Supplementary Material
Instantaneous Perception of Moving Objects in 3D

Di Liu[1] Bingbing Zhuang[3] Dimitris N. Metaxas [1] Manmohan Chandraker[2,3]
[1]Rutgers University [2]University of California, San Diego [3]NEC Labs America

In this supplementary, we provide additional technical details and experimental results that are not included in the main paper due to space limit.

S1. Supportive Explanations

Network architecture. Our network builds upon the encoder-decoder structure presented in [4] and adapts the method from [2] to treat height and temporal dimensions as channel dimensions, facilitating the use of 2D convolutional layers for enhanced processing efficiency. For the occupancy grid prediction, we take a 5-frame point cloud as input, each voxelized with a grid size $[100, 100, 100]$, and use an encoder with a sequence of convolutional layers to encode the input data into a lower-dimensional feature space. This is followed by a decoding process, which reconstructs the occupancy grid from the encoded features. The output is a sigmoid-activated occupancy grid, predicting the presence or absence of object points for each voxel in all five frames. For the motion detection and flow estimation, we pass the output of the occupancy grid prediction through another encoder-decoder structure to classify objects as static or moving and regress a motion vector for each occupied voxel. The encoder extracts features from the occupancy grid, and the subsequent decoder outputs the motion segmentation and flow estimation. The final output consists of the predicted flow for each voxel and the motion segmentation.

Implementation details. We empirically determined the weights for the training losses $\mathcal{L}_{\text{occ}}, \mathcal{L}_{\text{mot}}, \mathcal{L}_{\text{epe}}, \mathcal{L}_{\text{rel}}$ and $\mathcal{L}_{\text{ang}}$ as 1:1:1:1:1. For the occupancy prediction loss, since the number of empty voxels is far larger than that of occupied voxels, we apply class-balanced sampling in dataloader for improved training robustness. Similarly, it is also applied to ensure a class balance between static and moving objects.

S2. Additional Experimental Results

Performance Analysis of PCAcc. To further understand the performance of generic motion estimation approaches in terms of motion magnitude, we study the behavior of the state-of-the-art method in [1], which estimates object scene flows by analyzing temporally point clouds accumulation (PCAcc). To analyze the impact of motion magnitude, we categorize objects based on their speed ranges, from low to high, and calculate the average angle error of the flow for each group. As shown in Fig. S1, the errors increase dramatically as the speed decreases. This trend underscores that the generic motion estimator falters in the regime of subtle motion, emphasizing the need for methods dedicated to finer motion analysis.

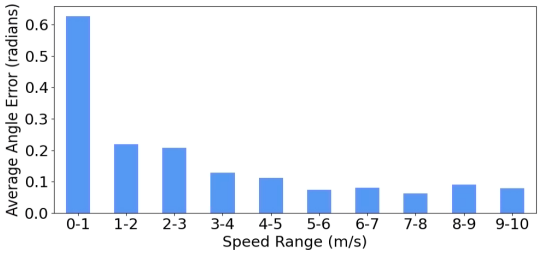


Figure S1. **Performance Analysis of PCAcc [1]** under varying object motion speed.

End-to-end training. In the main paper, we have trained the entire network in an end-to-end manner, including both the occupancy completion network and the subsequent motion perception networks. Here, we study the two-stage training strategy – training the occupancy completion network first, and then training the motion perception networks with the occupancy network frozen. As shown in Tab. S1, we observe better performance from the end-to-end training strategy. This is likely because end-to-end training integrates awareness of the final motion perception task into the occupancy completion network training, providing task-oriented guidance. Such guidance is valuable, especially as the direct supervision of the occupancy completion itself is sparse.

Qualitative comparison on motion flow. We provide more qualitative results on the motion flow estimation in comparison to the baselines in Fig. S3. As shown, the motion flows predicted from our S'More lead to better point cloud alignment, indicating its high accuracy.

Additional discussions and results on FastNSF [3]. In

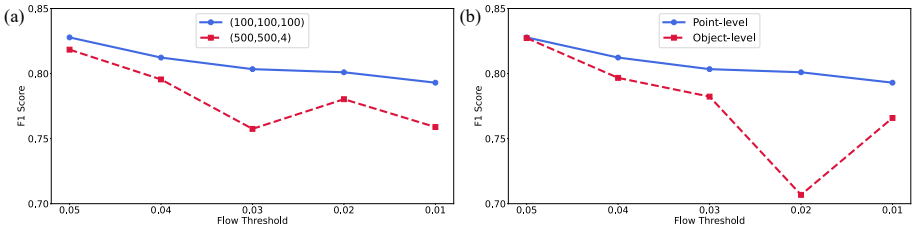
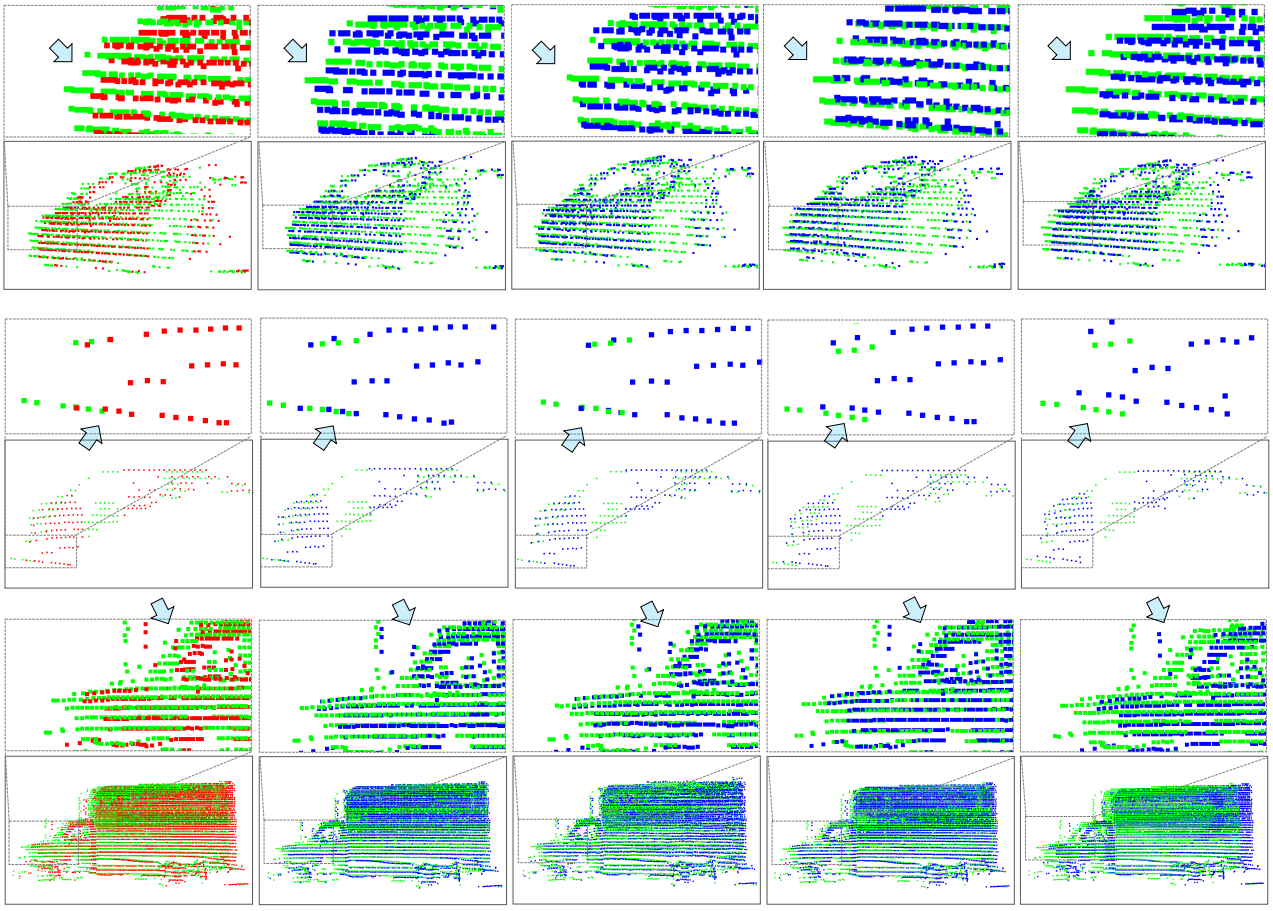


Figure S2. **Ablation study on grid size and point-/object-level prediction.**



(a) Input (b) Ground Truth (c) S'More (ours) (d) FastNSF (e) ICP

Figure S3. **Qualitative comparison.** We exhibit point cloud registration results for two point cloud sets: the first frame (in red) and the last frame (in green). The results are shown using (b) ground truth motion, and estimated motions by (c) S'More(ours), (d) FastNSF, and (e) ICP. The blue points indicate resultant positions after adding flow to the red points, which should ideally align with the green points.

Table S1. **Ablation study on the two-stage and end-to-end training strategy of S'More.**

	EPE (↓)	Angle Error (↓)	F1 Score (↑)
Two-stage	0.0518	0.4503	0.8192
End-to-end	**0.0437**	**0.3189**	**0.8323**

the main paper, we report the results from FastNSF with its standard scene-centric setting – we remove background points and feed the remaining points from all objects altogether to FastNSF for flow estimation. Here, we report additional results with an object-centric setting – we feed points from each object separately to FastNSF. As shown in Tab. S2, this still underperforms S'More. Empirically, we

Table S2. **Additional comparison with FastNSF.**

	EPE (↓)	Angle Error (↓)	F1 Score (↑)
FastNSF [3] (object-centric)	1.0574	0.5309	0.7123
S'More	**0.0437**	**0.3189**	**0.8323**

Table S3. **Study on the impact of training data purity.**

f_{min}	EPE (↓)	Angle Error (↓)	F1 Score (↑)
10.0	0.0979	0.5550	0.1414
5.0	0.0898	0.5796	0.0321
1.0	0.0663	0.4661	0.5910
0.5	0.0720	0.5465	0.4930
0.2 (S'More)	**0.0437**	**0.3189**	**0.8323**

observe that while the motion flows predicted by FastNSF are sufficiently accurate for large motions, their accuracy remains relatively low when it comes to small motions. This may indicate the necessity of dedicated supervised learning, as we do, for precise small motion estimation in the presence of swimming effect.

Purity of training data. In the main paper, we report the accuracy when including large-motion samples in the training data, where we set the threshold $f_{\text{min}} = 10$. Here, we report the accuracy for a range of thresholds, *i.e*. $f_{\text{min}} \in \{0.5, 1.0, 5.0, 10.0\}$. with results reported in Tab. S3. We observe dropped performance with increased values of f_{min}. This implies the important benefit brought about by the high purity of the training data in terms of motion magnitude. Empirically, it indicates that the dedicated training on the regime of small motion facilitates the learning to distinguish subtle motions from static objects, despite the swimming artifact.

Grid size. In the main paper, we compare the flow estimation error under two options in grid size, [100, 100, 100] and [500, 500, 4], respectively. Here, we also report the static/moving classification performance with F1 score in Fig. S2 (a). Again, we observe that higher resolution along the vertical z-axis benefits the performance.

Object-level flow with rigidity constraint. Instead of regressing a motion flow vector separately for each occupied voxel, we also attempt to regress a single rigid transformation (consisting of rotation and translation) for the entire object thereby deriving the motion flows. We have reported the comparison in flow estimation error in the main paper. Here, we further report the F1 score in Fig. S2 (b). We again observe empirically superior performance from the point-level regression.

Data curation. Our training data and evaluation benchmark are derived from the Waymo Open Dataset. We extract data samples from every 5 consecutive frames, *i.e*. a 0.5s temporal window; we only extract samples with small motion and then further classify objects as static or moving depending on the motion flow, as depicted in the main paper. We keep the data sample only if the objects are visible across all five frames, *i.e*. we ignore corner cases where objects completely leave the field of view or become occluded. We use the same split of training and test set as in [1].

S3. Demonstration Videos

In the supplementary folder, we provide video results by running S'More on entire sequences and comparing them with the ground truth, to demonstrate its effectiveness. We suggest readers watch them as videos are the best way to perceive and understand motions.

References

[1] Shengyu Huang, Zan Gojcic, Jiahui Huang, Andreas Wieser, and Konrad Schindler. Dynamic 3d scene analysis by point cloud accumulation. In *European Conference on Computer Vision*, 2022. 1, 3

[2] Tarasha Khurana, Peiyun Hu, David Held, and Deva Ramanan. Point cloud forecasting as a proxy for 4d occupancy forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1

[3] Xueqian Li, Jianqiao Zheng, Francesco Ferroni, Jhony Kaesemodel Pontes, and Simon Lucey. Fast neural scene flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 1, 3

[4] Wenyuan Zeng, Wenjie Luo, Simon Suo, Abbas Sadat, Bin Yang, Sergio Casas, and Raquel Urtasun. End-to-end interpretable neural motion planner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8660–8669, 2019. 1