IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Robust Unsupervised Domain Adaptation for 3D Point Cloud Segmentation Under Source Adversarial Attacks

Haosheng Li[1§], Junjie Chen[1§], Yuecong Xu[2], and Kemi Ding[1]

***Abstract*—Unsupervised domain adaptation (UDA) frameworks have shown good generalization capabilities for 3D point cloud semantic segmentation models on clean data. However, existing works overlook adversarial robustness when the source domain itself is compromised. To comprehensively explore the robustness of the UDA frameworks, we first design a stealthy adversarial point cloud generation attack that can significantly contaminate datasets with only minor perturbations to the point cloud surface. Based on that, we propose a novel dataset, AdvSynLiDAR, comprising synthesized contaminated LiDAR point clouds. With the generated corrupted data, we further develop the Adversarial Adaptation Framework as the countermeasure. Specifically, by extending the key point sensitive loss towards the Robust Long-Tailed loss and utilizing a decoder branch, our approach enables the model to focus on long-tailed classes during the pre-training phase and leverages high-confidence decoded point cloud information to restore point cloud structures during the adaptation phase. We evaluated our method on the AdvSynLiDAR dataset, where the results demonstrate that our method can mitigate performance degradation under source adversarial perturbations for UDA in the 3D point cloud segmentation application.**

***Index Terms*—Transfer learning, semantic scene understanding, object detection, segmentation and categorization.**

I. Introduction

In applications such as autonomous driving and robotic navigation [1], 3D point cloud semantic segmentation [2] is a critical technology. Due to the complexity of 3D data and the high annotation cost, UDA for 3D point cloud semantic segmentation has emerged as an effective strategy for transferring knowledge from a labelled source domain to an unlabeled target domain. While existing UDA methods perform well on clean data, their robustness against adversarial source perturbations remains unexplored. In real-world scenarios, source domain data used for transfer can be compromised by subtle adversarial manipulations, such as geometry distortion or label noise. Although these perturbations may appear visually as minor noise, they can significantly degrade model performance, leading to erroneous decisions in robot systems [3], [4].

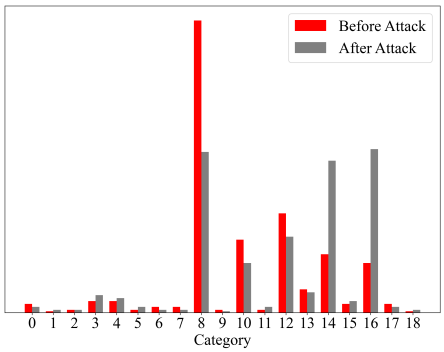

Fig. 1: Illustration of the distribution of various classes within the 3D point cloud data before and after adversarial attacks of SynLiDAR [5]. The long-tailed classes exhibit relatively stable distribution, with minimal changes observed across the pre-attack and post-attack states.

Existing cross-domain adaptation studies lack adversarially perturbed datasets, limiting the development of defense methods for 3D point cloud segmentation. To fill this gap, we propose AdvSynLiDAR, an adversarial version of SynLiDAR [5], simulating severe source-domain attacks. We apply a distance-weighted PGD [6] to adjust perturbation strength relative to viewpoint proximity, and inject high-confidence incorrect labels to introduce subtle yet harmful errors. In the face of such attacks, our task faces two significant difficulties: first, the stealthiness of adversarial attacks makes them difficult to be detected; second, the data distribution in 3D point cloud segmentation is inherently imbalanced — that is, a few classes dominate while many others are rare and underrepresented. These long-tailed classes are typically more challenging to learn due to limited supervision. These difficulties present both a challenge and an opportunity. We observe that for long-tailed data, which is common in 3D point clouds, despite its scarcity, it also has a much lower probability of being targeted by attacks. This offers a potential advantage to leverage upon long-tailed data for robust cross-domain segmentation.

To this end, we propose the Adversarial Adaptation Framework (AAF) defense strategy. Our approach is divided into two phases: pre-training and adaptation. The pre-training phase is conducted in a supervised manner using labeled source-domain data, while the adaptation phase is unsupervised, utilizing only unlabeled target-domain data. During the pre-training phase, we focus mainly on handling long-tailed classes. As shown in Fig. 1, long-tailed classes exhibit stronger

Manuscript received: April 2, 2025; Revised July 8, 2025; Accepted September 20, 2025.

This paper was recommended for publication by Editor Faust Aleksandra upon evaluation of the Associate Editor and Reviewers' comments.

The work is supported in part by National Science and Technology Major Project (No. 2025ZD1605600), and in part by National Natural Science Foundation of China (No. 62303212).

H. Li, J. Chen and K. Ding are with the Department of Automation and Intelligent Manufacturing (AIM), Southern University of Science and Technology, Shenzhen, China. Email: {12332662, 12332651} @mail.sustech.edu.cn, dingkm@sustech.edu.cn.

Y. Xu is with the Department of Electrical and Computer Engineering, National University of Singapore. Email: yc.xu@nus.edu.sg

Our code is available at https://github.com/Hash-Lee-777/Robust_3D_UDA.

Digital Object Identifier (DOI): see top of this page.

§Equal contribution.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

robustness against attacks than head classes. Inspired by the KPS loss [7] in 2D classification tasks, we introduce the Robust Long-Tailed Loss (RLT loss) to enhance the model's attention to 3D long-tailed class data. We first redefine the key points in KPS loss based on the characteristics of 3D point clouds. We then design a new margin adjustment mechanism that dynamically adjusts the margins based on the actual distribution of each class to address the sparseness and uneven distribution of long-tailed classes.

Further, while some methods mitigate corrupted inputs, they struggle to recover spatial structures under stealthy attacks. Inspired by VRCNet [8], we introduce a decoder branch during adaptation to reconstruct point distributions via probabilistic modeling. The refined outputs, fused with high-confidence neighborhood semantics, yield robust pseudo-labels without requiring explicit attack detection.

In summary, our contributions are threefold. (i) To the best of our knowledge, our research is the first to explore cross-domain robustness and defense strategies in 3D point cloud semantic segmentation. (ii) Our defense strategy AAF improves the model's robustness by introducing the RLT loss during the pre-training phase to enhance the focus on long-tailed classes and effectively utilizing high-confidence decoded point cloud information during the adaptation phase. (iii) Experimental results demonstrate that AAF achieves a remarkable improvement on the SemanticKITTI [9] and SemanticPOSS [10] datasets after the attack, with an increase of 11.61 and 9.85 mIoU, respectively.

II. Related Work

A. Unsupervised Domain Adaptation

To mitigate domain shift in unsupervised settings, domain adaptation methods align distributions at different levels. Input-level approaches [11] use style transfer to harmonize visual appearance. Feature-level methods [12] extract domain-invariant features by aligning latent spaces. Output-level adaptation [13] aligns prediction maps via adversarial training and target statistics. In the 2D domain, methods such as FDA [14] and SRoUDA [15] also explore robustness in UDA, but are not directly applicable to 3D point clouds due to structural differences.

Recent 3D UDA works have also started exploring unsupervised domain adaptation for 3D semantic segmentation. For example, M. Rochan et al. [16] introduce self-supervised learning and gated adapters to enhance LiDAR semantic segmentation across domains. Similarly, T-UDA [17] leverages temporal consistency for better domain transfer. Beyond semantic segmentation, recent work [18] explores source-free domain adaptation for primitive segmentation in industrial scenarios, using multi-confidence cues to refine pseudo-labels. Though targeting a different task, it illustrates the broader relevance of domain adaptation to geometric understanding.

B. Point Cloud Adversarial Attacks and Countermeasures

Adversarial attacks on 3D point clouds include gradient-based [19] and optimization-based [3] methods, as well as structure-based attacks like point shift [20], addition [4], and drop [21]. While effective for classification and segmentation, their influence on cross-domain adaptation, especially under source perturbations, remains underexplored.

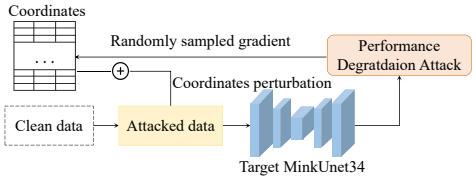

Fig. 2: Generating adversarial source domain coordinates, γ^i controls the intensity of perturbations based on the distance to the viewpoint center.

Countermeasures against 3D point cloud adversarial attacks include data-driven and model-driven approaches. Data-driven methods like PointGuard [22] use voting over random subsets, while model-driven methods such as LPC [23] transform point clouds into 2D lattices with structured encoding to improve robustness. However, to the best of our knowledge, existing research has not addressed defense mechanisms for UDA in 3D point cloud segmentation under adversarial attacks targeting the source domain.

III. The AdvSynLiDAR Dataset

Source domain point clouds are vulnerable to perturbations from sensor noise, environmental changes, and preprocessing artifacts [24], which exacerbate domain gaps and hinder UDA performance. However, standardized evaluation protocols for such perturbations remain lacking. To this end, we propose the AdvSynLiDAR dataset based on SynLiDAR [5], simulating adversarial attacks on source data by assuming access to $\mathbf{x}_\mathcal{S}$ and $\mathbf{y}_\mathcal{S}$. To obtain a point-wise classification and confidence output of point cloud, the pre-trained model θ provides the probability distribution of the i-th point in $\mathbf{x}_\mathcal{S}$, which is denoted as $\Phi_\theta(\mathbf{x}_\mathcal{S})^i = p(\cdot|\mathbf{x}_\mathcal{S})^i \in \mathbb{R}^{|c|}$, where $|c|$ denotes the total number of semantic classes.

A. Imperceptible Perturbation Points

In practice, adversarial perturbations are subtle and spatially dependent, which complicates their detection. To simulate realistic attacks, we adopt a perceptually aware PGD strategy [6], which applies more substantial perturbations to distant regions while preserving dense, perceptually sensitive areas. This enables precise evaluation of UDA robustness under adversarial source conditions. Motivated by the observation that central regions near the viewpoint exhibit higher point density and perceptual sensitivity, we introduce a dynamic adjustment factor γ^i to regulate the magnitude of perturbations based on their proximity to the viewpoint center Z. For each point $\mathbf{x}_\mathcal{S}^i$ in the point cloud, the distance d_i from the viewpoint center is calculated by $d(\mathbf{x}_\mathcal{S}^i) = \|\mathbf{x}_\mathcal{S}^i - Z\|$, and the dynamic factor γ^i is denoted as:

$$\gamma^i = \frac{d(\mathbf{x}_\mathcal{S}^i)}{d_{\max}}, \tag{1}$$

where $d_{\max}$ denotes the maximum distance to the viewpoint.

The adjusted PGD attack is denoted as:

$$\mathbf{x}_{i+1}^{\text{adv}} = \text{clip}_{\mathbf{x},\epsilon}\left(\mathbf{x}_i^{\text{adv}} + \gamma^i \odot \delta \cdot \text{sign}(\nabla_{\mathbf{x}_i^{\text{adv}}} L(f(\mathbf{x}_i^{\text{adv}};\theta), \mathbf{y}_i))\right), \tag{2}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

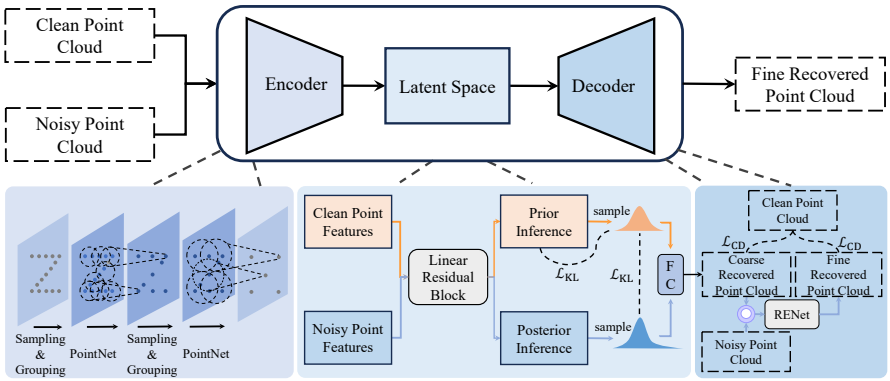

Fig. 3: The figure illustrates the pre-training process of the decoder, where the symmetric Chamfer Distance $\mathcal{L}_{\mathbf{CD}}$ quantifies the discrepancy between the reconstructed and original point clouds. Additionally, the Kullback-Leibler divergence loss $\mathcal{L}_{\mathbf{KL}}$ aligns the probability distribution from prior and posterior inference, promoting consistency in the latent space.

where θ is the parameters of the target model, δ denotes the perturbation intensities of the original point cloud. Term $\nabla_{\mathbf{x}_i^{\text{adv}}} L$ is the gradient value of the adversarial loss function concerning the adversarial point cloud in the i-th iteration, and y_i corresponds to the ground truth label of the original point. The clip$(\cdot)$ function constrains the perturbation range of the adversarial point cloud within $[0, \epsilon]$ relative to the original data. As shown in Fig. 2, by dynamically adjusting the magnitude of perturbations, we generate adversarial point clouds that effectively maintain the potency of the attack while remaining less detectable.

B. Noisy Semantic Labels

Manual 3D point cloud annotations often suffer from label noise, particularly for occluded or distant objects [25]. While most UDA methods assume clean source labels, real-world settings involve systematic high-confidence mislabels. To evaluate this, we inject such noise into the source domain by deriving adversarial labels based on prediction confidence from the model's logits: given a point cloud $\mathbf{x}_{\mathcal{S}}$ and a pre-trained model $\theta_{\mathcal{S}}$, labels are generated via:

$$\tilde{\mathbf{y}}^i = \underset{\mathbf{y} \in \mathbf{Y}_{\mathcal{S}} \setminus \{\mathbf{y}^i\}}{\arg\max} \{p(\mathbf{y} \mid \mathbf{x}_{\mathcal{S}}^i) \mid p(\mathbf{y} \mid \mathbf{x}_{\mathcal{S}}^i) > \tau_{adv}\}, \quad (3)$$

where τ_{adv} denotes the confidence threshold, and $p(\mathbf{y}|\mathbf{x}_{\mathcal{S}}^i; \theta_{\mathcal{S}})$ denotes the predicted class probabilities. This approach generates the highest-confidence incorrect class as the adversarial target, mimicking real-world scenarios with noisy labels.

IV. Methodology

To address the vulnerability of cross-domain point cloud segmentation to adversarial perturbed source domains, we propose an Adversarial Adaptation Framework (AAF), which enhances robustness by enforcing resilient training objectives and maintaining semantic consistency under source-domain perturbations.

A. Preliminary and Overview

In the context of adversarial cross-domain adaptation, the labeled source domain is denoted as $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{S}|}$. Here, $\mathbf{x}_{\mathcal{S}} \in \mathbb{R}^{N \times 3}$ represents the 3D point cloud coordinates of the source domain. The label $\mathbf{y}_{\mathcal{S}}^j \in \mathbf{Y}_{\mathcal{S}} = \{0, 1, \ldots, c-1\}$ denotes the semantic label for each point, where $\mathbf{Y}_{\mathcal{S}}$ is the label space of point clouds across domains, and c is the number of classes. The objective of our cross-domain adaptation task is to extract semantic information from the source domain and transfer it to an unlabeled target domain, which is denoted as $\mathcal{T} = \{(\mathbf{x}_i)\}_{i=1}^{|\mathcal{T}|}$. To obtain a point-wise classification of point clouds, the pre-trained model θ provides the probability distribution of the j-th point in a source point cloud $\mathbf{x}_{\mathcal{S}}$, which is denoted as $\Phi_\theta(\mathbf{x}_{\mathcal{S}})^j = p(\cdot|\mathbf{x}_{\mathcal{S}})^j \in \mathbb{R}^{|c|}$.

B. Adversarial Adaptation Framework (AAF)

1) Robust Long-Tailed Loss (RLT Loss): During the pre-training phase, to balance the advantages of KPS loss [7] in handling long-tailed classes with the strengths of Soft-DICELoss in managing head classes, we introduce the Robust Long-Tailed loss (RLT loss). SoftDiceLoss is widely used in segmentation tasks to measure the overlap between predictions and ground truth labels. It is defined as:

$$\mathcal{L}_{\text{SD}} = 1 - \frac{2\sum_{i=1}^{N} \mathbf{p}_{\mathcal{S}}^i \mathbf{y}_{\mathcal{S}}^i}{\sum_{i=1}^{N} \left(\mathbf{p}_{\mathcal{S}}^i\right)^2 + \sum_{i=1}^{N} \left(\mathbf{y}_{\mathcal{S}}^i\right)^2}. \quad (4)$$

This formula calculates the Dice similarity [26] coefficient between the predicted results $\mathbf{p}_{\mathcal{S}}^i$ and the ground truth labels $\mathbf{y}_{\mathcal{S}}^i$, where N is the number of samples. Initially designed for 2D classification tasks, KPS loss aims to enhance the separability of features by adjusting the boundaries between key points and non-key points in the feature space. To apply KPS loss to higher-dimensional 3D segmentation tasks, we make several improvements to the original KPS loss:

Geometric Importance in 3D Key Point Definition. We introduce geometric importance as the criteria for defining key points in 3D point clouds, allowing for better identification of critical points under attack conditions. Specifically, we define the geometric importance of a key point using the following formula:

$$\text{Importance}\left(\mathbf{x_i}\right) = \frac{\sum_{j=1}^{k} \text{Distance}\left(\mathbf{x_i}, \mathbf{x_j}\right)}{k}, \quad (5)$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

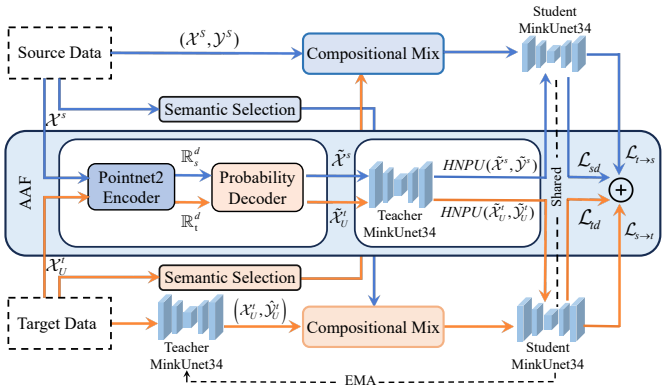

Fig. 4: Illustration of cross-domain adaptation with the AAF framework. The probability decoder branch dynamically adjusts input distributions while HNPU enhances pseudo generation throughout the adaptation process.

where $\mathbf{x_i}$ is the current point, $\mathbf{x_j}$ are its k nearest neighbors, and Distance $(\mathbf{x_i}, \mathbf{x_j})$ calculates the Euclidean distance between point $\mathbf{x_i}$ and $\mathbf{x_j}$. This enhances the discriminative ability of KPS loss for long-tailed classes.

Dynamic Boundary Adjustment for Class Imbalance. Standard classification assumes shared decision boundaries across classes, which can be suboptimal for long-tailed classes with sparse samples. Given the severe class imbalance in 3D point clouds (Fig. 1), we introduce a dynamic boundary adjustment mechanism that adapts decision margins based on class density and classification difficulty:

$$m_{\mathbf{y}_i} = \alpha \cdot \log\left(\frac{F_{\mathbf{y}_i}}{F_{max}}\right), \tag{6}$$

where α is a scaling factor, $F_{\mathbf{y}_i}$ is the sample count for class $\mathbf{y}_i$, and F_{max} is the sample count of the largest class. In this way, the smaller the class size (i.e., long-tailed classes), the larger the boundary adjustment term $m_{\mathbf{y}_i}$, giving these classes more decision space during classification.

In summary, the KPS loss in 3D tasks is defined as:

$$\mathcal{L}_{\text{KPS}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s\cdot\left(\cos\left(\theta_{\mathbf{y}_i}\right)-m_{\mathbf{y}_i}\right)}}{e^{s\cdot\left(\cos\left(\theta_{\mathbf{y}_i}\right)-m_{\mathbf{y}_i}\right)}+\sum_{j\neq\mathbf{y}_i}e^{s\cdot\cos(\theta_j)}}, \tag{7}$$

where s is a scaling factor, θ is the angle between the feature vector and the class anchor vector, and $m_{\mathbf{y}_i}$ is the boundary for class $\mathbf{y}_i$. This loss function encourages more significant boundaries for key points, thereby promoting the separability of long-tailed class features.

The overall loss function $\mathcal{L}_{\text{RLT}}$, can thus be defined as:

$$\mathcal{L}_{\text{RLT}} = \lambda_b \cdot \mathcal{L}_{\text{KPS}} + (1-\lambda_b)\cdot\mathcal{L}_{\text{SD}}\,, \tag{8}$$

where we employ Bayesian optimization [27] to dynamically adjust the regularization parameter λ_b in the loss function, aiming to identify the optimal λ_b at each iteration and prevent overfitting.

2) Probability Decoder Branch: As shown in Fig. 4, our AAF framework incorporates a probability decoder and a High-confidence Nearest Pseudo Update (HNPU) module to improve robustness against source-domain adversarial noise. The decoder reconstructs clean geometric features from corrupted source inputs. Meanwhile, HNPU improves pseudo-label reliability by aggregating semantic information from high-confidence neighborhoods.

Algorithm 1 High-Confidence Nearest Pseudo Update (HNPU)

Input: Decoded coordinates P_{fine}, original coordinates P, original labels L, confidence scores C_{conf}, confidence threshold τ, distance threshold d_{th}, number of neighbors k.

Output: Updated points P_{u}, updated labels L_{u}

```
Construct a KD-tree T = KD-Tree(P).
for each point x ∈ P_fine do
    Find k nearest neighbors N_k(x, T).
    Assign pseudo-label ŷ(x) using Eq. (13).
end for
return P_u, L_u
```

Probability Decoder. To mitigate adversarial distortions, we adopt a variational probability decoder inspired by VRCNet [8] to restore clean geometry. As shown in Fig. 3, we first construct perturbed inputs $\mathbf{x}_g$ by injecting Gaussian noise and applying random augmentations to clean source coordinates $\mathbf{x}_\mathcal{S}$, i.e., $\mathbf{x}_g = r(\mathbf{x}_\mathcal{S} + \mathcal{N}(0,\sigma^2))$. The decoder reconstructs clean geometry in three stages: reconstruction, completion, and enhancement.

In the reconstruction stage, the encoder maps the clean source input $\mathbf{x}_\mathcal{S}$ to a latent variable $\mathbf{z}_g$ with posterior $q_\phi(\mathbf{z}_g|\mathbf{x}_\mathcal{S})$, while the prior $p(\mathbf{z}_g) = \mathcal{N}(\mathbf{0},\mathbf{I})$ is a standard Gaussian. A coarse reconstruction $\mathbf{x}'_r$ is is sampled from $p^r_\theta(\mathbf{x}_\mathcal{S}|\mathbf{z}_g)$. The reconstruction loss encourages latent regularity and geometric fidelity:

$$\begin{aligned}\mathcal{L}_{rec} =& -\lambda\mathbf{KL}\left[q_\phi(\mathbf{z_g}|\mathbf{x}_\mathcal{S})\|p(\mathbf{z_g})\right]\\ &+\mathbb{E}_{p_{data}(\mathbf{x}_\mathcal{S})}\mathbb{E}_{q_\phi(\mathbf{z_g}|\mathbf{x}_\mathcal{S})}\left[\log p^r_\theta(\mathbf{x}_\mathcal{S}|\mathbf{z_g})\right],\end{aligned} \tag{9}$$

where $p_{data}(\mathbf{x}_\mathcal{S})$ represents the true data distribution of point clouds in the source domain, and λ is a weighting parameter.

In the completion stage, the perturbed input $\mathbf{x}_g$ is encoded to produce a latent posterior distribution $p_\psi(\mathbf{z}_g|\mathbf{x}_g)$, which is encouraged to match the posterior $q_\phi(\mathbf{z}_g|\mathbf{x}_\mathcal{S})$ derived from clean inputs. The reconstructed output $\mathbf{x}'_c$ shares decoder weights with the reconstruction stage:

$$\begin{aligned}\mathcal{L}_{com} =& -\lambda\mathbf{KL}\left[q_\phi(\mathbf{z_g}|\mathbf{x}_\mathcal{S})\|p_\psi(\mathbf{z_g}|\mathbf{x_g})\right]\\ &+\mathbb{E}_{p_{data}(\mathbf{x_g})}\mathbb{E}_{p_\psi(\mathbf{z_g}|\mathbf{x_g})}\left[\log p^c_\theta(\mathbf{x}_\mathcal{S}|\mathbf{z_g})\right].\end{aligned} \tag{10}$$

The enhancement stage uses self-attention and multi-scale upsampling to generate fine-grained outputs $\mathbf{x}'_f$. To capture geometric errors and stabilize training, we adopt the symmetric Chamfer Distance $\mathcal{L}_{\mathbf{CD}}$ as a surrogate for log-likelihood. The overall decoder loss is:

$$\begin{aligned}\mathcal{L}_d =& \lambda_{rec}\mathcal{L}_{rec} + \lambda_{com}\mathcal{L}_{com} + \lambda_{fine}\mathcal{L}_{fine}\\ =& \lambda_{rec}\left[\mathcal{L}_{\mathbf{KL}}(q_\phi(\mathbf{z_g}|\mathbf{x}_\mathcal{S}),\mathcal{N}(\mathbf{0},\mathbf{I})) + \mathcal{L}_{\mathbf{CD}}(\mathbf{x}'_r,\mathbf{x}_\mathcal{S})\right]\\ &+\lambda_{com}\left[\mathcal{L}_{\mathbf{KL}}(p_\psi(\mathbf{z_g}|\mathbf{x_g}),q_\phi(\mathbf{z_g}|\mathbf{x}_\mathcal{S})) + \mathcal{L}_{\mathbf{CD}}(\mathbf{x'_c},\mathbf{x}_\mathcal{S})\right]\\ &+\lambda_{fine}\mathcal{L}_{\mathbf{CD}}(\mathbf{x}'_f,\mathbf{x}_\mathcal{S}).\end{aligned} \tag{11}$$

High-Confidence Nearest Pseudo Update. Inspired by prior work [28], we further enhance robustness by leveraging high-

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I: Results for class-wise semantic segmentation on SynLiDAR to SemanticKITTI. Selection-perc is the hyperparameter that regulates the ratio of selected classes. Source classes are selected randomly with a selection-perc of 0.5 originally.

Source	sel-p.	AAF	car	bi.cle	mt.cle	truck	oth-v.	pers.	b.clst	m.clst	road	park.	sidew.	oth-g.	build.	fence	veget.	trunk	terra.	pole	traff.	mIoU
SynLiDAR	0.5		79.30	9.12	30.68	20.88	11.54	24.91	27.00	17.62	78.03	14.65	47.12	0.19	53.52	13.66	68.72	32.75	31.94	38.17	13.96	32.30
	0.9		80.07	5.46	25.70	30.32	10.86	23.32	21.51	7.77	80.04	18.70	49.36	0.15	49.45	13.46	67.26	29.56	34.01	40.04	12.46	31.55
	0.5	✓	77.51	6.59	30.51	19.01	9.45	24.83	29.26	18.93	77.72	13.79	45.94	0.09	50.09	13.74	68.05	28.27	29.20	37.76	14.00	31.30
AdvSynLiDAR	0.5		43.22	0.72	5.96	5.65	2.83	18.50	4.24	9.17	42.57	1.33	8.88	1.15	36.71	9.20	35.90	19.29	5.24	33.28	6.47	15.28
	0.9		40.50	9.47	6.46	10.39	4.94	20.74	28.77	15.50	46.65	3.70	14.00	1.51	43.09	10.74	41.66	23.66	6.67	38.48	10.53	19.87
	0.5	✓	55.32	5.16	4.52	15.78	6.11	14.26	19.75	7.72	56.92	6.79	16.77	1.09	41.12	10.90	61.21	25.63	7.46	36.42	8.12	**22.50**

TABLE II: Results for class-wise semantic segmentation on SynLiDAR to SemanticPOSS. Selection-perc is the hyperparameter that regulates the ratio of selected classes. Source classes are selected randomly with a selection-perc of 0.5 originally.

Source	sel-p.	AAF	pers.	rider	car	trunk	plants	traf.	pole	garb.	buil.	cone.	fence	bike	groun.	mIoU
SynLiDAR	0.5		55.56	52.78	35.09	23.43	71.01	22.97	31.88	30.35	67.48	21.22	24.82	10.00	78.60	40.40
	0.9		54.59	51.67	36.06	23.55	71.40	23.56	33.41	30.48	66.8	19.25	24.71	9.74	79.05	40.33
	0.5	✓	55.63	51.19	37.69	21.73	72.50	18.66	36.89	23.25	68.60	21.56	26.20	9.95	79.23	40.24
AdvSynLiDAR	0.5		55.90	34.94	11.01	14.36	27.51	14.28	18.19	10.63	52.12	5.12	26.77	16.78	71.38	27.61
	0.9		54.80	42.37	13.56	15.91	40.45	13.24	21.50	12.89	52.68	3.96	21.63	14.57	73.12	29.28
	0.5	✓	55.70	46.36	10.99	16.51	28.18	21.43	31.52	8.54	49.27	9.77	40.00	14.80	69.89	**30.30**

confidence regions. We propose the HNPU Algorithm 1, which aggregates information from confident neighborhoods. For each decoded point $\mathbf{x}$, we retrieve its k neighbors from the original point set P using a KD-tree T:

$$\mathcal{N}_k(\mathbf{x}, T) = \{\mathbf{x}' \in P \mid \|\mathbf{x} - \mathbf{x}'\| \leq d_{\text{th}}, |\mathcal{N}_k| = k\}, \quad (12)$$

where d_{th} is the distance threshold. Then the pseudo-label for $\mathbf{x}$ is updated with aggregated confidence scores:

$$\hat{\mathbf{y}}(\mathbf{x}) = \begin{cases} \mathbf{y}(\mathbf{x}), & C_{\text{conf}}(\mathbf{x}) > \tau, \\ \mathbf{y}(\mathbf{x}'), & C_{\text{conf}}(\mathbf{x}) \leq \tau \wedge \max\limits_{\mathbf{x}' \in \mathcal{N}_k(\mathbf{x}, P)} C_{\text{conf}}(\mathbf{x}') > \tau. \\ -1, & \text{otherwise.} \end{cases} \quad (13)$$

As shown in Fig. 4, we adopt a teacher-student framework for cross-domain adaptation using the total loss $\mathcal{L}_{tot} = \mathcal{L}_{s\to t} + \mathcal{L}_{t\to s}$ to enforce semantic consistency. In the decoder branch, we further include segmentation losses derived from decoded coordinates and pseudo-labels refined by HNPU. The updated source-domain segmentation loss is defined as:

$$\mathcal{L}_{sd} = \mathcal{L}_{seg}\left(\Phi_\theta\left(\text{HNPU}(\mathbf{x}_\mathcal{S}, \mathbf{y}_\mathcal{S})\right), L_{\mathcal{S}\text{u}}\right), \quad (14)$$

and the updated segmentation loss for the target domain point clouds is denoted as:

$$\mathcal{L}_{td} = \mathcal{L}_{seg}\left(\Phi_\theta\left(\text{HNPU}(\mathbf{x}_\mathcal{T}, \hat{\mathbf{y}}_\mathcal{T})\right), L_{\mathcal{T}\text{u}}\right), \quad (15)$$

in which $L_{\mathcal{T}\text{u}}$ and $L_{\mathcal{S}\text{u}}$ are the pseudo-labels updated by HNPU with decoded coordinates and the confidence information for both domains. The overall objective function is defined as the sum of the decoder segmentation loss and the loss for training the decoder, which is defined as follows:

$$\mathcal{L} = \mathcal{L}_{tot} + \lambda_d \mathcal{L}_d + \lambda_{sd} \mathcal{L}_{sd} + \lambda_{td} \mathcal{L}_{td}, \quad (16)$$

where λ_{sd}, λ_{td}, and λ_d are the weighting hyper-parameters for the loss functions. The decoder branch reduces adversarial shifts by aligning inputs with clean source geometry, while HNPU alleviates the impact of corrupted source data, jointly enhancing cross-domain adaptation performance.

TABLE III: Evaluation of our AAF framework combined with different UDA methods. SK and ASK denote the SemanticKITTI and adversarial SemanticKITTI (ASK) source domains, respectively. NS and WO denote the nuScenes-lidarseg and Waymo target domains. ✓ indicates that the AAF module is applied.

Method	AAF	SK→NS	SK→WO	ASK→NS	ASK→WO
CosMix	✗	25.14	**26.50**	19.99	21.69
CosMix	✓	**25.18**	26.39	**24.10**	**24.64**
T-UDA	✗	**36.58**	**36.70**	31.96	32.89
T-UDA	✓	36.52	36.21	**34.39**	**35.14**

V. EXPERIMENTS

In this section, we evaluate the performance of our proposed Adversarial Adaptation Framework (AAF) through adversarial cross-domain semantic segmentation experiments, with a specific focus on the synthetic-to-real domain shift. We present class-wise semantic segmentation results for two distinct scenarios and provide a detailed comparison to illustrate the effectiveness of our approach. Additionally, ablation studies and empirical analysis are also presented to justify the design of AAF.

A. Datasets

We perform adversarial cross-domain adaptation for 3D semantic segmentation tasks using widely adopted autonomous driving datasets, covering both synthetic-to-real and real-to-real scenarios. For the synthetic-to-real setting, we adopt SynLiDAR [5] as the synthetic source domain, and use SemanticPOSS [10] and SemanticKITTI [9] as the real-world target domains. SynLiDAR is a large-scale synthetic dataset created using Unreal Engine, comprising 198,396 annotated point clouds with 32 semantic classes. SemanticPOSS includes 2,988 real-world point clouds categorized into 14 semantic classes. SemanticKITTI is a comprehensive segmentation dataset derived from LiDAR acquisitions of the KITTI dataset, consisting of 43,552 annotated point clouds with over 19

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

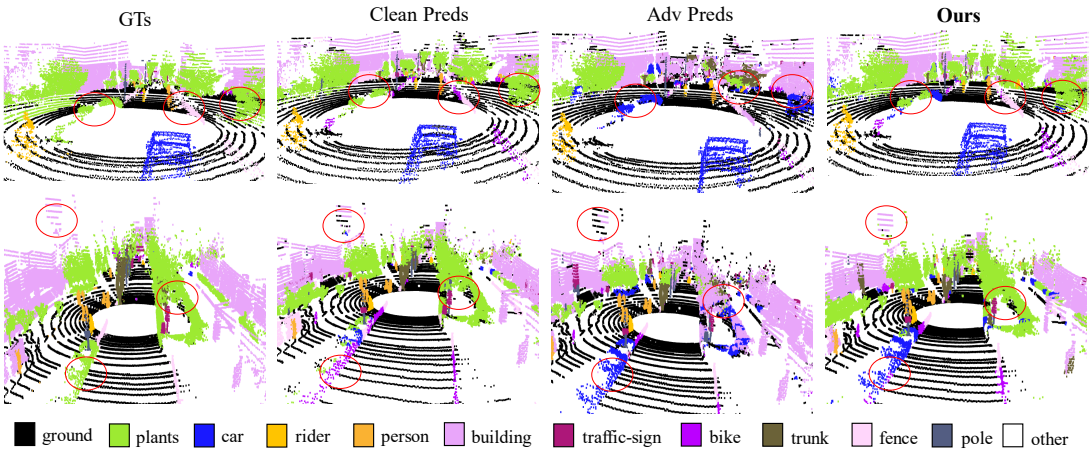

Fig. 5: Visualization of cross-domain adaptation semantic segmentation results on SynLiDAR → SemanticPOSS. From left to right: point clouds with the ground truth labels, adaptation with clean SynLiDAR, adaptation with AdvSynLiDAR, and Ours (adaptation with AdvSynLiDAR + AAF)

semantic classes. To assess the generalization ability of our method under realistic deployment conditions, we further consider real-to-real adaptation tasks, using SemanticKITTI as the source domain and nuScenes-lidarseg [29] and Waymo Open Dataset [30] as target domains. The nuScenes-lidarseg dataset contains more than 40,000 LiDAR scans captured with a 32-beam Velodyne sensor in diverse urban environments; however, the point clouds are relatively sparse due to the lower scan resolution and lower capture frequency. In contrast, the Waymo Open Dataset comprises over 100,000 dense LiDAR frames captured using a 64-beam sensor, providing richer geometry and more complete object structures.

B. Experimental Settings

Following CosMix [31], we adopt the same training and validation splits. We select the pre-trained MinkUNet34 on a clean SynLiDAR dataset as our target semantic segmentation model. For generating adversarial examples, we adopt a perceptually-aware perturbation method with $\delta = 0.01$ and $\epsilon = 0.05$. The confidence threshold τ_{adv} is set to 0.85. In the RLT loss module, the regularization parameter λ_{b} is set to 0.9. In the probability decoder module, we first train the decoder with 4096 sampled points. During adaptation, we continue to train the decoder with randomly sampled point clouds to prevent overfitting. The learning rate is empirically set to 0.0001. The reconstruction loss in uses $\lambda = 1.0$, and the balancing weights in the overall decoder loss are: $\lambda_{\text{rec}} = 10$, $\lambda_{\text{com}} = 0.5$, and $\lambda_{\text{fine}} = 0.01$. For the HNPU module, we use a confidence threshold $\tau = 0.95$, number of neighbors $k = 6$, and a distance threshold $d_{\text{th}} = 0.5$. Additionally, λ_d, $\lambda_{\mathcal{S}d}$, $\lambda_{\mathcal{T}d}$ in the overall objective function are empirically set to 0.1 and are fixed.

For real-to-real adaptation tasks, we adopt a consistent training strategy to ensure a fair comparison between CosMix and T-UDA [17]. Specifically, we pretrain the segmentation model on the clean SemanticKITTI dataset for 20 epochs with a learning rate of 0.012. During the adaptation stage, we train the model on the target domain for 10 epochs using a reduced learning rate of 0.001. The optimizer and other hyperparameters are kept identical to those used in T-UDA to maintain consistency across all experimental settings. All experiments in this section are conducted using the PyTorch [32] library and are conducted on 4× NVIDIA GeForce RTX 4090.

C. Performances and Comparisons

Tables. I and II present class-wise cross-domain semantic segmentation results on SynLiDAR → SemanticKITTI and SynLiDAR → SemanticPOSS, respectively.

1) Impact of Adversarial Perturbations: Even minor adversarial perturbations in the source domain lead to significant performance degradation in the target domain across multiple semantic classes. For instance, vegetation IoU in SemanticKITTI drops from 68.72 to 35.90, highlighting the vulnerability of head classes to adversarial attacks. Head classes refer to frequently occurring semantic classes with a large number of points, which dominate the training distribution. Similarly, classes such as plants in SemanticPOSS also show a considerable drop in IoU. This highlights the necessity for robust cross-domain adaptation techniques, as adversarial modifications propagate through the network and compromise semantic transferability.

2) Robustness of Long-Tailed Classes: Notably, we observe the impact of adversarial perturbations is less severe on long-tailed classes like persons, poles, and bicyclists, which tend to have sparser representations in the dataset. These classes maintain relatively stable segmentation performance compared to head classes, illustrating the inherent distribution imbalance within the source domain and how adversarial perturbations primarily affect the more frequent classes. We observe an improvement in semantic segmentation performance under adversarial attacks by empirically adjusting the selection-perception parameter to 0.9, which increases the focus on long-tailed classes. Specifically, the mIoU on the SemanticKITTI dataset improved from 15.28 to 19.87, while on the SemanticPOSS dataset, it increased from 27.61 to 29.28. This

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE IV: Ablation experiments with the components of AAF. RLT Loss is the Robust Long-Tailed Loss used in AAF, Pro.D. represents the probability decoder branch with HNPU.

Methods	Fine-tuning	RLT Loss	Pro.D.	mIoU	
				SemanticKITTI	SemanticPOSS
CosMix	-	-	-	15.28	27.75
(a)		✓		22.53	28.11
(b)			✓	20.07	30.14
(c)		✓	✓	22.78	30.30
(d)	✓			26.17	35.66
(e)	✓	✓		26.24	35.81
(f)	✓		✓	26.43	36.45
(g)	✓	✓	✓	**26.89**	**37.60**

adjustment allows long-tailed classes to partially recover their segmentation performance while mitigating the performance degradation observed in head classes.

3) Effectiveness of AAF: The AAF mitigates the adverse effects caused by perturbations in the source domain. With AAF, we observe substantial improvements in segmentation performance across both datasets. For example, on the SemanticKITTI dataset, applying the AAF leads to an improvement of mIoU from 15.28 to 22.78 when the selection percentage is 0.5. AAF also demonstrates the ability to stabilize the performance degradation of long-tailed classes while improving the overall segmentation. Specifically, in the case of SemanticPOSS, the performance of long-tailed classes such as rider and pole is preserved, maintaining performance under adversarial conditions.

To further validate the effectiveness of AAF in realistic deployment settings, we evaluate our framework under real-to-real domain adaptation tasks. As shown in Table III, our method achieves performance gains of 4.11 and 2.95 mIoU under the adversarial SemanticKITTI (ASK) source domain for nuScenes and Waymo, respectively, when integrated with CosMix. These results demonstrate that AAF effectively mitigates the negative impact of source perturbations and generalizes well to practical adaptation scenarios. The RLT loss and HNPU module exhibit robustness across diverse point cloud characteristics. Despite the structural sparsity of 32-beam LiDAR in nuScenes and the higher density of 64-beam scans in Waymo, AAF consistently improves segmentation performance. These results suggest that the RLT loss effectively enhances the representation of underrepresented classes in sparse scenes, while HNPU reliably refines pseudo-labels under varying neighborhood densities.

D. Ablation Studies, Analysis and Discussions

To validate the AAF framework and its components, we perform extensive ablation studies (Table IV), evaluating the contributions of RLT loss during pre-training, the probability decoder (Pro.D.) during adaptation, and the effect of fine-tuning with a small portion of clean source data under adversarial conditions.

1) Fine-tuning: Prior work [33] shows that limited clean data can significantly enhance robustness. We adopt 5% clean source data for fine-tuning, updating only the penultimate and final layers while freezing others. Training halts if source validation performance plateaus for three epochs. As shown in row (d), fine-tuning boosts mIoU from 15.28 to 26.17 and 27.75 to 35.66, confirming its effectiveness.

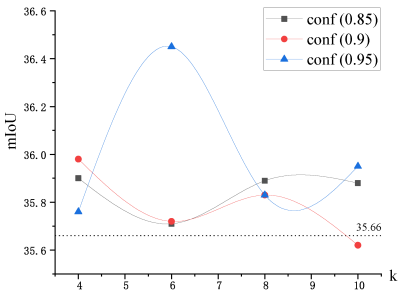

Fig. 6: mIoU results on SemanticPOSS demonstrating the effects of varying neighboring points (k) and confidence levels (conf) in the AAF framework.

2) RLT Loss: Incorporating RLT loss without fine-tuning (row (a)) yields substantial improvements, particularly on SemanticKITTI, where the mIoU increases from 15.28 to 22.53. By concentrating on these long-tailed classes, RLT loss enables the model to focus more effectively on critical semantic information, enhancing its ability to generalize across domains. Combining RLT loss with fine-tuning (row (e)) further improves the mIoU to 26.24 on SemanticKITTI, highlighting that RLT loss works synergistically with fine-tuning.

3) Probability Decoder Branch (Pro.D.): Introducing the probability decoder (Pro.D., row (b)) improves mIoU from 27.75 to 30.14 on SemanticPOSS. Combined with RLT loss (row (c)), it further increases to 30.30, showing their complementarity. The full model (row (g)) achieves the best performance (26.89 on SemanticKITTI, 37.60 on SemanticPOSS), demonstrating the synergy of all components under adversarial settings. Fig. 5 visualizes the cross-domain adaptation results under adversarial source domain, highlighting how the AAF framework restores the semantic structure. Recovery in head classes (e.g., building', plants') benefits from fine-tuning on clean data, while region-level consistency is mainly attributed to the HNPU strategy.

E. Parameter Sensitive Analysis

As illustrated in Fig. 6, the performance of our AAF framework is notably affected by the confidence threshold τ and the number of neighboring points k. A higher confidence threshold ($\tau = 0.95$) yields the best results, with the highest mIoU of 36.45 observed when $k = 6$. Conversely, lower thresholds ($\tau = 0.85$ and 0.9) can include more points but introduce more significant uncertainty, leading to less stable outcomes. The number of nearest neighbors k also plays a critical role in performance. A $k = 6$ value strikes an optimal balance between capturing local context and minimizing noise. Smaller values ($k = 4$) provide limited contextual information, while larger values ($k = 10$) can introduce excessive noise, particularly under adversarial conditions.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

VI. Conclusions

In this paper, we explore the robustness of 3D point cloud semantic segmentation frameworks under source-domain adversarial attacks in UDA settings, which is a critical yet underexplored challenge in safety-sensitive applications like autonomous driving. To our knowledge, this is the first study to benchmark and defend 3D UDA models under such threats. We propose a novel point cloud adversarial sample generation method to contaminate the dataset, resulting in the new AdvSynLiDAR dataset. Further, we introduce the AAF framework to address the performance degradation by introducing the Robust Long-Tailed loss (RLT loss) and designing a decoder-branch-based approach. Experiments show that our proposed AAF framework enhances the model's robustness and ensures effective semantic segmentation in cross-domain tasks.

References

[1] M. Spencer, R. Sawtell, and S. Kitchen, "Sphere-graph: A compact 3d topological map for robotic navigation and segmentation of complex environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2567–2574, 2024.

[2] Y. Sun, W. Zuo, H. Huang, P. Cai, and M. Liu, "Pointmoseg: Sparse tensor-based end-to-end moving-obstacle segmentation in 3-d lidar point clouds for autonomous driving," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 510–517, 2021.

[3] J. Zhang, L. Chen, B. Liu, B. Ouyang, Q. Xie, J. Zhu, W. Li, and Y. Meng, "3d adversarial attacks beyond point cloud," *Information Sciences*, vol. 633, pp. 491–503, 2023.

[4] Z. Shi, Z. Chen, Z. Xu, W. Yang, Z. Yu, and L. Huang, "Shape prior guided attack: Sparser perturbations on 3d point clouds," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 8, 2022, pp. 8277–8285.

[5] A. Xiao, J. Huang, D. Guan, F. Zhan, and S. Lu, "Transfer learning from synthetic to real lidar point cloud for semantic segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 3, 2022, pp. 2795–2803.

[6] A. Madry, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[7] M. Li, Y.-M. Cheung, and Z. Hu, "Key point sensitive loss for long-tailed visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4812–4825, 2022.

[8] L. Pan, X. Chen, Z. Cai, J. Zhang, H. Zhao, S. Yi, and Z. Liu, "Variational relational point completion network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8524–8533.

[9] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9297–9307.

[10] Y. Pan, B. Gao, J. Mei, S. Geng, C. Li, and H. Zhao, "Semanticposs: A point cloud dataset with large quantity of dynamic instances," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 687–693.

[11] D. Tsai, J. S. Berrio, M. Shan, S. Worrall, and E. Nebot, "See eye to eye: A lidar-agnostic 3d detection framework for unsupervised multi-target domain adaptation," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7904–7911, 2022.

[12] M. K. Wozniak, M. Hansson, M. Thiel, and P. Jensfelt, "Uada3d: Unsupervised adversarial domain adaptation for 3d object detection with sparse lidar and large domain gaps," *IEEE Robotics and Automation Letters*, vol. 9, no. 12, pp. 11 210–11 217, 2024.

[13] T. Spadotto, M. Toldo, U. Michieli, and P. Zanuttigh, "Unsupervised domain adaptation with multiple domain discriminators and adaptive self-training," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 2845–2852.

[14] Y. Yang and S. Soatto, "Fda: Fourier domain adaptation for semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4085–4095.

[15] W. Zhu, J.-L. Yin, B.-H. Chen, and X. Liu, "Srouda: meta self-training for robust unsupervised domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 3, 2023, pp. 3852–3860.

[16] M. Rochan, S. Aich, E. R. Corral-Soto, A. Nabatchian, and B. Liu, "Unsupervised domain adaptation in lidar semantic segmentation with self-supervision and gated adapters," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2649–2655.

[17] A. H. Gebrehiwot, D. Hurych, K. Zimmermann, P. Pérez, and T. Svoboda, "T-uda: Temporal unsupervised domain adaptation in sequential point clouds," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7643–7650.

[18] S. Wang, Y. Tong, X. Shang, and Z. Zhang, "Multi-confidence guided source-free domain adaption method for point cloud primitive segmentation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 737–743.

[19] J. Kim, B.-S. Hua, T. Nguyen, and S.-K. Yeung, "Minimal adversarial examples for deep learning on 3d point clouds," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7797–7806.

[20] K. Tang, Y. Shi, J. Wu, W. Peng, A. Khan, P. Zhu, and Z. Gu, "Normalattack: Curvature-aware shape deformation along normals for imperceptible point cloud attack," *Security and Communication Networks*, vol. 2022, no. 1, p. 1186633, 2022.

[21] H. Naderi, C. Dinesh, I. V. Bajic, and S. Kasaei, "Model-free prediction of adversarial drop points in 3d point clouds," *CoRR*, 2022.

[22] H. Liu, J. Jia, and N. Z. Gong, "Pointguard: Provably robust 3d point cloud classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6186–6195.

[23] K. Li, Z. Zhang, C. Zhong, and G. Wang, "Robust structured declarative classifiers for 3d point clouds: Defending adversarial attacks with implicit gradients," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 294–15 304.

[24] Z. Gao, K. Huang, R. Zhang, D. Liu, and J. Ma, "Towards better robustness against common corruptions for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 18 882–18 893.

[25] S. Ye, D. Chen, S. Han, and J. Liao, "Learning with noisy labels for robust point cloud segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6443–6452.

[26] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*. Ieee, 2016, pp. 565–571.

[27] P. I. Frazier, "A tutorial on bayesian optimization," *arXiv preprint arXiv:1807.02811*, 2018.

[28] X. Wu, U. Jang, J. Chen, L. Chen, and S. Jha, "Reinforcing adversarial robustness using model confidence induced by adversarial training," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80, 10–15 Jul 2018, pp. 5334–5342.

[29] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[30] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2446–2454.

[31] C. Saltori, F. Galasso, G. Fiameni, N. Sebe, F. Poiesi, and E. Ricci, "Compositional semantic mix for domain adaptation in point cloud segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[32] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[33] A. Agarwal, N. Ratha, R. Singh, and M. Vatsa, "Robustness against gradient based attacks through cost effective network fine-tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2023, pp. 28–37.