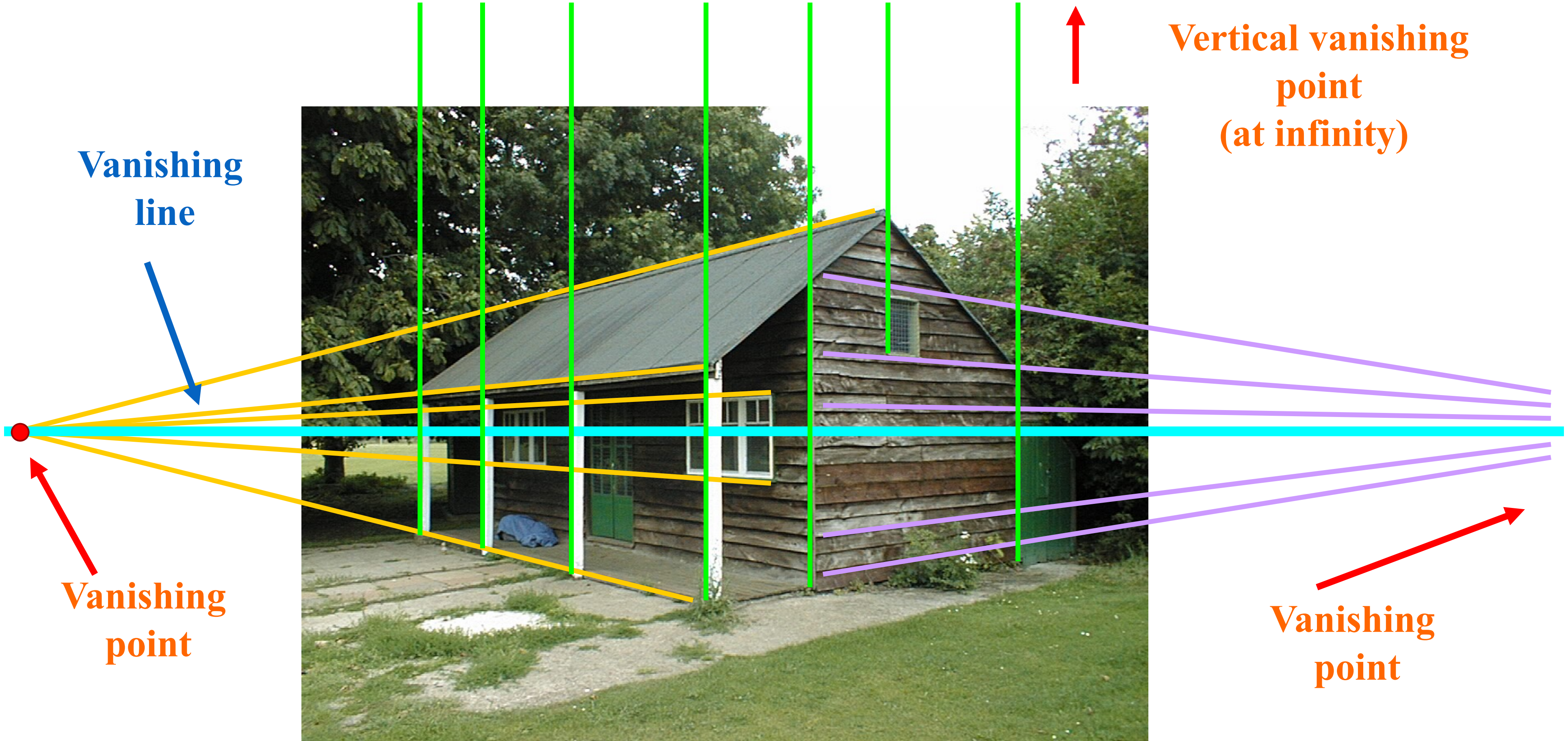
Calibrating the Camera

Method 2: Use vanishing points

- Find vanishing points corresponding to orthogonal directions



Calibration by orthogonal vanishing points

- Intrinsic camera matrix
  - Use orthogonality as a constraint
  - Model K with only f, u_0, v_0

For vanishing points

$$\mathbf{p}_i = \mathbf{KRX}_i \qquad \mathbf{X}_i{}^T\mathbf{X}_j = 0$$

$$\mathbf{X_i} = \mathbf{R^{-1}K^{-1}p_i}$$

$$\mathbf{p_i^\top}\left(\mathbf{K^{-1}}\right)^\top(\mathbf{R})(\mathbf{R^{-1}})(\mathbf{K^{-1}})\mathbf{p}_{\boldsymbol{j}} = \mathbf{0}$$

- What if you don't have three finite vanishing points?
  - Two finite VP: solve f, get valid u_0, v_0 closest to image center
  - One finite VP: u_0, v_0 is at vanishing point; can't solve for f

Calibration by vanishing points

- Intrinsic camera matrix

$$\mathbf{p}_i = \mathbf{KRX}_i$$

- Rotation matrix
  - Set directions of vanishing points
    - e.g., $\mathbf{X}_1 = [1, 0, 0]$
  - Each VP provides one column of **R**
  - Special properties of **R**
    - inv(**R**)=$\mathbf{R}^T$
    - Each row and column of **R** has unit length

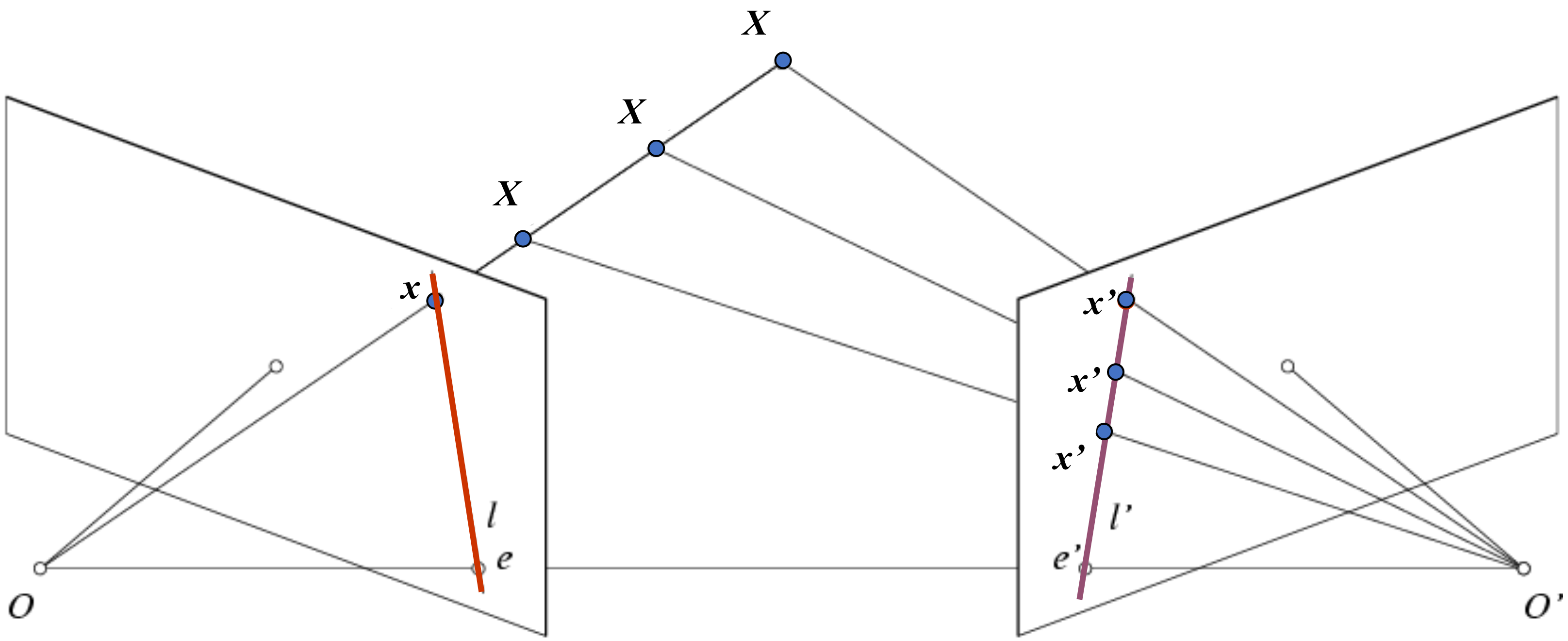
$$\mathbf{p_i} = \mathbf{Kr_i}$$

7. Epipolar Geometry and Stereo Vision

Epipolar constraint, Essential Matrix, Fundamental Matrix, Stereo 3D Reconstruction

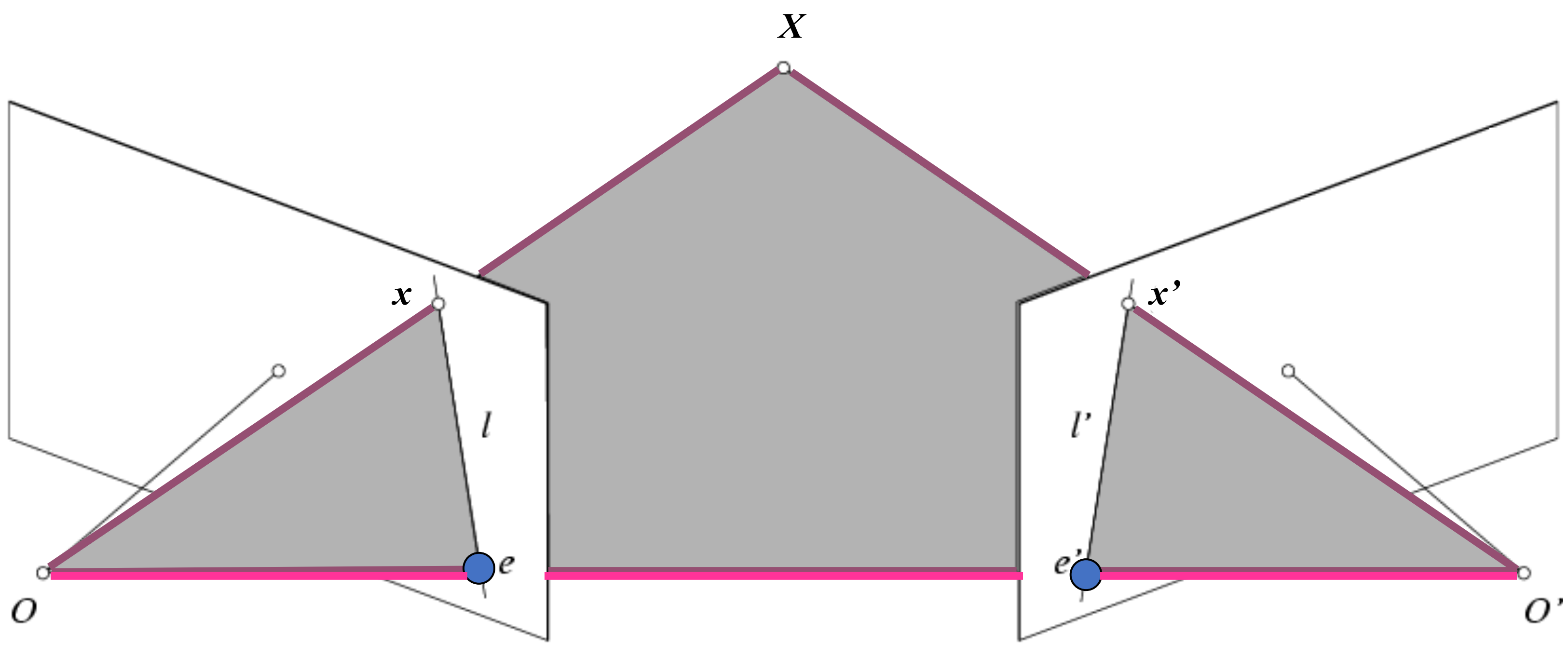
Many Slides from Prof. S. Seitz

Epipolar Geometry



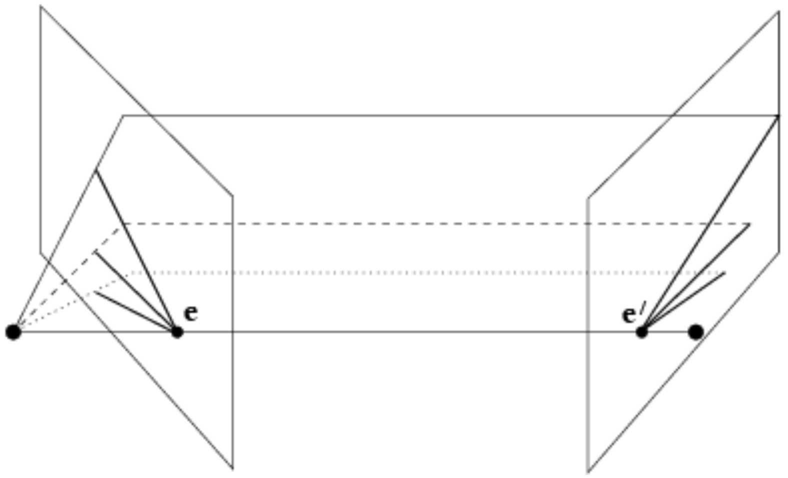
Relation between x and x’ if they project the point X

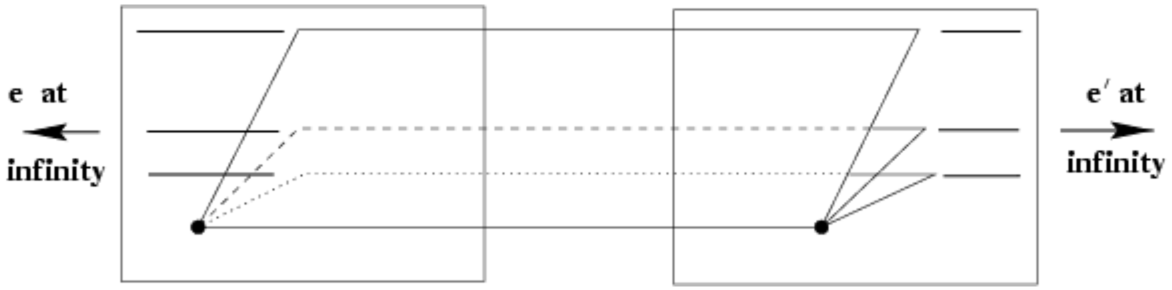
Epipolar geometry: notation



- **Baseline** – line connecting the two camera centers
- **Epipoles**
= intersections of baseline with image planes
= projections of the other camera center
- **Epipolar Plane** – plane containing baseline (1D family)

Epipolar geometry: notation



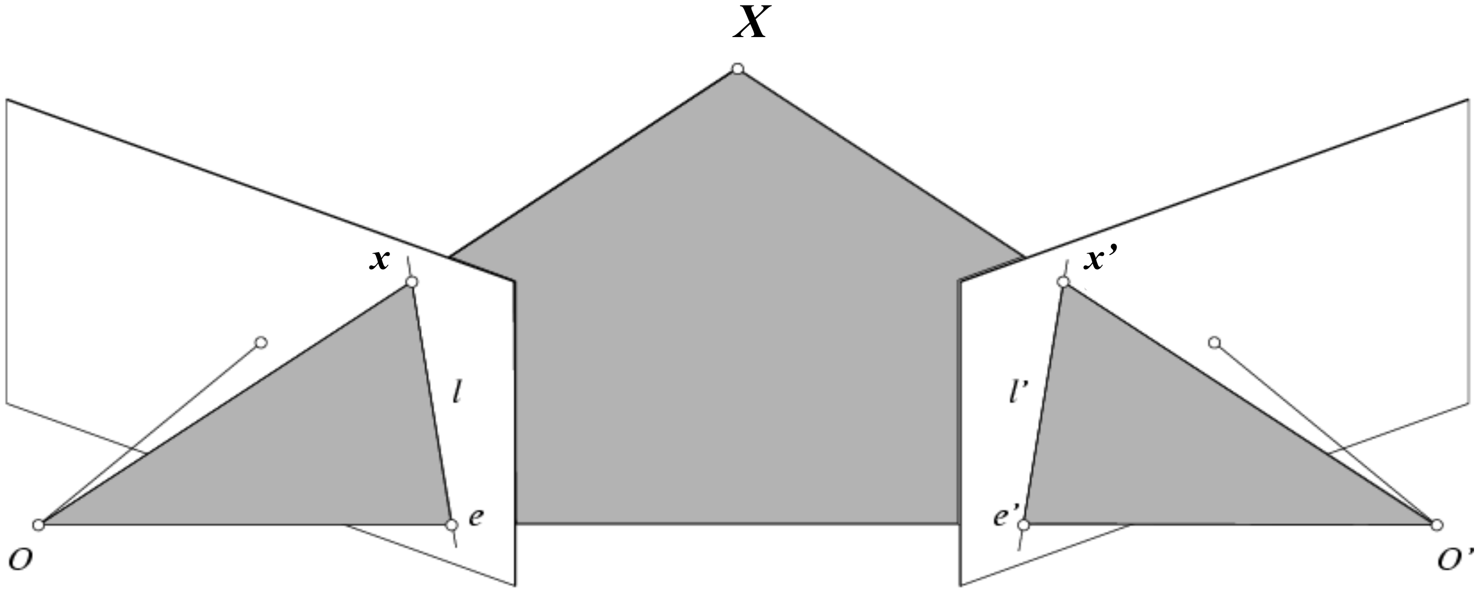


Converging lines:
Motion non-parallel to image plane

Parallel lines:
Motion parallel to image plane

What happens when camera moves forward/backward?

Epipolar constraint: Calibrated case



Given the intrinsic parameters of the cameras:

1. Convert to normalized coordinates by pre-multiplying all points with the inverse of the calibration matrix; set first camera's coordinate system to world coordinates

$$\hat{x} = K^{-1}x = X$$

Homogeneous 2d point (3D ray towards X)

2D pixel coordinate (homogeneous)

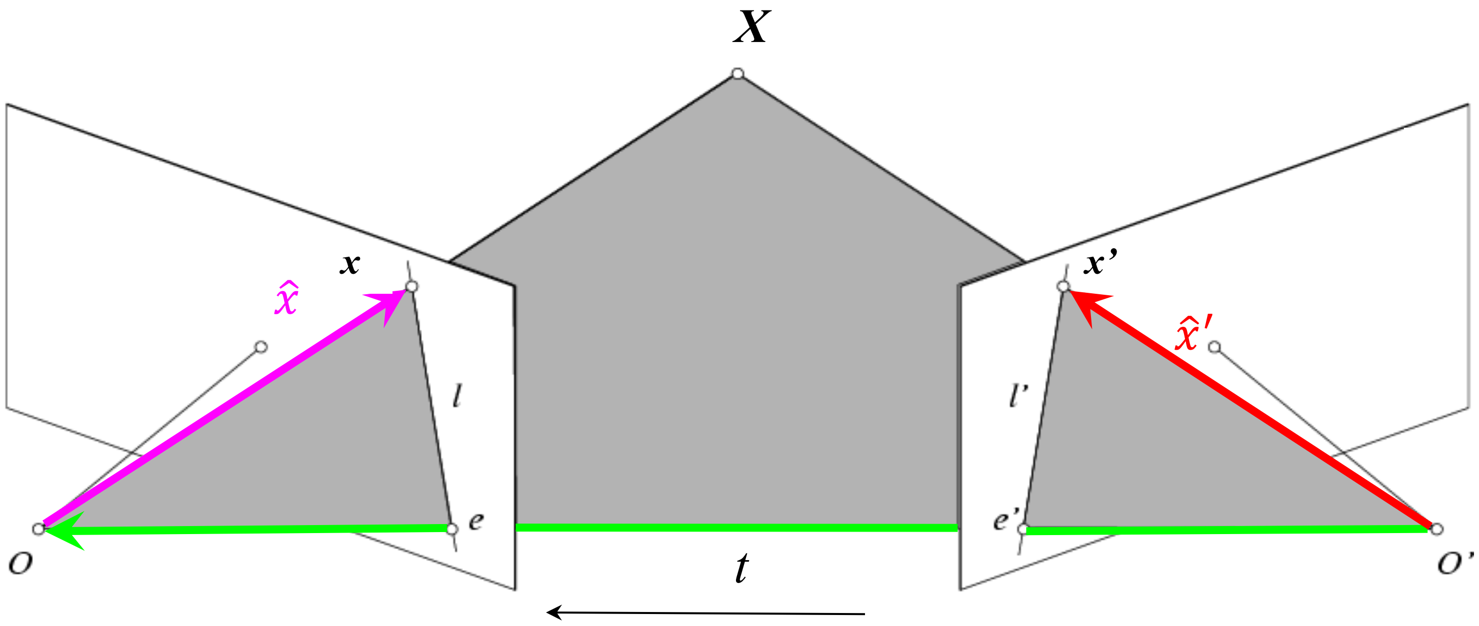
3D scene point

$$\boxed{\hat{x} = R\hat{x}' + t}$$

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

3D scene point in 2nd camera's 3D coordinates

Epipolar constraint: Calibrated case



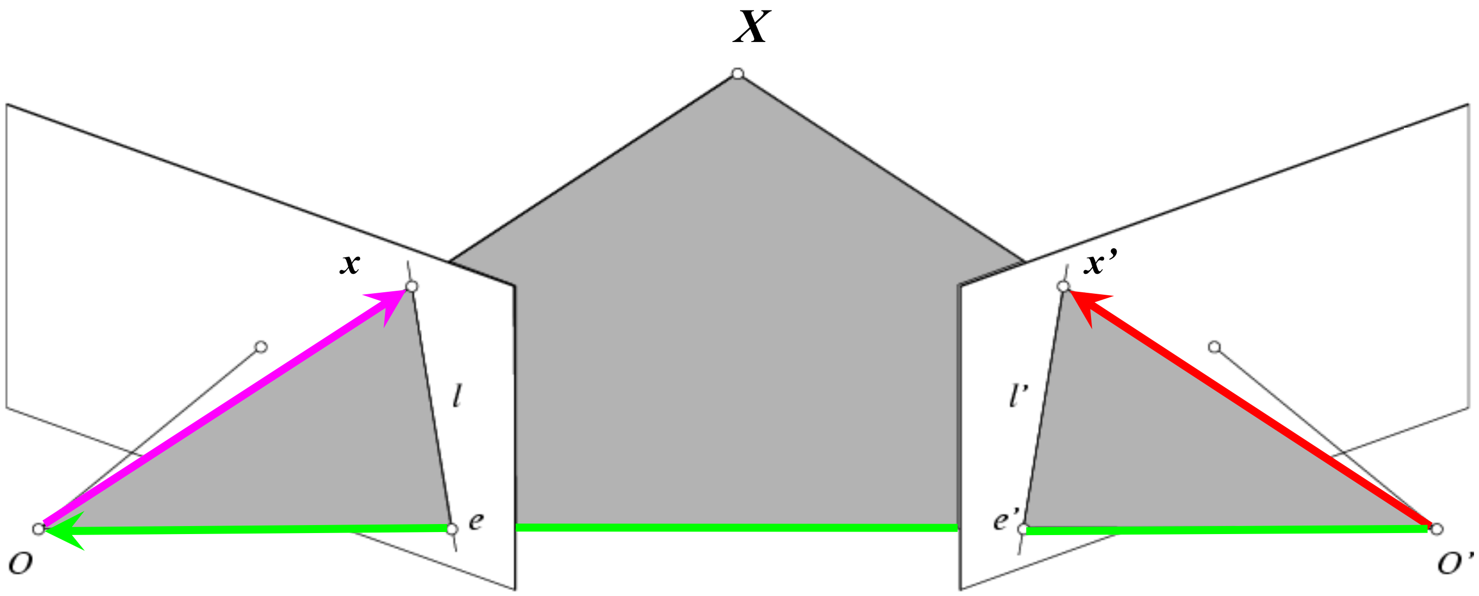
$$\hat{x} = K^{-1}x = X \qquad \hat{x}' = K'^{-1}x' = X'$$

$$\hat{x} = R\hat{x}' + t \quad \Rightarrow \quad \hat{x} \cdot [t \times (R\hat{x}')] = 0$$

(because $\hat{x}$, $R\hat{x}'$, and t are co-planar)

$$t\times\hat{x} = t\times(R\,\hat{x}' + t) = \mathrm{t}\times(R\,\hat{x}') \quad \Rightarrow \quad \hat{x}\cdot(t\times\hat{x}) = \hat{x}[\,\mathrm{t}\times(R\,\hat{x}')] = 0$$

Essential matrix



$$\hat{x} \cdot [t \times (R\hat{x}')] = 0 \quad \Rightarrow \quad \hat{x}^T E \hat{x}' = 0 \quad \text{with} \quad E = [t]_\times R$$

$$\mathbf{a} = (a_1 \; a_2 \; a_3)^\mathrm{T}$$

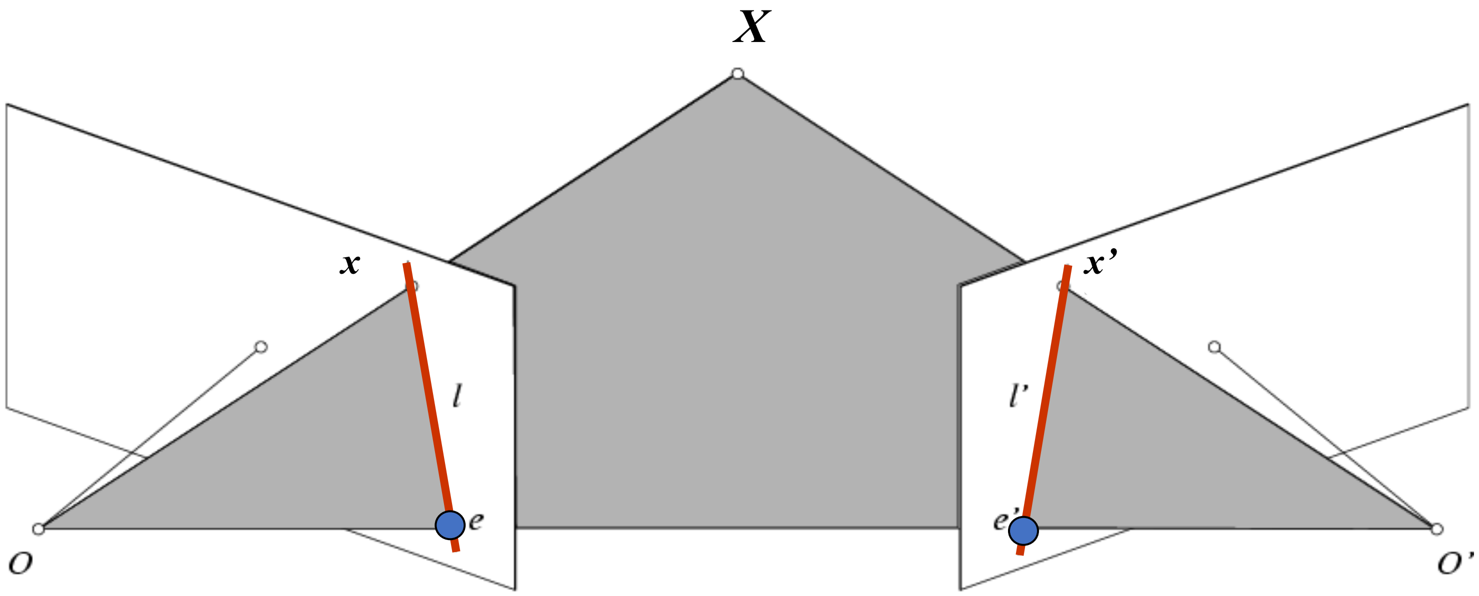
$$\mathbf{b} = (b_1 \; b_2 \; b_3)^\mathrm{T}$$

$$[\mathbf{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \quad \mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b}$$

$\Downarrow$

Essential Matrix
(Longuet-Higgins, 1981)

Properties of the Essential matrix



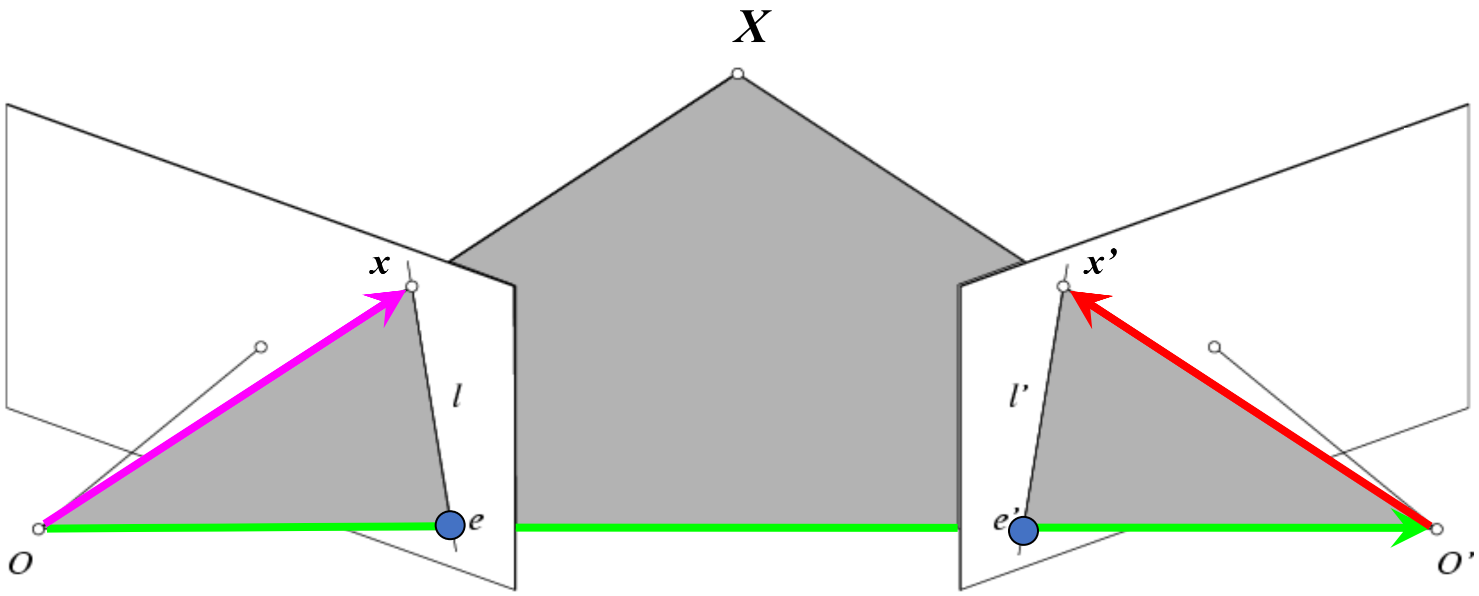
$$\hat{x} \cdot [t \times (R\hat{x}')] = 0 \quad \Rightarrow \quad \hat{x}^T E \hat{x}' = 0 \quad \text{with} \quad E = [t]_\times R$$

Drop ^ below to simplify notation

Skew-symmetric matrix

- $E\,x'$ is the epipolar line associated with x' ($l = E\,x'$)
- $E^T x$ is the epipolar line associated with x ($l' = E^T x$)
- $E\,e' = 0$ and $E^T e = 0$
- E is singular (rank two)
- E has five degrees of freedom
  - (3 for R, 2 for t because it's up to a scale)

Epipolar constraint: Uncalibrated case



- If we don't know *K* and *K'*, then we can write the epipolar constraint in terms of *unknown* normalized coordinates:

$$\hat{x}^T E \hat{x}' = 0 \qquad\qquad x = K\hat{x}, \quad x' = K'\hat{x}'$$

The Fundamental Matrix

Without knowing K and K', we can define a similar relation using *unknown* normalized coordinates

$$\hat{x}^T E \hat{x}' = 0$$

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

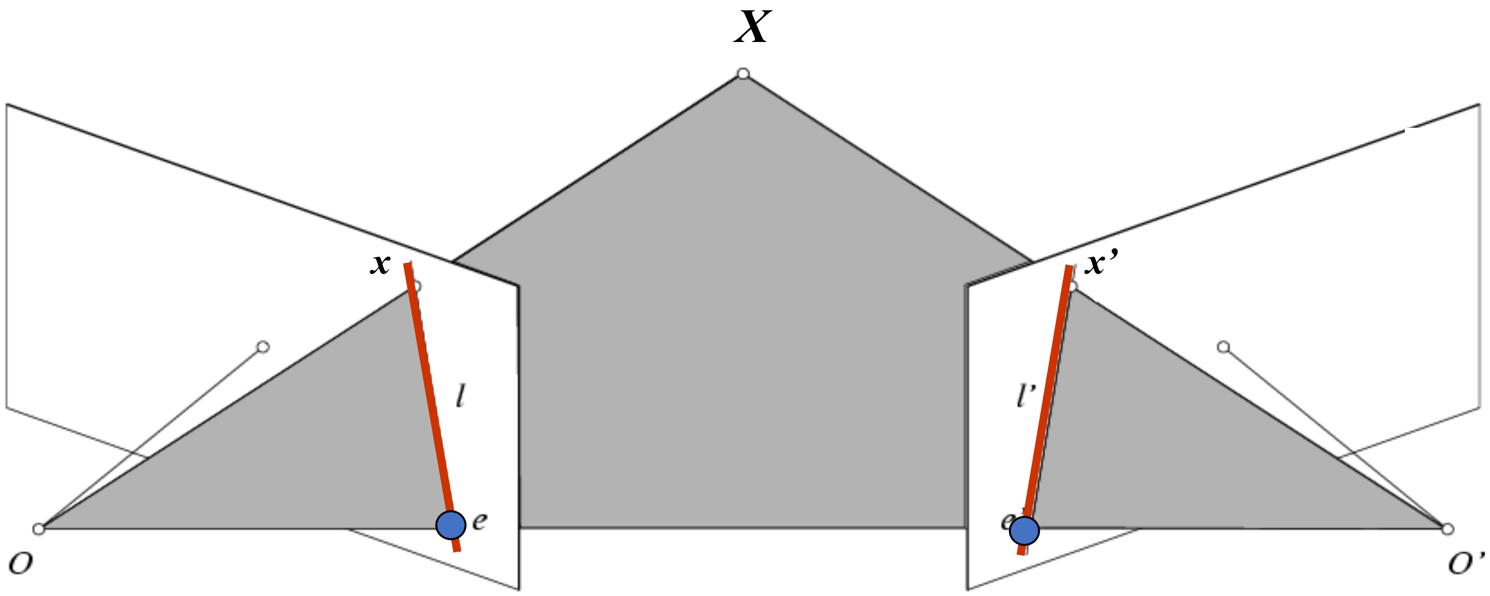
$$\hat{x}' = K'^{-1} x'$$

$$\Rightarrow \quad x^T F x' = 0 \quad \text{with} \quad F = K^{-T} E K'^{-1}$$

Fundamental Matrix
(Faugeras and Luong, 1992)

- Estimating the fundamental matrix is known as "weak calibration"
- If we know the calibration matrices of the two cameras, we can estimate the essential matrix: $E = K^T F K'$
- The essential matrix gives us the relative rotation and translation between the cameras, or their extrinsic parameters

Properties of the Fundamental matrix



$$x^T F x' = 0 \quad \text{with} \quad F = K^{-T} E K'^{-1}$$

- $F x$ is the epipolar line associated with x' $(l' = F x)$
- $F^T x'$ is the epipolar line associated with x $(l = F^T x')$
- $F e' = 0$ and $F^T e = 0$
- F is singular (rank two): det(F)=0
- F has seven degrees of freedom: 9 entries but defined up to scale, det(F)=0

Estimating F: 8-point algorithm

1. Solve a system of homogeneous linear equations
   a. Write down the system of equations
   b. For det(F) = 0, use SVD: S(3,3) = 0

$$\mathbf{x}^T F \mathbf{x'} = 0$$

$$uu'f_{11} + uv'f_{12} + uf_{13} + vu'f_{21} + vv'f_{22} + vf_{23} + u'f_{31} + v'f_{32} + f_{33} = 0$$

$$\mathrm{A}\boldsymbol{f} = \begin{bmatrix} u_1 u_1{}' & u_1 v_1{}' & u_1 & v_1 u_1{}' & v_1 v_1{}' & v_1 & u_1{}' & v_1{}' & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n u_v' & u_n v_n{}' & u_n & v_n u_n{}' & v_n v_n{}' & v_n & u_n{}' & v_n{}' & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ \vdots \\ f_{33} \end{bmatrix} = \mathbf{0}$$

8-point algorithm

1. Solve a system of homogeneous linear equations
   a. Write down the system of equations
   b. Solve **f** from A**f**=**0** using SVD
2. Resolve det(F) = 0 constraint by SVD

Notes:

- Use RANSAC to deal with outliers (sample 8 points)
  - How to test for outliers?

$|x'F\ x\ | < threshold?$

- Solve in normalized coordinates
  - mean=0
  - standard deviation ~= (1,1,1)
  - just like with estimating the homography for stitching

Homography vs Fundamental Matrix

Assume we have matched points x $\Leftrightarrow$ x' with outliers

Homography (No Translation)

- Correspondence Relation

$$\mathbf{x}' = \mathbf{H}\mathbf{x} \Rightarrow \mathbf{x}' \times \mathbf{H}\mathbf{x} = \mathbf{0}$$

1. Normalize image coordinates

$$\tilde{\mathbf{x}} = \mathbf{T}\mathbf{x} \quad \tilde{\mathbf{x}}' = \mathbf{T}'\mathbf{x}'$$

2. RANSAC with 4 points
   - Solution via SVD
3. De-normalize: $\mathbf{H} = \mathbf{T}'^{-1}\tilde{\mathbf{H}}\mathbf{T}$

Fundamental Matrix (Translation)

- Correspondence Relation

$$\mathbf{x}'^{T}\mathbf{F}\mathbf{x} = 0$$

1. Normalize image coordinates

$$\tilde{\mathbf{x}} = \mathbf{T}\mathbf{x} \quad \tilde{\mathbf{x}}' = \mathbf{T}'\mathbf{x}'$$

2. RANSAC with 8 points
   - Initial solution via SVD
   - Enforce $\det\left(\tilde{\mathbf{F}}\right) = 0$ by SVD
3. De-normalize: $\mathbf{F} = \mathbf{T}'^{T}\tilde{\mathbf{F}}\mathbf{T}$

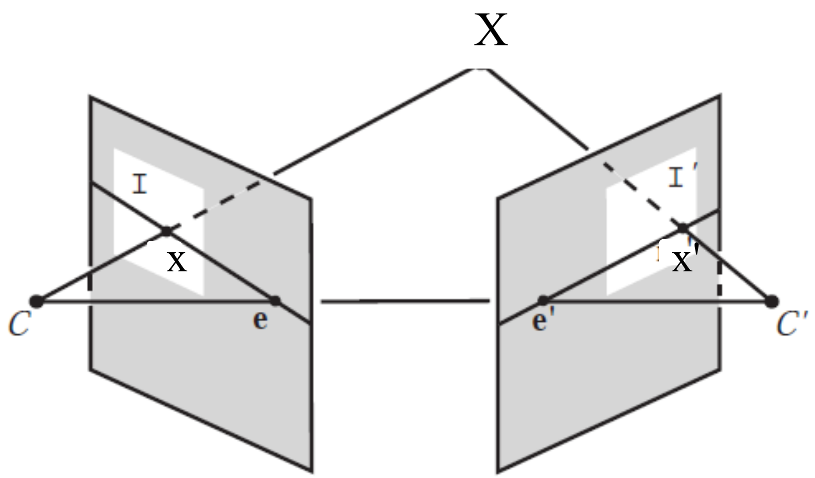
Estimating the Fundamental Matrix

- 8-point algorithm
  - Least squares solution using SVD on equations from 8 pairs of correspondences
  - Enforce det(F)=0 constraint using SVD on F
- 7-point algorithm
  - Use least squares to solve for null space (two vectors) using SVD and 7 pairs of correspondences
  - Solve for linear combination of null space vectors that satisfies det(F)=0
- Minimize reprojection error
  - Non-linear least squares

Note: estimation of F (or E) is degenerate for a planar scene.

“Gold standard” algorithm

- Use 8-point algorithm to get initial value of F
- Use F to solve for P and P’ (discussed later)
- Jointly solve for 3d points **X** and **F** that minimize the squared re-projection error



See Algorithm 11.2 and Algorithm 11.3 in Hartley and Zissermann (pages 284-285) for details

Let’s recap…

- Fundamental matrix song

Stereo Reonstruction

Fuse a calibrated binocular stereo pair to produce a depth image

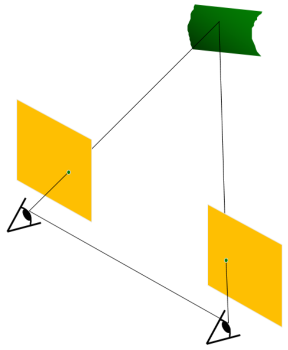
image 1

image 2

Dense depth map

Basic stereo matching algorithm

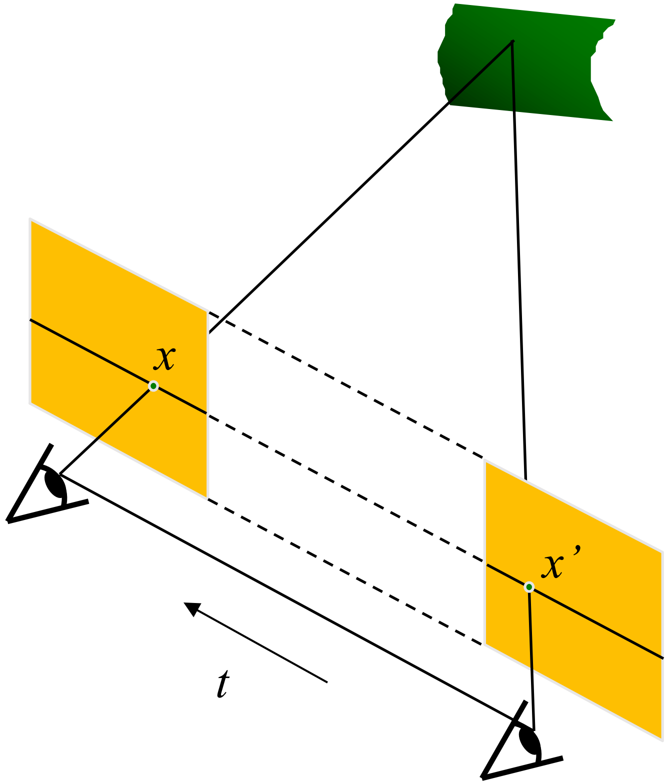
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Search along epipolar line and pick the best match
  - Triangulate the matches to get depth information



Simplest Case: Parallel images

- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same
- Then, epipolar lines fall along the horizontal scan lines of the images

Simplest Case: Parallel images



Epipolar constraint:

$$x^T E x' = 0, \quad E = t \times R$$

$$R = I \qquad t = (T, 0, 0)$$

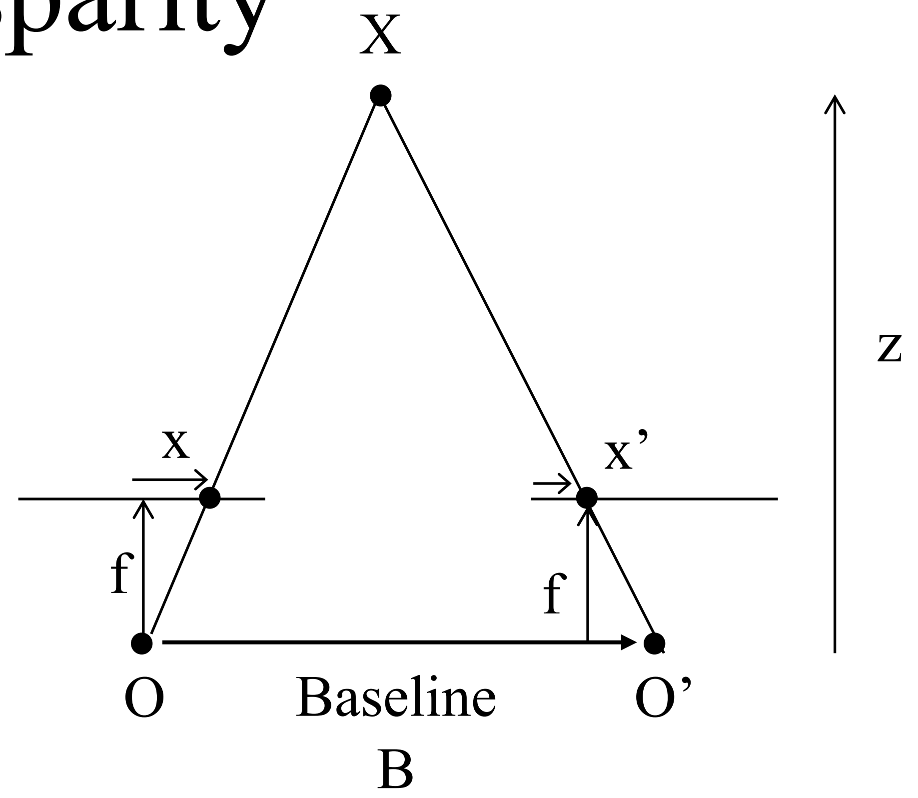
$$E = t \times R = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

$$\begin{pmatrix} u & v & 1 \end{pmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = 0 \qquad \begin{pmatrix} u & v & 1 \end{pmatrix} \begin{pmatrix} 0 \\ -T \\ Tv' \end{pmatrix} = 0 \qquad Tv = Tv'$$

The y-coordinates of corresponding points are the same

Depth from disparity

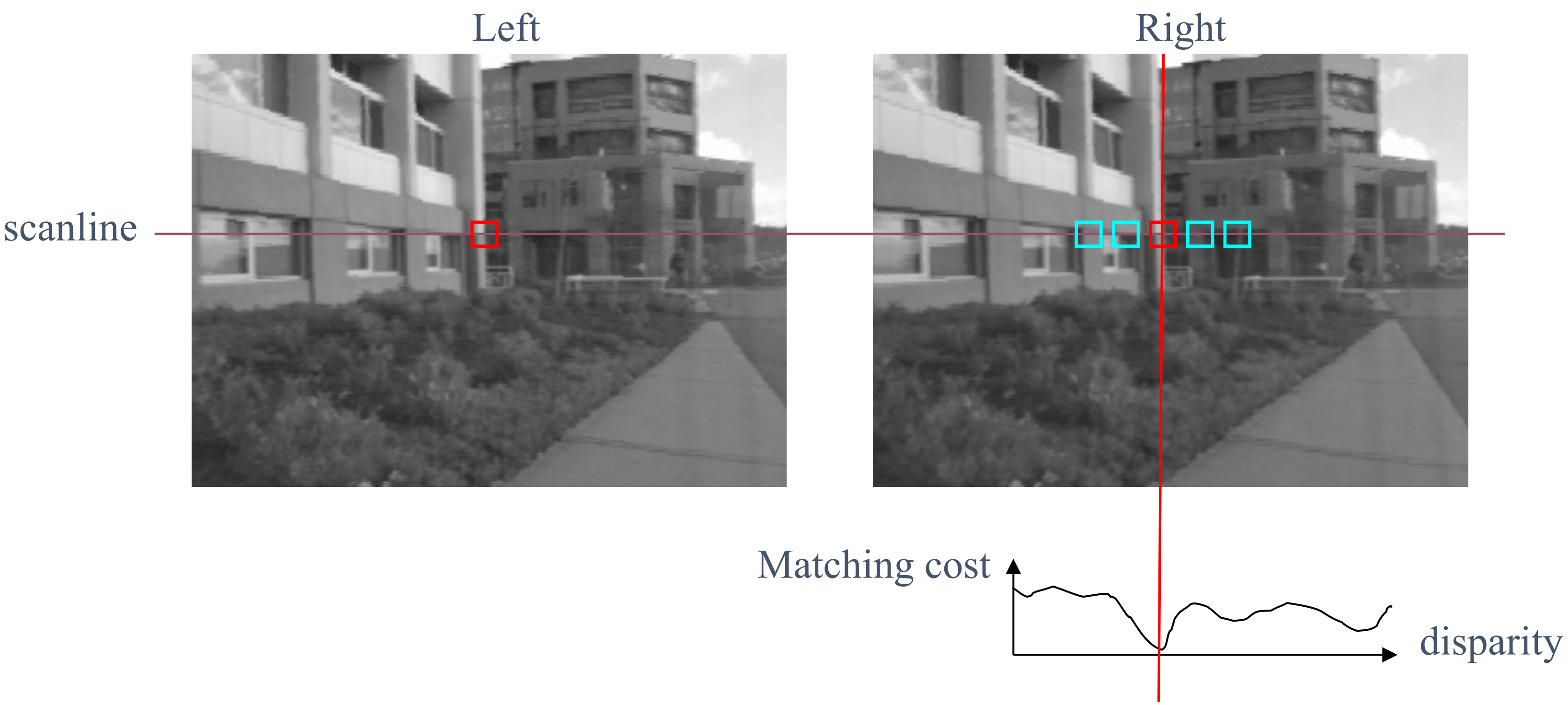
$$\frac{x - x'}{O - O'} = \frac{f}{z}$$



$$disparity = x - x' = \frac{B \cdot f}{z}$$

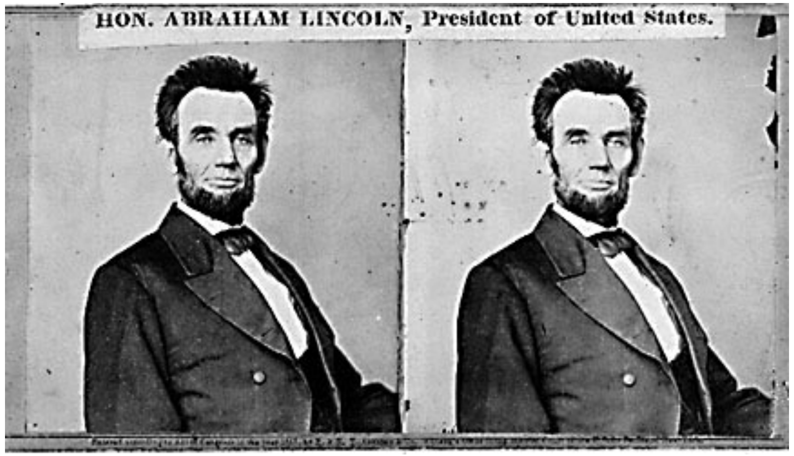
Disparity is inversely proportional to depth.

Correspondence search



- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

Failures of correspondence search



Textureless surfaces

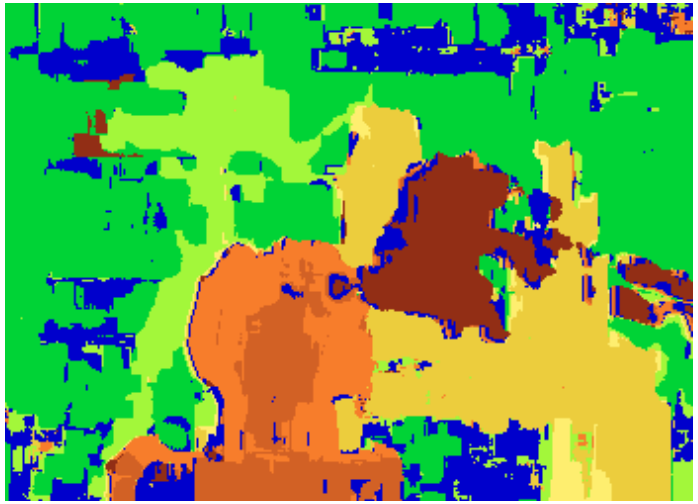
Occlusions, repetition

Non-Lambertian surfaces, specularities

Results with window search

Data

Window-based matching

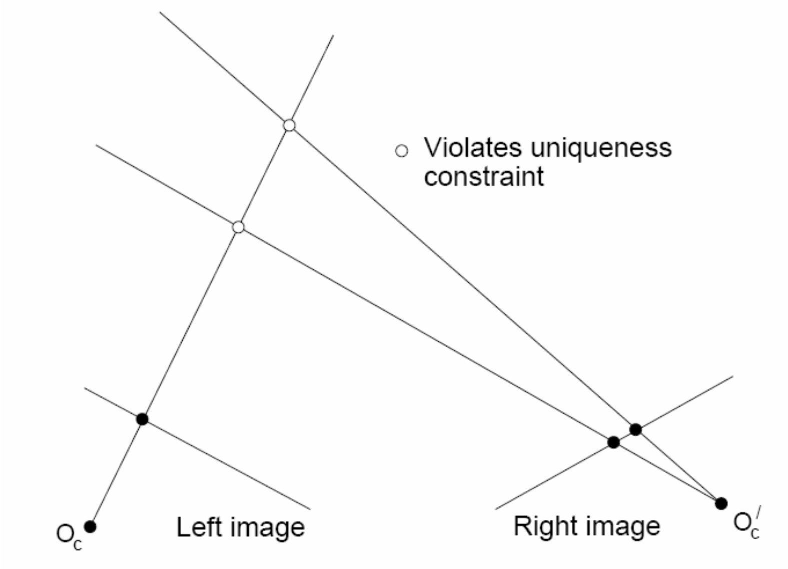


Ground truth

Improve window-based matching: Graph-Cut

So far, matches are independent for each point. Add constraints or priors

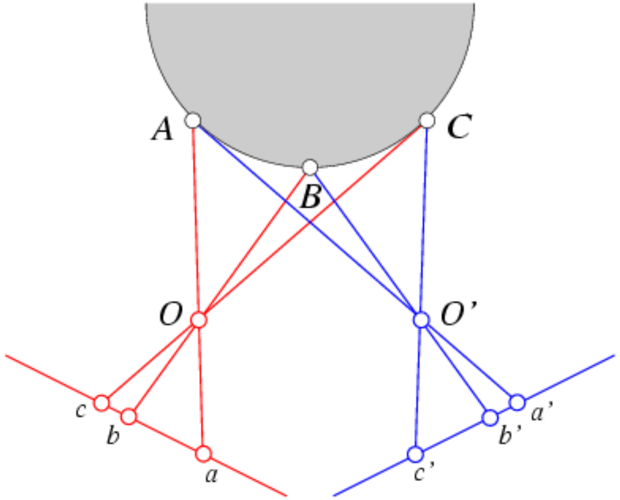
- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image

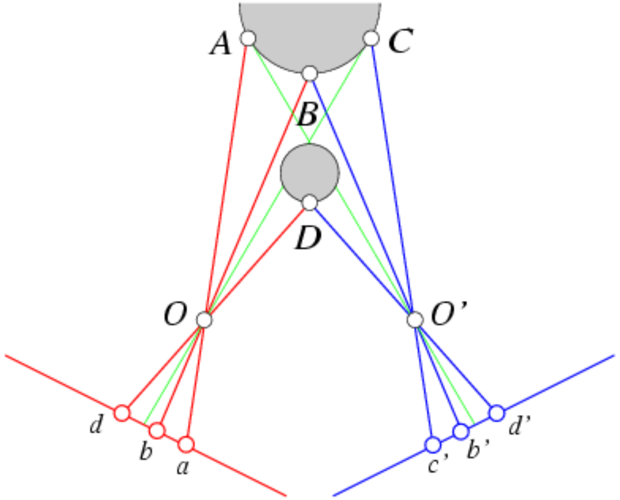


Improve window-based matching

So far, matches are independent for each point. Add constraints or priors

- Ordering
  - Corresponding points should be in the same order in both views



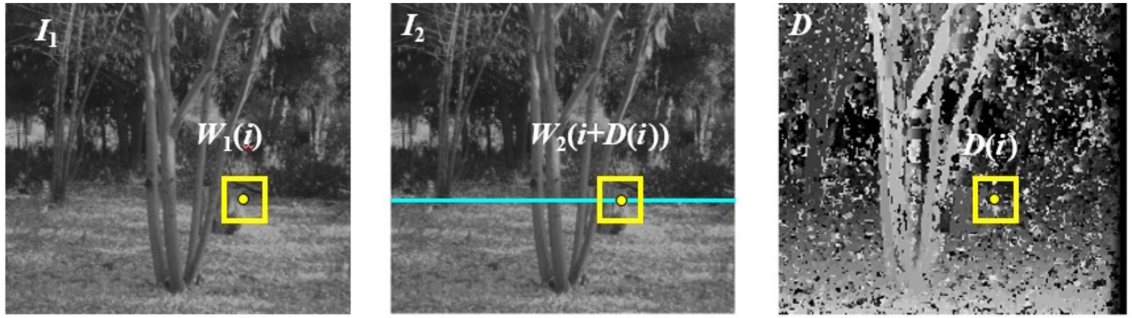


Ordering constraint doesn't hold

Improve window-based matching

So far, matches are independent for each point. Add constraints or priors

- Smoothness: We expect disparity values to change slowly (for the most part)



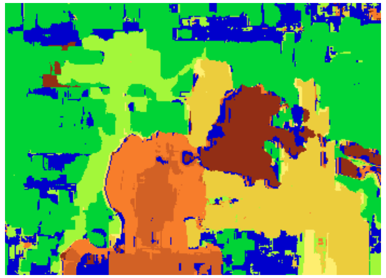
$$E = E_{\text{data}}(D; I_1, I_2) + \beta E_{\text{smooth}}(D)$$

$$E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2 \qquad E_{\text{smooth}} = \sum_{\text{neighbors } i,j} \left\| D(i) - D(j) \right\|^2$$

- Energy functions of this form can be minimized using *graph cuts*

Y. Boykov, O. Veksler, and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Improve window-based matching

- Before: 
- Unique, Ordering and Smoothness (Added using Graph-cut)

Graph cuts

Ground truth

Y. Boykov, O. Veksler, and R. Zabih, [Fast Approximate Energy Minimization via Graph Cuts](), PAMI 2001

Lab4: F estimation

- Take 2 images of same scene and find F using 8-point algorithm.
- What happens if you don't enforce det(F) =0?
- What happens if you don't normalize?

Lab4: Stereo reconstruction with non-parallel cameras

- Take 2 images with non-parallel motion of the camera and perform stereo reconstruction.