Self-Supervised Pillar Motion Learning for Autonomous Driving
SUPPLEMENTARY MATERIAL

Chenxu Luo[1,2] Xiaodong Yang[1] Alan Yuille[2]
[1]QCraft [2]Johns Hopkins University

In this supplementary material, Section 1 reports an additional experiment to study how our self-supervised pre-trained models benefit to the supervised training under different amounts of training data. Section 2 discusses how to make use of pillar motion to facilitate various downstream tasks. Section 3 describes more details on the differences between our pillar motion prediction and the existing scene flow estimation. Section 4 shows the impact of the smoothness loss term to the overall performance. Section 5 demonstrates the robustness of our approach to the important hyper-parameters. Section 6 presents more qualitative results to illustrate the effect of the proposed probabilistic motion masking design.

1. Self-Supervision for Supervised Training

Our self-supervised learning can be utilized as unsupervised pre-training to improve supervised training. In this section, we investigate the benefits of our self-supervised pre-trained models to supervised training under different amounts of training data with the derived motion annotations. Specifically, we randomly sample 20%, 40%, 60% and 80% of the entire training data. We compare the models trained from scratch against the ones initialized from the self-supervised pre-trained models. Here we summarize the important findings from the comparisons reported in Table 4. (1) It is observed that the self-supervised pre-trained models consistently and significantly outperform the randomly initialized models across all cases. Such improvements are more remarkable under fewer training data and for the fast speed group. (2) Our self-supervised model fine-tuned with a small amount of training data (i.e., 20%) is able to achieve comparable performance compared to the randomly initialized model trained with a large amount of training data (i.e., 80%). This suggests that the model with self-supervised pre-training requires much fewer annotations and is therefore more labeling efficient. (3) We find that the self-supervised pre-trained models converge faster, taking about 60% of the training iterations as the models initialized from scratch.

This work was done while C. Luo was interning at QCraft.

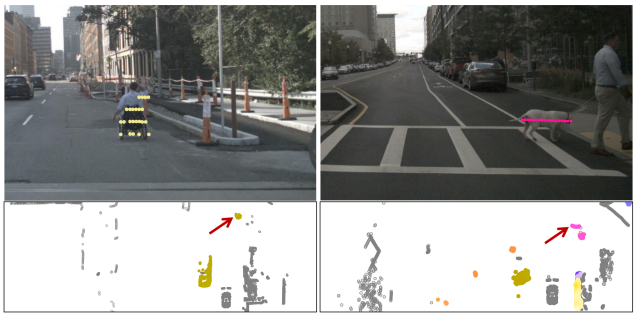

Figure 6. Examples of perceiving the rare objects: wheelchair and dog, which are not seen in the training of point cloud based 3D object detection. We show the results (indicated by the two red arrows) of our self-supervised model, which can correctly estimate the class-agnostic pillar motion.

2. Pillar Motion for Downstream Tasks

Our pillar motion can be potentially applied to enhance a variety of downstream modules. For perception, we show one advantage of our model to deal with the unknown instances that are not seen during the training of 3D object detection, as illustrated in Figure 6. For tracking, it is empirically demonstrated in [17] that integrating the low-level motion information improves the object tracking performance. As for planning, knowing the moving state of an agent is particularly helpful to tackle rare objects although the specific classes are unknown.

3. Comparison with Scene Flow Estimation

In addition to the comparisons described in the introduction and related work of the paper, here we elaborate more on the differences between our work and the previous methods for point cloud based scene flow estimation. First, scene flow aims to estimate the point correspondences between two point clouds, while our goal is to predict the motion of each pillar or the displacement vector that indicates the future position of each pillar. Second, although apart from the synthetic data (e.g., FlyingThings3D), the existing scene flow methods also experiment with the self-driving data (e.g., KITTI Scene Flow), they do not use the

Amount	Self-Supervised	Static		Speed $\leq$ 5m/s		Speed $>$ 5m/s	
		Mean	Median	Mean	Median	Mean	Median
0%	✓	0.1620	0.0010	0.6972	0.1758	3.5504	2.0844
20%	✗	0.0473	0.0001	0.4635	0.1400	2.0946	1.1676
	✓	0.0394	0.0001	0.2970	0.1309	1.028	0.6055
40%	✗	0.0459	0.0001	0.3712	0.1385	1.7060	0.8950
	✓	0.0329	0.0000	0.2813	0.1280	0.8923	0.5287
60%	✗	0.0412	0.0001	0.3082	0.1338	1.0912	0.6830
	✓	0.0352	0.0000	0.2801	0.1297	0.8499	0.5148
80%	✗	0.0347	0.0001	0.2930	0.1322	0.9824	0.6110
	✓	0.0247	0.0000	0.2301	0.0933	0.7788	0.4700

Table 4. Benefits of our self-supervised pre-training under different amounts of training data. ✓: the models are first self-supervised pre-trained and then supervised fine-tuned with the annotations of nuScenes. ✗: the models are randomly initialized from scratch and supervised trained with the annotations of nuScenes. We report the mean and median errors on the three speed groups. Note: no fine-tuning is performed under 0%, which is provided as a baseline reference.

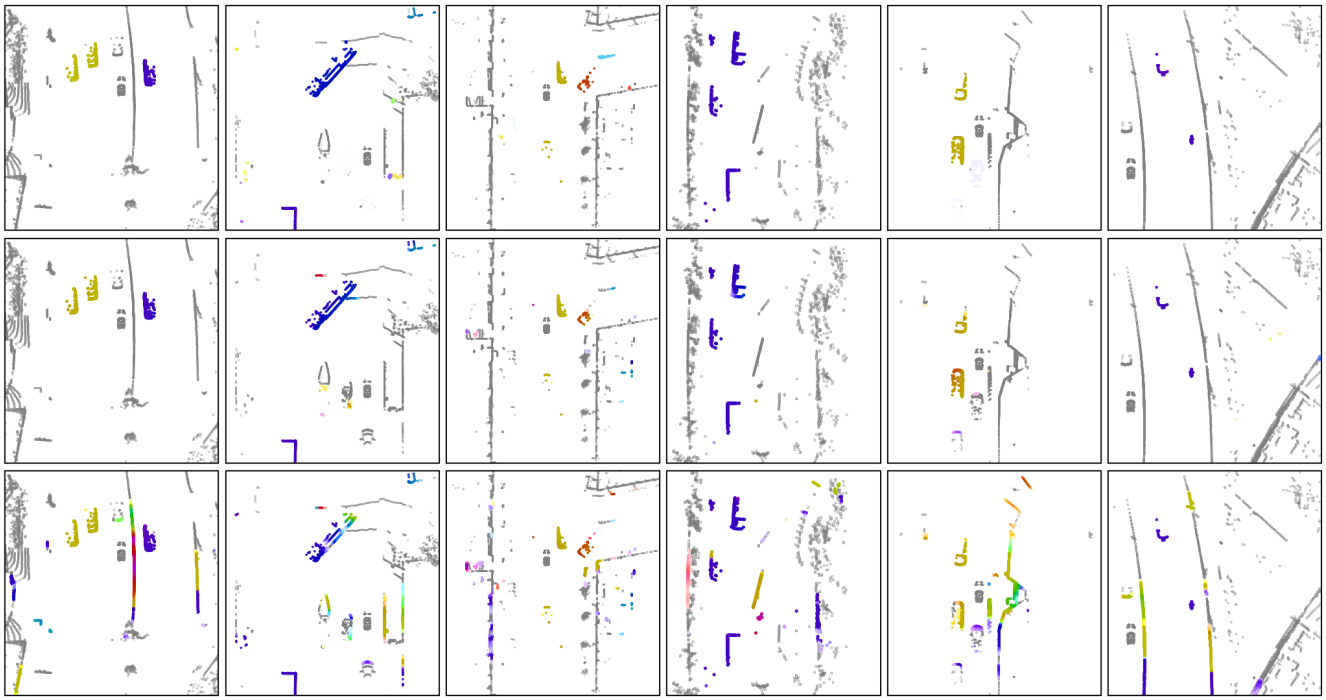

Figure 7. Comparison of the predicted pillar motion. We show the ground truth motion field in the first row, the results estimated by our full model in the second row, and the predictions by the model without using probabilistic motion masking in the third row. Each column demonstrates one scene. We remove the ground points for better visualization.

raw LiDAR scans. Instead, they combine 2D optical flow with depth map and convert them into 3D scene flow. Compared with the point clouds collected by LiDAR, the converted point clouds are much more dense. However, for the raw point clouds used by self-driving vehicles, this does not hold in most cases, making the task harder, in particular for directly doing self-supervision. Third, the prior scene flow methods usually take hundreds of milliseconds when operating on a partial point cloud that is even largely subsampled. Our approach can achieve pillar motion prediction of a complete point cloud in real-time.

4. Ablation Study on Smoothness

Removing the smoothness term in Eq. (8) slightly increases the mean errors, e.g., 0.0058 (Speed $\leq$ 5m/s), 0.0041 (Speed $>$ 5m/s), and 0.0042 (Moving). Overall, the smoothness loss in pillar motion is not as significant as in optical flow. This is due to the fact that the form of pillar motion representation already implies the smoothness prior as each pillar shares the same motion in $0.25\text{m} \times 0.25\text{m}$. In addition, the motion prediction of empty pillars that occupy a large portion of areas can be directly masked out.

5. Hyper-Parameters

We set the hyper-parameters to roughly balance the different loss terms: $\lambda_{\text{consist}} = \lambda_{\text{smooth}} = 1$ and $\lambda_{\text{regular}} = 0.01$. We also experiment with $\lambda_{\text{regular}} = 0.02, 0.03, 0.04, 0.05$. Under the five values of λ_{regular}, the standard deviations of mean errors of the three speed groups are very low: 0.0004, 0.0010 and 0.0070, indicating the robustness of our model to the hyper-parameter setting.

6. More Qualitative Results

Next we provide more qualitative results to reveal the efficacy of the proposed probabilistic motion masking. In Figure 7, we compare the predicted pillar motion fields by our full model and the model without using probabilistic motion masking. As shown in this figure, we present 6 scenes with diverse traffic scenarios and multiple zoom-in scales. In comparison to our full model, the model not using probabilistic motion masking tends to produce more false positive motion predictions at the background regions, such as building, wall and vegetation. This comparison further validates the effect of probabilistic motion masking to reduce the noise incurred by the moving ego-vehicle to the pillars of the background regions.