LoS: Local Structure-guided Stereo Matching

Supplementary Material

6. Monocular Depth Prior

As described in Sec.3.2, we refrain from using $\mathbf{D}_{mono}$ as the initial disparity map, and there are two main reasons. First, $\mathbf{D}_{mono}$, being estimated from a low-resolution image, lacks details of the scene. Second, despite alignment, $\mathbf{D}_{mono}$ still exhibits some scale mismatches. However, $\mathbf{D}_{mono}$ proves to be an effective prior for updating $\mathbf{D}_{raw}$ through the LSGP. Fig. 10 and Table 4 demonstrate that the incorporation of $\mathbf{D}_{mono}$ significantly enhances the accuracy of the initial disparity map. Models without LSGP, denoted as $\mathbf{D}_{raw}$, and those utilizing only local relations in LSGP of initialization step, indicated as $\mathbf{D}^0$ (w/o $\mathbf{D}_{mono}$), both exhibit a considerable decrease in performance.

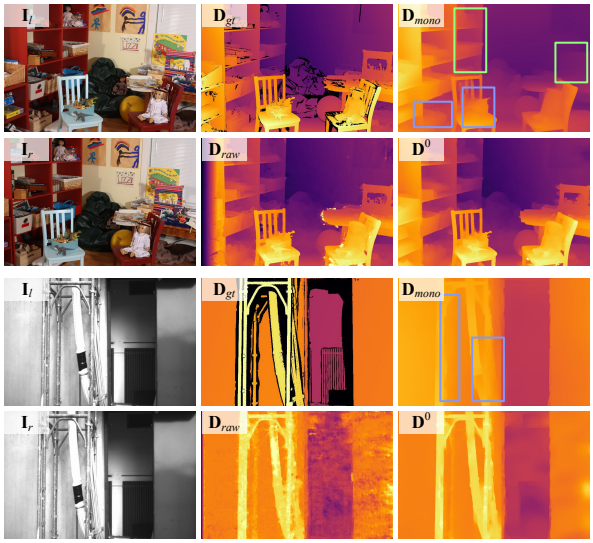

Figure 10. Illustration of depth prior. $\mathbf{D}_{mono}$ usually lacks of object details (blue boxes) and containing scale mismatches (green boxes). However, comparing $\mathbf{D}_{raw}$ and $\mathbf{D}^0$, we can find that $\mathbf{D}_{mono}$ serves as a good prior when working with LSGP.

	Middlebury		ETH3D	
	Bad2.0	AvgErr	Bad1.0	AvgErr
$\mathbf{D}_{raw}$	26.8	8.719	17.5	0.999
$\mathbf{D}^0$ (w/o $\mathbf{D}_{mono}$)	48.5	6.062	10.5	0.582
$\mathbf{D}^0$	25.3	3.834	8.7	0.513

Table 4. Comparison between $\mathbf{D}_{raw}$ and $\mathbf{D}^0$. With the guidance from $\mathbf{D}_{mono}$, LoS gets a much better initial disparity.

7. Experiment Details

7.1. Data Augmentation

We adopt the data augmentation techniques introduced in CREStereo [18] as our standard data augmentation process. Initially, asymmetric chromatic augmentations are applied separately to each image of a stereo pair, encompassing adjustments in brightness, contrast, and gamma. Subsequently, the left image undergoes several spatial augmentations, which include a slightly random homography transformation and a small vertical shift (less than 2 pixels). This is followed by random resizing and cropping to standardize the input sizes of the stereo pairs. Lastly, to enhance the model's capability in handling occluded areas, a rectangular section is randomly masked in the right image.

During the fine-tuning phase for KITTI and RVC models, we utilize a considerably simpler augmentation strategy, as outlined in RAFT-Stereo [20]. This change is due to the sparse nature of the disparity ground truth in the KITTI datasets. The process begins with the same asymmetric chromatic augmentations. Then, the stereo pairs undergo random resizing and cropping to achieve uniform input sizes. Finally, random masking is applied to the right images.

7.2. Settings of Efficiency Evaluation

We conduct the time consumption comparison in Table 2, and we introduce the detailed default settings of each model here.

RAFT-Stereo [20]. The validation iteration numbers are set to 32, the correlations are sampled from a 4-level correlation pyramid, and the sample radius is set to 4.

CREStereo [18]. The validation iteration numbers are set to 20, the 2-level cascaded framework is adopted, and 2D-1D alternative search strategy is adopted with 9 correlation candidates.

IGEV [46]. The validation iteration numbers are set to 32, the correlations are sampled from a 2-level CGEV pyramid, and the sample radius is set to 4.

LoS (Ours). The validation iteration numbers N, N_1, N_2 are set to 10, 64, 4 respectively, and 2D-1D alternative search strategy is adopted with 9 correlation candidates.

7.3. Settings of Challenging Areas Evaluation

We assess the performance of our model in challenging areas using the UnrealStereo4K dataset [38] in Table 3. Below, we provide the specifics of our experimental setup.

Datasets. There are 8 scenes in the UnrealStereo4K [38] dataset. However, the number of samples in each scene are not balanced. Therefore, we randomly choose 200 samples

from the dataset, allocating 25 samples from each scene, to construct our evaluation dataset.

Mask Generation. Class 1 and class 2 masks are generated straightforwardly based on the dense disparity ground truth. To generate the class 3 mask, we calculate the Structural Similarity Index (SSIM) between the original image and the image shifted by one pixel. Specifically, we horizontally and vertically shift the original images by one pixel and compute SSIM values for each pixel within a 33×33 patch. Pixels in textureless areas are identified where the SSIM value exceeds 0.95. To maintain image resolution during class 3 mask generation, we employ reflection padding. For the class 4 mask, we designate pixels with disparity gradients exceeding 5 as belonging to edge areas, *i.e.* $\sqrt{\|\mathbf{D}(x,y)-\mathbf{D}(x+1,y)\|^2+\|\mathbf{D}(x,y)-\mathbf{D}(x,y+1)\|^2} \geq 5$.

8. Additional Experiments

8.1. Ablation Study

The models in Table 5 are trained with a batch size of 8 for $150k$ steps, and the training set consists of the BTS, Middlebury and ETH3D, while other settings remained the same as Sec. 4.1.

LSGP. Table 5 demonstrates that the LSGP markedly enhances model performance. When LSGP is employed for initialization, updating, and refinement phases, the metric bad2.0 on the Middlebury dataset is reduced by 26.6%, 35.5%, and 7.6%, respectively (comparing models 1, 2, 3 with model 0). Conversely, omitting LSGP from these stages results in an increase in the bad2.0 metric by 5.4%, 16.7%, and 0.4% respectively (comparing models 4, 5, 6 with model 10). LSGP facilitates the propagation of useful information on the basis of GRU, and GRU in turn augments LSGP's efficacy through the updating of LSI. Thus, the most substantial performance gains are attributed to the use of LSGP in disparity updating. Although the improvements from LSGP in the refinement phase are modest, we maintain this step in the pipeline due to the efficiency and non-destructive integration of LSGP.

Loss Functions. Supervision of the LSI significantly improves model performance. With $\mathcal{L}_o$ providing supervision to both $\mathbf{G}$ and $\mathbf{O}$, the enhancement in performance is more pronounced. However, the marginal improvements offered by $\mathcal{L}_g$ are attributable to its inability to fully capture structural details, resulting in an elevated bad1.0 metric on the ETH3D dataset. The model attains optimal performance when it is under the combined supervision of $\mathcal{L}_o$ and $\mathcal{L}_g$.

Iteration Steps. We conduct an ablation experiment on the iteration steps. As shown in Table 6, increasing the iterations of LSGP (N_1 and N_2) is a more effective and efficient way to improve the model performance than increasing the iterations of GRU (N). However, as discussed in Sec. 4.2, due to the lack of constraints on $\mathbf{O}$ in the LoS model, unlimited increasing N_2 may cause a rapid performance drop, see models s20 and s21.

No.	LSGP			Loss		Middlebury		ETH3D	
	I	U	R	$\mathcal{L}_o$	$\mathcal{L}_g$	Bad2.0	AvgErr	Bad1.0	AvgErr
0						15.84	2.66	3.40	0.29
1	✓			✓	✓	11.63	1.97	1.94	0.21
2		✓		✓	✓	10.21	1.57	1.74	0.21
3			✓	✓	✓	14.63	2.45	2.65	0.25
4		✓	✓	✓	✓	10.17	1.57	1.73	0.21
5	✓		✓	✓	✓	11.26	1.92	1.90	0.21
6	✓	✓		✓	✓	9.69	1.79	1.67	0.20
7	✓	✓	✓			13.82	2.72	2.18	0.29
8	✓	✓	✓	✓		10.68	1.79	1.70	0.20
9	✓	✓	✓		✓	11.99	1.90	3.25	0.24
10	✓	✓	✓	✓	✓	9.65	1.79	1.67	0.20

Table 5. Ablation study. 'I' stands for LSGP in initialization step (Sec. 3.2.2 and Fig. 3 (c)), 'U' stands for LSGP in disparity updating (Sec. 3.3.1 and Fig. 3 (d)) and 'R' refers to LSGP used in refinement (Sec. 3.3.2 and Fig. 3 (e)).

No.	N	N_1	N_2	ETH3D			Middlebury		
				Bad1.0	AvgErr	time	Bad2.0	AvgErr	time
s1	2			4.2	0.374	**0.134**	22.0	6.893	**0.481**
s2	5†			3.6	0.322	0.175	20.3	6.497	0.663
s3	10‡			**3.1**	**0.305**	0.257	19.6	5.908	1.008
s4	15	64‡	4†‡	3.2	0.313	0.331	**19.4**	**5.678**	1.334
s5	20			**3.1**	**0.305**	0.405	19.7	5.689	1.663
s6	25			3.2	0.311	0.482	19.9	5.765	1.995
s7	30			**3.1**	0.309	0.559	20.2	5.821	2.320
s8		16		3.4	0.363	0.249	20.0	7.474	**0.987**
s9		32†		3.1	0.322	**0.246**	19.8	6.615	0.989
s10		48		3.1	0.317	0.250	**19.6**	6.564	0.995
s11	10‡	64‡	4†‡	3.1	**0.305**	0.257	**19.6**	5.908	1.008
s12		128		3.0	0.284	0.267	19.9	5.297	1.020
s13		192		**2.9**	0.277	0.280	20.2	**5.057**	1.037
s14		256		**2.9**	0.275	0.293	20.3	5.117	1.062
s15			2	3.1	0.319	**0.249**	19.8	6.123	**0.992**
s16			4†‡	3.1	0.305	0.257	19.6	5.908	1.008
s17			6	3.1	0.308	0.254	**19.2**	5.746	1.017
s18	10‡	64‡	8	**3.0**	0.298	0.259	**19.2**	5.790	1.056
s19			16	3.1	**0.294**	0.268	19.6	5.301	1.160
s20			24	3.3	0.298	0.284	20.5	5.196	1.258
s21			32	3.8	0.310	0.302	22.0	**5.161**	1.345

Table 6. Ablation Study on Iteration Steps. For N, N_1 and N_2, the training settings and default test settings are indicated with † and ‡ respectively. The best results are denoted with **bold**.

	ETH3D	Middlebury	KITTI 15	KITTI 12
	Bad1.0	Bad2.0	Bad3.0	Bad3.0
MiDaS + scale-shift (original)	3.1	19.6	5.5	4.4
MiDaS + median	3.1	**19.3**	5.5	4.4
ZoeDepth + scale-shift	**2.9**	19.5	**5.4**	**4.3**
ZoeDepth + median	3.0	20.0	5.4	4.4

Table 7. Depth Prior Comparison with Domain Generalization Task.

Depth Prior. For LSI initialization, we evaluate four combinations in Table 7. ZoeDepth [2] provides better LSI initialization than MiDaS while scale-shift alignment and median alignment achieves similar performance. However, we con-

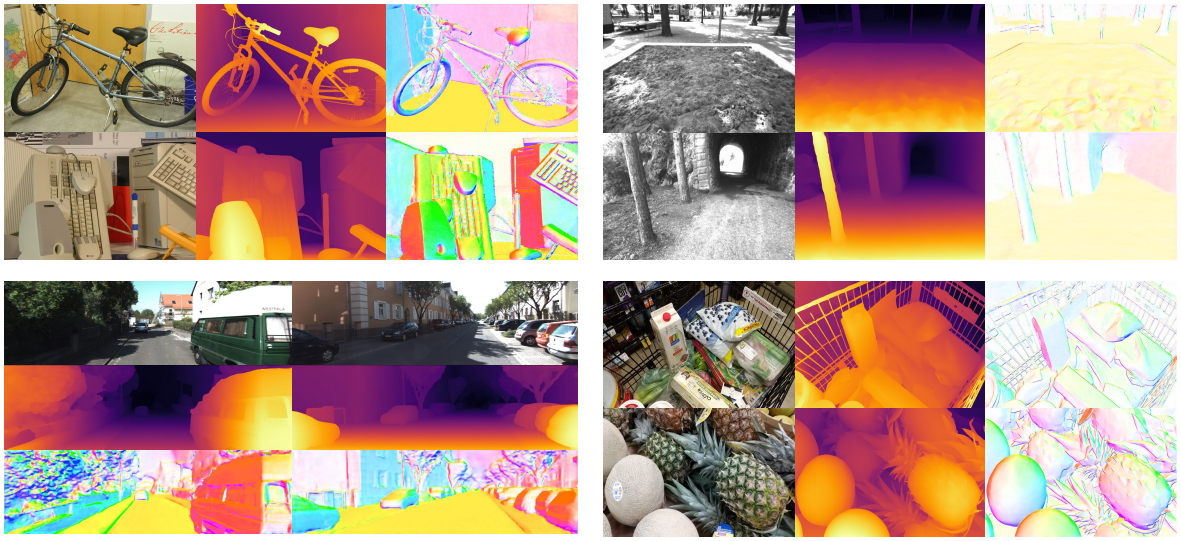

Figure 11. Quantitative results. We show the left image, estimated disparity map and disparity gradient map for each sample.

	ETH3D	Middlebury	KITTI 15	KITTI 12
	Bad1.0	Bad2.0	Bad3.0	Bad3.0
PSMNet [8]	23.8	39.5	16.3	15.1
DSMNet [50]	6.2	21.8	6.5	6.2
CFNet [31]	5.8	28.2	5.8	4.7
RAFT-Stereo [20]	3.3	18.3	5.8	4.7
CREStereo [18]	5.5	**15.3**	6.7	6.7
LoS (Ours)	**3.1**	19.6	**5.5**	**4.4**

Table 8. Domain Generalization Results.

duct this experiment after we submitting benchmarks results, and due to limitations specified in the benchmarks, we report the benchmark results based on *MiDaS + scale-shift* in our main paper.

8.2. Domain Generalization

We evaluate LoS's domain generalization ability, which is to generalize from a synthetic training dataset to unseen real-world test datasets. We train the LoS model on only SceneFlow [24] with data augmentation, and test the model on ETH3D [30], Middlebury [29], KITTI 2015 [25] and KITTI 2012 [11]. As shown in Table 8, our LoS achieves the best overall performance, which demonstrates the robustness and superiority of local structure guidance.

9. Visualization

We show more results on ETH3D, Middlebury, KITTI 12 and Holopix 50k in Fig. 11. Our LoS achieves satisfying result on cross datasets and scenes.