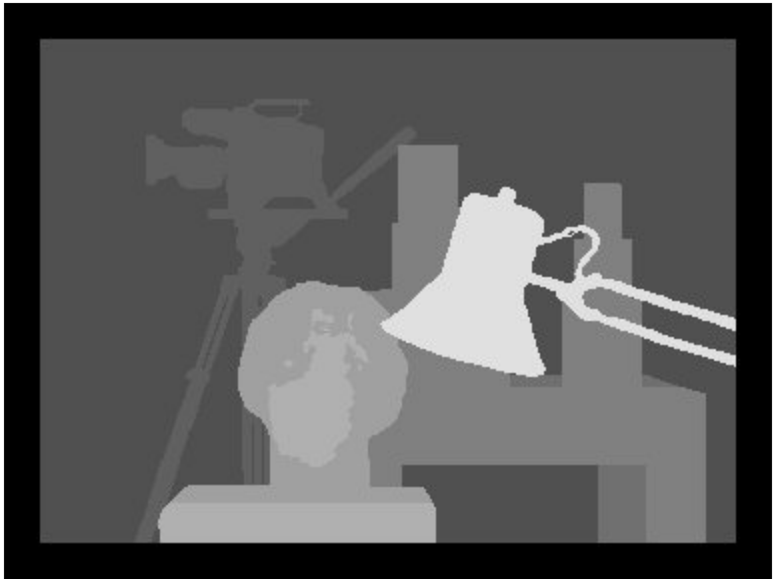
Binocular Stereo

Dr. Elli Angelopoulou

Pattern Recognition Lab (Computer Science 5)

University of Erlangen-Nuremberg

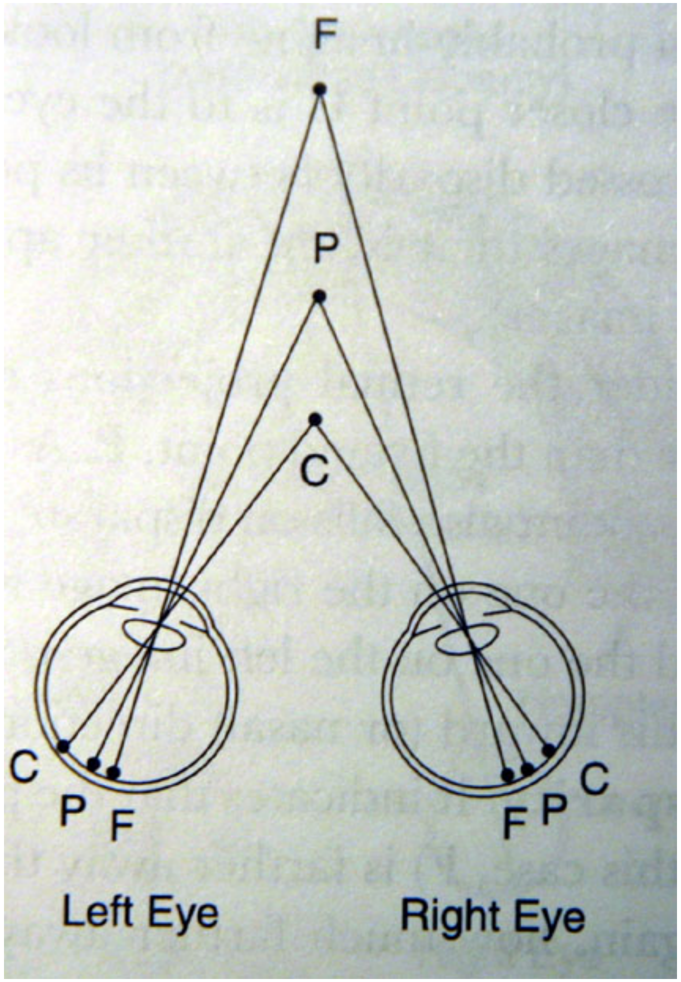
Binocular Stereo Example



Stereo Vision

- Goal: Infer information about the 3-D structure and distances of a scene from two or more images taken from different viewpoints.
- A stereo system must solve two subproblems:
  - Correspondence problem
  - Reconstruction
- *Correspondence Problem*: which pixel (point) on the left image and on the right image are projections of the same scene point.
- Once the point correspondence is established, we can compute the relative shift, the *disparity*, between the two projections.
- *Reconstruction*: The disparity data is then converted to a 3D map. In order to transform the disparity data to 3D measurements, we need some form of knowledge about the geometry of the stereo system.

Depth from Binocular Disparity



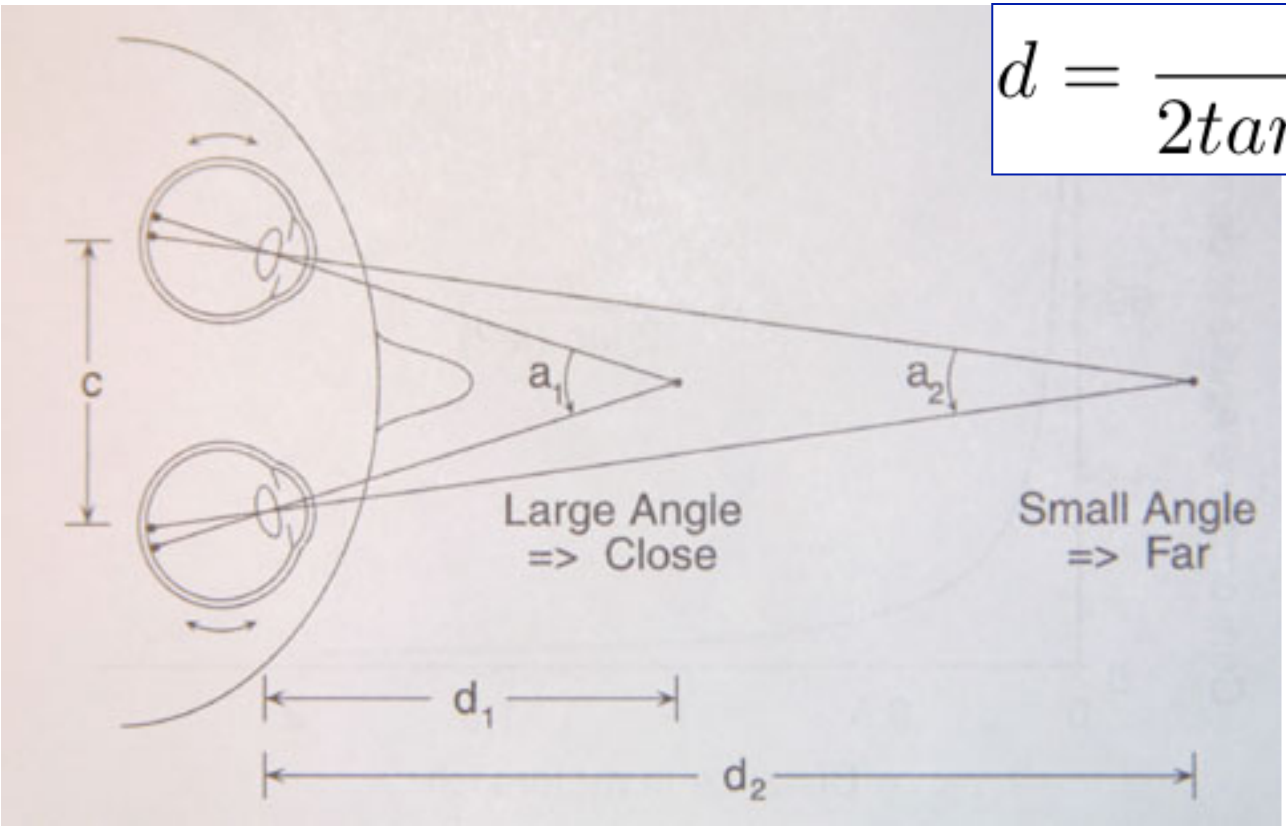
Sign and magnitude of disparity

P: converging point

C: object nearer projects to the outside of the P, disparity = +

F: object farther projects to the inside of the P, disparity = -

Depth from Convergence

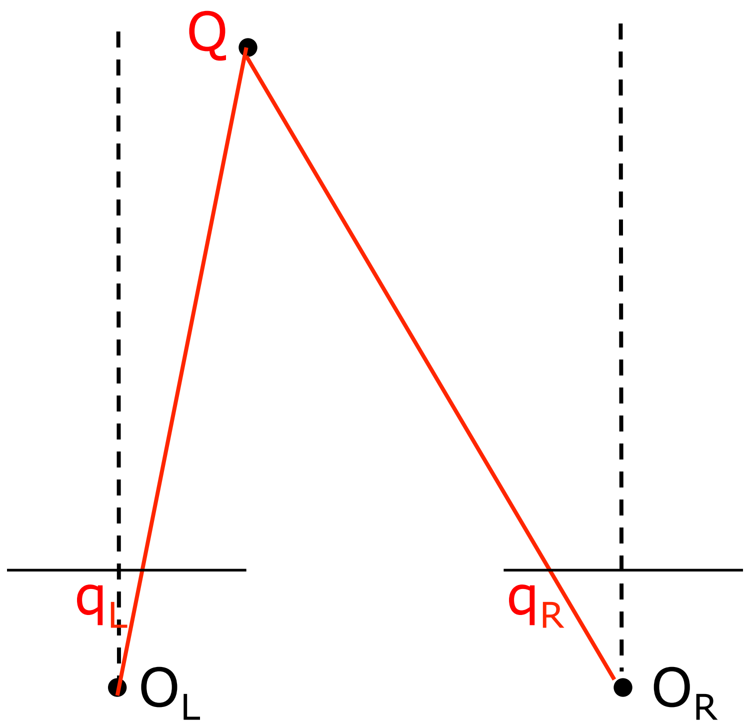


$$d = \frac{c}{2tan(a/2)}$$

Human performance: up to 2-2.5 meters

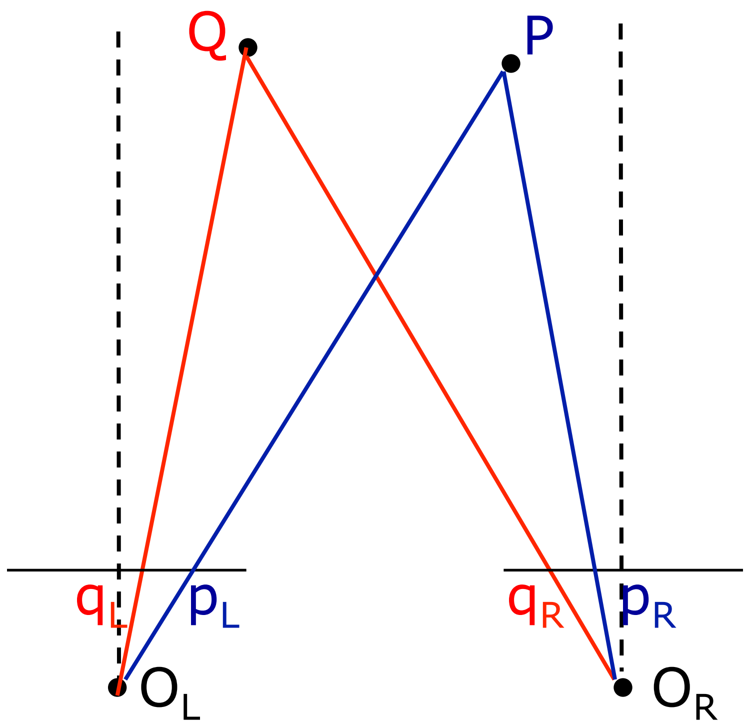
Simple Binocular Stereo Setup

- Parallel optic axes, i.e. the fixation point (the point where the 2 optic axes intersect) is at infinity.
- Both image planes lie on the same plane.
- Their scan lines are aligned (scan-line coherence), i.e. $y_L = y_R$.



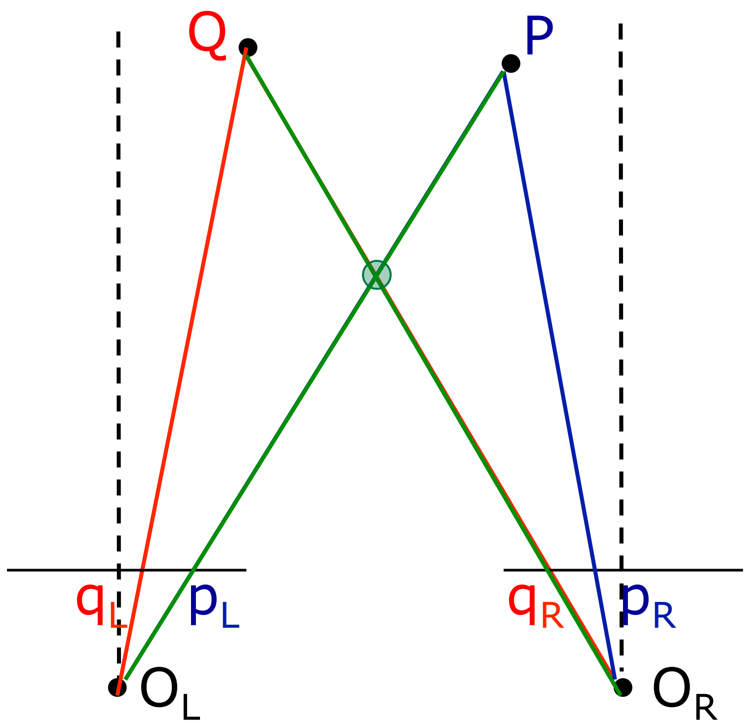
Correspondence and Triangulation

- When we correspond correctly (i.e. q_L with q_R and p_L with p_R), the intersection of the corresponding rays gives the 3D location of scene point that generated the projections (i.e. Q and P accordingly).



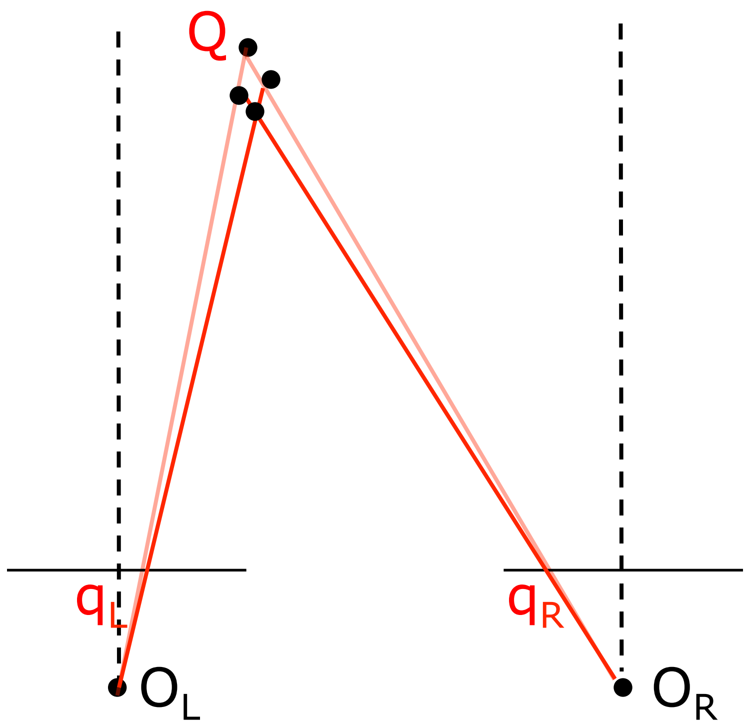
Impact of Correspondence

- A mistake in correspondence, e.g. q_R is matched with p_L, will result in the intersection of rays that correspond to projections of distinct points (Q and P) . As a result the wrong 3D location is recovered.

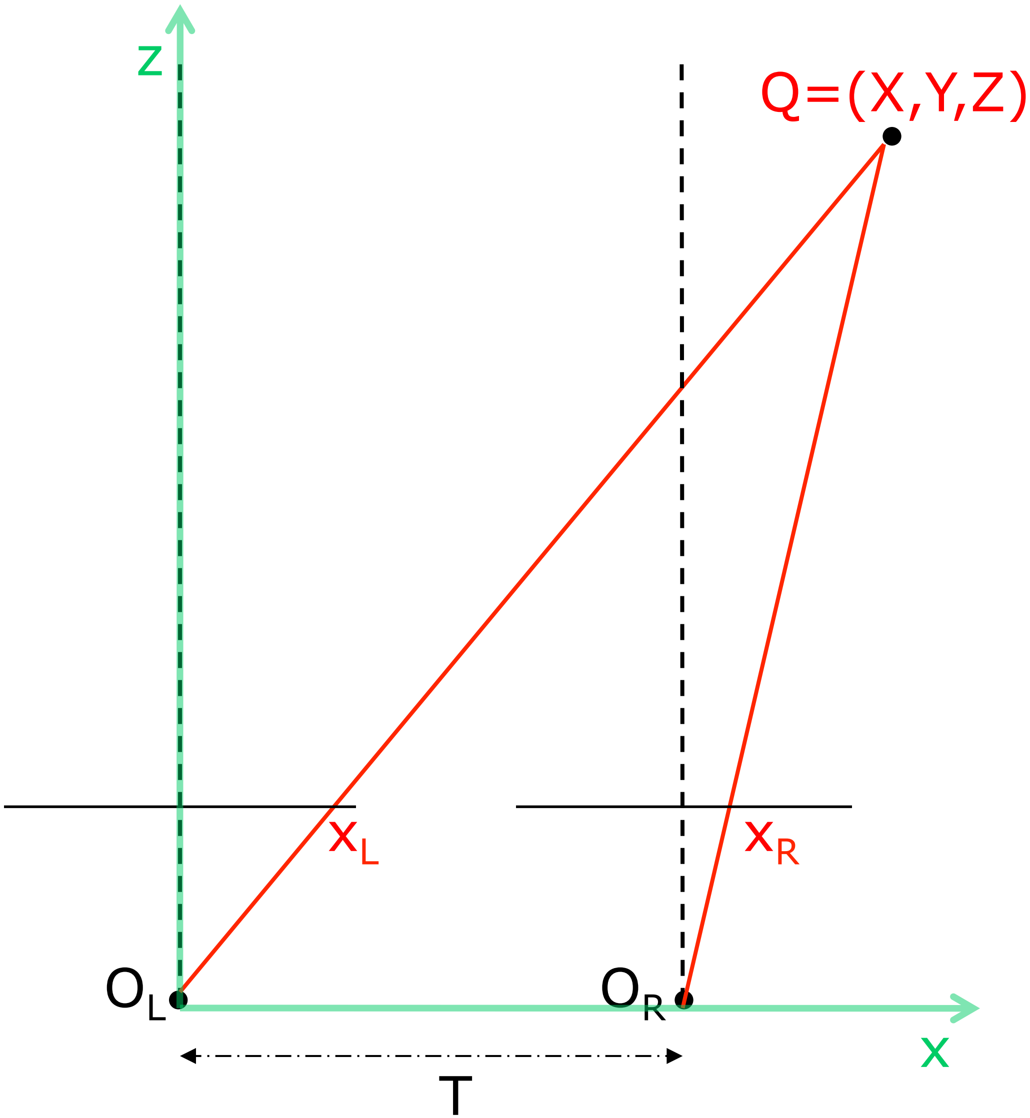


Noise and Correspondence

- The noise in the image capture process (sensor noise, quantization, discretization) introduces inaccuracies in the projection rays that directly affect the triangulation process.

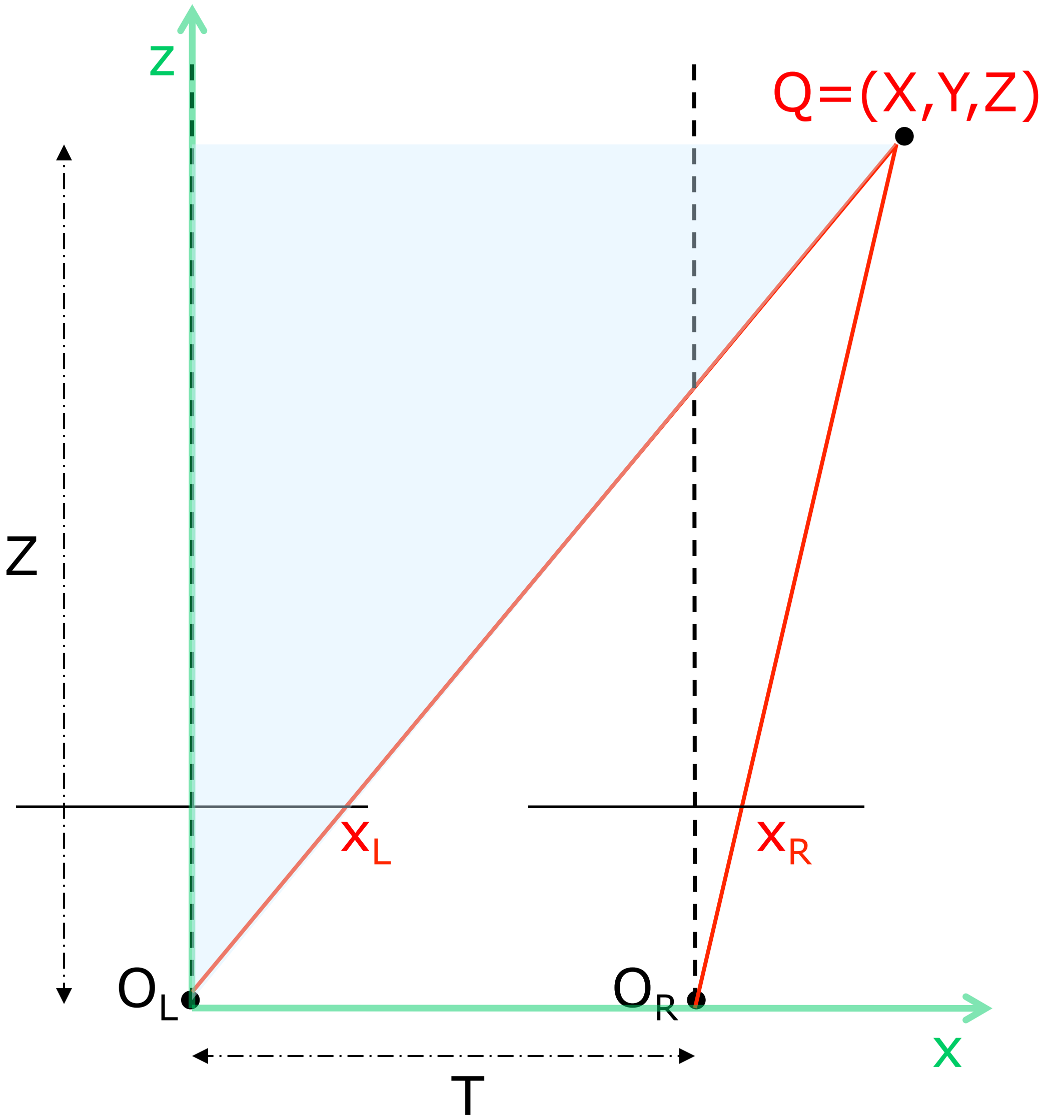


Triangulation

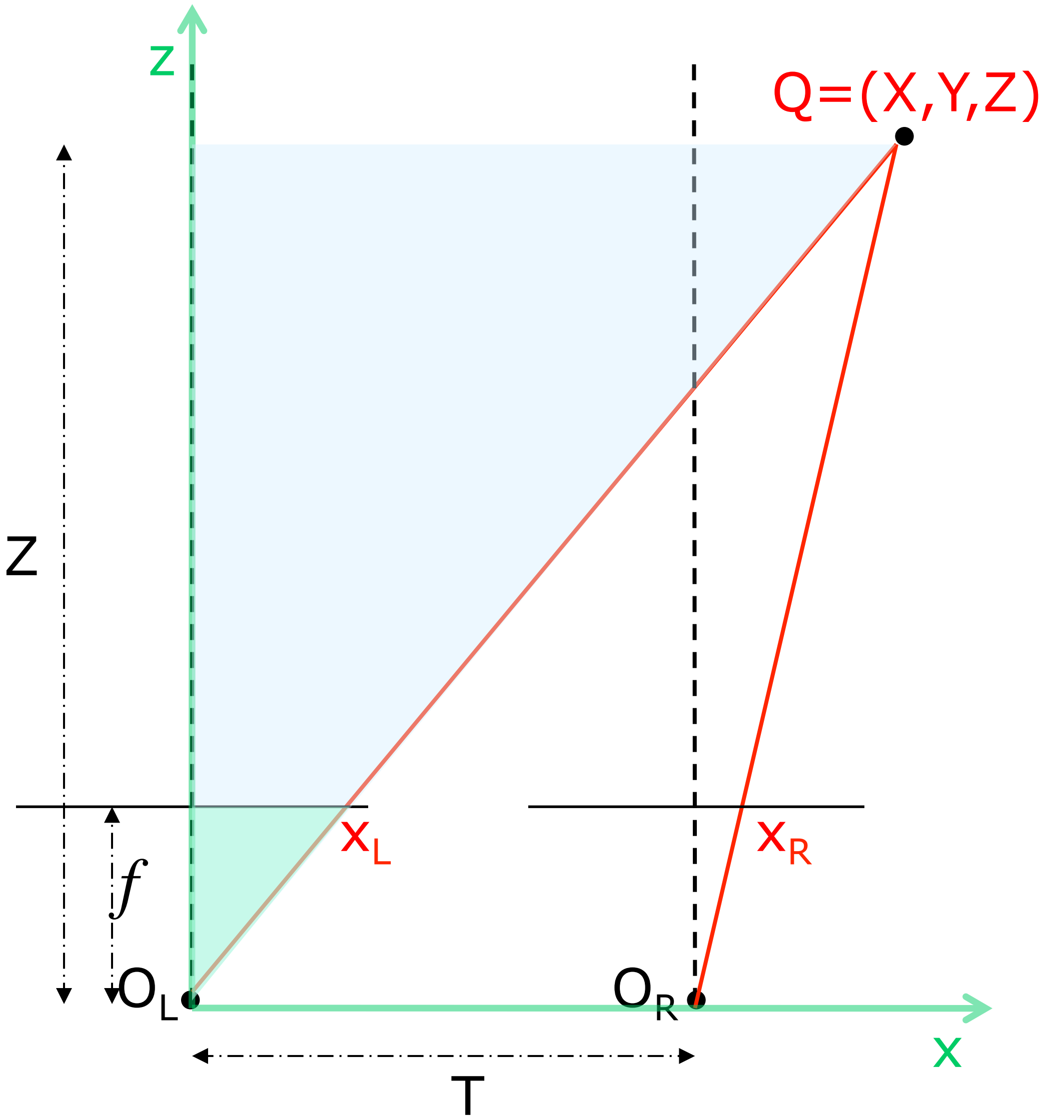


- Assume that the correspondence has been correctly established.
- Under the simple binocular setup (parallel optic axes and scan-line coherence), the only difference between the two projections q_L and q_R is in the x-component, i.e. x_L versus x_R.
- Let T be the **baseline**, i.e. the distance between the two COPs.

Triangulation



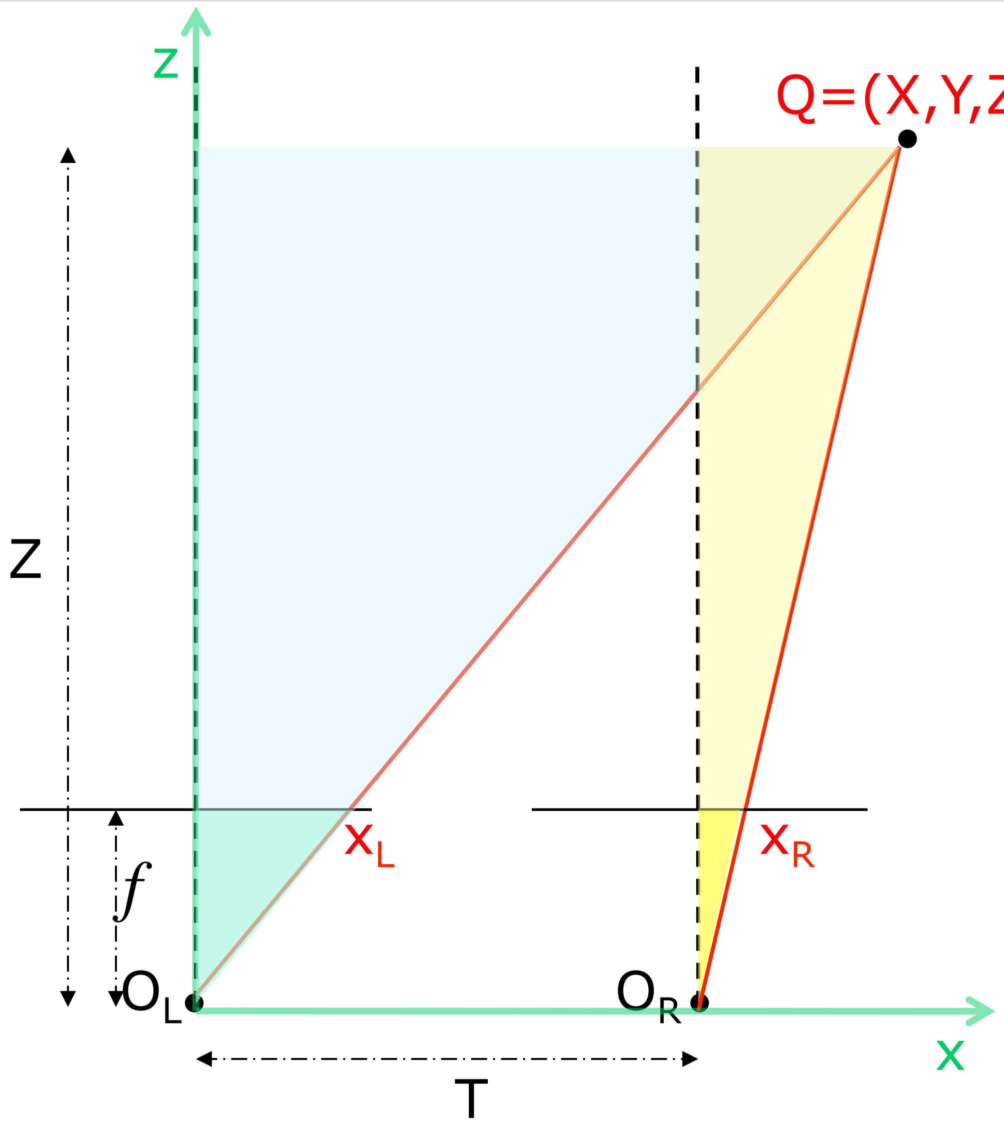
Triangulation



From the similar triangles:

$$\frac{x_L}{f} = \frac{X}{Z} \Rightarrow X = x_L \frac{Z}{f}$$

Triangulation



From the similar triangles:

$$\frac{x_L}{f} = \frac{X}{Z} \Rightarrow X = x_L \frac{Z}{f}$$

From the 2nd set of similar triangles:

$$\frac{x_R}{f} = \frac{X - T}{Z}$$

By replacing X in the 2nd eq.:

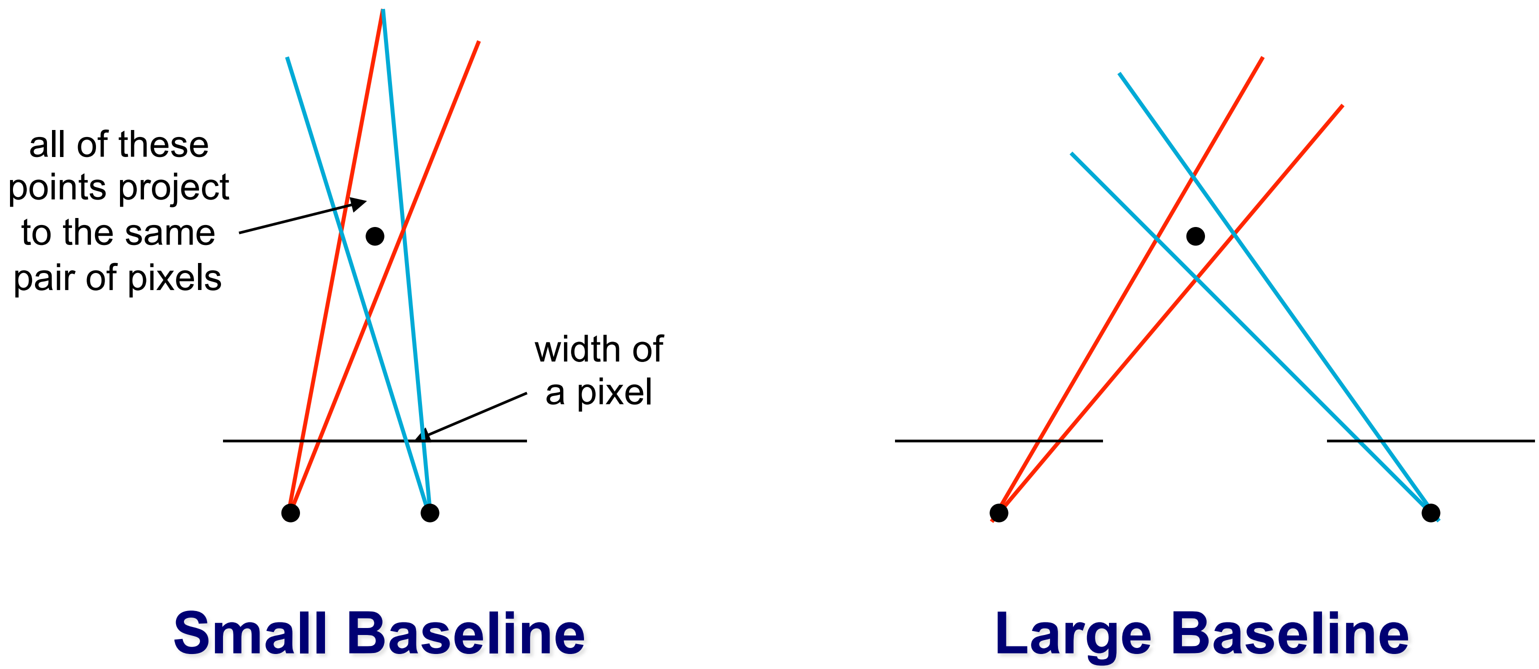
$$\frac{x_R}{f} = \frac{x_L \frac{Z}{f} - T}{Z} \Rightarrow x_R Z = x_L Z - fT$$

$$\Rightarrow \boxed{Z = f \frac{T}{x_L - x_R} = f \frac{T}{d}}$$

where *d* is the *disparity*:

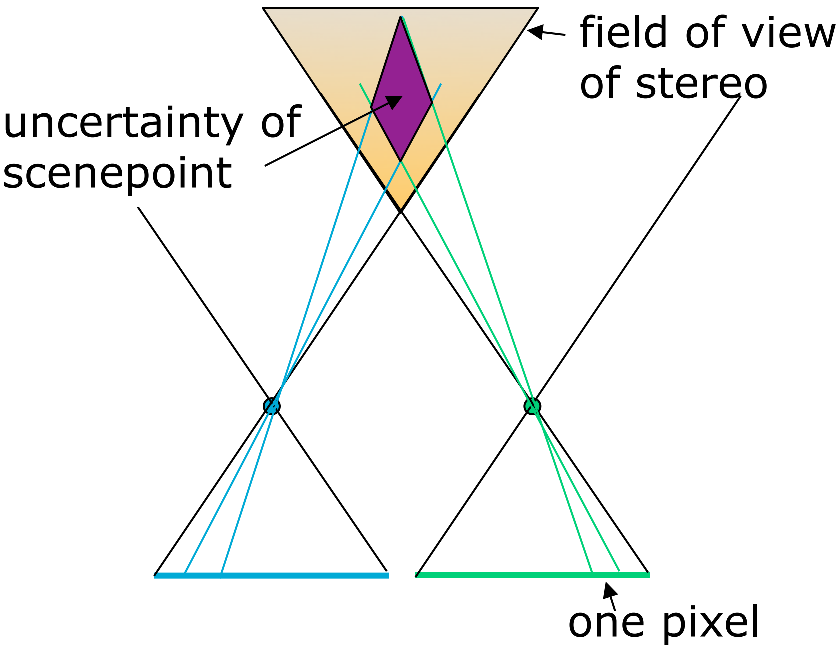
$$d = x_L - x_R$$

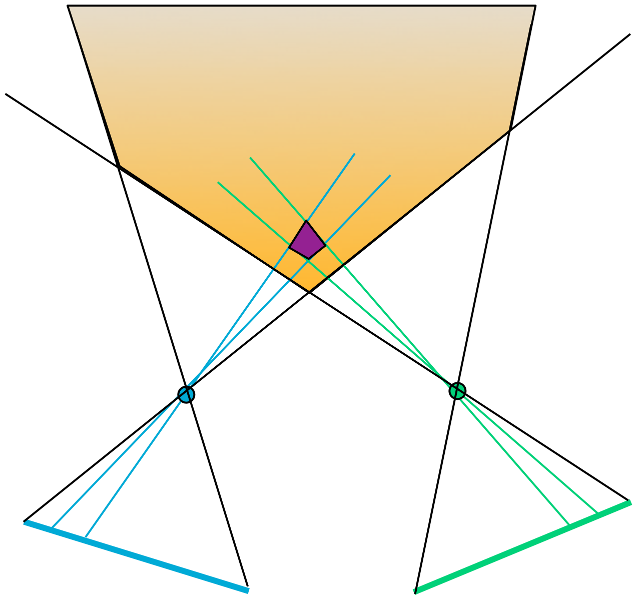
Impact of Baseline



- What's the optimal baseline?
  - Too small: large depth error
  - Too large: difficult search problem
    - Appearance may change between the 2 viewpoints
    - Decrease in the region of the scene that is mutually visible.

Vergence



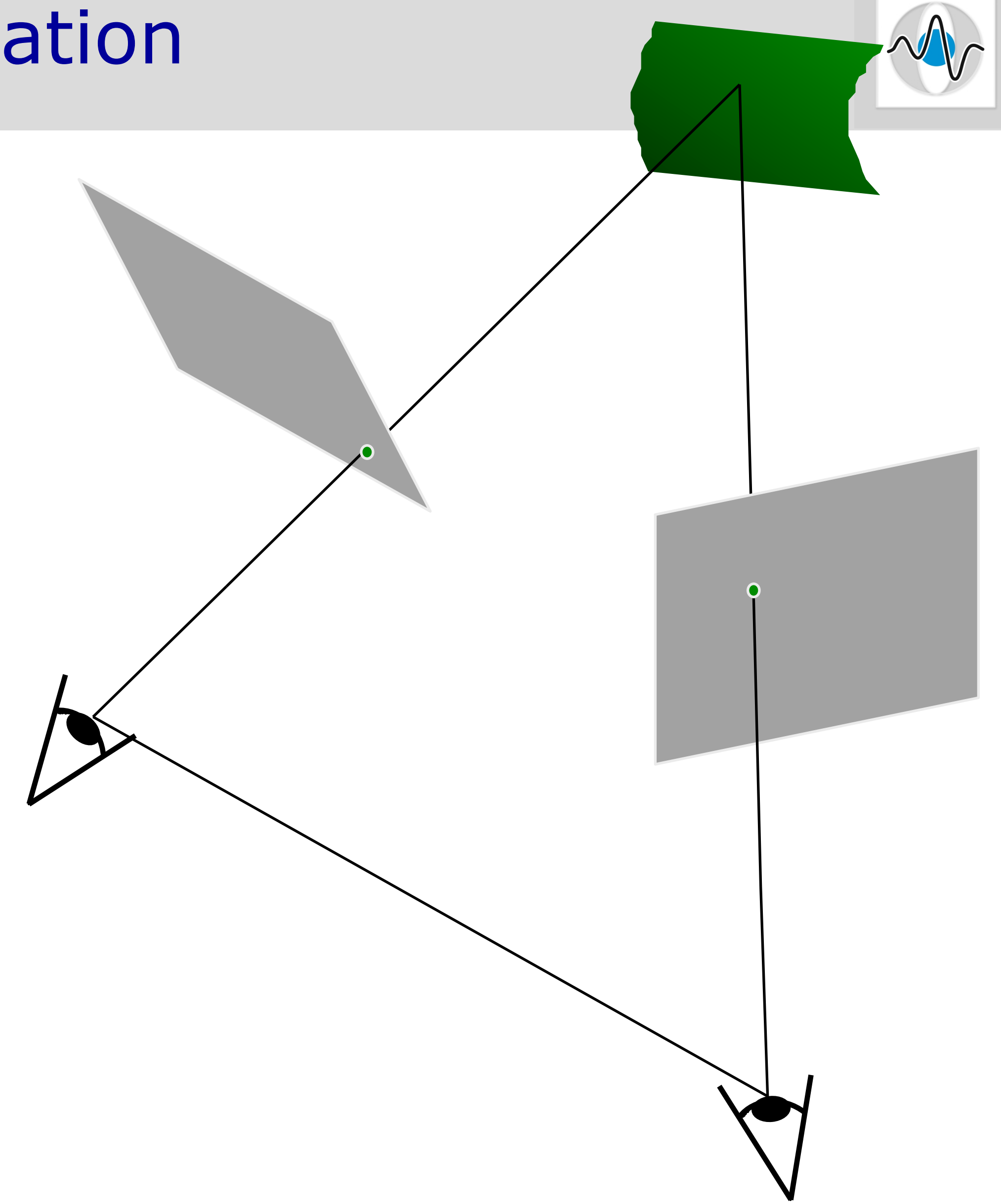


Optical axes of the two cameras need not be parallel

- Solution: Vergence (turn cameras towards each other)
  - Increases the field of view
  - Increases accuracy in the correspondence

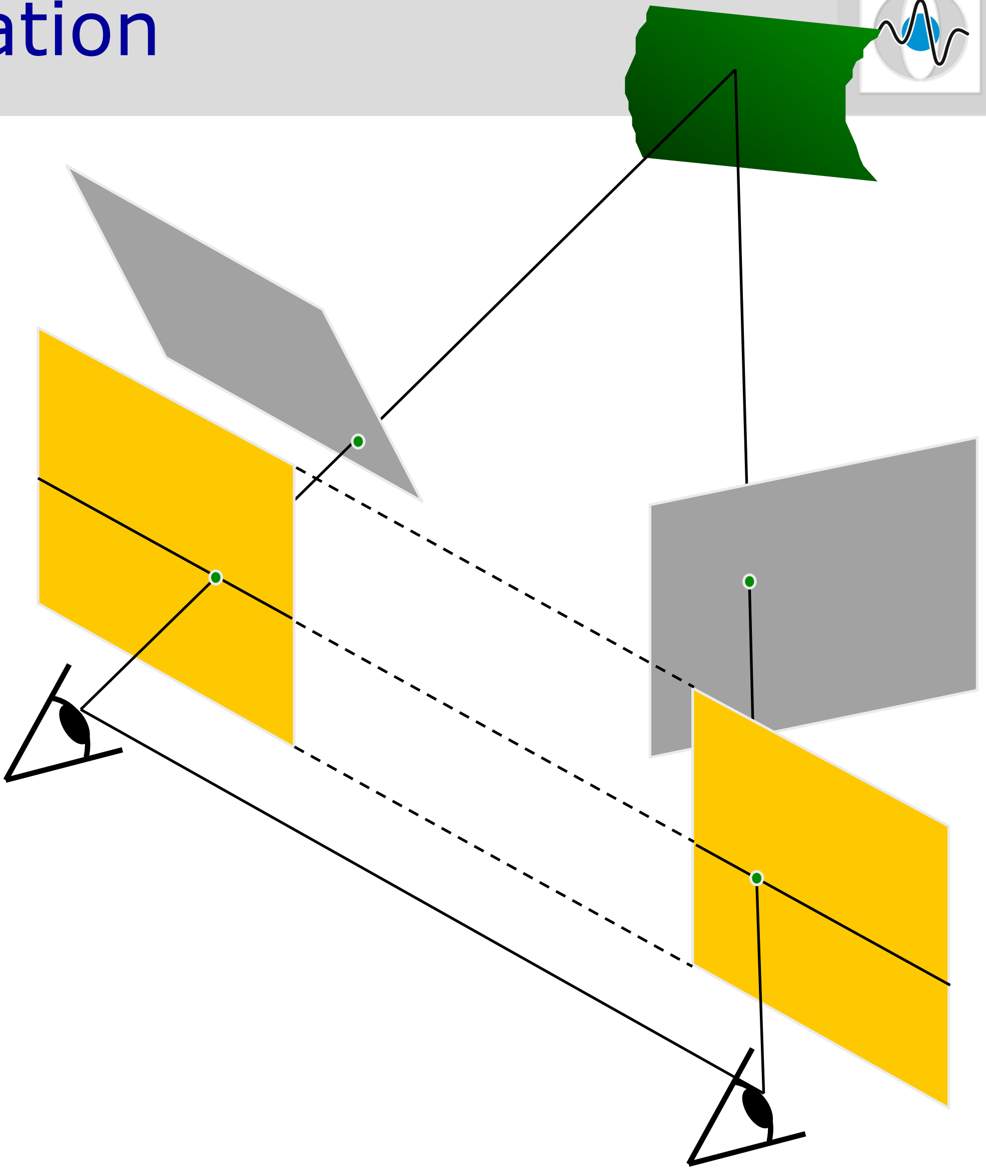
Stereo Image Rectification

- So far we have assumed:
  - parallel optic axes
  - scan-line coherence
- Such a setup can lead to inaccuracies.
- More commonly cameras are *verged*, i.e. the 2 optic axes intersect each other.
- Can we use the same math?

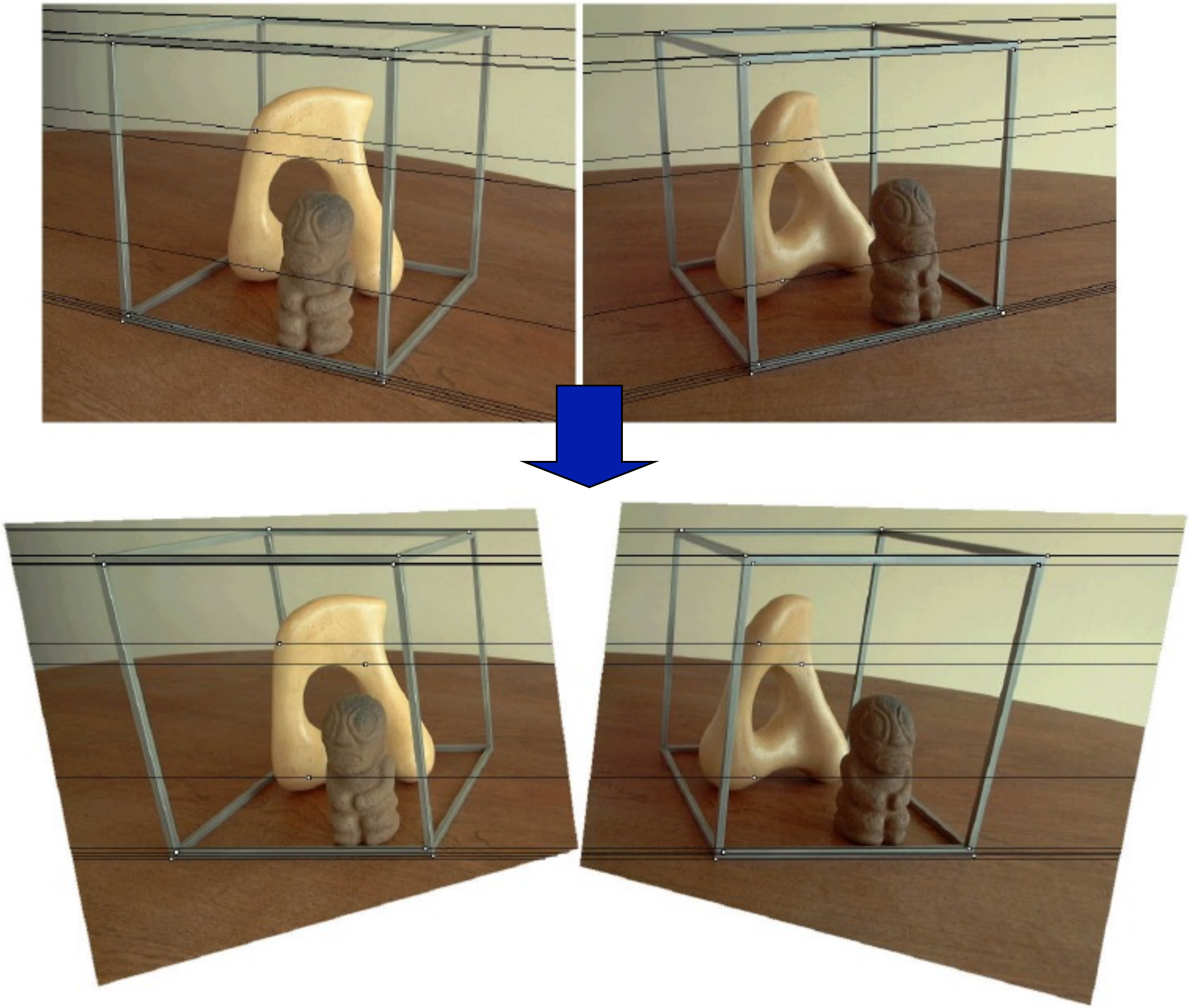


Stereo Image Rectification

- Yes!
- Re-project the image planes onto a common plane parallel to the baseline (the line between the two centers of projection).
- Two virtual image planes are created, which are now scan-line coherent.
- Do all the computations on these rectified (virtual) image planes.



Stereo Rectification Example

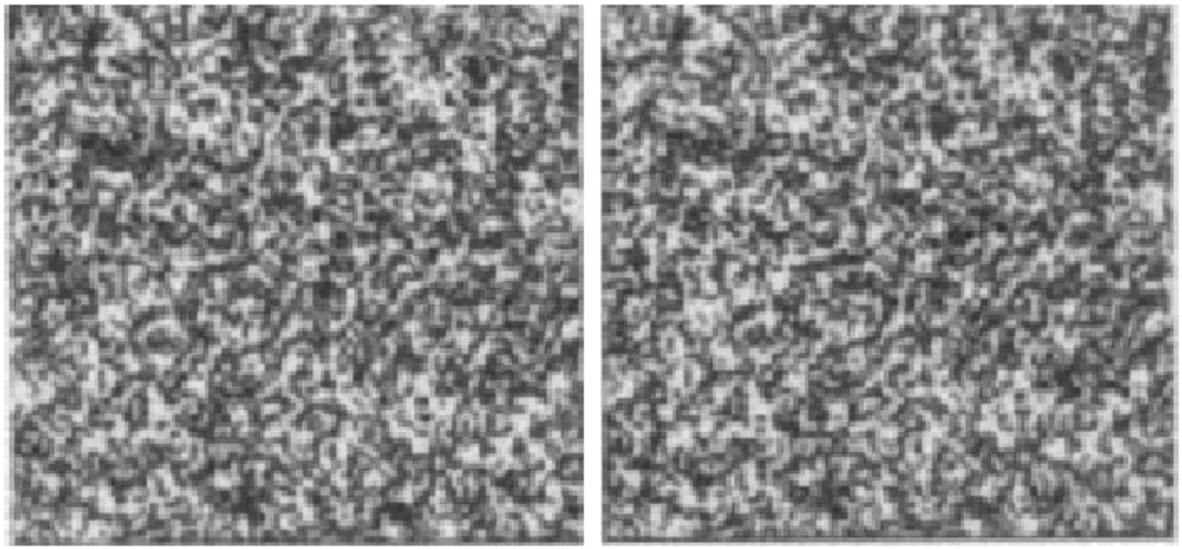


Correspondence Problem

- Assumptions:
  - Most scene points are visible from both viewpoints
  - Corresponding image regions look similar
- It is a search problem: Given an element in the left image, search the right image to find the corresponding element.
- Three underlying questions:
  1. What do we match between the two images? (objects, edges, pixels, sets of pixels?)
  2. What measure of similarity do we use?
  3. Can we search in a systematic way?

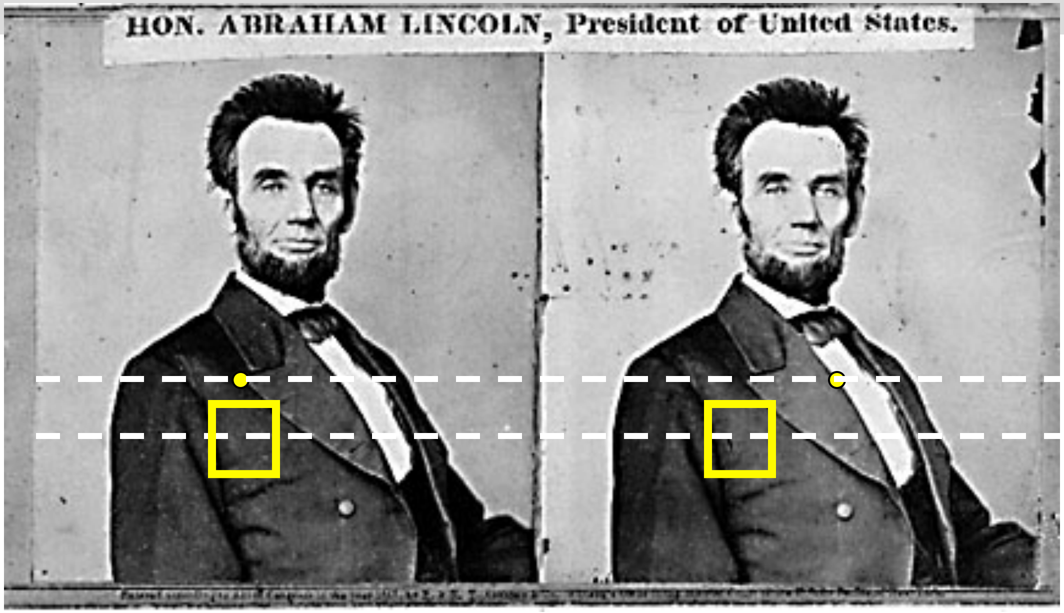
Point Correspondence

Random dot stereograms



Julesz: had huge impact because it showed that recognition not needed for stereo.

Point Correspondence in Practice



For each scan-line (more properly epipolar line)

For each pixel in the left image

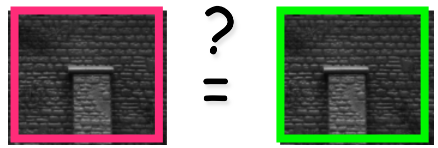
- compare with every pixel on same epipolar line in right image
- pick pixel with closest intensity value (or more general minimum match cost).
- This will never work, so:

Improvement: match ***windows***

Compare Regions around Points

- Idea: Compare intensity profiles around neighborhoods of potential points.
- Elements to be matched are now image windows of fixed size.
- The similarity measure is the correlation between windows in the two images.

Similarity Metrics



f g

$$SSD = \sum_{[i,j]\in R} (f(i,j) - g(i,j))^2$$

$$C_{fg} = \sum_{[i,j]\in R} f(i,j)g(i,j)$$

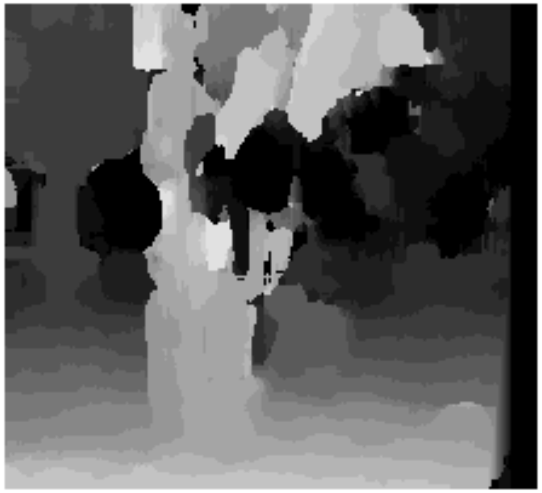
$$NC_{fg} = \frac{1}{n-1}\sum_{[i,j]\in R} \frac{(f(i,j)-\bar{f})(g(i,j)-\bar{g})}{\sigma_f \sigma_g}$$

Most popular

For each window, match to the closest window on the horizontal (epipolar) line in the other image.

Window Size

W = 3

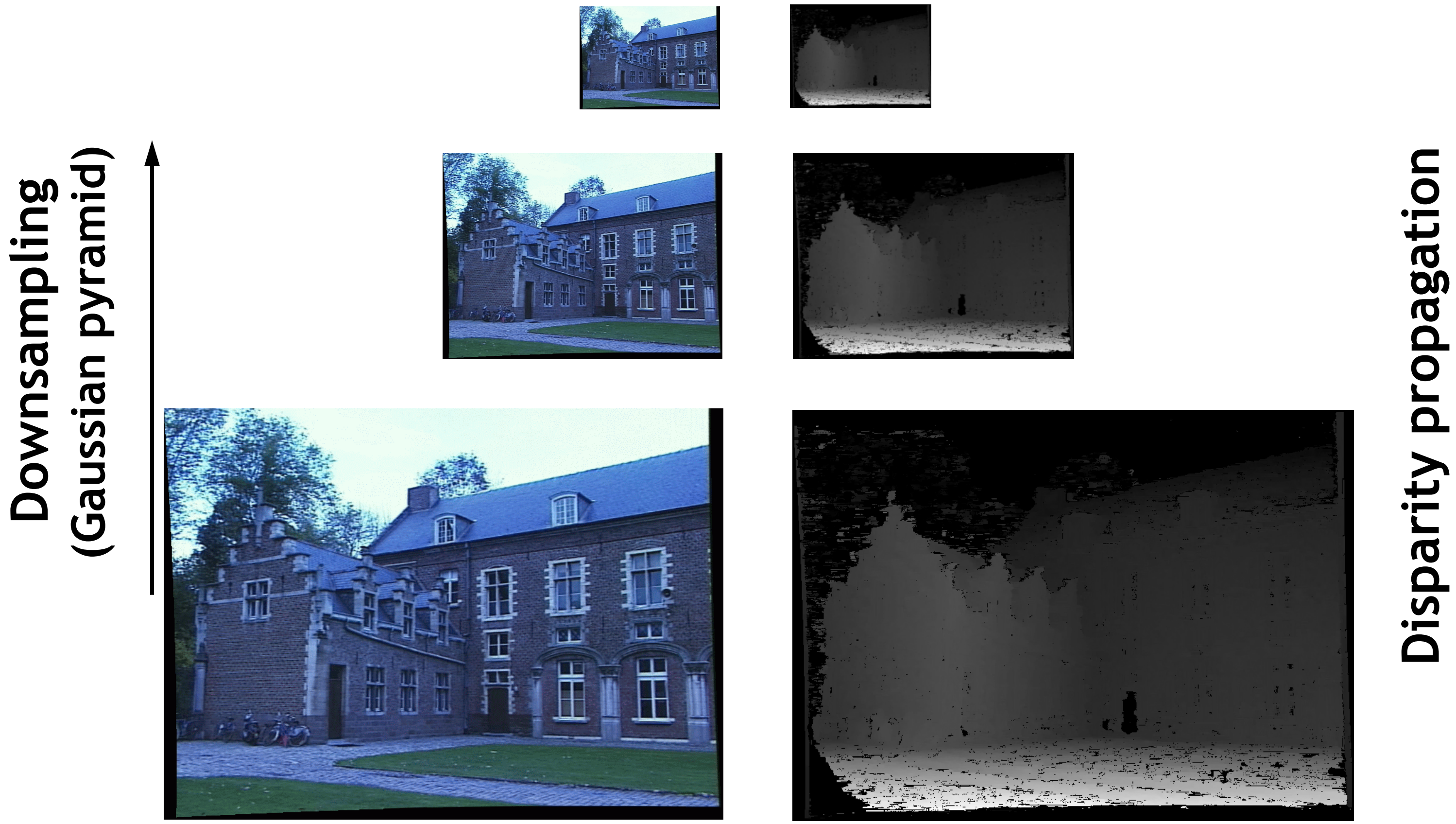


W = 20

- Smaller window: more detail, more noise.
- Larger window: less noise, less detail
- Better results with adaptive window size

Hierarchical Correspondence

- Allows faster computation
- Can handle large disparity ranges



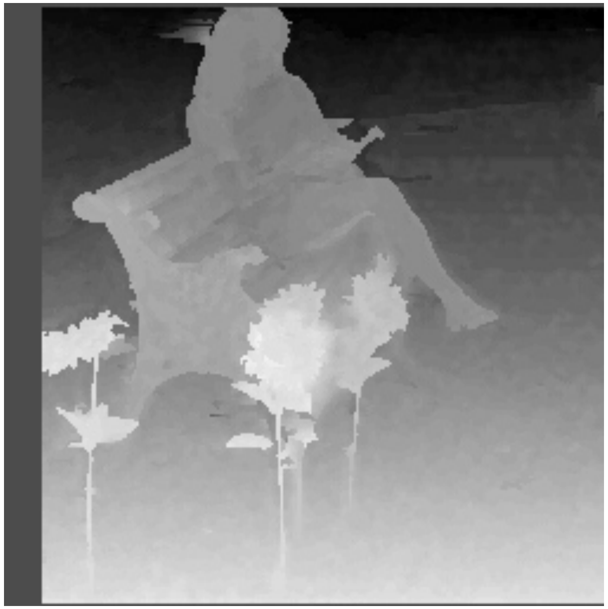
Compare Features

- Another Idea: Compute features and match only pixels based on their feature values.
- Possible features:
  - Edges
  - Lines…
- Pros: Possibly more unique values => easier correspondence
- Cons: Not all the pixels have a feature value => sparse correspondence; need for interpolation
- Often used in combination with hierarchical correspondence.

Stereo Example


left image


right image


depth map

H. Tao et al. "Global matching criterion and color segmentation based stereo"

Reconstruction

H. Tao et al. "Global matching criterion and color segmentation based stereo"

Image Sources

1. The slides on image rectification are courtesy of J. Chai, http://faculty.cs.tamu.edu/jchai/cpsc641_spring10/lectures/lecture9.ppt
2. A number of slides in this presentation have been adapted by the presentation of S. Narasimhan, http://ww.cs.cmu.edu/afs/cs/academic/class/15385-s06/lectures/ppts/lec-14.ppt
3. The Lincoln image is courtesy of S. Seitz.
4. The window-matching slide is courtesy of O. Camps.
5. The example slide on hierarchical correspondence algorithms is courtesy of ETH, http://www.inf.ethz.ch/personal/pomarc/courses/gcv/class07.ppt