Supplementary Material: Do End-to-end Stereo Algorithms Under-utilize Information?

Anonymous 3DV submission

Paper ID 108

In this supplementary material, we show more details of the network architectures, inference times, and experimental results, some of which are mentioned but not fully discussed in the main paper due to the page limit.

1. Network Details and Runtime

Backbone architectures. The backbones in our architecture are state-of-the-art 2D and 3D CNNs for stereo matching, including DispNetC [3], GCNet [2], PSMNet [1] and GANet [4]. Each convolutional layer is followed by a batch normalization (BN) layer and a ReLU layer, with the following exceptions, as specified in the corresponding papers: i) In DispNetC and our variants, there is no BN layer after the six convolution layers (i.e., disp6 to disp1) which output disparity maps. ii) In GCNet and our variants, there is no BN layer or ReLU after the last 2D convolutional layer (i.e., conv18) in the feature extraction module, and after the last 3D convolutional layer (i.e., conv37) in the cost volume regularization module.

Runtime In Table 1, we compare the GPU memory consumption and runtime in inference mode on pairs of frames with dimension 384×1280. All experiments are run on the same machine, with the same configuration of disparity range $D = 192$, filter window size $s = 5$ and dilation rate $r = 2$.

2. Ablation Study and Additional Quantitative Results

We perform ablation studies to investigate how the filter window s and the dilation rate r can affect the filtering output and disparity estimation. Out of a large number of possible combinations, we show two representatives *DispNetC+SABF* and *PSMNet+SABF* in Table 2. We find that $s = 5$ with $r = 2$ achieve a good balance in accuracy, space and runtime. The 500-run averaged memory consumption and runtime in GPUs are measured when we test a 384×1280 stereo pair, and the *bad-3* (noc,all) errors are evaluated on the KITTI 2015 validation set. Please note that a 5×5 filter (with dilation rate 2) covers 9×9 regions in the cost feature space, which is equivalent to 33×33 regions in the RGB image space, due to the cost volume being a quarter of the size of the input images. In the following experiments, we keep using $s = 5$ with $r = 2$ for our different architectures.

In Table 4, we further investigate the effectiveness among those variants in terms of parameter increase (column $\delta P\%$) and error decrease (column $\delta E\%$) evaluated on the KITTI 2015 validation set, as shown in Table 3[1].

3. Qualitative Results

In Figs. 1–4, we show reference images and disparity maps generated by each backbone without modifications and the same backbone after integrating one of the filtering techniques.

References

[1] J.-R. Chang and Y.-S. Chen. Pyramid stereo matching network. In *CVPR*, pages 5410–5418, 2018. 1

[2] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry. End-to-end learning of geometry and context for deep stereo regression. In *ICCV*, pages 66–75, 2017. 1

[3] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, pages 4040–4048, 2016. 1

[4] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *CVPR*, 2019. 1

[1] Table 3 is originally included in the main paper. We repeat it here to explain the effectiveness in Table 4.

Filters	DispNetC		PSMNet		GANet		GCNet	
	Mem.(MiB)	Time (ms)	Mem.(MiB)	Time (ms)	Mem.(MiB)	Time (ms)	Mem.(MiB)	Time (ms)
W/O	**1394**	**18.35**	**5151**	**315.57**	**7178**	**1894.70**	**4280**	**146.83**
SABF	1888	24.32	5386	563.42	7920	2488.72	4424	379.37
DFN	1422	28.33	5246	432.32	7466	2041.53	4298	255.20
PAC	1535	25.34	5168	514.91	8274	2383.44	4400	334.73
SGA	7066	489.60	11070	823.00	-	-	9916	655.18

Table 1: Runtime (ms) and GPU memory consumption (MiB). Results are shown in the entries of filters (rows in skyblue) and backbones (columns) w.r.t. the baselines (rows W/O). The largest values are in bold. GANet already contains SGA, resulting in blank entries "-".

filter size	DispNetC+SABF						PSMNet+SABF					
	$r=1$		$r=2$		Mem.	Time	$r=1$		$r=2$		Mem.	Time
s	EPE(px)	$\geq$3(%)	EPE(px)	$\geq$3(%)	(MiB)	(ms)	EPE(px)	$\geq$3(%)	EPE(px)	$\geq$3(%)	(MiB)	(ms)
$s=3$	0.875	3.14	0.845	2.99	2132	45.26	0.639	1.63	0.643	1.61	4681	449.40
$s=5$	0.867	3.13	0.841	2.90	2228	48.99	0.657	1.54	0.630	1.46	4989	633.42
$s=7$	0.832	2.83	0.795	2.46	2588	54.21	0.650	1.50	0.642	1.54	4709	939.40
$s=9$	0.825	2.84	0.854	3.00	3008	60.56	0.868	1.77	0.689	1.91	4953	1226.72

Table 2: Illustration of the effects of different filter window sizes s and dilation rates r. We computer the bad-3 (noc,all) errors on the KITTI 2015 validation set and the averaged GPU memory consumption and runtime to test a pair of frames with dimension 384×1280.

Filters	DispNetC		PSMNet		GANet		GCNet	
	noc	all	noc	all	noc	all	noc	all
W/O	2.59	3.02	1.46	1.60	**0.97**	**1.10**	2.06	2.64
SABF	2.26	2.63	1.28	1.40	1.07	1.17	1.76	2.10
DFN	2.37	2.78	1.23	1.34	0.99	1.11	1.70	2.08
PAC	2.38	2.72	1.29	1.48	1.13	1.23	1.71	2.03
SGA	**1.90**	**2.18**	**1.17**	**1.32**	-	-	**1.69**	**1.91**

Table 3: KITTI 2015 bad-3 validation results. Improved results are highlighted in gray, and best ones are in bold. GANet contains SGA, resulting in blank entries "-".

Filters	DispNetC		PSMNet		GANet		GCNet	
	δE(%)	δP(%)	δE(%)	δP(%)	δE(%)	δP(%)	δE(%)	δP(%)
SABF	12.9	4.2	12.4	34	-5.9	27	20.6	62.4
DFN	7.9	0.8	16.2	6.4	-0.1	5.1	21.5	11.8
PAC	9.9	0.1	7.8	2.0	-12	1.6	23.3	3.6
SGA	27.8	7.0	17.7	58.8	-	-	27.7	108

Table 4: Effectiveness comparison on the KITTI 2015 val-30 dataset. For each combination of network backbone and filtering, columns $\delta E(\%)$ and $\delta P(\%)$ indicate the relative decrease of error and increase of the number of parameters, respectively, w.r.t. the backbone baselines.

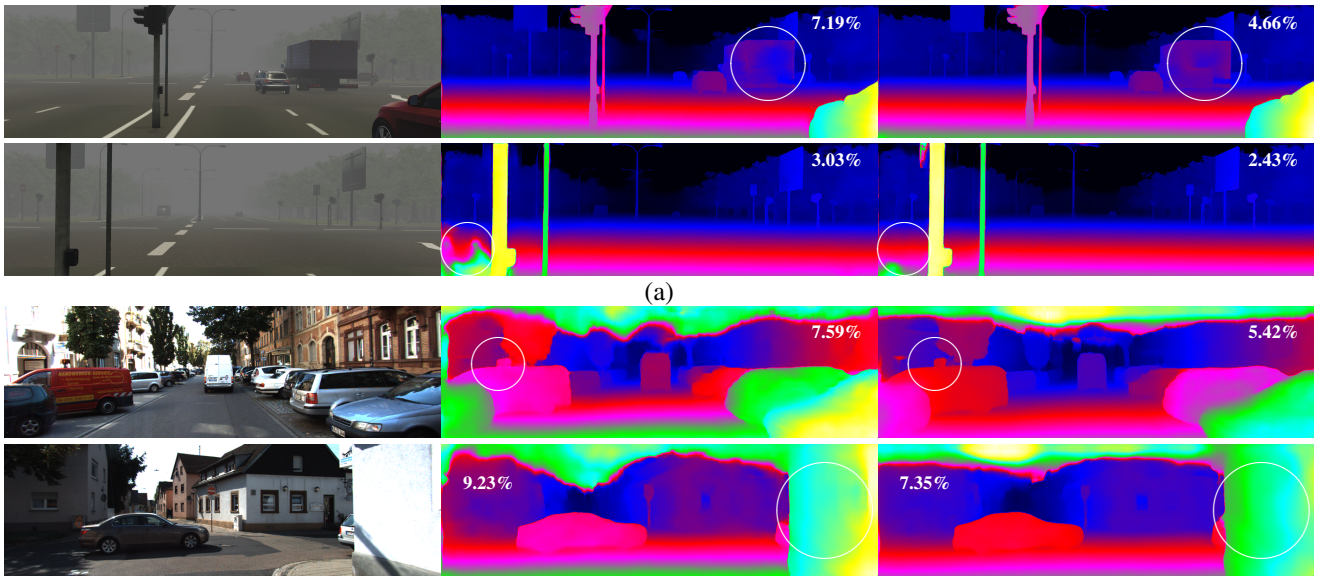


Figure 1: Results using DispNetC as backbone. (a) DispNetC vs DispNetC+PAC on Virtual KITTI 2 Scene06 validation set. (b) DispNetC vs DispNetC+SABF on KITTI 2015 validation set. In all rows, the left image is the reference image of the stereo pair, the middle column in the disparity map from the unmodified backbone, and the right image is the disparity map of the backbone with the integrated filter.

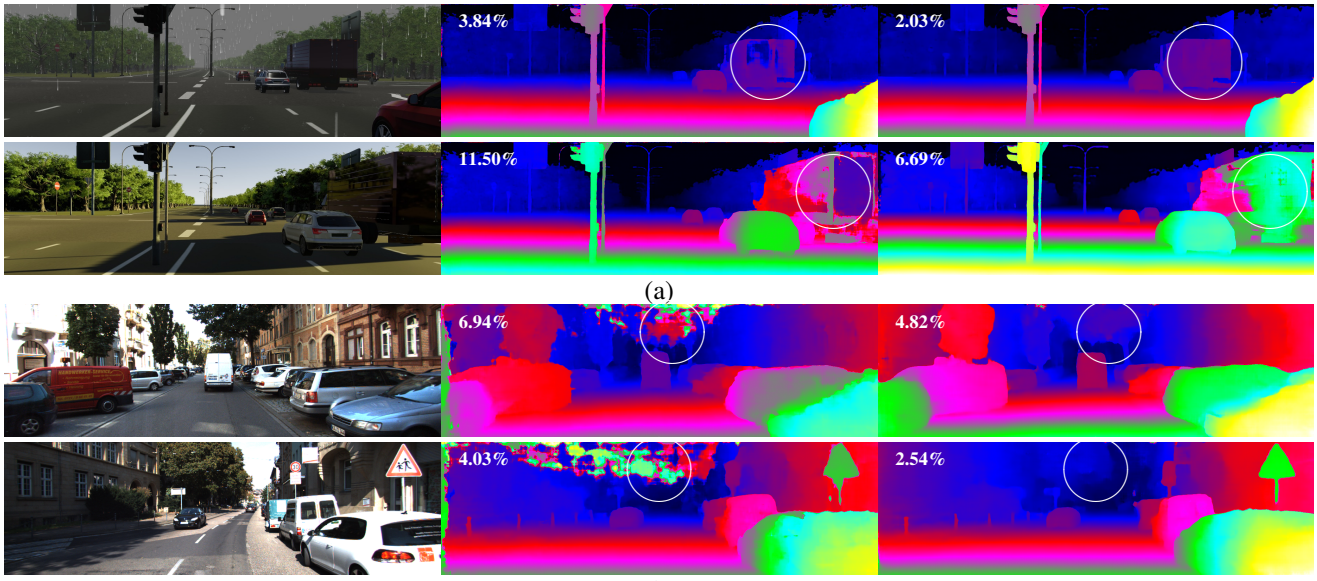
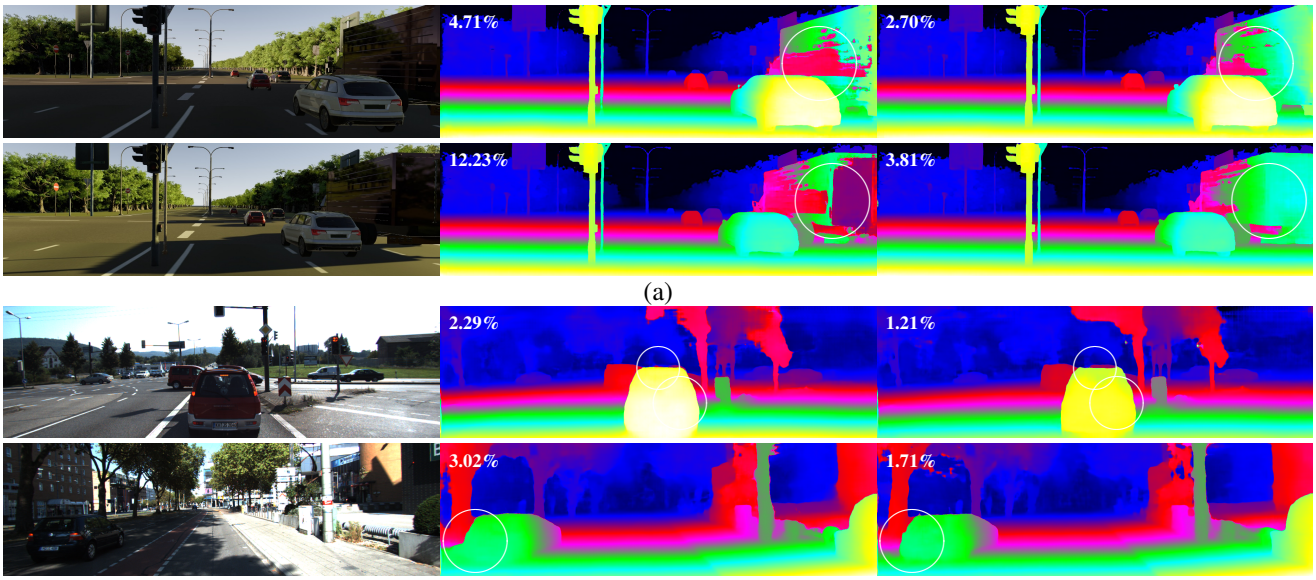


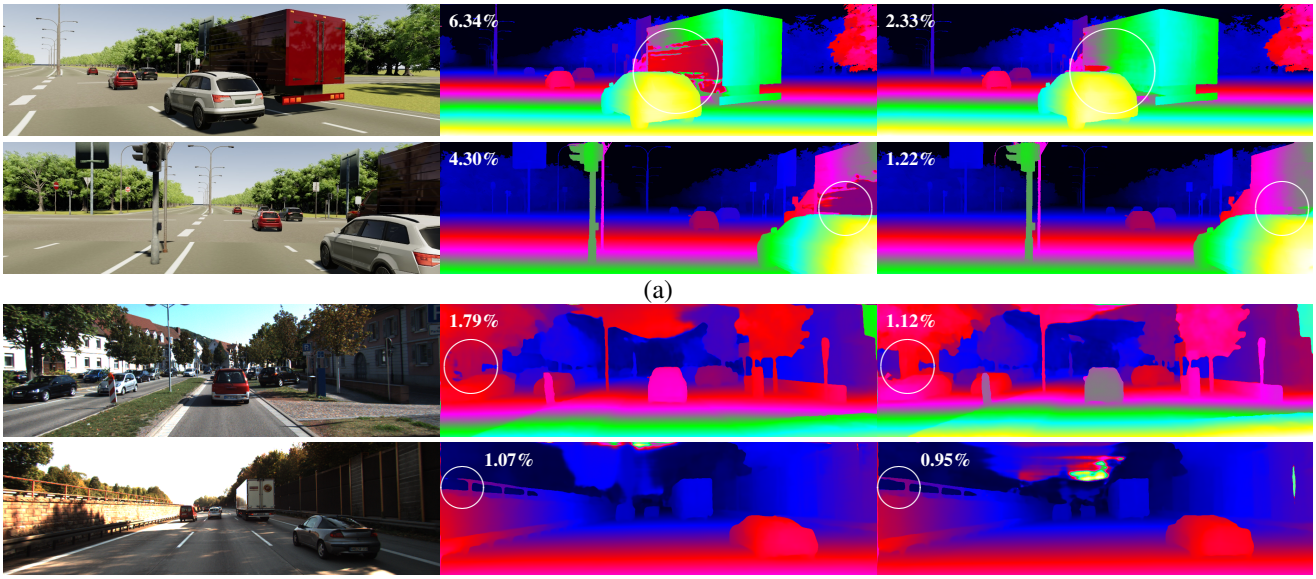
Figure 2: Results using GCNet as backbone. (a) GCNet vs GCNet+SGA on Virtual KITTI 2 Scene06 validation set. (b) GCNet vs GCNet+SGA on KITTI 2015 validation set. In all rows, the left image is the reference image of the stereo pair, the middle column in the disparity map from the unmodified backbone, and the right image is the disparity map of the backbone with the integrated filter.



(a)

(b)

Figure 3: Results using PSMNet as backbone. (a) PSMNet vs PSMNet+DFN on Virtual KITTI 2 Scene06 validation set.(b) PSMNet vs PSMNet+PAC on KITTI 2015 validation set. In all rows, the left image is the reference image of the stereo pair, the middle column in the disparity map from the unmodified backbone, and the right image is the disparity map of the backbone with the integrated filter.



(a)

(b)

Figure 4: Results using GANet as backbone. (a) GANet vs GANet+SABF on Virtual KITTI 2 Scene06 validation set. (b) GANet vs GANet+PAC on KITTI 2015 validation set. In all rows, the left image is the reference image of the stereo pair, the middle column in the disparity map from the unmodified backbone, and the right image is the disparity map of the backbone with the integrated filter.