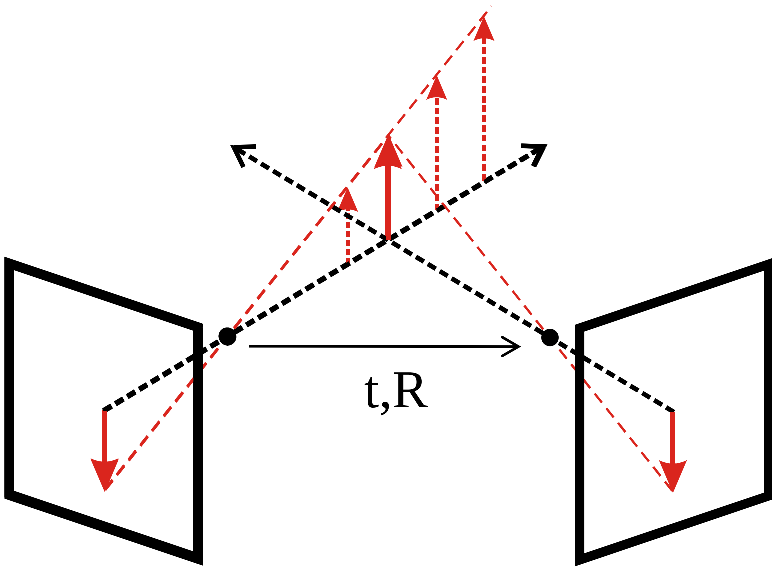
ME/CS 132: Introduction to Vision-based Robot Navigation

Stereo Vision

Roland Brockers
brockers@jpl.nasa.gov

Stereo vision

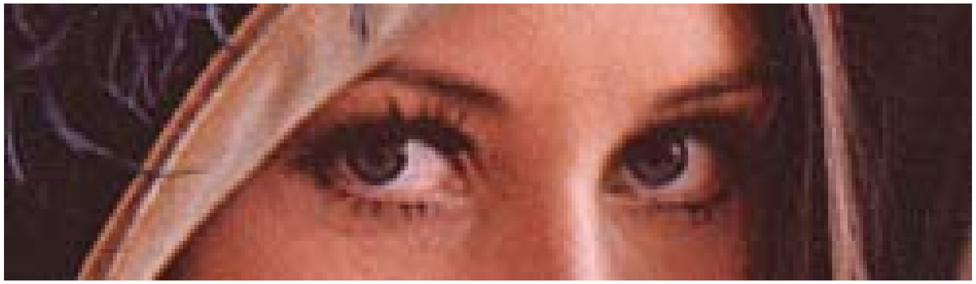
3D reconstruction using cameras as passive sensors

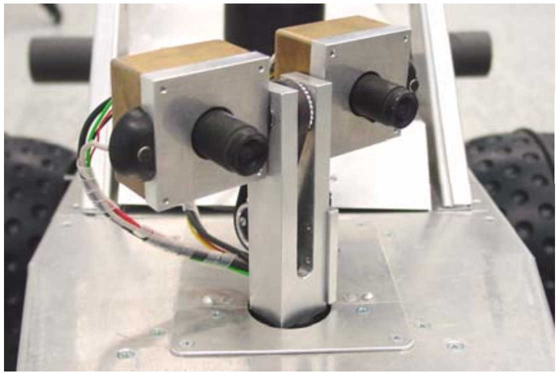


- 2D projection of 3D world
- At least 2 views for reconstruction
- Reconstruction relative to translation t

- Stereo: shape from known "motion" between two views

Motivation





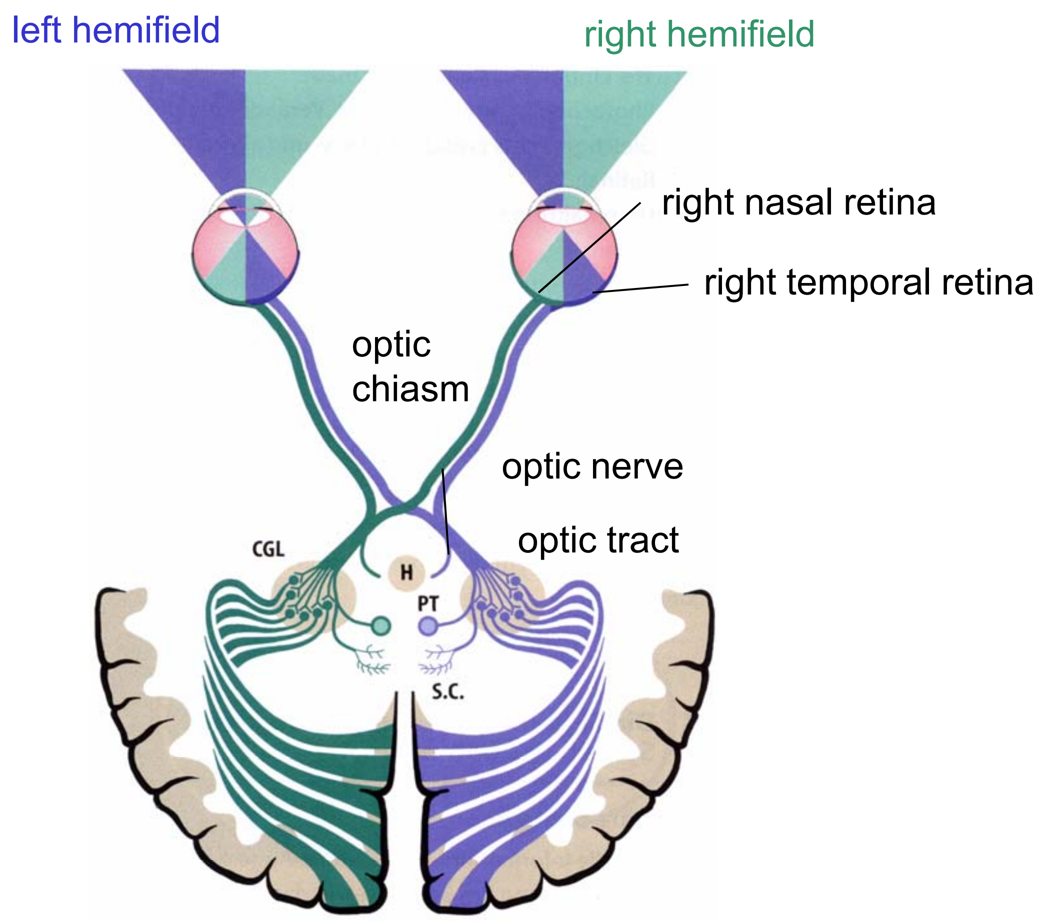
Stereograms

Invented by
Sir Charles Wheatstone,
1838

Anaglyphs

Public Library,
Stereoscopic Looking Room,
Chicago, by Phillips, 1923

3D vision



Primary visual cortex

[Schmidt & Lang, 2005]

Random dot stereogram

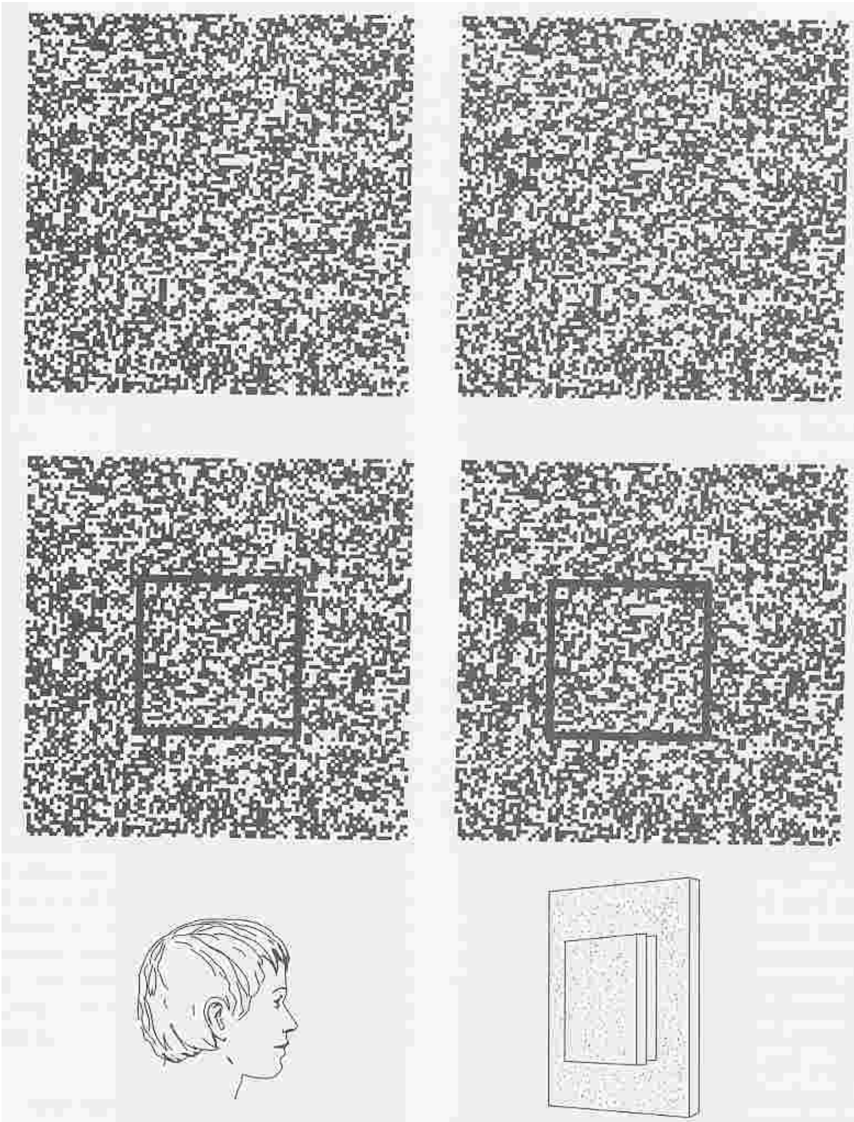


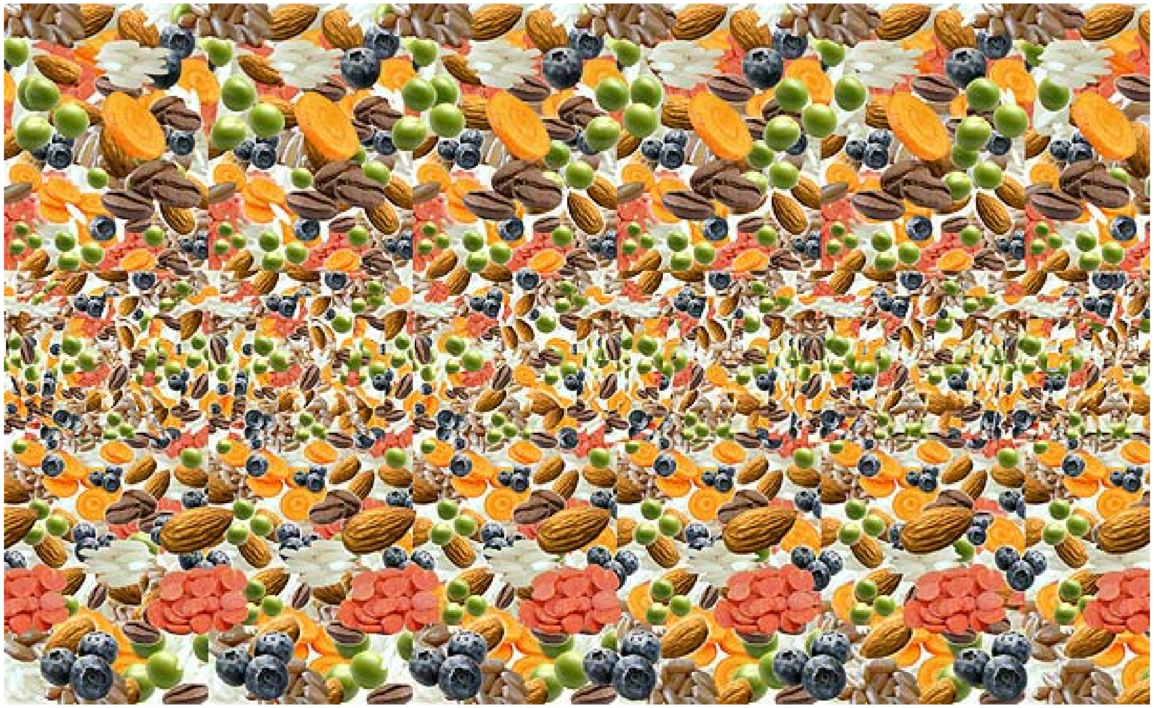
FIGURE 8.13

1	0	1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1	0	0
0	0	1	1	0	1	1	0	1	0
0	1	0	Y	A	A	B	B	0	1
1	1	1	X	B	A	B	A	0	1
0	0	1	X	A	A	B	A	1	0
1	1	1	Y	B	B	A	B	0	1
1	0	0	1	1	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0

1	0	1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1	0	0
0	0	1	1	0	1	1	0	1	0
0	1	0	A	A	B	B	X	0	1
1	1	1	B	A	B	A	Y	0	1
0	0	1	A	A	B	A	Y	1	0
1	1	1	B	B	A	B	X	0	1
1	0	0	1	1	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0

Bela Julesz, 1971

Autostereograms

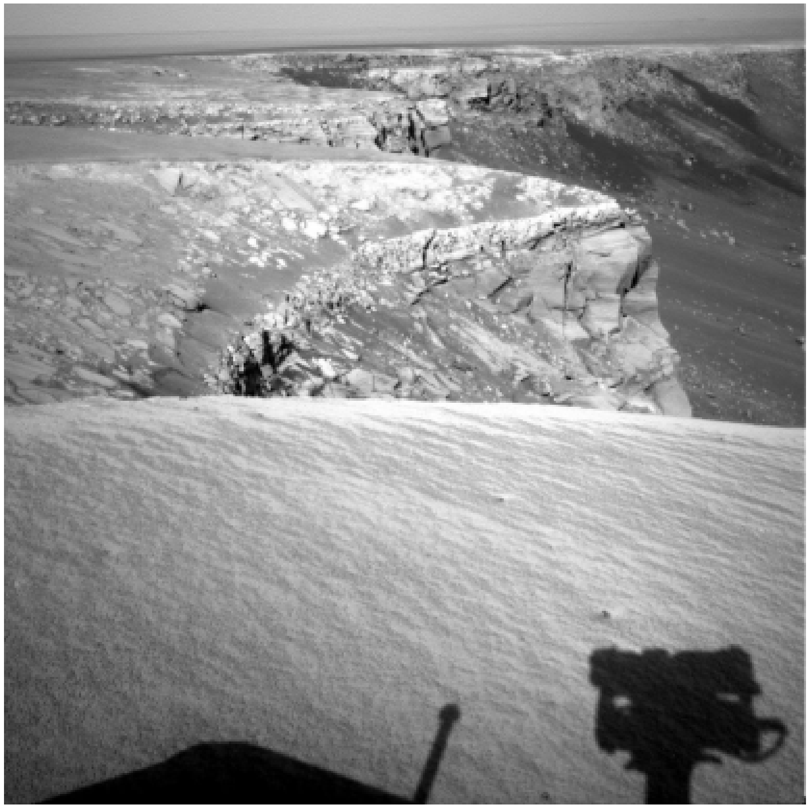


Exploit disparity as depth cue using single image.

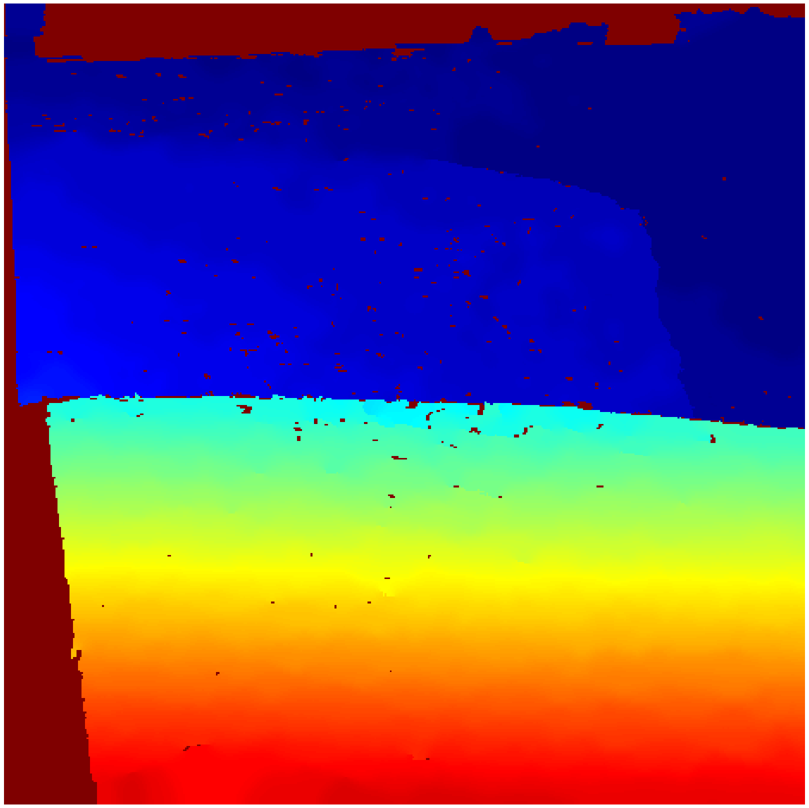
(Single image random dot stereogram, Single image stereogram)

Image from magiceye.com

3D vision



Left view



Depth image

MER rover 'Opportunity' at Victoria Crater

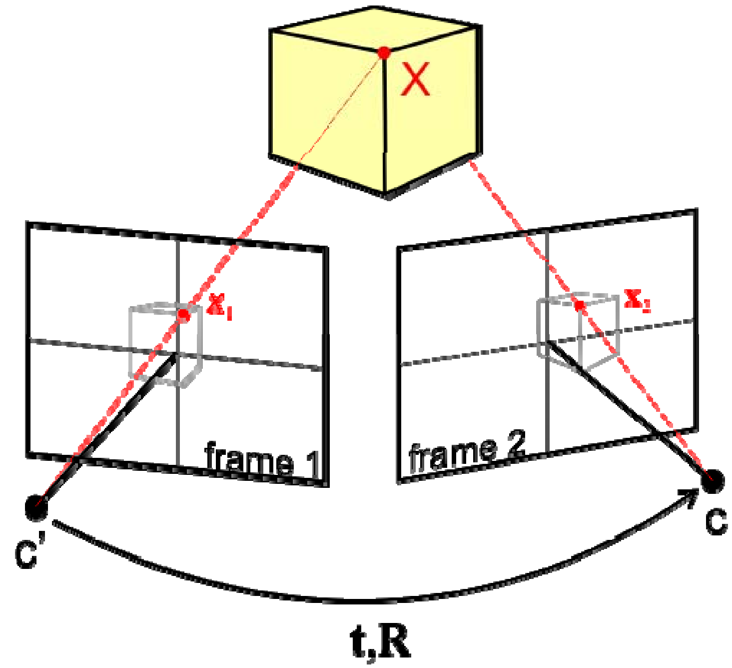
Two view geometry

Perspective projection

- in homogeneous coordinates
- 3D points $\mathbf{X} = [X, Y, Z, W]^T \in \Re^4$
- Image points $\mathbf{x} = [x, y, w]^T \in \Re^3, \quad (w = 1)$
- Perspective projection

$$\lambda \mathbf{x} = \mathbf{X}$$

$$\lambda = Z, \quad x = \frac{X}{Z}, \quad y = \frac{Y}{Z}$$



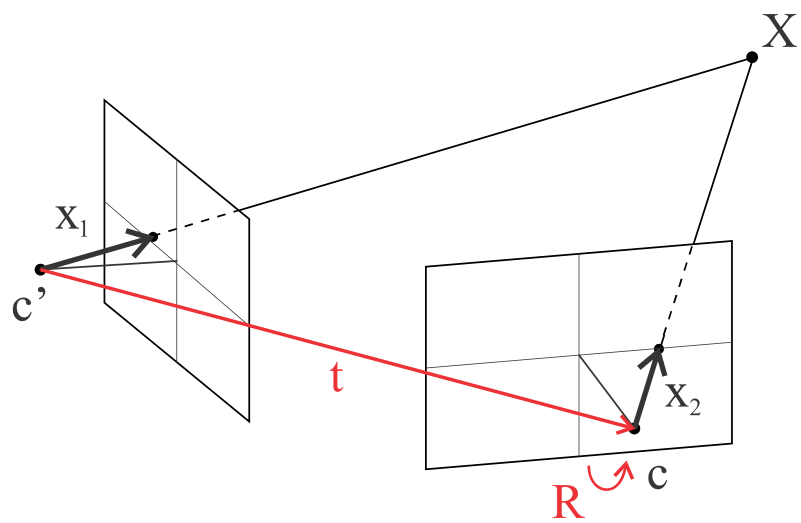
- Rigid body motion $\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{t}$
- Rigid body motion + perspective projection $\Pi = [\mathbf{R}, \mathbf{t}] \in \Re^{3x4} \qquad \lambda \mathbf{x}' = \Pi \mathbf{X} = [\mathbf{R}, \mathbf{t}]\mathbf{X}$

$$\boxed{\lambda_2 \mathbf{x}_2 = \mathbf{R} \lambda_1 \mathbf{x}_1 + \mathbf{t}}$$

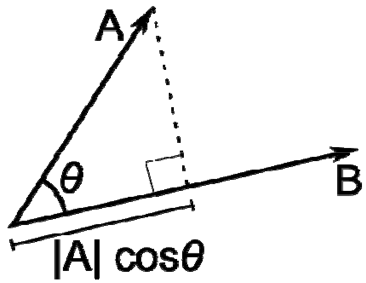
Two view geometry

Epipolar constraint:

$\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{t}$ are coplanar



Insert: Vector products



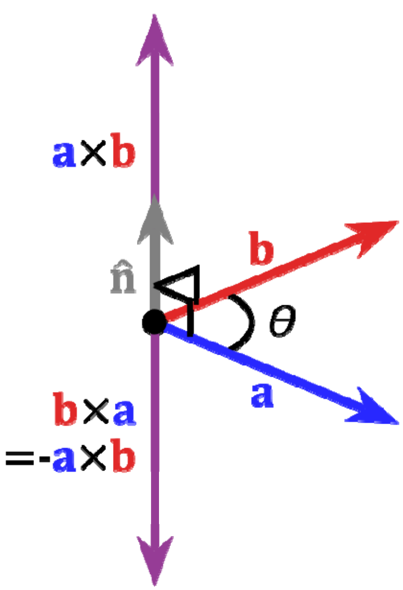
Scalar product

$$\vec{a} \cdot \vec{b} = |a| \cdot |b| \cdot \cos\theta$$

Matrix form of cross product

$$a \times b = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} b = [a_\times] b$$

$$a \times b = [a_x] b \qquad [a_x] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$



$$a \cdot (a \times b) = 0$$

$$b \cdot (a \times b) = 0$$

Two view geometry

Epipolar constraint:

$\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{t}$ are coplanar

$$\underbrace{\mathbf{t}\times\mathbf{x_2}}_{\text{plane normal}} = \mathbf{t}\times(\mathbf{R}\mathbf{x_1}+\mathbf{t})$$

$$= \mathbf{t}\times\mathbf{R}\mathbf{x_1} + \mathbf{t}\times\mathbf{t}$$

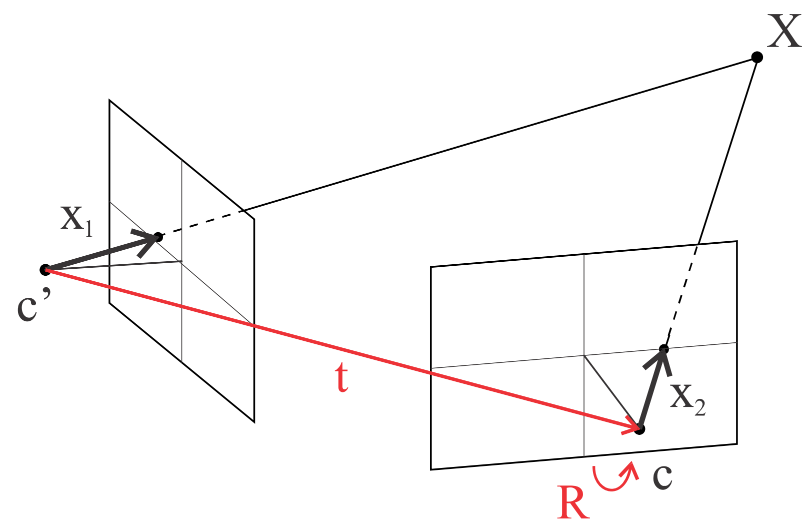
$$\mathbf{x_2}\cdot(\mathbf{t}\times\mathbf{x_2}) = \mathbf{x_2}\cdot(\mathbf{t}\times\mathbf{R}\mathbf{x_1}) = 0$$

$$\mathbf{t}\times\mathbf{R}\mathbf{x}_1 = [\mathbf{t}_\times]\mathbf{R}\mathbf{x}_1$$

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

Essential matrix
[Longuet-Higgings 1981]

$$\mathbf{x}_2{}^T\mathbf{E}\mathbf{x}_1 = 0$$



Two view geometry

Epipolar constraint:

$\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{t}$ are coplanar

$$\underbrace{\mathbf{t}\times\mathbf{x}_2}_{\text{plane normal}} = \mathbf{t}\times(\mathbf{R}\mathbf{x}_1+\mathbf{t}) = \mathbf{t}\times\mathbf{R}\mathbf{x}_1 + \mathbf{t}\times\mathbf{t}$$

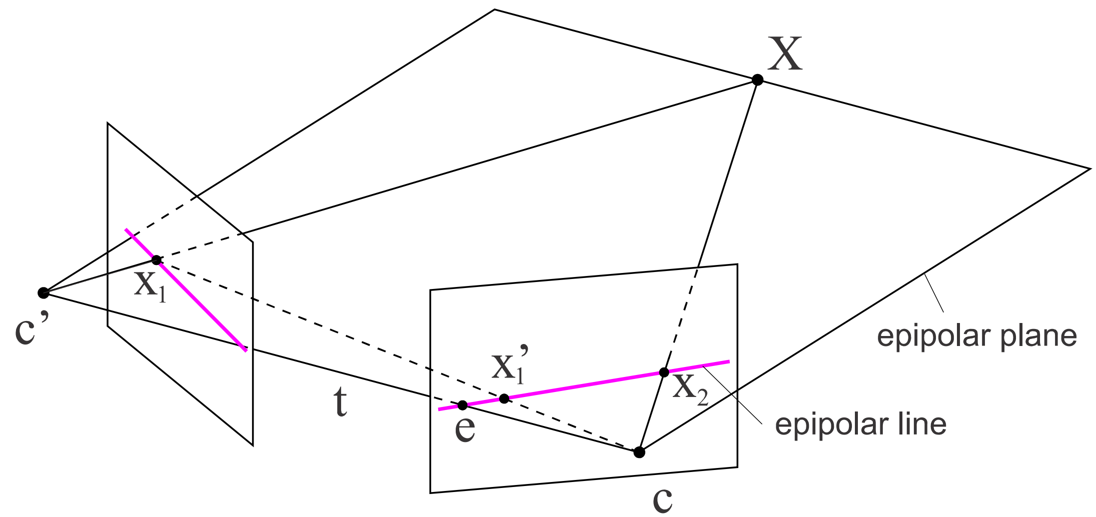
$$\mathbf{x}_2\cdot(\mathbf{t}\times\mathbf{x}_2) = \mathbf{x}_2\cdot(\mathbf{t}\times\mathbf{R}\mathbf{x}_1) = 0$$

$$\mathbf{t}\times\mathbf{R}\mathbf{x}_1 = [\mathbf{t}_\times]\mathbf{R}\mathbf{x}_1$$

$$\mathbf{E} = [\mathbf{t}_\times]\mathbf{R}$$ Essential matrix [Longuet-Higgings 1981]

$$\mathbf{x}_2^{\,T}\mathbf{E}\mathbf{x}_1 = 0$$

- Maps a point $\mathbf{x}_1$ to an epipolar line $\mathbf{E}\mathbf{x}_1$
- 5 independent parameters (up to scale)



Single moving camera: **t**, **R** unknown → prior to 3D reconstruction, do motion estimation

Stereo vision: **t**, **R** fixed and known from calibration
→ epipolar geometry can be pre-calculated
→ use rectification + epipolar constraint for correspondence search

today's lecture

Stereo vision

What are some possible algorithms?

• Feature based:	match “features” and interpolate
• Edge based:	match edges and interpolate
• Dense methods:	match all pixels with windows

- use optimization:
  - iterative updating
  - dynamic programming
  - energy minimization (regularization, stochastic)
  - graph algorithms

Stereo vision

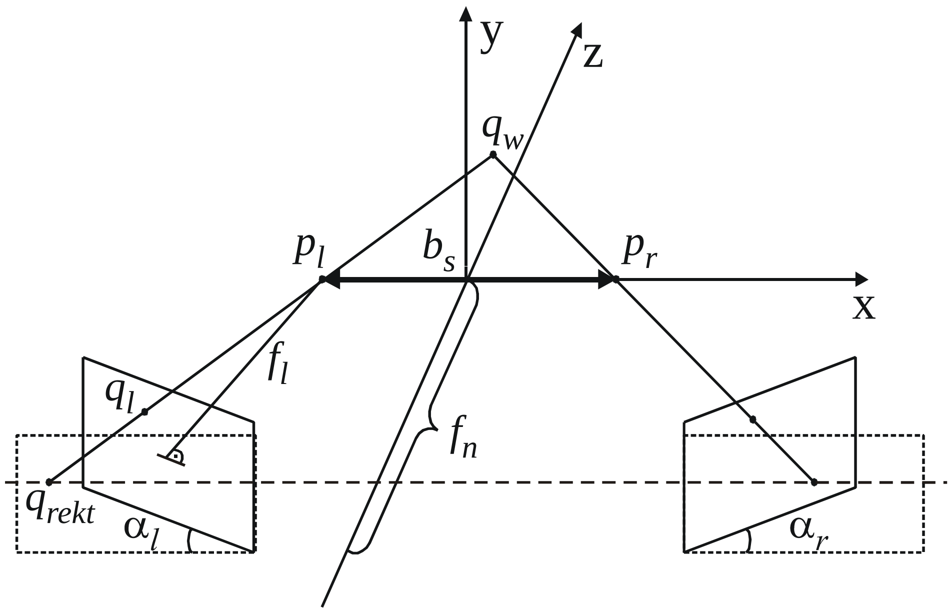
Image processing pipeline

- Image acquisition & Pre-processing
- Correspondence search
- Passive triangulation

Stereo vision

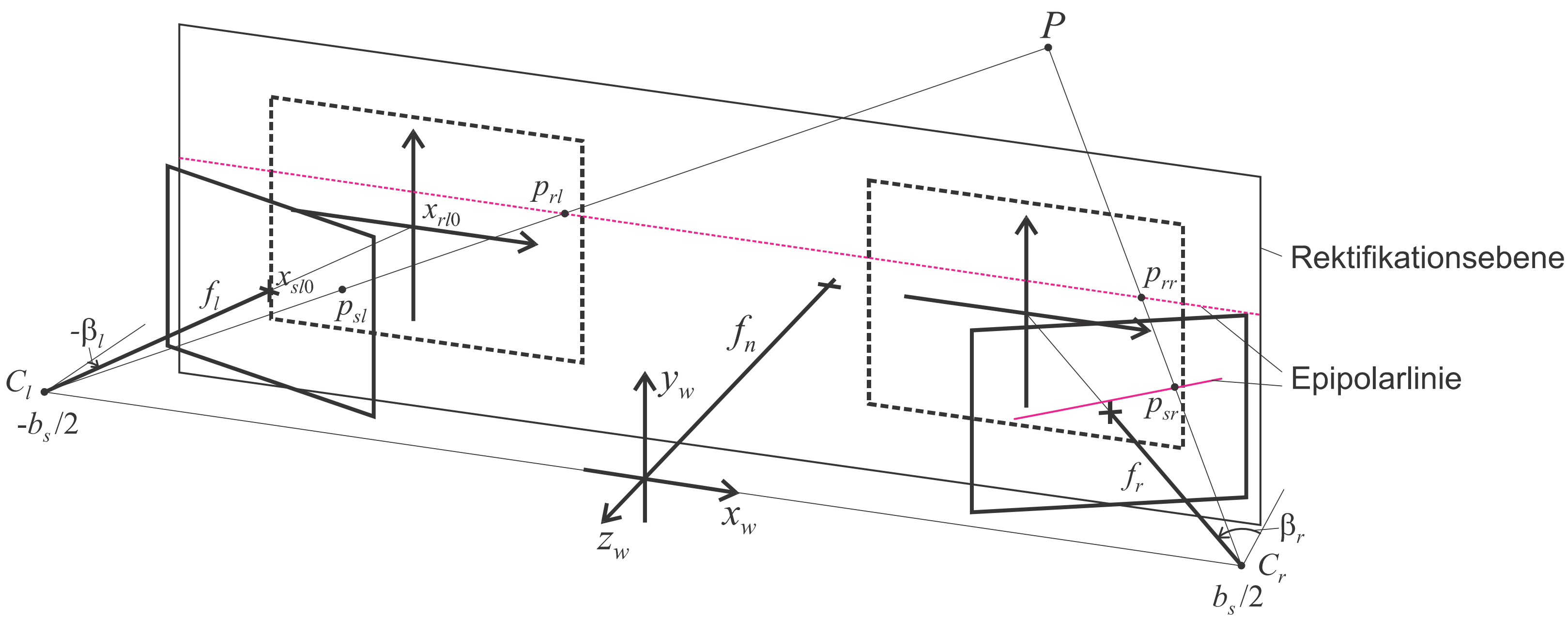
Image processing pipeline

- Image acquisition & Pre-processing
  - Image correction (e.g. removing of lens distortion)
  - Rectification

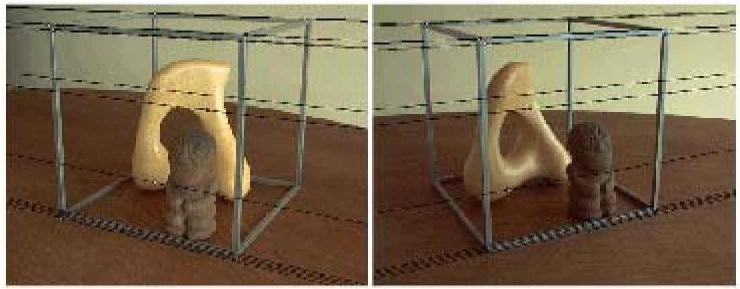


Rectification

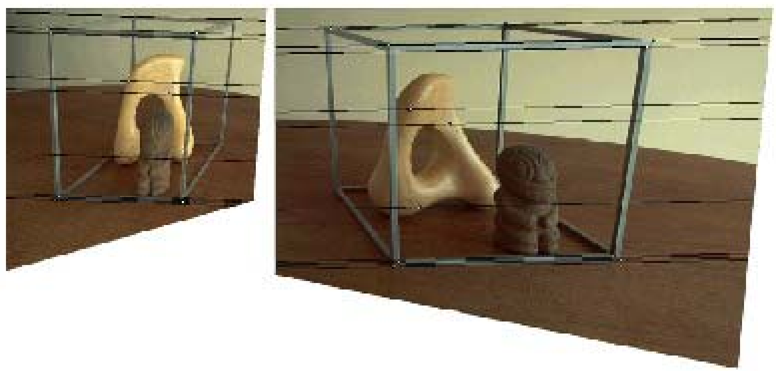
- Project each image onto same plane, which is parallel to the baseline
- Resample lines (and shear/stretch) to minimize distortion



Rectification



(a) Original image pair overlayed with several epipolar lines.

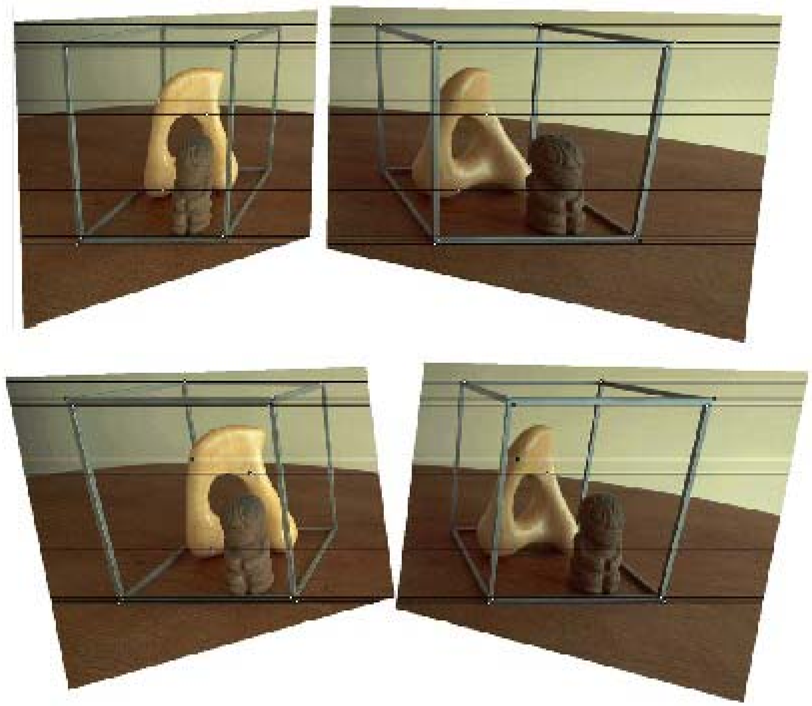


(b) Image pair transformed by the specialized projective mapping $\mathbf{H}_p$ and $\mathbf{H}'_p$. Note that the epipolar lines are now parallel to each other in each image.

BAD!

C. Loop, Z. Zhang: Computing rectifying homographies for stereo vision. CVPR 1999

Rectification



(c) Image pair transformed by the similarity $\mathbf{H}_r$ and $\mathbf{H}'_r$. Note that the image pair is now rectified (the epipolar lines are horizontally aligned).

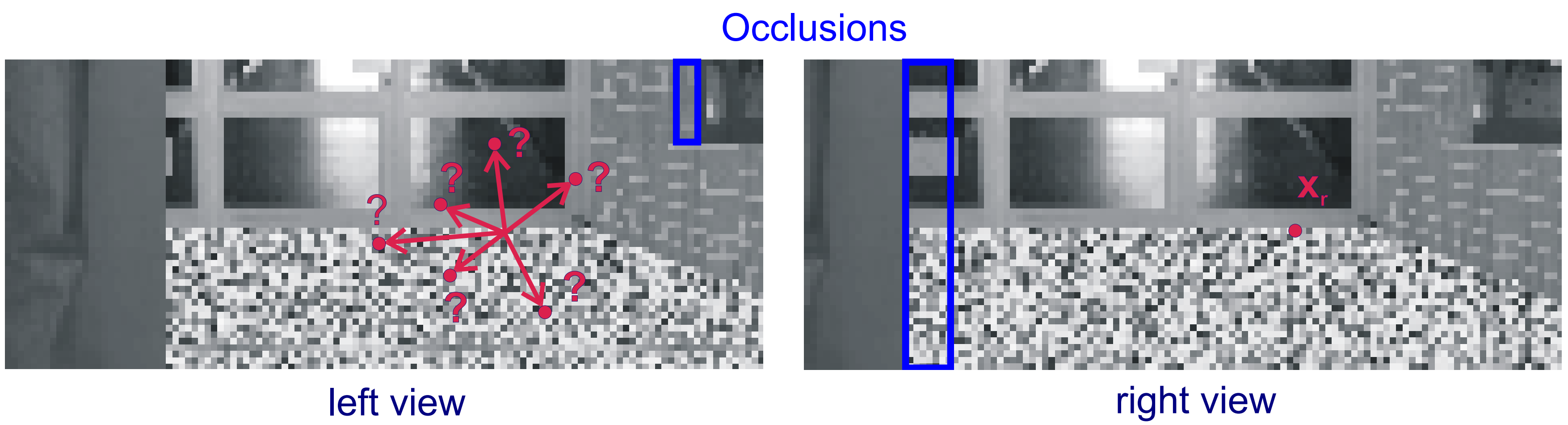
(d) Final image rectification after shearing transform $\mathbf{H}_s$ and $\mathbf{H}'_s$. Note that the image pair remains rectified, but the horizontal distortion is reduced.

GOOD!

C. Loop, Z. Zhang: Computing rectifying homographies for stereo vision. CVPR 1999

Image processing pipeline

- Image acquisition & Pre-processing
- Correspondence search
  - Difficulties due to ambiguities or occlusions

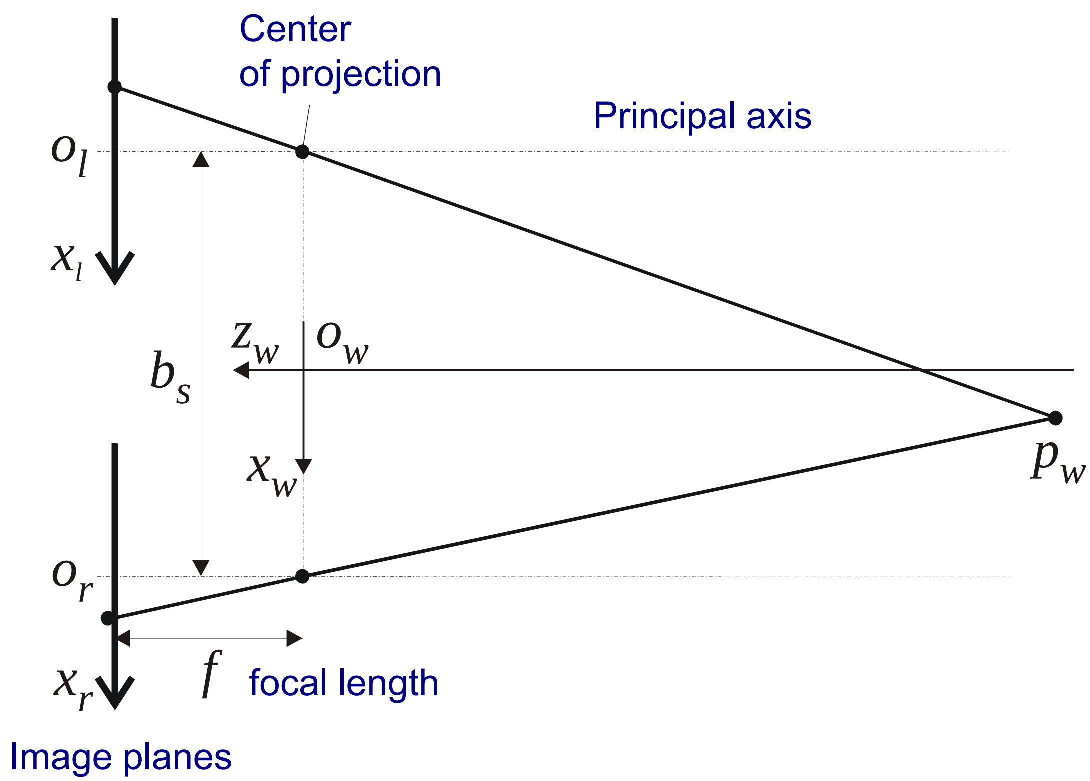


Stereo vision

Image processing pipeline

- Image acquisition & Pre-processing
- Correspondence search
- Passive triangulation

Passive triangulation



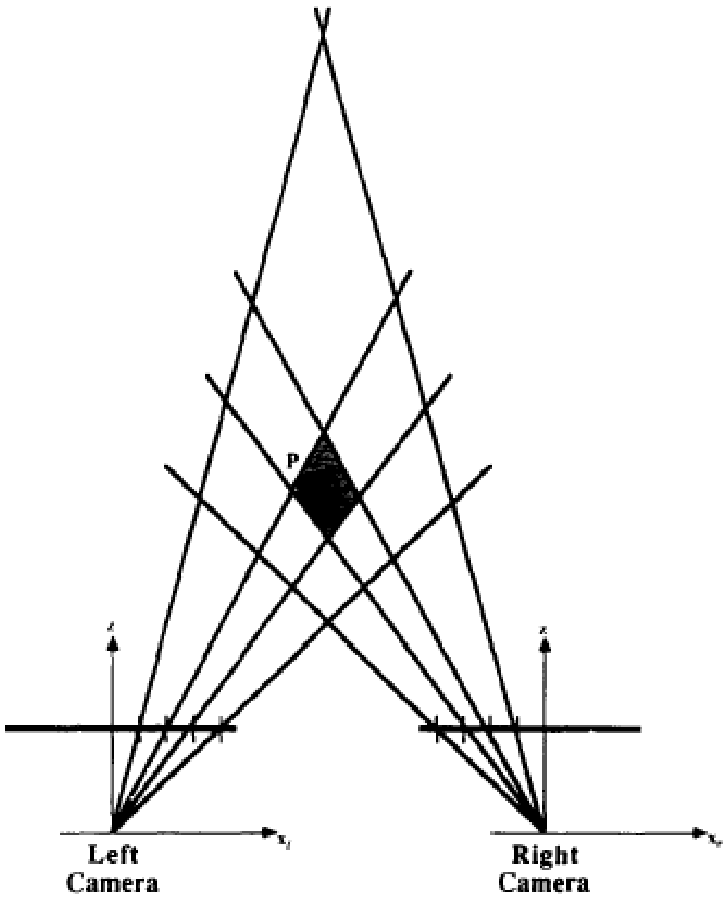
Disparity: $d_x = x_l - x_r$

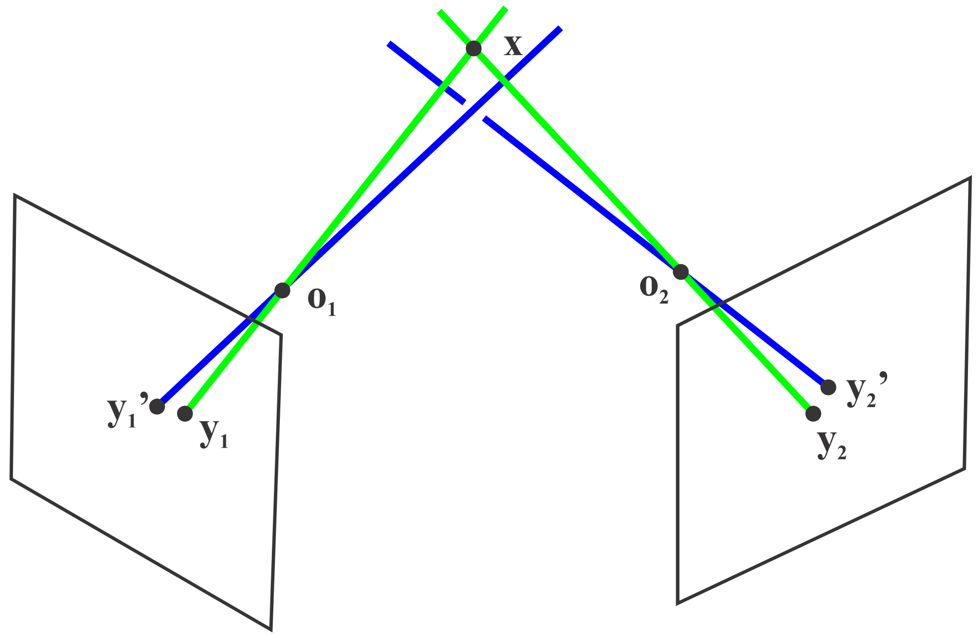
Range: $$x_w = \frac{b_s(x_l + x_r)}{2d_x}$$

$$z_w = -\frac{f\,b_s}{d_x}$$

Passive triangulation

Beware of triangulation error due to match uncertainty (alias effects)





Because of missalignment, reconstructed (blue) rays usually don't intersect !

Stereo vision

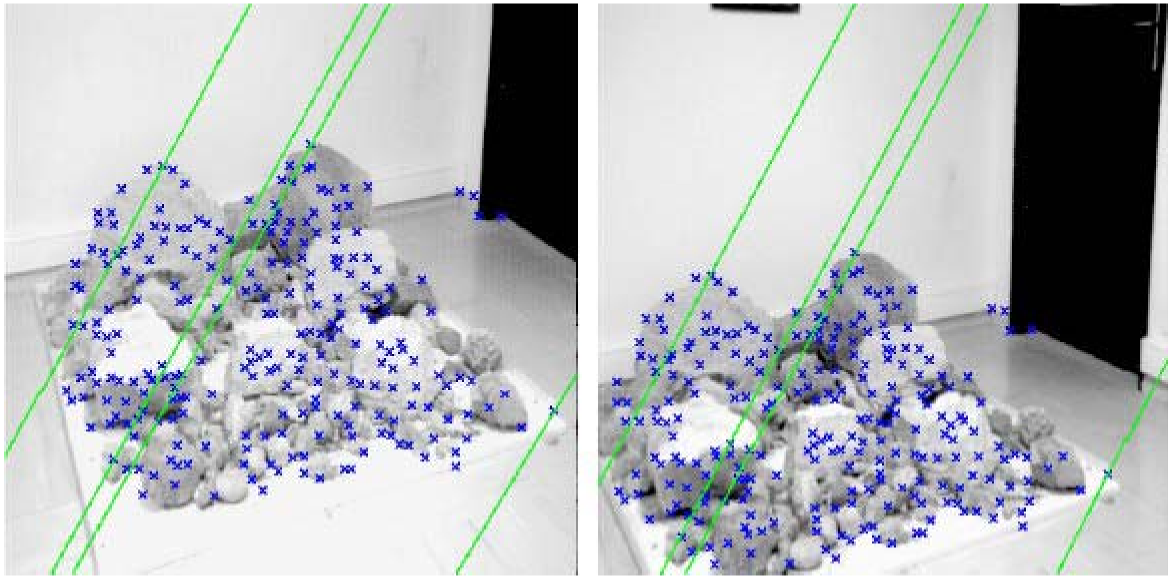
Image processing pipeline

- Image acquisition & Pre-processing
- Correspondence search
- Passive triangulation

Correspondence search

Feature based:

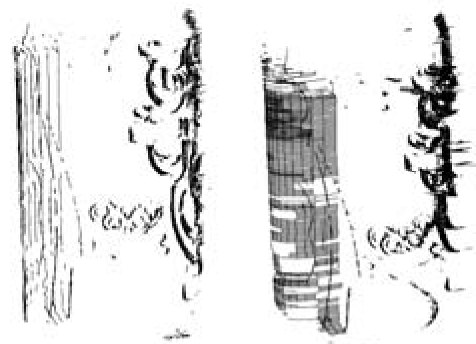
- Extract feature points
- Match features using outlier rejection (epipolar constraint)
- Triangulate



Correspondence search

Edge based:

- Extract edge contours and match edge segments (edgels)

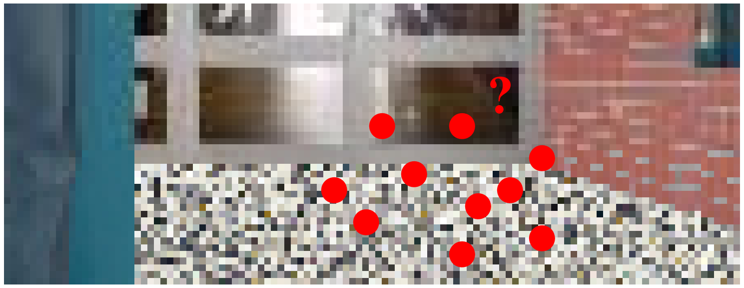


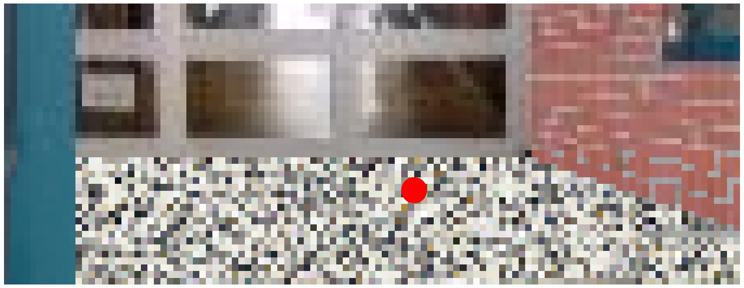
R. Szeliski, R. Weiss: Robust shape recovery from occluding contours using a linear smoother. IJCV 1998

Correspondence search

Image based:

- Find correspondes for every pixel in reference image



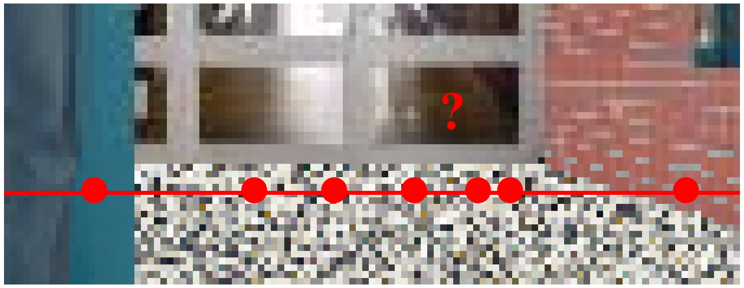


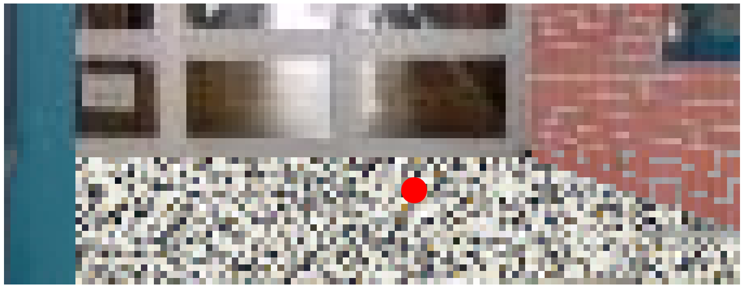
- Looks difficult ? Use stereostopic constraints !

Correspondence search

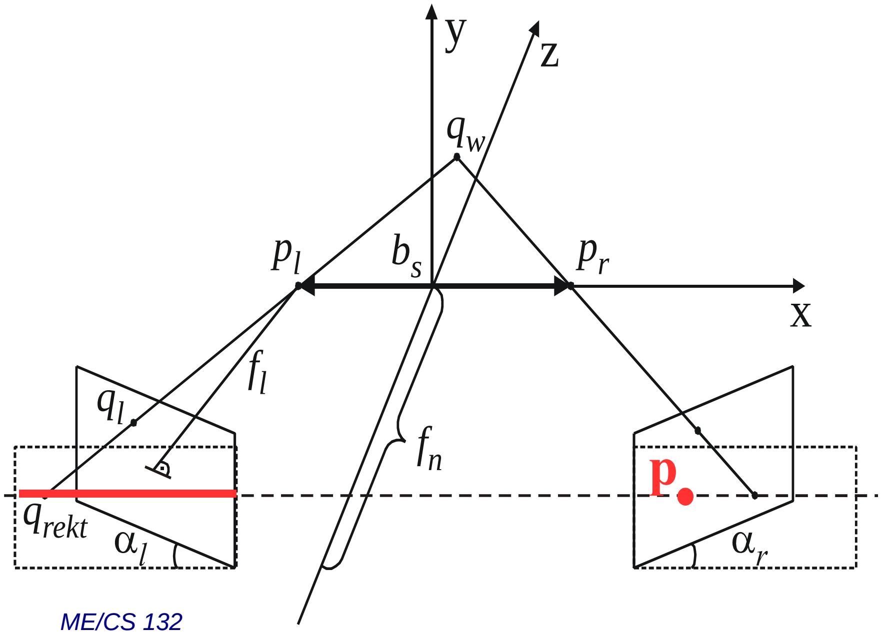
Image based:

- Find correspondes for every pixel in reference image





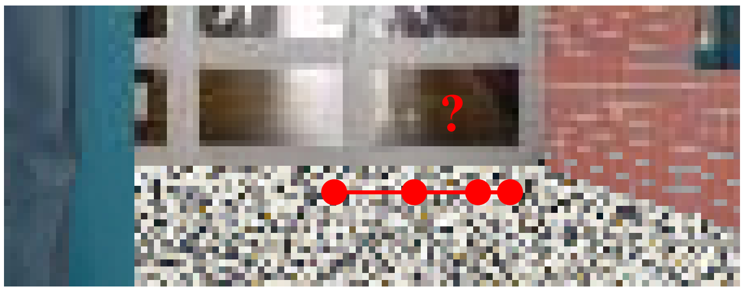
  - Epipolar constraint

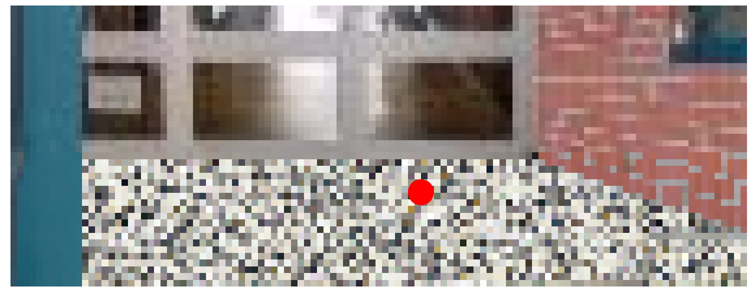


Correspondence search

Image based:

- Find correspondes for every pixel in reference image



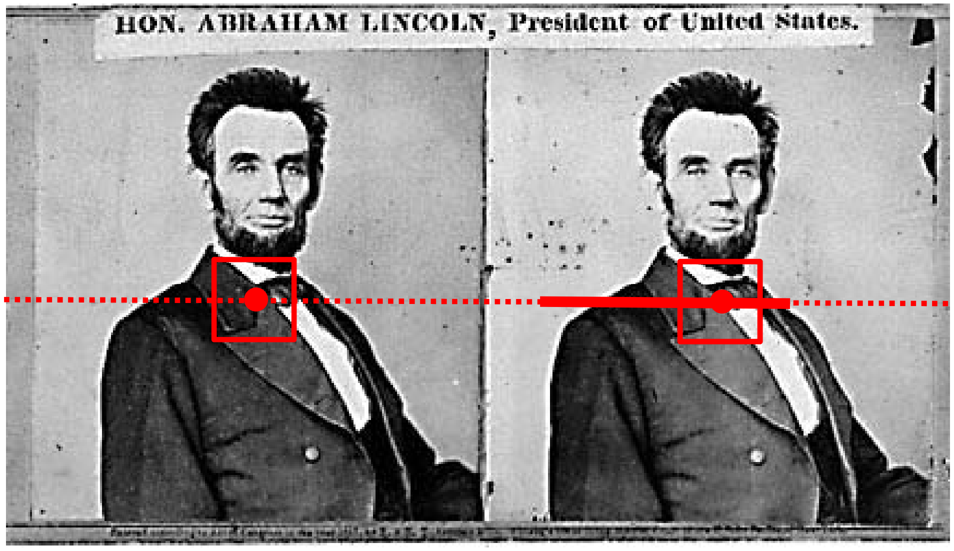


  - Epipolar constraint
  - Limited disparity range

  - No transparent objects: Uniqueness constraint [Marr & Poggio, 1976]
  - Solid objects: Continuity constraint [Marr & Poggio, 1976]

  - Disparity gradient limit (humans: < 1.0) [Burt & Julesz, 1980]
  - Order constraint

Basic algorithm



For each epipolar line

For each pixel in the reference image

- compare with each pixel on same epipolar line within search range
- pick pixel with minimum match cost

Improvement: Use match windows!

Example algorithm: SAD real time stereo

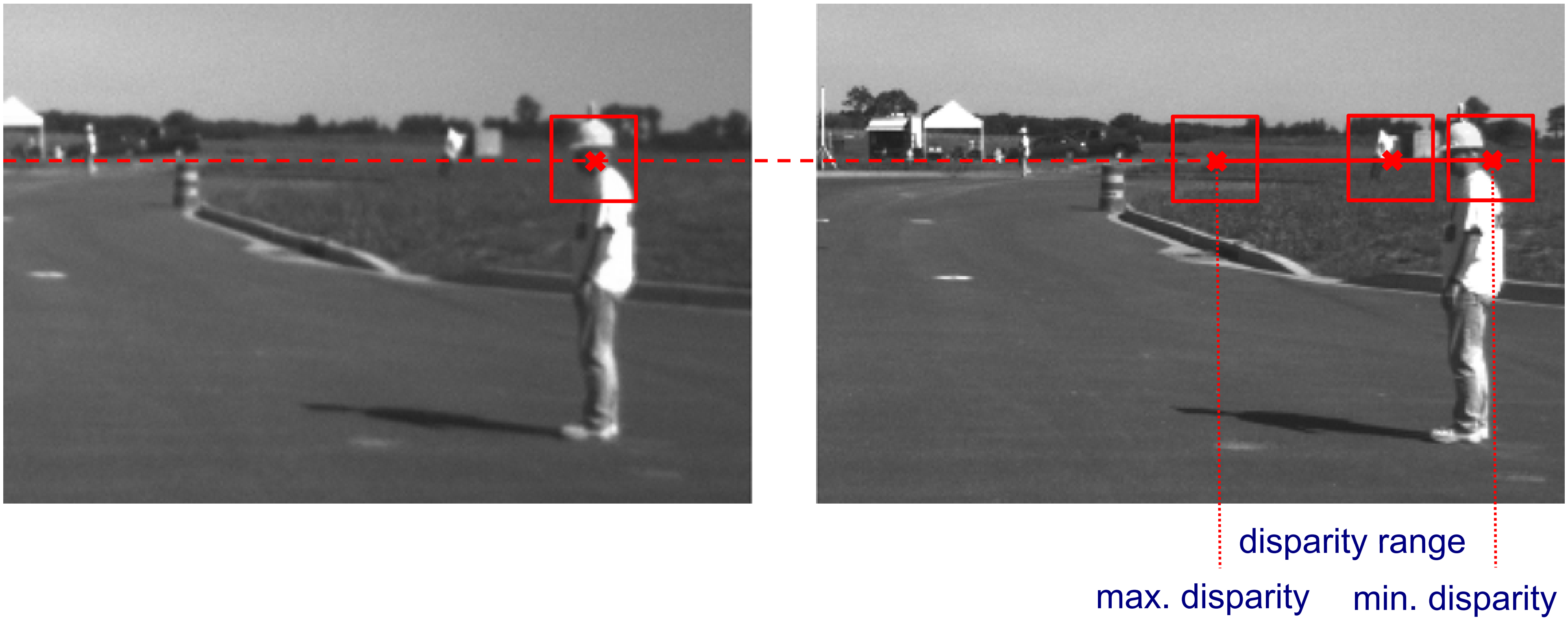
- Uses a Sum of Absolute Differences (SAD) to calculate matching scores
- Runs on many mobile robot platforms, including the MER exploration rovers on Mars

L

R

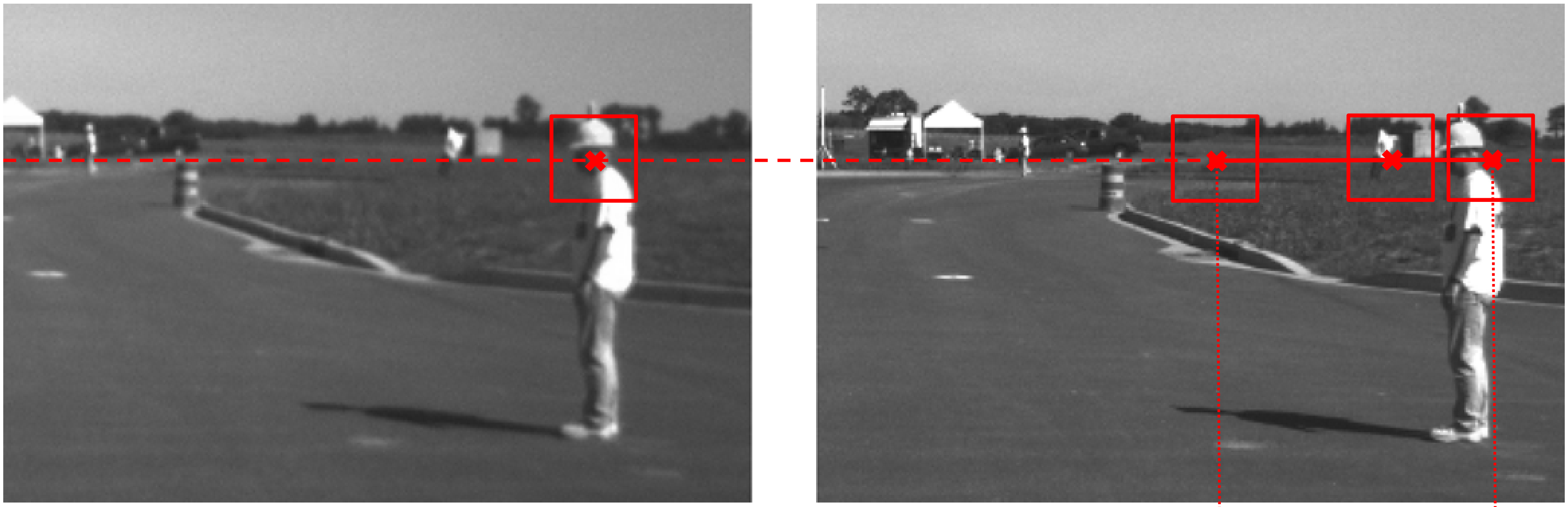
SAD real time stereo

SAD

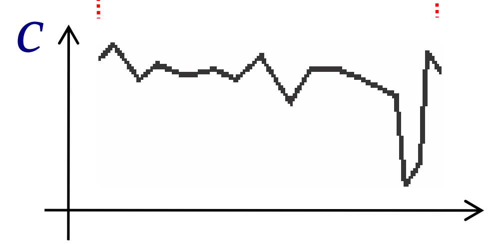


SAD real time stereo

SAD



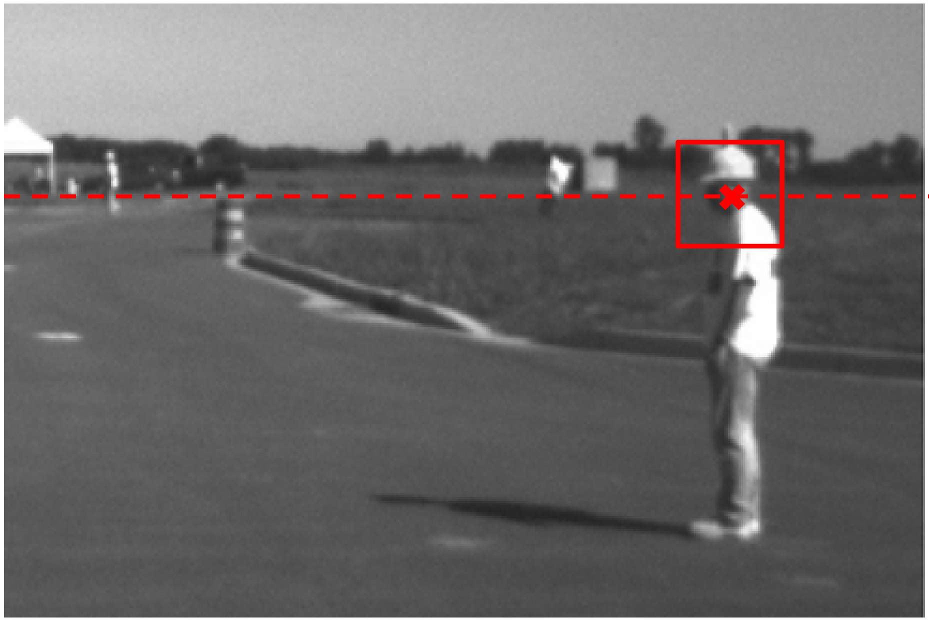
- Calculate scores for all pixels in the reference view and all displacements [d_{min} … d_{max}]

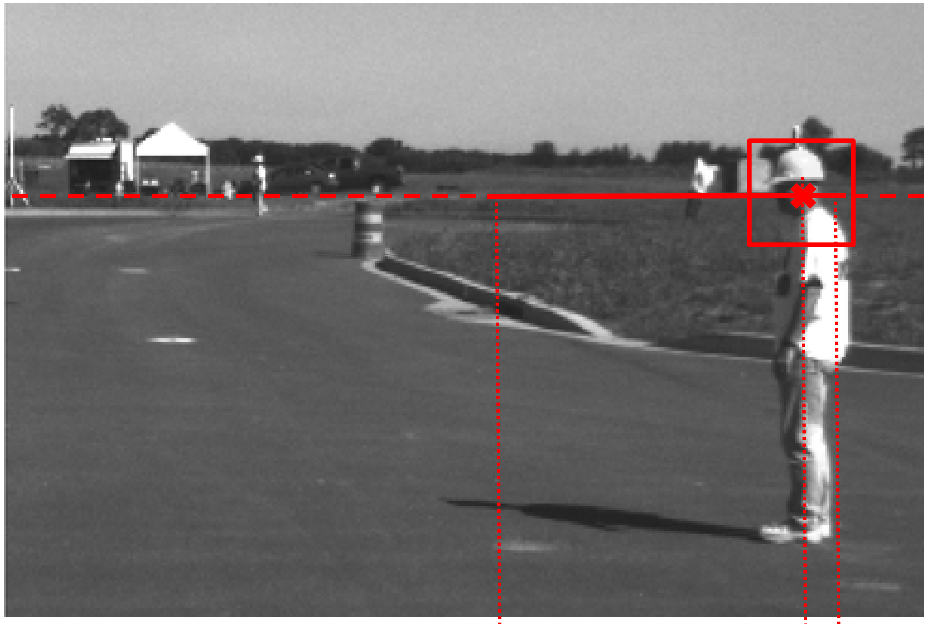


SAD Correlation score

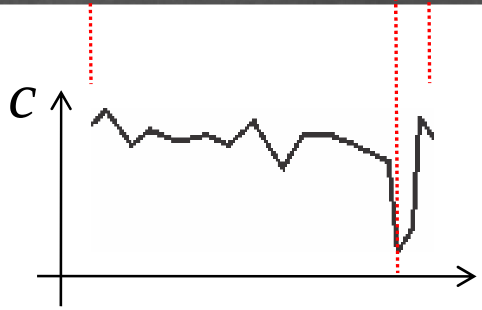
SAD real time stereo

SAD





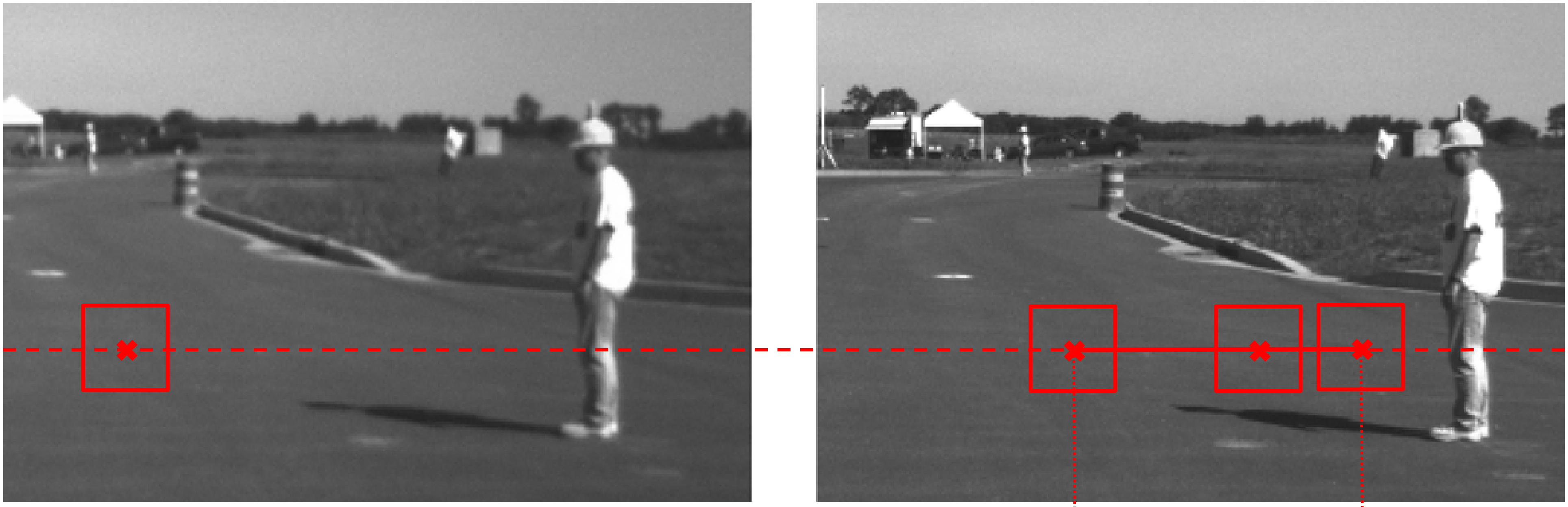
- Find the corresponding pixel with minimum SAD score
- And the winner takes all !



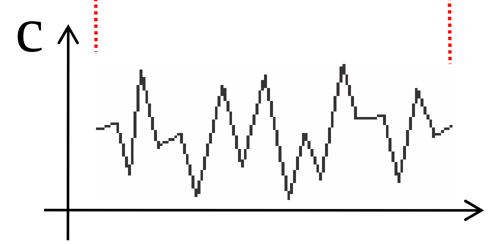
SAD Correlation score

SAD real time stereo

SAD



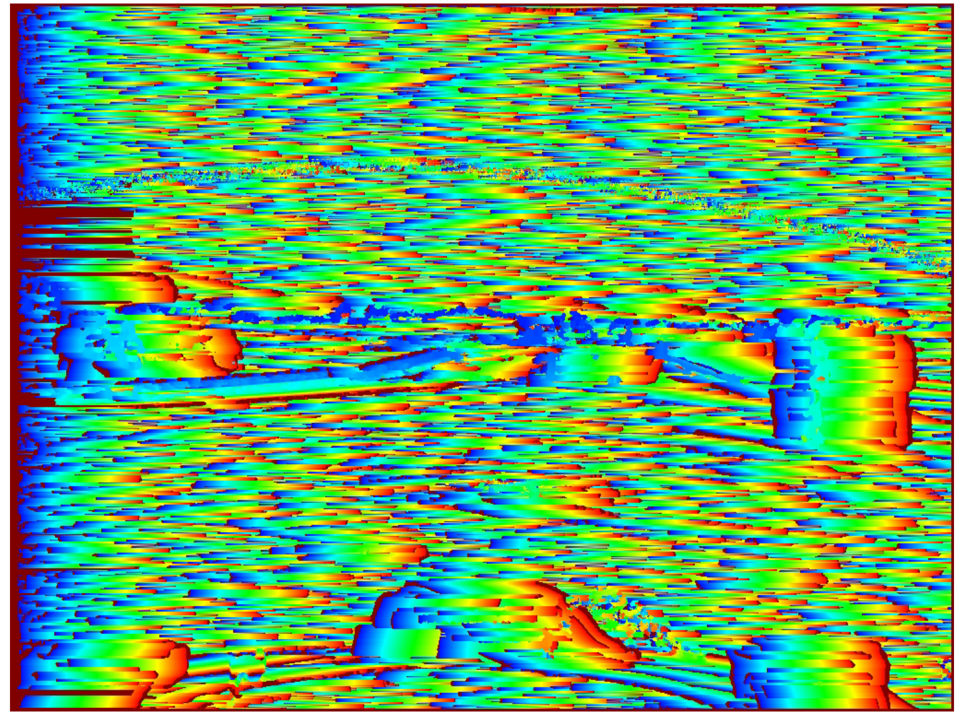
But what do we do here?



SAD Correlation score

SAD real time stereo

SAD, 7x7 correlation windows



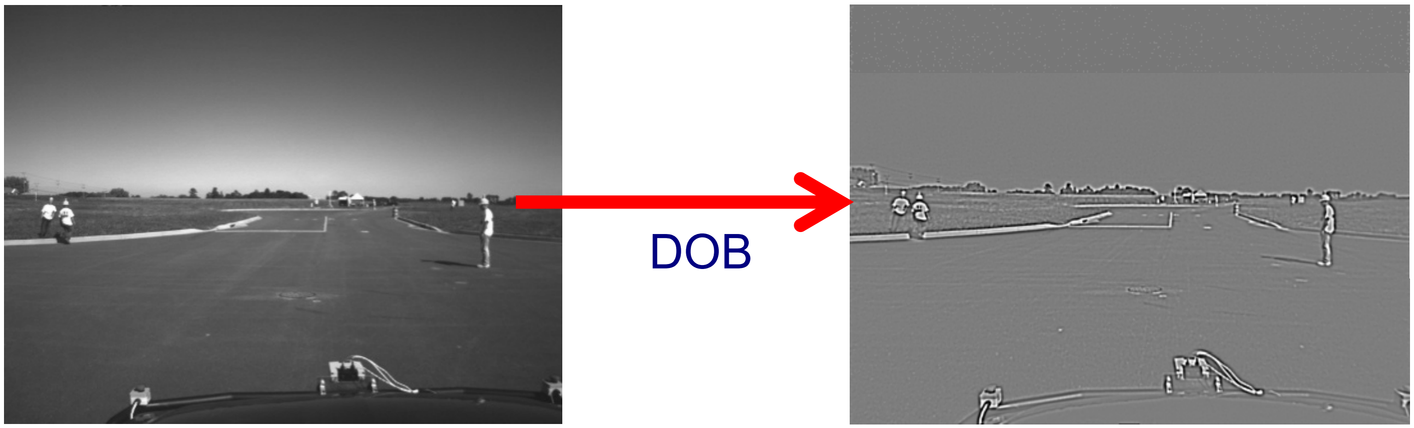
How can we get better results?

- Image pre-filtering to enhance contrast and reduce noise
- Post processing to clean-up

near far

Pre filter

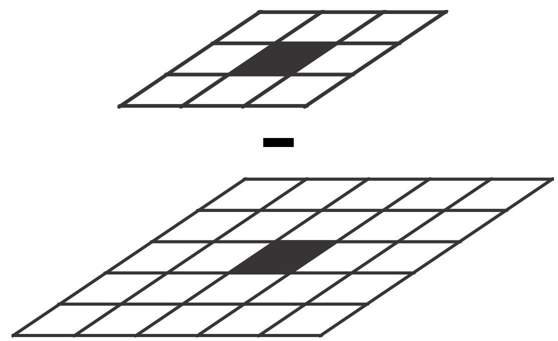
SAD, 7x7 correlation windows



Differences of boxes (DOB) bandpass filter

- Average I over small and large window and subtract

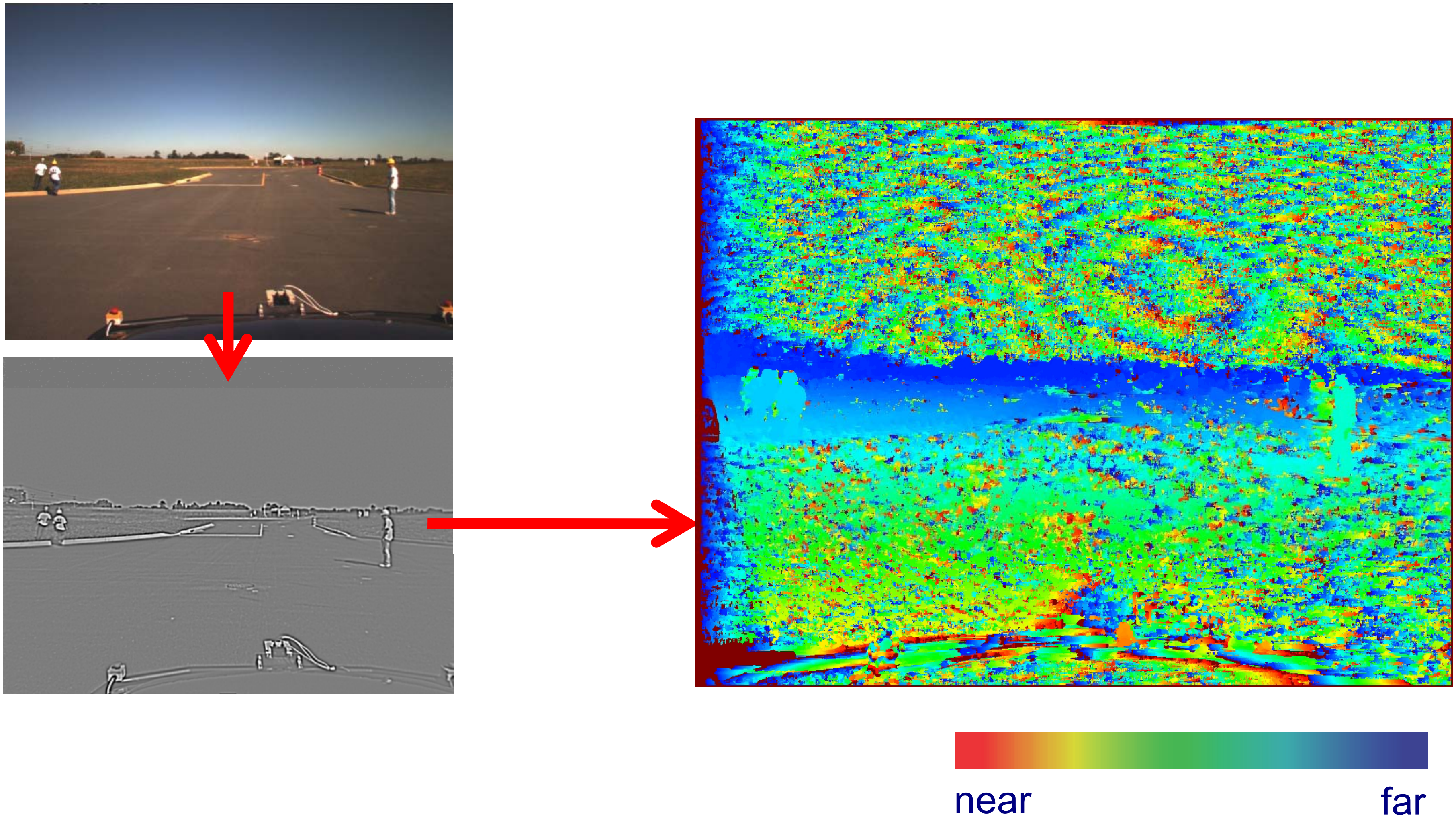
$$DOB(x,y) = \underset{w_{small}}{mean}(I(x,y)) - \underset{w_{large}}{mean}(I(x,y))$$



- Crude approximation of a Laplace filter (LoG) But very fast!
- Think of it as a background subtraction + noise reduction

Pre filter

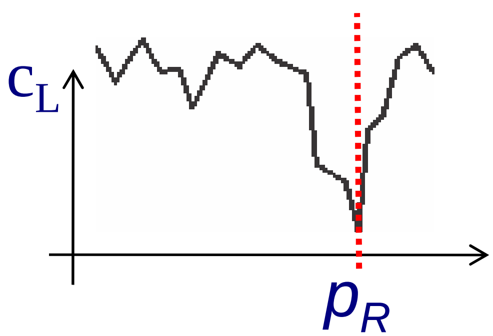
SAD, 7x7 correlation windows, DOB pre-filter



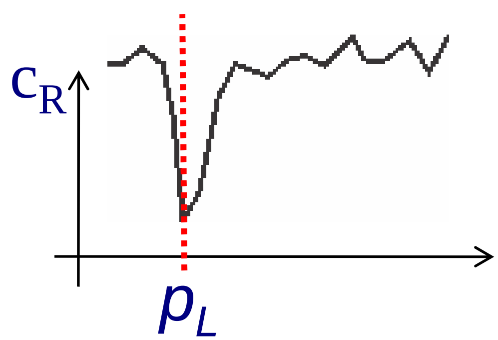
Post filter

Left to right consistency check

- Make sure the matching result is symmetrical:
- Check if a pixel in the reference view p_L for which we found a winning correspondence partner in the other view p_R is also the winning pixel for the correspondence search of p_R.



SAD Correlation score
for $p_L \rightarrow$ right image



SAD Correlation score
for $p_R \rightarrow$ left image

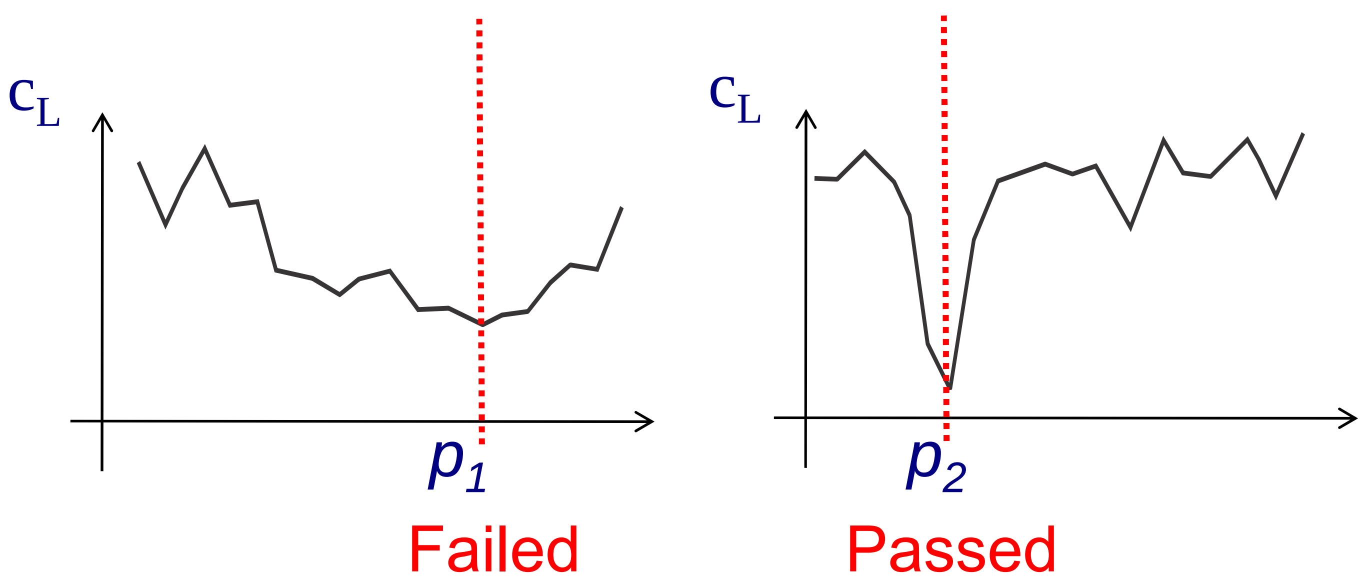
$$p_L \leftrightarrow p_R$$

Post filter

Curvature check:

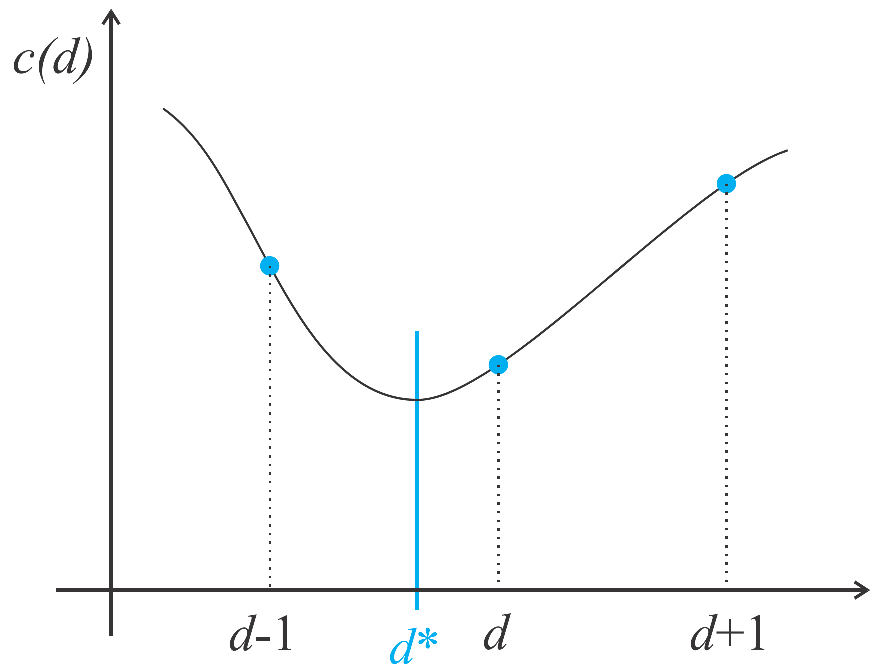
- Look at the curvature at the peak of the correlation function
- If the curvature is too flat, we don't have a high confidence in the winning disparity → reject this pixel
- Simple measure for SAD curvature:

$$curv(x,y) = c(x-1,y) - 2c(x,y) + c(x+1,y) > k$$



Sub-pixel precision

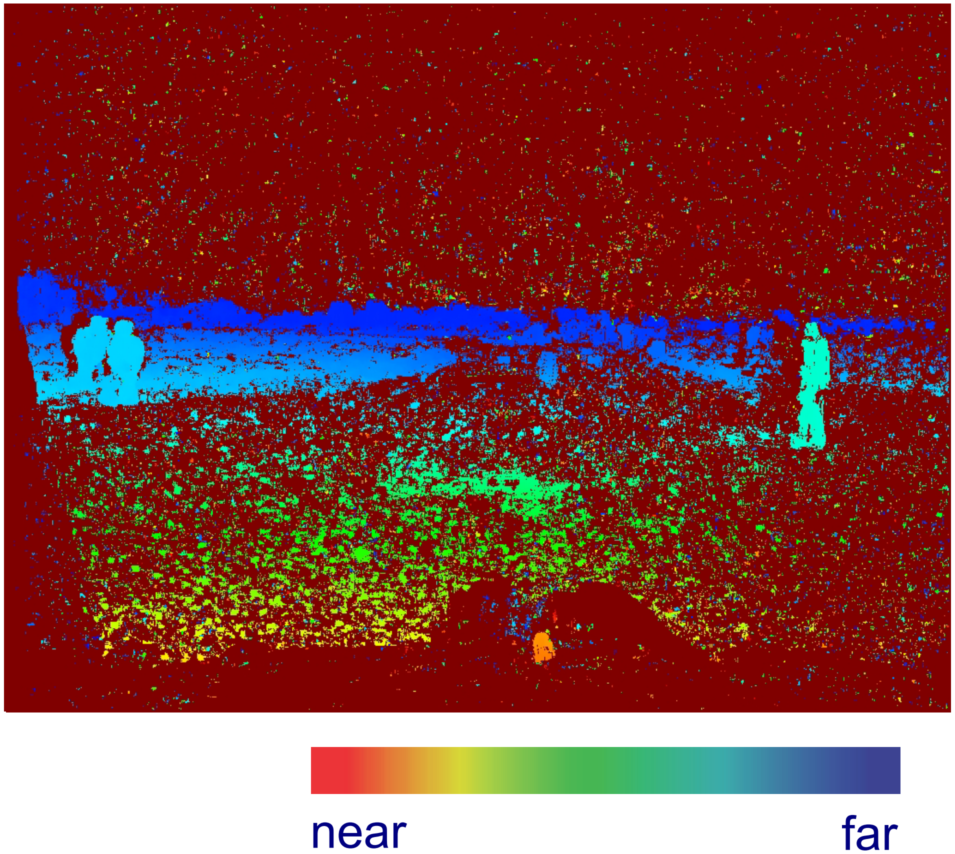
Use bilinear interpolation to estimate the minimum position



$$d^* = \frac{c(-1) - c(1)}{2(c(1) + c(-1) - 2c(0))}$$

Post filter

SAD, 7x7 correlation windows, DOB pre-filter, LR & curv check

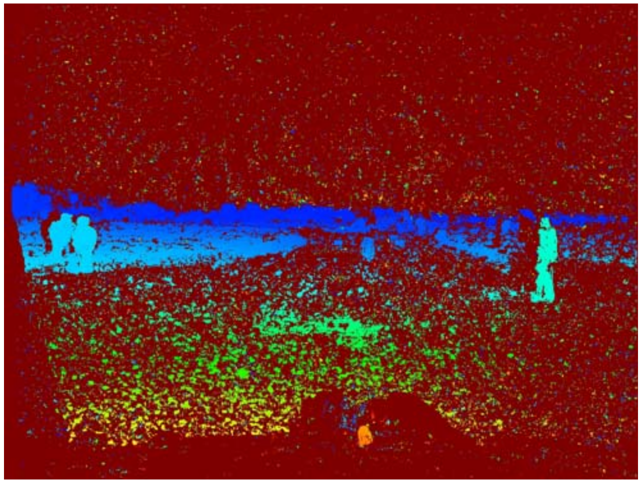


Post filter

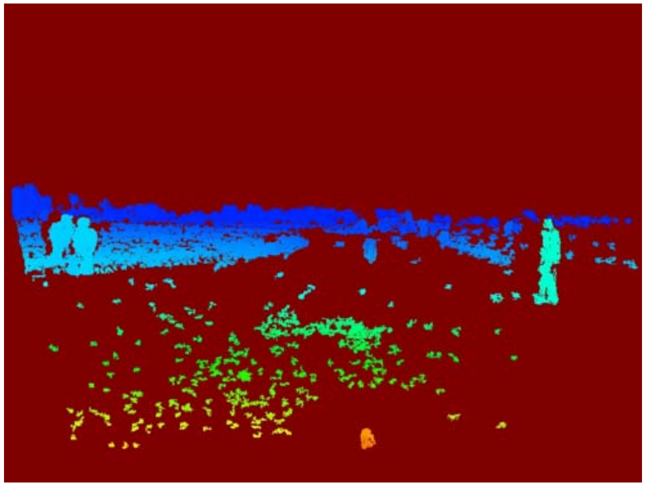
Blob filter

- Eliminate small blobs under a 'specific' size

Left view



Left-right consistency check



Blob filter

How can we further improve our stereo result?
Let's get complicated!

Optimization methods

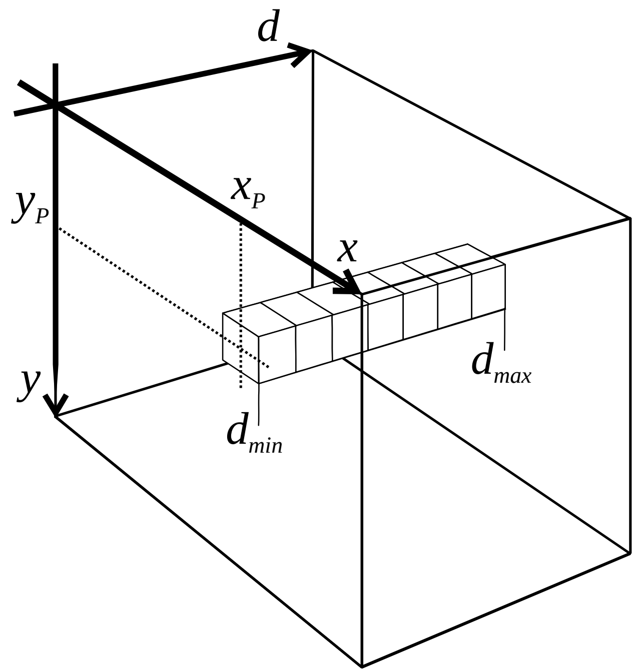
Optimize the correlation values before picking the winning disparity

How do we optimize?

- define an energy function *E*

$$E = f(s(x, y, d))$$

- pick the solution for which the energy is minimal



Disparity space image $s(x,y,d)$

Optimization methods

Local Optimization:

Optimize the disparity of every pixel independently

Semi-global optimization:

Optimize the disparity for groups of pixels

Global optimization:

Optimize the disparity for the whole image simultaneously

Optimization methods

Local Optimization:

- Fixed matching window

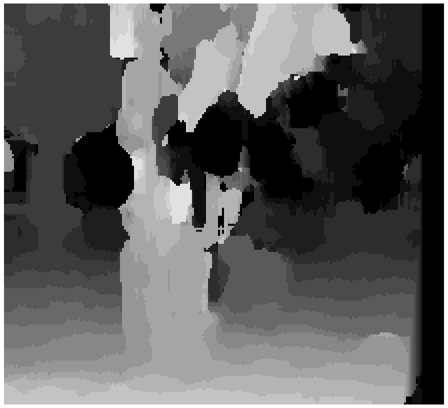
SAD scores: $$E(x,y,d) = \sum_{(u,v)\in w(x,y)} \left| I_L(u+d,v) - I_R(u,v) \right|$$

How big should the neighborhood be?

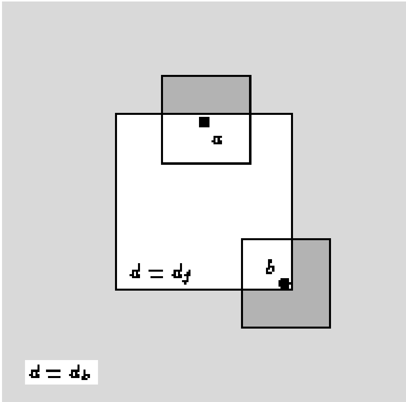
Optimization methods

- Smaller neighborhood: more details
- Larger neighborhood: fewer isolated mistakes

w = 3

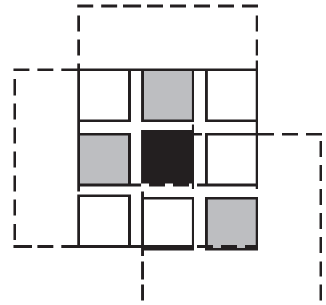


w = 20



Optimization methods

Local Optimization:



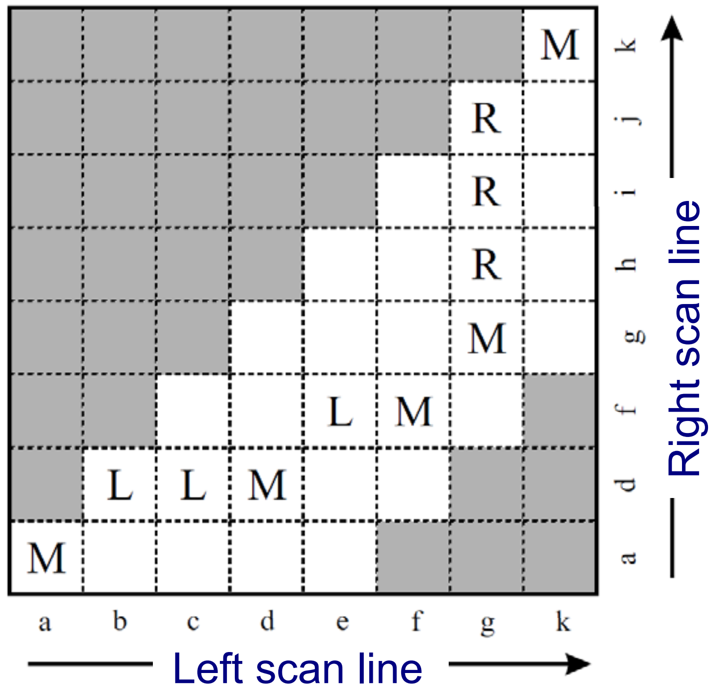
- Shifted windows
  Pick the minimum matching score
  of all surrounding windows

- Color based adaptive windows
  adaptive weights based on color similarity
  segmentation based

Optimization methods

Semi-global optimization:

- Dynamic Programming
  - 1D simultaneous optimization of disparities for each scan line



M: match,
L: left image occlusion,
R: right image occlusion

Dynamic Programming

Search for the minimum cumulative cost path in each scan line

Matching cost c_{ij}. E.g. SSD: $c_{ij} = \left(I_L(i) - I_R(j)\right)^2$

2D energy function for potential correspondence pair

$$E(i,j) = \min\Big(\underbrace{E(i-1,j-1)+\boxed{c_{ij}}}_{\text{No disparity change}},\ \underbrace{E(i-1,j)+\boxed{q_0}}_{\text{disparity changes by +1}},\ \underbrace{E(i,j-1)+\boxed{q_0}}_{\text{disparity changes by -1}}\Big)$$

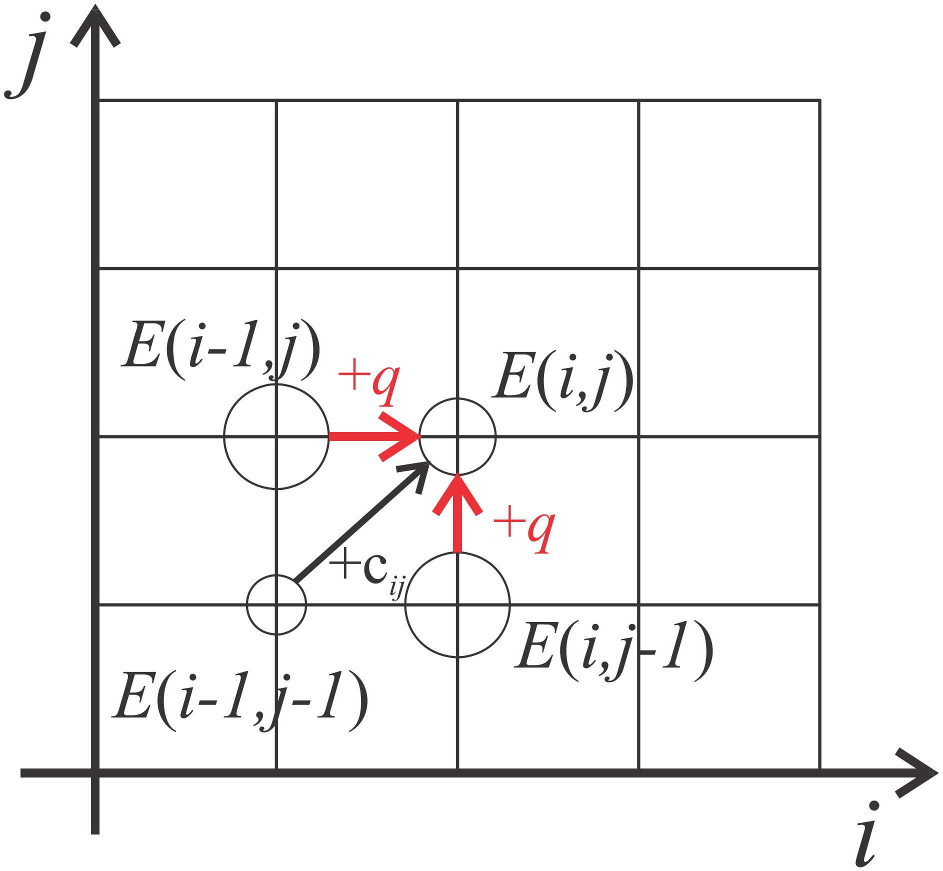
Start condition:

$$E(1,1) = e_{11}$$

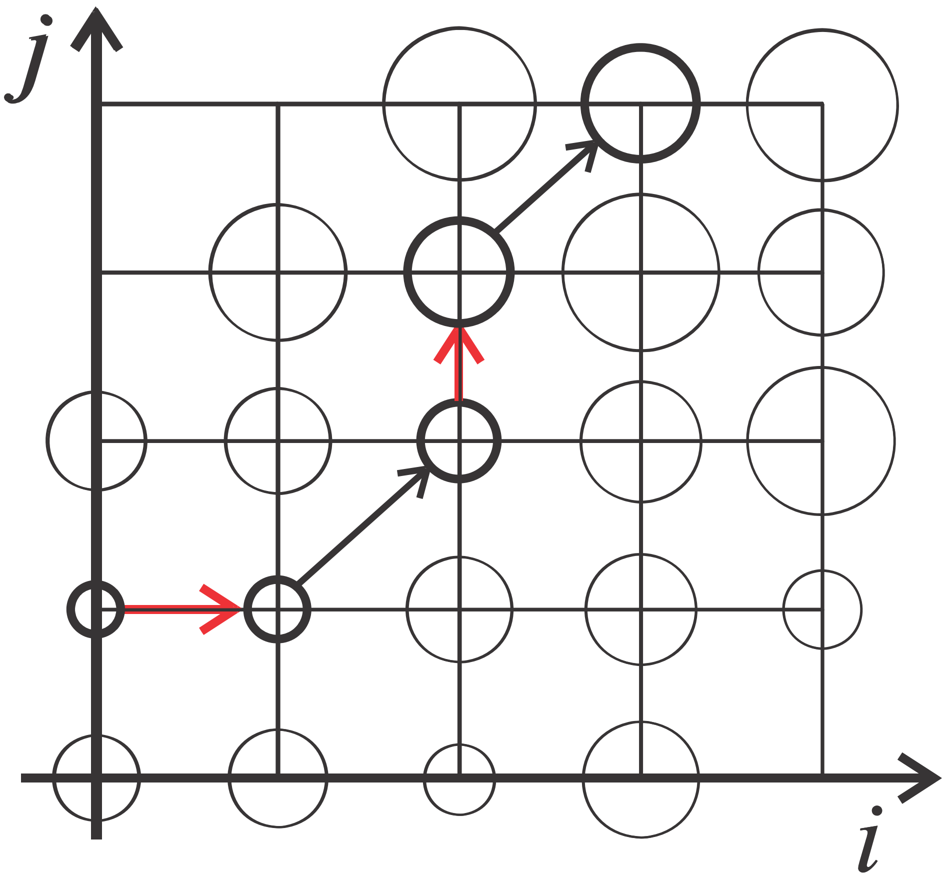
Initial matching scores

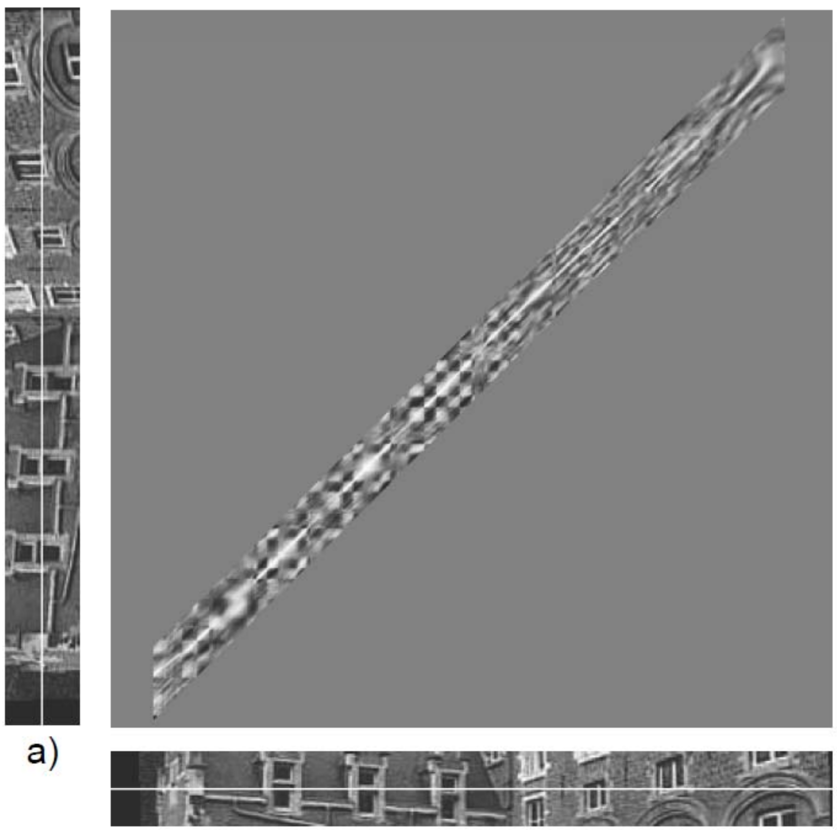
Penalty for change

Dynamic Programming



Dynamic Programming

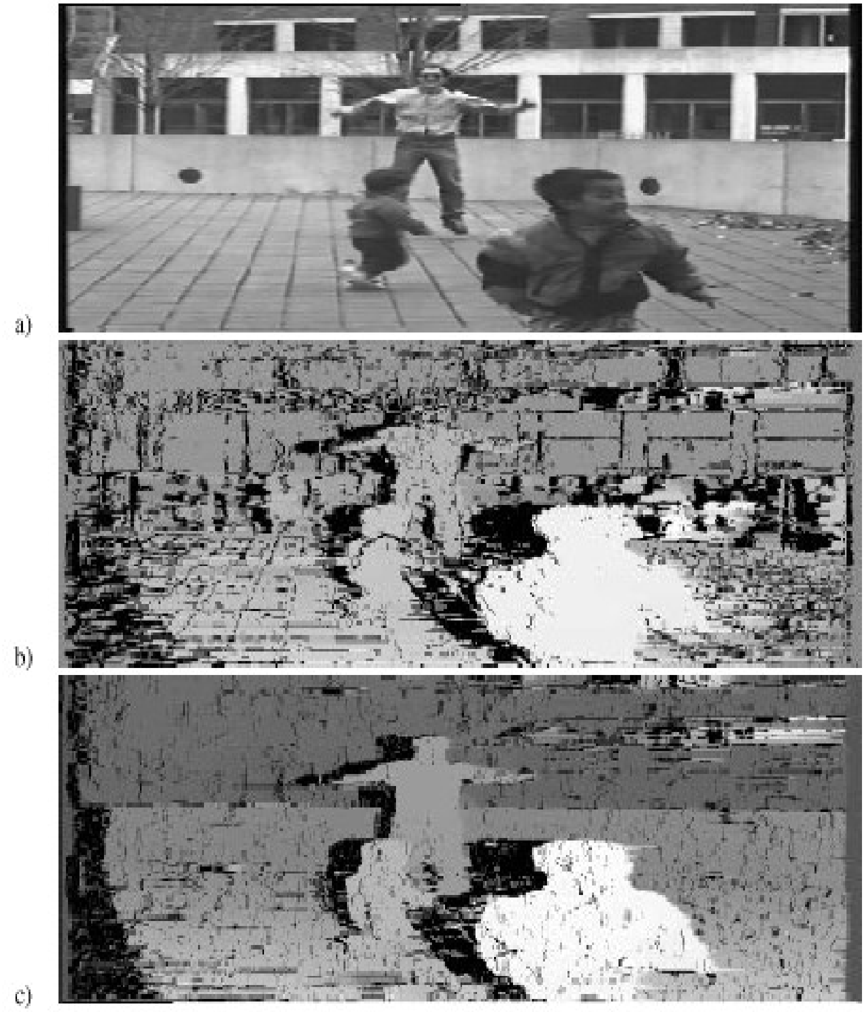




Dynamic Programming

Sample results

(Note horizontal artifacts)



A. F. Bobick, S. S. Intille: Large occlusion stereo. IJCV1999

Optimization methods

Global optimization:

- Graph Cut
  - 2D simultaneous optimization for whole image
  - 3D global energy function

$$E(x,y,d) = E_{data}(x,y,d) + \lambda E_{smoothness}(x,y,d)$$

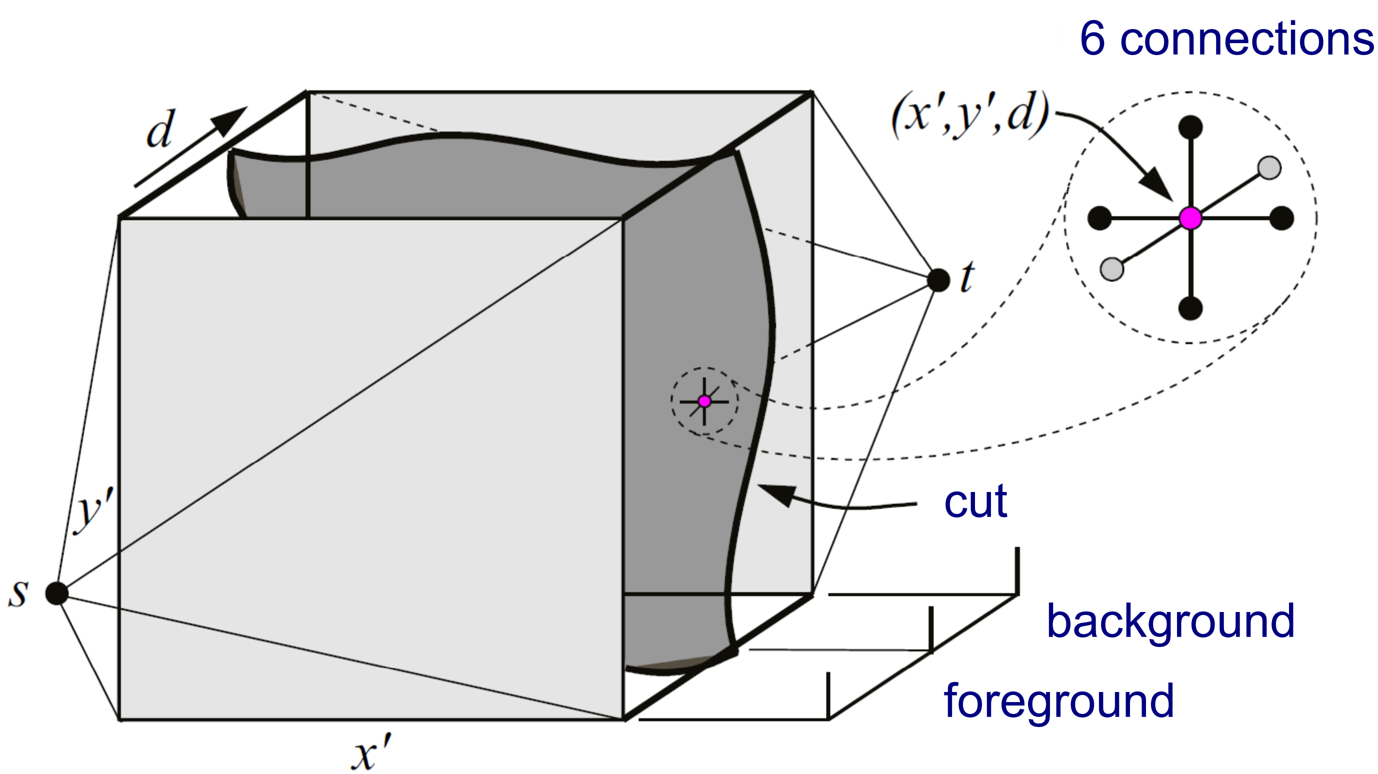
$$E_{data}(x,y,d) = \sum_{x,y,d} f\left(\boxed{c(x,y,d)}\right)$$ Initial matching scores

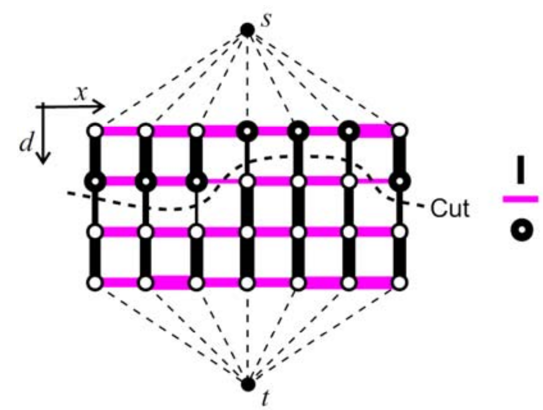
$$E_{smoothness}(x,y,d) = \sum_{x,y,d} \boxed{\sum_{u,v,w} f(x+u, y+v, d+w)}$$

Connection with neighbors

Graph Cut

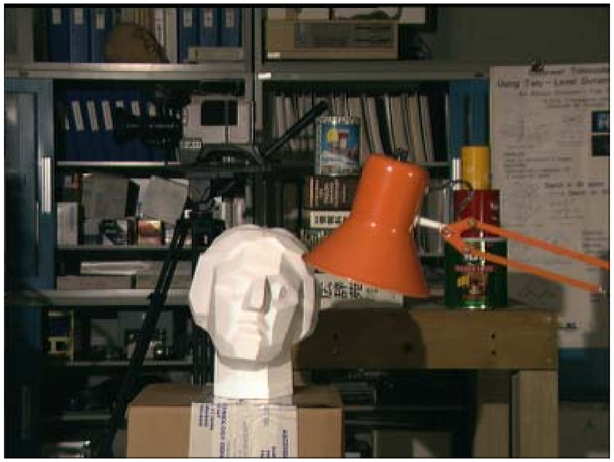
- Constructs a 3D graph with connections between neighboring cells
- Connection weights defined by global energy function *E*
- Cut finds the minimum energy surface



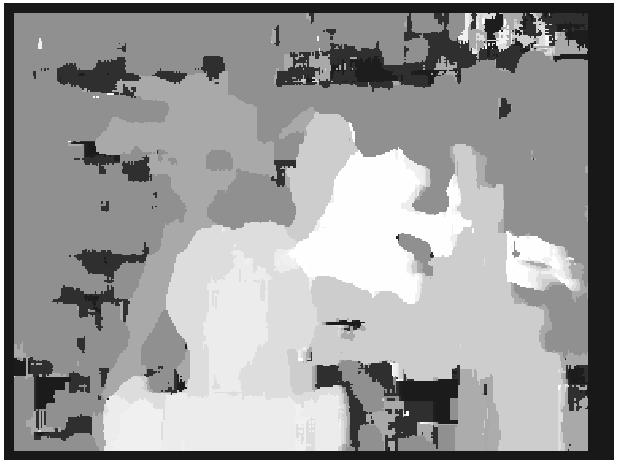


Graph Cut

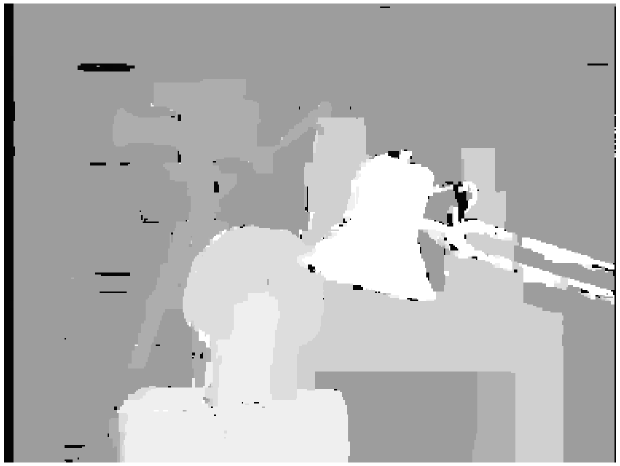
- Iterative optimization update of energy function
  (α-β swap, α expansion, modify smoothness penalty based on edges)
- Compute best pixel precise disparity



Input image



Sum Abs Diff



Graph cuts

Cooperative Stereo

- Global 3D energy function

$$E(x,y,d) = E_{data}(x,y,d) + \lambda E_{smoothness}(x,y,d)$$

$$E_{data}(x,y,d) = \sum_{x,y,d} f(e_{x,y,d} - c_{x,y,d})$$

$$E_{smoothness}(x,y,d) = \sum_{x,y,d} \sum_{u,v,w \in U} f(e_{x,y,d} - e_{x+u,y+v,d+w})$$

Local support area

- Iteratively update the energy for each pixel

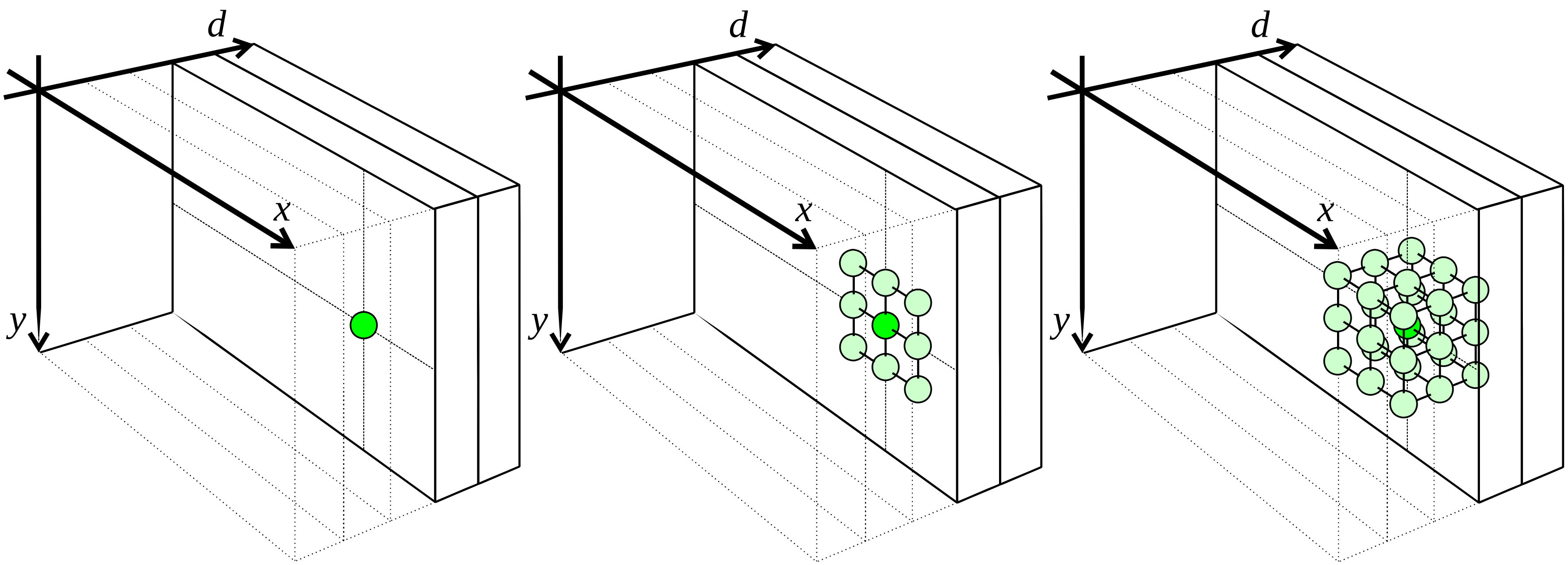
$$e_{i+1} = e_i + f(e_i, c_0, \sum_U e_{U,i})$$

- When converged, pick the disparity for the minimum score for each pixel

Cooperative Stereo

Local support area *U*

- inside a constant disparity level (2D-window function)
- inside disparity space (3D-window function)

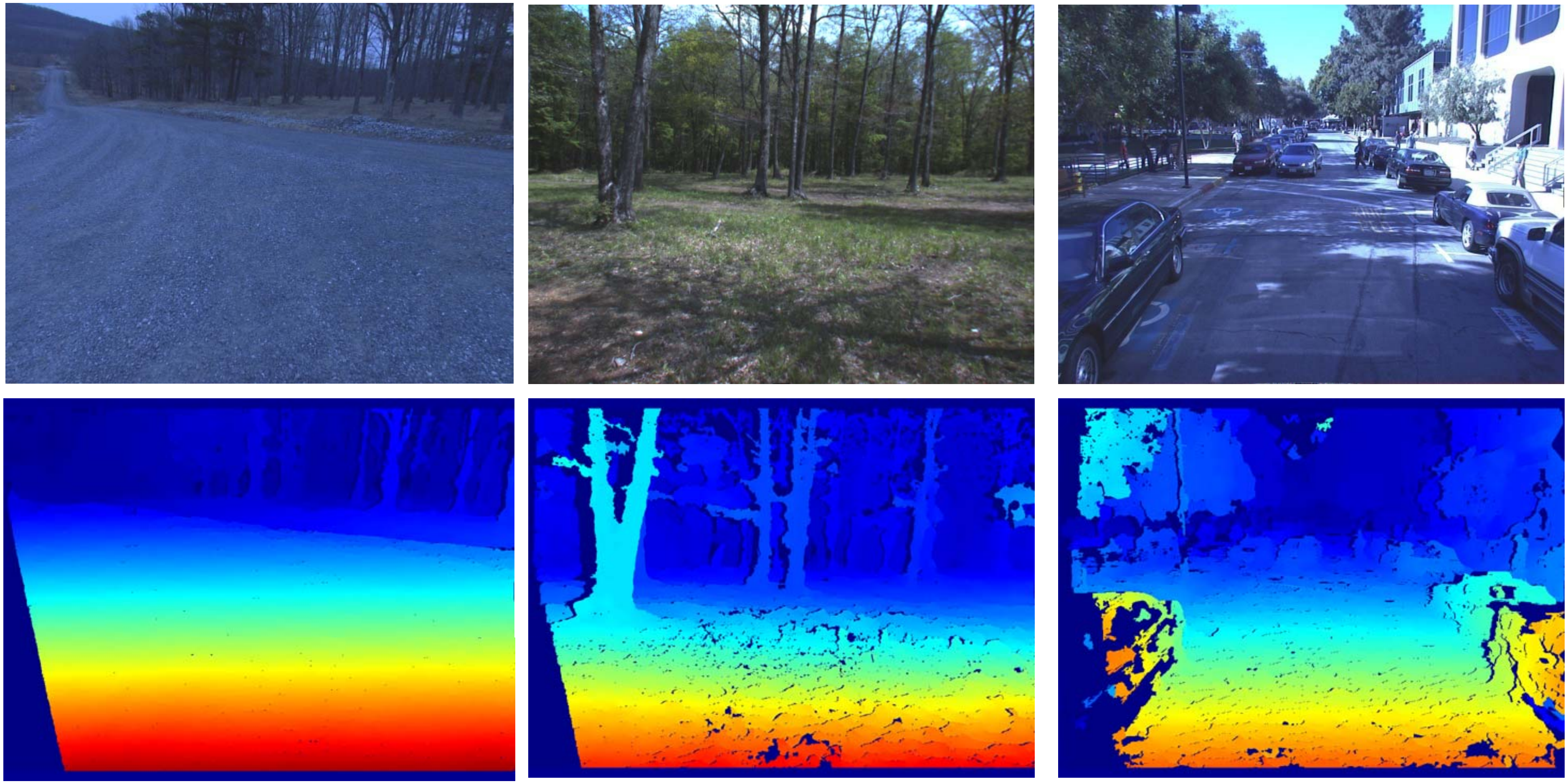


2D-support area

3D-support area

Interpretation of the coupled variable system as a neural network
[Marr & Poggio, 1976], [Reimann & Haken, 1994]

Cooperative Stereo



Stereo algorithms

Very active research field for more than 30 years

Lots of other methods:

- Bayesian inference
- Markov Random Fields
- Relaxation methods
- Belief Propagation
- Layered stereo

…

Stereo evaluation

vision.middlebury.edu

stereo • mview • MRF • flow

Stereo Evaluation • Datasets • Code • Submit

Daniel Scharstein • Richard Szeliski

Welcome to the Middlebury Stereo Vision Page, formerly located at www.middlebury.edu/stereo. This website accompanies our taxonomy and comparison of two-frame stereo correspondence algorithms [1]. It contains:

- An on-line evaluation of current algorithms
- Many stereo datasets with ground-truth disparities
- Our stereo correspondence software
- An on-line submission script that allows you to evaluate your stereo algorithm in our framework

How to cite the materials on this website:
We grant permission to use and publish all images and numerical results on this website. If you report performance results, we request that you cite our paper [1]. Instructions on how to cite our datasets are listed on the datasets page. If you want to cite this website, please use the URL "**vision.middlebury.edu/stereo/**".

References:

[1] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1/2/3):7-42, April-June 2002. Microsoft Research Technical Report MSR-TR-2001-81, November 2001.

Stereo – best algorithms

vision.middlebury.edu

stereo • mview • MRF • flow • color

Stereo Evaluation • Datasets • Code • Submit

Middlebury Stereo Evaluation - Version 2

New features and main differences to version 1.
Submit and evaluate your own results.

☐ Open a new window for each link

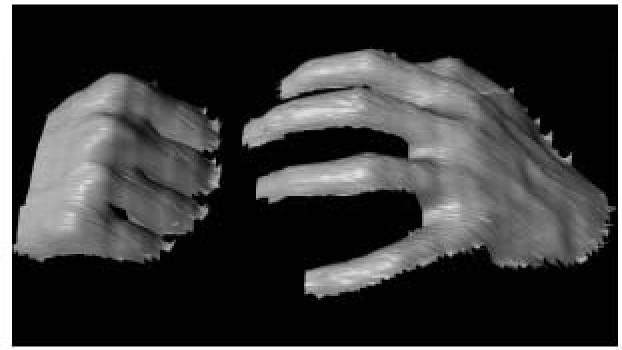
Error Threshold = 1 | Error Threshold... | Sort by nonocc | Sort by all | Sort by disc

Algorithm	Avg. Rank	Tsukuba ground truth nonocc	all	disc	Venus ground truth nonocc	all	disc	Teddy ground truth nonocc	all	disc	Cones ground truth nonocc	all	disc	Average percent of bad pixels (explanation)
ADCensus [94]	5.3	1.07 11	1.48 9	5.73 13	**0.09** 1	0.25 6	**1.15** 1	4.10 4	6.22 3	10.9 4	2.42 3	7.25 5	6.95 4	3.97
AdaptingBP [17]	6.6	1.11 13	1.37 5	5.79 14	0.10 2	0.21 3	1.44 3	4.22 6	7.06 6	11.8 7	2.48 4	7.92 9	7.32 7	4.23
CoopRegion [41]	6.7	0.87 2	**1.16** 1	**4.61** 1	0.11 3	0.21 2	1.54 5	5.16 14	8.31 9	13.0 11	2.79 12	7.18 4	8.01 16	4.41
DoubleBP [35]	9.1	0.88 4	1.29 2	4.76 4	0.13 6	0.45 16	1.87 10	3.53 3	8.30 8	9.63 2	2.90 17	8.78 24	7.79 13	4.19
RDP [102]	9.7	0.97 7	1.39 6	5.00 6	0.21 20	0.38 14	1.89 11	4.84 8	9.94 15	12.6 9	2.53 5	7.69 7	7.38 8	4.57
OutlierConf [42]	10.2	0.88 3	1.43 8	4.74 3	0.18 14	0.26 8	2.40 18	5.01 10	9.12 12	12.8 10	2.78 11	8.57 20	6.99 5	4.60
SubPixDoubleBP [30]	13.6	1.24 21	1.76 23	5.98 16	0.12 5	0.46 18	1.74 8	3.45 2	8.38 10	10.0 3	2.93 19	8.73 23	7.91 15	4.39
SurfaceStereo [79]	14.3	1.28 26	1.65 15	6.78 30	0.19 16	0.28 9	2.61 25	**3.12** 1	**5.10** 1	**8.65** 1	2.89 16	7.95 11	8.26 21	4.06
WarpMat [55]	15.9	1.16 14	1.35 4	6.04 17	0.18 15	0.24 5	2.44 20	5.02 11	9.30 13	13.0 13	3.49 27	8.47 19	9.01 33	4.98
ObjectStereo [98]	16.2	1.22 20	1.62 11	6.36 22	0.59 43	0.69 32	4.61 43	4.13 5	7.59 7	11.2 6	**2.20** 1	6.99 3	**6.36** 1	4.46
Undr+OvrSeg [48]	21.0	1.89 48	2.22 41	7.22 38	0.11 4	0.22 4	1.34 2	6.51 26	9.98 16	16.4 29	2.92 18	8.00 12	7.90 14	5.39
GC+SegmBorder [57]	21.5	1.47 37	1.82 25	7.86 45	0.19 17	0.31 10	2.44 20	4.25 7	5.55 2	10.9 5	4.99 62	**5.78** 1	8.66 27	4.52
GlobalGCP [104]	22.1	**0.87** 1	2.54 44	4.69 2	0.16 12	0.53 21	2.22 16	6.44 23	11.5 23	16.2 26	3.59 30	9.49 35	8.95 32	5.60
CostFilter [95]	22.4	1.51 39	1.85 29	7.61 42	0.20 19	0.39 15	2.42 19	6.16 21	11.8 26	16.0 23	2.71 9	8.24 15	7.66 12	5.55
AdaptOvrSegBP [33]	23.3	1.69 41	2.04 36	5.64 11	0.14 8	**0.20** 1	1.47 4	7.04 40	11.1 19	16.4 31	3.60 32	8.96 28	8.84 29	5.59

Other methods

Illumination based stereo matching







Li Zhang, Brian Curless, and Steven M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. *3DPVT* 2002

Lyubomir Zagorchev and A. Ardeshir Goshtasby*:* A paint-brush laser range scanner.

Bibliography

- Chapter 11 of Szeliski's book

- R. Hartley, A. Zisserman: Multiple view geometry in computer vision. Cambridge University Press, Cambridge, 2000. (good book about image based reconstruction)
- B. Julesz: Foundations of cyclopean perception. University of Chicago Press, Chicago, 1971. (Random dot stereograms)
- H. Hirschmüller, P. R. Innocent, J. Garabaldi: Real-time correlation-based stereo vision with reduced border error. In: International Journal of Computer Vision, vol. 47 (1-3), 2002, pp. 229-246. (Real-time SAD stereo algorithm)
- D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision, 47(1):7-42, 2002. (Algorithm evaluation)
- A.F. Bobick and S.S. Intille. Large occlusion stereo. Int Journal of Computer Vision, 33(3), 1999. pp. 181-200. (Dynamic programming)
- Roy & Cox: A Maximum-Flow Formulation of the N-camera Stereo Correspondence Problem. In: Proc. ICCV, 1998, pp. 492-499. (Graph Cut)
- V. Kolmogorov, R. Zabih: Computing visual correspondence with occlusions using graph cuts. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV'01), Vancouver, Kanada, 2001, pp. 508-515. (Graph Cut)
- D. Marr, T. Poggio: Cooperative computation of stereo disparity. In: Science, vol. 194, 1976, pp. 283-287. (Biologically inspired cooperative stereo algorithm)
- D. Reimann, H. Haken: Stereovision by self-organisation. In: Biological Cybernetics, vol. 71, 1994, pp. 17-26. (Biologically inspired cooperative stereo algorithm)
- R. Brockers: Cooperative Stereo Matching with Color-Based Adaptive Local Support. CAIP, 2009. (Cooperative stereo algorithm)
- Li Zhang, Brian Curless, and Steven M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. In *Proceedings of the 1st International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, Padova, Italy, June 19-21, 2002, pp. 24-36. (Structured light stereo)
- D. Scharstein, R. Szeliski: High-accuracy stereo depth maps using structured light. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'03), Madison, WI, USA, 2003, pp. 195-202. (Structured light stereo)
- L. Zagorchev, A. Goshtasby: A paintbrush laser range scanner. *Computer Vision and Image Understanding,* vol. 101, pp. 65-86, 2006. (Laser based stereo)