A Appendix

A.1 Additional quantitative results

SKFlow-RAFT. We further test the effectiveness of our proposed SKBlock on other optical networks, e.g., RAFT [4]. In SKFlow-RAFT, we adopt the same SKBlock architecture as that in SKFlow. As shown in Table 1, SKFlow-RAFT outperforms RAFT on Sintel training and test set with a modest increase in parameters. All models are trained with the same settings. From the table, we could learn that SKBlock could be easily adopted by other flow networks.

Table 1: Comparison of model size and performance on Sintel dataset.

Model	Clean (train)	Final (train)	Clean (test)	Final (test)	Parameters
RAFT	0.76	1.22	1.61	2.86	5.26M
SKFlow-RAFT	0.62	0.91	1.46	2.61	5.59M
GMA	0.62	1.06	1.39	2.47	5.88M
SKFlow	**0.52**	**0.78**	**1.28**	**2.27**	6.27M

Occlusion analysis on Sintel test set. We also test the occluded areas on the Clean and Final pass test set. Although it is hard to obtain accurate occluded areas on the test set since we authors have no access to those occlusion maps, the Sintel server provides statistics on the matched and unmatched areas. According to the Sintel website, the matched areas are regions that remain visible in adjacent frames, and the unmatched denote regions visible only in one of two adjacent frames. Therefore, we could still evaluate the performance in the generalized occluded areas. The results are shown in Table 2. We could learn that both SKFlow-RAFT and SKFlow predict more accurate flows on the occluded areas and non-occluded areas on the test set.

Table 2: Performance on the matched and unmatched areas on Sintel test set.

Model	Matched (train)	Unmatched (train)	Matched (test)	Unmatched (test)
RAFT	0.623	9.647	1.405	14.680
SKFlow-RAFT	0.617	8.346	1.288	13.352
GMA	0.582	7.963	1.241	12.501
SKFlow	**0.554**	**7.239**	**1.145**	**11.511**

Super dilated convolution kernels. We also explore the application of dilated convolutions to obtain large receptive fields with modest computation. In our dilated version, the large 15×15 depth-wise convolutions are replaced with a 9×9 dilated convolutions with a dilation rate of 2. Quantitive results are shown in Table 3. All models are trained using the $C \rightarrow T$ schedule and then validated on Sintel. GMA denotes the result in the original paper and GMA* denotes our reproduced result.

The dilated version indeed obtains less computation. Nevertheless, although our method and the dilated version have a receptive field of a similar size, there is a small gap in the performance. We

Table 3: Comparison of dilated and non-dilated models on the Sintel dataset.

Model	Sintel (clean)	Sintel (final)	Parameters
GMA	1.30	2.74	5.88M
GMA*	1.36	2.72	5.88M
SKFlow-Dilated	1.32	2.55	6.10M
SKFlow	**1.22**	**2.46**	6.27M

Table 4: Runtime comparison of different methods on KITTI.

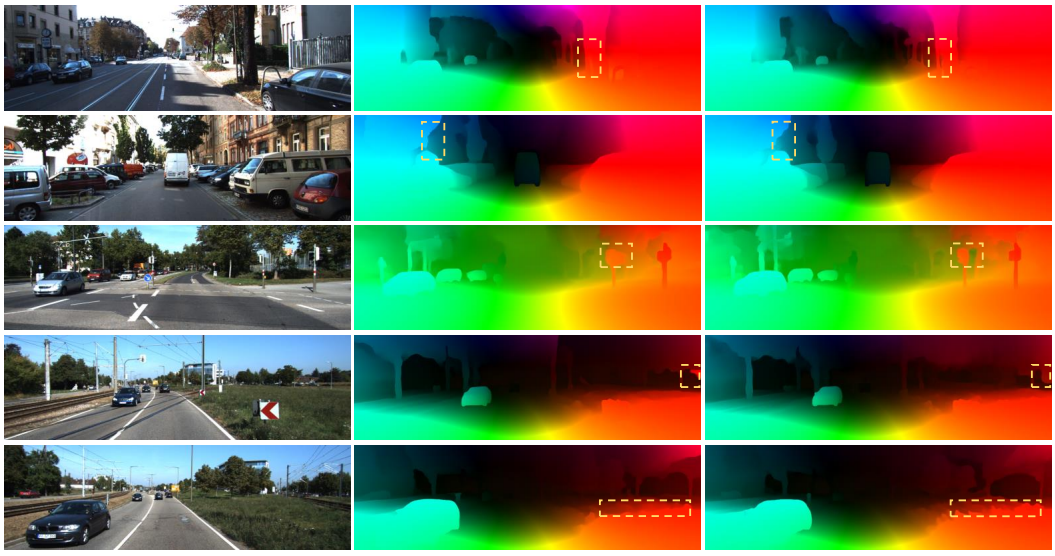
Model	Time
RAFT	0.13s
GMA	0.16s
SKFlow-Dilated	0.21s
SKFlow	0.22s

argue that the gap may be caused by the gridding effect [5]. Namely, the receptive field of a dilated convolution kernel covers an area with checkerboard patterns. Therefore, the sampled locations contribute to the calculation but the neighboring information is lost. In this case, the gridding effect leads to two issues that might affect the estimating of per-pixel motion: (1) Absence of local information. (2) Irrelevant information across large distances due to the sparse sample of input. However, given the efficiency and the various applications of dilated convolution, we believe that how to more properly apply it to optical flow networks is still a question worth further studying.

Runtime comparison. Runtime comparison for different methods is shown in Table 4. Test settings were introduced in the experiment section. We could learn that the runtime of SKFlow is a little bit longer in practice but the increase is still modest. The additional latency is caused by the lack of efficient implementation for large depth-wise and dilated kernels in the current PyTorch library.

A.2 Qualitative evaluation in realistic scenes

In addition, we compare the qualitative results in more realistic scenes. As shown in Figure 1, we visualize the predicted flow on the KITTI [3] test set, which contains more realistic frames compared with Sintel [1]. From Figure 1 we can see that SKFlow also improves the state-of-the-art method in realistic scenes.



(a) Input views **(b) GMA** **(c) Ours**

Figure 1: Visualization on the KITTI test set. (b) Results of GMA [2]. (c) Results of our SKFlow. Differences are highlighted by the dash boxes. Compared with state-of-the-art method, our proposed SKFlow predicts more accurate flow in realistic scenes.

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*, pages 611–625. Springer, 2012.

[2] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9772–9781, 2021.

[3] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. In *ISPRS Workshop on Image Sequence Analysis (ISA)*, 2015.

[4] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020.

[5] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 1451–1460. Ieee, 2018.