Supplementary Material for ADFactory

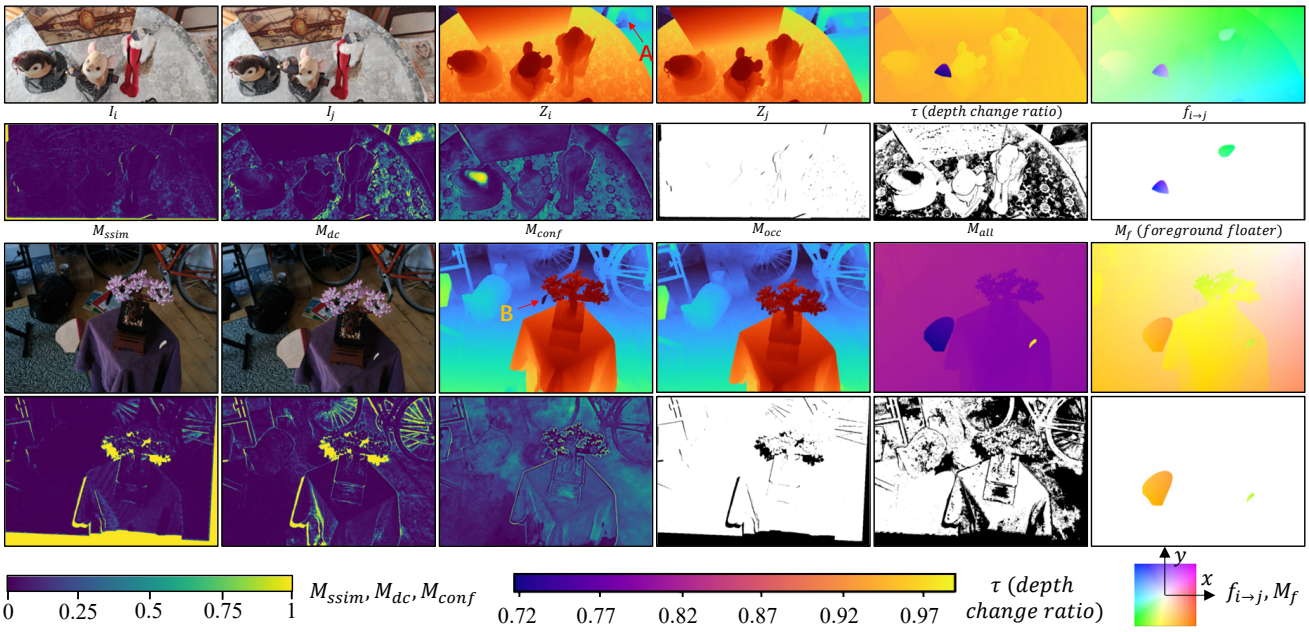

Figure 1. **Complete ADF dataset example.** Two examples of complete ADF training sets were presented. Here, M_{all} is the final filter mask and equal to $M_{conf}^{th1} M_{ssim}^{th2} M_{dc}^{th3} M_{occ}$. It is noted that M_{all} also uses M_{occ}, as explained in Section B.3.

A. Details of the Dataset ADF58

First of all we have to thank Fabio Tosi[13] for the photo set (taken with a mobile phone, including static scenes both indoors and outdoors), from which about 200 of our 300 scenes came from them. The remaining scenes were captured from existing Nerf datasets and handheld devices by the authors of this article.

A.1. How ADFactory is different.

In previous self-supervised optical flow work[5, 14–16], almost all methods tended to design a customized miniaturized network. This is easy to understand, as they train based on indirect losses, making convergence difficult when the network structure is complex. ADF is different. We have demonstrated in experiments that it can efficiently train complex baselines such as RAFT, Scale-flow, and GMFlow (including Transformer). It is one of the few self-supervised training methods that can be applied to most optical flow baselines.

A.2. Dataset Example

In Fig. 1, we show examples of two complete dataset outputs, noting that there are reconstruction failures at both A and B. Location A is the white wall behind the chair backrest, which caused an incorrect depth due to reconstruction failure. Observing the corresponding M_{dc} mask, it was found that the depth consistency at this location is very poor, which is in line with our original design intention. B is a pure black object on the floor, which has been filtered out in both M_{dc} and M_{conf}.

ADF also provides the rate of change in depth (dynamic foreground has yet to be added to the depth, but this is not difficult). When training Scale-flow, we share a M_{all} with optical flow.

A.3. Difficult Sample

At the beginning, the composition of the generated data consisted of 50% difficult samples (with an average optical flow value greater than 300 pixels) and 50% simple samples (with an average optical flow value less than 20 pixels). We found the network challenging to converge during training. In the final generated version, we retained 15% of the difficult samples and achieved more ideal results. Although most of the test data is simple, we believe it is necessary to retain difficult samples, which can help the network better handle fast moving small objects in the real world. In future work, we will also consider creating difficult samples (mo-

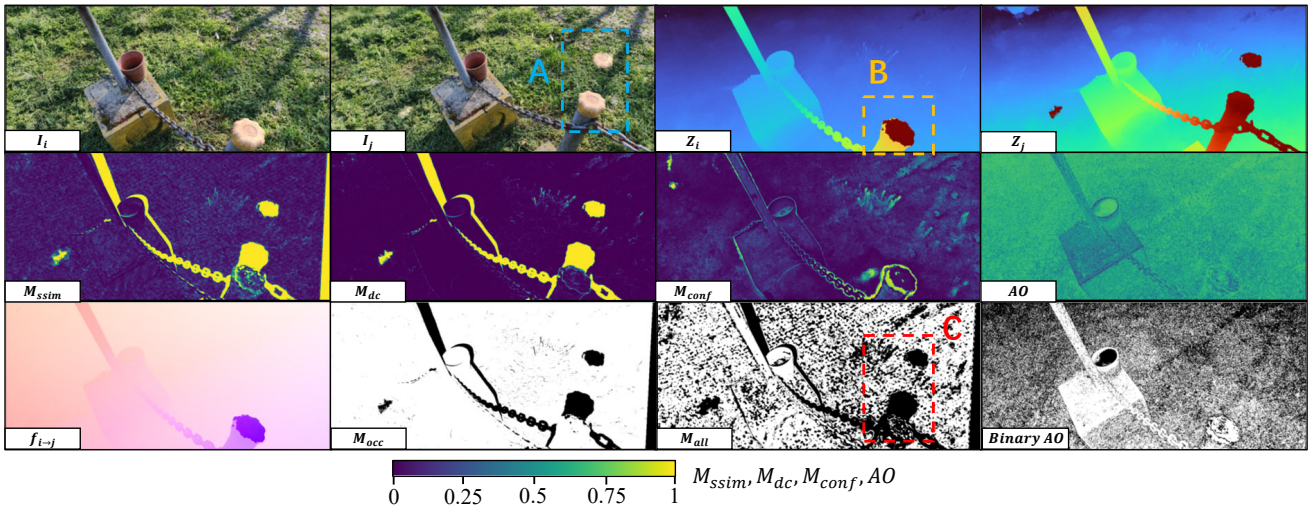

Figure 2. **AO failure.** Binary AO is an AO result binarized with a threshold of 0.65. Comparing the C part in Binary AO and M_{all}, it can be found that our metrics can filter out errors that AO cannot handle.

tion blur, smoke, etc.) to approximate real scenes further.

B. More Results

In this section, we have presented more results to answer readers' possible doubts, including: 1. Why is AO not effective? 2. Visualization of typical scenes. 3. Occlusion. 4. Detailed comparison results of ADF in optical flow baseline. 5. Reconstruction of vehicles.

B.1. Ambient Occlusion (AO)

This section mainly explains why we did not use the commonly used AO mask to screen for poor data. Firstly, let us review the definition of AO: the probabilities of the existence of a surface from the observation point to the depth of the ray. It reveals whether there will be floating objects in the air, which reflects the reconstruction quality of the neural field. However, more is needed for a comprehensive evaluation of the dataset. We summarize the following reasons:

- The AO indicator is not applicable to anti-aliasing methods because the smoothness of anti-aliasing results in high AO values even at the correct position. As shown in Fig. 2 (AO), the value of AO is much higher than the indicator we proposed, and it contains much noise, making it challenging to detect actual errors.
- AO cannot take effect when there is an error in the second frame. As shown in Fig. 2 A, the image of the first frame is normal at this time, but there is a significant depth error in the second frame. Simply measuring the reconstruction quality of the first frame is meaningless, but M_{dc} and M_{ssim} can easily identify this geometric inconsistency.
- When the reconstruction field is stable, but reconstruction error occurs, AO cannot take effect. As shown in Fig. 2 B, there is clearly an anomalous depth in the depth field, but its radiation field is stable. Neither AO nor M_{conf} can identify the issue, but M_{dc} and M_{ssim} can easily identify it.

Observation of Fig. 2 C, our comprehensive indicator M_{all} successfully removed all outliers and retained geometrically stable and high-quality data.

B.2. Visualization

We presented several typical real-world zero-shot scenes in the **demo** folder, including:

- **Occlusion:** libby
- **Portrait + large movements:** breakdance; breakdance-flare; motocross-bumps; soapbox
- **Fluctuating water surface:** blackswan; flamingo
- **Small amplitude movement:** camel
- **Traffic Scenarios:** train; car-roundabout

Except for traffic scenes, our method subjectively has higher clarity and accuracy than supervised training methods (**training process of supervised methods is C+T/S $\rightarrow$ K15**). This is not difficult to understand, as existing real-world datasets mainly focus on driving scenarios (actually only driving scenarios).

B.3. Occlusion

Although we did not pay attention to occlusion, the method trained by ADF still has good anti-occlusion performance. As shown in Fig. 3, our ADF method performs amazingly when there is a large amount of occlusion in a zero shot scene.

In the final training version, we used all masks (including M_{occ}) because we found in the experiment that adding M_{occ} did improve a small amount of performance and did not

Table 1. **Evaluation of Optical Flow.** The best results in the same category are bolded.

				K15		K12		K15-test
Training type	Method	Source	Training data	Fl_{epe}	F_{all}	Fl_{epe}	F_{all}	F_{all}
Supervised generalization	LiteFlowNet2[2]	TPAMI2020	C + T/S	8.97	25.9	3.42	-	-
	VCN[17]	NeurIPS2019	C + T/S	8.36	25.1	-	-	-
	RAFT[12]	ECCV2020	C + T/S	**5.04**	**17.4**	-	-	-
Self-supervised fine-tuning	UFlow[5]	ECCV2020	Km+Kraw	2.71	-	1.68	-	11.13
	MDFlow-fast[6]	TCSVT2022	Km	4.44	**12.3**	1.83	**6.8**	-
	UPFlow[10]	CVPR2021	Km+Kraw	**2.45**	-	**1.27**	-	**9.38**
	SMMSF[3]	CVPR2021	Km+Kraw	6.04	18.81	-	-	15.97
Self-supervised generalization	COTR[4]	ICCV2021	MD	6.12	16.9	2.26	10.5	-
	GLU-Net[14]	CVPR2020	COCO	7.49	33.83	3.14	19.76	-
	PDC-Net+[16]	TPAMI2023	COCO	4.53	**12.62**	1.76	6.6	-
	PDC-Net[15]	CVPR2021	COCO	5.22	15.13	2.08	7.98	-
	MDFlow-fast[6]	TCSVT2022	S	10.05	23.12	3.49	12.17	-
	MDFlow-fast[6]	TCSVT2022	GTA5	9.13	25.01	3.85	14.33	-
	Scale-flow [9]	MM2022	ADF58(ours)	**3.88**	13.36	**1.59**	6.97	13.47
	RAFT [12]	ECCV2020	ADF58(ours)	4.17	13.9	1.59	**6.43**	**13.41**

reduce the method's estimation ability for occluded parts. This mainly stems from three points:

- The motion foreground M_f obscures the background, and the occlusion does not disappear.
- Because a large number of training samples will be generated in one scene, covering all parts of the scene, even if a certain frame is partially occluded, the occluded part can still be learned in other frames.
- M_{occ} can assist in removing some erroneous depth results (occlusion caused by incorrect depth). As shown in Fig. 2 , M_{occ} removed the error incorrect in part C.

B.4. Optical Flow Performance

This section extends Tab.3 in the main text, showcasing more detailed results. Firstly, let us briefly introduce the training data used and the evaluation items.

- C: Flyingchairs for Flyingthings dataset[11].
- T: Flyingthings3D for Flyingthings dataset[11].
- S: Sintel dataset generated from movie scenes[1].
- K15: 200 images from KITTI 2015 training set[1].
- K12: 194 images from KITTI 2012 training set[2].
- Km: Multi-frame images of K15.
- Kraw: KITTI raw data[3].
- K15-test: KITTI15 official testing benchmark[4].
- MD: MegaDepth[7].
- COCO: Semantic Segmented Image Dataset[8].
- GTA5: From game GTA5, including game graphics and corresponding realistic depths.

Evaluation performance In the group of self-supervised generalization, our ADF58 always maintains optimal accuracy in most cases, proving our scheme's superiority in terms of performance.

From the perspective of training modes, the performance of the RAFT method trained by ADF also surpasses (5.04 v.s. 4.17; 17.4 v.s. 13.9) the pre-training method based on synthetic data (C+T/S), which proves the excellent potential of the ADF scheme as a universal large model pre-training scheme. Moreover, more importantly, ADF is an easily expandable solution that can quickly expand excellent optical flow datasets from monocular videos at a meagre cost, further enhancing the dataset's size and the semantic priors it contains.

B.5. Rebuilding Vehicles

From the results in the **more results** folder, it can be seen that the performance of ADF in driving scenarios is significantly limited compared to daily scenarios. As shown in Fig. 4, although the correct RGB image I_i can be reconstructed, the depth value Z_i at the car window needs to be corrected. This defect directly results in most labels being filtered out by the mask in the vehicle scene, making it im-

[1] https://www.cvlibs.net/datasets/kitti/eval_scene_flow.php

[2] https://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=flow

[3] https://www.cvlibs.net/datasets/kitti/raw_data.php

[4] https://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=flow

Figure 3. **Occlusion in zero shot generalization scenes.** When faced with unseen obstacles (tree trunks, iron pillars), the algorithm trained by ADF still exhibits stunning visual performance (this image is from libby.mp4 in the **demo** folder).

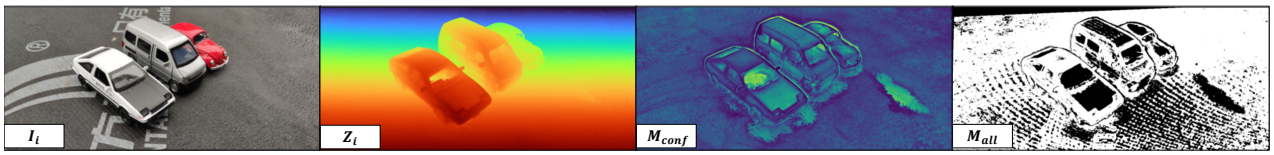

Figure 4. **Reconstruction of vehicles.** Zip-Nerf cannot accurately estimate the depth of transparent and reflective surfaces, making it challenging to generate a dataset about vehicles.

possible to conduct sufficient training. We will attempt to improve this defect in future work by using more advanced Nerf models.

References

[1] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision (ECCV)*, pages 611–625, 2012. 3

[2] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. A lightweight optical flow cnn—revisiting data fidelity and regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 43(8):2555–2569, 2020. 3

[3] Junhwa Hur and Stefan Roth. Self-supervised multi-frame monocular scene flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2684–2694, 2021. 3

[4] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 6207–6217, 2021. 3

[5] Rico Jonschkowski, Austin Stone, Jonathan T Barron, Ariel Gordon, Kurt Konolige, and Anelia Angelova. What matters in unsupervised optical flow. In *European Conference on Computer Vision (ECCV)*, pages 557–572, 2020. 1, 3

[6] Lingtong Kong and Jie Yang. Mdflow: Unsupervised optical flow learning by reliable mutual knowledge distillation. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 33(2):677–688, 2022. 3

[7] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755, 2014. 3

[9] Han Ling, Quansen Sun, Zhenwen Ren, Yazhou Liu, Hongyuan Wang, and Zichen Wang. Scale-flow: Estimating 3d motion from video. In *Proceedings of the 30th ACM International Conference on Multimedia (ACMMM)*, pages 6530–6538, 2022. 3

[10] Kunming Luo, Chuan Wang, Shuaicheng Liu, Haoqiang Fan, Jue Wang, and Jian Sun. Upflow: Upsampling pyramid for unsupervised optical flow learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 1045–1054, 2021. 3

[11] 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/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4040–4048, 2016. 3

[12] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision (ECCV)*, pages 402–419, 2020. 3

[13] Fabio Tosi, Alessio Tonioni, Daniele De Gregorio, and Matteo Poggi. Nerf-supervised deep stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 855–866, 2023. 1

[14] Prune Truong, Martin Danelljan, and Radu Timofte. Glu-net: Global-local universal network for dense flow and correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 6258–6268, 2020. 1, 3

[15] Prune Truong, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning accurate dense correspondences and when to trust them. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 5714–5724, 2021. 3

[16] Prune Truong, Martin Danelljan, Radu Timofte, and Luc Van Gool. Pdc-net+: Enhanced probabilistic dense correspondence network. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023. 1, 3

[17] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. *Advances in neural information processing systems*, 32, 2019. 3