Lecture 09:
Stereo Algorithms

Robert Collins
CSE486, Penn State

Recall: Simple Stereo System

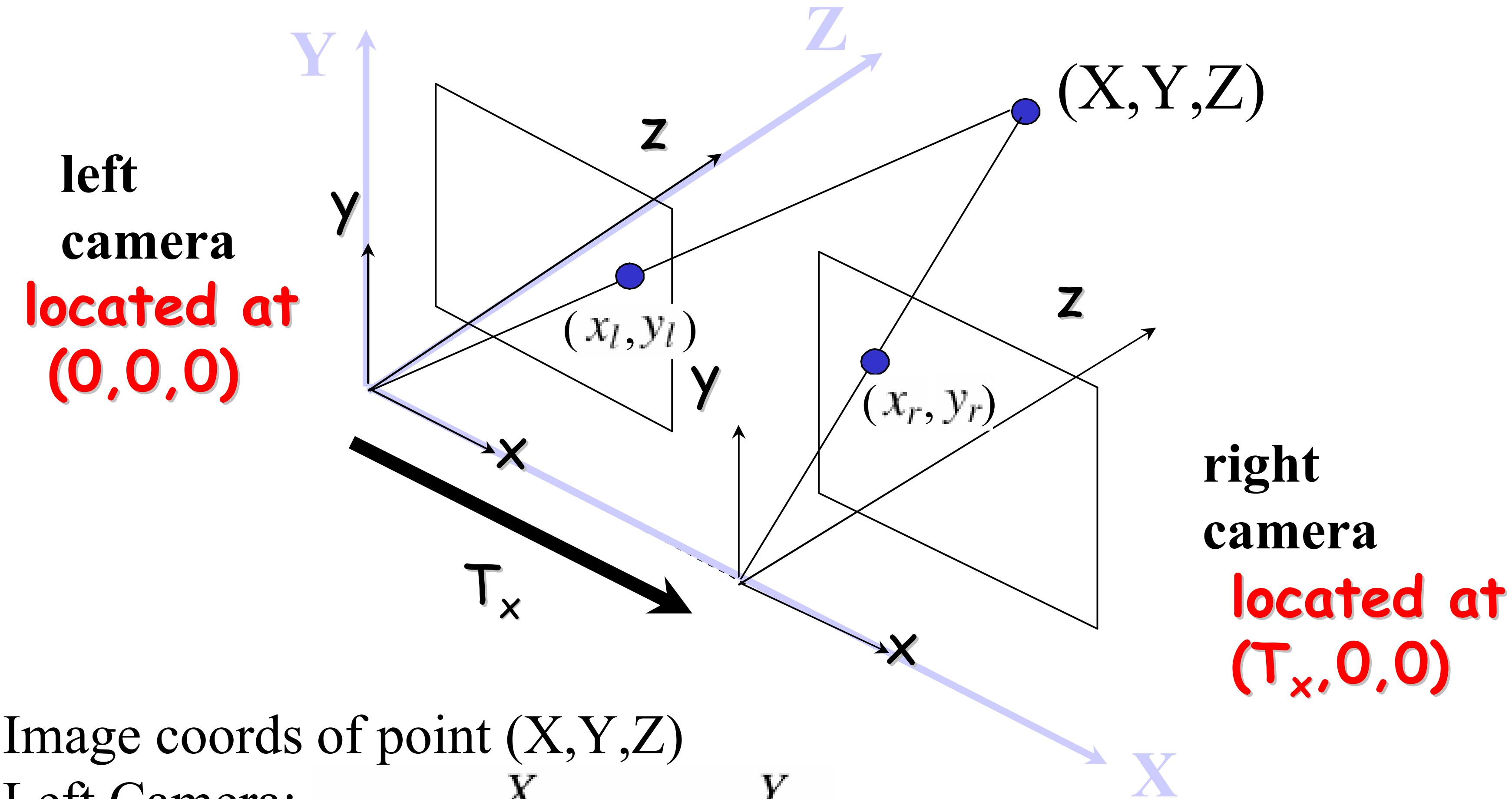


Image coords of point (X,Y,Z)

Left Camera: $$x_l = f\,\frac{X}{Z} \qquad y_l = f\,\frac{Y}{Z}$$

Right Camera: $$x_r = f\,\frac{X - T_x}{Z} \qquad y_r = f\,\frac{Y}{Z}$$

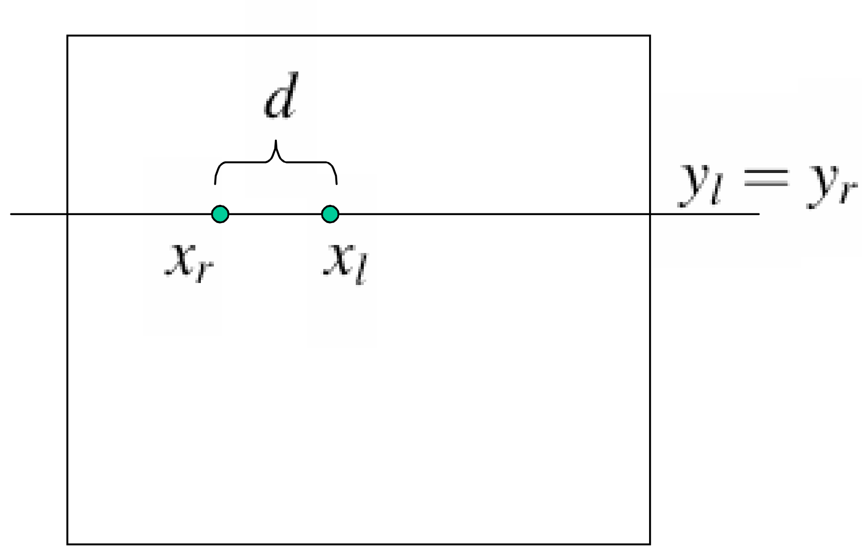
Recall: Stereo Disparity

Left camera

$$x_l = f\,\frac{X}{Z} \qquad y_l = f\,\frac{Y}{Z}$$

Right camera

$$x_r = f\,\frac{X - T_x}{Z} \qquad y_r = f\,\frac{Y}{Z}$$



Stereo Disparity

$$d = x_l - x_r = f\frac{X}{Z} - \left(f\frac{X}{Z} - f\frac{T_x}{Z}\right)$$

$$d = \frac{f\,T_x}{Z}$$

baseline

depth

$$Z = \frac{f\,T_x}{d}$$

disparity

Important equation!

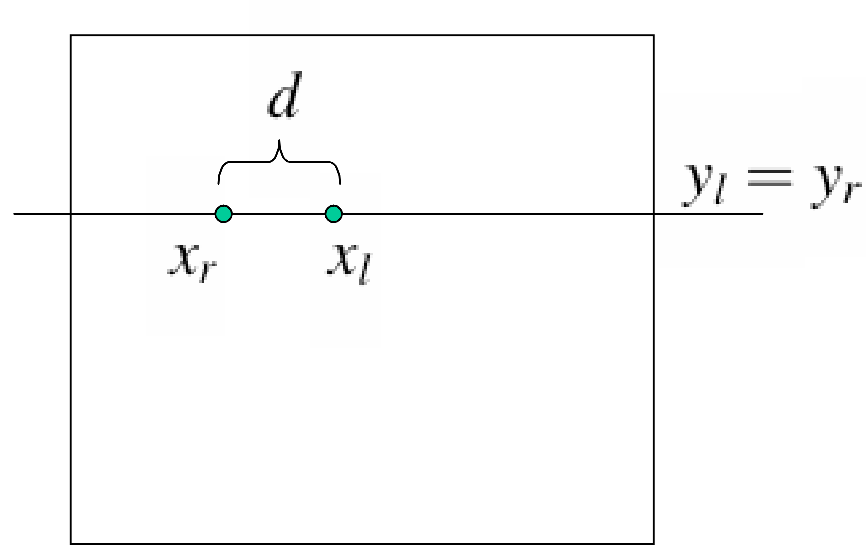
Recall: Stereo Disparity

Left camera

$$x_l = f\frac{X}{Z} \qquad y_l = f\frac{Y}{Z}$$

Right camera

$$x_r = f\frac{X - T_x}{Z} \qquad y_r = f\frac{Y}{Z}$$



Note: Depth and stereo disparity are inversely proportional

depth

$$Z = \frac{f\,T_x}{d}$$

disparity

Important equation!

Stereo Example

Left Image

Right Image

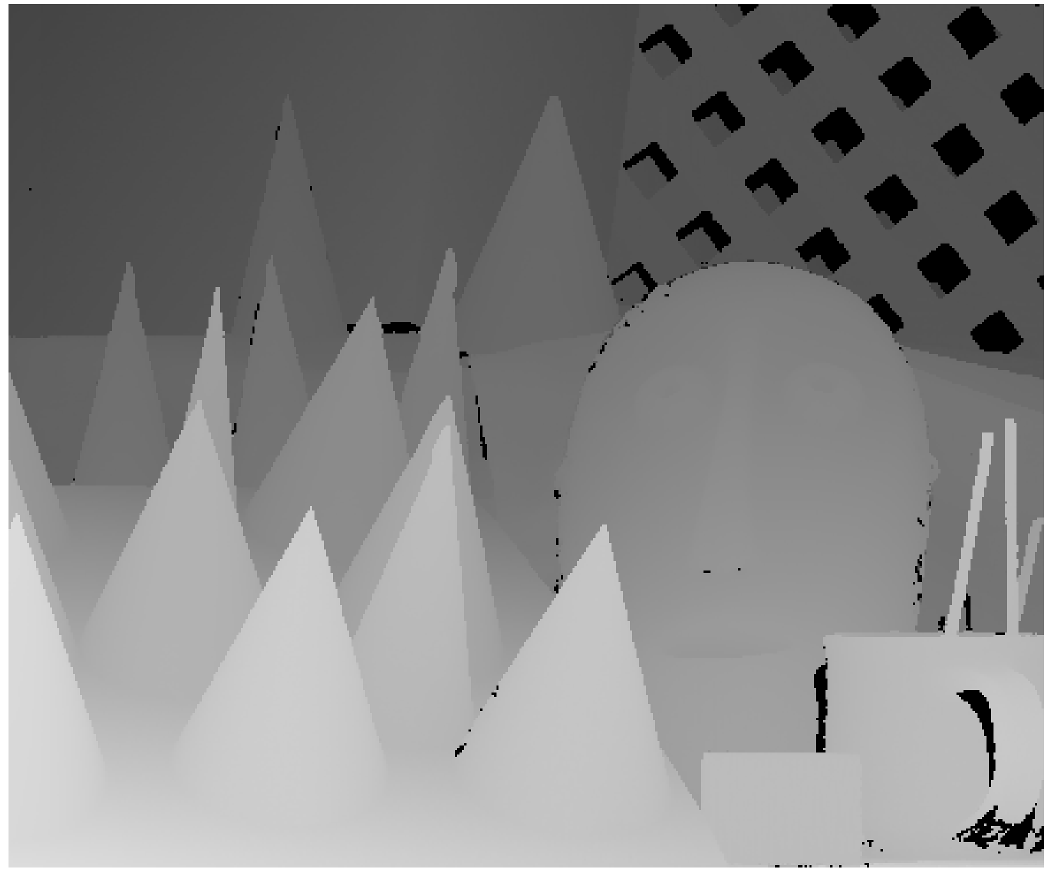
From Middlebury stereo evaluation page
http://www.middlebury.edu/stereo/

Stereo Example

Left image

Right image

Disparity values (0-64)



Note how disparity is larger (brighter) for closer surfaces.

Computing Disparity

- Correspondence Problem:
  - Determining which pixel in the right image corresponds to each pixel in the left image.
  - Disp = x_coord(left) - x_coord(right)

Recall our discussion of computing correspondences of image patches (Lecture 7).
SSD - sum of squared difference measure
NCC - normalized cross correlation measure

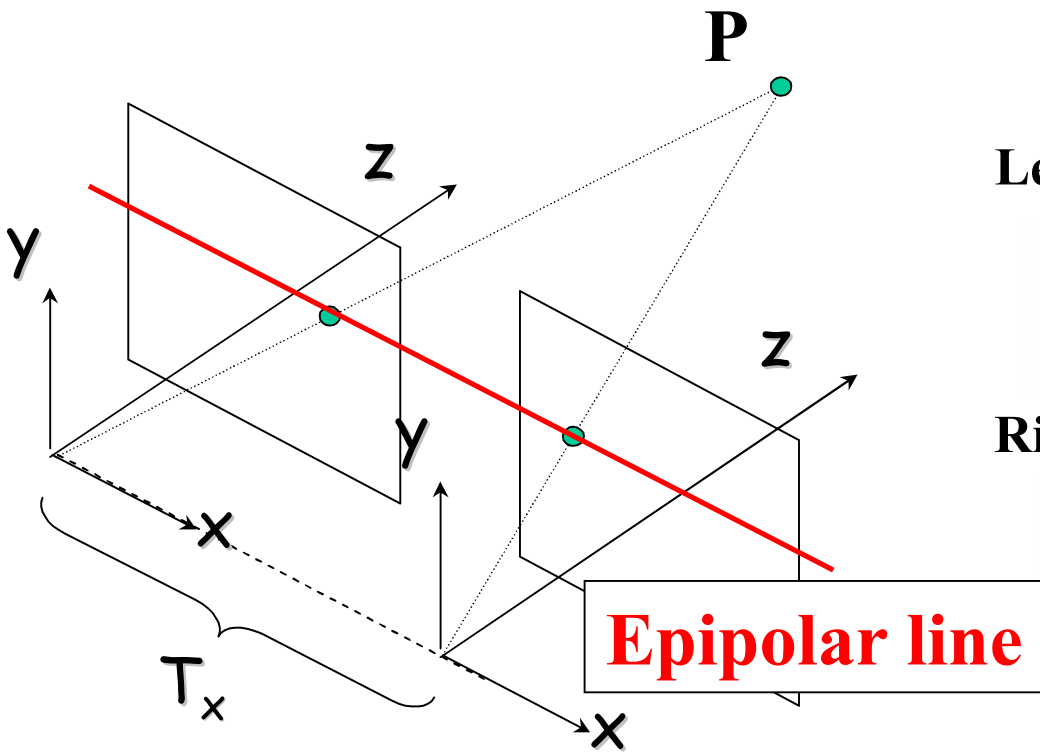
Epipolar Constraint

Important Concept:

For stereo matching, we don't have to search the whole 2D right image for a corresponding point.

The "epipolar constraint" reduces the search space to a one-dimensional line.

Recall : Simple Stereo System



Left camera

$$x_l = f\frac{X}{Z} \qquad \boxed{y_l = f\frac{Y}{Z}}$$

Right camera

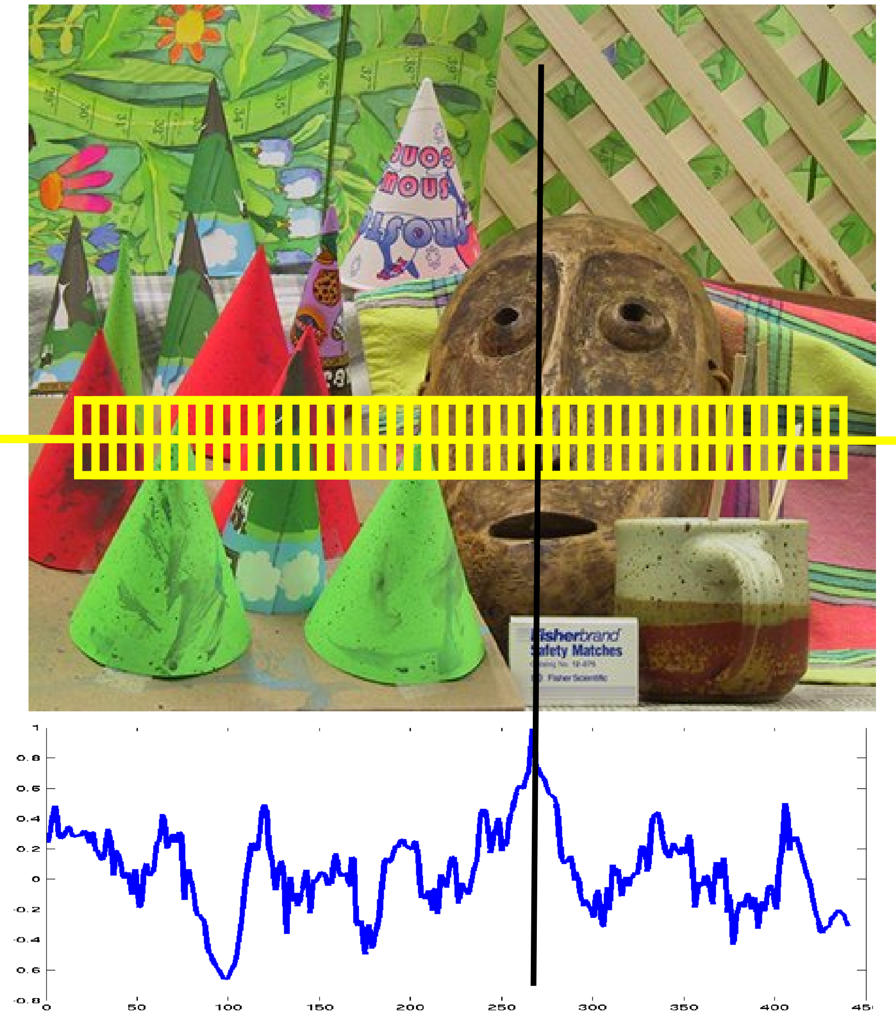
$$x_r = f\frac{X - T_x}{Z} \qquad \boxed{y_r = f\frac{Y}{Z}}$$

Same Y Coord!

Matching using Epipolar Lines

Left Image

Right Image



For a patch in left image

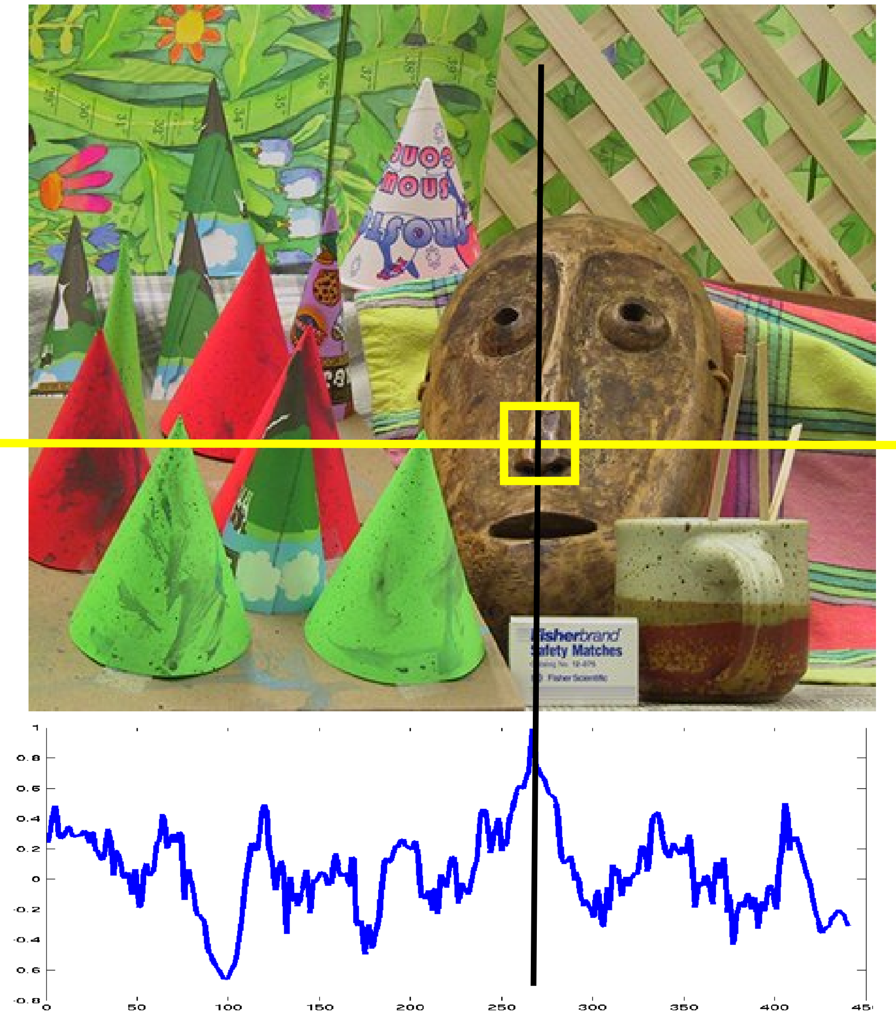
Compare with patches along same row in right image

Match Score Values

Matching using Epipolar Lines

Left Image

Right Image

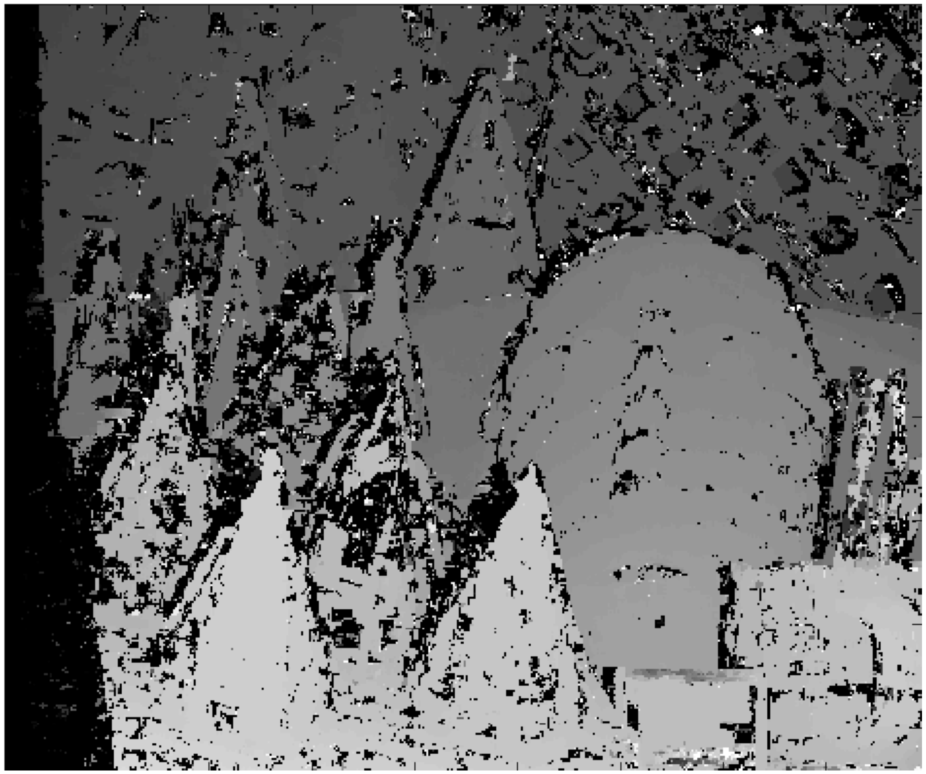


Match Score Values

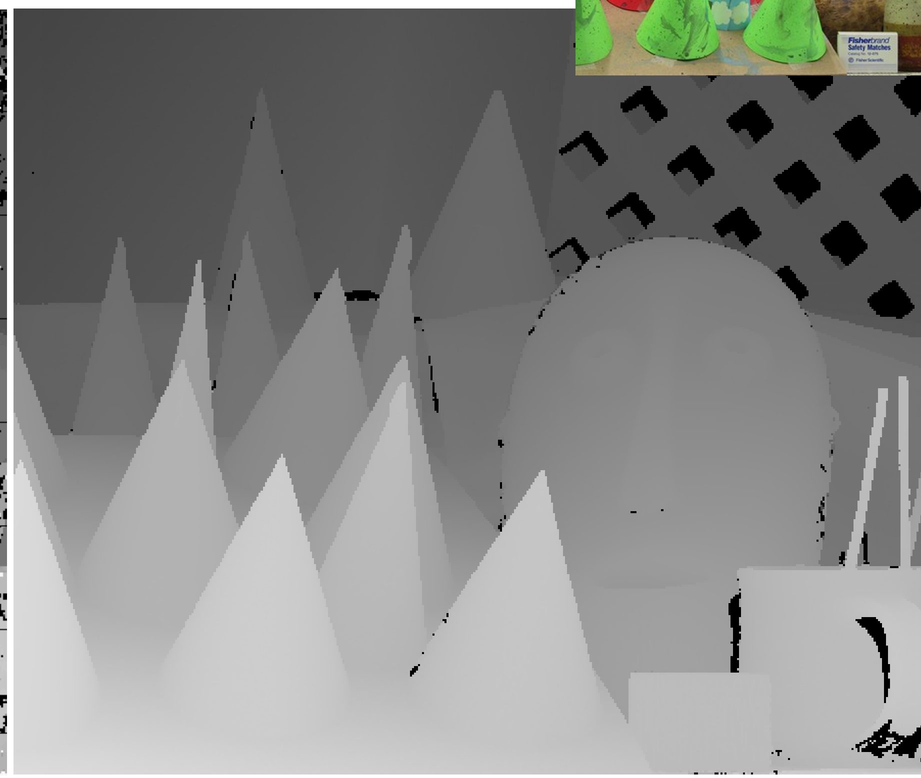
Select patch with highest
match score.

Repeat for all pixels in
left image.

Example: 5x5 windows NCC match score



Computed disparities

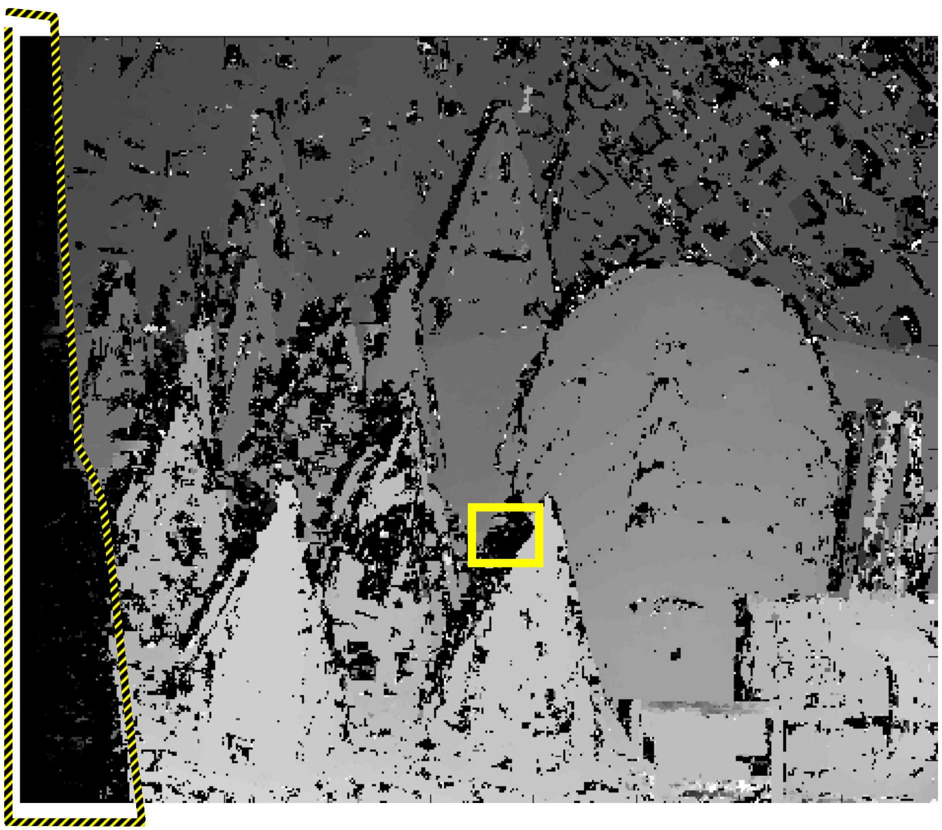


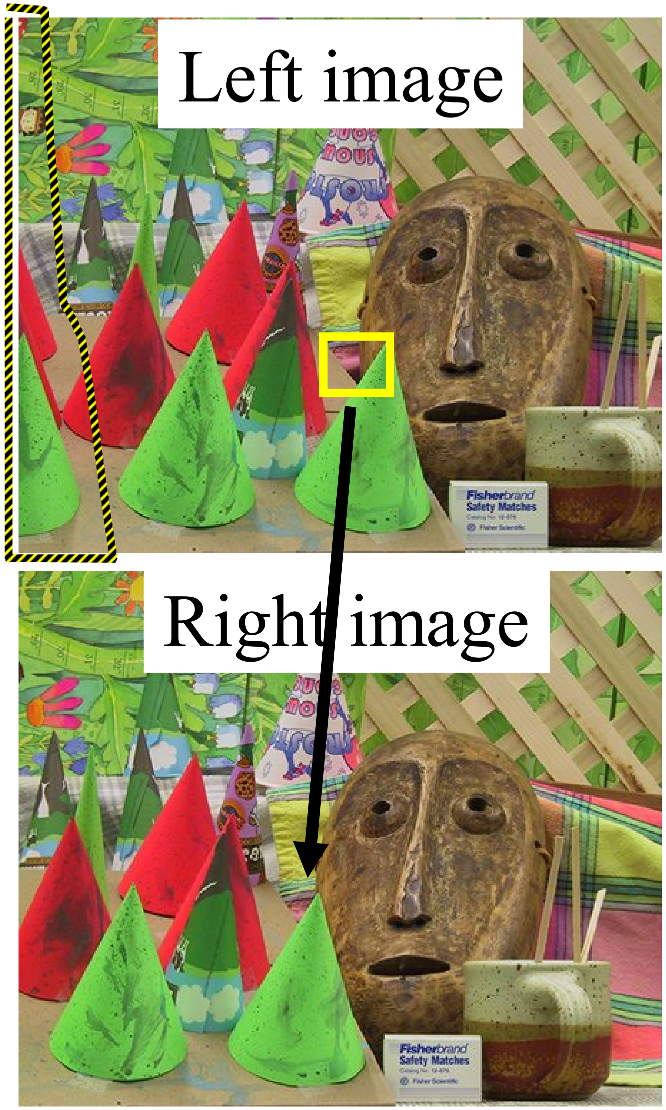
Ground truth

Black pixels: bad disparity values, or no matching patch in right image

Robert Collins
CSE486, Penn State

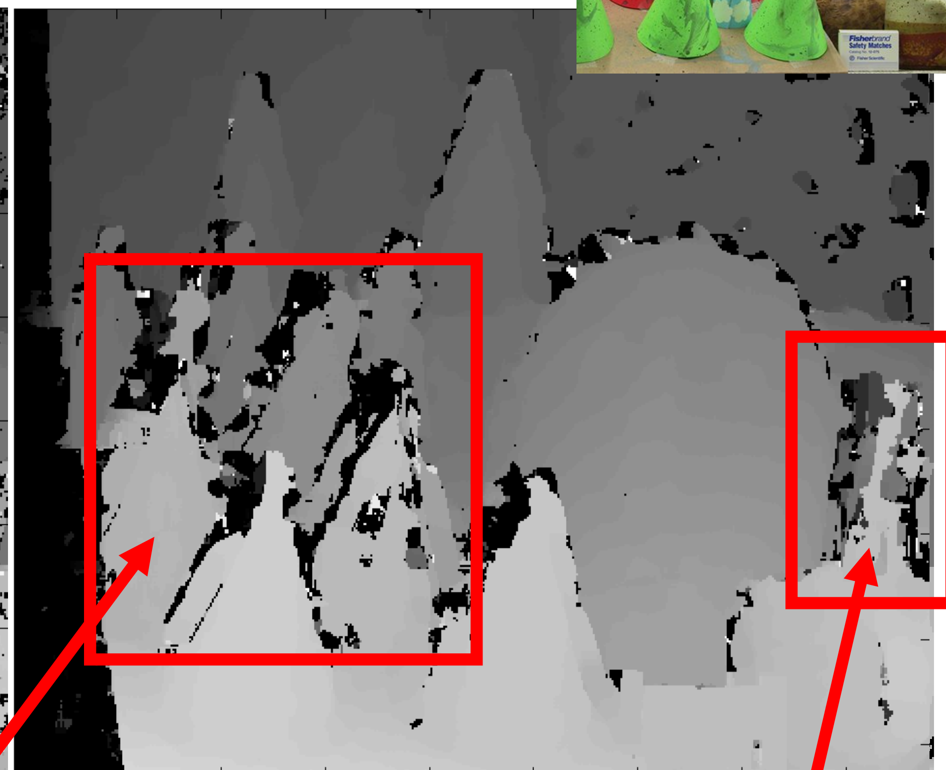
Occlusions: No matches





Effects of Patch Size

5x5 patches



11x11patches

Smoother in some areas

Loss of finer details

Adding Inter-Scanline Consistency

So far, each left image patch has been matched independently along the right epipolar line.

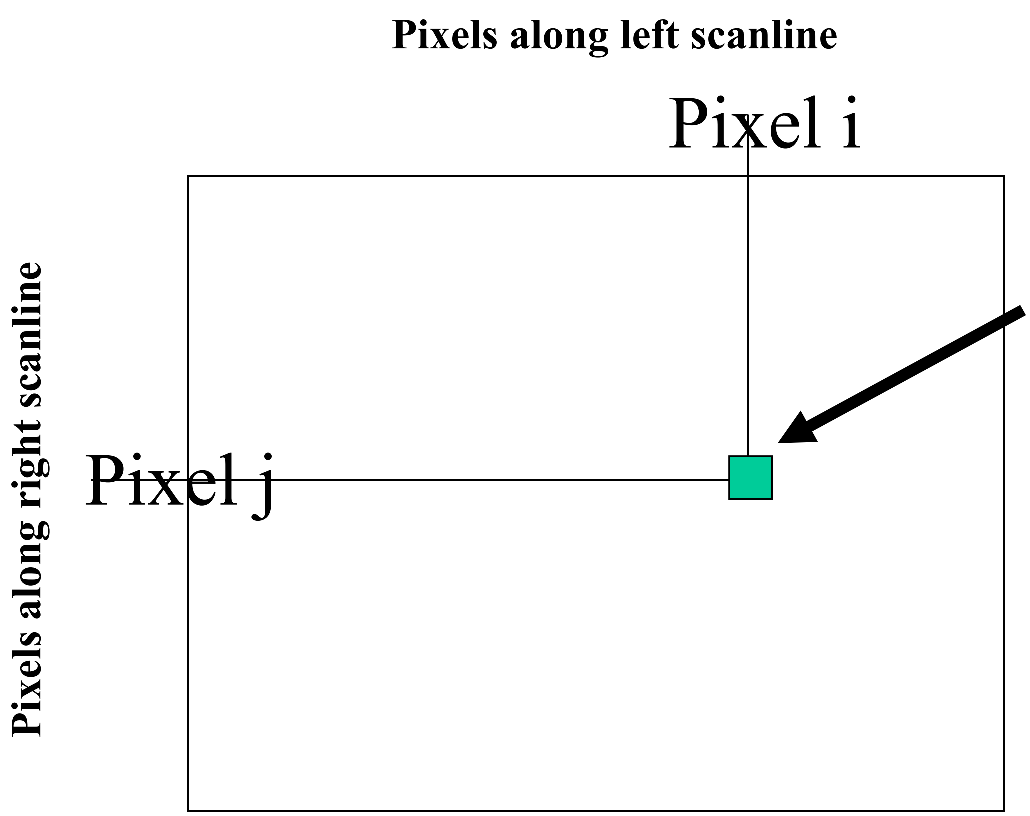
This can lead to errors.

We would like to enforce some consistency among matches in the same row (scanline).

Disparity Space Image

First we introduce the concept of DSI.

The DSI for one row represents pairwise match scores between patches along that row in the left and right image.



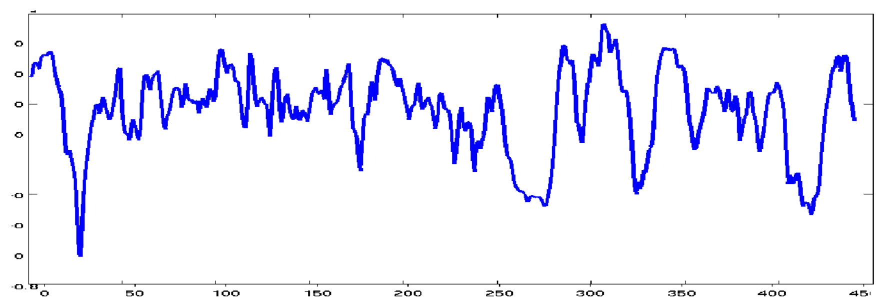
C(i,j) = Match score for patch centered at left pixel i with patch centered at right pixel j.

Robert Collins
CSE486, Penn State

Disparity Space Image (DSI)

Left Image

Right Image

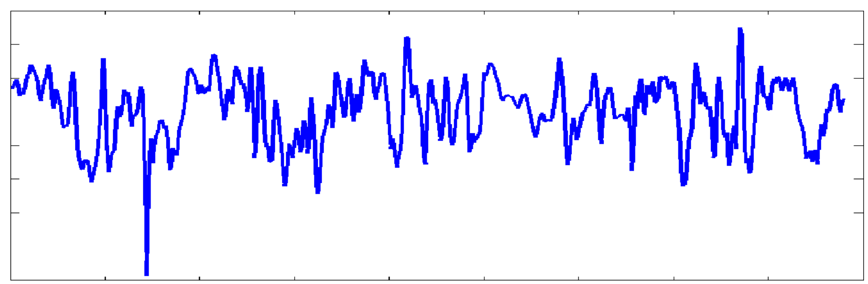


Dissimilarity Values
(1-NCC) or SSD

Disparity Space Image (DSI)

Left Image

Right Image



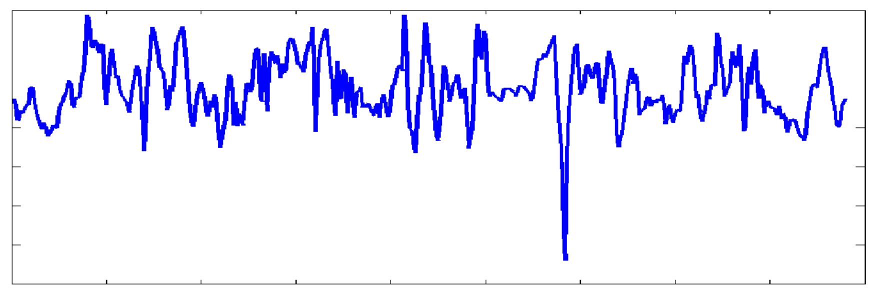
Dissimilarity Values
(1-NCC) or SSD

Robert Collins
CSE486, Penn State

Disparity Space Image (DSI)

Left Image

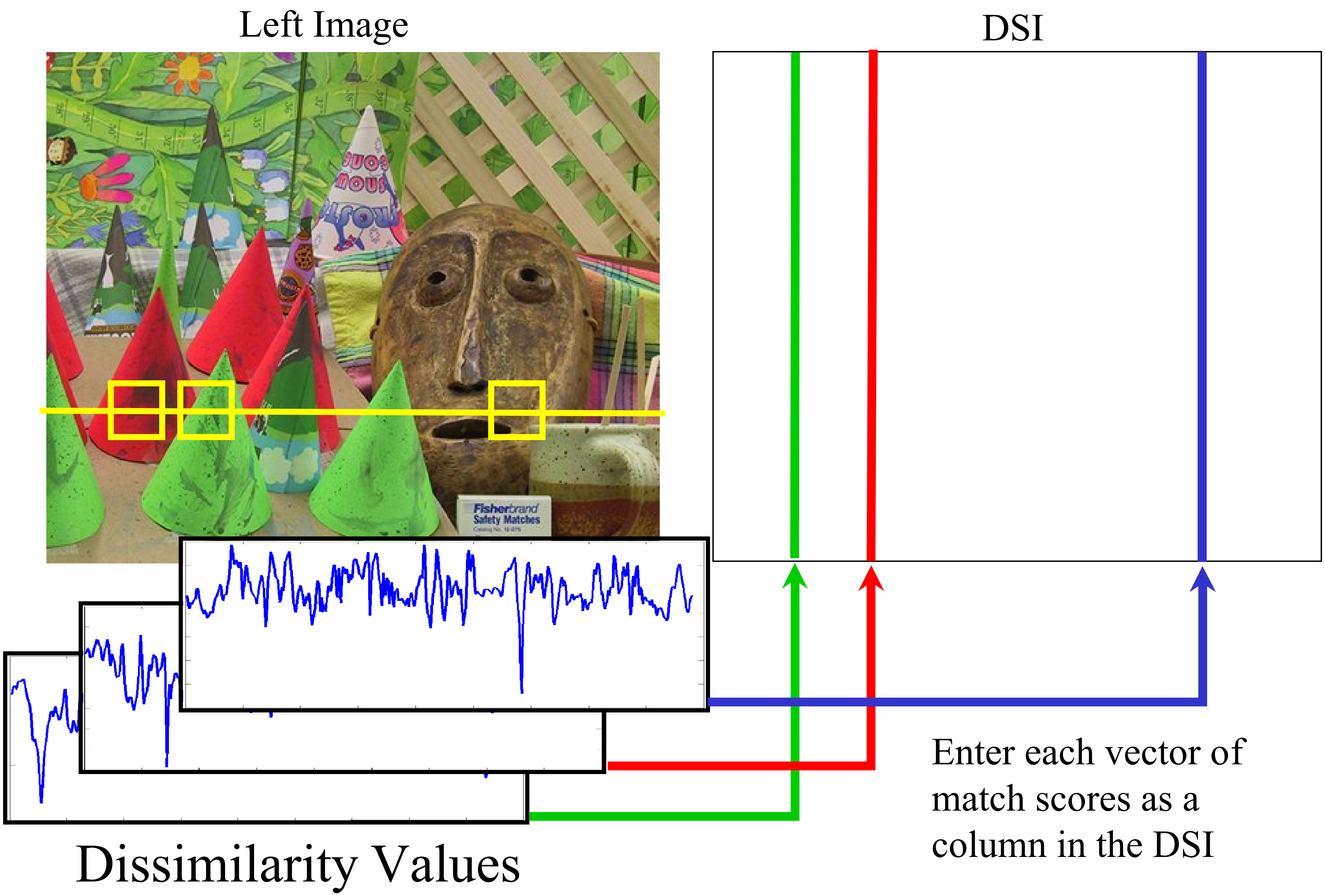
Right Image



Dissimilarity Values
(1-NCC) or SSD

Robert Collins
CSE486, Penn State

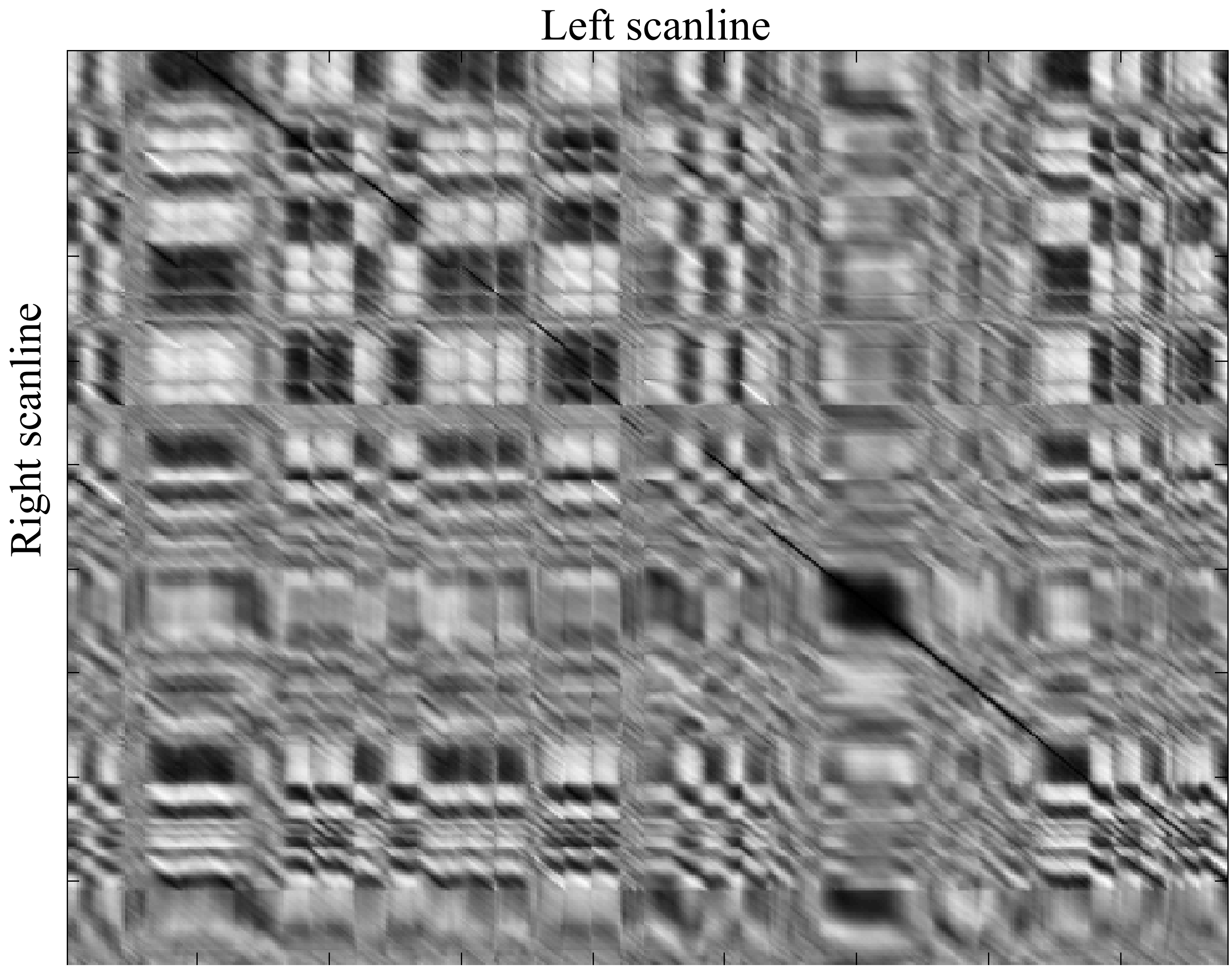
Disparity Space Image (DSI)



Enter each vector of match scores as a column in the DSI

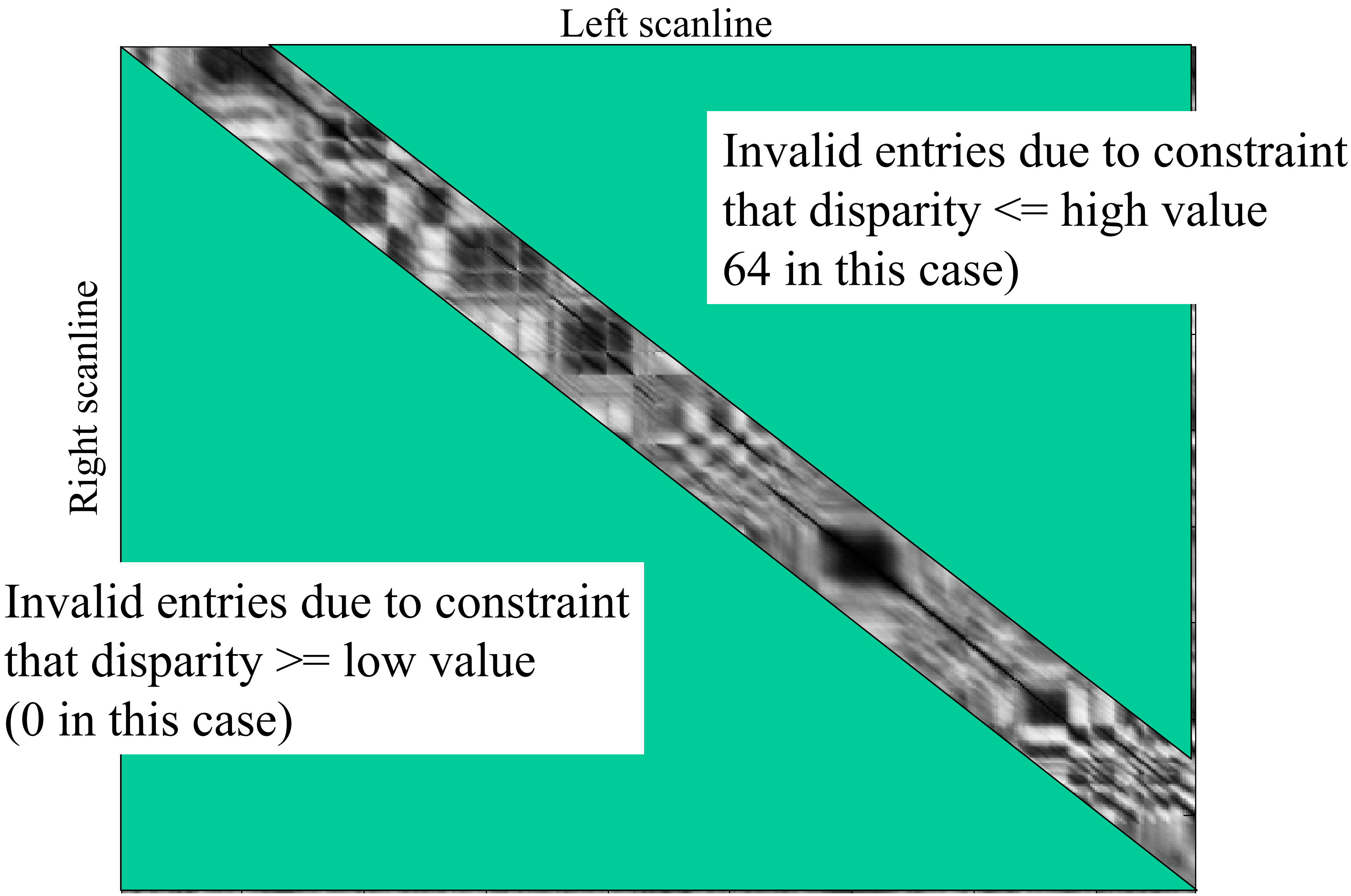
Robert Collins
CSE486, Penn State

Disparity Space Image



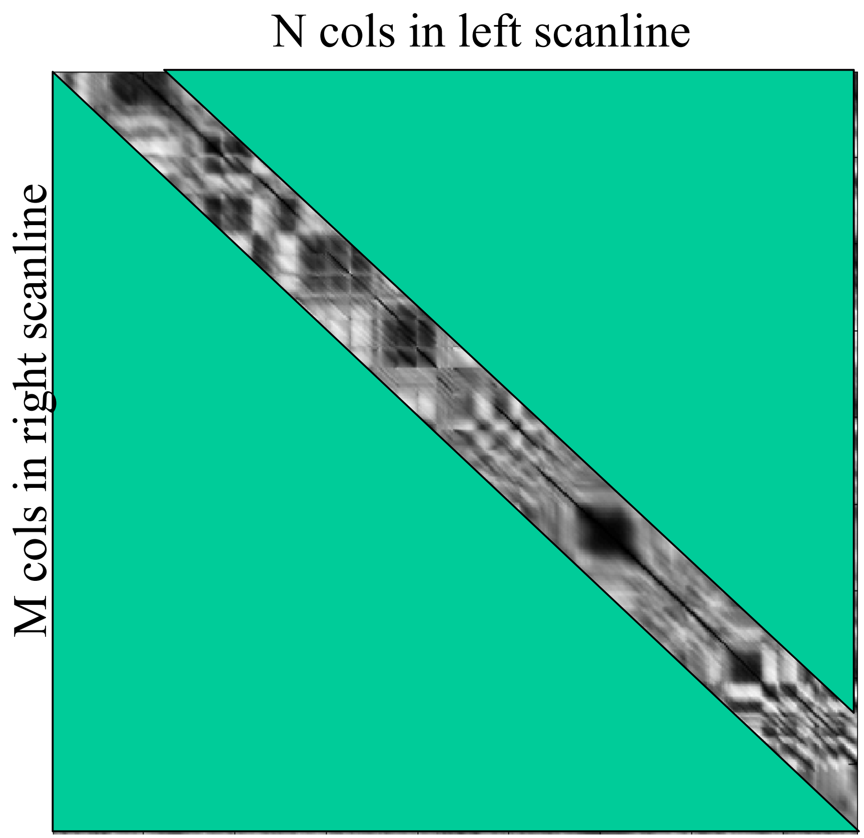
Robert Collins
CSE486, Penn State

Disparity Space Image



Robert Collins
CSE486, Penn State

Disparity Space Image

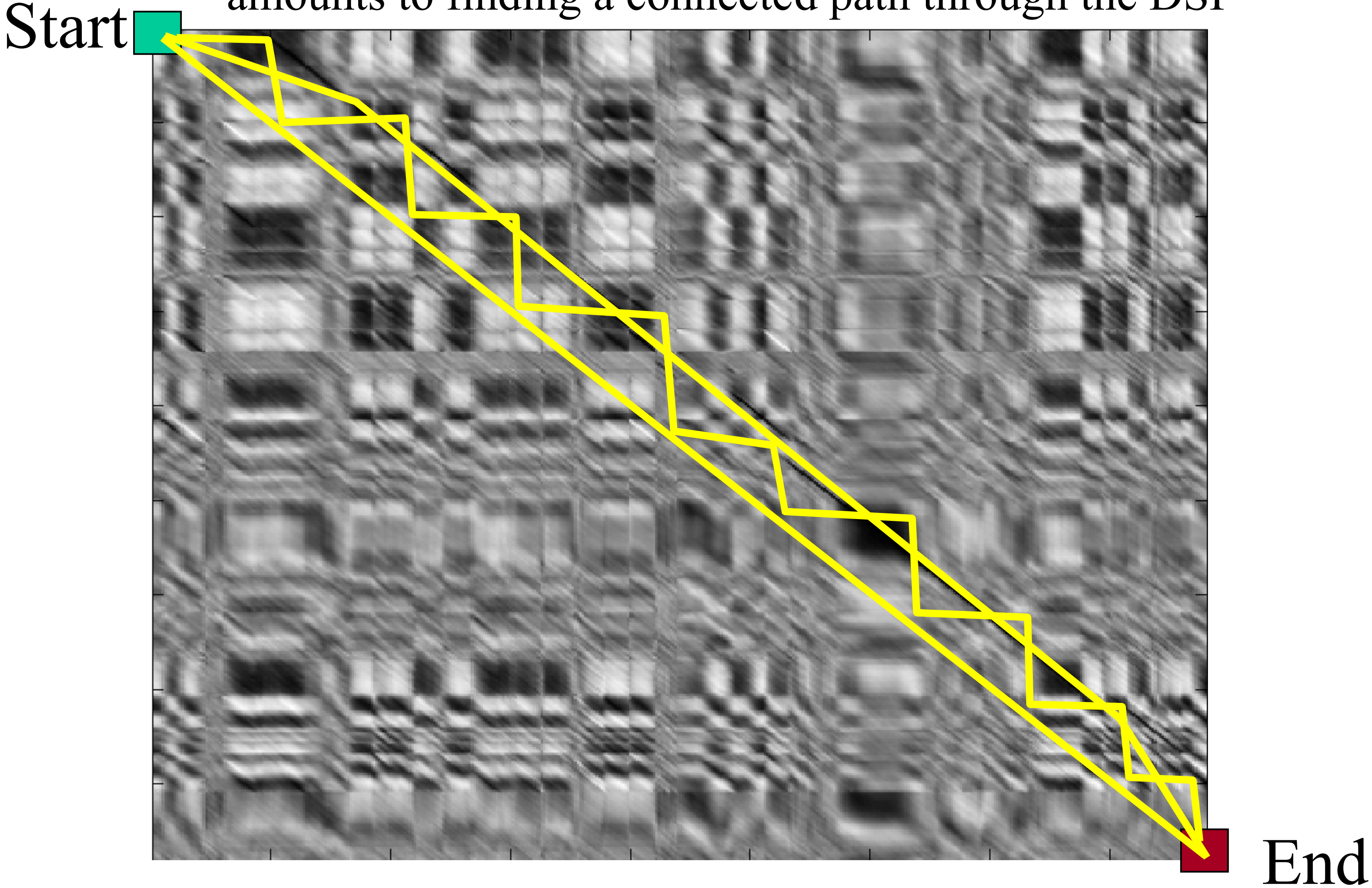


If we rearrange the diagonal band of valid values into a rectangular array (in this case of size 64 x N), that is what is traditionally known as the DSI

However, I'm going to keep the full image around, including invalid values (I think it is easier to understand the pixel coordinates involved)

DSI and Scanline Consistency

Assigning disparities to all pixels in left scanline now amounts to finding a connected path through the DSI

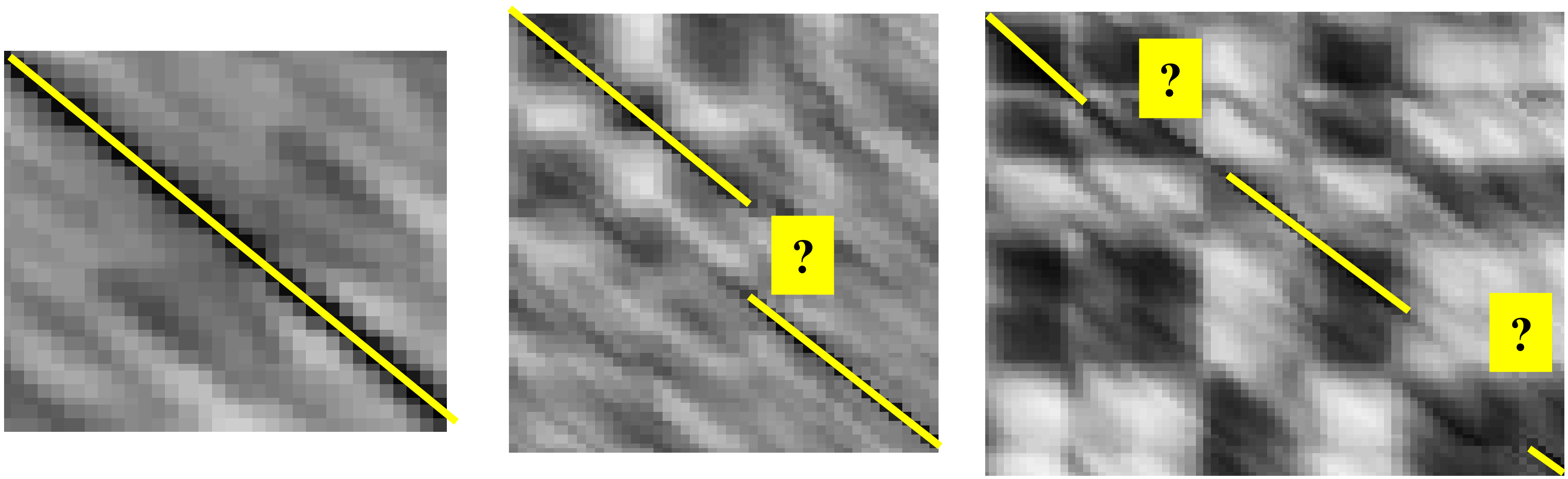


Robert Collins
CSE486, Penn State

Lowest Cost Path

We would like to choose the “best” path.

Want one with lowest “cost” (Lowest sum of dissimilarity scores along the path)

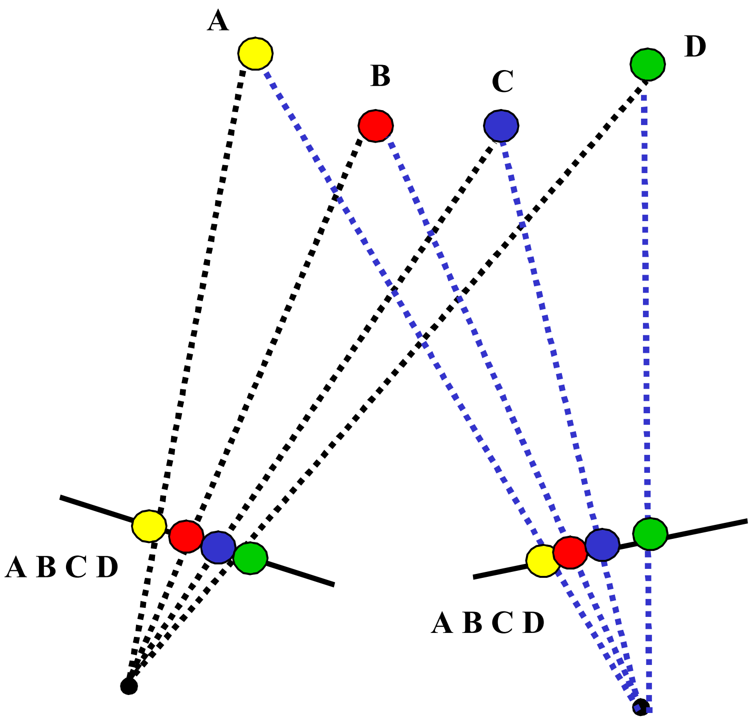


Constraints on Path

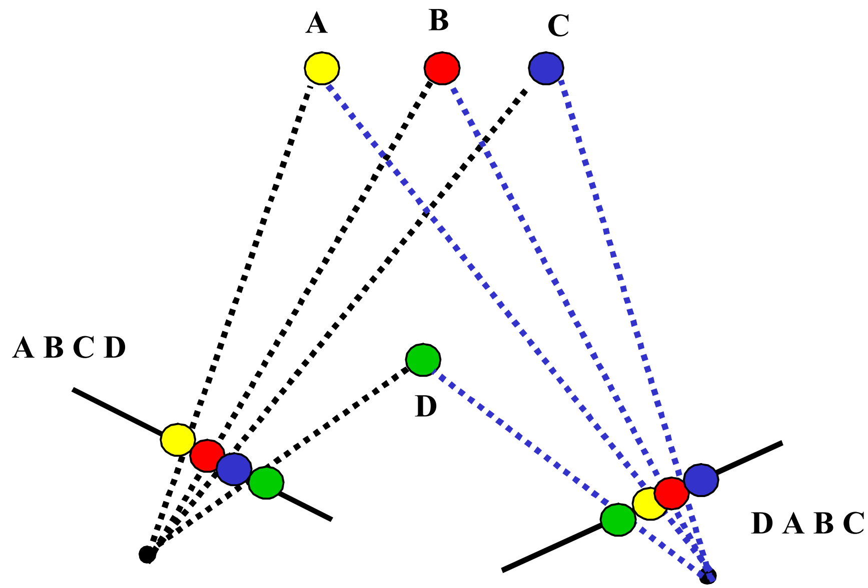
It is common to impose an ordering constraint on the path. Intuitively, the path is not allowed to “double back” on itself.

Robert Collins
CSE486, Penn State

Ordering Constraint



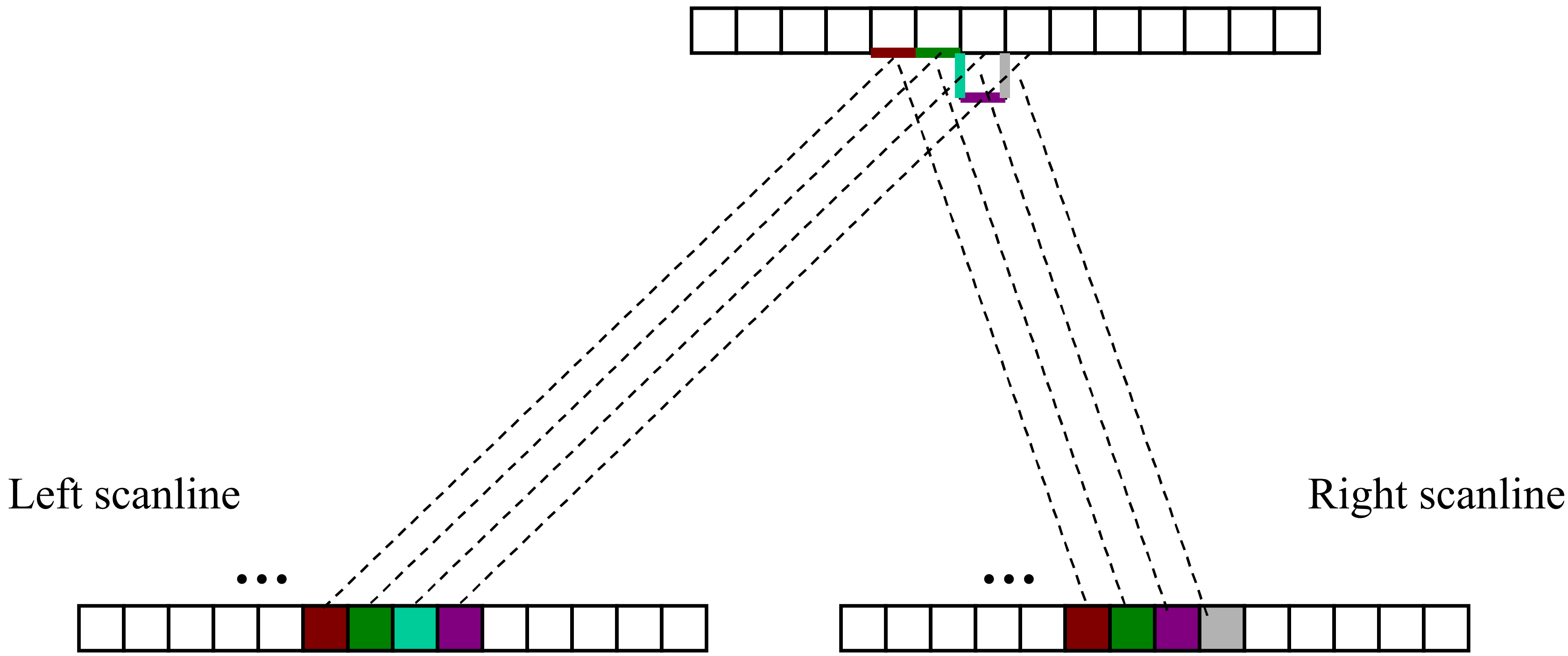
Ordering constraint…



…and its failure

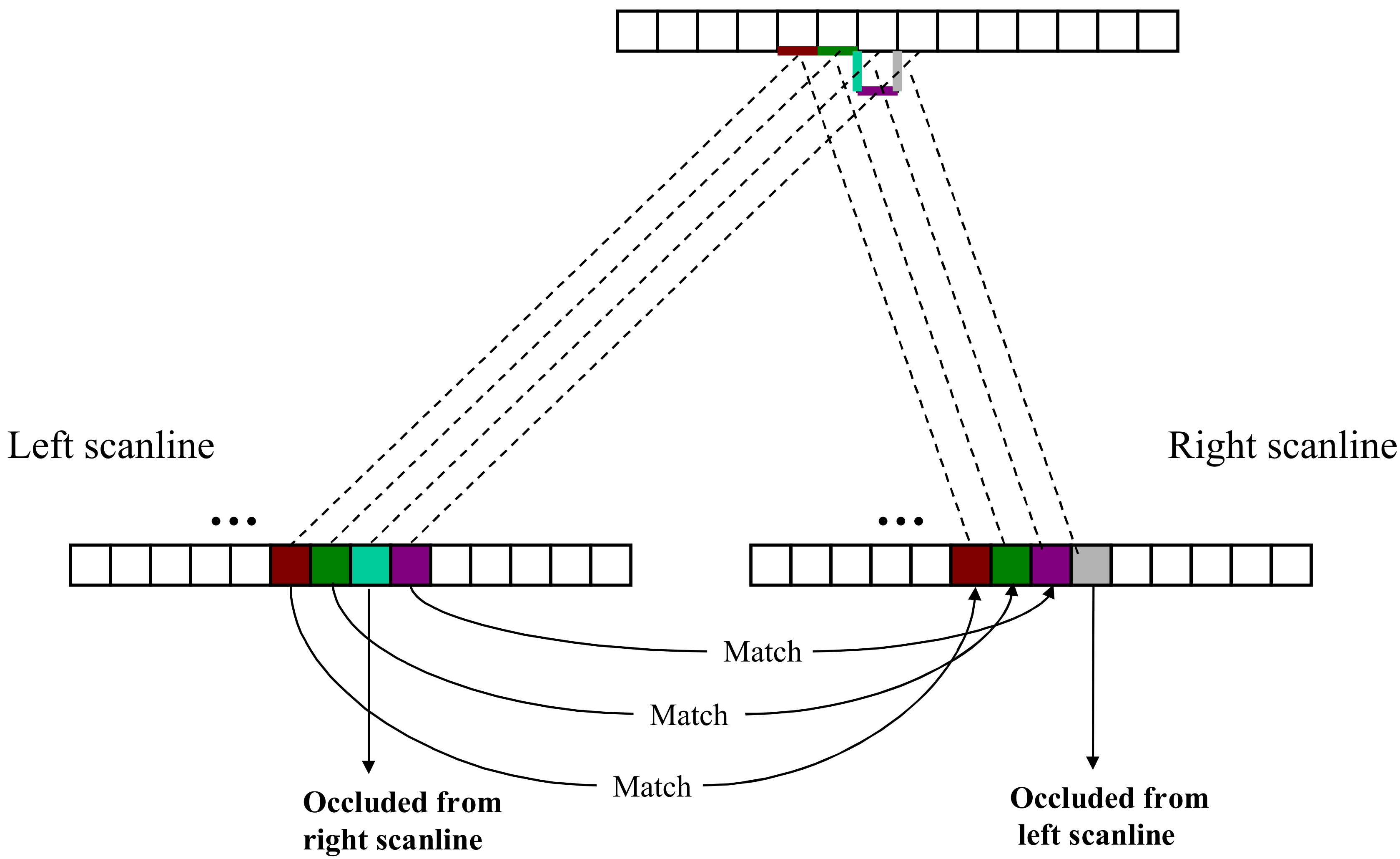
Robert Collins
CSE486, Penn State

Dealing with Occlusions



Robert Collins
CSE486, Penn State

Dealing with Occlusions



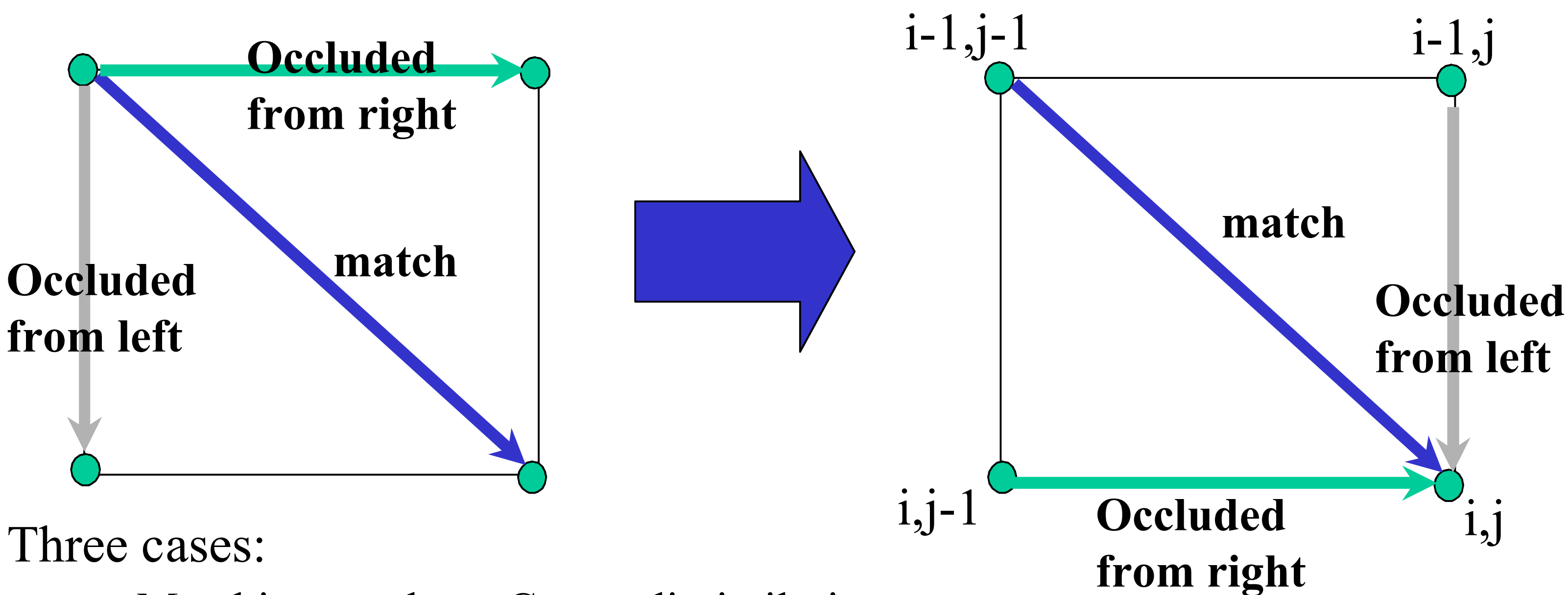
An Optimal Scanline Strategy

- We want to find best path, taking into account ordering constraint and the possibility of occlusions.

Algorithm we will discuss now is from
Cox, Hingorani, Rao, Maggs, "A Maximum Likelihood Stereo Algorithm," Computer Vision and Image Understanding, Vol 63(3), May 1996, pp.542-567.

Robert Collins
CSE486, Penn State

Cox et.al. Stereo Matching

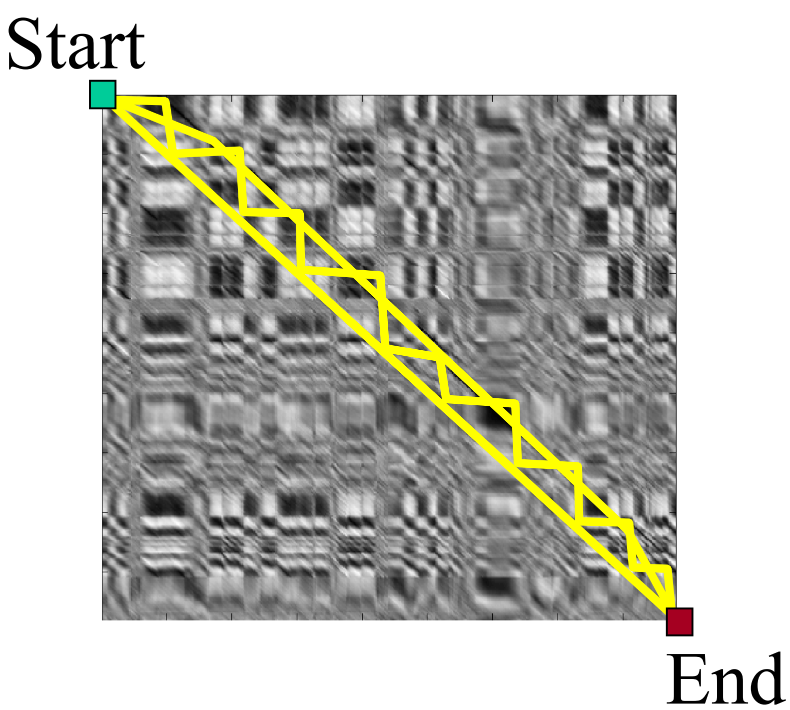


Three cases:

- Matching patches. Cost = dissimilarity score
- Occluded from right. Cost is some constant value.
- Occluded from left. Cost is some constant value.

$$\begin{aligned} C(i,j) = \min([\, & C(i-1,j-1) + \text{dissimilarity}(i,j) \\ & C(i-1,j) + \text{occlusionConstant}, \\ & C(i,j-1) + \text{occlusionConstant}]); \end{aligned}$$

Cox et.al. Stereo Matching



Recap: want to find lowest cost path from upper left to lower right of DSI image.

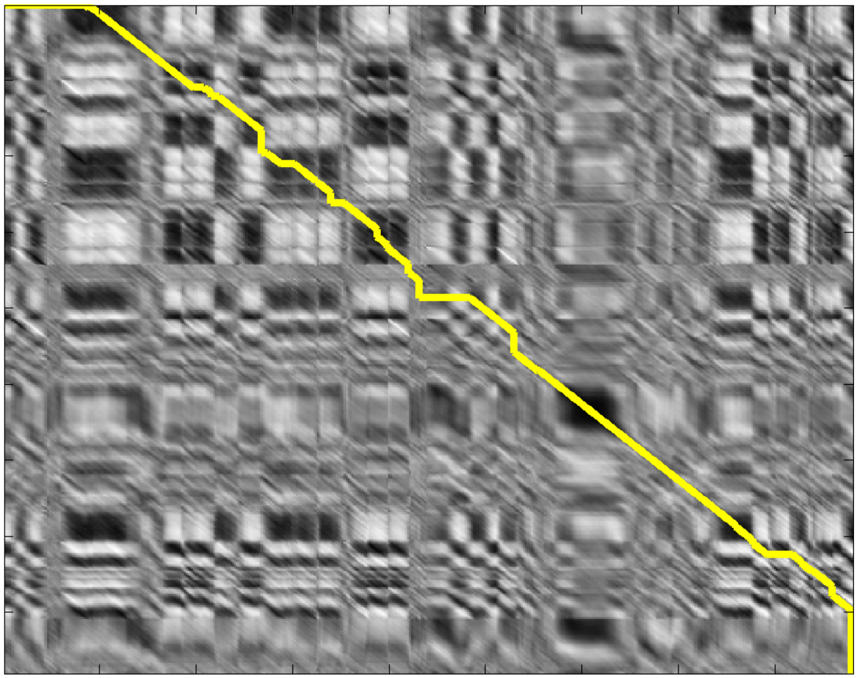
At each point on the path we have three choices: step left, step down, step diagonally.

Each choice has a well-defined cost associated with it.

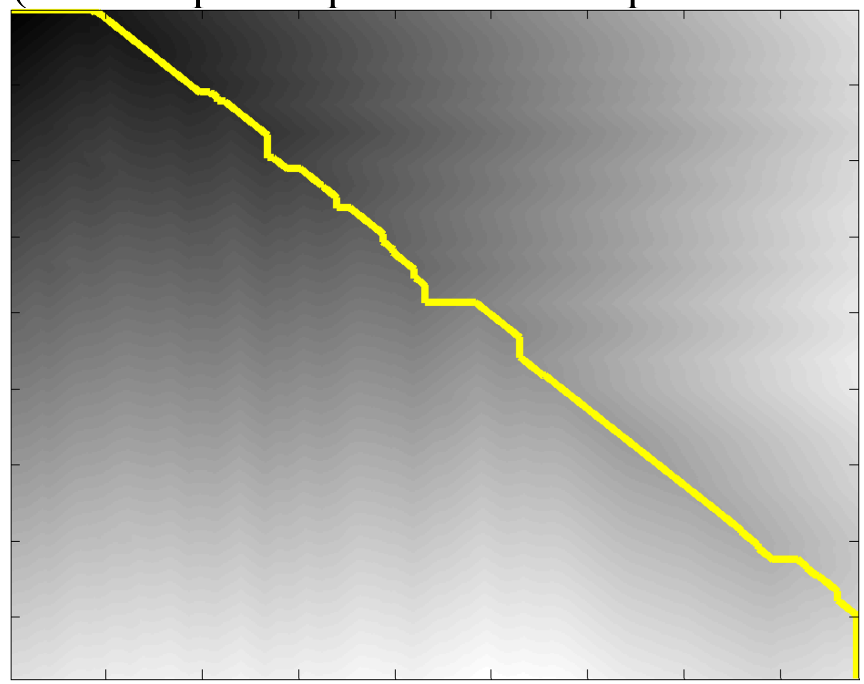
This problem just screams out for Dynamic Programming!
(which, indeed, is how Cox et.al. solve the problem)

Real Scanline Example

DSI



DP cost matrix
(cost of optimal path from each point to END)



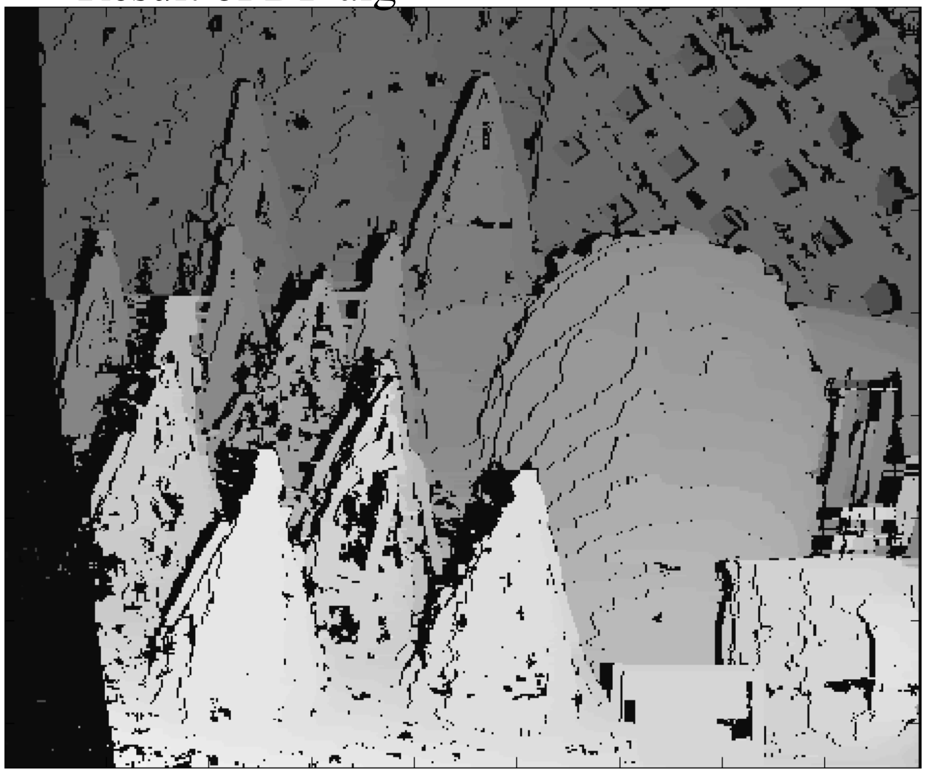
Every pixel in left column now is marked with either a disparity value, or an occlusion label.

Proceed for every scanline in left image.

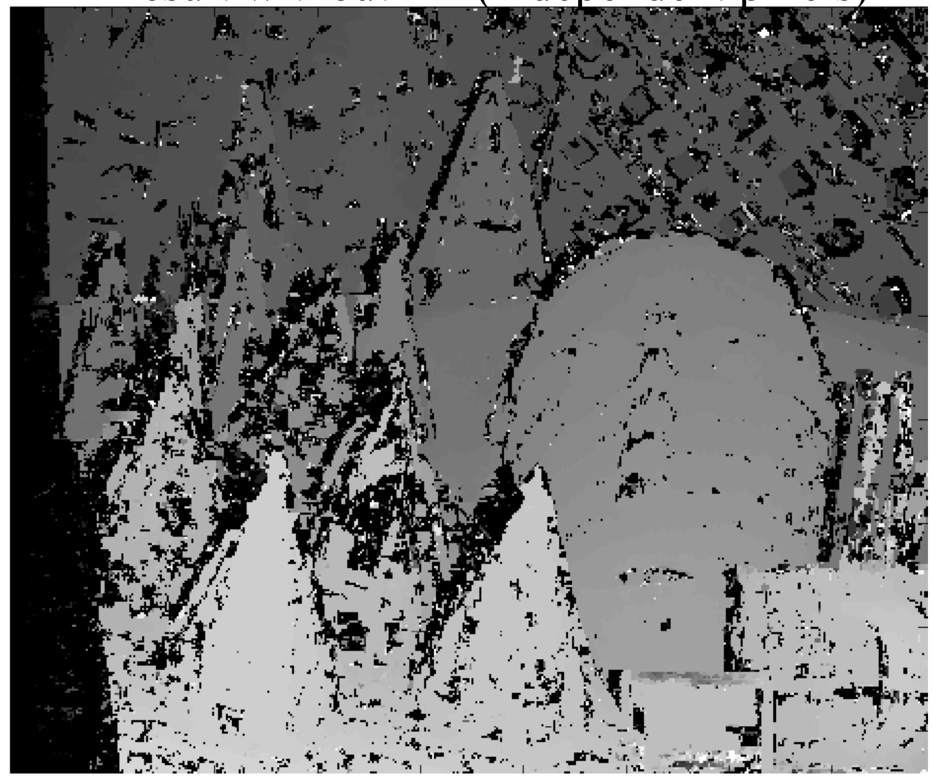
Robert Collins
CSE486, Penn State

Example

Result of DP alg



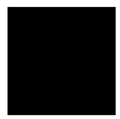
Result without DP (independent pixels)



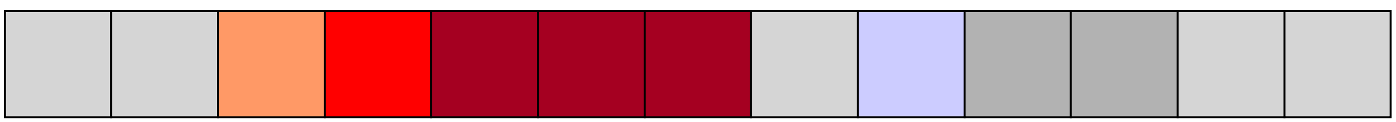
Result of DP alg. Black pixels = occluded.

Occlusion Filling

Simple trick for filling in gaps caused by occlusion.

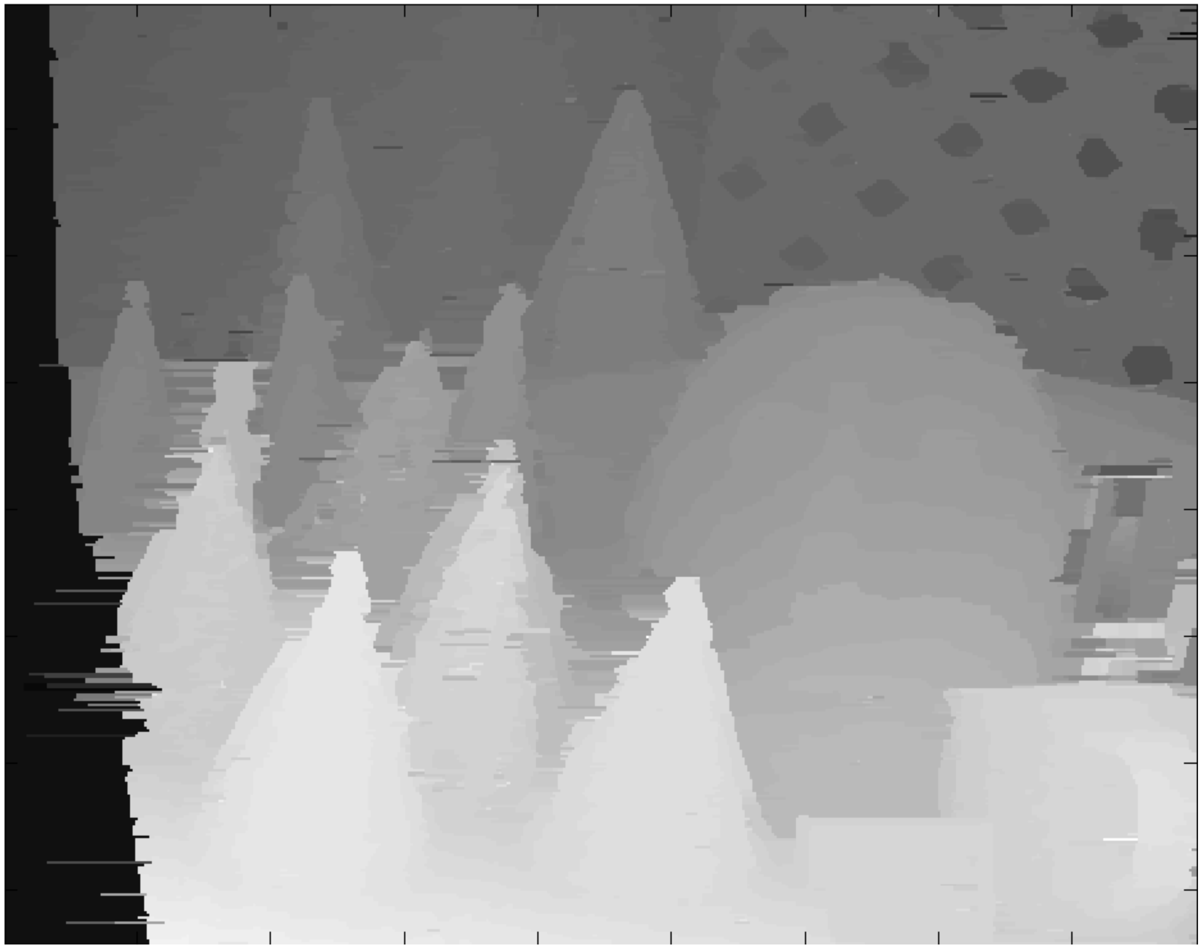
 = left occluded

Fill in left occluded pixels with value from the nearest valid pixel preceding it in the scanline.



Similarly, for right occluded, look for valid pixel to the right.

Example

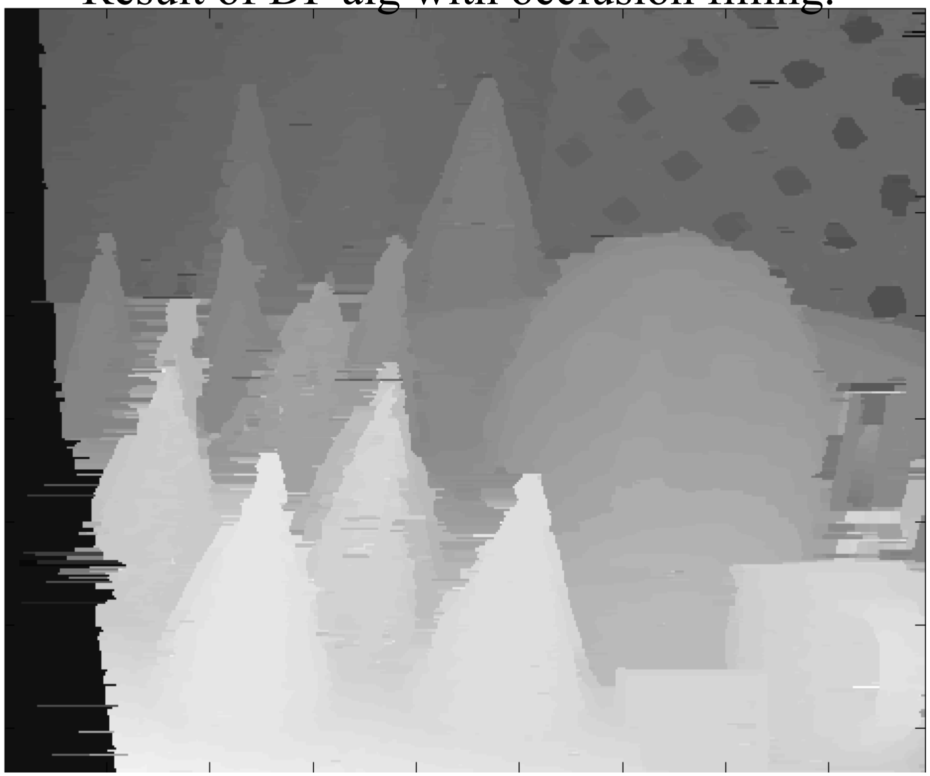


Result of DP alg with occlusion filling.

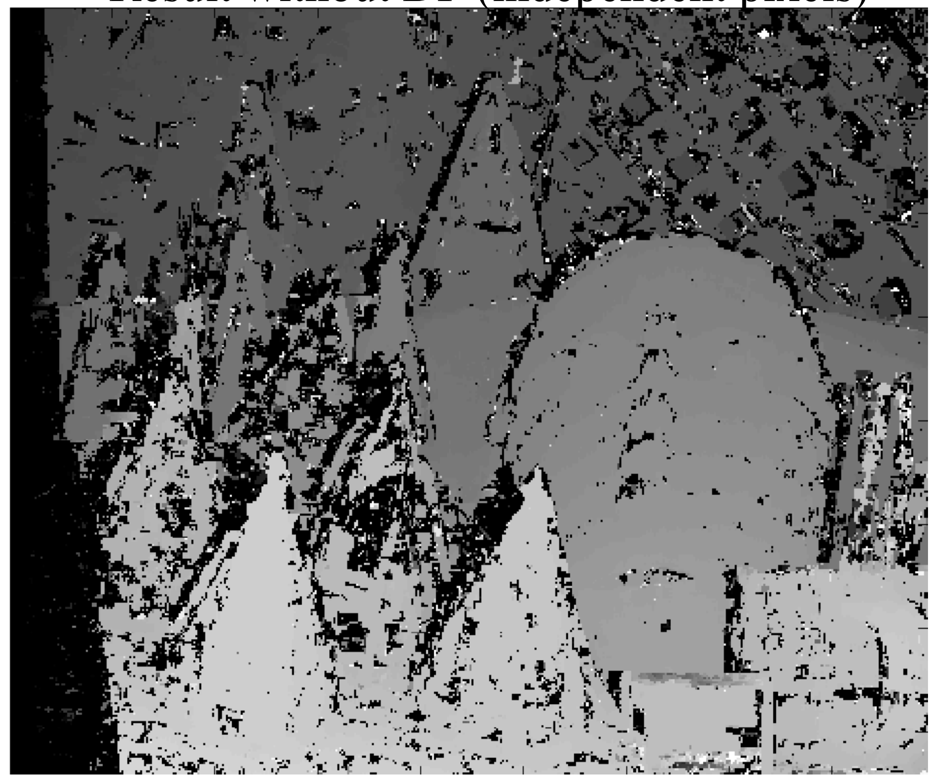
Robert Collins
CSE486, Penn State

Example

Result of DP alg with occlusion filling.

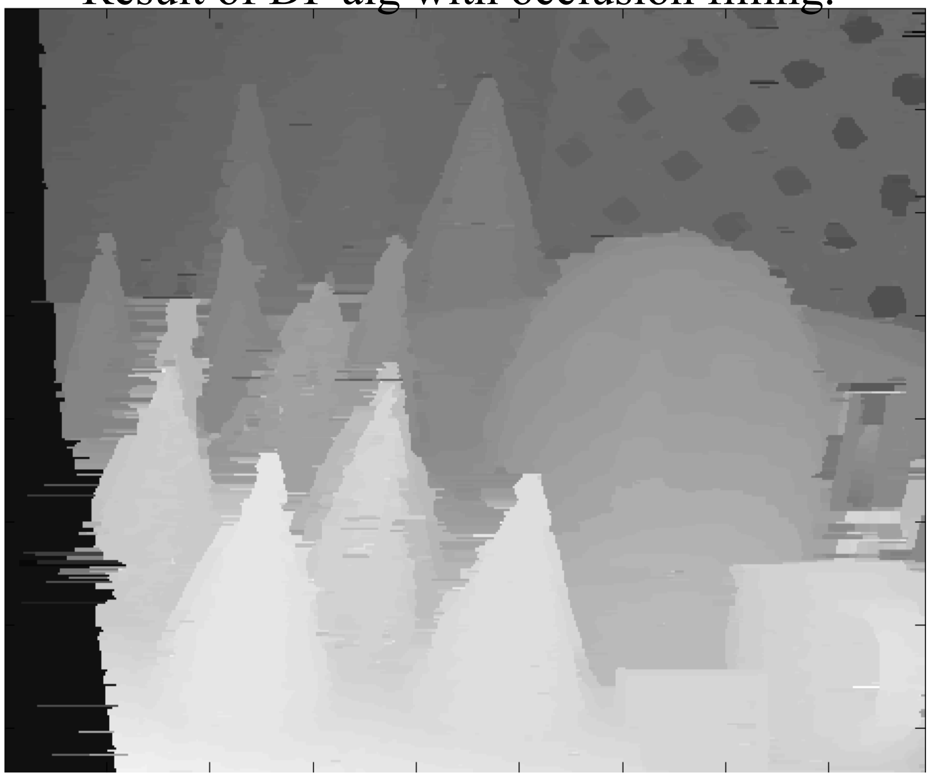


Result without DP (independent pixels)

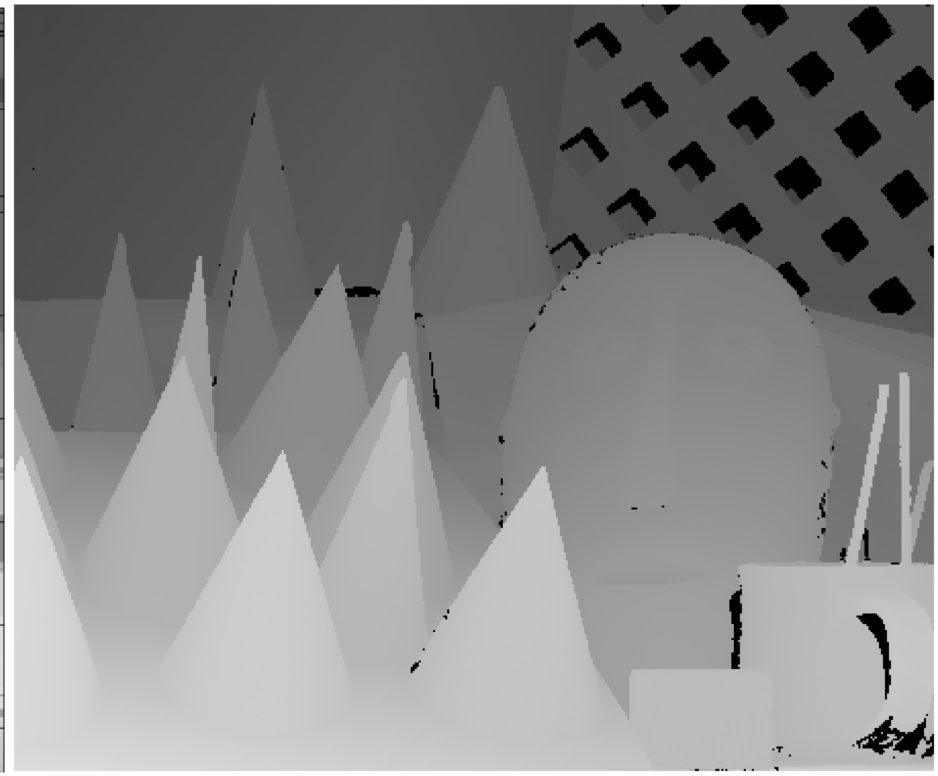


Example

Result of DP alg with occlusion filling.



Ground truth



www.middlebury.edu/stereo/

File Edit View Favorites Tools Help

Back Search Favorites

Address http://cat.middlebury.edu/stereo/ Go Links

Middlebury College
Stereo Vision Research Page

Daniel Scharstein
Richard Szeliski

Home | Data | Evaluation | Results | Code | **New Data**

Welcome to the Middlebury Stereo Vision Page

This website contains material accompanying our taxonomy and experimental comparison of stereo correspondence algorithms [1]. It contains stereo data sets with ground truth, the overall comparison of algorithms, instructions on how to evaluate your stereo algorithm in our framework, and our stereo correspondence software.

Also available are two new stereo data sets with ground truth obtained using our structured lighting technique [2]. These data sets have a more complex geometry and larger disparity ranges than the original data sets.

We are continually inviting other researchers to run their stereo algorithms on the four image pairs used in our overall comparison, and to send us the results. We will then run our evaluator, and report the resulting disparity error statistics. If you are interested in participating, please go to the evaluation page.

How to cite the materials on this web site:
We grant permission to use and publish all images and numerical results on this website. However, if you use our data sets, and/or report performance results, we request that you cite the appropriate paper(s) [1, 2]. If you want to cite this website, please use the "stable" URL "**www.middlebury.edu/stereo**". (This URL is currently auto-forwarded to "cat.middlebury.edu/stereo", but that may change.)

References:

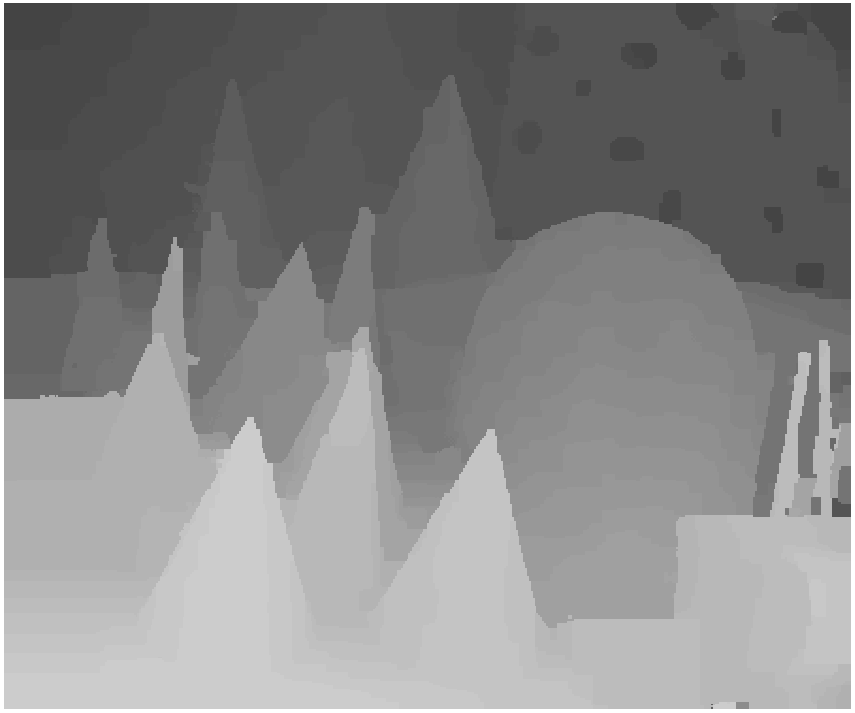
[1] D. Scharstein and R. Szeliski. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms.
IJCV 47(1/2/3):7-42, April-June 2002. PDF file (1.15 MB) - includes current evaluation.
Microsoft Research Technical Report MSR-TR-2001-81, November 2001. PDF file (1.27 MB).

[2] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light.
In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, volume 1, pages 195-202, Madison, WI, June 2003. PDF file (1.2 MB)

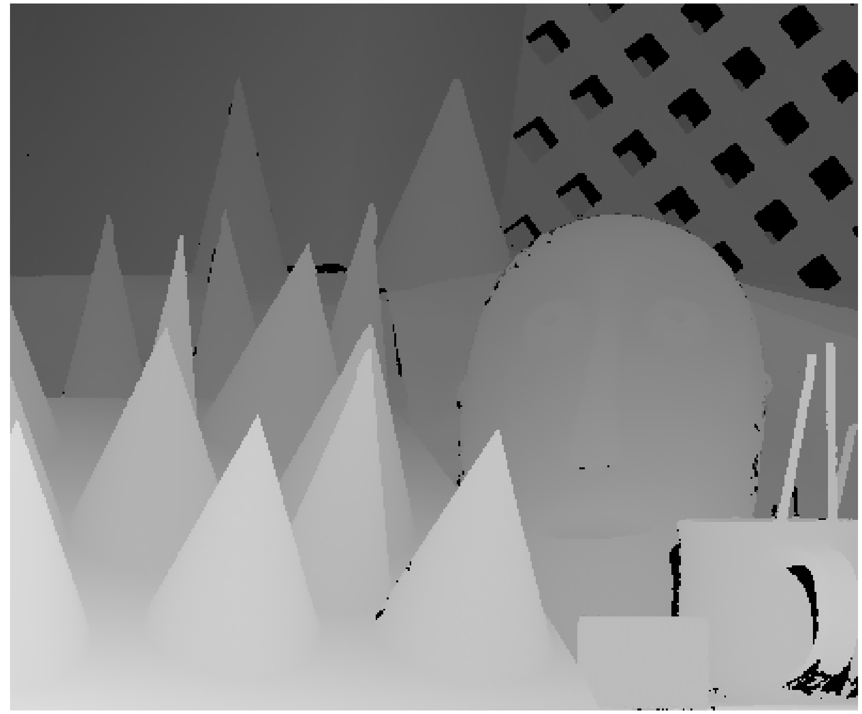
Support for this work was provided in part by NSF CAREER grant 9984485. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Robert Collins
CSE486, Penn State

State-of-the-Art Results



Algorithm Results



Ground truth

J. Sun, Y. Li, S.B. Kang, and H.-Y. Shum. "Symmetric stereo matching for occlusion handling". IEEE Conference on Computer Vision and Pattern Recognition, June 2005.