Master in Computer Vision Barcelona

Module 4: 3D Vision

Lecture 5: Epipolar geometry

Lecturer: Javier Ruiz Hidalgo

Outline

- Why is stereo useful?
  - Monocular / Binocular depth perception
- Epipolar contraints
  - Calibrated cameras: Essential matrix
  - Uncalibrated cameras: Fundamental matrix
- Estimating Fundamental matrix
  - Linear: 8 point algorithm
  - Non-linear
  - Robust methods: RANSAC
- Rectification

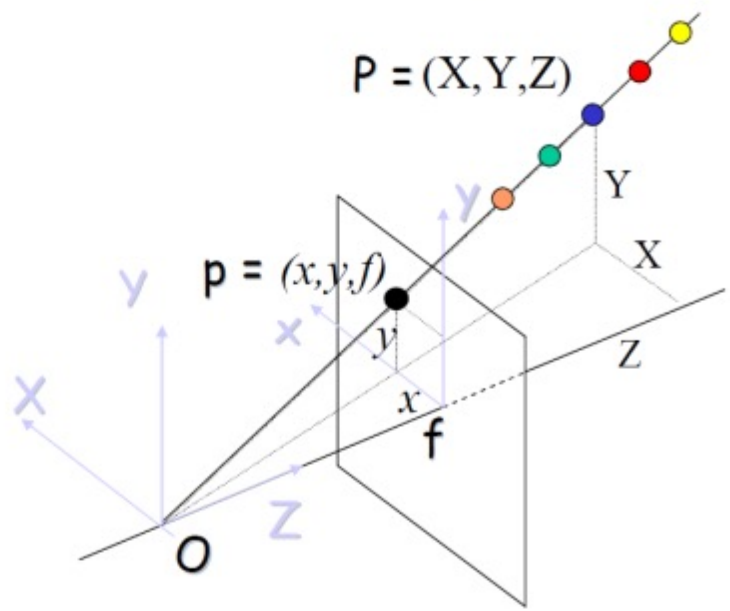
Why is stereo useful?

- **Structure and depth** are inherently **ambiguous** from single views

Source: L. Lazebnik

Why is stereo useful?

- **Fundamental ambiguity**
  - Any point ***P*** on the ray ***OP*** projects to the same point ***p*** in the image plane



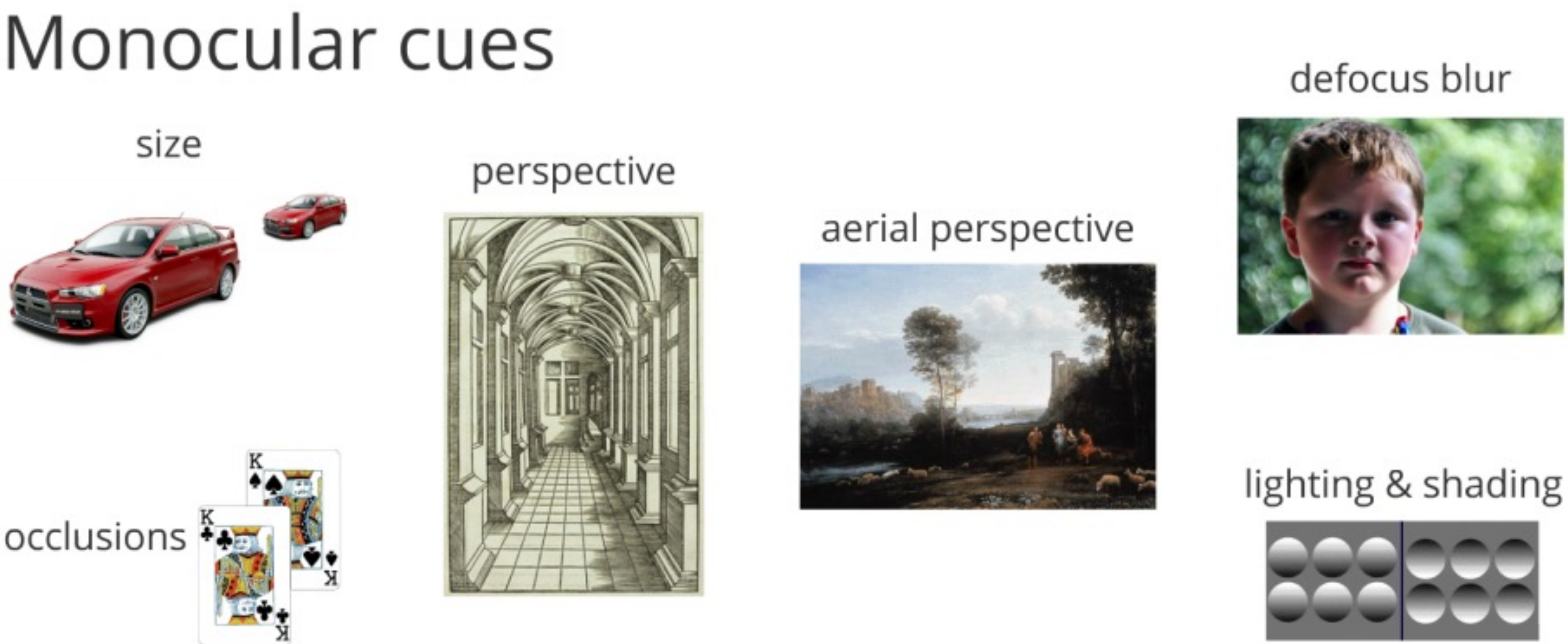
$$x = f\frac{X}{Z} = f\frac{kX}{kZ}$$

$$y = f\frac{Y}{Z} = f\frac{kY}{kZ}$$

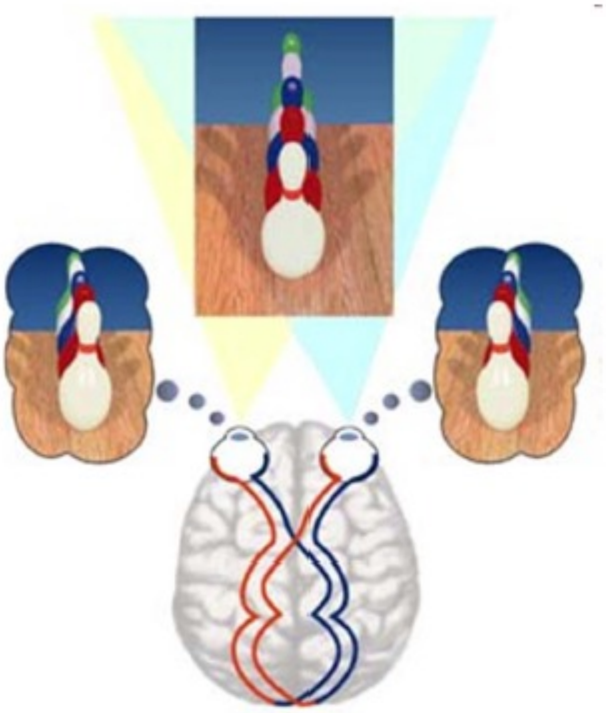
Source: R. Collins

Monocular depth perception

- What cues help us to perceive 3D structure and depth?

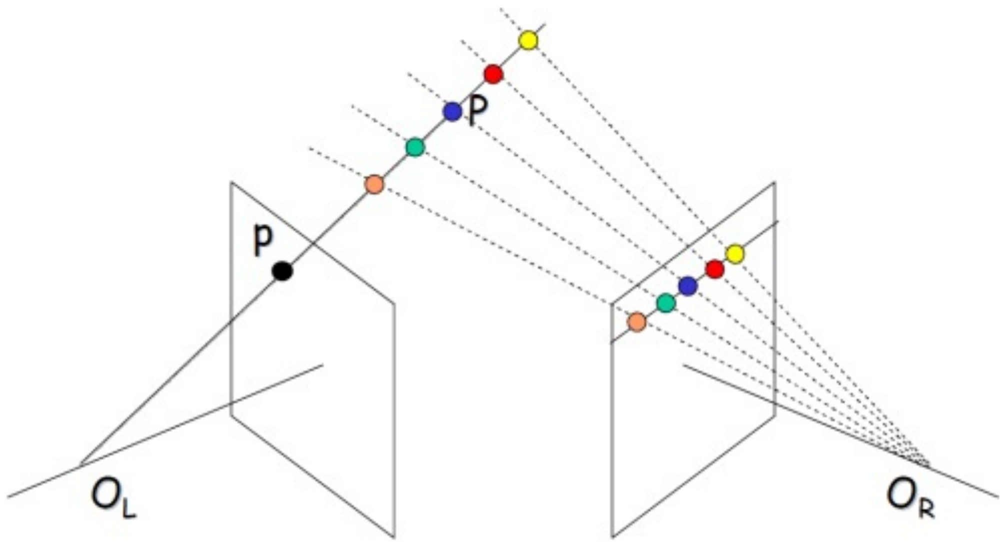


Two eyes help!



Two eyes help!

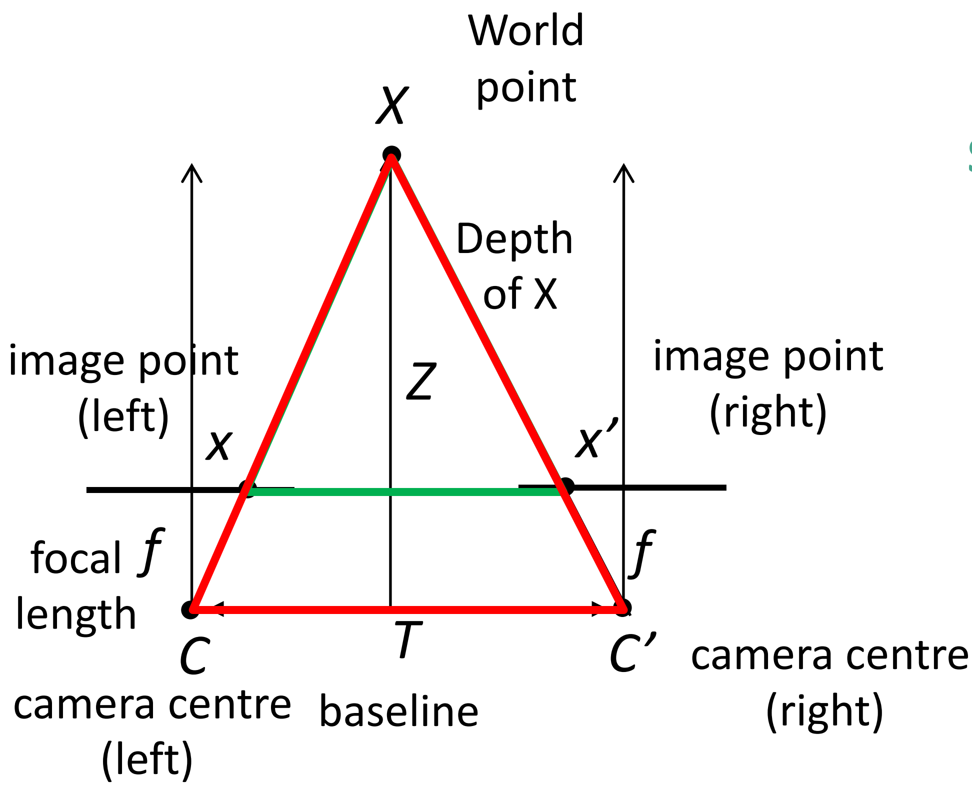
- A second camera can resolve the ambiguity, enabling measuring of depth via **triangulation**
  - Camera parameters (intrinsic & extrinsic) are known



Source: R. Collins

Geometry for a simple stereo system

- Parallel optical axes and known camera parameters



Similar triangles (x, X, x') and (C, X, C'):

$$\frac{T - x + x'}{Z - f} = \frac{T}{Z}$$

$$Z = f \frac{T}{x - x'}$$

disparity

Source: K. Grauman

Depth from disparity

image $I(x,y)$

Disparity map $D(x,y)$

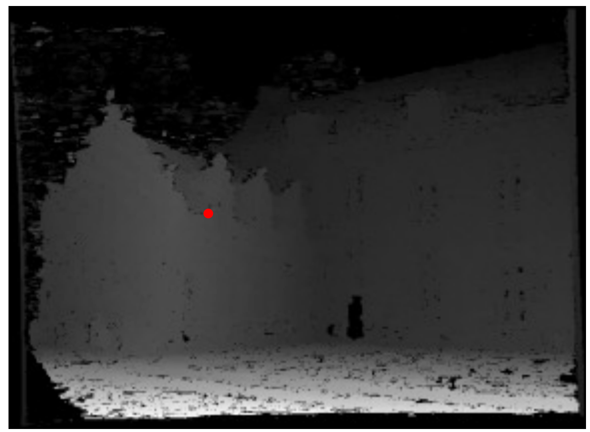


image $I'(x',y')$

$$(x',y')=(x+D(x,y), y)$$

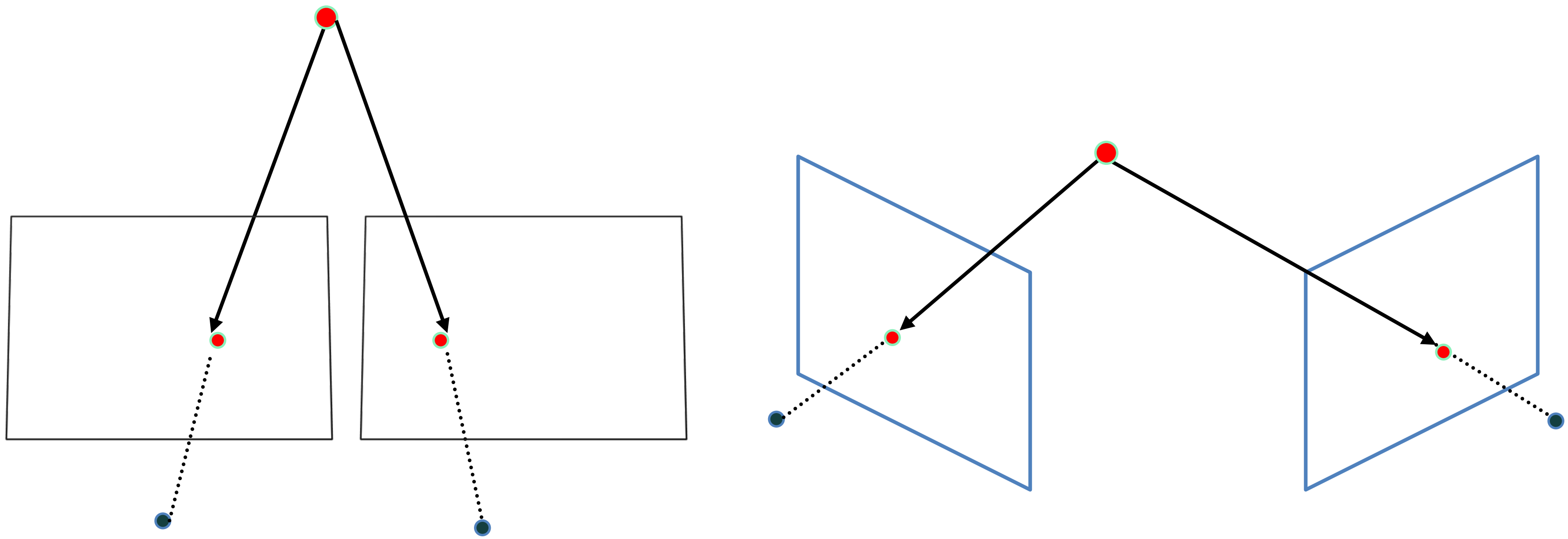
Source: K. Grauman

Outline

- Why is stereo useful?
  - Monocular / Binocular depth perception
- Epipolar contraints
  - Calibrated cameras: Essential matrix
  - Uncalibrated cameras: Fundamental matrix
- Estimating Fundamental matrix
  - Linear: 8 point algorithm
  - Non-linear
  - Robust methods: RANSAC
- Rectification

General case with calibrated cameras

- Two cameras without parallel optical axes

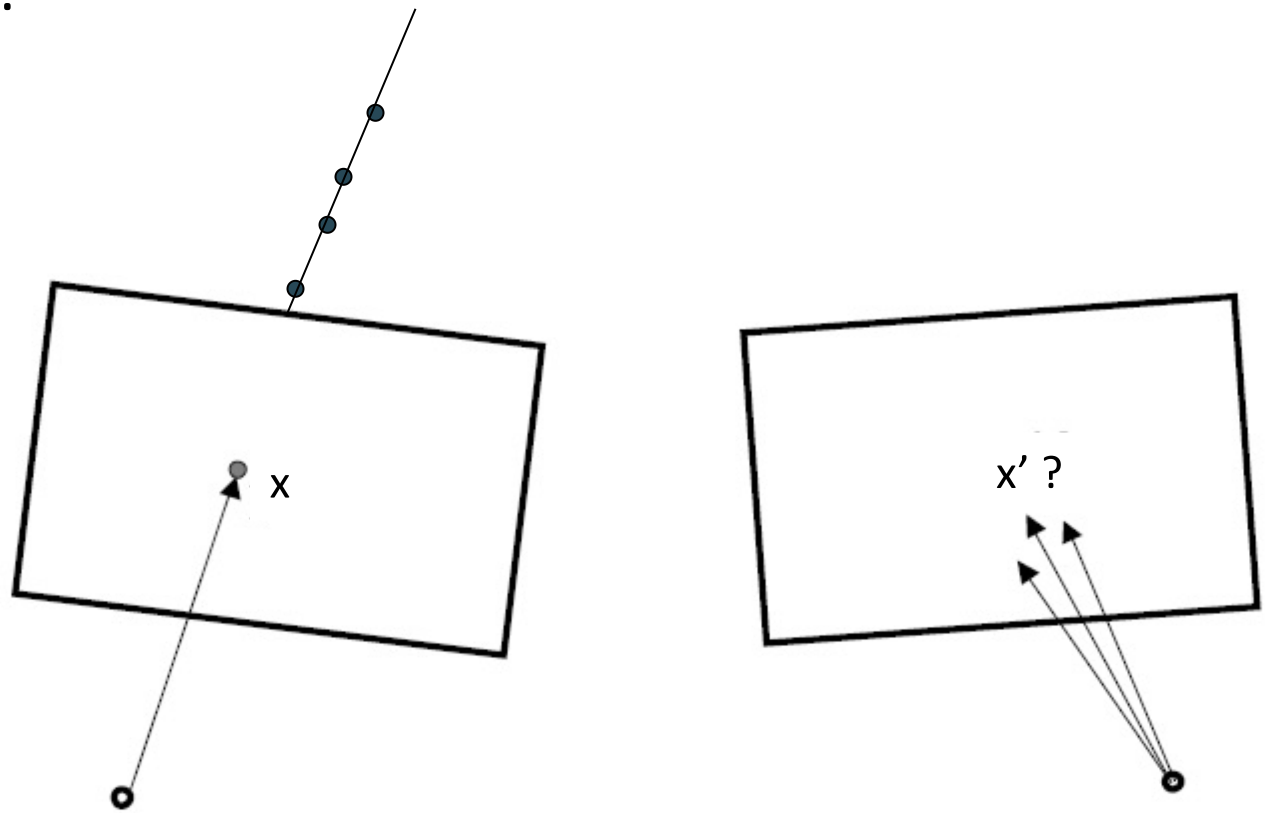


vs.

Source: K. Grauman

Stereo correspondence constraints

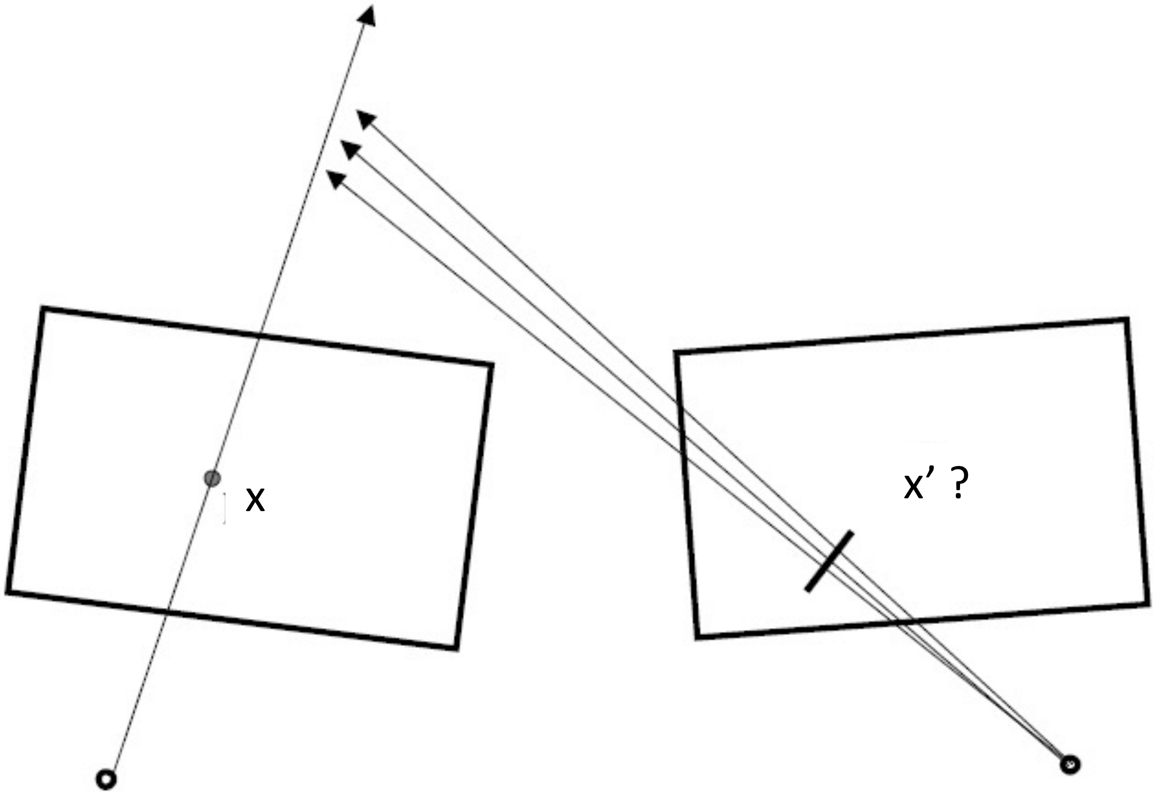
- Given ***x*** in the left image, where can the correspondence point ***x'*** be?



Source: K. Grauman

Stereo correspondence constraints

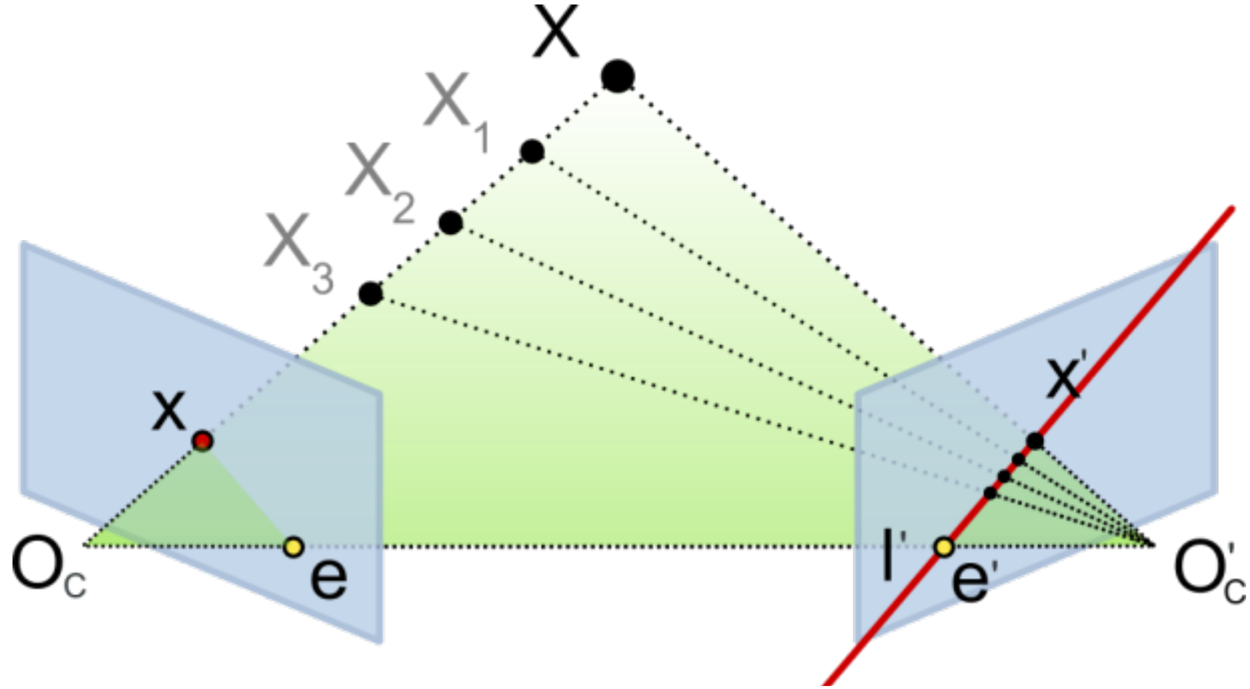
- Given ***x*** in the left image, where can the correspondence point ***x'*** be?



Source: K. Grauman

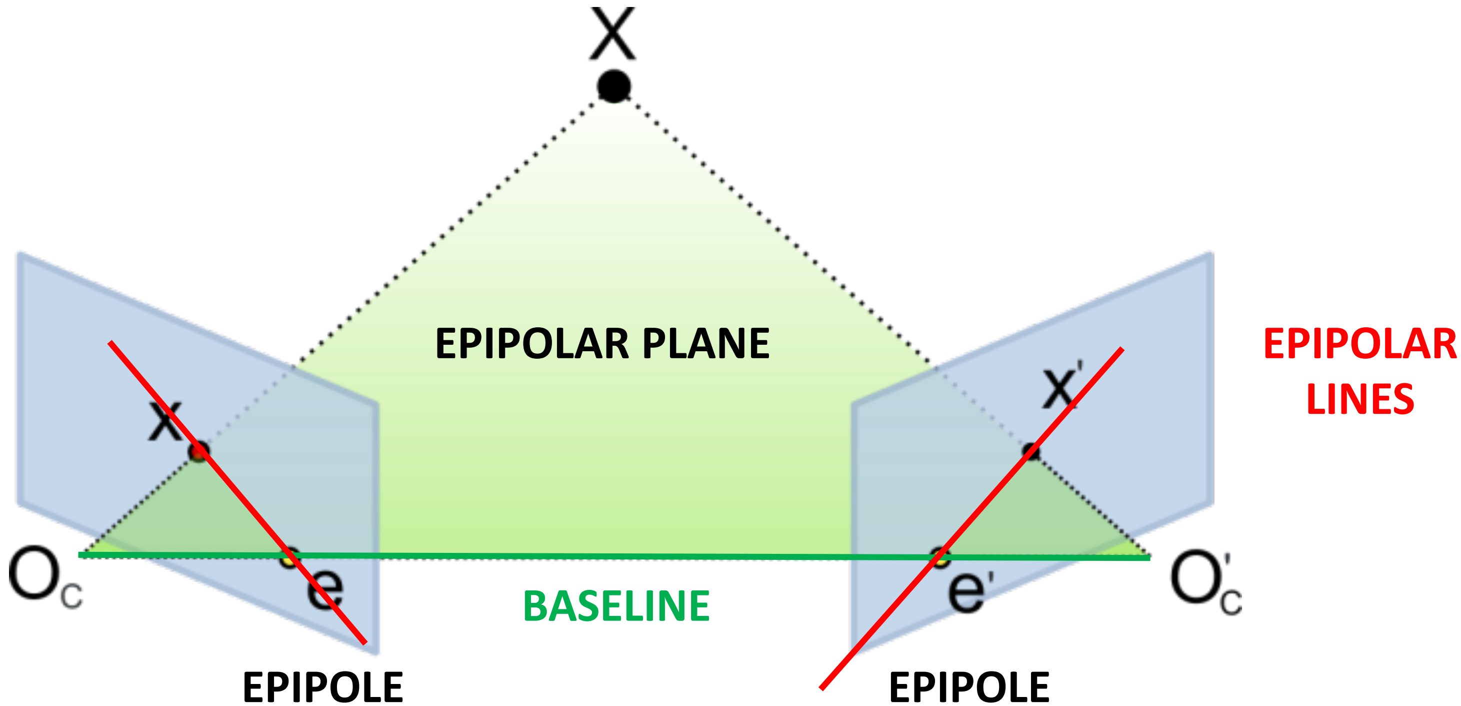
Epipolar constraint

- Geometry of two views constrains where the corresponding pixel for some image point in the left view must occur in the right view
  - It must be on the line carved out by a plane connecting the world point and optical centres



Source: K. Grauman

Epipolar geometry



http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Source: K. Grauman

Epipolar geometry terms

- **Baseline**: line joining the camera centres
- **Epipole**: point of intersection of baseline with image plane
- **Epipolar plane**: plane containing baseline and world point
- **Epipolar line**: intersection of epipolar plane with the image plane

- All epipolar lines intersect at the epipole
- An epipolar plane intersects the left and right image planes in epipolar lines

- **WHY IS THE EPIPOLAR CONSTRAINT USEFUL?**

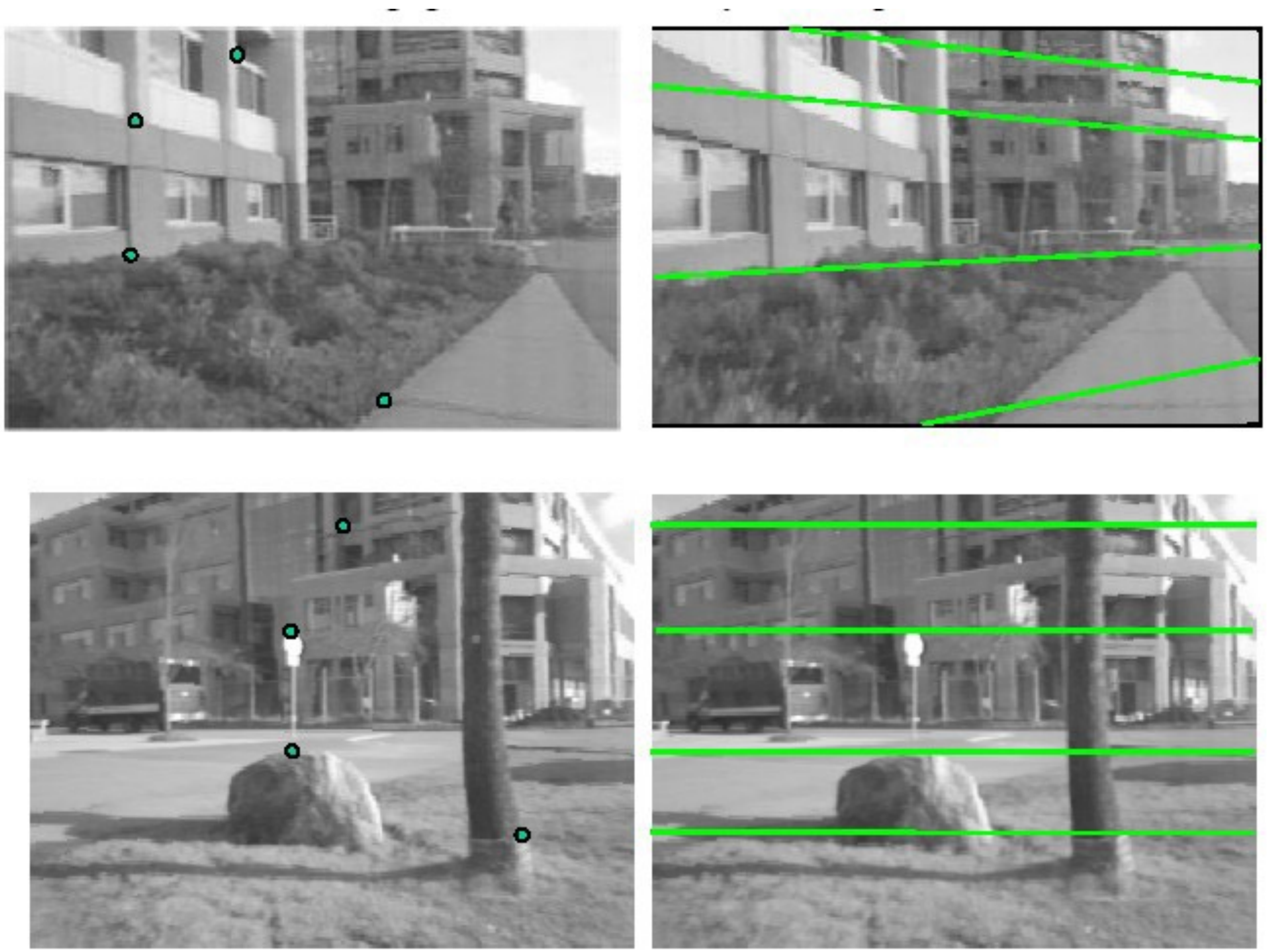
Source: K. Grauman

Epipolar constraint

- This is useful because it reduces the correspondence problem to a 1D search along an epipolar line

Source: K. Grauman

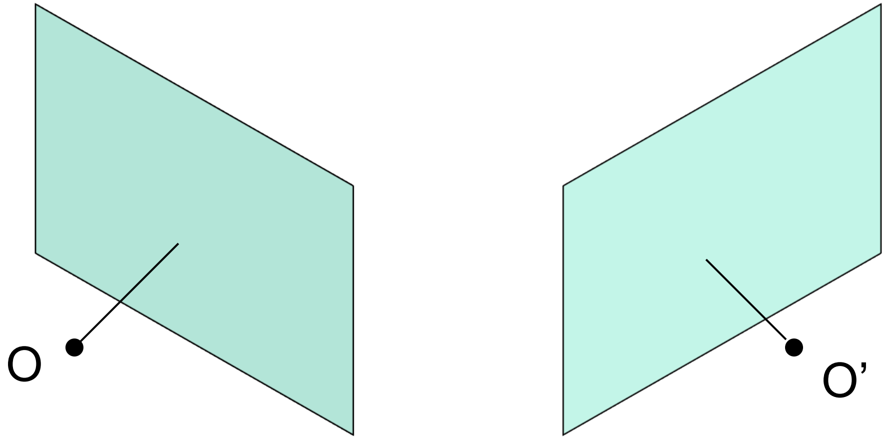
Epipolar constraint: Example



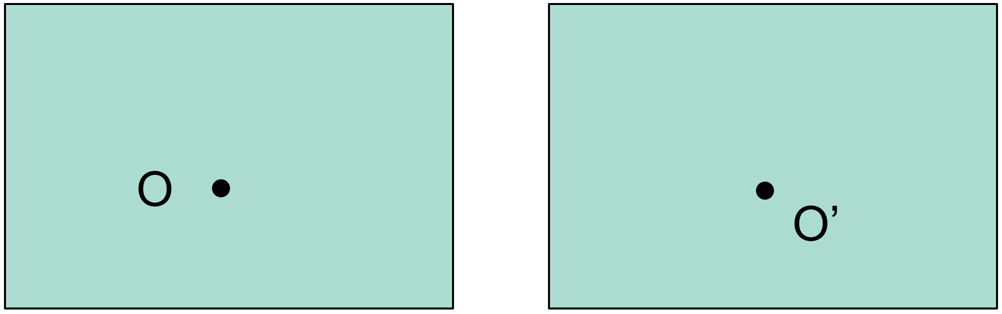
Source: K. Grauman

What do epipolar lines look like?

1.

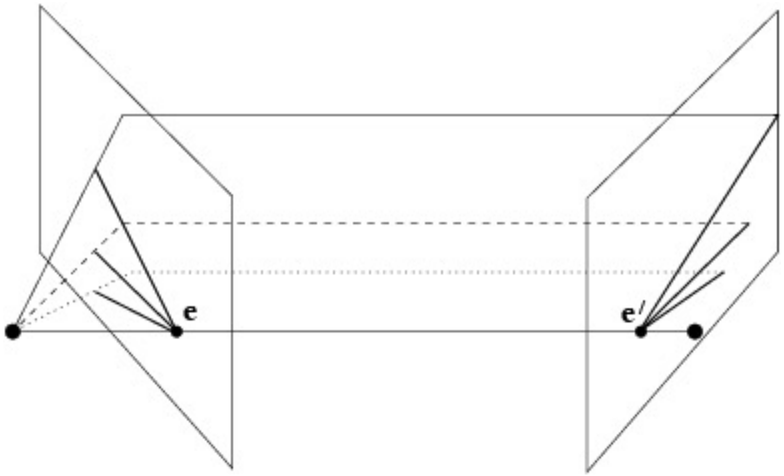


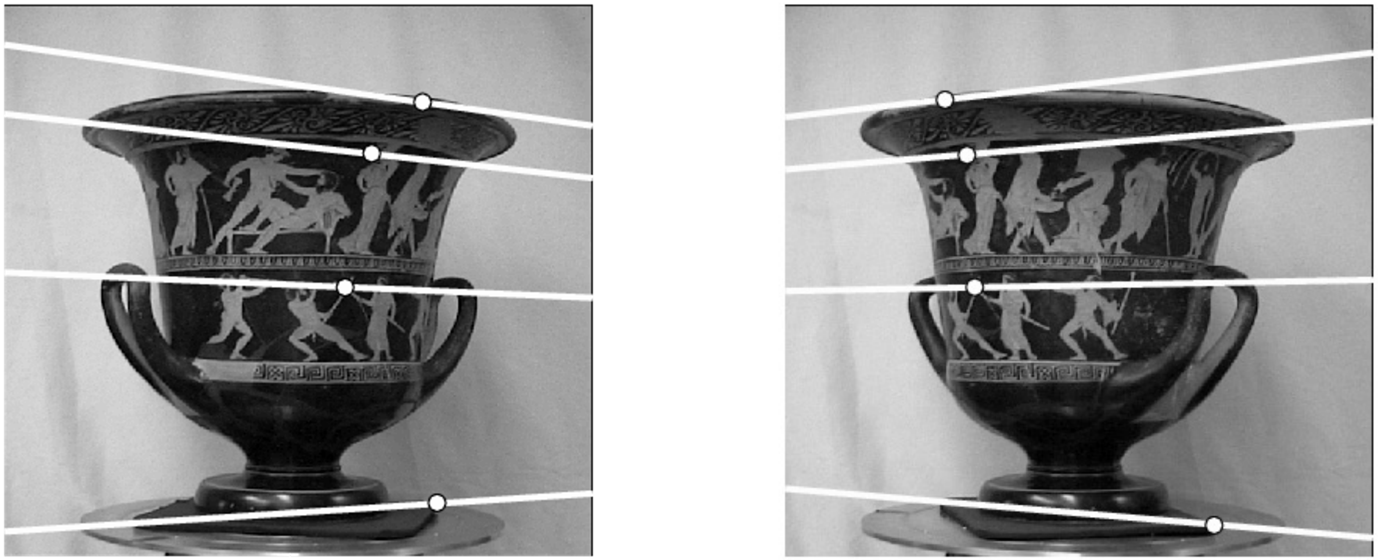
2.



Source: K. Grauman

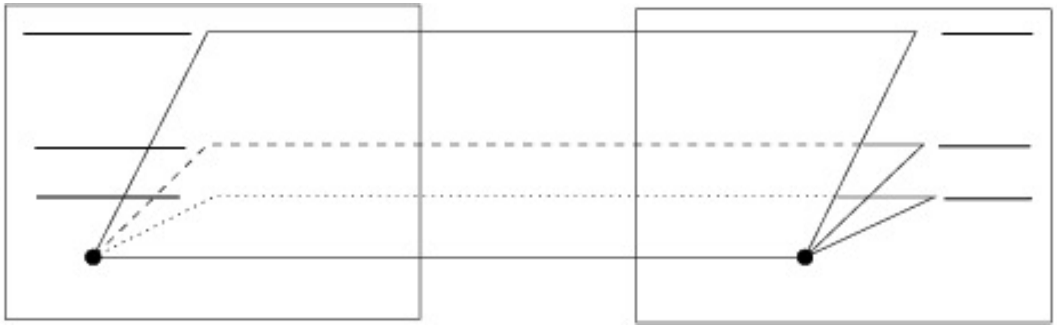
1. Converging cameras



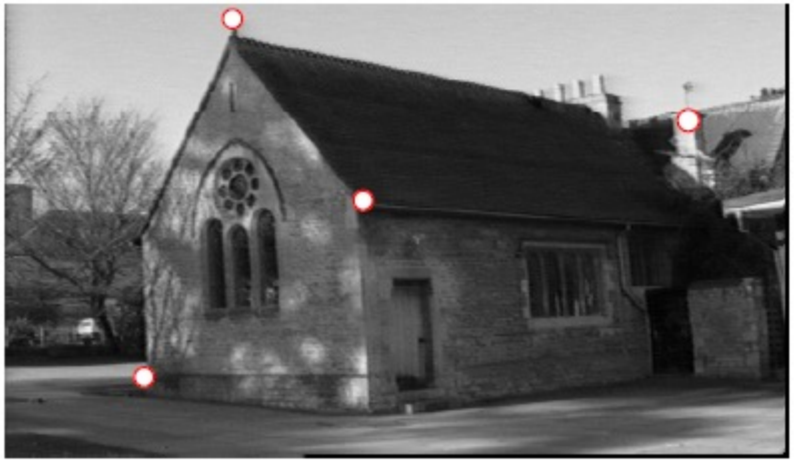


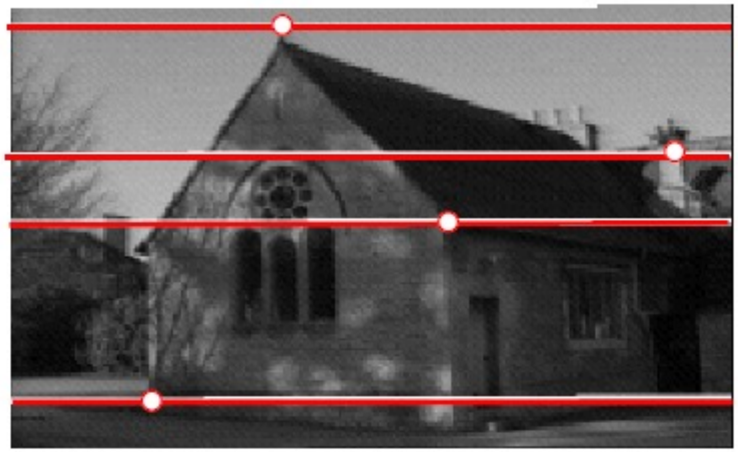
Source: K. Grauman

1. Parallel cameras



Where are the epipoles?

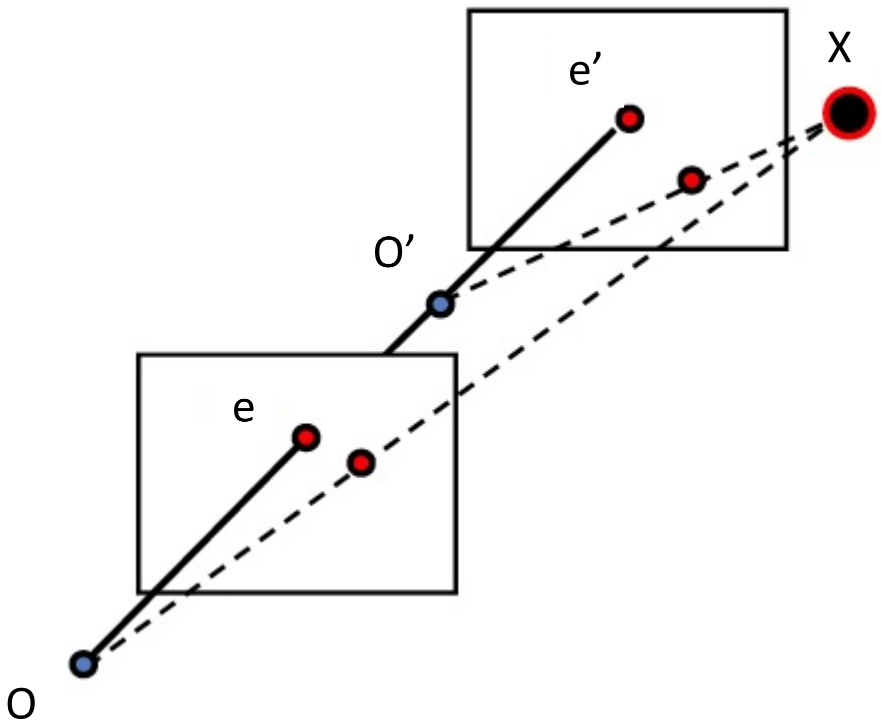




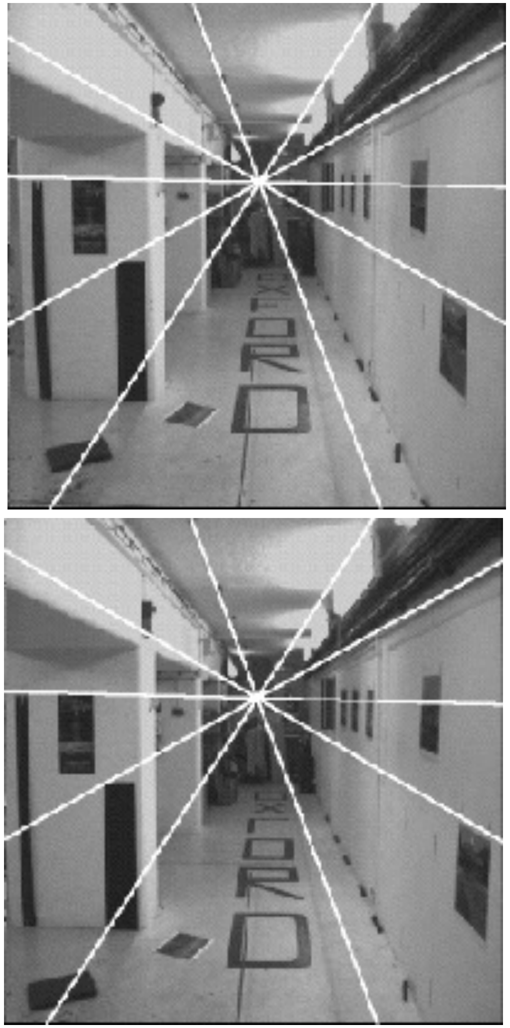
Source: K. Grauman

Another example

- Forward motion. Camera moves directly forward in the camera axes direction



- Epipoles have the same pixel position in both images
- Epipole called "Focus of expansion"

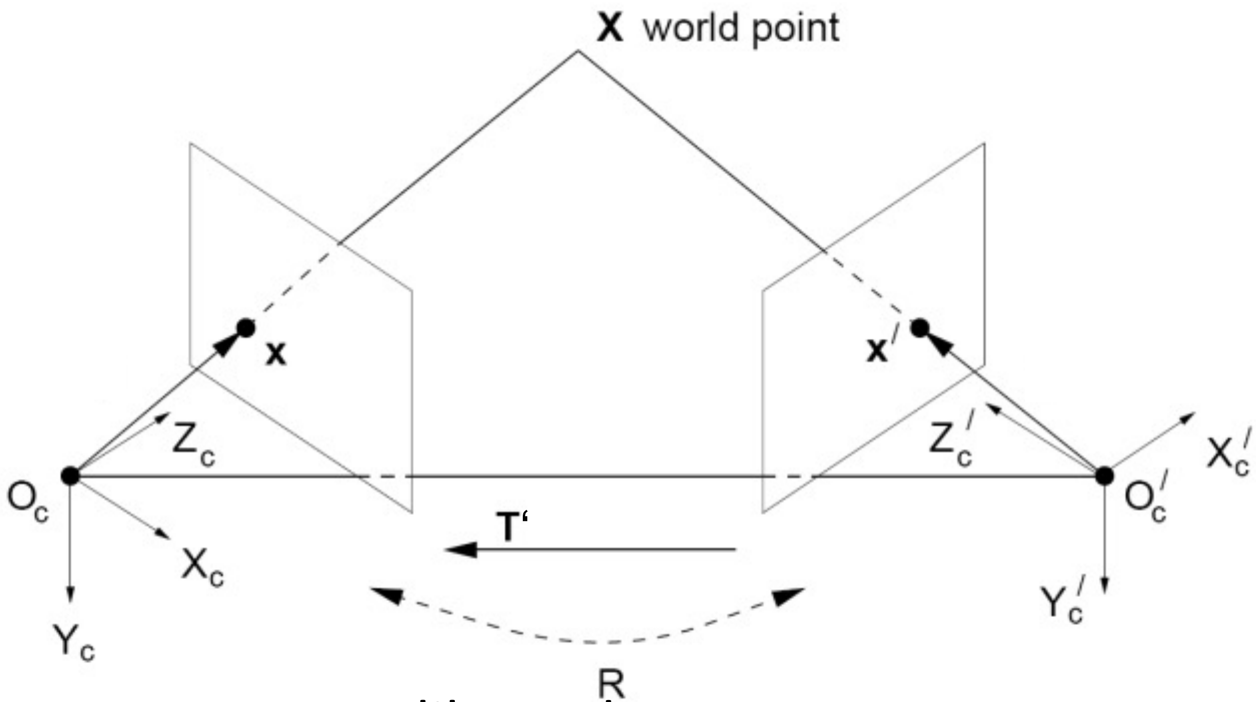


Source: K. Grauman & Fei-Fei Li

Epipolar constraint

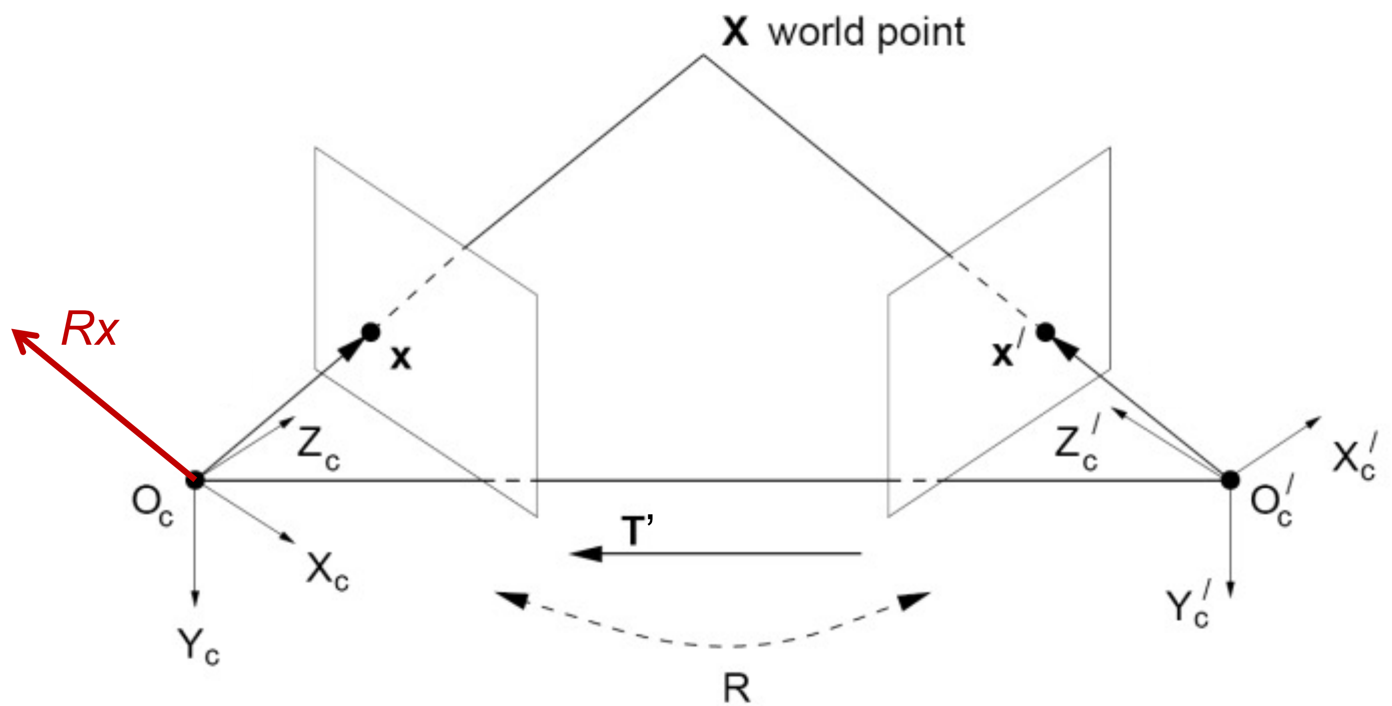
Source: K. Grauman

- Let's express this epipolar constraint algebraically



- Both cameras are calibrated
  - Know how to rotate & translate from camera 1 to camera 2 (prime)
  - Rotation: 3 x 3 matrix ***R*** ; translation: 3 vector ***T'*** $X' = RX + T'$

Epipolar constraint



- Epipolar plane is defined by vectors ***(X,T')*** or ***(x,T')*** or ***(x',T')***
- Vector ***Rx*** also included in epipolar plane
- Vector $T' \times Rx$ perpendicular to epipolar plane

$$x' \cdot (T' \times Rx) = 0$$

Source: K. Grauman

Cross product multiplication

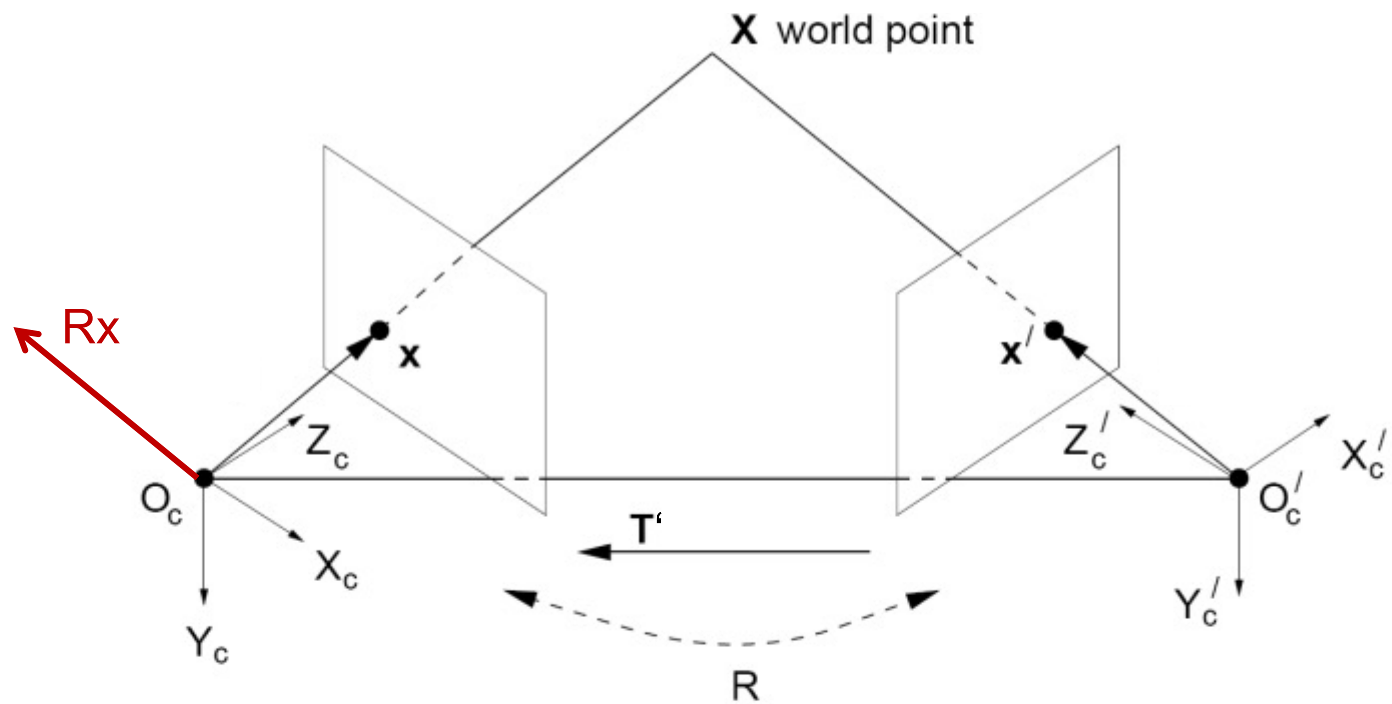
- We can express the cross product multiplication of two vectors **a** and **b** as a matrix multiplication

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

- Remember that

$$\mathbf{a}\cdot(\mathbf{a}\times\mathbf{b})=\mathbf{a}^T\left[\mathbf{a}_\times\right]\mathbf{b}=0$$
$$\mathbf{b}\cdot(\mathbf{a}\times\mathbf{b})=\mathbf{b}^T\left[\mathbf{a}_\times\right]\mathbf{b}=0$$

Epipolar constraint



- Epipolar plane is defined by vectors ***(X,T')*** or ***(x,T')*** or ***(x',T')***
- Vector ***Rx*** also included in epipolar plane
- Vector $T' \times Rx$ perpendicular to epipolar plane

Essential matrix
(Longuet-Higgins, 1981)

$$x' \cdot (T' \times Rx) = 0 \rightarrow x' \cdot ([T'_\times] Rx) = x'^T \boxed{[T'_\times] R} x = 0$$

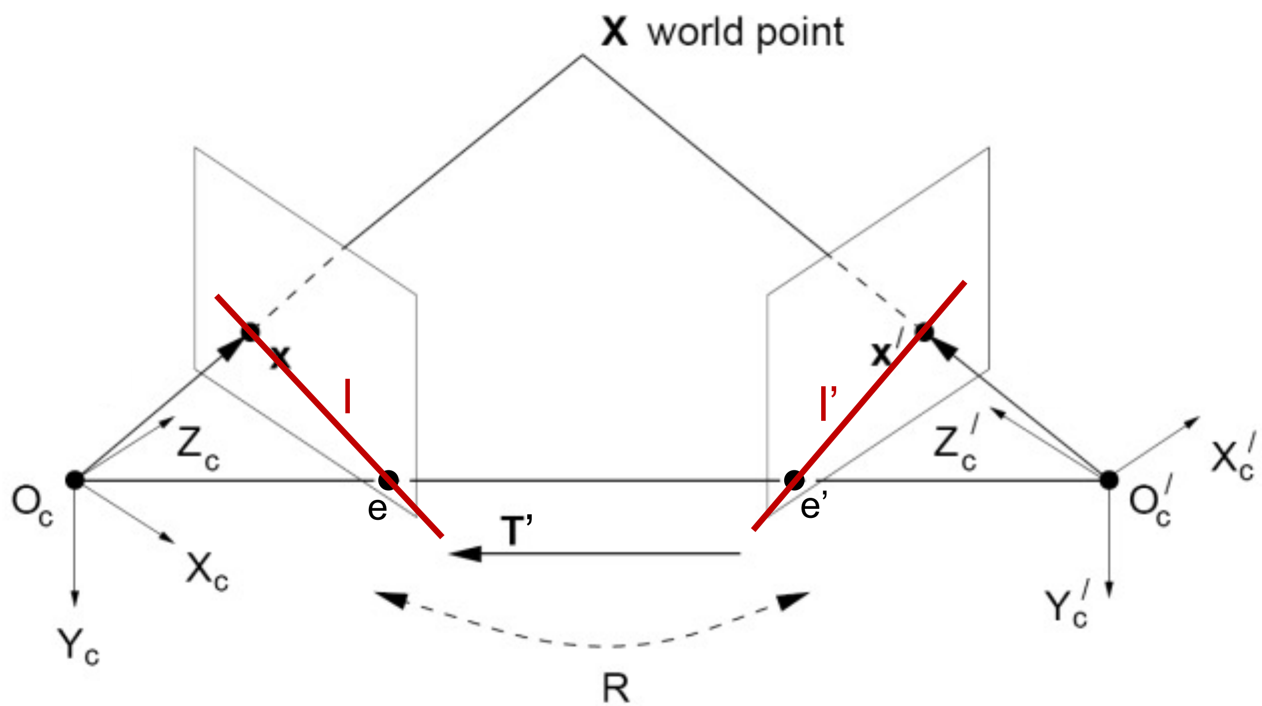
Source: K. Grauman

Essential matrix (1)

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

- ***E=[T'$_X$]R*** is called the **essential matrix**
- Relates corresponding points between both cameras given the rotation and translation
- Defined **up-to-scale**
- ***x*** and ***x'*** are in **camera coordinates system** (3 component vectors).

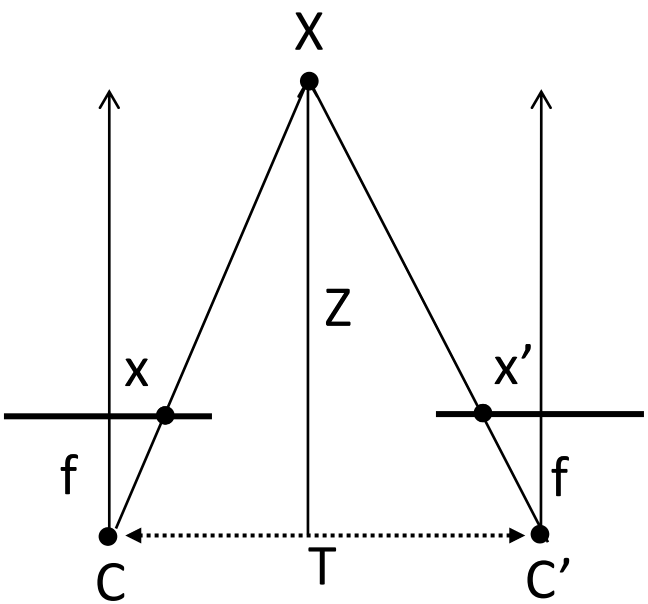
Essential matrix (2)



- Epipoles
  - ***Ee=0*** and ***E^Te'=0***
- Epipolar lines
  - ***l=E^Tx'*** and ***l'=Ex***

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

Essential matrix: Example



$$\mathbf{R} = \mathbf{I}$$

$$\mathbf{T}' = [-d, 0, 0]^{\mathrm{T}}$$

$$\mathbf{E} = [\mathbf{T}'_{\times}]\mathbf{R}$$

$$\mathbf{x} = [x, y, f]^{T}$$

$$\mathbf{x}' = [x', y', f]^{T}$$

$$\mathbf{x}'^{\mathrm{T}} E \mathbf{x} = 0$$

$$\begin{bmatrix} x' & y' & f \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & d \\ 0 & -d & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ f \end{bmatrix} = 0$$

$$\Leftrightarrow \begin{bmatrix} x' & y' & f \end{bmatrix} \begin{bmatrix} 0 \\ df \\ -dy \end{bmatrix} = 0$$

$$\boxed{\Leftrightarrow y = y'}$$

For the parallel cameras, image of any point must lie on same horizontal line in each image plane.

Essential matrix (3)

- 3x3 matrix with 5 degrees of freedom, det(***E***)=0
- ***E*** is singular (rank 2)
- Two of its singular values are equal, and the third is zero

$$E = UDV^T = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$$

Essential matrix: Extrinsic parameters (1)

- Computing extrinsic parameters from essential matrix is possible → factorizing
- Assume extrinsics of first camera are P = [I | 0] with Essential matrix:

$$E = [T'_\times] R = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$$

- E = SR determines S (and thus T’) up to scale
- Considering that ||T’|| = 1 and ST’ = 0 (S related to the cross product of T’) then:

Last column of U

$$T' = \boxed{\mathbf{u}_3}$$

  – Baseline between cameras normalized
    - Groud control points needed to recover true distances

Essential matrix: Extrinsic parameters (2)

- Sign of T' cannot be determined → 4 possible configurations depending of the sign of T' and two choices of R

$$P_1' = \left[UWV^T \mid +\mathbf{u}_3 \right] \quad P_2' = \left[UWV^T \mid -\mathbf{u}_3 \right]$$

$$P_3' = \left[UW^TV^T \mid +\mathbf{u}_3 \right] \quad P_4' = \left[UW^TV^T \mid -\mathbf{u}_3 \right]$$

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad Z = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Essential matrix: Extrinsic parameters (3)

- Geometrical interpretation:

$$P_1' = \left[UWV^T \mid +\mathbf{u}_3 \right] \quad P_2' = \left[UWV^T \mid -\mathbf{u}_3 \right]$$

$$P_3' = \left[UW^TV^T \mid +\mathbf{u}_3 \right] \quad P_4' = \left[UW^TV^T \mid -\mathbf{u}_3 \right]$$

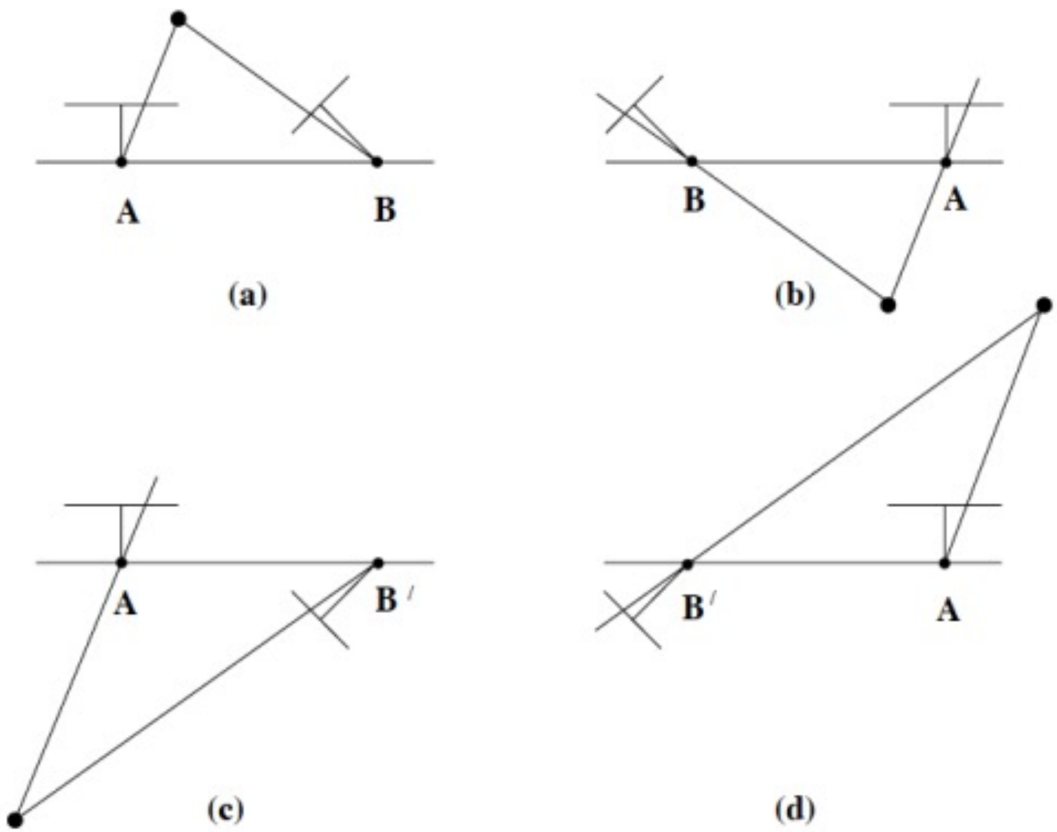
  - Direction of translation vector is reversed: $+u_3 \leftarrow\rightarrow -u_3$
  - Rotation of 180º along the baseline:

$$\left[UW^TV^T \mid u_3 \right] = \left[UWV^T \mid u_3 \right] \begin{bmatrix} VW^TW^TV^T & \\ & 1 \end{bmatrix} =$$

$$= \left[UWV^T \mid u_3 \right] \begin{bmatrix} V \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T & \\ & 1 \end{bmatrix}$$

Essential matrix: Extrinsic parameters (2)

- Up to 4 possible solutions → Only one solution (a) the reconstructed point is in front of both cameras



Source: R. Hartley & A. Zisserman

Epipolar constraint: Uncalibrated case

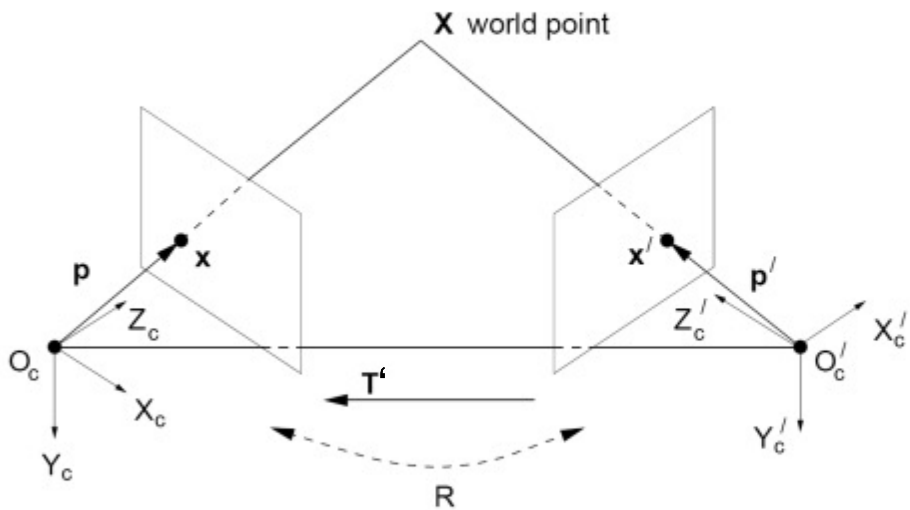
- Can we establish a similar relation between pixel coordinates $\mathbf{p=[u\ v]^T}$ instead of points in camera coordinates $\mathbf{x=[x\ y\ z]^T}$???

- Yes we can! using homogeneous coordinates
  - Points are represented as lines in a higher dimension

Epipolar constraint: Uncalibrated case

$$\tilde{p} = M\tilde{X} = K[I \quad 0]\tilde{X} = Kx \rightarrow x = K^{-1}\tilde{p}$$

$$\tilde{p}' = M'\tilde{X} = K'[R \quad T']\tilde{X} = K'x' \rightarrow x' = K'^{-1}\tilde{p}'$$



$$x'^{T} E x = \left(K'^{-1}\tilde{p}'\right)^{T} E K^{-1}\tilde{p} = \tilde{p}'^{T}\boxed{K'^{-T} E K^{-1}}\tilde{p} = \tilde{p}'^{T}\boxed{K'^{-T}\left[T'\right]_{\times} R K^{-1}}\tilde{p}$$

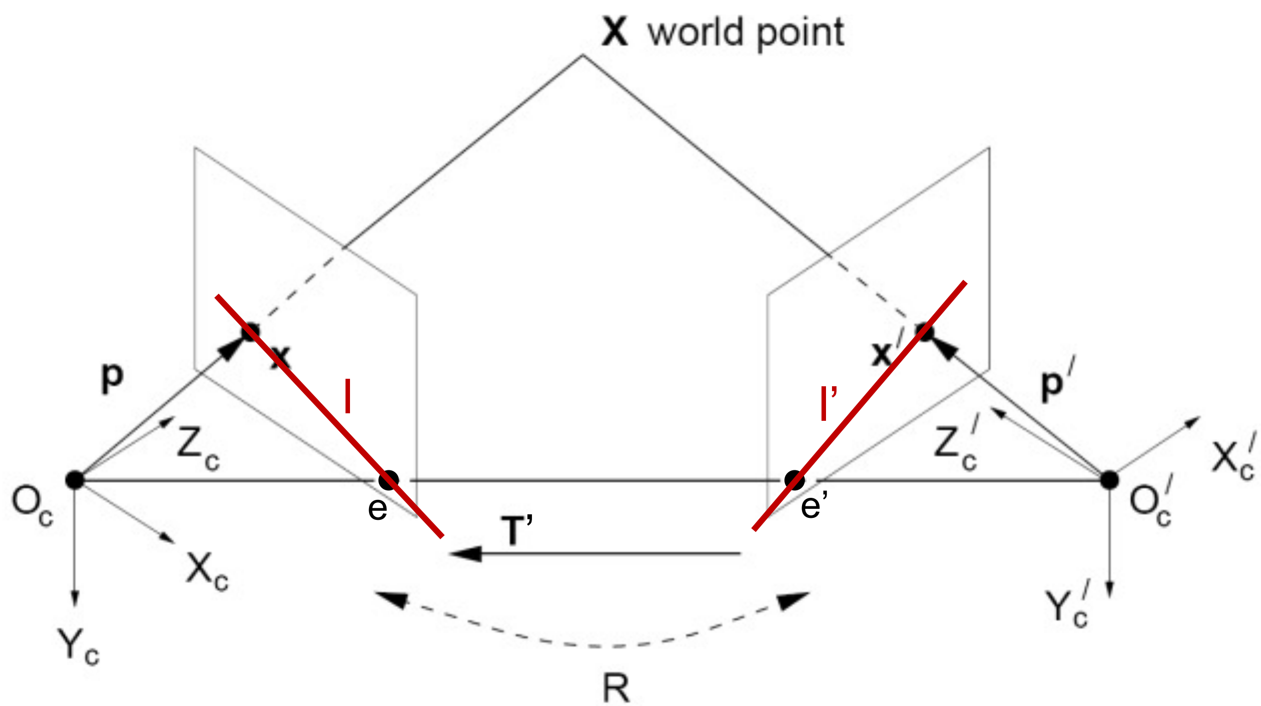
Fundamental matrix
(Faugeras and Luong, 1992)

Fundamental matrix (1)

$$\tilde{p}'^{T} F \tilde{p} = 0$$

- **$F=K'^{-T}[T'_X]RK^{-1}$** is called the **fundamental matrix**
- Relates corresponding points between uncalibrated cameras
- 3x3 matrix with 7 degrees of freedom, det(***F***)=0
- ***F*** is singular (rank 2)
- Defined **up-to-scale**
- ***p*** and ***p'*** are expressed in **pixels** (homogeneous coordinates)

Fundamental matrix (2)



- Epipoles
  - $F\tilde{e}=0$ and $F^T\tilde{e}'=0$
- Epipolar lines
  - $l=F^T\cdot\tilde{p}'$ and $l'=F\tilde{p}$

$$\tilde{p}'^T F \tilde{p} = 0$$

Fundamental matrix: Reconstruction ambiguity (1)

- Computing camera matrices from fundamental matrix
  - A pair of projection matrices ***M*** and ***M'*** uniquely determine ***F***
  - However, the fundamental matrix ***F*** determines the pair of camera matrices **up to a projective transformation**
    - **Possible solution:**

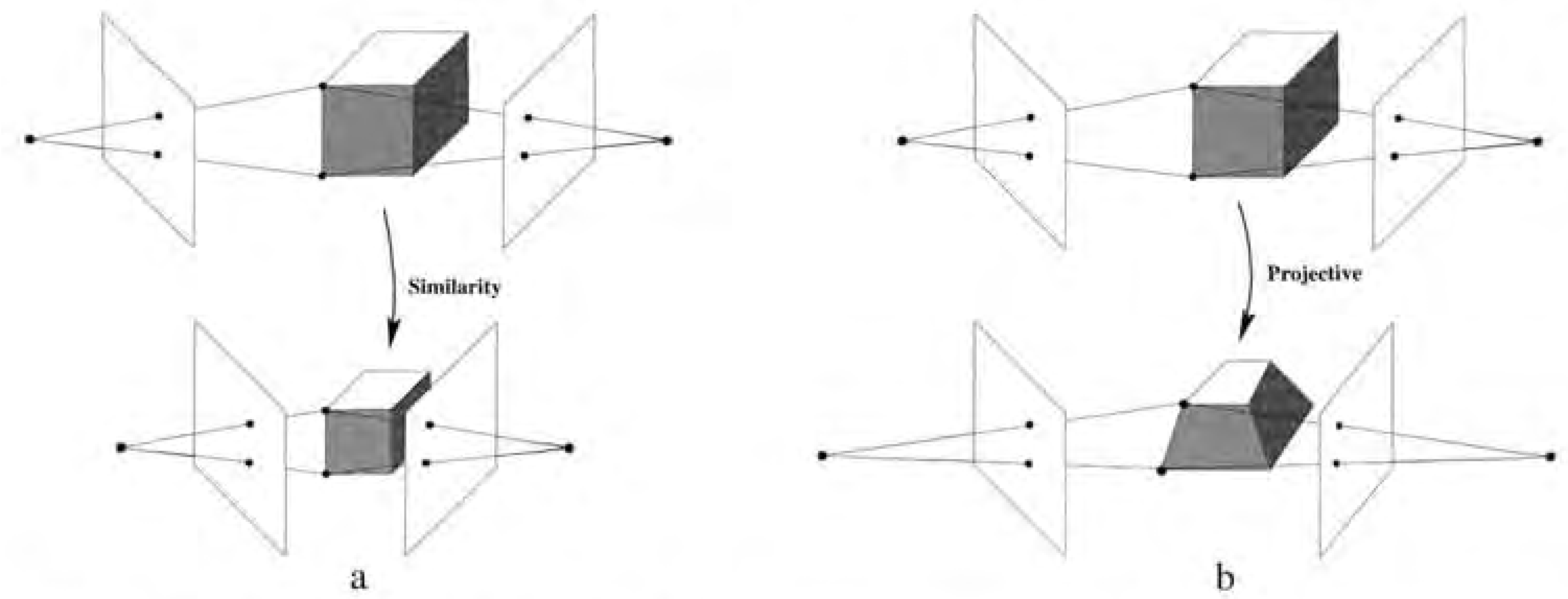
$$M = [I \mid 0] \quad M' = [SF \mid \mathbf{e}'] = [[\mathbf{e}'_x]F \mid \mathbf{e}']$$

- Taking ***H*** as a 4x4 matrix representing a projective transformation
  - Fundamental matrices corresponding to the pairs of camera matrices(***M*** ; ***M'***) and (***MH*** ; ***M'H***) are the same

$$H = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^T & s \end{bmatrix}$$

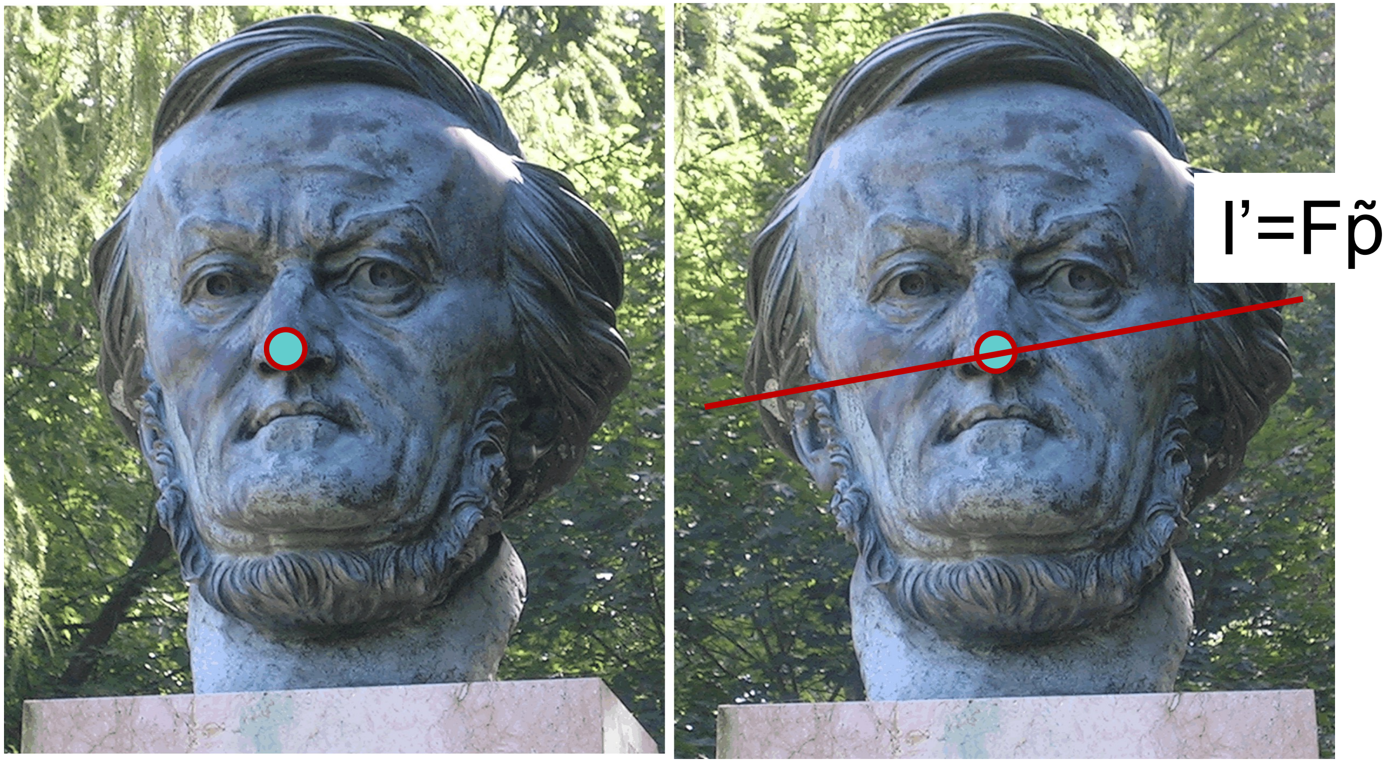
Fundamental matrix: Reconstruction ambiguity (2)

(a) Calibrated case (b) Uncalibrated case



Source: R. Hartley & A. Zisserman

Why is F useful?



- Two views of the same object
  - Suppose I know the camera positions and camera matrices
- Given a point on left image, how can I find the corresponding point on right image?

Source: Fei-Fei Li

Why is F useful?

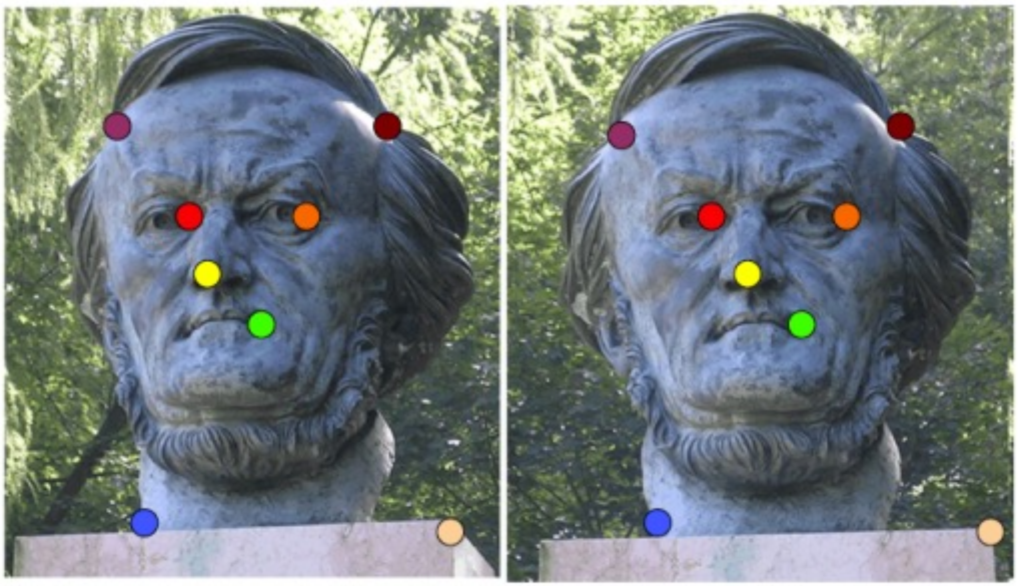
- F captures information about the epipolar geometry of 2 views + camera parameters
- **MORE IMPORTANTLY**: ***F*** gives constraints on how the scene changes under view point transformation (without reconstructing the scene!)
- Powerful tool in:
  - 3D reconstruction
  - Multi-view object/scene matching

Outline

- Why is stereo useful?
  - Monocular / Binocular depth perception
- Epipolar contraints
  - Calibrated cameras: Essential matrix
  - Uncalibrated cameras: Fundamental matrix
- Estimating Fundamental matrix
  - Linear: 8 point algorithm
  - Non-linear
  - Robust methods: RANSAC
- Rectification

Estimating F

- Estimate the fundamental matrix between two images
- Intrinsic and extrinsic parameters of the camera are unknown
- A set of points $\boldsymbol{p}_i$ and $\boldsymbol{p'}_I$ correspondences between the two images



- How to obtain these points?
  - Feature detection and correspondences

Estimating F

- For each point correspondence $\mathbf{p_i=[u\ v]^T}$ and $\mathbf{p'_i=[u'\ v']^T}$ the fundamental matrix constrains their relation to (in homogeneous coordinates):

$$\tilde{p}'^T F \tilde{p} = 0 \rightarrow \begin{bmatrix} u' & v' & 1 \end{bmatrix} \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0$$

$$\begin{bmatrix} uu' & vu' & u' & uv' & vv' & v' & u & v & 1 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0$$

8-point algorithm

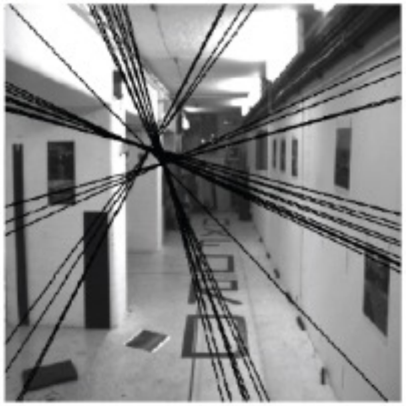
- Let's take 8 corresponding points

$$\begin{bmatrix} u_1u_1' & v_1u_1' & u_1' & u_1v_1' & v_1v_1' & v_1' & u_1 & v_1 & 1 \\ u_2u_2' & v_2u_2' & u_2' & u_2v_2' & v_2v_2' & v_2' & u_2 & v_2 & 1 \\ & & & & \vdots & & & & \\ u_8u_8' & v_8u_8' & u_8' & u_8v_8' & v_8v_8' & v_8' & u_8 & v_8 & 1 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0 \rightarrow \mathbf{Wf} = 0$$

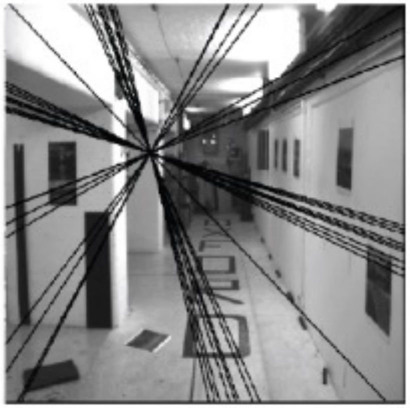
- Linear least square method
  - 0 is always a solution
  - Non-zero solution
    - Minimize $||\mathbf{Wf}||^2$ under the constraint $||\mathbf{f}||=1$

8-point algorithm

- Use singular value decomposition (SVD)
  - **W** = UDV^T where U and V are orthogonal matrices and D diagonal matrix with non-negative entries (descending order).
  - Last column of V gives **f** → ***F***
- However, ***F*** is still of rank 3
  - Epipolar lines will not coincide in the epipole!
  - Force ***F*** to be of rank 2



Rank 3



Rank 2

8-point algorithm

- Use singular value decomposition (**AGAIN!**) to construct ***F*** of rank 2

$$F_{RANK\_3} = U\begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix}V^T$$

$$F_{RANK\_2} = U\begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{bmatrix}V^T$$

8-point algorithm: Steps

1. Create matrix **W** from p_i and p'_i correspondences
2. Compute the SVD of matrix **W**$=UDV^T$
3. Create vector **f** from last column of V
4. Compose fundamental matrix ***F***_rank3
5. Compute the SVD of fundamental matrix ***F***_rank3 $= UDV^T$
6. Remove last singular value of D to create Ď
7. Re-compute matrix ***F*** $= U \check{D} V^T$ (rank 2)

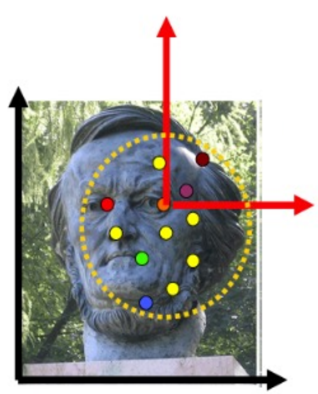
R. I. Hartley, ***In Defence of the 8-point Algorithm***, IEEE transaction on pattern analysis and machine intelligence, Vol. 19, No. 6, June 1997

The Fundamental matrix song

http://danielwedge.com/fmatrix/

8-point algorithm

- Least-square methods are
  - Very **sensitive** to **noise**
    - Usually more points (than 8) are used
  - Very sensitive to **false matches** (false correspondences)
    - Ransac
  - **Highly unstable** (in the numerical sense)
    - Normalize Input Data



- **IDEA**: Transform the image coordinate system to normalize input data before estimating ***F*** (Hartley,1995)

Robust methods

- Exploit heuristics
- Try to remove "outliers"
- **RAN**dom **SA**mple **C**onsensus
  - RANSAC loop (Iterate the procedure K times):
    - Randomly select 8 points
    - Estimate ***F*** from selected points
    - Determine # inliers where $d\left(p_i, F^T p_i'\right) + d\left(p_i', F p_i\right) < \varepsilon$
  - Re-estimate ***F*** using all inliers

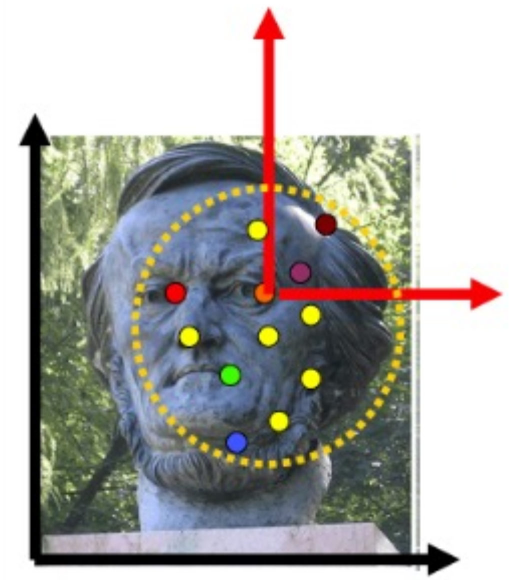
Normalized 8-point algorithm

1. Compute ***H*** and ***H'*** so the new origin is the centroid of points and scale it so the mean square distance between the origin and the data points is 2 pixels

$$\tilde{q}_i = H\tilde{p}_i \qquad \tilde{q}'_i = H'\tilde{p}'_i$$

2. Use the 8-point algorithm to estimate $\boldsymbol{F_q}$ from points $\boldsymbol{q_i}$ and $\boldsymbol{q_i'}$

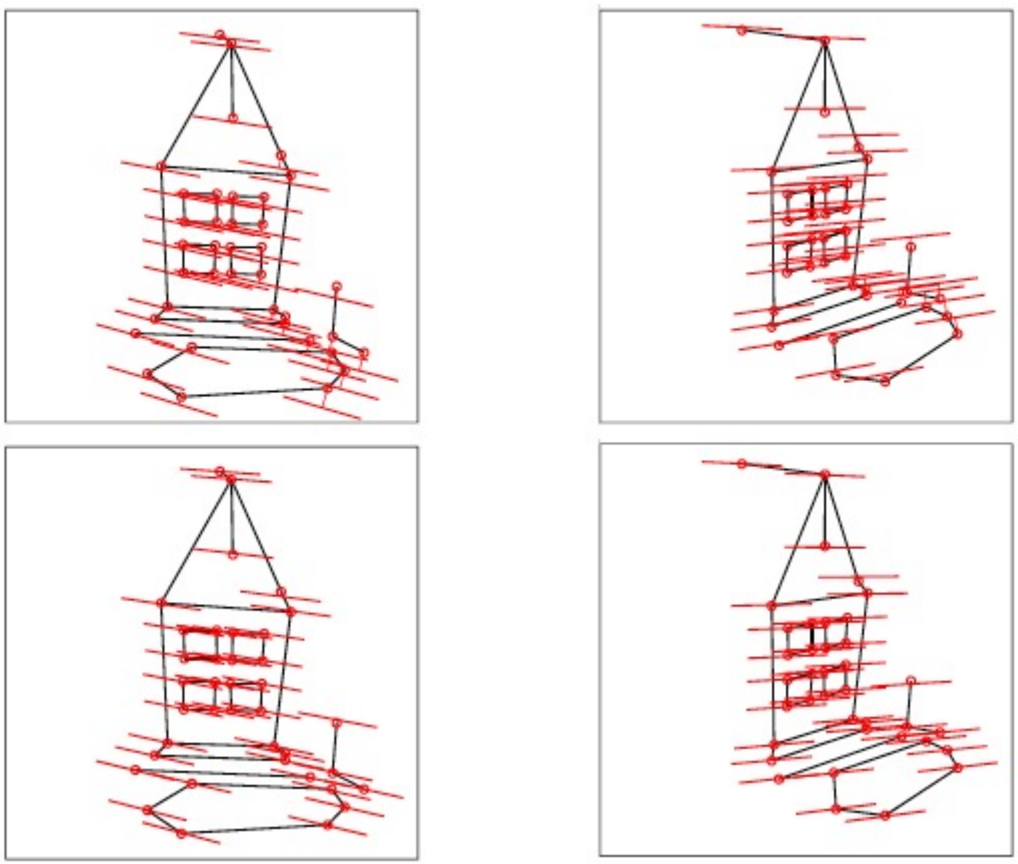
3. De-normalize $\boldsymbol{F_q}$ → $F = H'^T F_q H$



Example

Without transformation

With transformation



Mean errors:
10.0pixel
9.1pixel

Mean errors:
1.0pixel
0.9pixel

Non-linear methods

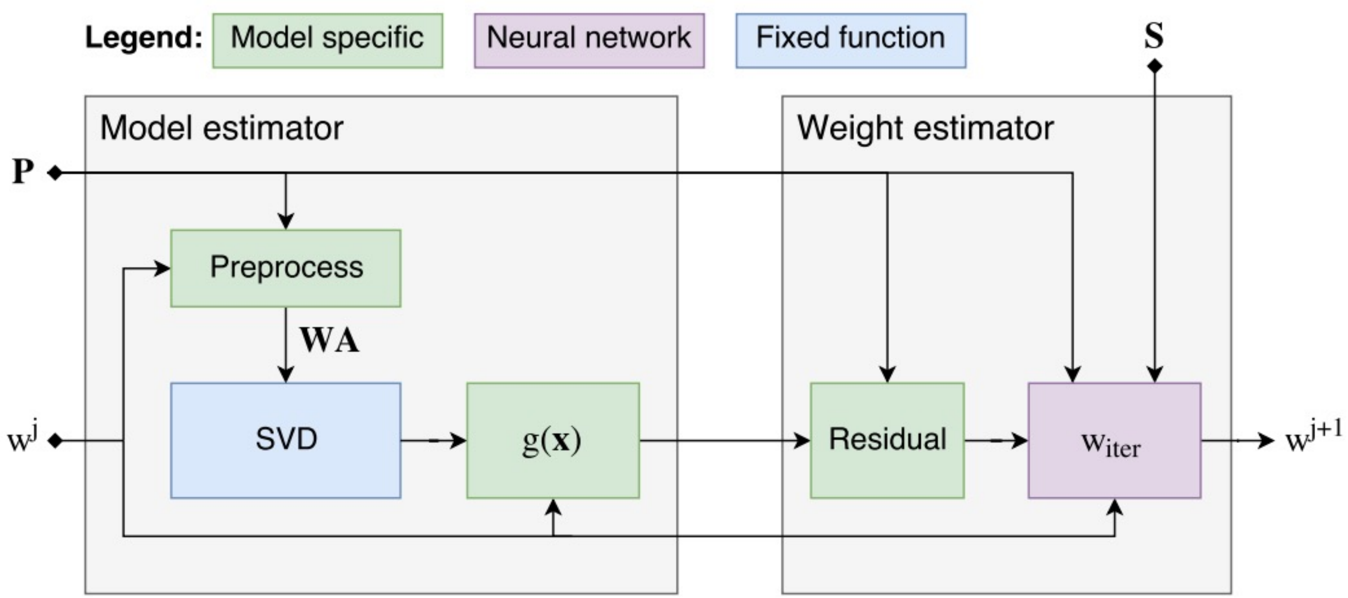
- Non-Linear Least-Squares Approach (Luong et al., 1993)
  - Minimize with respect to the coefficients of F , using an appropriate rank 2 parameterization

$$\sum_{i=1}^{N}\left[d^2\left(p_i, F^T p_i'\right) + d^2\left(p_i', F p_i\right)\right]$$

  - Still problems with false correspondences

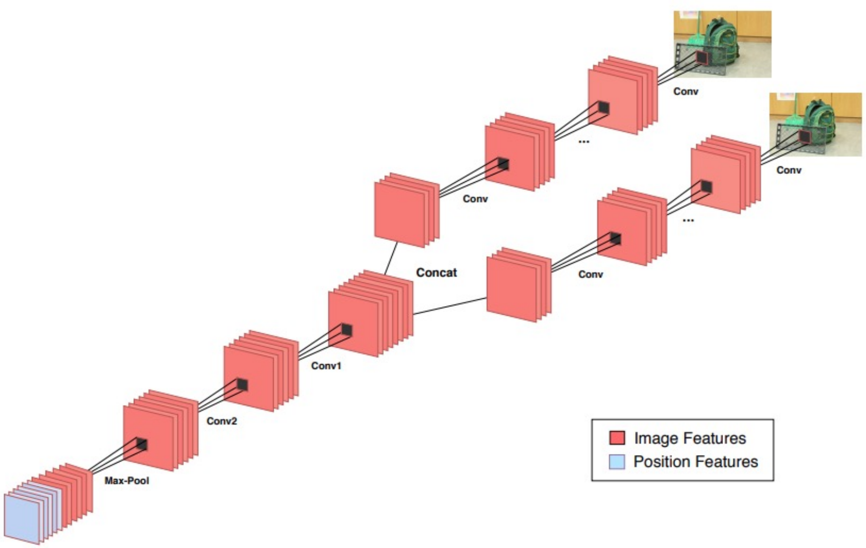
Learning methods (1)

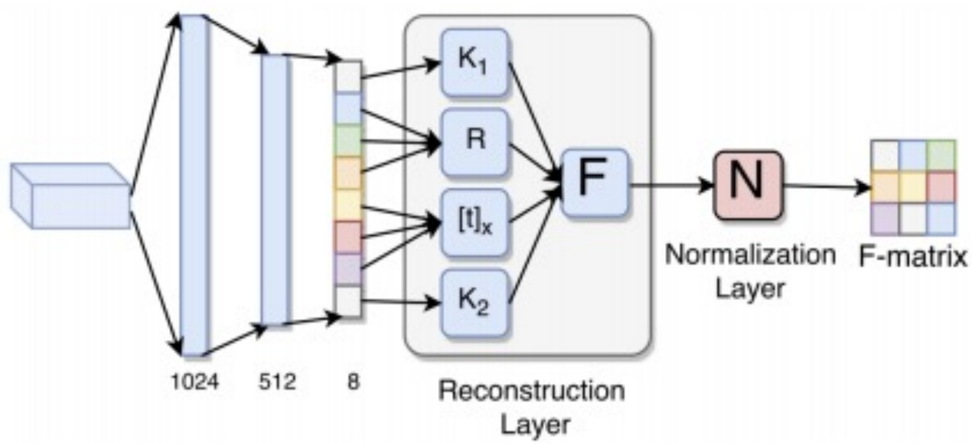
- R. Ranftl and V. Koltun, [Deep Fundamental Matrix Estimation](), ECCV, 2018
  - Uses neural network to refine weights for least-squares
  - Automatic detection of outliers



Learning methods (2)

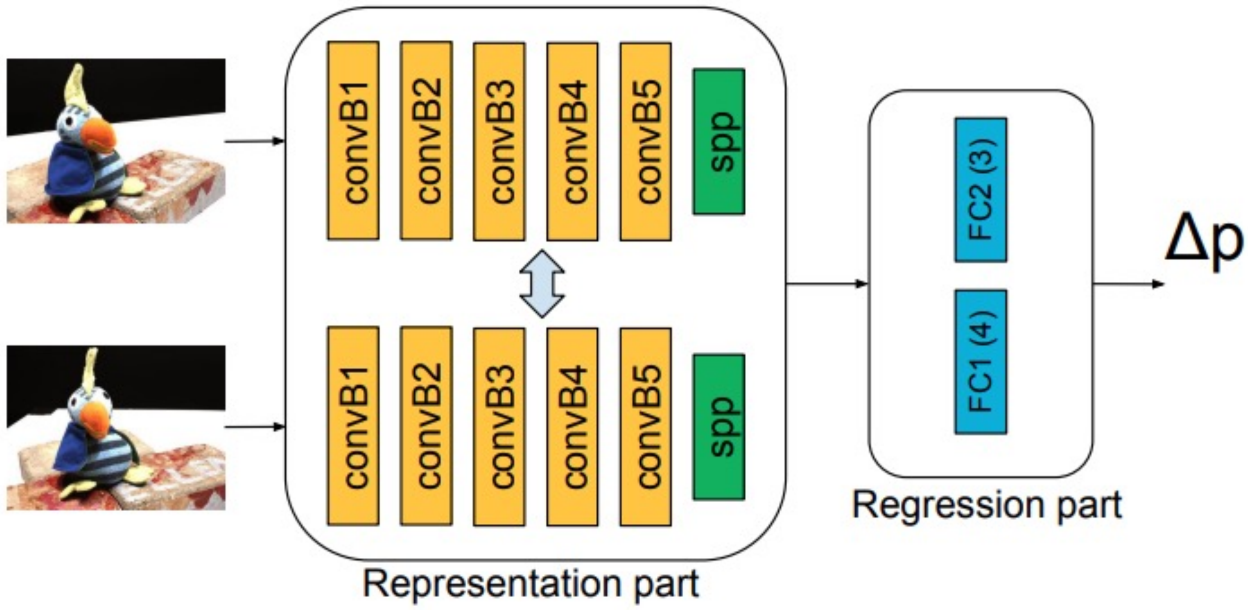
- O. Poursaeed, G. Yang, et.al. Deep Fundamental Matrix Estimation without Correspondences, 2018.
  - Directly estimate Fundamental matrix from two images using a siamese network and convolutional layers
- Reconstruction layer to force properties of **F**





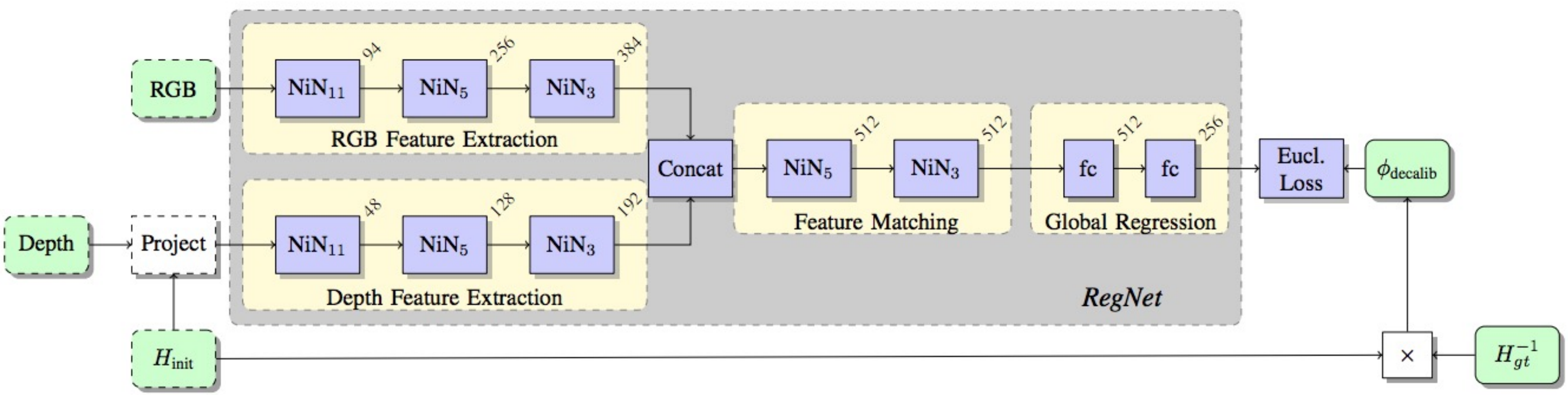
Learning methods (3)

- . Melekhov, et.al. Relative Camera Pose Estimation Using Convolutional Neural Networks, ACIVS, 2017.
  - Estimate relative rotation and translation between two images



Learning methods (4)

- N. Schneider, F. Piewak, C. Stiller, U. FrankeR, RegNet: Multimodal Sensor Registration Using Deep Neural Networks, IEEE Intelligent Vehicles Symposium, 2017
  - Infers extrinsic calibration between multimodal sensors
  - LiDAR and a RGB monocular camera



Outline

- Why is stereo useful?
  - Monocular / Binocular depth perception
- Epipolar contraints
  - Calibrated cameras: Essential matrix
  - Uncalibrated cameras: Fundamental matrix
- Estimating Fundamental matrix
  - Linear: 8 point algorithm
  - Non-linear
  - Robust methods: RANSAC
- Rectification

Image rectification (1)

- Transformation of two or more images into a common image plane

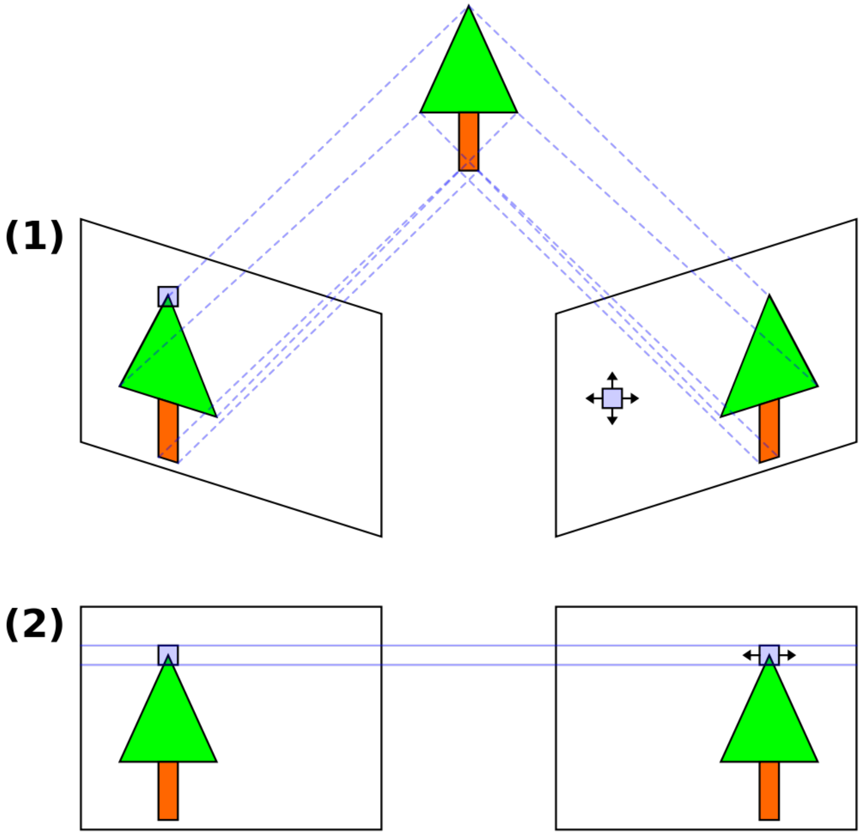
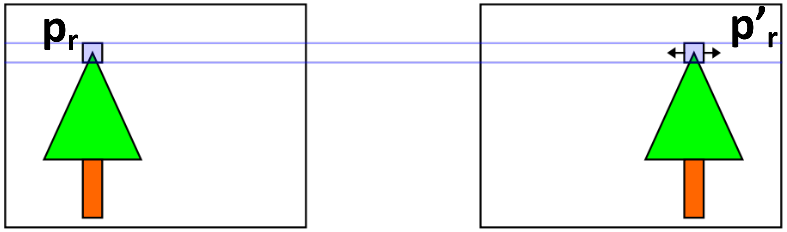


Image rectification (2)

- 3D reconstruction is a solution but …
  - Applying 2D projective transforms, or homographies, to each image can also solve the problem (easier)
    - C. Loop and Z. Zhang, *Computing Rectifying Homographies for Stereo Vision,* Technical Report, Microsoft Research, 1998.

- Finding homographies to transform images so:
  - Epipolar lines map to horizontally aligned lines in the transformed images (parallel to X axis)
  - Epipoles are at infinity

Image rectification (3)



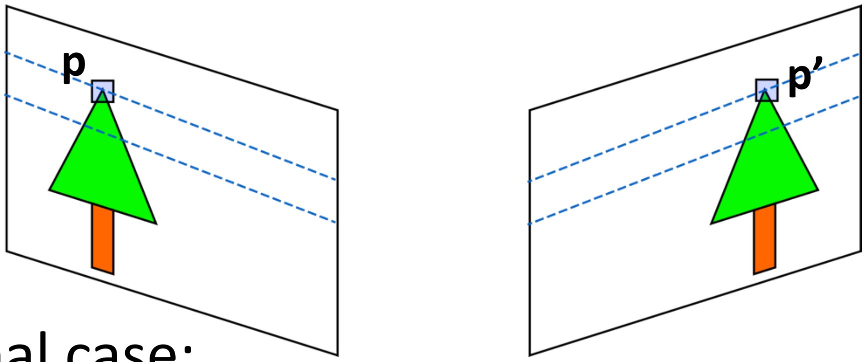
- In the rectified case:
  - Given a point $\mathbf{p}_r$ and its correspondence $\mathbf{p'}_r$ in the other image. The epipolar constraint is defined as:

$$\tilde{p}_r'^T F_r \tilde{p}_r = 0$$

  - And the fundamental matrix $\mathbf{F}_r$ is equal to:

$$F_r = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Image rectification (4)



- In the normal case:
  - Given point **p** and its correspondence **p'**: $\tilde{p}'^T F \tilde{p} = 0$
  - Let **H** and **H'** be the homographies to be applied to both images and consider rectified image points $\mathbf{p}_r$ and $\mathbf{p'}_r$ defined as:

$$\tilde{p}_r = H\tilde{p} \qquad \tilde{p}'_r = H\tilde{p}'$$

  - It follows that:

$$\tilde{p}'^T_r F_r \tilde{p}_r = 0$$

$$\tilde{p}'^T \underbrace{H'^T F_r H}_{F} \tilde{p} = 0$$

Image rectification (5)

– Knowing **F** we need to find **H** and **H'** so:

$$F = H'^{T} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} H$$

- Multiple solutions:
  – Find **H** and **H'** with less distortion
  – Decompose homography **H** (and similarly **H'**) into 3 different homographies:

$$H = H_s H_a H_p$$

Warping modes

- Different types of parametric warps include:

translation

rotation

scale

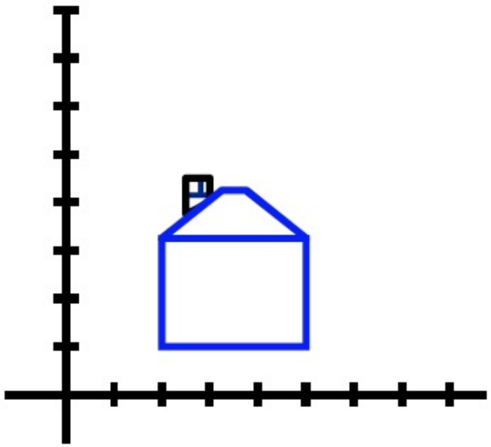
affine

perspective

Source: R. Szeliski

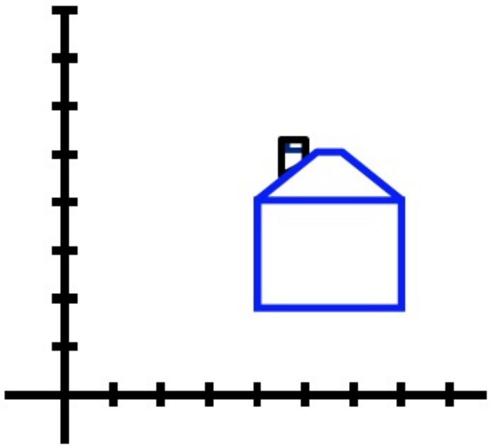
Warping models: Translation

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & h_{13} \\ 0 & 1 & h_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$



$h_{13} = 2$

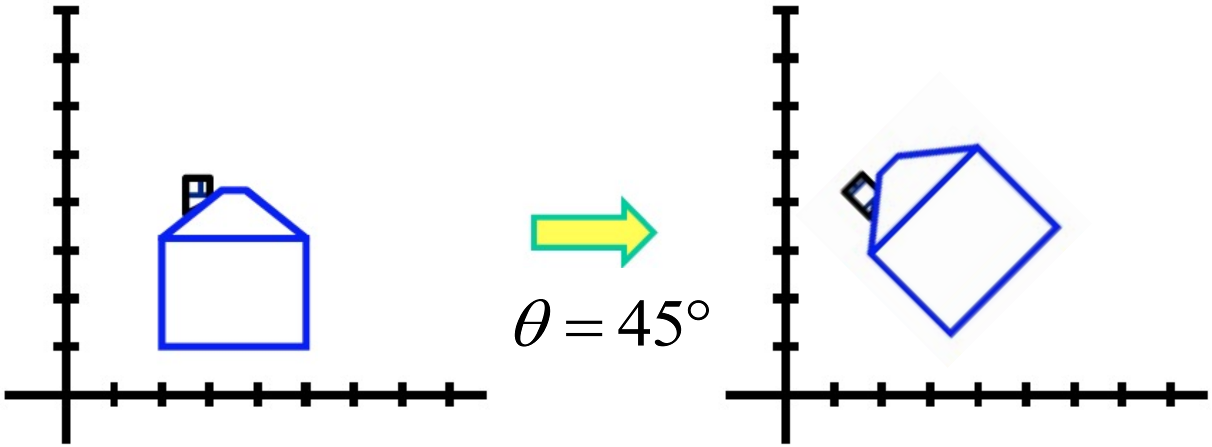
$h_{23} = 1$



Source: A. Efros

Warping models: Rotation

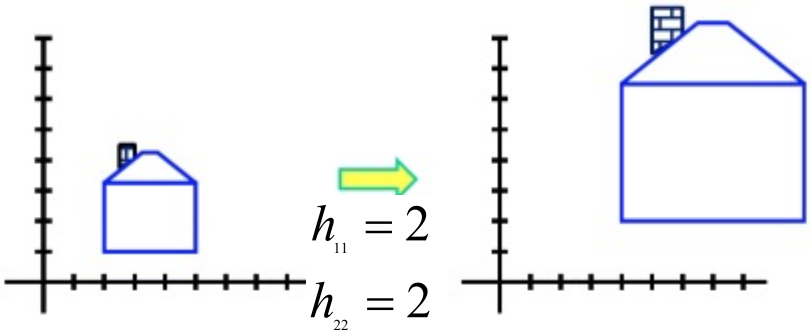
$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & 0 \\ h_{21} & h_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$



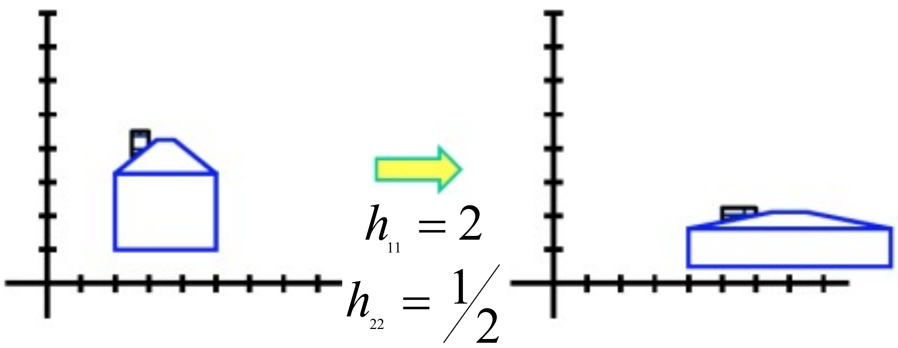
Source: A. Efros

Warping models: Scale

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & 0 & 0 \\ 0 & h_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$



Uniform Scaling



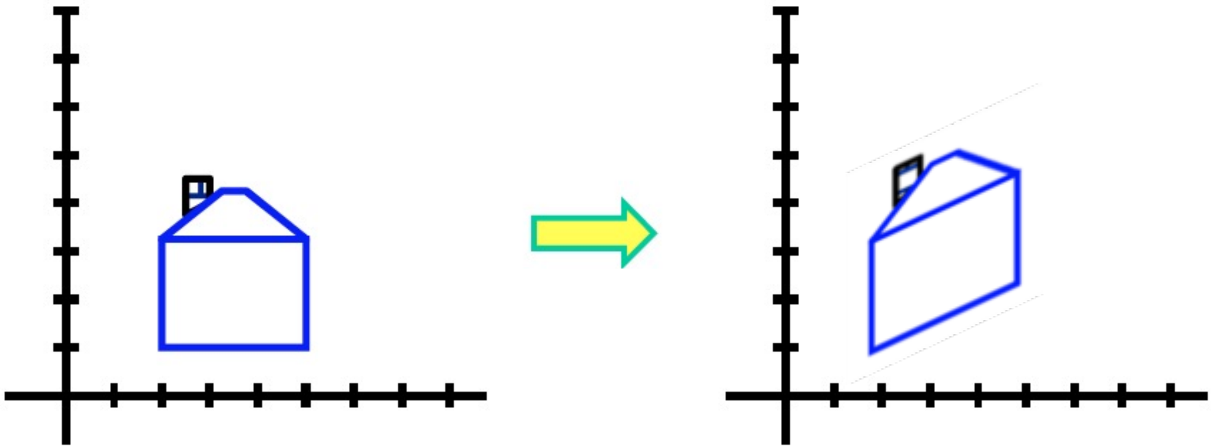
Non-uniform Scaling

Source: A. Efros

Warping models: Affine

- Combination of **rotation**, **scaling** and **translation**
- **Parallel** lines remain **parallel**

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$



Warping models: Perspective

- Combination of **affine** and **projective** warps
- **Parallel** lines **do not** necessarily **remain parallel**

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{31} & h_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

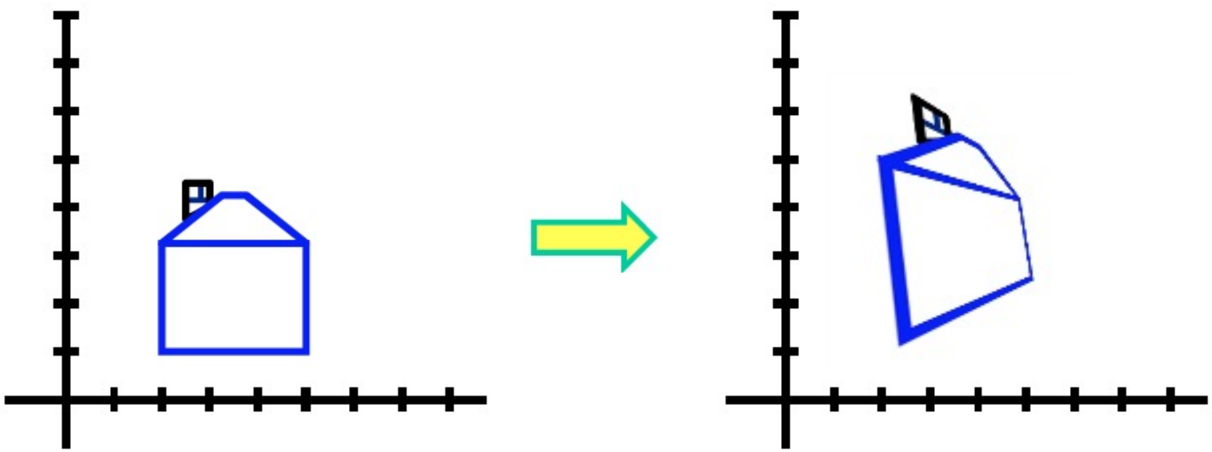


Image rectification (5) [REVIEW]

- Knowing **F** we need to find **H** and **H'** so:

$$F = H'^T \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} H$$

- Multiple solutions:
  - Find **H** and **H'** with less distortion
  - Decompose homography **H** (and similarly **H'**) into 3 different homographies:

$$H = H_s H_a H_p$$

Image rectification (6)

- $\mathbf{H_p}$ corresponds to a perspective transform

$$H_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ p_1 & p_2 & 1 \end{bmatrix}$$

  - Map epipoles to points at infinity
  - Epipolar lines become paralel
  - Use a Distortion minimization criterion
    - Minimize the variation of the weights of points in the image

Image rectification (7)

Original (un-rectified images)

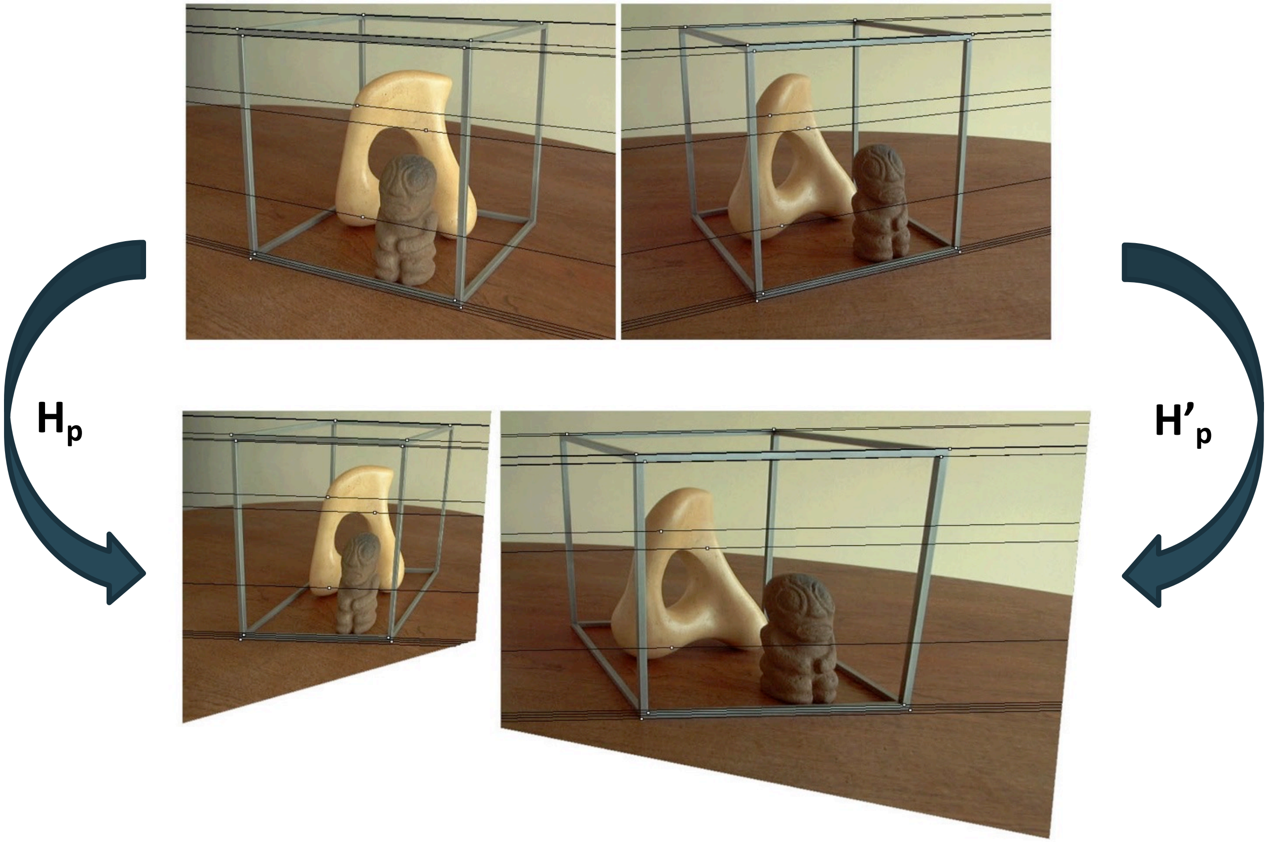


Image rectification (8)

- $\mathbf{H_a}$ corresponds to an affine transform
  - Rotate epipoles to be aligned in the X direction
  - Epipolar lines become rectified (parallel to X axes)

$$H_a = \begin{bmatrix} a_1 & a_2 & 0 \\ a_3 & a_4 & a_5 \\ 0 & 0 & 1 \end{bmatrix}$$

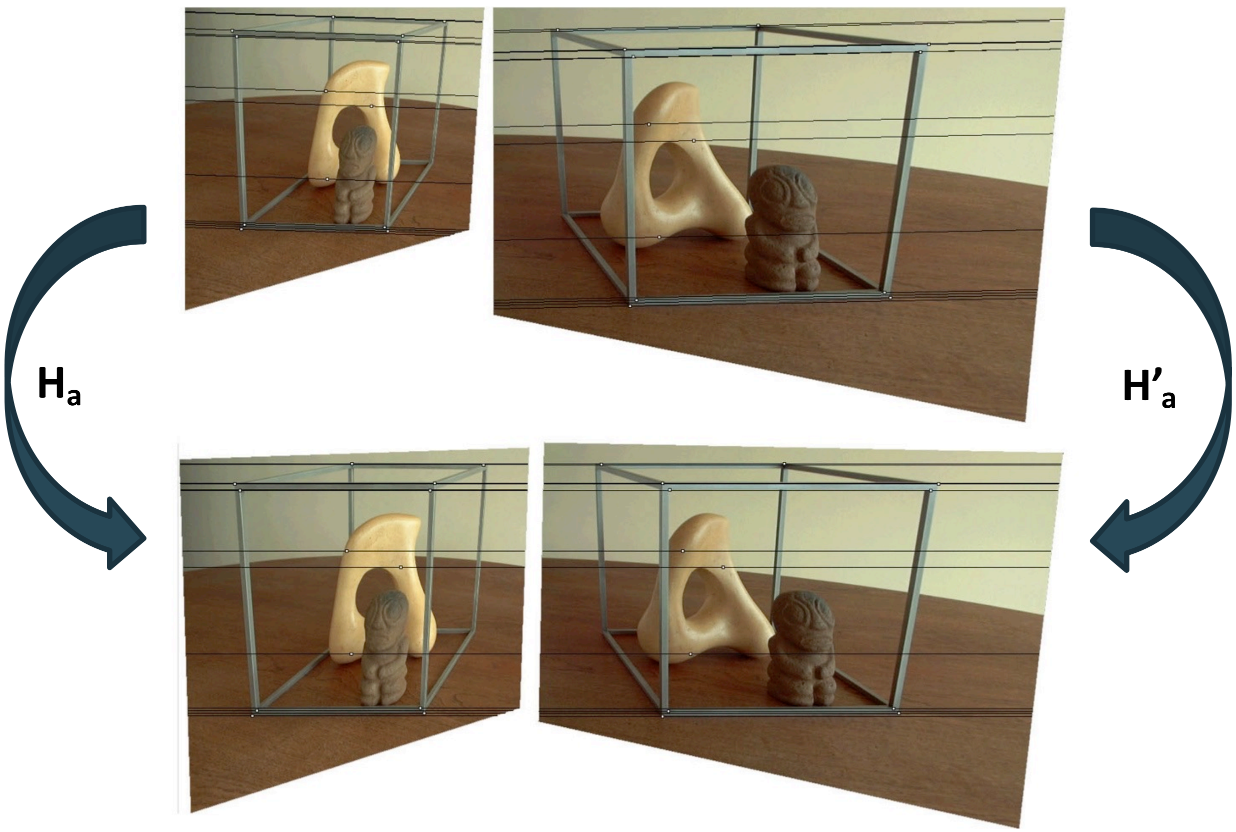
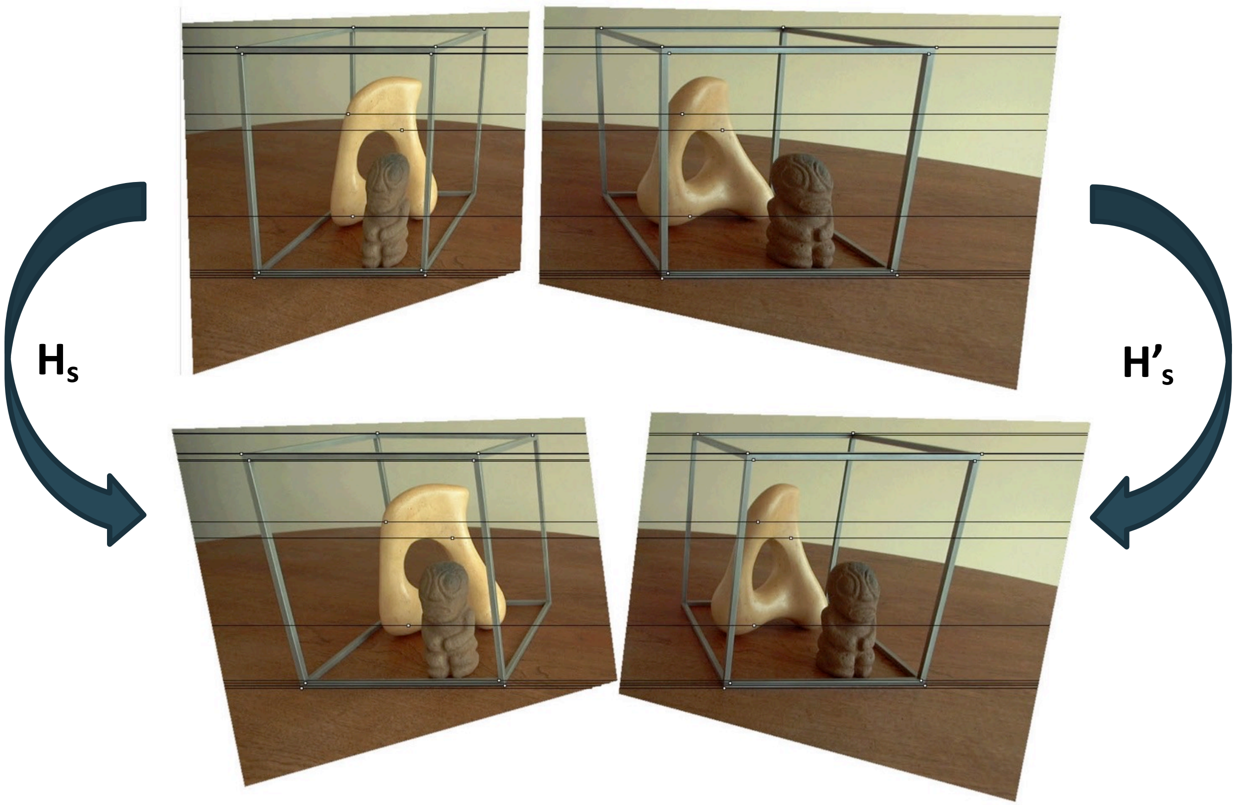


Image rectification (9)

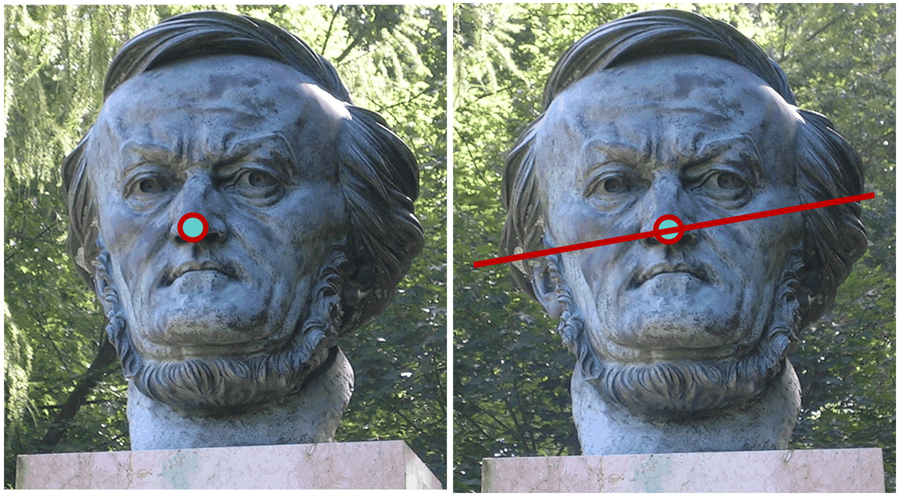
- $\mathbf{H_s}$ corresponds to a shearing transform
  - Reduce horizontal distortion (X coordinate)
  - No effect on rectification

$$H_s = \begin{bmatrix} s_1 & s_2 & s_3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



In this two lessons we have seen…

- Epipolar contraints
  - Essential and Fundamental matrices
- Estimating Fundamental matrix
  - 8 point algorithm
  - RANSAC
- Image rectification



References

- Epipolar geometry
  - R. Hartley and A. Zisserman, ***Multiple View Geometry in Computer Vision***, Cambridge University Press, June 2000. [Chapter 9]
  - G. Xu and Z. Zhang, ***Epipolar Geometry in Stereo Motion and Object Recognition***, Kluwer Academic Publishers, 1996. ISBN 0-7923-4199-6
  - R. I. Hartley, ***In Defence of the 8-point Algorithm***, IEEE transaction on pattern analysis and machine intelligence, Vol. 19, No. 6, June 1997
- Image rectification
  - C. Loop and Z. Zhang, ***Computing Rectifying Homographies for Stereo Vision***, Technical Report, Microsoft Research, 1998.