Noise-Aware Unsupervised Deep Lidar-Stereo Fusion
– Supplementary Material –

*Xuelian Cheng[1,2], *Yiran Zhong[2,4,5], Yuchao Dai[1], Pan Ji[3], Hongdong Li[2,4]
[1]Northwestern Polytechnical University [2]Australian National University
[3]NEC Laboratories America, [4]ACRV, [5]Data61 CSIRO

Abstract

In this supplementary material, we provide our detailed network structure, qualitative comparison of hard and soft slanted plane constraint, qualitative and quantitative comparison to stereo matching algorithms and qualitative results on the Synthia dataset.

1. Detailed Network Structure

The core architecture of our LidarStereoNet contains three blocks: 1) Feature extraction and fusion; 2) Feature matching, and 3) Disparity computing. We provide the detailed structure of the feature extraction and fusion block in Table 1. The feature matching block and disparity computing block share the same structures with PSMnet [1].

2. Hard versus Soft Plane Fitting

There are two kinds of plane fitting constraints. Conventional CRF based methods use one slanted plane model to describe all disparities in one segment, *i.e.*, disparities insides one segment exactly obeys one slanted plane model. We term it as "Hard" plane fitting constraint. Our method, on the other hand, only applies this term as part of the whole optimization target. In other words, we only require the recovered disparities to fit a plane in a segment if possible but it can still be balanced by other loss terms.

Fig. 1 illustrates the difference between our soft constraint and the CRF-style hard constraint in a recovered disparity map. As can be seen in Fig. 1, strictly applying the slanted plane model in recovered disparity map decreases its performance from 3.27% to 3.97% and it is very sensitive to segments as well. By switching segments from Stereo SLIC to SLIC, its performance further decreases from 3.97% to 4.52%.

Table 1. Feature extraction and fusion block architecture, where **k**, **s**, **chns** represent the kernel size, stride and the number of the input and the output channels. We use "+" to represent feature concatenation.

Lidar feature extraction			
layer	**k , s**	**chns**	**input**
conv_s1	11×11, 1	1/16	disparity
conv_s2	7×7, 2	16/16	conv_s1
conv_s3	5×5, 1	16/16	conv_s2
conv_s4	3×3, 2	16/16	conv_s3
conv_s5	3×3, 1	16/16	conv_s4
conv_mask	1×1, 1	17/16	conv_s5+mask
Stereo feature extraction			
layer	**k , s**	**chns**	**input**
conv0_1	3×3, 2	3/32	image
conv0_2	3×3, 1	32/32	conv0_1
conv0_3	3×3, 1	32/32	conv0_2
conv1_n	$\begin{bmatrix} 3\times3, 1 \\ 3\times3, 1 \end{bmatrix} \times 3$	32/32	conv0_3
conv2_1	$\begin{bmatrix} 3\times3, 2 \\ 3\times3, 1 \end{bmatrix}$	$\begin{bmatrix} 32/64 \\ 64/64 \end{bmatrix}$	conv1_3
conv2_n	$\begin{bmatrix} 3\times3, 1 \\ 3\times3, 1 \end{bmatrix} \times 15$	64/64	conv2_1
conv3_1	$\begin{bmatrix} 3\times3, 1 \\ 3\times3, 1 \end{bmatrix}$	$\begin{bmatrix} 64/128 \\ 128/128 \end{bmatrix}$	conv2_16
conv3_n	$\begin{bmatrix} 3\times3, 1 \\ 3\times3, 1 \end{bmatrix} \times 2$	128/128	conv3_1
conv4_n	$\begin{bmatrix} 3\times3, 1 \\ 3\times3, 1 \end{bmatrix} \times 3$	128/128	conv3_3
branch1	64×64, 64	128/32	conv4_3
branch2	32×32, 32	128/32	conv4_3
branch3	64×16, 16	128/32	conv4_3
branch4	8×8, 8	128/32	conv4_3
lastconv	$\begin{bmatrix} 3\times3, 1 \\ 1\times1, 1 \end{bmatrix}$	$\begin{bmatrix} 320/128 \\ 128/32 \end{bmatrix}$	conv2_16+conv4_3 +branch1+branch2 +branch3+branch4
Feature fusion			
lastconv + conv_mask			

*These authors contributed equally in this work.

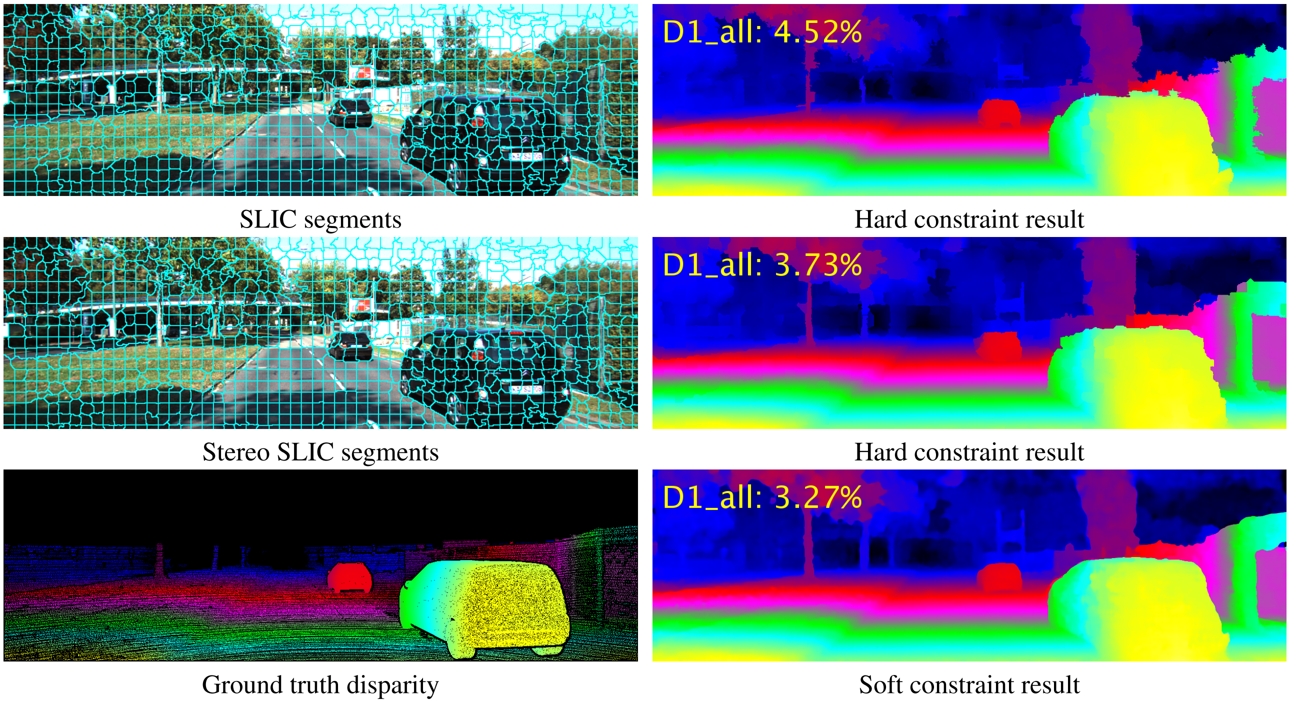


Figure 1. **Comparison of soft and hard constraints on slanted plane model with different superpixel segmentation methods.** Note that our recovered disparity map has more aligned boundaries with the color image.

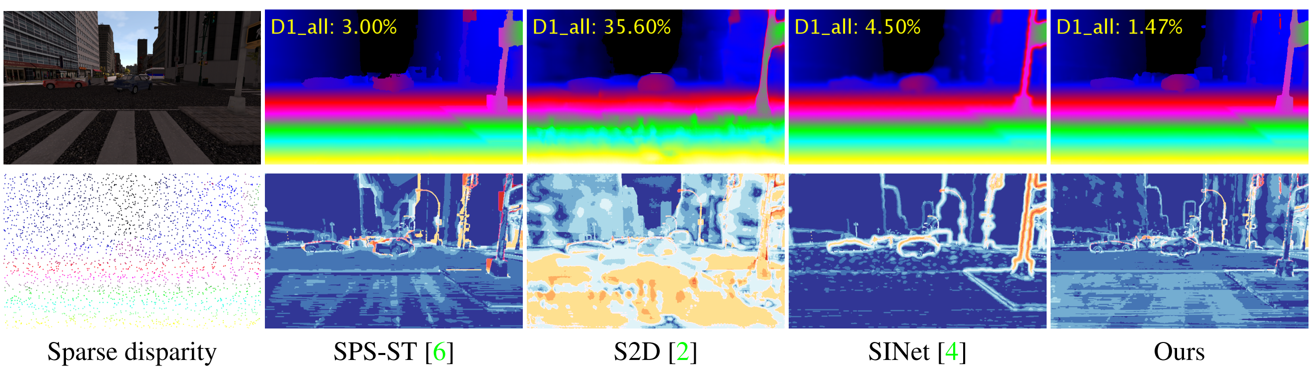


Figure 2. **Qualitative results on the Synthia dataset.** The first raw is the colorized disparity results, and the second row is the corresponding error maps.

3. Comparisons with STOA stereo matching methods

For the sake of completeness, we provide qualitative and quantitative comparisons with state-of-the-art stereo matching methods. We choose SPS-ST [6], MC-CNN[5], PSMnet [1] and SsSMnet[7] for reference. Note that the SPS-ST method is a traditional (non-deep) method, and its meta-parameters were tuned on KITTI dataset. For deep MC-CNN we used a model which was firstly trained on Middlebury dataset and for PSMnet we used the model that was trained on SceneFlow [3] dataset and the model ("-ft") that we fine-tuned on KITTI VO dataset. We also compared our method with state-of-the-art self-supervised stereo matching network SsSMnet [7].

4. Qualitative results on Synthia dataset

In Fig. 2, we show qualitative comparison results on Synthia dataset. Our method achieves the lowest bad pixel ratio.

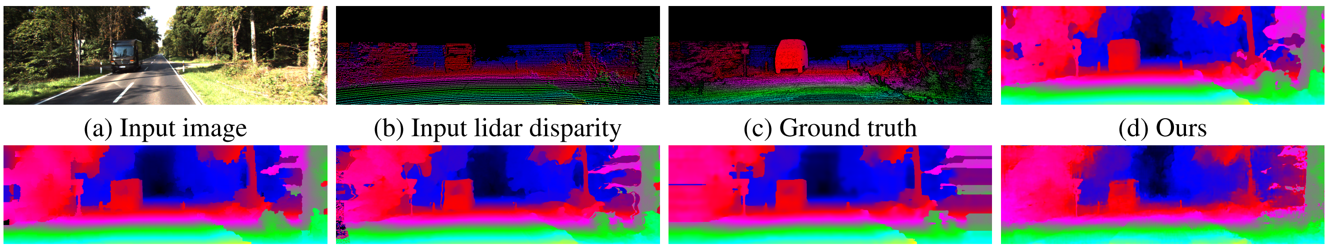
References

[1] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 5410–5418, 2018. 1, 2, 3

[2] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *IEEE International Conference on Robotics and Automation*, pages 1–8. IEEE, 2018. 2

Table 2. **Quantitative comparison on the selected KITTI 141 subset.** We compare our LidarStereoNet with various state-of-the-art stereo matching methods, where our proposed method outperforms all the competing methods with a wide margin.

Methods	Input	Supervised	Abs Rel	> 2 px	> 3 px	> 5 px	$\delta < 1.25$	Density
MC-CNN [5]	Stereo	Yes	0.0798	0.1070	0.0809	0.0555	0.9472	100.00%
PSMnet [1]	Stereo	Yes	0.0807	0.2480	0.1460	0.0639	0.9399	100.00%
PSMnet-ft [1]	Stereo	Yes	0.0609	0.0635	0.0410	0.0277	0.9689	100.00%
SPS-ST [6]	Stereo	No	0.0633	0.0702	0.0413	0.0265	0.9660	100.00%
SsSMnet [7]	Stereo	No	0.0619	0.0743	0.0498	0.0334	0.9633	100.00%
Our method	Stereo	No	**0.0572**	**0.0540**	**0.0345**	**0.0220**	**0.9731**	100.00%
Our method	Stereo + Lidar	No	**0.0350**	**0.0287**	**0.0198**	**0.0126**	**0.9872**	100.00%



(a) Input image (b) Input lidar disparity (c) Ground truth (d) Ours

(e)SPS-ST [6] (f) SsSMNet [7] (g) MC-CNN [5] (h) PSMnet [1]

Figure 3. **Qualitative results of the methods from Table 2.** Our method is trained on KITTI VO dataset and tested on the selected unseen KITTI 141 subset without any finetuning.

[3] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 4040–4048, 2016. 2

[4] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *International Conference on 3D Vision*, 2017. 2

[5] Jure Žbontar and Yann LeCun. Stereo matching by training a convolutional neural network to compare image patches. *J. Mach. Learn. Res.*, 17(1):2287–2318, Jan. 2016. 2, 3

[6] Koichiro Yamaguchi, David McAllester, and Raquel Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *Proc. Eur. Conf. Comp. Vis.*, pages 756–771. Springer, 2014. 2, 3

[7] Yiran Zhong, Yuchao Dai, and Hongdong Li. Self-supervised learning for stereo matching with self-improving ability. *arXiv preprint arXiv:1709.00930*, 2017. 2, 3