Robust Synthetic-to-Real Transfer for Stereo Matching

Supplementary Material

Overview

We organize the material as follows. Section A shows more details of the experiment settings. In Section B, we provide comparison results between using GT and PL for fine-tuning with the additional stereo matching network architecture. Section C and Section D conducts additional experiments about DKT. Section E presents more qualitative results of the domain generalization performance.

A. Details of Experimental Setting

Dataset. We conduct our experiments by initializing stereo matching networks with synthetic dataset pre-trained weights and fine-tuning them in real-world scenarios. We mainly focus on the robustness and domain generalization ability after fine-tuning networks, and we also show their target-domain performances to ensure networks actually learn from target domains. When not specifically mentioned, all networks in our experiments are pre-trained on the SceneFlow [5] dataset. The introduction of the synthetic and real-world datasets are as follows:

- *SceneFlow* [5] is a large synthetic dataset that consists of 35,454 pairs of stereo images for training and 4,370 pairs for evaluation. Both sets have dense ground-truth disparities. The resolution of the images is 960×540. Besides the original clean pass, the dataset also contains a final pass. The final pass has motion blur and defocus blur, making it more similar to real-world images. SceneFlow is currently the most commonly used dataset for pre-training stereo matching networks.
- *KITTI 2012* [2] collects outdoor driving scenes with sparse ground-truth disparities. It contains 194 training samples and 195 testing samples with a resolution of 1226×370.
- *KITTI 2015* [6] collects driving scenes with sparse disparity maps. It contains 200 training samples and 200 testing samples with a resolution of 1242×375.
- *Booster* [7] contains 228 samples for training and 191 samples for online testing in 64 different scenes with dense ground-truth disparities. Most of the collected scenes have challenging non-Lambertian surfaces. We use the quarter resolution in our experiments.
- *Middlebury* [8] consists of 15 training and 15 testing stereo pairs captured indoors. The dataset offers images at full, half, and quarter resolutions. We use the half-resolution training set for domain generalization evaluation.
- *ETH3D* [9] consists of 27 grayscale image pairs for training and 20 for testing. It includes both indoor and outdoor scenes. We use the training set for domain generalization evaluation.
- *DrivingStereo* [15] is a large-scale real-world driving dataset. A subset of it contains 2,000 stereo pairs collected under different weather (sunny, cloudy, foggy, and rainy). We use the half resolution of these challenging scenes to evaluate the robustness after fine-tuning on the KITTI datasets.

Local Dataset Split. Except for online submissions, we conduct the experiments based on local train and validation splits. For the KITTI 2012 and 2015 datasets, we follow GWCNet [3] to split 14 stereo pairs of 2012 and 20 pairs of 2015 for validation, the remaining 360 pairs are used for training. For the booster dataset, we use the 'Washer' and 'OilCan' scenes (15 stereo pairs) for validation, and the remaining 213 pairs for training. In this material, we also conduct fine-tuning experiments on Middlebury and ETH3D datasets, following the data split in [4]. We use the 'ArtL' and 'Playroom' scenes (2 stereo pairs) for Middlebury validation and the 'facade' and 'forest' scenes (3 stereo pairs) for ETH3D validation.

B. Network Architecture for GT vs. PL

In the main paper, we investigate the distinct behaviors of Ground Truth (GT) and Pseudo Label (PL) during fine-tuning. We achieve this by dividing pixels into different regions ($X_c(\tau)$, $X_{inc}(\tau)$, $X_{invalid}$) and conducting comprehensive comparisons between them. In addition to the iterative optimization based IGEV-Stereo [12], we employ the 3D convolution-based CFNet [10] to affirm that our findings are applicable across diverse stereo matching network architectures. As presented in Table I, learning new knowledge without sufficient regularization and overfitting GT details are two primary contributors to the degradation of domain generalization ability during the fine-tuning.

C. Additional Ablations about DKT

C.1. Fine-grained Permutations

In F&E-GT, we leverage the exponential moving average (EMA) Teacher's prediction to serve as fine-grained permutations for GT. In this section, we present ablations with alternative permutations. Specifically, we apply F&E-GT using the EMA Teacher for filtering out inconsistent regions but with variations in permutations. We use random noise from (-1, 1), PL from the frozen Teacehr, and the EMA Teacher's prediction for ablation and visualize the three kinds of fine-grained permutations in Figure I. The

Supervision	2012	2015	Midd	ETH3D	Booster
Training set	**KITTI 2012 & 2015**				
zero-shot	5.71	4.84	15.77	5.48	38.84
GT(valid)	**2.15**	**1.39**	19.83	29.94	30.95
GT($X_c(3)$)	2.26	1.67	17.92	24.52	30.88
GT($X_{inc}(3)$)	21.33	18.49	31.78	58.35	43.06
GT($X_c(1)$)	2.67	1.95	16.27	14.67	31.16
PL(all)	5.05	4.26	**13.71**	**4.86**	**29.92**
PL(valid)	5.58	4.64	14.78	6.05	30.79
PL($X_c(3)$)	3.32	3.01	14.09	5.50	30.97
PL($X_{inc}(3)$)	8.91	8.08	18.30	12.45	40.17
PL($X_c(1)$)	2.94	2.57	15.38	5.80	31.34
Training set	**Booster**				
zero-shot	4.97	6.31	15.77	5.48	35.03
GT(valid)	56.20	71.41	18.45	80.53	**25.86**
GT($X_c(3)$)	4.39	6.04	13.53	20.90	26.85
GT($X_{inc}(3)$)	97.27	97.86	77.44	99.89	45.69
GT($X_c(1)$)	4.31	6.19	13.77	21.80	26.13
PL(all)	4.24	5.19	11.38	5.42	28.34
PL(valid)	4.33	5.11	11.48	5.51	28.05
PL($X_c(3)$)	3.98	**4.65**	11.25	**5.39**	27.40
PL($X_{inc}(3)$)	5.56	7.68	16.81	6.08	36.44
PL($X_c(1)$)	**3.79**	4.98	**11.03**	5.59	27.54

Table I. Results of using different regions of GT or PL to fine-tune CFNet [10]. Different regions play varied roles during fine-tuning.

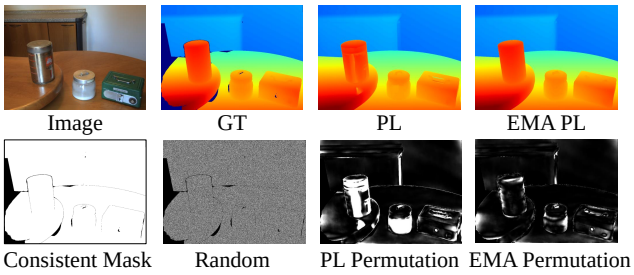

Figure I. Visualization with the absolute value of three kinds of fine-grained permutations.

results are presented in Table II. Our findings demonstrate that employing the frozen Teacher or EMA Teacher to add permutations better preserves domain generalization ability than random noise. Moreover, utilizing the EMA Teacher yields better target-domain performance compared to the frozen Teacher. We attribute this improvement to the EMA Teacher progressively predicting more accurate disparities.

Method	2012	2015	Midd	ETH3D	Booster
Training set	**KITTI 2012 & 2015**				
random noise	1.94	1.39	11.08	19.79	17.93
F.T.	1.94	1.40	**9.60**	12.34	17.57
EMA.T.	**1.93**	**1.38**	9.62	**12.31**	**17.46**
Training set	**Booster**				
random noise	11.55	12.28	9.54	7.73	12.85
F.T.	9.48	**10.96**	7.81	**5.93**	12.89
EMA.T.	**9.42**	11.04	**7.56**	6.11	**12.76**

Table II. Ablation of fine-grained permutations. F.T.: the frozen Teacher. PL serves as better permutations than random noise.

C.2. Effects of the Frozen Teacher

An alternative way to using the frozen Teacher's prediction with F&E-PL is directly using the EMA Teacher's prediction, which progressively predicts more accurate disparities. An overview of DKT without the frozen Teacher is shown in Figure II. We show the comparison in Table III. Using the frozen Teacher's prediction gets improvements in target domains than using the frozen Teacher's prediction with F&E-PL, however, it leads to a slight drop in domain generalization ability than using the Frozen Teacher's prediction.

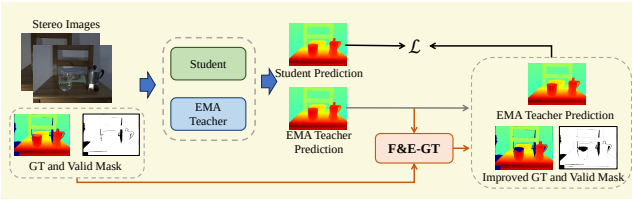

Figure II. DKT framework without the frozen Teacher.

Method	2012	2015	Midd	ETH3D	Booster
Training set	**KITTI 2012 & 2015**				
DKT(w/o F.T.)	**1.97**	**1.34**	8.02	4.43	15.65
DKT(full)	1.98	1.39	**7.11**	**3.64**	**15.51**
Training set	**Booster**				
DKT(w/o F.T.)	3.72	4.86	7.00	3.89	**12.23**
DKT(full)	**3.49**	**4.71**	**6.61**	**2.60**	12.63

Table III. Effects of the using frozen Teacher to produce PL. F.T.: the frozen Teacher. Using the frozen Teacher to produce PL preserves the domain generalization ability better.

D. Additional Experiments about DKT

D.1. DKT vs. DG Methods

Training robust stereo matching networks on synthetic datasets has been well-researched recently [1, 16, 17]. We compare our methods with domain generalization methods and verify if these methods work well for real-world fine-tuning. We use ITSA [1] and Asymmetric Augmentation [14] for comparison. As shown in Table IV, domain generalization methods designed for synthetic data pre-training fail in this case. We think differences between synthetic and real data may render previous methods unsuitable: existing methods reduce shortcuts learning caused by synthetic artifacts [1], while real-world data is actually real and the factors degrade generalization ability can be different.

D.2. Stereo Matching Network Architectures

In the main paper, we conduct our experiments with robust iterative optimization based stereo matching networks. Here we conduct experiments using other network architectures. We fine-tune CFNet [10] and CGI-Stereo [13],

Method	2012	2015	Midd	ETH3D	Booster
Training set	**KITTI 2012 & 2015**				
baseline	**1.94**	1.36	12.23	23.88	18.43
ITSA [1]	2.01	1.43	12.59	25.72	17.98
Asy.Aug	1.98	**1.34**	12.67	23.37	18.02
DKT	1.98	1.39	**7.11**	**3.64**	**15.51**
Training set	**Booster**				
baseline	52.30	55.44	19.78	93.31	12.88
ITSA [1]	51.95	56.97	18.77	98.78	13.01
Asy.Aug	55.79	58.36	18.51	95.43	12.79
DKT	**3.49**	**4.71**	**6.61**	**2.60**	**12.63**

Table IV. Comparison with domain generalization methods. The previous methods designed for building domain generalized stereo networks during synthetic data pre-training fail to preserve the domain generalization ability during fine-tuning.

which have great domain generalization ability after pre-training on synthetic data. We show the results in Table V. Compared to the baseline fine-tuning strategy with GT, networks fine-tuned with DKT show better generalization ability. Furthermore, We explore fine-tuning the recent Croco-Stereo [11] that builds transformers and train networks with the self-supervised task on a large scale of data. After self-supervised pre-training, Croco-Stereo trains networks to conduct stereo matching jointly on various datasets including SceneFlow, Middlebury, ETH3D, and Booster. We fine-tune Croco-Stereo in the KITTI datasets and evaluate the target and cross domain performance. We note that the cross-domain evaluation in this setting is not to represent the domain generalization ability of the model, but can represent how the model forgets previously seen scenarios. We do not fine-tune Croco-Stereo in the Booster datasets because it has seen the validation set during pre-training.

Method	2012	2015	Midd	ETH3D	Booster
Training set	**KITTI 2012 & 2015**				
CFNet [10] *	5.71	4.84	15.77	5.48	38.84
CFNet(ft)	2.15	1.39	19.83	29.94	30.95
DKT-CFNet	2.23	1.47	12.98	6.16	30.27
CGI-Stereo [13] *	6.55	5.49	13.91	6.30	33.38
CGI-Stereo(ft)	2.41	1.58	18.62	29.84	30.51
DKT-CGI	2.26	1.63	14.31	7.12	29.09
Croco-Stereo [11] *	12.21	18.16	2.62	0.13	8.30
Croco-Stereo(ft)	1.81	1.22	7.83	2.19	23.12
DKT-Croco	1.78	1.26	3.08	1.27	9.81
Training set	**Booster**				
CFNet [10] *	4.97	6.31	15.77	5.48	35.03
CFNet(ft)	56.20	71.41	18.45	80.53	25.86
DKT-CFNet	3.57	4.26	11.17	5.38	26.11
CGI-Stereo [13] *	5.90	6.02	13.91	6.30	30.23
CGI-Stereo(ft)	30.93	46.84	20.34	46.79	23.87
DKT-CGI	5.38	5.11	13.83	6.37	23.60

Table V. Results of fine-tuning with more network architectures. * uses pre-trained weights provided by the authors. Our proposed DKT framework can be applied to various network architectures and preserves their domain generalization ability.

D.3. Fine-tuning on More Datasets

We perform fine-tuning on the Middlebury and ETH3D datasets, following the data split in MCV-MFC [4]. The experimental results are presented in Table VI. Notably, we observe that fine-tuning on these two datasets can lead to a degradation in generalization ability to some unseen domains. However, it's noteworthy that fine-tuning on Middlebury and ETH3D can enhance performance on specific unseen datasets, and overall, the degradation in domain generalization is less pronounced compared to fine-tuning on KITTI and Booster. We think this difference is attributed to the fact that Middlebury and ETH3D datasets contain little transparent or mirrored (ToM) surfaces, which have a substantial impact on degrading domain generalization ability. The modest performance gaps between pre-trained networks and those subjected to fine-tuning suggest that the acquisition of new knowledge during the fine-tuning process may be relatively limited. Moreover, for both datasets, employing DKT during fine-tuning demonstrates better domain generalization ability than using only GT.

Method	2012	2015	Midd	ETH3D	Booster
Training set	**Middlebury 2014**				
IGEV-Stereo [12] *	5.13	6.04	5.03	3.61	17.62
IGEV-Stereo(ft)	4.02	5.01	**3.81**	4.97	15.26
DKT-IGEV	**3.47**	**4.62**	3.83	**2.97**	**14.23**
Training set	**ETH3D**				
IGEV-Stereo [12] *	5.13	6.04	**7.06**	3.09	17.62
IGEV-Stereo(ft)	5.19	5.62	12.31	2.26	22.57
DKT-IGEV	**4.81**	**5.59**	7.32	**2.23**	**17.33**

Table VI. Results of fine-tuning networks on more datasets. * uses pre-trained weights provided by the authors. Networks fine-tuned by the DKT framework show better robustness to unseen domains.

D.4. Joint Generalization

In addition to fine-tuning stereo matching networks on individual real-world scenarios, we employ DKT for joint fine-tuning across multiple domains. Besides assessing performance in target domains, we also evaluate the domain generalization ability on previously unseen DricingStereo scenarios. The results, presented in Table VII, demonstrate that using DKT for joint fine-tuning yields comparable results across multiple seen domains, while exhibiting superior robustness on unseen scenarios.

E. Additional Qualitative Results

In this section, we provide additional qualitative results of the domain generalization performance of stereo matching networks fine-tuned with only GT and DKT. Compared to using only GT for fine-tuning, DKT effectively preserves the networks' robustness to unseen domains after fine-tuning. Figures III to V use the same networks fine-tuned on the KITTI datasets and show the performance on

Method	2012	2015	Middlebury	ETH3D	Booster	DrivingStereo				
	>3px(%)	>3px(%)	>2px(%)	>1px(%)	>2px(%)	sunny	cloudy	foggy	rainy	avg
CFNet [10]	2.47	1.78	6.96	1.99	30.43	2.75	2.49	2.03	6.39	3.42
DKT-CFNet	2.51	1.80	5.92	**1.81**	18.55	2.20	2.34	1.89	3.55	2.50
IGEV-Stereo [12]	**2.00**	1.56	3.80	1.98	12.83	2.29	1.89	1.49	8.19	3.47
DKT-IGEV	2.02	**1.54**	**3.79**	2.01	**11.19**	**2.23**	**1.81**	**1.42**	**3.31**	**2.19**

Table VII. Results of joint generalization. Networks are fine-tuned on a combination of KITTI 2012, KITTI 2015, Middlebury, ETH3D, and Booster datasets. Networks fine-tuned by the DKT framework show competitive joint generalization performance, as well as better robustness to unseen challenging weather.

unseen Middlebury, Booster, and ETH3D domains. Figures VI to IX use the same networks fine-tuned on the Booster dataset and show the performance on unseen KITTI 2012, KITTI 2015, Middlebury, and ETH3D domains.

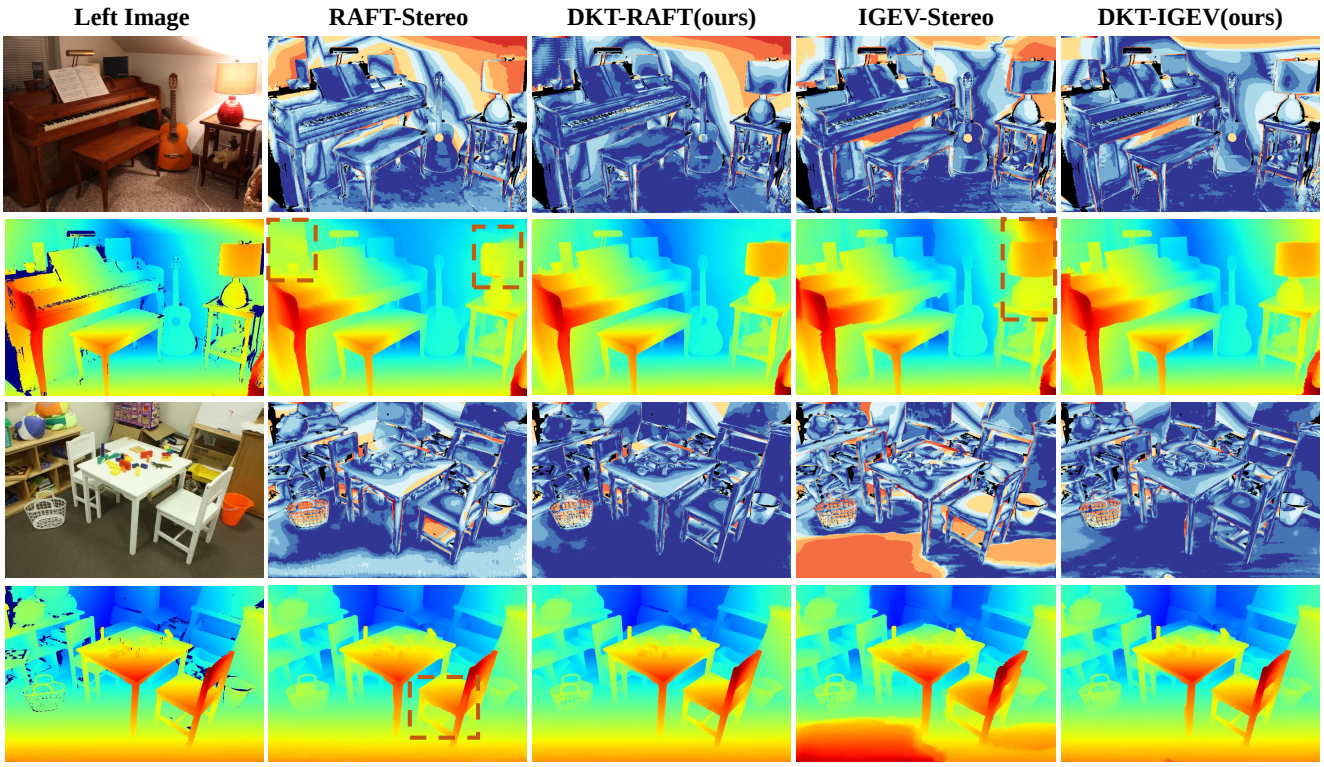

Figure III. Qualitative results of KITTI fine-tuned networks on the Middlebury training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

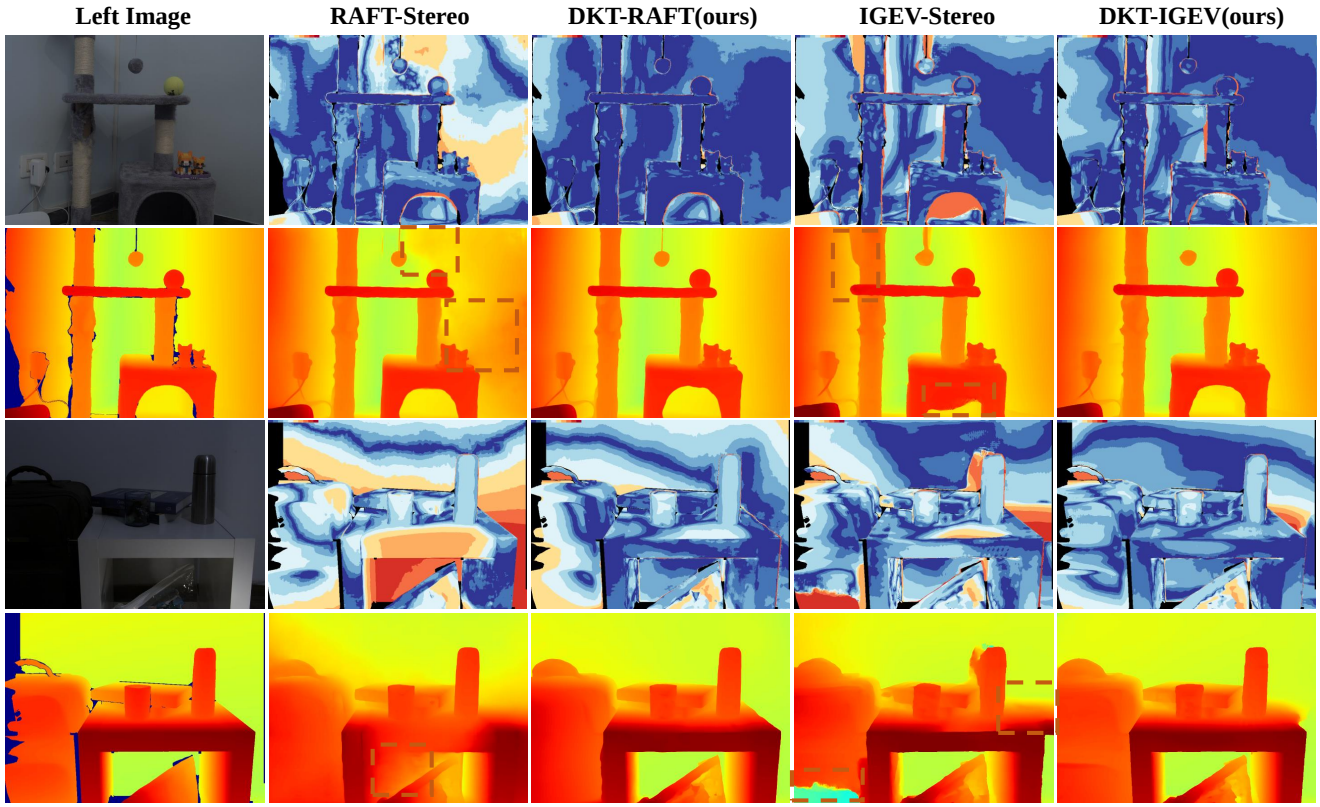

Figure IV. Qualitative results of KITTI fine-tuned networks on the Booster training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

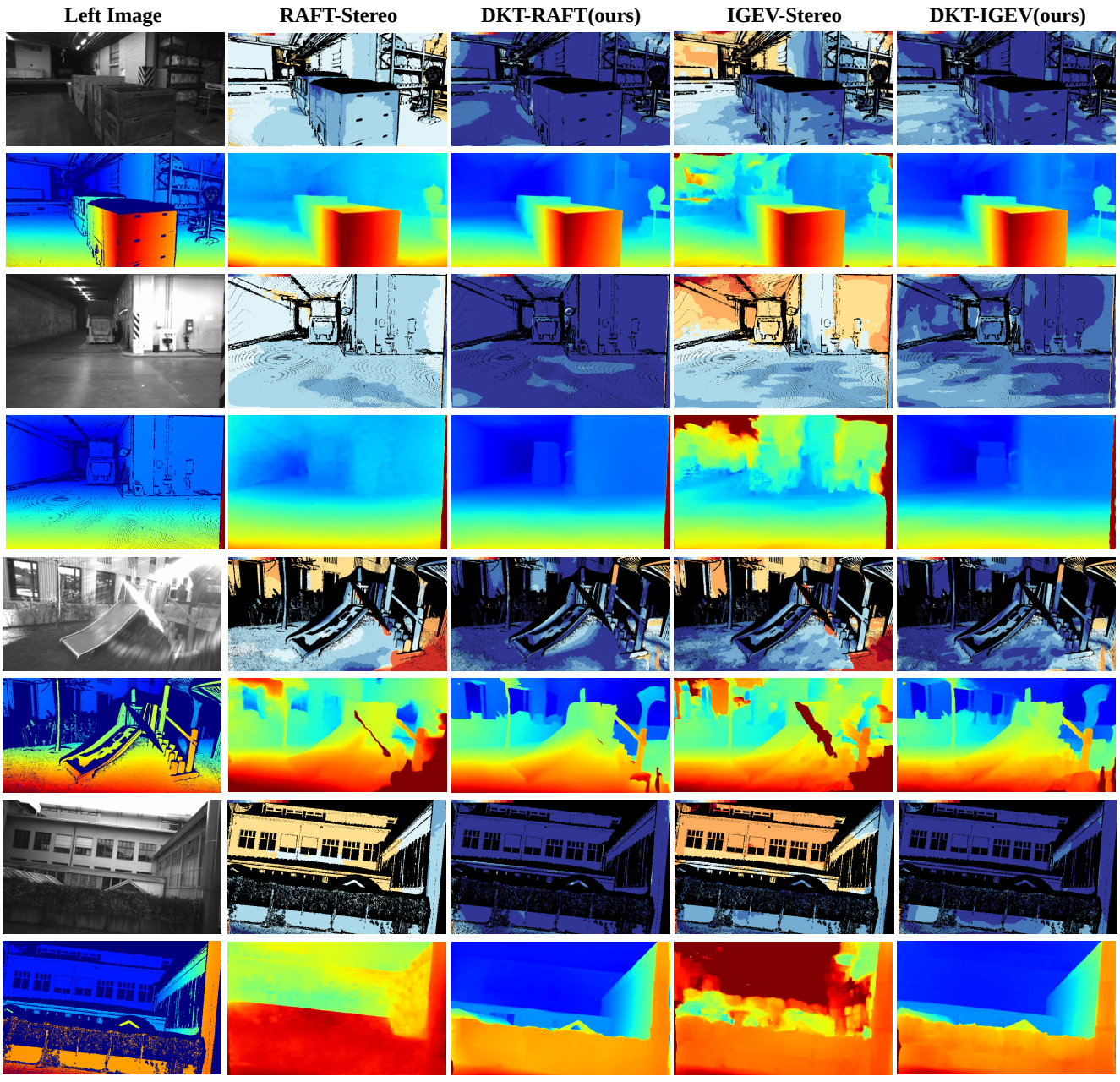

Figure V. Qualitative results of KITTI fine-tuned networks on the ETH3D training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

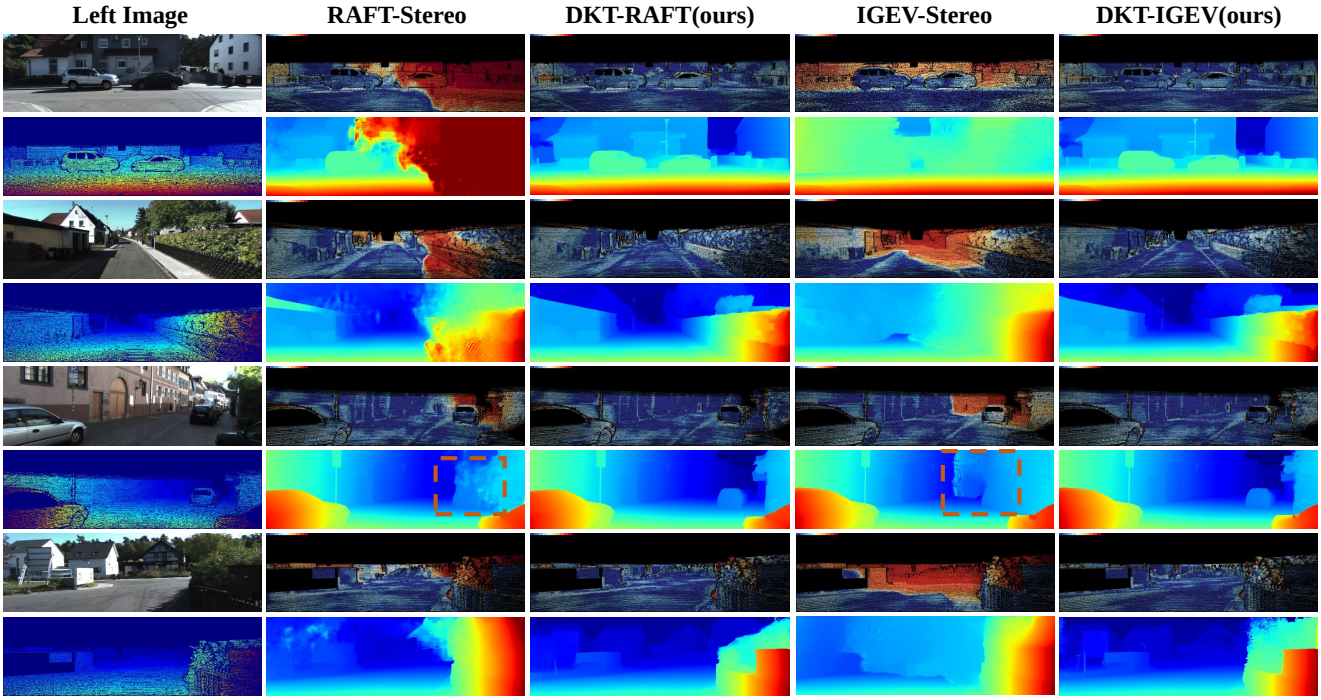

Figure VI. Qualitative results of Booster fine-tuned networks on the KITTI 2012 training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

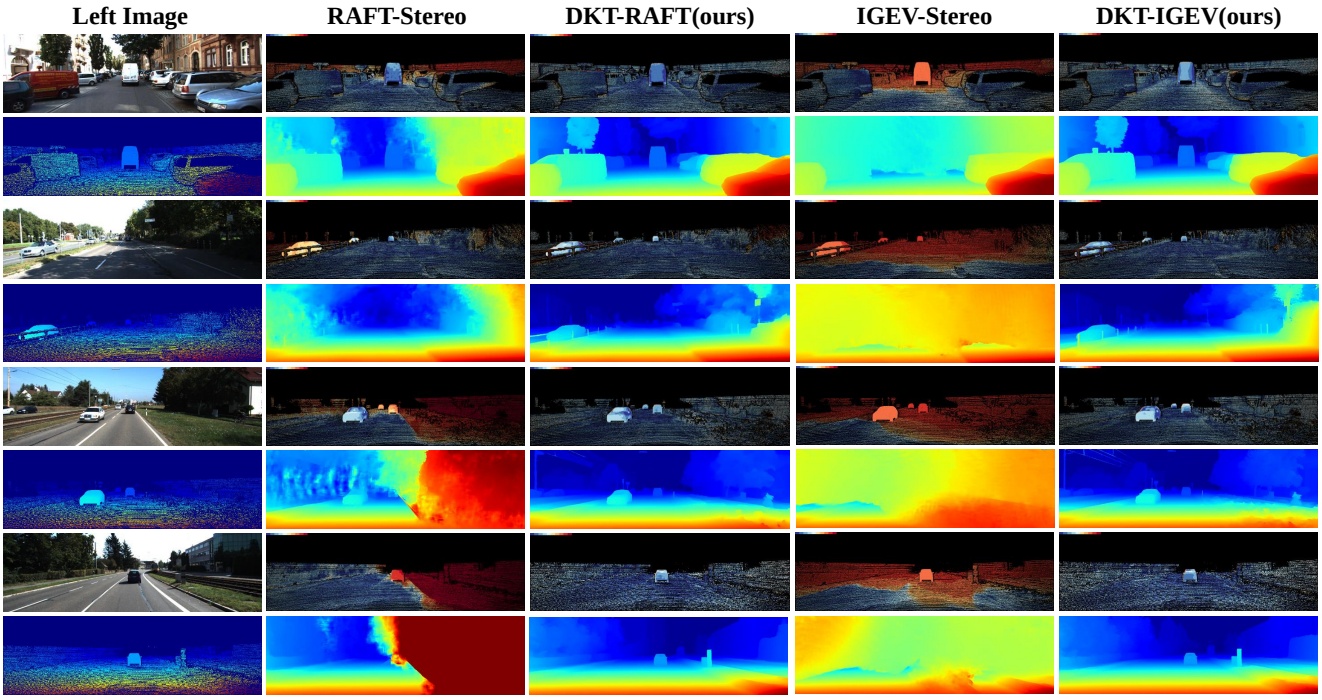

Figure VII. Qualitative results of Booster fine-tuned networks on the KITTI 2015 training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

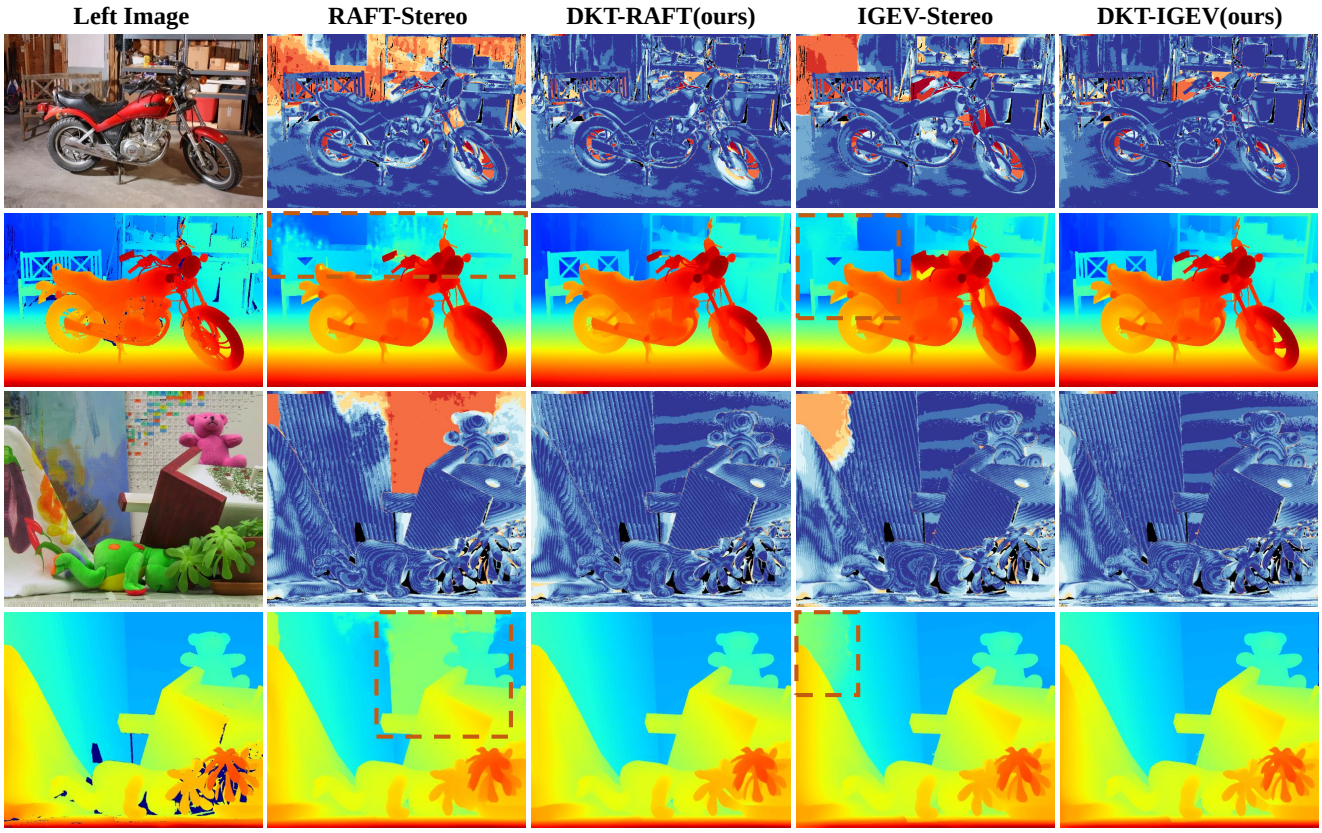

Figure VIII. Qualitative results of Booster fine-tuned networks on the Middlebury training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

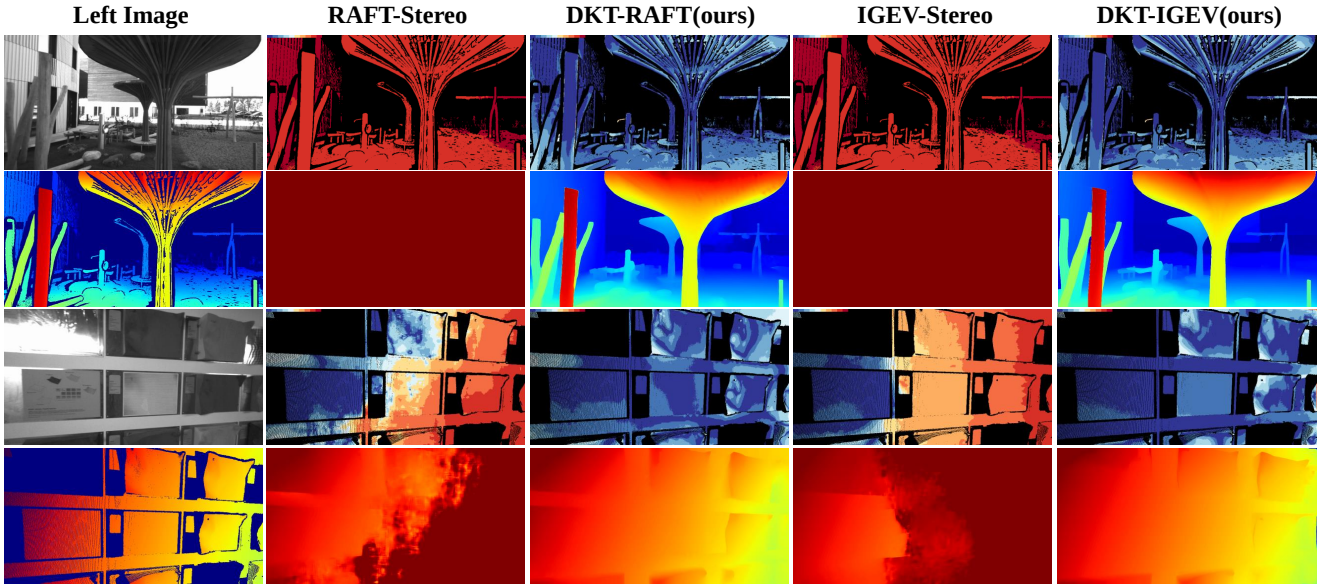

Figure IX. Qualitative results of Booster fine-tuned networks on the ETH3D training set. The left panel shows the left input image and the ground truth disparity. For each example, the first row shows the error map and the second row shows the colorized disparity prediction.

References

[1] WeiQin Chuah, Ruwan Tennakoon, Reza Hoseinnezhad, Alireza Bab-Hadiashar, and David Suter. Itsa: An information-theoretic approach to automatic shortcut avoidance and domain generalization in stereo matching networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13022–13032, 2022. 2, 3

[2] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 1

[3] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3273–3282, 2019. 1

[4] Zhengfa Liang, Yulan Guo, Yiliu Feng, Wei Chen, Linbo Qiao, Li Zhou, Jianfeng Zhang, and Hengzhu Liu. Stereo matching using multi-level cost volume and multi-scale feature constancy. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):300–315, 2019. 1, 3

[5] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016. 1

[6] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3061–3070, 2015. 1

[7] Pierluigi Zama Ramirez, Fabio Tosi, Matteo Poggi, Samuele Salti, Stefano Mattoccia, and Luigi Di Stefano. Open challenges in deep stereo: the booster dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21168–21178, 2022. 1

[8] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German conference on pattern recognition*, pages 31–42. Springer, 2014. 1

[9] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3260–3269, 2017. 1

[10] Zhelun Shen, Yuchao Dai, and Zhibo Rao. Cfnet: Cascade and fused cost volume for robust stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13906–13915, 2021. 1, 2, 3, 4

[11] Philippe Weinzaepfel, Vaibhav Arora, Yohann Cabon, Thomas Lucas, Romain Brégier, Vincent Leroy, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Improved cross-view completion pre-training for stereo matching. *arXiv preprint arXiv:2211.10408*, 2022. 3

[12] Gangwei Xu, Xianqi Wang, Xiaohuan Ding, and Xin Yang. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21919–21928, 2023. 1, 3, 4

[13] Gangwei Xu, Huan Zhou, and Xin Yang. Cgi-stereo: Accurate and real-time stereo matching via context and geometry interaction. *arXiv preprint arXiv:2301.02789*, 2023. 2, 3

[14] Gengshan Yang, Joshua Manela, Michael Happold, and Deva Ramanan. Hierarchical deep stereo matching on high-resolution images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5515–5524, 2019. 2

[15] Guorun Yang, Xiao Song, Chaoqin Huang, Zhidong Deng, Jianping Shi, and Bolei Zhou. Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 899–908, 2019. 1

[16] Feihu Zhang, Xiaojuan Qi, Ruigang Yang, Victor Prisacariu, Benjamin Wah, and Philip Torr. Domain-invariant stereo matching networks. In *European Conference on Computer Vision*, pages 420–439. Springer, 2020. 2

[17] Jiawei Zhang, Xiang Wang, Xiao Bai, Chen Wang, Lei Huang, Yimin Chen, Lin Gu, Jun Zhou, Tatsuya Harada, and Edwin R Hancock. Revisiting domain generalized stereo matching networks from a feature consistency perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13001–13011, 2022. 2