What Matters for 3D Scene Flow Network (Supplementary Material)

Guangming Wang[1], Yunzhe Hu[1], Zhe Liu[2], Yiyang Zhou[3], Masayoshi Tomizuka[3],
Wei Zhan[3], and Hesheng Wang[1]⋆

[1] Department of Automation, Key Laboratory of System Control and Information Processing of Ministry of Education, Key Laboratory of Marine Intelligent Equipment and System of Ministry of Education, Shanghai Jiao Tong University
[2] MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University
[3] Mechanical Systems Control Laboratory, University of California, Berkeley
{wangguangming,huyz7830,liuzhesjtu,wanghesheng}@sjtu.edu.cn
{yiyang.zhou,tomizuka,wzhan}@berkeley.edu

Module		Layer type	K	Sampling rate	MLP width
Hierarchical Point Feature Abstraction		Set conv layer for PC_1 and PC_2 at level $l = 1$	32	0.25	[16,16,32]
		Set conv layer for PC_1 and PC_2 at level $l = 2$	24	0.5	[32,32,64]
		Set conv layer for PC_1 and PC_2 at level $l = 3$	16	0.25	[64,64,128]
		Set conv layer for PC_2 at level $l = 4$	16	0.25	[128,128,256]
All-to-All Point Mixture		All-to-all flow embedding layer	4, 256	1	[256,128,128], [256,128]
		First set conv layer	8	0.25	[128,128,256]
		Second set conv layer	8	0.25	[256,256,512]
Generation of Initial Flow Embedding and Scene Flow		Set upconv layer for initial flow embedding	8	4	[256,256,512],[512]
		FC for initial scene flow	—	1	[3]
Hierarchical Flow Refinement	Flow Refine Layer for E_4 and F_4	Attentive flow re-embedding layer	4, 6	1	[512,256,256], [512,256]
		First set conv layer in flow encoding	16	1	[32,32,32]
		Second set conv layer in flow encoding	8	1	[16,16,16]
		Scene Flow Predictor for E_4	—	1	[512,256,256]
		FC for F_4	—	1	[3]
	Flow Refine Layer for E_3 and F_3	Set upconv layer	8	4	[256,128,128], [128]
		Attentive flow re-embedding layer	4, 6	1	[256,128,128], [256,128]
		First set conv layer in flow encoding	16	1	[32,32,32]
		Second set conv layer in flow encoding	8	1	[16,16,16]
		Scene Flow Predictor for E_3	—	1	[256,128,128]
		FC for F_3	—	1	[3]
	Flow Refine Layer for E_2 and F_2	Set upconv layer	8	4	[256,128,128], [128]
		Attentive flow re-embedding layer	4, 6	1	[128,64,64], [128,64]
		First set conv layer in flow encoding	16	1	[32,32,32]
		Second set conv layer in flow encoding	8	1	[16,16,16]
		Scene Flow Predictor for E_2	—	1	[256,128,128]
		FC for F_2	—	1	[3]
	Flow Refine Layer for E_1 and F_1	Set upconv layer	8	2	[128,64,64], [64]
		Attentive flow re-embedding layer	4, 6	1	[64,32,32], [64,32]
		First set conv layer in flow encoding	16	1	[32,32,32]
		Second set conv layer in flow encoding	8	1	[16,16,16]
		Scene Flow Predictor for E_1	—	1	[128,64,64]
		FC for F_1	—	1	[3]

Table 1. Details on our network parameters. K denotes the number of points of K Nearest Neighbors (KNN) in set conv layer, set upconv layer, and flow embedding layer. Sampling rate means the ratio of the number of output points to the number of input points for one layer. Multi-Layer Perceptron (MLP) width means the number of output channels for each layer of MLP. The variables in the table are defined the same as the main manuscript.

⋆ Corresponding Author. The first two authors contributed equally.

1 Overview

In this supplementary material, the details about our network parameters are presented in Sec. 2. Detailed descriptions about the datasets and data preprocess procedure can be found in Sec. 3. In Sec. 4, the definition of evaluation metrics will be given. Ablation studies on KITTI Scene Flow dataset [4] will be conducted in Sec. 5 to show the effectiveness of our designs across domain. Progressive ablation studies on FlyingThings3D dataset [3] are presented in Sec. 6. Runtime and memory consumption are tested in Sec. 7. We provide more qualitative visualization among different approaches in Sec. 8. Finally, qualitative results of our different ablated models are given in Sec. 9.

2 Network Parameters

In our network, shared Multi-Layer Perceptron (MLP) is implemented by 1×1 convolution with 1 stride and the ReLU activation function is used. Particularly, LeakyReLU activation function is used in FC layer. The slope for negative values equal to 0.1 in the LeakyReLU activation function. The detailed layer parameters including K values in K Nearest Neighbors (KNN), the sampling rate of each layer, and each linear layer width in MLP are described in Table 1.

When testing the generalization ability of our model on KITTI Scene Flow dataset [4] in the manuscript, we double the number of input points to our network as 16,384 points. In addition, the set conv layer at level $l = 1$ in the Hierarchical Point Feature Abstraction module is modified to downsample 4,096 points rather than 2,048 points in the manuscript. The K values of the set conv layers at level $l = 1, 2$ in the module are changed to 64 accordingly. The above modification is only for testing on KITTI Scene Flow dataset [4], and still uses the model trained on FlyingThings3D dataset [3]. We find that the performance on KITTI Scene Flow dataset [4] can be improved in this way, similar to the implementation of FlowNet3D [2] on github [4].

3 Descriptions about Datasets and Data Preprocess

We conduct our training and evaluation on two versions of FlyingThings3D dataset [3] and KITTI scene flow dataset [4] prepared by Gu *et al.* [1] [5] and Liu *et al.* [2] [6], respectively, as mentioned in the manuscript. We provide detailed descriptions on data preprocess as follows.

FlyingThings3D dataset [3] is originally comprised of RGB images, occlusion map, disparity map, and ground-truth optical flow. Gu *et al.* [1] construct the training and evaluation set through recovering the 3D point cloud and ground-truth scene flow from disparity map and ground-truth optical flow. Points with depths greater than $35m$ are removed and there will be 19,640 pairs of point clouds in the training set and 3,824

[4] https://github.com/xingyul/flownet3d.

[5] Datasets available at https://github.com/laoreja/HPLFlowNet.

[6] Datasets available at https://github.com/xingyul/flownet3d.

Table 2. The ablation study results on KITTI Scene Flow dataset prepared by Gu *et al.* [1].

	Method	EPE3D	Acc3D Strict	Acc3D Relax	Outliers	EPE2D	Acc2D
(a)	Ours w/o backward validation	0.0355	0.8788	0.9443	0.1778	1.3028	0.9311
	Ours w/o backward validation and all-to-all mechanism	0.0376	0.8853	**0.9593**	0.1716	1.4732	0.9264
	Ours (full, with backward validation and all-to-all mechanism)	**0.0332**	**0.8931**	0.9528	**0.1690**	**1.2186**	**0.9337**
(b)	Ours (with product similarity)	0.0460	0.8196	0.9262	0.2116	1.7203	0.8943
	Ours (with cosine product similarity)	0.0447	0.8261	0.9285	0.2159	1.8341	0.8731
	Ours (with normalized product similarity)	0.0405	0.8623	0.9478	0.1843	1.5930	0.9175
	Ours (full, with concatenated similarity)	**0.0332**	**0.8931**	**0.9528**	**0.1690**	**1.2186**	**0.9337**
(c)	Ours (replace Scene Flow Predictor with GRU)	0.0374	0.8864	0.9463	0.1780	1.2941	0.9331
	Ours (full, with Scene Flow Predictor)	**0.0332**	**0.8931**	**0.9528**	**0.1690**	**1.2186**	**0.9337**
(d)	Ours w/o features of PC_1 in Scene Flow Predictor	0.0365	0.8775	0.9472	0.1736	1.3482	0.9158
	Ours w/o up-sampled flow embedding in Scene Flow Predictor	0.0399	0.8528	0.9396	0.1936	1.4745	0.9178
	Ours w/o coarse flow in Scene Flow Predictor	0.0352	0.8799	0.9509	0.1709	1.2956	0.9296
	Ours w/o flow feature in Scene Flow Predictor	0.0402	0.8616	0.9414	0.1864	1.4741	0.9099
	Ours (full, with complete five inputs in Scene Flow Predictor)	**0.0332**	**0.8931**	**0.9528**	**0.1690**	**1.2186**	**0.9337**
(e)	Ours (with interpolation estimating 2048 points' flow)	0.0354	0.8837	**0.9587**	0.1704	1.3515	0.9252
	Ours (with interpolation estimating 8192 points' flow)	0.0360	0.8709	0.9452	0.1730	1.3808	0.9216
	Ours (full, with flow refinement estimating 2048 points' flow)	**0.0332**	**0.8931**	0.9528	**0.1690**	**1.2186**	**0.9337**

pairs in the evaluation set. This version of preparation keeps input points without occlusion, which means one can always find a corresponding point in PC_2 for each point in PC_1 warped by its ground-truth scene flow. Liu *et al.* [2], however, provide a more challenging version of preparation, in which points with occlusion are included. They provide masks indicating occluded points that do not have corresponding points in the adjacent frame. For the inference of the network, the occluded points are used as inputs without the help of masks. For the calculation of training loss and evaluation metrics, masks are included. This version of FlyingThings3D dataset contains 20,006 training samples and 2,007 test samples.

KITTI scene flow dataset [4] is another broadly used dataset with real-world data for scene flow estimation. In the preparation of Gu *et al.* [1], points with depth greater than $35m$ are removed, and points with height less than $0.3m$ are treated as the ground and therefore removed as well. No occlusion is retained. Since there is no ground-truth scene flow in the testing set, 142 scenes from the training set are selected for evaluation. In the preparation of Liu *et al.* [2], points with depth greater than $35m$ and ground points are also removed. There is still occlusion for the input points, but no masks are provided. 150 samples from the training set are included for evaluation in this preparation.

As mentioned in the manuscript, the input point clouds to the network are created by random sampling in each frame to simulate real-world data, which does not have direct point-to-point correspondences.

4 Details on Evaluation Metrics

Assume f_i be the estimated scene flow and $GT(f_i)$ be the ground-truth scene flow. The evaluation metrics used in the manuscript are defined as follows:

EPE3D(m): $\frac{1}{N}\sum\limits_{i=1}^{N}\|f_i - GT(f_i)\|_2$.

Table 3. The progressive ablation study results on FlyingThings3D dataset prepared by [1].

	Method	EPE3D	Acc3D Strict	Acc3D Relax	Outliers	EPE2D	Acc2D
(a)	Ours (full, with all investigated beneficial components in ablation studies, **best model**)	**0.0317**	**0.9109**	**0.9757**	**0.1673**	**1.7436**	**0.9108**
(b)	Ours (− backward validation from (a))	0.0332	0.9044	0.9743	0.1766	1.8221	0.9065
(c)	Ours (− all-to-all mechanism from (b))	0.0349	0.9001	0.9725	0.1798	1.9819	0.9032
(d)	Ours (− refining 2048 points' flow replaced with interpolating 2048 points' flow from (c))	0.0375	0.8845	0.9671	0.1944	2.1268	0.8853
(e)	Ours (− Scene Flow Predictor replaced with GRU from (d))	0.0396	0.8691	0.9606	0.2007	2.2522	0.8696
(f)	Ours (− flow feature in Scene Flow Predictor from (d))	0.0384	0.8786	0.9656	0.2046	2.1687	0.8805
(g)	Ours (− coarse flow in Scene Flow Predictor from (f))	0.0390	0.8759	0.9652	0.2071	2.2100	0.8797
(h)	Ours (− features of PC_1 in Scene Flow Predictor from (g))	0.0406	0.8664	0.9639	0.2198	2.2771	0.8748
(i)	Ours (− up-sampled flow embedding in Scene Flow Predictor from (h))	0.0472	0.8169	0.9543	0.2651	2.5917	0.8451
(j)	Ours (− concatenated similarity replaced with cosine product similarity from (i))	0.0553	0.7470	0.9342	0.3314	3.1161	0.7893

Table 4. The ablation study results at a poor baseline on FlyingThings3D dataset prepared by [1].

	Method	EPE3D	Acc3D Strict	Acc3D Relax	Outliers	EPE2D	Acc2D
(a)	Ours (w/o all investigated beneficial components in ablation studies, **baseline model**)	0.0553	0.7470	0.9342	0.3314	3.1161	0.7893
(b)	Ours (+ all-to-all mechanism with backward validation to (a))	0.0494	0.7918	0.9380	0.2885	2.7963	0.8045
(c)	Ours (+ interpolating 2048 points' flow replacing refining 2048 points' flow to (a))	0.0520	0.7810	0.9374	0.3038	2.9657	0.8033
(d)	Ours (+ four input predictor components with GRU Predictor to (a))	0.0449	0.8314	0.9505	0.2501	2.5740	0.8363
(e)	Ours (+ four input predictor components with Scene Flow Predictor to (a))	0.0439	0.8377	0.9548	0.2466	2.5170	0.8403
(f)	Ours (+ cosine product similarity replacing concatenated similarity to (a))	0.0472	0.8169	0.9543	0.2651	2.5917	0.8451

Acc3D Strict: Percentage of f_i such that $\|f_i - GT(f_i)\|_2 < 0.05m$ or $\frac{\|f_i - GT(f_i)\|_2}{\|GT(f_i)\|_2} < 5\%$.

Acc3D Relax: Percentage of f_i such that $\|f_i - GT(f_i)\|_2 < 0.1m$ or $\frac{\|f_i - GT(f_i)\|_2}{\|GT(f_i)\|_2} < 10\%$.

Outliers3D: Percentage of f_i such that $\|f_i - GT(f_i)\|_2 > 0.3m$ or $\frac{\|f_i - GT(f_i)\|_2}{\|GT(f_i)\|_2} > 10\%$.

EPE2D(px): $\frac{1}{N}\sum_{i=1}^{N}\|of_i - GT(of_i)\|_2$. $GT(of_i)$ denotes the ground-truth optical flow, and of_i stands for the estimated optical flow projected from PC_1 and predicted PC_2 via f_i.

Acc2D: Percentage of of_i such that $\|of_i - GT(of_i)\|_2 < 3px$ or $\frac{\|of_i - GT(of_i)\|_2}{\|GT(of_i)\|_2} < 5\%$.

5 Ablation Study on KITTI Scene Flow Dataset

In Sec. 5.2 of our manuscript, we conduct ablation studies on FlyingThings3D dataset [3] prepared by Gu *et al.* [1] to demonstrate the effectiveness of our proposed network designs. In this section, we will provide more ablation studies on KITTI scene flow dataset [4] prepared by Gu *et al.* [1] to demonstrate the effectiveness of our network designs across domain. The models are trained on $\frac{1}{4}$ of the training set (4,910 pairs) from FlyingThings3D [3], the same way as in the manuscript, and then tested on KITTI scene flow dataset [4]. The results in Table 2 demonstrate the generalization ability on real-world data of each design.

6 Progressive Ablation Study

In the manuscript, we demonstrated the effectiveness of each component by ablation experiments based on the best model, but this made the improvement for each component seem small. We change to the progressive ablation experiment by removing each component in turn (Table 3). As can be seen, the integration effect of these components combined yields huge benefits (**42.7% improvement**).

In addition, we investigate the benefits of each component at a poor baseline by adding each component without other improvements (Table 4). The experimental results show that each brings very large benefits (**average 14.1% improvement**). That is, at a better baseline, additional improvements will improve the results less than improvement at a poor baseline. This also explains why the ablation experiment presented in the manuscript showed a small improvement. The accuracy of this task has been greatly improved from the previous poorer baselines, because of the effort we put into every detail.

7 Runtime and Memory

Because our all-to-all is performed on sparse 256 points, it does not increase the consumption of memory and runtime very much (Table 5). The all-to-all cost volume with 1024 points significantly increases the memory and runtime. Therefore, we chose 256 sparse points for our all-to-all initial matching. Compared to other methods, ours has the lowest runtime and memory cost.

Table 5. The runtime and memory. We test on a single Titan RTX GPU for comparison.

Method	Input Point Number	Implementation	Runtime	Memory
HALFlow (TIP 2021) [5]	8192	TensorFlow	111.0ms	5766M
PV-RAFT (CVPR 2021) [6]	8192	PyTorch	212.0ms	5323M
Ours (with KNN mechanism w/o backward validation)	8192	PyTorch	**40.6ms**	**2391M**
Ours (with all-to-all mechanism w/o backward validation)	8192	PyTorch	40.9ms	2789M
Ours (full, with 256 all-to-all mechanism and backward validation)	8192	PyTorch	42.7ms	3045M
Ours (full, with 1024 all-to-all mechanism and backward validation)	8192	PyTorch	166.4ms	7845M

8 More Comparison on Visualization

We also provide more visualization comparison between our proposed method and other methods as shown in Fig. 1. In addition to FlowNet3D [2] and HALFlow [5] that are used in the manuscript for qualitative comparison, a recent state-of-the-art method, PV-RAFT [6], is also included here. The results demonstrate that our approach outperforms previous state-of-the-art methods qualitatively on both FlyingThings3D [3] and KITTI scene flow [4] datasets.

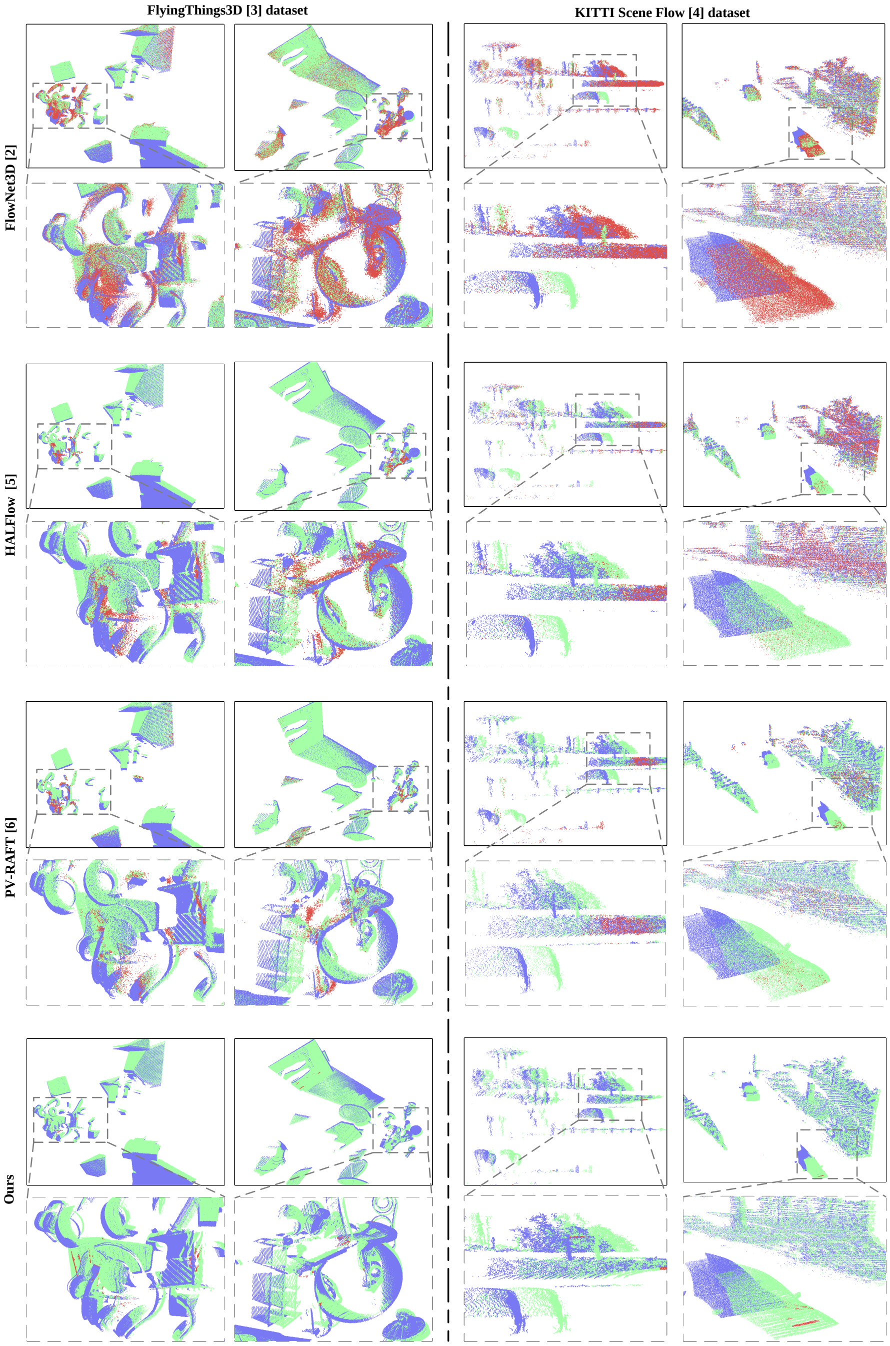

Fig. 1. More visualization results of our proposed method, compared with FlowNet3D [2], HALFlow [5] and PV-RAFT [6], on FlyingThings3D (left) and KITTI scene flow (right) datasets prepared by Gu *et al*. [1]. Blue points indicate PC_1. Green points indicate accurate predictions $\widetilde{PC}_2 = PC_1 + F$ and red points indicate inaccurate predictions (measured by Acc3D Relax).

9 Qualitative Results of Ablation Study

The qualitative results of ablation study (Fig. 2) show that the ablation components can improve the scene flow estimation performance of similar structures, repeated patterns, large motions, fragmented shrubs, etc. The yellow lines show the direction of wrong predictions.

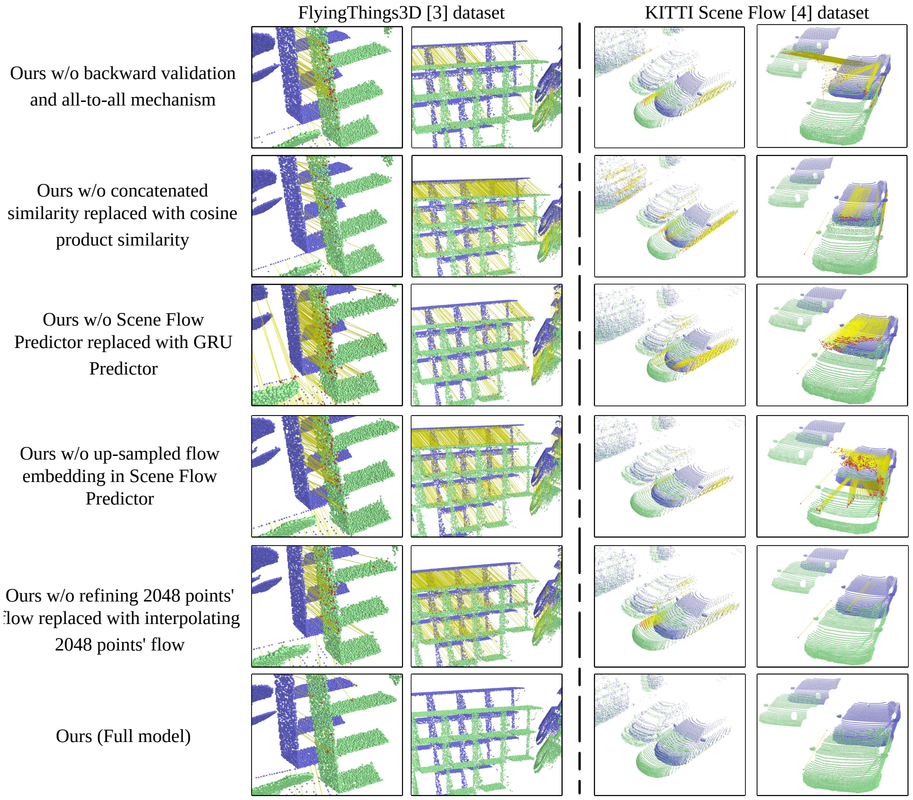
FlyingThings3D [3] dataset
KITTI Scene Flow [4] dataset
Ours w/o backward validation and all-to-all mechanism
Ours w/o concatenated similarity replaced with cosine product similarity
Ours w/o Scene Flow Predictor replaced with GRU Predictor
Ours w/o up-sampled flow embedding in Scene Flow Predictor
Ours w/o refining 2048 points' low replaced with interpolating 2048 points' flow
Ours (Full model)

Fig. 2. Each of ablation components contributes to performance improvement. Different components enhance qualitative results in different details.

References

1. Gu, X., Wang, Y., Wu, C., Lee, Y.J., Wang, P.: Hplflownet: Hierarchical permutohedral lattice flownet for scene flow estimation on large-scale point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3254–3263 (2019)
2. Liu, X., Qi, C.R., Guibas, L.J.: Flownet3d: Learning scene flow in 3d point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 529–537 (2019)

3. Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: 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. pp. 4040–4048 (2016)
4. Menze, M., Heipke, C., Geiger, A.: Object scene flow. ISPRS Journal of Photogrammetry and Remote Sensing **140**, 60–76 (2018)
5. Wang, G., Wu, X., Liu, Z., Wang, H.: Hierarchical attention learning of scene flow in 3d point clouds. IEEE Transactions on Image Processing **30**, 5168–5181 (2021)
6. Wei, Y., Wang, Z., Rao, Y., Lu, J., Zhou, J.: Pv-raft: point-voxel correlation fields for scene flow estimation of point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6954–6963 (2021)