Computer Vision

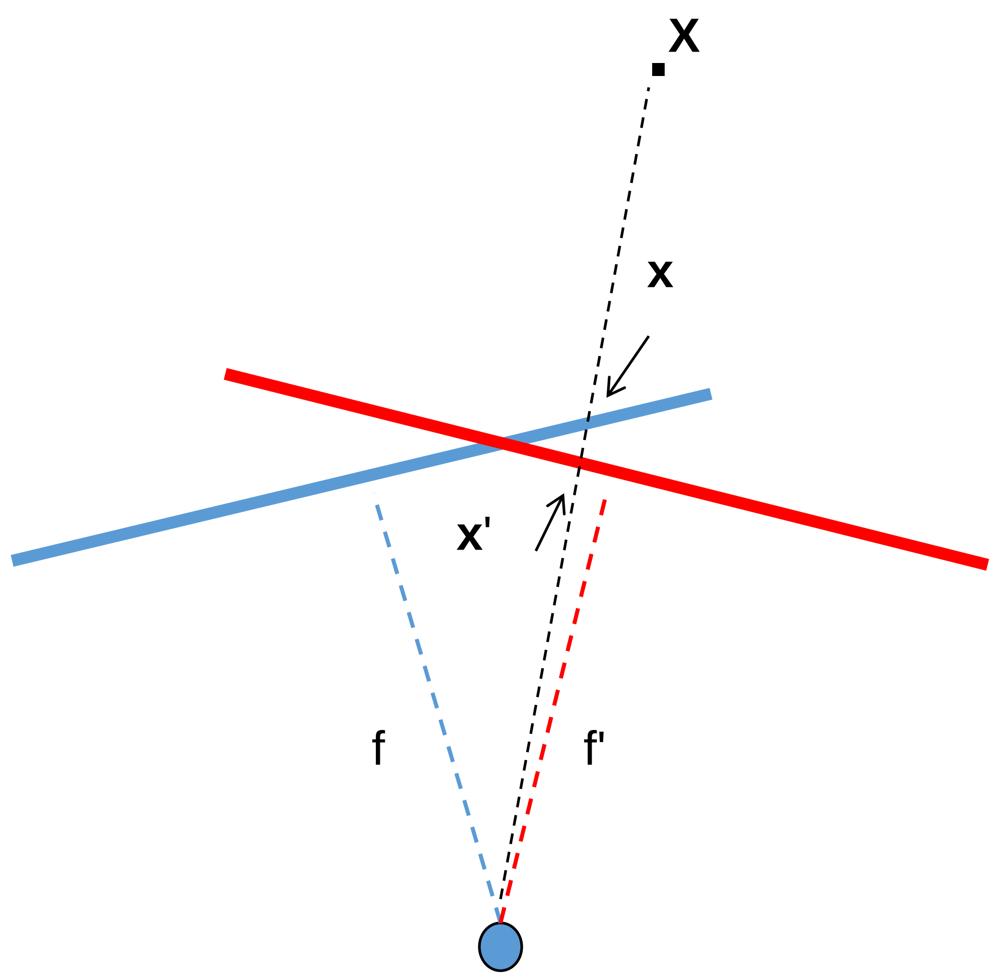
Epipolar Geometry and Stereo Vision

Dr. Mrinmoy Ghorai

Indian Institute of Information Technology
Sri City, Chittoor

Last class: Image Stitching

- Two images with rotation/zoom but no translation



Perspective and 3D Geometry

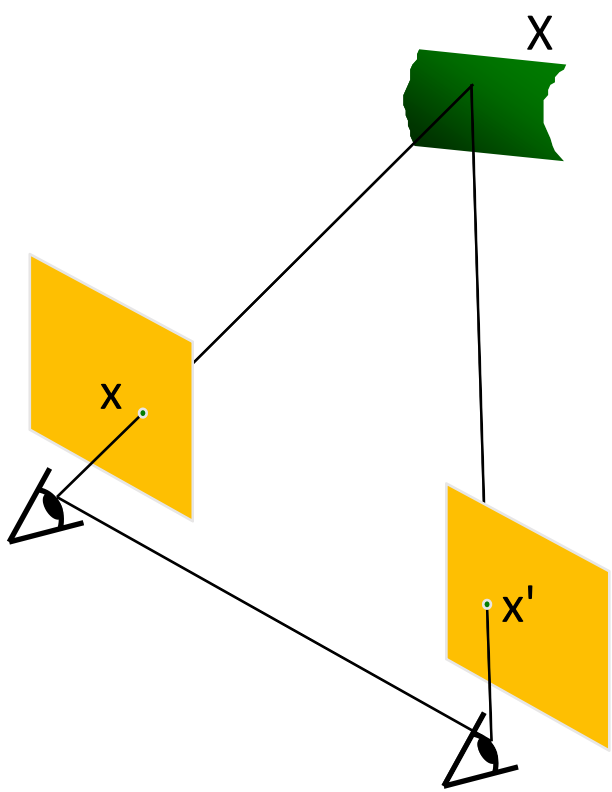
- **Camera models and Projective geometry**
  - What's the mapping between image and world coordiantes?
- **Projection Matrix and Camera calibration**
  - What's the projection matrix between scene and image coordiantes?
  - How to calibrate the projection matrix?
- **Single view metrology and Camera properties**
  - How can we measure the size of 3D objects in an image?
  - What are the important camera properties?
- **Photo stitching**
  - What's the mapping from two images taken without camera translation?
- **Epipolar Geometry and Stereo Vision**
  - What's the mapping from two images taken with camera translation?
- **Structure from motion**
  - How can we recover 3D points from multiple images?

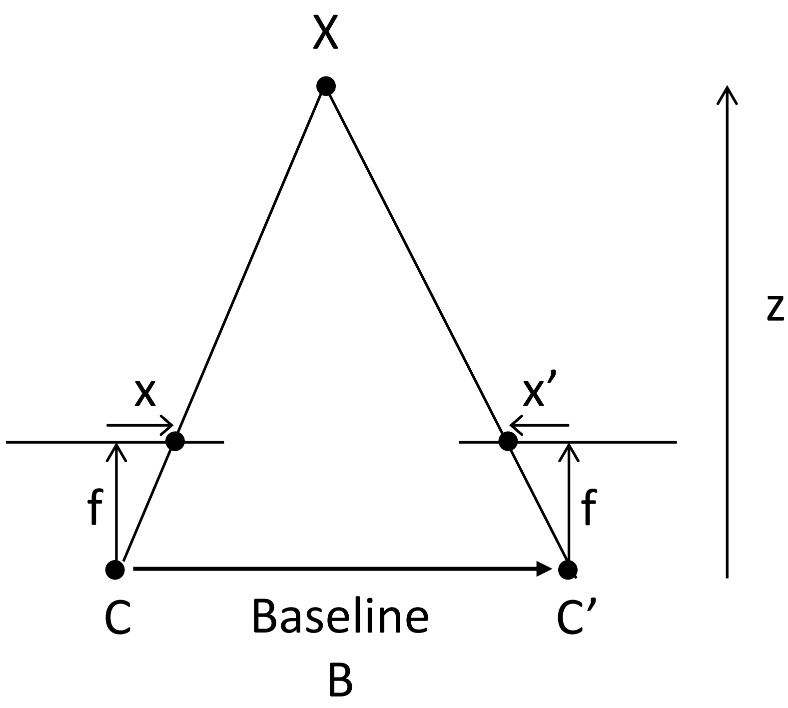
This class: Two-View Geometry

- Epipolar geometry
  - Relates cameras from two positions
- Stereo depth estimation
  - Recover depth from two images

Depth from Stereo

- Goal: Recover depth by finding image coordinate x' that corresponds to x

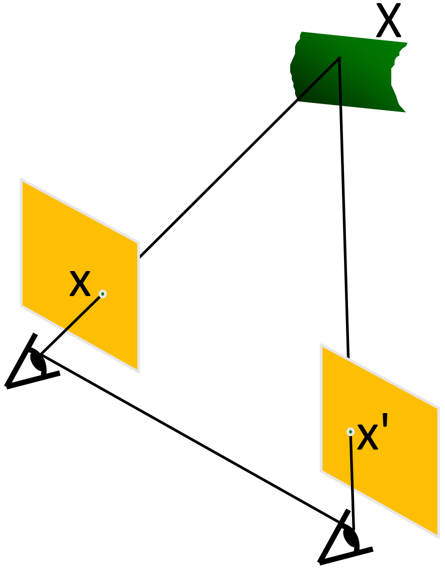




Depth from Stereo

- Goal: Recover depth by finding image coordinate x’ that corresponds to x

- Sub-Problems
  1. **Calibration:**
     What’s the relation of the two cameras?
  2. **Correspondence:**
     Where is the matching point x’?



Correspondence Problem

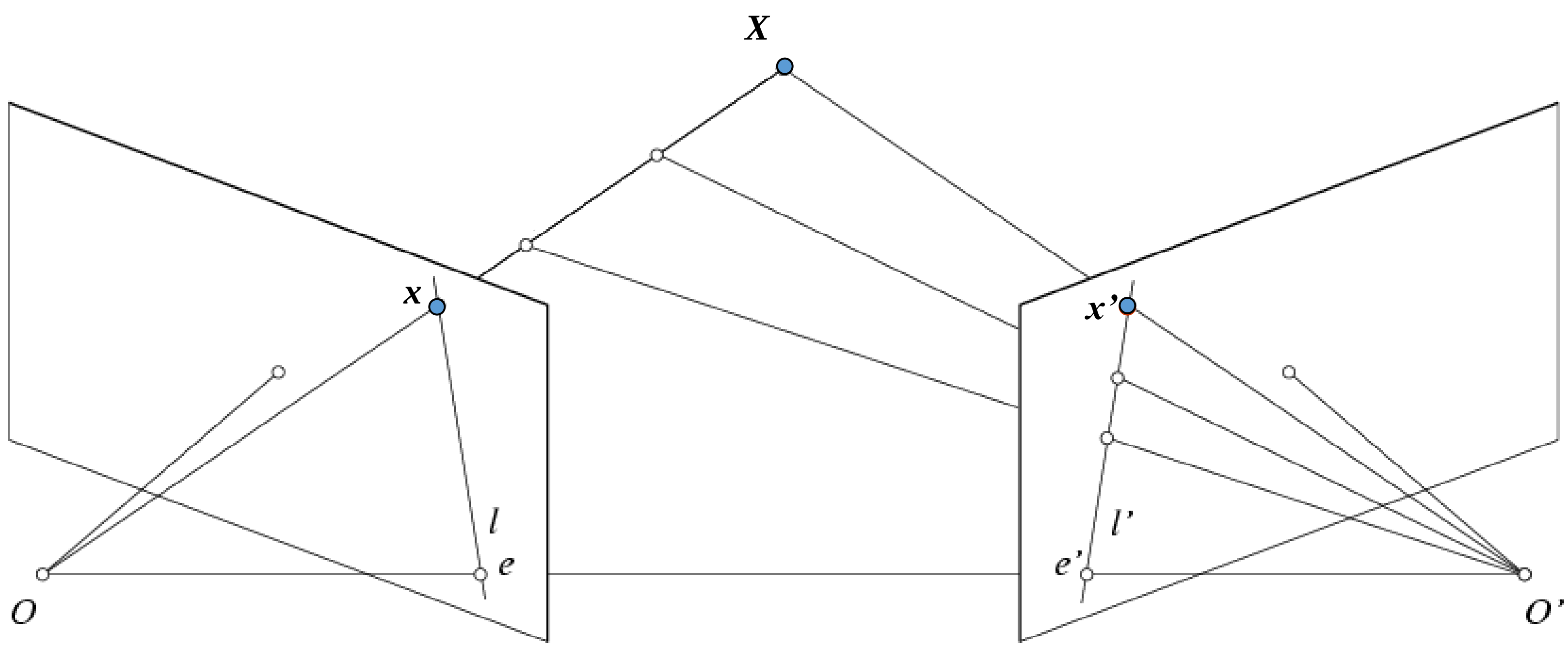
- Two images taken from cameras with different intrinsic and extrinsic parameters

- How do we match a point in the first image to a point in the second?

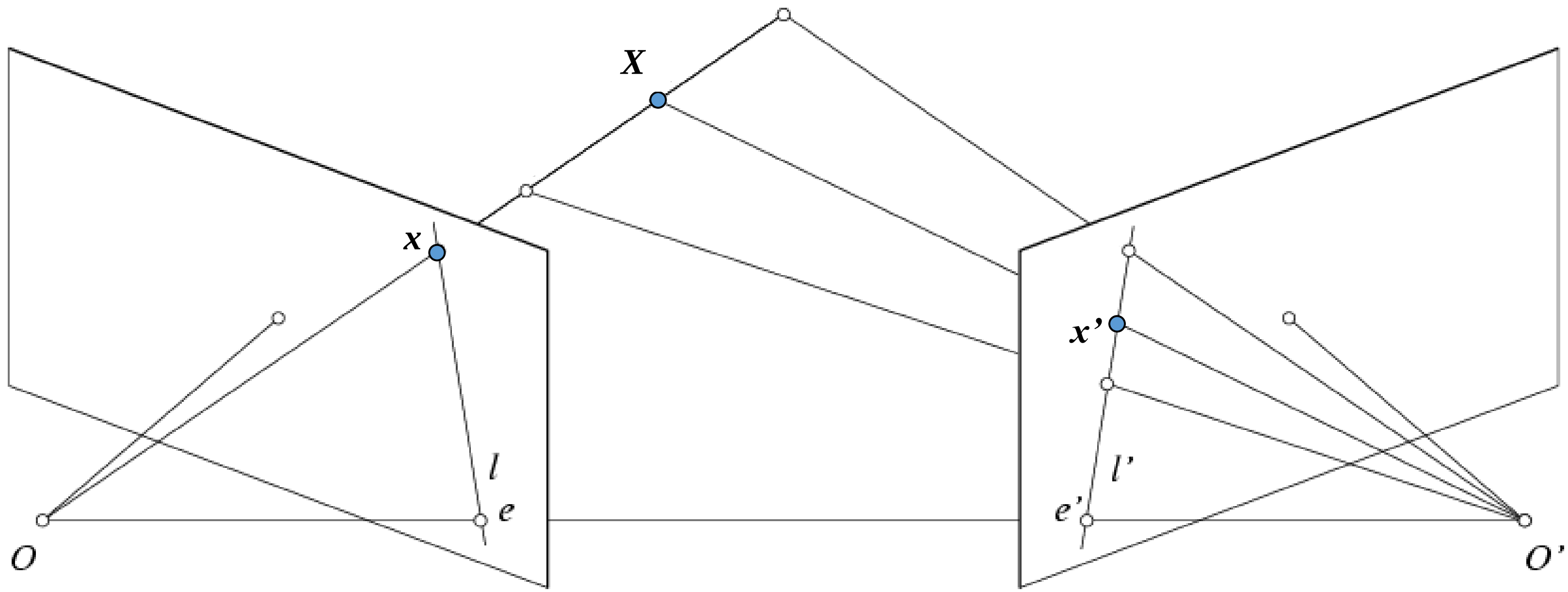
- How can we constrain our search?

Key idea: Epipolar constraint

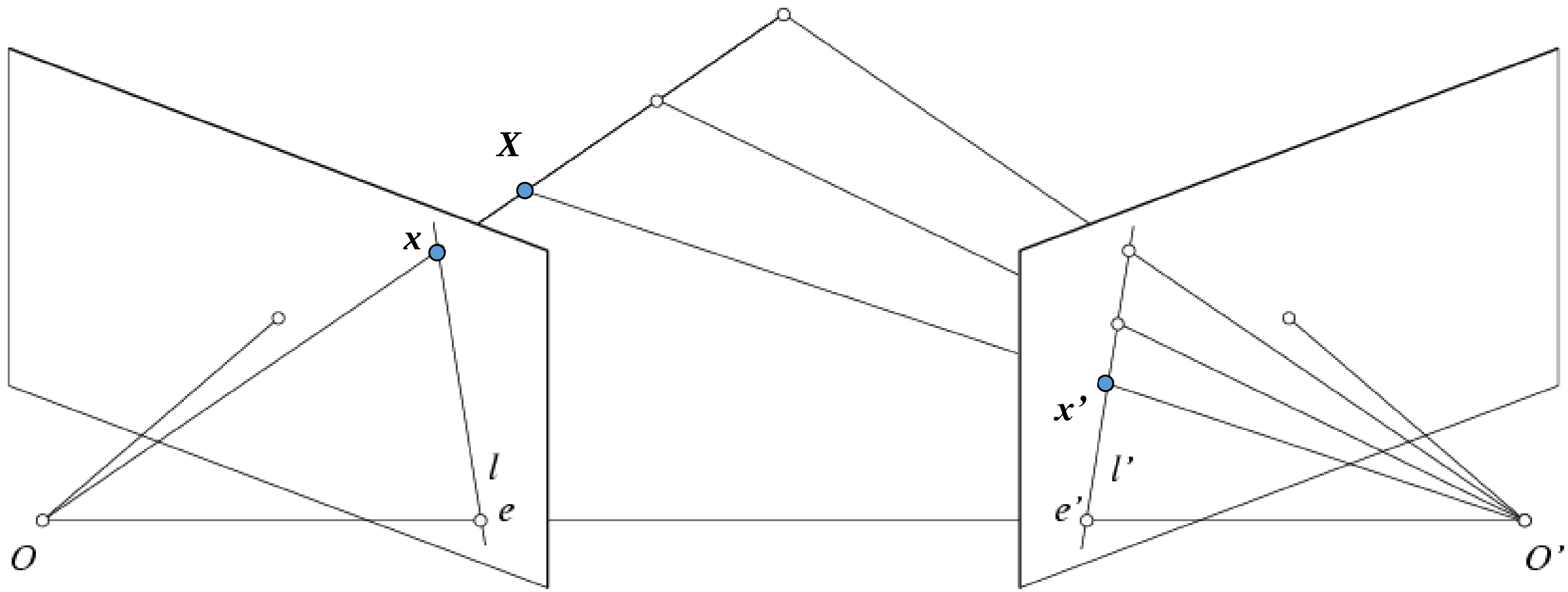
Key idea: Epipolar constraint



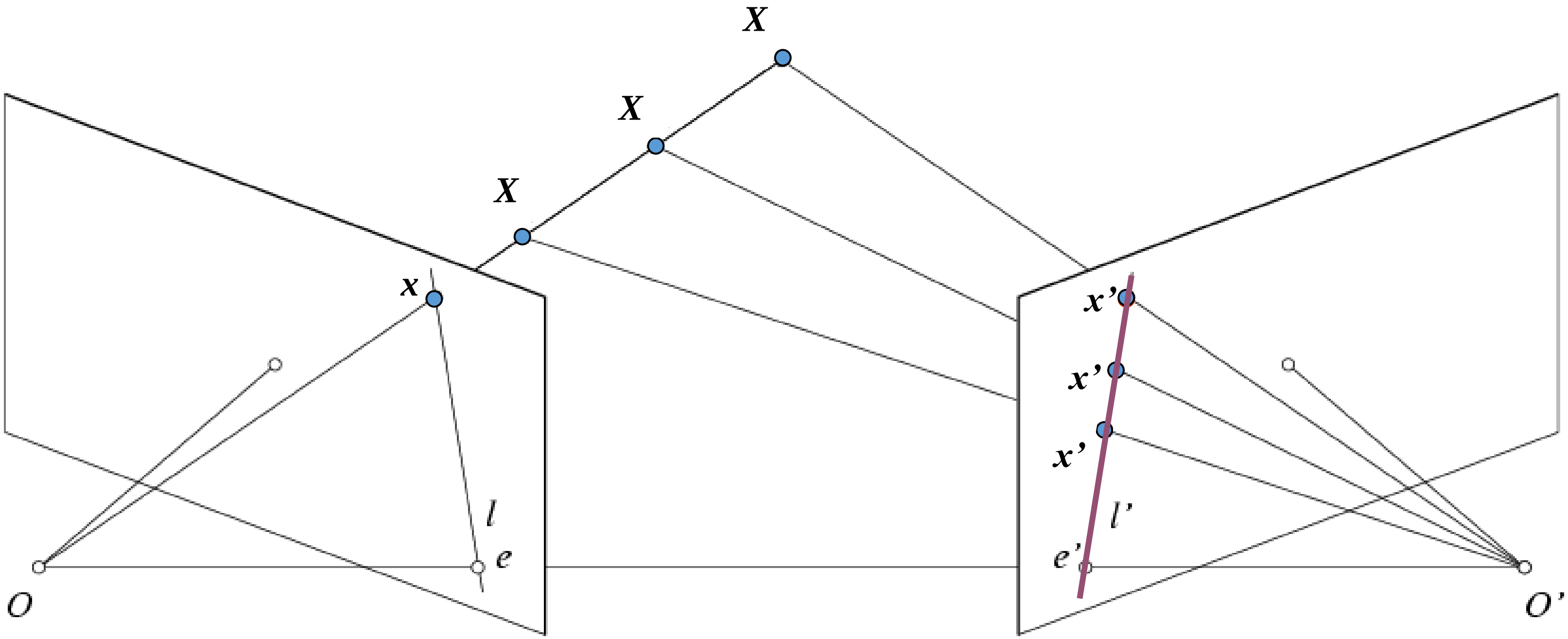
Key idea: Epipolar constraint



Key idea: Epipolar constraint

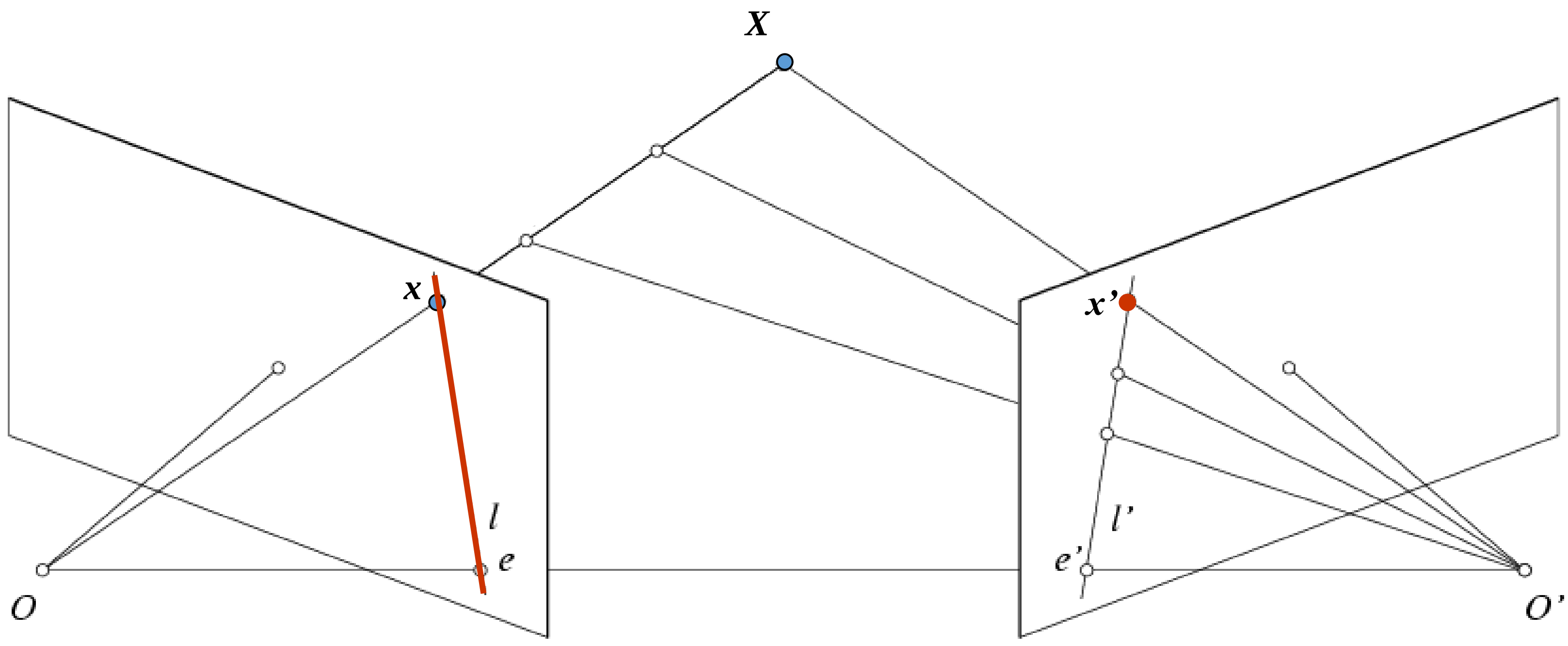


Key idea: Epipolar constraint



Potential matches for *x* have to lie on the corresponding line *l'*.

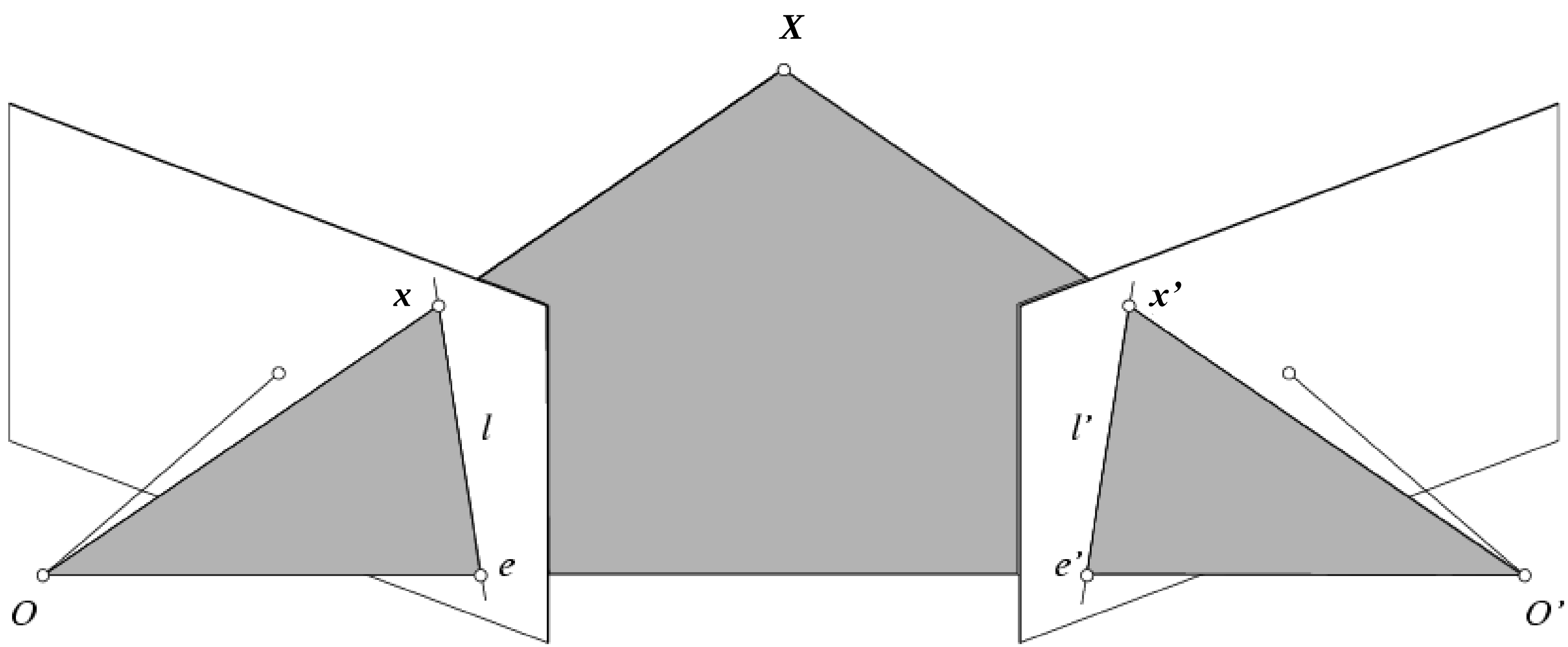
Key idea: Epipolar constraint



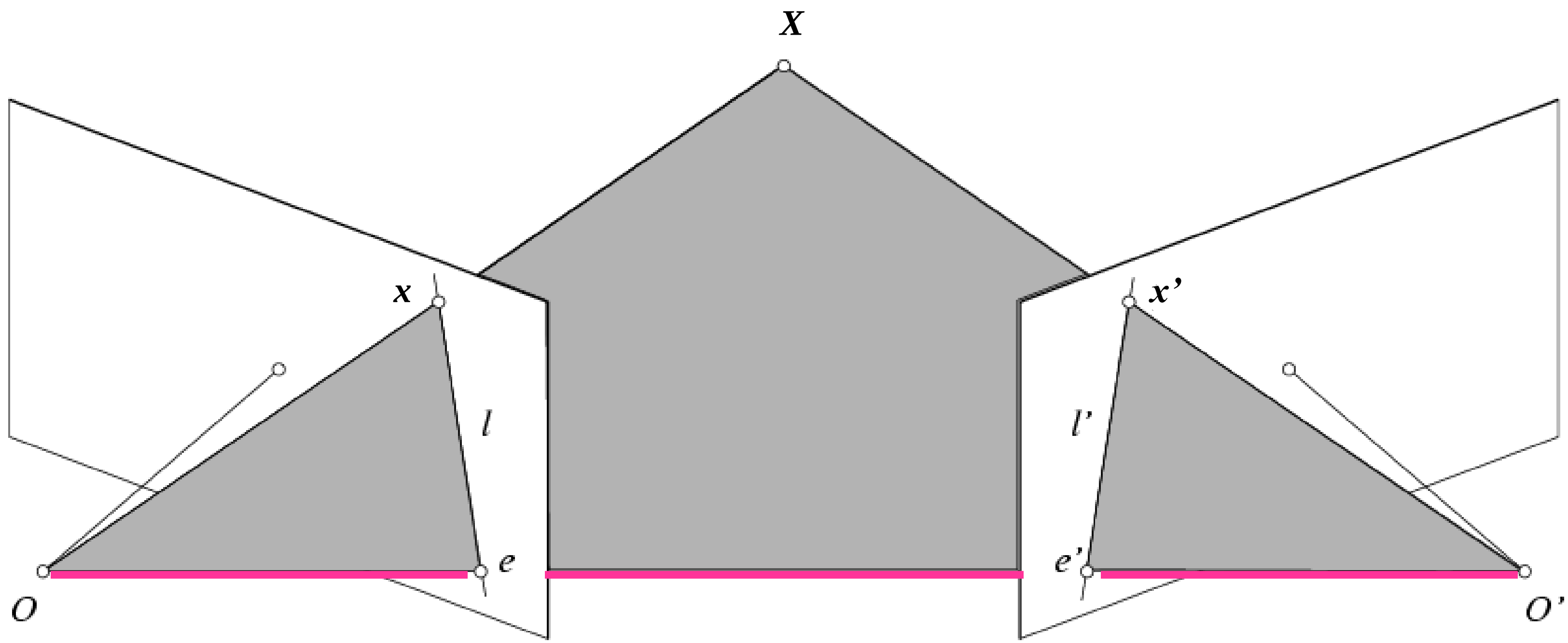
Potential matches for *x* have to lie on the corresponding line *l'*.

Potential matches for *x'* have to lie on the corresponding line *l*.

Epipolar geometry: notation

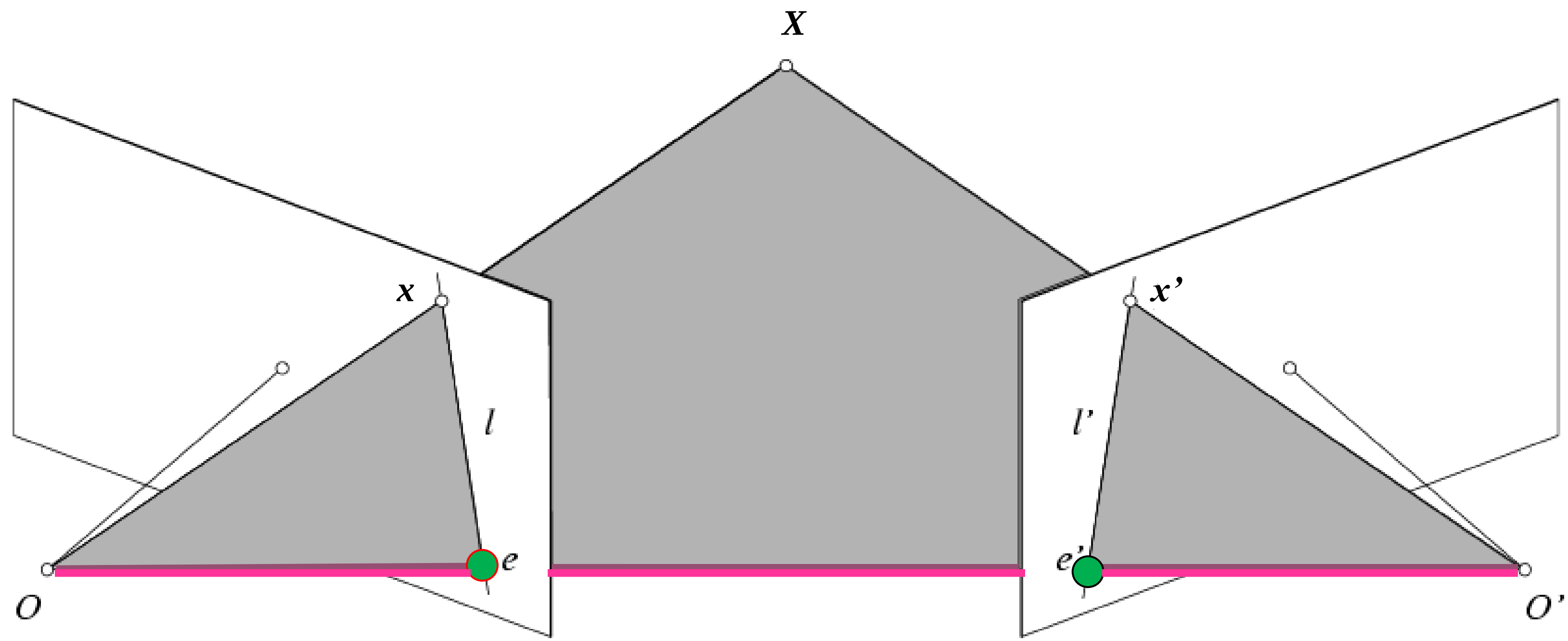


Epipolar geometry: notation



- **Baseline** – line connecting the two camera centers

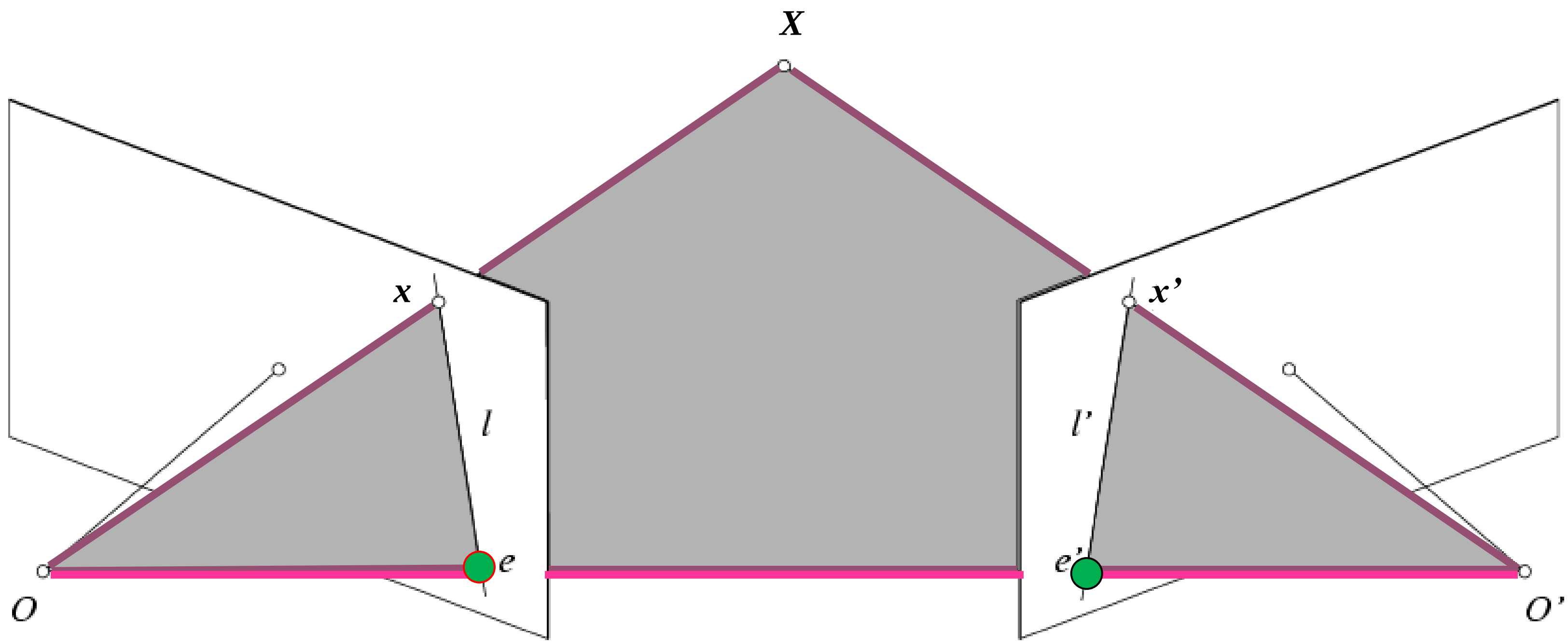
Epipolar geometry: notation



- **Baseline** – line connecting the two camera centers
- **Epipoles**

= intersections of baseline with image planes
= projections of the other camera center

Epipolar geometry: notation



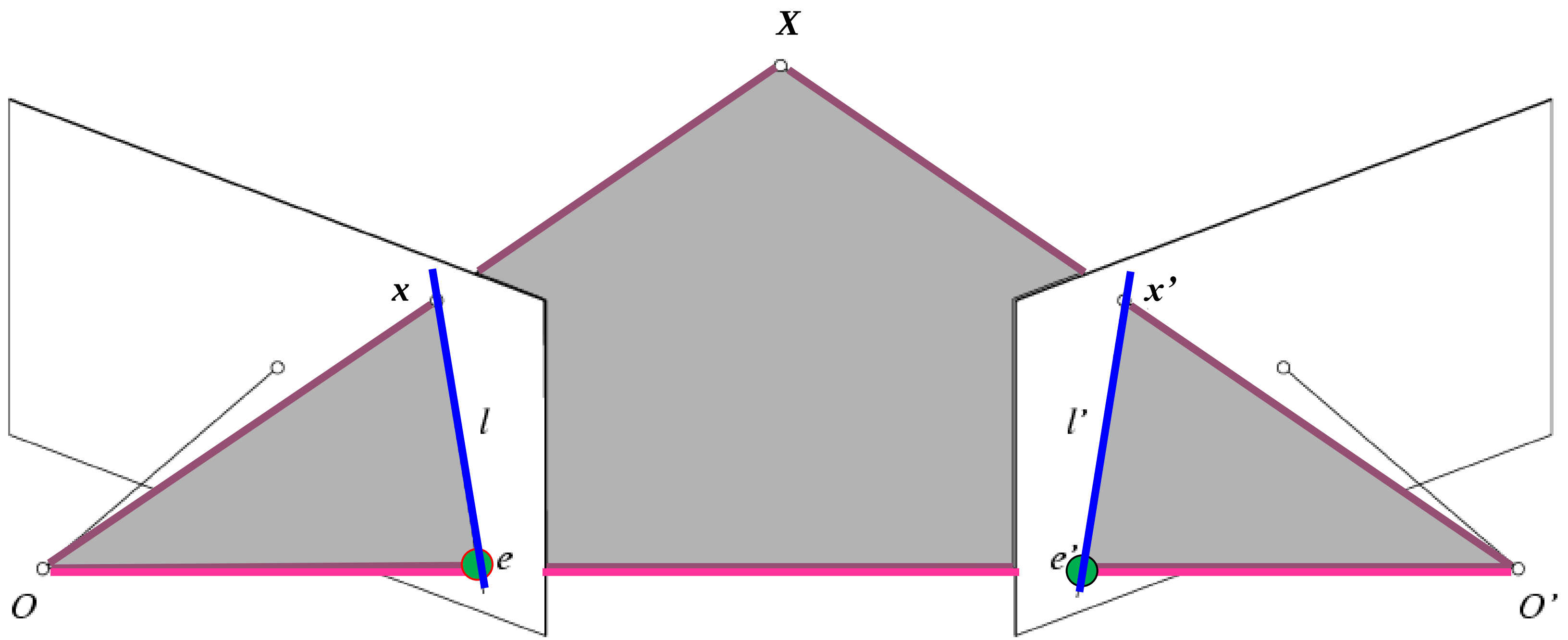
- **Baseline** – line connecting the two camera centers
- **Epipoles**

= intersections of baseline with image planes

= projections of the other camera center

- **Epipolar Plane** – plane containing baseline and 3d point

Epipolar geometry: notation



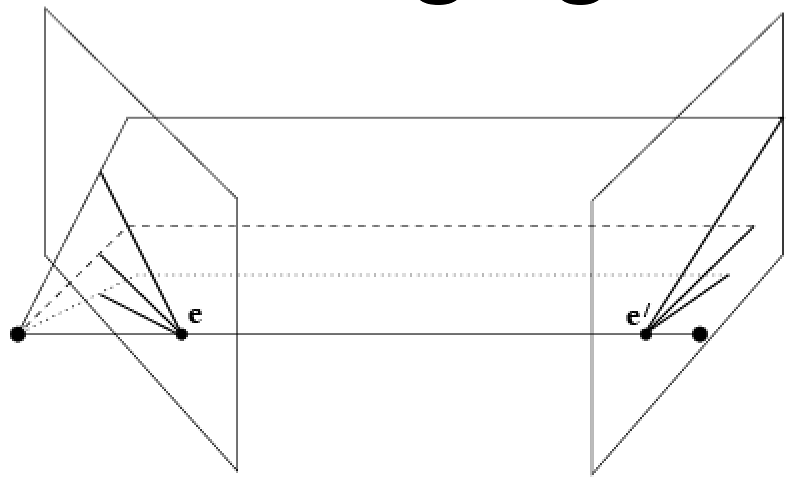
- **Baseline** – line connecting the two camera centers
- **Epipoles**

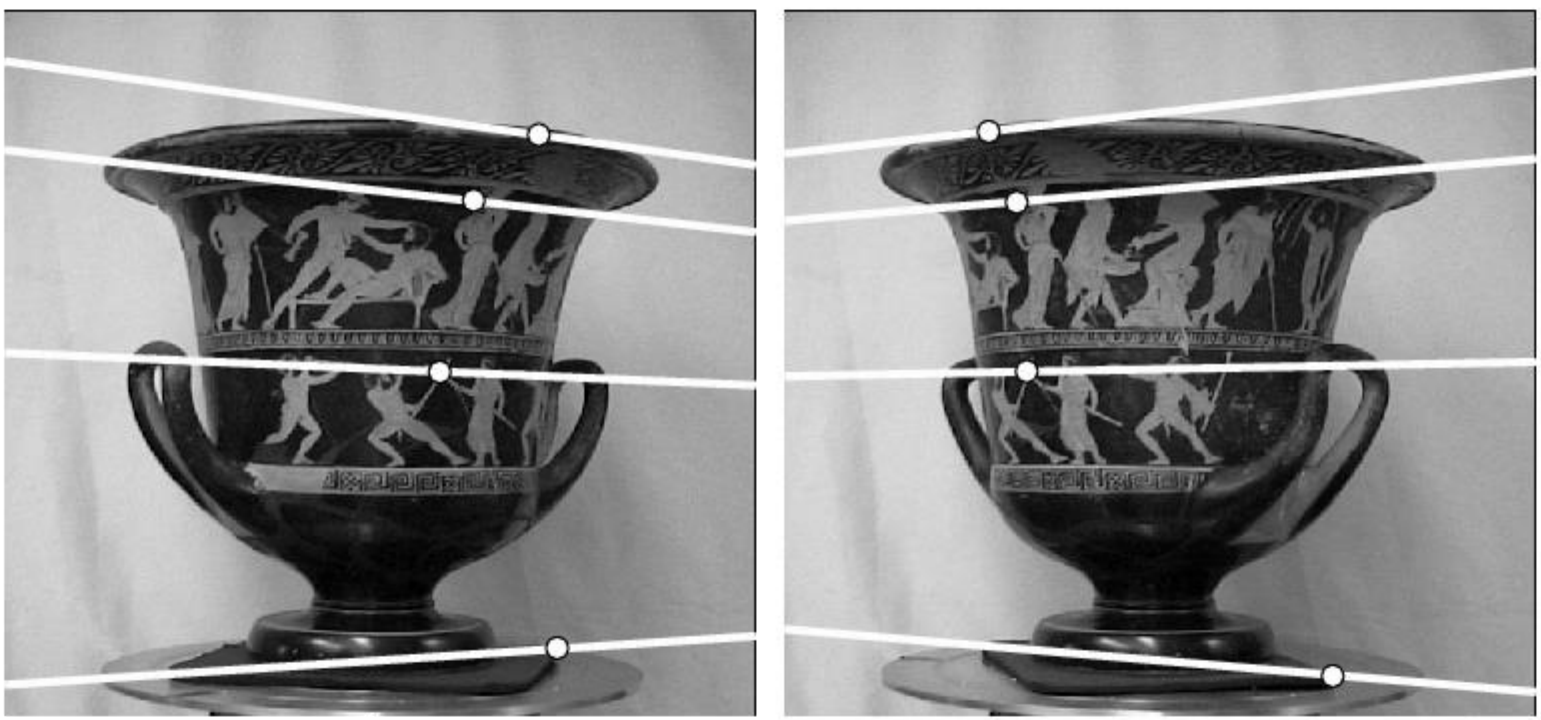
= intersections of baseline with image planes

= projections of the other camera center

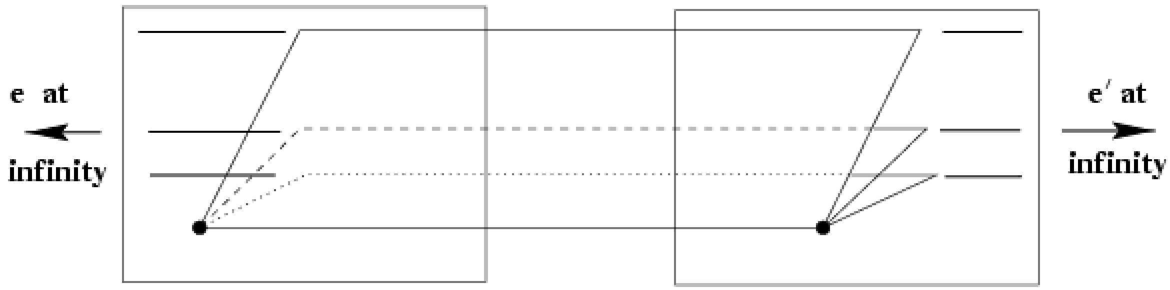
- **Epipolar Plane** – plane containing baseline and 3d point
- **Epipolar Lines** - intersections of epipolar plane with image planes (always come in corresponding pairs)

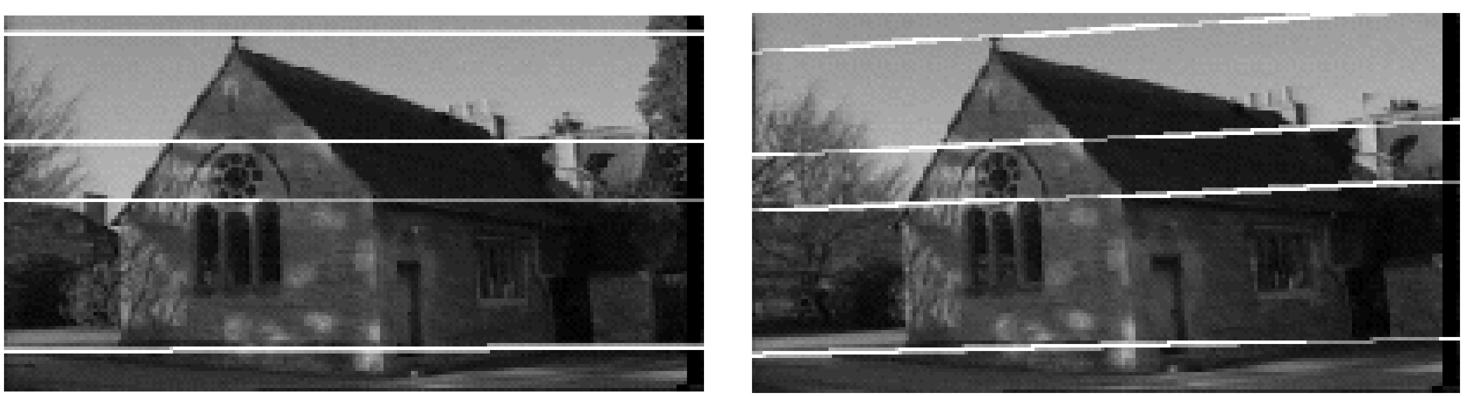
Example: Converging cameras





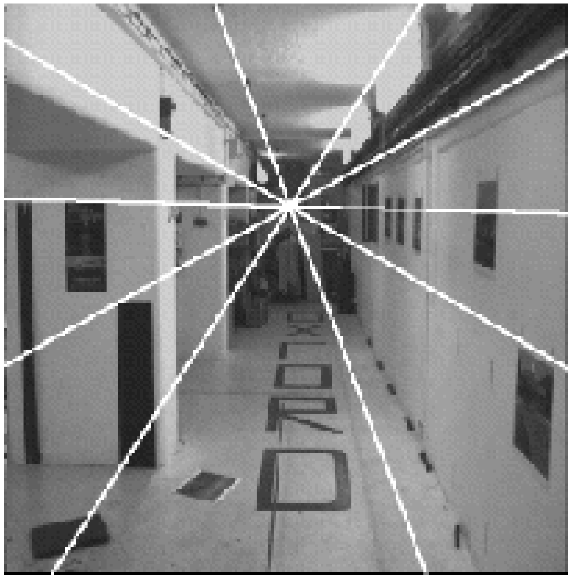
Example: Parallel cameras

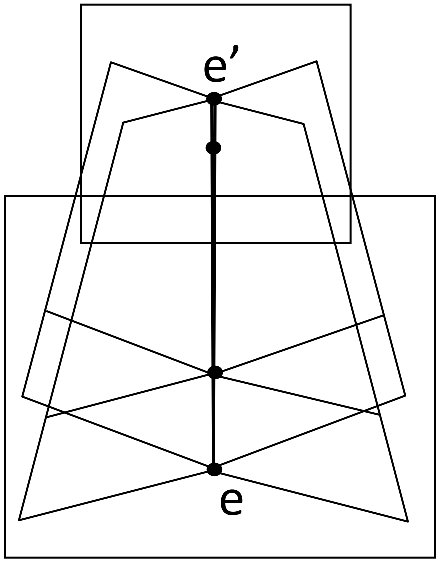




Example: Forward motion

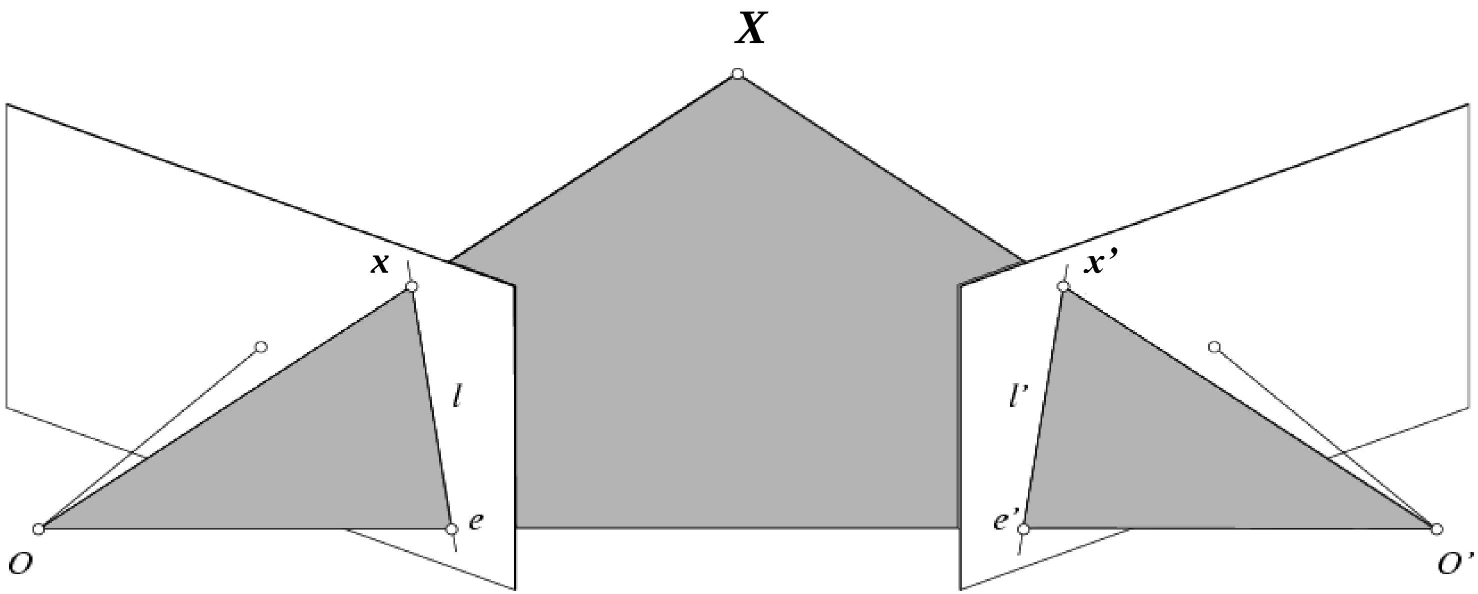
The camera moves directly forward





Epipole has same coordinates in both images.
Points move along lines radiating from e:
“Focus of expansion”

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)

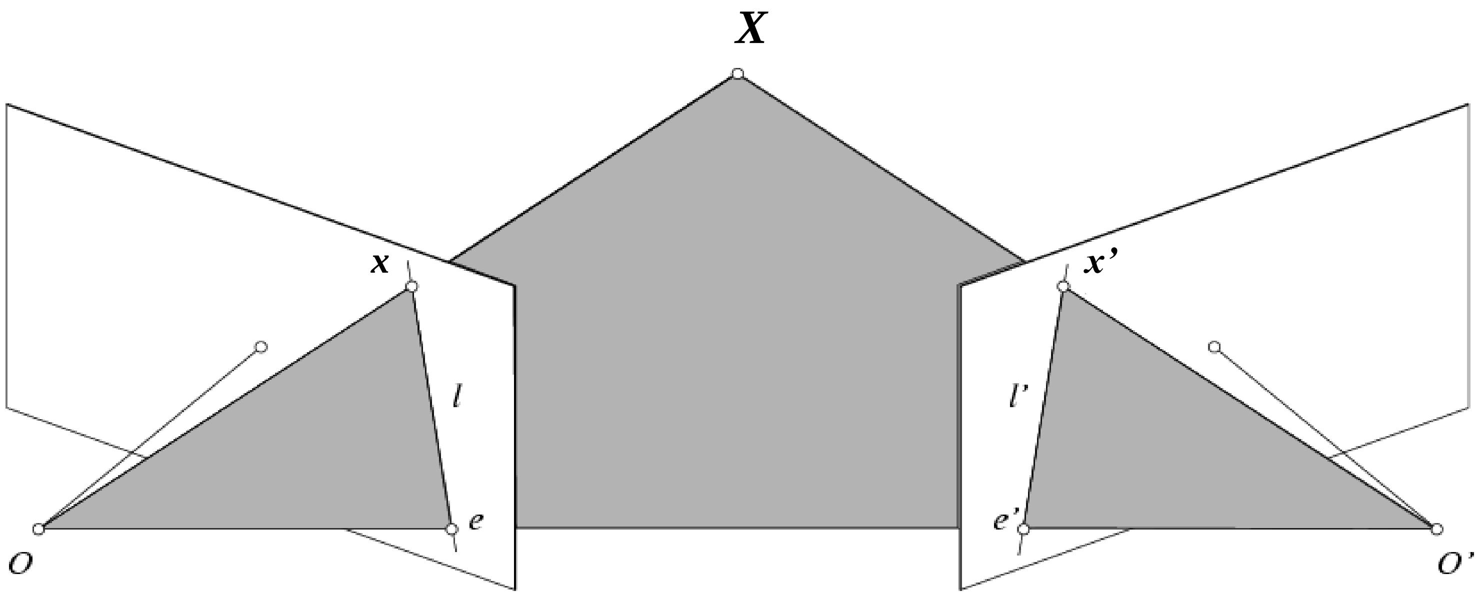
2D pixel coordinate (homogeneous)

3D scene point

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

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

Epipolar constraint: Calibrated case



Given the intrinsic parameters of the cameras:

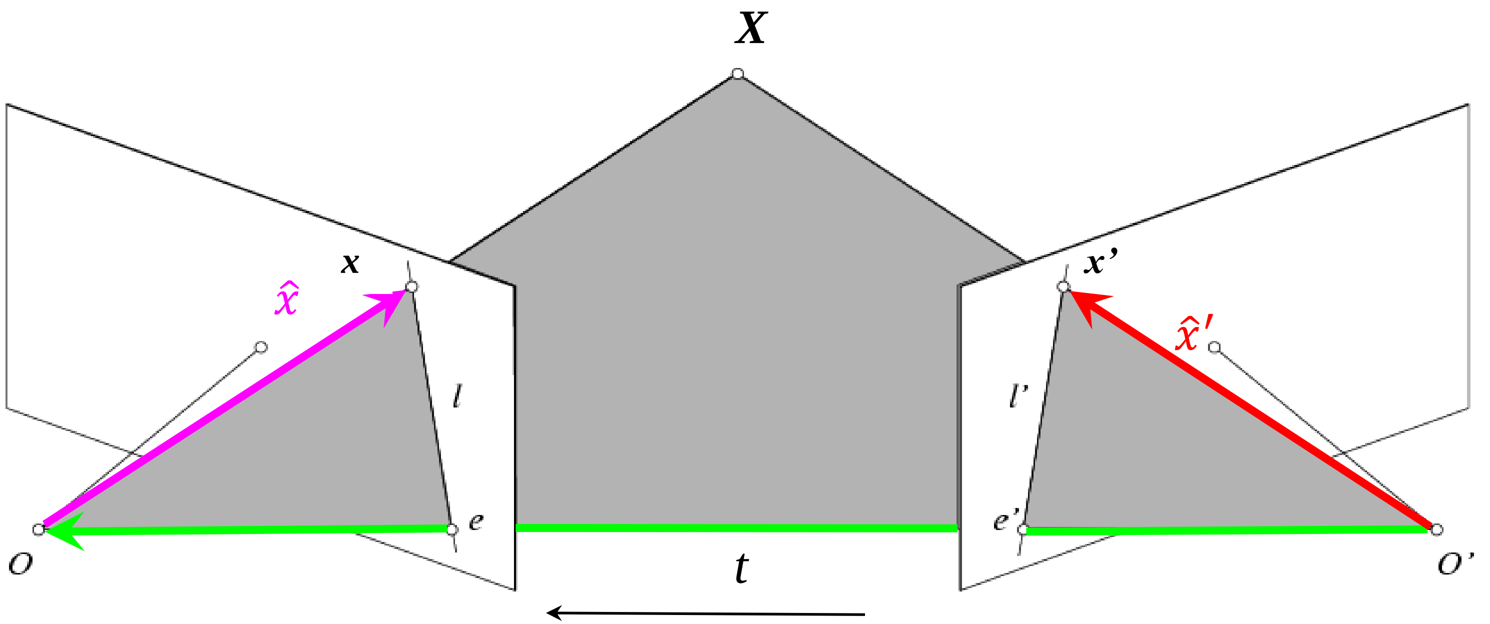
1. Convert to normalized coordinates by pre-multiplying all points with the inverse of the calibration matrix; set first camera's coordinate system to world coordinates
2. Define some *R* and *t* that relate X to X' as below

for some scale factor

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

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

Epipolar constraint: Calibrated case

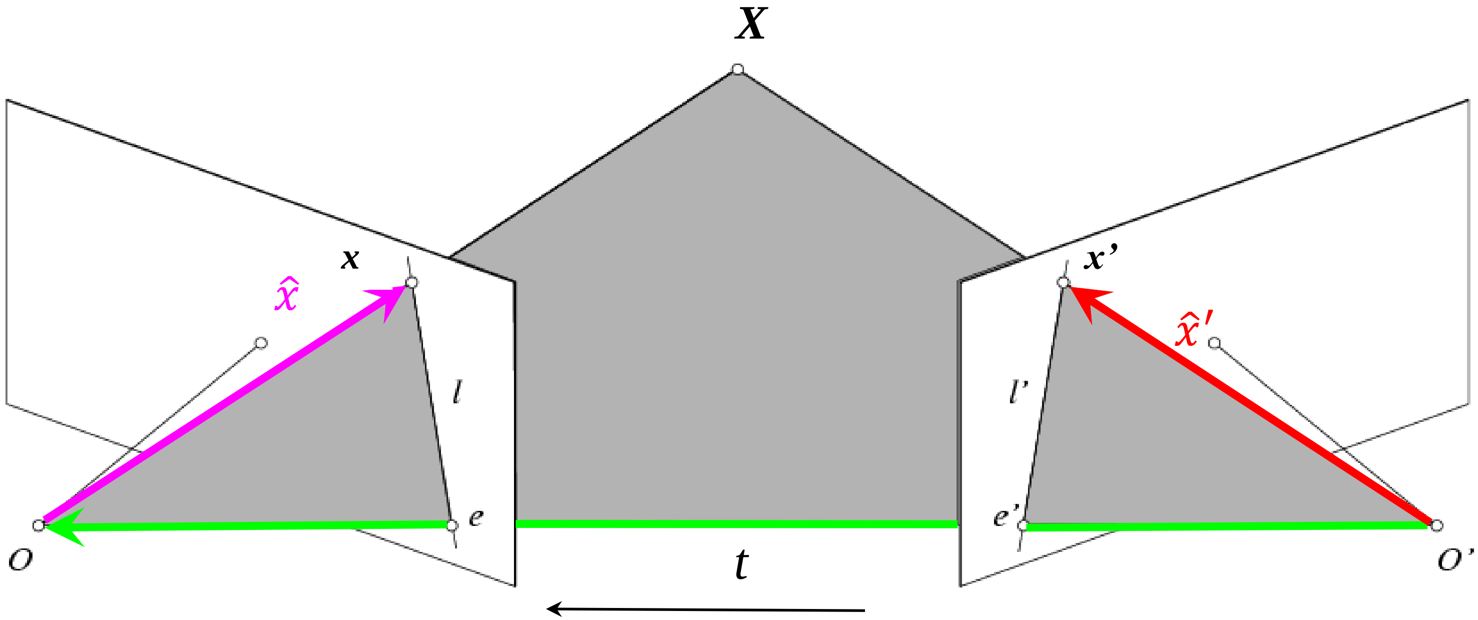


$$\hat{x} = K^{-1}x = X$$

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

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

Epipolar constraint: Calibrated case

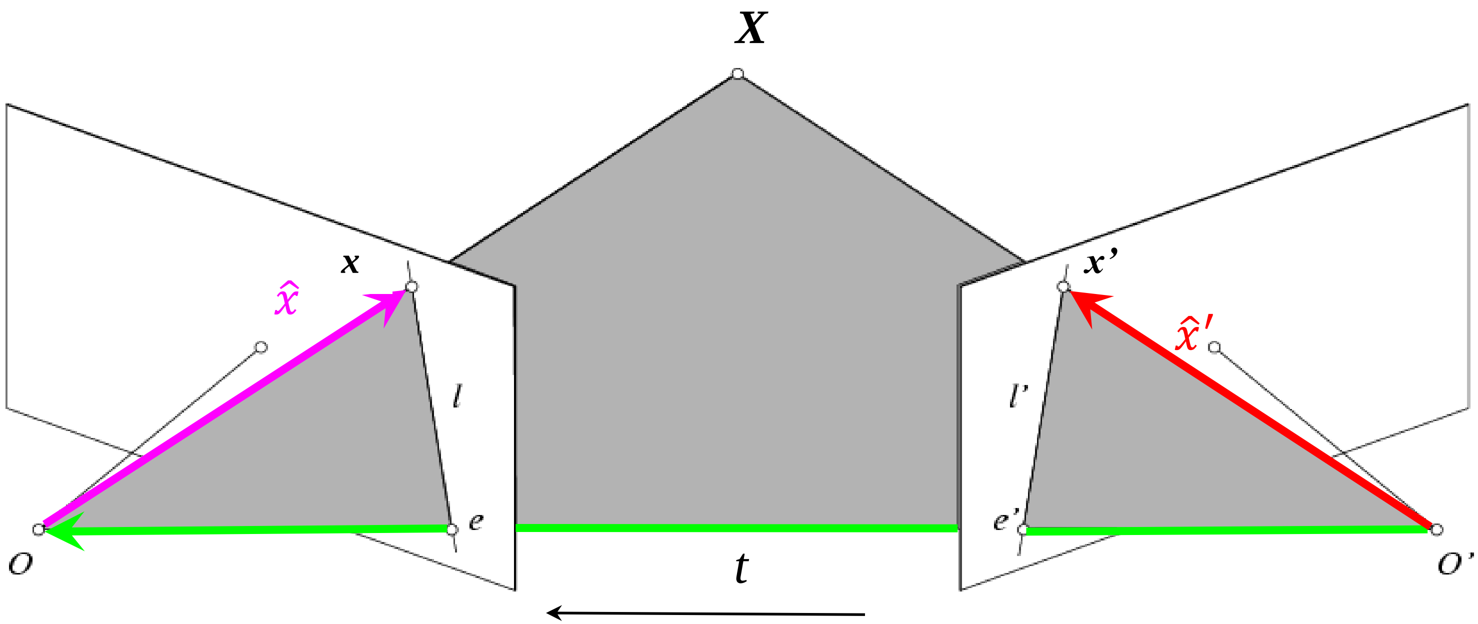


$$\hat{x} = K^{-1}x = X \qquad \hat{x}' = K'^{-1}x' = X'$$

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

$$\hat{x}.[t \times \hat{x}] = 0$$

Epipolar constraint: Calibrated case



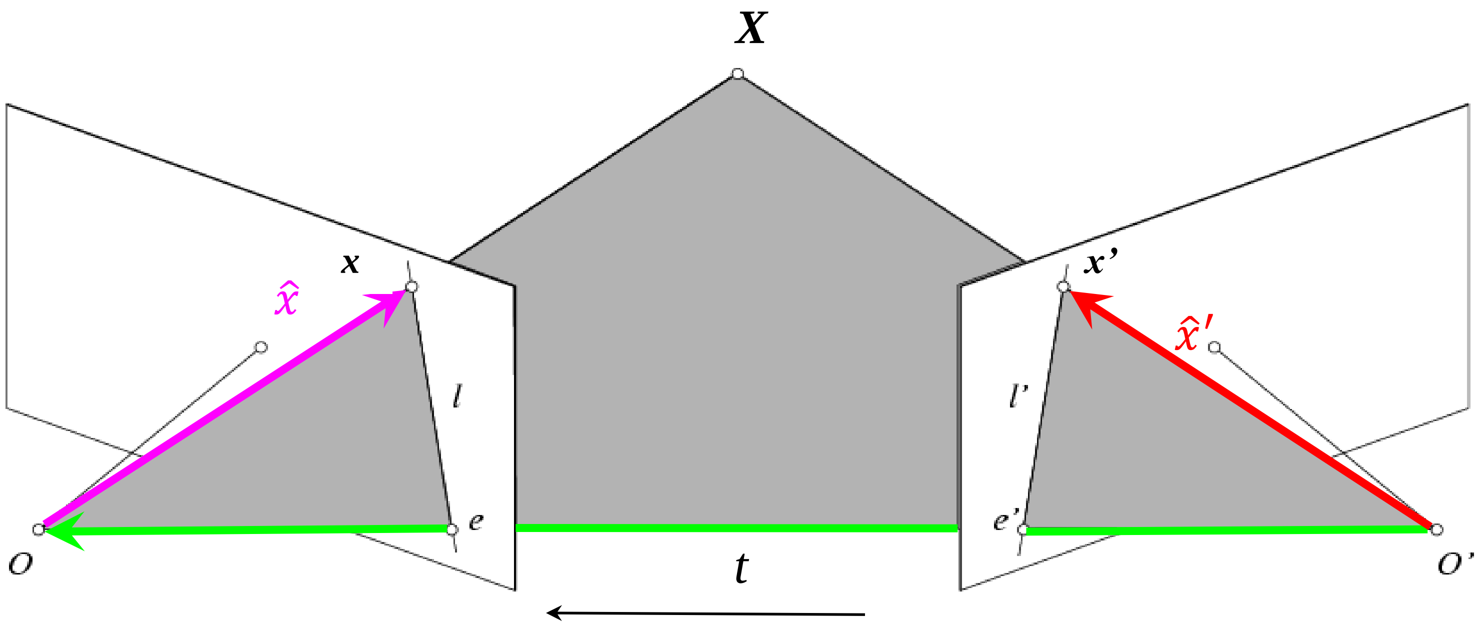
$$\hat{x} = K^{-1}x = X$$

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

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

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

Epipolar constraint: Calibrated case



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

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

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

$$\Rightarrow \hat{x}.[t \times (R\,\hat{x}')] = 0$$

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

Matrix representation of cross product

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

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

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

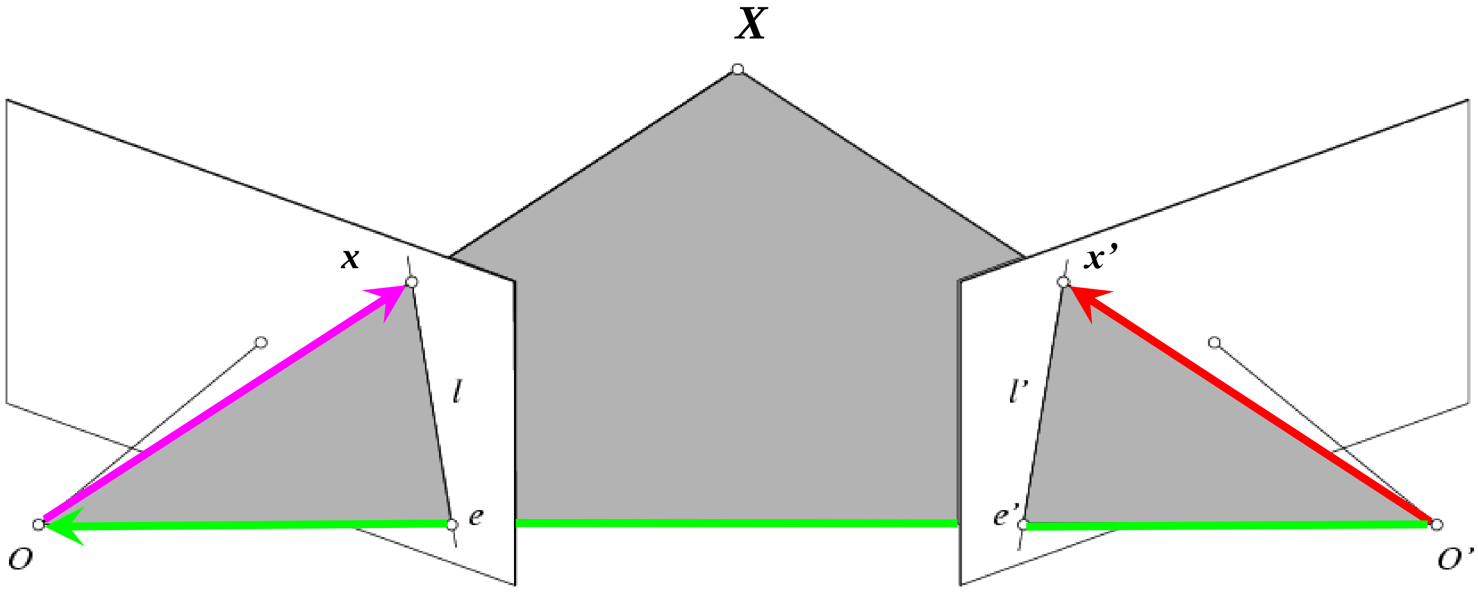
Skew-symmetric matrix

(Rank 2)

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b}$$

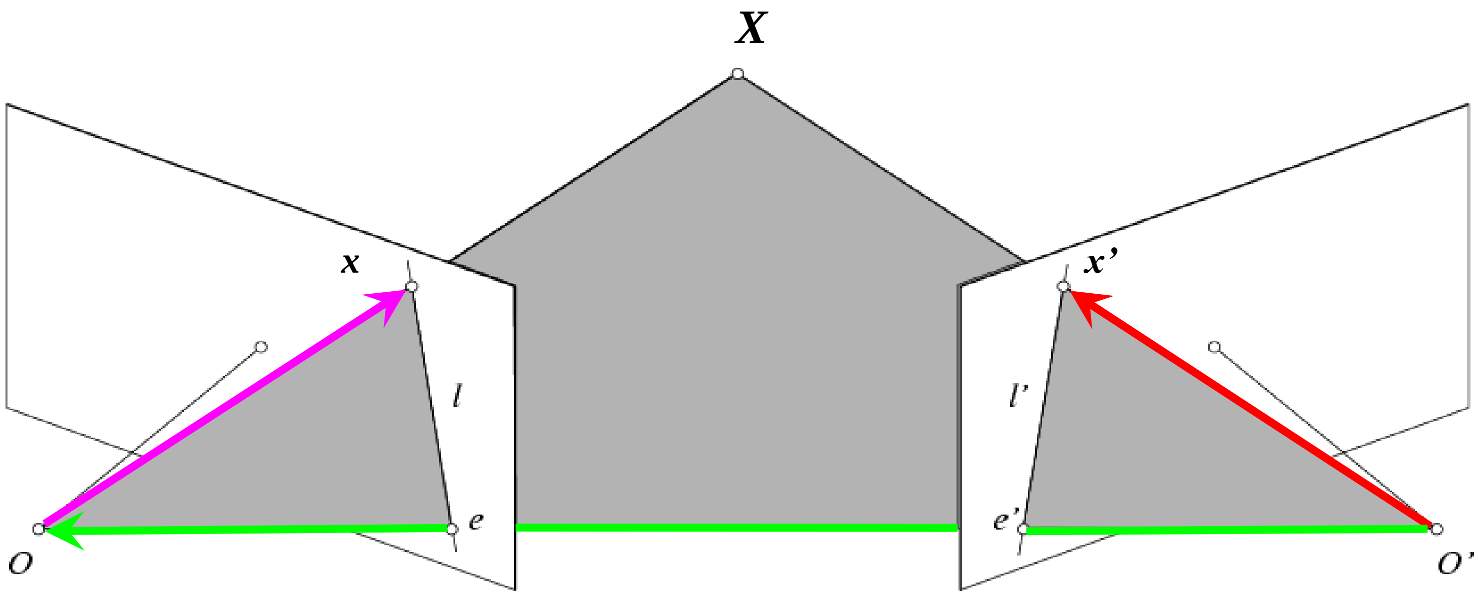
[Matrix representation of the cross product]

Essential matrix



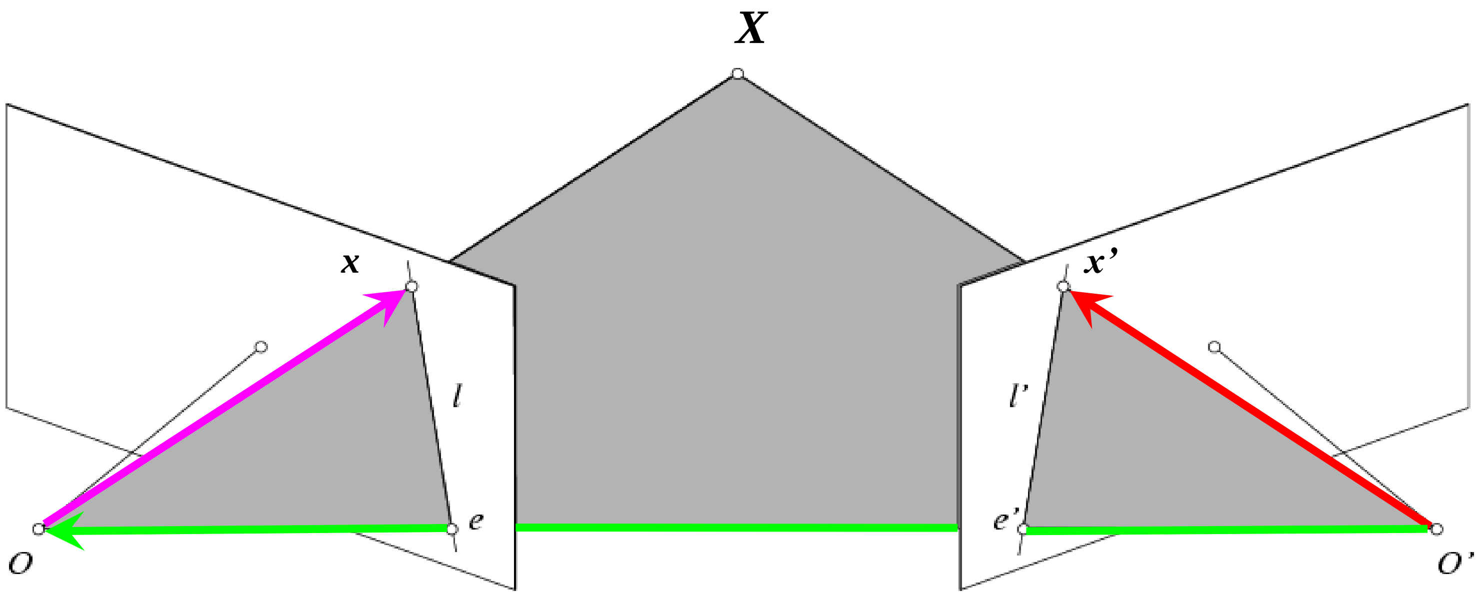
$$\hat{x}.[t \times (R\hat{x}')] = \hat{x}^T [t]_\times R\hat{x}'$$

Essential matrix



$$\hat{x}.[t \times (R\hat{x}')] = \hat{x}^T \left[t\right]_\times R\hat{x}' = \hat{x}^T E\,\hat{x}' = 0 \quad \text{with} \quad E = \left[t\right]_\times R$$

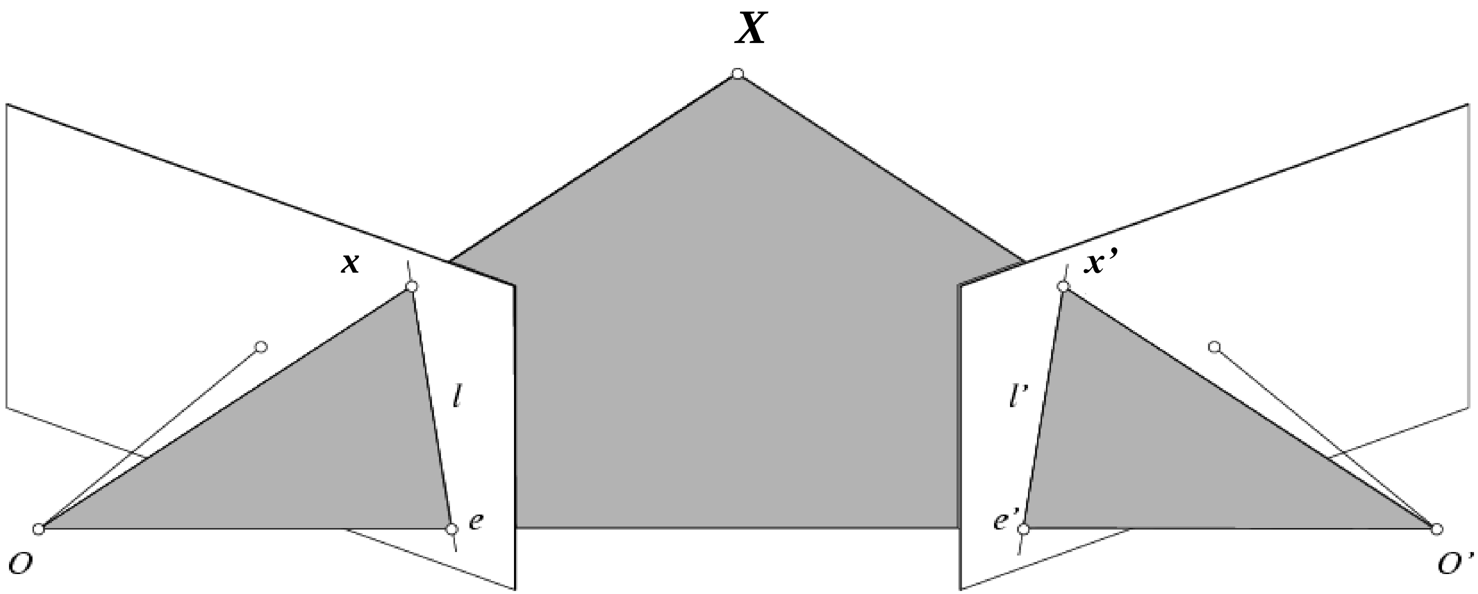
Essential matrix



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

Essential Matrix
(Longuet-Higgins, 1981)

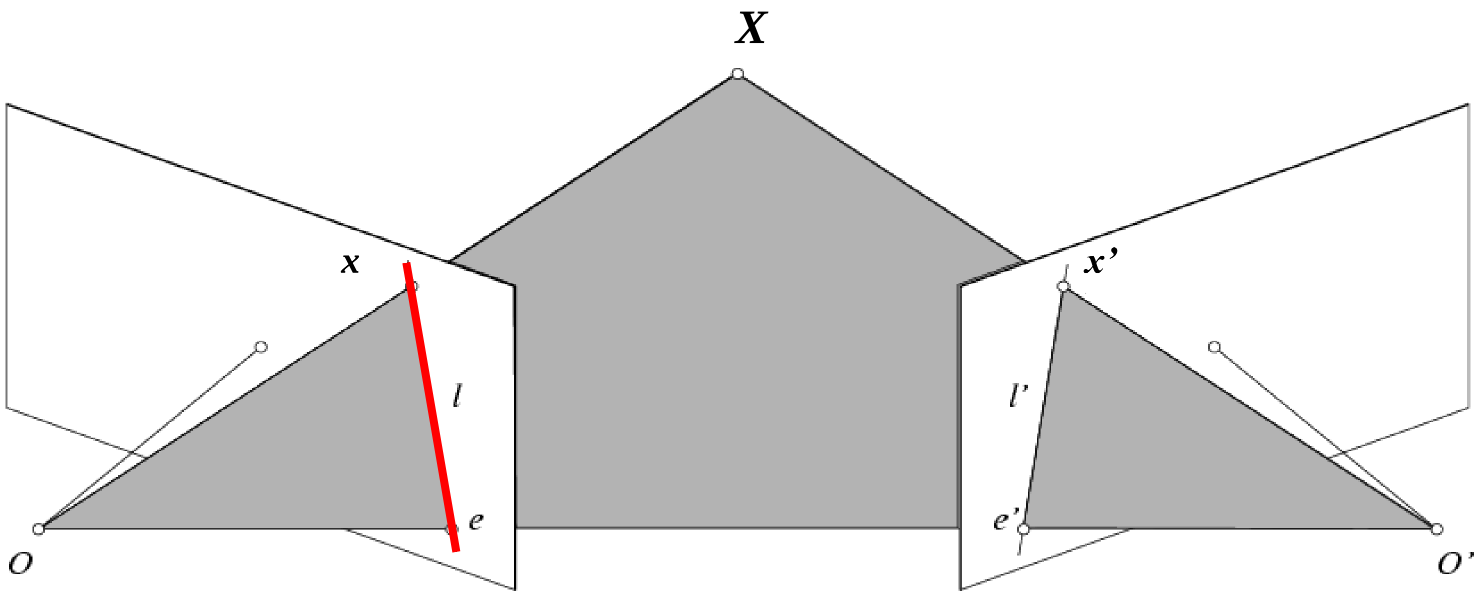
Properties of the Essential matrix



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

Skew-symmetric matrix

Properties of the Essential matrix

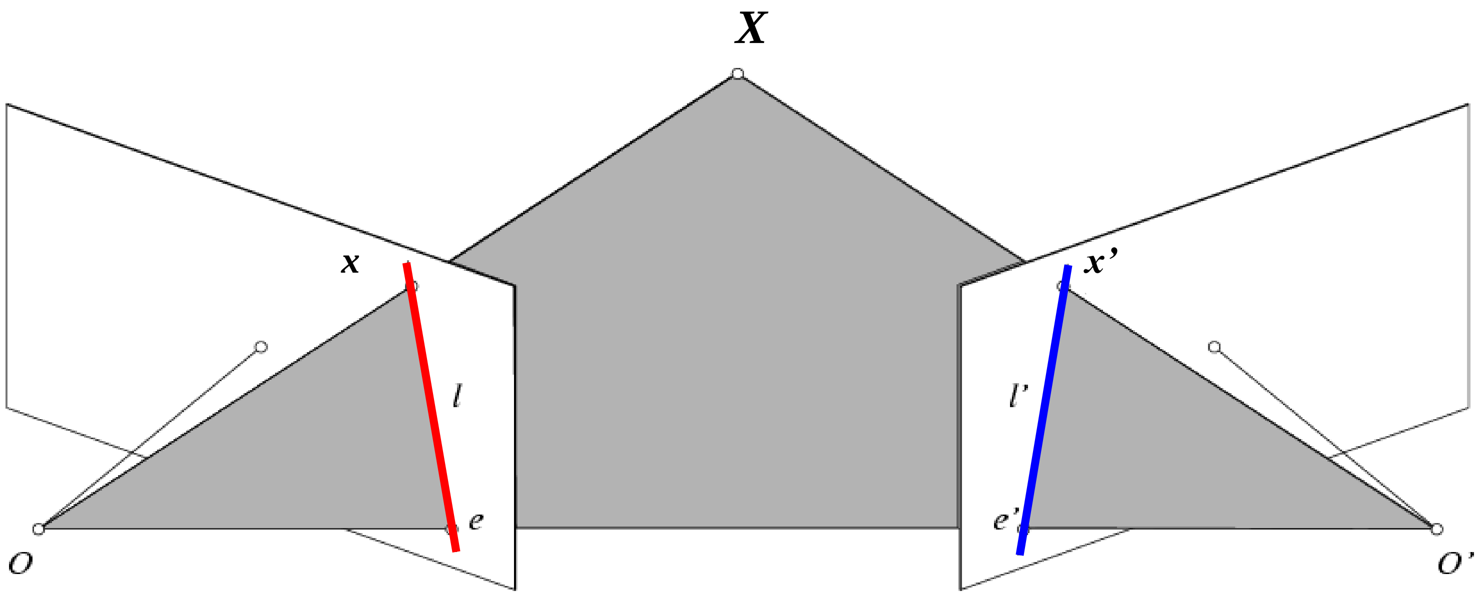


$$\hat{x} \cdot [t \times (R\hat{x}')] = 0 \quad \Rightarrow \quad \hat{x}^T E \hat{x}' = 0 \quad \text{with} \quad E = [t]_\times R$$

Skew-symmetric matrix

- *E x'* is the epipolar line associated with *x'* (*l* = *E x'*)

Properties of the Essential matrix

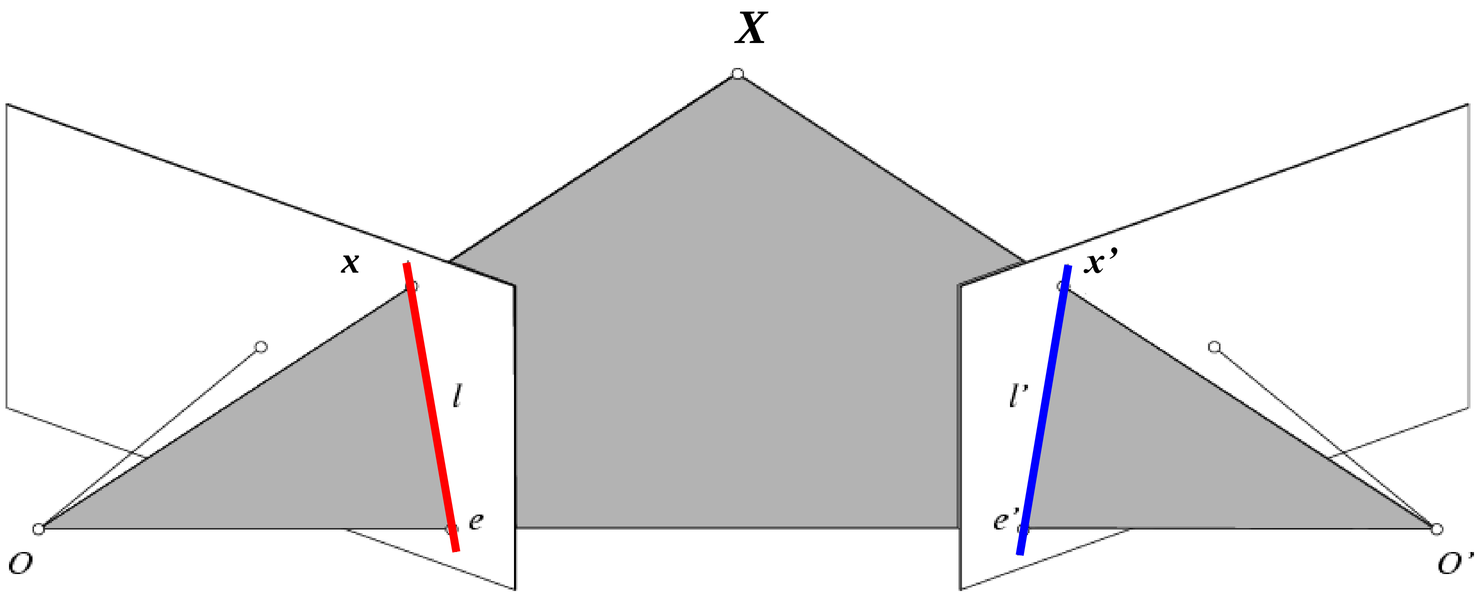


$$\hat{x} \cdot [t \times (R\hat{x}')] = 0 \quad \Rightarrow \quad \hat{x}^T E \hat{x}' = 0 \quad \text{with} \quad E = [t]_\times R$$

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* ($l' = E^T x$)

Properties of the Essential matrix

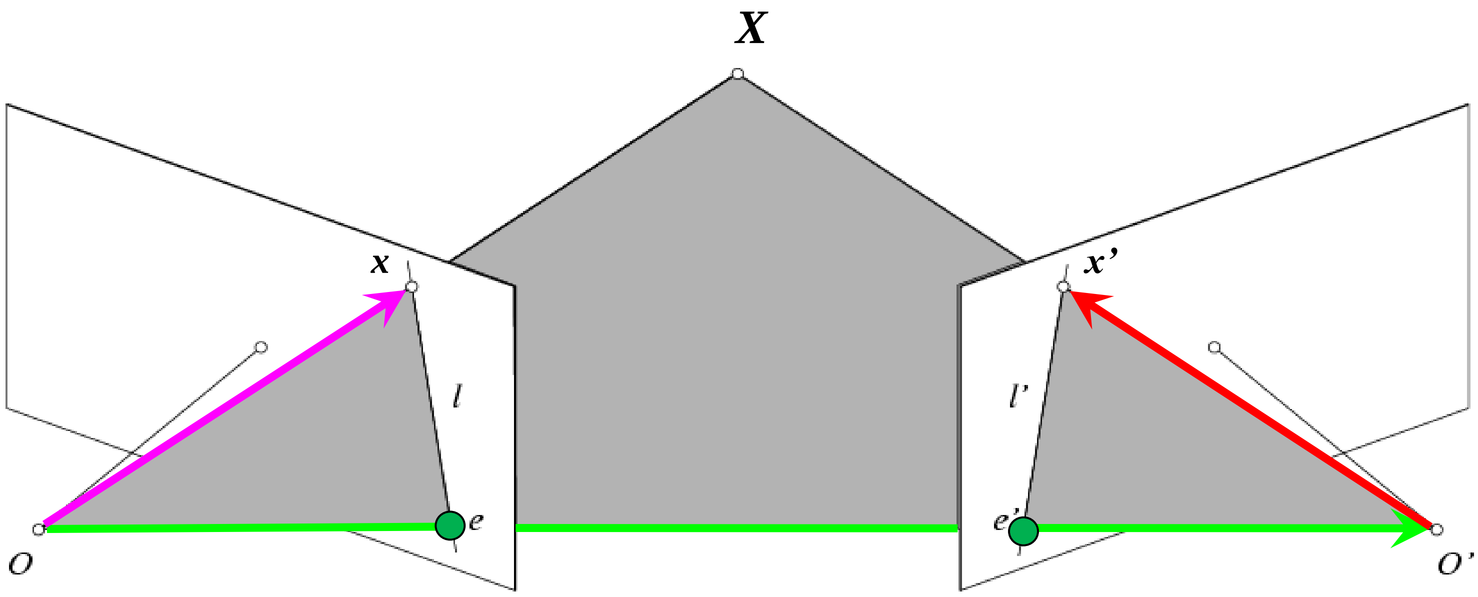


$$\hat{x} \cdot [t \times (R\hat{x}')] = 0 \quad \Rightarrow \quad \hat{x}^T E \hat{x}' = 0 \quad \text{with} \quad E = [t]_\times R$$

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* is singular (rank two): det(E)=0

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:

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

The Fundamental Matrix

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

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

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

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

The Fundamental Matrix

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

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

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

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

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

The Fundamental Matrix

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

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

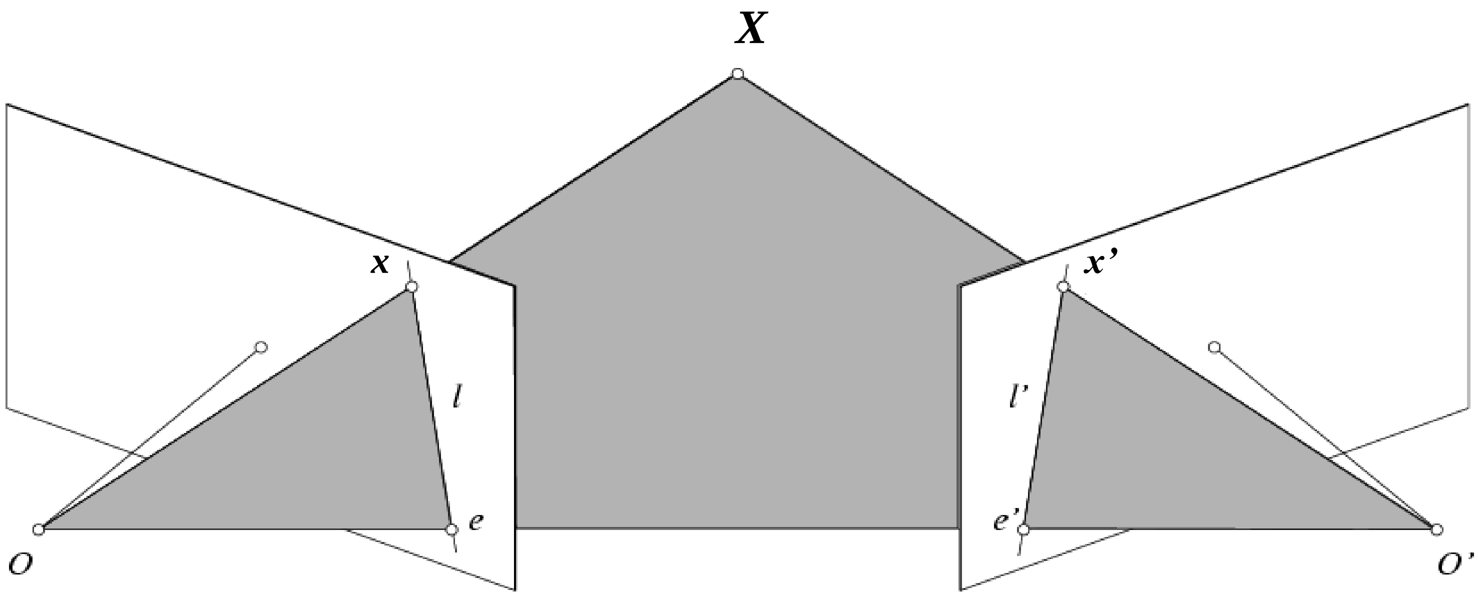
$$\hat{x} = K^{-1} x$$

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

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

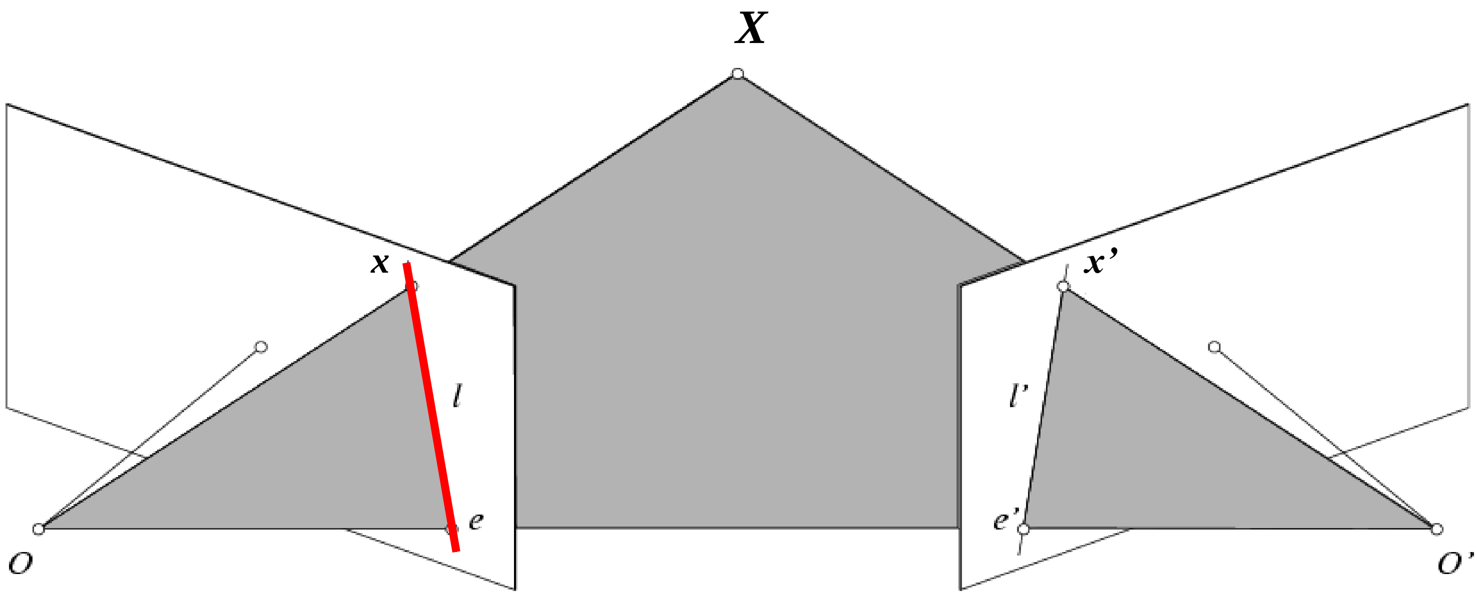
Fundamental Matrix
(Faugeras and Luong, 1992)

Properties of the Fundamental matrix



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

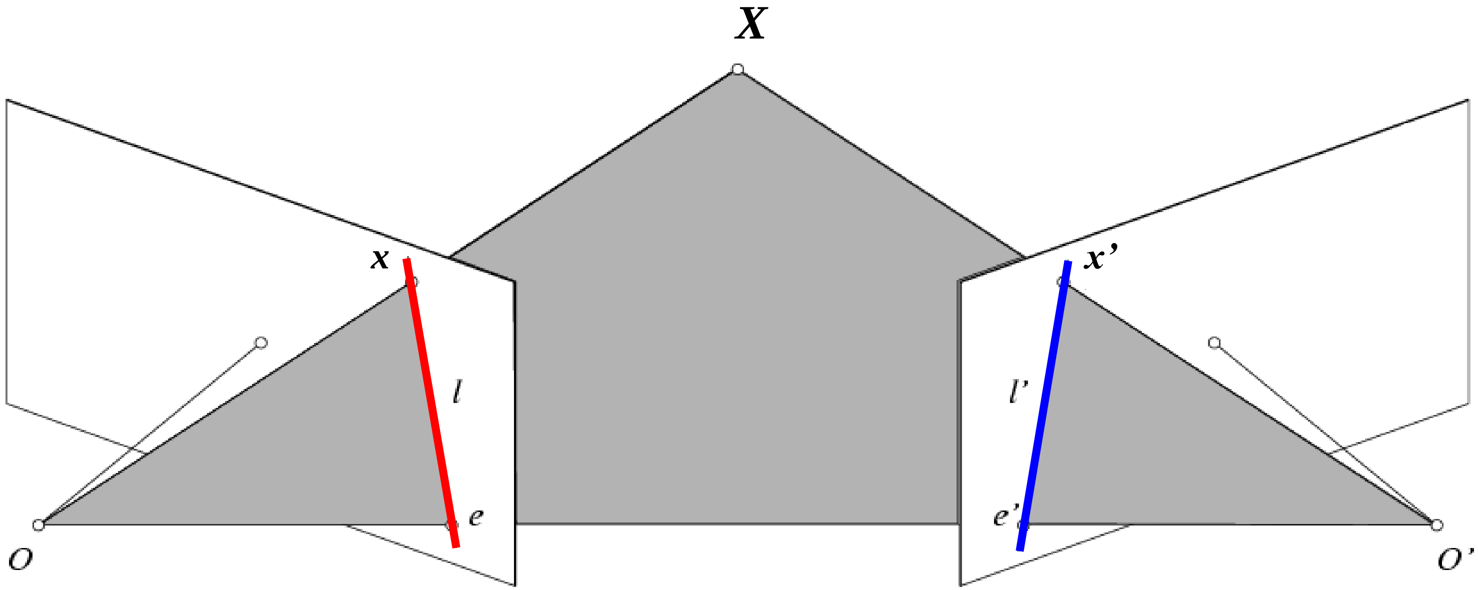
Properties of the Fundamental matrix



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

- *F x'* is the epipolar line associated with *x'* (*l* = *F x'*)

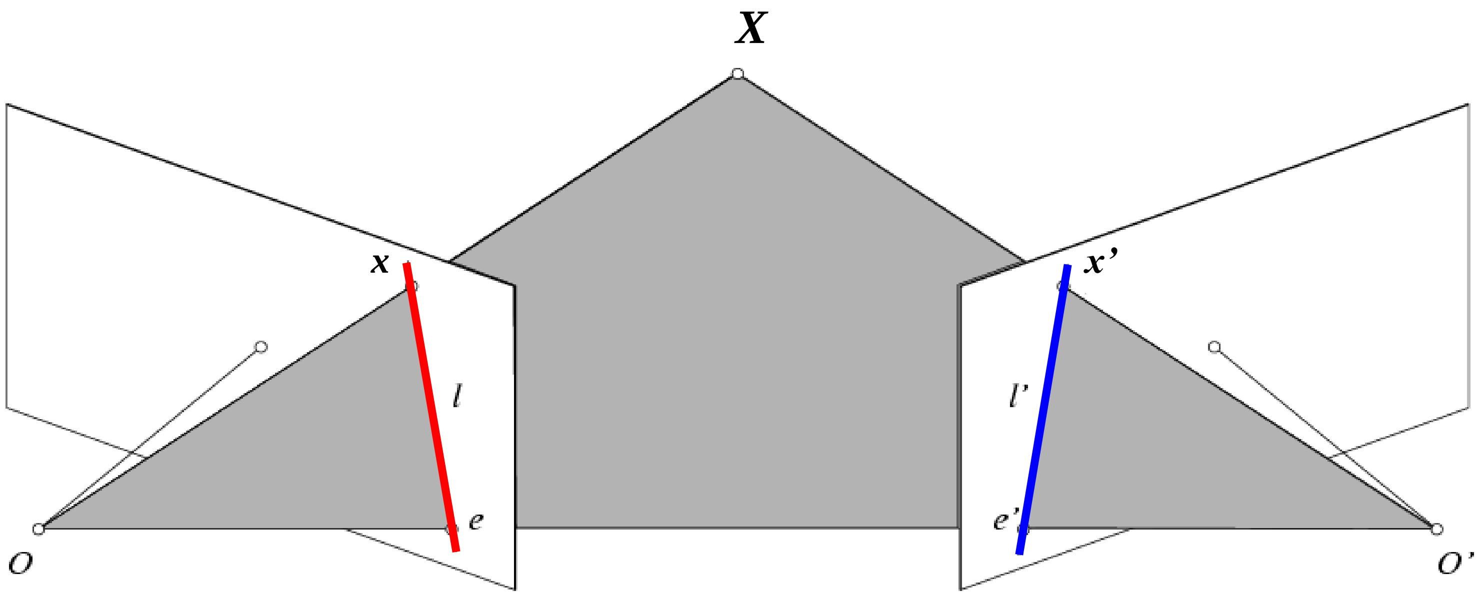
Properties of the Fundamental matrix



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

- *F x'* is the epipolar line associated with *x'* (*l* = *F x'*)
- $F^T x$ is the epipolar line associated with *x* ($l' = F^T x$)

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* is singular (rank two): det(F)=0

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

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

8-point algorithm

1. Solve a system of homogeneous linear equations
   a. Write down the system of equations

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

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

8-point algorithm

1. Solve a system of homogeneous linear equations
    a. Write down the system of equations

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

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

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

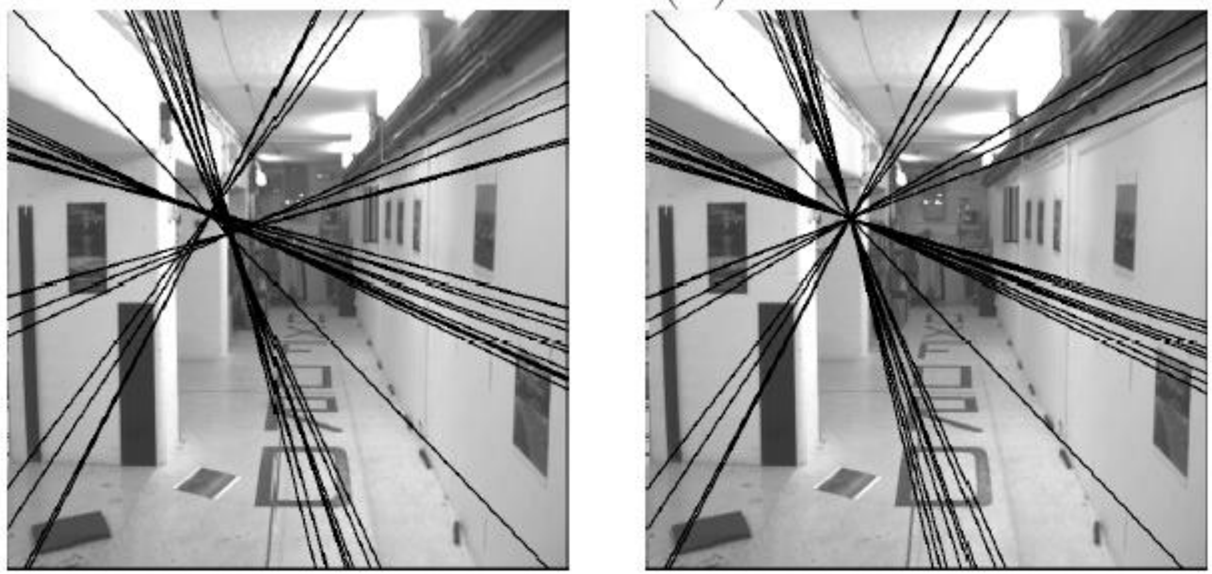
8-point algorithm

1. Solve a system of homogeneous linear equations
   a. Write down the system of equations
   b. Solve **f** from A**f**=**0** using SVD

Matlab:
```
[U, S, V] = svd(A);
f = V(:, end);
F = reshape(f, [3 3])';
```

Need to enforce singularity constraint

Fundamental matrix has rank 2 : $\det(\mathtt{F}) = 0$.



Left : Uncorrected $\mathtt{F}$ – epipolar lines are not coincident.

Right : Epipolar lines from corrected $\mathtt{F}$.

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

      Matlab:
      ```
      [U, S, V] = svd(A);
      f = V(:, end);
      F = reshape(f, [3 3])';
      ```
2. Resolve det(F) = 0 constraint using SVD

   Matlab:
   ```
   [U, S, V] = svd(F);
   S(3,3) = 0;
   F = U*S*V';
   ```

8-point algorithm

Notes:

- Use RANSAC to deal with outliers (sample 8 points)
  - How to test for outliers? $|\boldsymbol{x}^T \boldsymbol{F} \boldsymbol{x}'| < \boldsymbol{threshold}$
- Solve in normalized coordinates
  - mean=0
  - standard deviation ~= (1,1,1)

Moving on to stereo…

Fuse a calibrated binocular stereo pair to produce a depth image

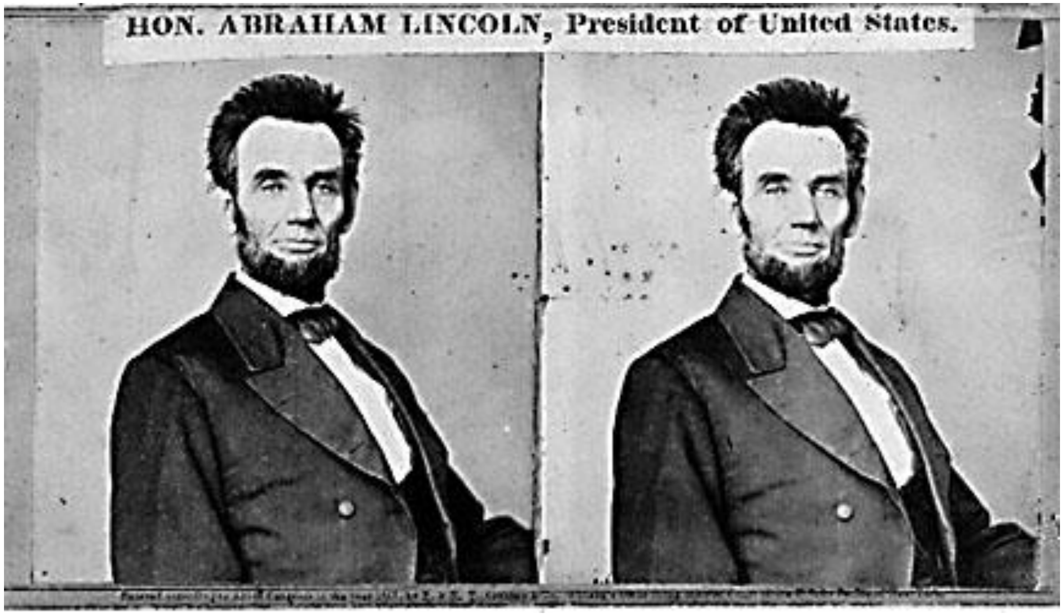
image 1

image 2

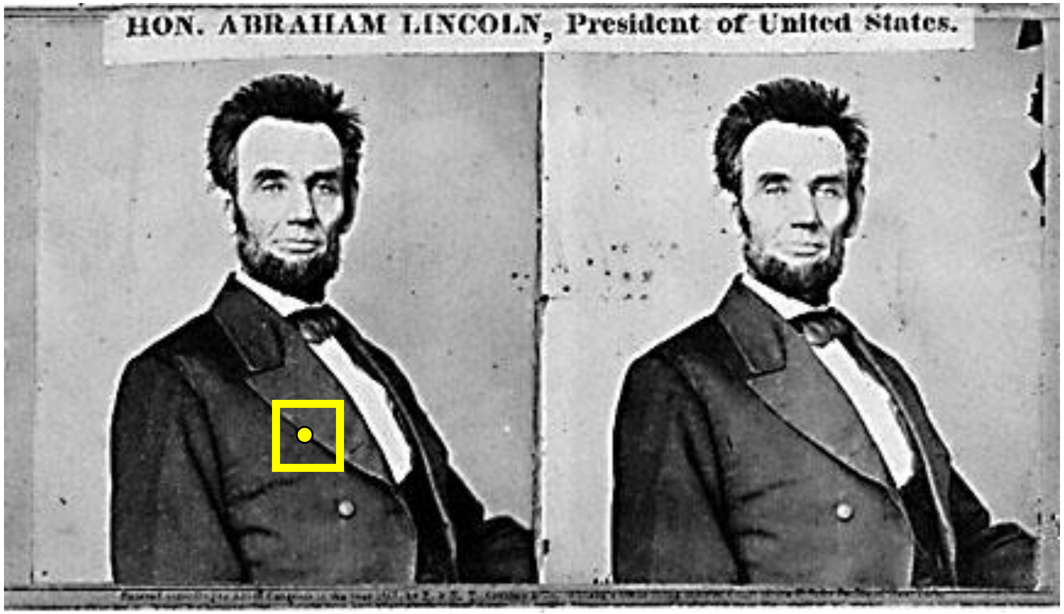
Depth map

Many of these slides adapted from Steve Seitz and Lana Lazebnik

Basic stereo matching algorithm

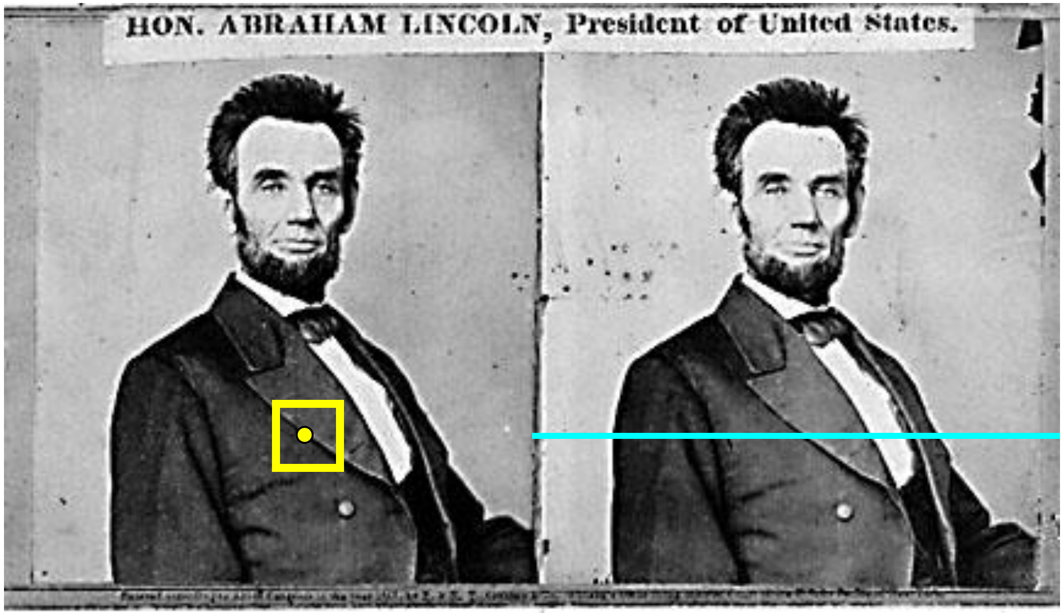


Basic stereo matching algorithm



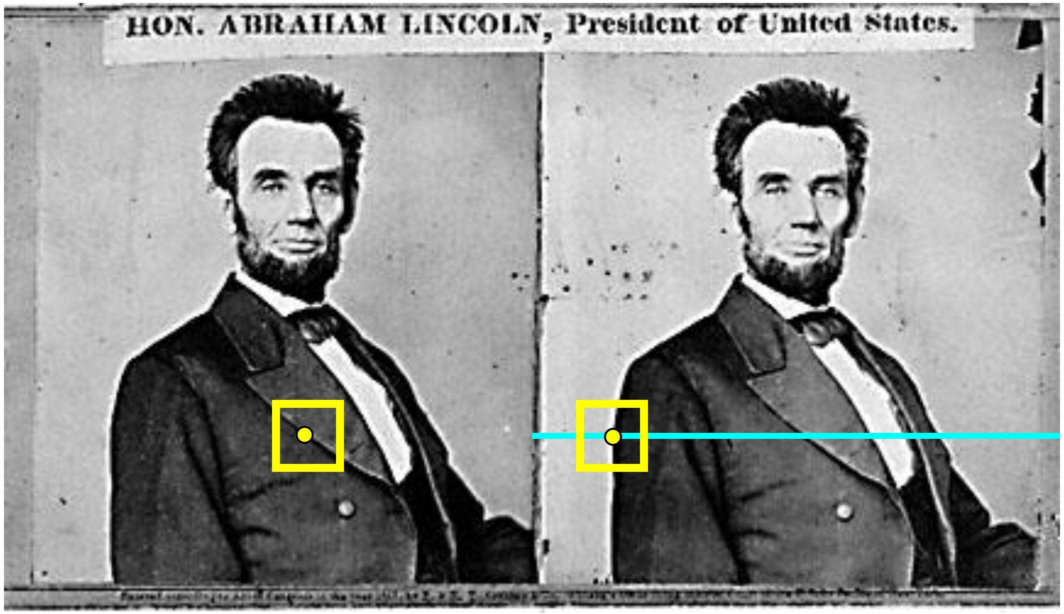
- For each pixel in the first image

Basic stereo matching algorithm



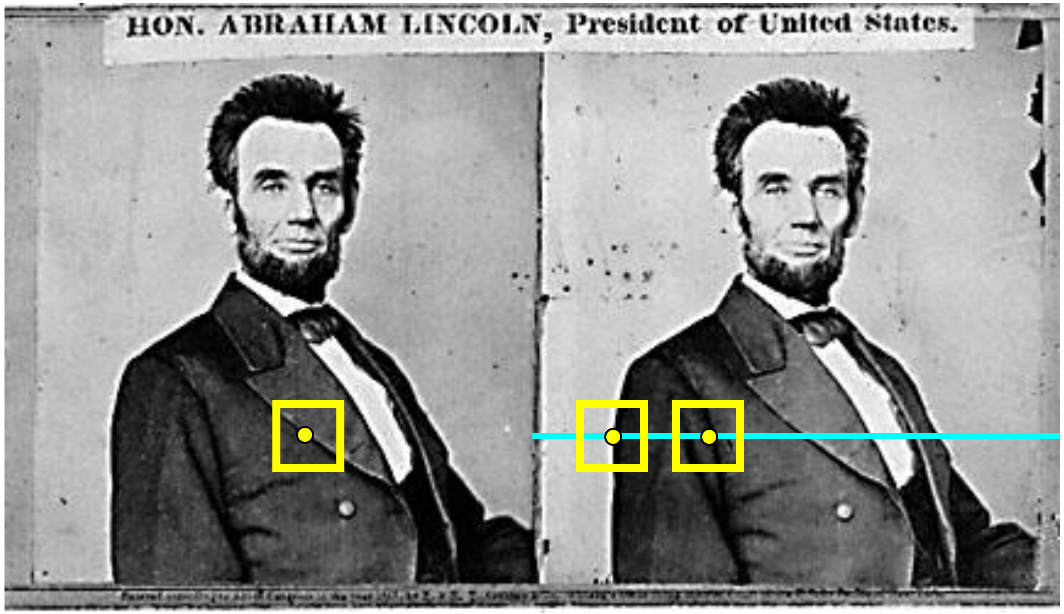
- For each pixel in the first image
  - Find corresponding epipolar line in the right image

Basic stereo matching algorithm



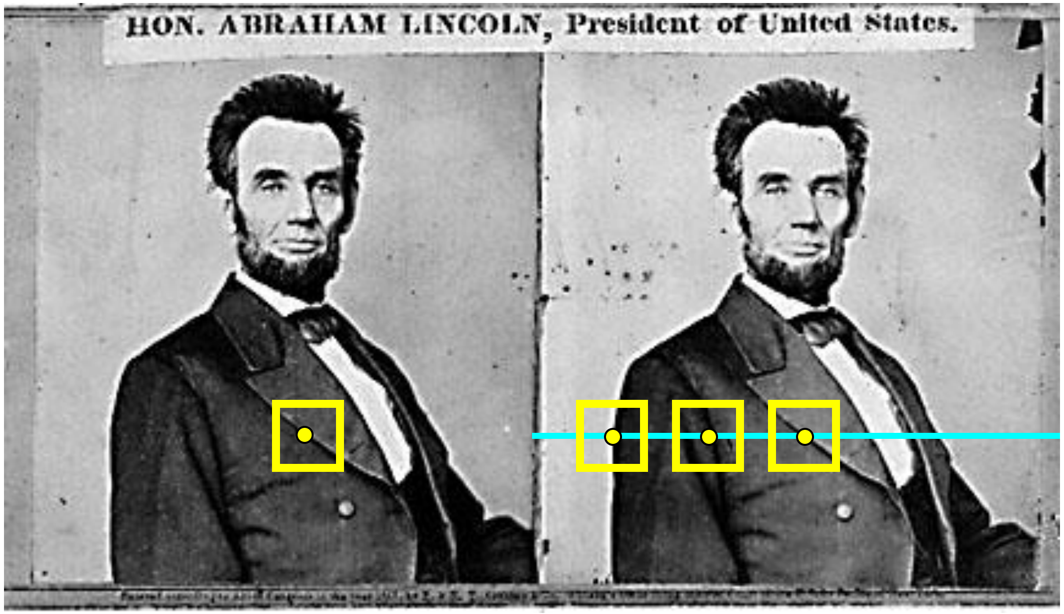
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Search along epipolar line and pick the best match

Basic stereo matching algorithm



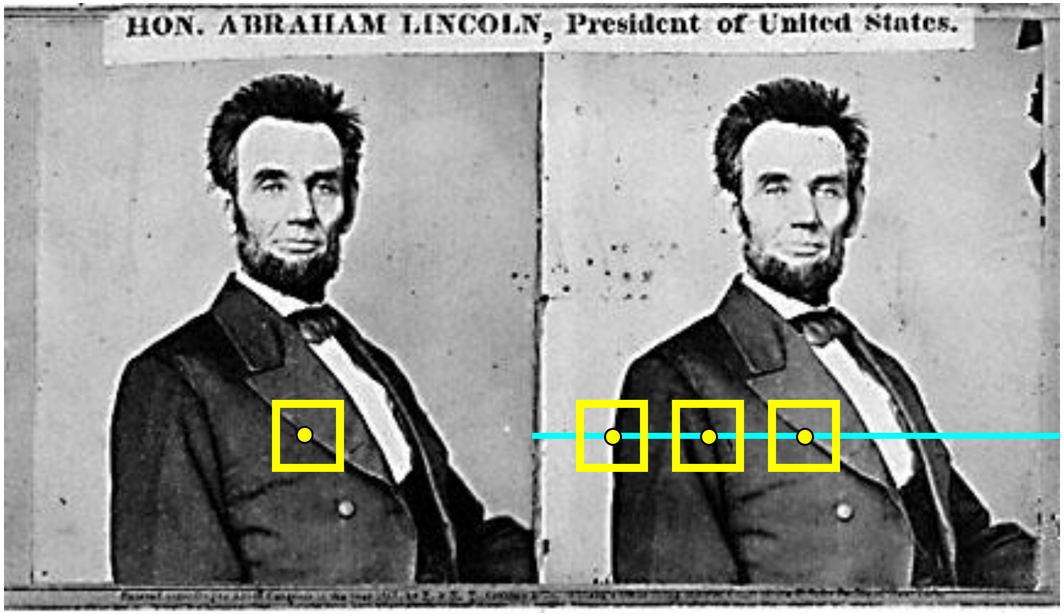
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Search along epipolar line and pick the best match

Basic stereo matching algorithm



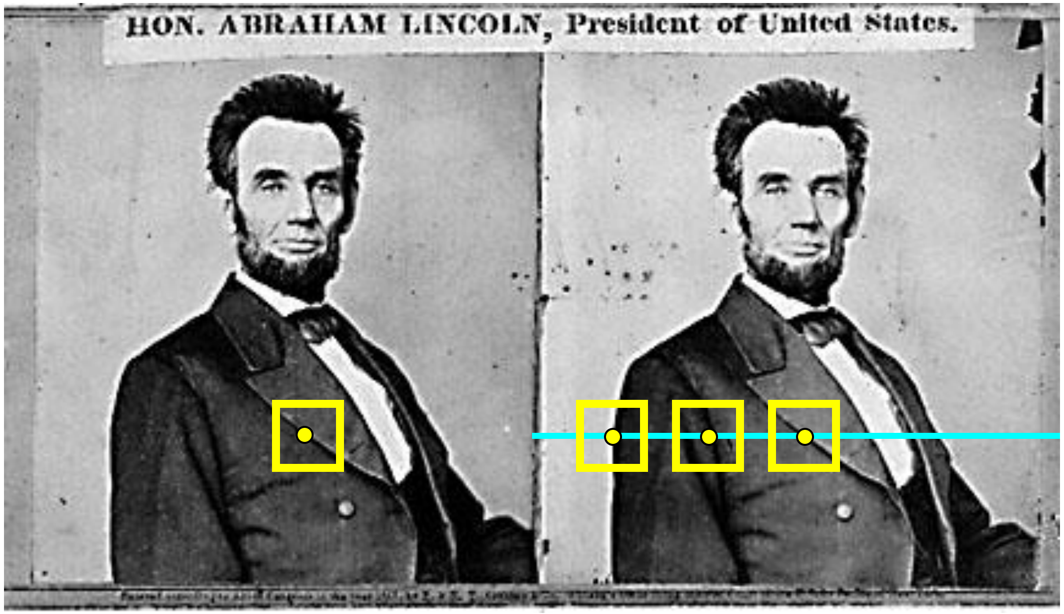
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Search along epipolar line and pick the best match

Basic stereo matching algorithm



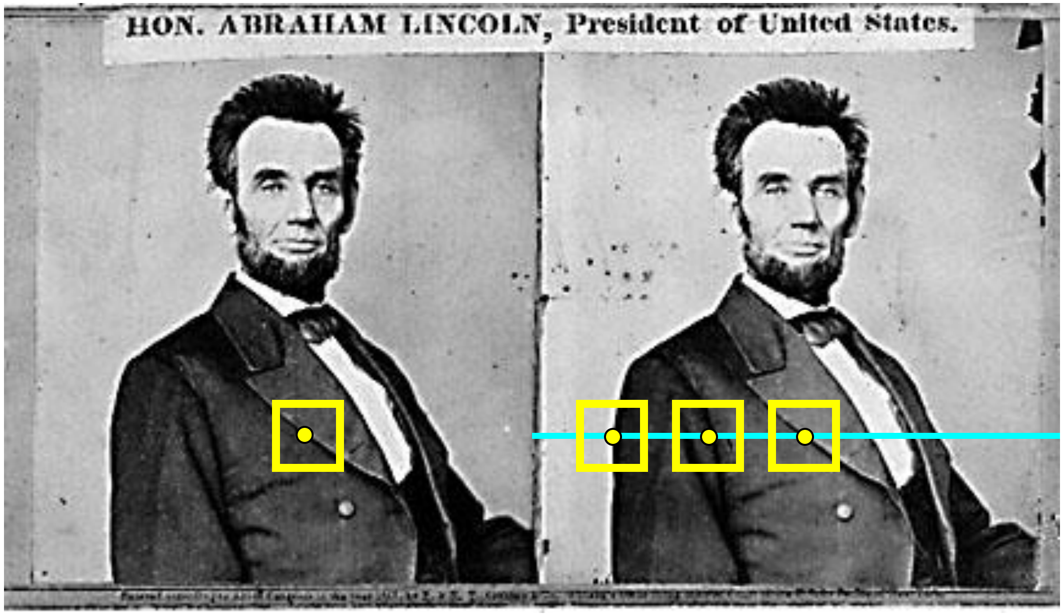
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Search along epipolar line and pick the best match
  - Triangulate the matches to get depth information

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: epipolar lines are scanlines

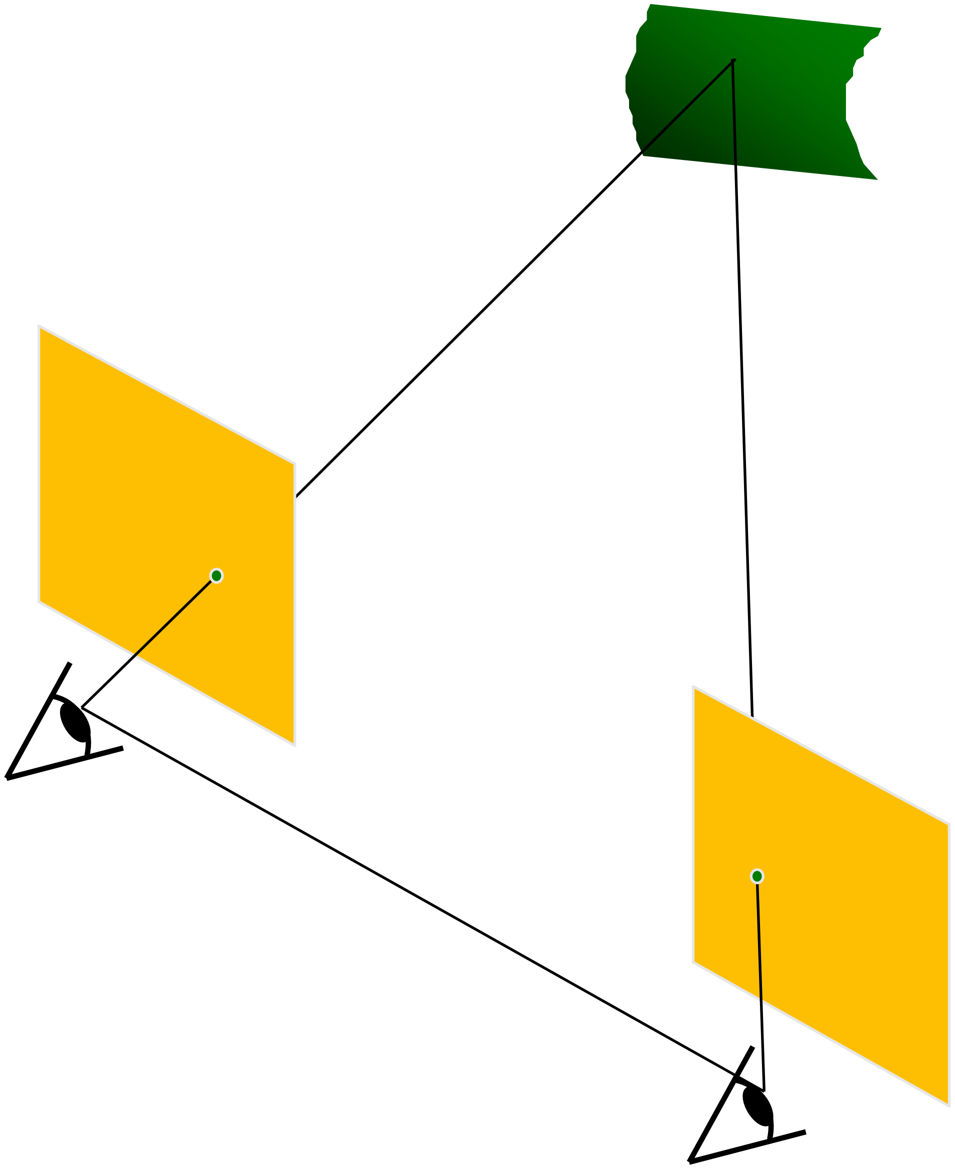
Basic stereo matching algorithm



- 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

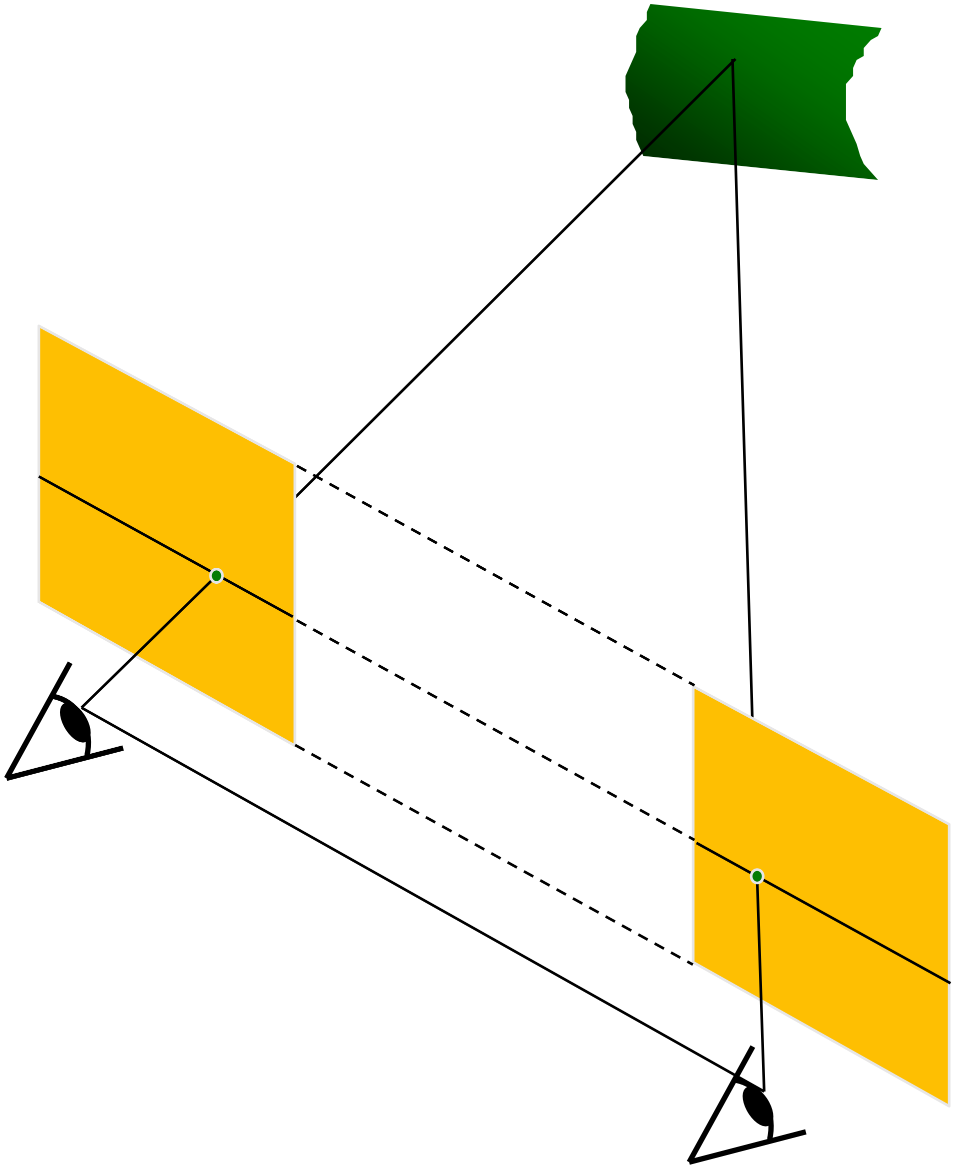
- Simplest case: epipolar lines are scanlines
  - When does this happen?

Simplest Case: Parallel images



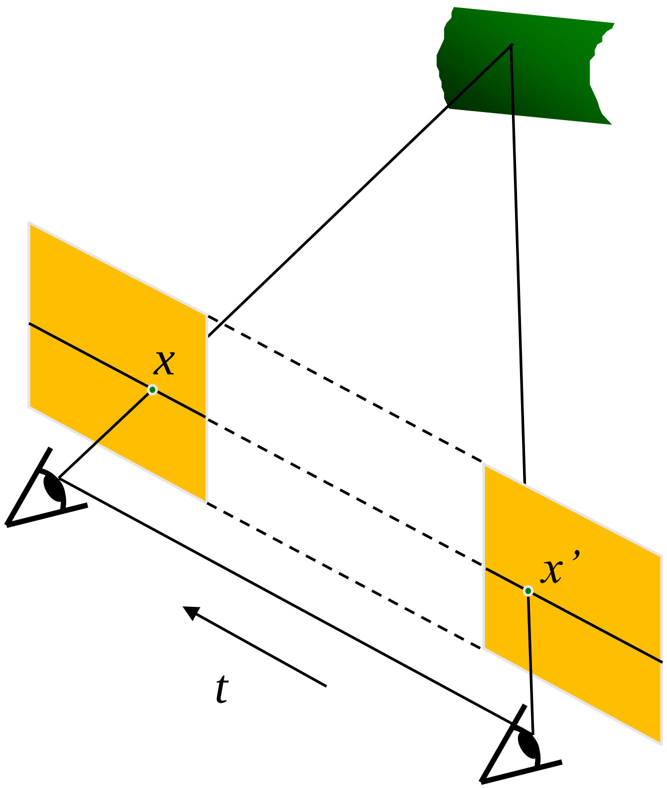
- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same

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



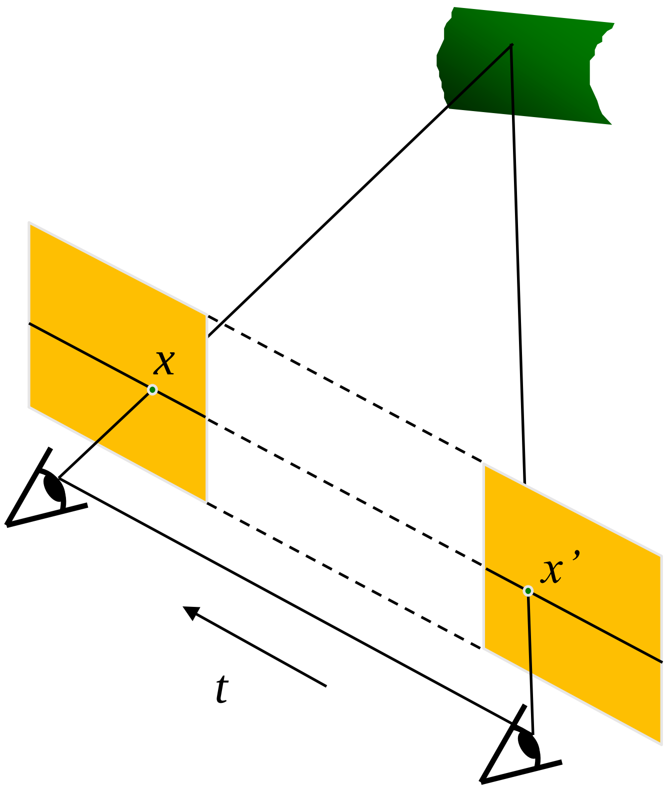
Epipolar constraint:

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

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

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

Simplest Case: Parallel images



Epipolar constraint:

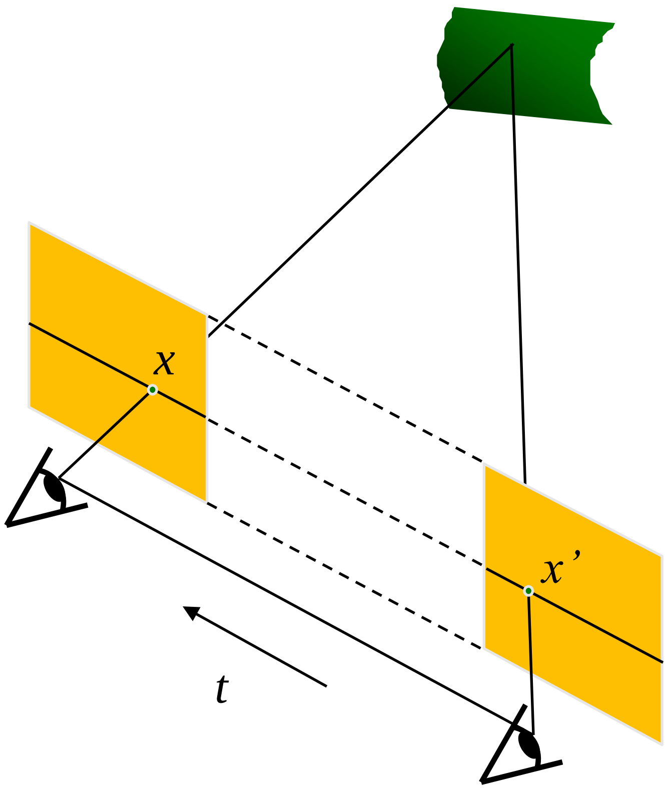
$$x^T E x' = 0, \quad E = t \times R$$

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

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

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

Simplest Case: Parallel images



Epipolar constraint:

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

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

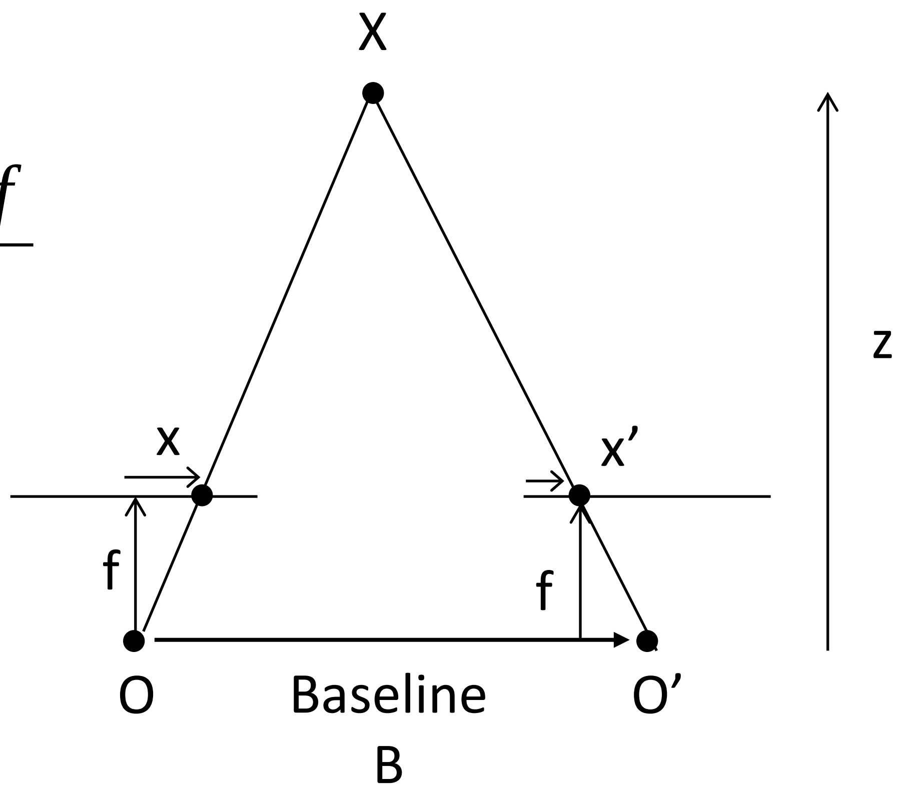
$$E = t \times R = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

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

The y-coordinates of corresponding points are the same

Depth from disparity

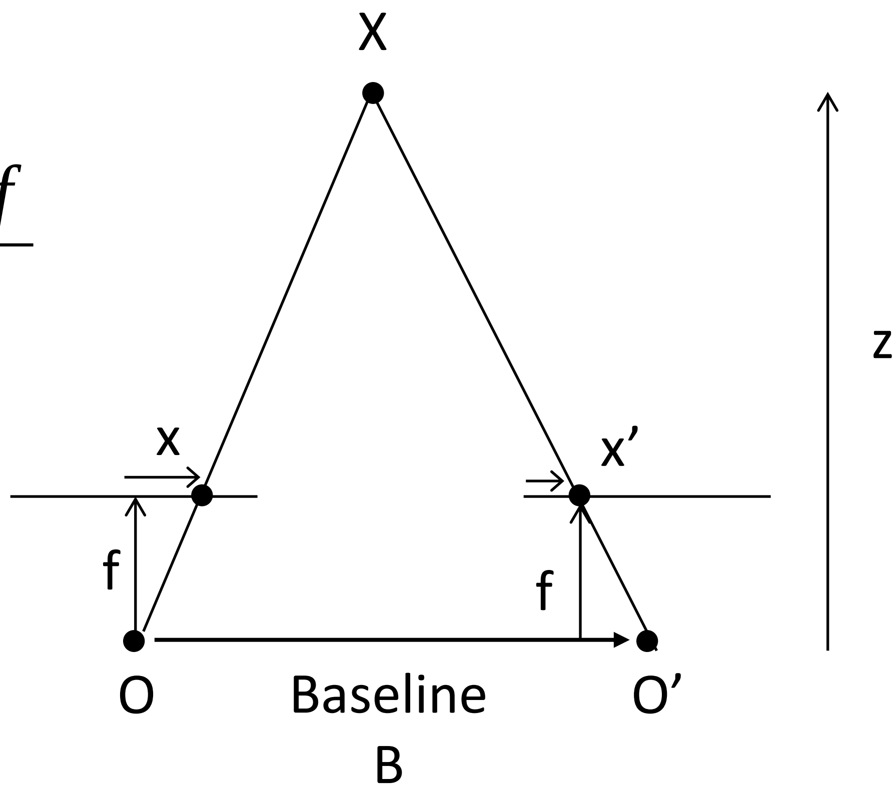
$$\frac{B - x + x'}{B} = \frac{z - f}{z}$$



Depth from disparity

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

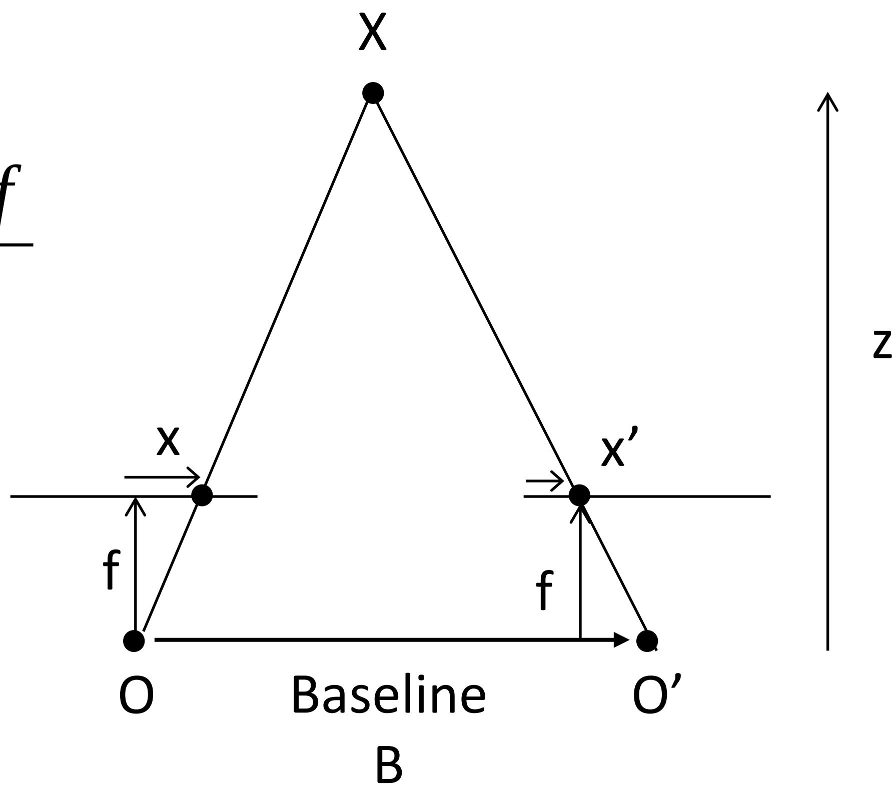
$$\Rightarrow \quad \frac{x - x'}{B} = \frac{f}{z}$$



Depth from disparity

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

$$\Rightarrow \quad \frac{x - x'}{B} = \frac{f}{z}$$

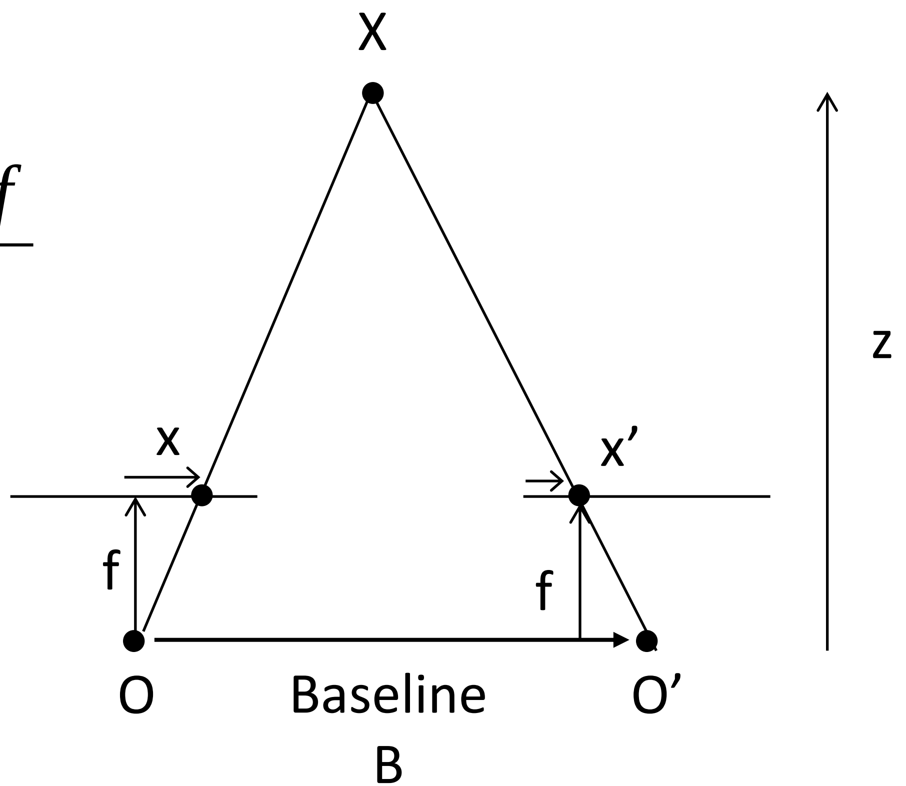


$$disparity = x - x' = \frac{B \cdot f}{z}$$

Depth from disparity

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

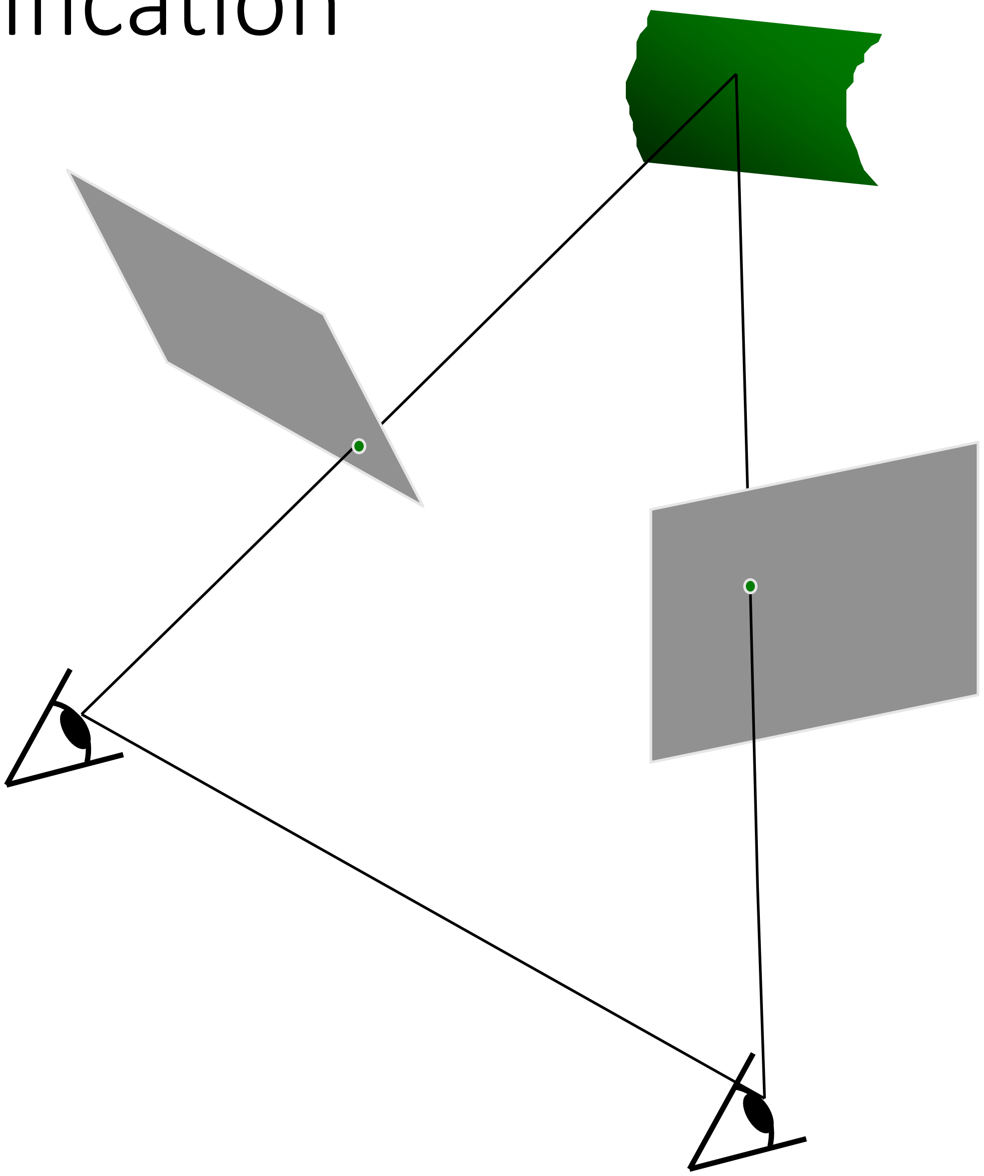
$$\Rightarrow \frac{x - x'}{B} = \frac{f}{z}$$



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

Disparity is inversely proportional to depth.

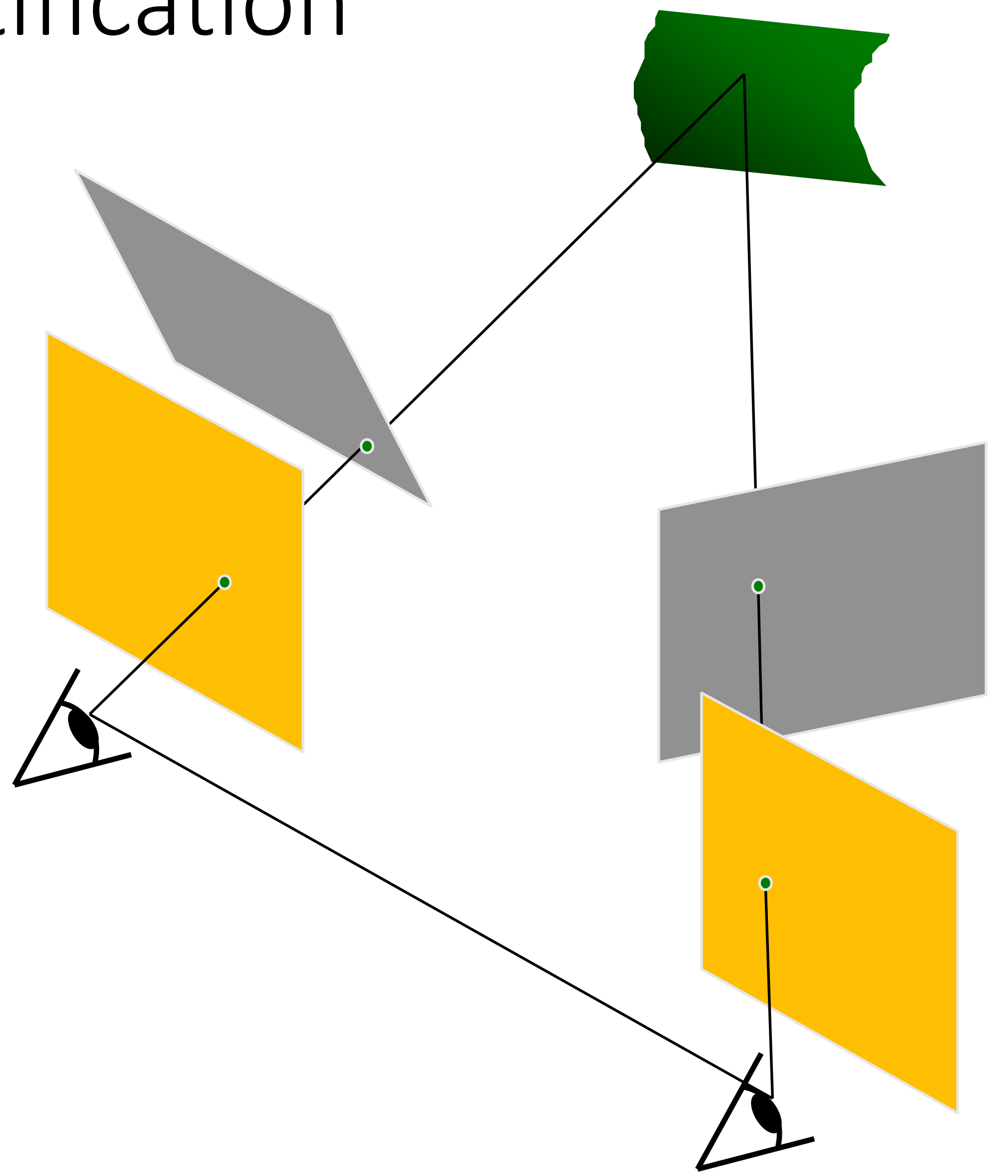
Stereo image rectification



C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). IEEE Conf. Computer Vision and Pattern Recognition, 1999.

Stereo image rectification

- Reproject image planes onto a common plane parallel to the line between camera centers

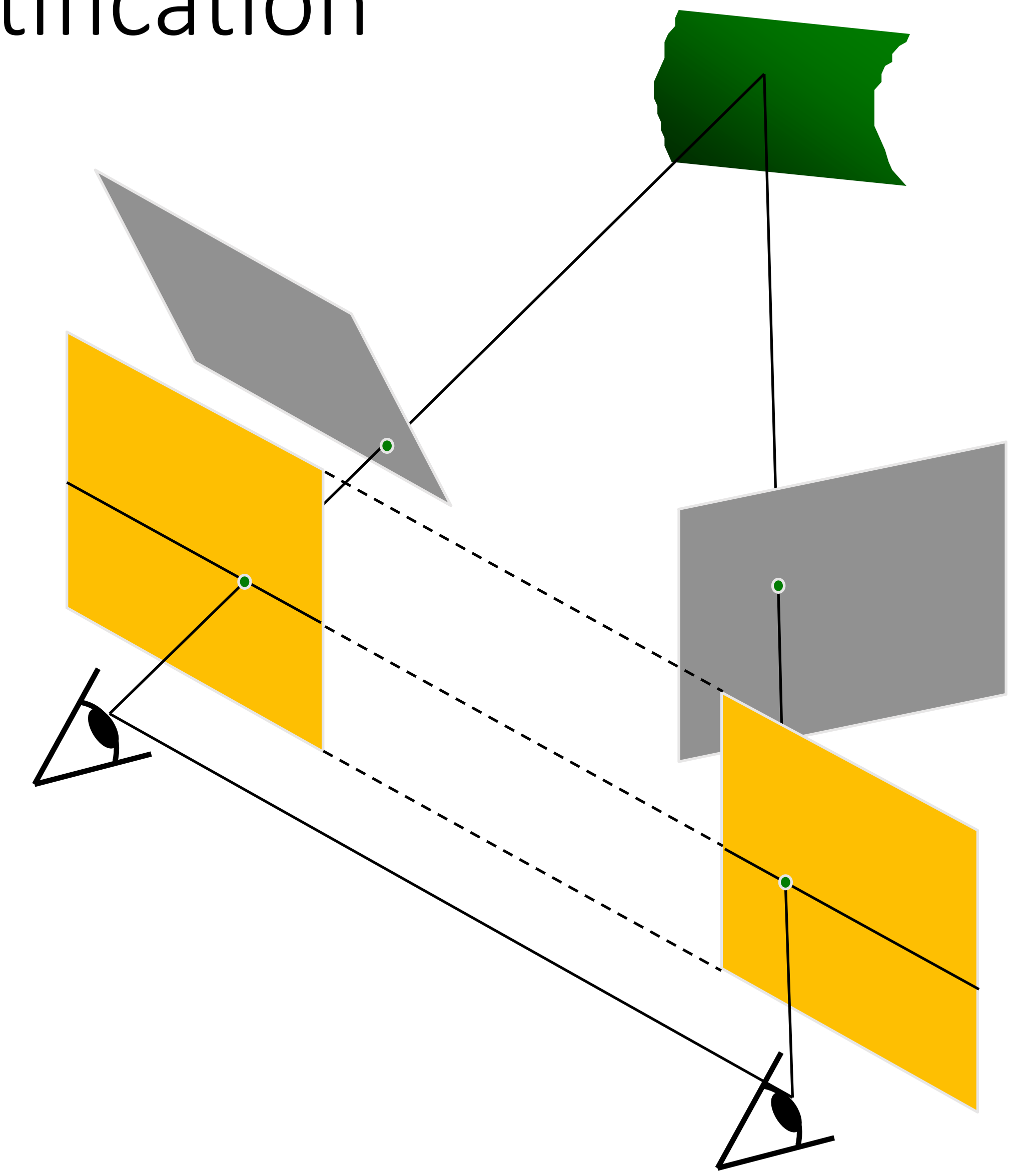


C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). IEEE Conf. Computer Vision and Pattern Recognition, 1999.

Stereo image rectification

- Reproject image planes onto a common plane parallel to the line between camera centers

- Pixel motion is horizontal after this transformation



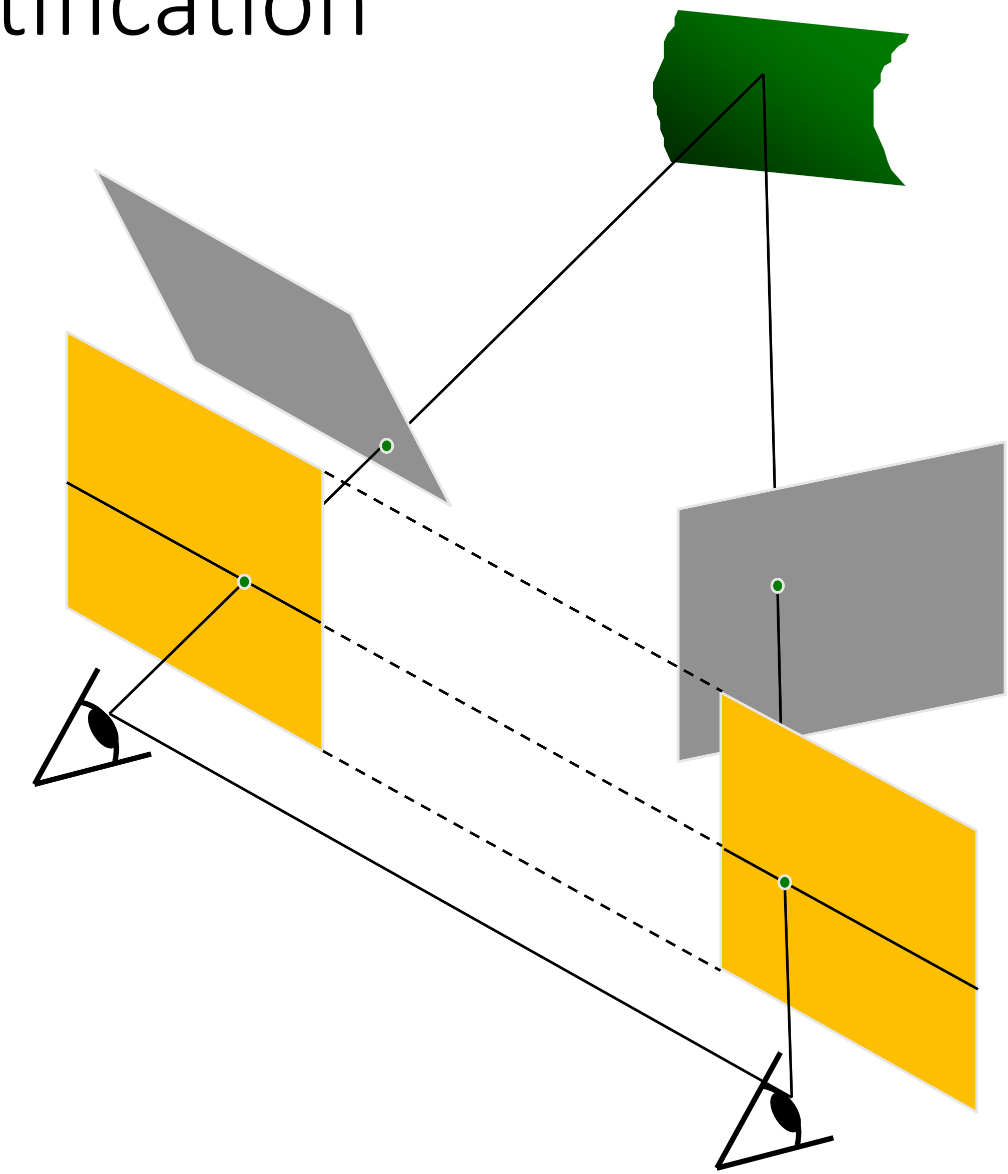
C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). IEEE Conf. Computer Vision and Pattern Recognition, 1999.

Stereo image rectification

- Reproject image planes onto a common plane parallel to the line between camera centers

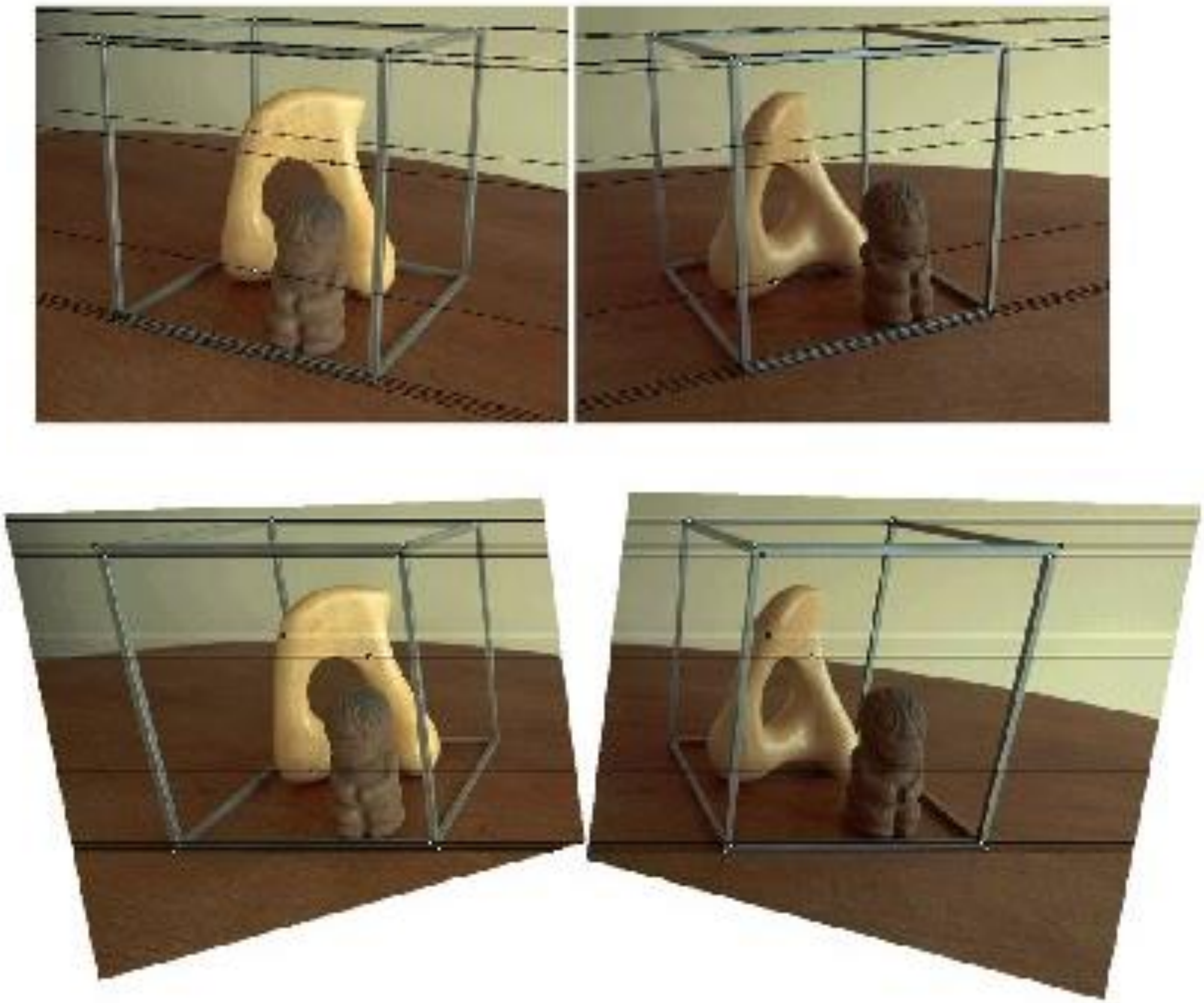
- Pixel motion is horizontal after this transformation

- Two homographies (3x3 transform), one for each input image reprojection

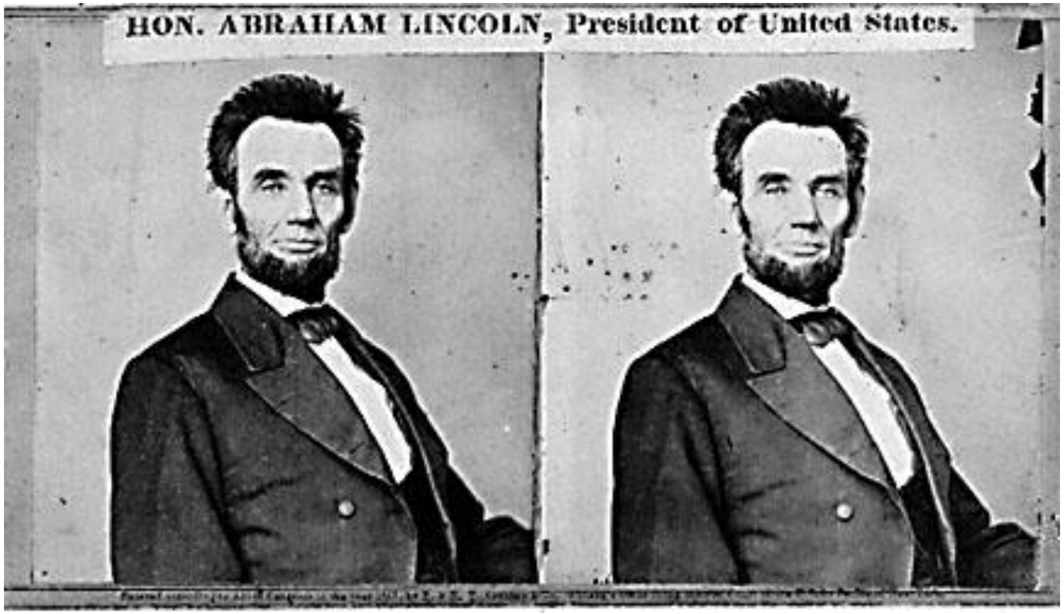


C. Loop and Z. Zhang. Computing Rectifying Homographies for Stereo Vision. IEEE Conf. Computer Vision and Pattern Recognition, 1999.

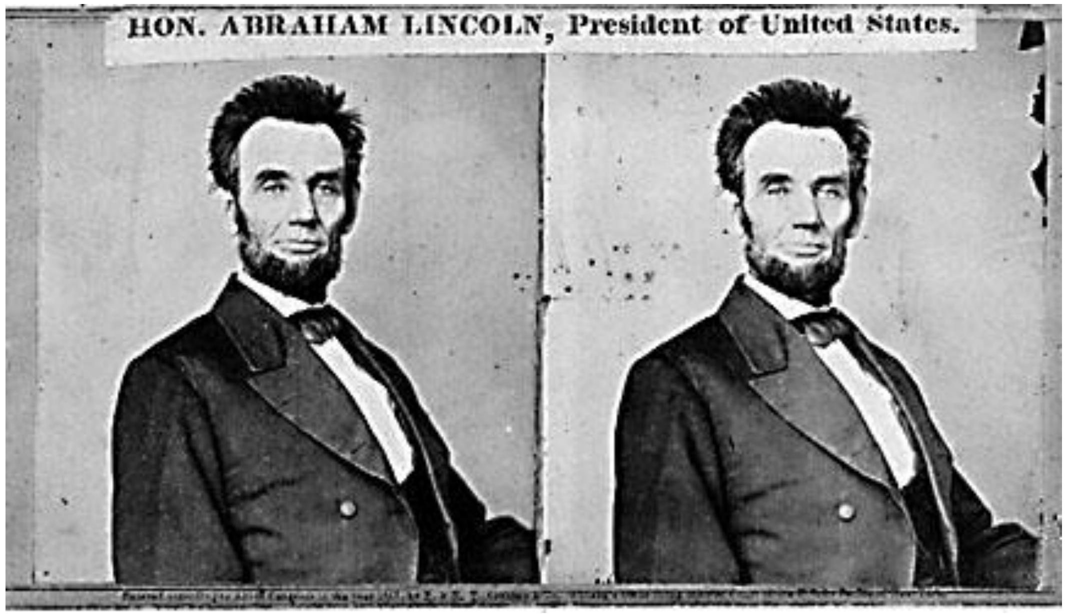
Rectification example



Basic stereo matching algorithm

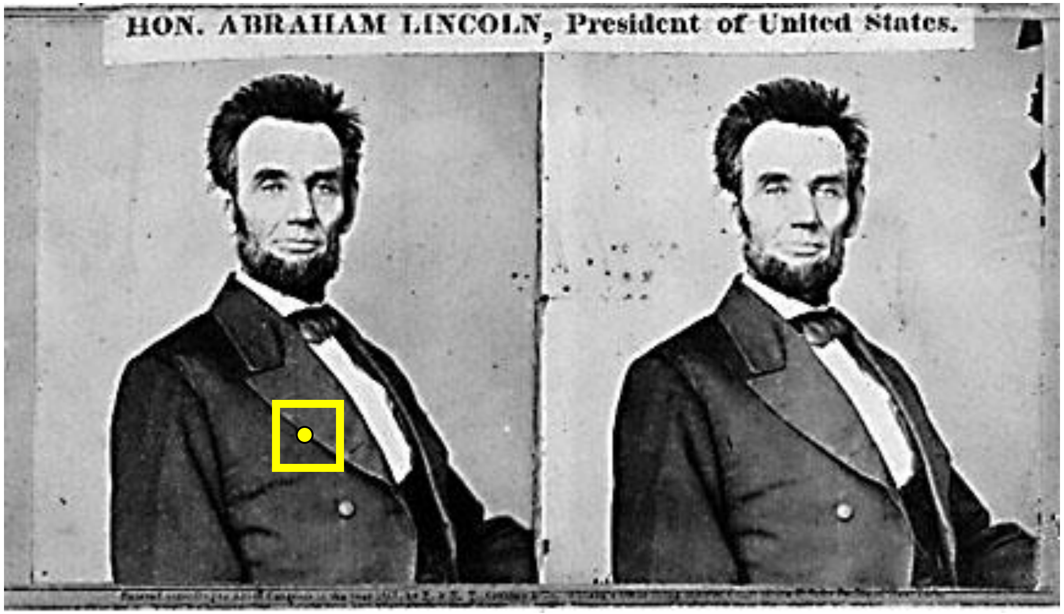


Basic stereo matching algorithm



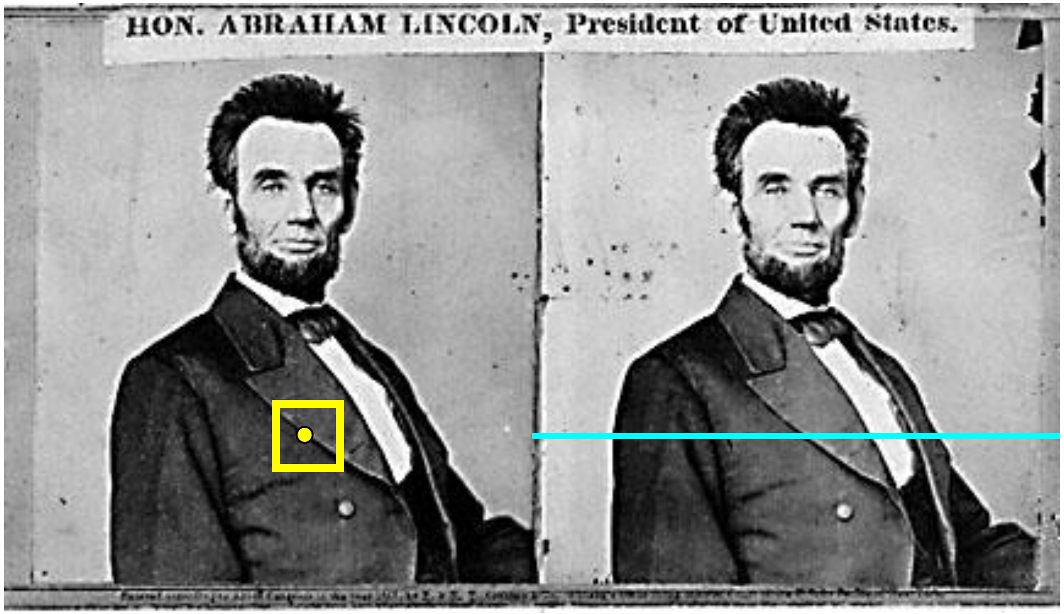
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines

Basic stereo matching algorithm



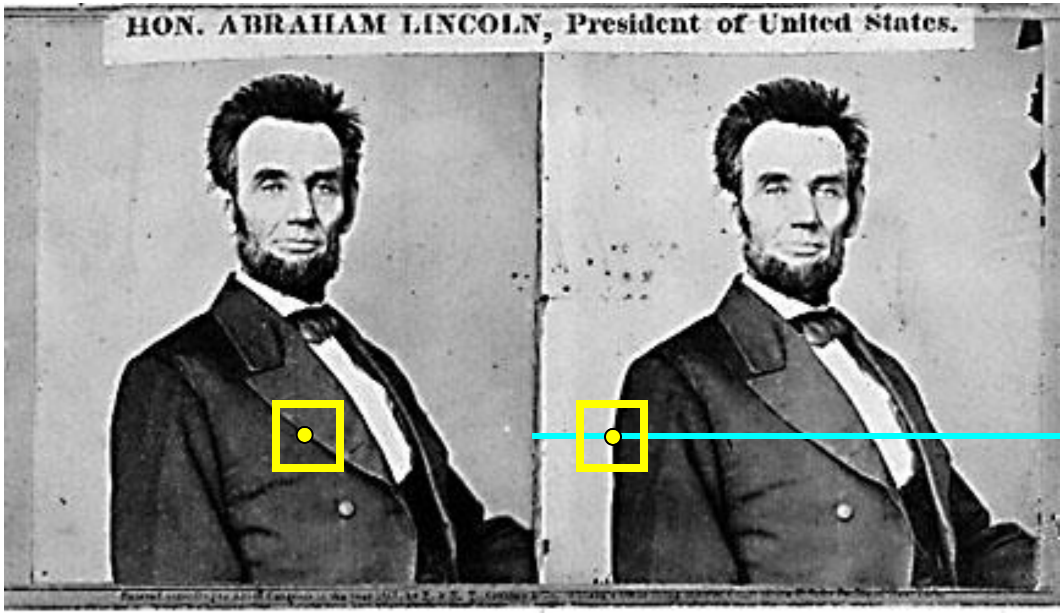
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image

Basic stereo matching algorithm



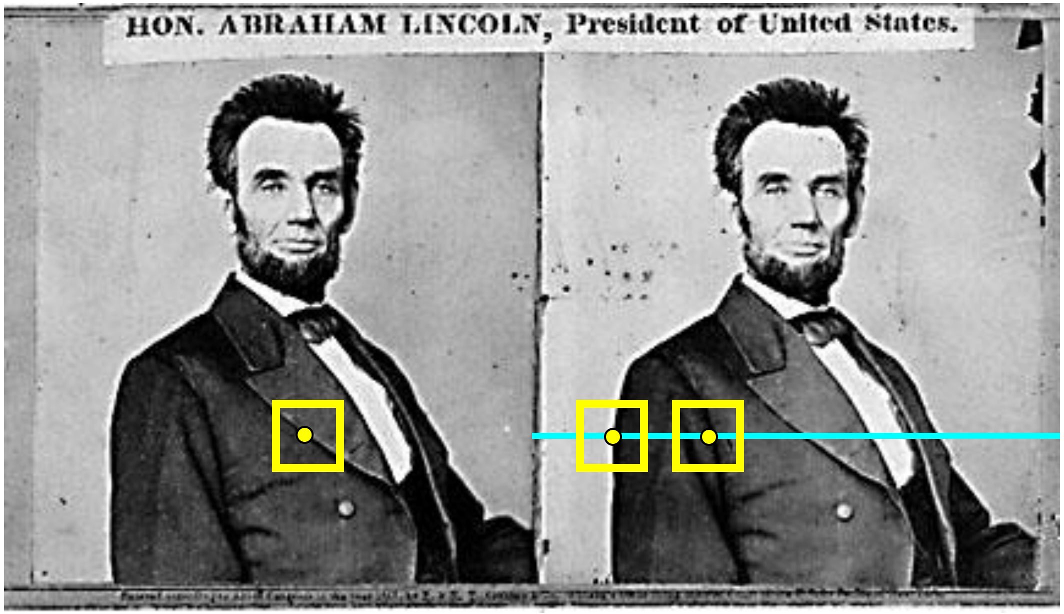
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image

Basic stereo matching algorithm



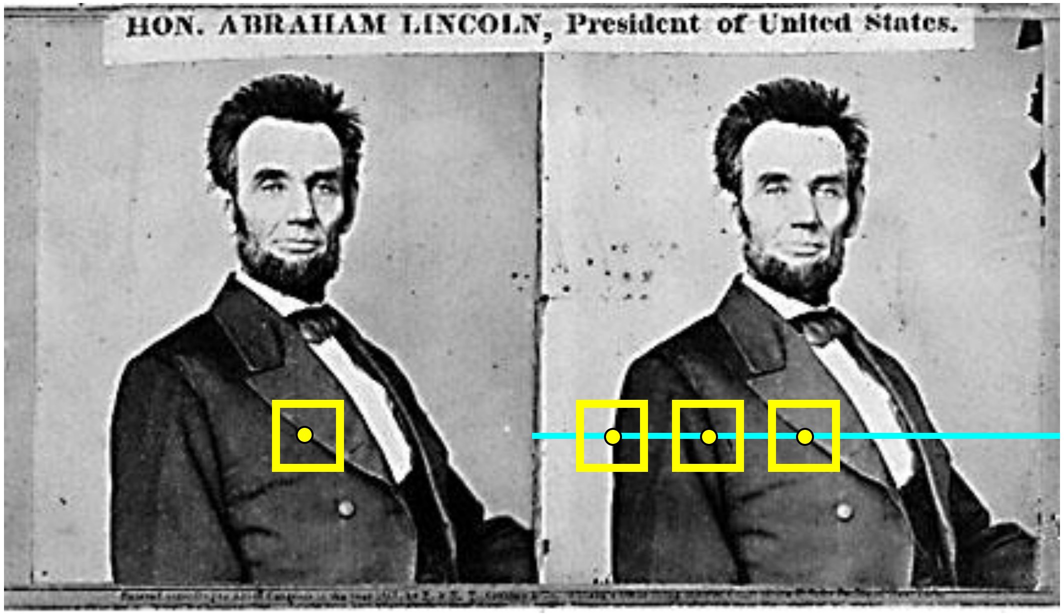
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Search the scanline and pick the best match x’

Basic stereo matching algorithm



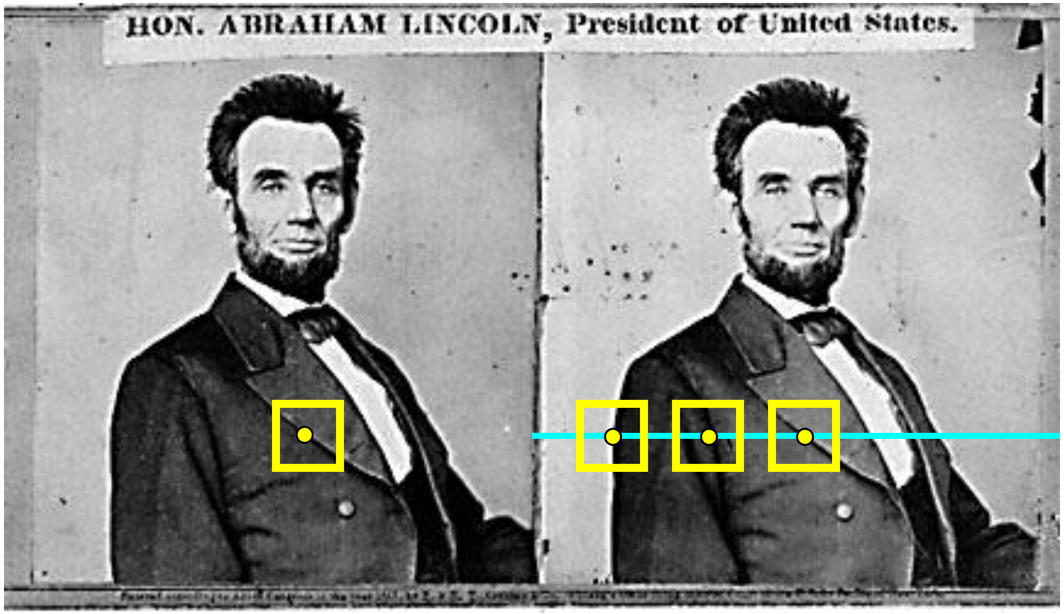
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Search the scanline and pick the best match x’

Basic stereo matching algorithm



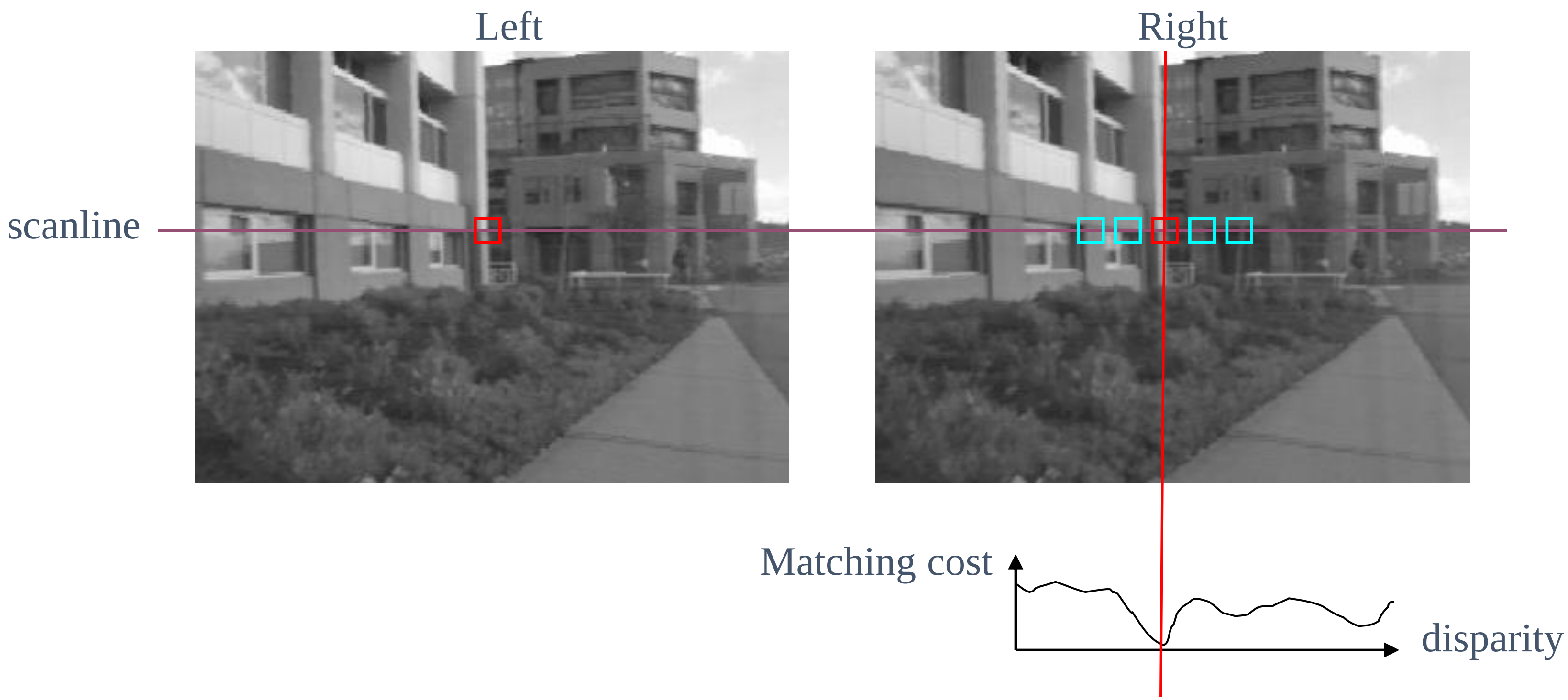
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Search the scanline and pick the best match x’

Basic stereo matching algorithm



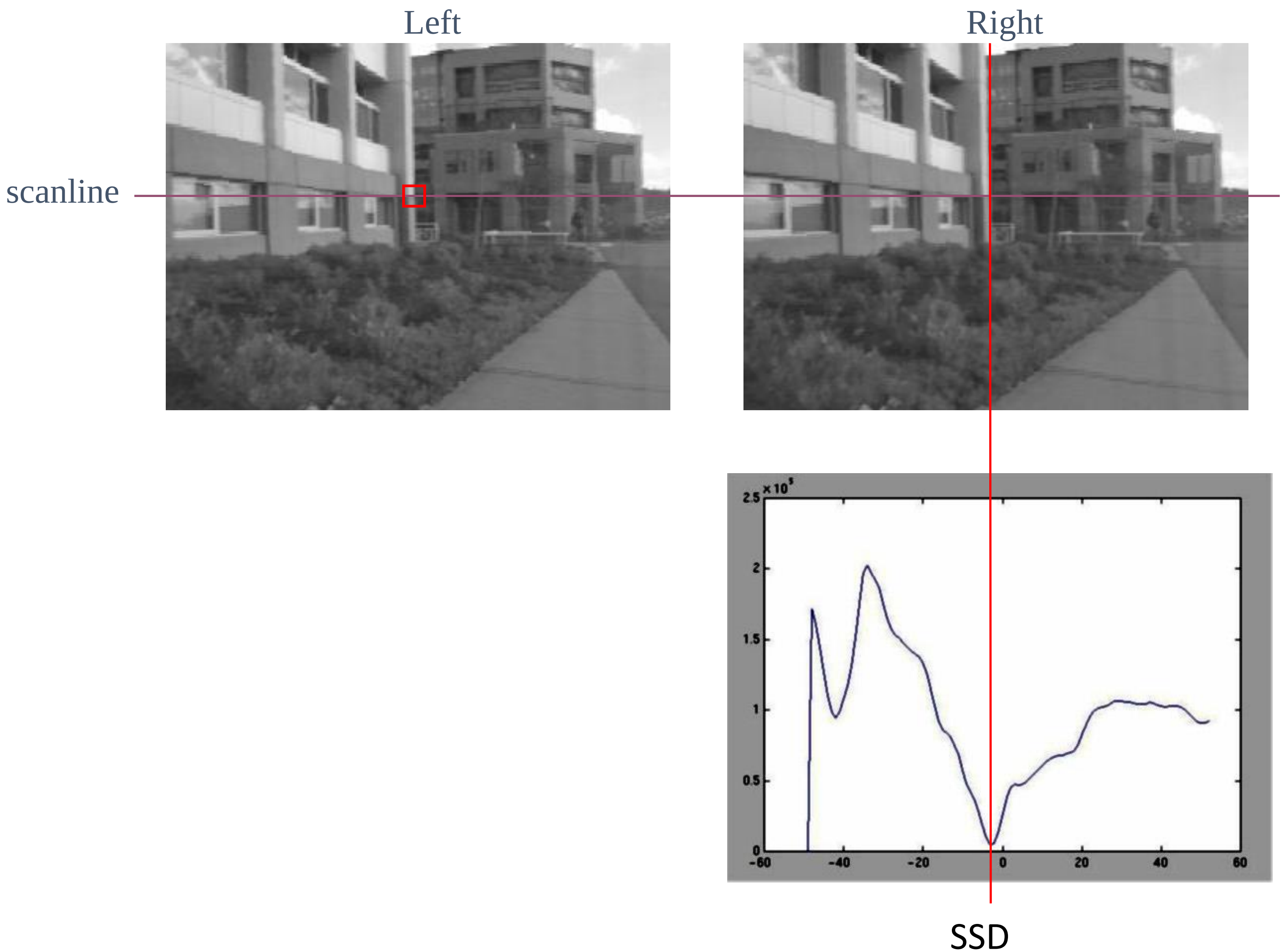
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Search the scanline and pick the best match x'
  - Compute disparity x-x' and set depth(x) = fB/(x-x')

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

Correspondence search

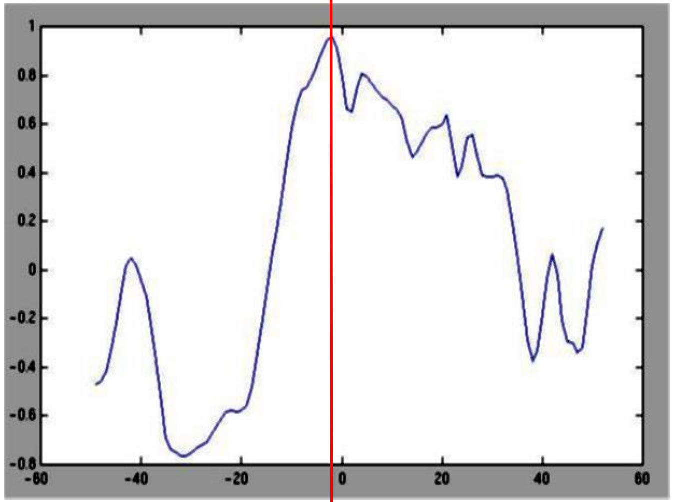


Correspondence search

Left

Right

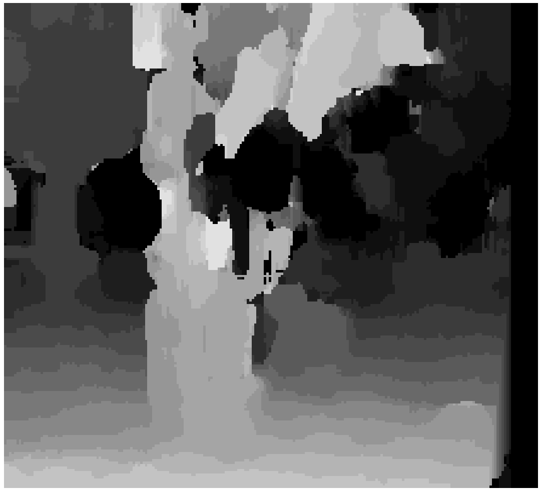
scanline



Norm. corr

Effect of window size

W = 3



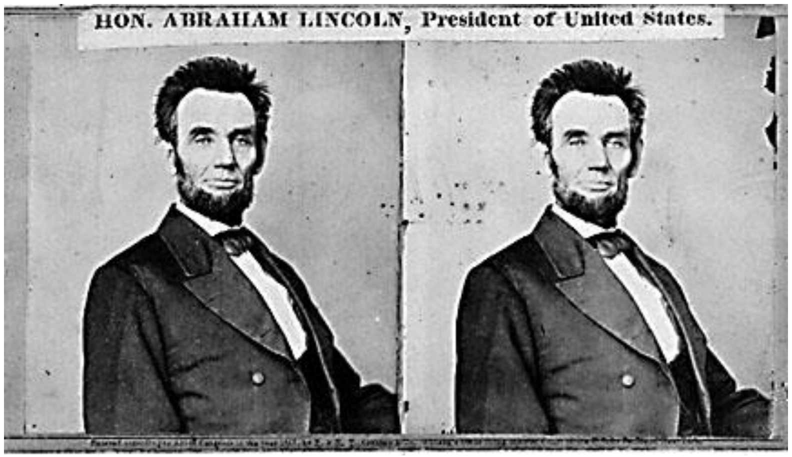
W = 20

- Smaller window
  - \+ More detail
  - – More noise

- Larger window
  - \+ Smoother disparity maps
  - – Less detail
  - – Fails near boundaries

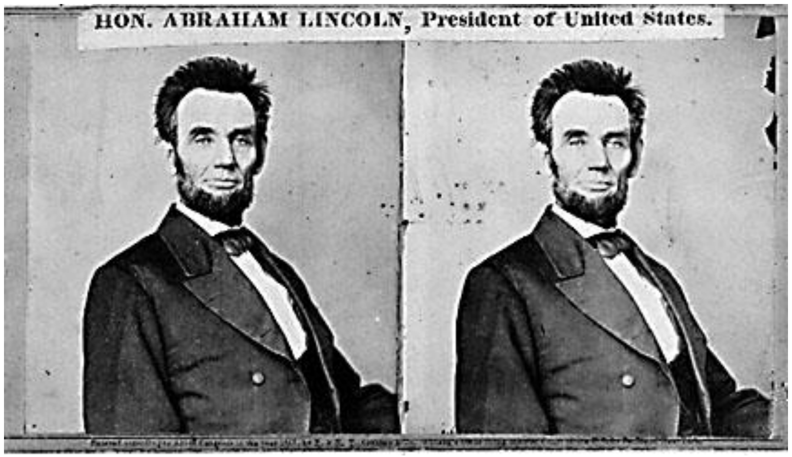
Failures of correspondence search

Failures of correspondence search



Textureless surfaces

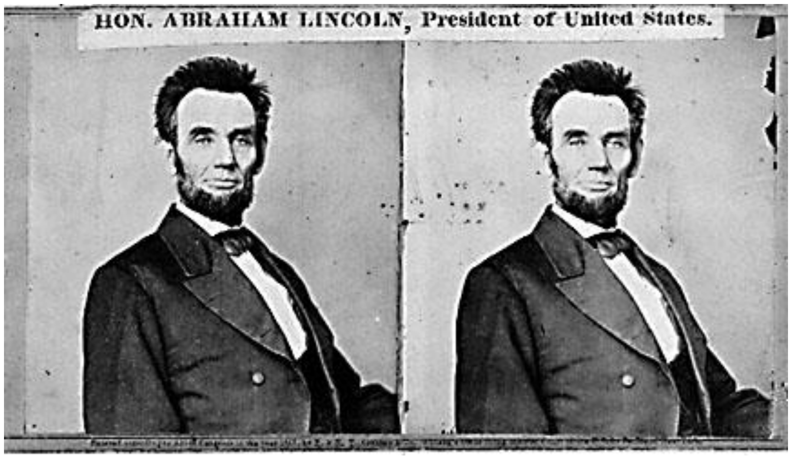
Failures of correspondence search



Textureless surfaces

Occlusions, repetition

Failures of correspondence search



Textureless surfaces

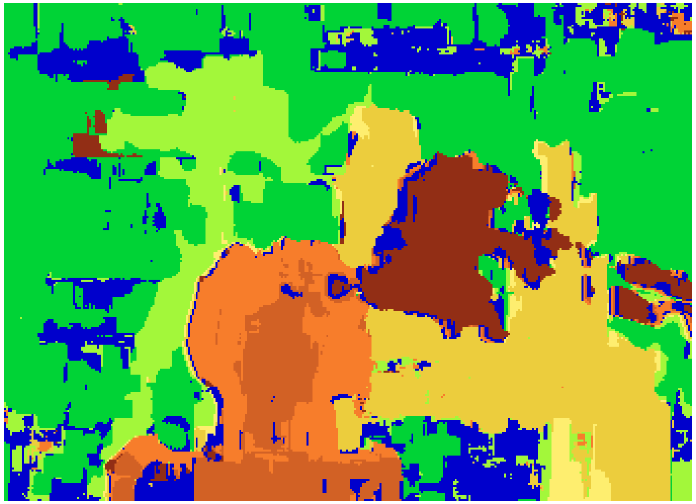
Occlusions, repetition

Non-Lambertian surfaces, specularities

Results with window search

Data

Window-based matching



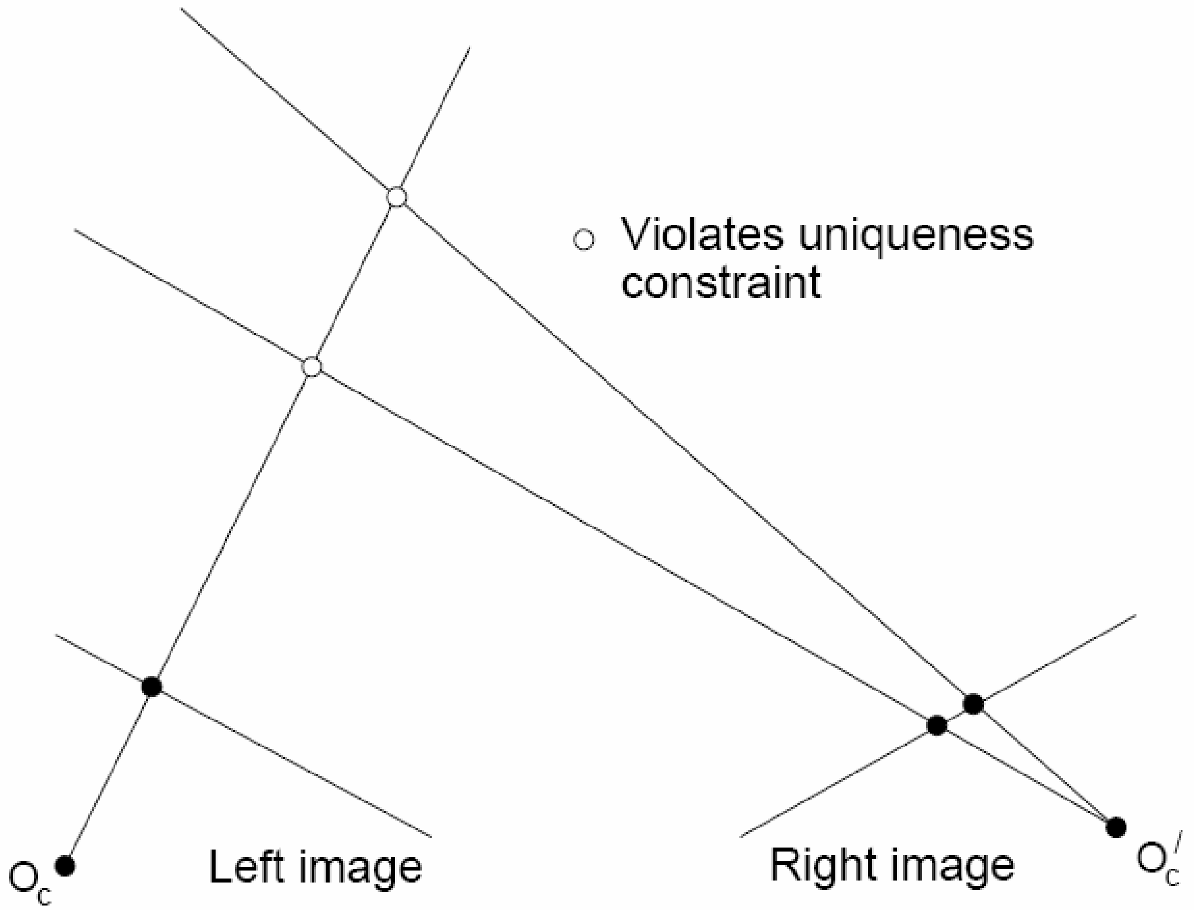
Ground truth

How can we improve window-based matching?

- So far, matches are independent for each point
- What constraints or priors can we add?

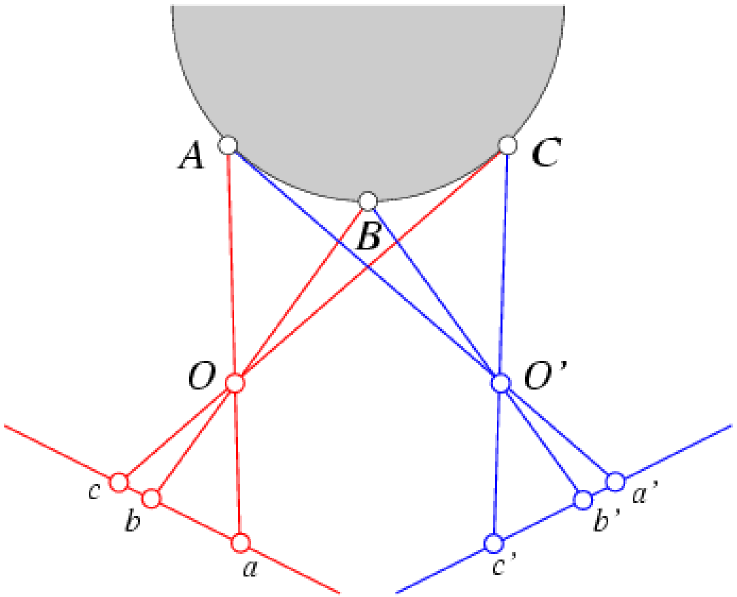
Stereo constraints/priors

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



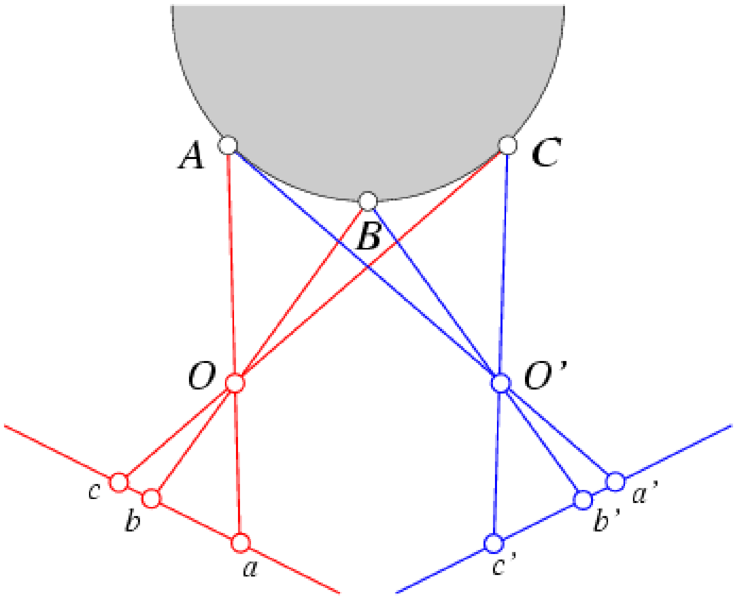
Stereo constraints/priors

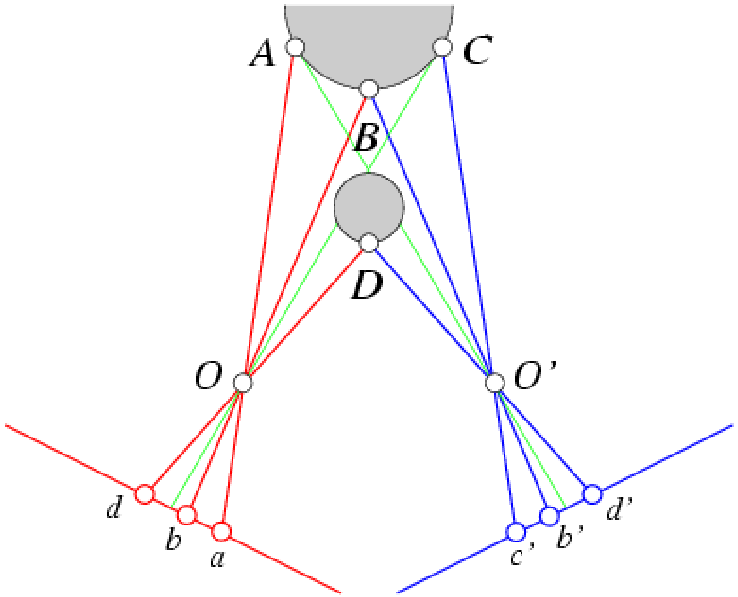
- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views



Stereo constraints/priors

- **Uniqueness**
  - For any point in one image, there should be at most one matching point in the other image
- **Ordering**
  - Corresponding points should be in the same order in both views



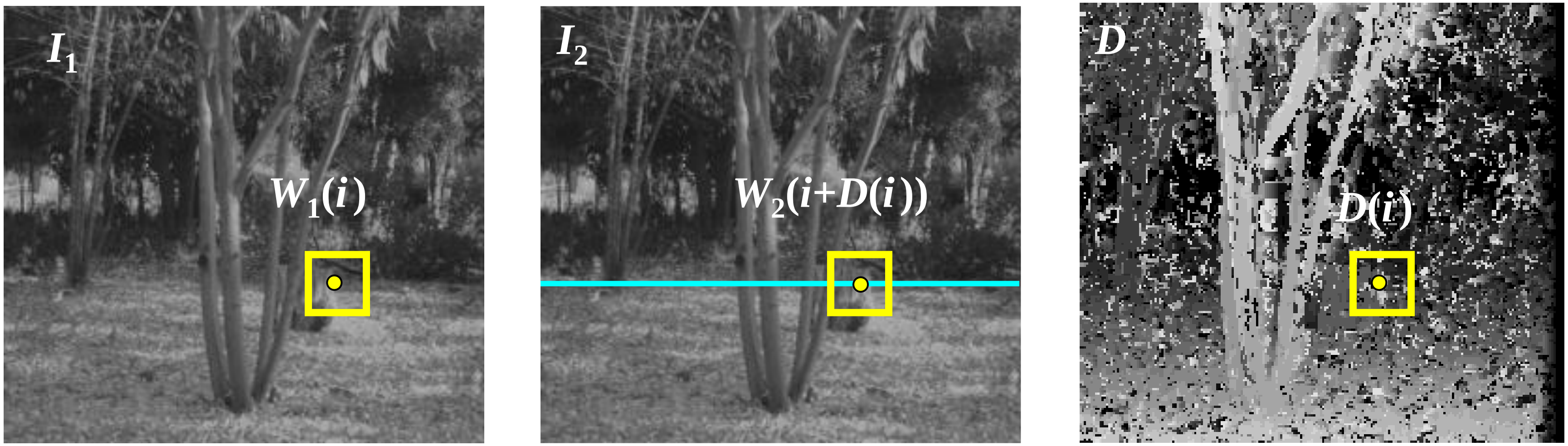


Ordering constraint doesn't hold

Stereo constraints/priors

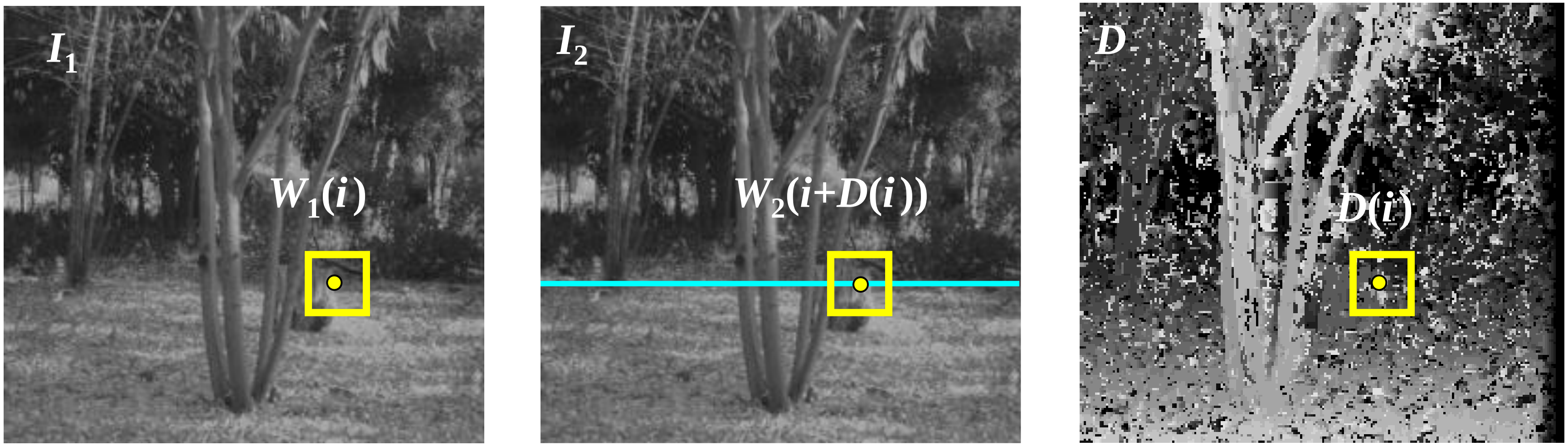
- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views
- Smoothness
  - We expect disparity values to change slowly (for the most part)

Stereo matching as energy minimization



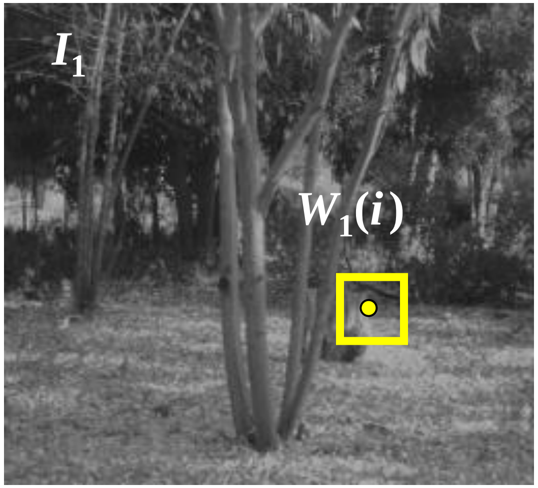
$$E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2$$

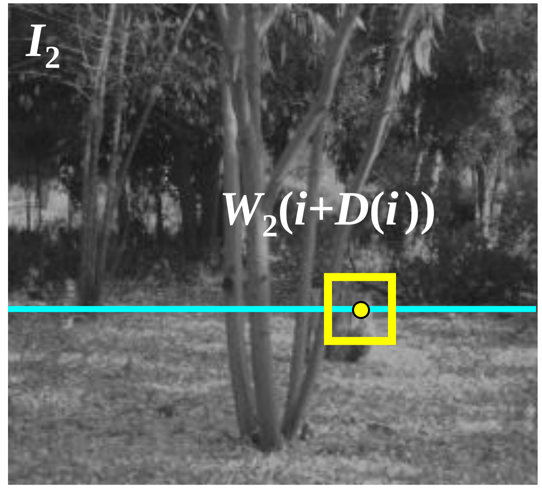
Stereo matching as energy minimization

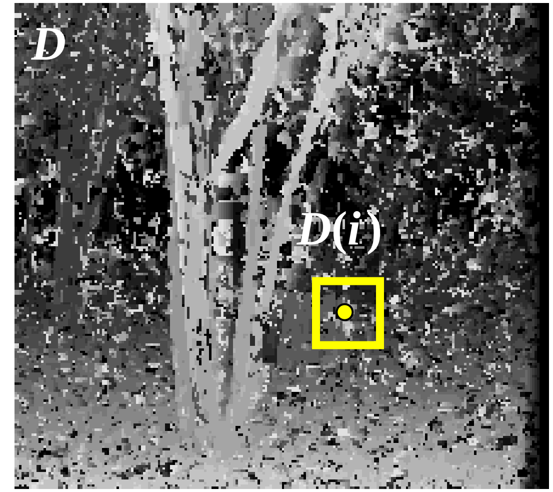


$$E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2 \qquad E_{\text{smooth}} = \sum_{\text{neighbors } i,j} \left\| D(i) - D(j) \right\|^2$$

Stereo matching as energy minimization



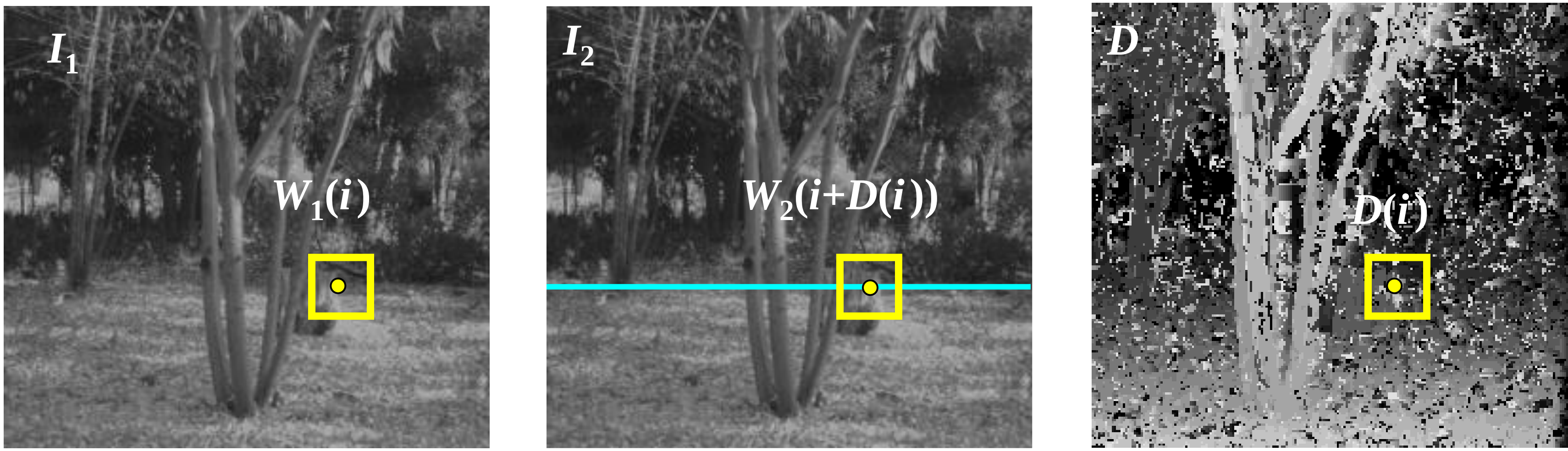




$$E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2 \qquad E_{\text{smooth}} = \sum_{\text{neighbors } i,j} \left\| D(i) - D(j) \right\|^2$$

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

Stereo matching as energy minimization



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

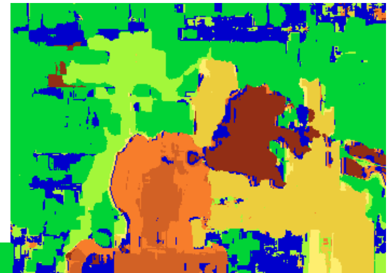
$$E = E_{\text{data}}(D; I_1, I_2) + \beta E_{\text{smooth}}(D)$$

- 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

Many of these constraints can be encoded in an energy function and solved using graph cuts

Before



Graph cuts

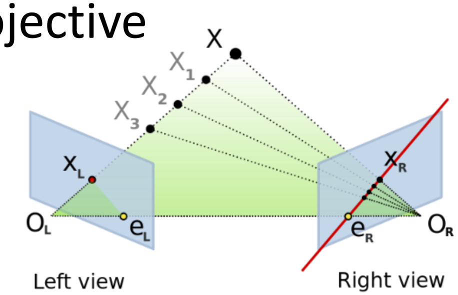
Ground truth

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

For the latest and greatest: http://www.middlebury.edu/stereo/

Things to remember

- **Epipolar geometry**
  - Epipoles are intersection of baseline with image planes
  - Matching point in second image is on a line passing through its epipole
  - Fundamental matrix maps from a point in one image to a line (its epipolar line) in the other
  - Can solve for F given corresponding points (e.g., interest points)
  - Can recover canonical camera matrices from F (with projective ambiguity)



- **Stereo depth estimation**
  - Estimate disparity by finding corresponding points along scanlines
  - Depth is inverse to disparity

Acknowledgements

- Thanks to the following researchers for making their teaching/research material online
  - Forsyth
  - Steve Seitz
  - Noah Snavely
  - J.B. Huang
  - Derek Hoiem
  - J. Hays
  - J. Johnson
  - R. Girshick
  - S. Lazebnik
  - K. Grauman
  - Antonio Torralba
  - Rob Fergus
  - Leibe
  - And many more ………..

Next class:

structure from motion