3D Vision: Theory, Application and New Trends

Dense Correspondence Estimation

Sudipta N. Sinha
Microsoft Research, Redmond, USA

July 4, 2018

3rd SUMMER SCHOOL ON COMPUTER VISION,
BASICS OF MODERN AI, 2—7 July 2018, IIIT Hyderabad

Overview

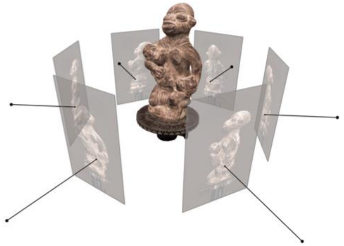
- Correspondence Problems in Computer Vision
- Stereo Matching
  - Semi Global Matching (SGM) and extensions
  - Priors and optimization
  - Deep Learning for stereo
- Scene Flow with Motion Segmentation

Image to Image correspondence

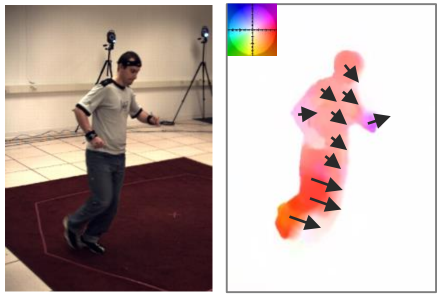
Geometric

Binocular Stereo

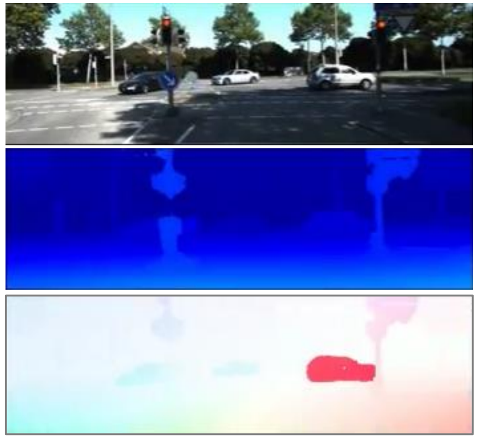
Multiview stereo



Optical flow



Scene Flow



Semantic

SIFT Flow (Liu+ 2008)

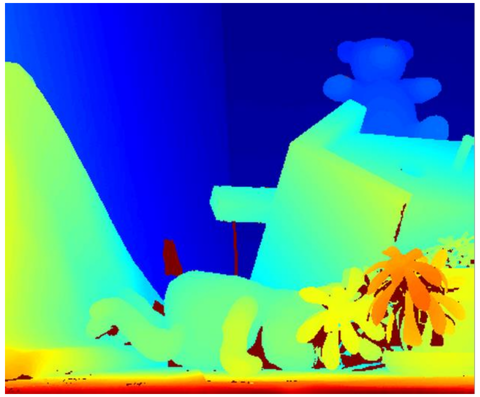
Deformable Spatial Pyramids (Kim+ 2013)

Joint Correspondence and Cosegmentation (Taniai+ 2016)

Stereo Matching

Left

Right



Left Disparity Map

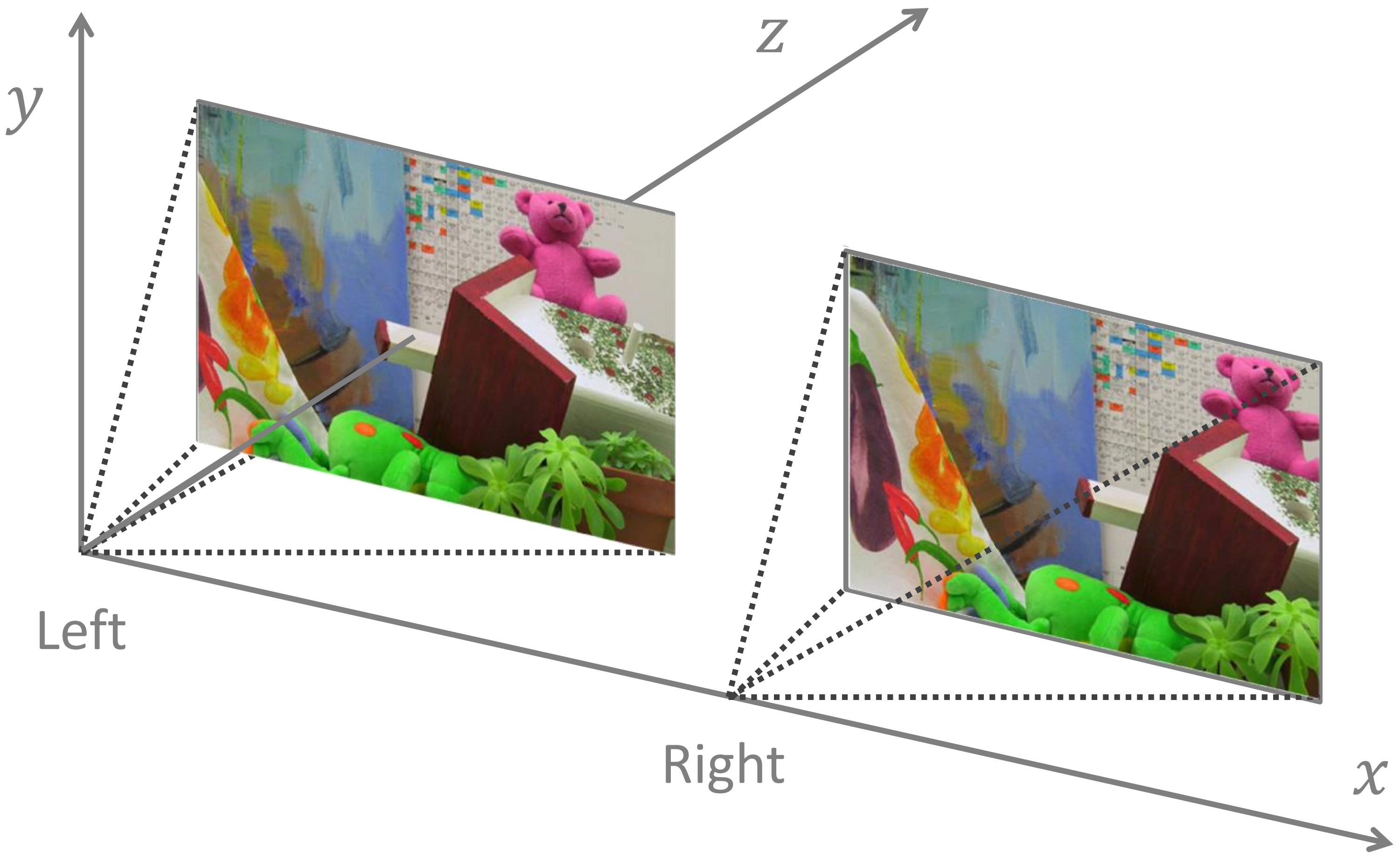
- Dense pixel correspondence in rectified pairs
- Disparity Map: $D(x, y)$

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

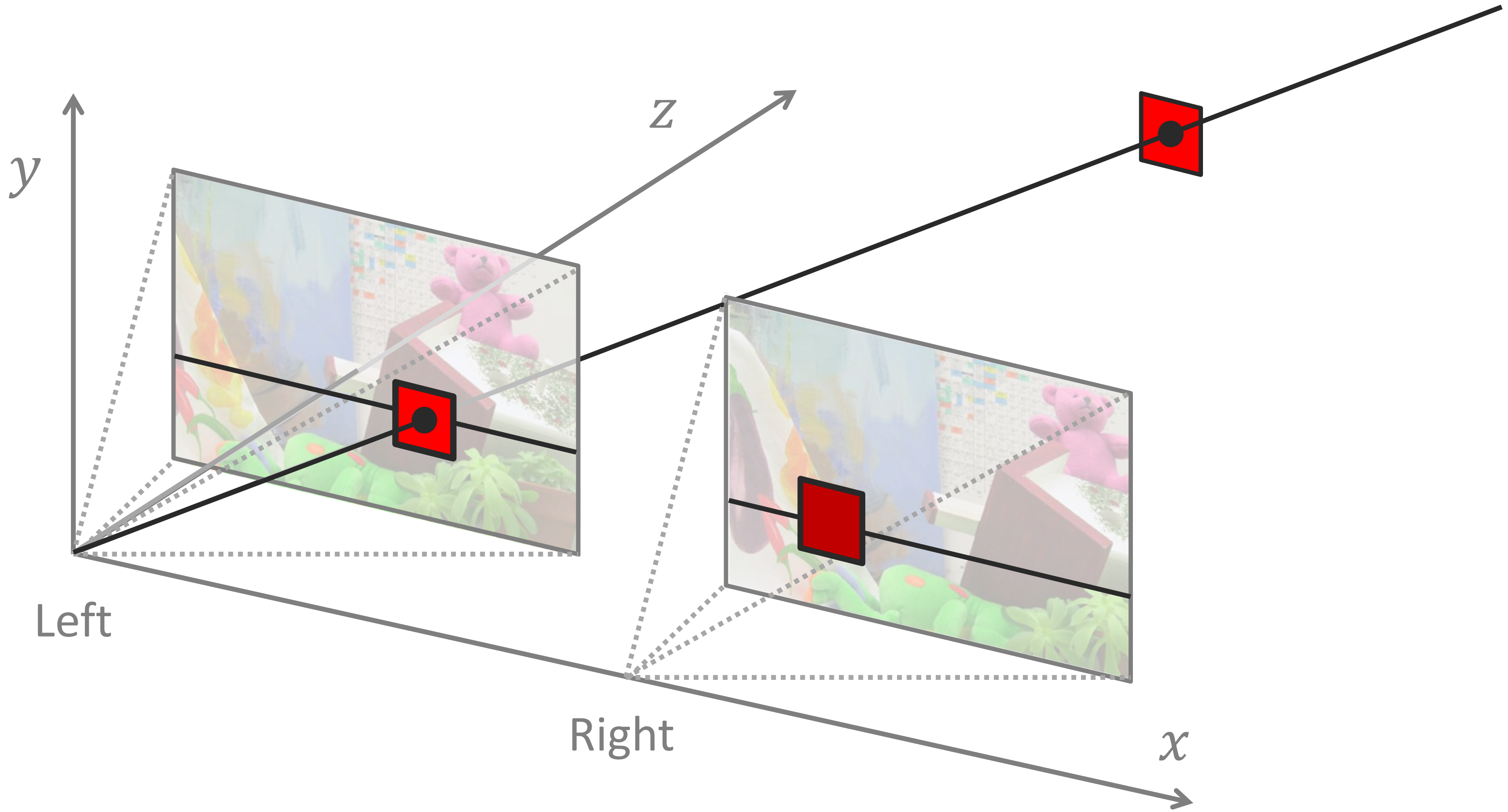
- Depth Map: $Z(x, y) = \frac{bf}{D(x,y)}$

Depth Map

Binocular Stereo Matching

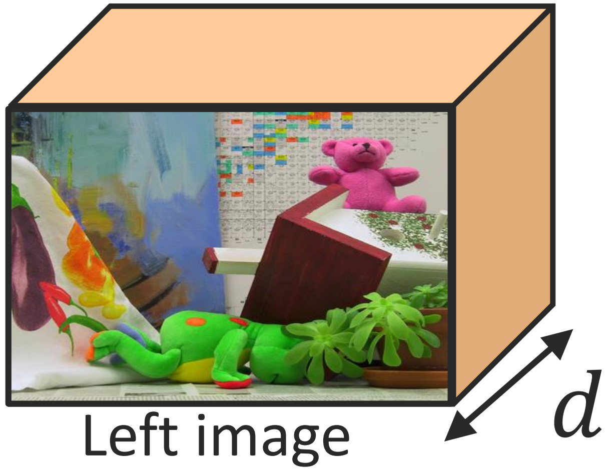


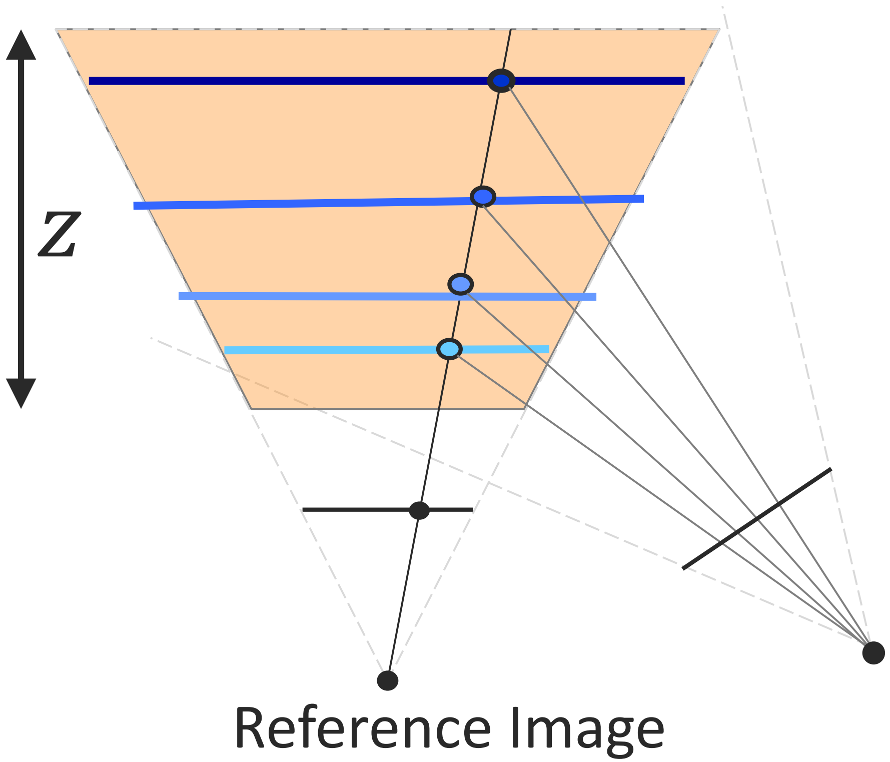
Binocular Stereo Matching



Discrete Search Space

- Disparity Space Image
  - 1D horizontal shifts $(d_{min}\ , d_{max})$
- Plane Sweep Volume
  - Search over depths ..
  - Stereo Rectification not needed
- Issue of fronto-parallel bias





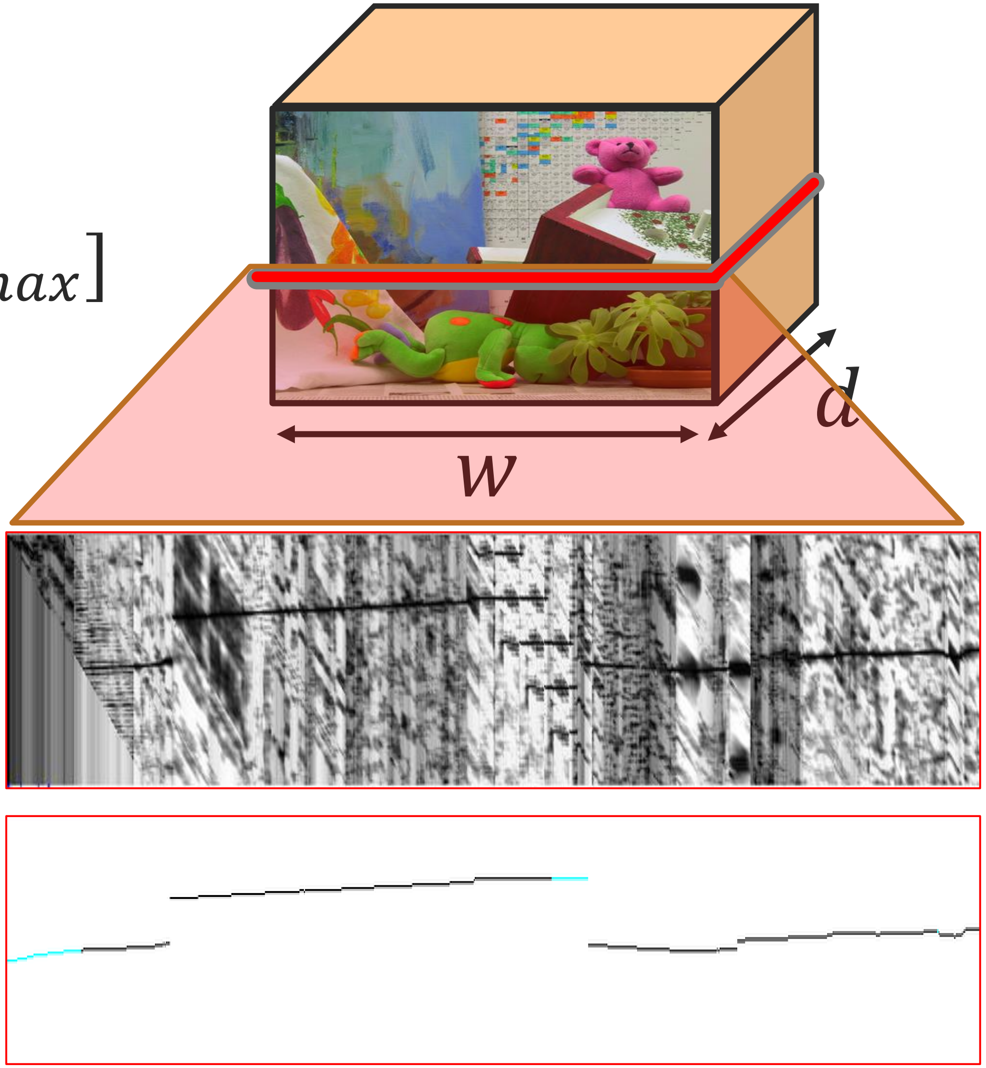
Matching Cost Volume

- Disparity Search Space
  - Discrete 1D horizontal shifts $[d_{min}, d_{max}]$
- Matching (dissimilarity) cost
  - Hand engineered or learned features

Objective:

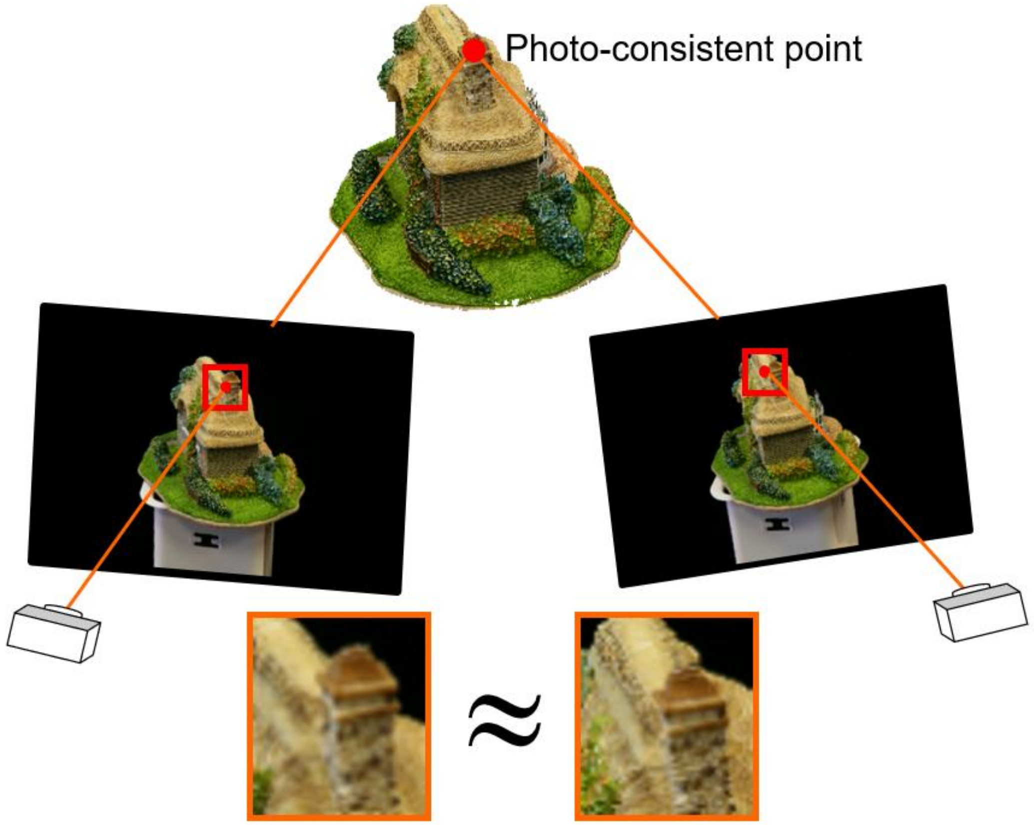
Assign per-pixel disparities that minimize the matching costs.

Left image DSI

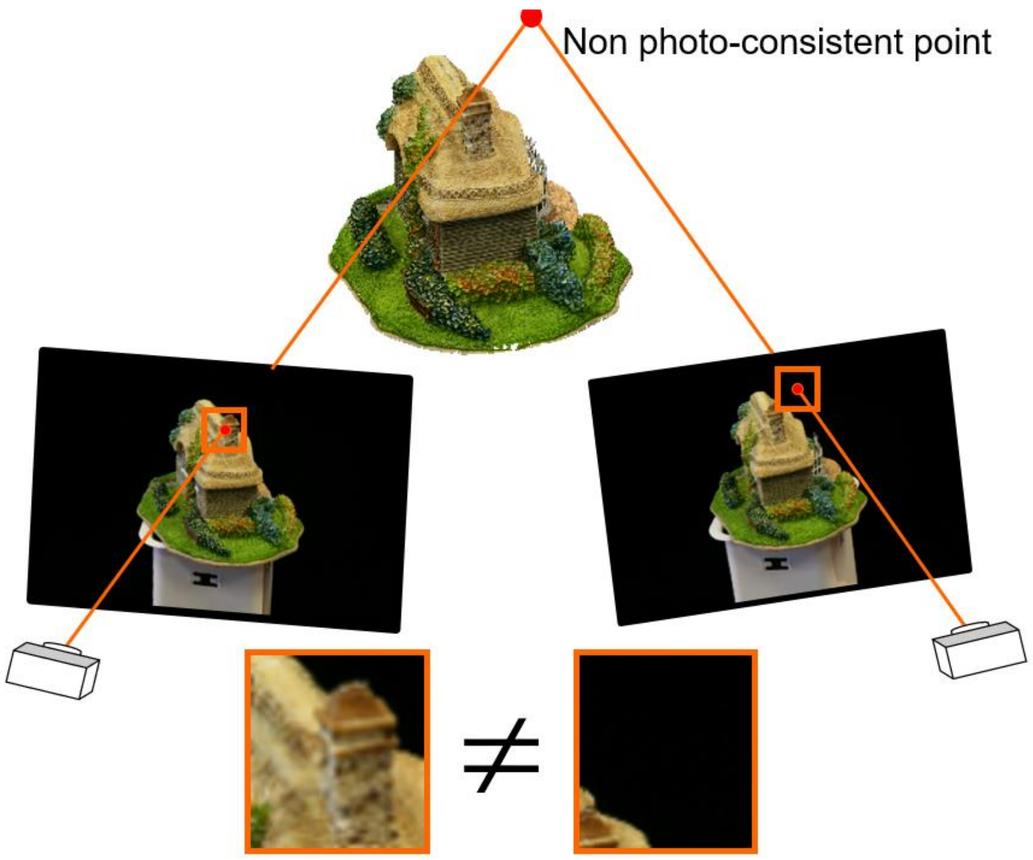


Ground truth surface (cross-section)

Need a way to compare an image patch



Correct match: low cost



Incorrect match: high cost

Matching costs

- Find pairs of pixels (or local patches) with similar appearance
- Minimize *matching cost (maximize photo-consistency)*
  - **Patch**-based (parametric vs non-parametric)
    - *Sum of Absolute Difference (SAD),*
    - *Sum of Squared Difference (SSD),*
    - *Normalized Cross Correlation (ZNCC)*
    - *Census, Rank filter, …*

$$C_{SAD}(\mathbf{p},\mathbf{d}) = \sum_{\mathbf{q}\in N_{\mathbf{p}}} |I_L(\mathbf{q}) - I_R(\mathbf{q}-\mathbf{d})|$$

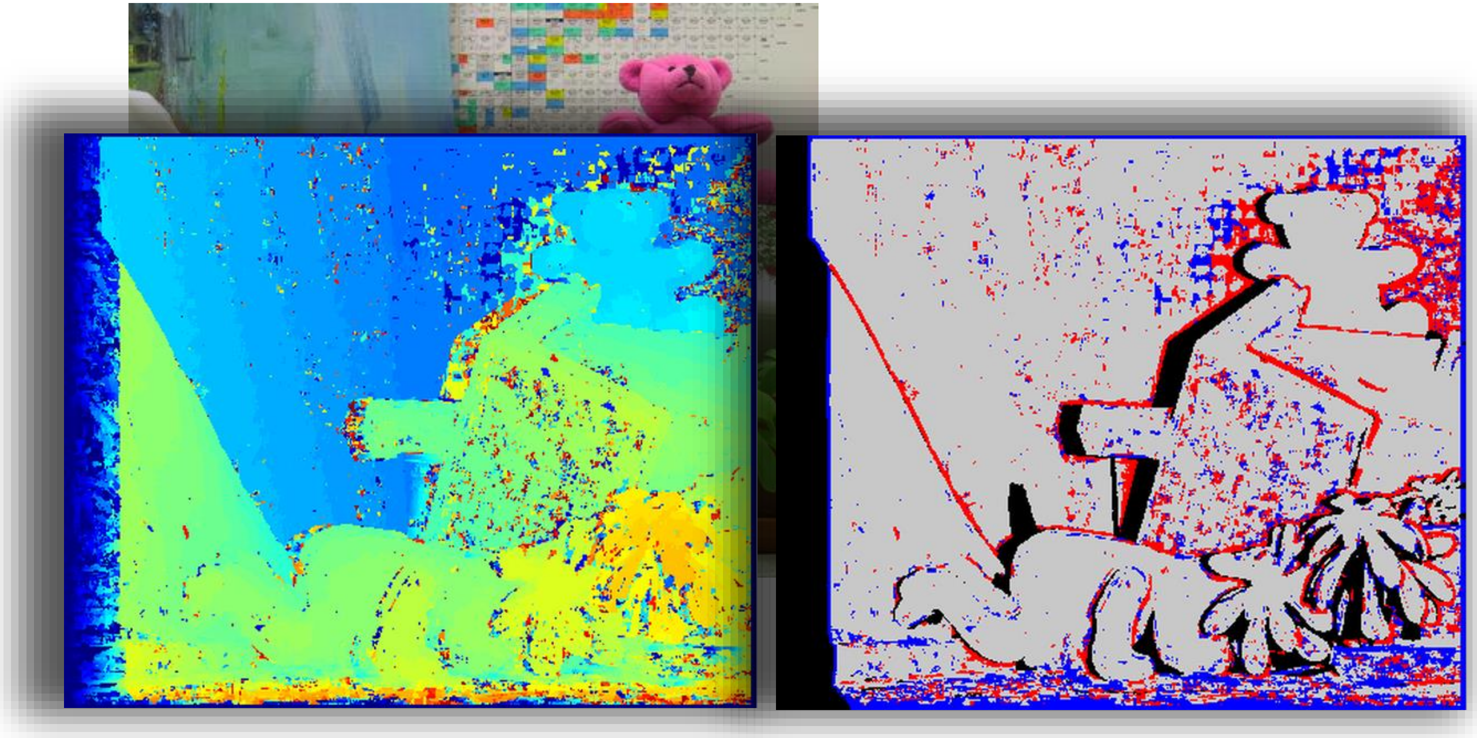
$$C_{ZNCC}(\mathbf{p},\mathbf{d}) = \frac{\sum_{\mathbf{q}\in N_{\mathbf{p}}}(I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}))(I_R(\mathbf{q}-\mathbf{d}) - \bar{I}_R(\mathbf{p}-\mathbf{d}))}{\sqrt{\sum_{\mathbf{q}\in N_{\mathbf{p}}}(I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}))^2 \sum_{\mathbf{q}\in N_{\mathbf{p}}}(I_R(\mathbf{q}-\mathbf{d}) - \bar{I}_R(\mathbf{p}-\mathbf{d}))^2}}$$

Evaluation of Stereo Matching Costs on Images with Radiometric Differences
[Hirschmuller and Scharstein, PAMI 2008]

  - **Descriptor**-based
    - *(hand-crafted features) SIFT, DAISY, …*
    - *(learnt features) Deep learning (revisit later)*

Local Optimization

- Minimize matching cost at each pixel in the left image <u>independently</u>
- Winner-take-all (WTA)



Local Optimization

- Minimize matching cost at each pixel in the left image <u>independently</u>
- Winner-take-all (WTA)
- Adaptive support weights

Image Patch

Adaptive Weights

Locally Adaptive Support-Weight Approach for Visual Correspondence Search
[Yoon and Kweon, CVPR 2005]

Local evidence not enough …

- Photometric Variations
- Fore-shortening
- Reflections
- Transparent surfaces
- Texture-less Areas
- Non-Lambertian Surfaces
- Repetitive patterns
- Complex Occlusions

(Image Source: Lectures on stereo matching, Christian Unger and Nassir Navab, TU Munchen)
http://campar.in.tum.de/twiki/pub/Chair/TeachingWs09Cv2/3D_CV2_WS_2009_Stereo.pdf

Local evidence not enough …

- Photometric Variations
- Fore-shortening ➡
- Reflections
- Transparent surfaces
- Texture-less Areas
- Non-Lambertian Surfaces
- Repetitive patterns
- Complex Occlusions

(Image Source: Lectures on stereo matching, Christian Unger and Nassir Navab, TU Munchen)
http://campar.in.tum.de/twiki/pub/Chair/TeachingWs09Cv2/3D_CV2_WS_2009_Stereo.pdf

Local evidence not enough …

- Photometric Variations
- Fore-shortening
- Reflections
- Transparent surfaces
- Texture-less Areas
- Non-Lambertian Surfaces
- Repetitive patterns
- Complex Occlusions

(Image Source: Lectures on stereo matching, Christian Unger and Nassir Navab, TU Munchen)
http://campar.in.tum.de/twiki/pub/Chair/TeachingWs09Cv2/3D_CV2_WS_2009_Stereo.pdf

Local evidence not enough …

- Photometric Variations
- Fore-shortening
- Reflections ➡
- Transparent surfaces ➡
- Texture-less Areas
- Non-Lambertian Surfaces
- Repetitive patterns
- Complex Occlusions

(Image Source: Lectures on stereo matching, Christian Unger and Nassir Navab, TU Munchen)
http://campar.in.tum.de/twiki/pub/Chair/TeachingWs09Cv2/3D_CV2_WS_2009_Stereo.pdf

Global Optimization

- Solve for all disparities simultaneously …
- Solve a pixel labeling problem
- Labels are discrete (ordered), $d \ \in L_D$

$$L_{\mathrm{D}} = [d_{min}, d_{max}]$$

- Incorporate regularization into objective

$$E(D) = \mathrm{E}_{\mathrm{data}}(D) + \mathrm{E}_{\mathrm{smooth}}(D)$$

- Data term encodes matching costs
- Smoothness term encodes priors
  - Encourage adjacent pixels to take similar disparities

Global Optimization

- Inference on Markov Random Fields (MRF)
- Minimize Energy Function.

$$E(D) = \mathrm{E}_{\mathrm{data}}(D) + \mathrm{E}_{\mathrm{smooth}}(L)$$
$$= \sum_{p \in I} C_p\left(d_p\right) + \sum_{(p,q) \in N} V_{pq}(d_p, d_q)$$

$C_p\left(d_p\right)$: matching cost term (*tabular representation*)

$V_{pq}(d, d')$: pairwise term (Potts, truncated linear or quadratic ...)

contrast sensitive Potts prefers discontinuity at image edges

Global Optimization

- Exact binary MRFs can be efficiently optimized
  - submodular $V_{pq}(*,*)$: equivalent to finding max-flow on graph
- But, multi-label case is NP-Hard, for suitable $V_{pq}(*,*)$
  - such as, *discontinuity-preserving* Potts model.
- Approximate energy minimization for multi-label MRF
  - Graph cuts [Boykov+ 98, Kolmogorov and Zabih 2002]
    - Alpha-expansion (calls max-flow in inner-loop)
  - Belief Propagation etc. – *(see previous tutorials)*
    - ICCV’07 tutorial (Discrete Optimization in Computer Vision)
    - IPAM’08 workshop (Graph Cuts and Related Discrete or Continuous Optimization Problems)

Semi Global Matching (SGM)

Scanline Optimization (1D)

Minimize

$$E(D) = \sum_{p \in I} C_p\left(d_p\right) + \sum_{(p,q) \in N} V_{pq}(d_p, d_q)$$

- Let the pairwise term be:

$$V(d, d') = \begin{cases} 0 & \text{if } d = d' \\ P_1 & \text{if } |d - d'| = 1 \\ P_2 & \text{if } |d - d'| \geq 2. \end{cases}$$

Scanline Optimization (1D)

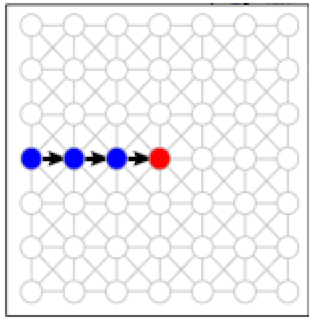
Minimize:

$$E(D) = \sum_{p \in I} C_p\left(d_p\right) + \sum_{(p,q) \in N} V_{pq}(d_p, d_q)$$

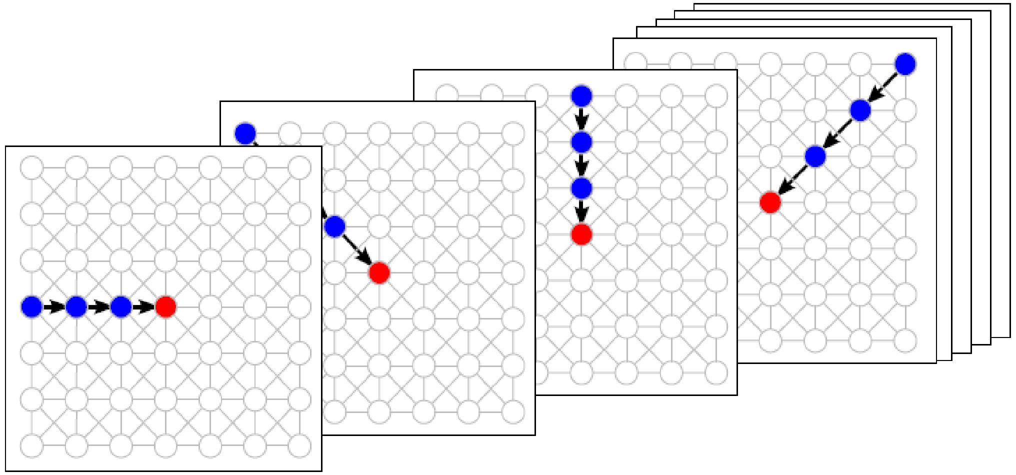
- Consider the above problem on a 1D scanline.
- Compute an aggregated matching cost

$$L_{\mathbf{r}}(\mathbf{p}, d) = C_{\mathbf{p}}(d) + \min_{d' \in \mathcal{D}} (L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d') + V(d, d')).$$

  - $\mathbf{r} = (1, 0)$: start at leftmost pixel, scan left



Semi Global Matching (SGM)



- For 8 directions
  - calculate aggregated costs

$$L_{\mathbf{r}}(\mathbf{p}, d) = C_{\mathbf{p}}(d) + \min_{d' \in \mathcal{D}} (L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d') + V(d, d')).$$

- Finally, sum the costs and select per-pixel minima.

$$S(\mathbf{p}, d) = \sum_{\mathbf{r}} L_{\mathbf{r}}(\mathbf{p}, d)$$

$$D_{\mathbf{p}} = \arg\min_{d} S(\mathbf{p}, d).$$

Efficient Update

$$L_{\mathbf{r}}(\mathbf{p}, d) = C_{\mathbf{p}}(d) + \boxed{\min_{d' \in \mathcal{D}} (L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d') + V(d, d')).}$$

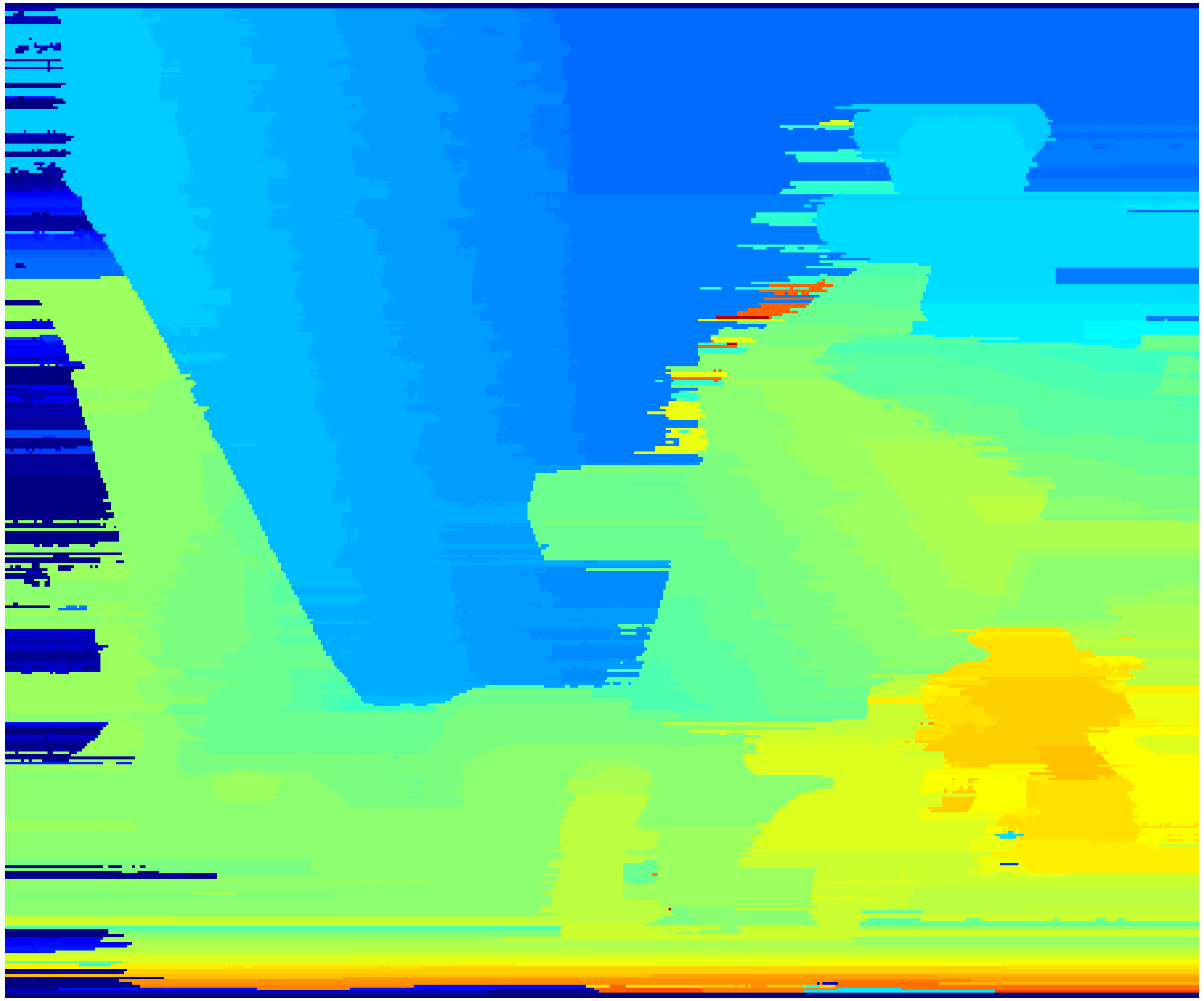
- The minimum can be computed efficiently because $V(d, d')$ has this special form

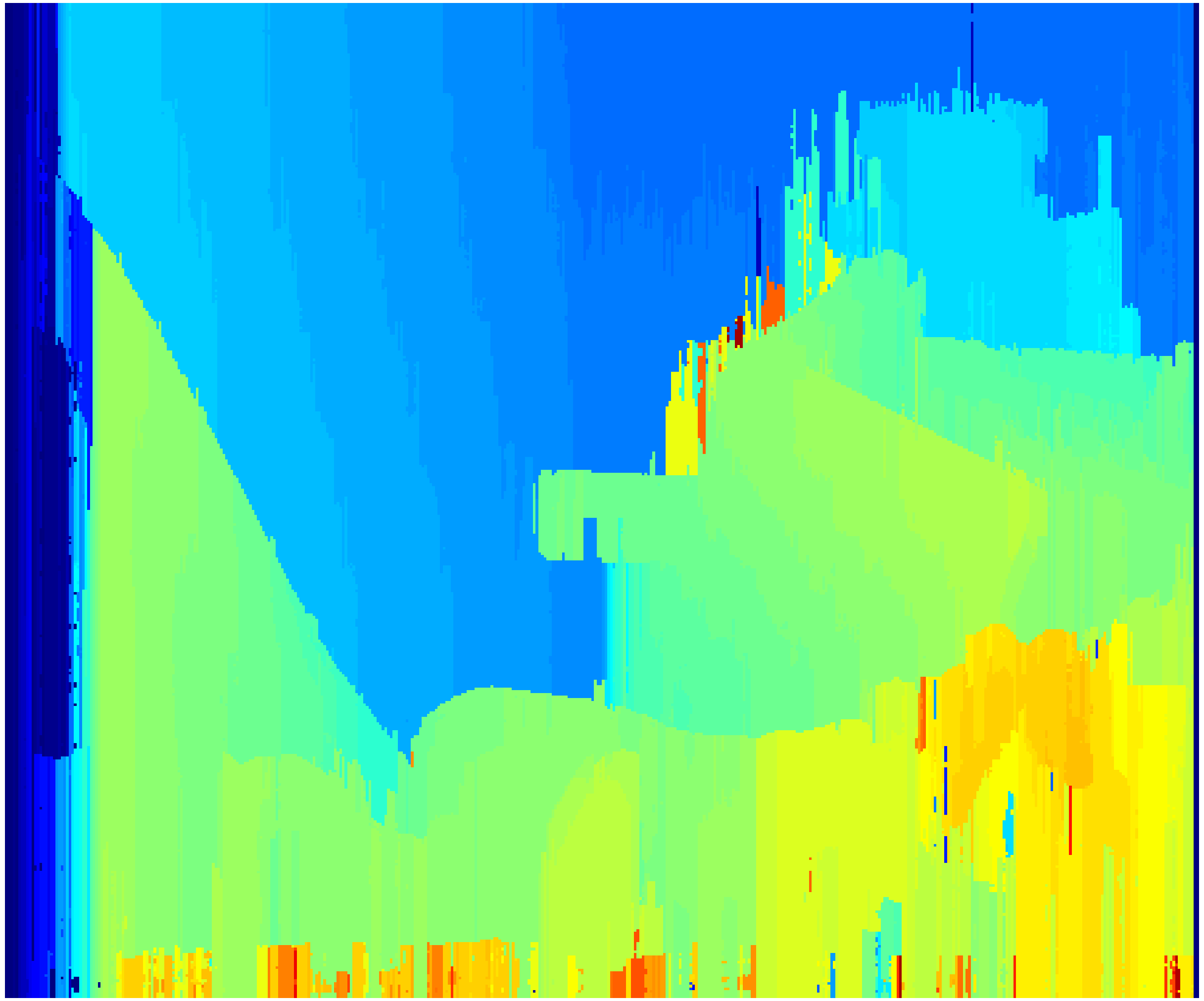
$$V(d, d') = \begin{cases} 0 & \text{if } d = d' \\ P_1 & \text{if } |d - d'| = 1 \\ P_2 & \text{if } |d - d'| \geq 2. \end{cases}$$

- Precompute $\boxed{\min_{d' \in \mathcal{D}} L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d')}$ for previous pixel
  - This term is constant for all disparities d
  - subtract the minimum value
- Then, compute

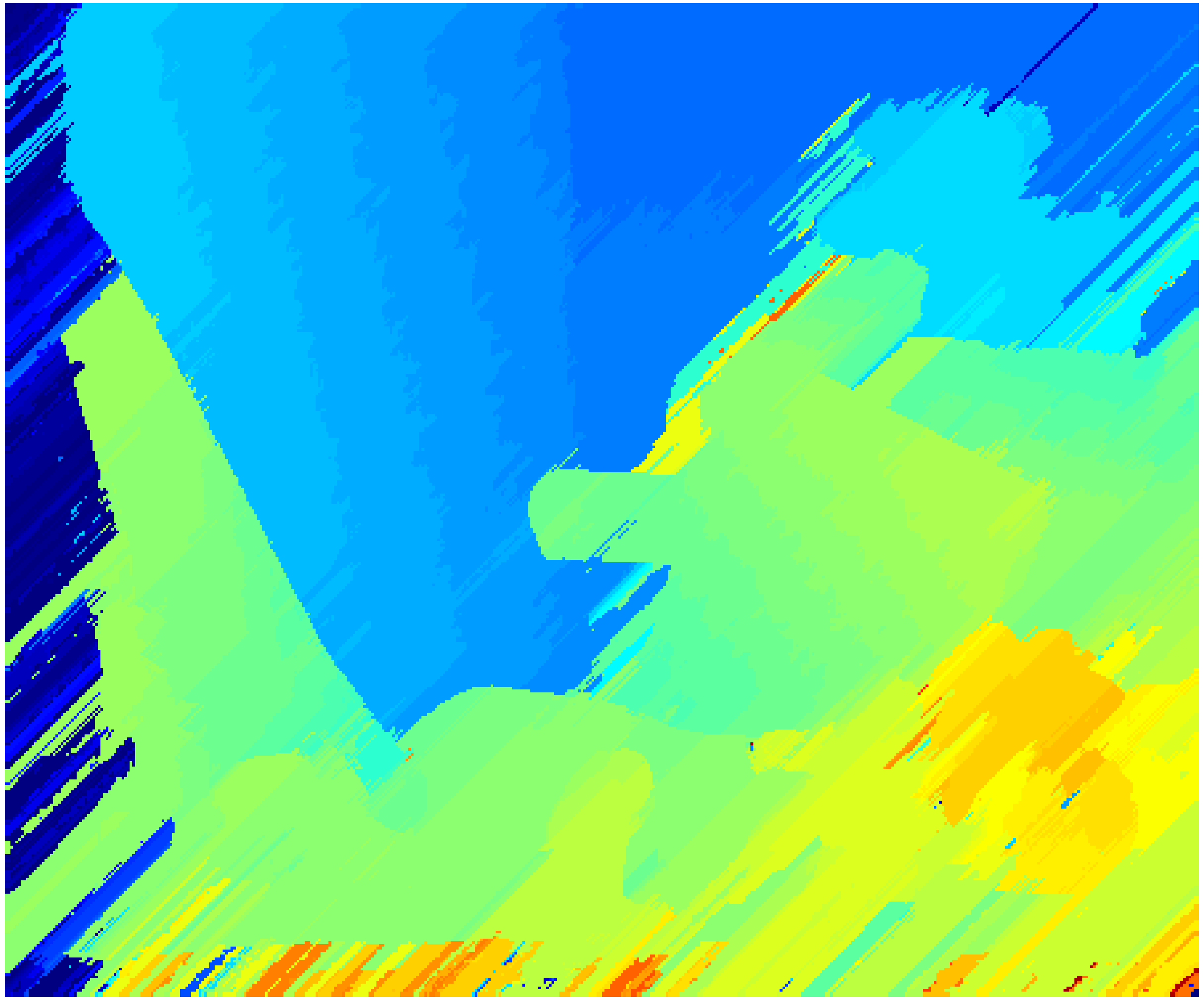
$$L_{\mathbf{r}}(\mathbf{p}, d) = C_{\mathbf{p}}(d) + \min(\underline{P_2}, \underline{L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d)}, \underline{L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d - 1) + P_1}, \underline{L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d + 1) + P_1})$$

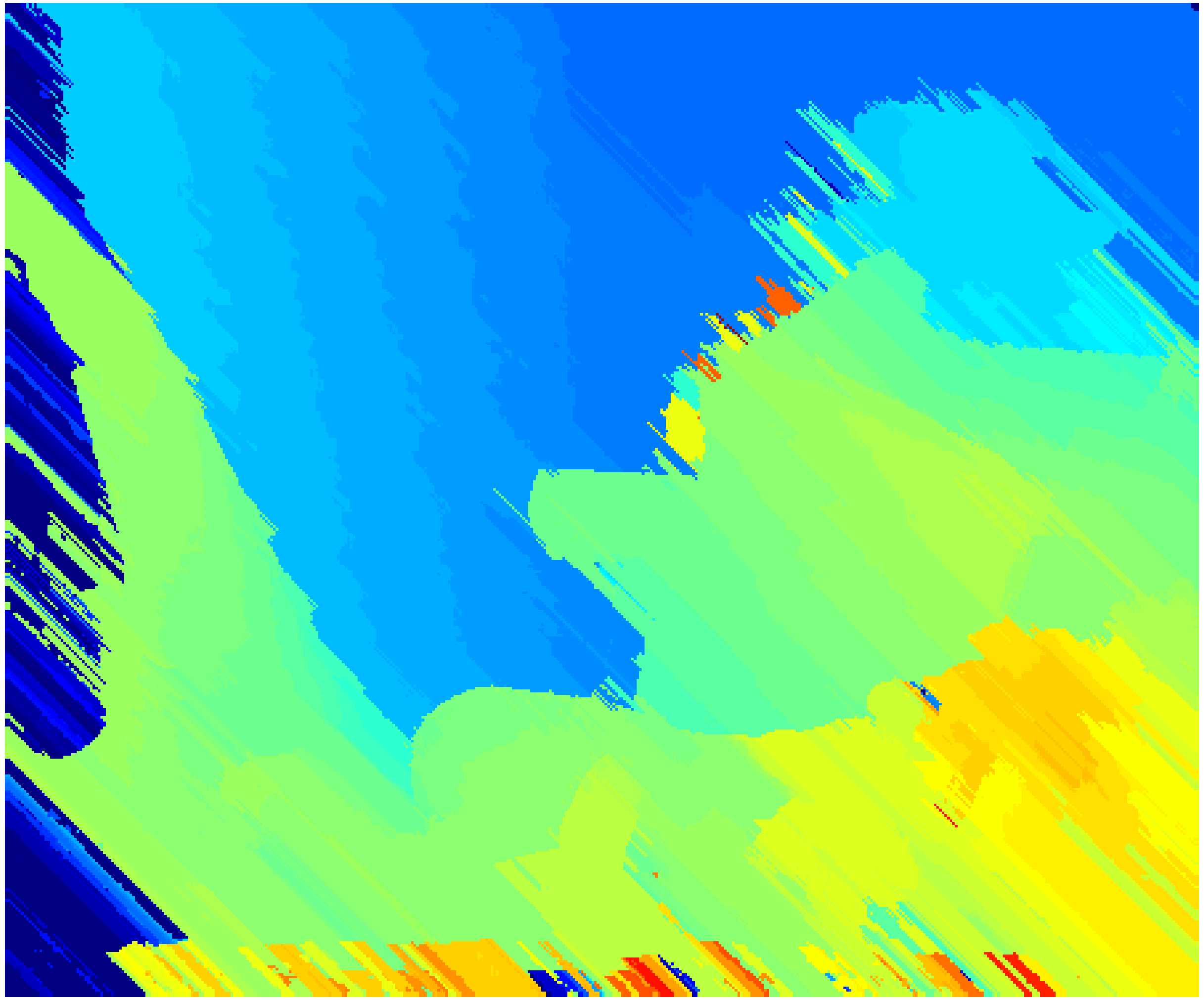
Semi Global Matching (SGM)



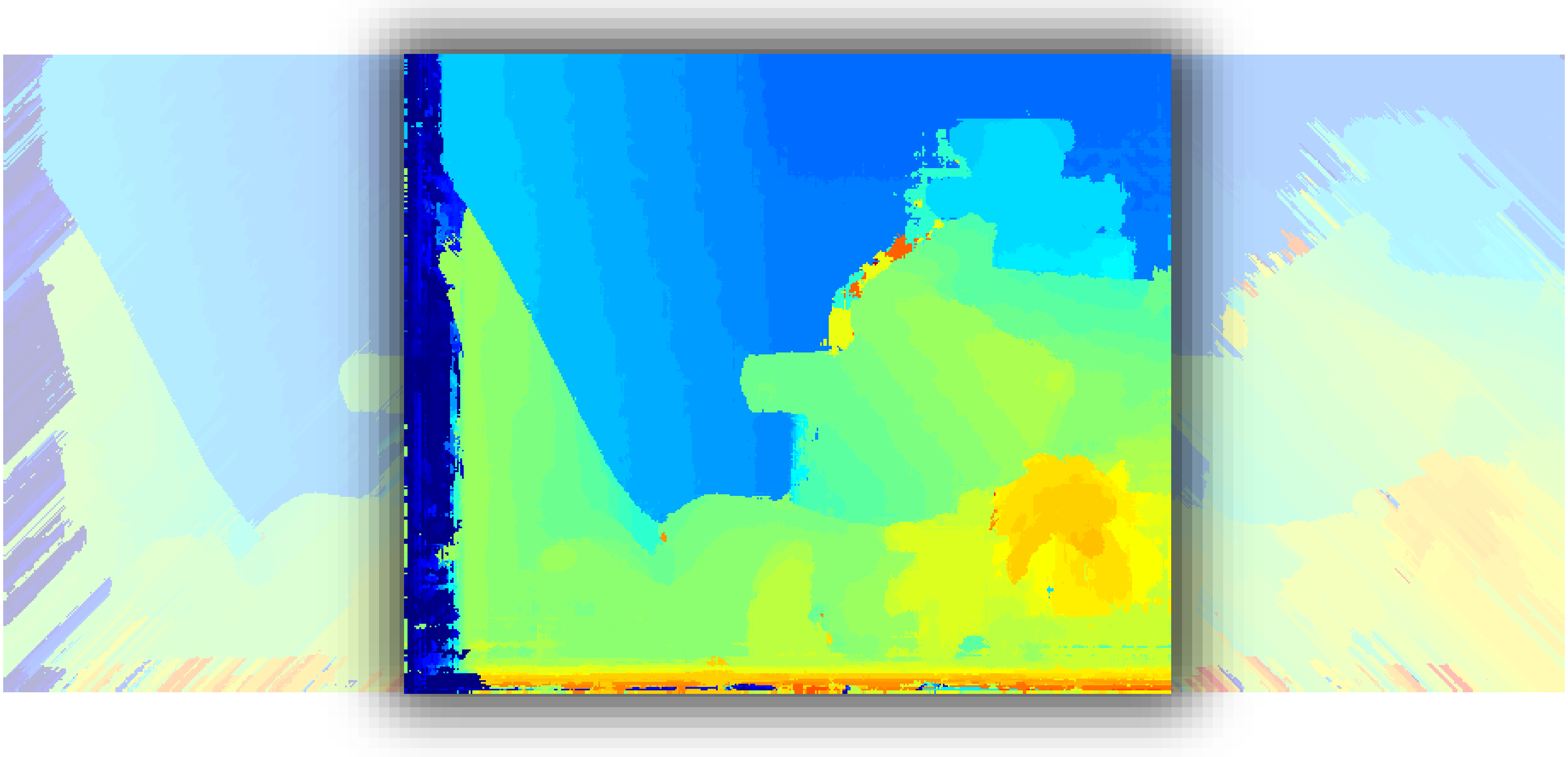


Semi Global Matching (SGM)



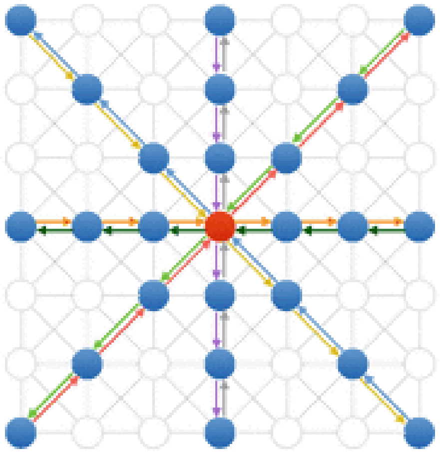


Semi Global Matching (SGM)

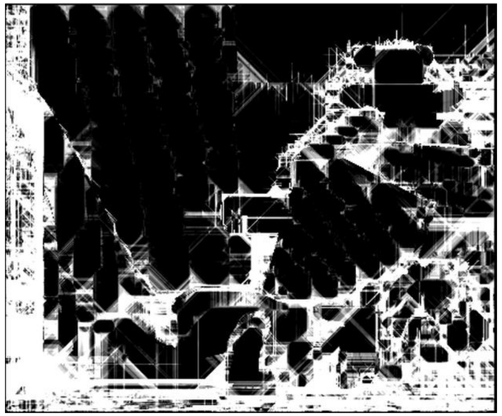


SGM and message passing (BP, TRW-S)

[Drory+ 2014, in Pattern Recognition]



- Insight 1: SGM interpreted as min-sum Belief Propagation on a star shaped subgraph
  - A different subgraph for every pixel.
- Insight 2: SGM's efficient reuse of messages
  - Minor adjustment to aggregated cost gives *min-marginals*
- Also related to tree-reweighted message passing
- Uncertainty measure
  - Gap between minimum of sums and sum of minimums for different directions



Black : low uncertainty

Summary

Pros

- Easy to implement
- Parallelizable
- Fit for real-time, embedded systems (FPGA, GPUs …)
- Related to established message passing techniques

Cons

- Cannot handle slanted weakly textured surfaces
- Fronto-parallel bias
- Somewhat large memory footprint

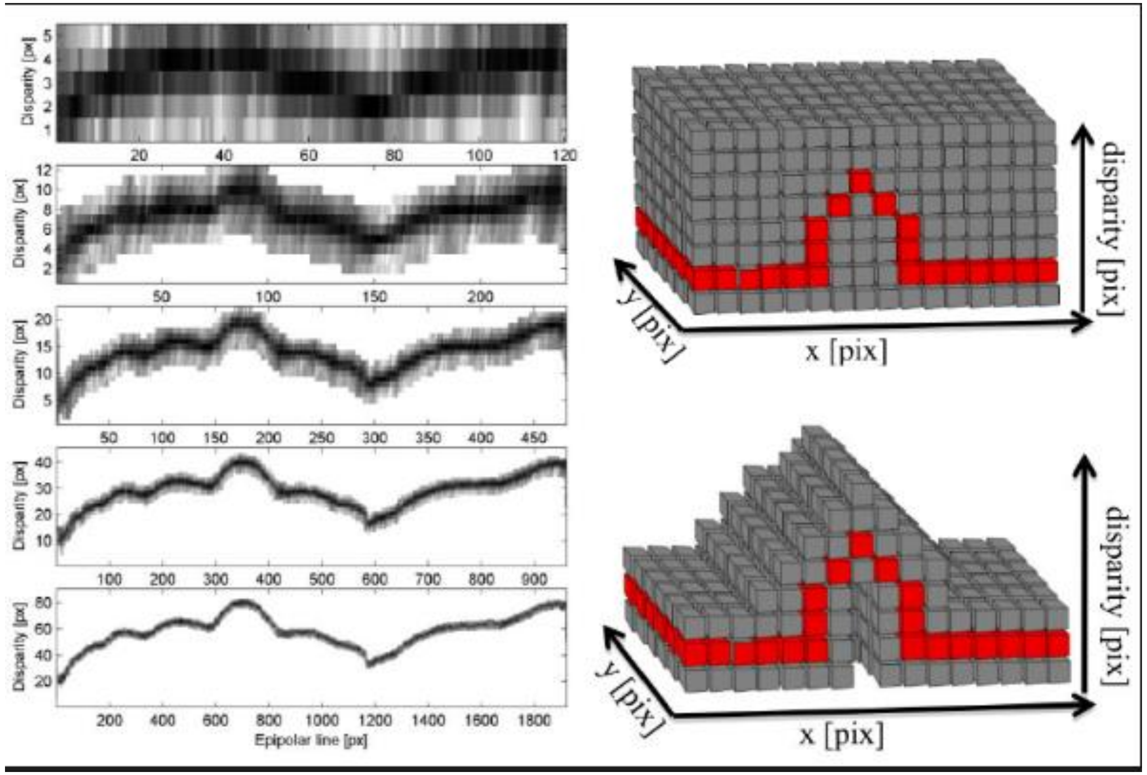
SGM extensions

1. Coarse to fine SGM

- Per-pixel disparity range
  - depth prior
  - interval size can vary

Iterative semi-global matching
for robust driver assistance systems
[Hermann and Klette, ACCV 2012]

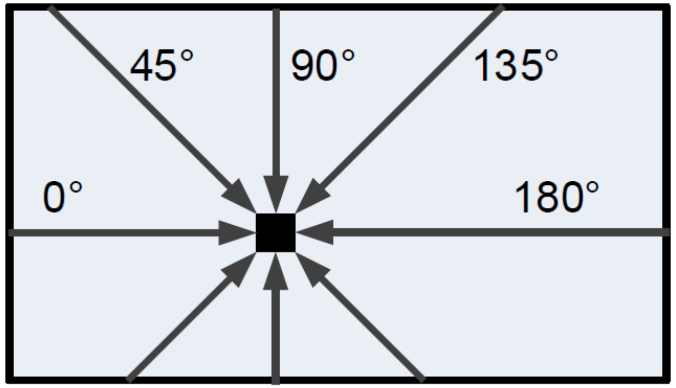
- Per-pixel disparity range
  - coarse to fine strategy
  - interval size is fixed
  - reduces memory footprint



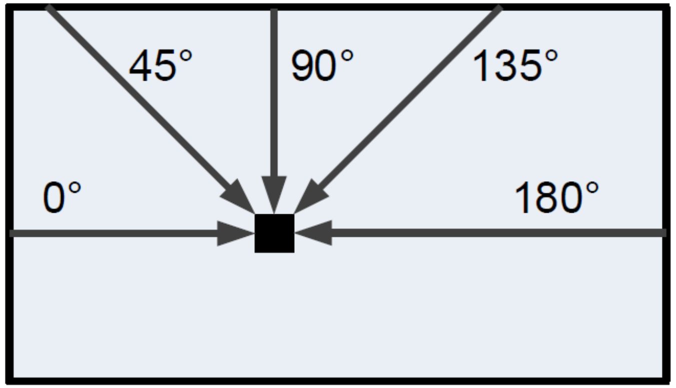
SURE: Photogrammetric Surface Reconstruction from Imagery
[Rothermel+ LC3D workshop]

3. Embedded SGM Stereo

Real-time and Low Latency Embedded Computer Vision Hardware Based on a Combination of FPGA and Mobile CPU
[Honegger, Oleynikova and Pollefeys, IROS 2014]



Normal SGM

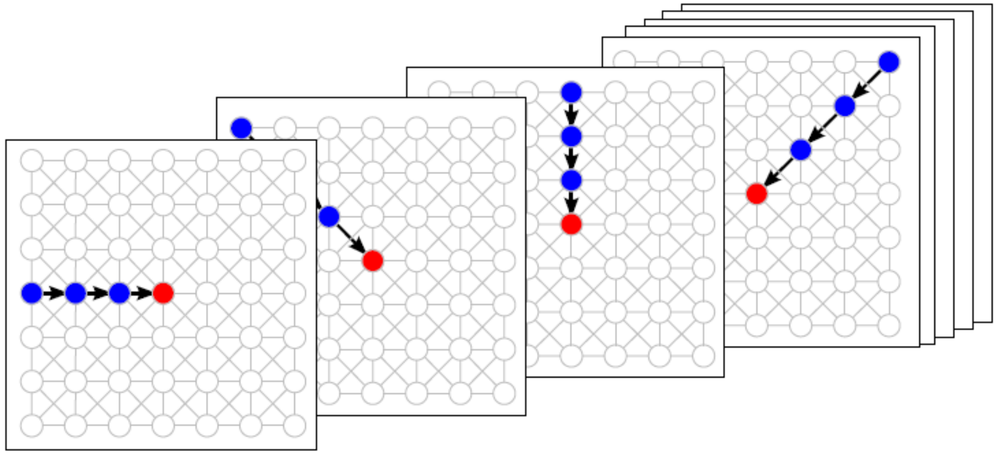


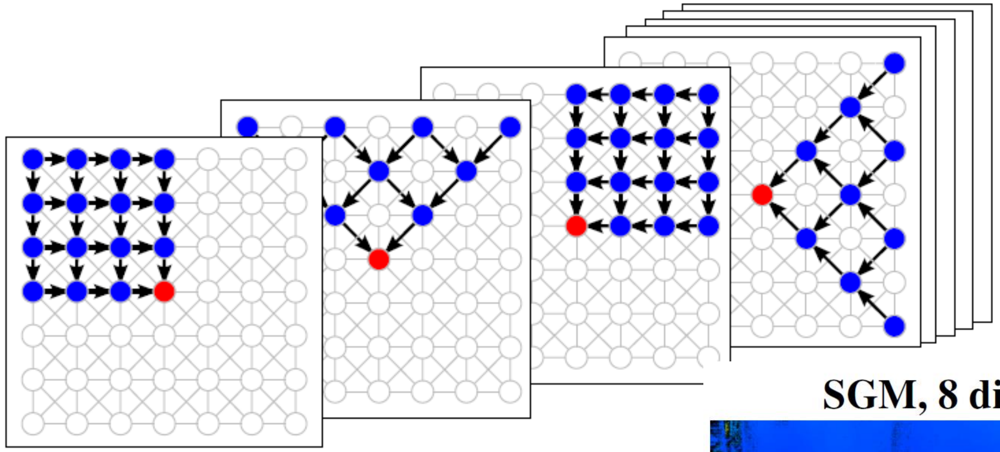
5 paths that avoid bottom to top scan

- Image processed one horizontal scanline at a time
- Low-latency, low-memory footprint
- 60 Hz at 752 x 480 resolution (FPGA for small UAVs and robots)

4. More Global Matching (MGM)

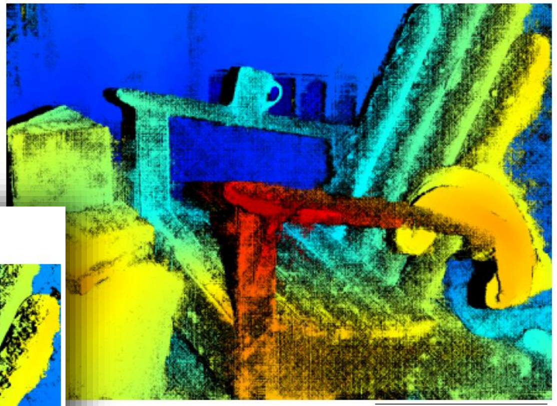
[Facciolo+ BMVC'15]



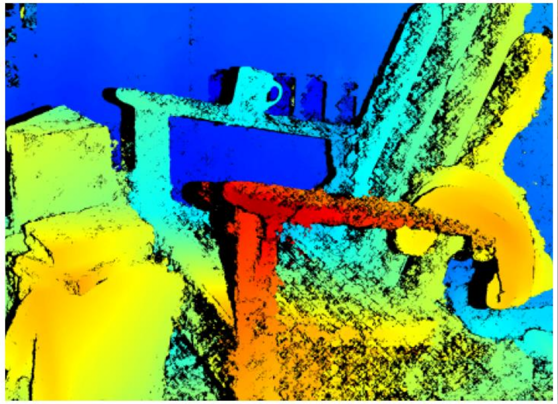


- gather evidence from two directions (quadrant)
  - SGM only uses one direction.
- minor change to SGM recursion (update) step.
- only few extra operations per pixel
- parallelizable

SGM, 8 directions



MGM, 8 directions



Geometric and Semantic Priors for stereo matching

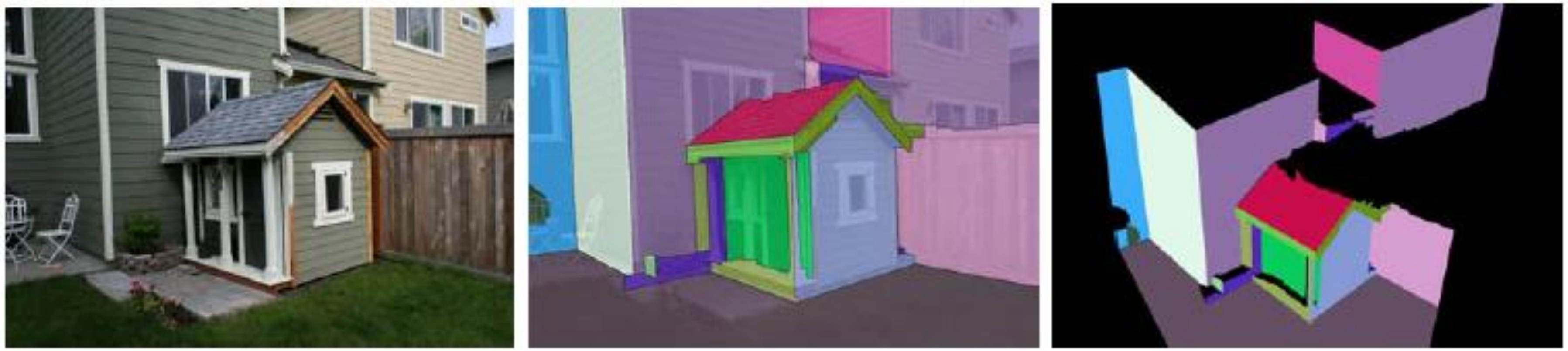
Stereo Matching with Structured Priors

- **Label space: go beyond disparity labels**
  - 3D Planes

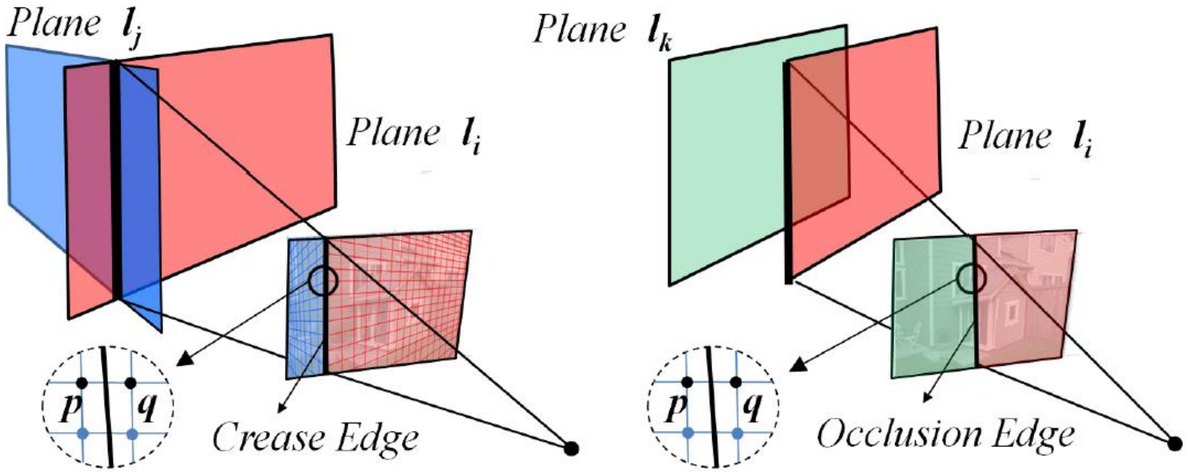
    [Birchfield and Tomasi 2001, Furukawa+ 2009, Sinha+ 2009, Gallup+ 2010]
  - Surfaces [Bleyer+ 2010]
  - 2-Layers [Sinha+ 2012]
- **Joint Stereo and Segmentation**
  - Appearance (color) models [Bleyer+ 2011, Kowdle+ 2012]
  - Semantic Segmentation [Ladicky+ 2010]

Piecewise Planar Stereo

[Sinha+ ICCV'09]

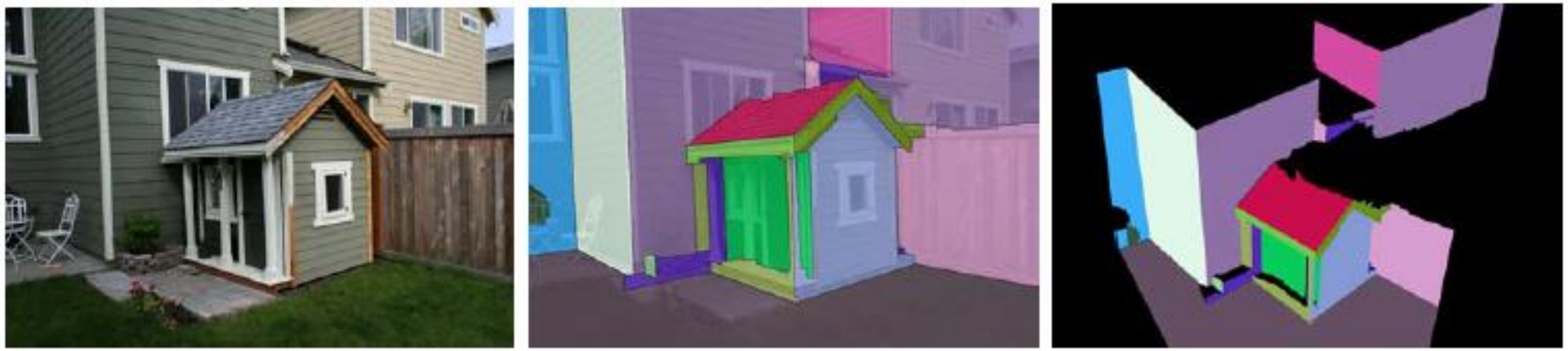


- Label set is a set of unbounded 3d planes: $L = [\pi_1, \pi_2, \ldots \pi_n]$
- Energy minimization via graph cuts
  - pixel-plane labeling
  - pairwise terms
    - Crease between planes
    - Line segments, vanishing points



Piecewise Planar Stereo

[Sinha+ ICCV'09]



<u>Pros</u>

- Piecewise planar bias good for urban scene
- Label-specific, spatially-varying smoothness
- Handles slanted planar surfaces
- Crease between planes modeled

<u>Cons</u>

- Not great for general scenes
- Correct plane may be missing
- Unbounded planes costly to evaluate

Piecewise Planar Stereo (+ color models)

[Kowdle+ ECCV'12]

SGM Stereo | Plane hypotheses | Plane labels | Depth map

- Run SGM stereo
- Extract planes
- per-plane color model (online learning)
- Pixel-plane labeling via graph-cuts
  - Trade-off stereo and color segmentation cues (unary terms)

Object Stereo

[Bleyer+ CVPR’11]

- Joint Stereo and Segmentation
- For both views, estimate
  - Disparity map
  - Object labeling
- Model
  - Scene has a few objects. Each has a
    - Object color model (GMM)
      - Distribution of pixel colors is *compact*
    - Object surface model (plane + parallax)
      - Pixels lie close to a 3D object plane

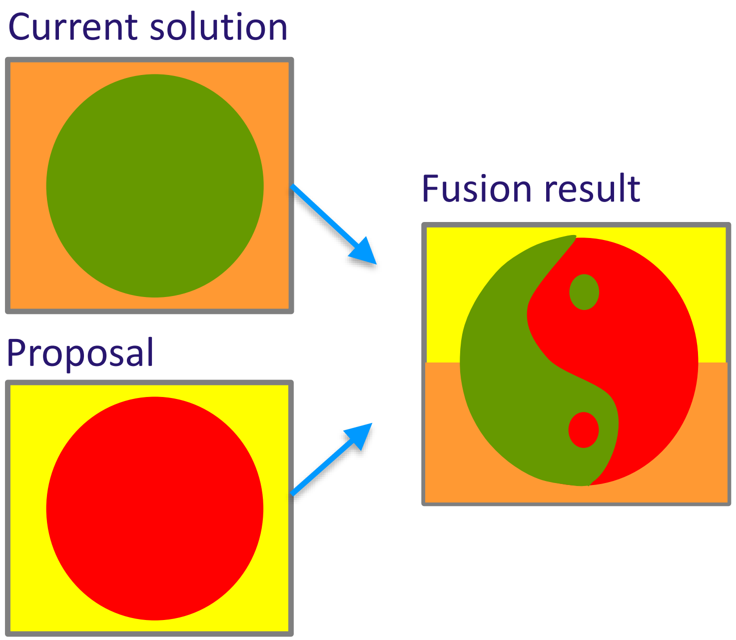
Object Stereo

[Bleyer+ CVPR'11]

Minimize:

$$E(D,O) = E_{photo}(D,O) + E_{smooth-D}(D,O) + E_{smooth-O}(D,O) + E_{mdl}(D,O) + \ \ldots$$

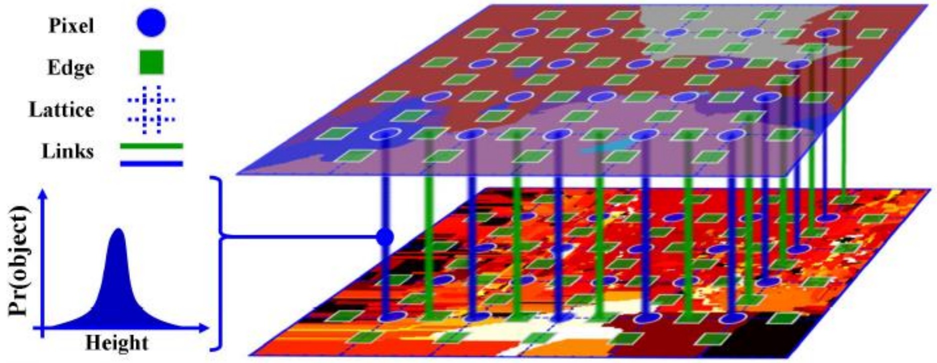
- Proposal generation
- Merge proposals optimally
  - MRF Fusion moves
  - Quadratic Pseudo Boolean Optimization
  - non-submodular Graph Cuts



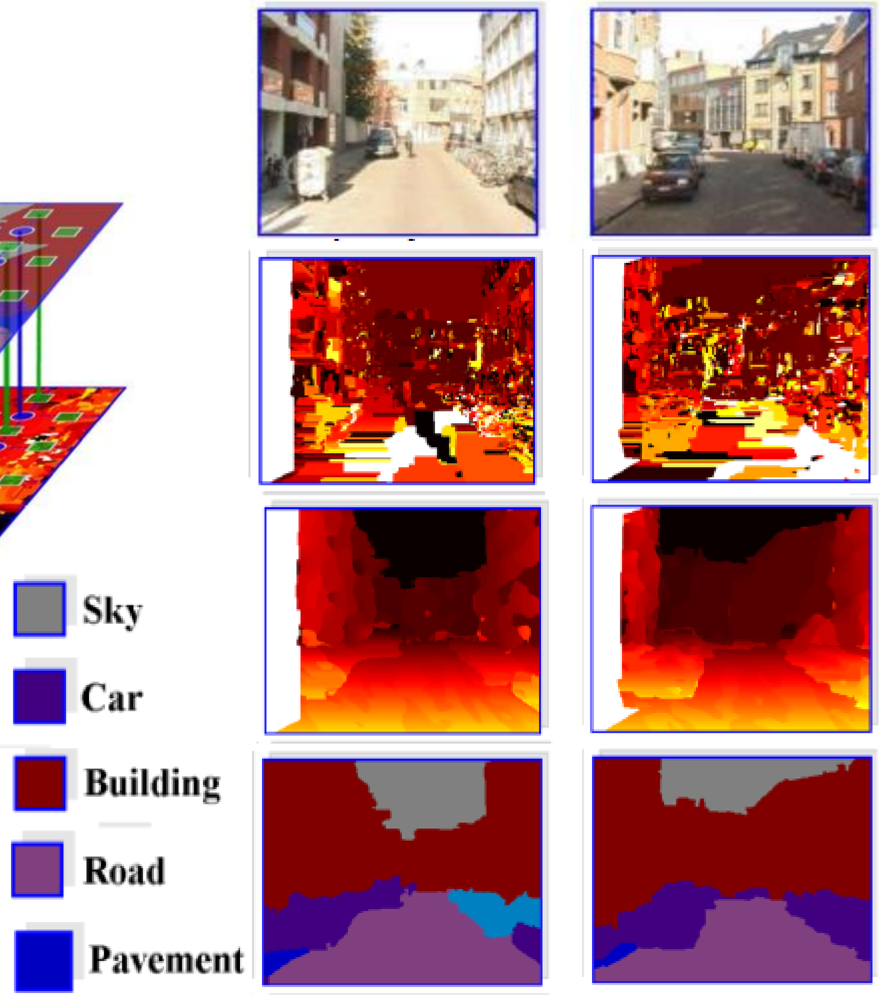
Joint Stereo and Semantic Segmentation

[Ladicky+ BMVC 2010]

- Object class and depth are mutually informative
- Each pixel takes label $z = (d, c) \in L_{Depth} \times L_{Obj}$



- Energy function:
  - Unary: wt. sum of likelihoods (class label, depth)
  - Pairwise: depth transition at object label boundary
  - Higher Order: consistency of superpixels ..
  - Optimized using graph cuts

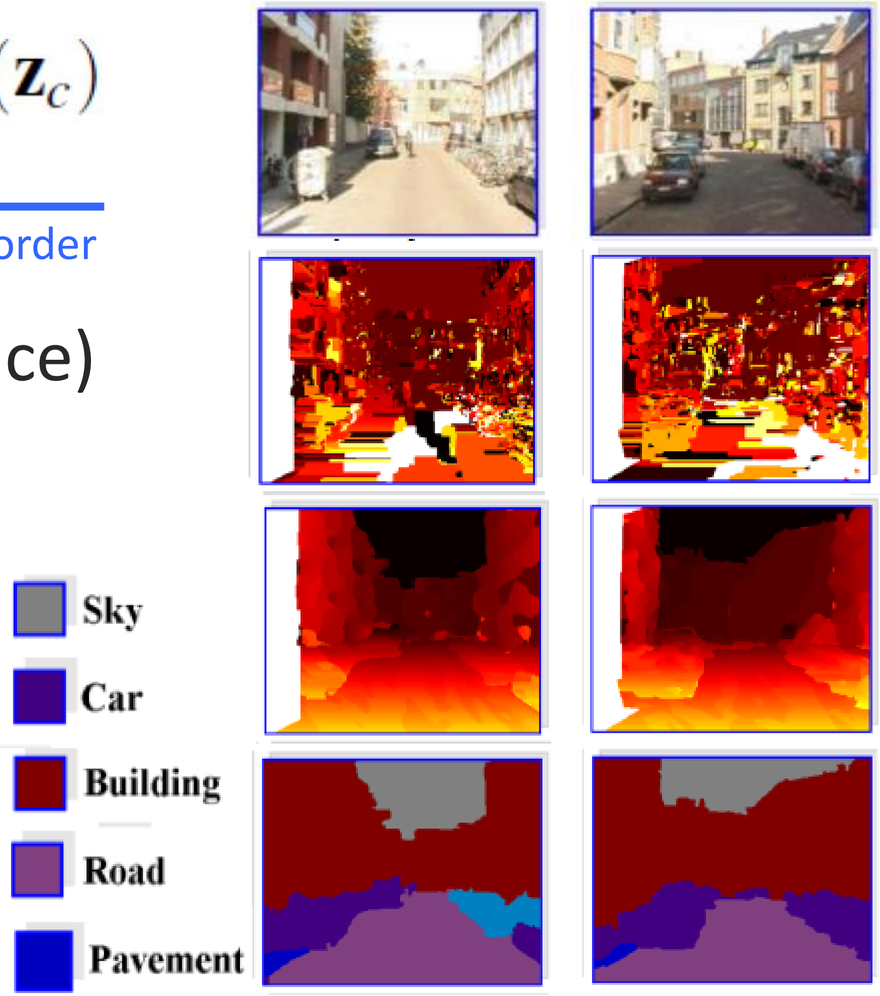


Joint Stereo and Semantic Segmentation

[Ladicky+ BMVC 2010]

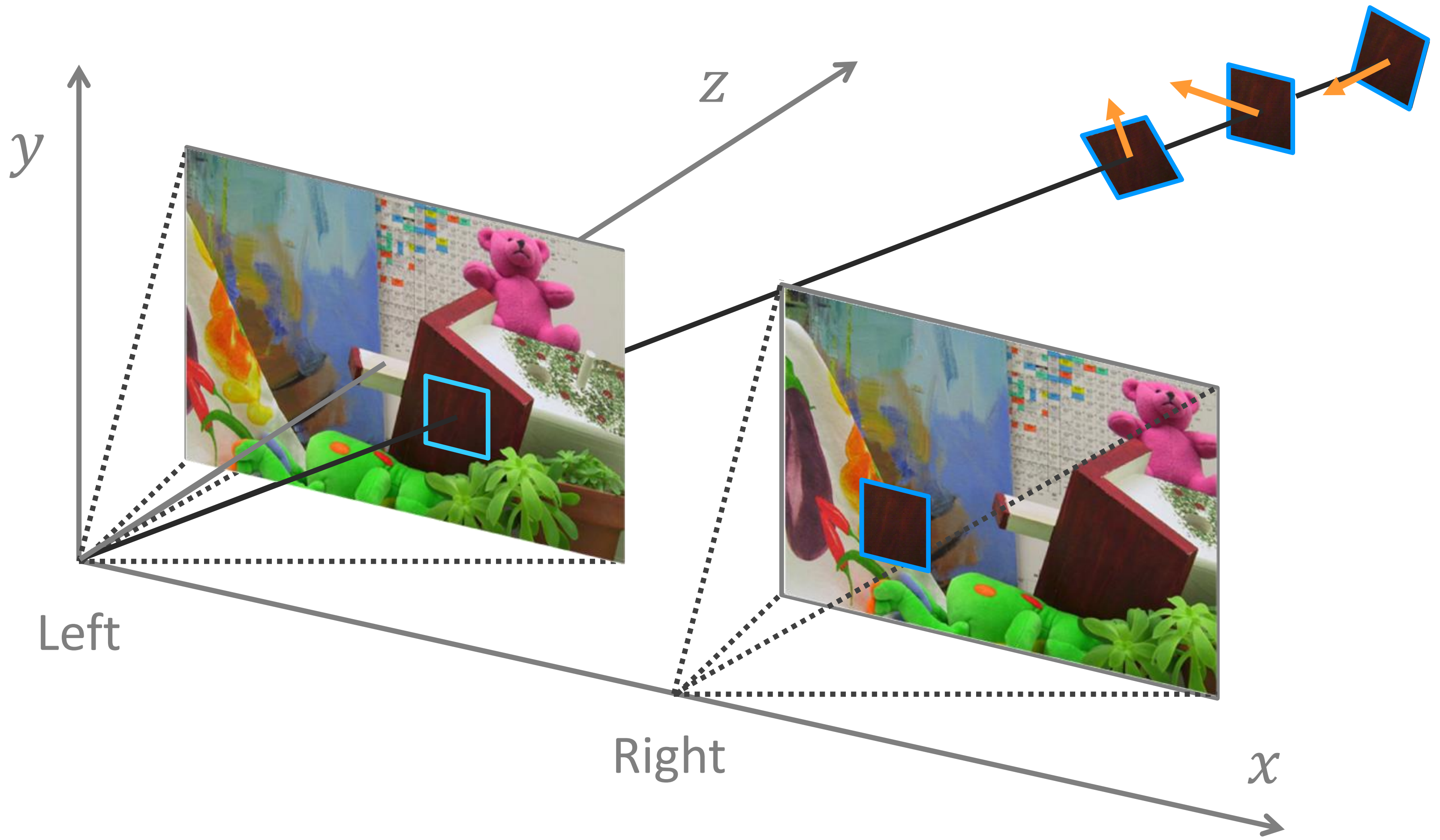
$$E(\mathbf{z}) = \underbrace{\sum_{i \in \mathcal{V}} \psi_i^J(z_i)}_{\text{unary}} + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \underbrace{\psi_{ij}^J(z_i, z_j)}_{\text{pairwise}} + \underbrace{\sum_{c \in \mathcal{C}} \psi_c^J(\mathbf{z}_c)}_{\text{higher-order}}$$

- Alpha expansion on label pairs (in product space)
  - Too many labels, slow ..
- Projected expansion move
  - Keep one of the two label components fixed
  - Expansion move in object class projection
  - Expansion move in depth projection



Continuous Stereo

3D Label Stereo



- Estimate per-pixel 3D tangent planes (depth z + normal n)
- Infinite and continuous label space

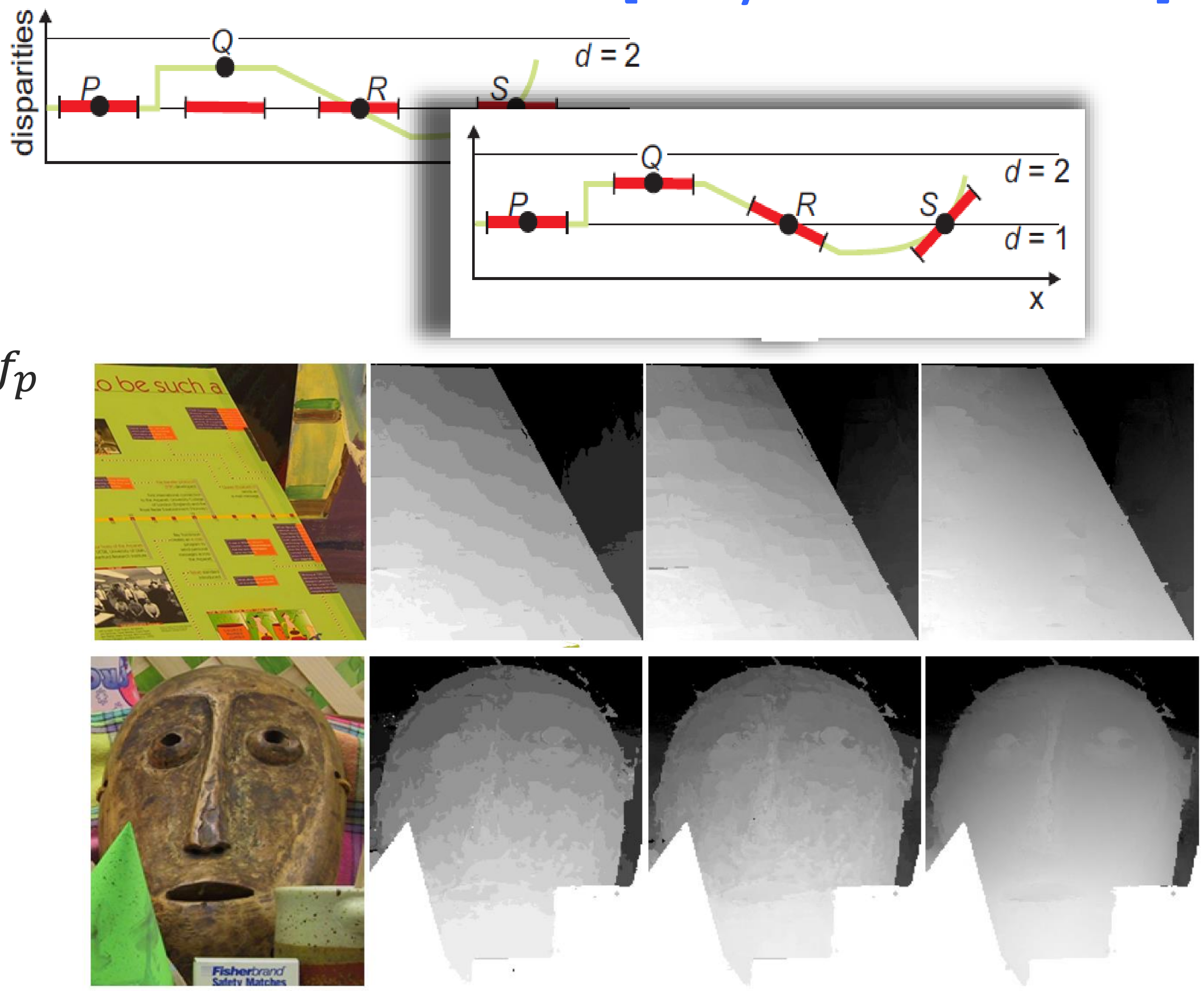
1. PatchMatch Stereo

[Bleyer+ BMVC'11]

- Representation:
  - slanted disparity plane f_p at pixe
    - Label $(a_{f_p}, b_{f_p}, c_{f_p}) \in R^3$

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

- Matching cost:
  - color and gradient difference
  - Adaptive support weights

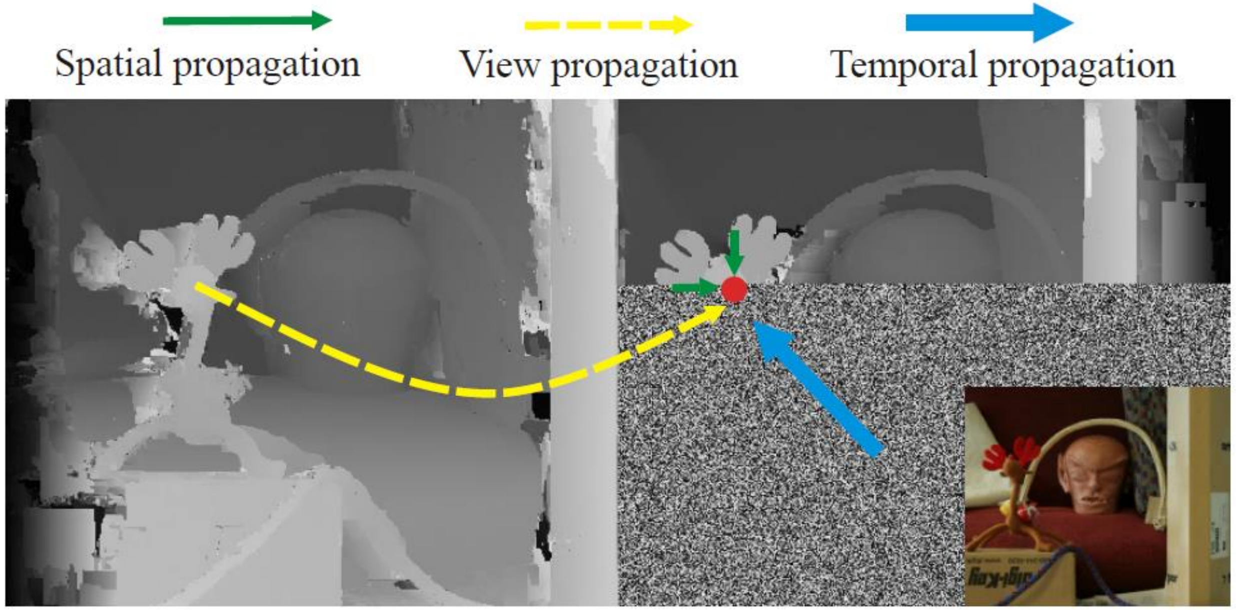


1. PatchMatch Stereo

[Bleyer+ BMVC’11]

Inference via PatchMatch [Barnes+ 2009]

- Randomly initialize disparity planes
- At each iteration
  - Propagate disparity labels
    - from neighbors
    - from other view
  - If cost decreases, accept
  - Re-fit planes
- Regularization added
  - PatchMatch BP [Besse+ 2012], Local Expansion Move [Taniai+ 2014]

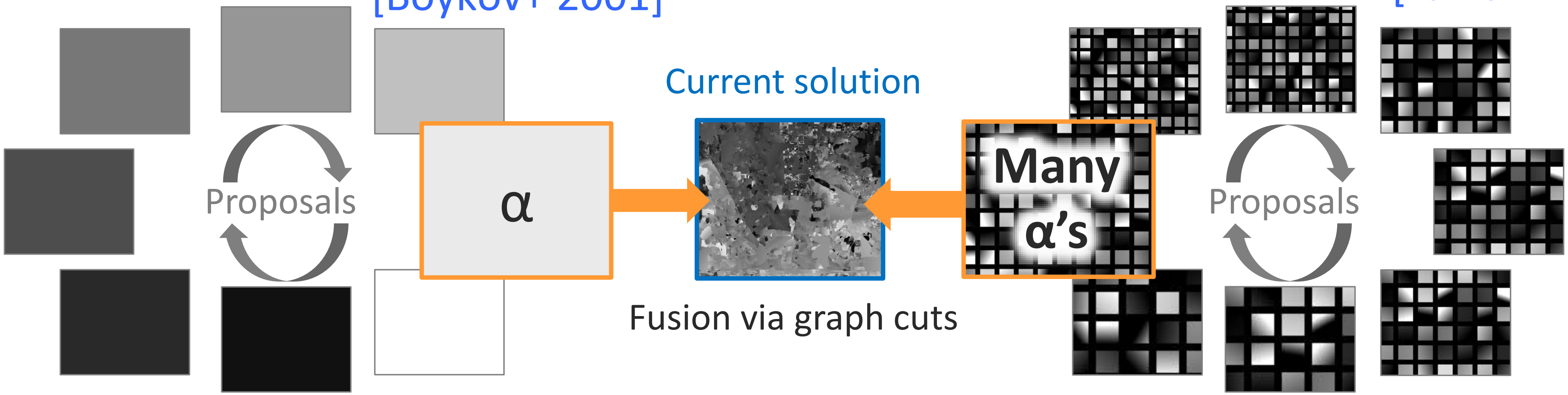


2. Local Expansion Moves

Continuous Stereo Matching using Local Expansion Moves
Taniai + 2017 (arxiv, TPAMI sub)

Traditional α-expansions [Boykov+ 2001]

Local α-expansions [Taniai+ 2014]

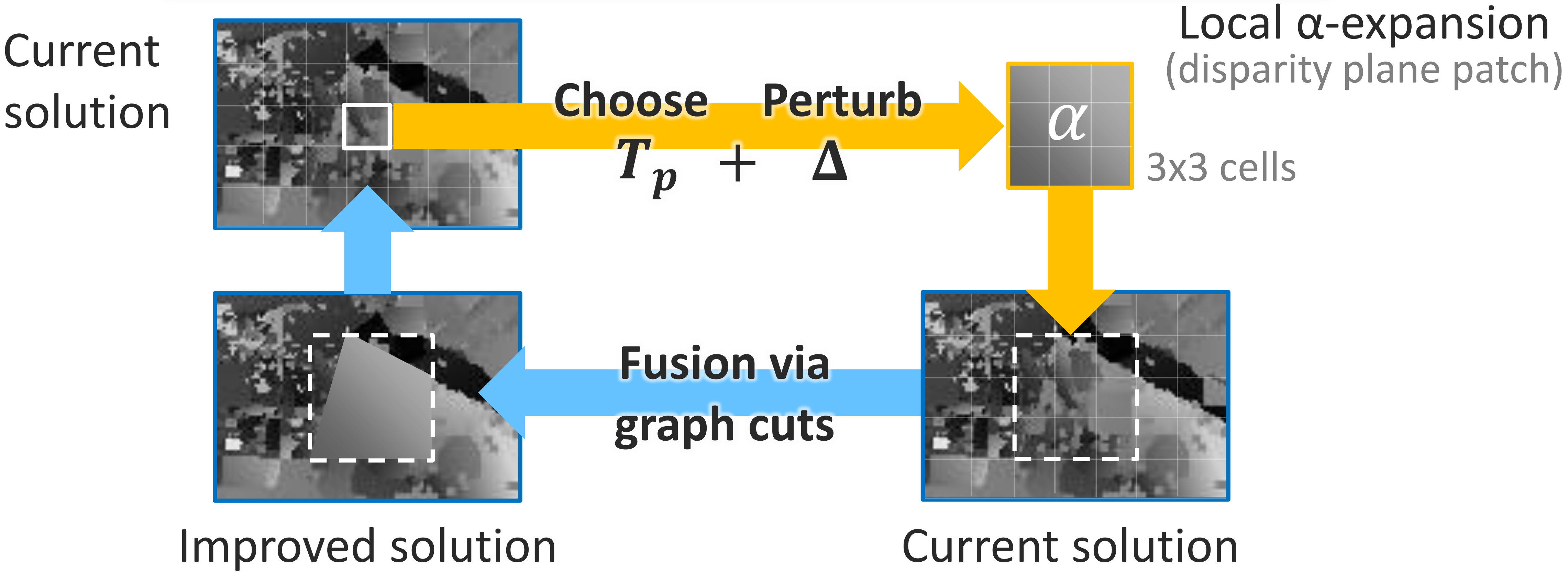


☹ Intractable due to the infinite label space

☺ Spatially localized label-space searching

2. Local Expansion Moves

Continuous Stereo Matching using Local Expansion Moves
Taniai + 2017 (arxiv, TPAMI sub)

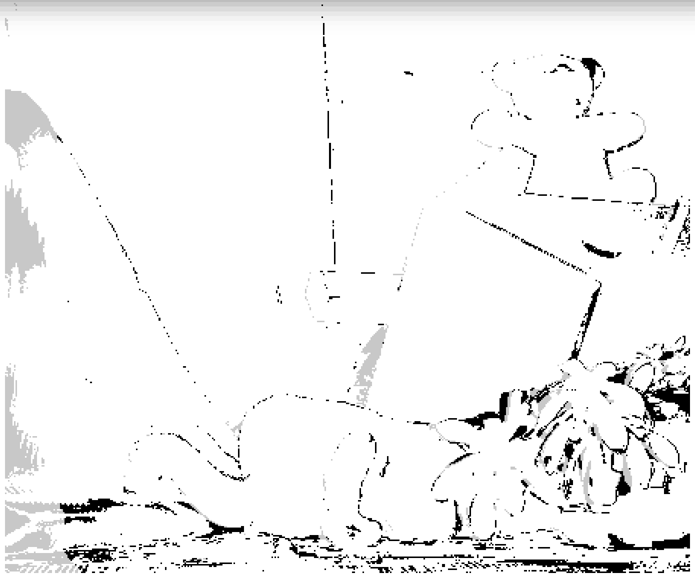


Propagation and randomized search like PatchMatch [Barnes+ ToG '09]

2. Local Expansion Moves

Continuous Stereo Matching using Local Expansion Moves
Taniai + 2017 (arxiv, TPAMI sub)

Middlebury V3 benchmark
1st rank amongst 64 methods (June 2017)



White: correct
Black: incorrect
Gray: incorrect but occluded

Error map

Ranking of all methods using MC-CNN [Zbontar and LeCun, 2016]

bad 2.0 (%) Name	Res	Weight Avg	Austr	AustrP	Bicyc2	Class	ClassE	Compu	Crusa	CrusaP	Djemb	DjembL	Hoops	Livgrm	Nkuba	Plants	Stairs
LocalExp (ours)	H	**5.43** 1	3.65 2	2.87 3	**2.98** 1	**1.99** 1	**5.59** 1	**3.37** 1	3.48 2	**3.35** 1	**2.05** 1	10.3 2	9.75 2	8.57 4	14.4 8	5.40 3	9.55 5
3DMST [26]	H	5.92 2	3.71 3	2.78 2	4.75 2	2.72 4	7.36 4	4.28 2	**3.44** 1	3.76 2	2.35 2	12.6 5	11.5 4	8.56 3	14.0 7	5.35 2	8.87 4
MC-CNN+TDSR [10]	F	6.35 3	5.45 8	4.45 12	6.80 13	3.46 10	10.7 10	6.05 7	5.01 7	5.19 8	2.62 6	10.8 3	**9.62** 1	**6.59** 1	**11.4** 1	6.01 6	**7.04** 1
PMSC [27]	H	6.71 4	**3.46** 1	**2.68** 1	6.19 9	2.54 2	6.92 2	4.54 3	3.96 3	4.04 4	2.37 3	13.1 7	12.3 5	12.2 6	16.2 13	5.88 5	10.8 8
NTDE [18]	H	7.44 8	5.72 12	4.36 11	5.92 7	2.83 5	10.4 7	5.71 5	5.30 8	5.54 9	2.40 4	13.5 8	14.1 9	12.6 8	13.9 6	6.39 8	12.2 13
MC-CNN-acrt [46]	H	8.08 9	5.59 11	4.55 15	5.96 8	2.83 5	11.4 14	5.81 6	8.32 12	8.89 16	2.71 7	16.3 12	14.1 10	13.2 10	13.0 3	6.40 9	11.1 10

Deep Learning in Stereo

Learning the Matching Cost

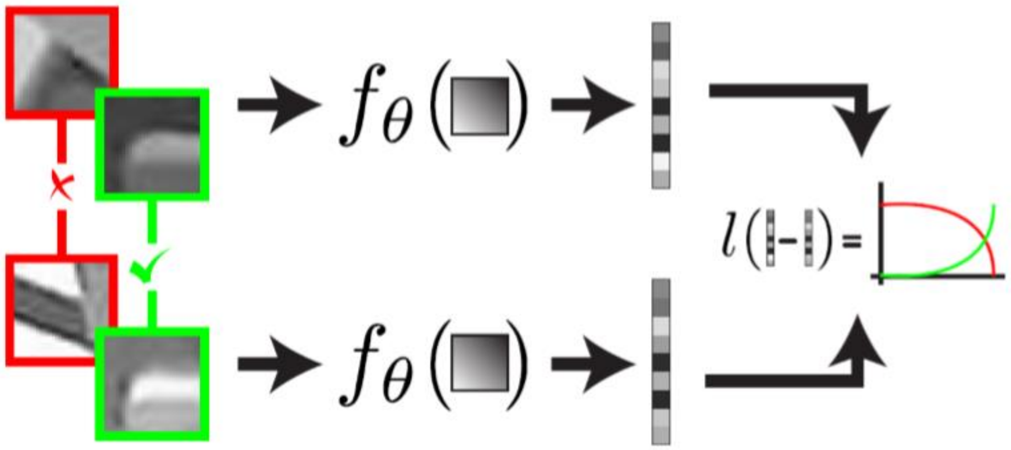
Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches
Zbontar and Lecun [CVPR 2015] [JMLR 2016]

- ConvNet compares two patches and predicts *true* vs. *false* match
- produces the disparity space image (DSI)
- trained on patches extracted from stereo ground truth
  - Positive pairs sampled directly from disparity maps
  - Negative pairs sampled with moderate perturbation

- Stereo Matching
  - Cross-based Cost Aggregation [Mei+ 2011]
  - Semi-Global Matching (SGM)

Local Feature Learning using Siamese Networks

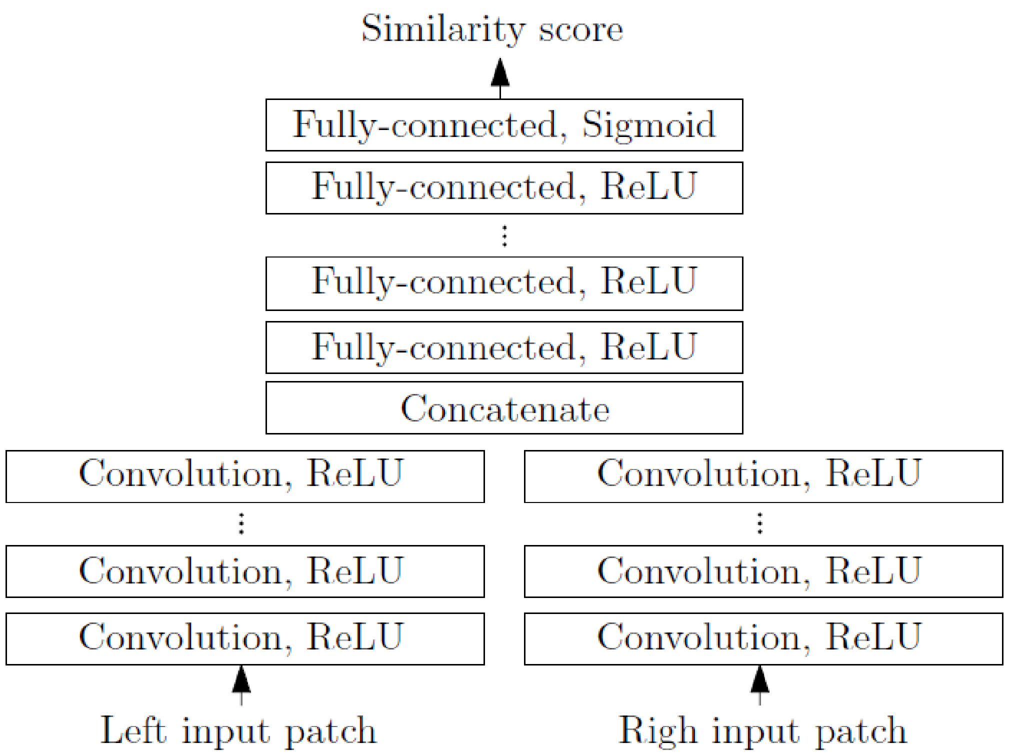
- Verification Tasks [Bromley et al. 1994]
  - Given pairs of entities (faces, signatures, ..),
  - Predict match vs. non-match
- Learning Image Descriptors



- Training Data: Stereo ground truth, CG datasets, Internet photos

Learning the Matching Cost

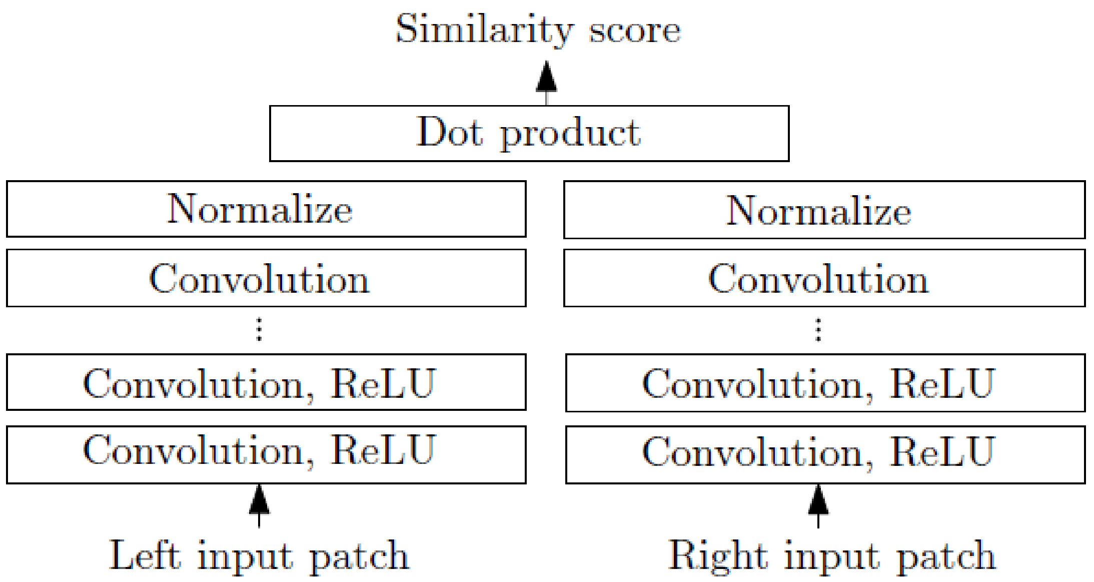
[Zbontar and Lecun JMLR 2016]



Accurate Architecture (MC-CNN acrt) [Siamese + Metric Network]

Learning the Matching Cost

[Zbontar and Lecun JMLR 2016]

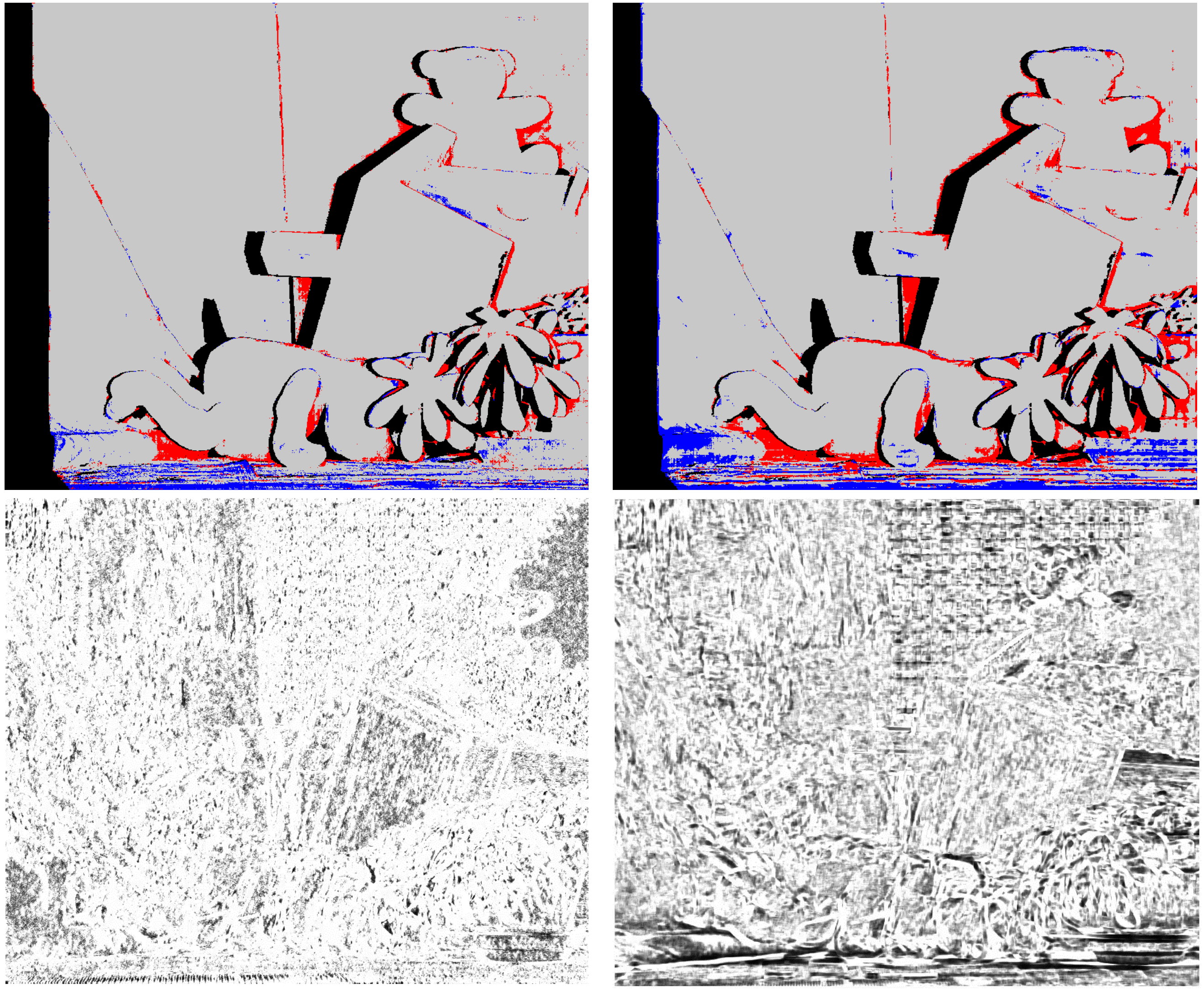


Fast Architecture (MC-CNN fst) [Siamese Network]

Visualizing the DSI (NCC vs MC-CNN-fst)

Error map
(w SGM)

DSI



MC-CNN

NCC 7x7

Advantages of MC-CNN

- discriminates weak, low frequency textures
- Accurate at depth boundaries, slanted surfaces
- Ignores horizontal edges

MC-CNN-acrt *vs.* **MC-CNN-fst**

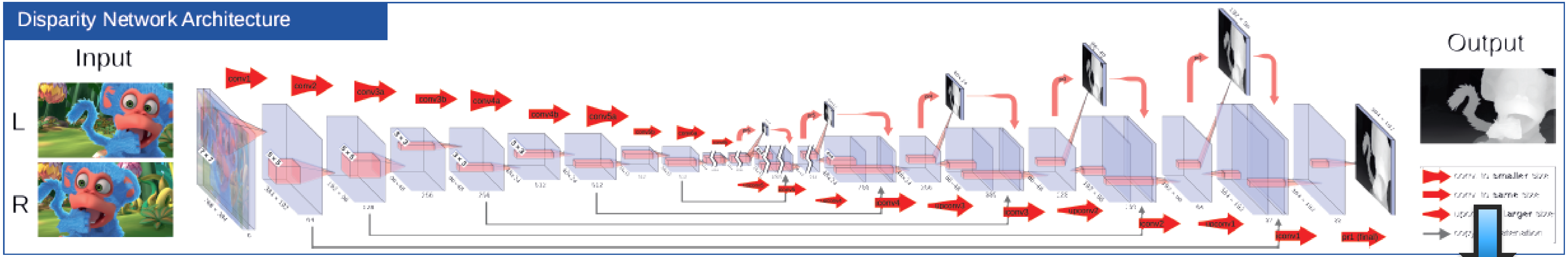
Deep visual correspondence embedding model for stereo matching costs
[Chen+ ICCV 2015]

- Also proposed faster Siamese network architecture
- Combines computation at two scale (full and half resolution)
- Smaller network, 100x faster than MC-CNN-*acrt*

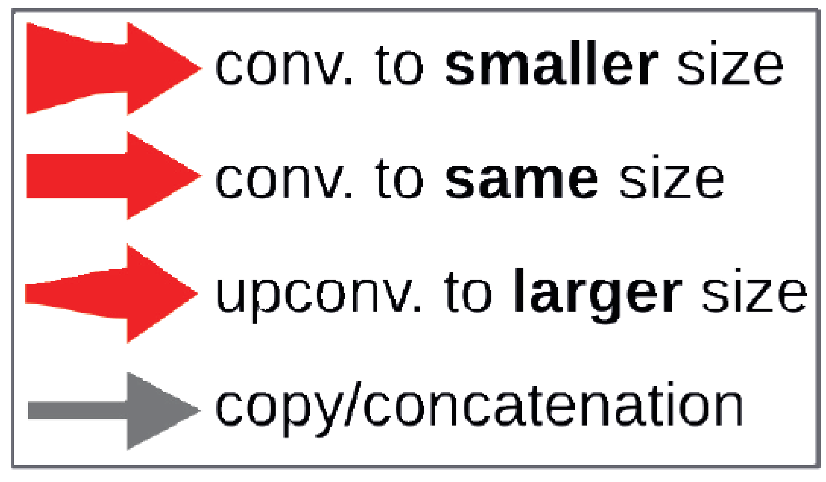
Efficient Deep Learning for Stereo Matching
[Luo+ CVPR 2016]

- Concurrent to [Chen+ 2015, Zbontar+ 2016]
- Tested small Siamese networks
- Multi-class classification loss instead of binary classification loss
- Analyzed receptive field size, showed larger is better

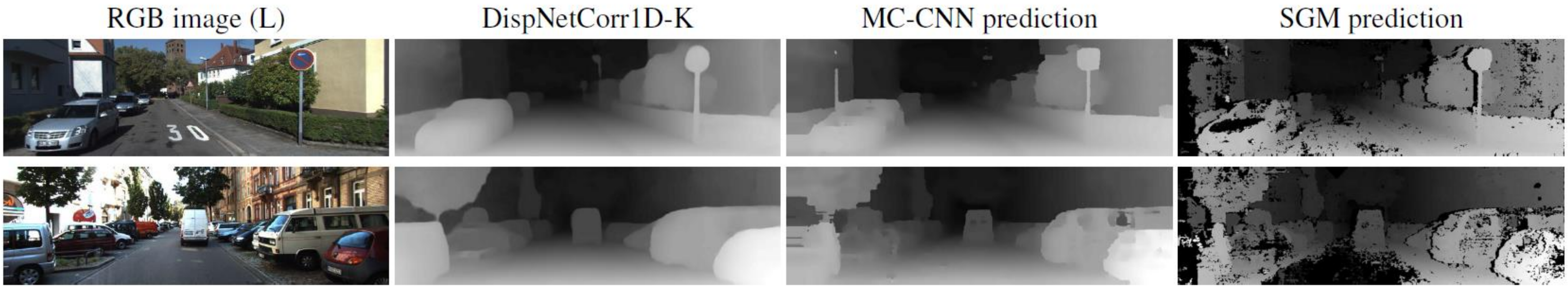
A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation [Mayer+ CVPR 2016]



- Contracting Part: convolutions
- Expanding Part (see FlowNet [Dosovitskiy+ ICCV 2015])
  - Up-convolutions (convolutional transpose)
  - Concatenated with feature maps from contracting part and the predicted disparity maps

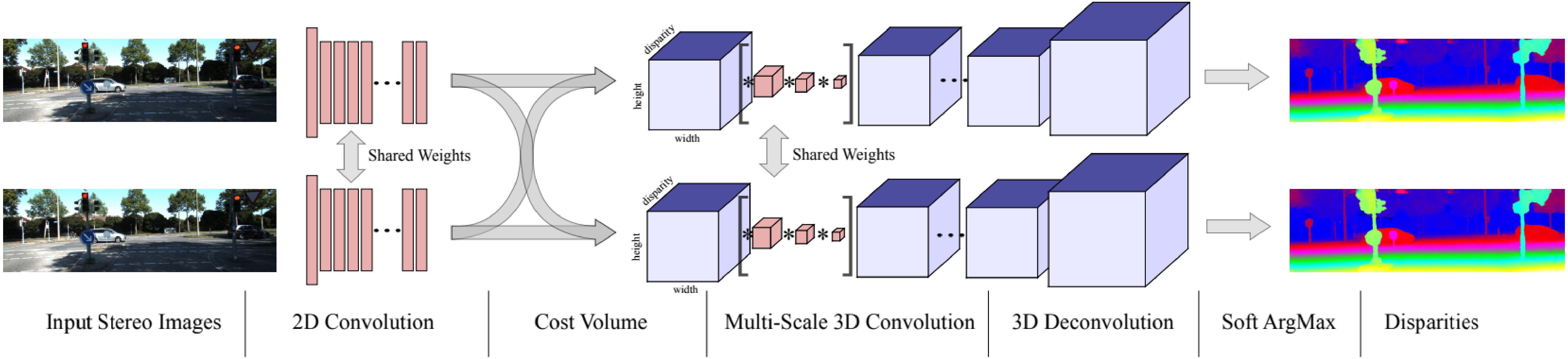


A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation [Mayer+ CVPR 2016]



- Network trained on synthetic data (Flying Chairs3D) and fine-tuned on KITTI2015
- Observations in the paper: Fine-tuning on KITTI improves the results on that dataset but increases errors on other datasets.
  - KITTI 2015 has small disparity range
  - Fine-tuning hurts performance on other datasets with larger disparity range.

End-to-End Learning of Geometry and Context for Deep Stereo Regression [Kendall+ arxiv 2017]



- Extensive use of 3D convolutions; capture context
- Differentiable soft-argmin (first proposed by [..., Bengio] ICLR 2014)

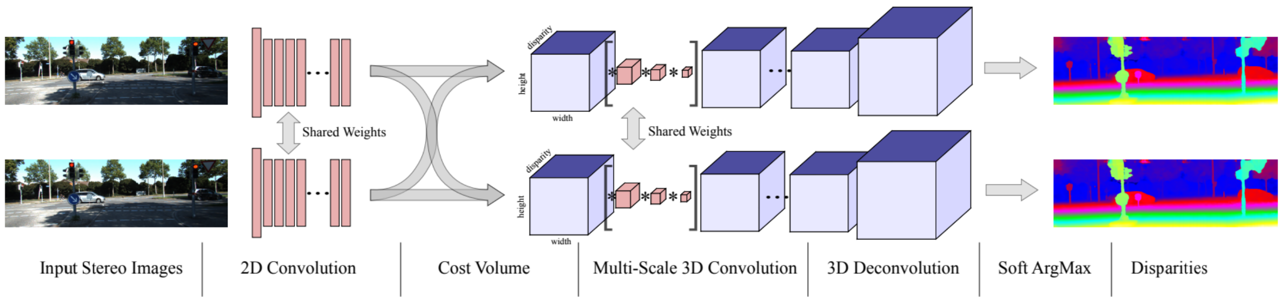
End-to-End Learning of Geometry and Context for Deep Stereo Regression [Kendall+ arxiv 2017]

KITTI 2015 Stereo Benchmark

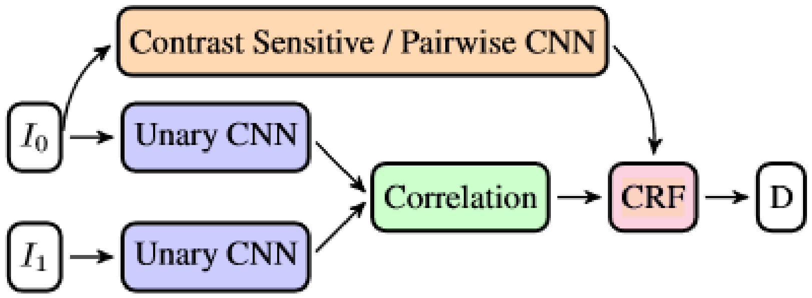
	All Pixels			Non-Occluded Pixels			Runtime
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	(s)
MBM [11]	4.69	13.05	6.08	4.33	12.12	5.61	0.13
ELAS [15]	7.86	19.04	9.72	6.88	17.73	8.67	0.3
Content-CNN [32]	3.73	8.58	4.54	3.32	7.44	4.00	1.0
DispNetC [34]	4.32	**4.41**	4.34	4.11	**3.72**	4.05	**0.06**
MC-CNN [50]	2.89	8.88	3.89	2.48	7.64	3.33	67
PBCP [40]	2.58	8.74	3.61	2.27	7.71	3.17	68
Displets v2 [18]	3.00	5.56	3.43	2.73	4.95	3.09	265
GC-Net (this work)	**2.21**	6.16	**2.87**	**2.02**	5.58	**2.61**	0.9

New Trends

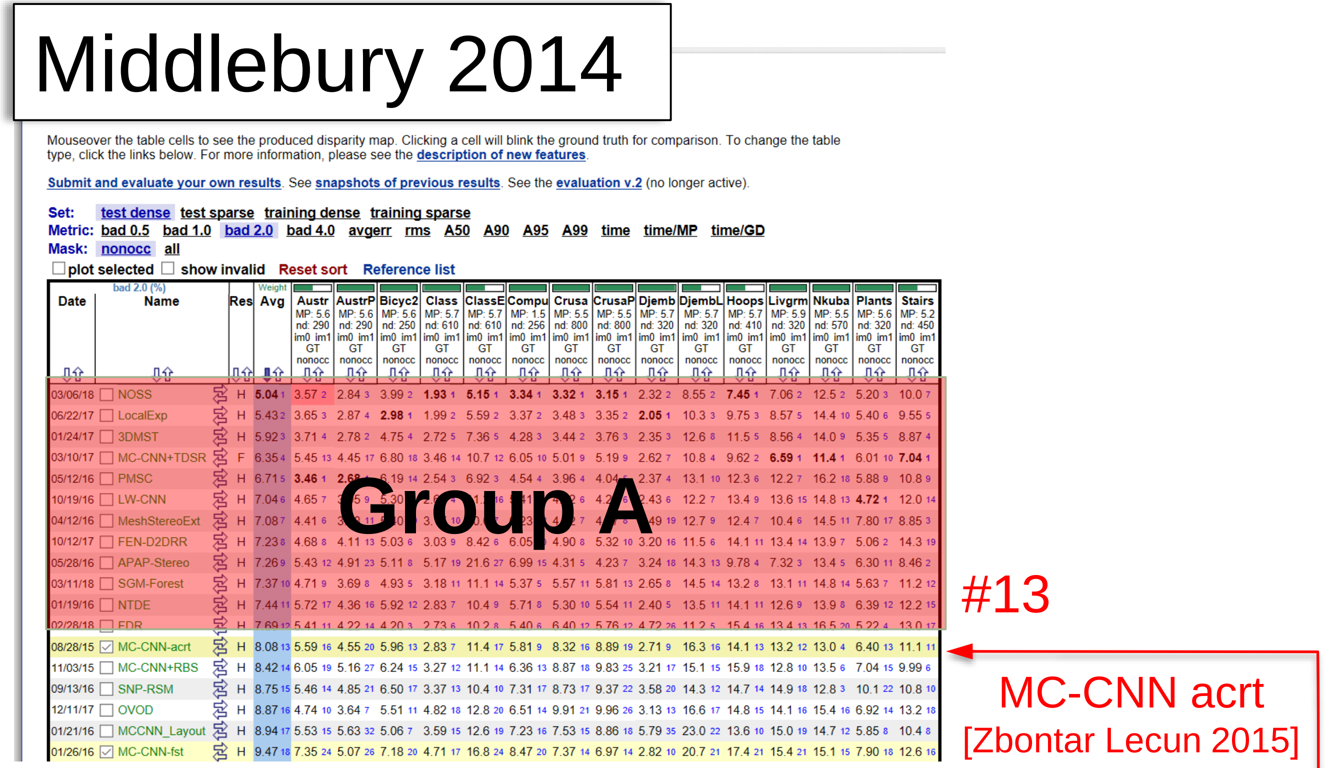
- Learning the matching cost:
  - MC-CNN [Zbontar + Lecun 2015], Chen+ 2015, Luo+ 2016
- Continuous MRFs: [Taniai+ 2017] (Rank 1 on Middlebury 2014!)
- Deep stereo regression (end to end training)
  - FlowNet [Dosovitskiy+ 2015], DispNet [Mayer+ 2016]
- Return of “Correlation”
  - DispNetCorr [Mayer+ 2016]
  - GC-Net [Kendall+ 2017]

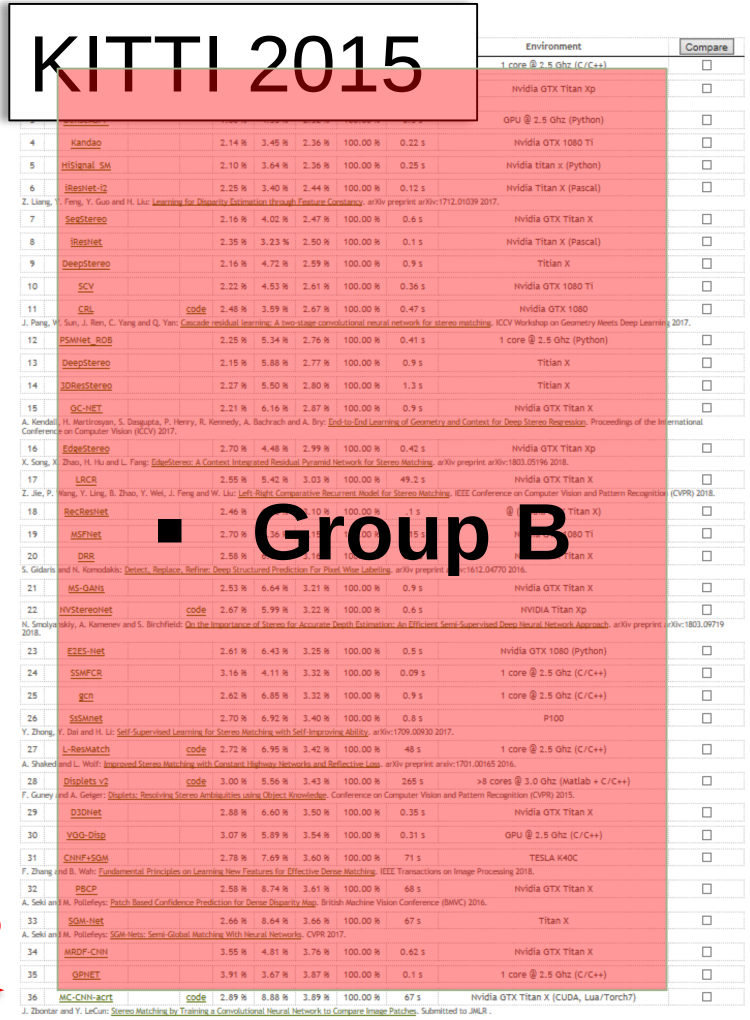


- Return of “CRFs” (Hybrid CNN-CRF models)
  - Seki and Pollefeys 2017, Knobelreiter+ 2017



Stereo Benchmark Rankings





- Group A and B have no methods in common!
- Group A entries all use MC-CNN acrt but no other “deep learning” technique!
- Group B methods do NOT use MC-CNN acrt; they use ResNet, 3D convolutions, 3D deconvolutions, U-shaped Nets, RNNs; End to end learning is very popular!

CVPR 2017 Robust Vision Challenge workshop

Must train **one** model on combined training set and submit to all benchmarks!

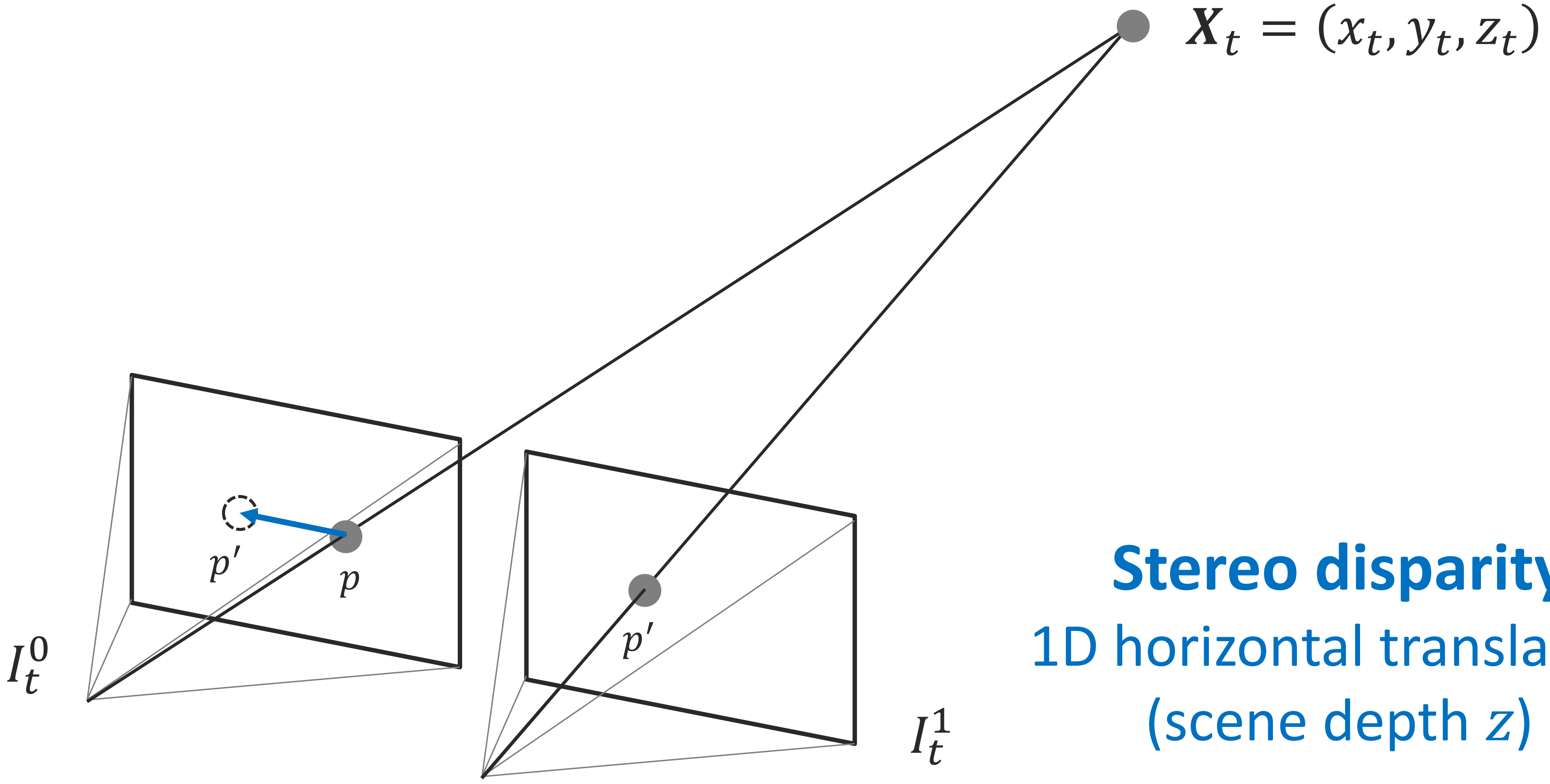
	Stereo	MVS	Flow	Depth	Semantic	Instance
Middlebury	✗	✗	✗			
KITTI	✗		✗	✗	✗	✗
MPI Sintel			✗			
ETH3D	✗	✗				
HD1K			✗			
ScanNet				✗	✗	✗
Cityscapes					✗	✗
WildDash					✗	✗

ROB methods (current rankings)

METHOD	Deep learning?	Middlebury Rank	KITTI Rank	ETH3D Rank
NOSS_ROB	?	1	133	2
DN-CSS_ROB	✓	40	40	1
PSMNet_ROB	✓	60+	9	7
NaN_ROB	✓	4	33	10
SGM		31	90+	21
total		80	144	39

Stereoscopic Scene Flow

Stereoscopic Scene Flow



Stereo disparity

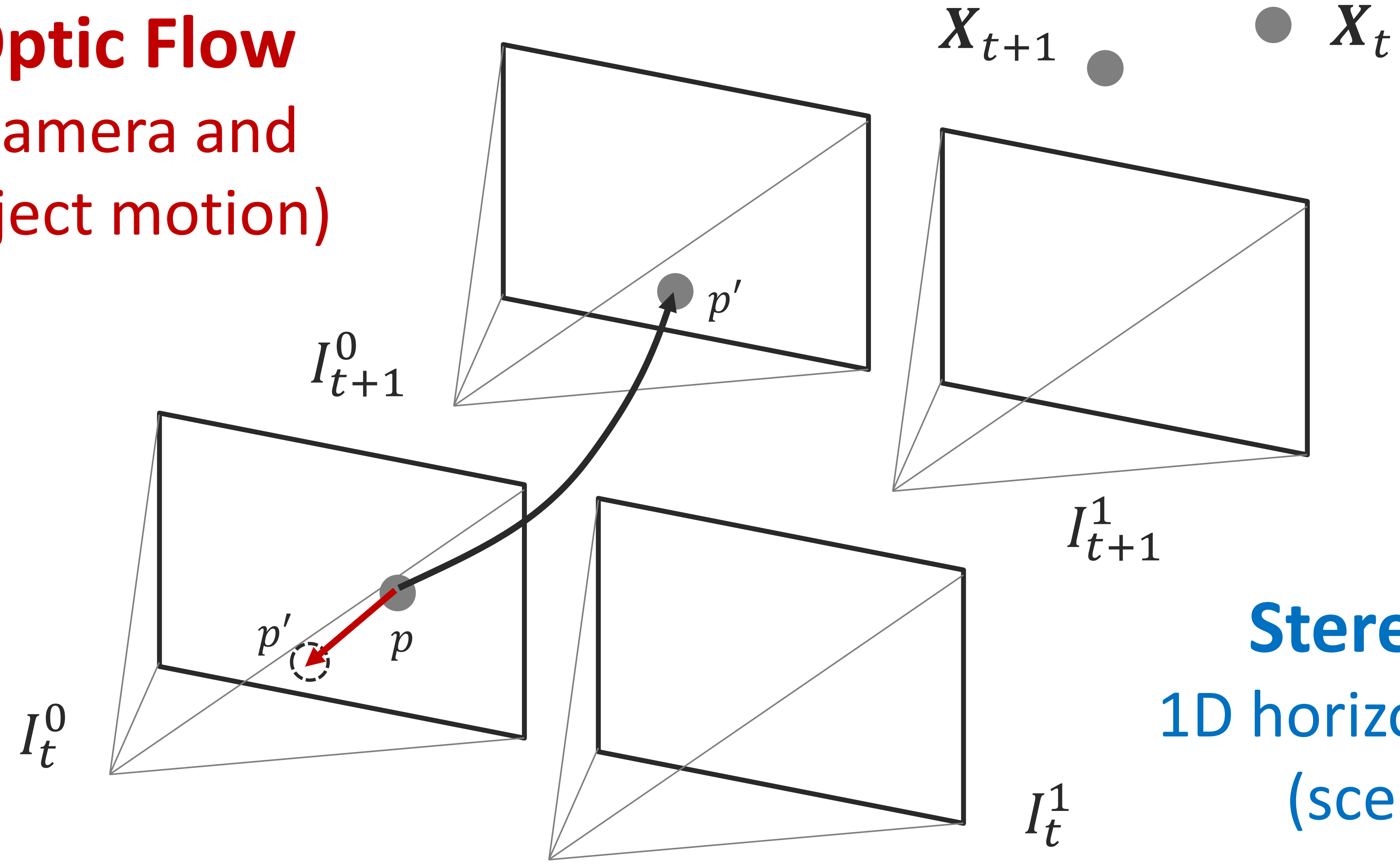
1D horizontal translation

(scene depth z)

Stereoscopic Scene Flow

Optic Flow

(camera and object motion)



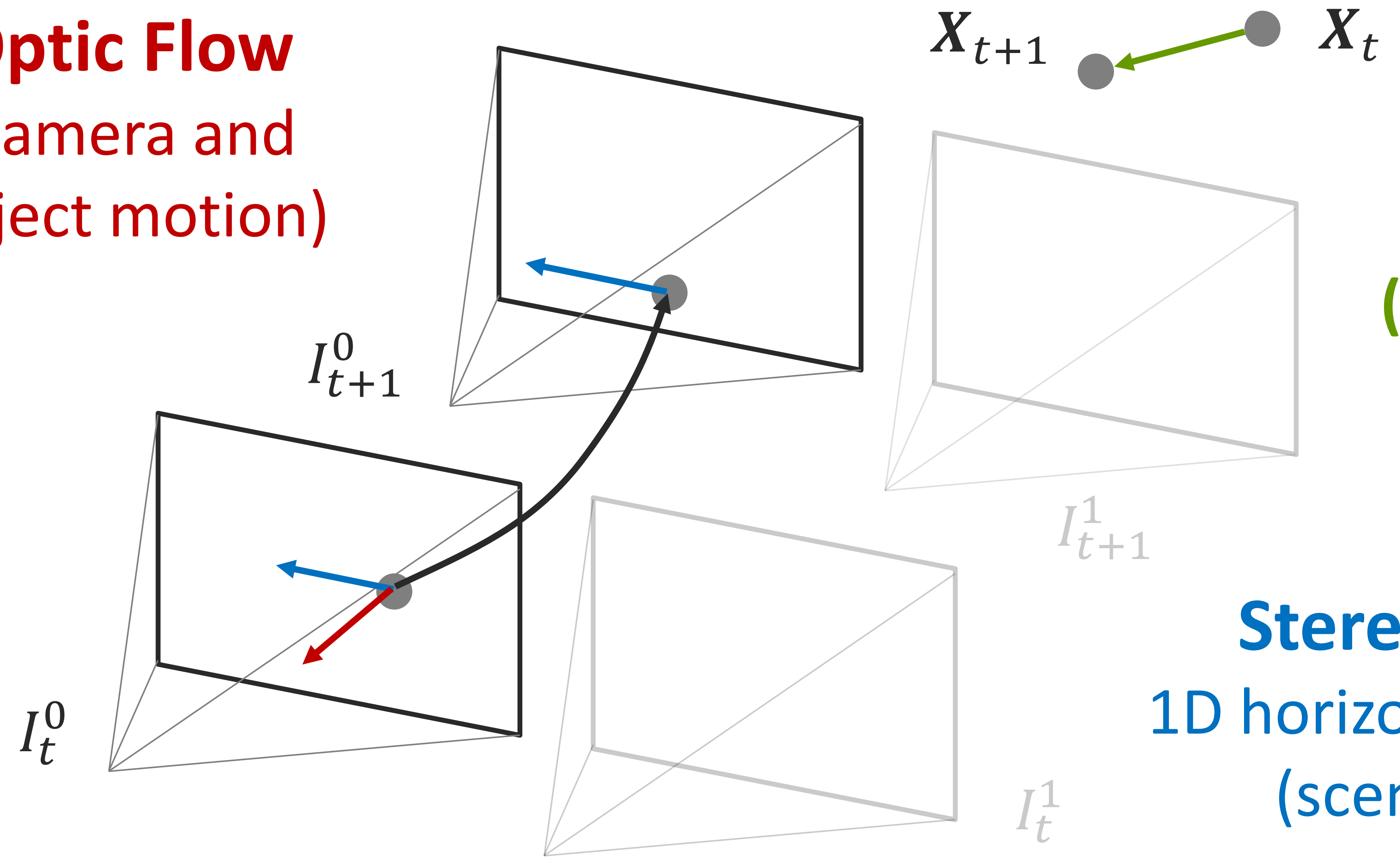
Stereo disparity

1D horizontal translation

(scene depth z)

Stereoscopic Scene Flow

Optic Flow
(camera and object motion)



Scene Flow
3D translation
(object motion)

Stereo disparity
1D horizontal translation
(scene depth z)

Scene Flow

Fast Multi-frame Stereo Scene Flow with Motion Segmentation

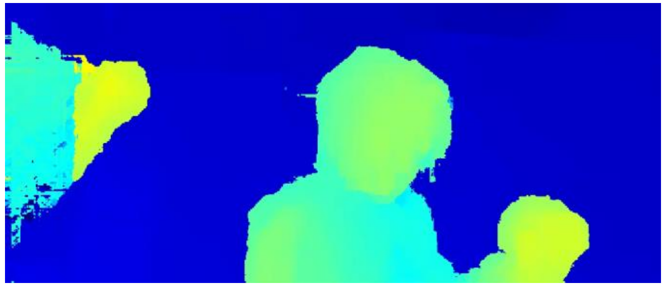
Taniai, Sinha, Sato CVPR 2017

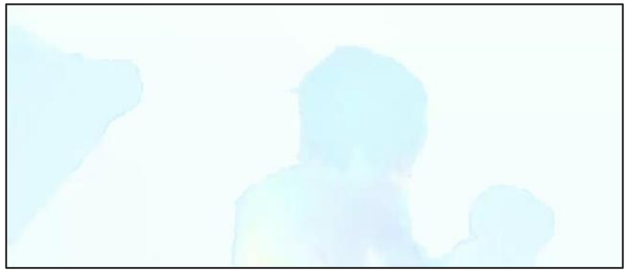
Input: Stereo Video


Left


Right

Output

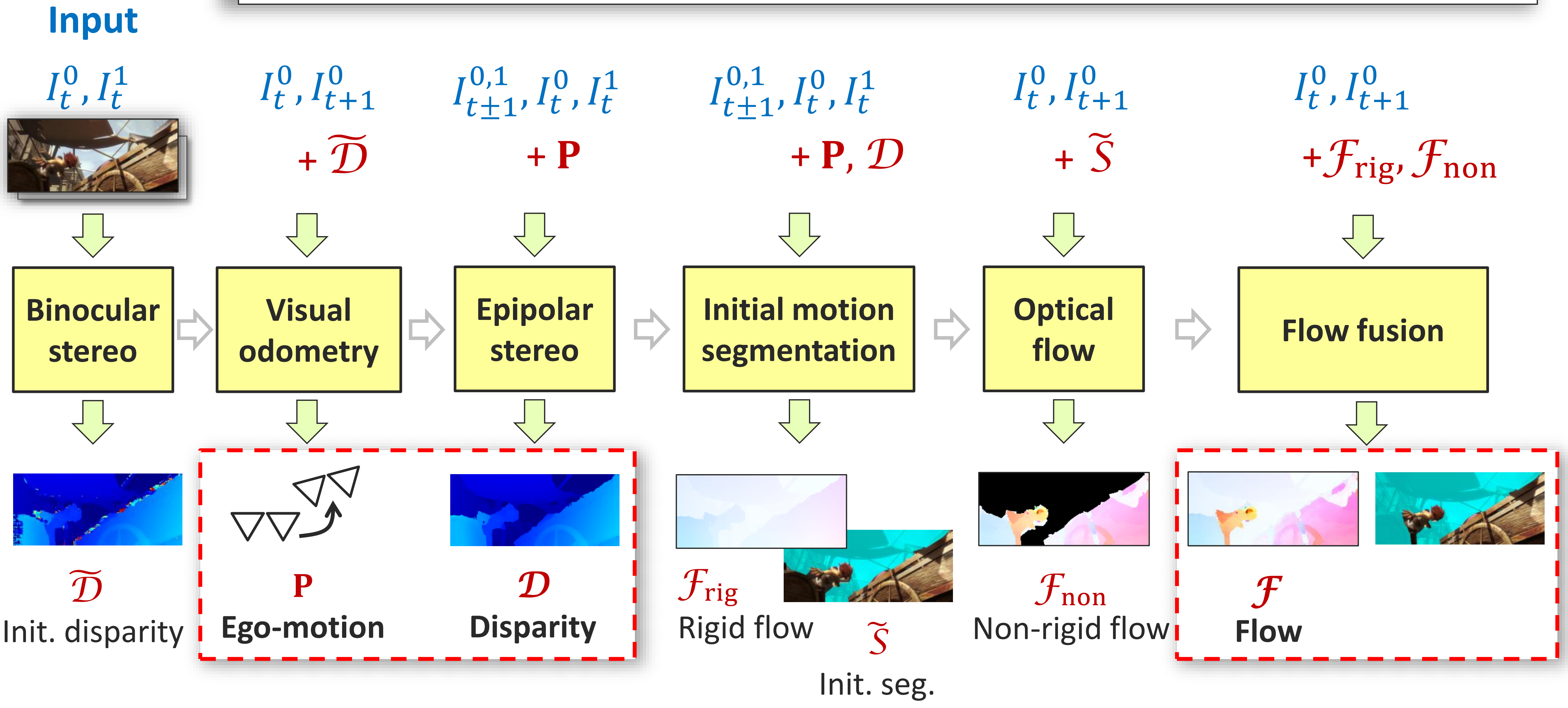

Disparity Map


Optical Flow


Moving Object segmentation

Fast Multi-frame Stereo Scene Flow with Motion Segmentation

Taniai, Sinha, Sato CVPR 2017



Fast Multi-frame Stereo Scene Flow with Motion Segmentation

Taniai, Sinha, Sato CVPR 2017

KITTI 2015 Scene Flow Benchmark (November 2016)

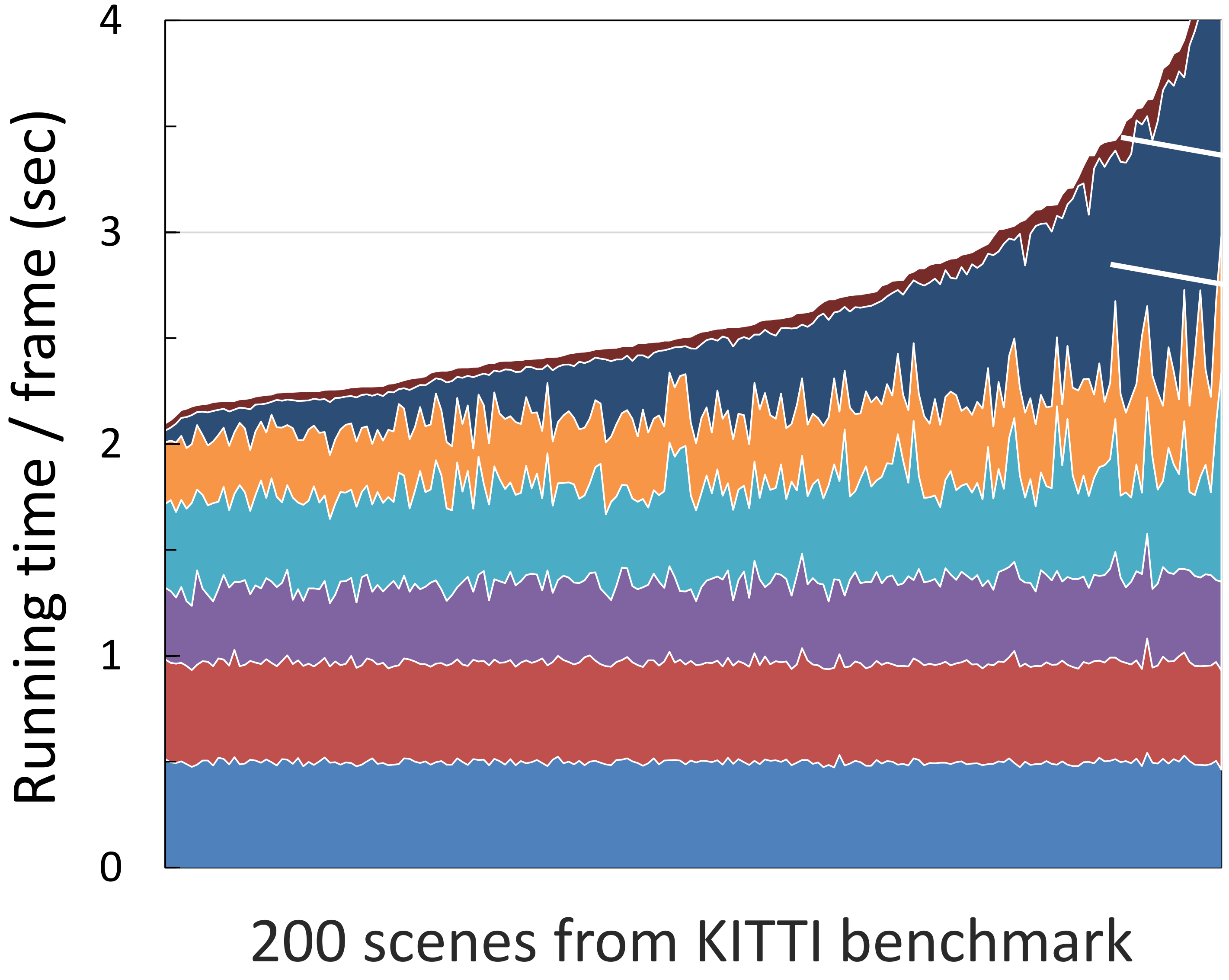
Rank	Method	D1-bg	D1-fg	D1-all	D2-bg	D2-fg	D2-all	Fl-bg	Fl-fg	Fl-all	SF-bg	SF-fg	SF-all	Time
1	PRSM [43]	**3.02**	**10.52**	**4.27**	**5.13**	**15.11**	**6.79**	**5.33**	**17.02**	**7.28**	**6.61**	**23.60**	**9.44**	300 s
2	OSF [30]	4.54	12.03	5.79	5.45	19.41	7.77	5.62	22.17	8.37	7.01	28.76	10.63	50 min
3	**FSF+MS** (ours)	5.72	11.84	6.74	7.57	21.28	9.85	8.48	29.62	12.00	11.17	37.40	15.54	2.7 s
4	CSF [28]	4.57	13.04	5.98	7.92	20.76	10.06	10.40	30.33	13.71	12.21	36.97	16.33	80 s
5	PR-Sceneflow [42]	4.74	13.74	6.24	11.14	20.47	12.69	11.73	27.73	14.39	13.49	33.72	16.85	150 s
8	PCOF + ACTF [10]	6.31	19.24	8.46	19.15	36.27	22.00	14.89	62.42	22.80	25.77	69.35	33.02	0.08 s (GPU)
12	GCSF [8]	11.64	27.11	14.21	32.94	35.77	33.41	47.38	45.08	47.00	52.92	59.11	53.95	2.4 s

200 road scenes with multiple moving objects

Rank	SF-all	Time
1	**9.44**	300 s
2	10.63	50 min
3	15.54	2.7 s
4	16.33	80 s
5	16.85	150 s
8	33.02	0.08 s (GPU)
12	53.95	2.4 s

Breakdown of Running times

CPU: 3.5 GHz × 4 Cores
Image: (1242 × 375) × 0.65 scale



0.07 sec Flow fusion
0.48 sec Optical flow
0.36 sec Initial segmentation
0.47 sec Epipolar stereo
0.38 sec Visual odometry
0.47 sec Binocular stereo
0.72 sec Initialization

2.72 sec per frame

Summary

- Semi Global Matching (SGM) and extensions
- Geometric and Semantic Priors
- Continuous optimization
- High Resolution Stereo
- Deep Learning in Stereo
- Stereoscopic Scene Flow