

CS 534: Computer Vision Stereo Imaging

Spring 2004
Ahmed Elgammal
Dept of Computer Science
Rutgers University

CS 534 – Stereo Imaging - 1

Talk

- Visitor: Dr. Sing Bing Kang Microsoft Research
- Talk next Monday at 2:30pm.
- Everybody in the class is **REQUIRED** to attend as if it is a regular lecture.

CS 534 – Stereo Imaging - 2

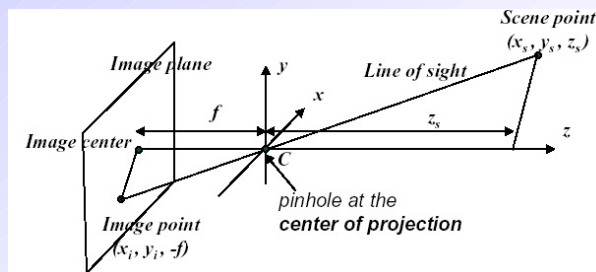
Outlines

- Depth Cues
- Simple Stereo Geometry
- Epipolar Geometry
- Stereo correspondence problem
- Algorithms

Recovering the World From Images

We know:

- 2D Images are projections of 3D world.
- A given image point is the projection of any world point on the *line of sight*
- So how can we recover depth information



Why to recover depth ?

- Recover 3D structure, reconstruct 3D scene model, many computer graphics applications
- Visual Robot Navigation
- Aerial reconnaissance
- Medical applications



The Stanford Cart, H. Moravec, 1979.



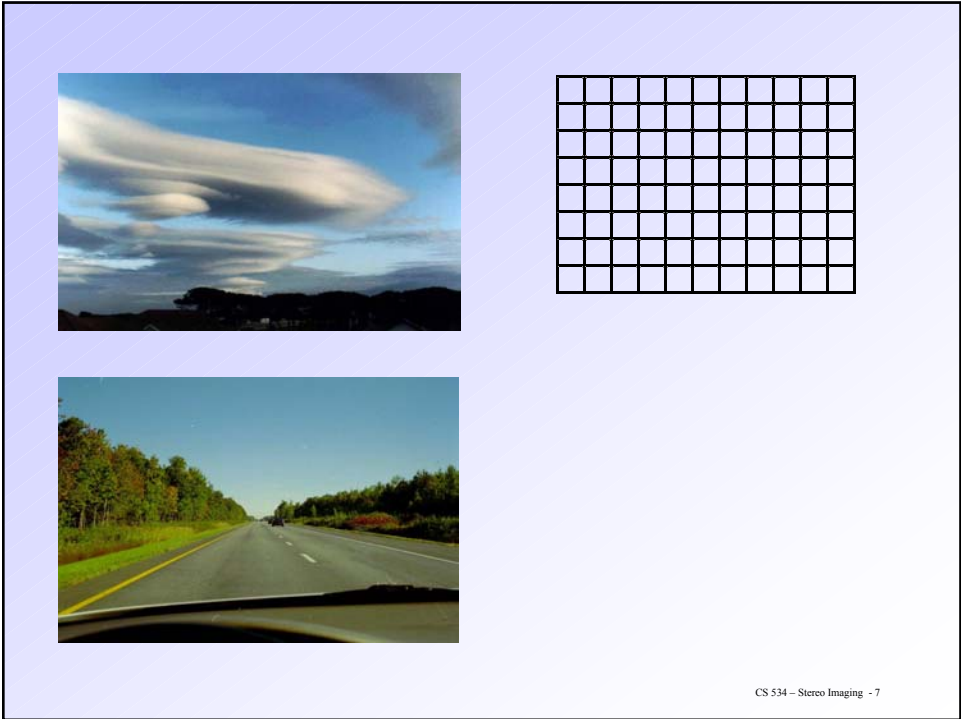
The INRIA Mobile Robot, 1990.

CS 534 – Stereo Imaging - 5

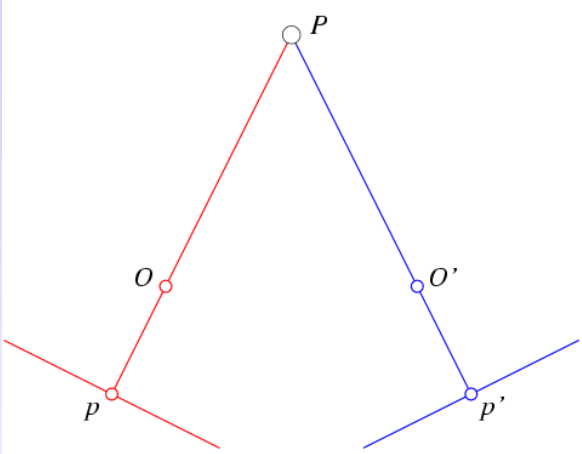
Depth Cues

- Monocular Cues
 - Occlusion – Interposition
 - Relative height
 - Familiar size
 - Texture Gradient
 - Shadows
 - Perspective
- Motion Parallax (also Monocular)
- Binocular Cues

CS 534 – Stereo Imaging - 6



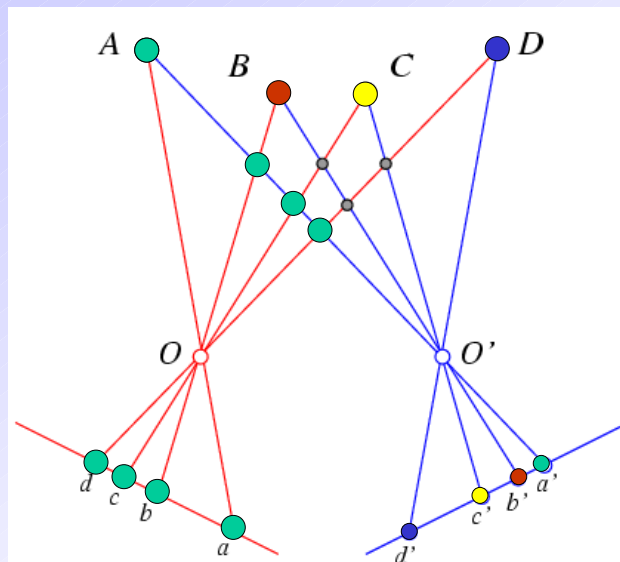
- Given multiple views we can recover scene point - Triangulation



Stereo vision involves two processes:

- *Fusion* of features observed by two or more cameras: which point corresponds to which point ?
- *Reconstruction* of 3D preimage : how to intersect the rays.

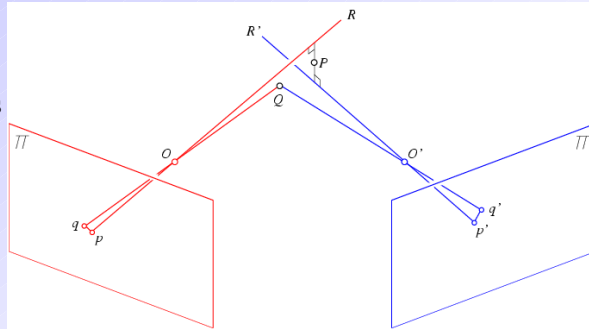
(Binocular) Fusion



Reconstruction

In practice rays never intersect:

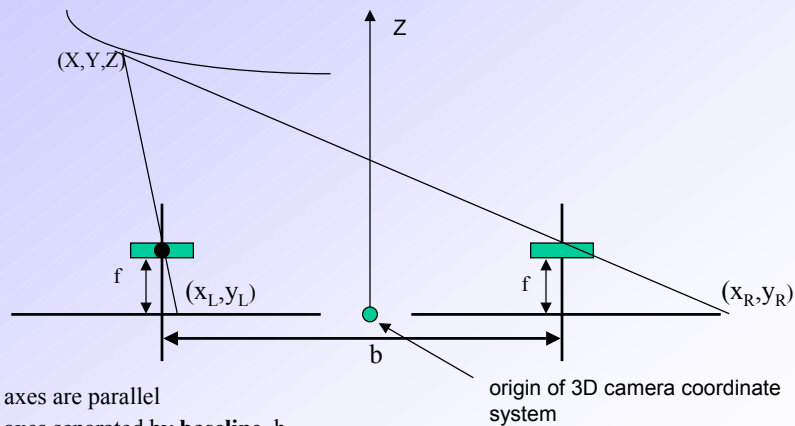
- calibration errors
- feature localization errors



- Algebraic linear method: four equations in three unknown – use linear least squares to find P

- Non-Linear Method: find Q minimizing $d^2(p, q) + d^2(p', q')$

Stereo imaging



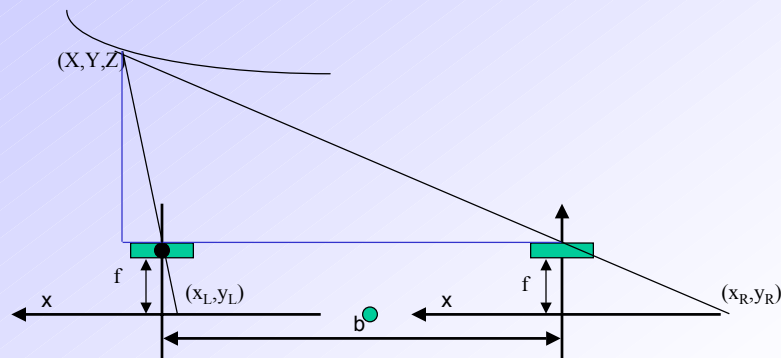
- Optical axes are parallel
- Optical axes separated by **baseline**, b .
- Line connecting lens centers is perpendicular to the optical axis, and the x axis is parallel to that line
- 3D coordinate system is a **cyclopean** system centered between the cameras

Stereo imaging

- (X, Y, Z) are the coordinates of P in the Cyclopean coordinate system.
- The coordinates of P in the left camera coordinate system are $(X_L, Y_L, Z_L) = (X - b/2, Y, Z)$
- The coordinates of P in the right camera coordinate system are $(X_R, Y_R, Z_R) = (X + b/2, Y, Z)$
- So, the x image coordinates of the projection of P are
 - $x_L = (X + b/2)f/Z$
 - $x_R = (X - b/2)f/Z$
- Subtracting the second equation from the first, and solving for Z we obtain:
 - $Z = bf/(x_L - x_R)$
- We can also solve for X and Y :
 - $X = b(x_L + x_R)/2(x_L - x_R)$
 - $Y = by/(x_L - x_R)$

CS 534 – Stereo Imaging - 13

Stereo imaging



- $x_L - x_R$ is called the **disparity**, d , and is always negative
- $X = (b[x_R + x_L])/2/d$ $Y = by/d$ $Z = bf/d$

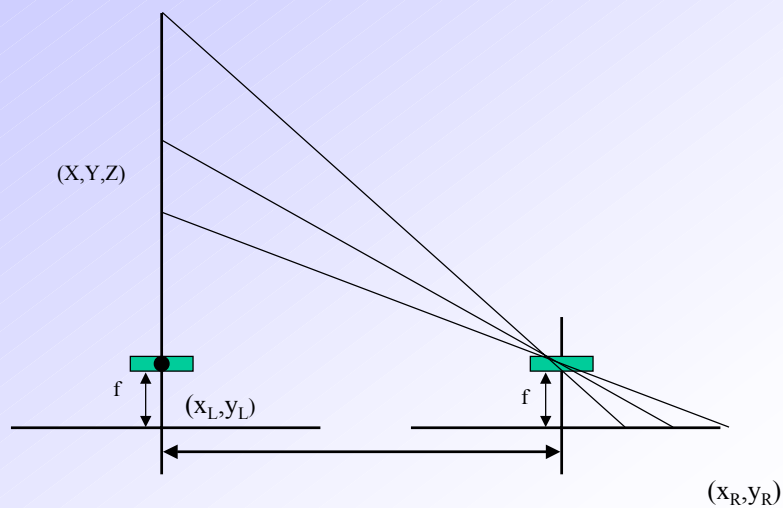
CS 534 – Stereo Imaging - 14

Stereo imaging

- Depth is inversely proportional to $|\text{disparity}|$
 - disparity of 0 corresponds to points that are infinitely far away from the cameras
 - in digital systems, disparity can take on only integer values (ignoring the possibility of identifying point locations to better than a pixel resolution)
 - so, a disparity measurement in the image just constrains distance to lie in a given range
- Disparity is directly proportional to b
 - the larger b , the further we can accurately range
 - but as b increases, the images decrease in common field of view

CS 534 – Stereo Imaging - 15

Range versus disparity



CS 534 – Stereo Imaging - 16

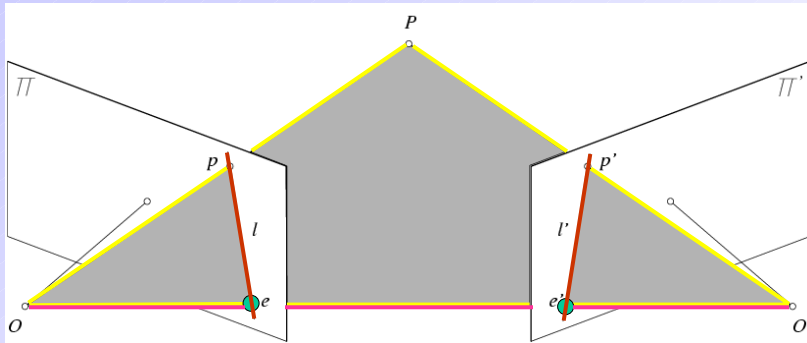
Stereo imaging

- Definition: A scene point, P , visible in both cameras gives rise to a pair of image points called a **conjugate pair**.
 - the conjugate of a point in the left (right) image must lie on the same image row (line) in the right (left) image because the two have the same y coordinate
 - this line is called the **conjugate line**.
 - so, for our simple image geometry, all conjugate lines are parallel to the x axis

A more practical stereo image model

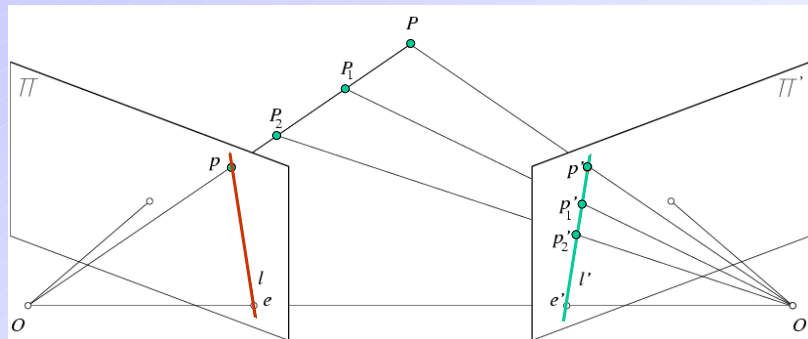
- Difficult, practically, to
 - have the optical axes parallel
 - have the baseline perpendicular to the optical axes
- Also, we might want to tilt the cameras towards one another to have more overlap in the images
- Calibration problem - finding the transformation between the two cameras
 - it is a rigid body motion and can be decomposed into a rotation, \mathbf{R} , and a translation, \mathbf{T} .

Epipolar Geometry



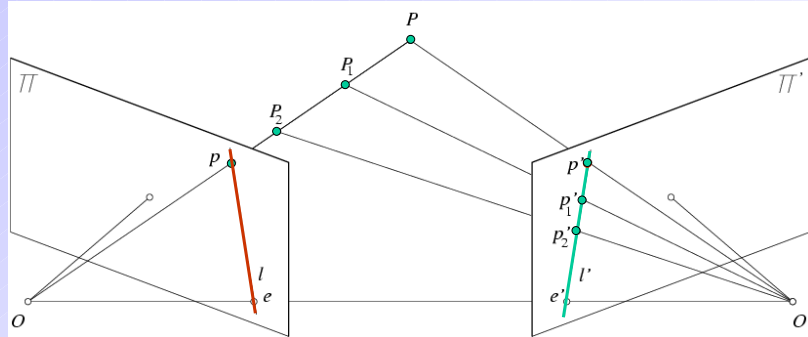
- Epipolar Plane
- Baseline
- Epipoles
- Epipolar Lines

Epipolar Constraint



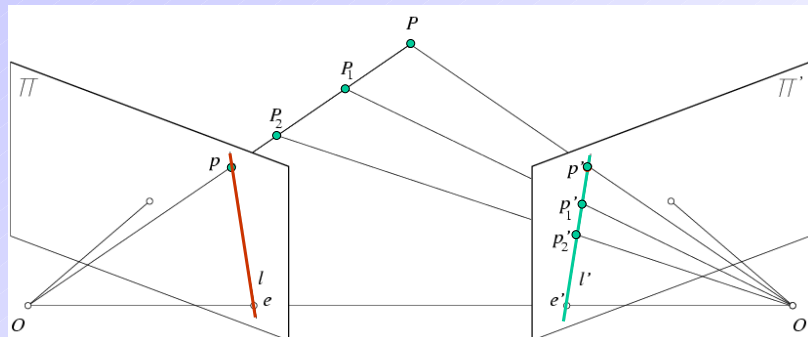
- Potential matches for p have to lie on the corresponding epipolar line l' .
- Potential matches for p' have to lie on the corresponding epipolar line l .

Epipolar Constraint



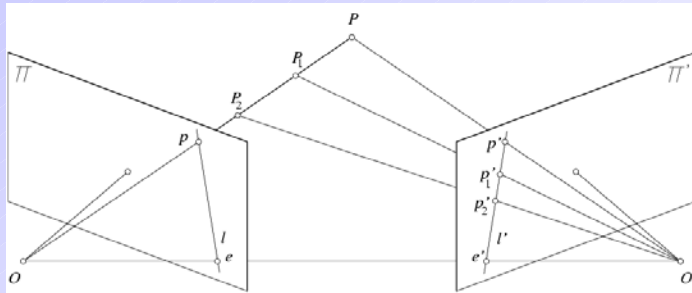
- First scene point possibly corresponding to p is O : (any point closer to the left image than O would be between the lens and the image plane, and could not be seen.)
- So, first possible corresponding point in the right image is e'

Epipolar Constraint



- Last scene point possibly corresponding to p is point at infinity along p line of sight
- but its image is the vanishing point of the ray Op in the right camera
- so we know two points on the epipolar line, any corresponding point p' is between e' and this vanishing point

Epipolar Constraint



epipole e'

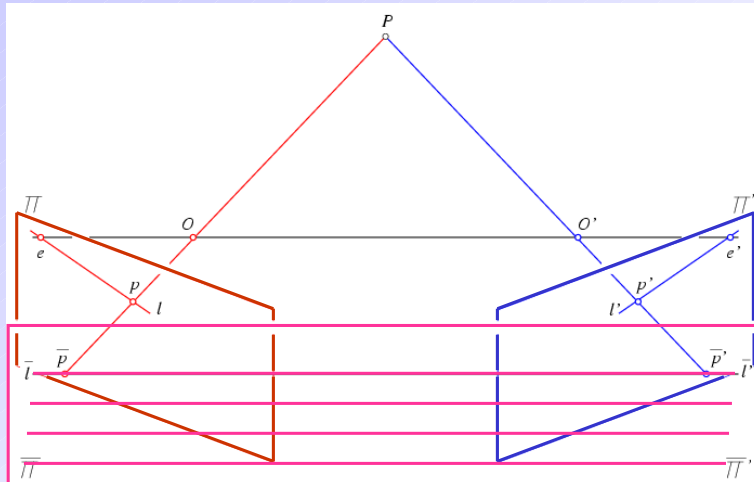
- this is image of the left lens center in the right image
- this point O lies on the line of sight for every point in the left image
- All epipolar lines for all points in the left image must pass through e'
- might not be in the finite field of view

Special case: image planes parallel to the baseline (standard stereo sitting):

- epipolar lines are scan lines
- epipoles at infinity

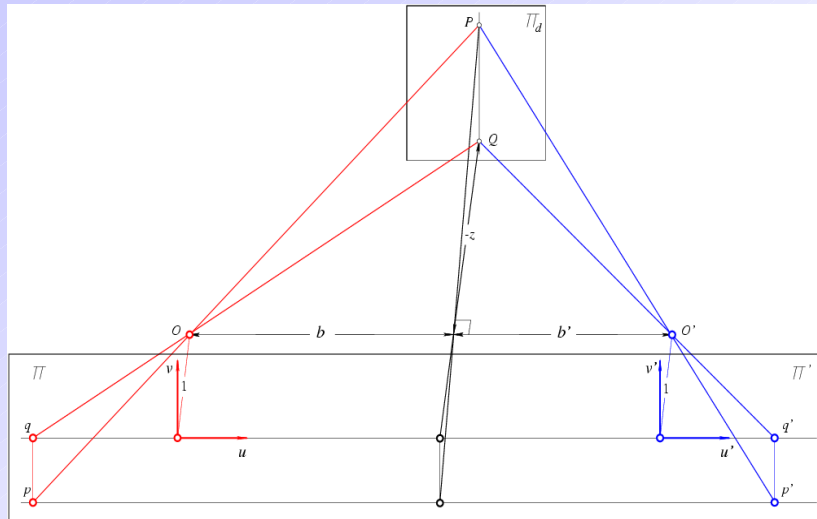
Image Rectification

Project original images to a common image plane parallel to the baseline



All epipolar lines are parallel in the rectified image plane.

Reconstruction from Rectified Images



Disparity: $d = u' - u$.



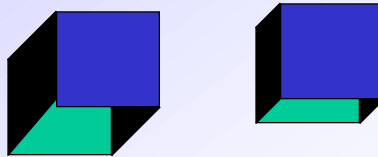
Depth: $z = -B/d$.

Stereo correspondence problem

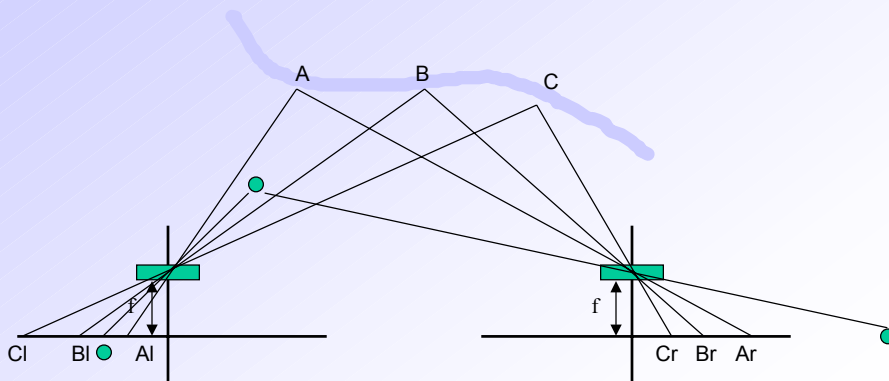
- Given a point, p , in the left image, find its conjugate point in the right image
 - called the stereo correspondence problem
 - Different approaches
- What constraints simplify this problem?
 - Epipolar constraint - need only search for the conjugate point on the epipolar line
 - Negative disparity constraint - need only search the epipolar line to the "right" of the vanishing point in the right image of the ray through p in the left coordinate system
 - Continuity constraint - if we are looking at a continuous surface, images of points along a given epipolar line will be ordered the same way

Stereo correspondence problem

- Similarity of correspondence functions along adjacent epipolar lines
- Disparity gradient constraint - disparity changes slowly over most of the image.
 - Exceptions occur at and near occluding boundaries where we have either discontinuities in disparity or large disparity gradients as the surface recedes away from sight.

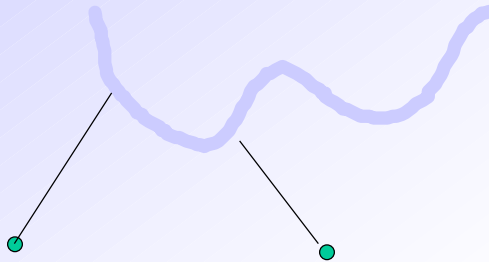


Continuity constraint

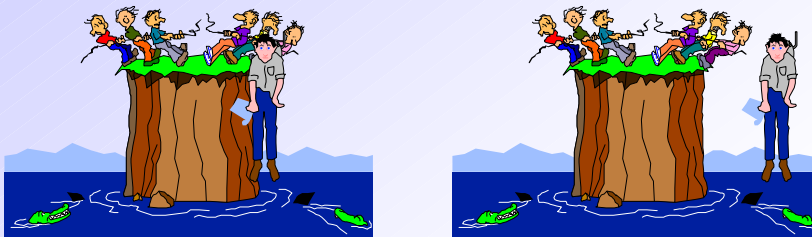


Why is the correspondence problem hard

- Occlusion
 - Even for a smooth surface, there might be points visible in one image and not the other
 - Consider aerial photo pair of urban area - vertical walls of buildings might be visible in one image and not the other
 - scene with depth discontinuities (lurking objects) violate continuity constraint and introduces occlusion



Why is the correspondence problem hard?



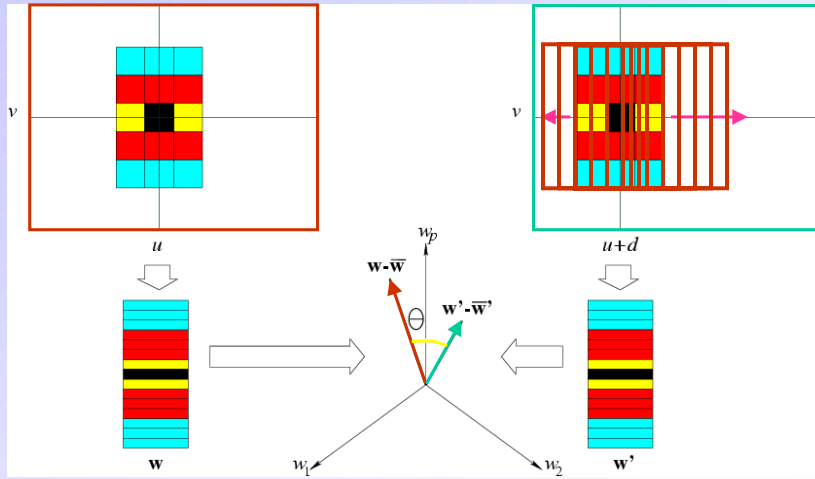
Why is the correspondence problem hard?

- Variations in intensity between images due to
 - noise
 - specularities
 - shape-from-shading differences
- Coincidence of edge and epipolar line orientation
 - consider problem of matching horizontal edges in an ideal left right stereo pair
 - will obtain good match all along the edge
 - so, edge based stereo algorithms only match edges that cross the epipolar lines

Approaches to Find Correspondences

- Intensity Correlation-based approaches
- Edge matching approaches
- Dynamic programming
- Probabilistic approaches (recent)

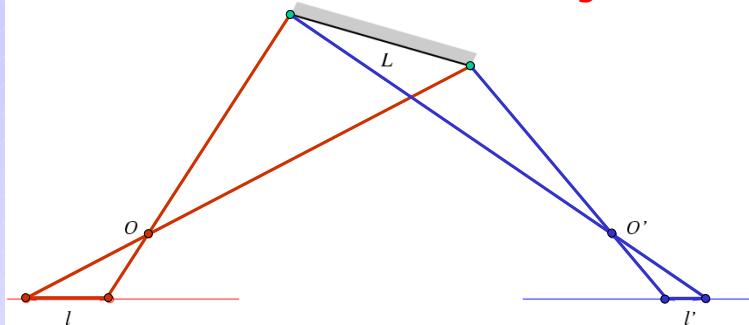
Correlation Methods (1970--)



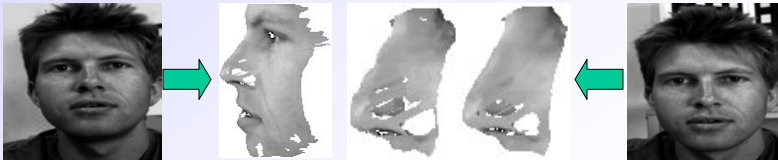
Slide the window along the epipolar line until $w \cdot w'$ is maximized.

Normalized Correlation: minimize θ instead. \leftrightarrow Minimize $|w - w'|$.

Correlation Methods: Foreshortening Problems

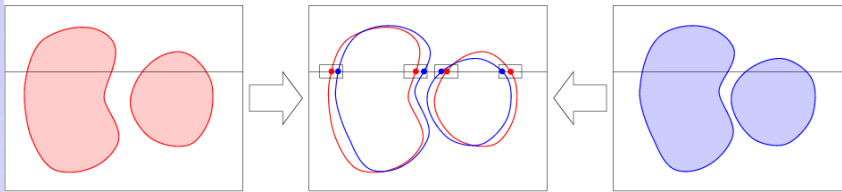


Solution: add a second pass using disparity estimates to warp the correlation windows, e.g. Devernay and Faugeras (1994).

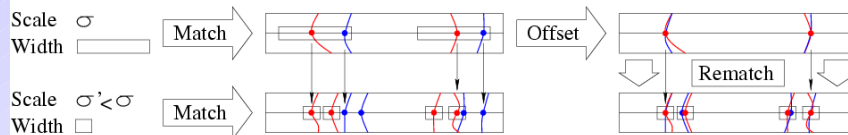


Multi-Scale Edge Matching (Marr, Poggio and Grimson, 1979-81)

Matching zero-crossings at a single scale

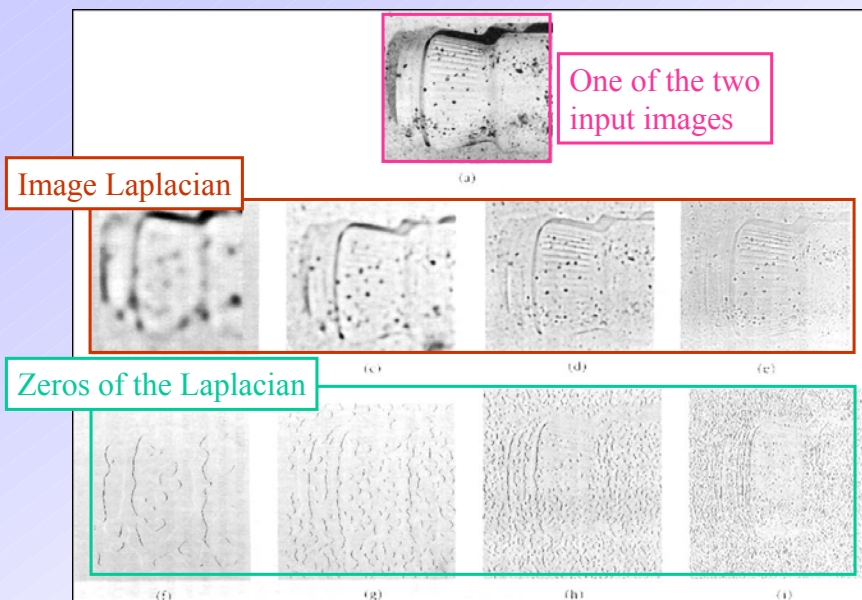


Matching zero-crossings at multiple scales

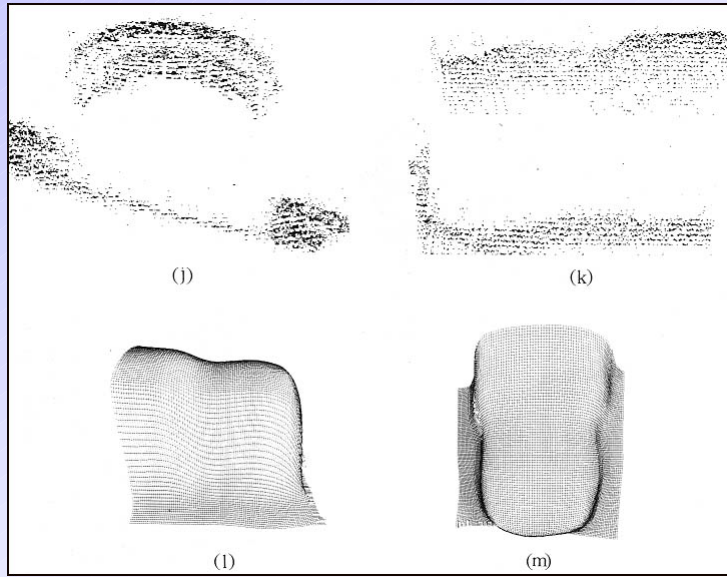


- Edges are found by repeatedly smoothing the image and detecting the zero crossings of the second derivative (Laplacian).
- Matches at coarse scales are used to offset the search for matches at fine scales (equivalent to eye movements).

Multi-Scale Edge Matching (Marr, Poggio and Grimson, 1979-81)

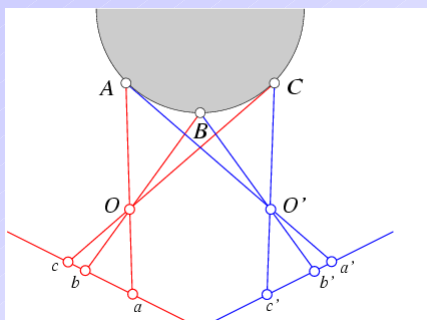


Multi-Scale Edge Matching (Marr, Poggio and Grimson, 1979-81)



Reprinted from Vision: A Computational Investigation into the Human Representation of Spatial Information, D. Marr and T. Poggio, MIT Press, 1983. CS 534 - Stereo Imaging - 37

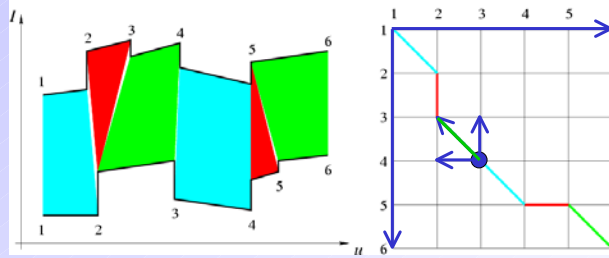
The Ordering Constraint



In general the points are in the same order on both epipolar lines.

But it is not always the case..

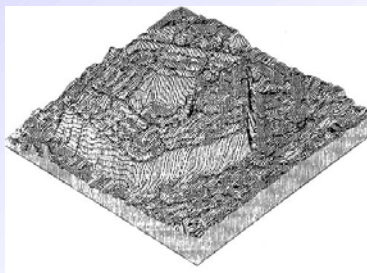
Dynamic Programming (Baker and Binford, 1981)



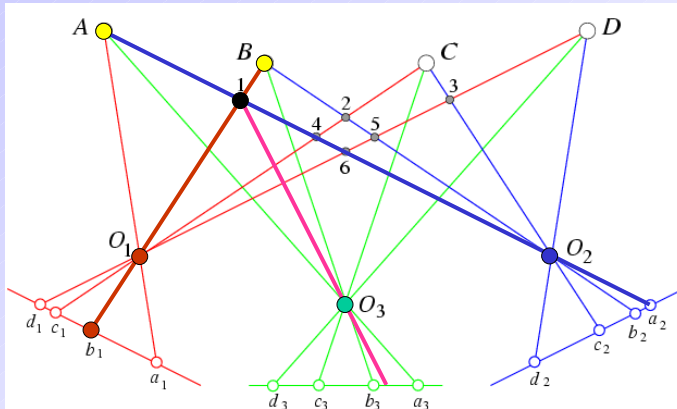
```

% Loop over all nodes (k, l) in ascending order.
for k = 1 to m do
  for l = 1 to n do
    % Initialize optimal cost C(k, l) and backward pointer B(k, l).
    C(k, l) ← +∞; B(k, l) ← nil;
    % Loop over all inferior neighbors (i, j) of (k, l).
    for (i, j) ∈ Inferior-Neighbors(k, l) do
      % Compute new path cost and update backward pointer if necessary.
      d ← C(i, j) + Arc-Cost(i, j, k, l);
      if d < C(k, l) then C(k, l) ← d; B(k, l) ← (i, j) endif;
    endfor;
  endfor;
% Construct optimal path by following backward pointers from (m, n).
P ← {(m, n)}; (i, j) ← (m, n);
while B(i, j) ≠ nil do (i, j) ← B(i, j); P ← {(i, j)} ∪ P endwhile.
    
```

Dynamic Programming (Ohta and Kanade, 1985)



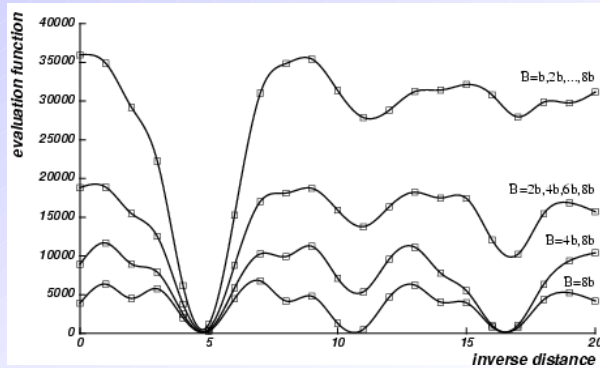
Three Views



The third eye can be used for verification.

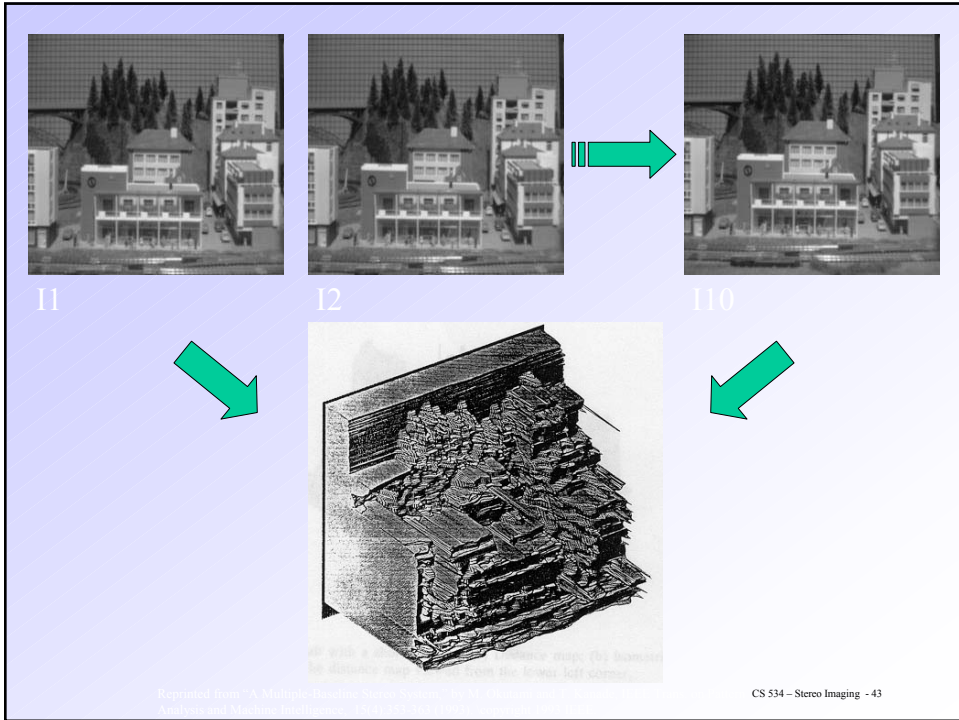
More Views (Okutami and Kanade, 1993)

Pick a reference image, and slide the corresponding window along the corresponding epipolar lines of all other images, using **inverse depth** relative to the first image as the search parameter.



Reprinted from "A Multiple-Baseline Stereo System," by M. Okutami and T. Kanade, Proceedings of the International Conference on Computer Analysis and Machine Intelligence, 151-157, 1993. Copyright 1993 by IEEE.

Use the sum of correlation scores to rank matches.



Sources