Recovering Image Correspondence: New Methods and Applications

Sudipta N. Sinha

Microsoft Research

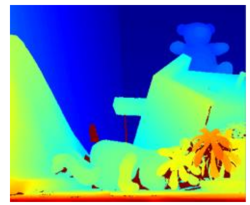
Redmond, USA

NII Shonan Meeting on Optimization Methods in Geometric Vision,

January 28, 2019

Overview

RGB Stereo Images



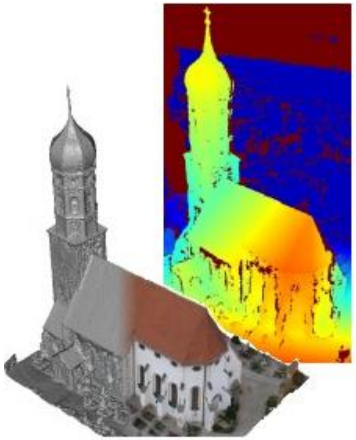
Disparity Map

RGB/NIR Image Alignment

- ❑ Dense stereo matching
  - Optimization via Semi Global Matching (SGM)
  - Two extensions to SGM
- ❑ Learning to align images from scratch
  - Joint framework for local feature descriptor learning and image alignment
  - Application: RGB / NIR image registration

Semi Global Matching (SGM)

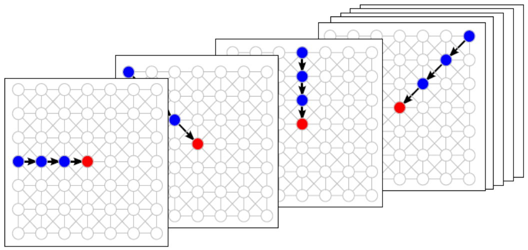
[Hirschmüller 2005]



- Motivation: Markov Random Field (MRF) inference via Graph Cuts, BP etc. is too slow and approximate. So why not approximate even more.
- SGM is parallelizable; runs on GPUs and FPGAs.
- Widely used: assisted driving, robotics, aerial mapping.

Semi Global Matching (SGM)

[Hirschmüller 2005]

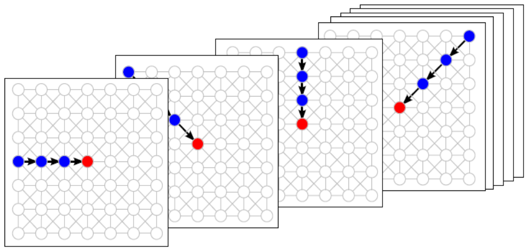


- Solve several independent 1D scanline optimization problems; one for each of 4 or 8 directions.

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

Semi Global Matching (SGM)

[Hirschmüller 2005]



- Solve several independent 1D scanline optimization problems; one for each of 4 or 8 directions.

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

- Sum the costs and select min cost disparity at each pixels.

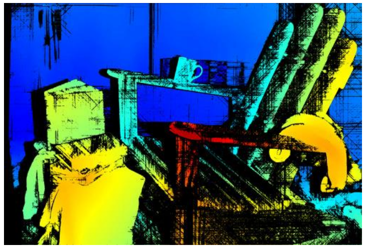
$$S(\mathbf{p}, d) = \sum_{\mathbf{r}} L_{\mathbf{r}}(\mathbf{p}, d)$$

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

Two Limitations of SGM

- Fronto-parallel bias due to pairwise smoothness term; leads to errors on slanted textureless surfaces.

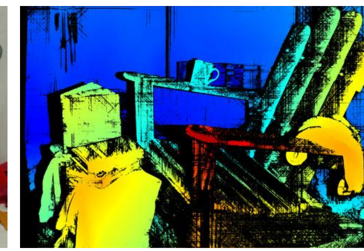
$$V(d, d') = \begin{cases} 0 & \text{if } d = d' \\ P_1 & \text{if } |d - d'| = 1 \\ P_2 & \text{if } |d - d'| \geq 2 \end{cases}$$



Two Limitations of SGM

- Fronto-parallel bias due to pairwise smoothness term; leads to errors on slanted textureless surfaces.

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

- Summing up costs and picking the best disparity (last two steps lack proper justification

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

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

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

SGM with Surface Orientation Prior

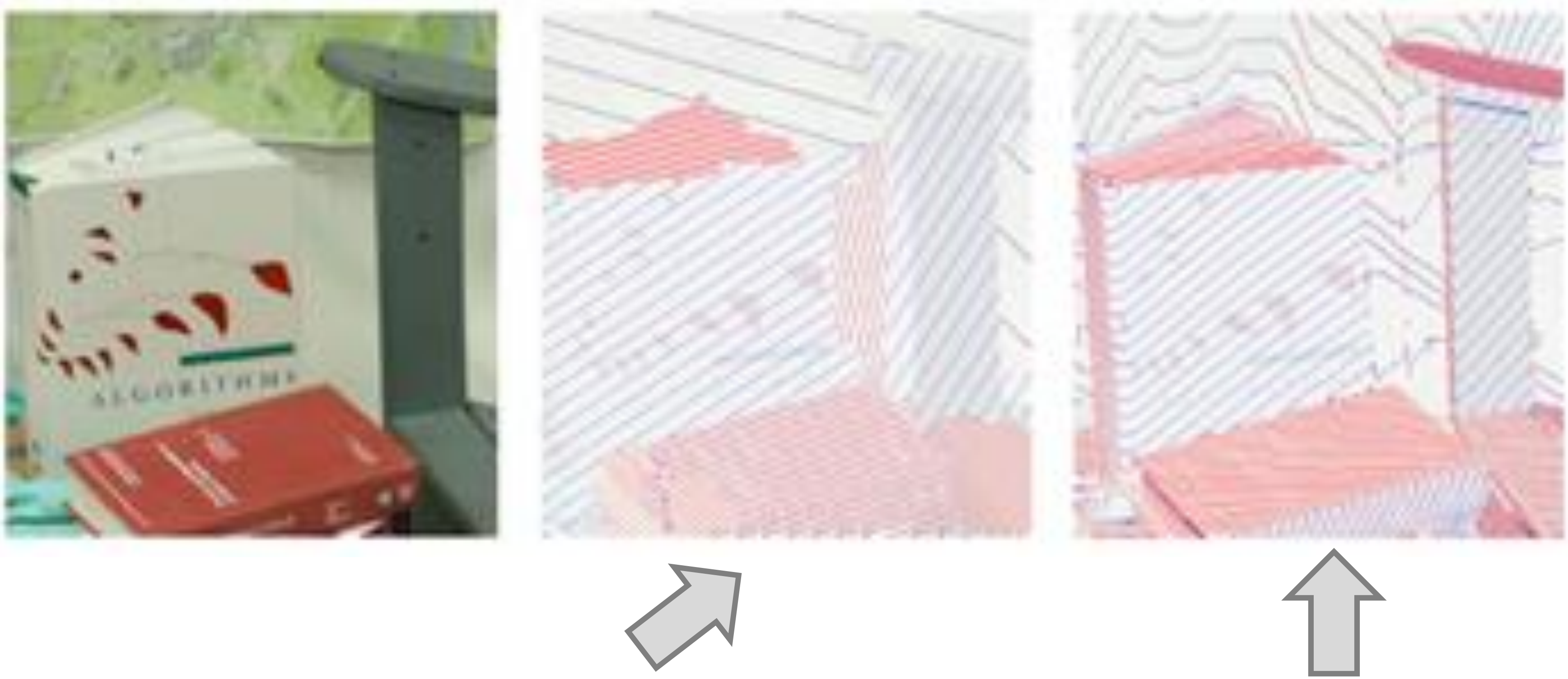
[Scharstein, Taniai, Sinha, 3DV 2017]

- If we knew the surface slant, we can replace the fronto-parallel bias with bias parallel to surface.

- <u>Approach:</u>
  - *Fit surfaces (planes) to an initial depth map.*
  - *Alternatively, integrate a given surface normal map.*
  - *Discretize* disparity surface and record pixels where the disparity "changes" (+/- 1).
  - During optimization, bias pairwise terms at those pixels.

SGM with Surface Orientation Prior

[Scharstein, Taniai, Sinha, 3DV 2017]



SGM-EP
Low-resolution stereo matching
+ Plane fitting

SGM-GS
Ground truth oracle

SGM with Surface Orientation Prior

- Pairwise Terms.
  - SGM

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

  - SGM-P (2D Prior)

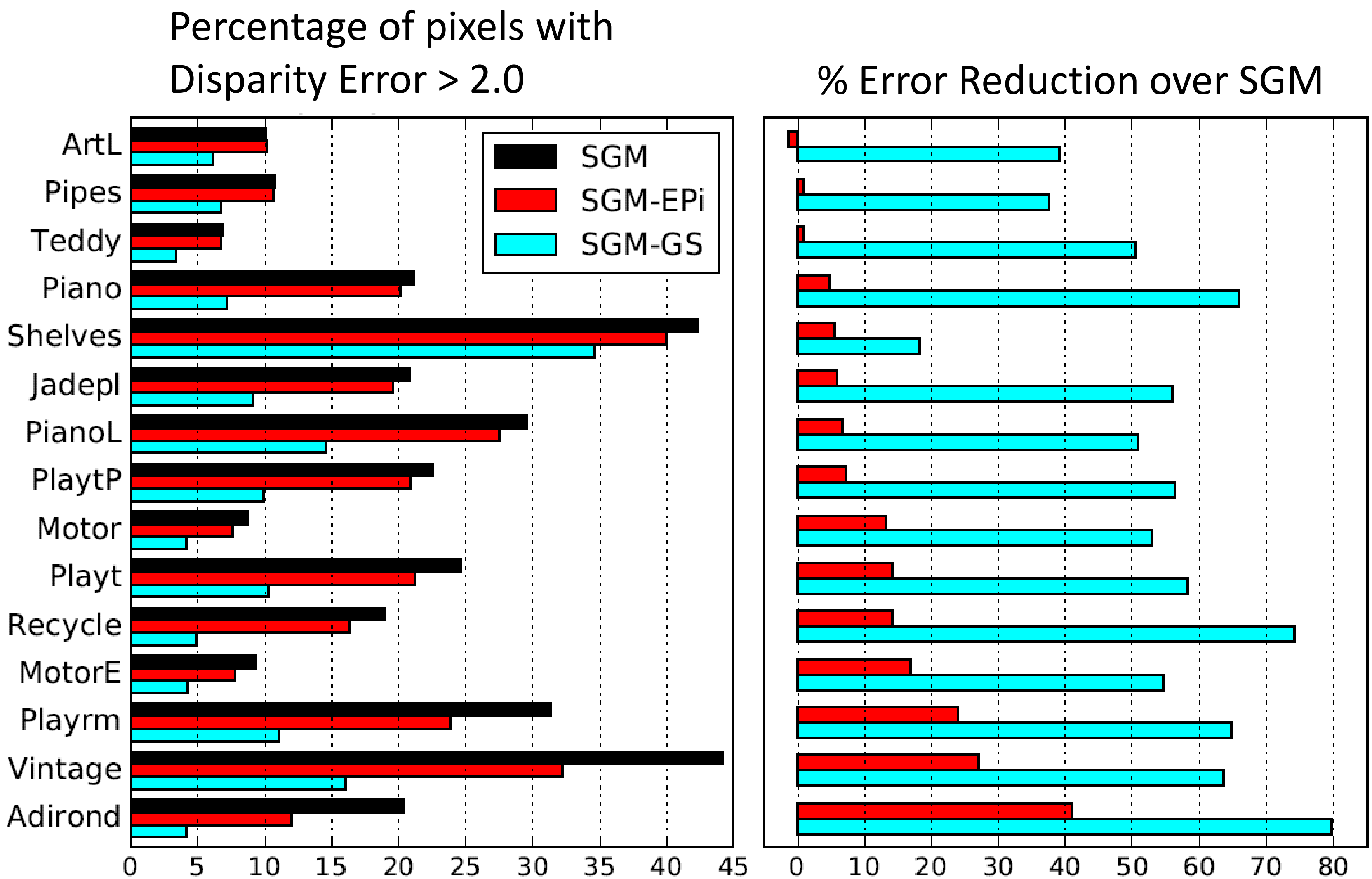
$$V_S(d_{\mathbf{p}}, d'_{\mathbf{p}}) = V(d_{\mathbf{p}} + j_{\mathbf{p}}, d'_{\mathbf{p}})$$

$$j_{\mathbf{p}} \in \{-1, 0, +1\}$$

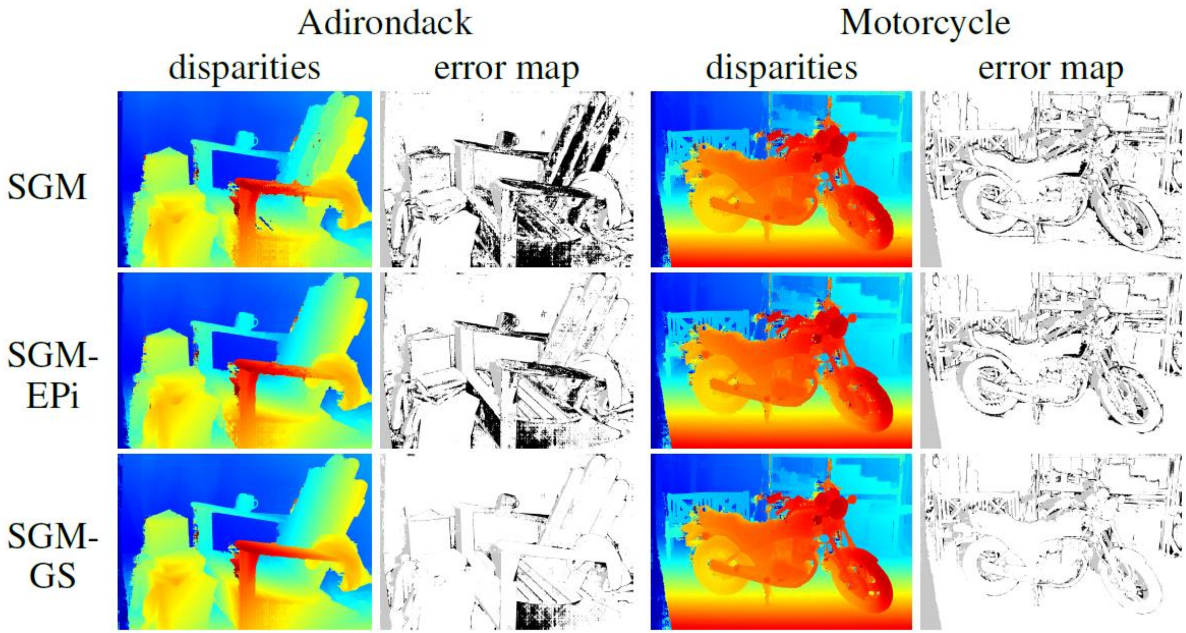
  - SGM-P (3D Prior)

$$V_S(d_{\mathbf{p}}, d'_{\mathbf{p}}) = V(d_{\mathbf{p}} + j_{\mathbf{p}}(d_{\mathbf{p}}), d'_{\mathbf{p}})$$

Results


Percentage of pixels with
Disparity Error > 2.0
% Error Reduction over SGM
SGM
SGM-EPi
SGM-GS
ArtL
Pipes
Teddy
Piano
Shelves
Jadepl
PianoL
PlaytP
Motor
Playt
Recycle
MotorE
Playrm
Vintage
Adirond
0 5 10 15 20 25 30 35 40 45
0 10 20 30 40 50 60 70 80

Conclusions



- Huge accuracy boost in scenes with slanted untextured surfaces.
- Soft constraint; inaccurate normals don't hurt accuracy.
- 2D prior version adds minimal computational overhead.
- Accurate estimation of surface orientation can be difficult.

Learning to Fuse Proposals in SGM

[Schoenberger, Sinha and Pollefeys, ECCV 2018]

- **SGM steps:**

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

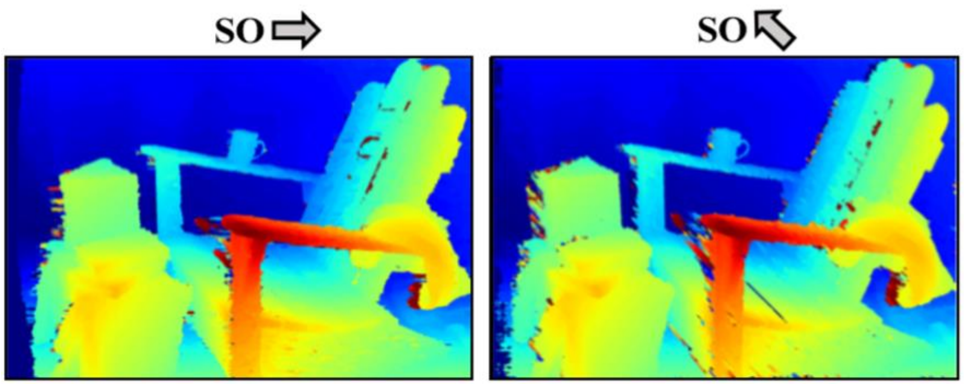
2. $$S(\mathbf{p}, d) = \sum_{\mathbf{r}} L_{\mathbf{r}}(\mathbf{p}, d)$$

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

- **Main Idea:**
  - Replace steps 2 and 3 with a learned predictor.
  - The predictor takes disparity maps obtained via scanline optimization and directly estimates the final disparity map.

Learning to Fuse Proposals in SGM

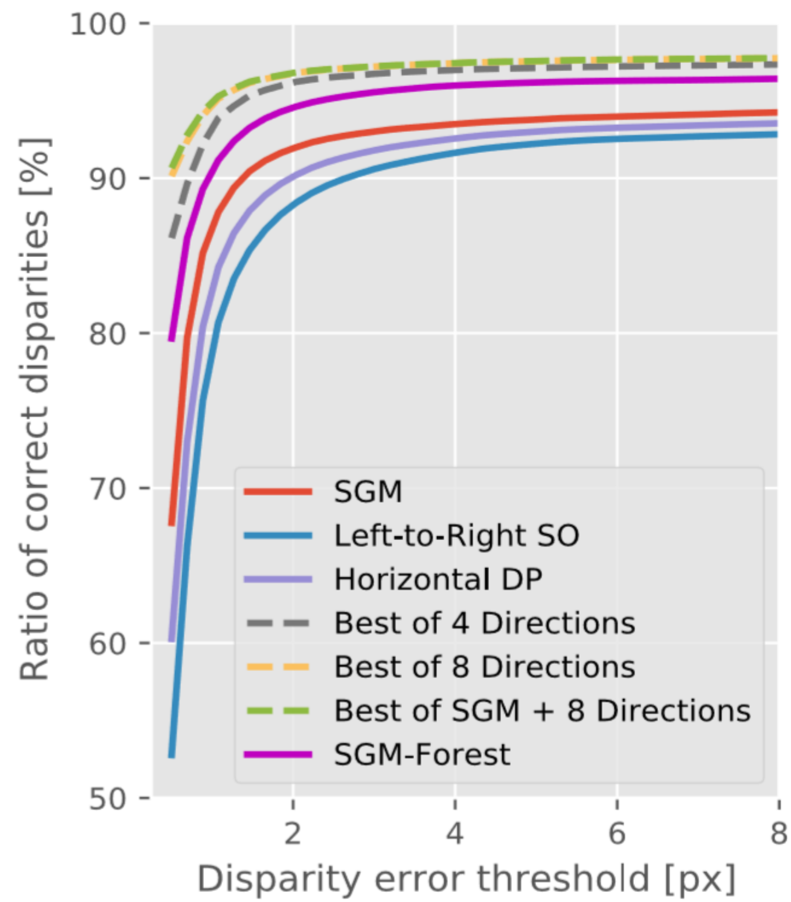
[Schoenberger, Sinha and Pollefeys, ECCV 2018]



Two candidates obtained via scanline optimization

❑ Motivation

- "Best of k directions" oracle is significantly better than SGM.

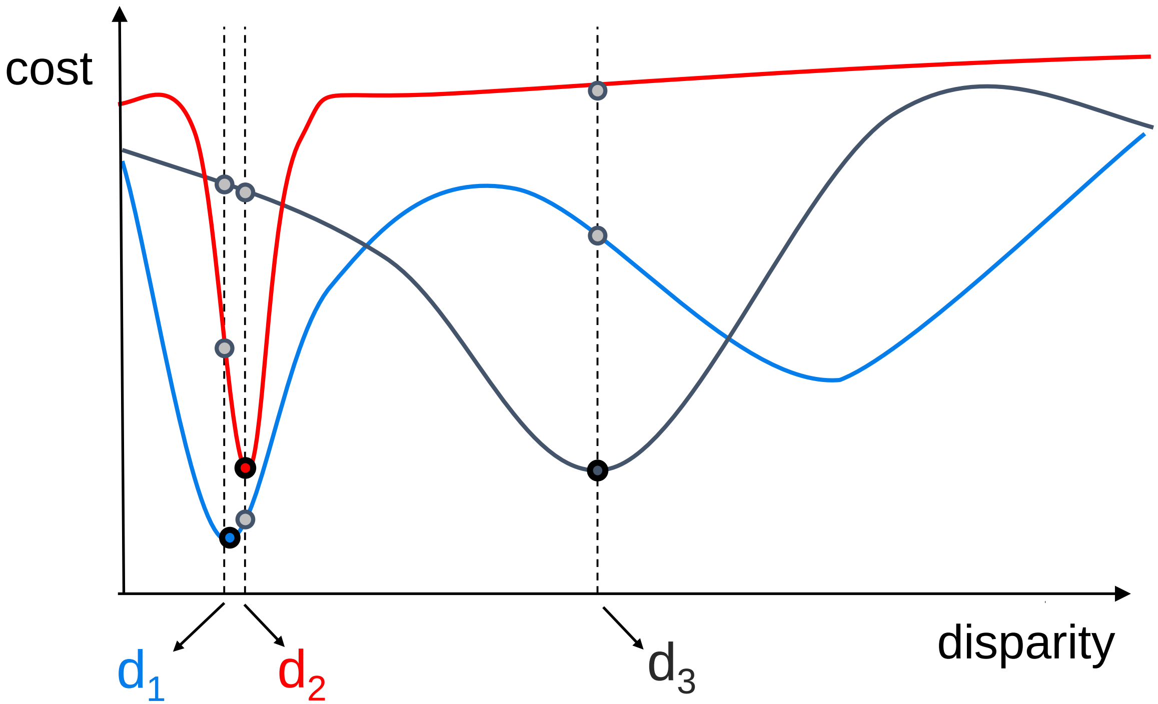


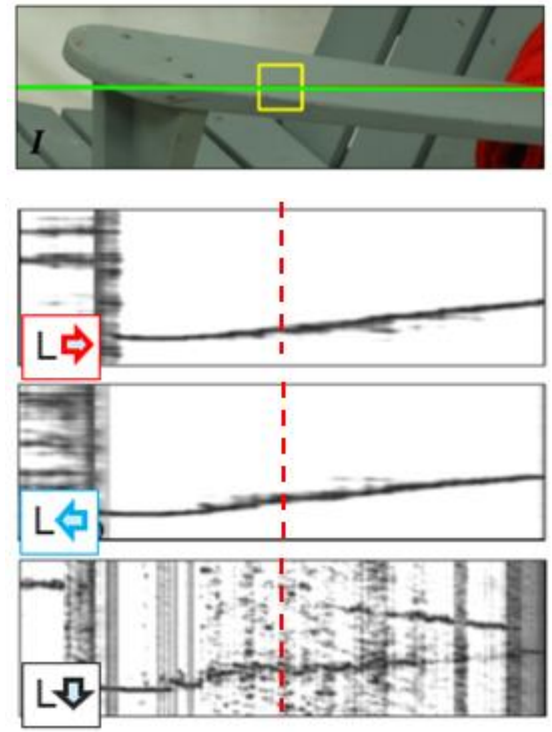
Learning to Fuse Proposals in SGM

[Schoenberger, Sinha and Pollefeys, ECCV 2018]

- **Approach** (SGM Forest):
  1. Run SO to get ***k*** disparity map proposals
  2. At each pixel
     - Construct ***per-pixel*** feature vector (*see next slide*)
     - Pick best disparity using a random forest classifier
     - Forest outputs probabilities
  3. Post-processing using probability map

Computing Per-pixel Features





SGM Forest: Results

Datacost	Method	Middlebury 2014				KITTI 2015				ETH3D 2017			
		0.5px	1px	2px	4px	0.5px	1px	2px	4px	0.5px	1px	2px	4px
all													
NCC	SGM	69.23	42.36	27.96	22.25	60.59	33.79	15.06	8.34	32.52	16.71	10.66	7.69
	SGM-F.	**64.00**	**37.22**	**22.85**	**17.09**	**52.39**	**25.80**	**10.11**	**4.69**	**22.48**	**11.26**	**6.36**	**4.35**
MC-CNN-fast	SGM	65.82	36.22	21.98	17.47	58.48	31.39	13.30	7.02	26.34	10.50	6.13	4.52
	SGM-F.	**62.04**	**32.96**	**18.22**	**13.16**	**51.03**	**24.05**	**8.73**	**3.78**	**17.62**	**7.17**	**3.66**	**2.51**
MC-CNN-acrt	SGM	65.58	36.08	20.66	16.24	57.24	28.55	9.54	5.26	39.03	16.34	9.14	6.67
	SGM-F.	**59.20**	**30.58**	**16.57**	**11.62**	**46.88**	**19.77**	**6.51**	**2.97**	**27.40**	**11.89**	**7.30**	**5.52**

MC-CNN [Zbontar and Lecun 2015]

- SGM-Forest consistently outperforms standard SGM and prior SGM variants.

SGM Forest: Ablation Study

Method	Left View Scanlines	Right View Scanlines	Filtering	Training Dataset	bad 0.5px [%]	bad 1px [%]	bad 2px [%]	bad 4px [%]	Time [s]
		all							
SGM	all			–	65.58	36.08	20.66	16.24	**3.0**
SGM − $\min_d L_{\mathbf{r}}(\mathbf{p}, d)$	all			–	66.79	38.35	23.32	18.36	3.1
SGM − $\min_d \operatorname{median}_{\mathbf{r}} L_{\mathbf{r}}(\mathbf{p}, d)$	all			–	67.53	39.75	23.34	18.12	3.2
SGM-SVM	all			M	60.89	32.59	20.31	16.16	323.7
SGM-MLP	all			M	60.49	32.61	20.25	16.14	21.0
SGM-Forest	horiz+vert			M	61.09	32.69	18.02	13.19	5.7
	top-down			M	61.31	32.85	18.31	13.37	5.8
	bottom-up			M	61.38	32.91	18.42	13.43	5.8
	all			M	60.28	32.15	17.90	13.14	6.1
	all	•		M	60.18	32.08	17.69	12.91	6.3
	all	•	•	E	59.89	30.69	16.78	11.67	8.2
	all	•	•	K	59.70	30.61	16.72	11.67	8.2
	all	•	•	M	**59.20**	**30.58**	**16.57**	**11.62**	8.2

Test Data:

Midd 2014 train set

Training Data:

E: ETH3D 2017

K: KITTI 2015

M: Midd 2005–06

- Excellent cross-dataset generalization.
- Model trained on 2005-06 data shows large accuracy gain on the significantly harder Middlebury 2014 scenes.
- Forest learns abstract patterns in the DSI; not in the input images.

SGM Forest: Benchmark Results

Middlebury 2014 (MC-CNN-acrt)

Method	non-occl.		all		Time
LocalExp	5.43%	#1	11.7%	#1	881s
3DMST	5.92%	#2	12.5%	#3	174s
MC-CNN+TDSR	6.35%	#2	12.1%	#3	657s
PMSC	6.71%	#4	13.6%	#4	599s
LW-CNN	7.04%	#5	17.8%	#15	314s
MeshStereoExt	7.08%	#6	15.7%	#9	161s
FEN-D2DRR	7.23%	#7	16.0%	#11	121s
APAP-Stereo	7.26%	#8	13.7%	#5	131s
SGM-Forest	**7.37%**	**#9**	**15.5%**	**#8**	**88s** *
NTDE	7.44%	#10	15.3%	#7	152s

Middlebury 2014 (MC-CNN-fast)

Method	non-occl.		all		Time
LocalExp	6.52 %	#1	12.1%	#1	846s
3DMST	7.08 %	#2	12.9%	#2	167s
APAP-Stereo	7.53%	#3	14.3%	#6	117s
FEN-D2DRR	7.89%	#4	14.1%	#4	73s
...					
MC-CNN-acrt	10.1%	#12	19.7%	#20	106s
...					
SGM-Forest	**11.1%**	**#19**	**17.8%**	**#14**	**9s** *
...					
MC-CNN-fast	11.7%	#21	21.5%	#27	1s

KITTI 2015

Method	Error	Time
CNNF+SGM	3.60% (#9)	71.0s
SGM-Net	3.66% (#11)	67.0s
MC-CNN-acrt	3.89% (#12)	67.0s
SGM-Forest	**4.38% (#14)**	**6.0s** *
MC-CNN-WS	4.97% (#18)	1.4s
SGM_ROB [17]	6.38% (#27)	0.1s
SGM+C+NL	6.84% (#31)	270.0s
SGM+LDOF	6.84% (#32)	86.0s
SGM+SF	6.84% (#33)	2700.0s
CSCT+SGM+MF	8.24% (#35)	6.4ms

ETH3D 2017

Method	non-occl.	all	Time
SGM-Forest	**5.40%**	**4.96%**	**5.21s** *
SGM_ROB [17]	10.08%	10.77%	0.15s
MeshStereo	11.94%	11.52%	159.24s
SPS-Stereo	15.83%	15.04%	1.59s
ELAS	17.99%	16.72%	0.13s

*** CPU impl.**

- #1 (ETH3D), #9 (Middlebury 2014), #14 (KITTI).
- Retains computational efficiency of SGM.

Learning to Align Images using Weak Geometric Supervision

Jing Dong[1,2] Byron Boots[1] Frank Dellaert[1]
Ranveer Chandra[2] Sudipta N. Sinha[2]

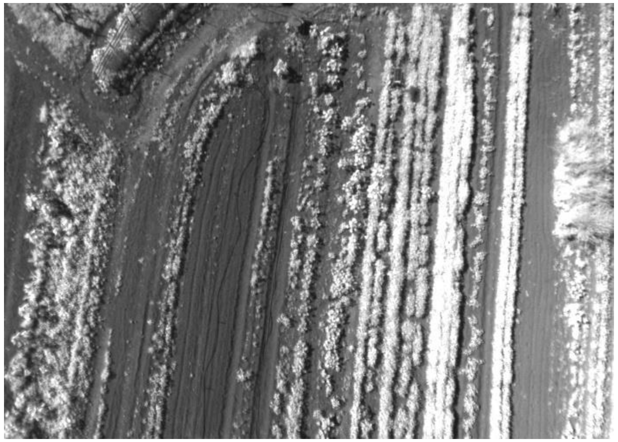
[1] Georgia Institute of Technology [2] Microsoft Research

3DV 2018

Learning Local Feature Descriptors

- Descriptor Learning typically needs supervised learning.
- Training them requires good image correspondences.
- For RGB images, easy to obtain such training data.
- However, not so easy for different imaging modalities (e.g. RGB/NIR).

RGB



NIR

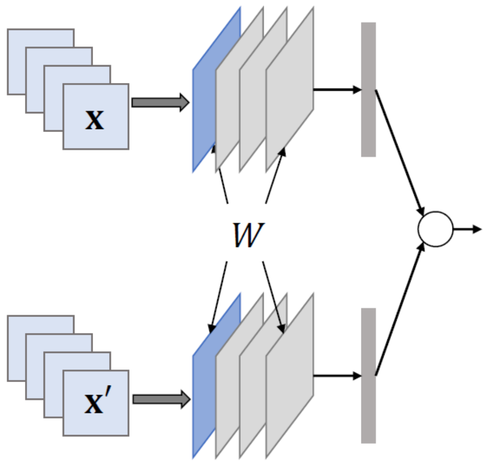
Goal

- Given two coarsely aligned images of scenes related by an unknown 2D homography, we compute the homography parameters.
- We do not assume any prior knowledge about features or image representations.

- **Main Idea:** We learn the feature descriptor representation from scratch on the image pair and jointly estimate the 2D homography parameters.

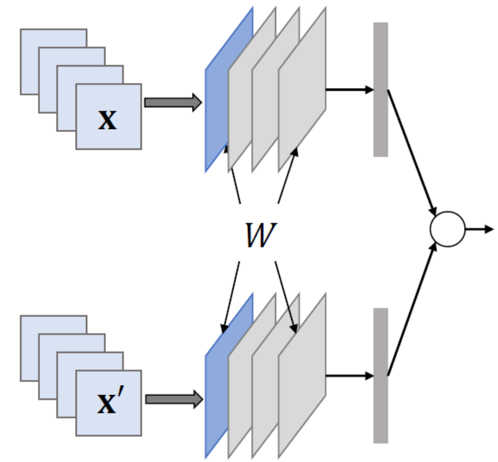
Siamese Networks

- Used for local descriptor learning
- Training set
  - P: true correspondence pairs
  - N: false correspondence pairs



Siamese Networks

- Used for local descriptor learning
- Training set
  - P: true correspondence pairs
  - N: false correspondence pairs



- Contrastive Loss

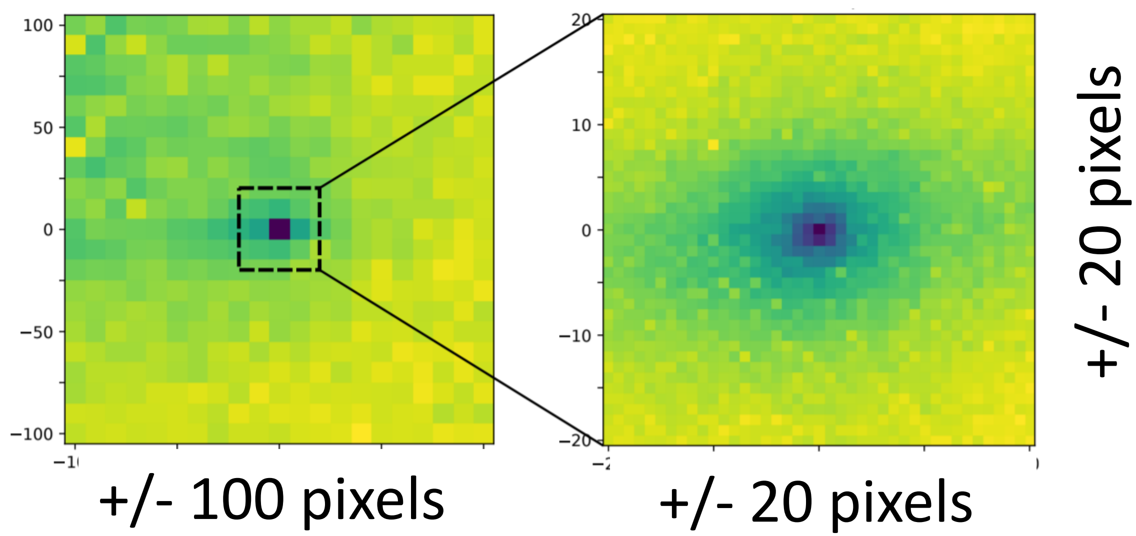
$$\boldsymbol{L}_0(\mathbf{x}, \mathbf{x}'; \theta) = \|\boldsymbol{f}(\mathbf{x}; \theta) - \boldsymbol{f}(\mathbf{x}'; \theta)\|_2$$

$$\boldsymbol{L}_1(\mathbf{x}, \mathbf{x}'; \theta) = \max(\,0,\ \mu - \|\boldsymbol{f}(\mathbf{x}; \theta) - \boldsymbol{f}(\mathbf{x}'; \theta)\|_2)$$

$$\underset{\theta}{\operatorname{argmin}} \left(\sum_{i=1}^{|\mathcal{P}|} \boldsymbol{L}_0(\mathbf{x}_i, \mathbf{x}'_i; \theta) + \sum_{j=1}^{|\mathcal{N}|} \boldsymbol{L}_1(\mathbf{x}_j, \mathbf{x}'_j; \theta) \right)$$

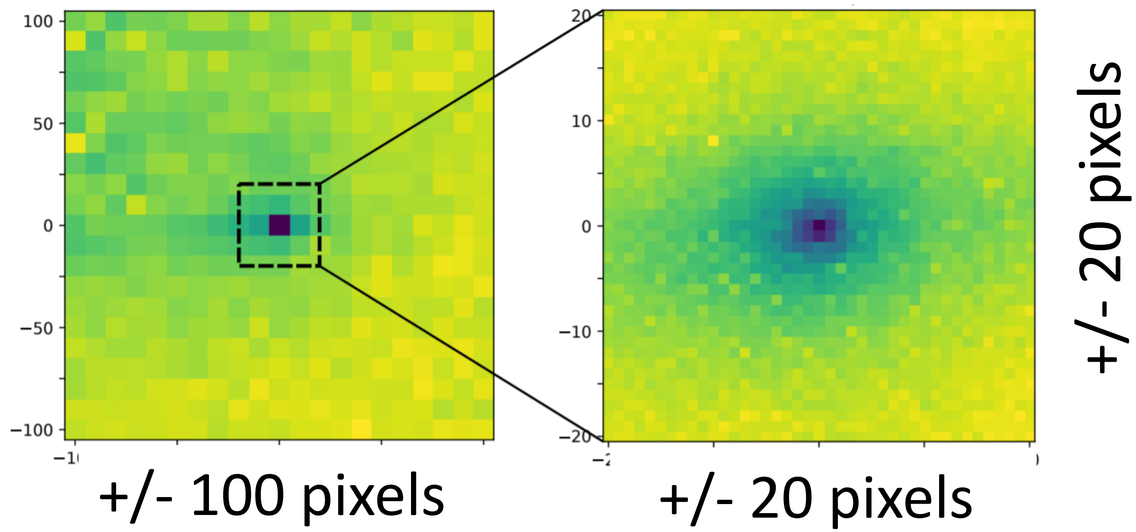
Insight

- Visualization of the training loss when severa networks are trained on misaligned image patches (shifted by 2D translational offsets).



Insight

- Visualization of the training loss when severa networks are trained on misaligned image patches (shifted by 2D translational offsets).



- Siamese network can be trained and homography parameters can be updated while minimizing the standard loss.
- Updates to the homography can also computed using backpropagation and SGD.

Our Formulation

- Positive set (true correspondences) re-estimated from current homography estimate

Homography-based image warping

Homography parameters

$$\boxed{\boldsymbol{L}_0(\mathbf{x};\psi,\theta)} = \|\boldsymbol{f}(\mathbf{x};\theta) - \boldsymbol{f}(\mathbf{w}(\mathbf{x};\psi);\theta)\|_2$$

$$\boldsymbol{L}_1(\mathbf{x},\mathbf{x}';\theta) = \max(\,0,\ \mu - \|\boldsymbol{f}(\mathbf{x};\theta) - \boldsymbol{f}(\mathbf{x}';\theta)\|_2)$$

Our Formulation

- Positive set (true correspondences) re-estimated from current homography estimate

Homography-based image warping

Homography parameters

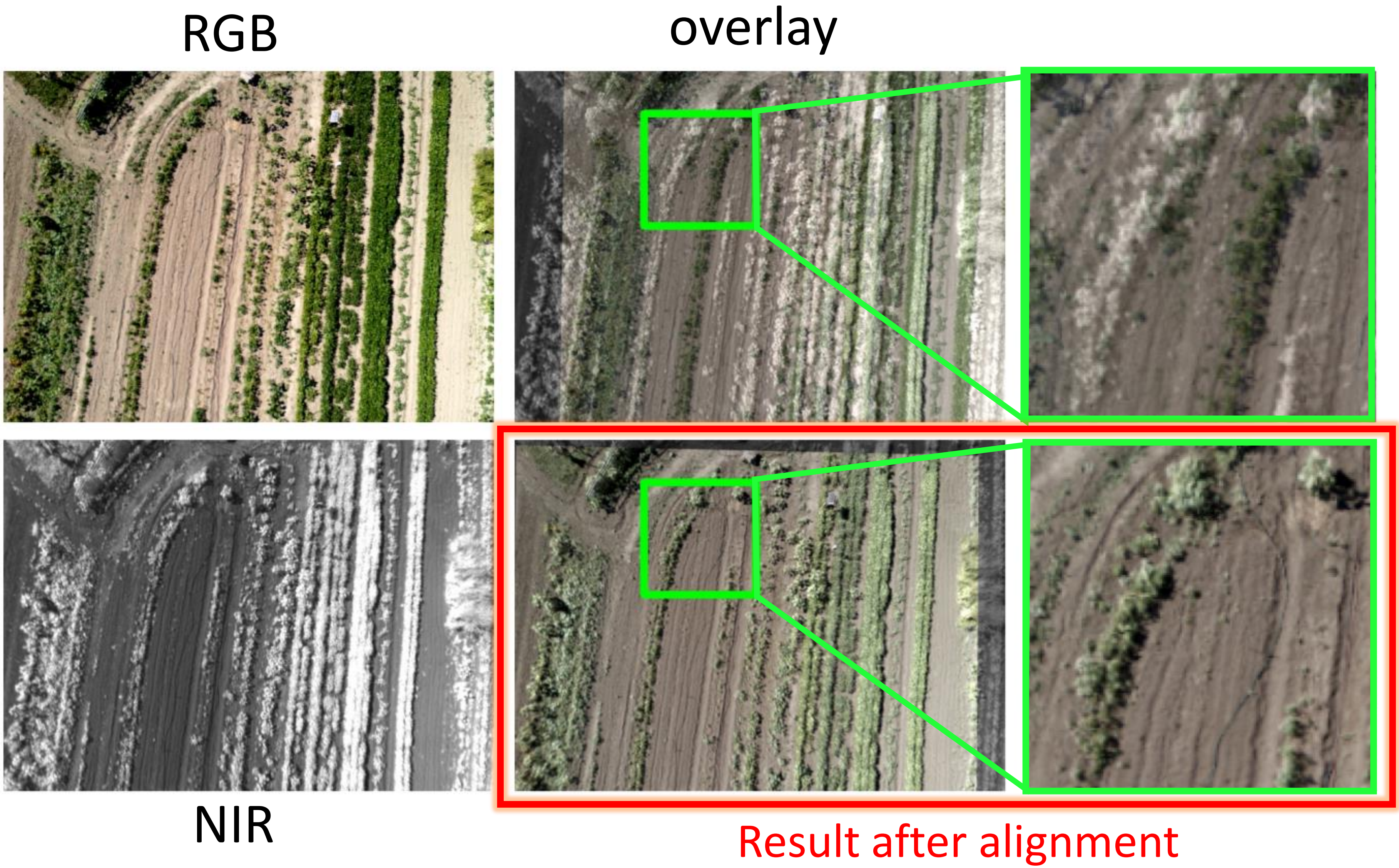
$$\boxed{\boldsymbol{L}_0(\mathbf{x};\psi,\theta)} = \|\boldsymbol{f}(\mathbf{x};\theta) - \boldsymbol{f}(\mathbf{w}(\mathbf{x};\psi);\theta)\|_2$$

$$\boldsymbol{L}_1(\mathbf{x},\mathbf{x}';\theta) = \max(\,0,\ \mu - \|\boldsymbol{f}(\mathbf{x};\theta) - \boldsymbol{f}(\mathbf{x}';\theta)\|_2)$$

Joint Optimization

$$\theta^*,\psi^* = \operatorname*{argmin}_{\theta,\psi}\left(\sum_{i=1}^{|\mathcal{P}|}\boldsymbol{L}_0(\mathbf{x}_i;\psi,\theta) + \sum_{j=1}^{|\mathcal{N}|}\boldsymbol{L}_1(\mathbf{x}_j,\mathbf{x}'_j;\theta)\right)$$

RGB-NIR image alignment



Summary of Results

- Learned RGB—NIR descriptor(s) perform better than existing descriptors.
- Competitive with supervised RGB descriptors.
  - Evaluated on HPatches [Balntas+ 2017].
- Robust to medium degree of initial misalignment
  - outperforms Mutual-Information (MI) methods.
- Bootstrapping:
  - Used our method to automatically obtain precise correspondences from multiple pairs.
  - Then, trained a supervised descriptor with improved generalization to new scenes.

Conclusions

- New extensions of Semi Global Matching (SGM)
  - Adding soft precomputed surface orientation priors.
  - Using learned strategy to fuse multiple proposals.

- Towards aligning images from scratch
  - Jointly trained a Siamese network and estimated a homography to align an image pair.
  - Weakly supervised local descriptor learning.
  - Extend to general scenes in the future.