计算机视觉与模式识别

Computer Vision and Pattern Recognition

-- Geometric Vision

Stereo Vision, Two View Geometry, Epipolar constraints

人工智能与机器人研究所

Institute of Artificial Intelligence and Robotics

袁泽剑

Email: yuan.ze.jian@xjtu.edu.cn

科学馆102室

Syllabus

- Physiology and theories of Vision (1)
- Image Formation and Camera Model (1)
- **Stereo Vision & SFM (1/3)**
- Image Filtering & Structure Extraction (2)
- Local Features & Image Matching (2)
  - Visual vocabularies and Image indexing(1)
- Segmentation (Clustering /Grouping) (2)
- Visual Recognition (4)
- Motion / Optical flow / Tracking (2-1)

Introduction

- **Geometric vision**
  - ➢ Visual cues
  - ➢ Stereo vision
- **Epipolar geometry**
  - ➢ Depth with stereo
  - ➢ Geometry for a simple stereo system
    - ✓ Case example with parallel optical axes
    - ✓ General case with calibrated cameras
- **Stereopsis & 3D Reconstruction**
  - ➢ Correspondence search
  - ➢ Additional correspondence constraints
  - ➢ Possible sources of error
  - ➢ Applications

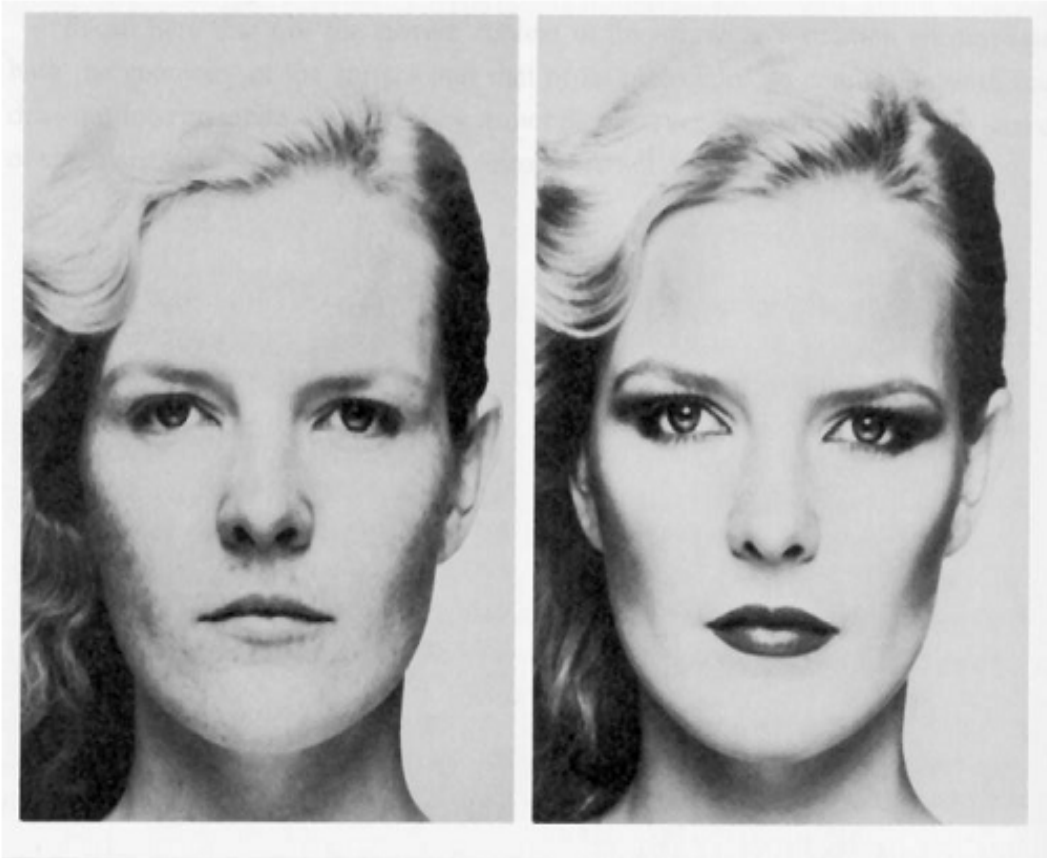
Geometric vision

- Goal: Recovery of 3D structure
  - What cues in the image allow us to do this?

Slide credit: Svetlana Lazebnik

Visual Cues

- Shading

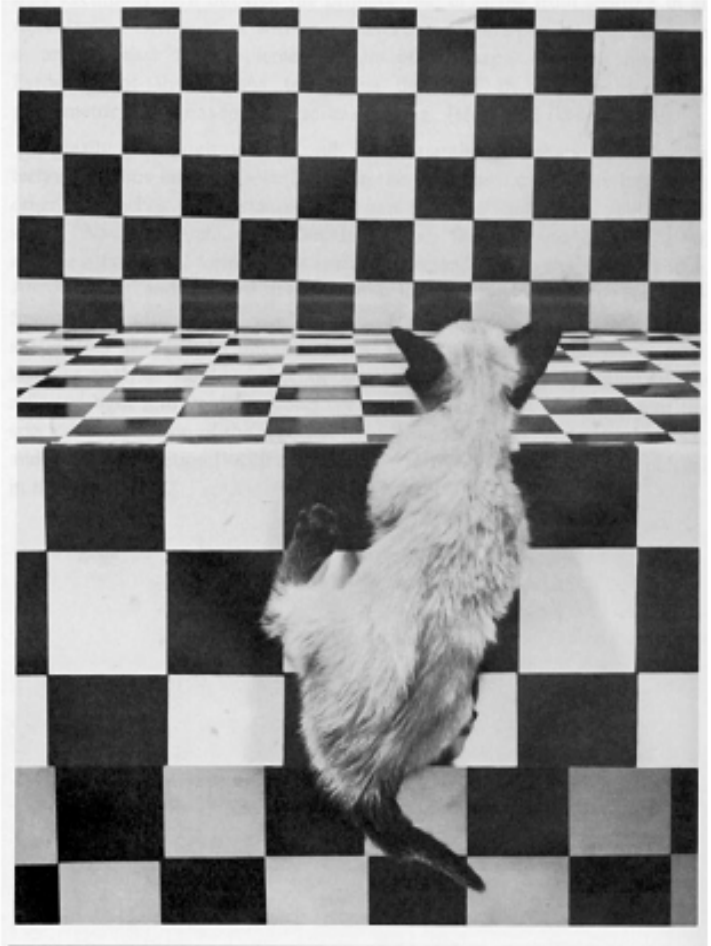


Merle Norman Cosmetics, Los Angeles

Slide credit: Steve Seitz

Visual Cues

- Shading
- Texture



The Visual Cliff, by William Vandivert, 1960

Slide credit: Steve Seitz

Visual Cues

- Shading
- Texture
- Focus

From *The Art of Photography*, Canon

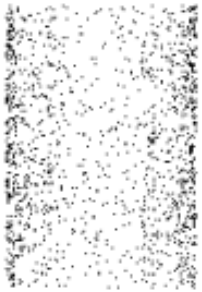
Visual Cues

- Shading
- Texture
- Focus
- Perspective

Visual Cues

- Shading
- Texture
- Focus
- Perspective
- Motion

Figures from L. Zhang

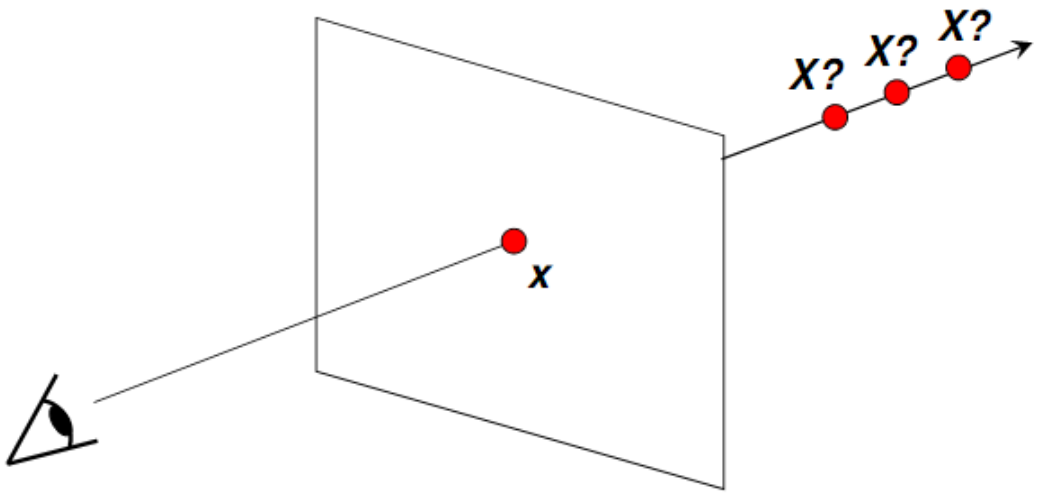


Slide credit: Steve Seitz, Kristen Grauman

Goal: Recovery of 3D Structure

- Focus on **perspective and motion** (SFM)
- **Multi-view geometry**

  (Recovery of structure from one image is inherently ambiguous)



Slide credit: Svetlana Lazebnik

Example

- Structure and depth are inherently ambiguous from single views.

Slide credit: Svetlana Lazebnik, Kristen Grauman

What Is Stereo Vision?

- Generic problem formulation: given several images of the same object or scene, compute a representation of its 3D shape

Slide credit: Svetlana Lazebnik, Steve Seitz

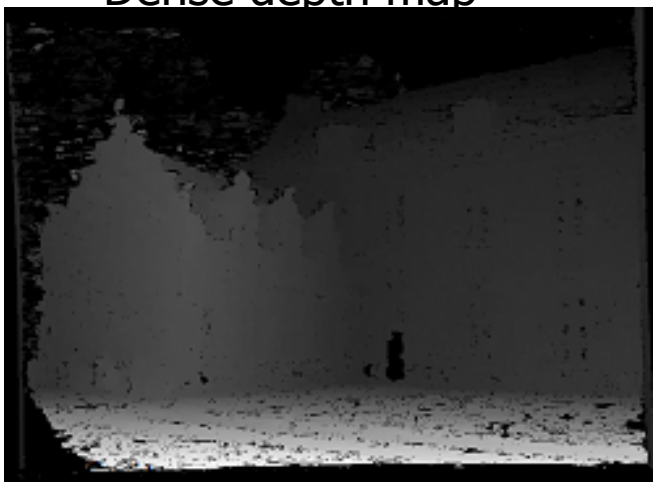
What Is Stereo Vision?

- Narrower formulation: given a calibrated binocular stereo pair, fuse it to produce a depth image

Image 1

Image 2

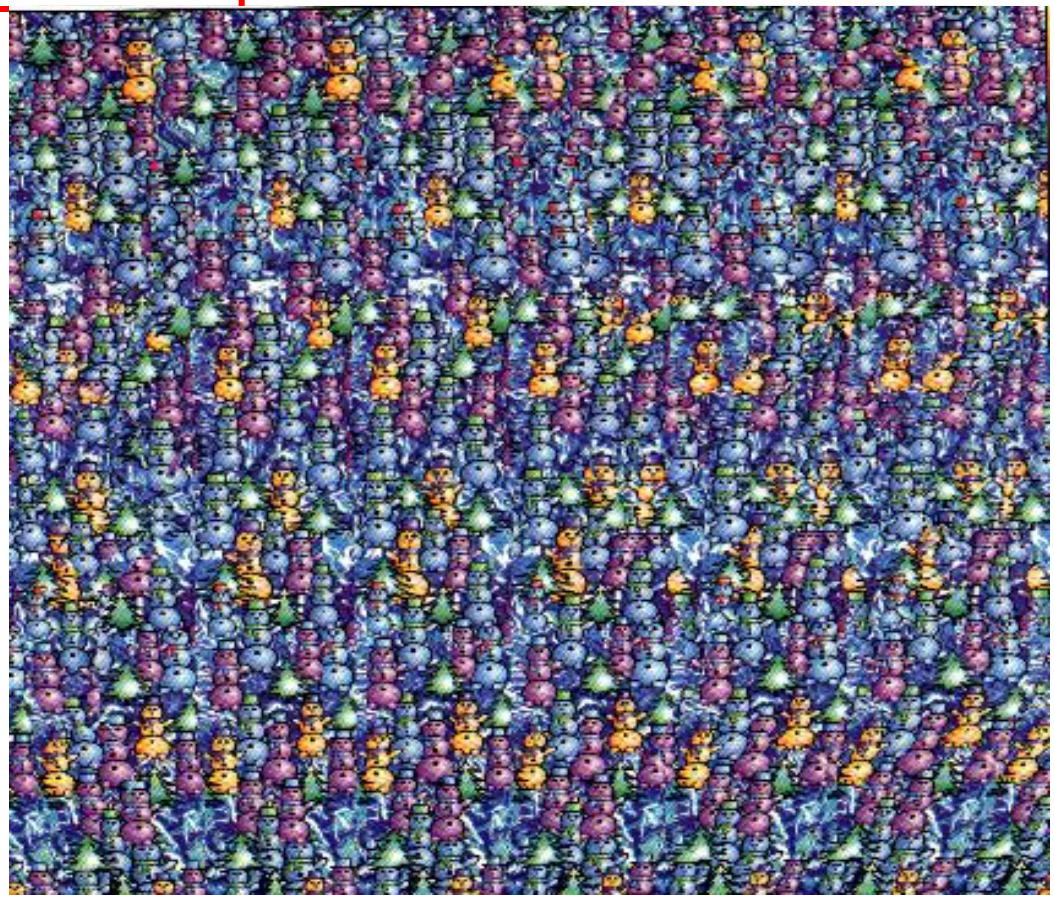
Dense depth map



Source: Svetlana Lazebnik, Steve Seitz

What Is Stereo Vision?

- Narrower formulation: given a calibrated binocular stereo pair, fuse it to produce a depth image.
  - Humans can do it

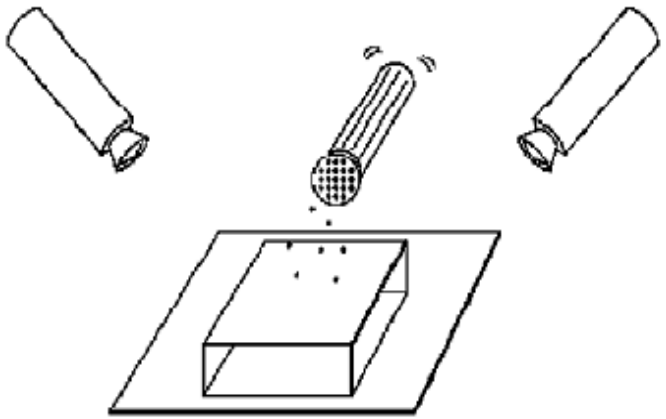


Source: Svetlana Lazebnik, Steve Seitz

Historic Origin: Random Dot Stereograms

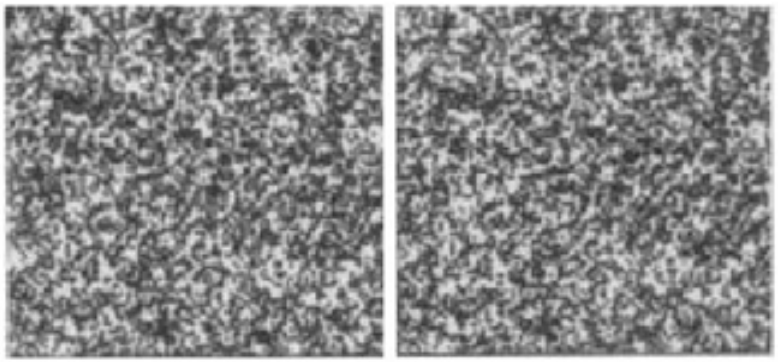
- Julesz 1960: Do we identify local brightness patterns before fusion (monocular process) or after (binocular)?

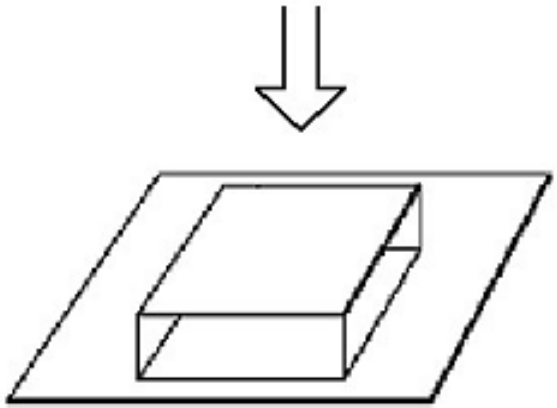
- To test: pair of synthetic images obtained by randomly spraying black dots on white objects



Slide credit: Kristen Grauman

Random Dot Stereograms



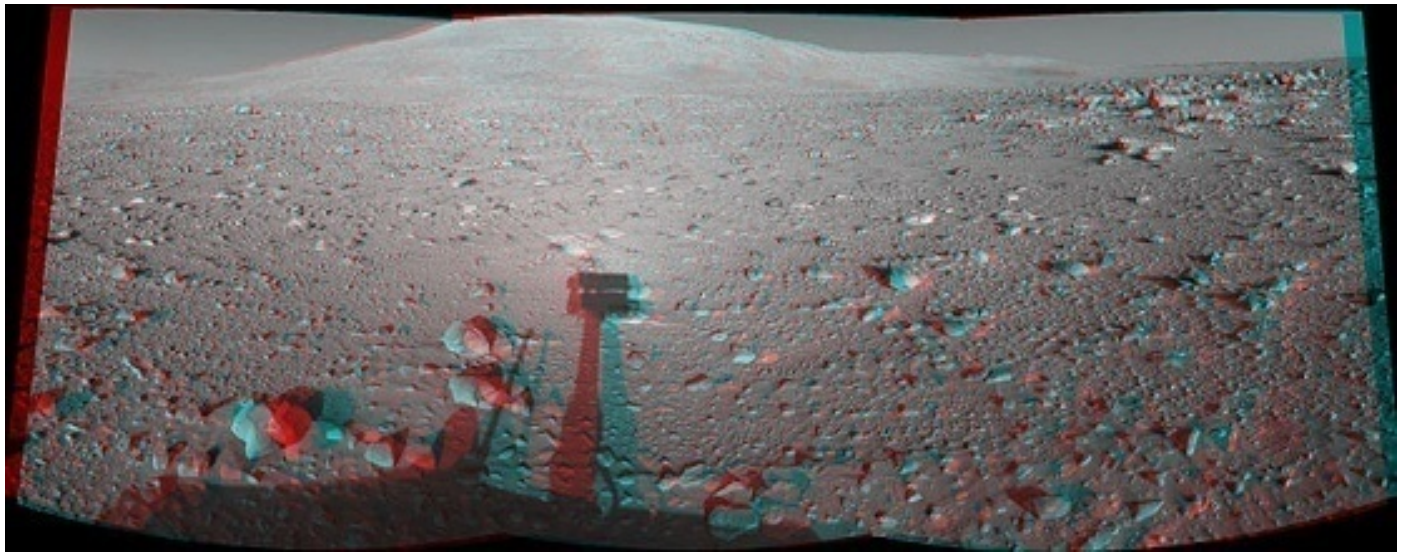


- When viewed monocularly, they appear random; when viewed stereoscopically, see 3d structure.

Slide credit: Kristen Grauman

Application of Stereo: Robotic Exploration

Real-time stereo on Mars



Slide credit: Steve Seitz

Two-View Geometry

- Correspondence (stereo matching): Given a point in just one image, how does it constrain the position of the corresponding point x' in another image?

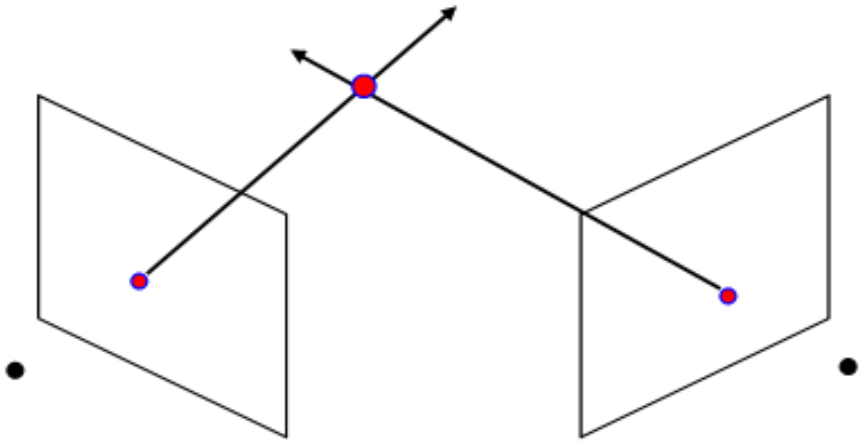
- Scene geometry (structure): Given corresponding points in two or more images, where is the pre-image of these points in 3D?

- Camera geometry (motion): Given a set of corresponding points in two images, what are the cameras for the two views?

Epipolar geometry
(对极几何/极线几何)

- Depth with stereo / triangulation
- Geometry for a simple stereo system
- Stereo system with parallel optical axes
- General case with calibrated cameras

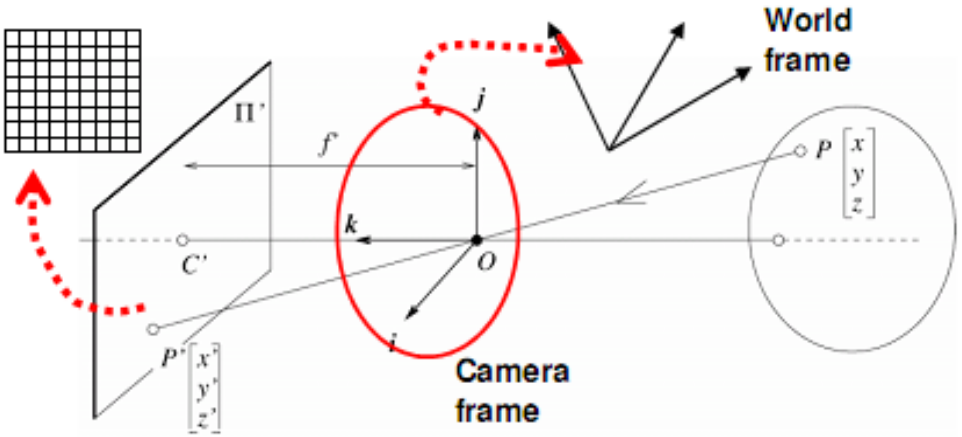
Depth with Stereo: Basic Idea



- Basic Principle: Triangulation
  - Gives reconstruction as intersection of two rays
  - Requires
    - Camera pose (calibration)
    - Point correspondence

Slide credit: Steve Seitz

Recall：Camera Calibration



Extrinsic parameters:
Camera frame ↔ Reference frame

Intrinsic parameters:
Image coordinates relative to camera ↔ Pixel coordinates

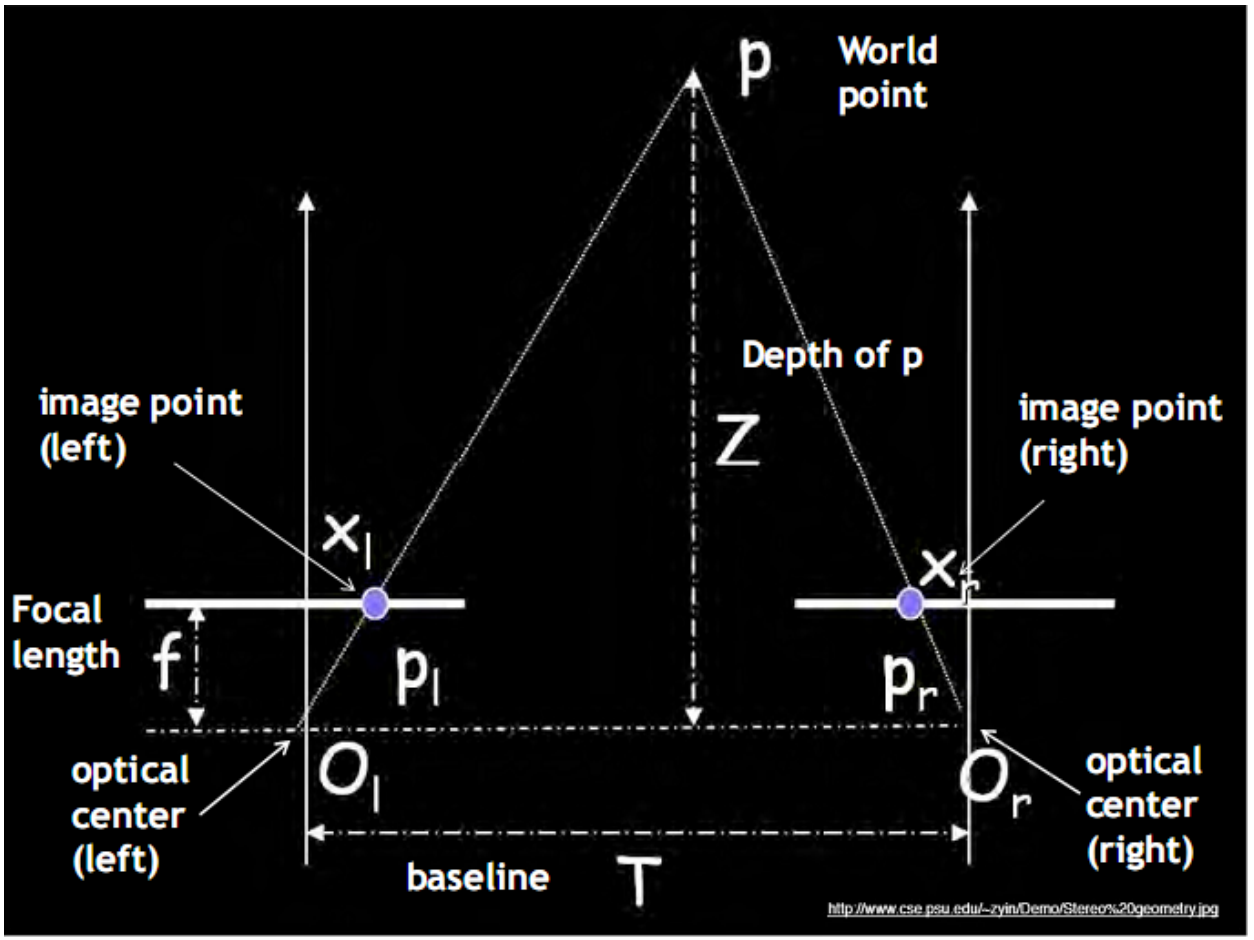
- *Extrinsic params*: rotation matrix and translation vector
- *Intrinsic params*: focal length, pixel sizes (mm), image center point, radial distortion parameters

Assumption： these parameters are given and fixed.

Slide credit: Kristen Grauman

Geometry for a Simple Stereo System

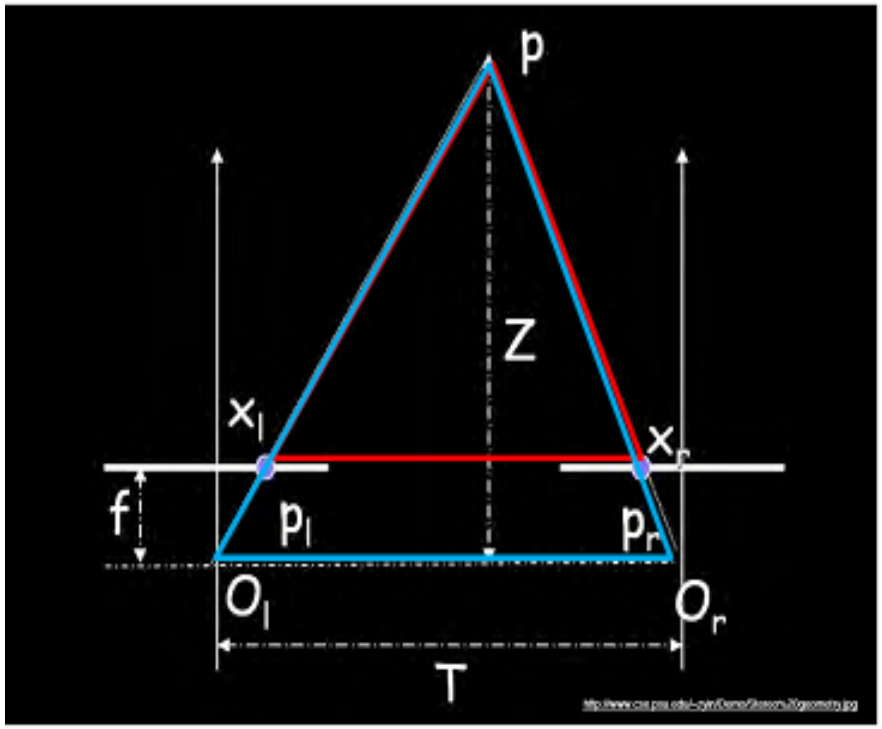
- Assuming parallel optical axes, known camera parameters (i.e., calibrated cameras):



Slide credit: Kristen Grauman

Geometry for a Simple Stereo System

- Assume parallel optical axes, known camera parameters (i.e., calibrated cameras). We can triangulate via:



Similar triangles (p_l, P, p_r) and (O_l, P, O_r):

$$\frac{T + x_l - x_r}{Z - f} = \frac{T}{Z}$$

$$Z = f\frac{T}{x_r - x_l}$$

disparity

Depth From Disparity

Image I(x,y)

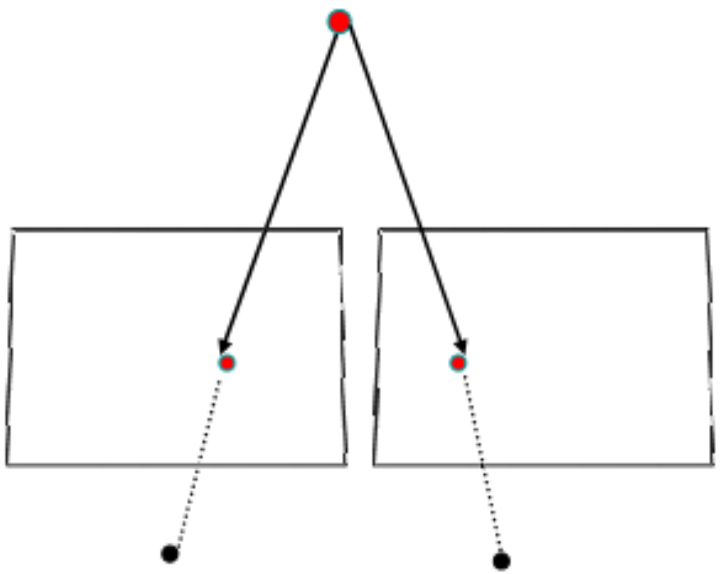
Disparity map D(x,y)

Image I´(x´,y´)

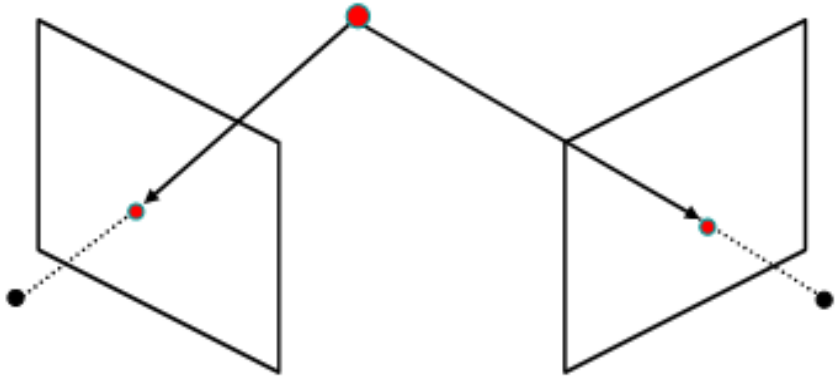
$$(x´,y´)=(x+D(x,y),\ y)$$

General Case With Calibrated Cameras

- The two cameras need not have parallel optical axes.



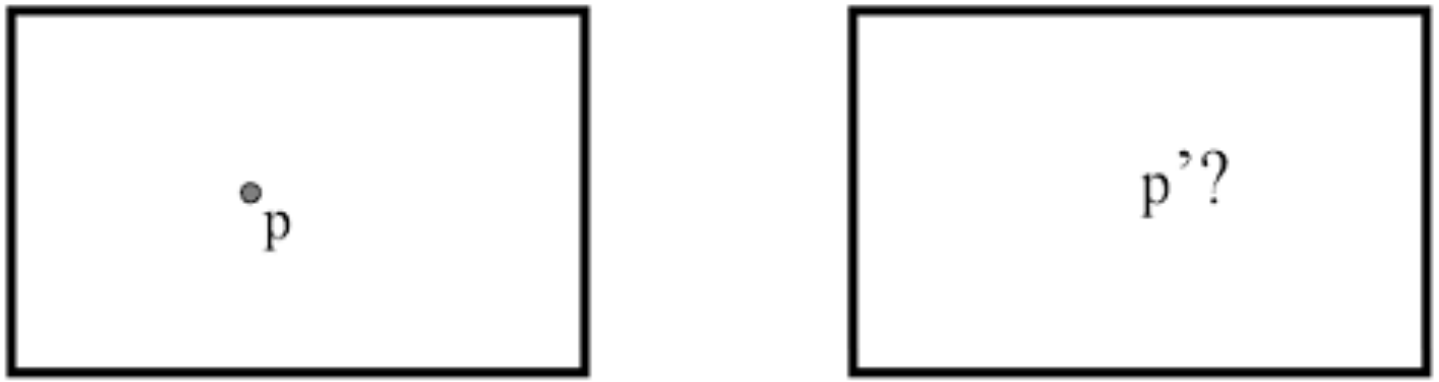
vs.



Slide credit: Kristen Grauman, Steve Seitz

Stereo Correspondence Constraints

- Given p in the left image, where can the corresponding point p' in the right image be?

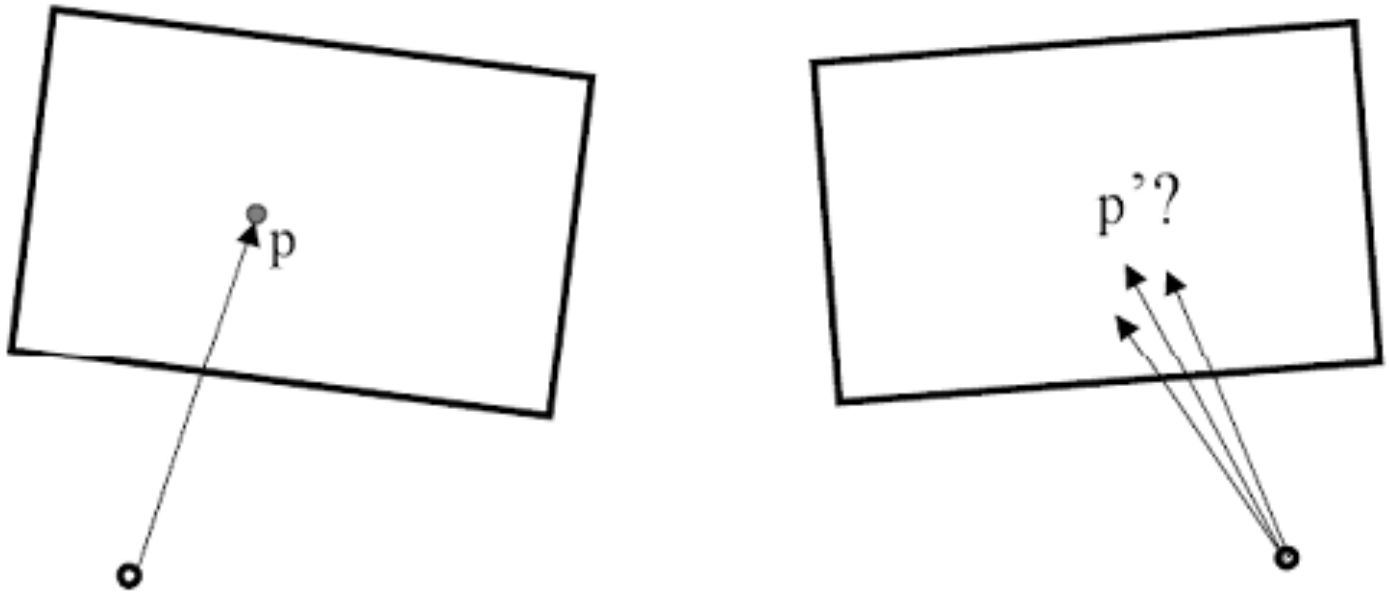


Parallel optical axes

Slide credit: Kristen Grauman

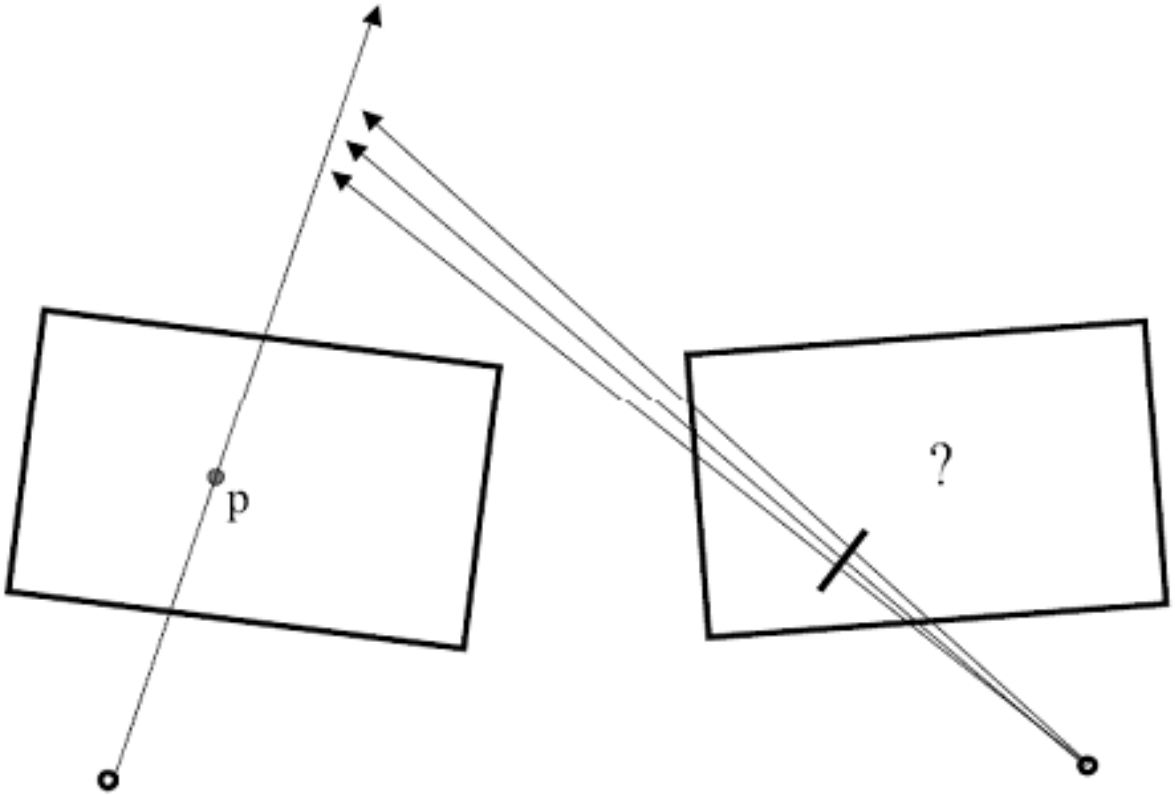
Stereo Correspondence Constraints

- Given p in the left image, where can the corresponding point p' in the right image be?



General case

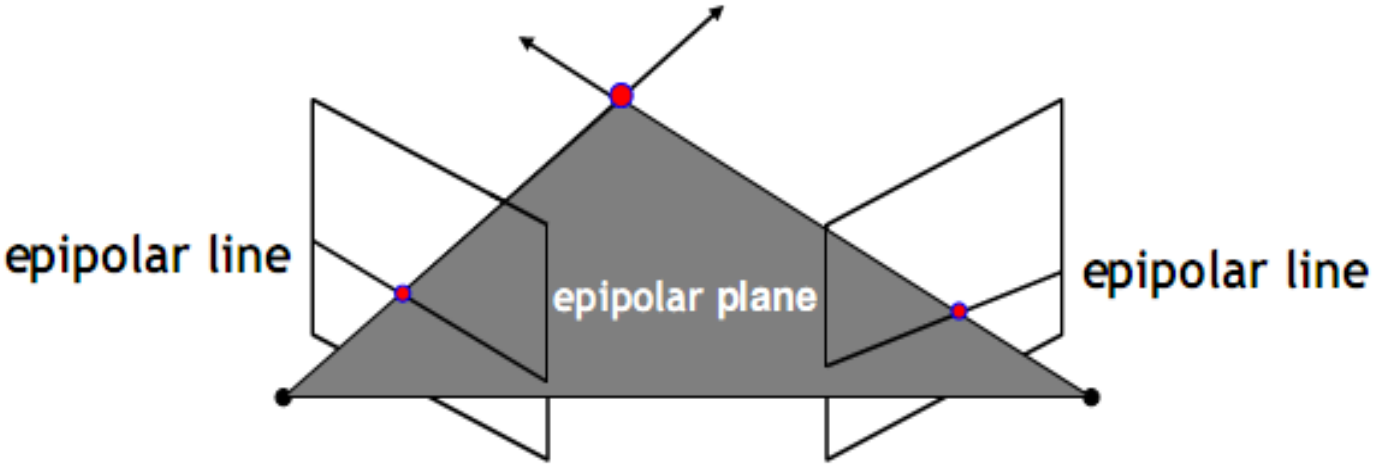
Stereo Correspondence Constraints



Slide credit: Kristen Grauman

Stereo Correspondence Constraints

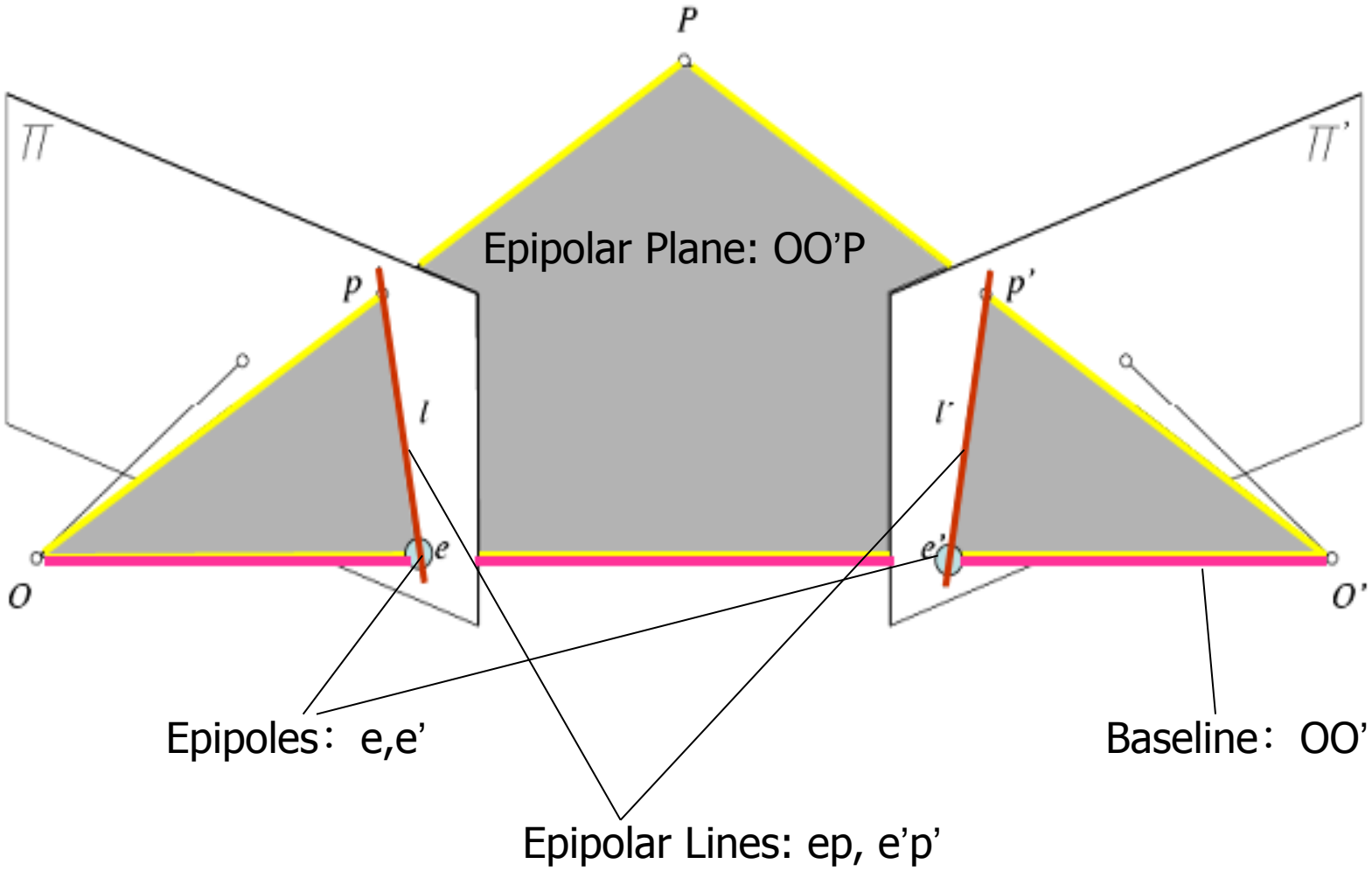
- Geometry of two views allows us to constrain where the corresponding pixel for some image point in the first view must occur in the second view.



- Epipolar constraint: Why is this useful?
  - Reduces correspondence problem to 1D search along conjugate epipolar lines.

Source: Steve Seitz

Epipolar Geometry



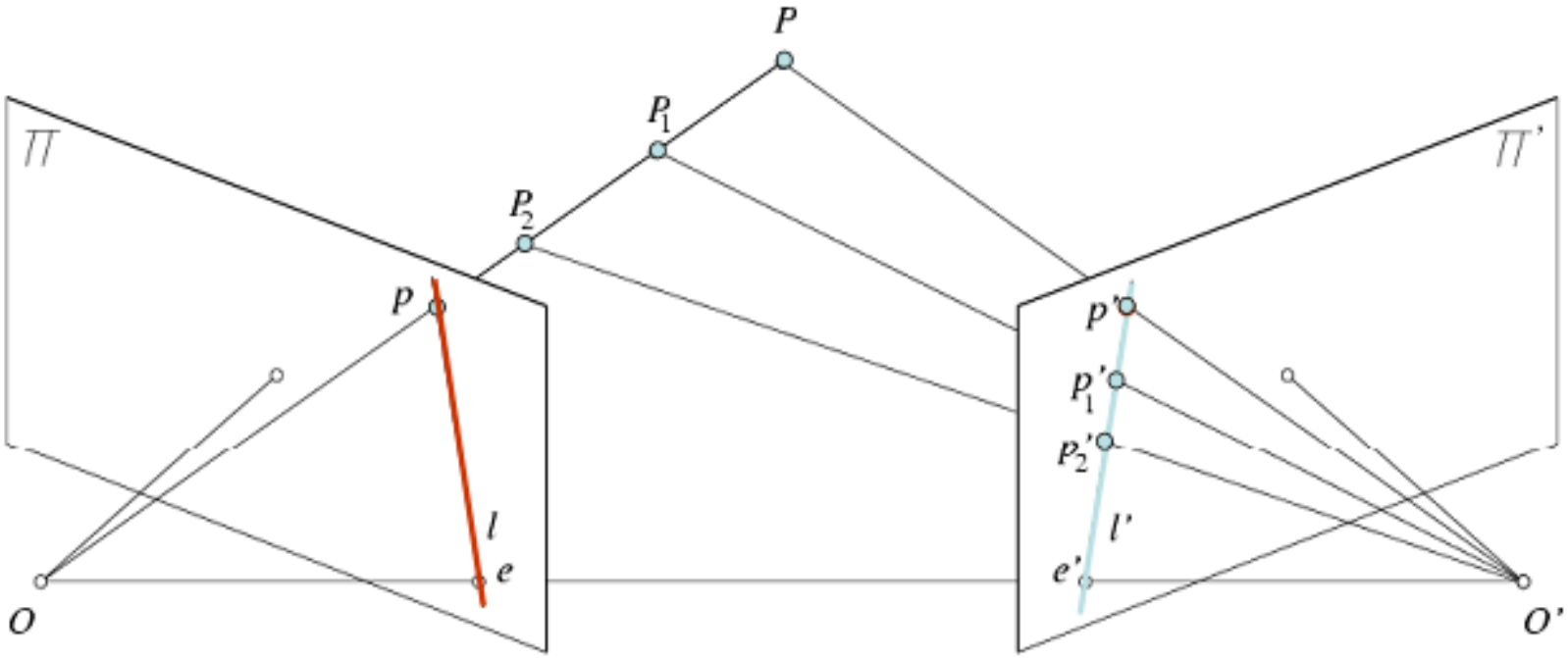
Source： Marc Pollefeys

Epipolar Geometry: Terms

- **Baseline**: line joining the camera centers
- **Epipole**: point of intersection of baseline with the 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**

Source: Marc Pollefeys

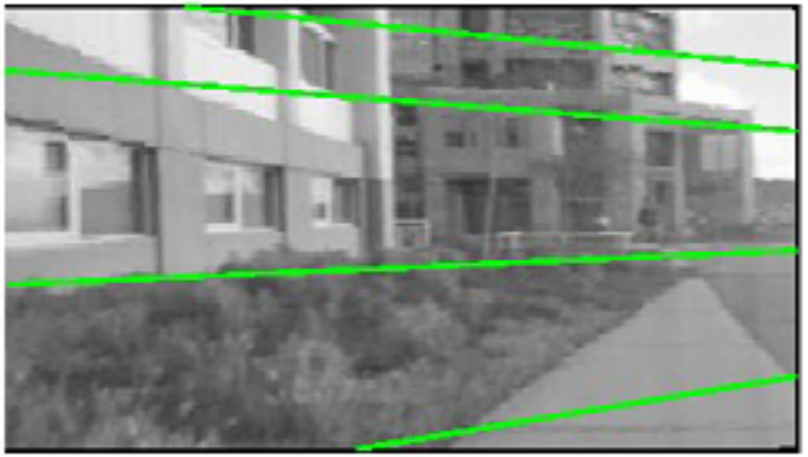
Epipolar Constraint



- Potential matches for p have to lie on the corresponding epipolar line l'.
- Potential matches for p' have to lie on the corresponding epipolar line l.

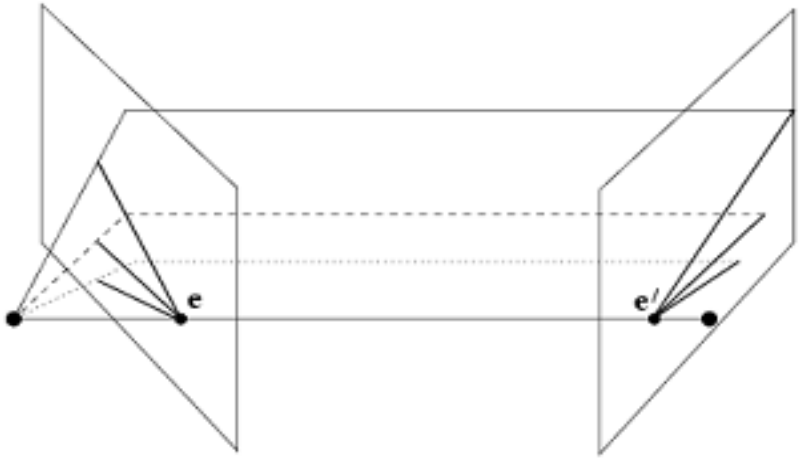
Source: Marc Pollefeys

Example

Slide credit: Kristen Grauman

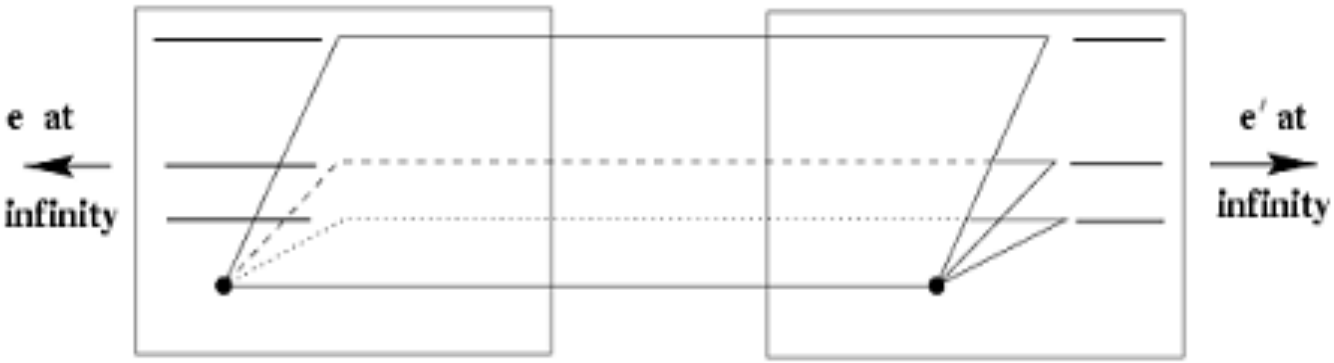
Example: Converging Cameras



As position of 3d point varies, epipolar lines "rotate" about the baseline

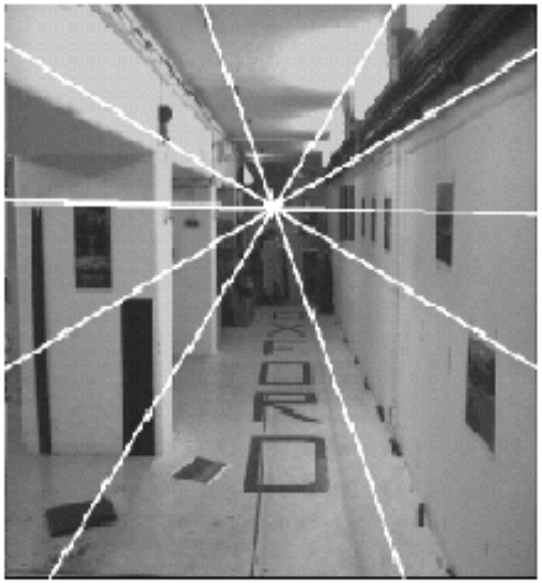
Slide credit: Kristen Grauman

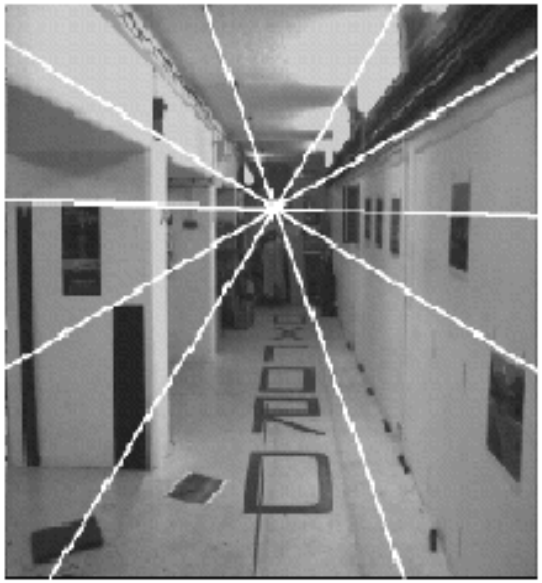
Example: Motion Parallel With Image Plane

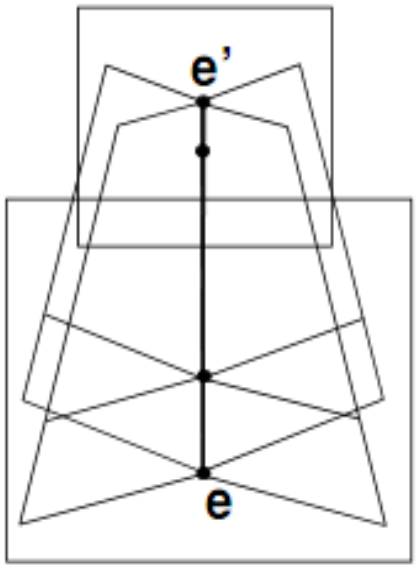


Slide credit: Kristen Grauman

Example: Forward Motion





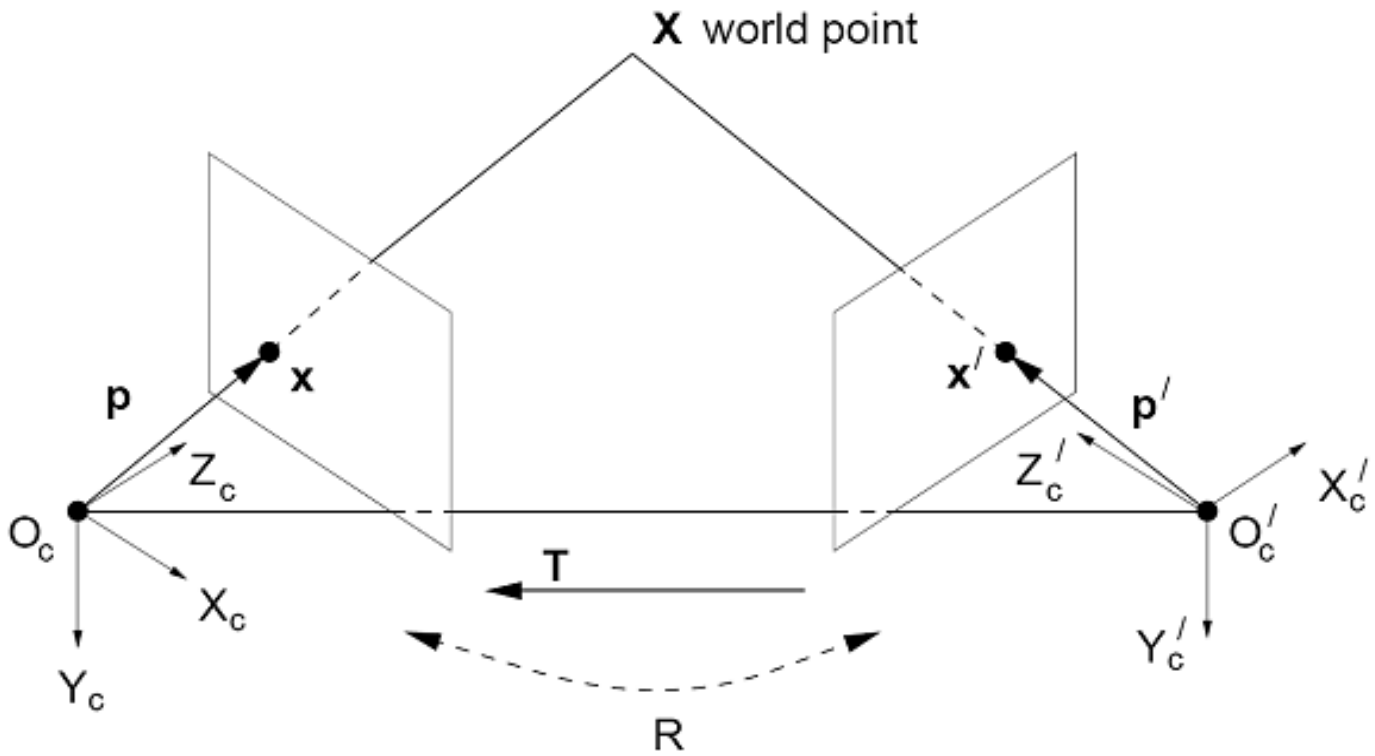


- Epipole has same coordinates in both images.
- Points move along lines radiating from e: "Focus of expansion"

Slide credit: Kristen Grauman

- For a given stereo rig, how do we express the epipolar constraints algebraically?

Stereo Geometry With Calibrated Cameras



- If the rig is calibrated, we know:
  - How to rotate and translate camera reference frame 1 to get to camera reference frame 2.
    - Rotation: 3 x 3 matrix; translation: 3 vector.

Slide credit: Kristen Grauman, Steve Seitz

Rotation Matrix

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Express 3d rotation as series of rotations around coordinate axes by angles α, β, γ

Overall rotation is product of these elementary rotations:

$$\mathbf{R} = \mathbf{R}_x \mathbf{R}_y \mathbf{R}_z$$

Slide credit: Kristen Grauman

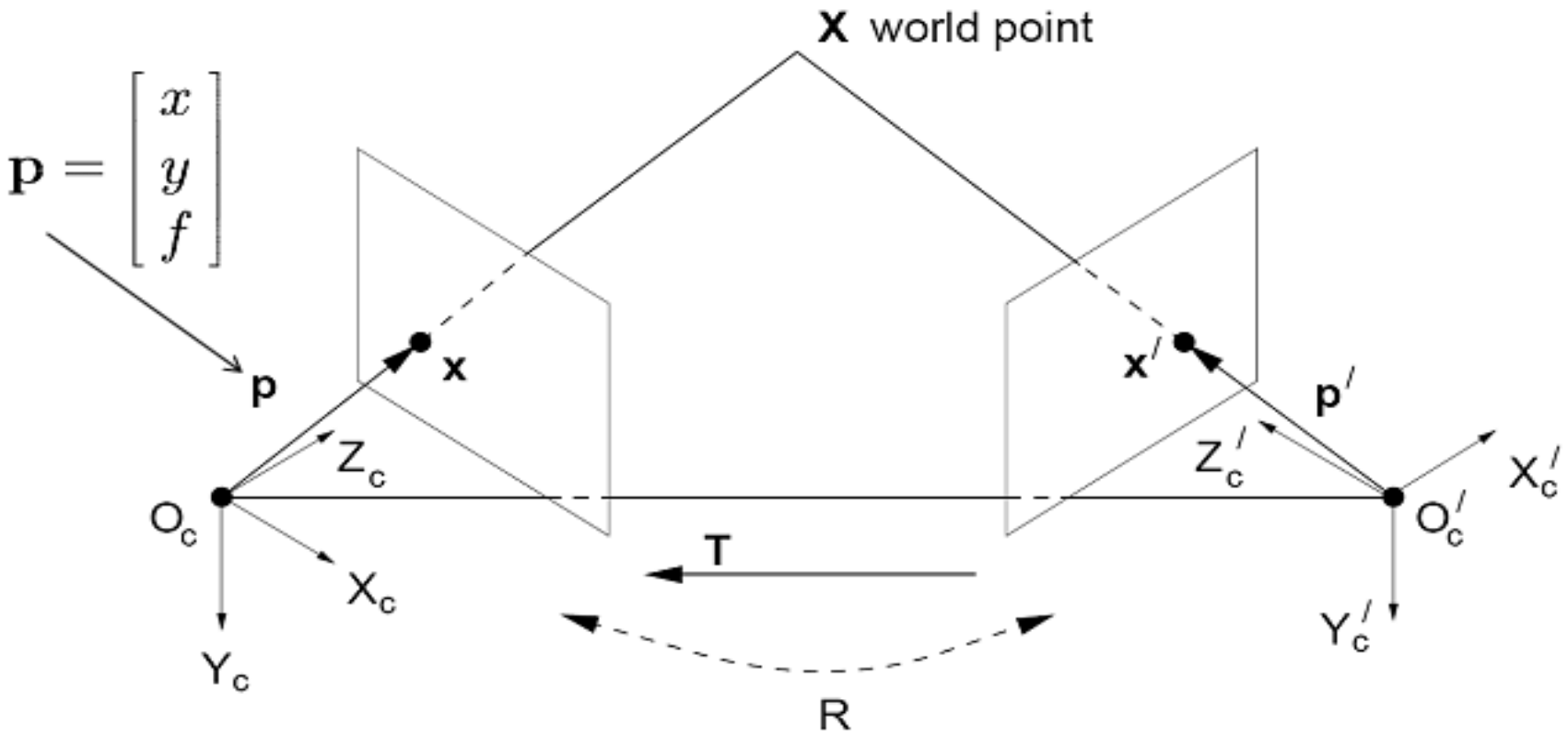
3D Rigid Transformation

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}$$

Slide credit: Kristen Grauman

Stereo Geometry With Calibrated Cameras



- Camera-centered coordinate systems are related by known rotation R and translation T:

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}$$

Cross Product & Dot Product

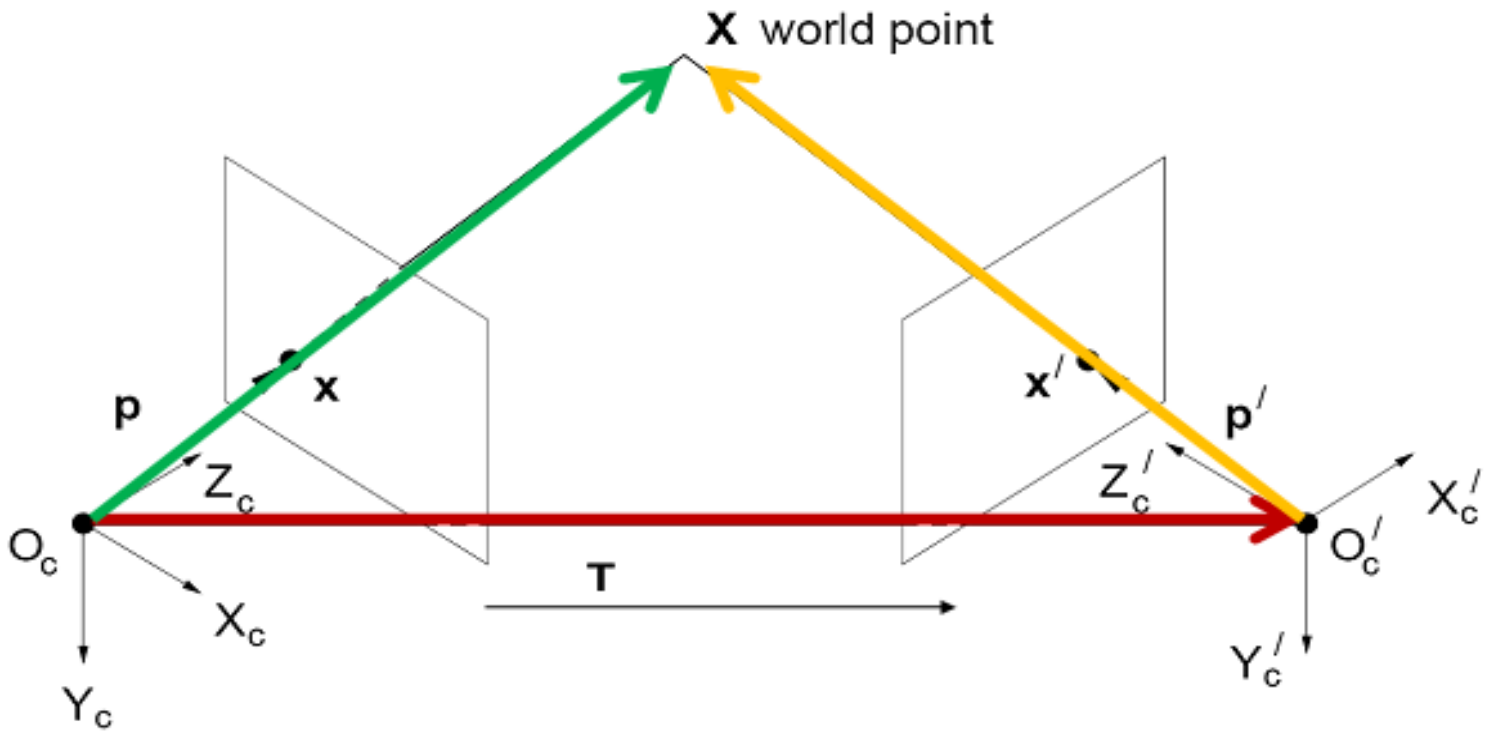
$$\vec{a} \times \vec{b} = \vec{c}$$

$$\vec{a} \cdot \vec{c} = 0$$
$$\vec{b} \cdot \vec{c} = 0$$

- Vector cross product takes two vectors and returns a third vector that's perpendicular to both inputs.
- So here, c is perpendicular to both a and b, which means the dot product = 0.

Slide credit: Kristen Grauman

From Geometry to Algebra



$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}$$

$$\mathbf{X}' \cdot (\mathbf{T} \times \mathbf{X}') = \mathbf{X}' \cdot (\mathbf{T} \times \mathbf{R}\mathbf{X})$$

$$\mathbf{T} \times \mathbf{X}' = \mathbf{T} \times \mathbf{R}\mathbf{X} + \mathbf{T} \times \mathbf{T}$$

$$= 0$$

Normal to the plane

$$= \mathbf{T} \times \mathbf{R}\mathbf{X}$$

Slide credit: Kristen Grauman

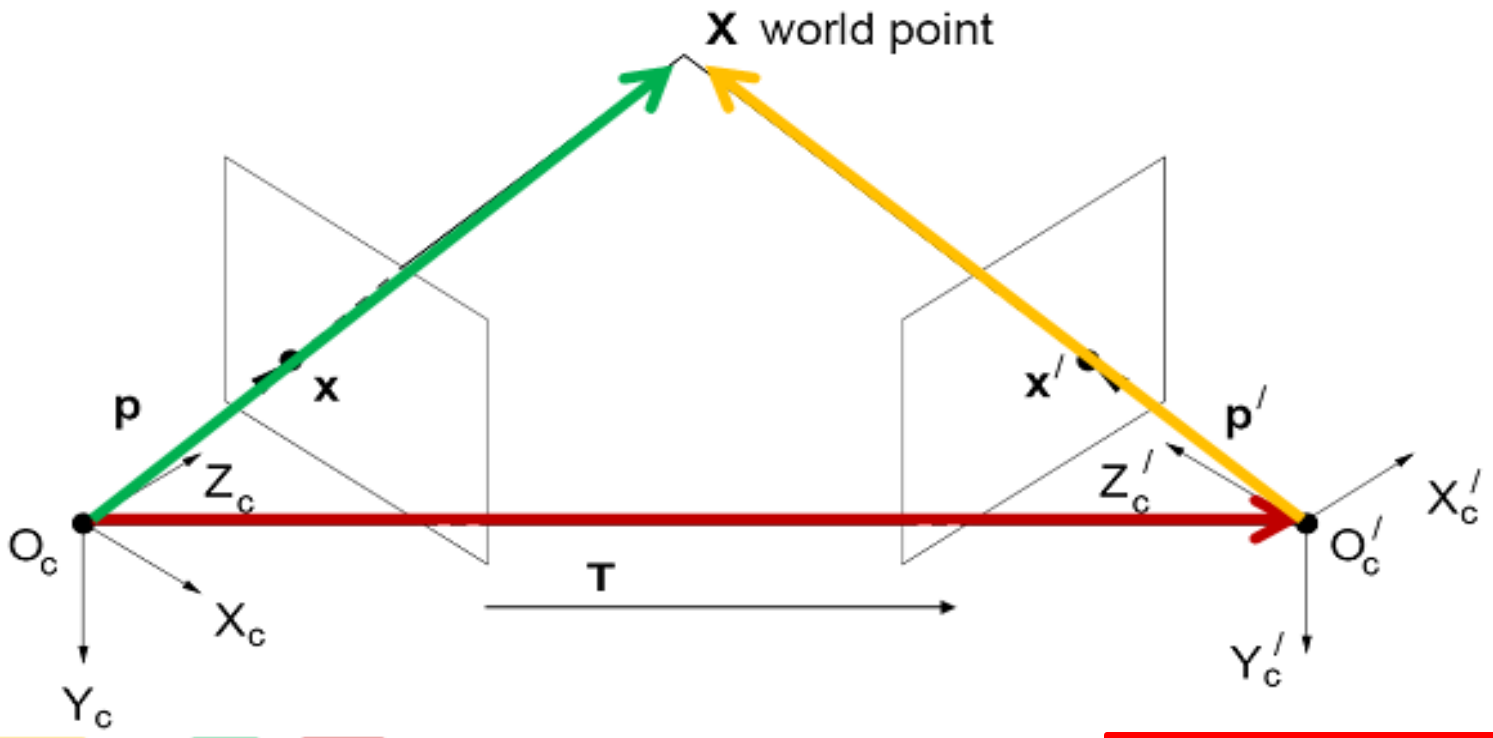
Matrix Form of Cross Product

$$\vec{a}\times\vec{b}=\begin{pmatrix}a_x\\a_y\\a_z\end{pmatrix}\times\begin{pmatrix}b_x\\b_y\\b_z\end{pmatrix}=\begin{pmatrix}a_yb_z-a_zb_y\\a_zb_x-a_xb_z\\a_xb_y-a_yb_x\end{pmatrix}$$

$$\vec{a}\times\vec{b}=[a_\times]\vec{b}$$

$$[a_\times]=\begin{bmatrix}0&-a_z&a_y\\a_z&0&-a_x\\-a_y&a_x&0\end{bmatrix}$$

From Geometry to Algebra



$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}$$

$$\mathbf{T}\times\mathbf{X}' = \mathbf{T}\times\mathbf{R}\mathbf{X} + \mathbf{T}\times\mathbf{T}$$

Normal to the plane

$$= \mathbf{T}\times\mathbf{R}\mathbf{X}$$

$$\mathbf{X}'\cdot(\mathbf{T}\times\mathbf{X}') = \mathbf{X}'\cdot(\mathbf{T}\times\mathbf{R}\mathbf{X})$$

$$= 0$$

Slide credit: Kristen Grauman

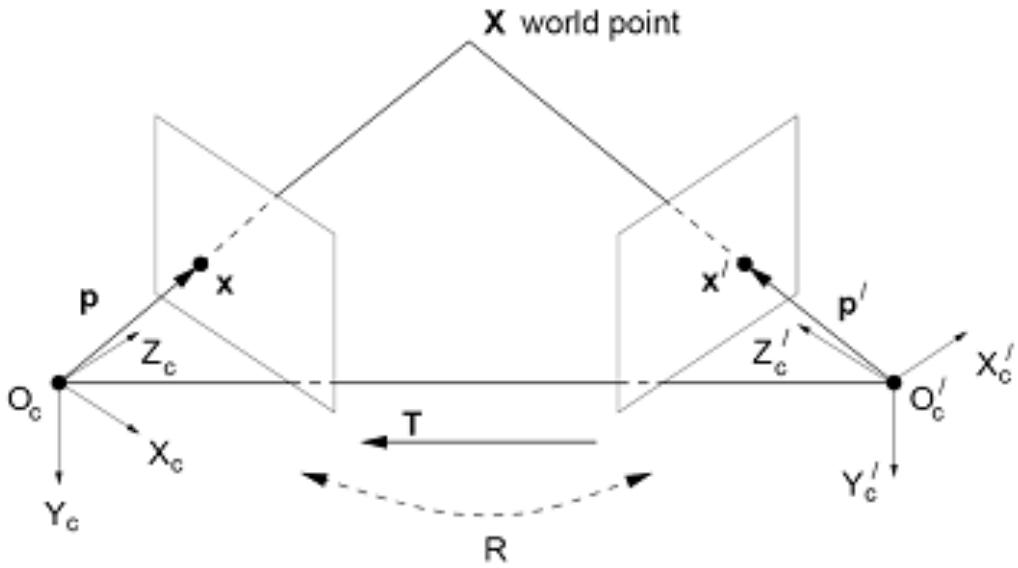
Essential Matrix

$$\mathbf{X}' \cdot (\mathbf{T} \times \mathbf{R}\mathbf{X}) = 0$$

$$\mathbf{X}' \cdot ([\mathbf{T}_\times]\mathbf{R}\mathbf{X}) = 0$$

Let $\mathbf{E} = [\mathbf{T}_\times]\mathbf{R}$

$$\mathbf{X}'^T \mathbf{E}\mathbf{X} = 0$$



- This holds for the rays p and p' that are parallel to the camera-centered position vectors X and X', so we have: $\mathbf{p}'^T \mathbf{E}\mathbf{p} = 0$
- $\mathbf{E}$ is called the essential matrix, which relates corresponding image points [Longuet-Higgins 1981]

Slide credit: Kristen Grauman

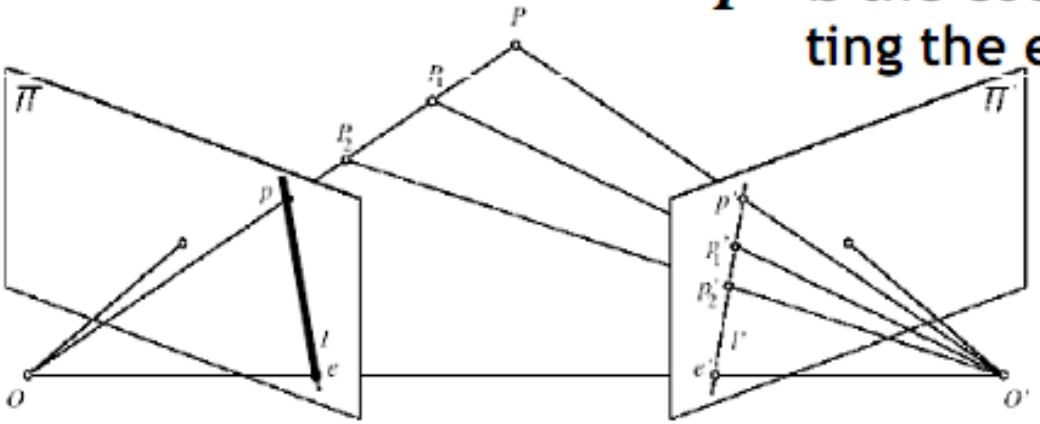
Essential Matrix and Epipolar Lines

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

Epipolar constraint: if we observe point p in one image, then its position p' in second image must satisfy this equation.

$l' = Ep$ is the coordinate vector representing the epipolar line for point p

(i.e., the line is given by: $l'^{\mathrm{T}}\mathbf{x} = 0$)



$l = E^T p'$ is the coordinate vector representing the epipolar line for point p'

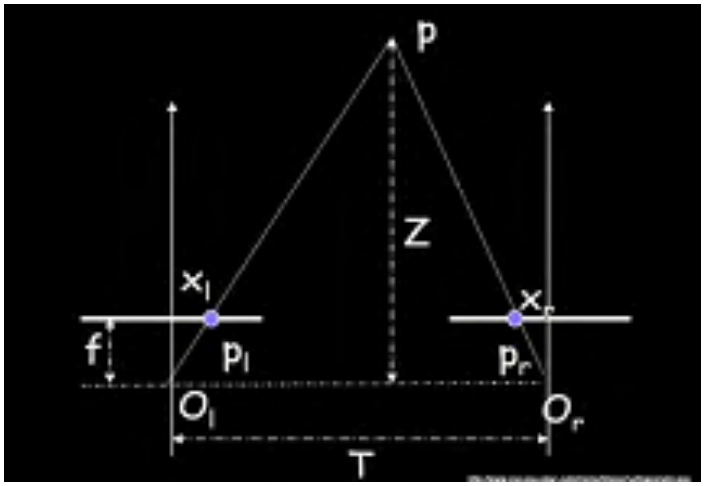
Slide credit: Kristen Grauman

Essential Matrix: Properties

- Relates image of corresponding points in both cameras, given rotation and translation.
- Assuming intrinsic parameters are known

$$\mathbf{E} = \left[\mathbf{T}_{\times}\right]\mathbf{R}$$

Essential Matrix Example: Parallel Cameras



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

$$\mathbf{T} = [-d, 0, 0]^{\mathrm{T}}$$

$$\mathbf{E} = [\mathbf{T}_x]\mathbf{R} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & d \\ 0 & -d & 0 \end{pmatrix}$$

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

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

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

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

$$\Leftrightarrow y = y'$$

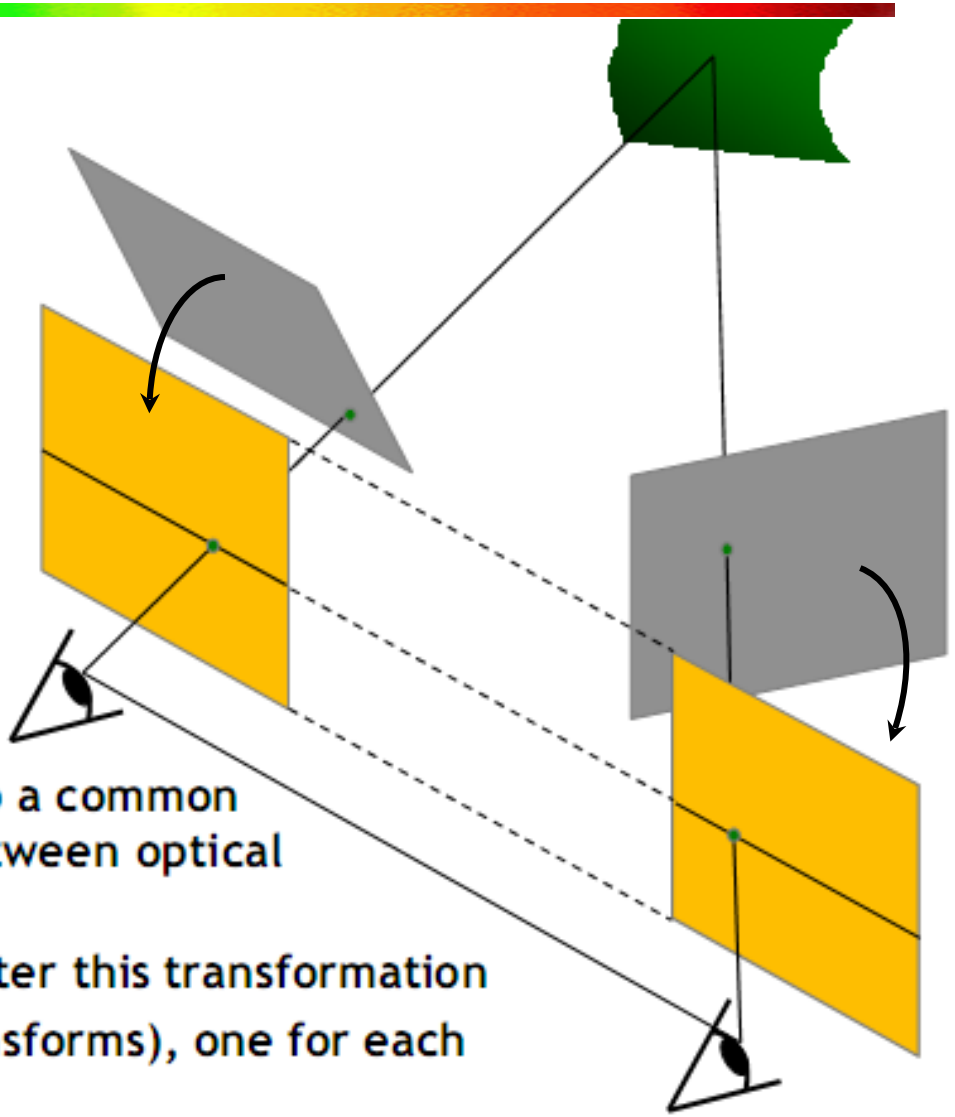
Image I(x,y) | Disparity map D(x,y) | Image I´(x´,y´)

$$(x´,y´)=(x+D(x,y),\ y)$$

- What about when cameras' optical axes are not parallel?

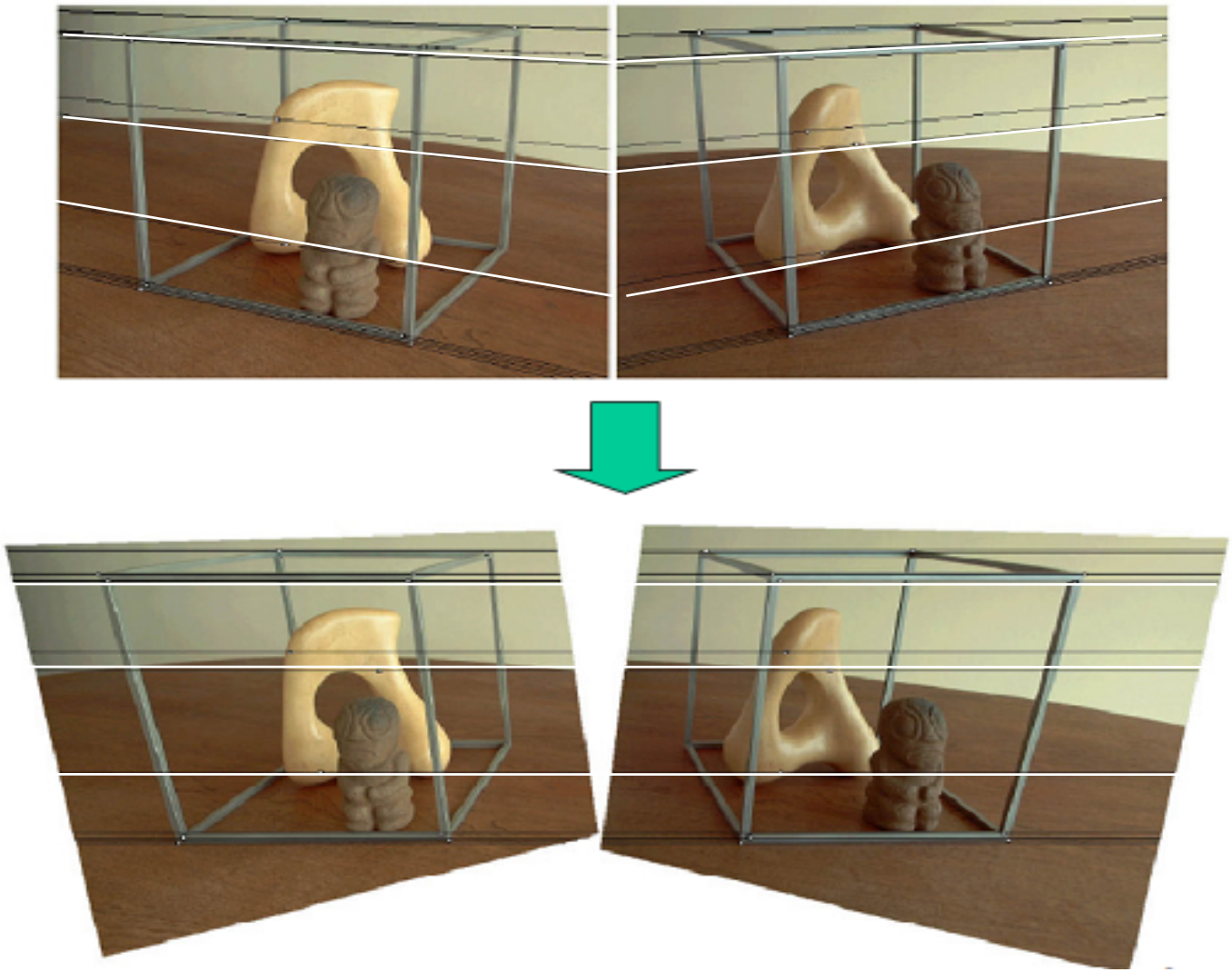
Stereo Image Rectification

- In practice, it is convenient if image scanlines are the epipolar lines.



- Algorithm
  - Reproject image planes onto a common plane parallel to the line between optical centers
  - Pixel motion is horizontal after this transformation
  - Two homographies (3x3 transforms), one for each input image reprojection

Stereo Image Rectification: Example



Source: Alyosha Efros

Stereopsis & 3D Reconstruction

- Correspondence search
- Additional correspondence constraints
- Possible sources of error
- Applications

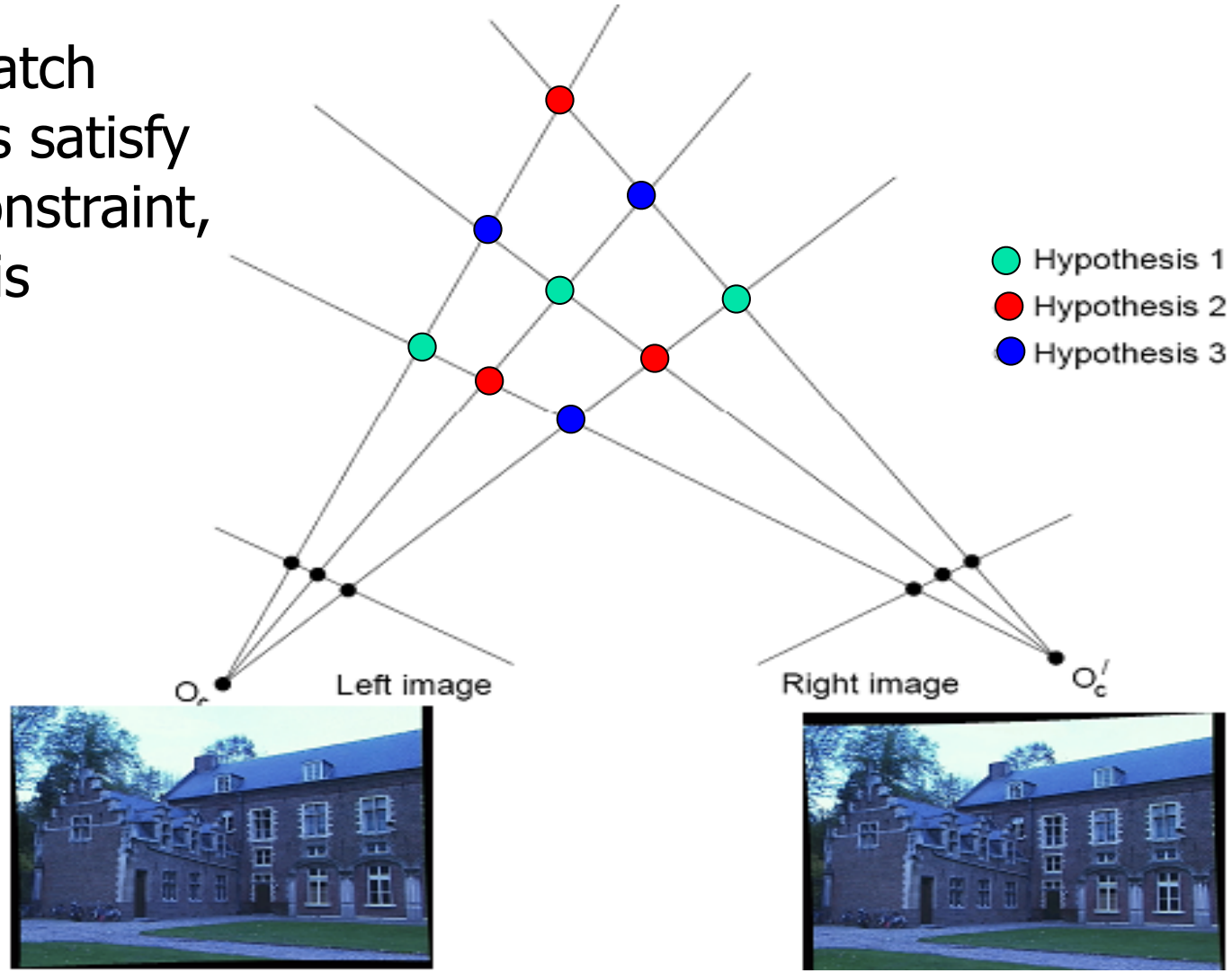
Stereo Reconstruction

- Main Steps
  - Calibrate cameras
  - Rectify images
  - Compute disparity
  - Estimate depth

Slide credit: Kristen Grauman

Correspondence Problem

Multiple match hypotheses satisfy epipolar constraint, but which is correct?



Slide credit: Kristen Grauman

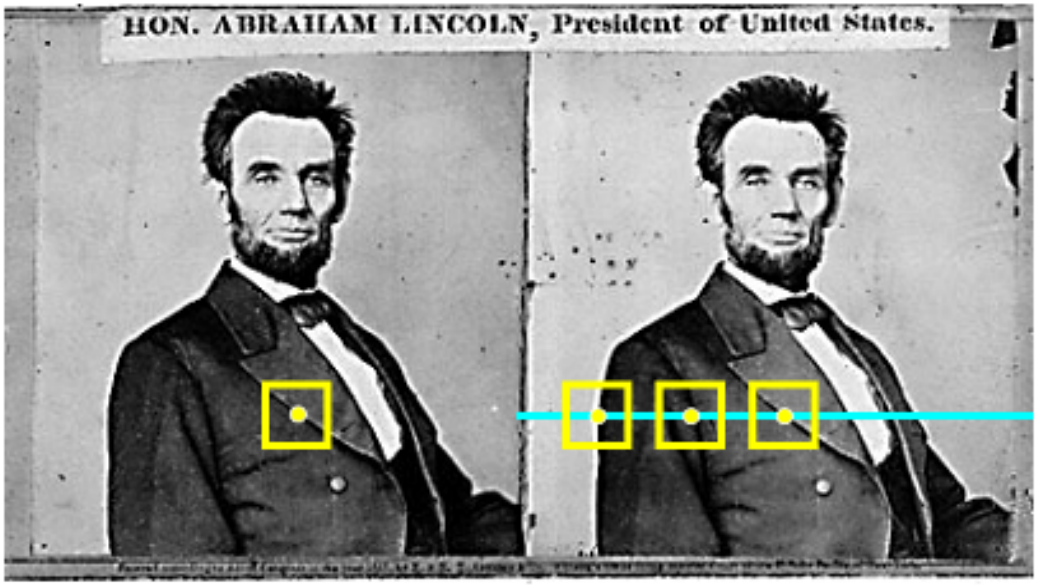
Correspondence Problem

- To find matches in the image pair, we will **assume**
  - Most scene points visible from both views
  - Image regions for the matches are similar in appearance

Additional Correspondence Constraints

- Similarity
- Uniqueness
- Ordering
- Disparity gradient

Dense Correspondence Search



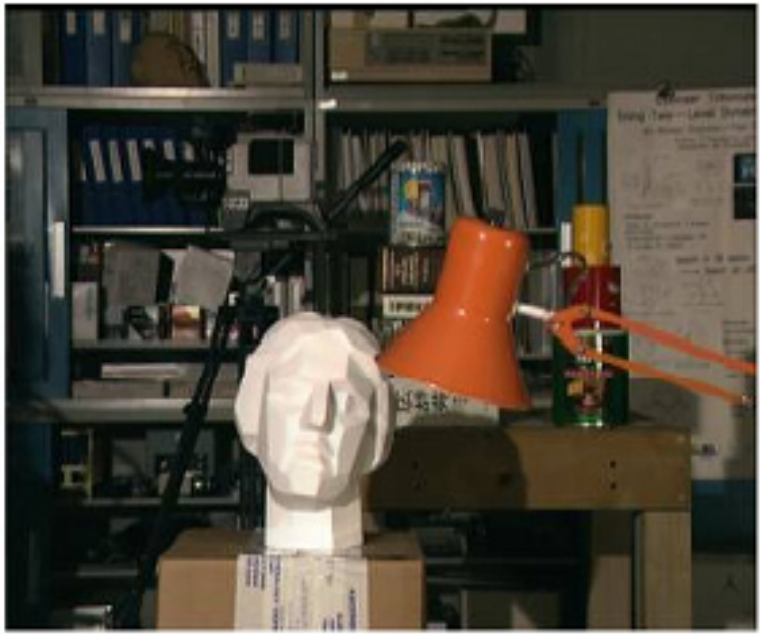
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match (e.g. SSD, correlation)
  - Triangulate the matches to get depth information
- This is easiest when epipolar lines are scanlines

  ⇒ Rectify images first

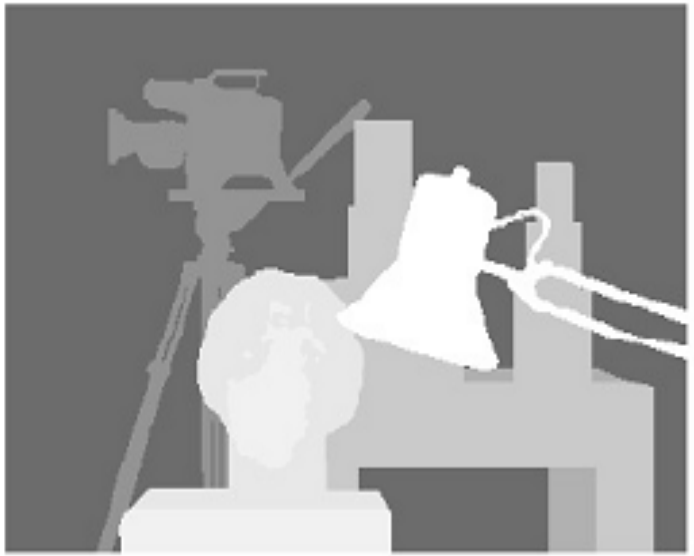
Source: Svetlana Lazebnik, Li Zhang

Example: Window Search

- Data from University of Tsukuba



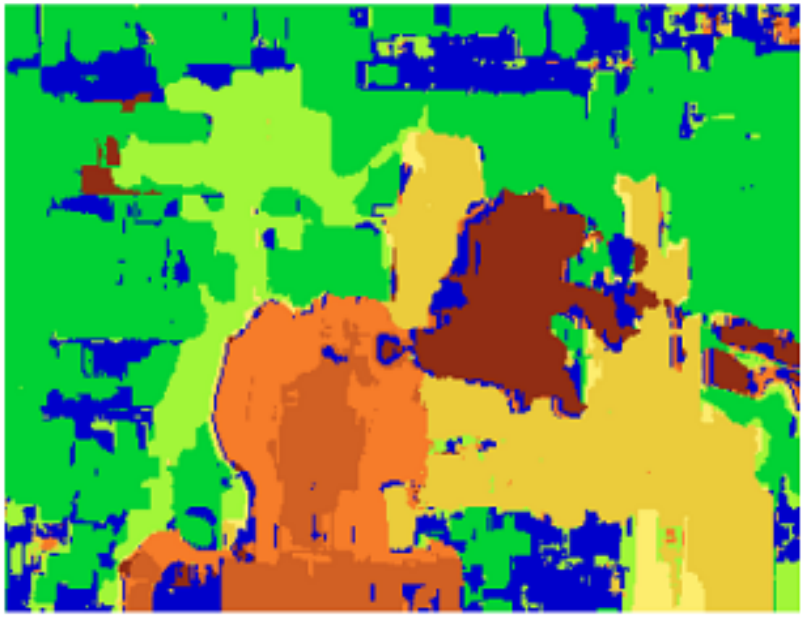
Scene



Ground truth

Example: Window Search

- Data from University of Tsukuba



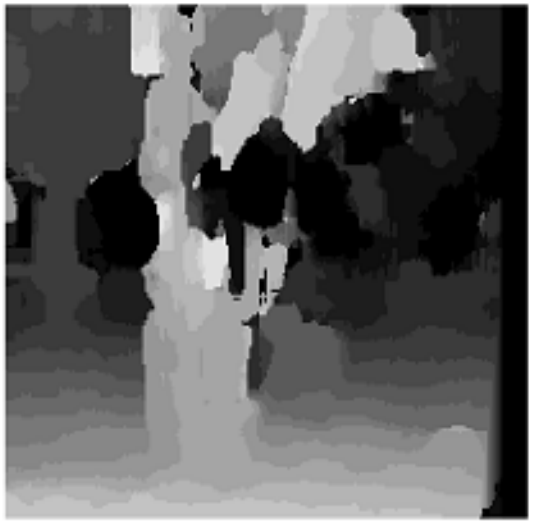
Window-based matching
(best window size)

Ground truth

Slide credit: Kristen Grauman

Effect of Window Size

W = 3

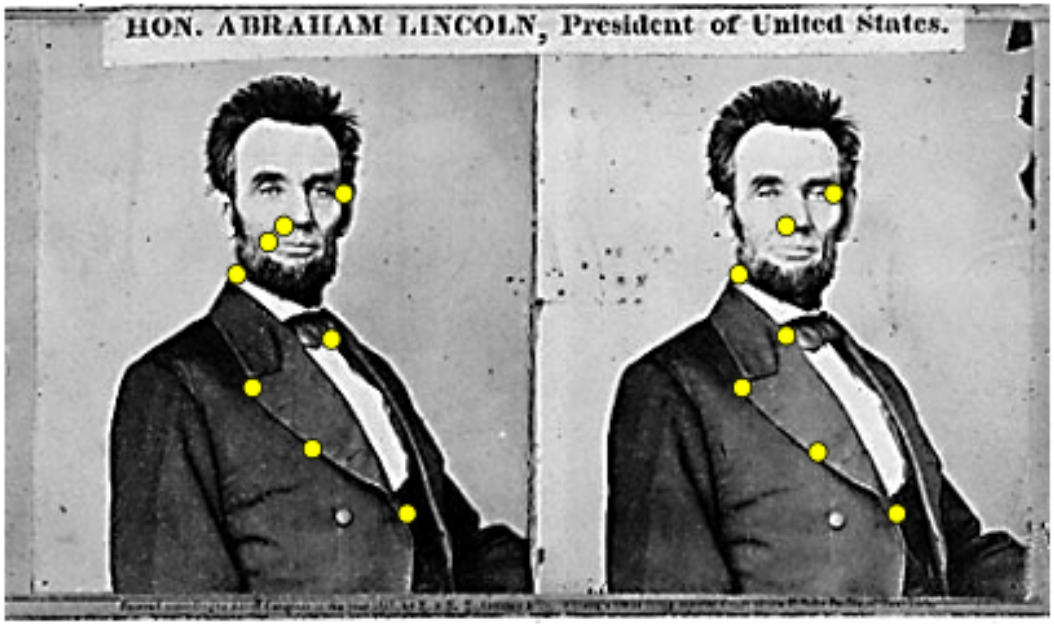


W = 20

Want window large enough to have sufficient intensity variation, yet small enough to contain only pixels with about the same disparity.

Slide credit: Kristen Grauman

Sparse Correspondence Search



- Restrict search to sparse set of detected features
- Rather than pixel values (or lists of pixel values) use *feature descriptor* and an associated *feature distance*
- Still narrow search further by epipolar geometry

What would make good features?

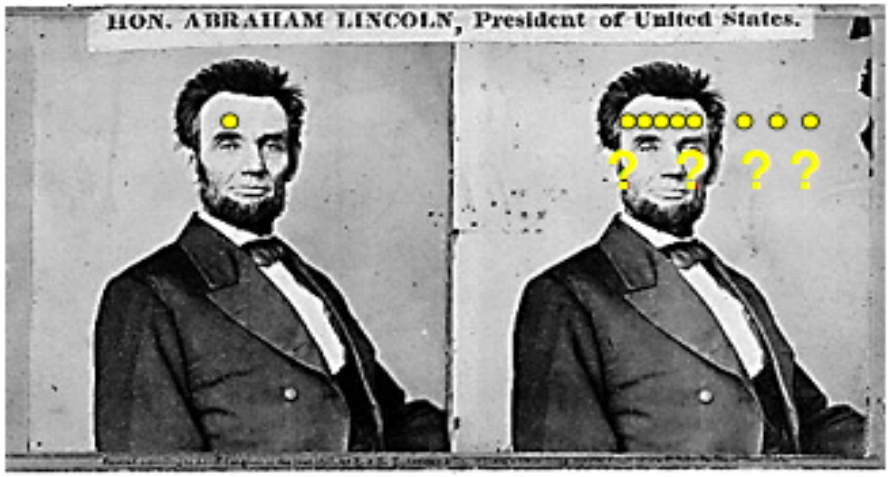
Slide credit: Kristen Grauman

Dense vs. Sparse

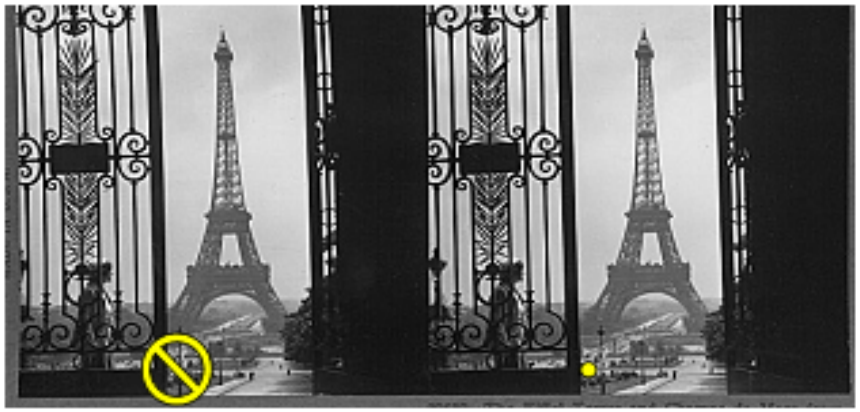
- Sparse
  - Efficiency
  - Can have more reliable feature matches, less sensitive to illumination than raw pixels, But…
  - Have to know enough to pick good features
  - Sparse depth information
- Dense
  - Simple process
  - More depth estimates, can be useful for surface Reconstruction, But…
  - Breaks down in textureless regions anyway
  - Raw pixel distances can be brittle
  - Not good with very different viewpoints

Slide credit: Kristen Grauman

Difficulties in Similarity Constraint



Untextured surfaces



Occlusions

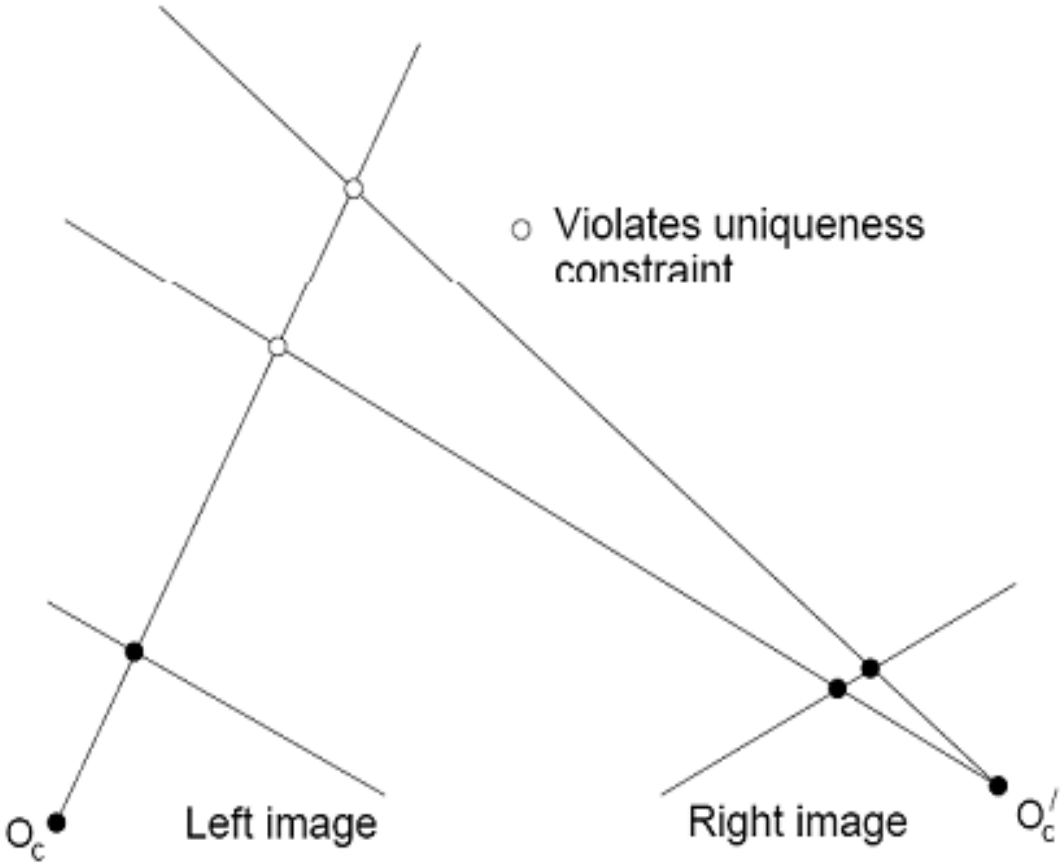
Slide credit: Kristen Grauman

Additional Correspondence Constraints

- Similarity
- Uniqueness
- Ordering
- Disparity gradient

Uniqueness

- For opaque objects, up to one match in right image for every point in left image



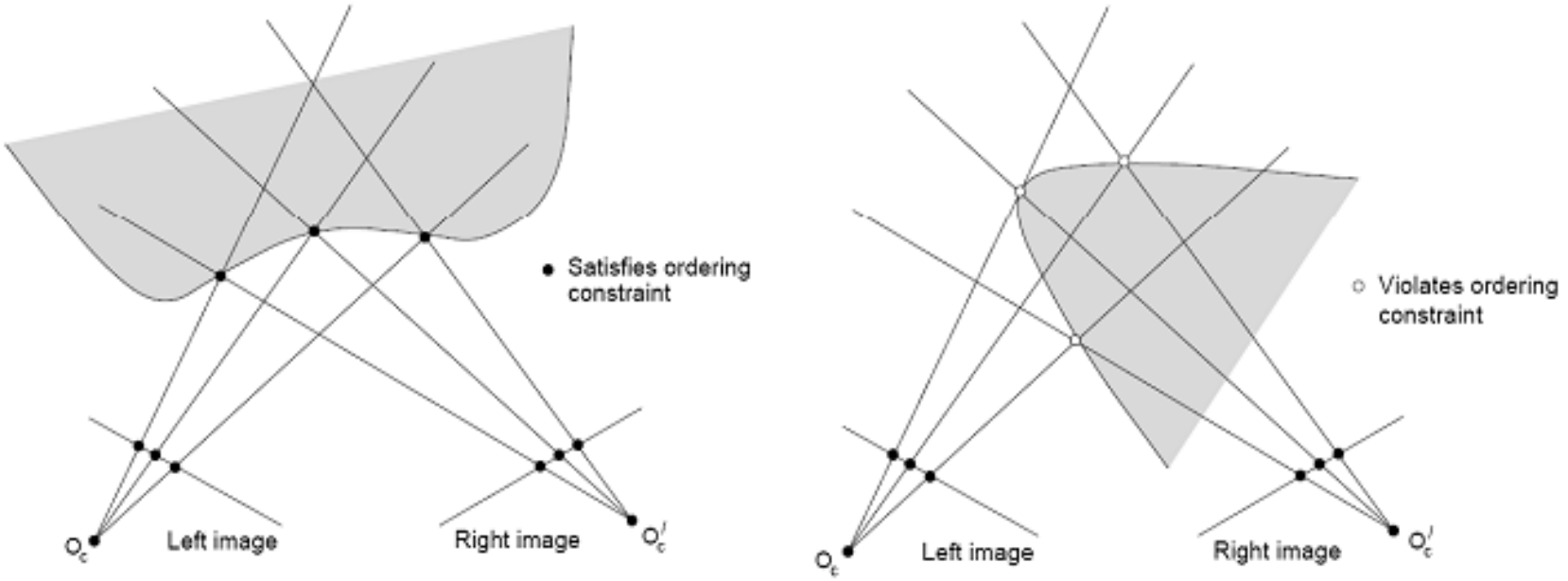
Slide credit: Kristen Grauman

Additional Correspondence Constraints

- Similarity
- Uniqueness
- Ordering
- Disparity gradient

Ordering

- Points on *same surface* (opaque object) will be in same order in both views



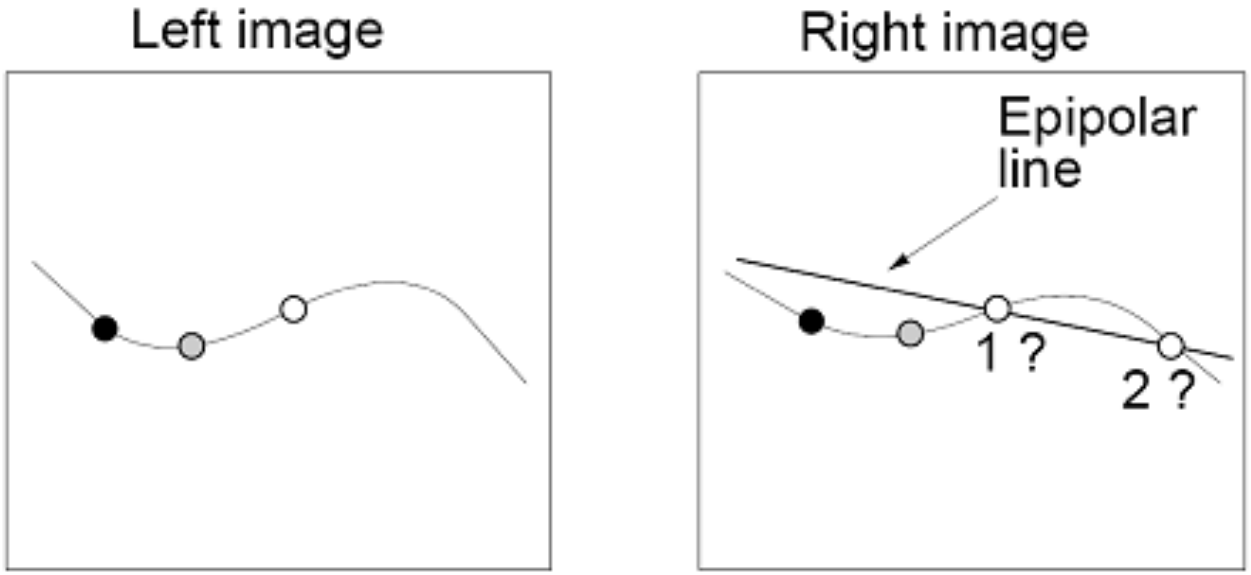
Slide credit: Kristen Grauman

Additional Correspondence Constraints

- Similarity
- Uniqueness
- Ordering
- Disparity gradient

Disparity Gradient

- Assume piecewise continuous surface, so want disparity estimates to be locally smooth



Given matches ● and ◎, point ○ in the left image must match point 1 in the right image. Point 2 would exceed the disparity gradient limit.

Slide credit: Kristen Grauman

Additional Correspondence Constraints

- Similarity
- Uniqueness
- Ordering
- Disparity gradient

- **Epipolar lines constrain**

Possible Sources of Error?

- Low-contrast / textureless image regions
- Occlusions
- Camera calibration errors
- Violations of brightness constancy (e.g., specular reflections)
- Large motions

Summary: Stereo Reconstruction

- Main Steps
  - Calibrate cameras
  - Rectify images
  - Compute disparity
  - Estimate depth

Left

Right

- So far, we have only considered calibrated cameras…

Left

Right

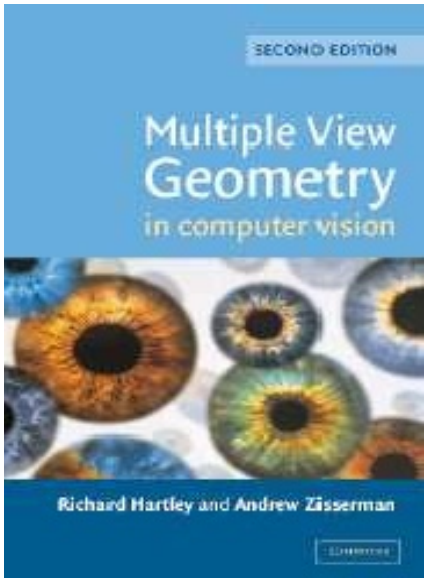
References and Further Reading

- Epipolar geometry and stereopsis：

  D. Forsyth, J. Ponce,Computer Vision – A Modern Approach. Prentice Hall, 2003, Chapters 10.1-10.2 and 11.1-11.3

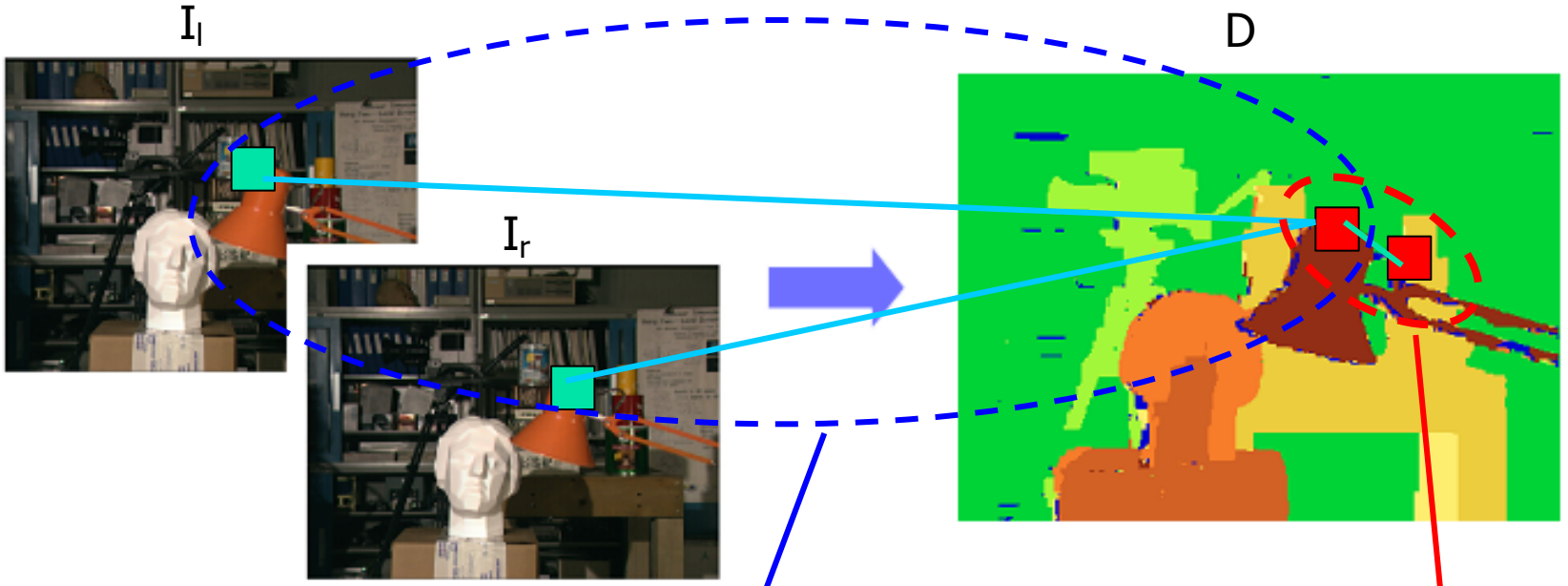
- 3D reconstruction algorithms:

  R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, 2nd Ed., Cambridge Univ. Press, 2004, Chaper 9-10.



MRFs for depth estimation

- **Stereo depth estimation**



$$E(D)=\sum_{\mathbf{p}} C\left(I_l(\mathbf{p}), I_r(\mathbf{p}+D_{\mathbf{p}})\right)+\sum_{(\mathbf{p},\mathbf{q})\in E} S\left(D_{\mathbf{p}}, D_{\mathbf{q}}\right)$$