Supplementary Material: Self-Point-Flow: Self-Supervised Scene Flow Estimation from Point Clouds with Optimal Transport and Random Walk

Table 1. Comparison of our pseudo-label generation algorithm (**PLGA**) with some point registration methods. Without FlowNet3D [3] involved, our **PLGA** outperforms the three point registration methods.

Method	ICP [1]	FGR [10]	CPD [8]	**PLGA**
EPE $(m)\downarrow$	0.406	0.402	0.489	**0.338**

A. Comparison with point registration methods

In this section, we regard our pseudo-label generation algorithm (**PLGA**), $i.e.$ the pseudo label generation module and refinement module in our paper, as a non-deep learning-based 3D point matching algorithm to estimate the scene flow between two point clouds, and compare it with some point cloud registration methods, such as ICP [1], FGR [10] and CPD [8].

In this case, deep neural networks are not involved in the pseudo-label generation algorithm. The pseudo label generation module directly matches points from the first point cloud $\boldsymbol{P}$ to the second point cloud $\boldsymbol{Q}$. And the refined pseudo labels $\widehat{\boldsymbol{D}}$ produced by the refinement module are regarded as the scene flow estimates of our pseudo-label generation algorithm.

This experiment is conducted on the FT3D$_s$ test set [2]. The results of the three point cloud registration methods are given in PointPWC-Net [9]. As shown in Table 1, when applied as a 3D point matching algorithm without FlowNet3D [3] involved, our algorithm outperforms the three point registration methods on the metric EPE.

B. Experimental details

B.1. Implementation Details

When comparing with PointPWC-Net [9], we first train the FlowNet3D model by our self-supervised method on FT3D$_\mathrm{s}$ training set and then evaluate on the test sets of FT3D$_\mathrm{s}$ and KITTI$_\mathrm{s}$, following the experimental settings in [9]. During training, in our pseudo label generation module, we set the iteration number L_o to 4, the regularization parameter ε to 0.03, θ_d to 1.22, and θ_c to 0.35. Because color is unavailable in FT3D$_\mathrm{s}$, we use 3D point coordinate and surface normal to build the transport cost matrix. In our pseudo label refinement module, we set θ_r to 0.63, λ to 0.8, and the iteration number L_r to 5. To speed up the training, for each sample with 8,192 points, we randomly select 2,048 points to produce initial pseudo labels and then use the refinement module to produce a refined pseudo label for each point. After obtaining dense pseudo labels, we use L_2-norm loss for scene flow supervision, and the batch size is 8. The learning rate starts from 0.001 and is multiplied by 0.7 at every 40 epochs.

When comparing with JGF [7], we train the FlowNet3D model by our self-supervised method on KITTI$_\mathrm{r}$. The settings of our two modules are the same as those of the last experiment, except that we use 3D point coordinate, color, and surface normal to build the transport cost matrix, and we set the iteration number of random walk L_r to ∞. Our models are trained from scratch with L_2-norm loss, and the batch size is 16. The learning rate starts from 0.001 and is multiplied by 0.7 at every 10 epochs.

B.2. Details about cycle-consistency regularization

In order to make the paper self-contained, we introduce the cycle-consistency regularization [3], which can be added into our self-supervised training loss.

Given two consecutive point clouds, $\boldsymbol{P}=\{\boldsymbol{p}_i\in\mathbb{R}^3\}_{i=1}^n$ at frame t and $\boldsymbol{Q}=\{\boldsymbol{q}_i\in\mathbb{R}^3\}_{j=1}^n$ at frame $t+1$, the neural network estimates the forward scene flow from $\boldsymbol{P}$ to $\boldsymbol{Q}$ as $\boldsymbol{F}=g(\boldsymbol{P},\boldsymbol{Q};\Theta)$, where $g(\cdot)$ is the neural network with model parameters Θ. Warping the first point cloud $\boldsymbol{P}$ by the predicted forward scene flow $\boldsymbol{F}$, we obtain the pre-warped first point cloud, denoted as $\hat{\boldsymbol{P}}$. And the cycle-consistency regularization is designed to encourage the predicted backward scene flow $\bar{\boldsymbol{F}}=g(\hat{\boldsymbol{P}},\boldsymbol{P};\Theta)$ to be consistent with the reverse of the predicted forward scene flow $\boldsymbol{F}$. The cycle-consistency regularization can be written as:

$$Loss_{cycle}=\|\bar{\boldsymbol{F}}+\boldsymbol{F}\|_2, \tag{1}$$

where $\|\cdot\|_2$ denotes the L_2-norm.

C. More visualizations

C.1. Qualitative comparisons with other self-supervised loss

In this section, we compare our proposed self-supervised learning method with the self-supervised ChamferSmoothCurvature loss proposed in PointPWC-Net [9]. The ChamferSmoothCurvature loss consists of three parts: Chamfer distance, Smoothness constraint, and Laplacian regularization. For comparison, we train a FlowNet3D model by the ChamferSmoothCurvature loss on the FT3D$_s$ training set following the training strategy that is adopted in our self-supervised learning. Qualitative comparisons between our self-supervised learning method and the ChamferSmoothCurvature loss [9] are shown in Fig. 1. This experiment is conducted on the FT3D$_s$ test set.

C.2. Visualizing produced pseudo ground truth

Additional qualitative results of our produced pseudo ground truth on FlyingThings3D [4] and KITTI [6, 5] are shown in Fig. 2.

C.3. Visualizing self-supervised scene flow estimation results

More qualitative results of our produced self-supervised scene flow estimation method on FT3D$_s$ and KITTI$_o$ are shown in Fig. 3.

References

[1] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992. 1

[2] Xiuye Gu, Yijie Wang, Chongruo Wu, Yong Jae Lee, and Panqu Wang. Hplflownet: Hierarchical permutohedral lattice flownet for scene flow estimation on large-scale point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3254–3263, 2019. 1

[3] Xingyu Liu, Charles R Qi, and Leonidas J Guibas. Flownet3d: Learning scene flow in 3d point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 529–537, 2019. 1

[4] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016. 2

[5] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 2, 2015. 2

[6] Moritz Menze, Christian Heipke, and Andreas Geiger. Object scene flow. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140:60–76, 2018. 2

[7] Himangi Mittal, Brian Okorn, and David Held. Just go with the flow: Self-supervised scene flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11177–11185, 2020. 1

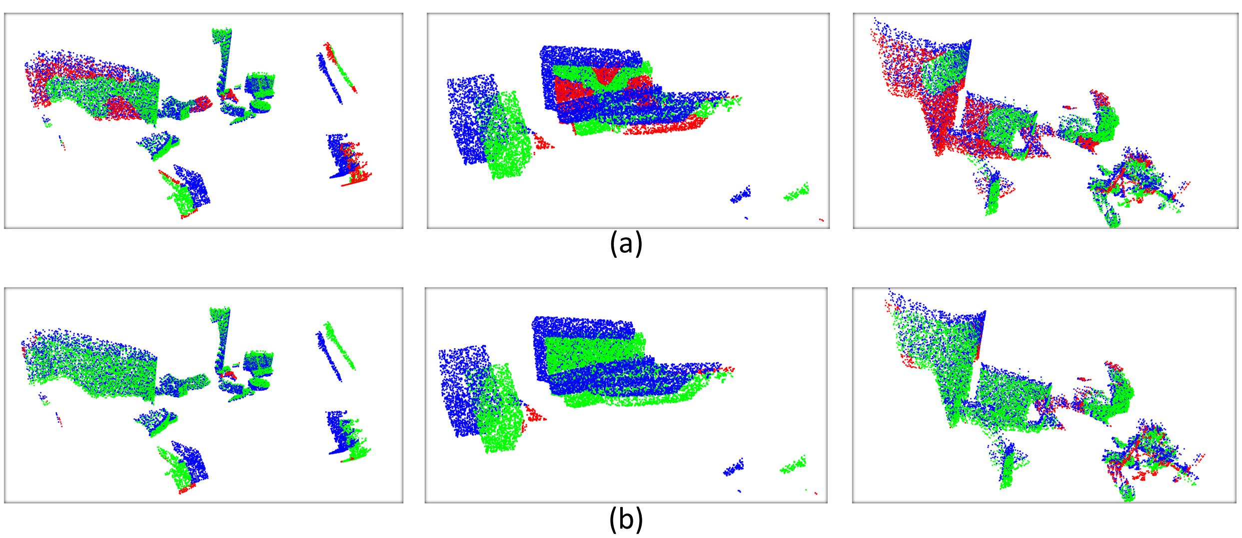

Figure 1. Qualitative comparisons with the self-supervised ChamferSmoothCurvature loss [9]. (a) Results produced by the FlowNet3D model trained by the ChamferSmoothCurvature loss; (b) results produced by the FlowNet3D model trained by our proposed self-supervised learning method. Blue points are the first point cloud $\boldsymbol{P}$. Green points are the points warped by the correctly predicted scene flow. The predicted scene flow belonging to **AR** is regarded as a correct prediction. For the points with incorrect predictions, we use the ground truth scene flow to warp them and the warped results are shown as red points.

[8] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE transactions on pattern analysis and machine intelligence*, 32(12):2262–2275, 2010. 1

[9] Wenxuan Wu, Zhi Yuan Wang, Zhuwen Li, Wei Liu, and Li Fuxin. Pointpwc-net: Cost volume on point clouds for (self-) supervised scene flow estimation. In *European Conference on Computer Vision*, pages 88–107. Springer, 2020. 1, 2

[10] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *European Conference on Computer Vision*, pages 766–782. Springer, 2016. 1

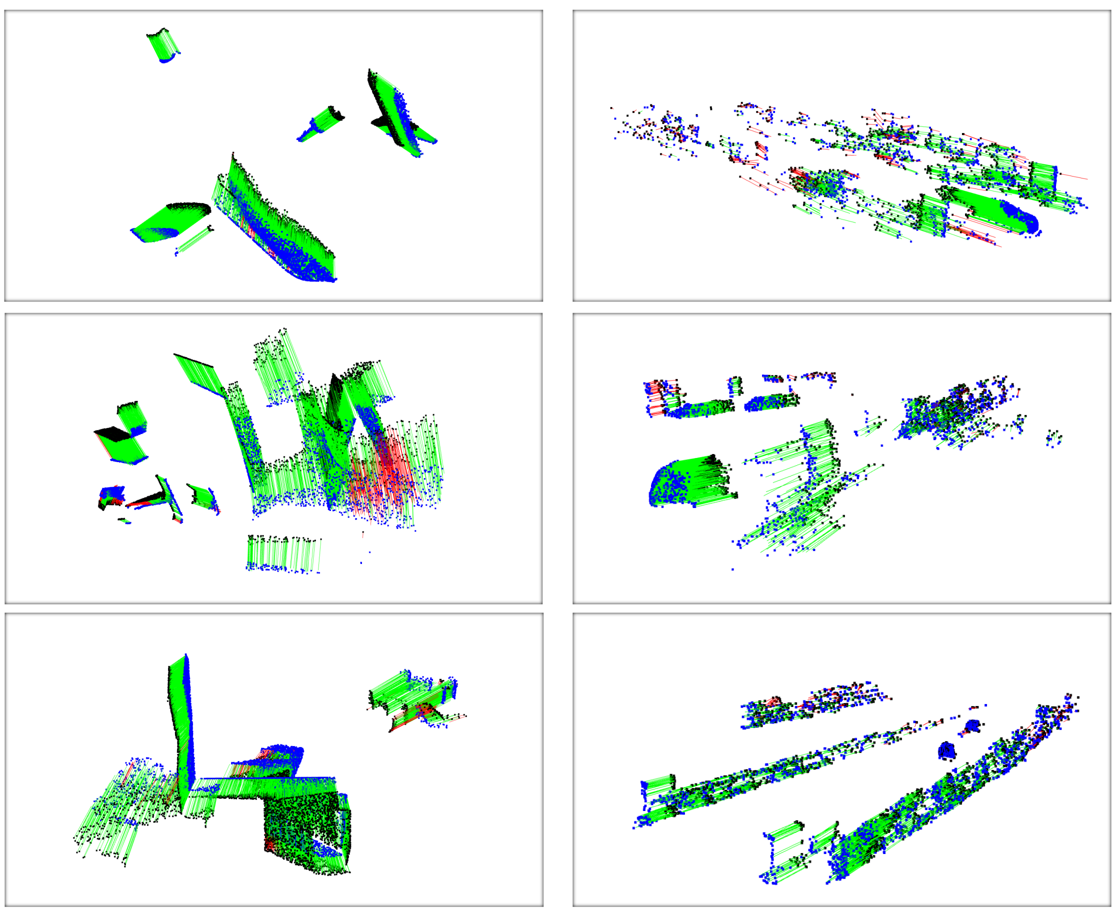

Figure 2. Produced pseudo ground truth on FlyingThings3D (left) and KITTI (right). Blue points are the first point cloud. Black points are the second point cloud. Green line represents the correct pseudo ground truth measured by **AR**. Red line represents the wrong pseudo ground truth.

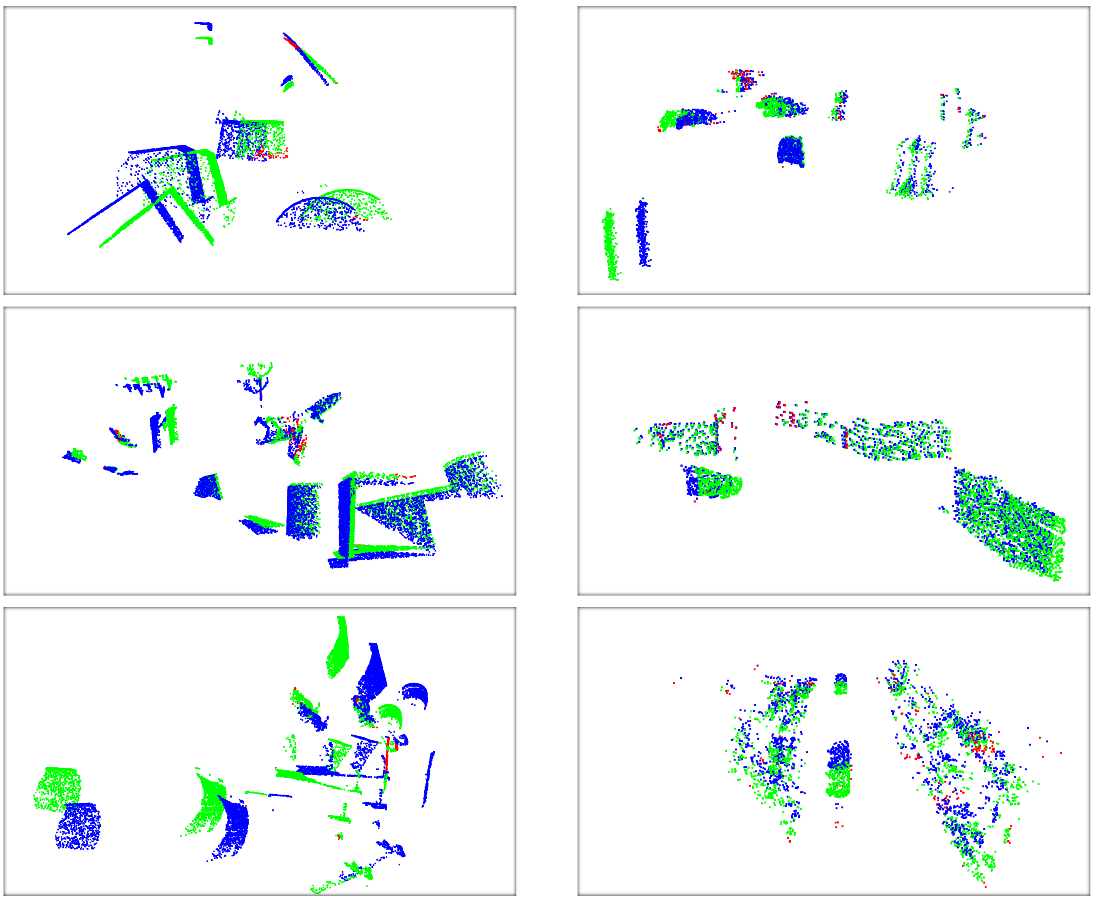

Figure 3. Qualitative results on FlyingThings3D (left) and KITTI (right). Blue points are the first point cloud $\boldsymbol{P}$. Green points are the points warped by the correctly predicted scene flow. The predicted scene flow belonging to $\mathbf{AR}$ is regarded as a correct prediction. For the points with incorrect predictions, we use the ground truth scene flow to warp them and the warped results are shown as red points.