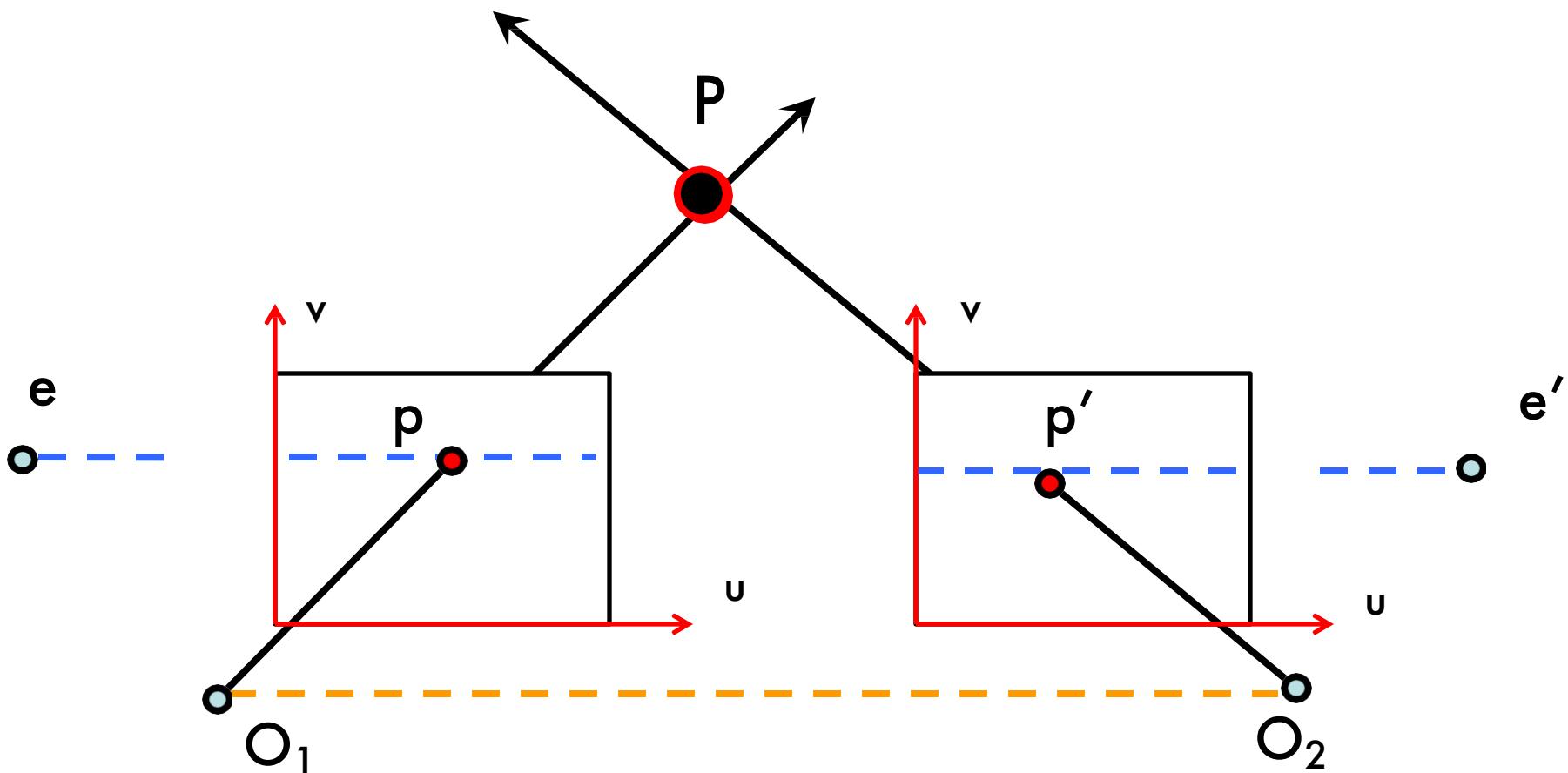


- Epipolar Plane
- Baseline
- Epipolar Lines
- Epipoles e, e'
 - = intersections of baseline with image planes
 - = projections of the other camera center

Example of epipolar lines



Example: Parallel image planes

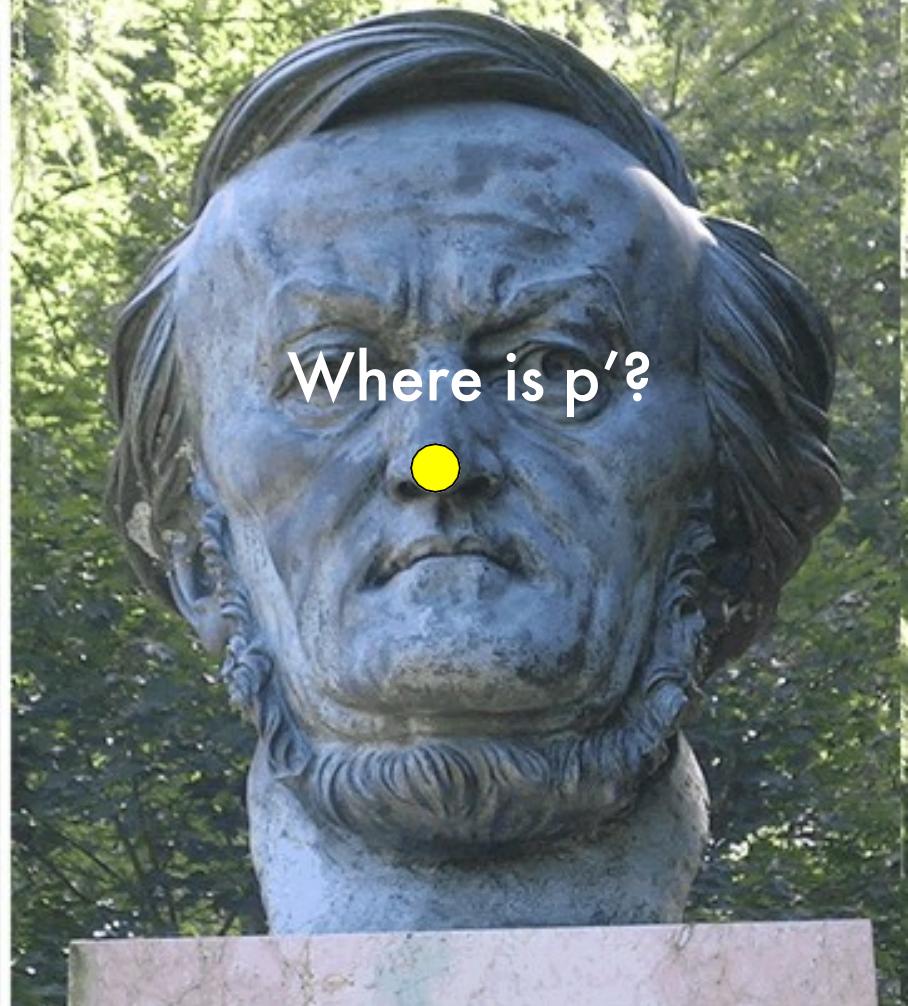


- Baseline intersects the image plane at infinity
- Epipoles are at infinity
- Epipolar lines are parallel to u axis

Example: Parallel Image Planes

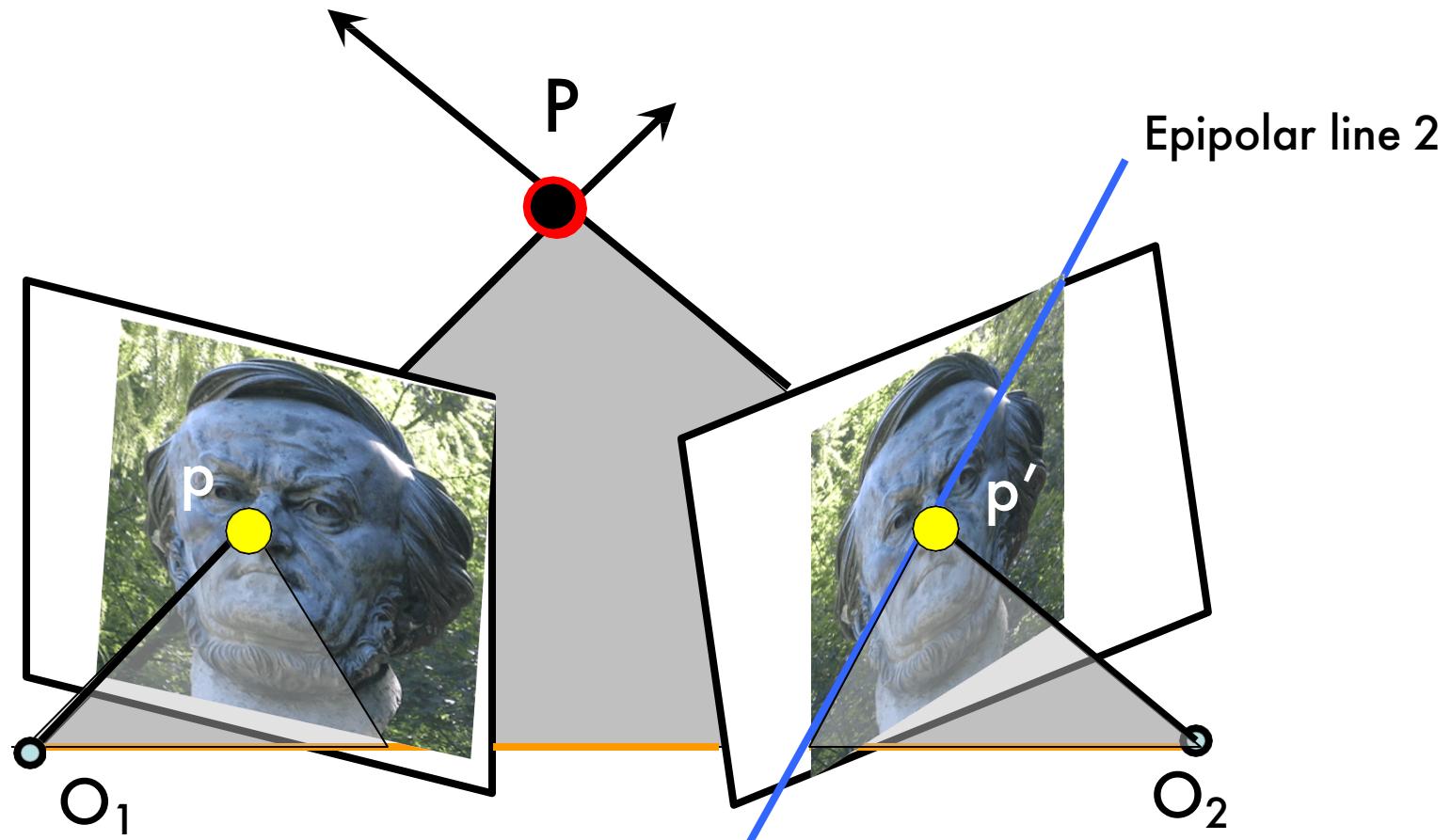


Epipolar Constraint

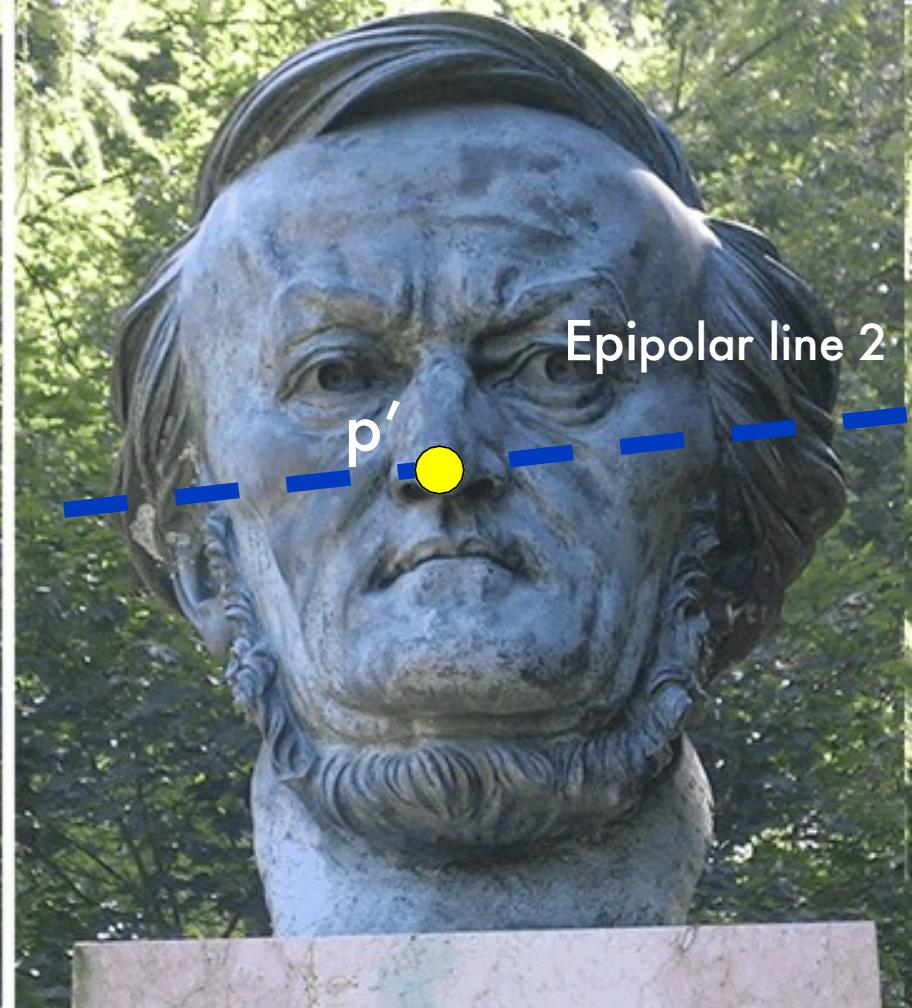


- Two views of the same object
- Given a point on left image, how can I find the corresponding point on right image?

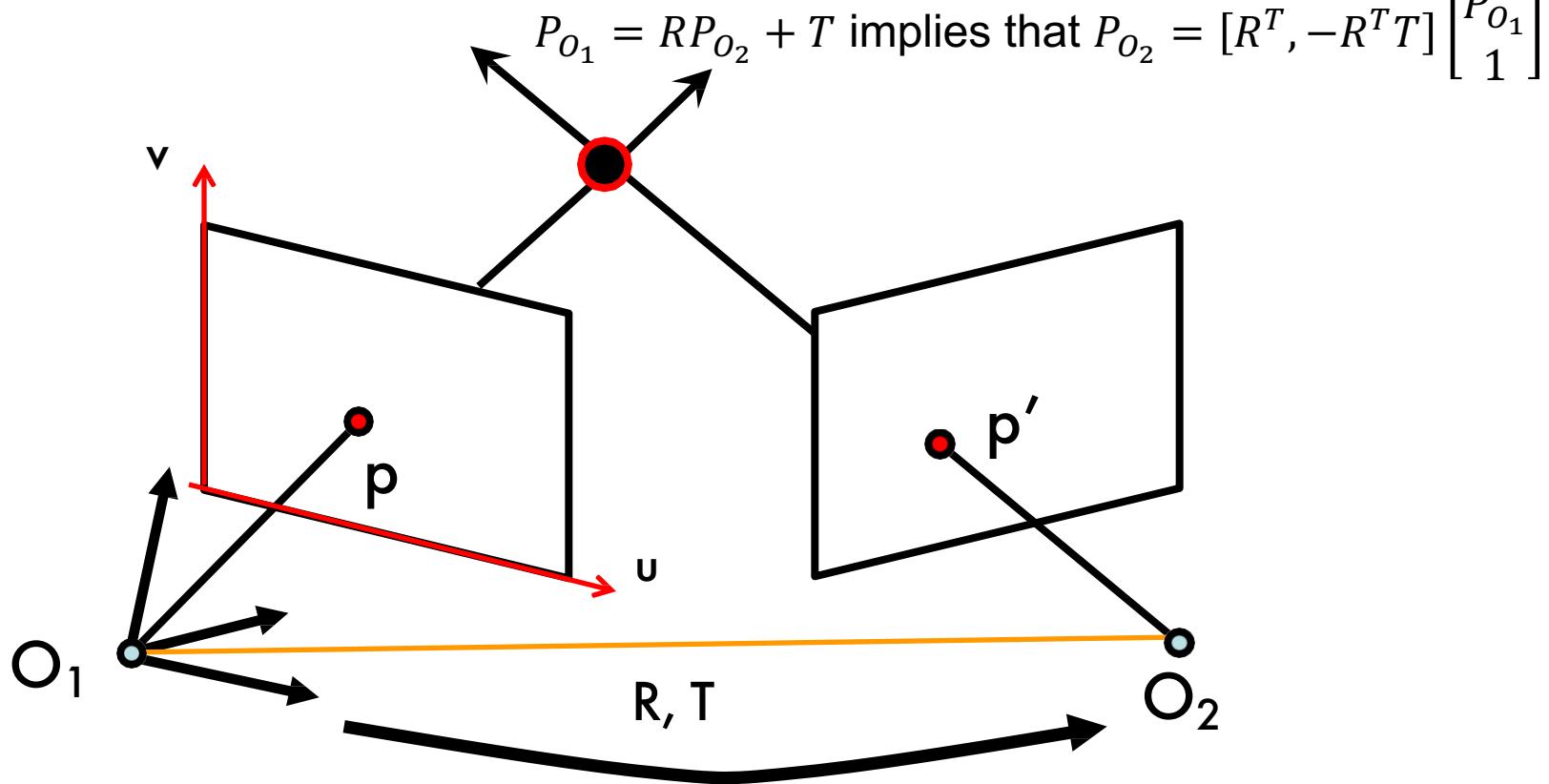
Epipolar geometry



Epipolar Constraint



Epipolar Constraint



$$M = K \begin{bmatrix} I & 0 \end{bmatrix}$$

$$M P = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = p \quad [\text{Eq. 3}]$$

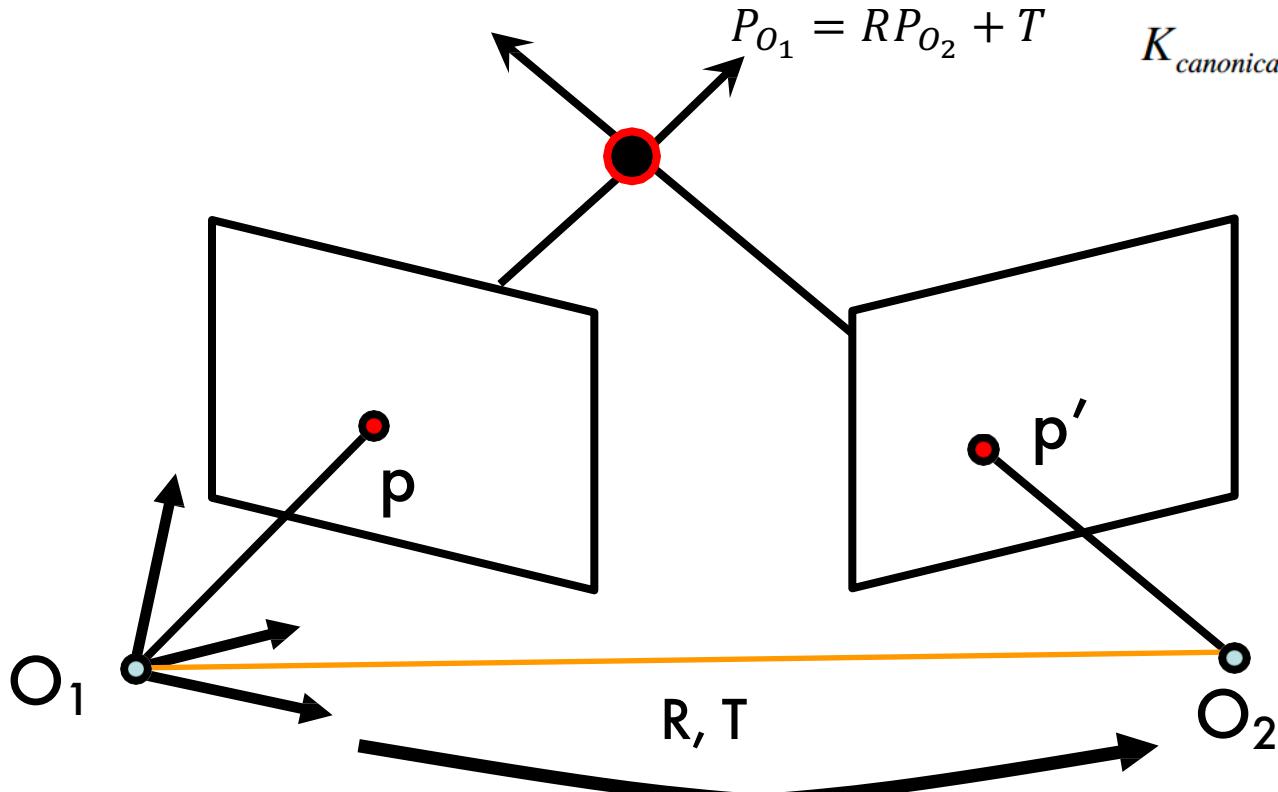
here, P means P_{O_1}

$$M' = K' \begin{bmatrix} R^T & -R^T T \end{bmatrix}$$

$$M' P = \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = p' \quad [\text{Eq. 4}]$$

here, P also means P_{O_1}

Epipolar Constraint



$$M = K \begin{bmatrix} I & 0 \end{bmatrix}$$

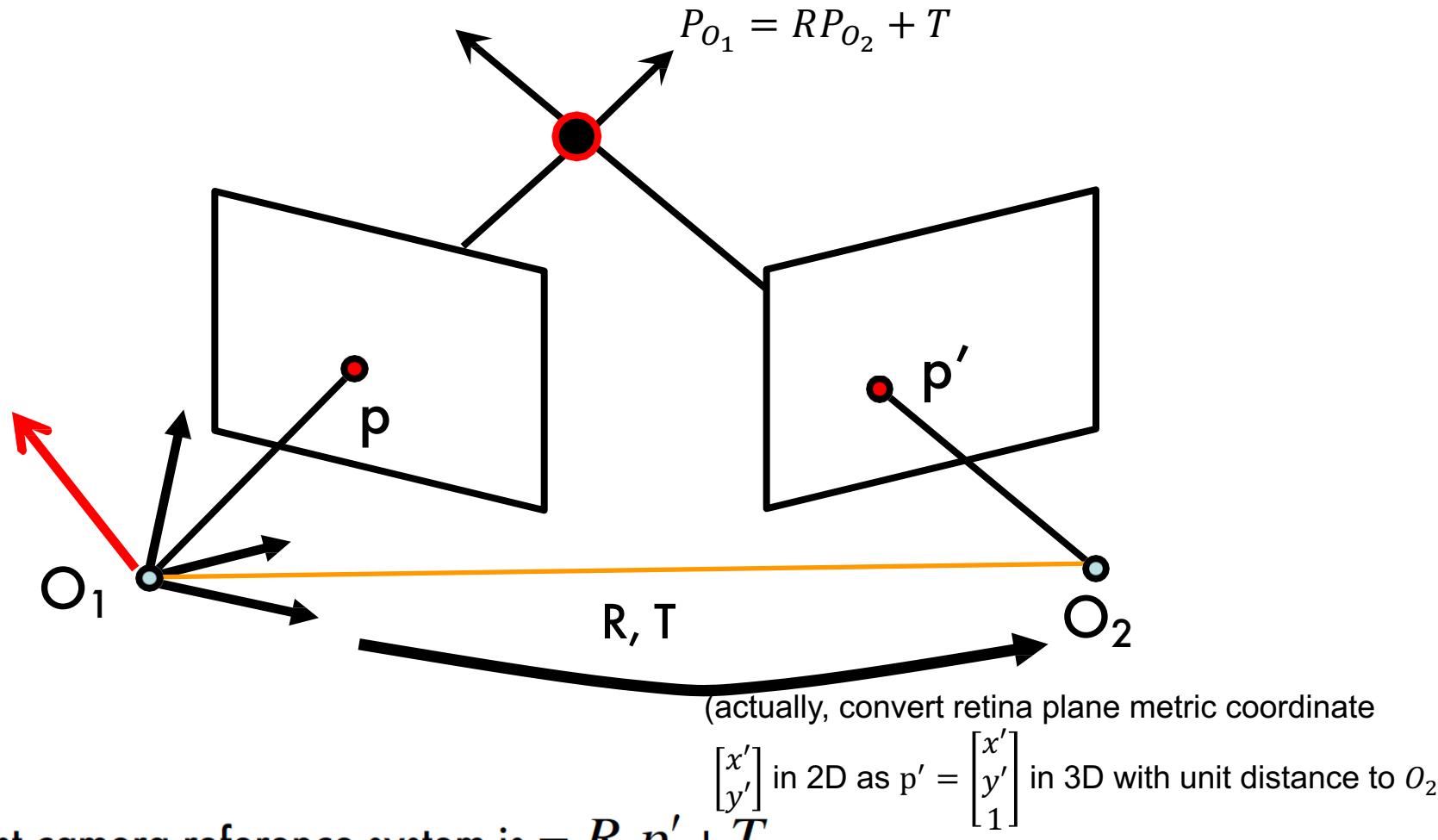
$K = K'$ are known
(canonical cameras)

$$M = [I \quad 0] \quad [\text{Eq. 5}]$$

$$M' = K' \begin{bmatrix} R^T & -R^T T \end{bmatrix}$$

$$M' = \begin{bmatrix} R^T & -R^T T \end{bmatrix} \quad [\text{Eq. 6}]$$

Epipolar Constraint



p' in first camera reference system is $= R p' + T$

$T \times ((R p') + T) = T \times (R p')$ is perpendicular to epipolar plane

$$\rightarrow p^T \cdot [T \times (R p')] = 0 \quad [\text{Eq. 7}]$$

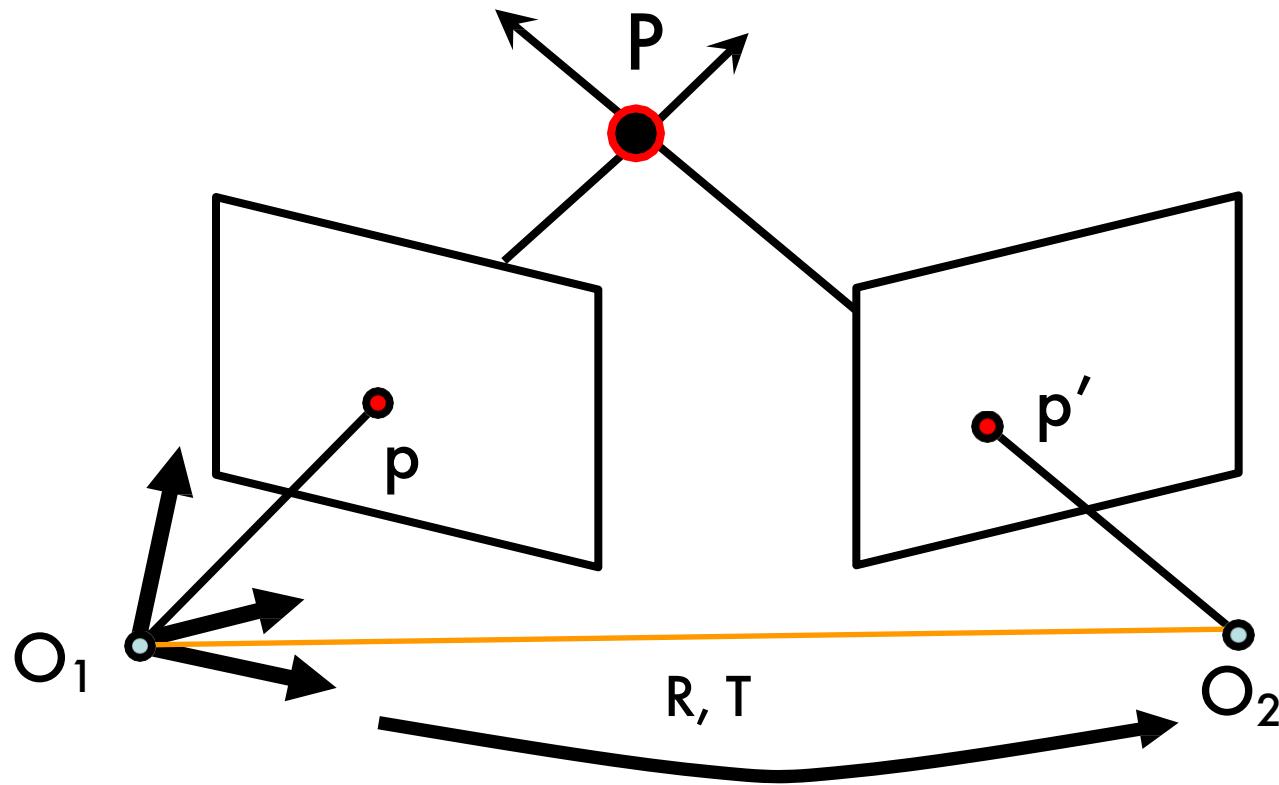
Cross product as matrix multiplication

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = [\mathbf{a}_\times] \mathbf{b}$$

$$[\mathbf{a}_\times] = -[\mathbf{a}_\times]^T$$

“skew-symmetric matrix”

Epipolar Constraint



$$p^T \cdot [T \times (R p')] = 0 \rightarrow$$

[Eq. 8]

$$p^T ([T \times] R) p' = 0$$

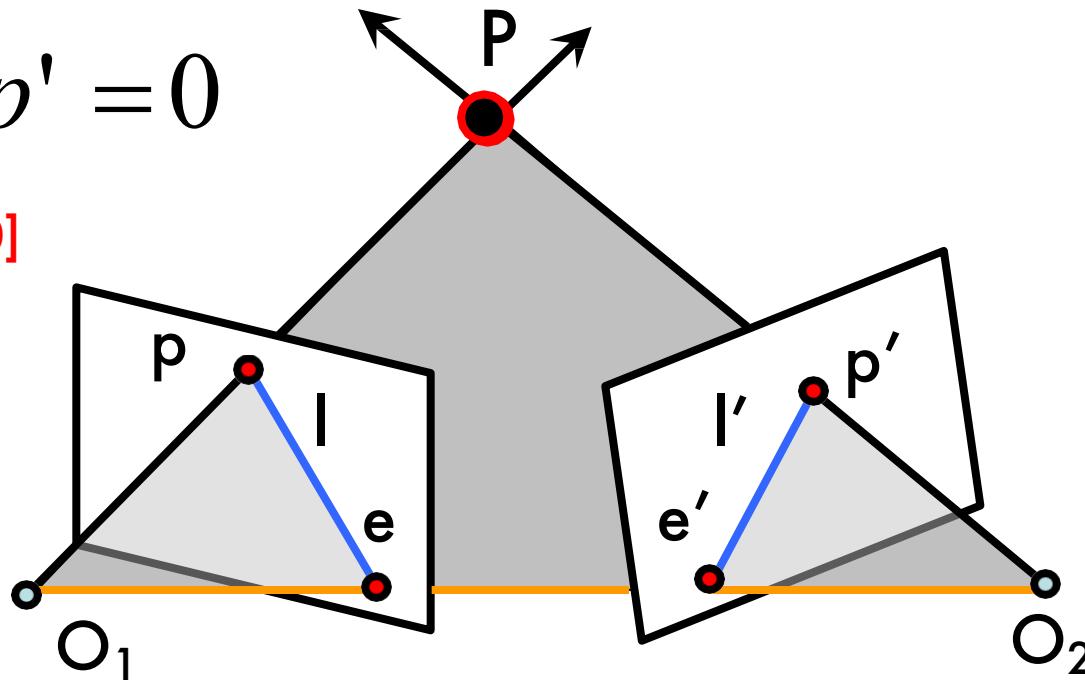
[Eq. 9]

E = Essential matrix
(Longuet-Higgins, 1981)

Epipolar Constraint

$$p^T \cdot E p' = 0$$

[Eq. 10]

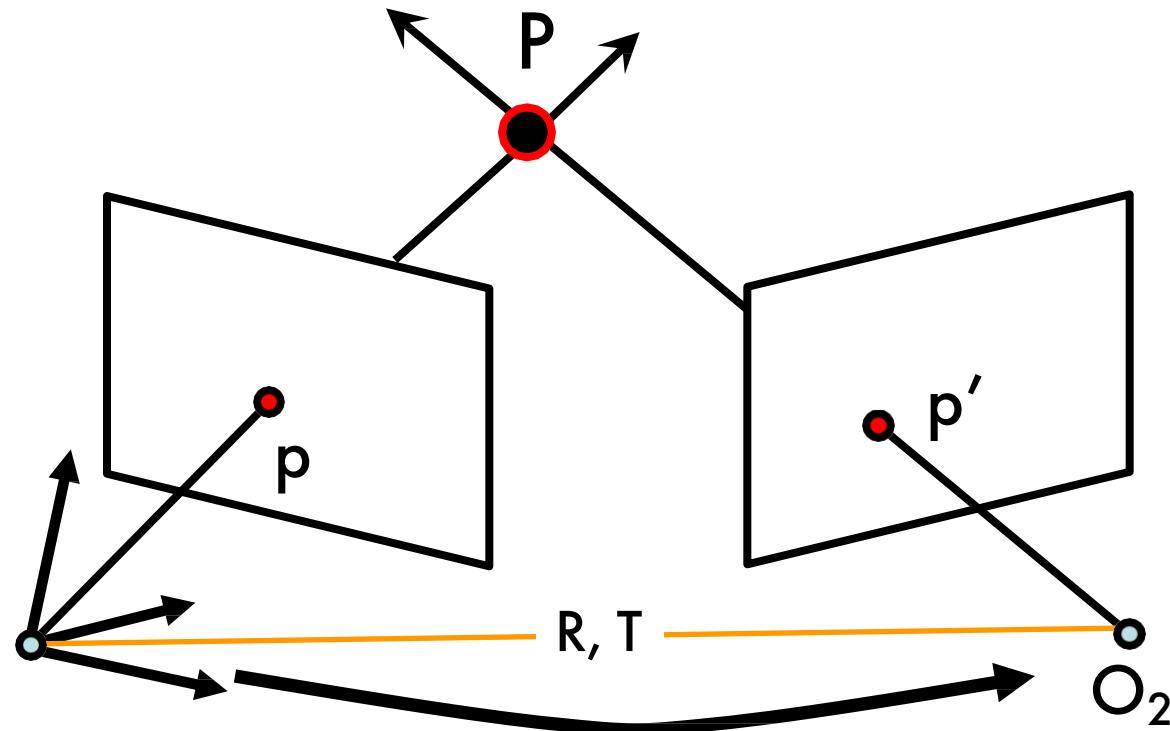


- $I = E p'$ is the epipolar line associated with p'
- $I' = E^T p$ is the epipolar line associated with p
- $E e' = 0$ and $E^T e = 0$
- E is 3x3 matrix; 5 DOF
- E is singular (rank two)

Agenda

- Why is stereo useful?
- Epipolar constraints
- **Fundamental matrix**
- Estimating F

Epipolar Constraint



$$M = K \begin{bmatrix} I & 0 \end{bmatrix}$$

“ideal” camera

$$p_c = K^{-1} p \quad [\text{Eq. 11}]$$

$$M' = K' \begin{bmatrix} R^T & -R^T T \end{bmatrix}$$

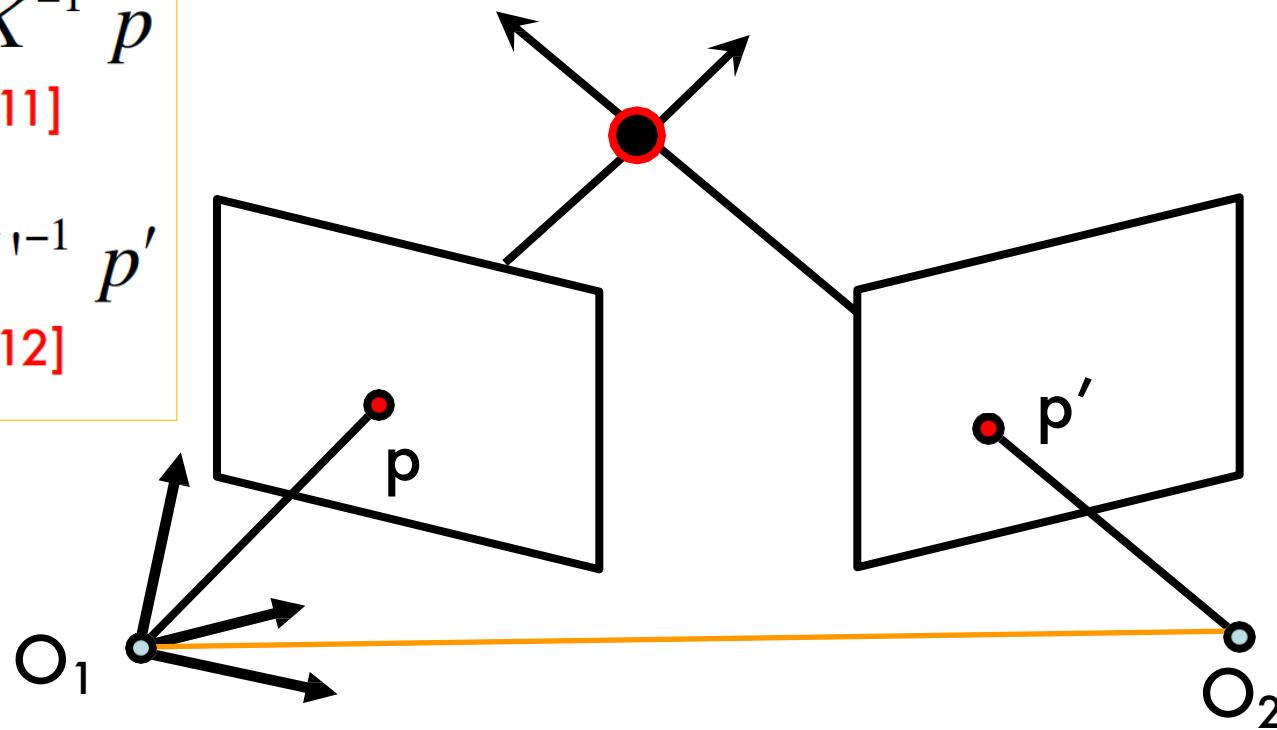
$$p'_c = K'^{-1} p' \quad [\text{Eq. 12}]$$

Goal: Cancel the adjustment made by K which was to accommodate the device

Epipolar Constraint

$$p_c = K^{-1} p \quad [\text{Eq. 11}]$$

$$p'_c = K'^{-1} p' \quad [\text{Eq. 12}]$$

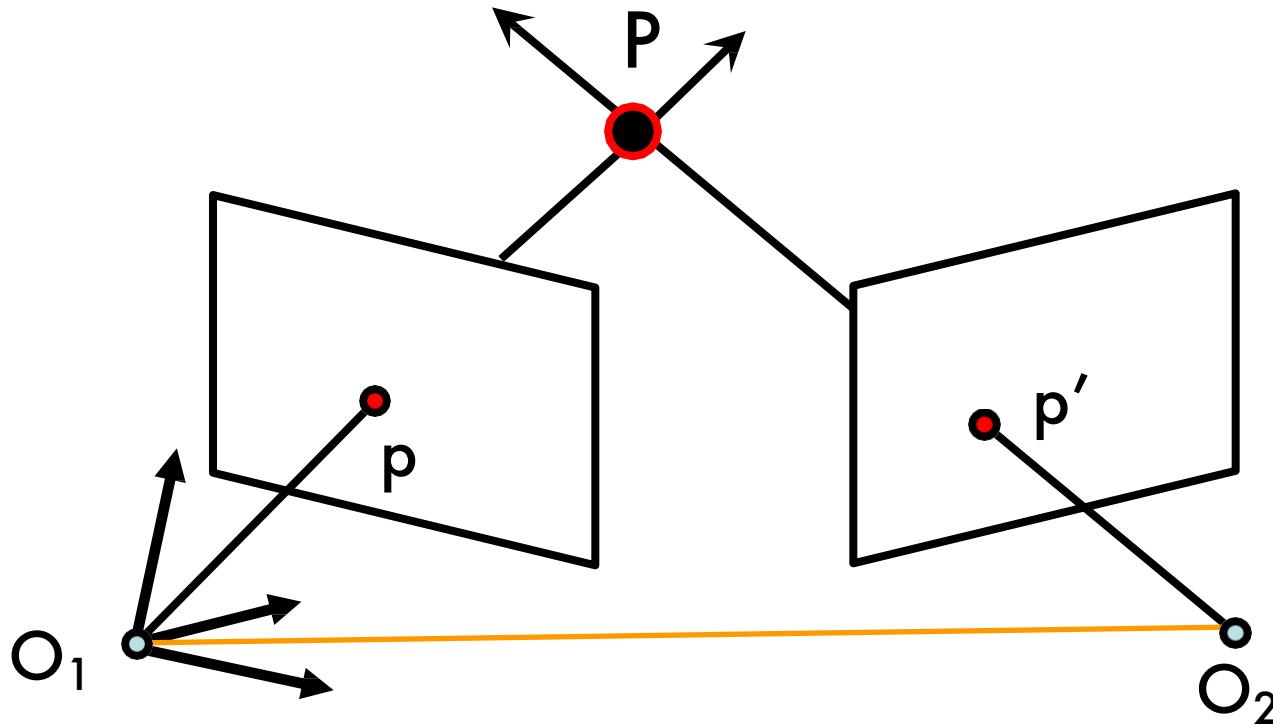


[Eq. 9]

$$p_c^T \cdot [T_x] \cdot R p'_c = 0 \rightarrow (K^{-1} p)^T \cdot [T_x] \cdot R K'^{-1} p' = 0$$

$$p^T \boxed{K^{-T} \cdot [T_x] \cdot R K'^{-1}} p' = 0 \rightarrow p^T \boxed{F} p' = 0 \quad [\text{Eq. 13}]$$

Epipolar Constraint



[Eq. 13]

$$p^T F p' = 0$$

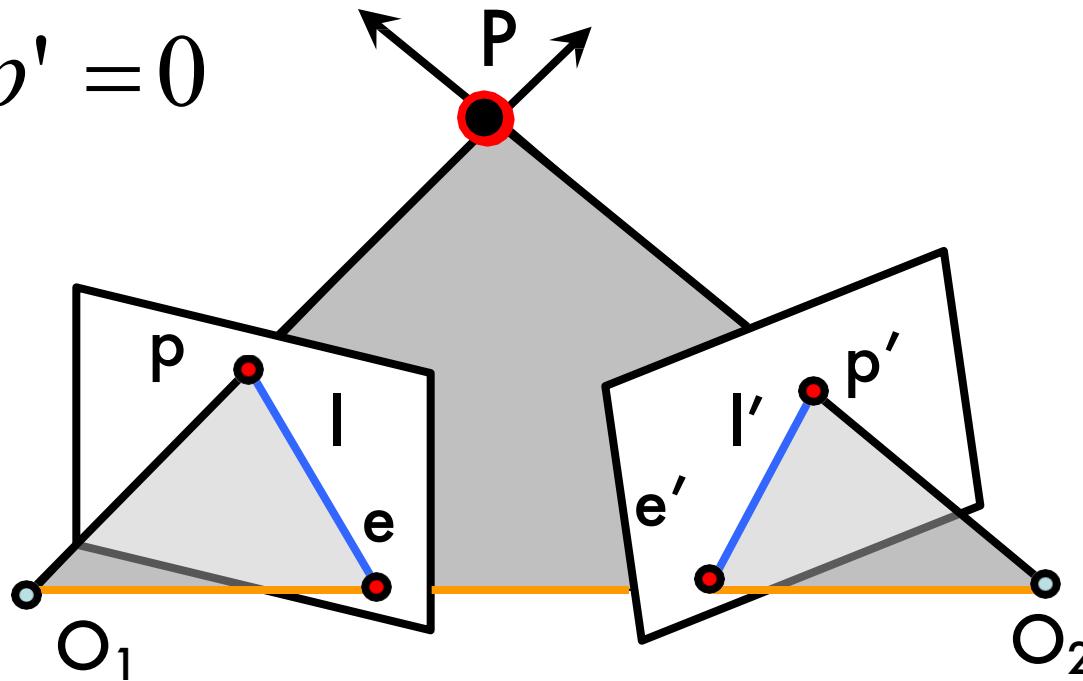
$$F = K^{-T} \cdot [T_x] \cdot R \cdot K'^{-1}$$

F = Fundamental Matrix
(Faugeras and Luong, 1992)

[Eq. 14]

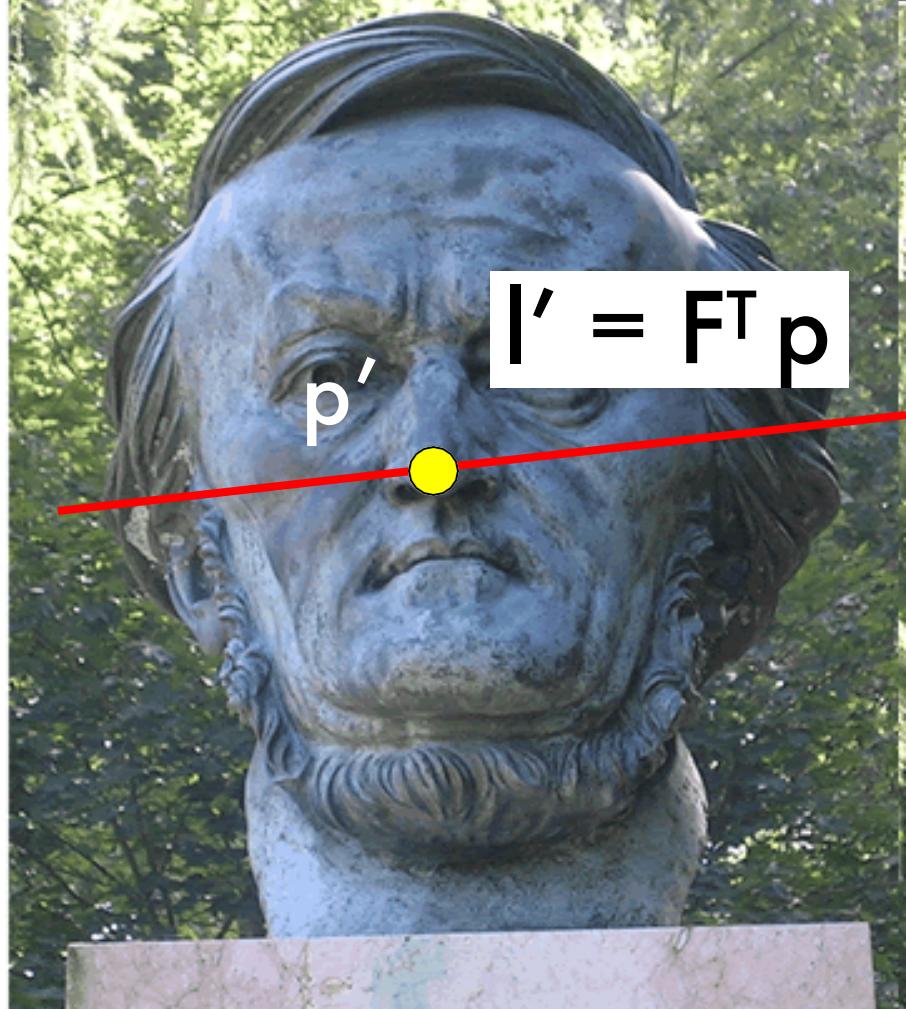
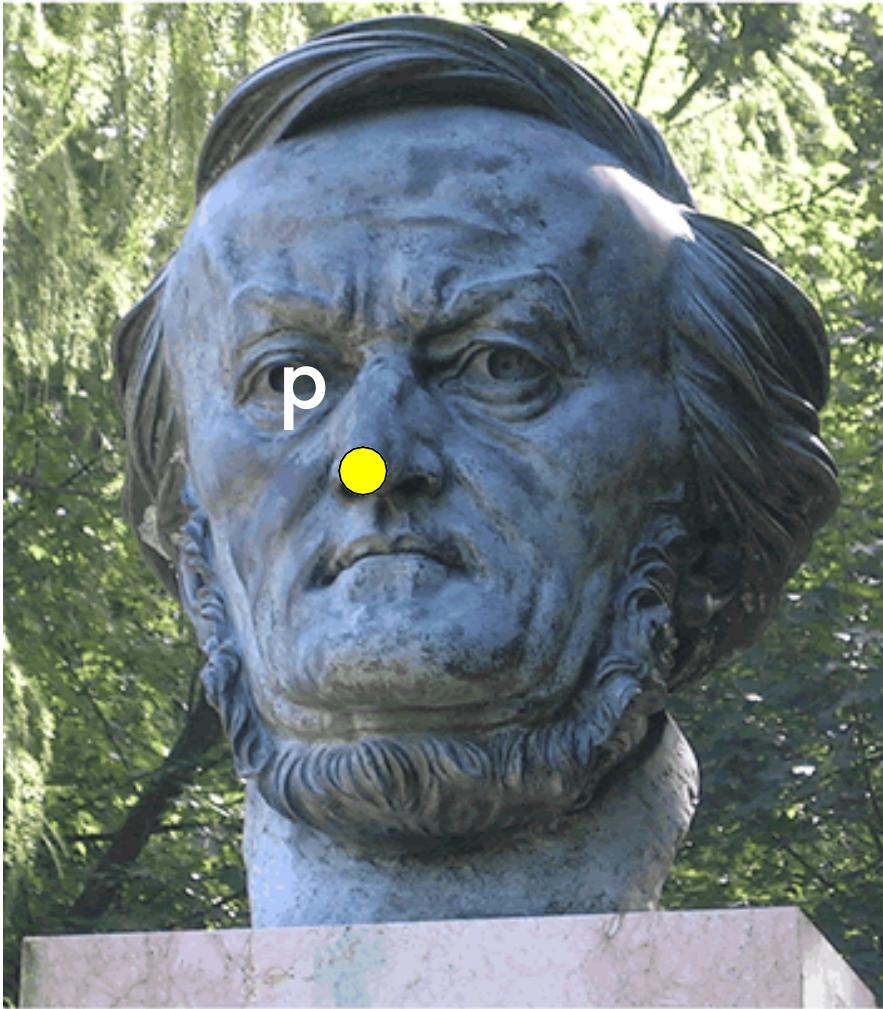
Epipolar Constraint

$$p^T \cdot F p' = 0$$



- $I = F p'$ is the epipolar line associated with p'
- $I' = F^T p$ is the epipolar line associated with p
- $F e' = 0$ and $F^T e = 0$
- F is 3x3 matrix; 7 DOF
- F is singular (rank two)

Why F is useful?



- Suppose F is known
- No additional information about the scene and camera is given
- Given a point on left image, we can compute the corresponding epipolar line in the second image

Why F is useful?

- F captures information about the epipolar geometry of 2 views + camera parameters
- **MORE IMPORTANTLY:** F gives constraints on how the scene changes under view point transformation (without reconstructing the scene!)
- Powerful tool in:
 - 3D reconstruction
 - Multi-view object/scene matching

Agenda

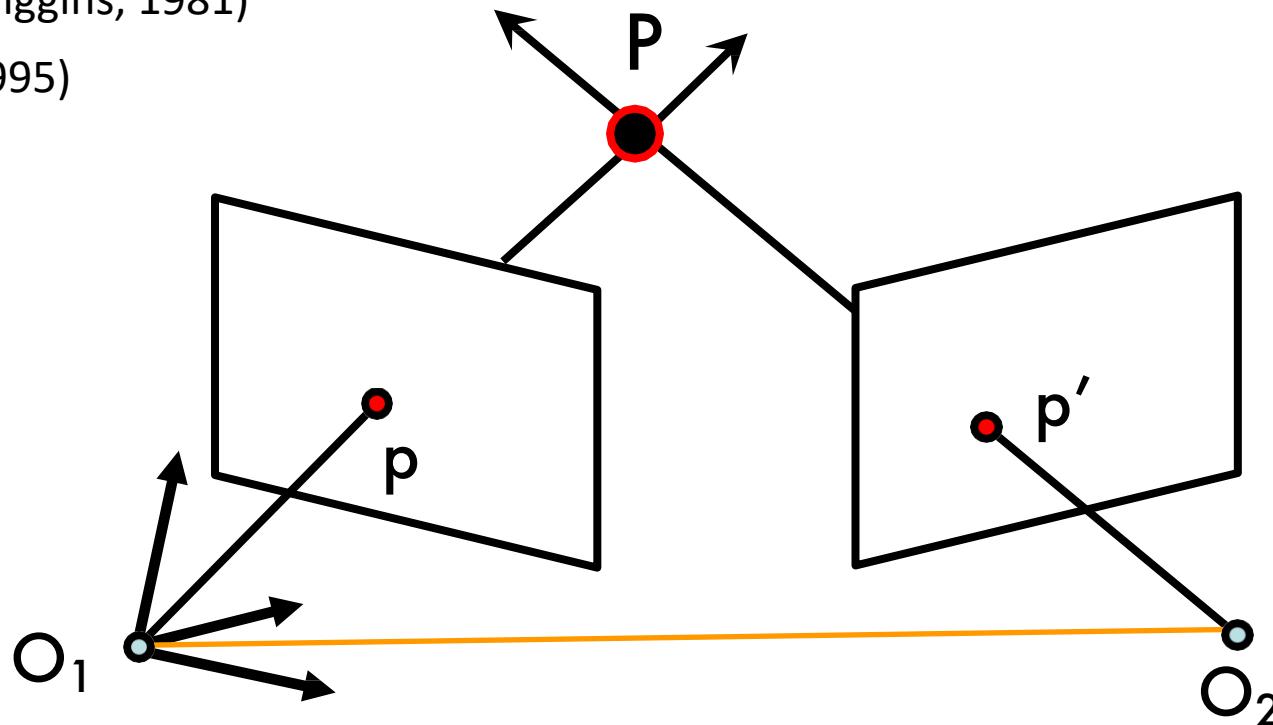
- Why is stereo useful?
- Epipolar constraints
- Fundamental matrix
- **Estimating F**

Estimating F

The Eight-Point Algorithm

(Longuet-Higgins, 1981)

(Hartley, 1995)



$$p^T F p' = 0$$

Estimating F

$$[\text{Eq. 13}] \quad p^T F p' = 0 \quad \rightarrow$$

$$p = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad p' = \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix}$$

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



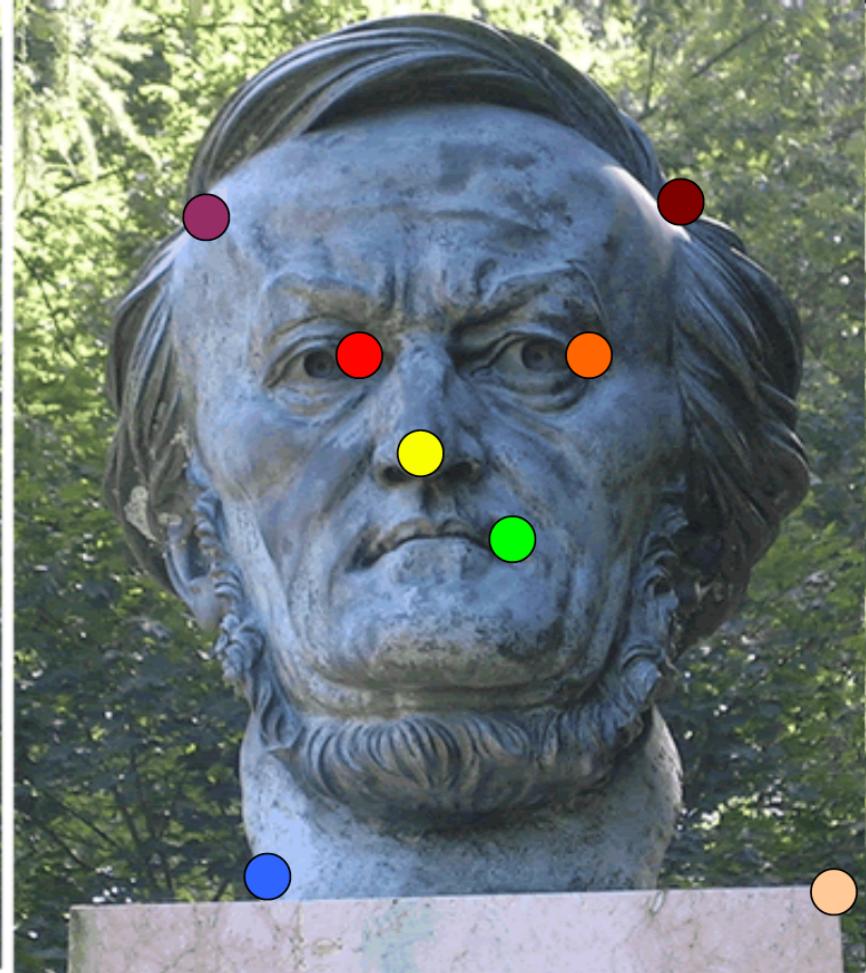
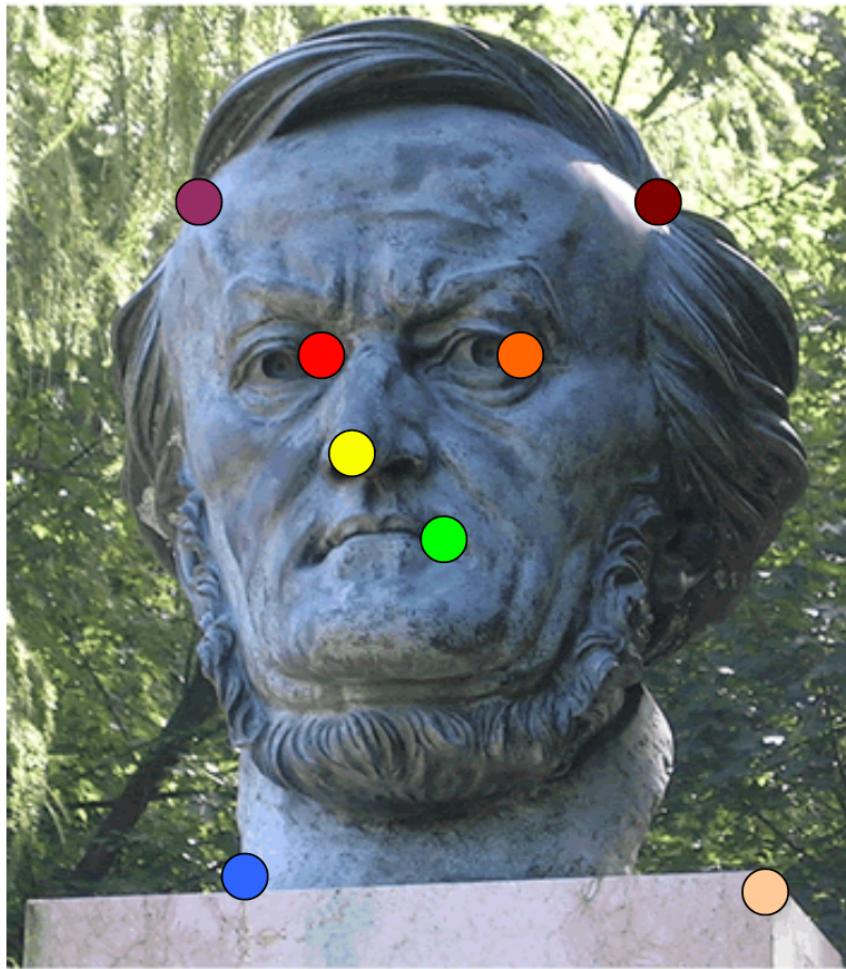
$$(uu', uv', u, vu', vv', v, u', v', 1)$$

$$\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

[Eq. 14]

Let's take 8 corresponding points

Estimating F



Estimating F

$$\begin{pmatrix} u_i u'_i, u_i v'_i, u_i, v_i u'_i, v_i v'_i, v_i, u'_i, v'_i, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0 \quad [\text{Eq. 14}]$$

Estimating F

$$\mathbf{W} \begin{pmatrix} u_1 u'_1 & u_1 v'_1 & u_1 & v_1 u'_1 & v_1 v'_1 & v_1 & u'_1 & v'_1 & 1 \\ u_2 u'_2 & u_2 v'_2 & u_2 & v_2 u'_2 & v_2 v'_2 & v_2 & u'_2 & v'_2 & 1 \\ u_3 u'_3 & u_3 v'_3 & u_3 & v_3 u'_3 & v_3 v'_3 & v_3 & u'_3 & v'_3 & 1 \\ u_4 u'_4 & u_4 v'_4 & u_4 & v_4 u'_4 & v_4 v'_4 & v_4 & u'_4 & v'_4 & 1 \\ u_5 u'_5 & u_5 v'_5 & u_5 & v_5 u'_5 & v_5 v'_5 & v_5 & u'_5 & v'_5 & 1 \\ u_6 u'_6 & u_6 v'_6 & u_6 & v_6 u'_6 & v_6 v'_6 & v_6 & u'_6 & v'_6 & 1 \\ u_7 u'_7 & u_7 v'_7 & u_7 & v_7 u'_7 & v_7 v'_7 & v_7 & u'_7 & v'_7 & 1 \\ u_8 u'_8 & u_8 v'_8 & u_8 & v_8 u'_8 & v_8 v'_8 & v_8 & u'_8 & v'_8 & 1 \end{pmatrix} \mathbf{f} = 0 \quad [\text{Eqs. 15}]$$

- Homogeneous system $\mathbf{W f} = 0$
- Rank 8 \rightarrow A non-zero solution exists (unique)
- If $N > 8 \rightarrow$ Lsq. solution by SVD! $\rightarrow \hat{\mathbf{F}}$
 $\|\mathbf{f}\| = 1$

\hat{F} satisfies: $p^T \hat{F} p' = 0$

and estimated \hat{F} may have full rank ($\det(\hat{F}) \neq 0$)

But remember: fundamental matrix is Rank2

Find F that minimizes

$$\|F - \hat{F}\| = 0$$

Frobenius norm (*)

Subject to $\text{rank}(F)=2$

SVD (again!) can be used to solve this problem

(*) Sq. root of the sum of squares of all entries

Find F that minimizes

$$\|F - \hat{F}\| = 0$$

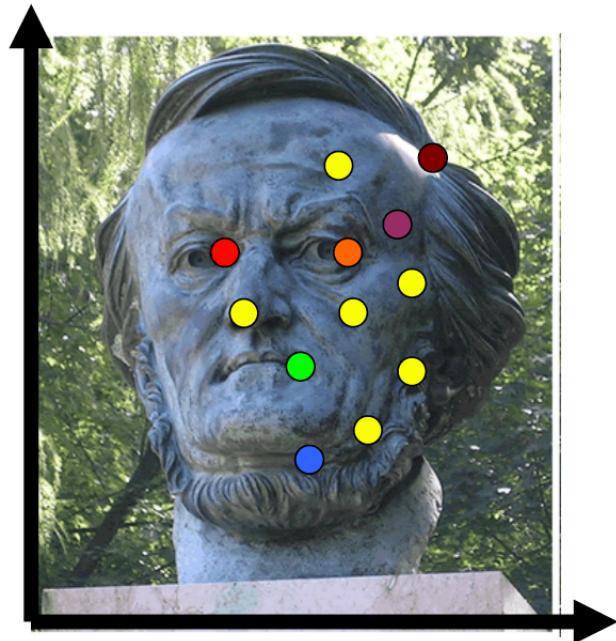
Frobenius norm (*)

Subject to $\det(F) = 0$

$$F = U \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T \quad \text{Where:}$$

$$U \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{bmatrix} V^T = SVD(\hat{F})$$

Problems with the 8-Point Algorithm



$$\mathbf{W} \mathbf{f} = 0, \quad \xrightarrow{\text{Lsq solution by SVD}} \mathbf{F}$$
$$\|\mathbf{f}\| = 1$$

- Recall the structure of \mathbf{W} :
 - do we see any potential (numerical) issue?

Problems with the 8-Point Algorithm

$$\mathbf{W}\mathbf{f} = 0$$

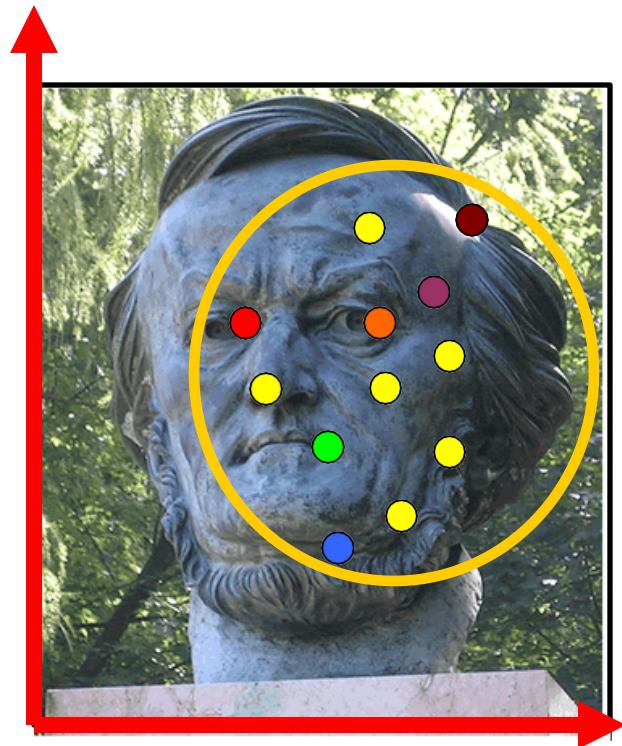
$$\begin{pmatrix} u_1 u'_1 & u_1 v'_1 & u_1 & v_1 u'_1 & v_1 v'_1 & v_1 & u'_1 & v'_1 & 1 \\ u_2 u'_2 & u_2 v'_2 & u_2 & v_2 u'_2 & v_2 v'_2 & v_2 & u'_2 & v'_2 & 1 \\ u_3 u'_3 & u_3 v'_3 & u_3 & v_3 u'_3 & v_3 v'_3 & v_3 & u'_3 & v'_3 & 1 \\ u_4 u'_4 & u_4 v'_4 & u_4 & v_4 u'_4 & v_4 v'_4 & v_4 & u'_4 & v'_4 & 1 \\ u_5 u'_5 & u_5 v'_5 & u_5 & v_5 u'_5 & v_5 v'_5 & v_5 & u'_5 & v'_5 & 1 \\ u_6 u'_6 & u_6 v'_6 & u_6 & v_6 u'_6 & v_6 v'_6 & v_6 & u'_6 & v'_6 & 1 \\ u_7 u'_7 & u_7 v'_7 & u_7 & v_7 u'_7 & v_7 v'_7 & v_7 & u'_7 & v'_7 & 1 \\ u_8 u'_8 & u_8 v'_8 & u_8 & v_8 u'_8 & v_8 v'_8 & v_8 & u'_8 & v'_8 & 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

- Highly un-balanced (not well conditioned)
- Values of W must have similar magnitude
- This creates problems during the SVD decomposition

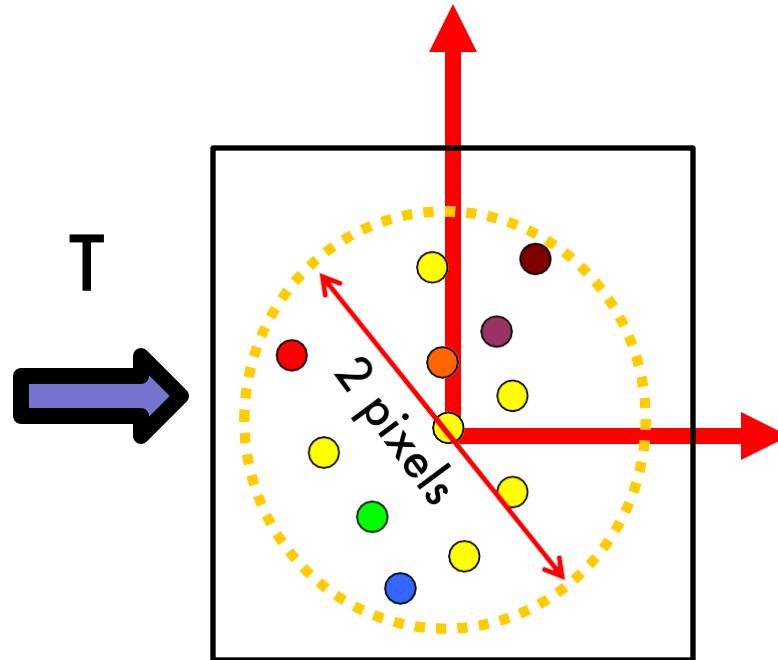
Normalization

- IDEA: Transform image coordinates such that the matrix W becomes better conditioned (pre-conditioning)
- For each image, apply a transformation T (translation and scaling) acting on image coordinates such that:
 - Origin = centroid of image points
 - Mean square distance of the image points from origin is ~ 2 pixels

Example of normalization



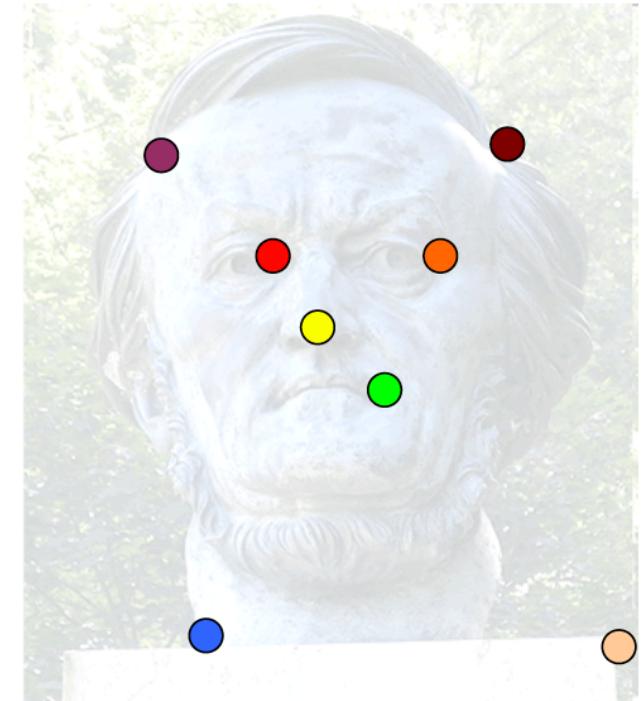
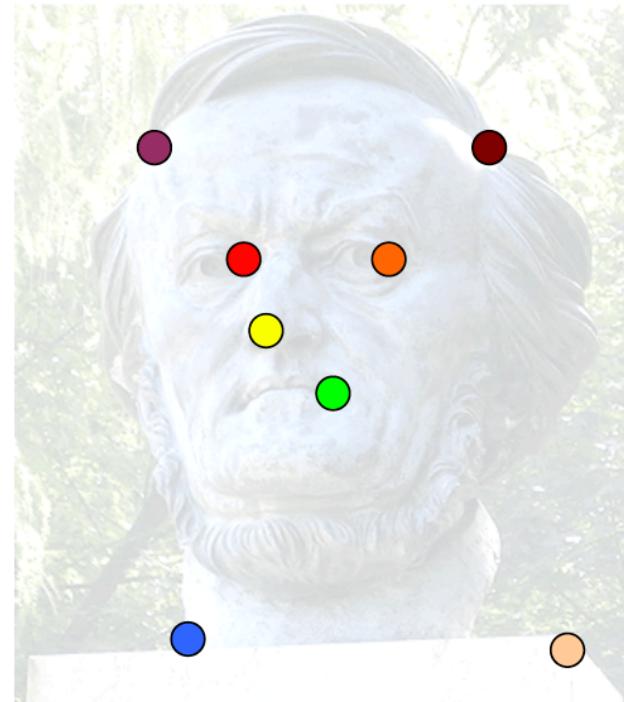
Coordinate system of the image before applying T



Coordinate system of the image after applying T

- Origin = centroid of image points
- Mean square distance of the image points from origin is ~2 pixels

Normalization



$$q_i = T \ p_i$$

$$q'_i = T' \ p'_i$$

The Normalized Eight-Point Algorithm

Compute T and T' for image 1 and 2, respectively

Normalize coordinates in images 1 and 2:

$$q_i = T p_i \quad q'_i = T' p'_i$$

Use the eight-point algorithm to compute \hat{F}_q from the corresponding points q_i and q'_i .

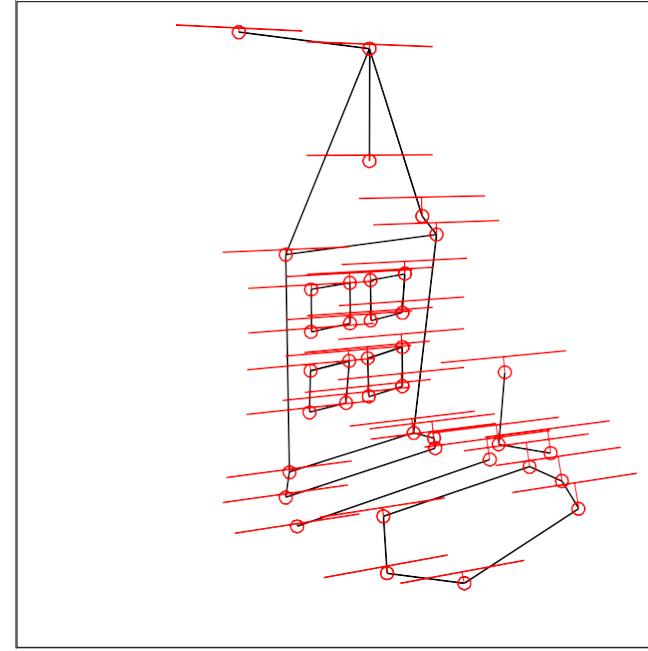
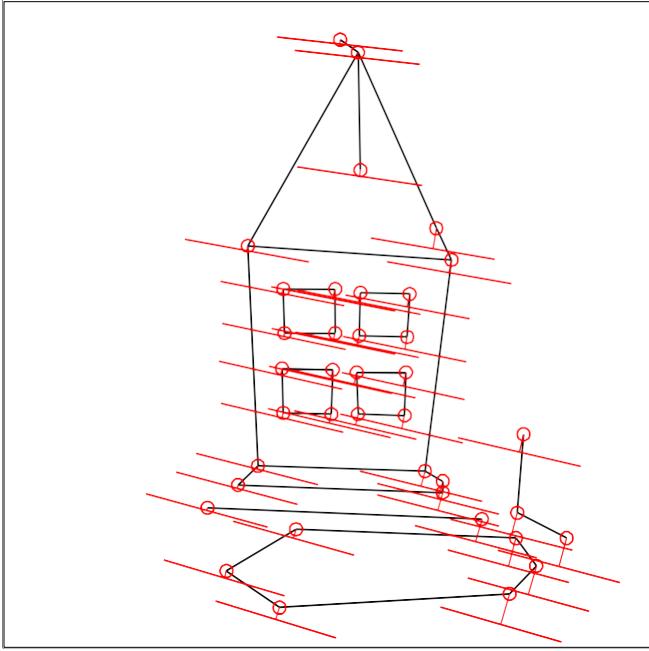
Enforce the rank-2 constraint: $\rightarrow F_q$ such that:

De-normalize F_q :

$$F = T^T F_q T'$$

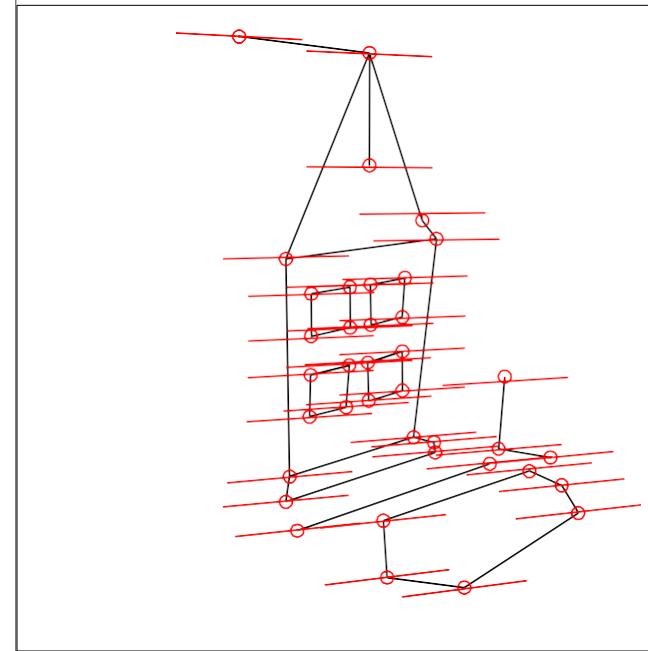
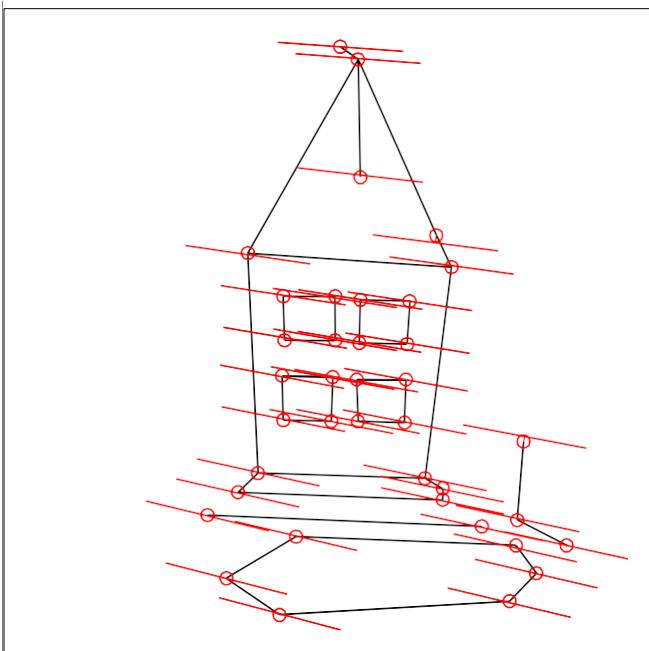
$$\begin{cases} q^T F_q q' = 0 \\ \det(F_q) = 0 \end{cases}$$

Without normalization



Mean errors:
10.0pixel
9.1pixel

With normalization



Mean errors:
1.0pixel
0.9pixel