Supplementary Material

EgoFlowNet: Non-Rigid Scene Flow from Point Clouds with Ego-Motion Support

Ramy Battrawy[1]
ramy.battrawy@dfki.de
René Schuster[1]
rene.schuster@dfki.de
Didier Stricker[1,2]
didier.stricker@dfki.de

[1] Augmented Vision
German Research Center for
Artificial Intelligence (DFKI)
Kaiserslautern, Germany

[2] Computer Science Department
The University of
Kaiserslautern-Landau (RPTU)
Kaiserslautern, Germany

1 Introduction

In our supplementary material, we explain details of implementation, training and augmentation and we perform further ablation studies to validate our design choices. We then add another comparison with a newer method that works under the assumption of rigidity. Finally, we discuss the possible shortcomings of our approach and show more qualitative results.

2 Implementation, Training and Augmentation

Following related approaches [3, 5], we train our method by considering all frames of the train split of semKITTI [1]. During training, the preprocessed data is randomly sub-sampled to a certain resolution (*i.e.*, 8192 points), where the order of the points is random and the correlation between consecutive frames is resolved by random selection. We use the Adam optimizer with default parameters and train our model for 150 epochs. We use an exponentially decaying learning rate, initialized at 0.001 and then decaying at a rate of 0.7 every 10 epochs. We apply batch normalization to all layers of our model except the last layer in each head (*i.e.*, segmentation, scene flow, and the layer providing confidence values in the ego-motion branch). We perform geometric augmentation, which is a random rotation of all points around one randomly chosen axis by a random degree uniformly selected between -10° and $+10^\circ$. Our entire architecture is implemented using TensorFlow.

Table 1: We explore the impact of our losses. For this experiment, we train on semanticKITTI [1] and test on lidarKITTI [2] in the presence of ground surface points. The marker (*) indicates that the self-supervised loss of scene flow is applied to all points without considering the segmentation masks.

$\mathcal{L}_{sf}$	$\mathcal{L}_{seg}$	$\mathcal{L}_{ego}$	**EPE3D$_{\text{all}}$** [m]	**EPE3D$_{\text{fg}}$** [m]	**EPE3D$_{\text{bg}}$** [m]	**Acc3DR** [%]
✓(*)	✗	✗	0.509	0.485	0.501	0.193
✓(*)	✓	✓	0.071	0.380	0.049	0.920
✓	✓	✓	**0.049**	**0.267**	**0.033**	**0.964**

Table 2: The use of hybrid features with the stop gradient in our EgoFlowNet almost matches the results of the task-specific segmentation network and provides the most accurate results for the ego-motion.

Task	$\mathbf{F}_{s,0}$	$\mathbf{F}_{encoder}$	**HF**	**HF** $\perp$	**lidarKITTI [2]** prec. FG ↑ [%]	rec. FG ↑ [%]	prec. BG ↑ [%]	rec. BG ↑ [%]	RAE ↓ [°]	RTE ↓ [m]
seg.	✓	✗	✗	✗	**0.8058**	**0.8895**	**0.9920**	**0.9800**	-	-
seg. + ego.	✓	✓	✗	✗	0.7083	0.8869	0.9918	0.9691	0.1143	0.0389
seg. + ego. + sf.	✓	✓	✗	✗	0.7207	0.8865	0.9913	0.9716	0.1046	0.0398
seg. + ego. + sf.	✓	✗	✓	✗	0.7133	0.8800	0.9916	0.9702	0.1128	0.0422
seg. + ego. + sf.	✓	✗	✗	✓	0.7958	0.8872	0.9917	0.9784	**0.0943**	**0.0293**

3 More Experiments

3.1 Additional Ablation Studies

Verification of Losses: We conduct further experiments to verify our losses. The results are shown in Table 1. Supervision for all points by the basic self-supervised loss for scene flow (marked with ✓(*) in the Table 1) and without the losses of segmentation $\mathcal{L}_{seg}$ and ego-motion $\mathcal{L}_{ego}$ results in extremely inaccurate scene flow. However, integrating both the additional losses significantly improves the scene flow in all metrics. Adding the binary masks to our self-supervised loss, as suggested in the paper, improves the scene flow over FG and BG points even further, as shown in the last row.

Impact of Hybrid Features with Stop Gradient: We verify our decision to develop hybrid features HF $\perp$ with stop gradient by evaluating the segmentation and ego-motion on the lidarKITTI data set [2] in the presence of the ground surface points in Table 2.

First, we verify the accuracy of our segmentation without the ego-motion and scene flow branches by training the segmentation task using only the features extracted by the decoder module $F_{s,0}$. Then, we add the ego-motion branch without scene flow, but using the features from the encoder module of the first feature extraction network $F_{encoder}$. The precision of the segmentation at FG points is negatively affected by the addition of the ego-motion branch. The addition of the scene flow branch slightly improves the segmentation precision at FG points, and the addition of the context encoder using the hybrid features without stop gradients still shows poor precision at FG points. With the stop gradient $\perp$, we improve the overall accuracy of the segmentation almost to the results of the specific-segmentation task 1st row and we also improve the relative angular error RAE and the relative

Table 3: In comparison to RSF [2], our method shows a consistently high accuracy, independent of the data set or the whether the ground surface is included or excluded.

				stereoKITTI [6]				lidarKITTI [6]			
	Method	**Sup.**	**Rigid.**	**EPE3D** ↓ [m]	**Out3D** ↓ [%]	**Acc3DS** ↑ [%]	**Acc3DR** ↑ [%]	**EPE3D** ↓ [m]	**Out3D** ↓ [%]	**Acc3DS** ↑ [%]	**Acc3DR** ↑ [%]
without ground	RSF [2]	*None*	✓	**0.035**	**0.146**	**0.932**	**0.971**	0.085	**0.239**	**0.883**	0.929
	Ours	***Weak***	✗	0.042	0.190	0.874	0.969	**0.069**	0.257	0.857	**0.932**
with ground	RSF [2]	*None*	✓	0.205	0.387	0.735	0.802	0.416	0.767	0.308	0.498
	Ours	***Weak***	✗	**0.039**	**0.212**	**0.922**	**0.966**	**0.049**	**0.267**	**0.918**	**0.964**

translational error RTE.

3.2 Additional Comparison

We compare with the very recent scene flow estimation method, RSF [2], which jointly optimizes a global ego-motion and a set of bounding boxes with their own rigid motions, without using any annotated labels. The RSF [2] approach provides a robust scene flow and outperforms most of the recent scene flow approaches when the ground surface is excluded. However, reliable exclusion of the ground surface is not always possible, may lead to an incomplete representation of the scene. Therefore, we compare our EgoFlowNet with RSF once with excluded ground points, and again when they are present. The comparison is presented in Table 3. We consider the default settings of RSF [2][1] for the evaluation. For the test without ground points, we feed our network with all points including the ground points, but we evaluate all remaining points after removing the ground points. The presence of ground points affects the overall accuracy of the RSF [2] method while our approach still shows a comparable result to RSF [2] when we evaluate without ground points.

In terms of efficiency, RSF [2] takes more than 30 seconds for each point cloud pair, while our EgoFlowNet takes 140*ms* on the same NVIDIA Titan V GPU.

3.3 Limitations

In terms of accuracy, we find that our EgoFlowNet can fail for moving objects that leave the field of view, so that they are partially occluded or disappear in the second LiDAR frame Q. In this case, the scene flow prediction for these areas is often partially or completely wrong. We illustrate such cases in Figure 1. Adding robustness against occlusions remains a challenge for future work.

3.4 Additional Qualitative Results

We visualize our predicted masks and the error maps of scene flow of six examples from stereoKITTI in Figure 2 and another six examples from lidarKITTI in Figure 3.

[1] https://github.com/davezdeng8/rsf

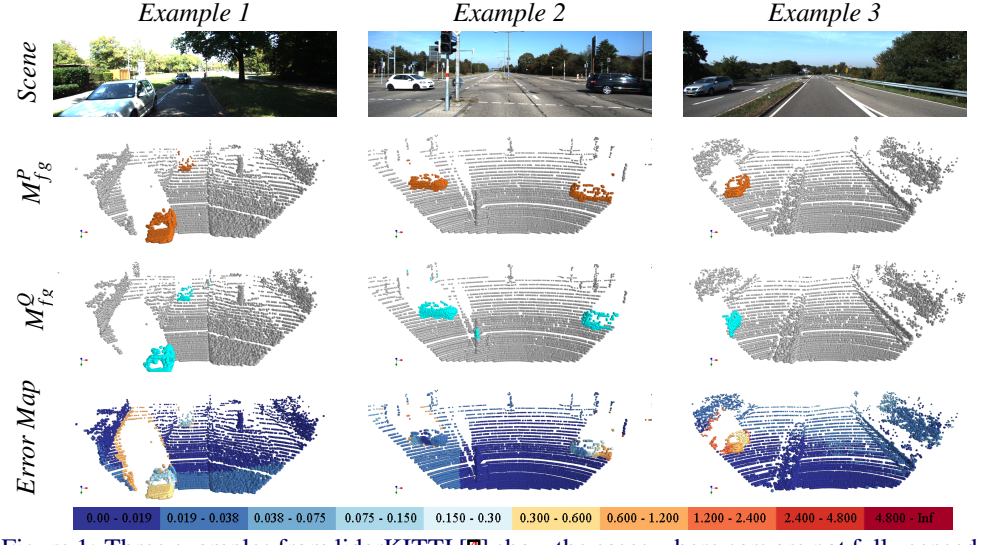

Figure 1: Three examples from lidarKITTI [4] show the cases where cars are not fully sensed in the second frame Q and our scene flow prediction partially fails. For visual enhancement only, we show the RGB images of each scene. We visualize the predicted binary mask, where BG and FG points are encoded by gray and orange or cyan colors, respectively. The error map for each scene (third row) shows the end-point error in meters and is colored according to the map shown in the last row.

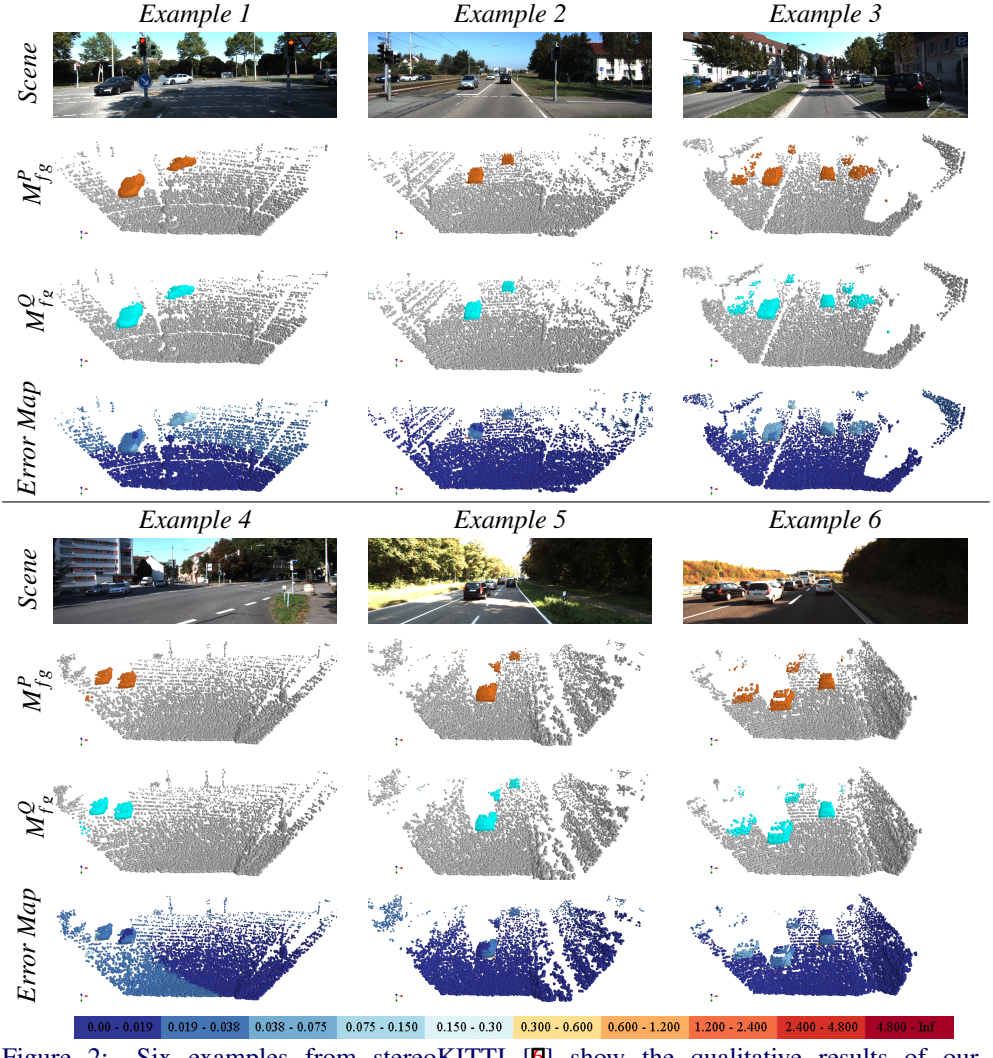

Figure 2: Six examples from stereoKITTI [8] show the qualitative results of our EgoFlowNet.

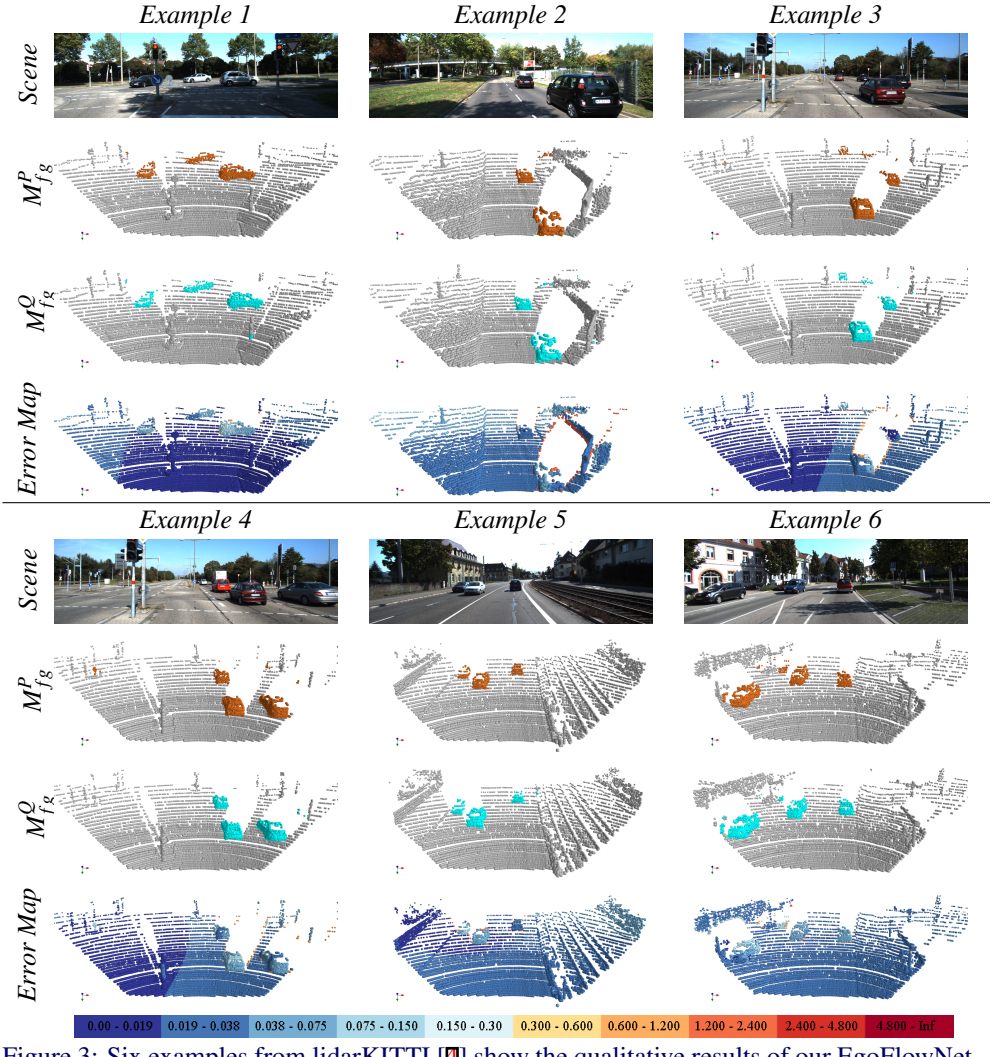

Figure 3: Six examples from lidarKITTI [4] show the qualitative results of our EgoFlowNet.

References

[1] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[2] David Deng and Avideh Zakhor. RSF: Optimizing Rigid Scene Flow From 3D Point Clouds Without Labels. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023.

[3] Guanting Dong, Yueyi Zhang, Hanlin Li, Xiaoyan Sun, and Zhiwei Xiong. Exploiting Rigidity Constraints for LiDAR Scene Flow Estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[4] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[5] Zan Gojcic, Or Litany, Andreas Wieser, Leonidas J Guibas, and Tolga Birdal. Weakly Supervised Learning of Rigid 3D Scene Flow. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[6] Moritz Menze and Andreas Geiger. Object Scene Flow for Autonomous Vehicles. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.