Rethinking Data Augmentation for Robust LiDAR Semantic Segmentation in Adverse Weather

Supplementary Material

In this appendix, we supplement more materials to support the findings and conclusions in the main body of this paper. Specifically, this appendix is organized as follows. Section A provides additional explanations of Selective Jittering. Section B provides detailed explanations for Learnable Point Drop. Section C presents toy experiments conducted with SynLiDAR. Section D details class-level results. Section E offers further ablation studies. Section F provides additional qualitative results.

A Detailed Explanation for Selective Jittering

Selective Jittering is devised to address the primary distortion of geometric perturbation and consists of two types: Depth-Selective Jittering (DSJ) and Angle-Selective Jittering (ASJ). DSJ adds noise to points along discretized depth levels. The number of depth levels, randomly selected within a specific range, is a hyperparameter. ASJ perturbs points along angles, with the maximum angular range as a hyperparameter. In addition, Range Jittering (RJ) perturbs points in the range direction with Gaussian noise, which DSJ and ASJ do not.

B Detailed Explanation for Learnable Point Drop

(1) Concept: In Section 3, we claimed that adverse weather causes point drops and degrades performance. Therefore, once critical point drops that degrade the model performance are identified, we presume it can sufficiently mimic the effects of adverse weather. This is why LPD receives a reward as $L_{LPD} - L_{aug} + H_{LPD} - H_{aug}$, encouraged to find adverse point drops.
(2) Loss: The update loss in LPD is identical to that in the original DQN paper [5], with reward r, discount rate γ, frozen target model Q' and policy model Q:

$$L_{DQN} = \mathbb{E}_{i,s,a}\left[\left(r + \gamma \max_{a'} Q'(s', a'; \theta_{i-1}) - Q(s, a; \theta_i)\right)^2\right]. \tag{1}$$

(3) The network needs only one backpropagation, and our training with LPD is performed only once. The loss of LPD from eqn. 1 and L_{aug} and L_{LPD} are combined and backpropagated together:

$$L_{total} = L_{aug} + L_{LPD} + L_{DQN}. \tag{2}$$

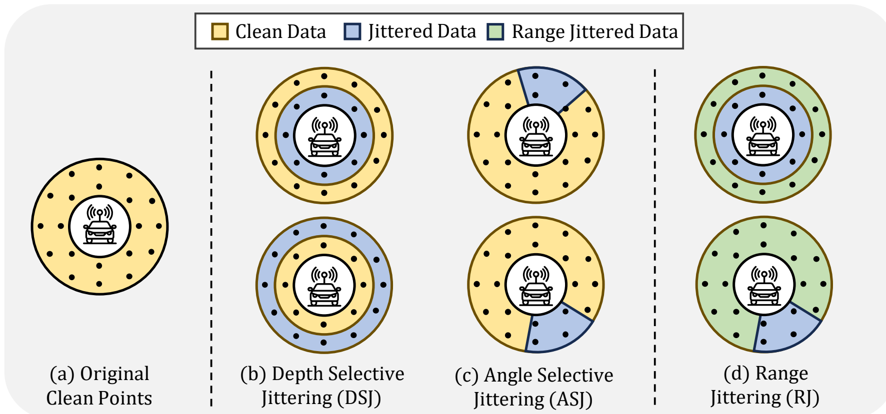

Fig. A.1: Illustration of proposed selective jittering. (a) Original points from clean data. (b) Depth-selective jittering (DSJ) adds Gaussian noise to a random range of depths. (c) Angle-selective jittering (ASJ) adds Gaussian noise to a random range of angles. (d) Range Jittering adds Gaussian noise in the range direction and is used alongside points treated with DSJ or ASJ.

C Toy Experiment in SynLiDAR

In the main paper, the toy experiment conducted on the SemanticKITTI [1] *validation set* was also performed using SynLiDAR [8]. As illustrated in Table A.1, the toy experiment in SynLiDAR yielded results akin to those observed in SemanticKITTI. Additionally, a detailed representation of the toy example's outcomes in SynLiDAR can be examined through Figure A.2.

Table A.1: Results of toy experiments from *validation set* of SynLIDAR [8].

Methods	car	bi.cle	mt.cle	truck	oth-v.	pers.	bi.clst	mt.clst	road	parki.	sidew.	oth-g.	build.	fence	veget.	trunk	terra.	pole	traf.	mIoU
Clean	0.0	54.9	94.1	0.0	0.0	94.8	93.7	0.0	94.8	0.0	89.8	0.0	98.2	53.7	76.8	90.3	0.0	86	95.2	53.8
D1 : soft	0.0	37.0	86.7	0.0	0.0	89.5	87.8	0.0	92.6	0.0	85.1	0.0	93.8	32.4	39.5	80.9	0.0	84.7	93.0	50.2
D1 : hard	0.0	9.1	53.9	0.0	0.0	56.0	48.6	0.0	20.9	0.0	23.7	0.0	31.0	0.4	4.8	26.7	0.0	39.6	74.4	**21.6**
D2 : soft	0.0	4.2	86.7	0.0	0.0	88.2	87.8	0.0	92.5	0.0	48.4	0.0	52.1	7.0	1.4	68.8	0.0	28.8	26.8	31.2
D2 : hard	0.0	0.2	52.5	0.0	0.0	50.3	48.4	0.0	20.6	0.0	3.1	0.0	4.0	0.2	0.1	10.5	0.0	2.2	8.7	**11.1**
D3 : soft	0.0	55.7	79.6	0.0	0.0	86.5	87.4	0.0	45.0	0.0	50.4	0.0	55.3	22.3	6.8	74.1	0.0	69.5	29.1	34.8
D3 : hard	0.0	12.5	11.3	0.0	0.0	63.3	32.9	0.0	0.2	0.0	3.5	0.0	6.6	1.9	2.8	45.6	0.0	4.1	0.1	**9.7**
D4 : soft	0.0	54.8	94.1	0.0	0.0	94.8	93.8	0.0	94.8	0.0	89.8	0.0	98.2	53.8	76.6	90.2	0.0	86	95.2	53.8
D4 : hard	0.0	54.8	94.1	0.0	0.0	94.8	93.8	0.0	94.8	0.0	89.8	0.0	98.2	53.8	76.6	90.2	0.0	86	95.2	53.8

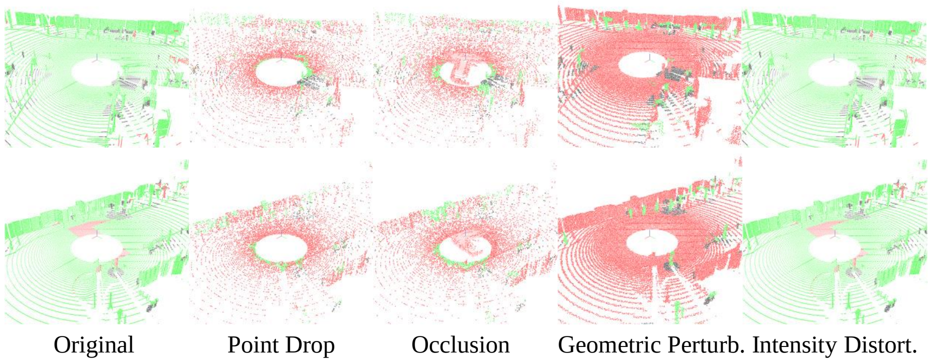

Fig. A.2: Qualitative results of toy experiments in SynLiDAR [8]. MinkUnet is employed and trained in *train set* of the SynLiDAR dataset. The red points represent incorrect predictions, whereas the green points indicate correct predictions.

D Detailed Class-level Results

We provide detailed class IoU for the Table 2 of the main paper. As seen in Table A.2 for SemanticKITTI→SemanticSTF, our method has significantly improved the performance of categories such as *other vehicle*, *motorcyclist*, *sidewalk*, *pole*, and *traffic sign*, which initially had lower baseline performance. Specifically, it achieved increases of +12.0 IoU for *other vehicle*, +36.6 IoU for *motorcyclist*, +11.9 IoU for *sidewalk*, +4.9 IoU for *pole*, and +28.2 IoU for *traffic sign*. This demonstrates that our methodology is effective in predicting classes that are likely to be overlooked due to adverse weather conditions. Furthermore, the superiority of our method is evident as it shows better performance than the existing state-of-the-art method, PointDR, across most classes.

Table A.2: LiDAR segmentation results (mIoU) on the SemanticSTF validation set of models trained with SemanticKITTI. D-fog and L-fog denote dense fog and light fog weather conditions in all experiments. * is the reproduced result with the same segmentation backbone. The best score is in **bold** and the second best is <u>underlined</u>.

Methods	car	bi.cle	mt.cle	truck	oth-v.	pers.	bi.clst	mt.clst	road	parki.	sidew.	oth-g.	build.	fence	veget.	trunk	terra.	pole	traf.	mIoU
Oracle	89.4	42.1	0.0	59.9	61.2	69.6	39.0	0.0	82.2	21.5	58.2	45.6	86.1	63.6	80.2	52.0	77.6	50.1	61.7	54.7
Baseline	67.1	<u>5.0</u>	**28.1**	38.5	<u>14.6</u>	45.8	8.3	13.8	40.1	**16.1**	26.1	3.3	71.6	**52.7**	53.8	33.9	39.2	<u>25.3</u>	12.7	31.4
LaserMix [4]	18.6	**5.4**	0.0	9.9	1.6	0.6	7.9	10.5	47.6	6	12.1	1.8	21.6	20.2	48.4	6.6	37.8	19	2.8	14.7
PolarMix [7]	21	2	0.0	3.8	1.6	2.8	0.6	0.0	<u>58.3</u>	4.4	17.4	1.4	40.7	36.4	41.3	6.6	35	14.6	2.8	15.3
PointDR* [9]	<u>69.2</u>	1	8.9	**41.9**	7.6	<u>48.9</u>	**17.0**	<u>36.2</u>	57.8	<u>15.9</u>	<u>32.3</u>	<u>4.0</u>	**75.7**	46.4	<u>54.0</u>	<u>36.2</u>	<u>43.9</u>	23.7	<u>24.2</u>	<u>33.9</u>
Baseline+SJ+LPD	**86.1**	4.8	<u>13.8</u>	<u>39.7</u>	**26.6**	**55.4**	<u>8.5</u>	**50.4**	**63.7**	14.9	**37.9**	**5.5**	<u>75.2</u>	**52.7**	**60.4**	**39.7**	**44.9**	**30.1**	**40.8**	**39.5**
Increments to baseline	+19.0	-0.2	-14.3	+1.1	+12.0	+9.6	+0.2	+36.6	+23.5	-1.2	+11.9	+2.2	+3.6	0.0	+6.7	+5.8	+5.7	+4.9	+28.2	+8.1

Table A.3: LiDAR segmentation results (mIoU) on the SemanticSTF validation set of models trained with SynLiDAR. D-fog and L-fog denote dense fog and light fog weather conditions in all experiments. * is the reproduced result with the same segmentation backbone. The symbol ‡ indicates that the validation for model selection was performed on sequence 0 of SynLiDAR, rather than SemanticSTF. The best score is in **bold** and the second best is <u>underlined</u>.

Methods	car	bi.cle	mt.cle	truck	oth-v.	pers.	bi.clst	mt.clst	road	parki.	sidew.	oth-g.	build.	fence	veget.	trunk	terra.	pole	traf.	mIoU
Oracle	89.4	42.1	0.0	59.9	61.2	69.6	39.0	0.0	82.2	21.5	58.2	45.6	86.1	63.6	80.2	52.0	77.6	50.1	61.7	54.7
Baseline	33.76	1.71	<u>3.29</u>	15.54	0.24	25.52	1.65	3.43	15.27	9.16	16.76	0.05	33.38	21.89	39.49	18.7	**44.03**	8.75	0.84	15.45
LaserMix [4]	**52.41**	**5.6**	1.05	7.91	<u>1.96</u>	25.59	1.62	2.92	**44.58**	6.44	<u>21.21</u>	0.88	16.95	23.61	40.75	18.91	41.45	5.65	0.7	16.85
PolarMix [7]	<u>48.93</u>	<u>4.23</u>	2.32	14.64	**2.37**	24.55	2.14	4.64	34.64	7.66	19.8	0.39	37.44	22.3	44.85	21.32	<u>43.18</u>	7.08	1.3	18.09
PointDR* [9]	41.07	2.81	**3.43**	18.05	0.17	**31.3**	**2.81**	3.3	34.39	**10.23**	19.72	<u>0.96</u>	**52.72**	21.98	**48.49**	21.33	38.31	19.19	5.61	19.78
PointDR*‡ [9]	36.13	3.47	2.15	**21.93**	0.31	28.72	1.69	**5.09**	<u>42.92</u>	9.31	20.71	0.58	50.83	<u>26.88</u>	46.85	**24.49**	37.69	<u>22.74</u>	6.45	<u>20.47</u>
Baseline+SJ+LPD	42.13	2.79	2.68	19.22	0.67	<u>29.22</u>	1.91	<u>4.8</u>	42.32	8.67	21.05	**1.56**	48.23	25.97	47.17	22.11	32.8	21.74	<u>6.54</u>	20.08
Increments to baseline	+8.4	+1.1	-0.6	+3.7	+0.4	+3.7	+0.3	+1.4	+27.0	-0.5	+4.3	+1.5	+14.9	+4.1	+7.7	+3.4	-11.2	+13.0	+5.7	+4.6
Baseline+SJ+LPD‡	39.26	2.89	0.89	<u>19.39</u>	0.75	27.68	<u>2.19</u>	3.78	42.5	<u>9.35</u>	**21.55**	0.3	<u>51.89</u>	**33.48**	<u>47.38</u>	<u>23.11</u>	33.31	**23.22**	**6.78**	**20.51**
Increments to baseline	+5.5	+1.2	-2.4	+3.8	+0.5	+2.2	+0.5	+0.4	+27.2	+0.2	+4.8	+0.3	+18.5	+11.6	+7.9	+4.4	-10.7	+14.5	+5.9	+5.1

In the experiment for SynLiDAR→SemanticSTF, our methodology significantly improves performance on key classes in a driving environment, increasing the IoU for 'car' by +5.5, 'person' by +2.2, and 'road' by +27.2 as shown in Tab. A.3. Additionally, it shows substantial performance gains compared to the baseline, with increases of +4.8 IoU for 'sidewalk', +18.5 IoU for 'building', and +11.6 IoU for 'fence'. This indicates that our methodology effectively aids in predicting classes that become challenging to forecast due to adverse weather conditions.

E More Ablation Studies

Analysis of DSJ and ASJ. Our study compared DSJ and ASJ against jittering applied to all points. Additionally, to evaluate Cube-Selective Jittering (CSJ), which involves jittering points within a random cubic area. According to Tab. A.4, jittering all points improved robustness in +5.5 mIoU but resulted in a -2.2 mIoU performance decrease on clean data. In contrast, CSJ led to a decrease in mIoU across all adverse weather conditions in a -0.2 mIoU. This outcome suggests that not considering the prior spherical shape of LiDAR data when choosing jittering locations may be inadequate for countering adverse weather. Conversely, ASJ demonstrated improvements (+2.6 mIoU in dense fog, +5.3 mIoU in light fog, +7.9 mIoU in rain, and +6.3 mIoU in snow) with a relatively minor -1.8 mIoU performance decrease in clean data. Also, DSJ showed similar trends with increases in mIoU across various adverse weather conditions and a more moderate -1.7 mIoU drop in clean data performance. These results indicate that ASJ and DSJ effectively simulate coordinate distortion while minimizing the performance gap with clean data.

Table A.4: Experiments on ASJ/DSJ relative to all-point jittering/CSJ. All models were trained on SemanticKITTI and validated on SemanticSTF. The values in parentheses indicate the performance improvement or decrease over the baseline model.

Methods	Clean	D-fog	L-fog	Rain	Snow	mIoU
Baseline	**63.9**	30.7	30.1	29.7	25.3	31.4
Jittering	61.7 (-2.2)	<u>34.3</u>	<u>36.3</u>	33.2	**32.4**	<u>36.9</u> (+5.5)
CSJ	<u>63.2</u> (-0.7)	32.9	31.1	28.8	23.2	31.2 (-0.2)
ASJ	62.1 (-1.8)	33.3	35.4	**37.8**	<u>31.6</u>	36.8 (+5.4)
DSJ	62.2 (-1.7)	**34.6**	**36.8**	<u>36.1</u>	31.2	**38.0** (+6.6)

Analysis of LPD. In our research, we investigated how LPD differs from random dropout. As indicated in Tab. A.5, LPD increasingly drops points as the distance increases. Fog tends to obscure objects that are further away more significantly. Therefore, this demonstrates that LPD effectively represents point drop due to occlusions such as fog, which it is designed to simulate. Consequently, LPD is a more reasonable point dropout method to deal with adverse weather conditions than random point drop.

Table A.5: Ratio of points remaining after LPD compared to original data. This contrasts with random drop, which exhibits a uniform remaining ratio across different distances.

Distance	0∼10m	10∼20m	20∼30m	30∼40m	40∼50m	50∼60m	60∼70m	70∼80m	80∼90m
Ratio	80.6	78.6	78.7	77.4	76.2	75.5	76.2	76.4	72.5

Augmentation Configurations and Parameter Ablation. (1) DSJ adds noise to points along discretized depth levels. The number of depth levels, randomly selected within a specific range, is a hyperparameter. More depth levels improved rain resistance. (2) ASJ perturbs points along angles, with the maximum angular range $\Delta\theta$ as a hyperparameter. A larger angle range improved dense fog resistance. (3) Gaussian noise determines the jittering severity in SJ with its standard deviation σ as a hyperparameter. Higher σ improved resistance to rain and snow; lower σ to fog. (4) LPD involves a batch size, discount rate γ, and exploration decay ϵ as hyperparameters related to DQN learning. Higher γ improved fog resistance. Ablation studies in Tab. A.6 show that we achieve generalized performance, less sensitive to hyperparameter choices.

Comparison with Domain Generalization and Adaptation Methods. CosMix [6] and UniMix [10] offer unsupervised domain adaptation (DA) techniques, with UniMix also providing domain generalization (DG) techniques. Domain generalization methods train models using only source domain data, while

Table A.6: Hyperparameter ablation in SemanticKITTI-to-SemanticSTF. The values we chose are highlighted in **bold**.

Selective Jittering					
DSJ range	mIoU	ASJ $\Delta\theta$	mIoU	Gaussian σ	mIoU
$[2, 5]$	39.3	$\frac{1}{2}\pi$	38.1	0.001	37.5
$[3, 6]$	38.8	$\frac{3}{4}\pi$	39.9	0.005	38.4
$[4, 7]$	37.8	π	**39.5**	**0.01**	**39.5**
[5,8]	**39.5**	$\frac{5}{4}\pi$	40.0	0.05	37.3
$[6, 9]$	37.8	$\frac{3}{2}\pi$	39.3	0.1	37.9
deviation	0.8	deviation	0.2	deviation	1.3

Learable Point Drop					
Batch size	mIoU	Discount γ	mIoU	Decay ϵ	mIoU
8	37.1	0.5	37.8	100	38.5
16	37.1	0.8	38.4	500	38.7
32	**39.5**	0.9	38.4	**1000**	**39.5**
64	38.2	**0.99**	**39.5**	2000	37.3
128	37.0	0.999	38.8	3000	40.3
deviation	1.7	deviation	0.9	deviation	0.7

unsupervised domain adaptation methods train models with unlabeled target domain data. Our method, however, exclusively uses source data, making it fundamentally more challenging than UDA and not directly comparable. As shown in Tab. A.7, our method outperforms CosMix and UniMix (DG) by +7.9 and +4.9 mIoU, respectively. This result demonstrates that our proposed method effectively represents adverse weather conditions without real weather-conditioned data. Additionally, our method performs comparably to UniMix (DA) despite not using target domain data as DA does. Because UDA can complement data augmentations, combining our method with UniMix could further enhance performance.

Table A.7: Comparison with CosMix and UniMix. DA and DG denote domain adaptation and domain generalization.

Methods	mIoU
CoSMix(DA)	28.4
UniMix(DA)	37.2
UniMix(DG)	31.4
MinkowskiNet18+Ours	36.3

Time cost of augmentation process. First, we emphasize that the inference runtime does not change. Regarding the training overhead, it takes 3.4 hours to train the original MinkowskiNet 15 epochs with four A6000 GPUs, and our augmentations take an extra 2.28 hours (+ 67%). In comparison, existing weather simulations [2,3] took over 12 hours for SemanticKITTI's train set, highlighting our method's efficiency relative to traditional simulations.

F More Qualitative Results

The figures below show the qualitative examples of LiDAR semantic segmentation results on the SemanticSTF validation set.

As observed in Fig. A.3, proposed SJ and LPD were found to improve the prediction of roads and sidewalks in dense fog conditions. Similarly, in light fog conditions, as demonstrated in the third and fourth rows of Fig. A.3, there was a significant enhancement in the performance for road predictions. Moreover, as shown in the black circle in the third row of Fig. A.3, accurate predictions were also achieved for road-related classes such as cars. This trend of improved performance under adverse weather conditions continued, as can be seen in Fig. A.4.

This observed trend of improved performance persists even when training on SynLiDAR and performing validation on SemanticSTF. As seen in Fig. A.5, our method successfully executes accurate predictions in locations where the baseline model previously erred. Similarly, in the third row of Fig. A.5, enhanced precision in predicting objects like cars is evident. This consistent trend of enhanced accuracy under various weather conditions can also be found in Fig. A.6.

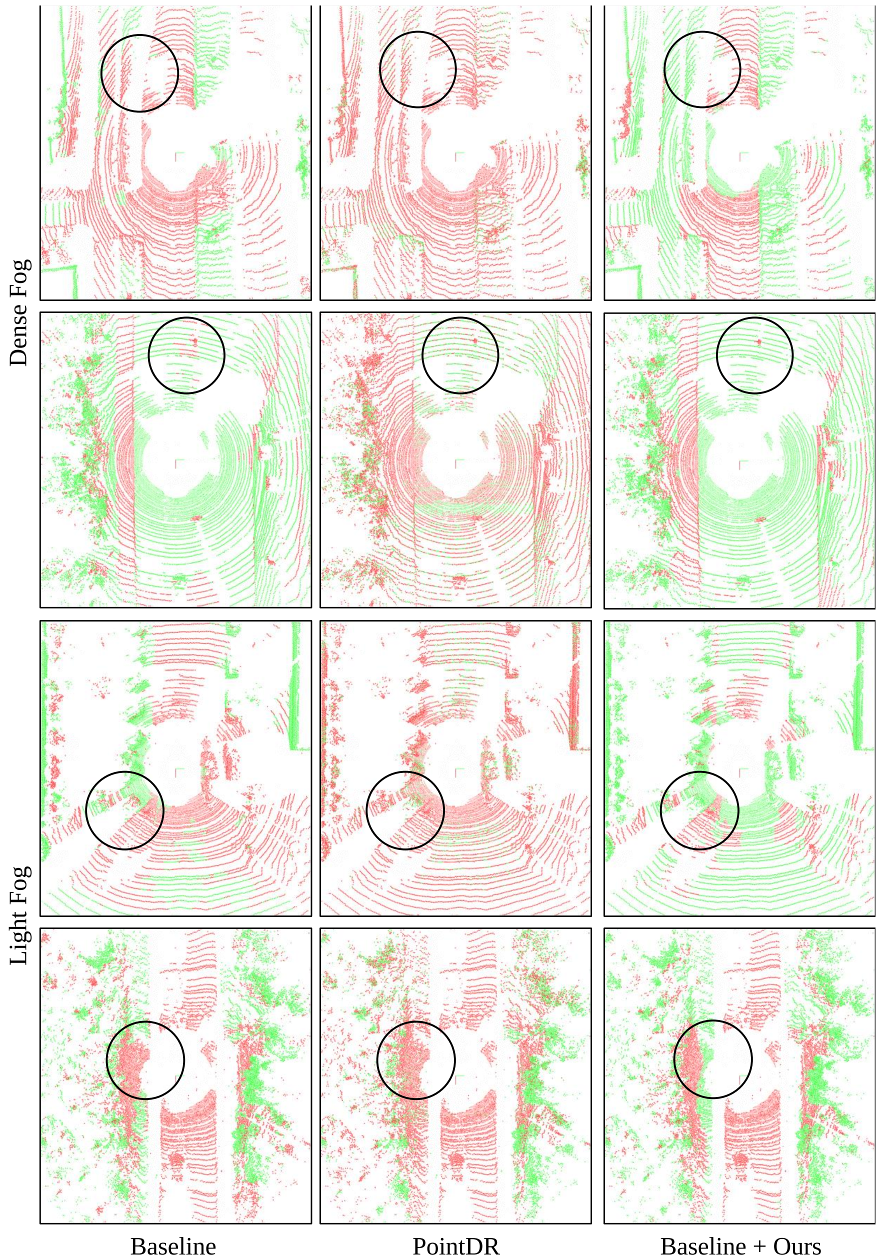

Fig. A.3: Qualitative examples of LiDAR semantic segmentation result on the ***dense fog*** and ***light fog*** condition of the SemanticSTF *validation set*. All models are trained on SemanticKITTI. In all qualitative results, the red and green points indicate incorrect and correct predictions respectively. Best viewed in colors.

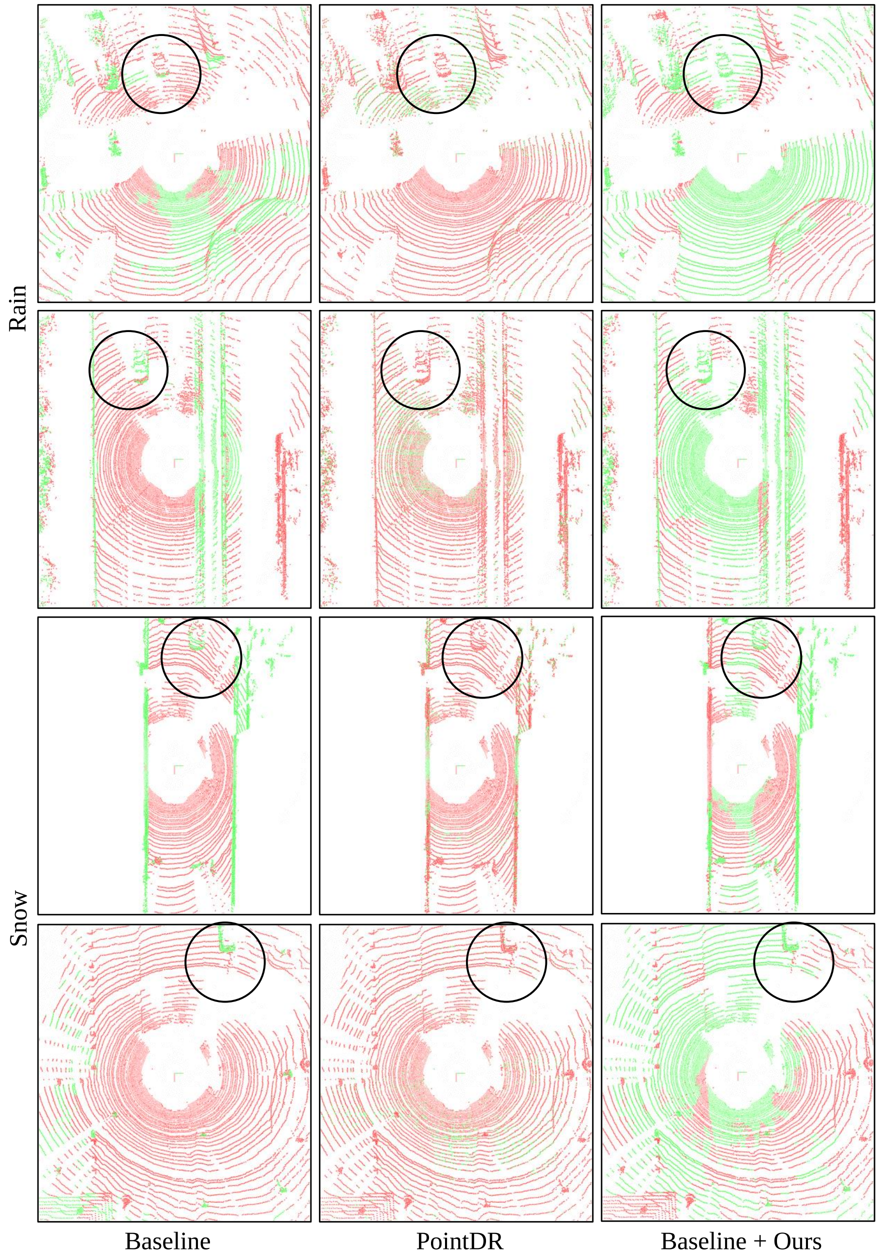

Fig. A.4: Qualitative examples of LiDAR semantic segmentation result on the ***rain*** and ***snow*** condition of the SemanticSTF *validation set*. All models are trained on SemanticKITTI. In all qualitative results, the red and green points indicate incorrect and correct predictions respectively. Best viewed in colors.

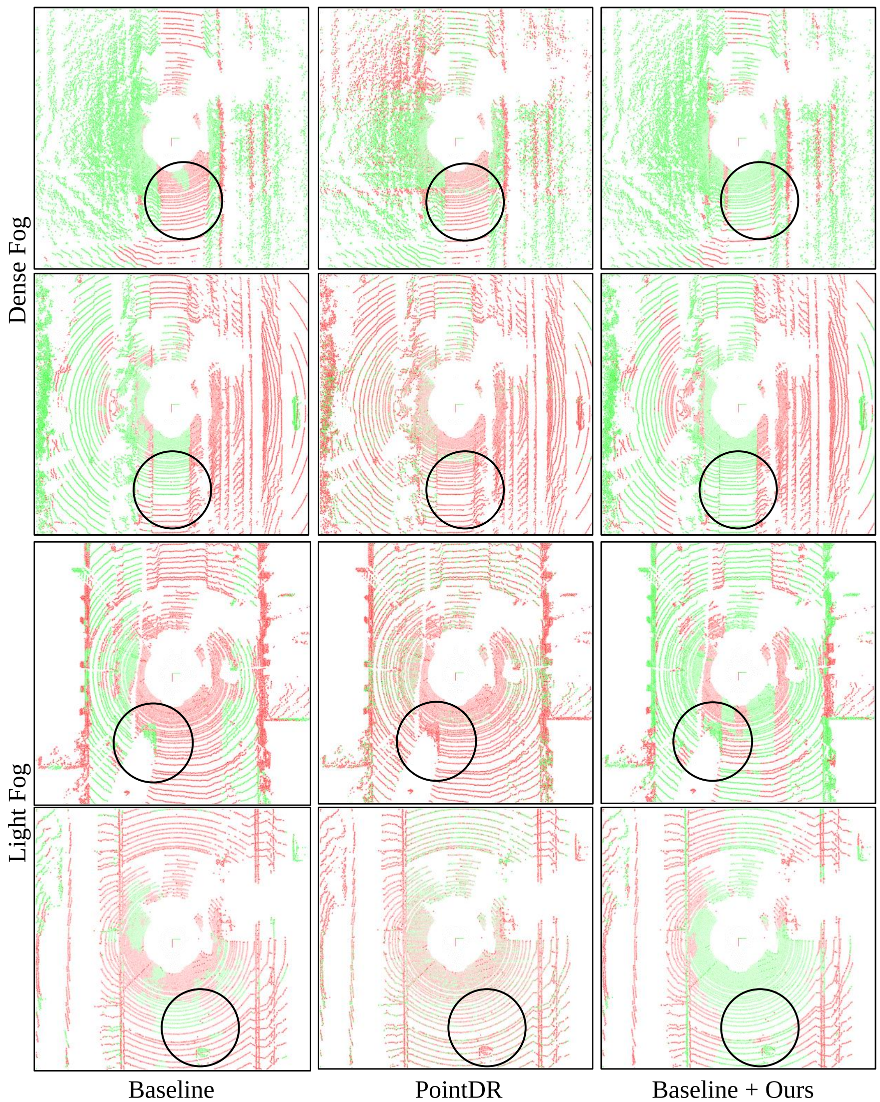

Fig. A.5: Qualitative examples of LiDAR semantic segmentation result on the ***dense fog*** and ***light fog*** condition of the SemanticSTF *validation set*. All models are trained on SynLiDAR. In all qualitative results, the red and green points indicate incorrect and correct predictions respectively. Best viewed in colors.

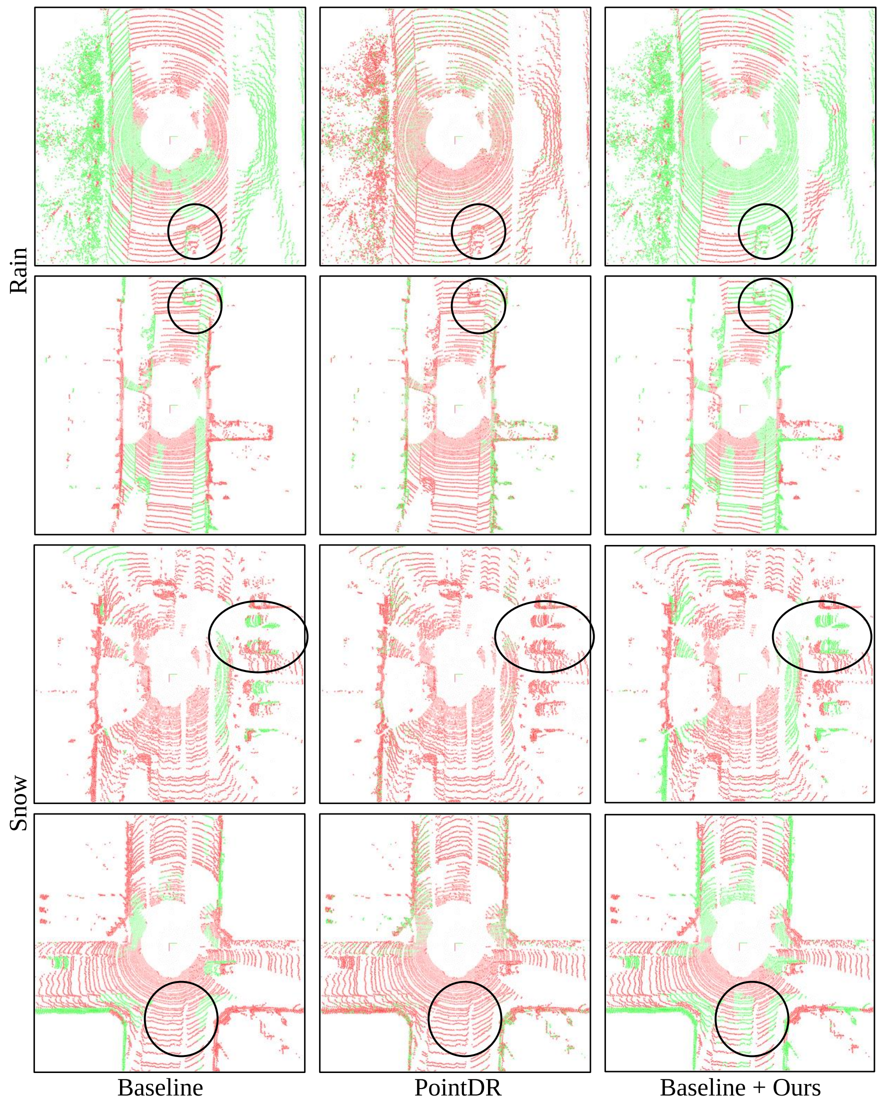

Fig. A.6: Qualitative examples of LiDAR semantic segmentation result on the ***rain*** and ***snow*** condition of the SemanticSTF *validation set.*. All models are trained on SynLiDAR. In all qualitative results, the red and green points indicate incorrect and correct predictions respectively. Best viewed in colors.

References

1. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: Semantickitti: A dataset for semantic scene understanding of lidar sequences. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9297–9307 (2019)
2. Hahner, M., Sakaridis, C., Bijelic, M., Heide, F., Yu, F., Dai, D., Van Gool, L.: Lidar snowfall simulation for robust 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16364–16374 (2022)
3. Kong, L., Liu, Y., Li, X., Chen, R., Zhang, W., Ren, J., Pan, L., Chen, K., Liu, Z.: Robo3d: Towards robust and reliable 3d perception against corruptions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19994–20006 (2023)
4. Kong, L., Ren, J., Pan, L., Liu, Z.: Lasermix for semi-supervised lidar semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21705–21715 (2023)
5. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602 (2013)
6. Saltori, C., Galasso, F., Fiameni, G., Sebe, N., Ricci, E., Poiesi, F.: Cosmix: Compositional semantic mix for domain adaptation in 3d lidar segmentation. In: European Conference on Computer Vision. pp. 586–602. Springer (2022)
7. Xiao, A., Huang, J., Guan, D., Cui, K., Lu, S., Shao, L.: Polarmix: A general data augmentation technique for lidar point clouds. Advances in Neural Information Processing Systems **35**, 11035–11048 (2022)
8. Xiao, A., Huang, J., Guan, D., Zhan, F., Lu, S.: Transfer learning from synthetic to real lidar point cloud for semantic segmentation. In: Proceedings of the AAAI conference on artificial intelligence. vol. 36, pp. 2795–2803 (2022)
9. Xiao, A., Huang, J., Xuan, W., Ren, R., Liu, K., Guan, D., El Saddik, A., Lu, S., Xing, E.P.: 3d semantic segmentation in the wild: Learning generalized models for adverse-condition point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9382–9392 (2023)
10. Zhao, H., Zhang, J., Chen, Z., Zhao, S., Tao, D.: Unimix: Towards domain adaptive and generalizable lidar semantic segmentation in adverse weather. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14781–14791 (2024)