3D Vision: Stereo

Marc Pollefeys
Daniel Barath

Spring 2024

http://cvg.ethz.ch/teaching/3dvision/

ETH

Schedule

Feb 19	Introduction
Feb 26	Geometry, Camera Model, Calibration
Mar 4	Guest lecture + Features, Tracking / Matching
Mar 11	**Project Proposals by Students**
Mar 18	3DV conference
Mar 25	Structure from Motion (SfM) + papers
Apr 1	Easter break
Apr 8	**Dense Correspondence (stereo / optical flow) + papers**
Apr 15	Bundle Adjustment & SLAM + papers
Apr 22	**Student Midterm Presentations**
Apr 29	Multi-View Stereo & Volumetric Modeling + papers
May 6	3D Modeling with Depth Sensors + papers
May 13	Guest lecture + papers
May 20	Holiday

Dense Correspondence & Stereo Matching

ETH

Dense Correspondence & Stereo Matching

Tsukuba dataset

http://cat.middlebury.edu/stereo/

Relationship Disparity - Depth

How to recover a 3D point from two corresponding image points?

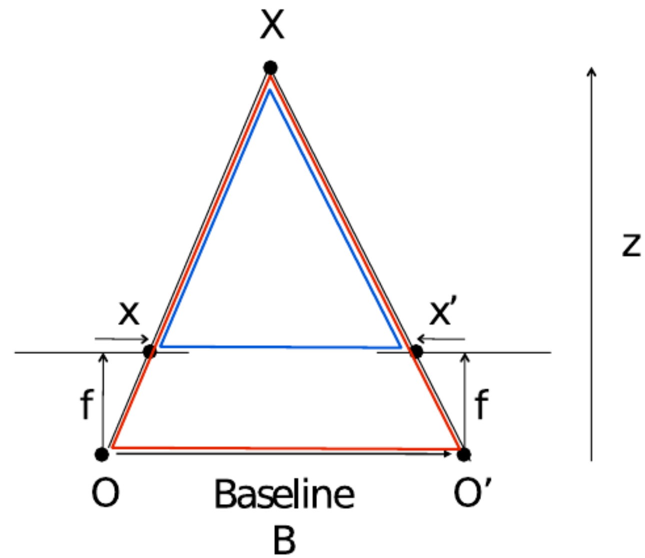
- Equal triangles (only when image planes are parallel)
- Using the definition $d = x - x'$:

$$\frac{Z - f}{B - (x - x')} = \frac{Z}{B}$$

$$ZB - fB = ZB - Z(x - x')$$

$$Z = \frac{fB}{x - x'} = \frac{fB}{d}$$

$$d = \frac{fB}{Z}$$



Overview

Task

- Construct a 3D model from 2 images of a calibrated camera

Pipeline:

1. Find a set of corresponding points
2. Estimate the **epipolar geometry**
3. Rectify both images
4. Dense feature matching
5. 3D reconstruction

Disparity map

image I(x,y)

Disparity map D(x,y)

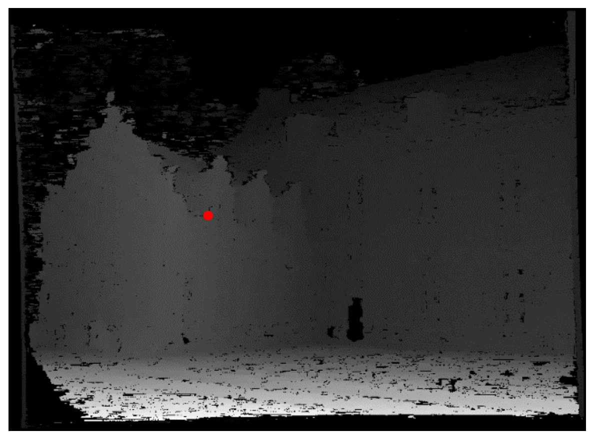
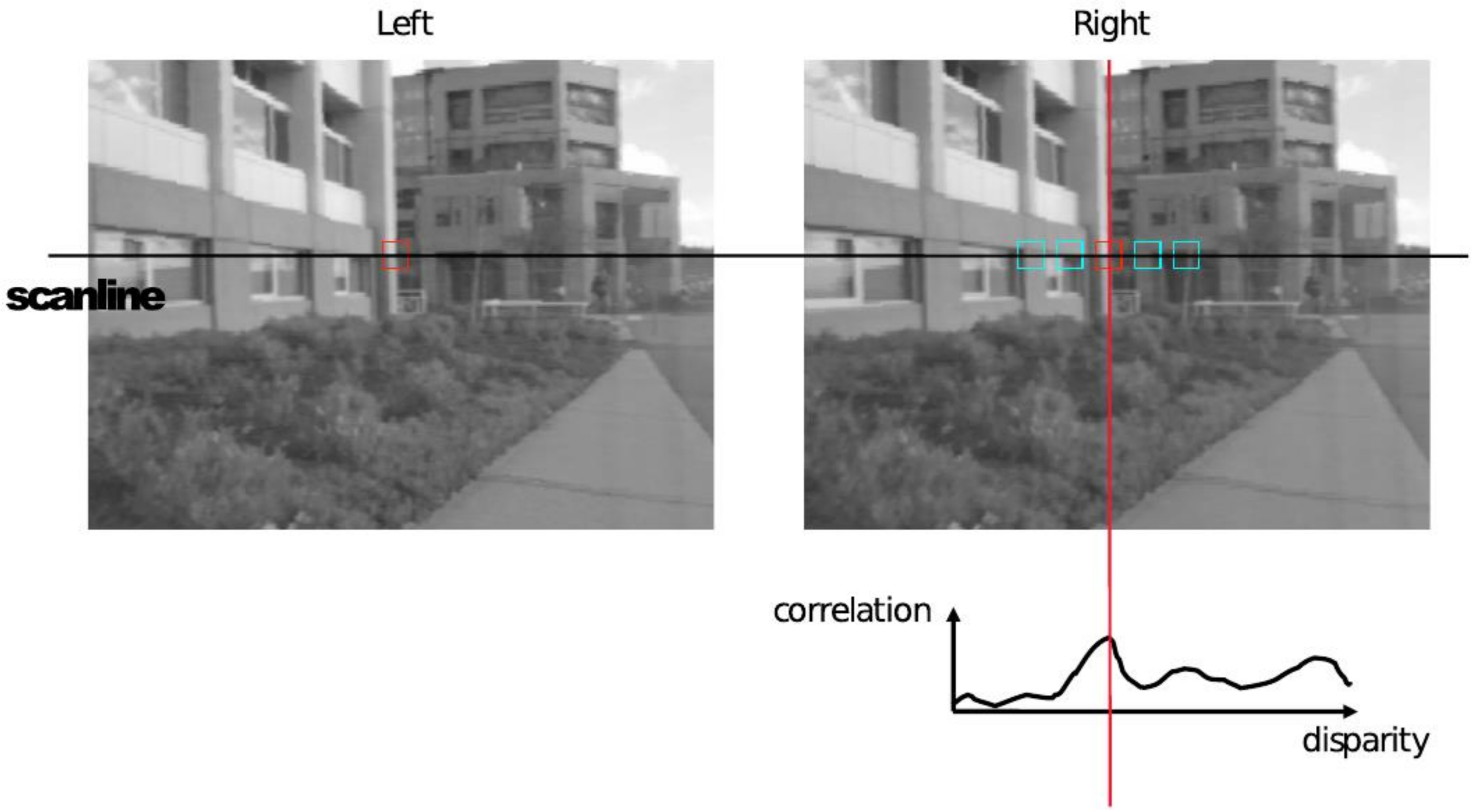


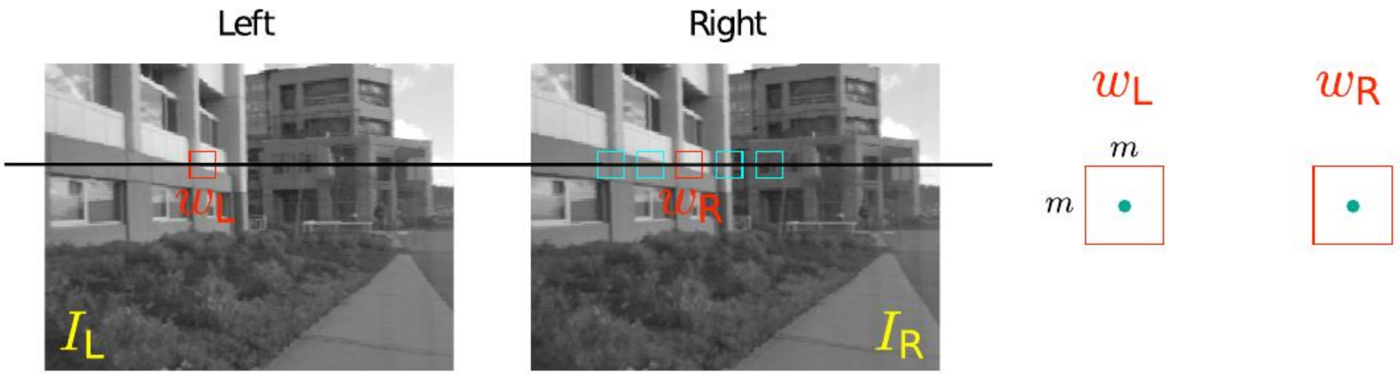
image I´(x´,y´)

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

Photoconsistency



Photoconsistency



- w_L and w_R are corresponding $m \times m$ windows of pixels
- We can write them as vectors: $\mathbf{w}_L, \mathbf{w}_R \in \mathbb{R}^{m^2}$
- Normalized correlation (cosine of the enclosed angle):

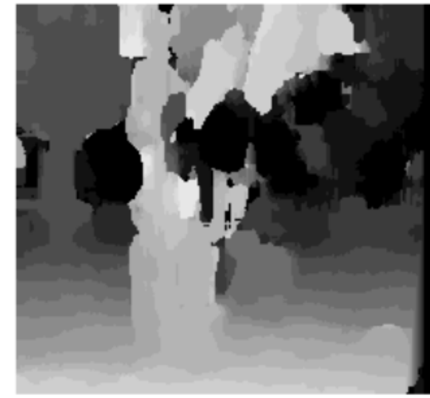
$$\mathrm{NC}(x, y, d) = \frac{(\mathbf{w}_L(x, y) - \bar{\mathbf{w}}_L(x, y))^T (\mathbf{w}_R(x - d, y) - \bar{\mathbf{w}}_R(x - d, y))}{\|\mathbf{w}_L(x, y) - \bar{\mathbf{w}}_L(x, y)\|_2 \, \|\mathbf{w}_R(x - d, y) - \bar{\mathbf{w}}_R(x - d, y)\|_2}$$

Sum of squared differences (SSD):

$$\mathrm{SSD}(x, y, d) = \|\mathbf{w}_L(x, y) - \mathbf{w}_R(x - d, y)\|_2^2$$

Photoconsistency

$m = 3$



$m = 20$

Block Matching:

- Choose some disparity range $[0, d_{max}]$
- For all pixels $\mathbf{x} = (x, y)$ try all disparities and choose the one that maximizes the normalized correlation or minimizes the SSD
- This strategy is called: Winner-takes-all (WTA)
- Do this for both images, apply left-right consistency check

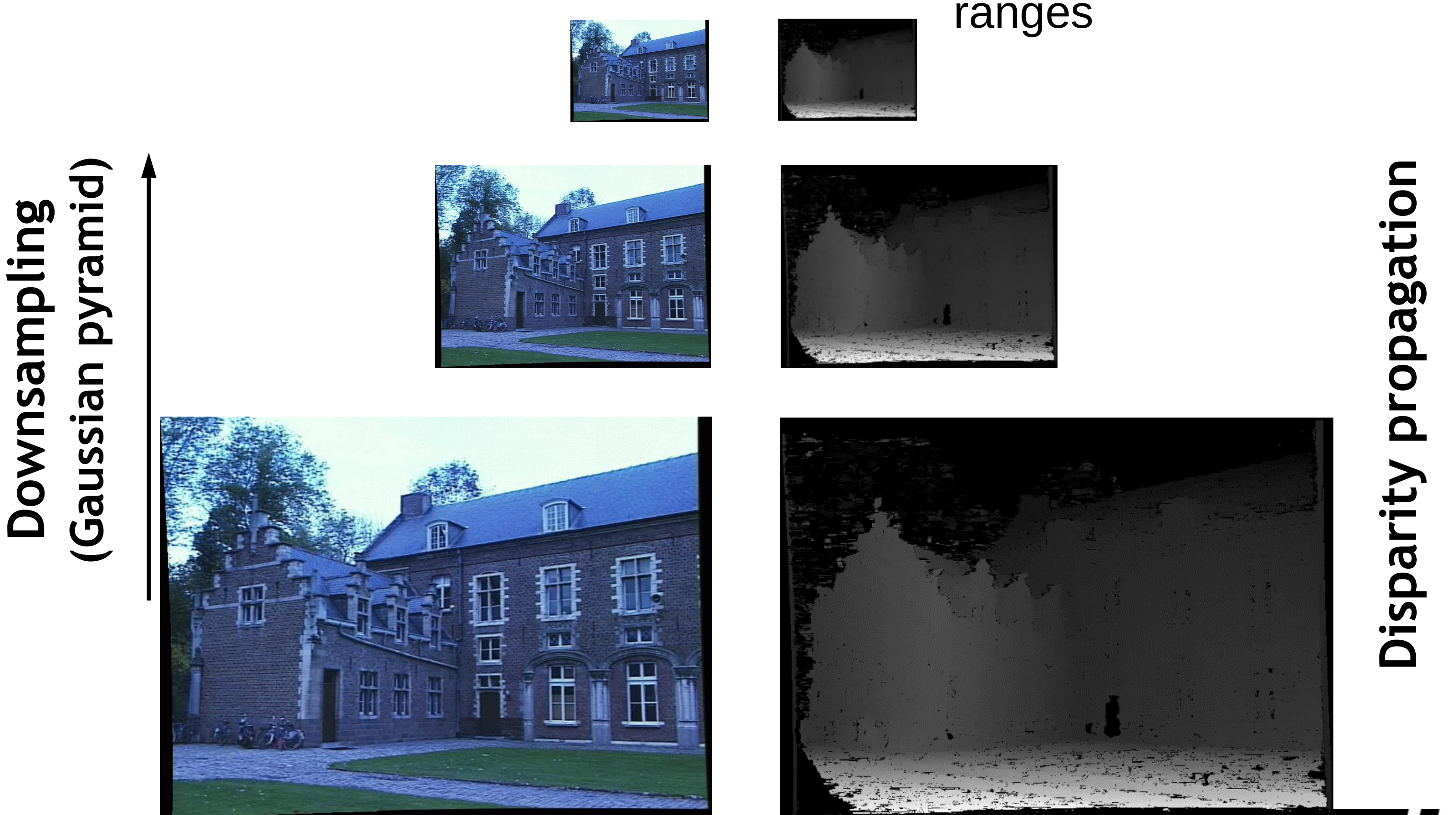
Challenges:

- Which window size to choose? Tradeoff: Ambiguity $\leftrightarrow$ Bleeding!
- Block matching = fronto-parallel assumption (often invalid!)

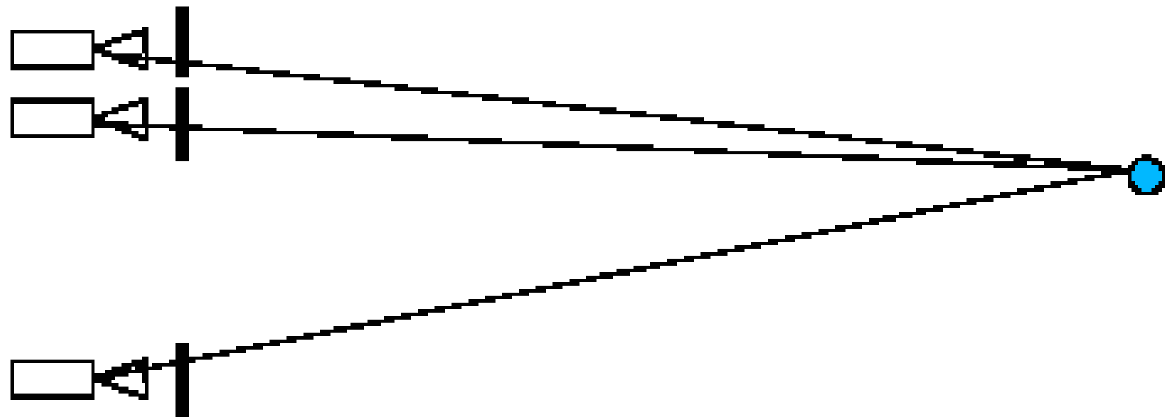
Hierarchical stereo matching

Allows faster computation

Deals with large disparity ranges



Stereo camera configurations

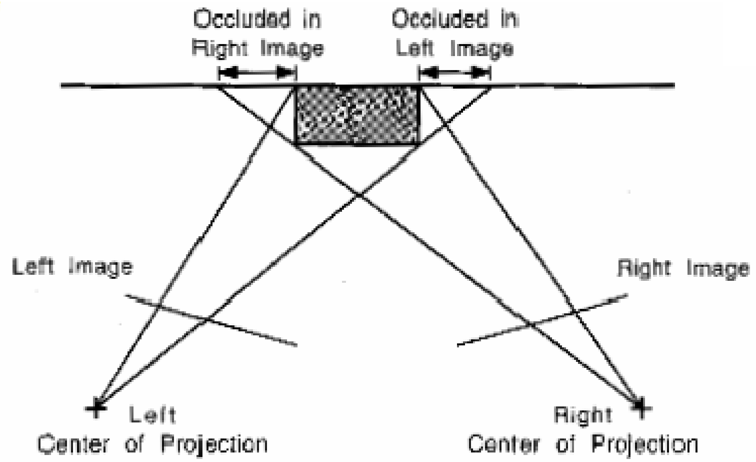


Short baseline:
- Good matches
- Few occlusions
- Poor precision

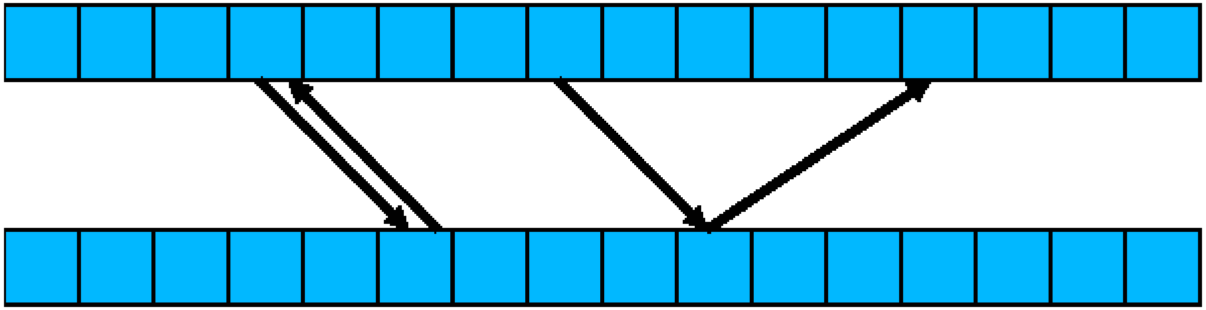
Long baseline:
- Harder to match
- More occlusions
- Better precision

ETH

Occlusions



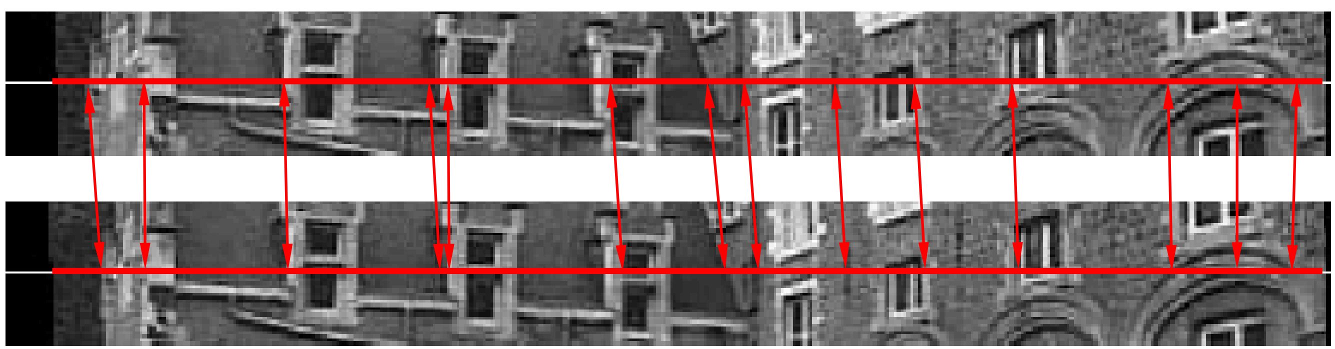
→ Consistency test



ETH

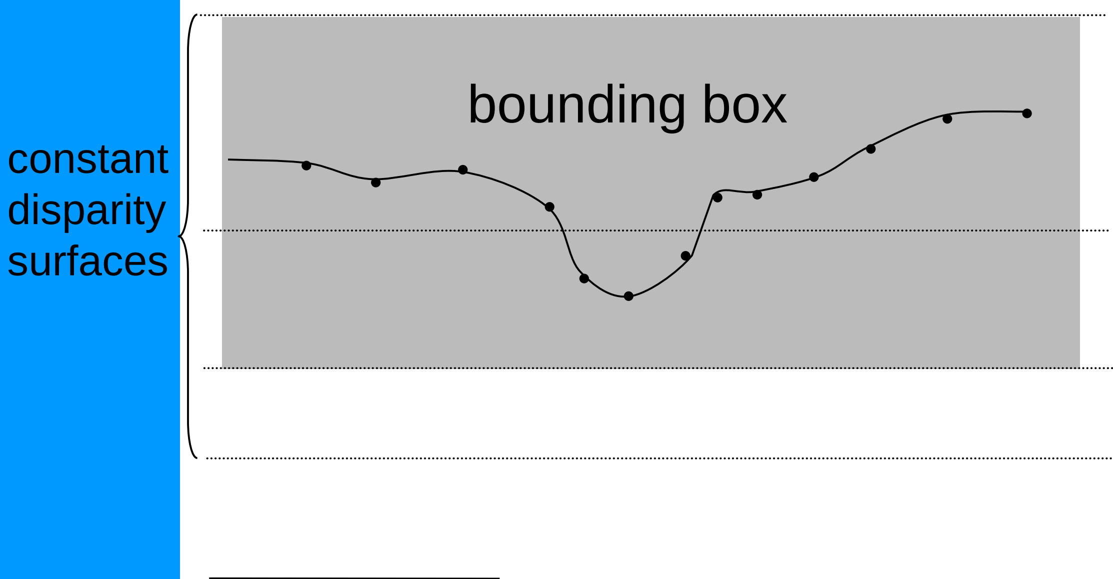
Uniqueness constraint

- In an image pair each pixel has at most one corresponding pixel
  - In general one corresponding pixel
  - In case of occlusion there is none

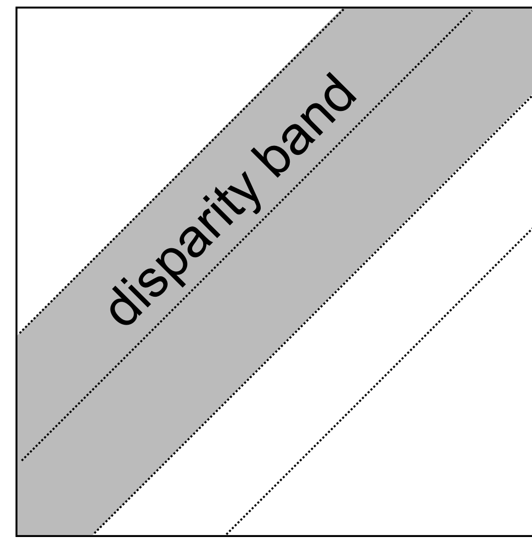


Disparity constraint

surface slice



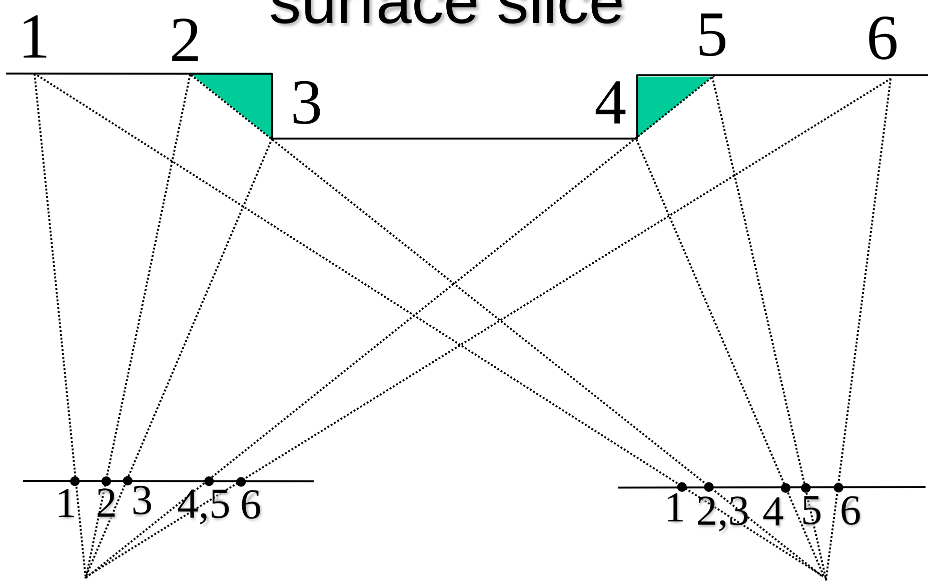
surface as a path



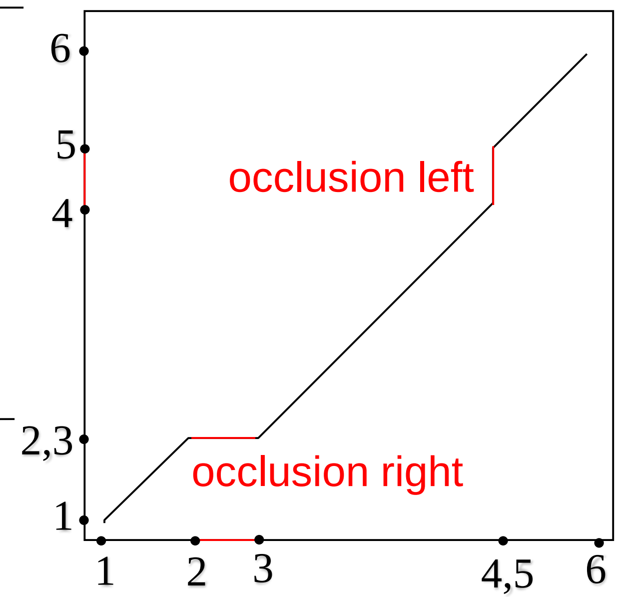
use reconstructed features
to determine bounding box

Ordering constraint

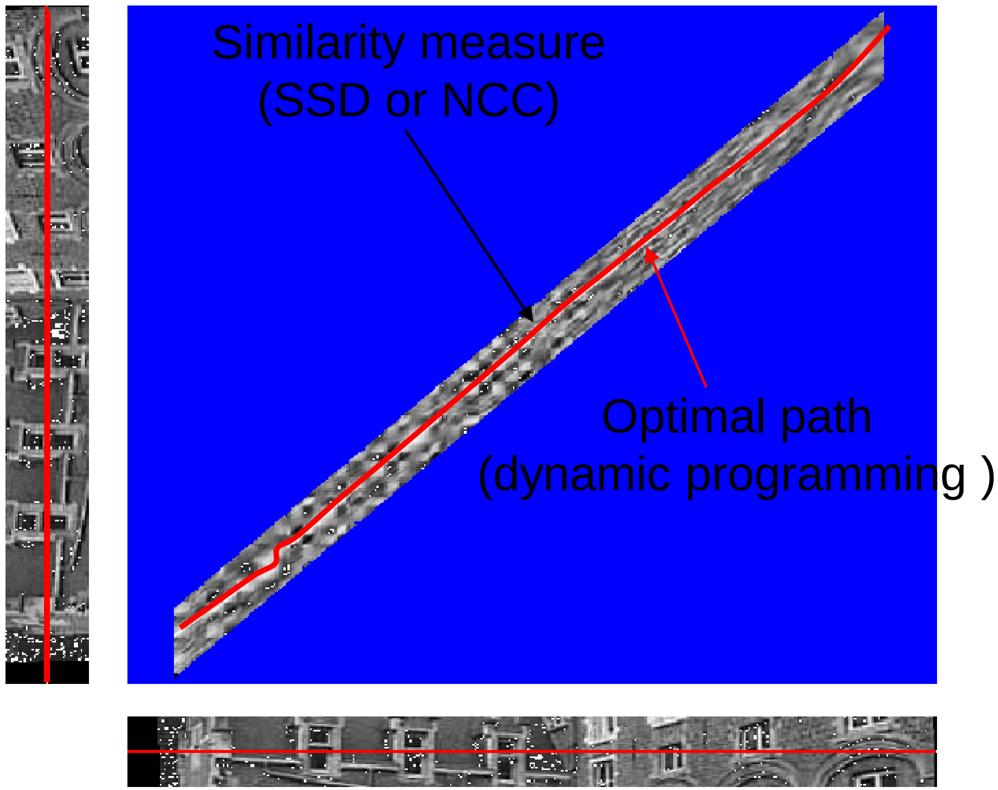
surface slice



surface as a path



Stereo matching



Constraints

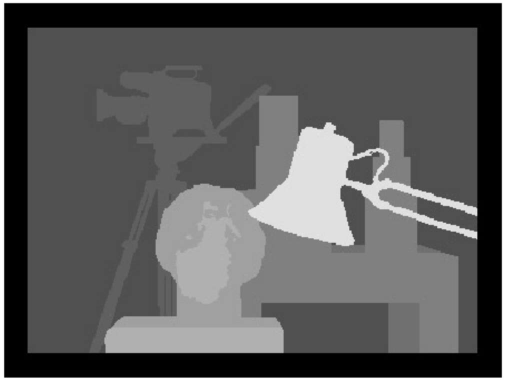
- epipolar
- ordering
- uniqueness
- disparity limit

Trade-off

- Matching cost (data)
- Discontinuities (prior)

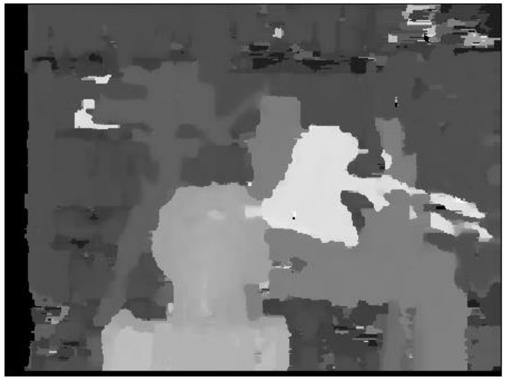
Consider all paths that satisfy the constraints

pick best using dynamic programming



True disparities

*2 – Dynamic progr.

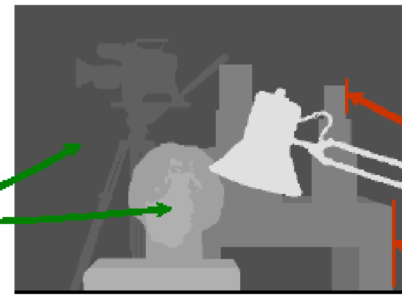


16 – Fast Correlation

(Scharstein & Szeliski, IJCV‘02)

Energy minimization

Disparity continuous in most places,



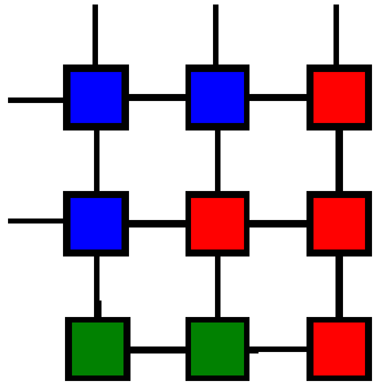
except at depth discontinuities

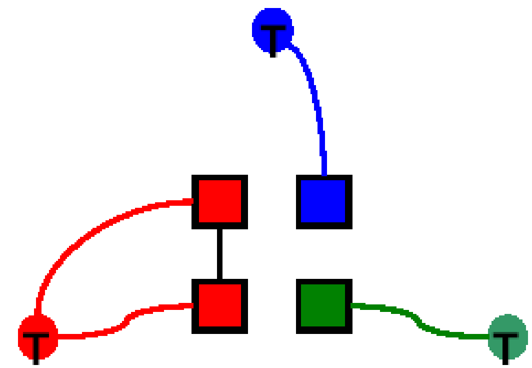
1. Matching pixels should have similar intensities.
2. Most nearby pixels should have similar disparities

→ Minimize

$$\sum\left[I_1(x+D(x,y),y)-I_2(x,y)\right]^2$$
$$+\lambda\sum\left[D(x+1,y)-D(x,y)\right]^2$$
$$+\mu\sum\left[D(x,y+1)-D(x,y)\right]^2$$

Graph Cut



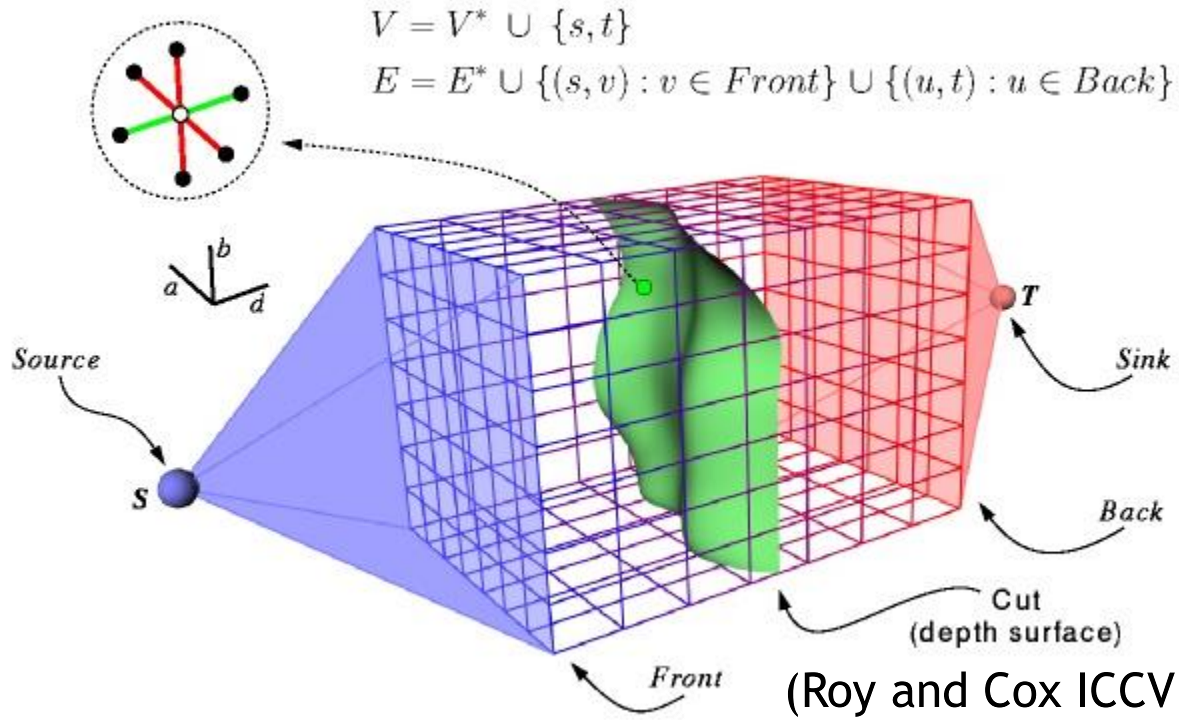


1. Stereo is a labeling problem
2. Graph cut corresponds to a labeling.

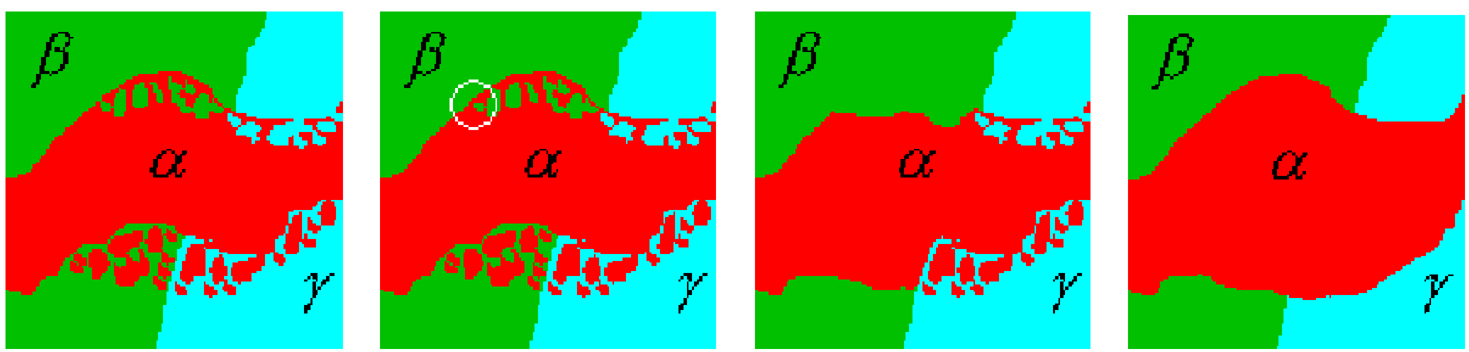
→ **Assign edge weights cleverly so that the min-weight cut gives the minimum energy!**

ETH

Simplified graph cut

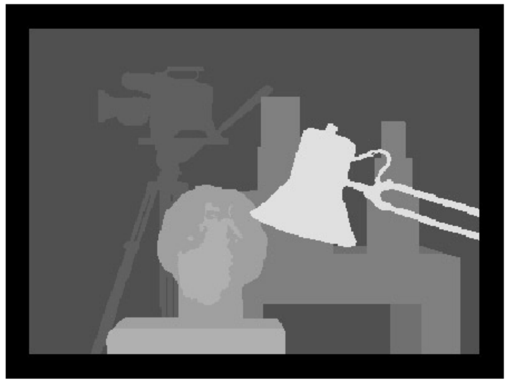


(Roy and Cox ICCV‘98)



(a) initial labeling (b) standard move (c) α-β-swap (d) α-expansion

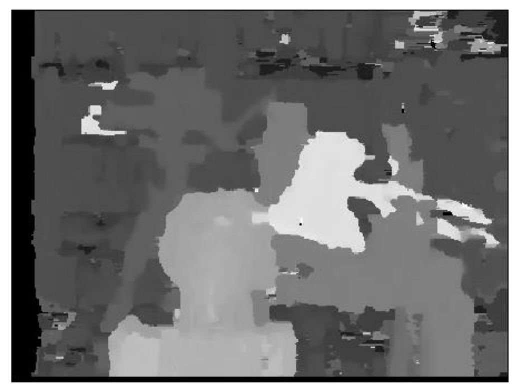
(Boykov et al ICCV‘99)



True disparities

11 – GC + occlusions

*2 – Dynamic progr.



16 – Fast Correlation

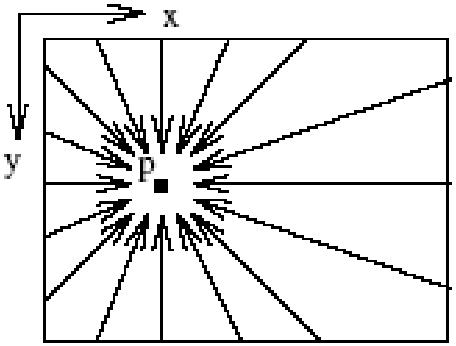
(Scharstein & Szeliski, IJCV‘02)

ETH

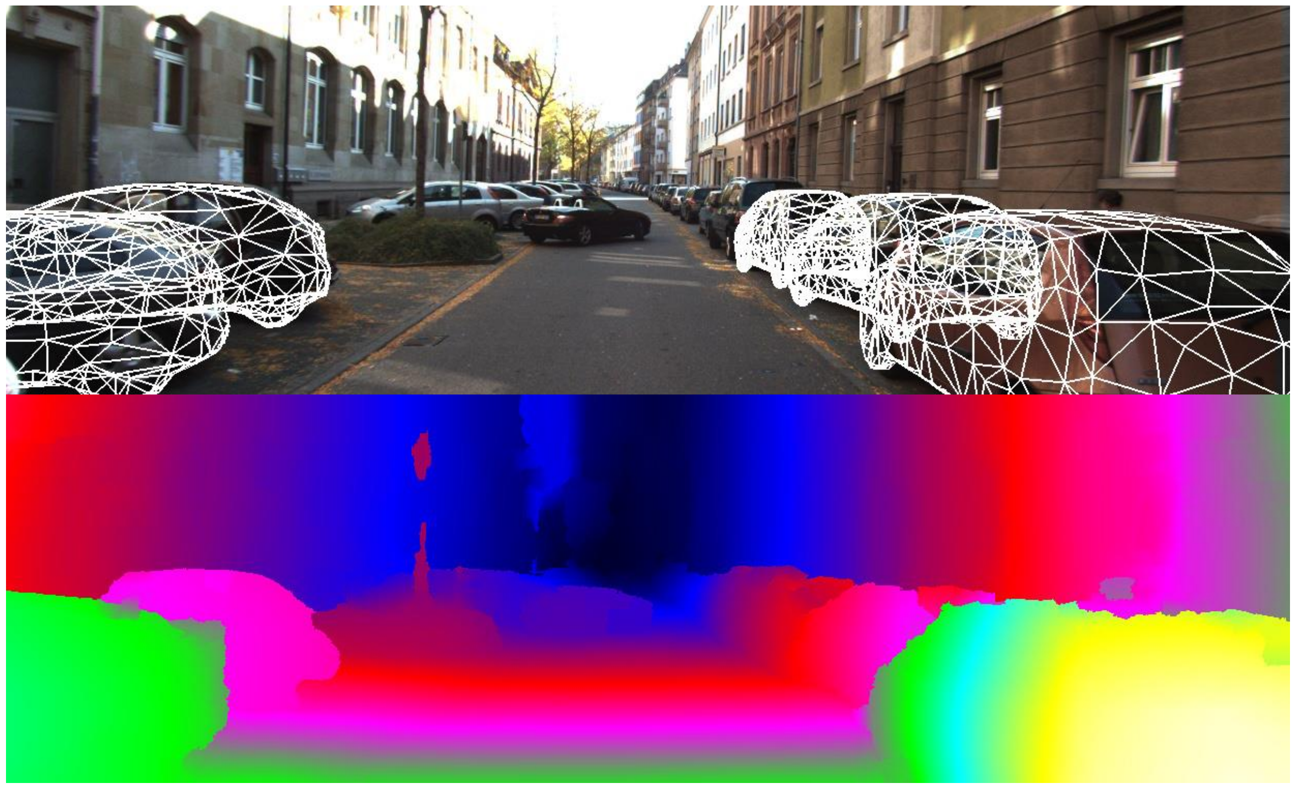
Semi-global optimization

- Optimize:

  $E = E_{data} + E(|D_p - D_q| = 1) + E(|D_p - D_q| > 1)$
  - Use mutual information as cost
  - [Hirschmüller CVPR05]
- NP-hard using graph cuts or belief propagation (2-D optimization)
- Instead do dynamic programming along many directions
  - Don’t use visibility or ordering constraints
  - Add costs of all paths



More Complex Priors



(Güney & Geiger, CVPR 2015)

Stereo matching with general camera configurations

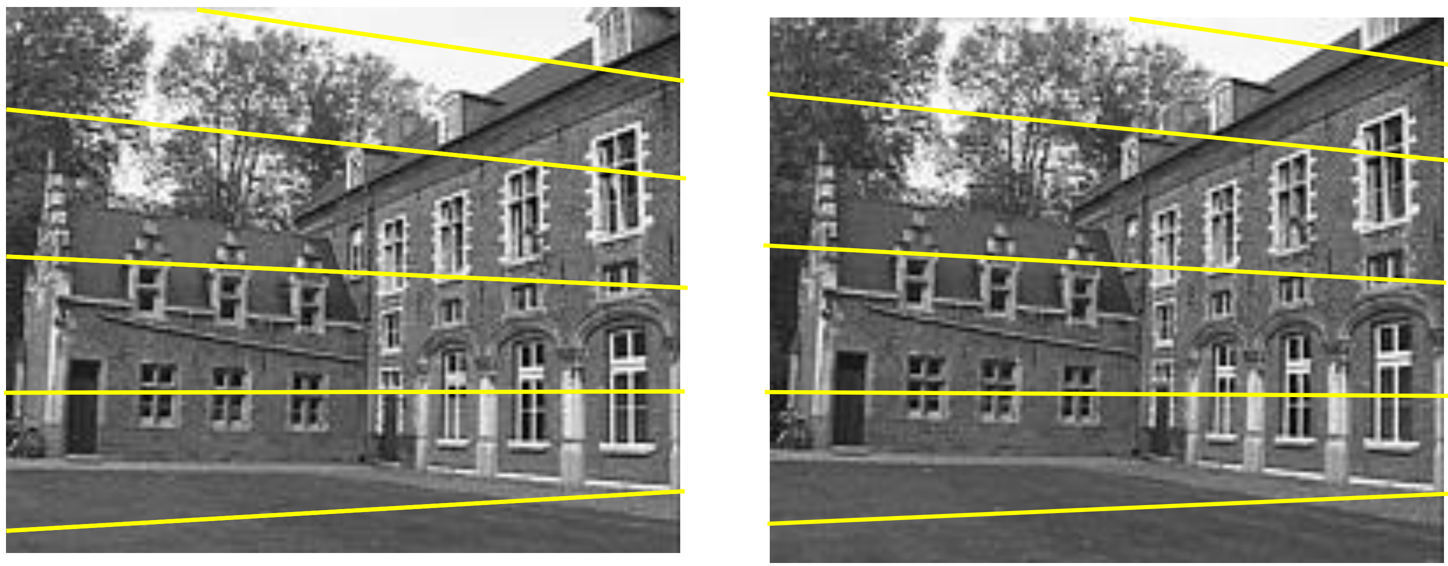
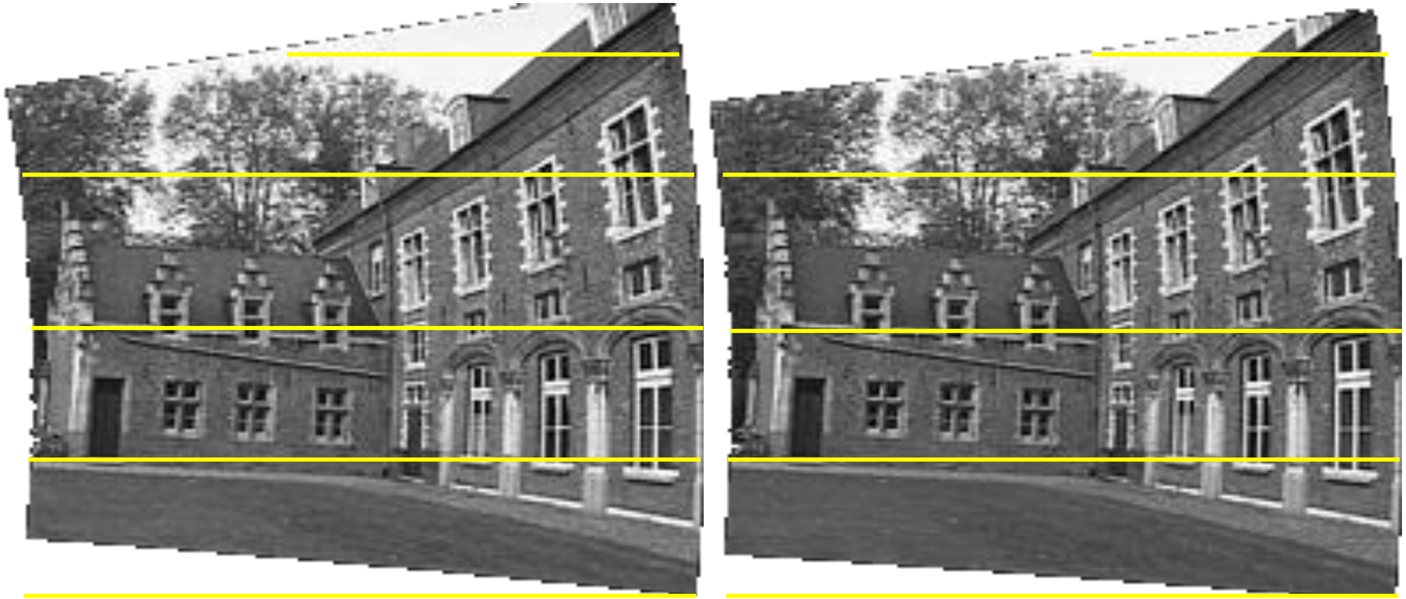
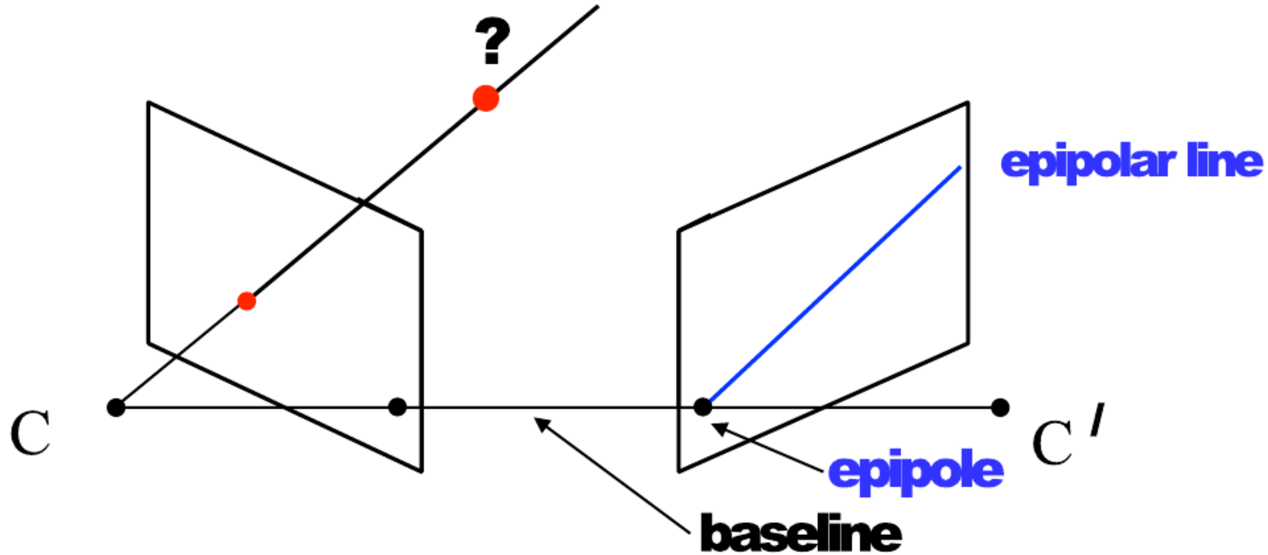


Image pair rectification

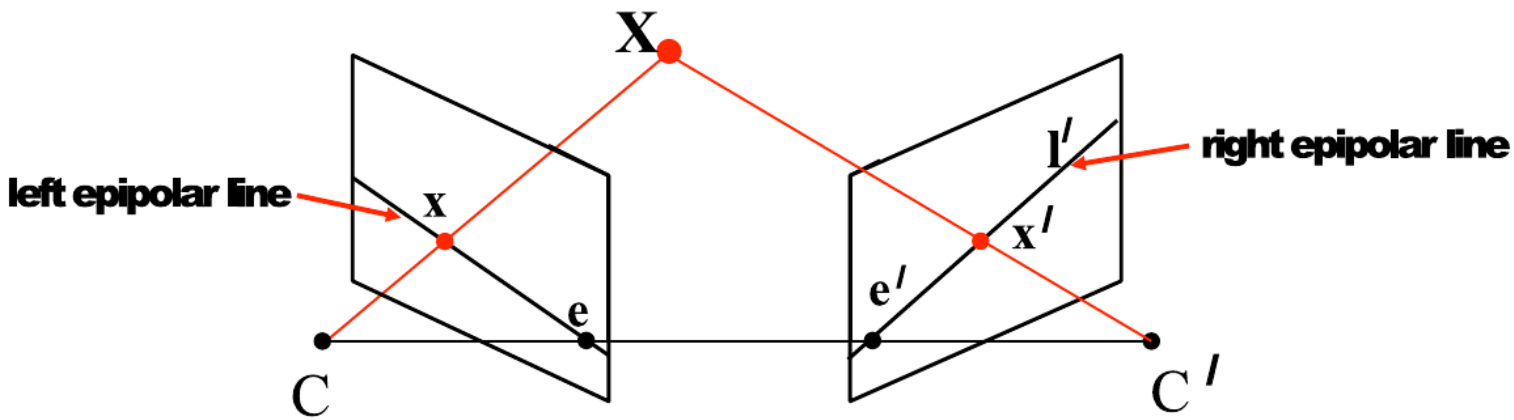


Epipolar Geometry



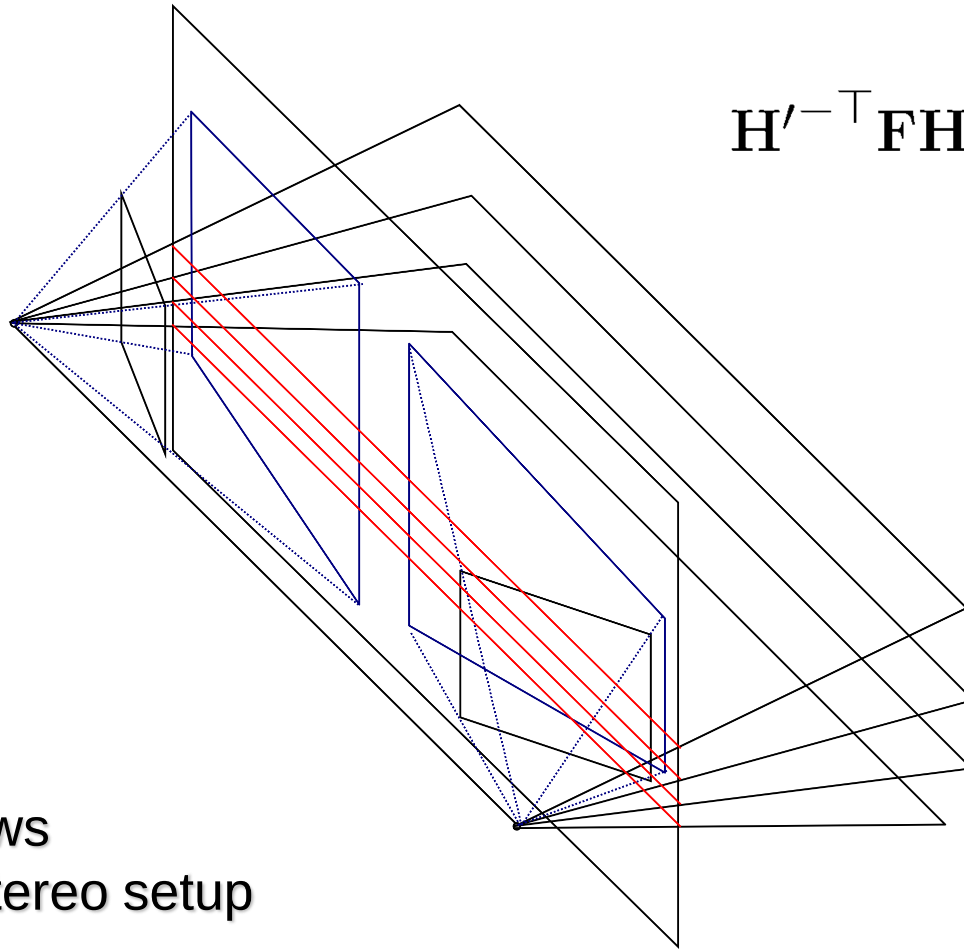
- Lets assume the camera parameters and geometry is known!
- Given a projection of a 3D point in the left image
- Where is it located in 3D?
- On the epipolar line defined by this point and the camera centers
- Reduces the search problem to 1D!

Epipolar Geometry

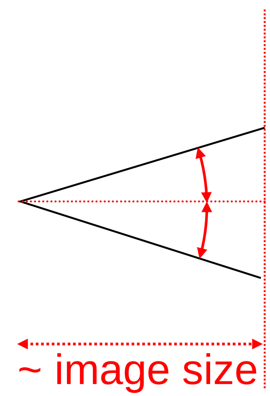


- $\overline{\mathbf{CC}'}$: Baseline (translation between cameras)
- $\mathbf{e}, \mathbf{e}'$: Epipole (intersection of image plane with baseline)
- $\mathbf{l}, \mathbf{l}'$: Epipolar line (intersection of image plane with epipolar plane)

Planar rectification



$$\mathbf{H}'^{-\top}\mathbf{F}\mathbf{H}^{-1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

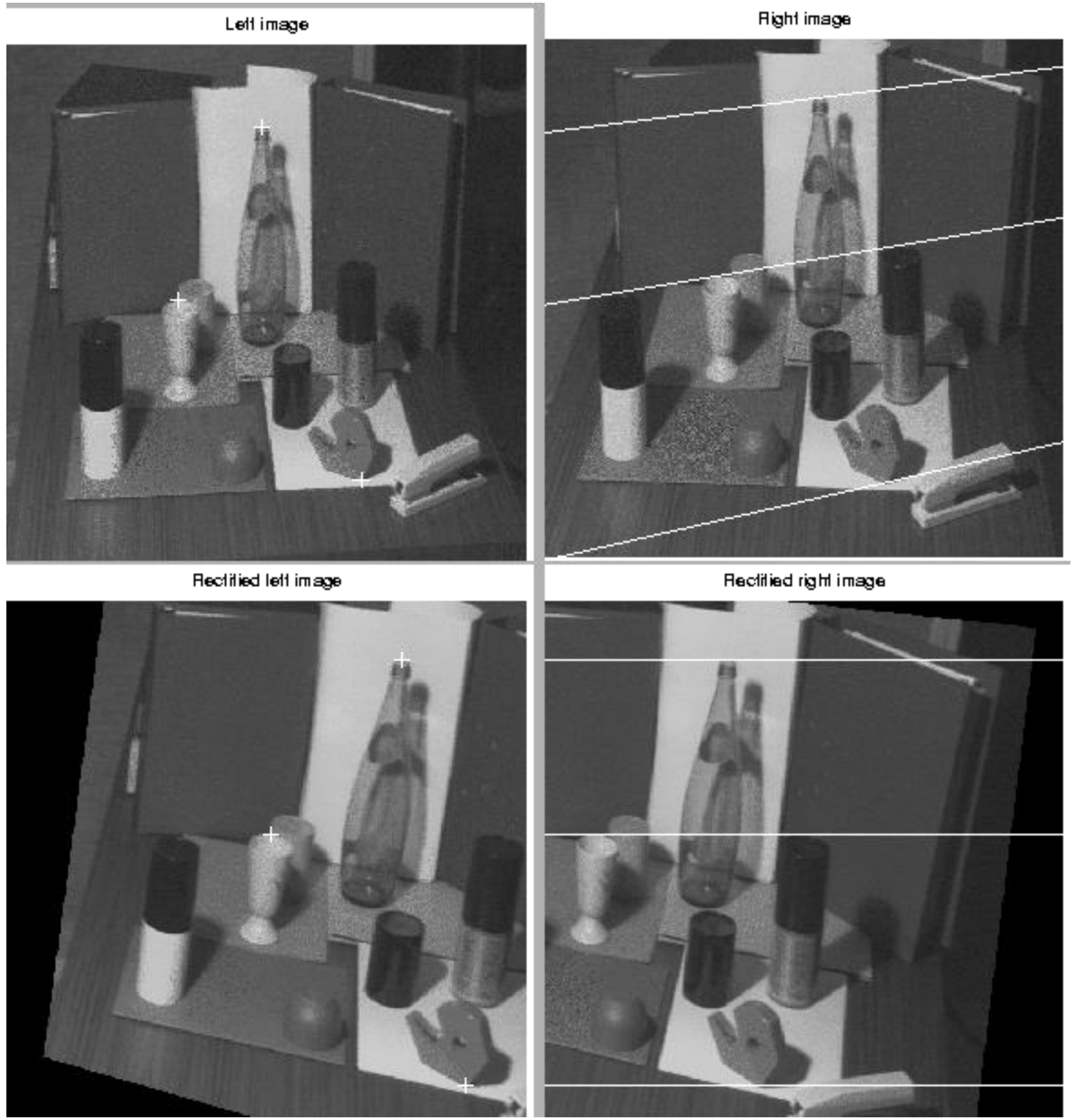


(calibrated)

Distortion minimization
(uncalibrated)

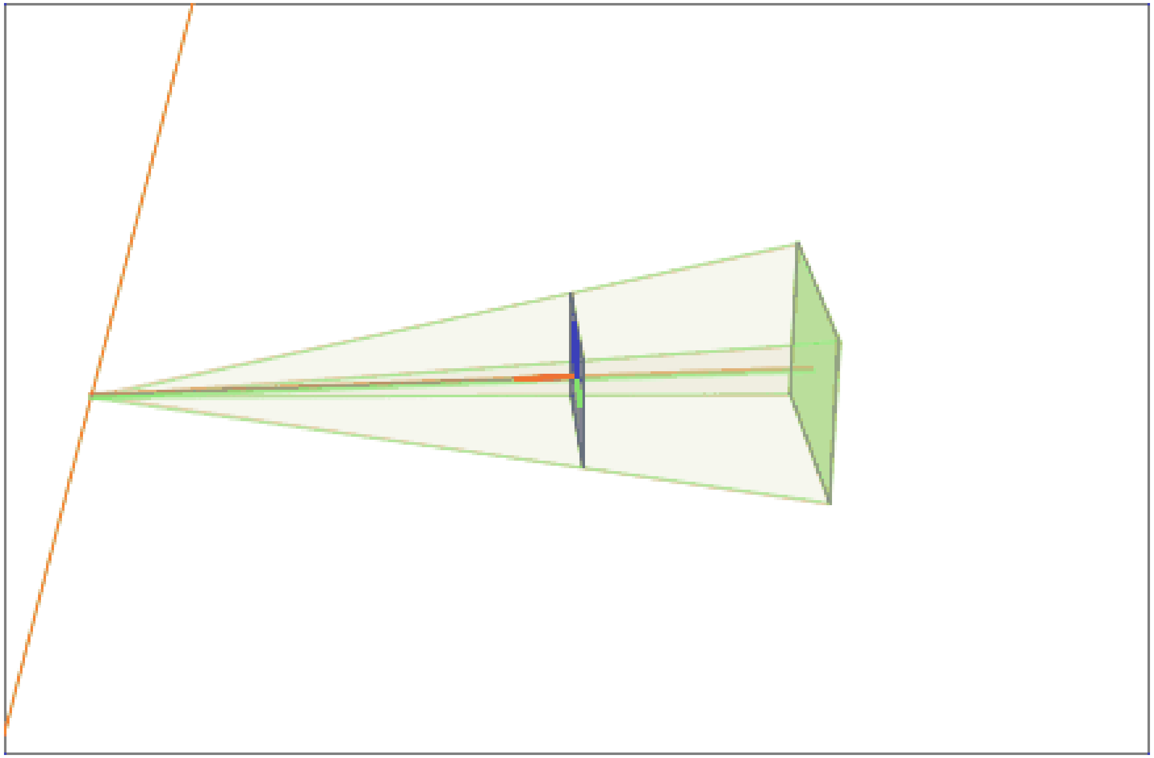
Bring two views
to standard stereo setup

(moves epipole to)
(not possible when in/close to image)


Left image
Right image
Rectified left image
Rectified right image

ETH

Planar rectification

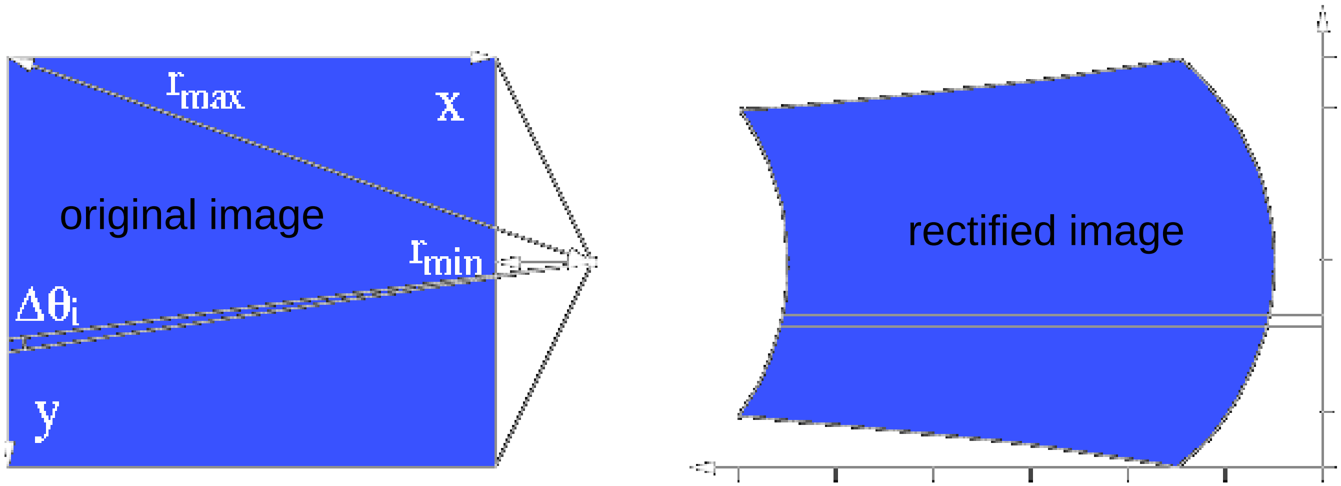


Source: https://en.wikipedia.org/wiki/Image_rectification

Polar rectification

(Pollefeys et al. ICCV’99)

- Polar re-parameterization around epipoles
- Requires only (oriented) epipolar geometry
- Preserve length of epipolar lines
- Choose $\Delta\theta$ so that no pixels are compressed



Works for all relative motions

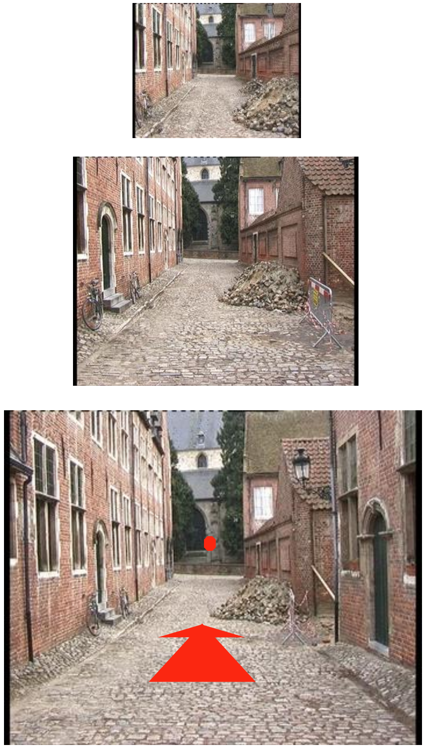
Guarantees minimal image size

original
image pair

planar
rectification

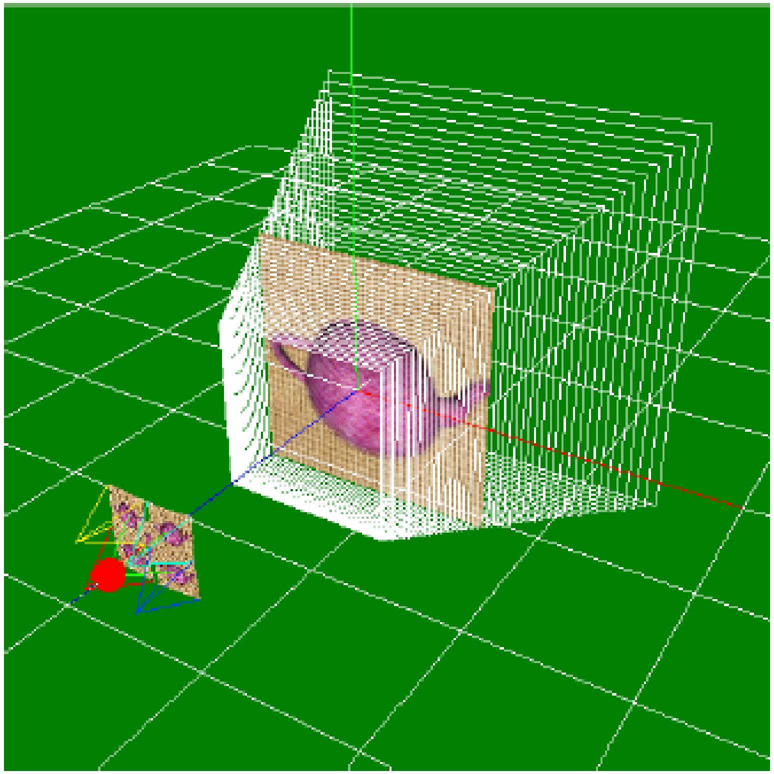
polar
rectification

Example: Béguinage of Leuven



Does not work with standard Homography-based approaches

Plane-sweep multi-view matching

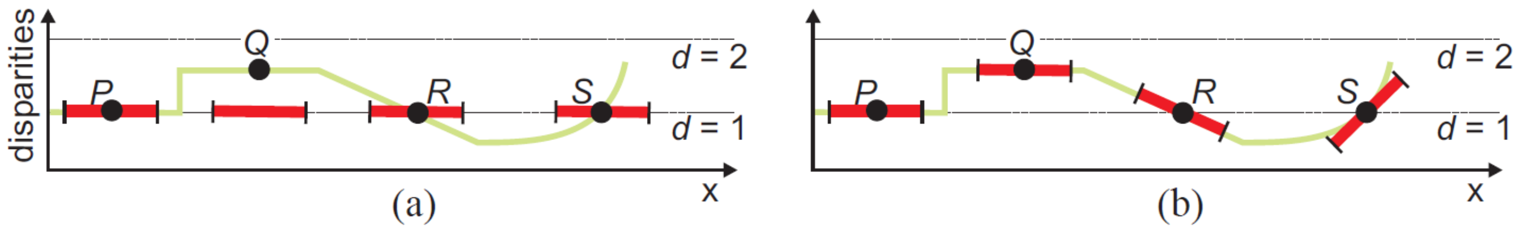


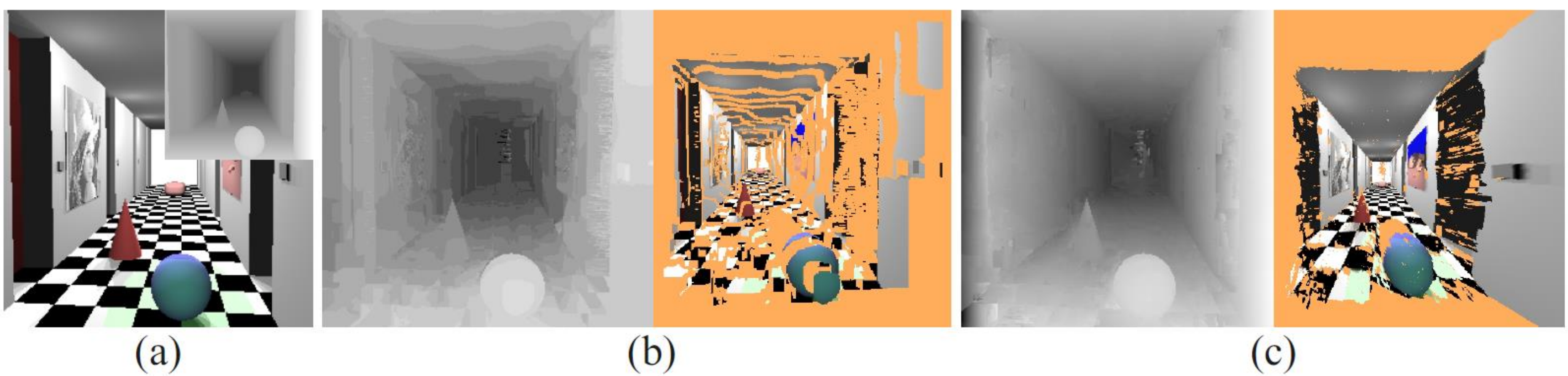
- Simple algorithm for multiple cameras
- no rectification necessary
- doesn't deal with occlusions

Collins'96; Roy and Cox'98 (GC)

PatchMatch Stereo

fronto-parallel windows vs. slanted support windows





(Bleyer et al. BMVC’11)

ETH

PatchMatch Stereo

(Bleyer et al. BMVC' 11)

- For a particular plane the disparity at a pixel is given by

$$d_p = a_{f_p} p_x + b_{f_p} p_y + c_{f_p}$$

- The plane with the minimal cost is chosen

$$f_p = \underset{f \in \mathscr{F}}{\operatorname{argmin}}\, m(p, f)$$

- The dissimilarity cost is calculated as

$$m(p, f) = \sum_{q \in W_p} w(p, q) \cdot \rho(q, q - (a_f q_x + b_f q_y + c_f))$$

with $w(p, q) = e^{-\frac{||I_p - I_q||}{\gamma}}$

$$\rho(q, q') = (1 - \alpha) \cdot \min(||I_q - I_{q'}||, \tau_{col}) + \alpha \cdot \min(||\nabla I_q - \nabla I_{q'}||, \tau_{grad})$$

PatchMatch Stereo

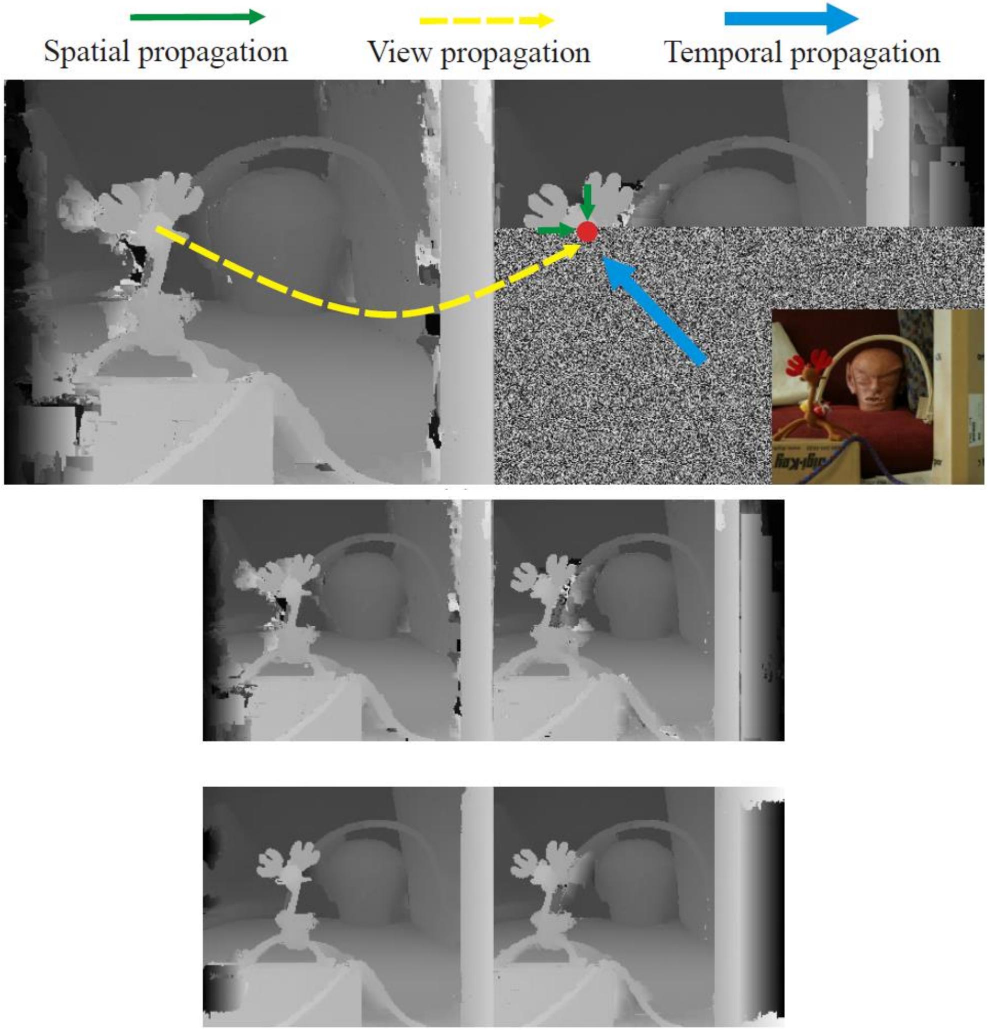
(Bleyer et al. BMVC'11)

Idea: Start with a random initialization of disparities and plane parameters for each pixel and update the estimates by propagating information from the neighboring pixels

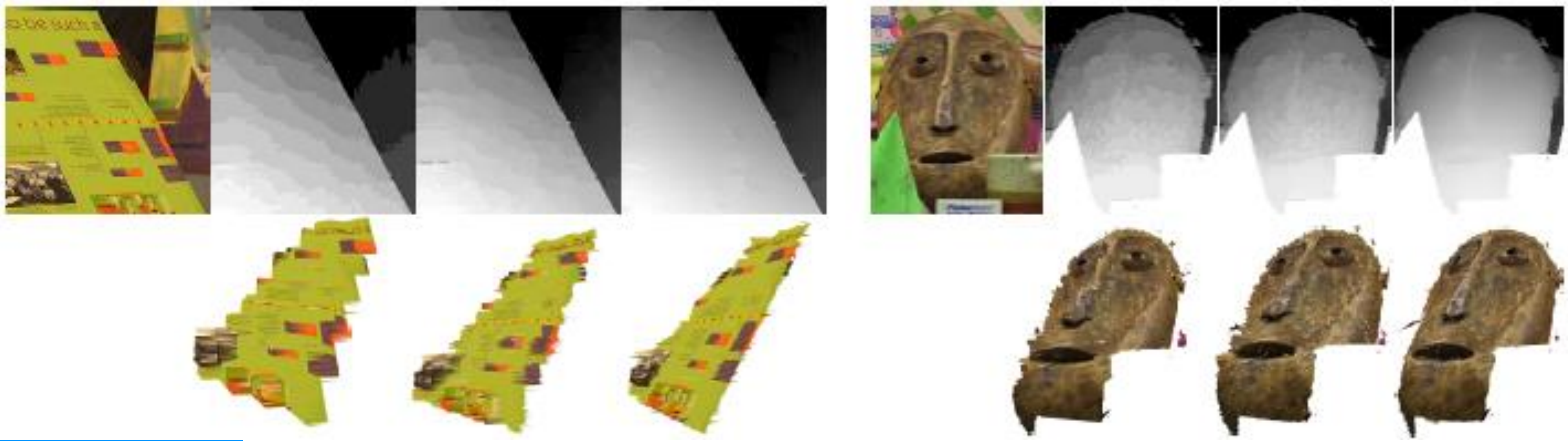
- *Spatial propagation*: Check for each pixel the disparities and plane parameters for the left and upper (right and lower) neighbors and replace the current estimates if matching costs are smaller
- *View propagation*: Warp the point in the other view and check the corresponding estimates in the other image. Replace if the matching costs are lower.
- *Temporal propagation*: Propagate the information analogously by considering the estimates for the same pixel at the preceding and consecutive video frame

PatchMatch Stereo

(Bleyer et al. BMVC'11)



PatchMatch Stereo



Left to right:

- Fronto-parallel, discrete disparities
- Fronto-parallel, continuous disparities
- PatchMatch Stereo (slanted, continuous disparities)

(Bleyer et al. BMVC’11)

ETH

Schedule

Feb 19	Introduction
Feb 26	Geometry, Camera Model, Calibration
Mar 4	Guest lecture + Features, Tracking / Matching
Mar 11	**Project Proposals by Students**
Mar 18	3DV conference
Mar 25	Structure from Motion (SfM) + papers
Apr 1	Easter break
Apr 8	**Dense Correspondence (stereo / optical flow) + papers**
Apr 15	Bundle Adjustment & SLAM + papers
Apr 22	**Student Midterm Presentations**
Apr 29	Multi-View Stereo & Volumetric Modeling + papers
May 6	3D Modeling with Depth Sensors + papers
May 13	Guest lecture + papers
May 20	Holiday

Next week:
Bundle Adjustment & SLAM

Now:
Papers!