MS-RAFT-3D: A MULTI-SCALE ARCHITECTURE FOR RECURRENT IMAGE-BASED SCENE FLOW SUPPLEMENTARY MATERIAL

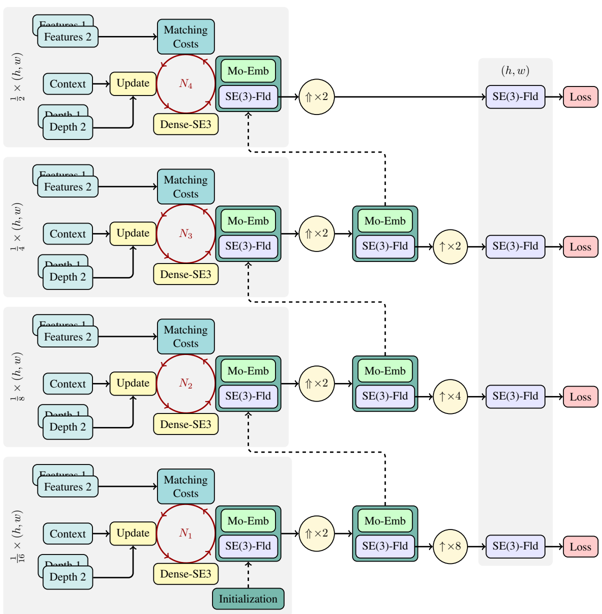

Fig. 1. Architecture of MS-RAFT-3D+

In the following, we first show the architecture of our 4-scale model. Then we elaborate on our employed context encoder and finally, we demonstrate some more visual results on KITTI[1] and Spring[2] benchmarks.

1. ARCHITECTURE OF MS-RAFT-3D+

Figure 1 shows the architecture of our 4-scale MS-RAFT-3D+ model. It can be seen that in addition to the three scales at $[\frac{1}{16}, \frac{1}{8}, \frac{1}{4}]$, the SE(3) field is also refined at $\frac{1}{2}$ resolution. This allows to capture more details from images. Besides, no bilinear upsampling is needed to upsample the SE(3) field to full resolution, as the results after convex upsampling are already at full resolution. Note that for computing the matching costs, we used the on-demand cost computation from [3].

2. CONTEXT ENCODER

We use a simple top-down feature extractor to compute context features. The architecture is shown in Figure 2. The num-

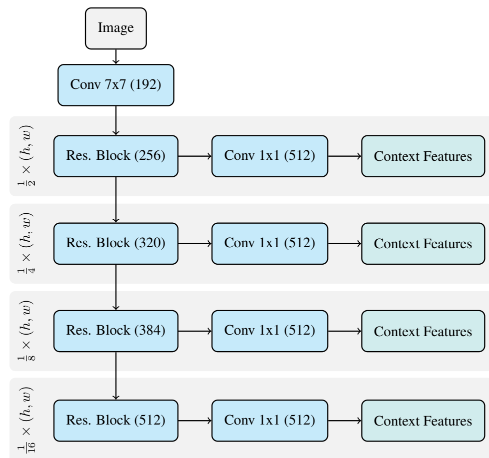

Fig. 2. Structure of the Context Encoder for Four Scales

bers in brackets show the number of channels that is output by each module. Note that the number of context encoder channels in the main paper's ablations correspond to the residual blocks, before applying the 1×1 *conv*. Note that the update unit (which is responsible for computing the residual flow) is shared among scales. This means, inputs of that module at each scale must have the same channels. Therefore we unify the number of channels via 1×1 *convs*. Importantly, Figure 2 shows the context encoder for the 4-scale network. In the case of our 3-scale model, the output of the first residual block at $\frac{1}{2}$ is not passed through a 1×1 *conv* and is not output by the encoder.

3. QUALITATIVE RESULTS

We present more qualitative results of our method on the Spring benchmark in Figure 3 and on the KITTI benchmark in Figure 4. In both cases, our approach achieves detailed results and lower errors. Note the specified areas in Figure 4 *i.e.* cars and traffic signs in both examples. Importantly, similar to [4], for KITTI, as the top 80 pixels of samples are not considered in the evaluation, they are also not computed, but extended from the last row's estimate.

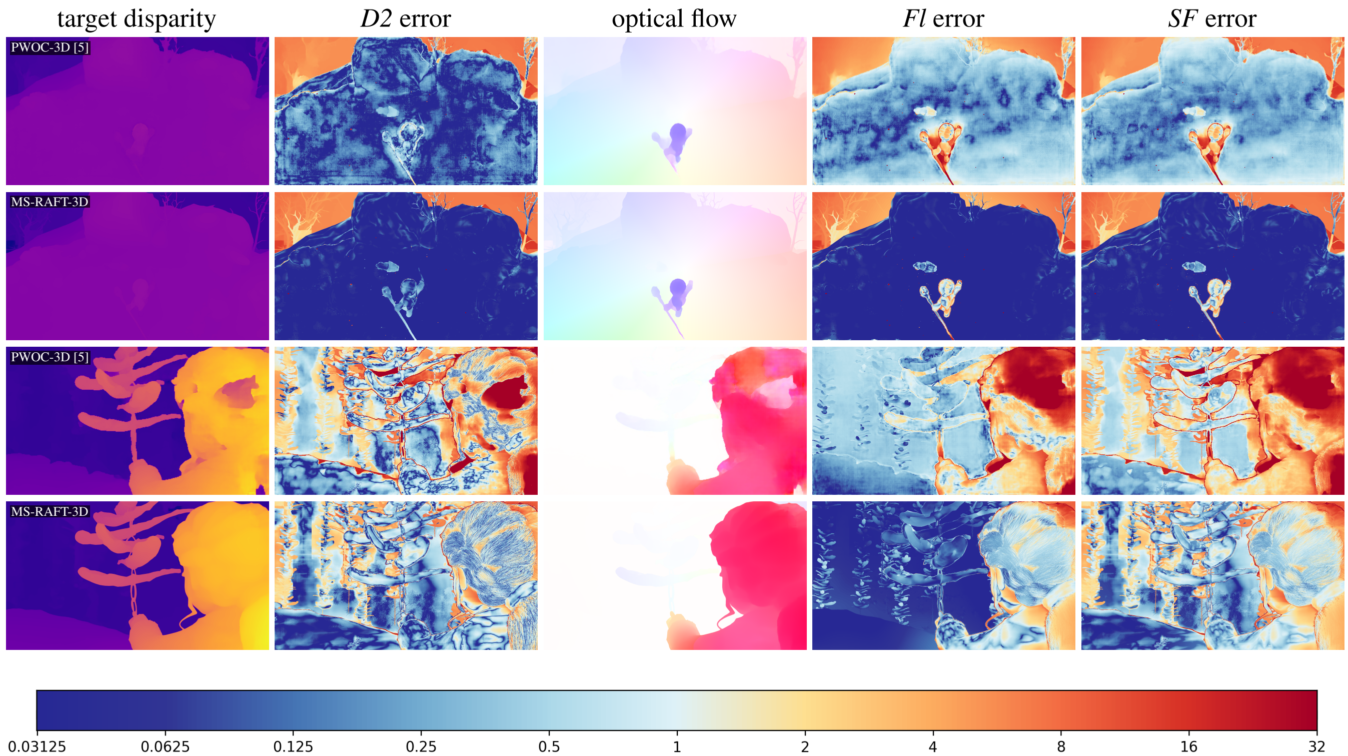

Fig. 3. Qualitative results of our method and the current SOTA on Spring

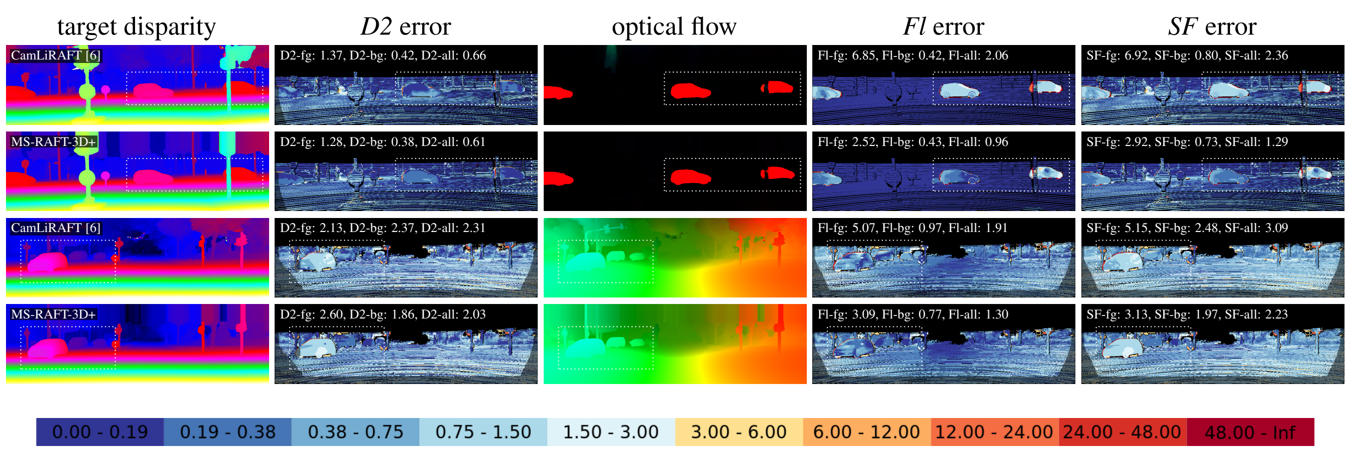

Fig. 4. Visual comparisons of our approach to a SOTA method on KITTI

4. REFERENCES

[1] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *CVPR*, 2015.

[2] L. Mehl, J. Schmalfuss, A. Jahedi, Y. Nalivayko, and A. Bruhn, "Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo," in *CVPR*, 2023.

[3] A. Jahedi, M. Luz, M. Rivinius, L. Mehl, and A. Bruhn, "MS-RAFT+: High resolution multi-scale RAFT," *IJCV*, vol. 132, no. 5, pp. 1835–1856, 2024.

[4] Z. Teed and J. Deng, "RAFT-3D: Scene flow using rigid-motion embeddings," in *CVPR*, 2021, pp. 8375–8384.

[5] R. Saxena, R. Schuster, O. Wasenmuller, and D. Stricker, "PWOC-3D: Deep occlusion-aware end-to-end scene flow estimation," in *IEEE IV*, 2019, pp. 324–331.

[6] H. Liu, T. Lu, Y. Xu, J. Liu, and L. Wang, "Learning optical flow and scene flow with bidirectional camera-lidar fusion," *IEEE TPAMI*, 2023.