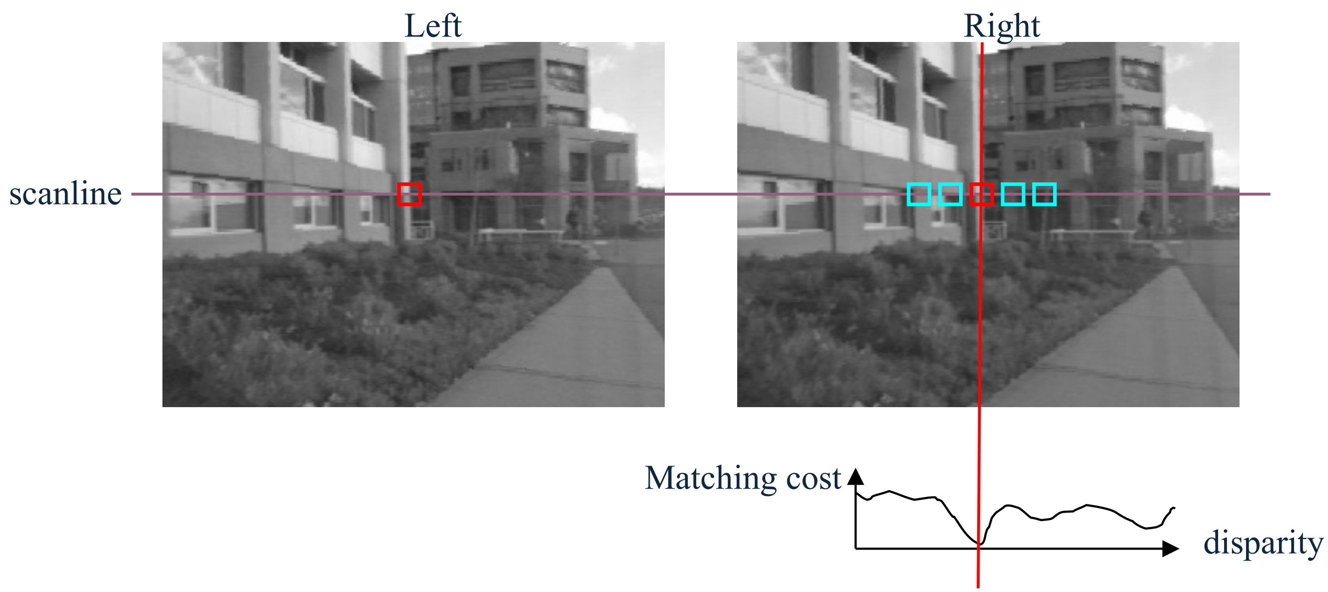
Stereo Matching Methods

D.A. Forsyth,

University of Illinois at Urbana Champaign

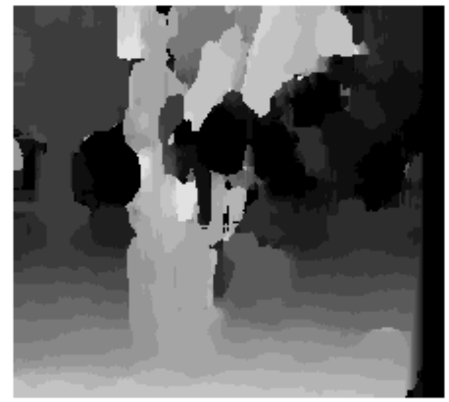
Correspondence search



- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

Effect of window size on correspondence search

Window size 3

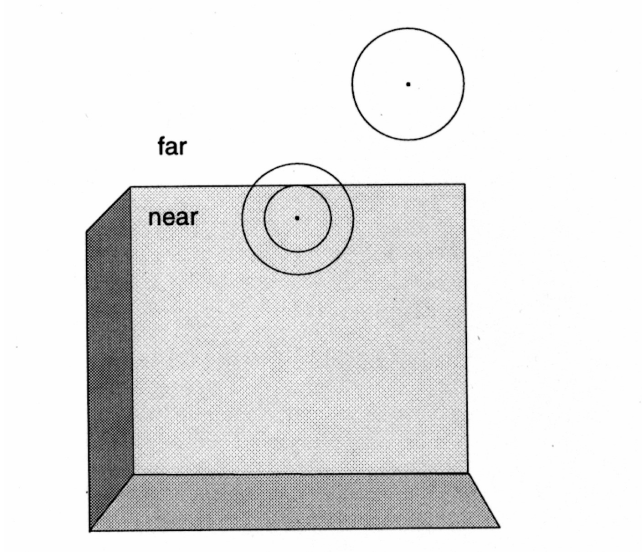


Window size 20

- Smaller window:
  - + More detail
  - • More noise
- Larger window:
  - + Smoother disparity maps
  - – Less detail

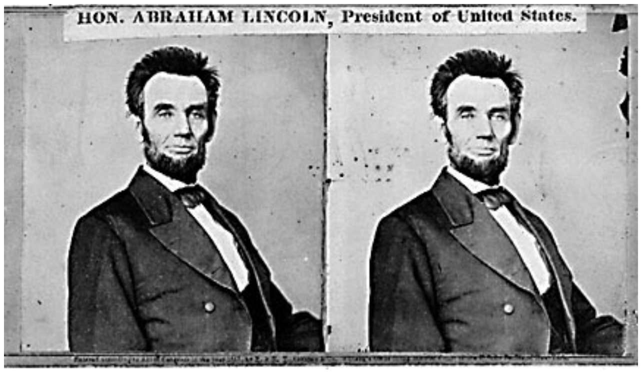
More interesting matching costs

- Compute filter outputs for many filters
- SSD those
- Advantage: more detailed description of points
- Disadvantage: Filter support interacts badly with fast changes in depth



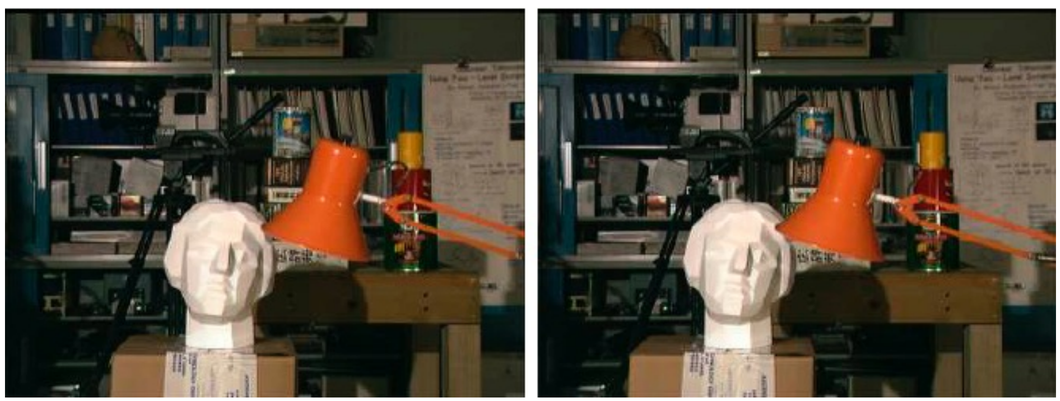
From Jones and Malik, “A computational framework for determining Stereo correspondences from a set of linear spatial filters

Where will basic search fail?



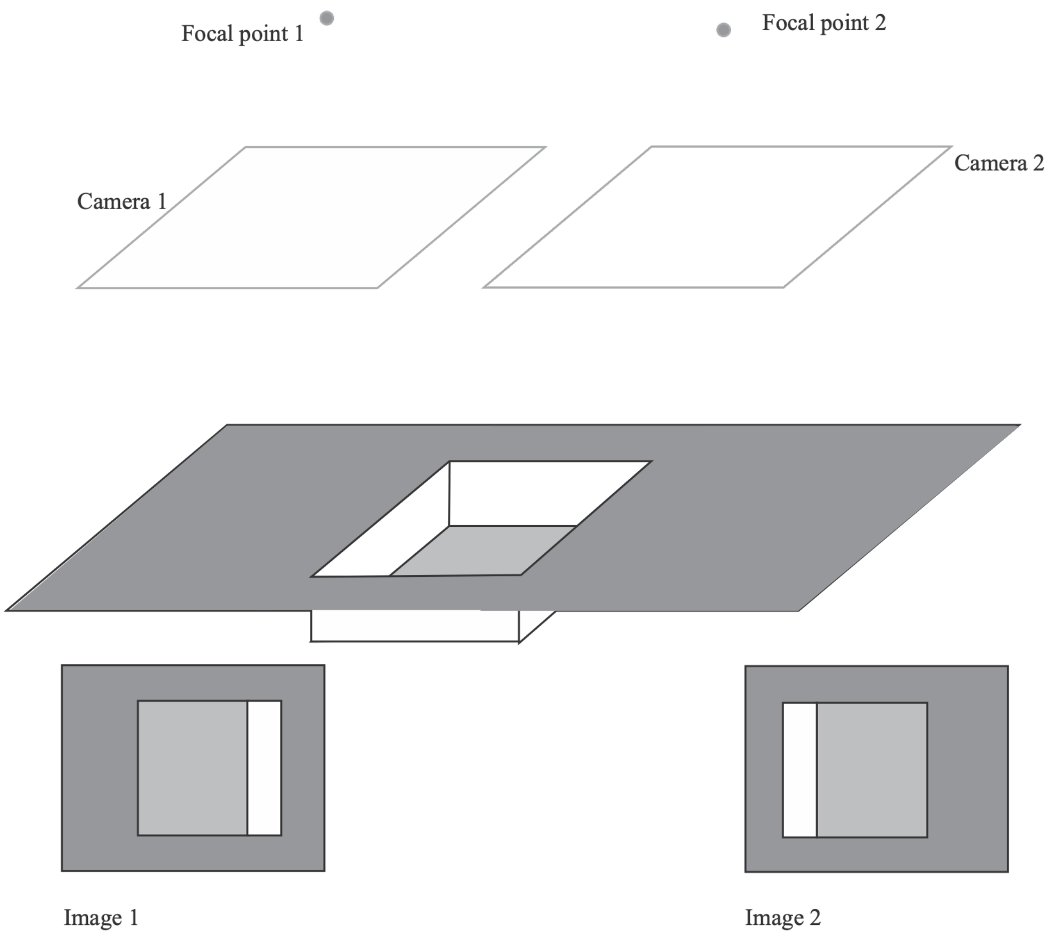
Textureless surfaces

Occlusions, repetition



Non-Lambertian surfaces, specularities

Dealing with Da Vinci



- Consequence:
- some points in L (R) image do not
- have matches in R (L) image
- Note asymmetry in previous algorithm
- (find best matching point in other side - what if best matches aren't consistent?)

- Strategy:
- match L to R
- match R to L
- use only consistent matches

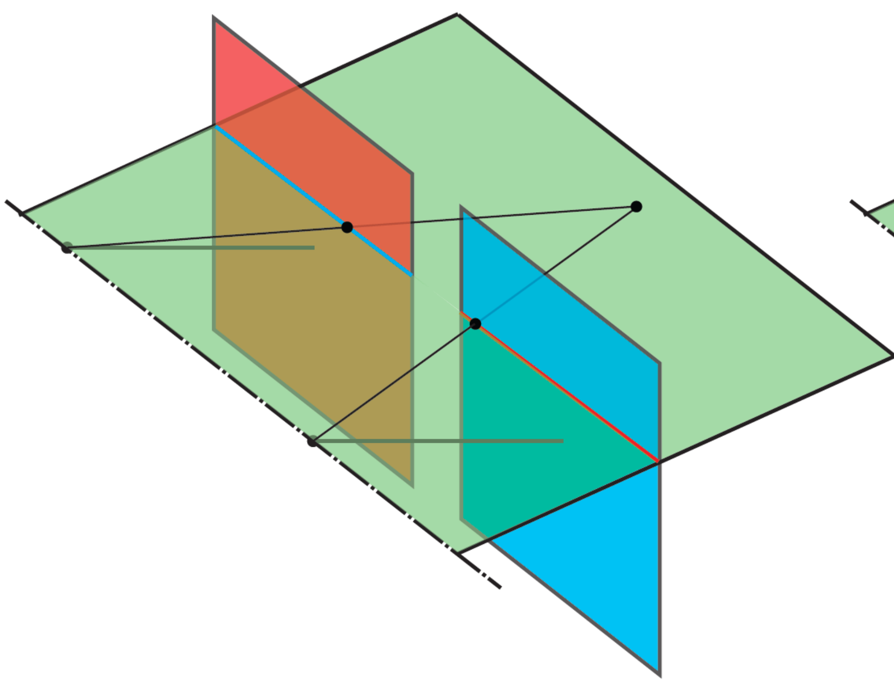
Stereo as optimization, simple

- Do this line by line
    - move along line in left image
        - parameters:
            - at each location x in left, best matching location b(x) in right.
    - objective:
        - photometric consistency
            - compare $I_2(b(x))$ to $I_1(x)$
- Constraints
    - Uniqueness
    - Ordering

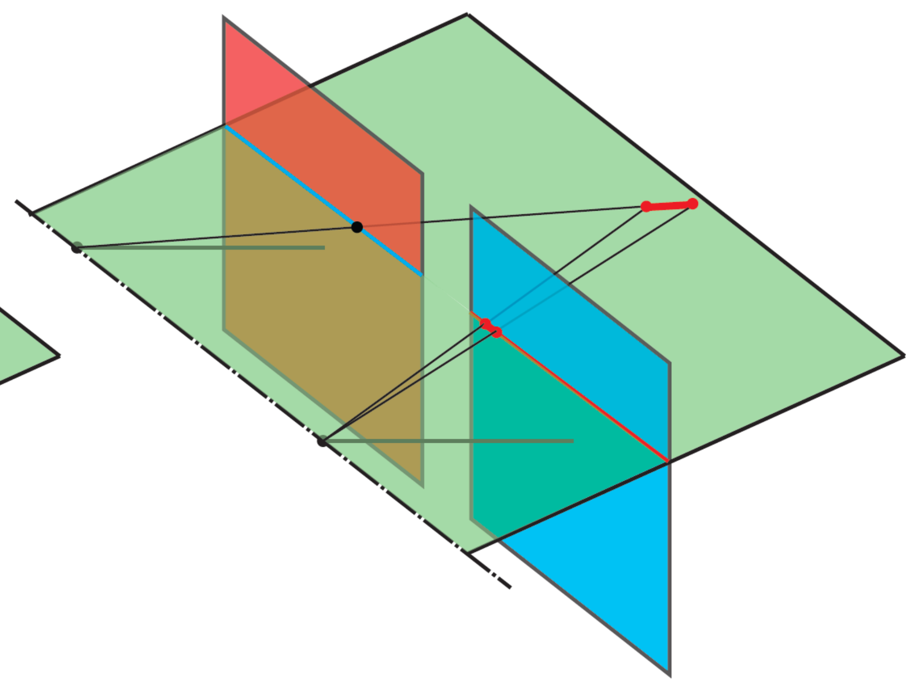
Uniqueness

- Each point in one should match at most one point in the other
  - Mostly, but not always, true

Uniqueness



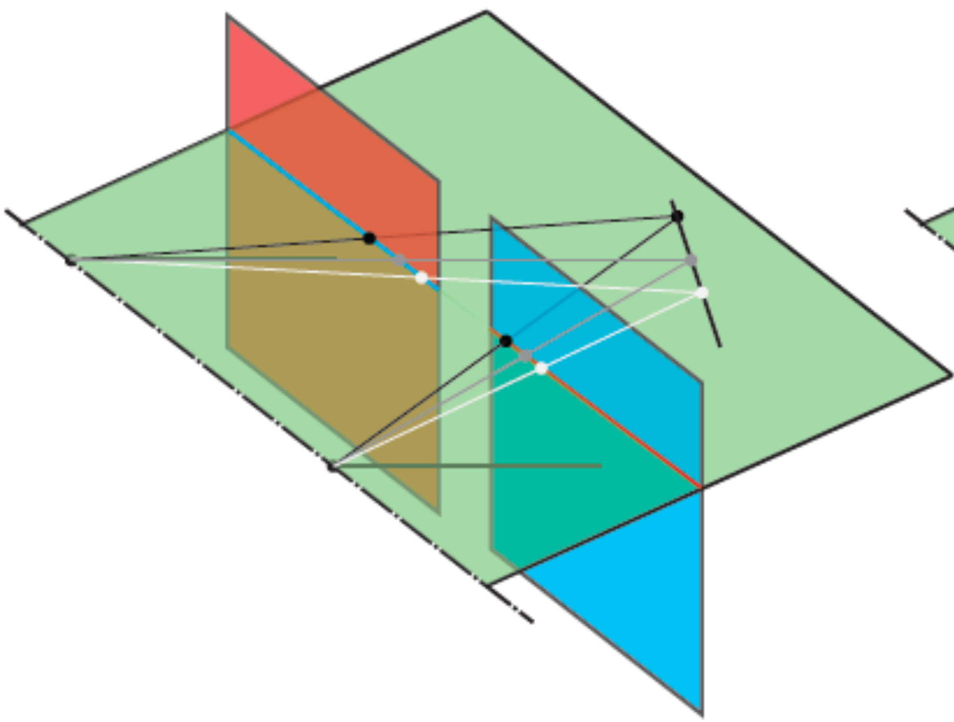
Violation



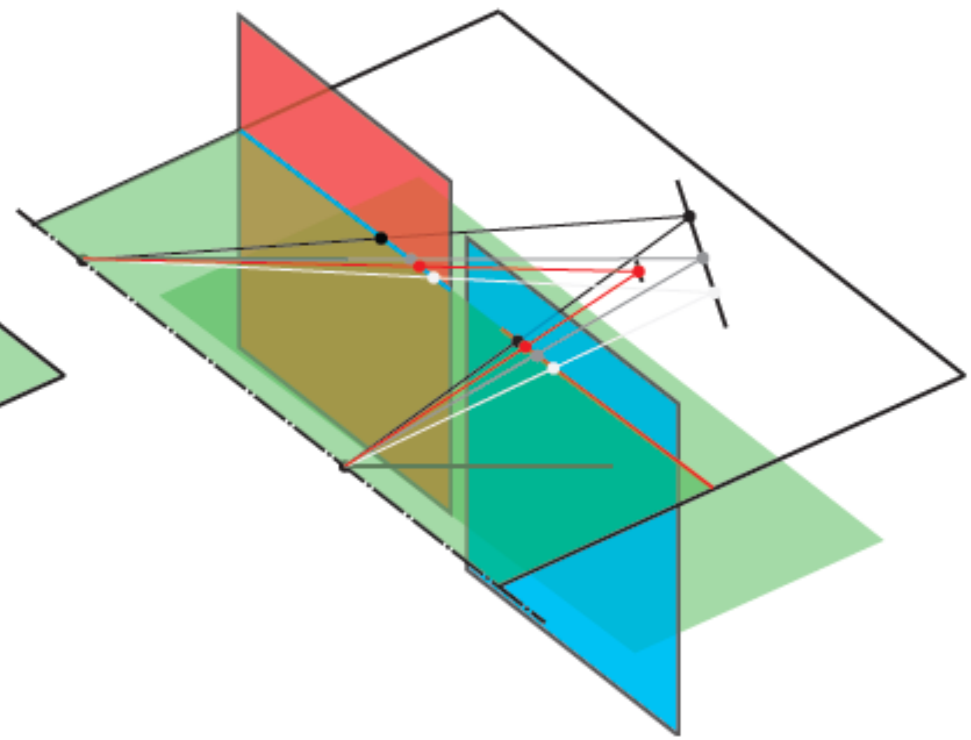
Ordering

- Corresponding points should appear in the same order
  - Mostly, but not always, true

Ordering

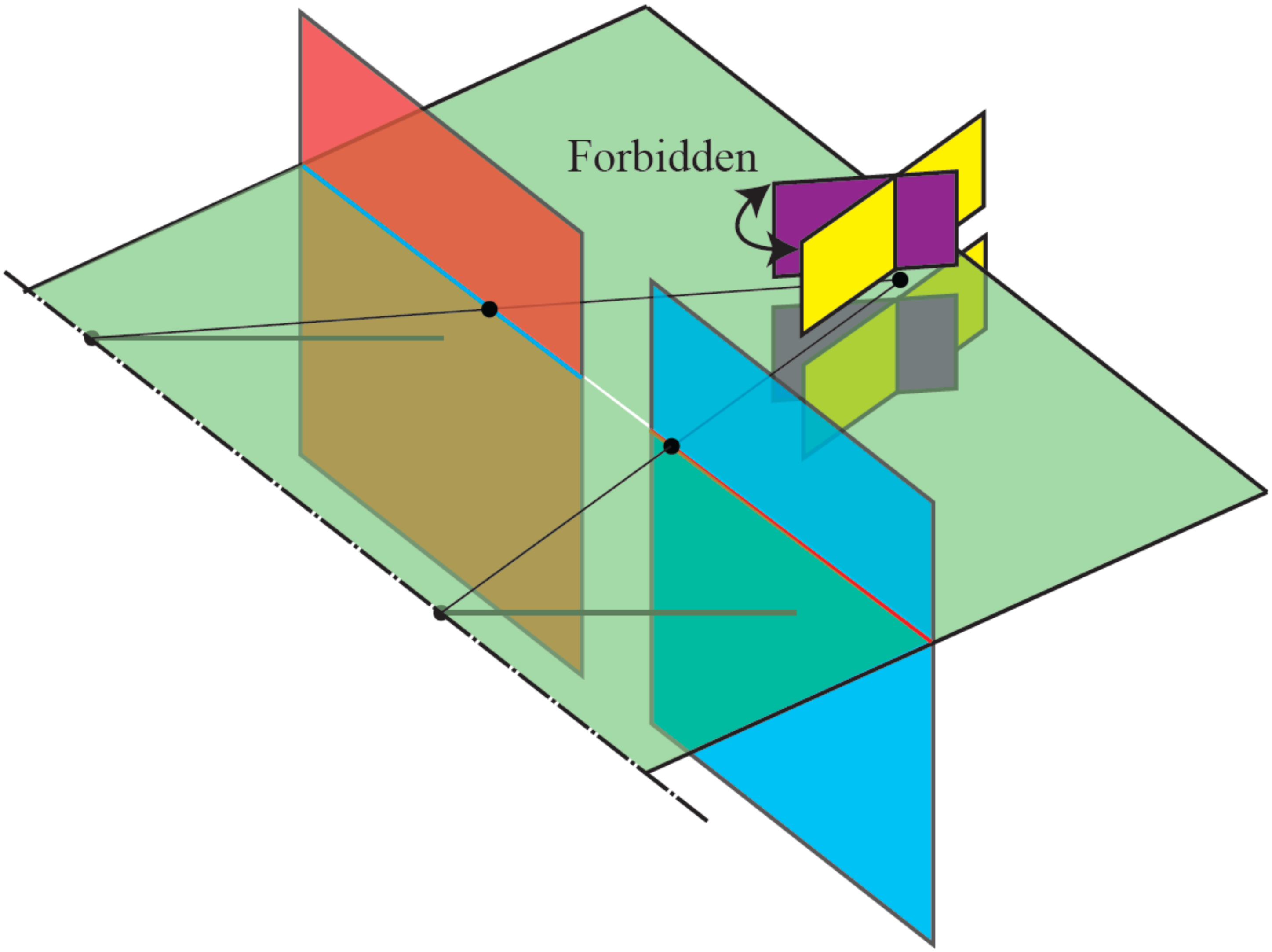


Violation



Disparity gradient constraint

- The disparity gradient is limited



Smoothness

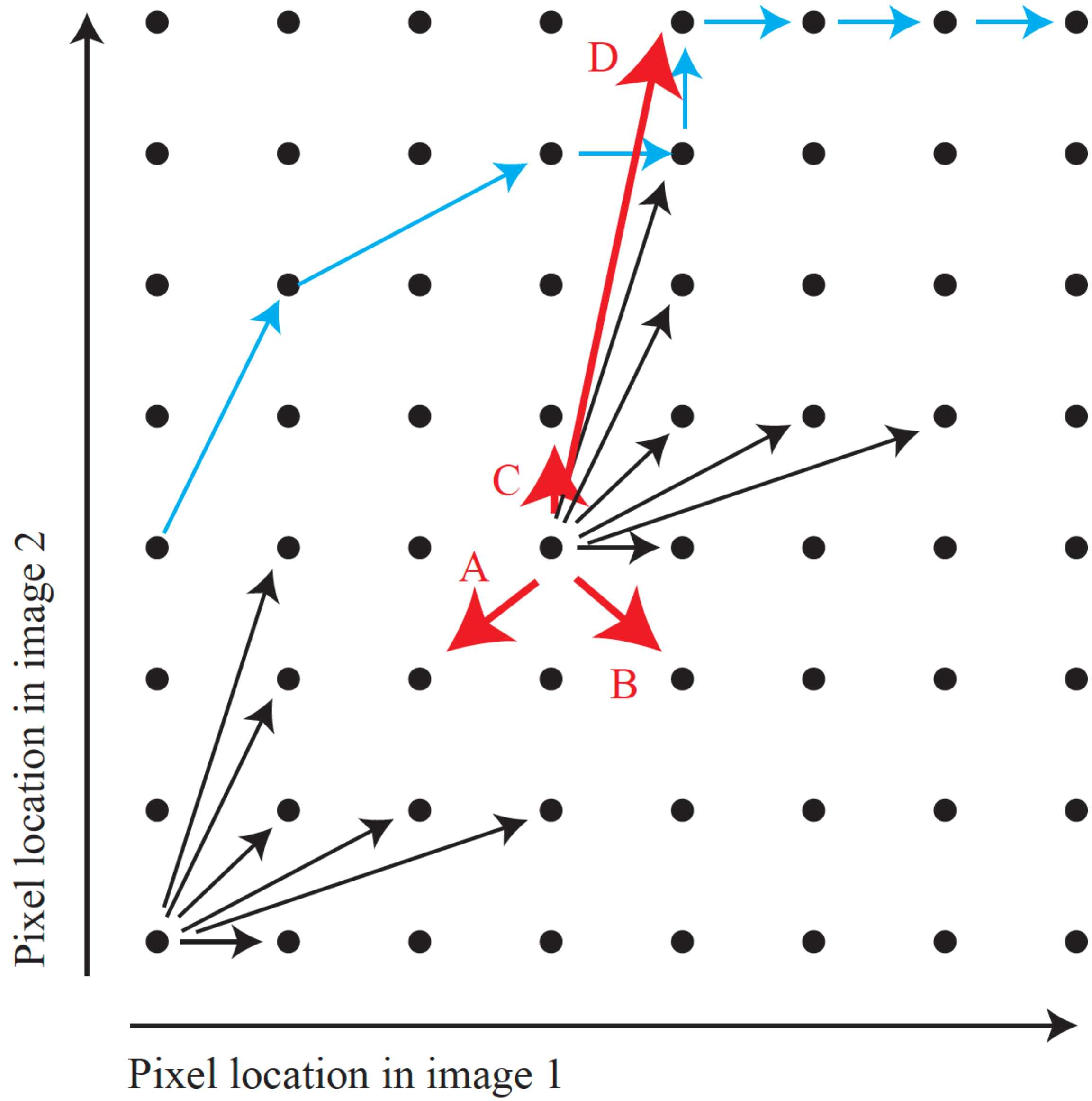
- Large disparity gradients should be rare
  - because they imply sharp depth gradients
  - and depth gradients tend to be small

Matching by dynamic programming

- Match pixels along the entire scanline while preserving uniqueness and ordering
- Different scanlines are still optimized independently

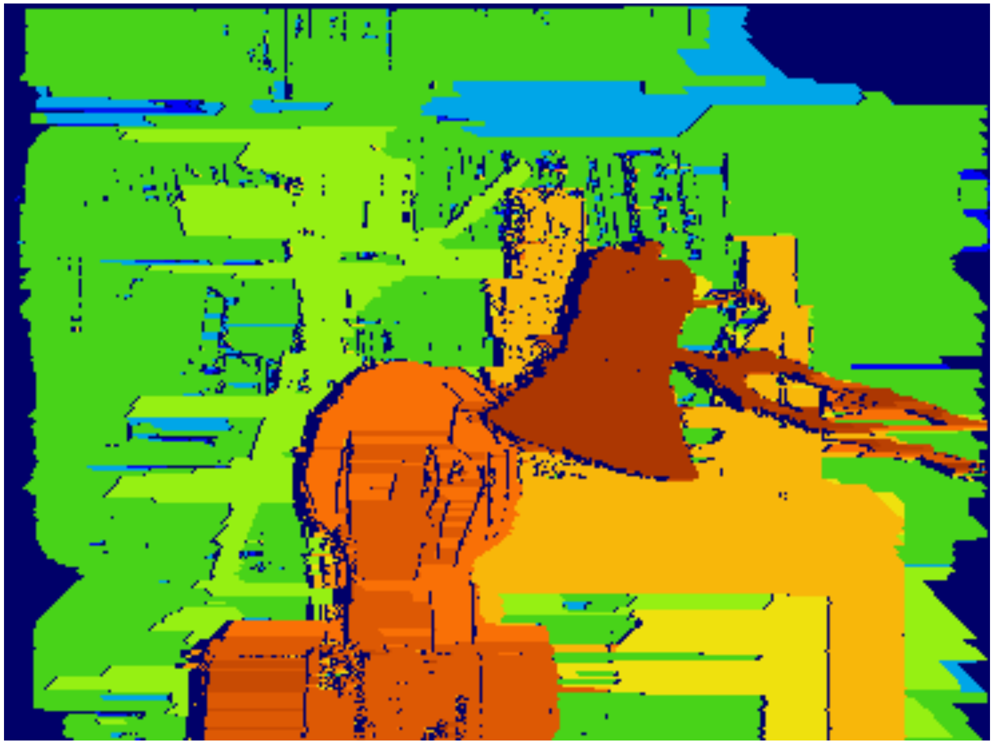
Y. Ohta and T. Kanade. [Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming]. IEEE Trans. PAMI, 1985

Matching by dynamic programming



Matching by dynamic programming

- Generates streaking artifacts!



Stereo as a conditional random field

- Originally:
  - find x_1, x_2 that match, compute depth
- Now:
  - choose depth/disparity at x_1 that causes x_2 to be best match

Setup

- Discretize disparity
  - For example, 256 values
  - Seek one-hot vector at each pixel
    - where the 1 tells which disparity at that pixel
- Set up cost function
  - Per pixel cost
    - disparity at this location implies match
      - is color (etc) at the matching location the same as at pixel?
  - Pairwise cost
    - is disparity at this location compatible with disparity at that?

Per pixel cost

- Decide photometric cost:

$$C(\delta; i, j) = \|\mathcal{I}(i, j) - \mathcal{I}(i + \delta, j)\|^2.$$

- Now use

$$\sum_{i,j} \left[\sum_{u \text{ in values}} [\mathbb{I}_{[\delta_{i,j}=u]}(\delta_{i,j}) C(u; i, j)] \right]$$

Basically, the one-hot vector

Pairwise costs

- Decide neighbors for a pixel
  - Examples:

$\mathcal{N}(i,j)$ could just be

$$\{(i+1,j)\}$$

(which would get us the dynamic programming case). Alternatively, it could be

$$\{(i+1,j),(i-1,j),(i,j+1),(i,j-1)\}$$

(the *four neighbors*). It could be

$$\{(i+1,j),(i-1,j),(i,j+1),(i,j-1),(i-1,j-1),(i+1,j-1),(i-1,j+1),(i+1,j+1)\}$$

(the *eight neighbors*).

Pairwise costs

Associated with each edge will be a cost function. This cost function penalizes pairs of disparities across the edge that are undesirable. Now write a cost function that compares disparity across pairs of pixels. For example, the cost function could be

$$B(u, v) = \begin{cases} \|u - v\|^2 & \text{if } \|u - v\| < k \\ L & \text{otherwise} \end{cases}$$

Now solve

per pixel costs

$$\sum_{i,j}\left[\sum_{u \text{ in values}}\left[\mathbb{I}_{[\delta_{i,j}=u]}(\delta_{i,j})C(u;i,j)\right]\right] \quad +$$

$$\sum_{i,j}\left[\sum_{k,l\in\mathcal{N}(i,j)}\left[\sum_{u,v \text{ in values}}\left[\mathbb{I}_{[\delta_{i,j}=u]}(\delta_{i,j})\mathbb{I}_{[\delta_{k,l}=v]}(\delta_{k,l})B(u,v)\right]\right]\right].$$

pairwise costs

Basically, the one-hot vector

Summary

- This is an intractable optimization problem
  - with excellent approximation properties
- For a small set of cases, true optimum is known
  - (by good luck)
- Yields:
  - excellent stereo algorithms

Y. Boykov, O. Veksler, and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Examples

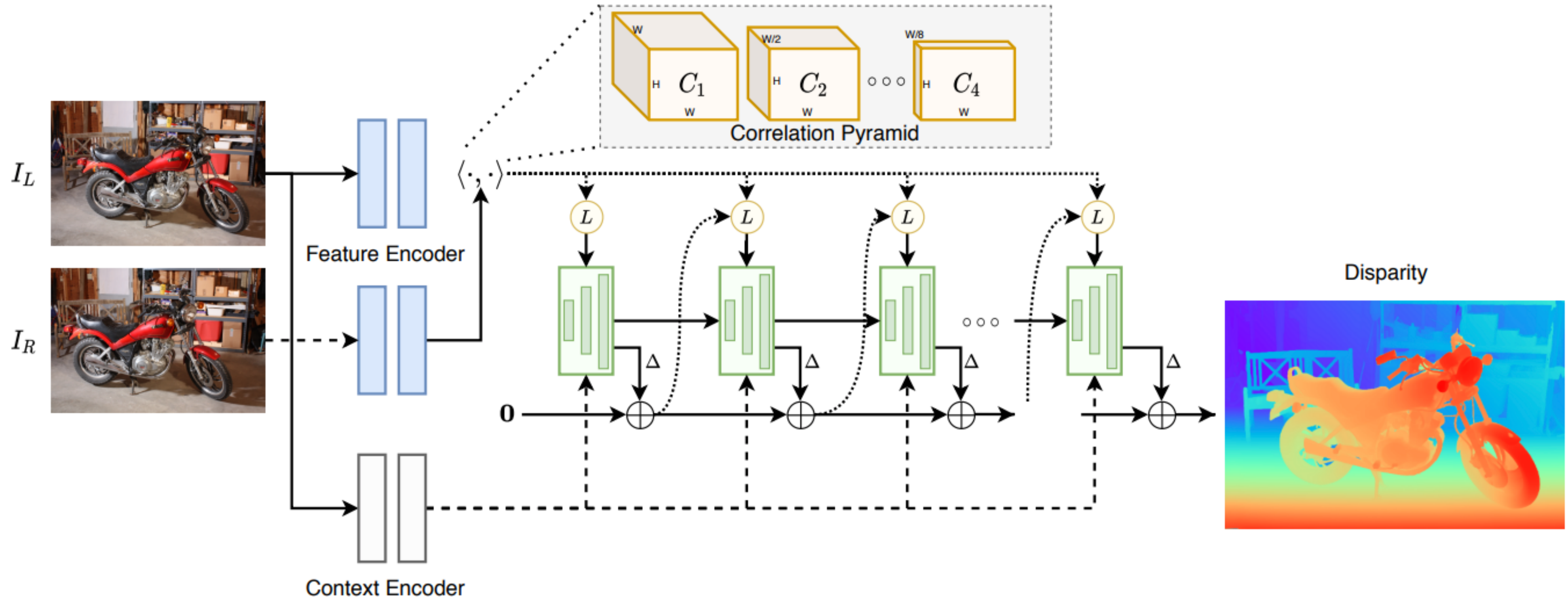
Resources

- Middlebury stereo
  - start at
    - https://vision.middlebury.edu/stereo/
- KITTI
  - start at
    - https://www.cvlibs.net/datasets/kitti/eval_stereo.php

The application of networks

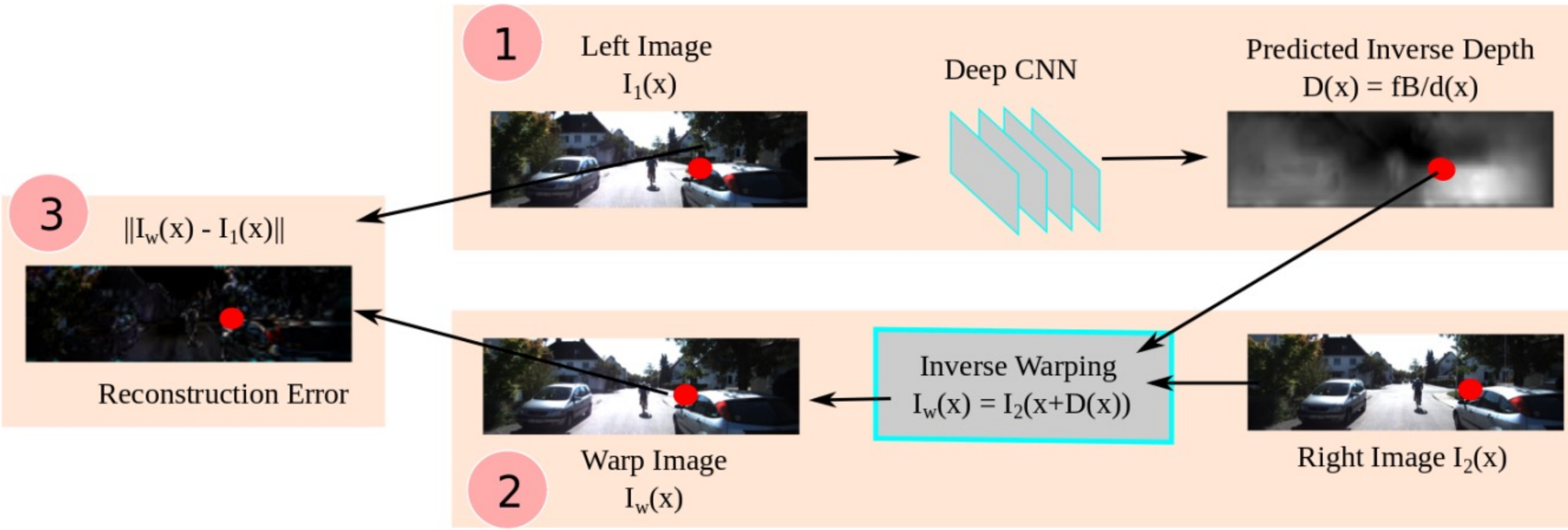
- VERY like optic flow
  - RAFT works rather well
- Tremendous opportunity for self-supervision
  - with photometric consistency

Stereo matching with deep networks



L. Lipson et al. [RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching](). arXiv 2021

Self-supervised depth estimation



R. Garg et al. [Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue](). ECCV 2016