Epipolar Geometry

CSE P576

Dr. Matthew Brown

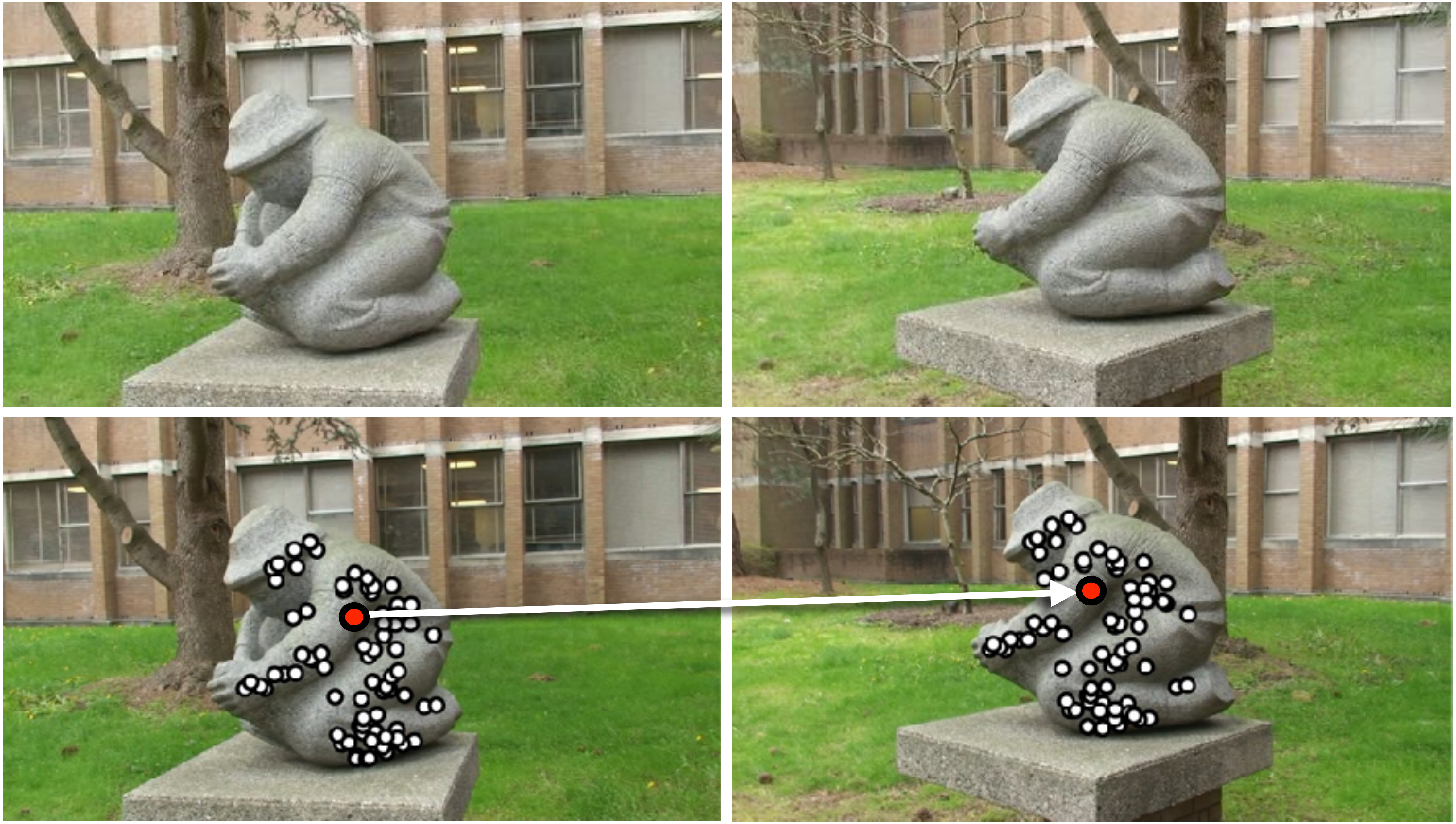
Epipolar Geometry

- Epipolar Lines, Plane Constraint
- Fundamental Matrix, Linear solution
- RANSAC for F, 2-view SFM

[Szeliski Chapters 7+11]

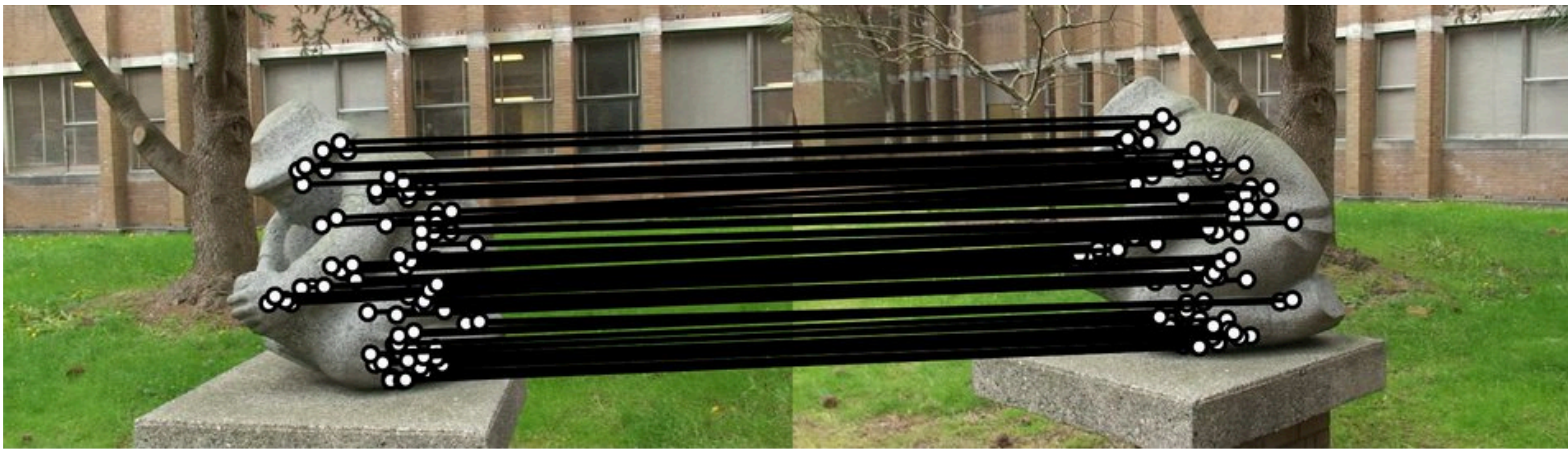
Correspondence

- Find all matches between views



Geometric Constraints

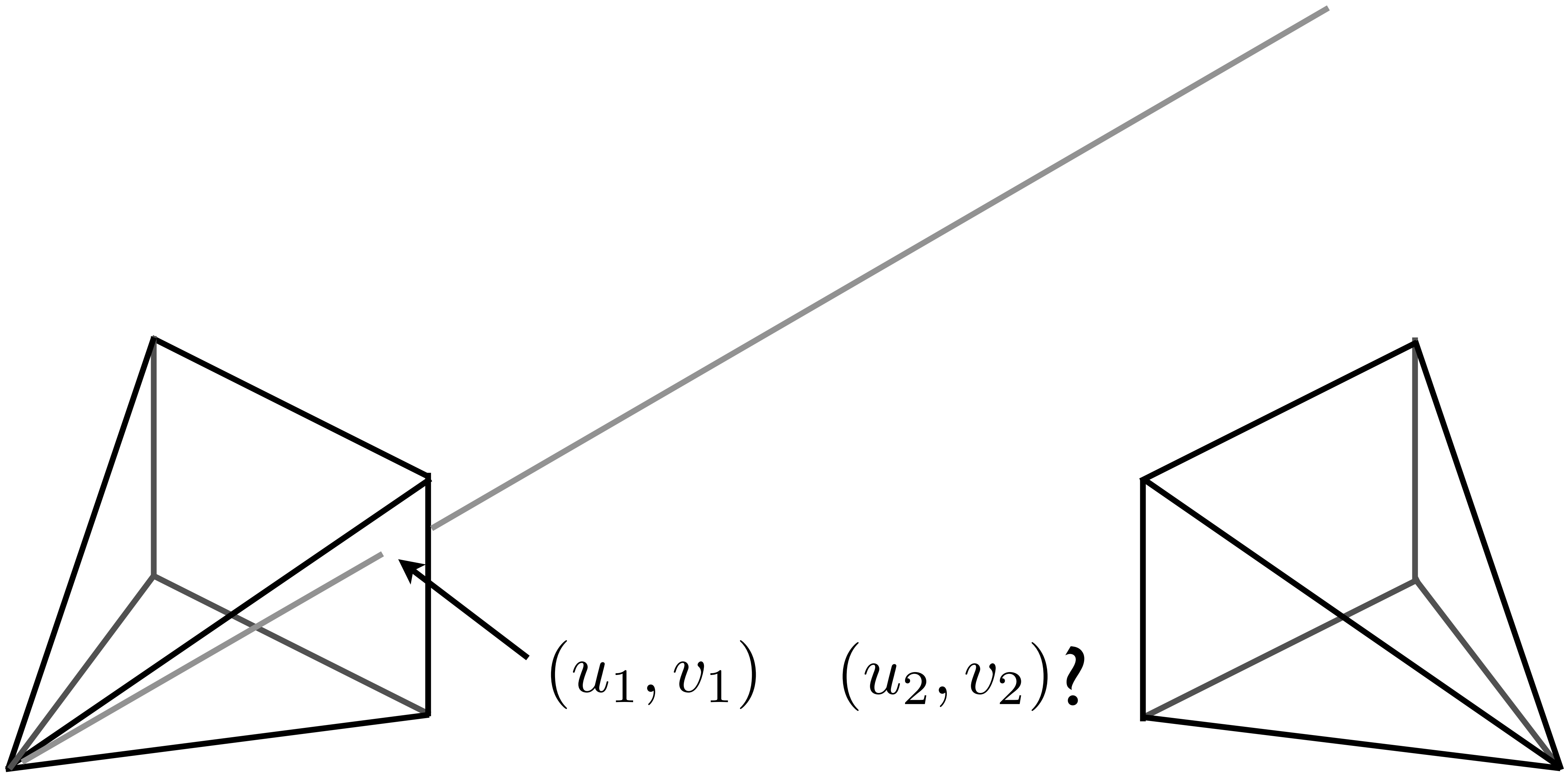
- Find subset of matches that are consistent with a geometric transformation



Consistent matches can be used for subsequent stages, e.g., 3D reconstruction, object recognition etc.

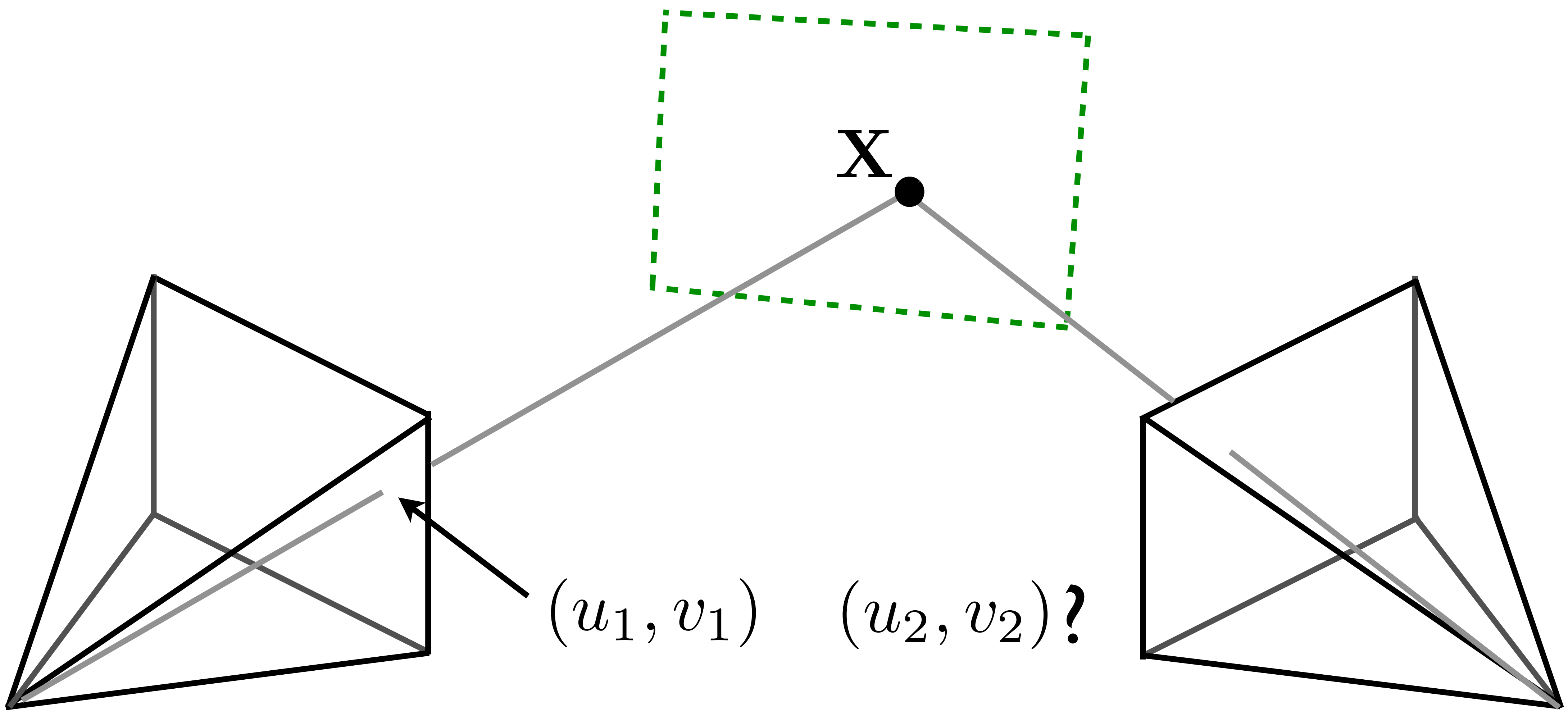
2-view Geometry

- How do we transfer points between 2 views?



2-view Geometry

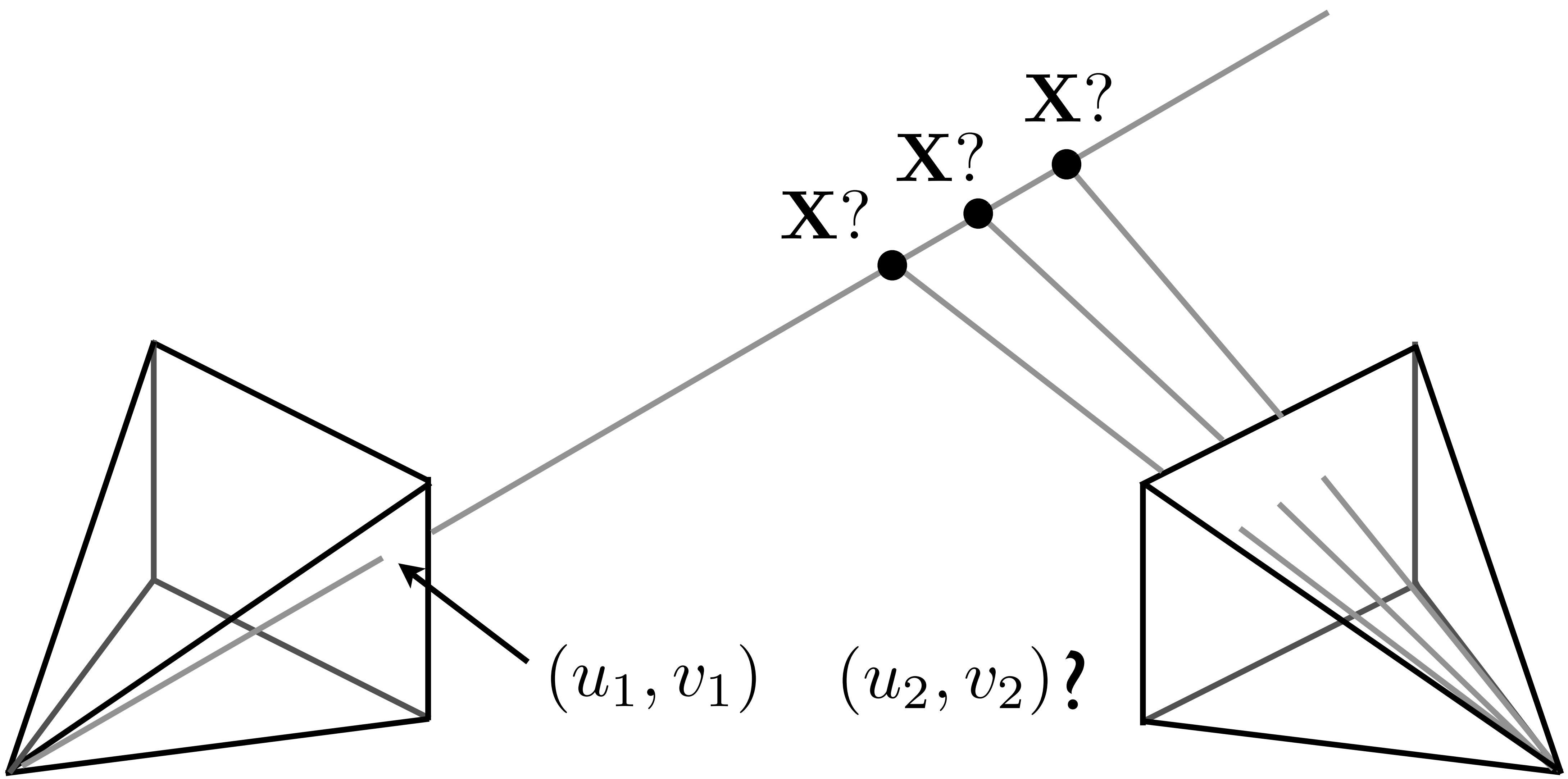
- How do we transfer points between 2 views? (planar case)



Planar case: one-to-one mapping via plane (Homography)

2-view Geometry

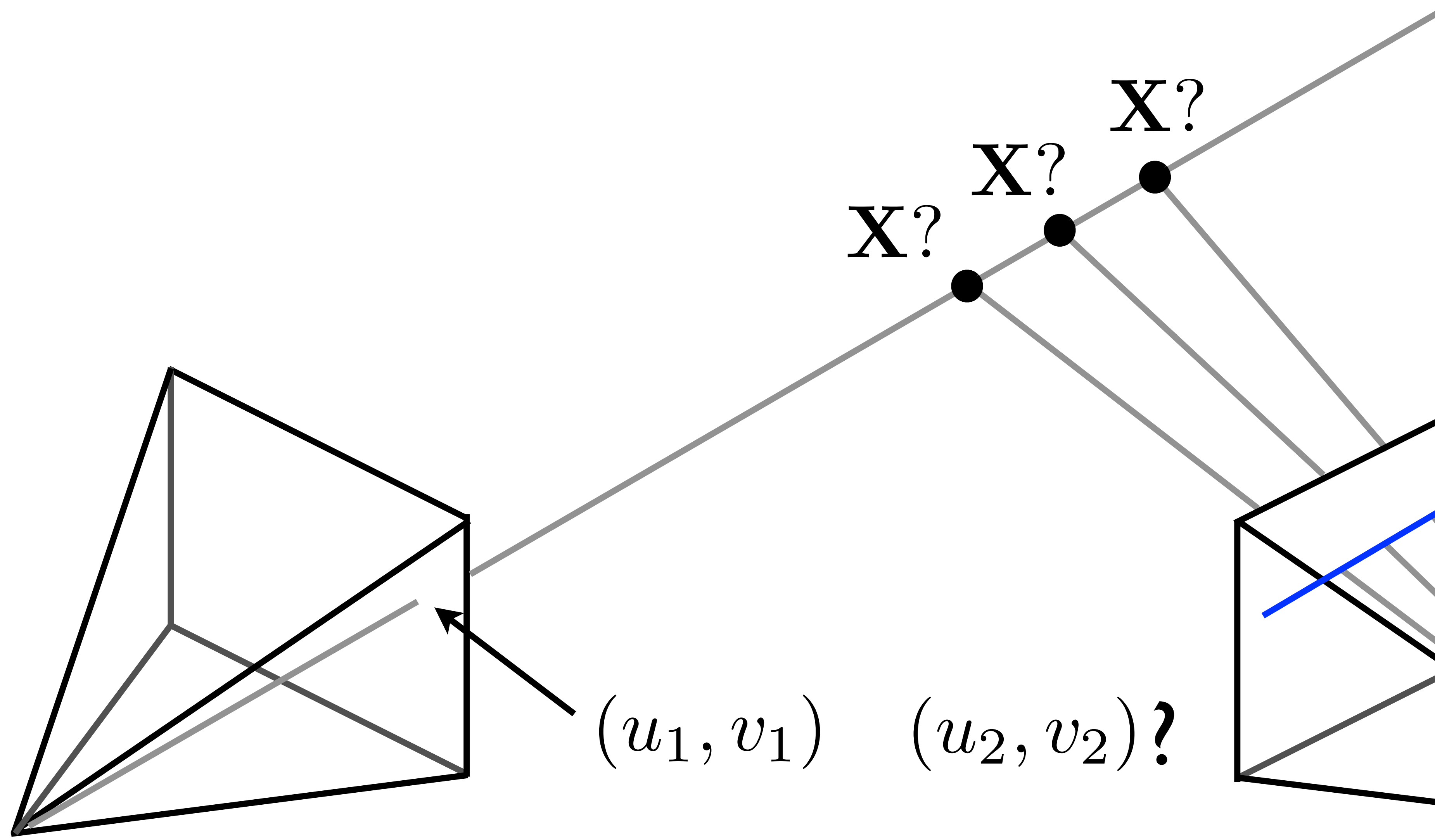
- How do we transfer points between 2 views? (non-planar)



Non-planar case: depends on the depth of the 3D point

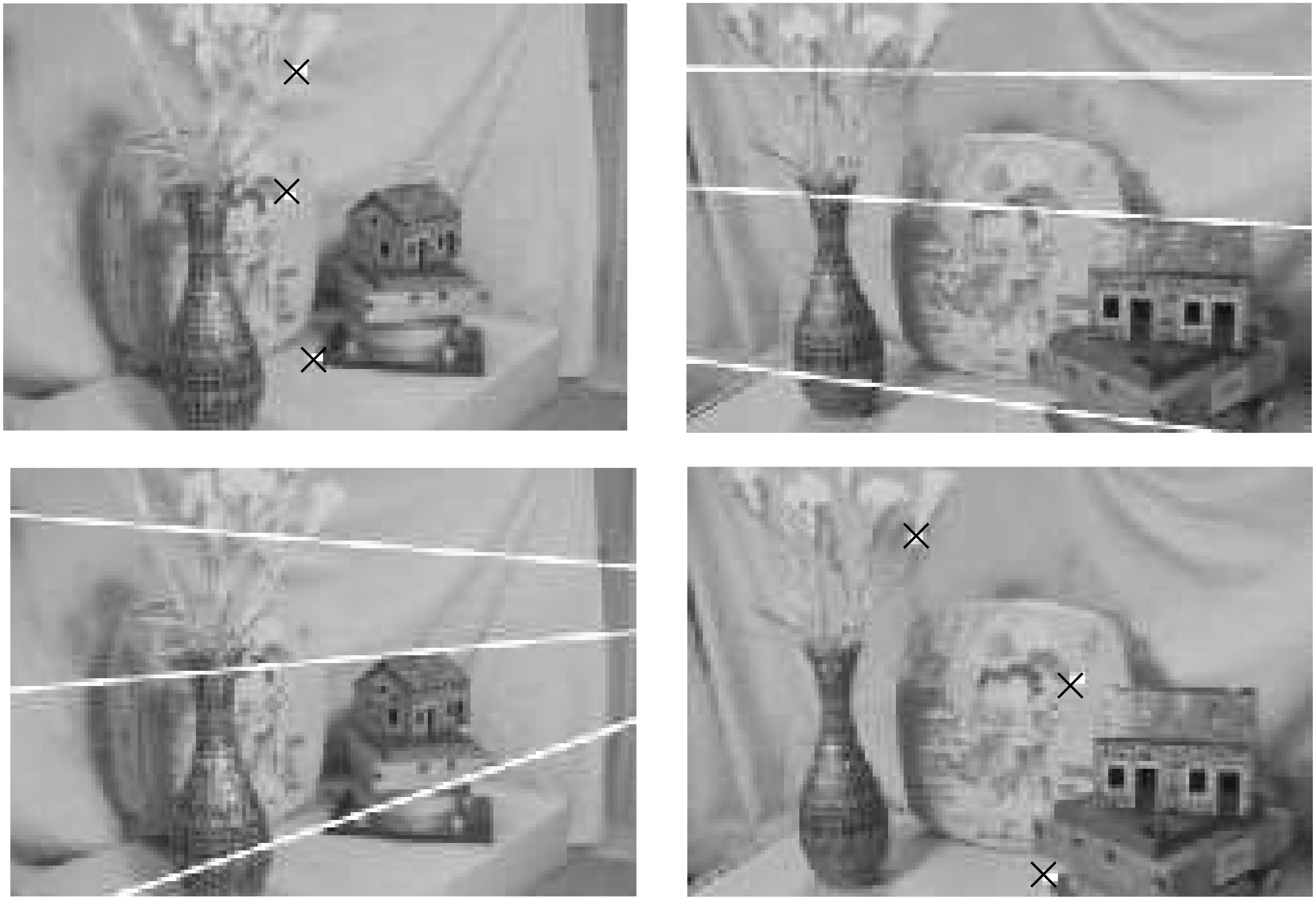
Epipolar Line

- How do we transfer points between 2 views? (non-planar)



A point in image 1 gives a **line** in image 2

Epipolar Lines



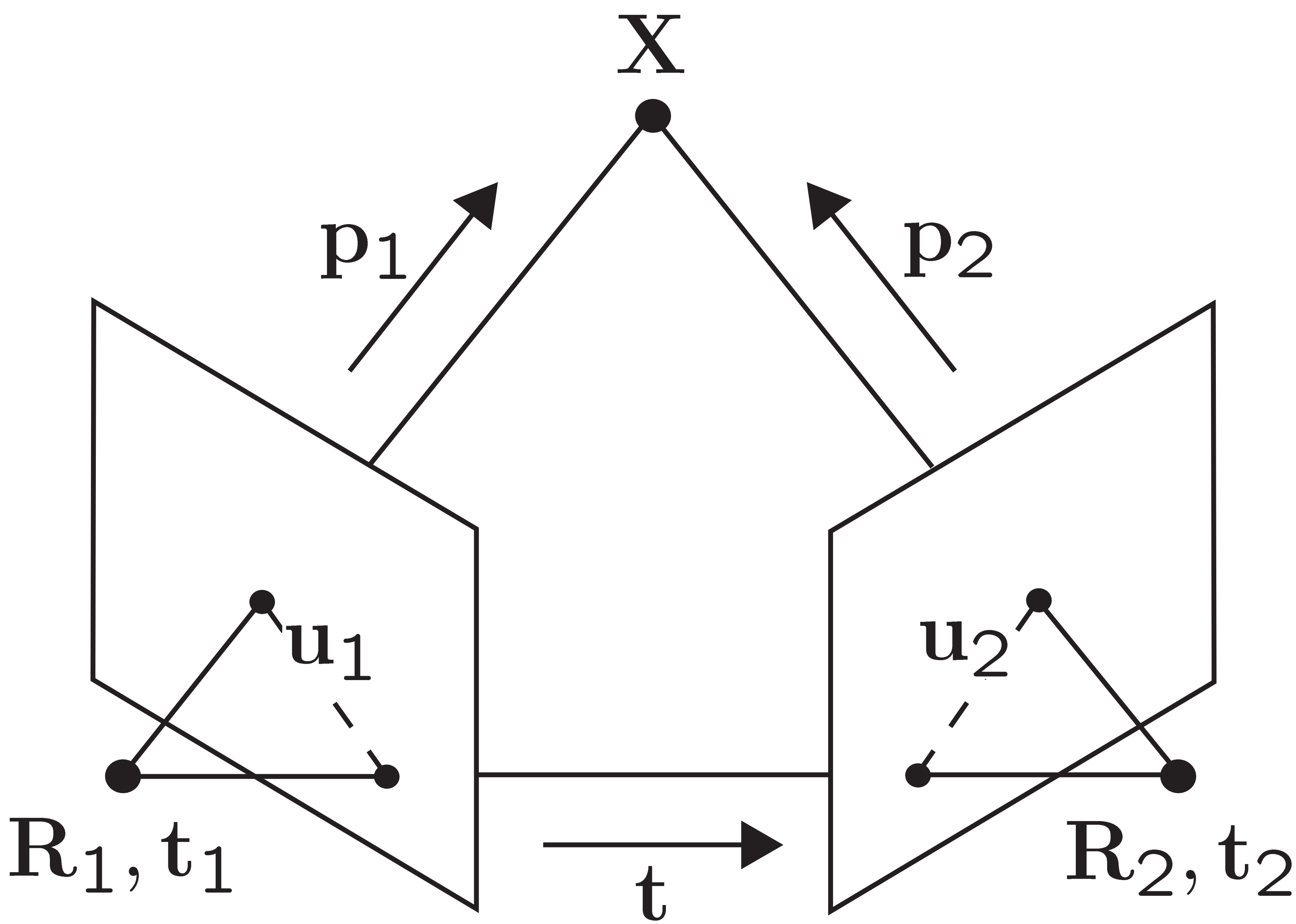
[R. Cipolla]

Epipolar Lines

[R. Cipolla]

The Epipolar Constraint

- For rays to intersect at a point (X), the two rays and the camera translation must lie in the same plane



4.1

[Szeliski 7.2, 11.1]

Computing F

- Single correspondence gives us one equation

$$\begin{bmatrix} u_1 & v_1 & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = 0$$

- Multiply out

$$\begin{aligned} u_1 x_1 f_{11} + u_1 y_1 f_{12} + u_1 f_{13} + v_1 x_1 f_{21} + v_1 y_1 f_{22} \\ + v_1 f_{23} + x_1 f_{31} + y_1 f_{32} + f_{33} = 0 \end{aligned}$$

Computing F

- Rearrange for unknowns, add points by stacking rows

$$\begin{bmatrix} u_1x_1 & u_1y_1 & u_1 & v_1x_1 & v_1y_1 & v_1 & x_1 & y_1 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_2x_2 & u_2y_2 & u_2 & v_2x_2 & v_2y_2 & v_2 & x_2 & y_2 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_3x_3 & u_3y_3 & u_3 & v_3x_3 & v_3y_3 & v_3 & x_3 & y_3 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_4x_4 & u_4y_4 & u_4 & v_4x_4 & v_4y_4 & v_4 & x_4 & y_4 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_5x_5 & u_5y_5 & u_5 & v_5x_5 & v_5y_5 & v_5 & x_5 & y_5 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_6x_6 & u_6y_6 & u_6 & v_6x_6 & v_6y_6 & v_6 & x_6 & y_6 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_7x_7 & u_7y_7 & u_7 & v_7x_7 & v_7y_7 & v_7 & x_7 & y_7 & 1 \end{bmatrix}$$
$$\begin{bmatrix} u_8x_8 & u_8y_8 & u_8 & v_8x_8 & v_8y_8 & v_8 & x_8 & y_8 & 1 \end{bmatrix}$$
$$\begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix}$$

- This is a linear system of the form $\mathbf{Af} = \mathbf{0}$

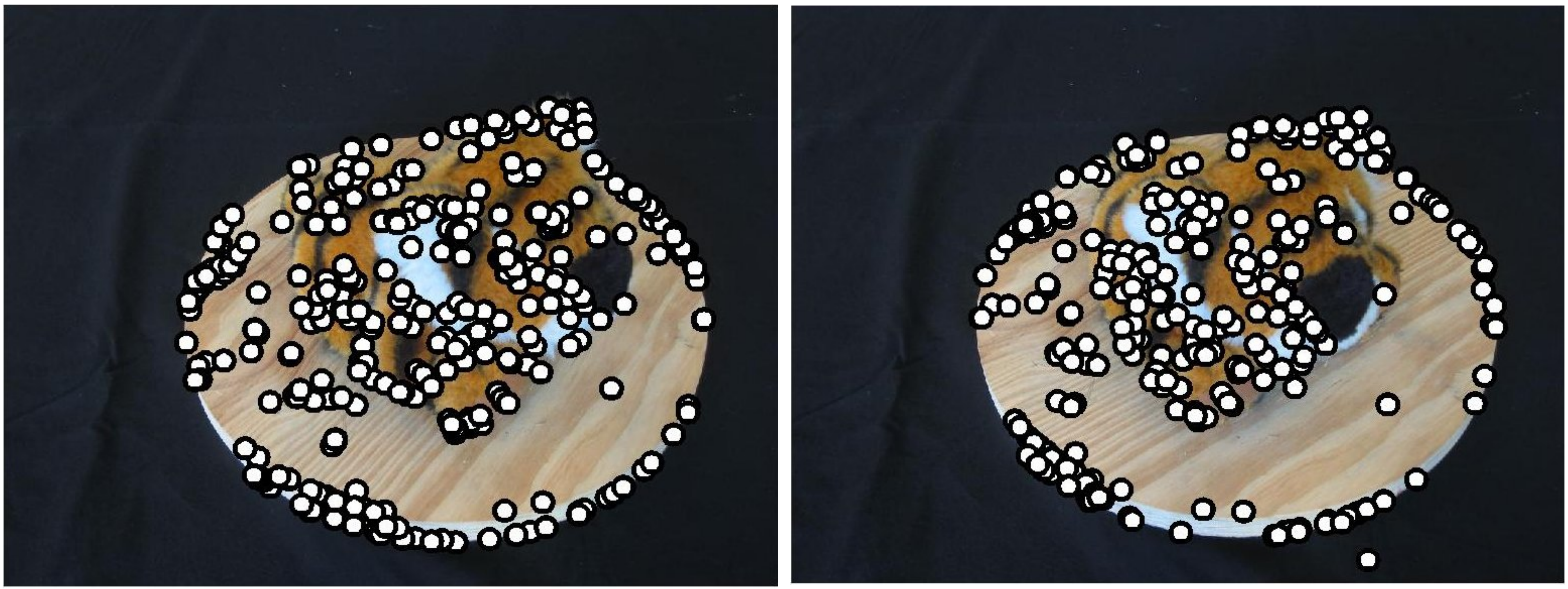
  can be solved using Singular Value Decomposition (SVD)

Epipolar Geometry

- Example: 2-view matching in 3D

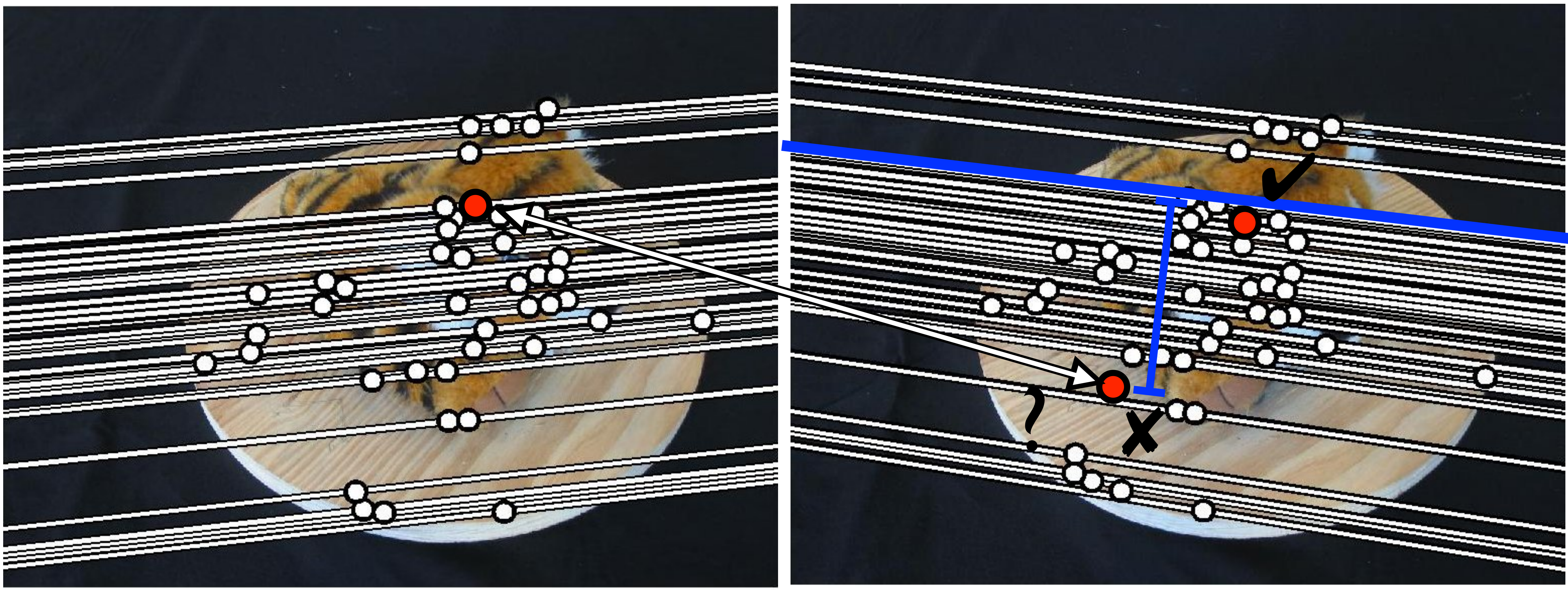

Epipolar Geometry

- Raw SIFT matches



Epipolar Geometry

- Epipolar lines



Can use RANSAC to find inliers with small distance from epipolar line

Epipolar Geometry

- Consistent matches

RANSAC for F

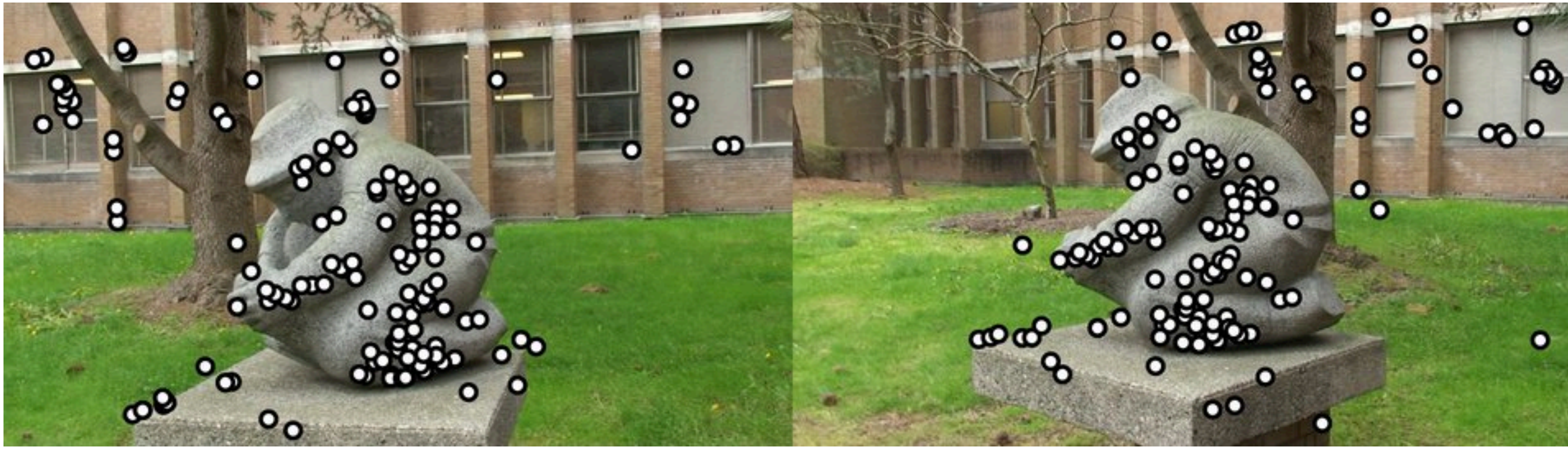
1. Match Features between 2 views
2. Randomly select set of 8 matches
3. Compute F using 8-point algorithm (SVD to solve $Af=0$)
4. Check consistency of all points with F, compute distances to epipolar lines and count #inliers with distance < threshold
5. Repeat steps 2-4 to maximise #inliers

Epipolar Lines from F

- What is the equation of the epipolar line for point x?

4.2

RANSAC for F

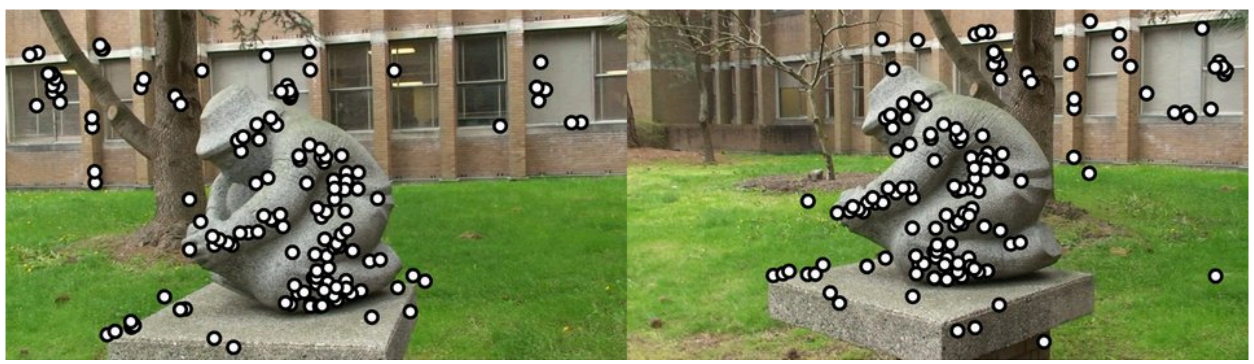


Raw feature matches (after ratio test filtering)

Solved for F and RANSAC inliers

2-view Structure from Motion

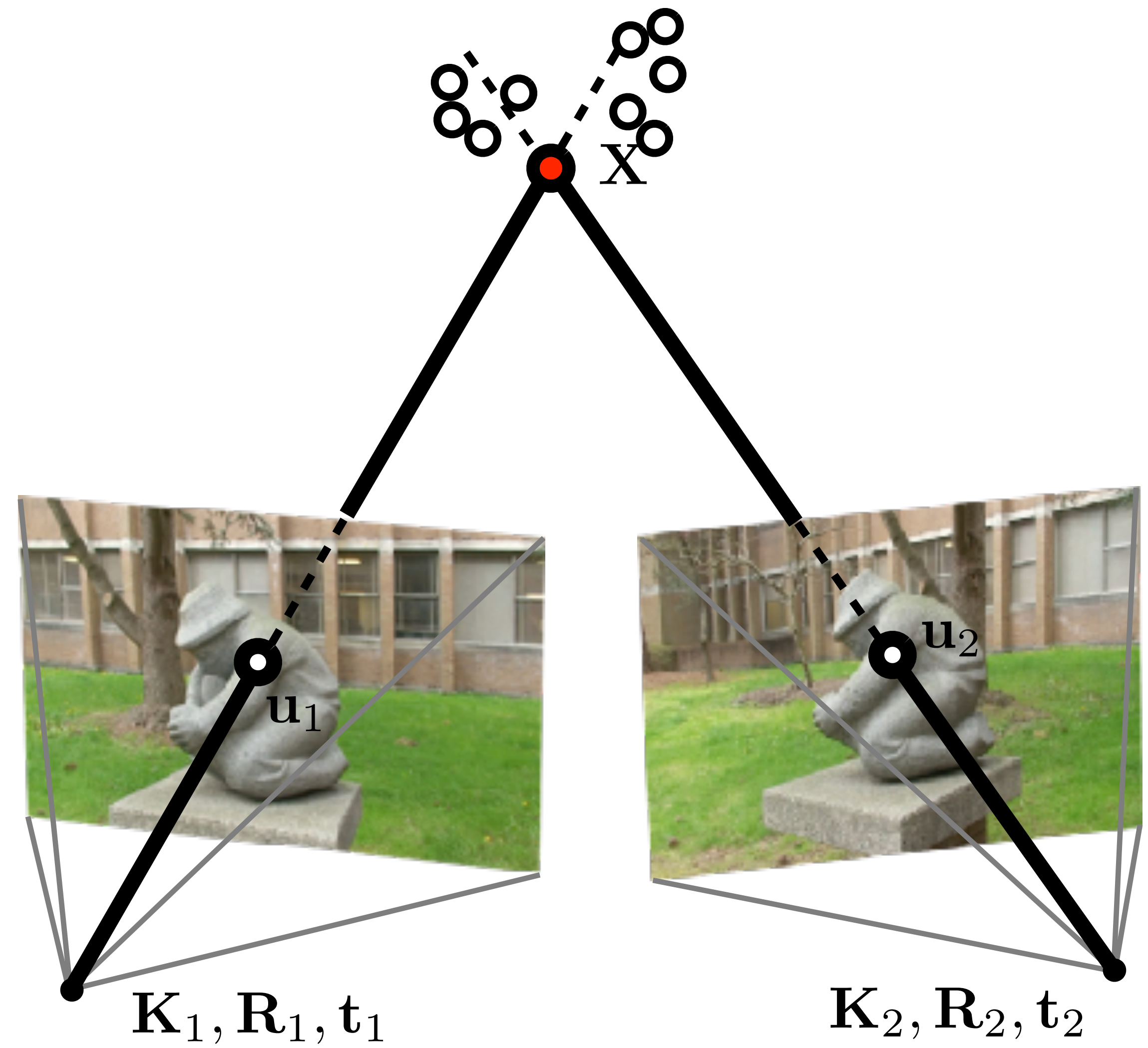
- We can use the combination of SIFT/RANSAC and triangulation to compute 3D structure from 2 views



Raw SIFT matches

RANSAC for F

Extract R, t



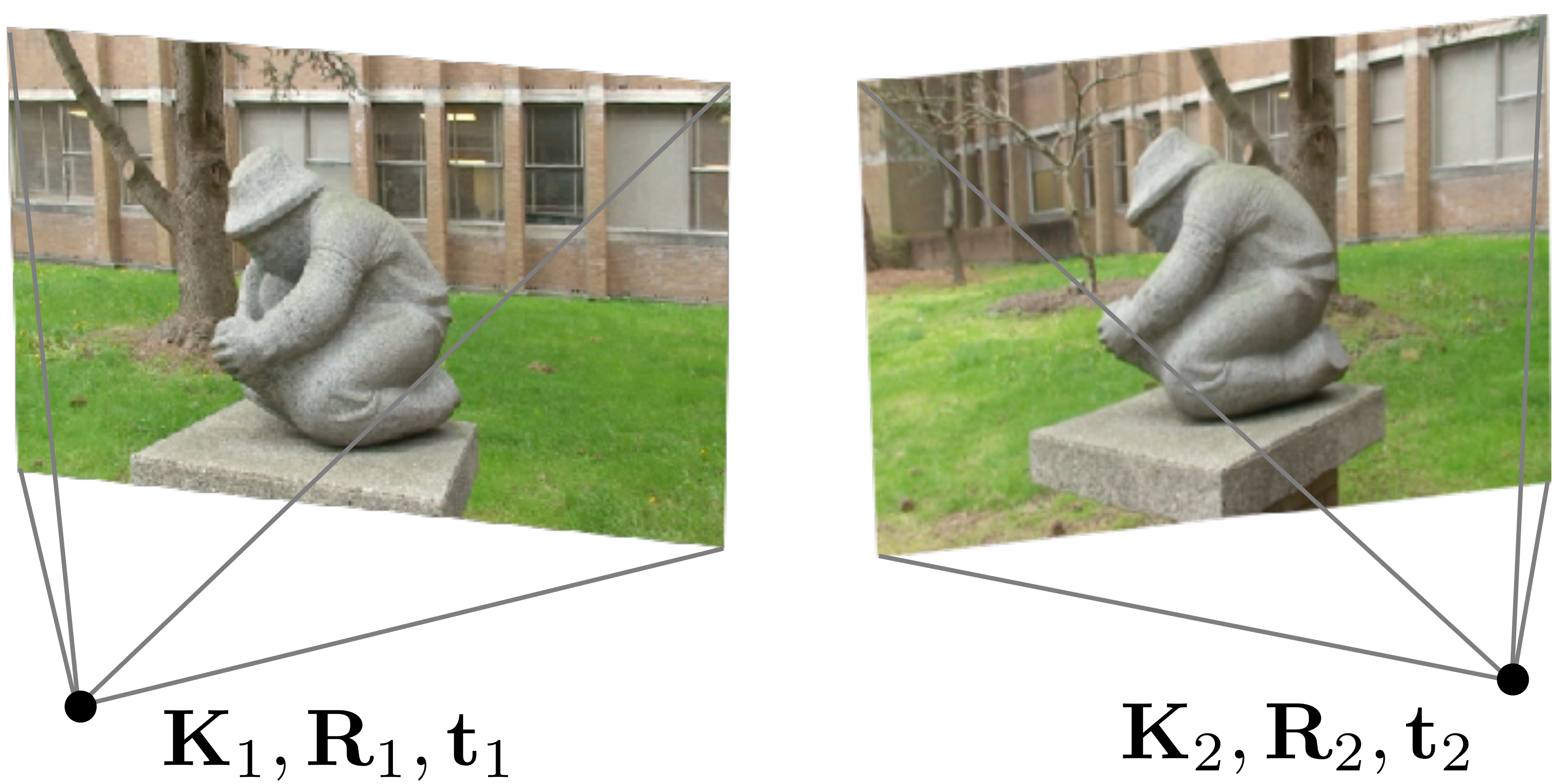
Triangulate to 3D Point Cloud

Cameras from F

- The Fundamental matrix is derived from the cameras

$$\mathbf{u}_2^T \underbrace{\mathbf{K}_2^{-T}\mathbf{R}_2^T(\mathbf{t}_2 - \mathbf{t}_1) \times \mathbf{R}_1\mathbf{K}_1^{-1}}_{\mathbf{F}} \mathbf{u}_1 = 0$$

Can we invert it to get the cameras from F?

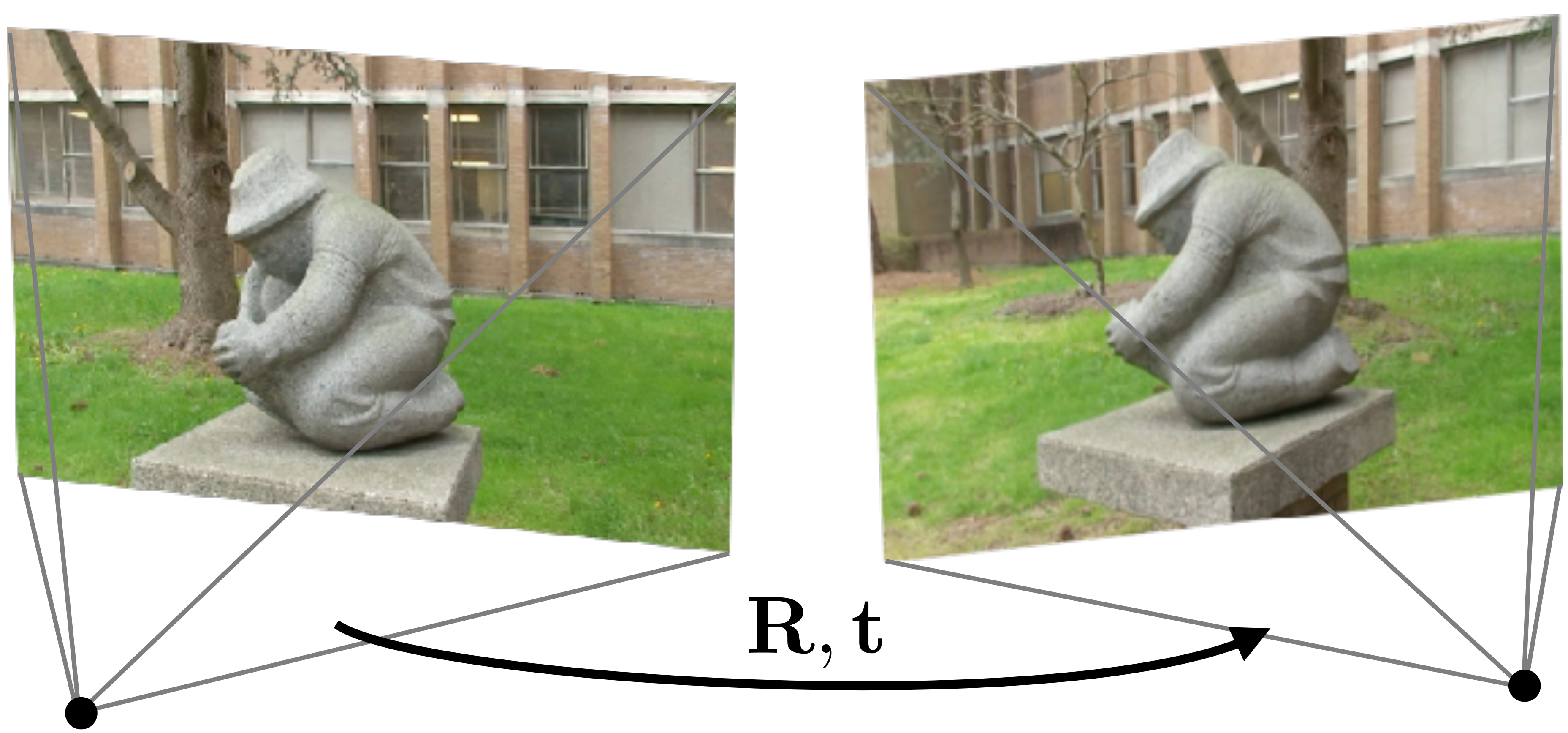


Cameras from F

- First simplify by writing in terms of relative translation/rotation and assume $\mathbf{K}_1, \mathbf{K}_2$ are known

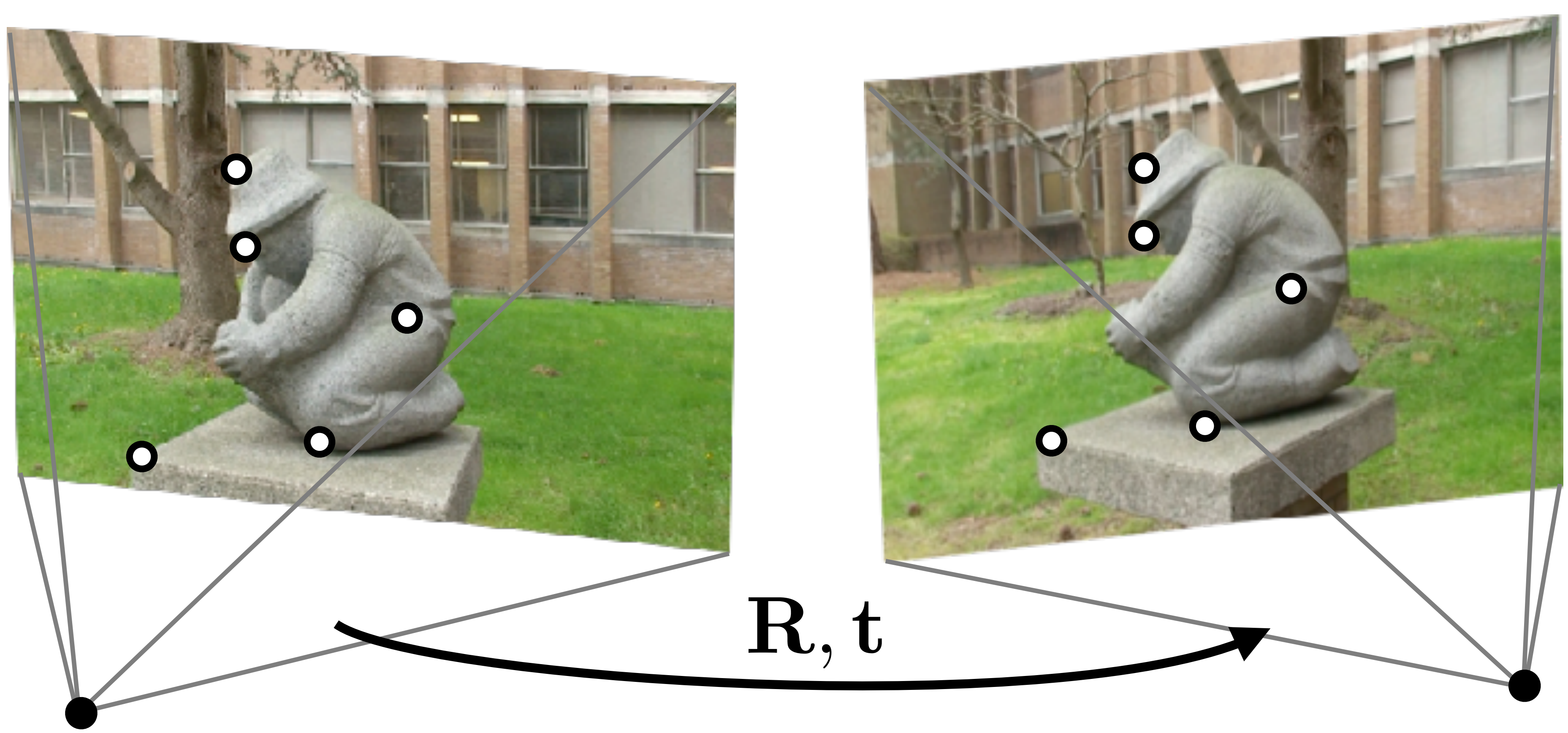
4.3

$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$ can be solved for **t**, **R** [Szeliski p350]



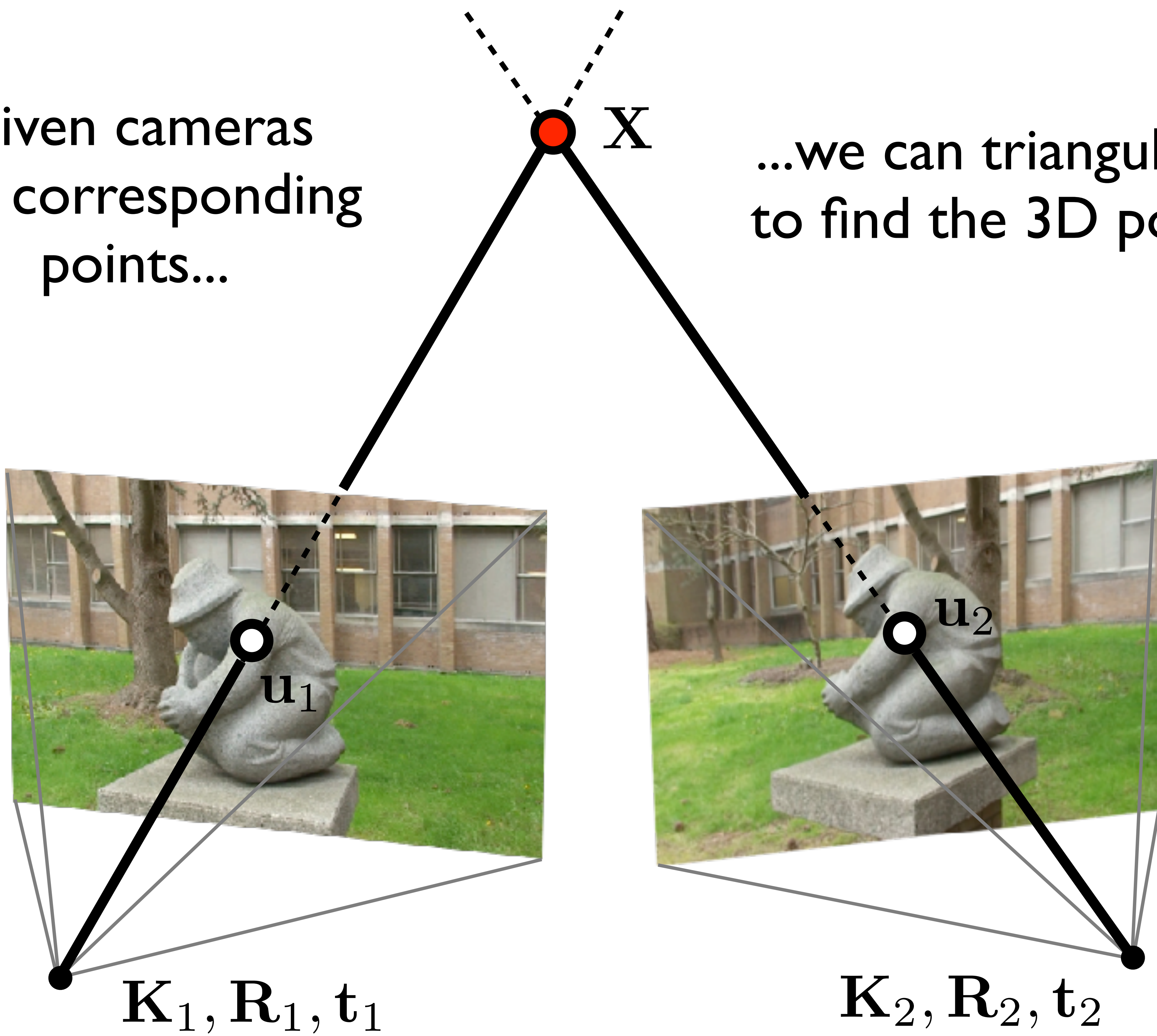
5 Point Algorithm

- Instead of using the 8 point algorithm to solve for F, we can directly solve for R and t using only 5 correspondences
- This involves solving a 10th degree polynomial [Nister 2004]
- Often we can guess the focal length (e.g., guess field of view), and solve for it later using bundle adjustment



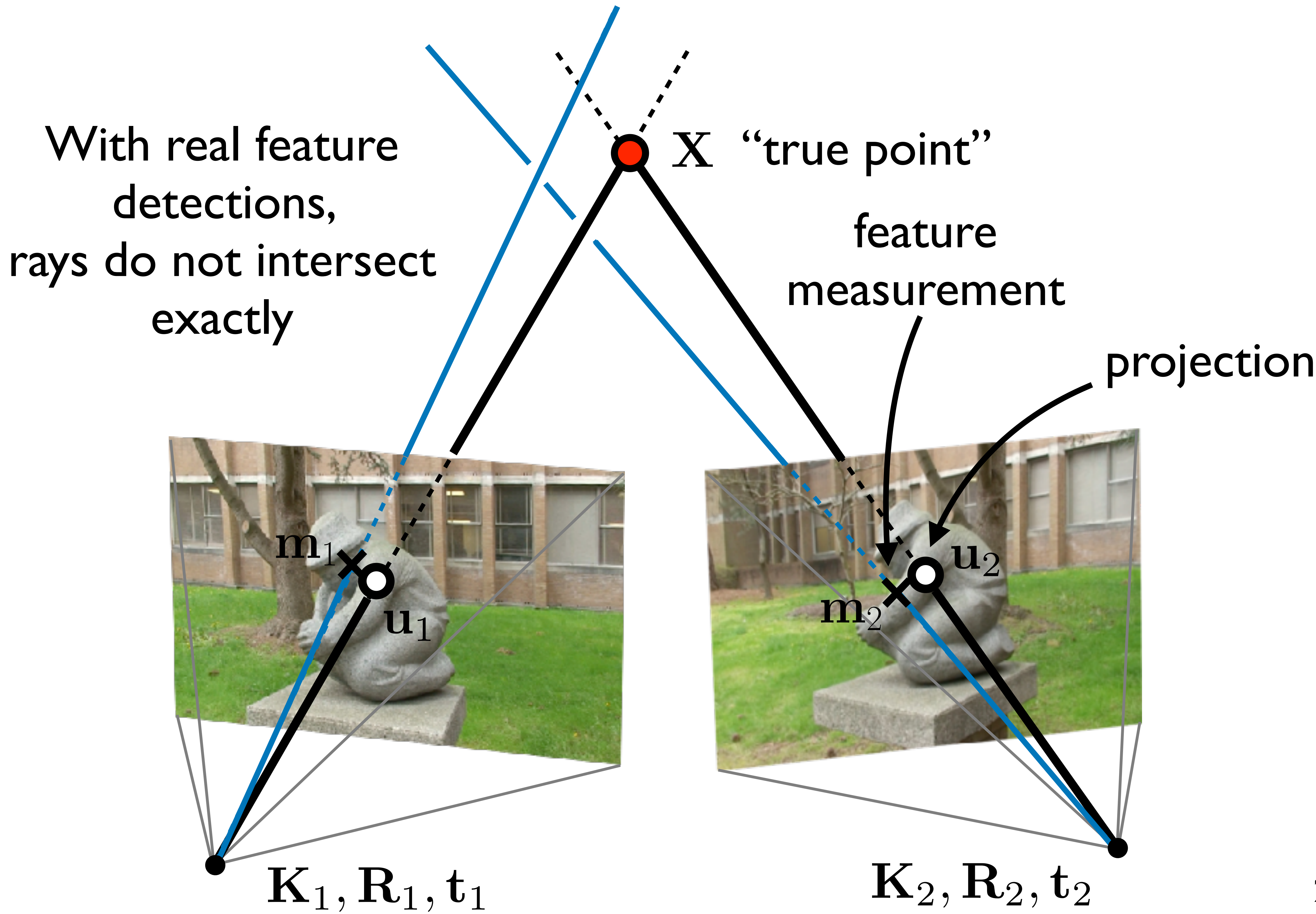
Triangulation

Given cameras and corresponding points...



...we can triangulate to find the 3D point

Triangulation



Triangulation

- We can solve for the 3D point X by minimising the closest approach of the rays in 3D (linear), or better find an X such that image measurement errors are minimised (non-linear)

 4.4

[Szeliski 7.1]

Recap: 2-view Geometry

- Planar geometry: one to one mapping of points

$$\mathbf{u} = \mathbf{H}\mathbf{x}$$

viewing a plane, rotation

Recap: 2-view Geometry

- Epipolar (3D) geometry: point to line mapping

$$\mathbf{u}^T \mathbf{F} \mathbf{x} = 0$$

moving camera, 3D scene

Next Lecture

- Multiview alignment, structure from motion