Learning to Identify Out-of-Distribution Objects for 3D LiDAR Anomaly Segmentation

Supplementary Material

A. Further Details on OoD Datasets

In this section, we provide additional details about the proposed mixed real-synthetic OoD datasets, including the construction process, the selected ModelNet objects inserted into the scans, the insertion protocol, and the technique used to align point distribution with the LiDAR sensor geometry.

A.1. Overview

As described in Sec. 4, we construct the proposed Out-of-Distribution (OoD) datasets based on three established autonomous driving benchmarks [4, 7, 45]. To insert anomaly objects, we use 3D models from the ModelNet dataset [67], carefully selecting those that do not conflict with the training or evaluation data and are not present in the original training sets, ensuring a totally diverse domain. The selected objects are listed in Tab. 9, while others, such as cars, persons, or objects unsuitable for the driving context, like airplanes and guitars, are excluded. Some objects that are not typically suited for the driving environment, such as bookshelves, dressers, range hoods, or wardrobes, are still included and appropriately resized when inserted to simulate debris, obstacles and other unexpected impediments on the road. This way, we can simulate anomalies that may occur in real-world scenarios, where the vehicle must identify and avoid them. Examples of selected models used for the OoD datasets creation are shown in Fig. 4.

Figure 4. Examples of ModelNet objects selected for the creation of the proposed mixed real-synthetic OoD datasets.

For each object, we uniformly sample 3D points across the surface of its CAD model to obtain a dense point representation. This facilitates the subsequent alignment of the object's point distribution to the LiDAR scan geometry in the insertion strategy, as described in Sec. 4. Since ModelNet models do not have intensity information, we assign a temporary value of 0 to each point to obtain the same feature configuration of LiDAR scans. We also apply a simple scaling augmentation to randomly reduce the size of the objects, increasing variability.

For each base dataset, we select different surfaces for insertion based on the *single* and *multi* splits introduced in Sec. 4. Specifically, for SemanticKITTI-OoD, anomaly objects in the *single* split are inserted only on the road class label, while the *multi* split also considers classes as parking, sidewalk, and other-ground. For SemanticPOSS-OoD, due to the limited number of class labels, ground is used as the insertion surface in both splits. In nuScenes-OoD, the *single* split contains anomaly objects only on the drivable surface class, whereas the *multi* split also considers other-flat and sidewalk classes.

For the *multi* split, we select the number of inserted objects with decreasing probability of 40%, 30%, 20%, and 10% for inserting 1, 2, 3, and 4 objects, respectively. All anomaly objects are placed within a 50 m radius from the center of the scan, where the LiDAR sensor is located (see Fig. 5), to be consistent with the evaluation setup of [43].

Regarding the class labels, we follow [43] and use the value 2 to denote anomaly points in both SemanticKITTI-OoD and SemanticPOSS-OoD. In nuScenes-OoD, label 2 is already assigned to the *human.pedestrian.adult* class, so we use label 100 for anomaly points to avoid any conflict.

A.2. Reflectivity Values

As introduced in Sec. 4, ModelNet objects do not provide intensity information, while real-world LiDAR scans usually have this attribute. To solve the mismatch and ensure consistency between inserted objects and the scene, we compute a per-point intensity value (see Eq. (15)) that approximates real-world behavior following the Lambertian reflectance model [44]. This requires assigning a reflectivity value ρ to each object, representing the intrinsic reflectance of its material.

We assign material types to the selected ModelNet objects based on plausible real-world composition, for example, treating chairs as wood or plastic and vases as ceramic.

Table 9. Details of the anomaly objects inserted in the proposed OoD datasets. For each object, we report the number of occurrences in each dataset, along with the total number of anomalies and the per-scan ratio.

Object	SemanticKITTI-OoD		SemanticPOSS-OoD		nuScenes-OoD	
	S	M	S	M	S	M
bathtub	59	174	5	17	105	304
bed	40	180	5	20	72	239
bookshelf	56	158	4	21	77	232
bottle	49	153	5	24	85	273
bowl	66	182	7	15	84	260
chair	64	175	9	16	86	251
cup	50	168	10	29	77	253
desk	60	153	9	22	87	276
dresser	60	177	6	16	76	253
flower pot	60	161	10	27	73	255
glass box	59	174	2	17	82	232
lamp	55	165	7	18	94	262
laptop	57	171	2	19	75	254
mantel	43	169	9	22	90	223
monitor	60	173	7	24	84	279
night stand	52	175	12	13	87	247
piano	60	177	7	21	74	265
radio	49	145	6	14	77	263
range hood	57	175	10	22	80	235
sink	53	189	6	20	77	212
sofa	67	167	6	20	94	275
stool	48	168	11	20	98	258
table	71	146	9	21	77	249
tent	68	164	2	25	88	241
toilet	62	169	8	28	75	232
tv stand	51	172	5	26	82	242
vase	45	165	7	15	85	237
wardrobe	61	172	8	19	78	238
xbox	53	177	2	15	79	228
Total	1634	4894	196	586	2398	7268
Ratio	0.40	1.20	0.39	1.17	0.40	1.21

We assign reflectivity values accordingly, based on empirical observations and conventions used in rendering engines (e.g., Blender). Materials such as glass, ice or dark surfaces show low reflectance, while mirrors or glossy metals reflect significantly more light, resulting in higher reflectivity values. These values range in $[0, 1]$ where 0 and 1 indicate respectively the minimum and maximum reflectance. Table 10 reports assigned materials and corresponding reflectivity values for all selected models. Figure 6 shows some examples of the computed intensity values obtained using the proposed technique, visualized on sections of the range image projections. In SemanticKITTI-OoD, the effect of the intensity computation on the inserted object is more evident, while in the other two datasets, the low mean intensity value makes anomalies more difficult to distinguish from the background, particularly for SemanticPOSS-OoD. Overall, results demonstrate that the proposed approach for intensity estimation adapts to the characteristics of corresponding LiDAR scans, producing values that correctly align with the sensor behavior.

Table 10. Material properties and reflectivity values for the selected ModelNet objects.

Object	Material	Reflectivity
bathtub	glossy ceramic	0.60
bed	fabric, wood	0.40
bookshelf	wood	0.40
bottle	glass, plastic	0.25
bowl	ceramic	0.60
chair	wood, plastic	0.35
cup	ceramic	0.60
desk	wood	0.40
dresser	wood	0.40
flower pot	clay	0.45
glass box	glass	0.20
lamp	metal, plastic	0.35
laptop	metal, plastic	0.30
mantel	stone	0.40
monitor	plastic, glass	0.25
night stand	wood	0.40
piano	polished wood	0.50
radio	plastic, metal	0.35
range hood	steel	0.45
sink	ceramic, metal	0.60
sofa	fabric	0.40
stool	wood, metal	0.35
table	wood, plastic	0.40
tent	fabric	0.30
toilet	glossy ceramic	0.60
tv stand	wood, plastic	0.40
vase	ceramic	0.55
wardrobe	wood	0.40
xbox	plastic	0.30

A.3. Insights on Protocol

We provide additional details on the proposed strategy for creating the OoD datasets, starting from each base autonomous driving benchmark. Given the combined point cloud $\mathbf{S} = [\mathbf{P}, \mathbf{O}]$, where $\mathbf{P} \in \mathbb{R}^{N \times 4}$ is the LiDAR scan containing N points and $\mathbf{O} \in \mathbb{R}^{M \times 4}$ the object point cloud with M points, we perform a spherical projection of 3D points onto a 2D surface, namely, a range image [37]. This projection provides essential geometric information for preserving realistic scan properties, such as point occlusion and the beam-like point distribution characteristic of LiDAR measurements. Each point $\mathbf{p}_n \in \mathbf{S}$ where $\mathbf{p}_n = (x_n, y_n, z_n, i_n)$ with coordinates and intensity value, is projected onto a range image $R(u, v) \in \mathbb{R}^{H \times W}$, where H and W are the height and width. We set H as the number of beams of the LiDAR sensor, for each dataset, and $W = 2048$ for a wider horizontal resolution, to keep more

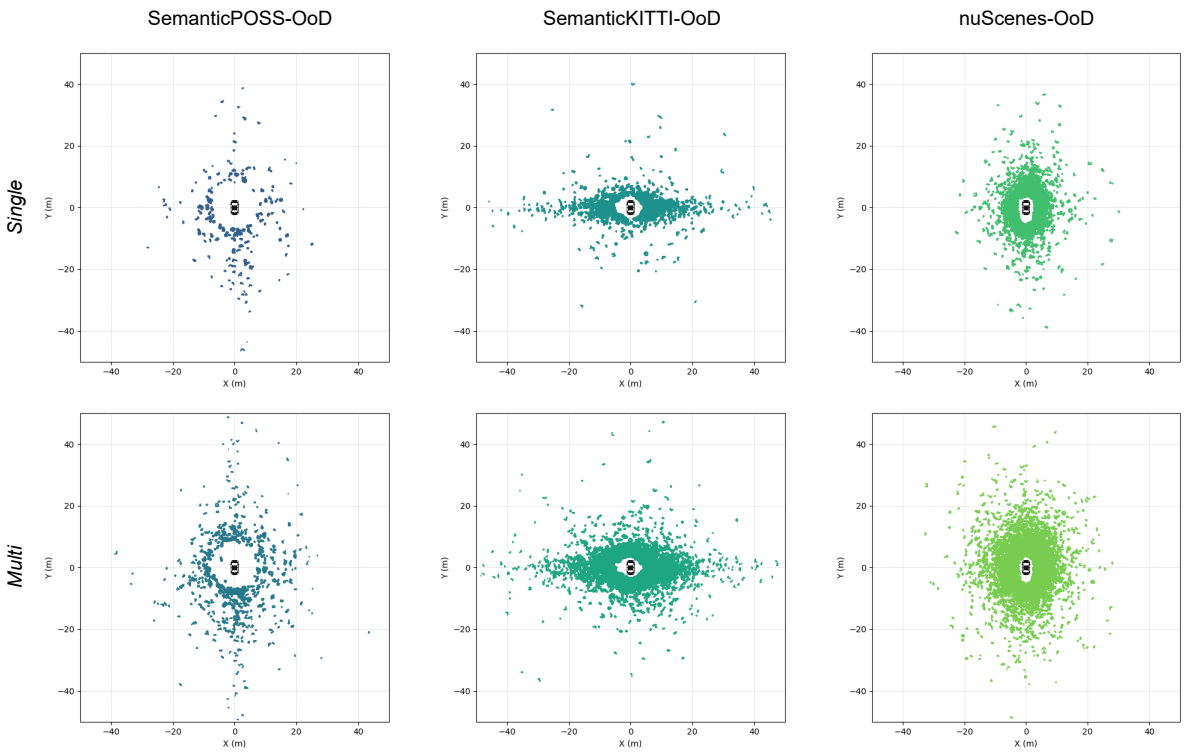


Figure 5. Distribution of anomaly points on the XY plane across all proposed OoD datasets.

anomaly points. The projection procedure is as follows:

$$\begin{pmatrix} u_n \\ v_n \end{pmatrix} = \begin{pmatrix} \frac{1}{2}[1 - \arctan(y_n, x_n)\pi^{-1}]W \\ [1 - (\arcsin(z_n, r_n^{-1}) + f_{\text{down}})f^{-1}]H \end{pmatrix}, \quad (16)$$

where $r_n = \sqrt{x_n^2 + y_n^2 + z_n^2}$ is the range of the point with respect to the sensor, $f = |f_{\text{up}} + f_{\text{down}}|$ denotes the vertical field of view of the sensor, and f_{up}, f_{down} are the upward and downward inclination angles, respectively. In the case of multiple points projected onto the same cell, we select the closest point to the sensor, i.e., the one having the minimum range value.

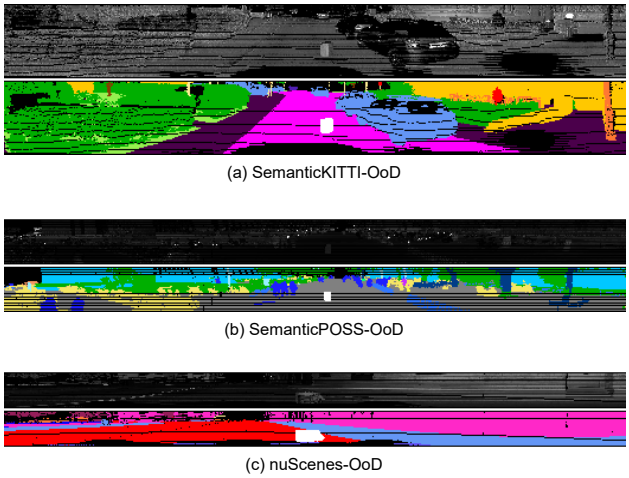
Each cell of the range image stores the range value of the 3D point projected in the corresponding cell, providing essential information for dataset construction and for geometrically aligning the point distribution of the inserted objects with that of the LiDAR scans. First, it provides a direct mapping between 3D points and 2D cells, allowing us to identify points in the original LiDAR scan $\mathbf{S}$ that become occluded by the inserted anomaly object, and should be removed in the final scan. Second, for points belonging to the inserted objects, rows of the range image correspond to LiDAR beams. We select points along each row, obtaining a beam-like sampling pattern for the object, aligned with the LiDAR data acquisition process. Furthermore, since each cell retains only the closest point to the sensor, this process ensures that only visible, front points of the object are preserved. The projection and reprojection operations also avoid any possible overlap of the inserted objects with instances in the LiDAR scan, maintaining a geometrically consistent real-world scenario.

B. Further Details on Experiments

B.1. Baselines

Due to the novelty of the task, as discussed in Sec. 5, there are only a few methods developed for 3D LiDAR anomaly segmentation. Moreover, only a subset of these approaches provides publicly available code [43]. Therefore, for the experiments on the proposed OoD datasets, we rely on the implementation from [43], which, however, includes only a portion of the methods tested in that work, namely max logit, RbA, and deep ensemble. For a fair comparison, we additionally evaluate standard OoD methods considered in [43], using the same backbone as our approach, i.e., MinkowskiNet [14], to assess any possible bias related to the backbone. We retrain these methods on the corresponding training split of each OoD dataset and evaluate them using the same inference setting.

For Void Classifier, we train the network with an additional class corresponding to the unlabeled/outlier regions and use its predicted confidence at inference time to identify



(a) SemanticKITTI-OoD

(b) SemanticPOSS-OoD

(c) nuScenes-OoD

Figure 6. Example of computed intensity values in the proposed OoD datasets. (Top) Remission value of the LiDAR scan. (Bottom) Semantic labels, with inserted anomaly objects shown in white.

Table 11. Ablation study on the threshold r in the contrastive head score Eq. (13). Results on STU validation set.

Threshold r	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑
1	93.17	36.47	18.87
2	94.51	34.66	26.92
3	94.79	**34.43**	27.04
4	94.90	34.65	27.46
5	**95.05**	34.86	**27.53**

Table 12. Ablation study of the LiDAR anomaly segmentation approach pipeline on SemanticKITTI-OoD (*Multi*) set, analyzing its impact on semantic segmentation.

$\mathcal{L}_{\text{prot}}$	$\mathcal{L}_{\text{cont}}$	$\mathcal{L}_{\text{obj}}$	mIoU (%)
			64.8
✓			60.7
✓	✓		61.0
✓	✓	✓	61.2

anomalies. For MC Dropout, we introduce dropout layers in the backbone during training and activate them in inference, where multiple forward pass are repeated (10 in our setup). For the deep ensemble, following [43], we train three models with different random seeds to obtain different checkpoints. We then extract the pre-softmax per-point features from each model and combine them by computing the mean and then calculating the entropy of the prediction probabilities to produce the final anomaly segmentation scores.

Table 13. Model runtime comparison across different datasets. Results are in reported in ms. Tested on a NVIDIA A40 GPU.

Method	KITTI-OoD	POSS-OoD	nuScenes-OoD	STU
Mask4Former3D [72]	348	315	168	392
Deep Ensemble (Sequential) [30]	2454	1695	861	2628
Deep Ensemble (Parallel) [30]	818	565	287	876
LIDO (ours)	**87**	**75**	**38**	**90**

Table 14. Range-based evaluation on STU validation set (AP, %).

Method	0-10m	10-20m	20-30m	30-40m	40-50m
RbA [41]	1.85	1.28	0.73	0.15	0.01
Max Logit [24]	2.25	1.53	1.20	0.27	0.01
Deep Ensemble [30]	7.63	8.49	3.42	**0.38**	**0.03**
MC Dropout [56]	0.16	0.53	0.06	0.04	0.01
Void Classifier [6]	2.95	1.78	1.98	0.28	**0.03**
LIDO (ours)	**40.77**	**44.07**	**5.51**	0.21	0.02

B.2. Additional Ablation Study

In Tab. 11, we report results obtained with different threshold values r, as defined in Eq. (13). Due to the large domain gap between training and evaluation, when training solely on SemanticKITTI and testing on the STU dataset, this parameter requires tuning. Differences in sensor resolution and feature distributions may produce variations in the feature norms. Unlike findings in the image domain [55], we find that feature norms of inlier classes, produced from LiDAR scans, tends to be larger. This explains the choice of a higher threshold, as reported in the table, with $r = 5$ obtaining the best performance. Other values have slightly lower FPR, but do not match the best AUROC and AP metrics.

We also analyze the impact of the loss functions introduced in our approach on semantic segmentation performance (mIoU). Results are reported in Tab. 12 on the proposed SemanticKITTI-OoD dataset, *multi* split, as STU [43] validation set does not provide semantic labels. We observe that introducing the prototype loss $\mathcal{L}_{\text{prot}}$ leads to a decrease in mIoU, while the addition of contrastive and objectosphere losses slightly recovers this drop, by improving the separation between class-specific features. Although this results in a slight reduction in semantic segmentation results, it yields improved anomaly segmentation results, consistent with the trends observed in the ablation study on STU (see Sec. 5).

B.3. Runtime

We report a runtime comparison between our proposed approach and other baselines [43] in Tab. 13, on the STU dataset and the proposed real-synthetic OoD datasets. All methods are tested on a single NVIDIA A40 GPU. The results show that our approach maintains real-time performance (<100 ms) across all datasets, including high-resolution configurations such as the 128-beam STU scans. Instead, Mask4Former3D [43] combined with post-

processing techniques as max logit or RbA, and especially when used in an ensemble setting, drastically increases the runtime, requiring seconds to produce a single prediction.

B.4. Additional Results

We compute the AP metric across different distance thresholds and report results in Tab. 14, following the protocol of [43]. We compare our approach with the Mask4Former3D baselines introduced in [43]. As expected, our method achieves superior performance at shorter ranges, with a decrease as the distance to anomalous objects increases, a trend observed across all approaches. These results highlight the effectiveness of our approach in identifying anomalies while still indicating room for improvements for distant objects, which remain a challenging setting.

We report extensive results on both STU and the proposed mixed real-synthetic OoD m, datasets in Tabs. 15 to 18, including additional baselines evaluated with the same backbone as our method, i.e., MinkowskiNet [14], for a fair and more comprehensive comparison. Table 15 presents results on the STU validation set, including object-level OoD metrics from [43] for a better evaluation of anomaly segmentation performance, such as Panoptic Quality (PQ), Unknown Quality (UQ) [61], Recognition Quality (RQ) and Segmentation Quality (SQ). We refer the reader to the original paper for more details on these metrics. Our method significantly outperforms all baselines in terms of AP, indicating strong capability in identifying anomalous objects, and also achieves overall better object-level performance. In contrast, applying standard OoD technique on top of the same MinkowskiNet backbone yield comparable or even inferior results compared to the Mask4Former3D baseline from [43]. Results on the other datasets further confirm the effectiveness of the proposed approach, consistently achieving strong performance. While standard OoD methods based on either Mask4Former3D or MinkowskiNet remain competitive, they generally do not surpass our method. When they do so, as in the case of Deep Ensemble, they incur in significantly higher computational and memory costs. nuScenes-OoD (Tab. 18) represents a particular challenging scenario, mainly due to the lower number of LiDAR beams and reduced point density. As discussed in Sec. 5, these factors affect the learning of good and robust class prototypes, which also reflects in the observed drop in semantic segmentation performance. In this dataset, our method achieves comparable results with standard OoD methods with the same backbone in terms of AP, while maintaining lower FPR values, highlighting the effectiveness despite the challenging setting and indicating room for further improvements in generalization.

B.5. LiDAR Semantic Segmentation

Tables 19 to 22 present detailed per-class LiDAR semantic segmentation results on STU inlier validation sequence and the proposed OoD datasets, comparing a standard semantic segmentation baseline with our approach that incorporates the losses described in Sec. 3. Our method achieves comparable performance with only a small degradation on nuScenes-OoD. This drop is justified by the lower resolution and reduced number of points per scan in nuScenes, which may affect the construction of robust prototypes. The effect is further amplified by the severe class imbalance, where bicycle and motorcycle classes are present for only 0.01% and 0.03% of the data (approximately 10^5 and 3×10^5 points, respectively, out of the total), affecting contrastive learning, prototype building and the learned feature space. Consequently, these classes are often misclassified as manmade, which is significantly more present (15%).

C. Qualitative Results

We report further qualitative results for each dataset, comparing our method with the deep ensemble model in [43], built upon Mask4Former [72]. Figures 7 to 10 show visualization of anomaly segmentation results on STU validation set, SemanticPOSS-OoD, SemanticKITTI-OoD and nuScenes-OoD datasets, respectively. We report both successful and failure cases. Our proposed approach demonstrates strong anomaly segmentation performance across the datasets, better or comparable to that of the deep ensemble model. For each figure, ground truth anomaly objects are shown in cyan, predicted anomaly scores follow the plasma color map from blue to orange, indicating increasing probability of a point being anomalous. Compared to the ensemble model used in [43], our approach produces fewer false positives. In the ensemble results, many inlier points that belong to the road and building classes are incorrectly marked in violet or red, meaning that the model assigns them a high probability of being an anomaly. In contrast, our approach demonstrates more robust predictions on inlier classes as most points are colored in blue, indicating low anomaly scores, with only limited uncertainty near class boundaries (e.g., road and sidewalk), within underrepresented classes or at long ranges, well-known challenges in standard semantic segmentation [28].

Table 15. Anomaly Segmentation performance on STU validation set.

Method	Point-Level OoD			Object-Level OoD				
	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑	RecallQ [%]	SQ [%]	RQ [%]	UQ [%]	PQ [%]
Mask4Former3D + MC Dropout [56]	65.76	79.82	0.17	3.54	74.36	3.48	2.63	2.59
Mask4Former3D + RbA [41]	73.00	100.0	1.64	21.84	78.58	2.75	17.16	2.16
Mask4Former3D + Max Logit [24]	87.27	68.76	2.02	26.64	79.26	2.06	21.12	1.63
Mask4Former3D + Void Classifier [6]	89.77	79.50	2.62	17.35	81.27	8.98	14.10	7.30
Mask4Former3D + Deep Ensemble [30]	90.93	37.34	6.94	17.70	79.96	9.10	14.15	7.27
MinkowskiNet + RbA [41]	53.93	92.07	0.05	4.76	**88.49**	0.26	4.21	0.23
MinkowskiNet + Void Classifier [6]	70.24	100.0	0.11	-	-	-	-	-
MinkowskiNet + Max Logit [24]	90.92	43.49	0.97	-	-	-	-	-
MinkowskiNet + MC Dropout [56]	91.98	35.64	1.41	35.25	75.83	1.35	26.73	1.02
MinkowskiNet + Deep Ensemble [30]	93.32	**29.67**	2.72	**39.88**	78.40	2.01	**31.27**	1.57
LIDO (ours)	**95.05**	34.86	**27.53**	31.92	74.82	**28.38**	23.88	**21.23**

Table 16. Anomaly Segmentation performance on SemanticPOSS-OoD dataset.

Method	Single			Multi		
	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑
Mask4Former3D + RbA [41]	49.09	100.0	0.13	50.42	100.0	0.26
Mask4Former3D + Max Logit [24]	61.05	91.85	0.19	63.77	88.74	0.40
Mask4Former3D + Deep Ensemble [30]	85.86	58.12	0.82	85.83	54.20	1.21
MinkowskiNet + RbA [41]	22.21	99.32	0.08	19.97	99.43	0.14
MinkowskiNet + Void Classifier [6]	46.22	98.38	0.12	49.02	97.30	0.24
MinkowskiNet + Max Logit [24]	87.15	48.21	1.36	89.86	47.72	2.94
MinkowskiNet + MC Dropout [56]	85.49	66.30	1.41	86.11	58.76	2.93
MinkowskiNet + Deep Ensemble [30]	89.82	45.91	2.54	**91.15**	46.33	4.87
LIDO (ours)	**91.51**	**45.10**	**3.97**	90.84	**44.74**	**5.92**

Table 17. Anomaly Segmentation performance on SemanticKITTI-OoD dataset.

Method	Single			Multi		
	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑
Mask4Former3D + RbA [41]	59.68	100.0	4.56	71.39	100.0	13.63
Mask4Former3D + Max Logit [24]	76.49	99.63	4.84	84.05	99.32	**13.84**
Mask4Former3D + Deep Ensemble [30]	92.87	**27.69**	6.20	**92.19**	**28.64**	12.04
MinkowskiNet + RbA [41]	40.38	95.45	0.19	36.90	96.54	0.41
MinkowskiNet + Void Classifier [6]	47.74	87.02	0.22	50.89	84.33	0.53
MinkowskiNet + Max Logit [24]	88.70	52.26	2.49	86.93	48.01	3.88
MinkowskiNet + MC Dropout [56]	87.49	50.75	3.04	84.93	57.46	4.48
MinkowskiNet + Deep Ensemble [30]	91.64	39.96	4.78	89.03	42.34	6.21
LIDO (ours)	**93.36**	31.19	**10.60**	89.89	39.04	9.42

Table 18. Anomaly Segmentation performance on nuScenes-OoD dataset.

Method	Single			Multi		
	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑	AUROC [%] ↑	FPR@95 [%] ↓	AP [%] ↑
Mask4Former3D + RbA [41]	32.65	100.0	3.56	71.60	98.68	7.95
Mask4Former3D + Max Logit [24]	66.17	98.60	3.54	43.54	100.0	8.98
Mask4Former3D + Deep Ensemble [30]	**91.79**	**34.59**	**18.34**	88.62	43.63	**23.87**
MinkowskiNet + RbA [41]	11.02	99.59	0.41	12.61	99.52	0.80
MinkowskiNet + MC Dropout [56]	84.14	50.80	4.48	81.94	53.09	6.64
MinkowskiNet + Max Logit [24]	90.82	41.11	8.14	**90.10**	**38.73**	12.11
MinkowskiNet + Deep Ensemble [30]	89.79	39.28	8.26	87.29	44.37	11.30
MinkowskiNet + Void Classifier [6]	86.09	61.40	8.30	86.64	61.57	16.34
LIDO (ours)	89.33	39.70	6.79	87.25	44.51	10.53

Table 19. LiDAR Semantic Segmentation results on STU inlier validation sequence.

Method	mIoU %	car	bicycle	truck	person	road	parking	sidewalk	building	fence	vegetation	trunk	terrain	pole	traffic-sign
Standard	**36.7**	62.4	0.0	2.4	49.8	67.3	16.0	57.0	**84.1**	**75.2**	**90.9**	**34.7**	67.1	**45.7**	**45.8**
LIDO (ours)	35.1	**70.3**	**0.8**	**53.1**	**67.6**	**73.2**	**19.4**	**59.7**	73.7	70.1	66.8	21.4	**67.7**	12.1	11.9

Table 20. LiDAR Semantic Segmentation results on the proposed SemanticKITTI-OoD dataset.

	Method	mIoU %	car	bicycle	motorcycle	truck	other-vehicle	person	bicyclist	motorcyclist	road	parking	sidewalk	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic-sign
Single	Standard	**64.9**	**96.8**	**32.6**	**80.8**	**78.6**	**61.6**	**71.6**	**89.8**	**0.1**	**93.5**	**52.0**	**81.0**	0.2	**91.0**	**60.2**	**87.9**	**67.0**	**75.7**	**64.9**	49.7
	LIDO (ours)	61.3	95.9	18.7	74.2	74.5	61.0	71.1	82.8	0.0	91.4	42.4	77.0	**0.8**	89.5	52.6	86.4	66.3	74.0	55.7	**51.0**
Multi	Standard	**64.8**	**96.9**	**32.7**	**80.9**	**77.8**	**60.8**	**71.4**	**89.7**	**0.1**	**93.4**	**51.5**	**80.8**	0.2	**90.9**	**59.9**	**87.9**	**67.0**	**75.7**	**64.8**	49.7
	LIDO (ours)	61.2	95.9	18.7	73.3	74.4	**60.8**	70.8	82.7	0.0	91.2	42.2	76.6	**0.8**	89.5	52.4	86.4	66.3	74.0	55.7	**50.9**

Table 21. LiDAR Semantic Segmentation results on the proposed SemanticPOSS-OoD dataset.

	Method	mIoU %	person	rider	car	truck	plants	traffic sign	pole	trashcan	building	cone/stone	fence	bike	ground
Single	Standard	**57.1**	78.0	**34.6**	80.7	**33.5**	**74.7**	24.4	**41.1**	**65.4**	79.5	40.9	**51.0**	**57.9**	**80.2**
	LIDO (ours)	55.6	**79.1**	32.2	**82.9**	21.6	67.9	**27.1**	34.4	58.9	**81.5**	**57.6**	49.0	53.6	77.3
Multi	Standard	**57.0**	78.0	**34.5**	80.7	**33.5**	**74.7**	24.4	**41.1**	**65.3**	79.5	40.9	**51.0**	**57.9**	**80.2**
	LIDO (ours)	55.5	**79.1**	32.2	**82.9**	21.6	67.9	**27.1**	34.4	58.4	**81.5**	**57.5**	49.0	53.6	77.2

Table 22. LiDAR Semantic Segmentation results on the proposed nuScenes-OoD dataset.

	Method	mIoU %	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	driv. surf.	other flat	sidewalk	terrain	manmade	vegetation
Single	Standard	**72.7**	**74.3**	**40.2**	**87.9**	**90.5**	**40.9**	**81.8**	**76.8**	**60.0**	**54.2**	**81.1**	**95.3**	66.1	**70.2**	**74.1**	**85.9**	**84.7**
	LIDO (ours)	60.6	57.7	5.1	70.8	88.0	16.2	51.1	76.0	26.4	36.5	77.6	94.4	**66.4**	67.6	72.1	81.7	82.0
Multi	Standard	**72.6**	**90.5**	**40.1**	**87.9**	**90.5**	**40.7**	**81.7**	**76.8**	**59.2**	**54.2**	**81.1**	**95.2**	65.6	**69.9**	**74.0**	**85.9**	**84.7**
	LIDO (ours)	60.4	57.6	5.1	70.8	88.1	16.2	51.2	76.1	25.4	36.6	77.6	94.1	**65.7**	66.9	72.0	81.7	82.0

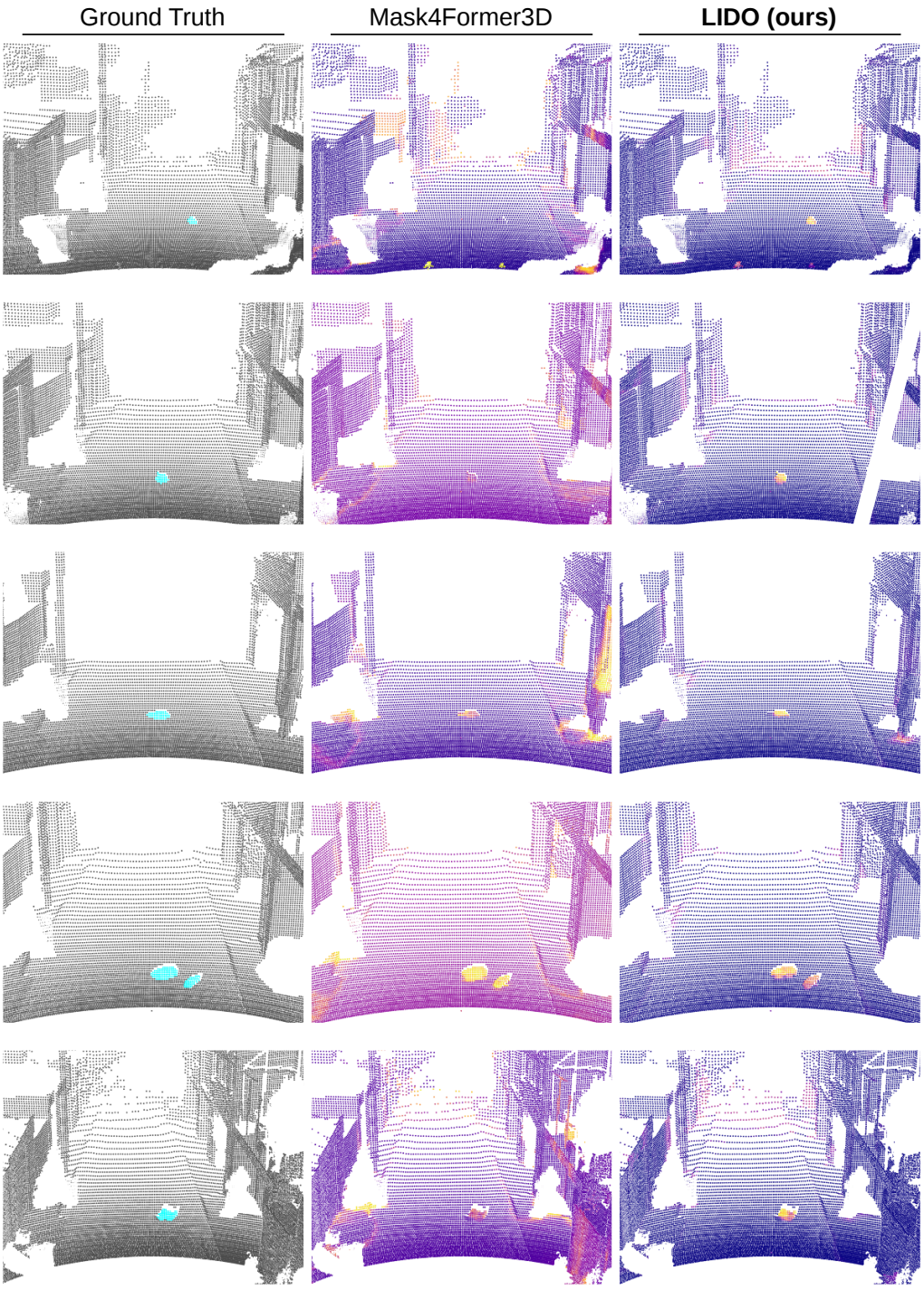


Figure 7. Qualitative comparison of 3D LiDAR anomaly segmentation results on STU validation set.

Figure 8. Qualitative comparison of 3D LiDAR anomaly segmentation results on SemanticPOSS-OoD *Multi* split.

Figure 9. Qualitative comparison of 3D LiDAR anomaly segmentation results on SemanticKITTI-OoD *Multi* split.

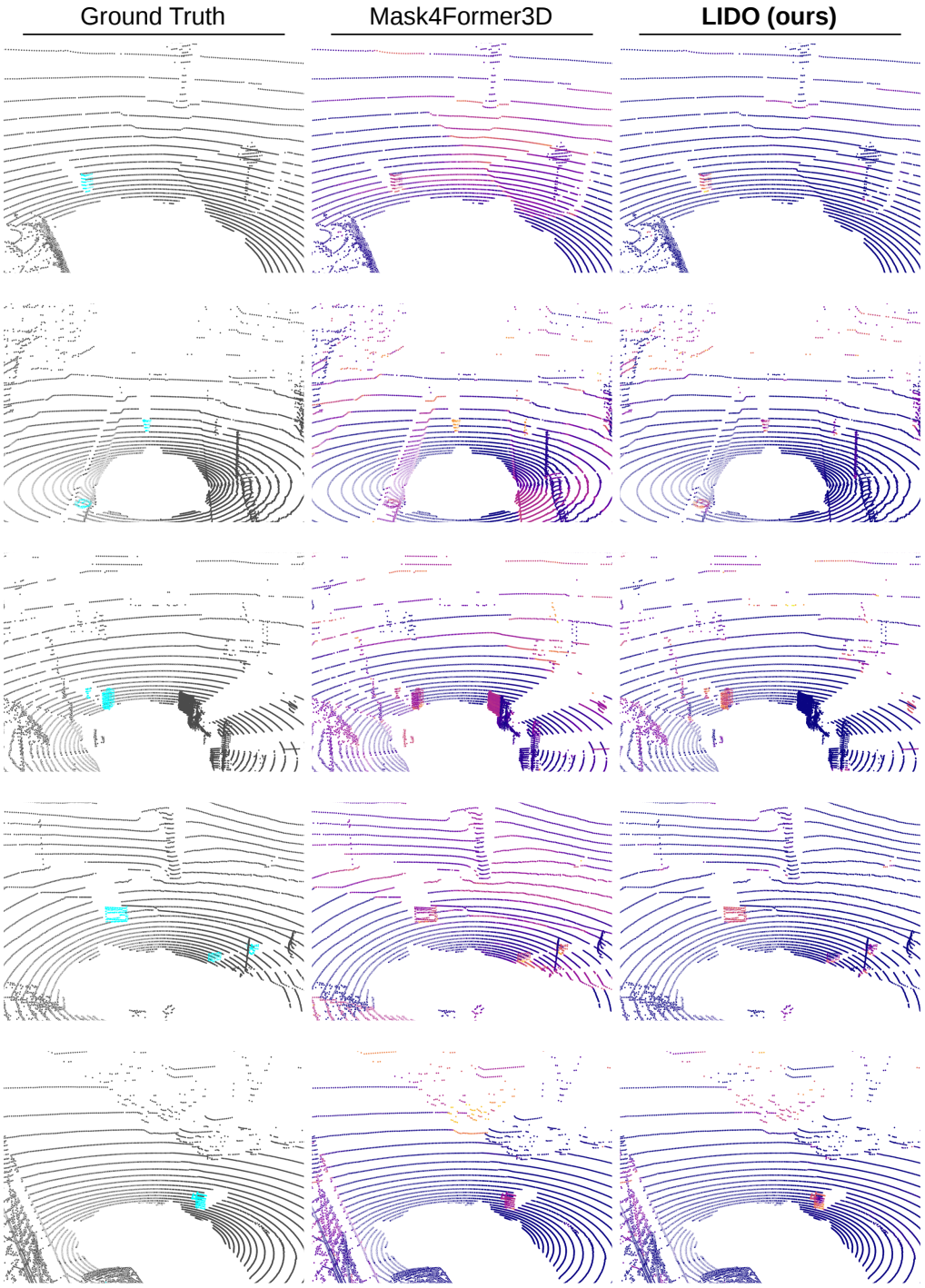


Figure 10. Qualitative comparison of 3D LiDAR anomaly segmentation results on nuScenes-OoD *Multi* split.