Implicit Surface Contrastive Clustering for LiDAR Point Clouds Supplementary Material

Zaiwei Zhang*
Nuro Inc
zaizhang@nuro.ai

Min Bai
AWS AI
baimin@amazon.com

Erran Li
AWS AI
lilimam@amazon.com

In the following, we first document the model architecture details used in our experiments, and discuss the implementation details for finetuning on both detection and segmentation task in Section A. Next we provide further details regarding the unsupervised semantic grouping task. Lastly, we show additional fine-grained experimental results in Section B.

A. Network Architecture and Implementation Details

Object detection for KITTI We adopt the same training and testing configurations in OpenPCDet [4] for the object detection task with the Part-A^2 model. We apply Adam-Onecyle [2] optimizer and use a base learning rate 0.003 with a 0.1x weight decay at 35 and 45 epochs. The model is trained for 80 epochs and we use a batch size of 4 per GPU. The model is trained with 4 NVIDIA V100 GPUs. We use the same configuration for training from scratch and finetuning from a pretrained model. We only load the pretrained weight for 3D Unet backbone used in OpenPCDet [4].

Semantic segmentation for SemanticKITTI and Waymo For segmentation task, we use the same 3D sparse convolutional U-Net backbone as in JS3C [6]. In order to support the pretraining task which reasons about point level feature embeddings, we project the backbone network's voxel feature outputs directly onto the original input points for subsequent reasoning. For fine-tuning, we attach a pointwise two-layer trainable MLP to the backbone to convert the pointwise feature embeddings to the required number of channels for the semantic segmentation task.

We slightly modify the number of epochs used for training since we have reduced the training dataset size greatly for our evaluation experiments. The model is trained for 800, 600, 600, and 500 epochs on the 1%, 2%, 5%, and 10% splits, respectively. We apply Adam [1] optimizer and use a base learning rate 0.001 with a 0.7x weight decay at every tenth of the training process. We use batch size 8 for each GPUs and the model is also trained with 4 GPUs. We apply the same configurations for training from scratch and finetuning experiments for both datasets.

*Work done at AWS AI.

B. More Results

More results on unsupervised semantic grouping We provide some additional per-category results on unsupervised semantic grouping for SemanticKITTI in Table 1. To save space in the main paper, we combine the results for bicycle and motorcycle and show the mean result as *Cycle*, and also combine the results for bicyclist and motorcyclist and show the mean result as *Cyclist*. As mentioned in the paper, for the less appearing objects, our model does perform a bit worse. However, as shown in Table 1, our approach is able to achieve the best result for other-vehicle class, which shows that our approach consistently extract expressive features for vehicles. For other classes related to the ground points, our approach is able to achieve the best result as the mean over all classes.

More results for 3D object detection on KITTI We show the detection results for *Easy* and *Hard* objects in the KITTI dataset in Table 2 and Table 3, respectively. For all the detection results in the paper, we set 0.7 as the IoU threshold in each axis for *Car* and 0.5 as the IoU threshold in each axis for *Pedestrian* and *Cyclist*. We can observe that similar to the results in the main paper, our approach performs the best in majority of the settings. For objects with hard occlusions, as shown in Table 3, our approach performs significantly better for all three classes. This suggests that our approach is able to infer missing shape information and thus provides larger gain for objects with strong occlusions.

Per-category results for semantic segmentation on SemanticKITTI In Table 4, we provide the detailed per-

	BiC	MotC	BiCl	MotCl	OtherV	Road	Park	SideW	Terrain	OtherG	Mean **
GT	16.6	77.2	8.9	90.2	84.9	37.8	0.1	31.3	32.1	3.8	38.3
No Pretraining	1.4	4.0	0.0	0.0	3.2	18.5	0.0	31.8	**16.9**	**2.4**	7.8
PointContrast [5]	1.2	**5.0**	1.3	4.1	3.4	34.9	**0.1**	0.0	15.1	1.3	6.6
DepthContrast [8]	1.1	3.3	0.0	0.0	4.7	31.3	0.0	26.2	2.0	1.8	7.0
SegContrast [3]	1.2	3.8	**2.8**	**6.8**	2.3	30.6	0.0	**28.1**	0.1	1.7	7.7
SSPL [7]	0.8	1.4	1.3	2.5	4.1	**40.2**	0.0	0.1	0.7	0.9	5.2
Ours	**3.6**	4.6	2.6	5.2	**5.0**	33.7	0.0	21.2	1.0	2.0	**7.9**

Table 1. Unsupervised semantic grouping on other classes in SemanticKITTI Dataset (IoU). BiC and MotC are bicycle and motorcycle. BiCl and MotCl are bicyclist and motorcyclist. OtherV and OtherG are other-vehicle and other-ground. Park is refers to the parking class and SideW refers to the sidewalk class. ** the mean is only over the listed classes here. Please see the main paper for the mean of the prominent classes.

	Car (Easy)				Pedestrian (Easy)				Cyclist (Easy)			
Self-Supervision Method	5%	10%	20%	50%	1%	2%	5%	10%	1%	2%	5%	10%
None	77.8	85.2	87.7	88.8	54.6	**65.1**	66.0	66.3	67.6	81.4	82.2	86.8
PointContrast [5]	79.1	85.8	86.9	88.1	55.0	64.9	65.3	66.2	69.1	82.1	83.2	87.2
DepthContrast [8]	80.0	86.6	88.2	88.7	54.7	62.6	64.3	65.4	70.4	82.8	83.7	87.4
SegContrast [3]	80.5	86.9	88.0	**88.9**	53.2	63.4	64.5	65.7	69.8	82.6	84.1	87.6
SSPL [7]	79.8	86.1	87.8	87.2	53.4	62.7	64.3	65.9	69.3	82.3	83.0	87.0
Ours	**83.5**	**87.5**	**88.6**	88.8	**55.2**	64.1	**67.2**	**67.8**	**73.8**	**83.2**	**86.5**	**88.1**

Table 2. 3D object detection fine-tuning performance on sub-sampled KITTI Dataset (mAP_R11)

category semantic segmentation results for finetuning with 1% annotations. Our approach performs better in majority classes. For *truck*, our approach achieves more than 10% absolute gain over the top performing baseline method. It shows that our approach is able to extract sharp features for some uniquely shaped objects.

Per-category results for semantic segmentation on Waymo Open Dataset In Table 5, we provide the detailed per-category results for finetuning with 1% annotations on waymo open dataset. Similarly, our approach performs better in most of classes. For *bus* class, finetuning with our approach provides more than 5% gain over the top performing baseline method, which emphasizes the benefits of feature learning with our approach.

References

[1] Andrew Adams, Jongmin Baek, and Myers Abraham Davis. Fast high-dimensional filtering using the permutohedral lattice. In *Computer Graphics Forum*, volume 29, pages 753–762. Wiley Online Library, 2010. 1

[2] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

[3] Lucas Nunes, Rodrigo Marcuzzi, Xieyuanli Chen, Jens Behley, and Cyrill Stachniss. Segcontrast: 3d point cloud feature representation learning through self-supervised segment discrimination. *IEEE Robotics and Automation Letters*, 7(2):2116–2123, 2022. 2, 3, 4

[4] OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020. 1

[5] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas J Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 3, 4

[6] Xu Yan, Jiantao Gao, Jie Li, Ruimao Zhang, Zhen Li, Rui Huang, and Shuguang Cui. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3101–3109, 2021. 1

[7] Zaiwei Zhang, Min Bai, and Erran Li. Self-supervised pre-training for large-scale point clouds. In *NeurIPS 2022*, 2022. 2, 3, 4

[8] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-supervised pretraining of 3d features on any point-cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10252–10263, 2021. 2, 3, 4

	Car (Hard)				Pedestrian (Hard)				Cyclist (Hard)			
Self-Supervision Method	5%	10%	20%	50%	1%	2%	5%	10%	1%	2%	5%	10%
None	51.8	66.6	69.6	74.5	41.4	**51.7**	52.6	54.4	41.1	55.9	56.8	67.6
PointContrast [5]	55.4	67.6	66.2	74.8	41.6	51.4	54.5	55.2	45.3	55.3	58.7	67.9
DepthContrast [8]	58.3	68.2	72.3	75.6	41.4	48.8	53.8	54.1	47.5	56.1	60.2	68.5
SegContrast [3]	59.1	68.4	72.0	75.8	41.1	51.0	53.9	54.7	47.0	55.8	60.5	69.0
SSPL [7]	57.6	67.9	70.9	74.0	41.3	49.0	53.7	54.9	46.4	55.6	59.3	68.8
Ours	**67.6**	**69.6**	**74.1**	**77.7**	**41.9**	50.1	**54.8**	**55.4**	**50.8**	**56.5**	**65.9**	**70.5**

Table 3. 3D object detection fine-tuning performance on sub-sampled KITTI Dataset (mAP_R11)

Pretraining	None	PointContrast [5]	DepthContrast [8]	SegContrast [3]	SSPL [7]	Ours
all	38.9	41.1	39.2	42.2	42.6	**45.0**
car	90.9	90.1	90.6	91.7	91.5	**92.6**
bicycle	**3.1**	0.8	1.6	1.7	1.1	2.1
motorcycle	5.1	2.1	4.0	3.0	3.7	**10.1**
truck	15.9	35.7	16.3	34.0	29.0	**47.9**
other-vehicle	13.1	15.4	14.5	**28.3**	23.4	26.6
person	28.2	25.3	24.0	31.9	**35.1**	31.4
bicyclist	11.1	8.1	13.0	12.4	**18.0**	9.9
motorcyclist	**0.0**	0.0	0.0	0.0	0.0	0.0
road	87.6	87.0	86.8	88.6	88.8	**90.8**
parking	22.0	19.0	22.2	25.9	22.8	**29.2**
sidewalk	66.4	67.2	65.6	66.5	69.5	**74.0**
other-ground	**1.0**	0.1	0.3	0.5	0.2	0.9
building	84.2	87.1	86.1	85.9	86.3	**87.9**
fence	34.0	**43.6**	39.6	39.1	40.9	43.4
vegetation	82.7	83.2	83.5	83.9	84.4	**85.2**
trunk	46.5	54.7	48.4	55.7	52.6	**56.0**
terrain	66.4	67.1	67.5	64.8	69.5	**72.0**
pole	43.3	**54.0**	49.7	53.4	51.6	52.7
traffic-sign	38.2	**41.9**	31.3	35.4	41.0	41.3

Table 4. Detailed semantic segmentation results on 1% SemanticKITTI (mIoU)

Pretraining	None	PointContrast [5]	DepthContrast [8]	SegContrast [3]	SSPL [7]	Ours
all	42.0	43.8	42.7	43.5	44.7	**46.0**
car	87.8	89.8	89.0	90.0	89.8	**90.9**
truck	40.0	45.3	44.4	**46.5**	44.1	45.5
bus	34.0	40.0	41.2	40.4	37.2	**47.6**
other vehicle	4.3	3.3	3.3	5.1	**6.7**	5.7
motorcyclist	0.2	0.0	0.0	**0.5**	0.0	0.3
bicyclist	15.9	20.6	14.8	18.6	**21.2**	20.7
pedestrian	67.5	70.7	70.5	69.8	72.3	**73.2**
sign	48.4	49.8	49.6	49.7	49.9	**50.9**
traffic light	23.3	21.9	20.9	21.4	23.0	**23.9**
pole	54.1	54.5	53.8	53.6	55.1	**55.7**
construction cone	27.4	30.5	24.0	28.7	29.8	**31.0**
bicycle	10.1	12.1	9.7	11.0	18.2	**18.3**
motorcycle	19.1	23.4	23.1	23.0	22.5	**27.4**
building	88.7	89.3	89.1	89.5	89.7	**90.2**
vegetation	80.9	82.1	81.7	82.1	82.4	**83.0**
tree trunk	53.4	54.8	53.1	55.2	55.2	**56.3**
curb	48.1	49.3	47.9	48.7	50.7	**52.2**
road	81.8	84.2	83.1	83.9	**85.3**	85.1
lane marker	19.1	20.8	19.8	19.4	23.1	**25.3**
other ground	10.5	10.2	11.0	8.1	11.3	**13.1**
walkable	57.8	59.1	58.3	58.9	61.1	**61.2**
sidewalk	50.9	52.7	51.2	52.1	54.4	**55.5**

Table 5. Detailed semantic segmentation results on 1% Waymo Open Dataset (mIoU)