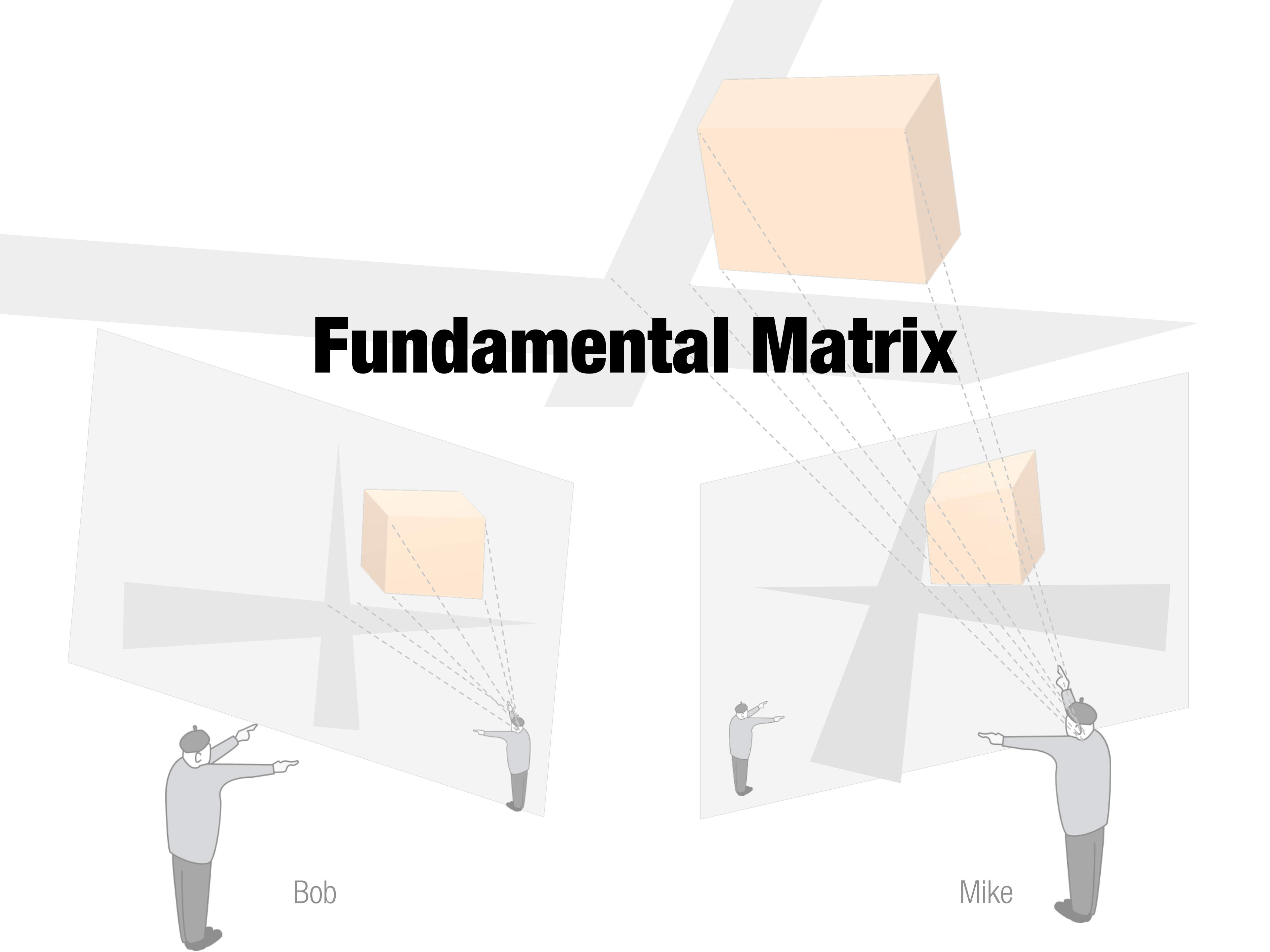
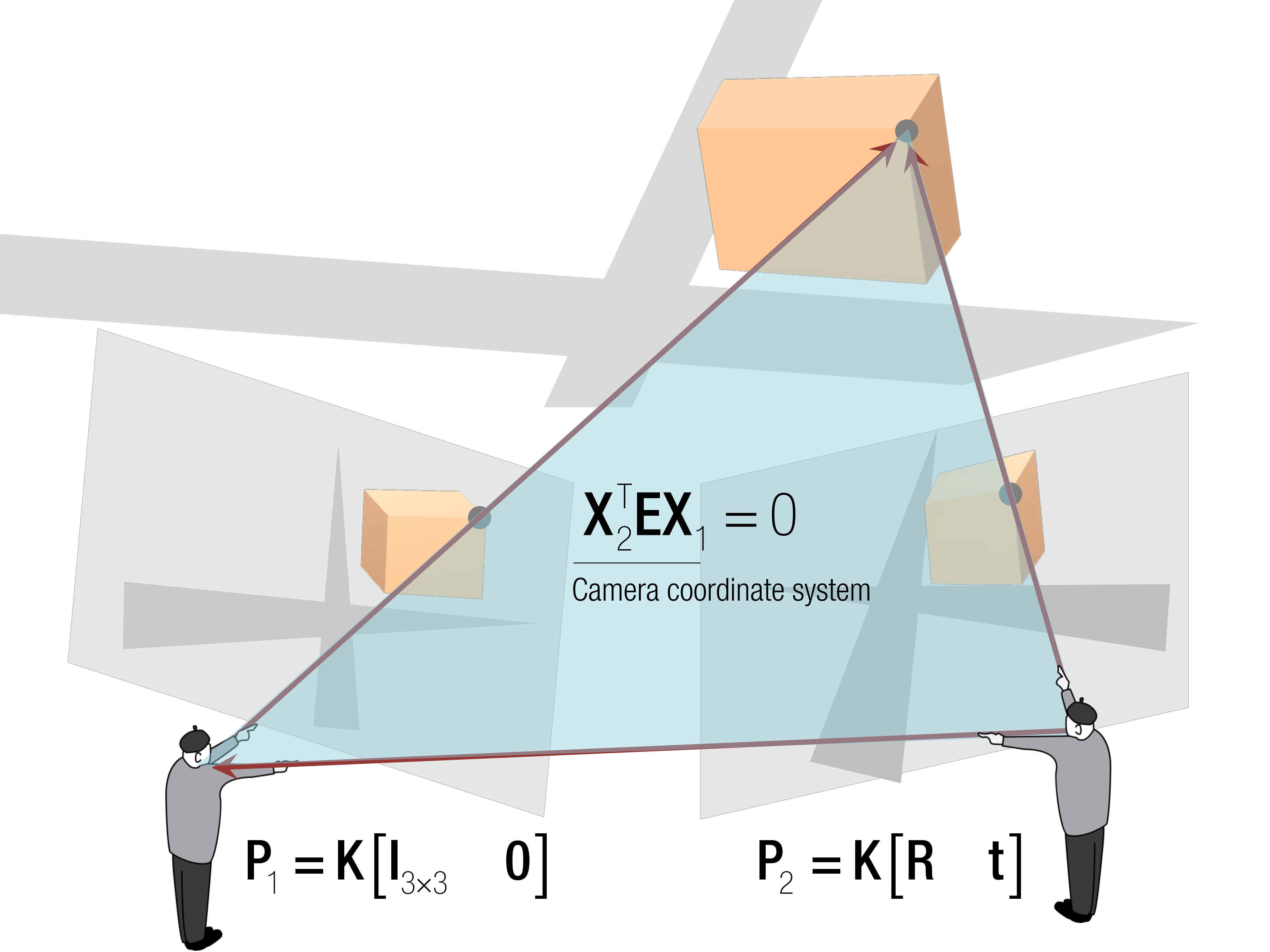
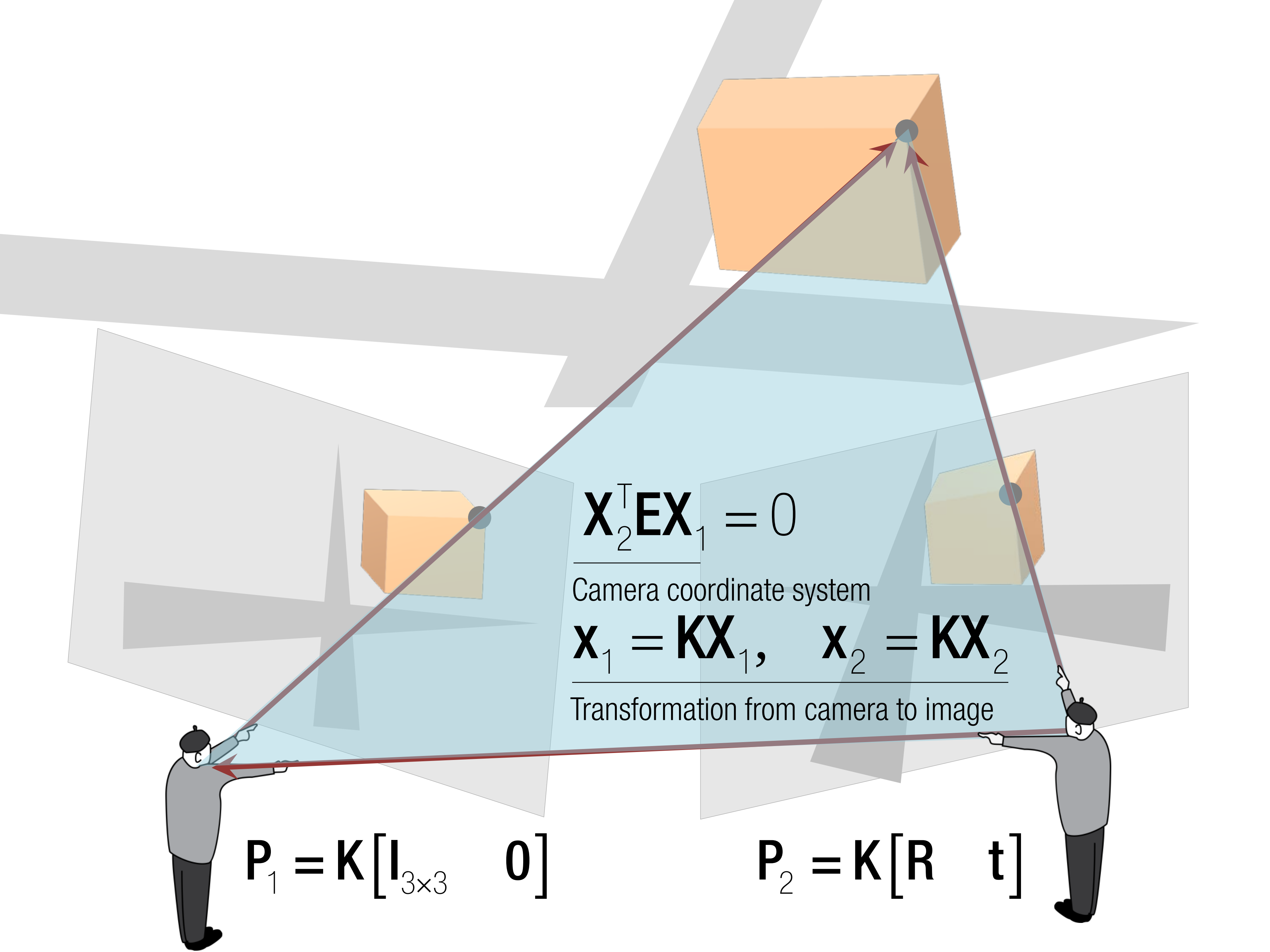
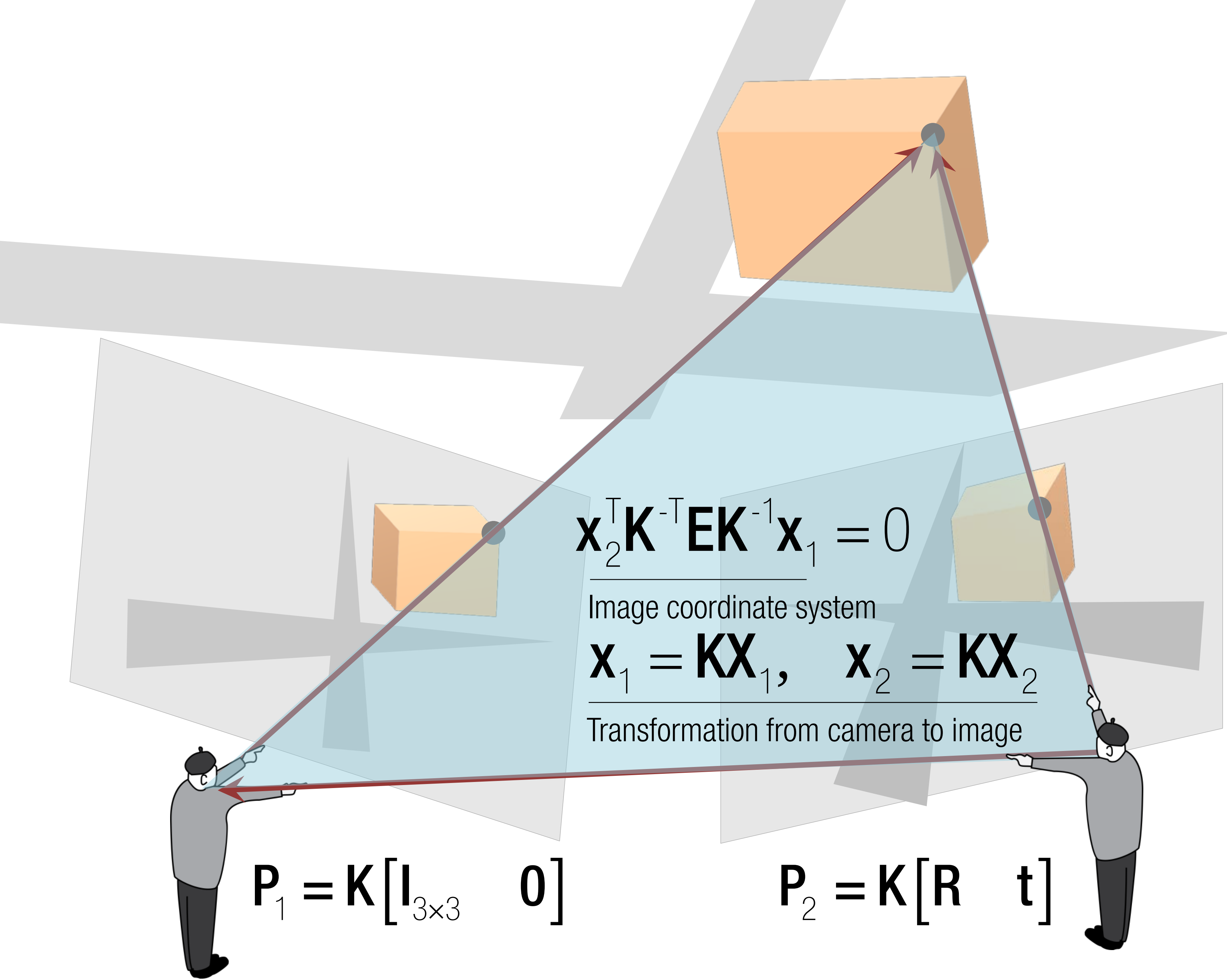
# Fundamental Matrix

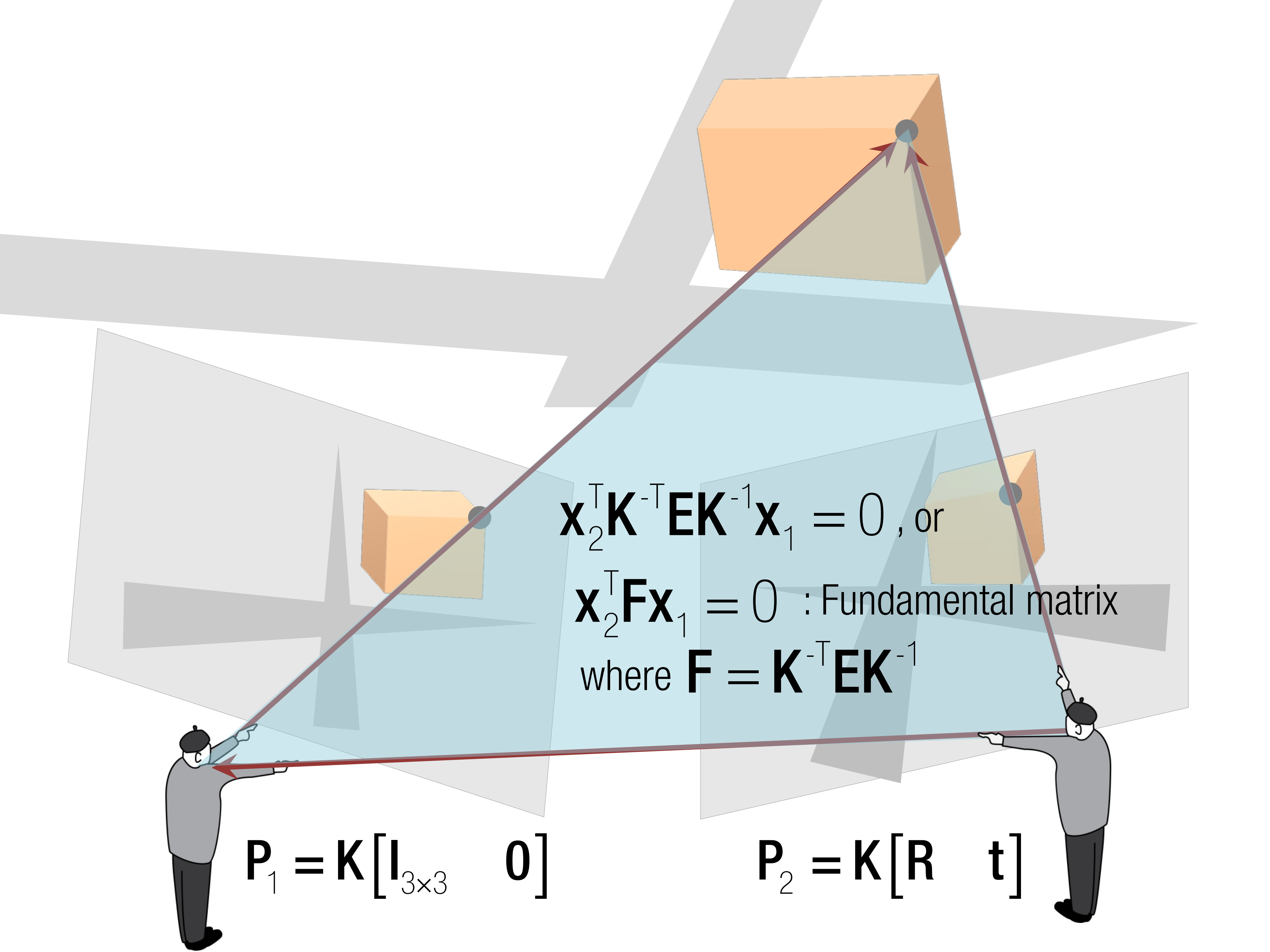


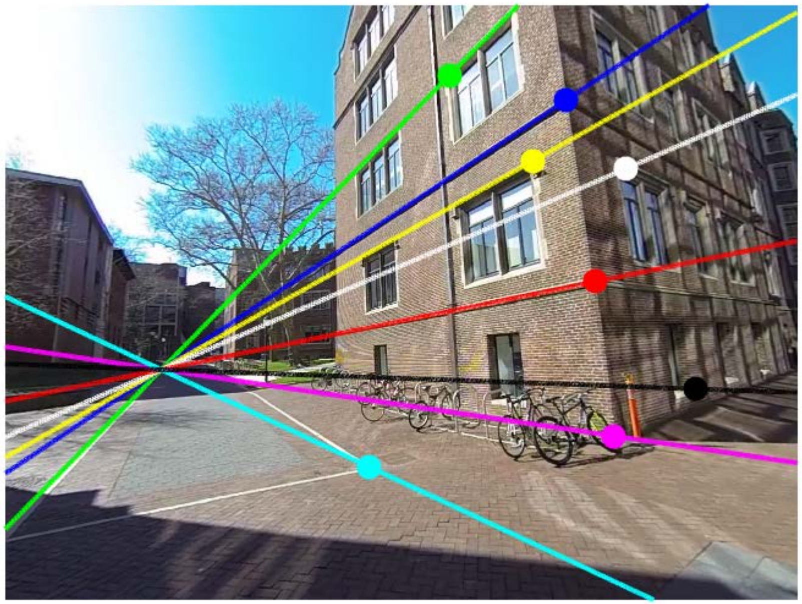
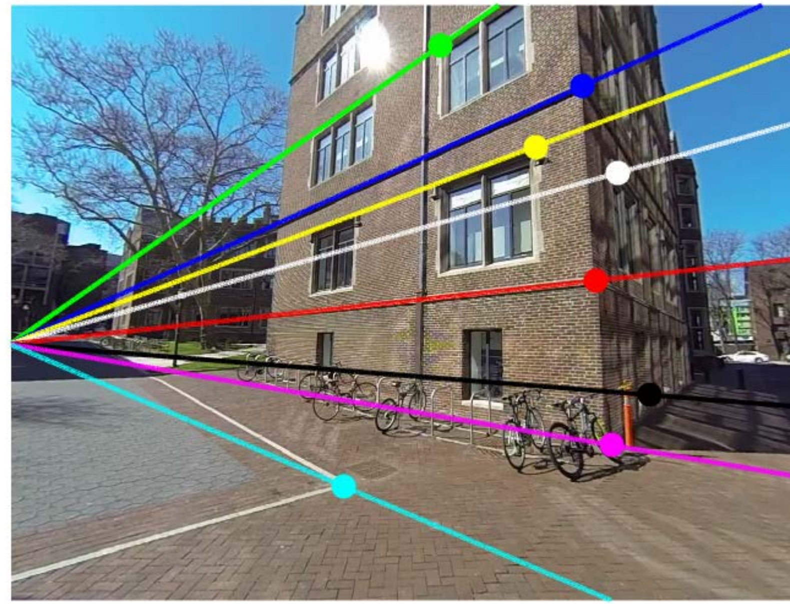

$X_2^T E X_1 = 0$
Camera coordinate system
$P_1 = K[I_{3\times3} \quad 0]$
$P_2 = K[R \quad t]$


$X_2^{\top} E X_1 = 0$
Camera coordinate system
$x_1 = K X_1, \quad x_2 = K X_2$
Transformation from camera to image
$P_1 = K[I_{3\times3} \quad 0]$
$P_2 = K[R \quad t]$

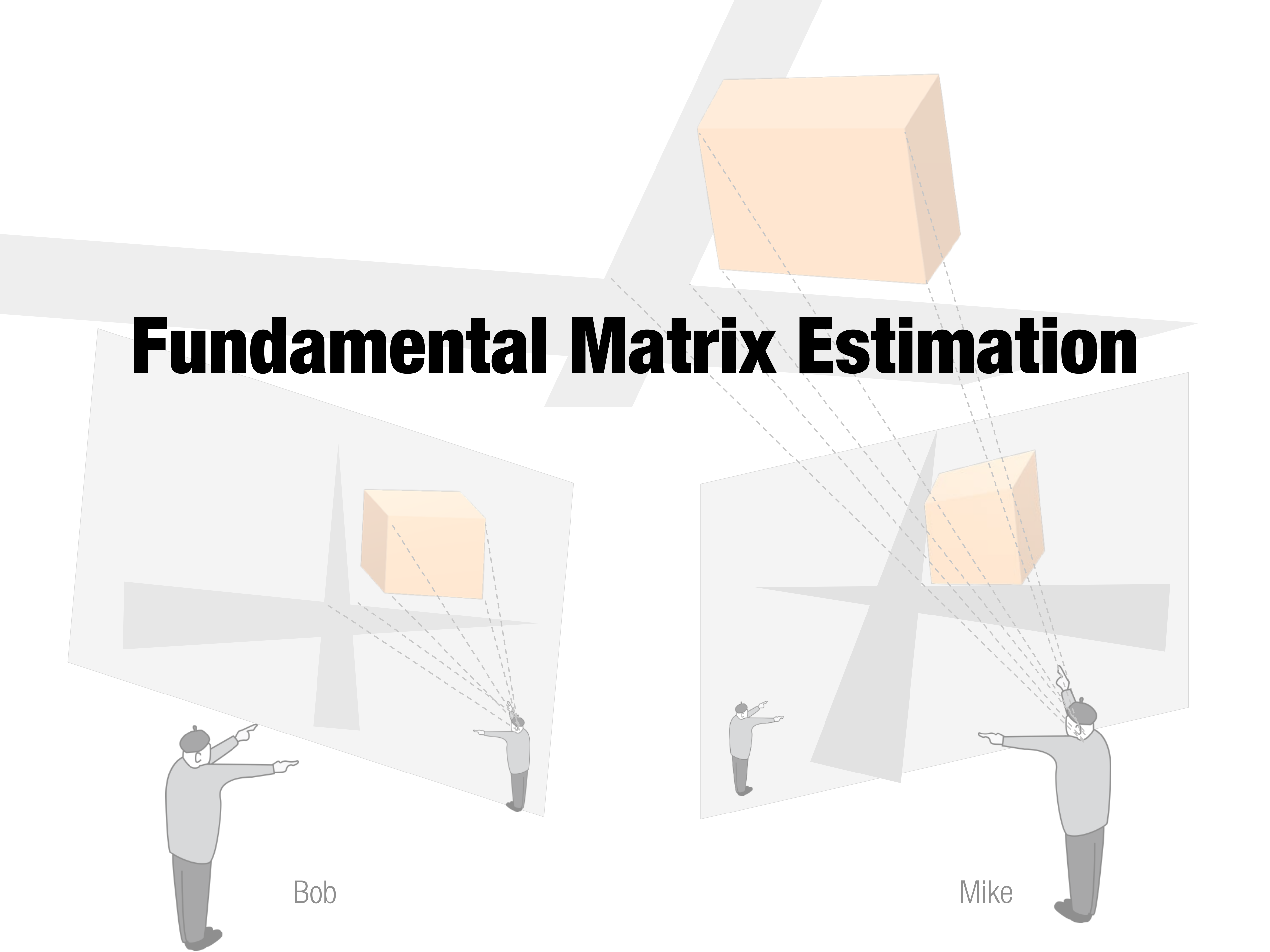


$x_2^T K^{-T} E K^{-1} x_1 = 0$
Image coordinate system
$x_1 = KX_1, \quad x_2 = KX_2$
Transformation from camera to image
$P_1 = K[I_{3\times3} \quad 0]$
$P_2 = K[R \quad t]$


$x_2^T K^{-T} E K^{-1} x_1 = 0$, or
$x_2^T F x_1 = 0$ : Fundamental matrix
where $F = K^{-T} E K^{-1}$
$P_1 = K[I_{3\times3} \quad 0]$
$P_2 = K[R \quad t]$

# Fundamental Matrix Estimation



# Fundamental Matrix

$$\mathbf{x}_2^\top \mathbf{F} \mathbf{x}_1 = 0$$

$$\mathbf{F} \in \mathbb{R}^{3\times 3}$$

$$\text{rank}(\mathbf{F}) = 2$$

Degree of freedoms: 3x3-1 = 8

Matrix dimensions (3x3)

scale factor (-1)

\# of unknowns: 8

\# of required equations: 8

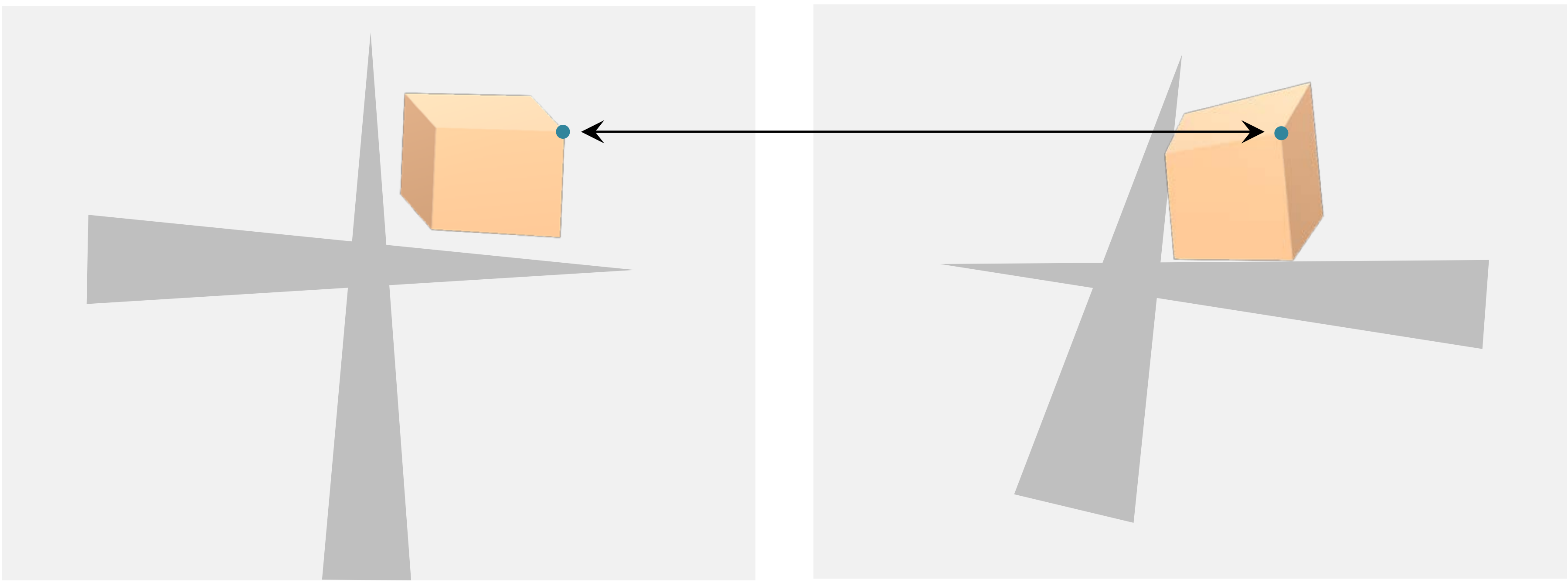
$$\mathbf{x}_{2,1}^\top \mathbf{F} \mathbf{x}_{1,1} = 0$$

$$\vdots$$

$$\mathbf{x}_{2,8}^\top \mathbf{F} \mathbf{x}_{1,8} = 0$$

8 correspondences

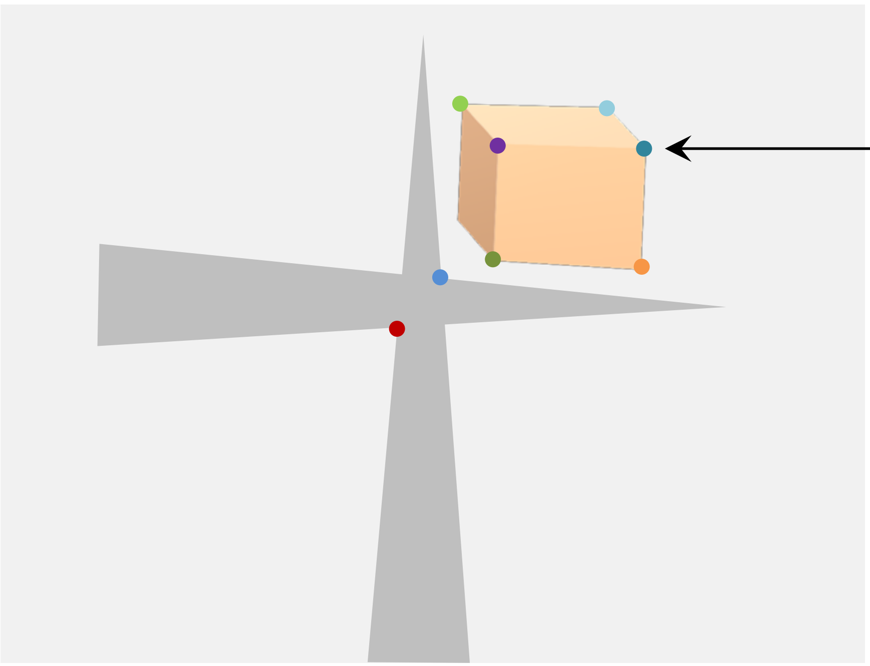
# Point correspondence



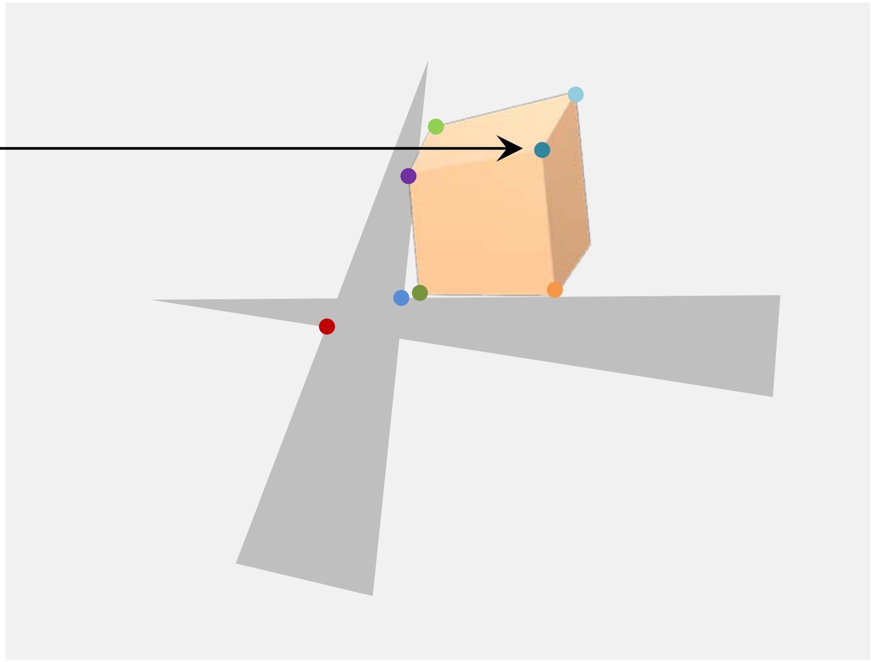
Bob's view

Mike's view

# Point correspondence



Bob's view



Mike's view

8 correspondences

$$\mathbf{x}_2^\top \mathbf{F} \mathbf{x}_1 = 0$$

$$\begin{bmatrix} u_i^2 & v_i^2 & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u_i^1 \\ v_i^1 \\ 1 \end{bmatrix} = 0$$

Linear equation in F:

$$u_i^1 u_i^2 f_{11} + u_i^1 v_i^2 f_{21} + u_i^1 f_{31} + v_i^1 u_i^2 f_{12} + v_i^1 v_i^2 f_{22} + v_i^1 f_{32} + u_i^2 f_{13} + v_i^2 f_{23} + f_{33} = 0$$

$$\mathbf{x}_2^{\top}\mathbf{F}\mathbf{x}_1 = 0$$

$$\begin{bmatrix} u_i^2 & v_i^2 & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u_i^1 \\ v_i^1 \\ 1 \end{bmatrix} = 0$$

Linear equation in F:

$$\begin{bmatrix} u_1^1 u_1^2 & u_1^1 v_1^2 & u_1^1 & v_1^1 u_1^2 & v_1^1 v_1^2 & v_1^1 & u_1^2 & v_1^2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_8^1 u_8^2 & u_8^1 v_8^2 & u_8^1 & v_8^1 u_8^2 & v_8^1 v_8^2 & v_8^1 & u_8^2 & v_8^2 & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0$$

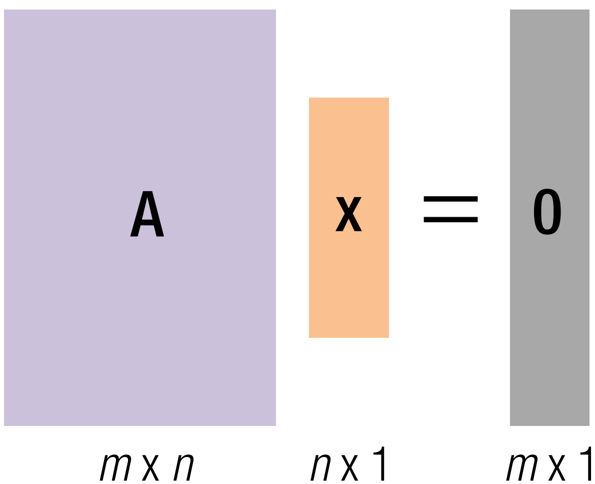
$$\mathbf{x}_2^\top \mathbf{F} \mathbf{x}_1 = 0$$

$$\begin{bmatrix} u_i^2 & v_i^2 & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u_i^1 \\ v_i^1 \\ 1 \end{bmatrix} = 0$$

Linear equation in F:

$$\underbrace{\begin{bmatrix} u_1^1 u_1^2 & u_1^1 v_1^2 & u_1^1 & v_1^1 u_1^2 & v_1^1 v_1^2 & v_1^1 & u_1^2 & v_1^2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_8^1 u_8^2 & u_8^1 v_8^2 & u_8^1 & v_8^1 u_8^2 & v_8^1 v_8^2 & v_8^1 & u_8^2 & v_8^2 & 1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ \text{[illegible]} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix}}_{\mathbf{X}} = \mathbf{0}$$

# Linear Homogeneous Equations




Linear least square solve produces a trivial solution:

$$\mathbf{x} = \left(\mathbf{A}^{T}\mathbf{A}\right)^{-1}\mathbf{A}^{T}\mathbf{b} \rightarrow \mathbf{x} = \mathbf{0}$$

An additional constraint on $\mathbf{X}$ to avoid the trivial solution: $\|\mathbf{x}\| = 1$

1) $\mathbf{rank(A)} = r < n-1$ : infinite number of solutions

$$\mathbf{x} = \lambda_{r+1}\mathbf{V}_{r+1} + \cdots + \lambda_n \mathbf{V}_n \quad \text{where} \quad \sum_{i=r+1}^{n} \lambda_i^2 = 1$$

2) $\mathbf{rank(A)} = n-1$ : one exact solution

$$\mathbf{x} = \mathbf{V}_n$$

3) $n < m$ : no exact solution in general (needs least squares)

$$\min_{\mathbf{x}} \|\mathbf{Ax}\|^2 \text{ subject to } \|\mathbf{x}\| = 1 \rightarrow \mathbf{x} = \mathbf{V}_n$$

# 8 Point Algorithm

- Construct 8x9 matrix **A**.

$$\begin{bmatrix} u_1^1 u_1^2 & u_1^1 v_1^2 & u_1^1 & v_1^1 u_1^2 & v_1^1 v_1^2 & v_1^1 & u_1^2 & v_1^2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_8^1 u_8^2 & u_8^1 v_8^2 & u_8^1 & v_8^1 u_8^2 & v_8^1 v_8^2 & v_8^1 & u_8^2 & v_8^2 & 1 \end{bmatrix}$$

**A**

# 8 Point Algorithm

- Construct 8x9 matrix **A**.
- Solving linear homogeneous equations via SVD:

  $\mathbf{x} = \mathbf{V}_{:,9}$ where $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\top}$

  $\mathbf{F} = \mathrm{reshape}(\mathbf{x}, 3, 3)$: constructing matrix from vector.

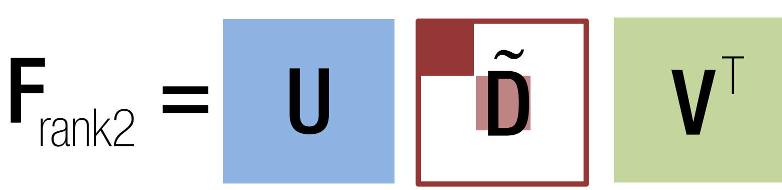
# 8 Point Algorithm

- Construct 8x9 matrix **A**.
- Solving linear homogeneous equations via SVD:

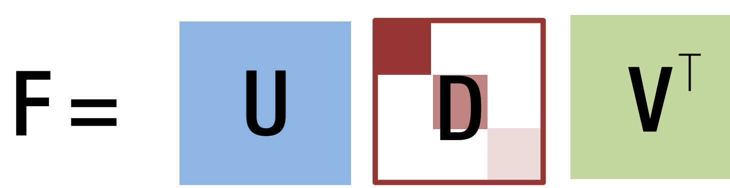
  $\mathbf{x} = \mathbf{V}_{:,8}$ where $\mathbf{A} = \mathbf{UDV}^{\top}$

  $\mathbf{F} = \text{reshape}(\mathbf{x}, 3, 3)$: constructing matrix from vector.
- Applying rank constraint, i.e., $\text{rank}(\mathbf{F}) = 2$ .

  $\mathbf{F}_{\text{rank2}} = \mathbf{U}\tilde{\mathbf{D}}\mathbf{V}^{\top}$ where $\tilde{\mathbf{D}}$ : $\mathbf{D}$ with the last element zero.



SVD cleanup



## 1.2 Match Outlier Rejection via RANSAC

**Goal** Given $N$ correspondences between two images ($N \geq 8$), $\mathbf{x}_1 \leftrightarrow \mathbf{x}_2$, implement the following function that estimates inlier correspondences using fundamental matrix based RANSAC:

```
[y1 y2 idx] = GetInliersRANSAC(x1, x2)
```

(INPUT) `x1` and `x2`: $N \times 2$ matrices whose row represents a correspondence.

(OUTPUT) `y1` and `y2`: $N_i \times 2$ matrices whose row represents an inlier correspondence where $N_i$ is the number of inliers.

(OUTPUT) `idx`: $N \times 1$ vector that indicates ID of inlier `y1`.

A pseudo code the RANSAC is shown in Algorithm 2.

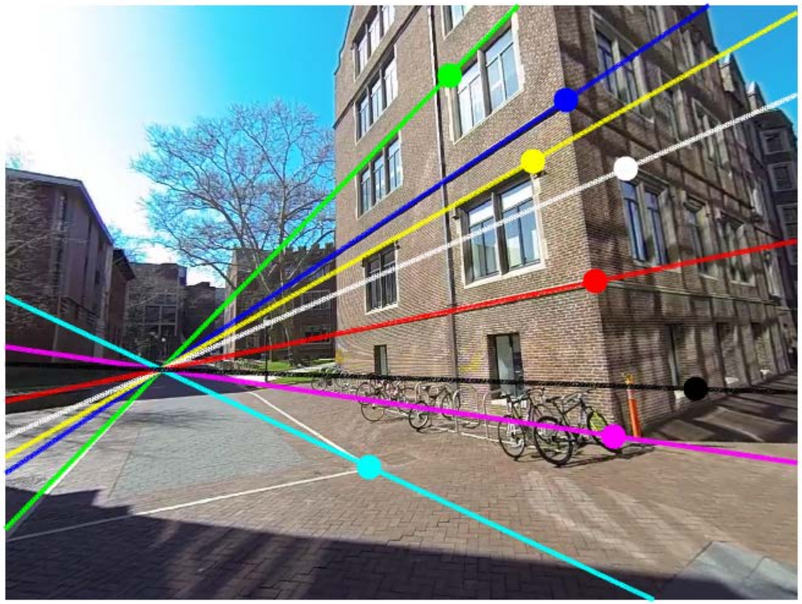
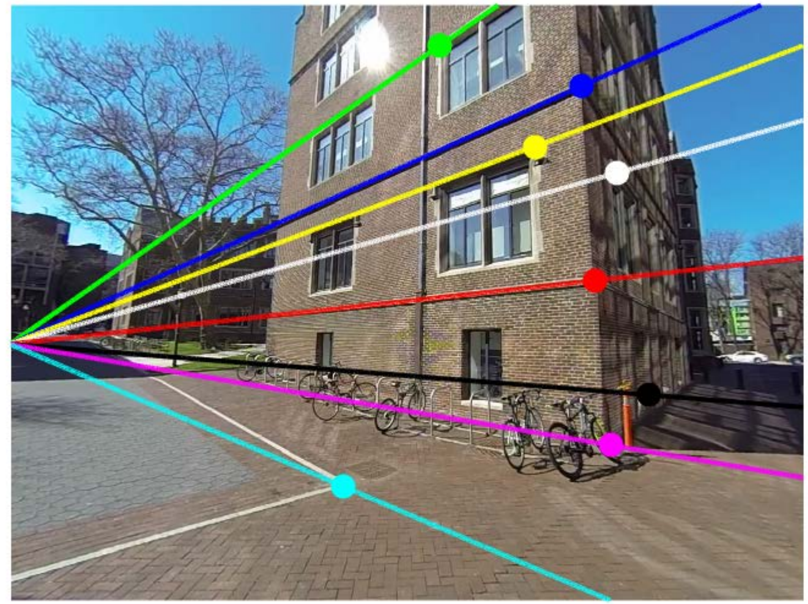
---

**Algorithm 2** GetInliersRANSAC

---

$n \leftarrow 0$
**for** $i = 1 : M$ **do**
    Choose 8 correspondences, $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$, randomly
    `F = EstimateFundamentalMatrix(`$\hat{\mathbf{x}}_1$`, `$\hat{\mathbf{x}}_2$`)`
    $\mathcal{S} \leftarrow \emptyset$
    **for** $j = 1 : N$ **do**
        **if** $|\mathbf{x}_{2j}^{\mathsf{T}} \mathbf{F} \mathbf{x}_{1j}| < \epsilon$ **then**
            $\mathcal{S} \leftarrow \mathcal{S} \cup \{j\}$
        **end if**
    **end for**
    **if** $n < |\mathcal{S}|$ **then**
        $n \leftarrow |\mathcal{S}|$
        $\mathcal{S}_{in} \leftarrow \mathcal{S}$
    **end if**
**end for**

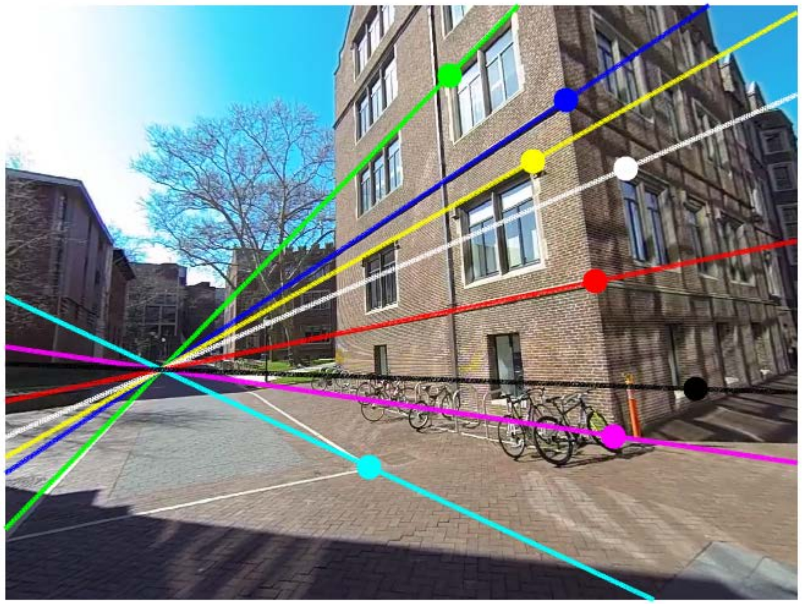
---

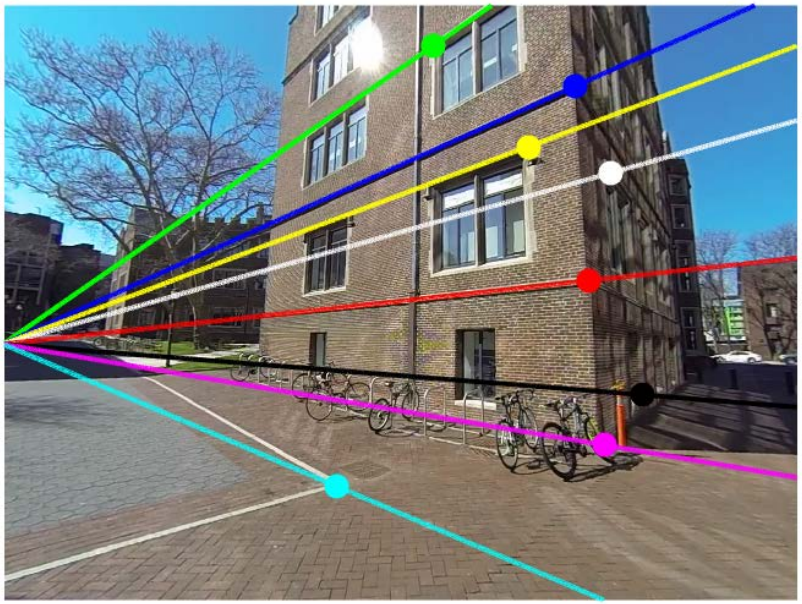
 

F =

1.0e+003 *

| | | |
|---|---|---|
| 0.0000 | 0.0001 | -0.0463 |
| -0.0001 | 0.0000 | 0.0181 |
| 0.0519 | -0.0043 | -9.9997 |





```
>> rank(F)
ans =
     3
>> [u,d,v] = svd(F);
>> d(3,3) = 0;
```

```
>> F = u * d * v'          : SVD cleanup
F =
  1.0e+003 *
    0.0000    0.0001   -0.0463
   -0.0001    0.0000    0.0181
    0.0519   -0.0043   -9.9997
>> rank(F)
ans =
     2
```


x1 =

950   450

L2 =

-0.1024   -0.9947   547.0942

$$\mathbf{L}_2 = \mathbf{F}\mathbf{x}_1$$

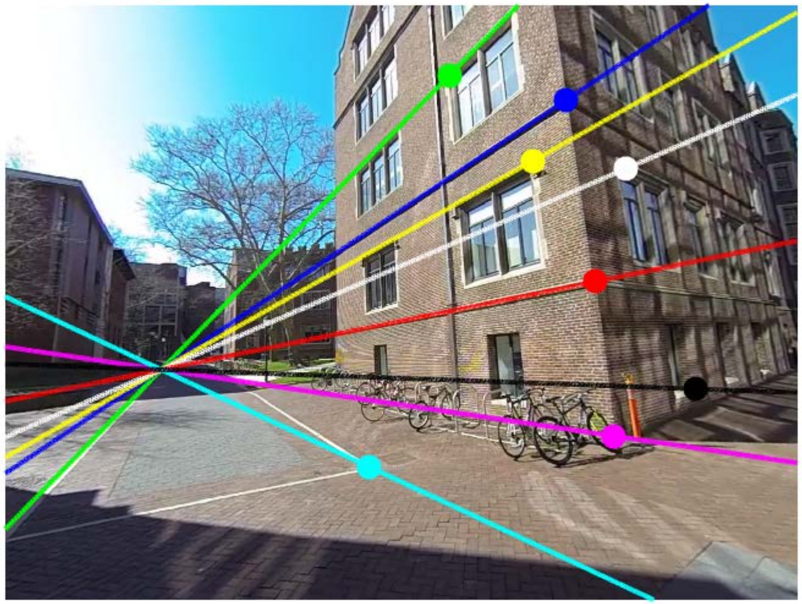

L1 =

0.5489 0.8359 -627.0515

$$\mathbf{L}_1 = \mathbf{F}^{\top}\mathbf{x}_2$$

x2 =

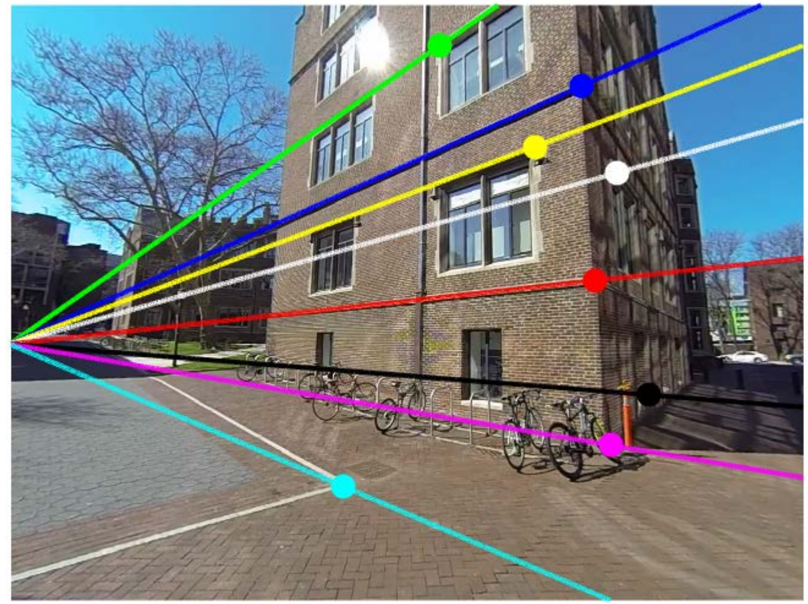
920 130



Epipole of left image: null space of F

```
[u,d] = eigs(F'*F);
u =
   -0.0052    0.9258   -0.3780
    0.0004   -0.3780   -0.9258
    1.0000    0.0050   -0.0016
d =
   1.0000          0                 0
        0    6.4719e-10          0
        0          0    -7.6511e-22
```

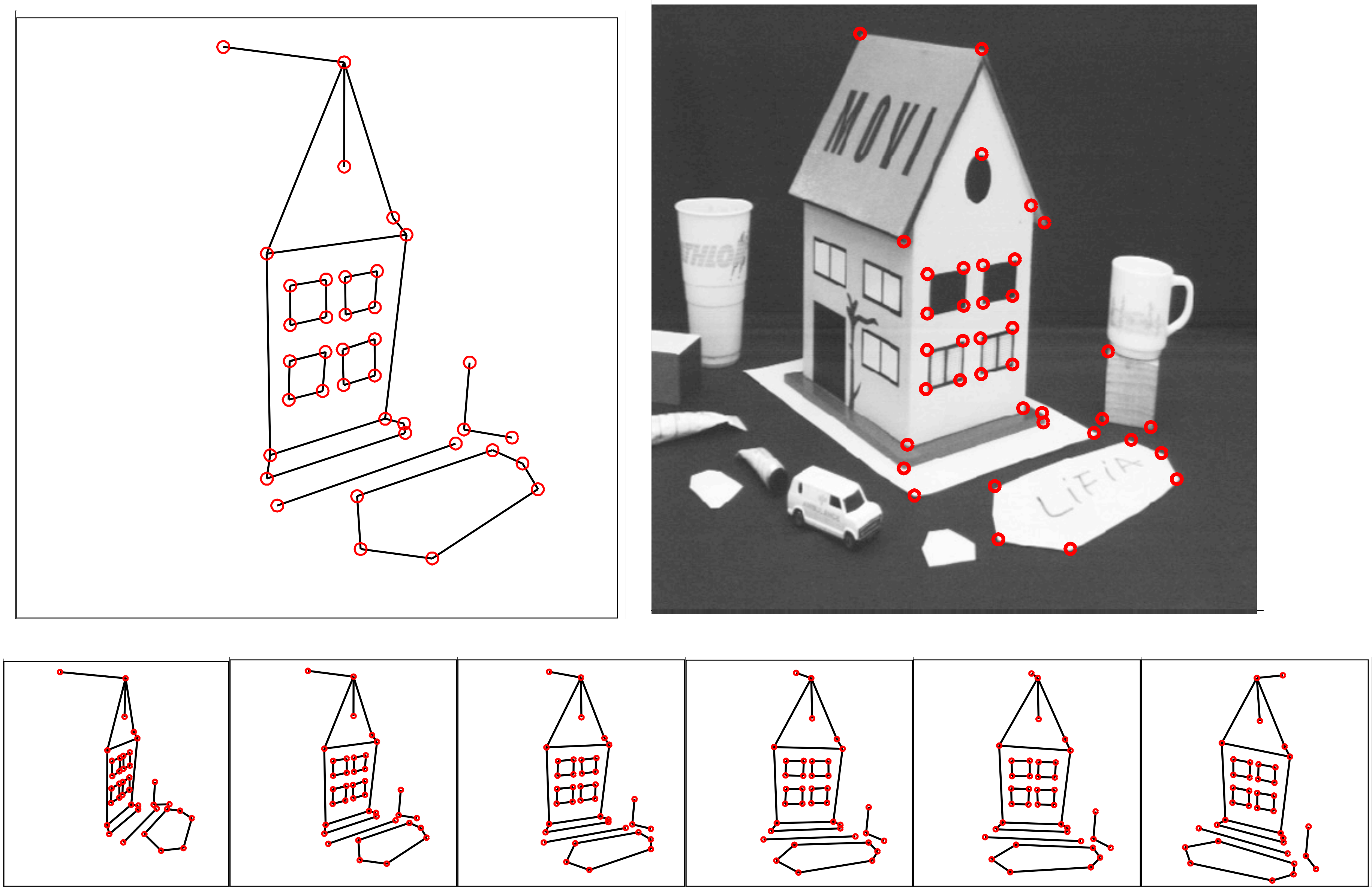
uu = u(:, 3) = [-0.3780, -0.9258, -0.0016]



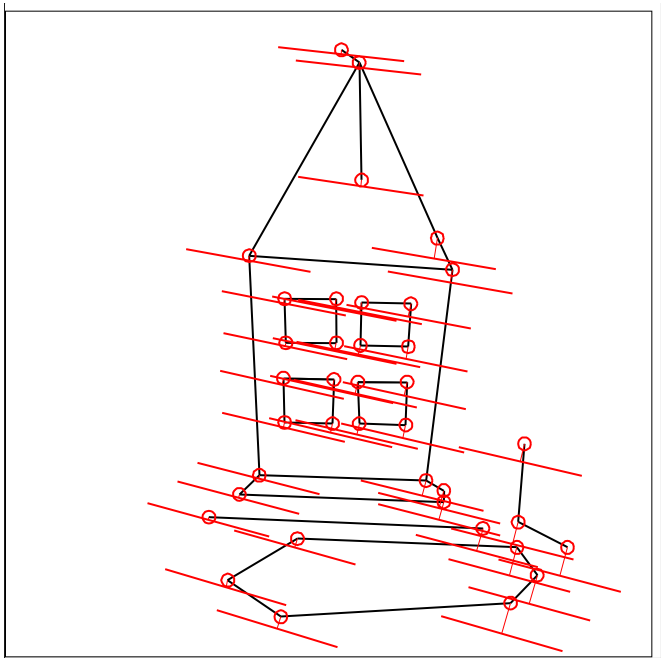
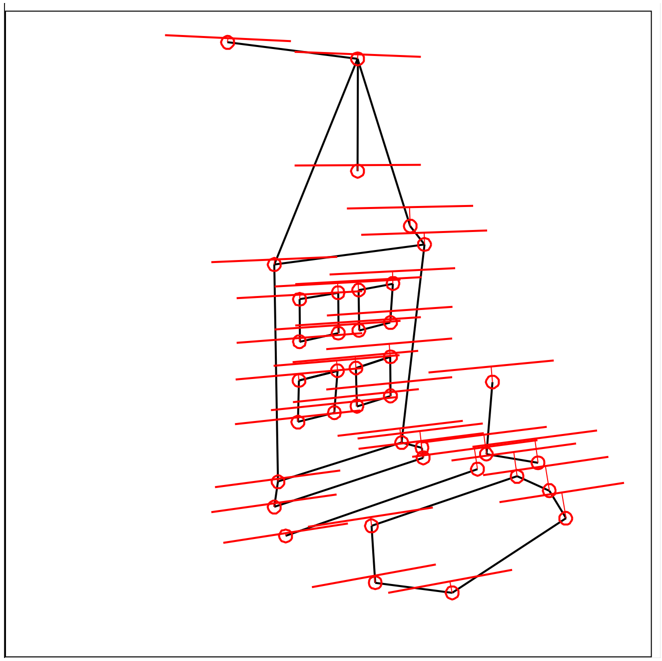
```
[u,d] = eigs(F*F');
u =
    0.0046    1.0000    0.0029
   -0.0018    0.0029   -1.0000
    1.0000   -0.0046   -0.0018
d =
   1.0000         0              0
        0    6.4719e-10          0
        0         0    -5.6583e-21
```

Epipole of right image: Null sapce of F transposed

uu = u(:, 3) = [0.0029, -1.0000, -0.0018]

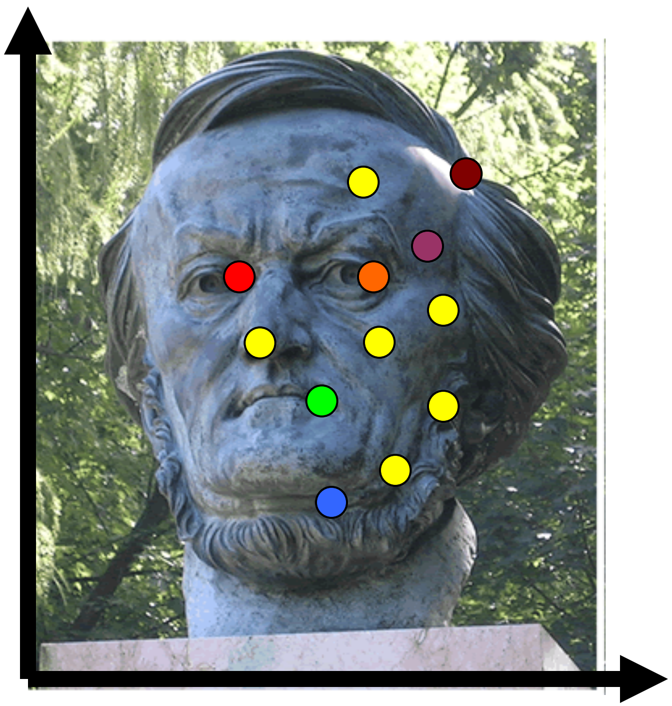



Data courtesy of R. Mohr and B. Boufama.

Mean errors:
10.0pixel
9.1pixel

# Problems with the 8-Point Algorithm



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

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

# Problems with the 8-Point Algorithm

$$\mathbf{Wf} = 0$$

$$\begin{pmatrix} u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 & 1 \\ u_2u'_2 & u_2v'_2 & u_2 & v_2u'_2 & v_2v'_2 & v_2 & u'_2 & v'_2 & 1 \\ u_3u'_3 & u_3v'_3 & u_3 & v_3u'_3 & v_3v'_3 & v_3 & u'_3 & v'_3 & 1 \\ u_4u'_4 & u_4v'_4 & u_4 & v_4u'_4 & v_4v'_4 & v_4 & u'_4 & v'_4 & 1 \\ u_5u'_5 & u_5v'_5 & u_5 & v_5u'_5 & v_5v'_5 & v_5 & u'_5 & v'_5 & 1 \\ u_6u'_6 & u_6v'_6 & u_6 & v_6u'_6 & v_6v'_6 & v_6 & u'_6 & v'_6 & 1 \\ u_7u'_7 & u_7v'_7 & u_7 & v_7u'_7 & v_7v'_7 & v_7 & u'_7 & v'_7 & 1 \\ u_8u'_8 & u_8v'_8 & u_8 & v_8u'_8 & v_8v'_8 & v_8 & u'_8 & v'_8 & 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

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

HZ pag 108

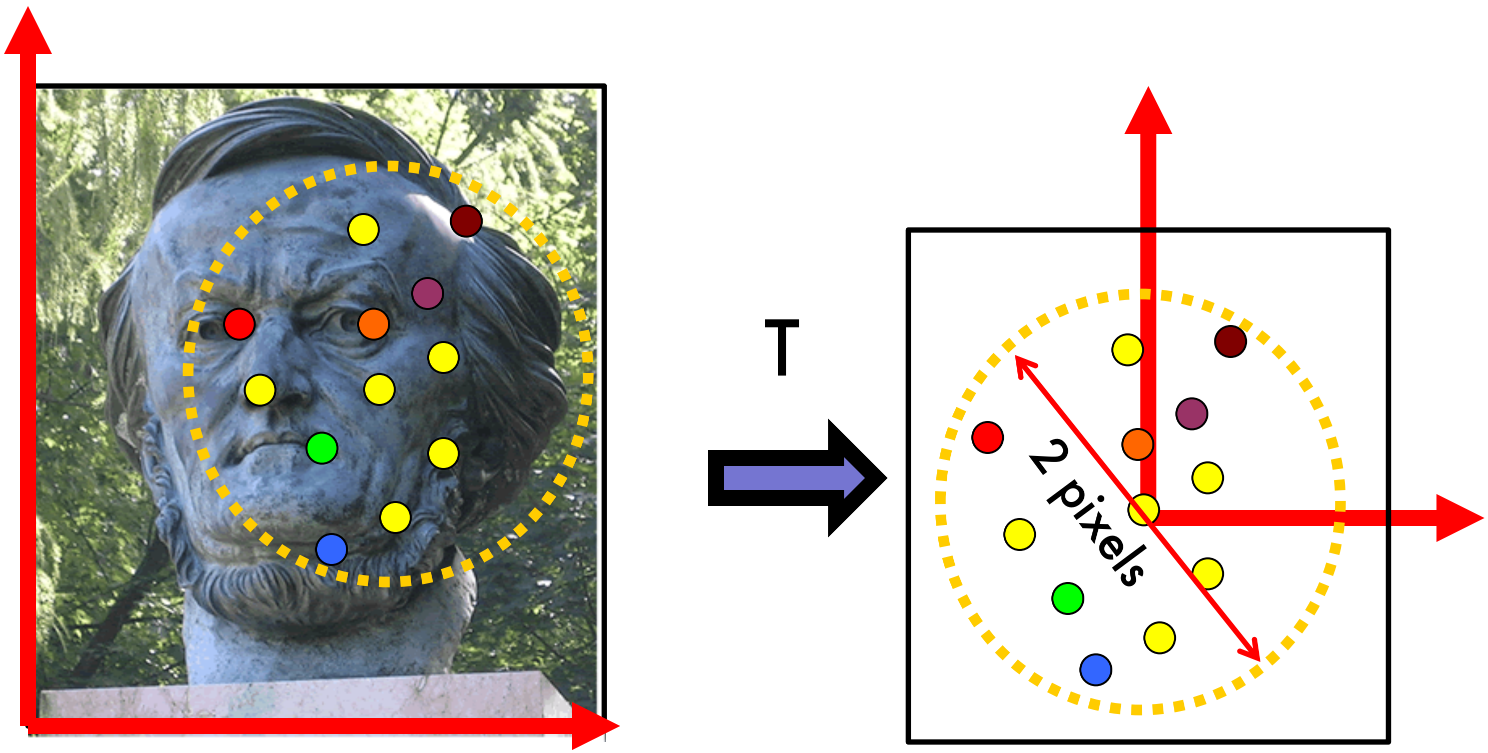
# Normalization

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

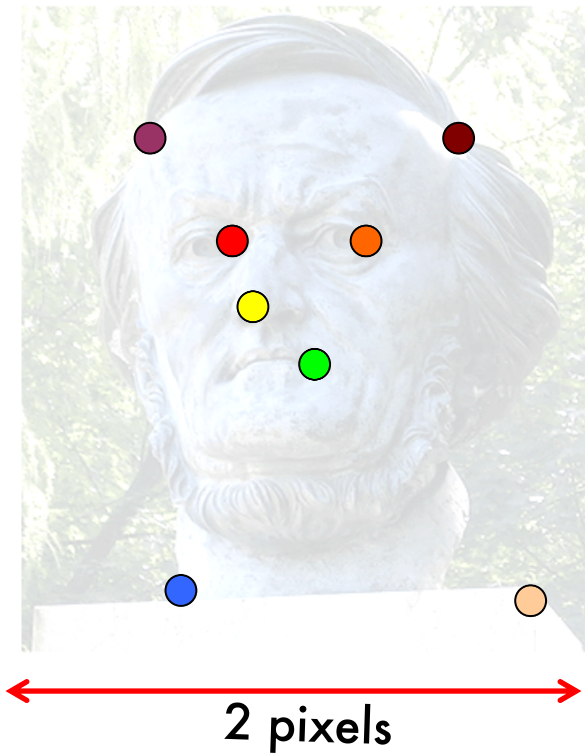


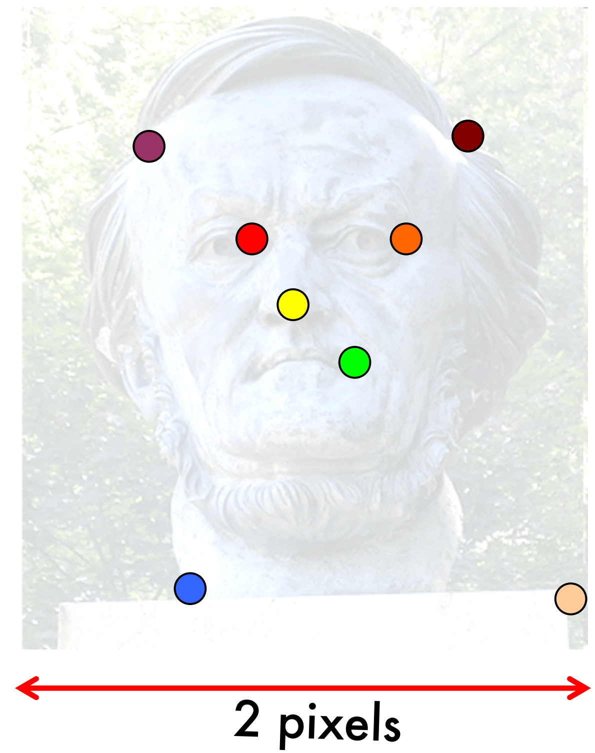

Coordinate system of the image before applying T

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

0. Compute T and T′ for image 1 and 2, respectively

1. Normalize coordinates in images 1 and 2:

$$q_i = T\ p_i \qquad q_i' = T'\ p_i'$$

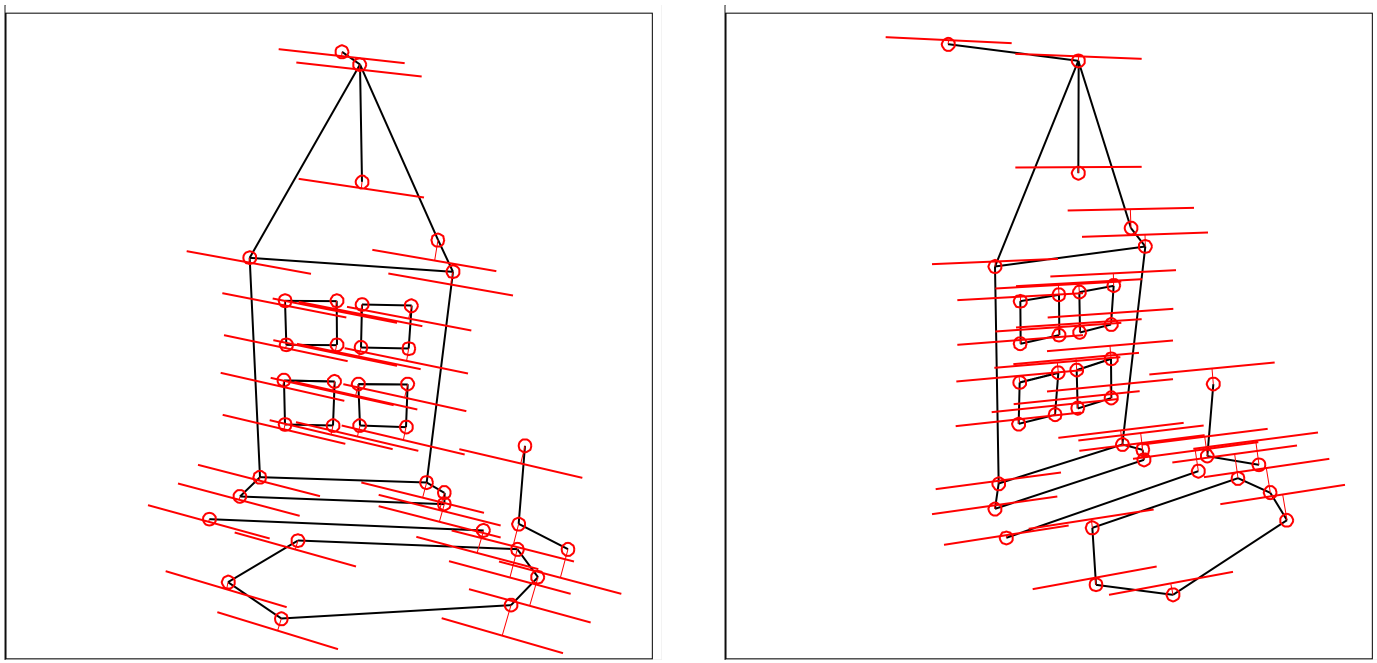
2. Use the eight-point algorithm to compute $\hat{F}_q$ from the corresponding points $q_i$ and $q_i'$ .

1. Enforce the rank-2 constraint: $\rightarrow \mathrm{F_q}$ such that:

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

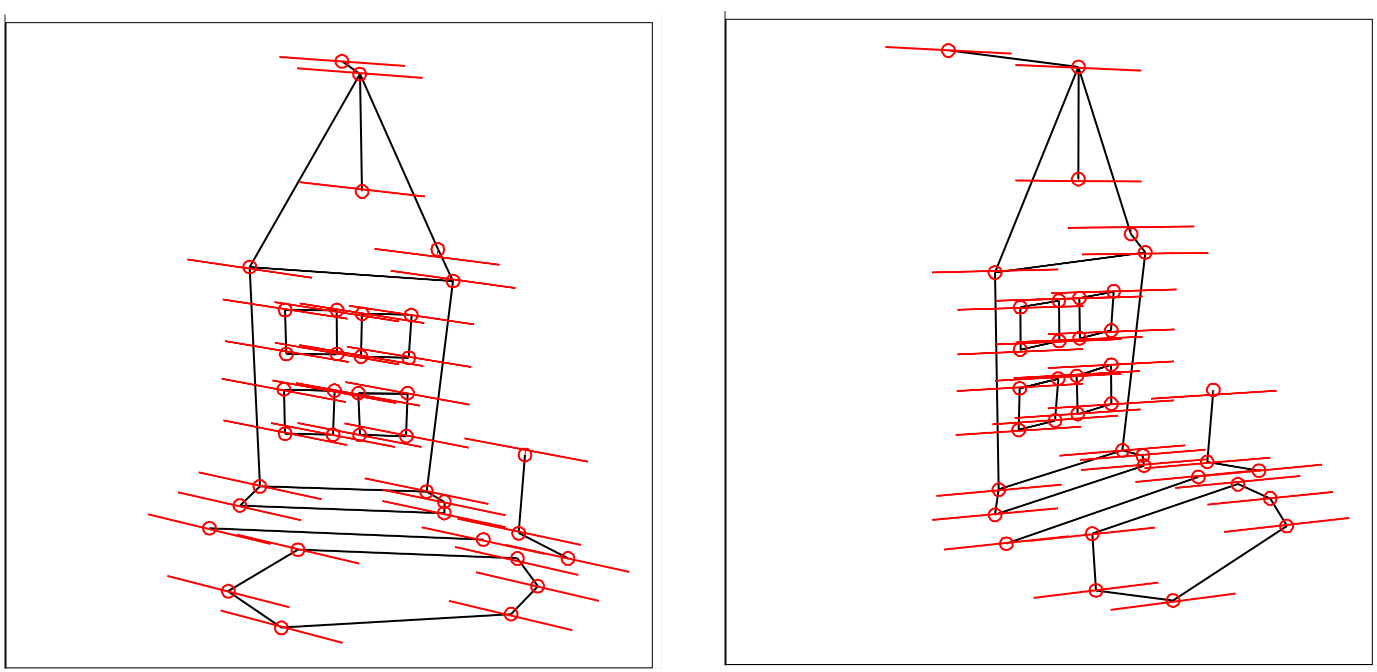
2. De-normalize $F_q$: $F = T^T F_q\ T'$

Without normalization



Mean errors:
10.0pixel
9.1pixel

With normalization



Mean errors:
1.0pixel
0.9pixel