ADAPTIVE AUGMENTATION-AWARE LATENT LEARNING FOR ROBUST LIDAR SEMANTIC SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Adverse weather conditions significantly degrade the performance of LiDAR point cloud semantic segmentation networks by introducing large distribution shifts. Existing augmentation-based methods attempt to enhance robustness by simulating weather interference during training. However, they struggle to fully exploit the potential of augmentations due to the trade-off between minor and aggressive augmentations. To address this, we propose A3Point, an adaptive augmentation-aware latent learning framework that effectively utilizes a diverse range of augmentations while mitigating the semantic shift, which refers to the change in the semantic meaning caused by augmentations. A3Point consists of two key components: semantic confusion prior (SCP) latent learning, which captures the model's inherent semantic confusion information, and semantic shift region (SSR) localization, which decouples semantic confusion and semantic shift, enabling adaptive optimization strategies for different disturbance levels. Extensive experiments on multiple standard generalized LiDAR segmentation benchmarks under adverse weather demonstrate the effectiveness of our method, setting new state-of-the-art results. The code will be released.

1 INTRODUCTION

LiDAR semantic segmentation is vital for 3D vision tasks such as autonomous driving (Li & Ibanez-Guzman, 2020; Li et al., 2020; Aksoy et al., 2020; Zhao et al., 2023). However, existing methods (Ando et al., 2023; Choy et al., 2019; Lai et al., 2023; Puy et al., 2023) often struggle in adverse weather (e.g., fog, snow, and rain) due to severe distribution shifts in point clouds, causing a mismatch between training and testing data. Since most outdoor scene point cloud datasets (Behley et al., 2019; Fong et al., 2022; Xiao et al., 2022b) are collected in normal weather, developing a robust network that generalizes across diverse conditions is increasingly crucial. Addressing this challenge is essential for achieving reliable, weather-invariant LiDAR semantic segmentation.

To mitigate performance degradation, existing studies (Xiao et al., 2023; Park et al., 2024; Kong et al., 2023b) enhance network robustness by simulating adverse weather during training via simulation-based or augmentation-based approaches. Simulation-based methods (Bijelic et al., 2018; Hahner et al., 2022; 2021) model physical equations to replicate weather effects on point clouds but require separate modeling for each condition, making it impractical to cover all variations. Augmentation-based methods (Xiao et al., 2023; Park et al., 2024; Kim et al., 2023) introduce geometric perturbations and point drop to mimic weather-induced distortions more flexibly. However, they remain underutilized for two reasons (Fig. 1): (1) mild augmentations fail to generalize to severe conditions, while (2) excessive augmentations distort point cloud distribution, causing semantic shift (Wang et al., 2021; Bai et al., 2022; Yuan et al., 2021), where augmented regions no longer align with original semantics, thereby hindering training. This dilemma leads existing methods to restrict augmentation range and magnitude, limiting their potential. *How to utilize a larger augmentation space while mitigating semantic shift remains a compelling challenge.*

This analysis motivates us to explore a broader, more aggressive augmentation space to better simulate weather-induced distortions at varying intensities. However, ensuring its effectiveness requires addressing the potential semantic shift in augmented point clouds. Directly modeling semantic shift is difficult, as prediction errors arise from two factors: (1) **semantic confusion** (Fig.2 (a)), which is the network's inherent property that struggles to distinguish similar classes (e.g., *road* vs. *sidewalk*)

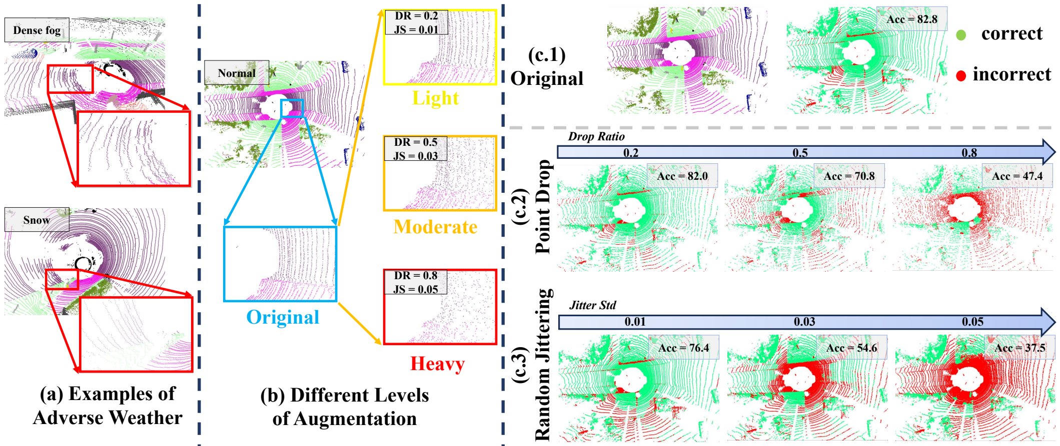


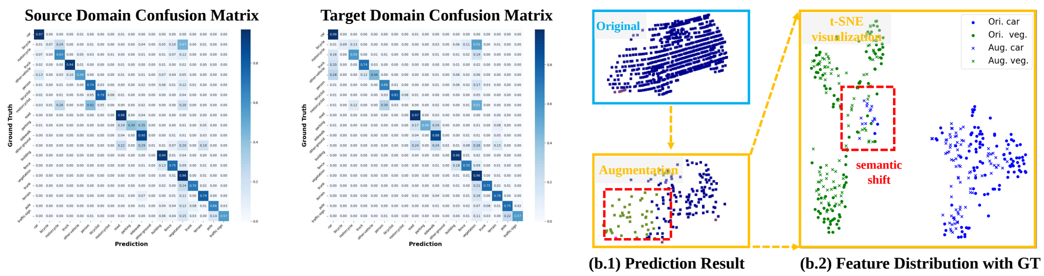
Figure 1: (a) Point cloud distortions caused by adverse weather conditions. (b) Augmentation at different levels (light, moderate, and heavy), where we adjust the drop ratio (DR) for point drop and jitter std (JS) for random jittering. (c) Visualization of segmentation accuracy under different augmentation levels, showing that aggressive distortions lead to significant performance degradation.

despite correct labels; (2) **semantic shift** (Fig.2 (b)), where excessive augmentation distorts point cloud distributions and the original labels fail to describe the corresponding regions. For semantic confusion, original labels should be preserved to enhance the network's discriminative ability. In contrast, semantic shift requires adapting supervision to avoid misleading signals. Thus, the key to solving semantic shift lies in disentangling these two factors in augmented point clouds.

In this paper, we model semantic shift in augmented point clouds to enable their effective use during training. Observing that semantic confusion exists in both raw and augmented data and is consistent across domains, while semantic shift occurs only in augmented data, we propose a two-step strategy to identify it: (1) Mining semantic confusion priors from normal-condition predictions. Inspired by VQVAE (Van Den Oord et al., 2017), we frame this as a **discrete latent representation learning task**: class-specific local confusion patterns are encoded into a latent space and represented by quantized latent variables, with representational capacity enforced via reconstruction. (2) Detecting semantic shift as anomalies in augmented point clouds. We apply the learned latent encoder to augmented predictions, formulating semantic shift localization as an **anomaly detection problem**: by comparing augmented latent representations with the learned priors, we distinguish semantic consistency regions (affected only by semantic confusion) from semantic shift regions (additionally affected by semantic shift). This separation enables targeted optimization during training.

Based on the above discussion, we propose **A**daptive **A**ugmentation-**A**ware latent learning for robust LiDAR semantic segmentation in adverse conditions, namely A3Point, which involves two key components: semantic confusion prior (SCP) latent learning and semantic shift region (SSR) localization, to fully explore the potential of a large and diverse augmentation space for robust point cloud segmentation. **To capture the network's inherent semantic confusion**, in the SCP latent learning module, we perform class-wise latent encoding on predictions from original point clouds, using quantized latent variables to represent local confusion patterns. Through vector quantization and reconstruction constraints, this process learns meaningful, representative embeddings capable of reconstructing prediction maps. **To disentangle semantic confusion from semantic shift**, in the SSR localization module, we dynamically track the representation distribution of each quantized latent variable and use a frozen prior latent encoder to implicitly represent augmented predictions. By treating semantic shift detection as an anomaly detection problem, we adaptively distinguish semantic consistency regions (SCR) from semantic shift regions (SSR). In SCR, original labels remain effective for optimization, while in SSR, we apply knowledge distillation, selecting the global nearest quantized latent variable as a supervisory constraint. By jointly localizing SSR and adapting optimization strategies accordingly, A3Point fully harnesses diverse augmentations to improve segmentation robustness under varying disturbance levels.

Our contributions are as follows: 1) We introduce a novel perspective to overcome limitations of augmentation-based approaches, enabling the effective utilization of a large and diverse augmen-



(a) Semantic Confusion **(b) Semantic Shift**

Figure 2: Demonstration of **semantic confusion** and **semantic shift**. (a) Confusion matrices from source (normal weather) and target (adverse weather) domains. Despite domain shifts, semantic confusion remains consistent. (b) Aggressive augmentations alter point cloud density and shape, leading to semantic misalignment (e.g., *car* $\rightarrow$ *veg.*).

tation space to improve LiDAR segmentation robustness under adverse weather. 2) We propose a two-step framework to decouple semantic confusion and mitigate semantic shift, comprising semantic confusion prior (SCP) latent learning and semantic shift region (SSR) localization. 3) We validate our approach through extensive experiments on multiple standard generalized LiDAR segmentation benchmarks under adverse weather, achieving new state-of-the-art results.

2 Related Work

2.1 3D Point Cloud Semantic Segmentation

Semantic segmentation of 3D point clouds assigns a label to each point and is typically approached via three paradigms: point-based, projection-based, and voxel-based. Point-based methods directly take 3D points as input. PointNet (Qi et al., 2017) was the first to introduce this approach, utilizing multi-layer perceptrons to extract per-point features. Subsequent works (Thomas et al., 2019; Zhao et al., 2021; Choe et al., 2022; Fan et al., 2021) further advanced this paradigm. While these methods minimize information loss and achieve strong performance, they demand high computational resources when applied to large-scale LiDAR data. Projection-based methods (Ando et al., 2023; Kong et al., 2023a; Milioto et al., 2019; Xiao et al., 2021; Zhang et al., 2020) map point clouds to 2D range-view images via spherical projection, enabling the use of 2D segmentation networks. These methods are computationally efficient but suffer from information loss during projection, leading to slightly lower accuracy. Voxel-based methods (Choy et al., 2019; Lai et al., 2023; Zhu et al., 2021; Graham et al., 2018; Tang et al., 2020) divide 3D point clouds into sparse voxel grids and aggregate points within the same voxel. The introduction of sparse convolutions significantly reduces computational costs, making this paradigm well-suited for large-scale outdoor LiDAR scenes. In this work, we adopt voxel-based architectures (Choy et al., 2019; Tang et al., 2020) as our baseline to balance inference efficiency and segmentation performance.

2.2 LiDAR under Adverse Weather Conditions

In real-world applications, robust scene understanding under adverse weather is critical, especially given the safety demands of autonomous driving (Kong et al., 2023b; Xiao et al., 2023; Sakaridis et al., 2021). However, extreme weather can significantly disturb point cloud distributions (Filgueira et al., 2017; Heinzler et al., 2019; Peynot et al., 2009; Ryde & Hillier, 2009), leading to a dramatic drop in the performance of LiDAR segmentation networks. To mitigate domain discrepancy between normal and adverse weather conditions, unsupervised domain adaptation (UDA) approaches (Hahner et al., 2022; 2021; Luo et al., 2021; Xiao et al., 2022a;b; Yang et al., 2021) have been explored. These methods leverage labeled source domain data and unlabeled target domain data to learn domain-invariant features, improving cross-domain performance. However, UDA-based methods are limited to specific and visible target domains, making them insufficient for generalizing to unknown weather disturbances. In this paper, we adopt the domain generalization setting (Kim et al., 2024; 2023; Li et al., 2023), aiming to train a model using a single source domain under normal weather conditions, without access to target data from adverse weather during training.

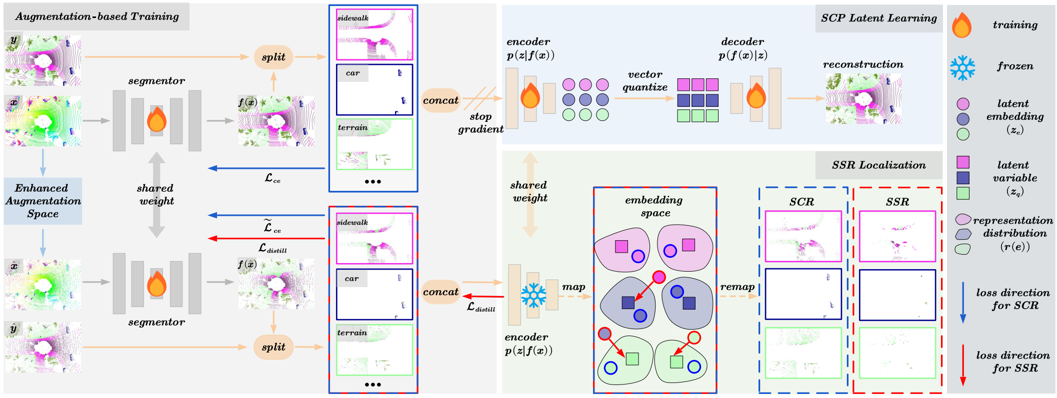


Figure 3: Pipeline of A3Point. We explore an abundant augmentation space (Sec.3.4) and propose two key components: SCP latent learning to capture inherent semantic confusion (Sec.3.5) and SSR localization to decouple semantic shift (Sec.3.6).

2.3 Augmentation for Robust LiDAR Segmentation

To enhance the robustness of LiDAR segmentation models, existing methods introduce point cloud corruptions during training to simulate the interference caused by adverse weather. Simulation-based methods (Bijelic et al., 2018; Hahner et al., 2022; 2021) rely on prior weather knowledge to construct physical models that artificially simulate point clouds under adverse conditions. Rather than explicitly modeling specific weather effects, augmentation-based methods (Xiao et al., 2023; Park et al., 2024; Kim et al., 2023) provide a more general and flexible approach. Inspired by 2D image augmentation, Mix-based methods (Kong et al., 2023c; Nekrasov et al., 2021; Xiao et al., 2022a; Zhao et al., 2024) blend two LiDAR scans to enhance training diversity. To better simulate disturbances caused by adverse weather, recent works (Park et al., 2024; Xiao et al., 2023) identify two primary degradation patterns: (1) geometric perturbation and (2) point drop. PointDR (Xiao et al., 2023) introduces a set of augmentations to randomly simulate weather-induced disturbances, while LiDARWeather (Park et al., 2024) proposes a learnable Point Drop strategy for adaptive augmentation. However, these methods remain constrained by a limited perturbation space. In contrast, our approach explores a broader range of perturbations and explicitly addresses the semantic shift problem caused by aggressive augmentation, enabling more robust network training.

3 Method

3.1 Problem Definition

In domain generalization (DG) for LiDAR semantic segmentation, the network is trained on labeled source domain data and need to be generalized to unseen target domain data. To be specific, the source domain can be denoted as $D_s = \{(x_i^S, y_i^S)\}_{i=1}^{N_S}$, where $x_i^S \in X_S$ represents a LiDAR point cloud scan with $y_i^S \in Y_S$ as the corresponding point-wise one-hot label covering C classes. The target domain can be denoted as $D_t = \{(x_i^T)\}_{i=1}^{N_T}$, where target label Y_T shares the same label space with Y_S. Since the target domain is not accessible during the training process, we omit the superscript S/T in the following notation for brevity.

3.2 Preliminaries

Augmentation-based Training for DG. Existing methods explore weather-induced disturbances and apply them as data augmentation. This approach can be viewed as a domain randomization paradigm for learning a domain-generalizable network. During training, the loss is first computed on the original scan to train a neural network f:

$$\mathcal{L}_{ce} = \frac{1}{N_S} \sum_{i=1}^{N_S} \frac{1}{n_i} \sum_{j=1}^{n_i} \ell_{ce}(f(x_{ij}), y_{ij}), \tag{1}$$

where ℓ_{ce} denotes the voxel-wise cross-entropy loss, and n_i is the number of valid voxels in x_i. Then, for each x_i, random augmentations are simultaneously applied to (x_i, y_i), obtaining $(\hat{x}_i, \hat{y}_i) = \mathcal{A}\{(x_i, y_i)\}$. The augmented training pair is also implemented through the cross-entropy loss:

$$\hat{\mathcal{L}}_{ce} = \frac{1}{N_S} \sum_{i=1}^{N_S} \frac{1}{\hat{n}_i} \sum_{j=1}^{\hat{n}i} \ell_{ce}(f(\hat{x}_{ij}), \hat{y}_{ij}). \tag{2}$$

The total training loss can be represented as: $\mathcal{L} = \mathcal{L}_{ce} + \hat{\mathcal{L}}_{ce}$.

Vector Quantized Variational AutoEncoder. VQ-VAE (Van Den Oord et al., 2017) is a variant of variational autoencoder (Kingma et al., 2013) that learns a discrete latent representation. It consists of an encoder $\mathbb{E}$, a decoder $\mathbb{D}$, and a codebook $\mathcal{C} = \{e_1, e_2, ..., e_K\}$ containing K learnable embeddings. The encoder maps the input x to a continuous latent representation $z_e = \mathbb{E}(x)$, which is then quantized to the nearest codebook entry e_k using a nearest-neighbor lookup (we use a single random variable z to represent the discrete latent variables for simplicity):

$$z_q = \text{quantize}(z_e) = e_k, \text{where } k = \arg\min_j ||z_e - e_j||_2. \tag{3}$$

The decoder reconstructs the input from the quantized latent representation: $\bar{x} = \mathbb{D}(z_q)$. Total training objective is:

$$\mathcal{L} = ||x - \bar{x}||_2^2 + ||\text{sg}(z_e) - z_q||_2^2 + ||z_e - \text{sg}(z_q)||_2^2, \tag{4}$$

which consists of reconstruction loss, codebook loss, and commitment loss, where $\text{sg}(\cdot)$ denotes the stop-gradient operation. The codebook loss brings the selected latent variables e close to encoder outputs, while the commitment loss encourages the encoder to produce latent representations close to the codebook entries. The discrete latent representation learned by VQ-VAE can capture the underlying structure of the data and has been successfully applied in various tasks, such as image generation (Razavi et al., 2019; De Fauw et al., 2019; Yu et al., 2021) and unsupervised representation learning (Liu et al., 2023; Chen et al., 2024; Takida et al., 2023).

3.3 Overview of A3Point Framework

We first define the enhanced augmentation space used during training (Sec.3.4). Then, we model a discrete latent representation learning process to learn semantic confusion prior (Sec.3.5). Next, we localize the semantic shift regions through a form of anomaly detection (Sec.3.6). Finally, we introduce region-specific optimization strategies (Sec.3.7). Our overall framework is shown in Fig.3.

3.4 Enhanced Augmentation Space

Previous works (Xiao et al., 2023; Park et al., 2024) identify that the main disturbances caused by adverse weather can be summarized as (1) geometric perturbation, caused by perceived distance shifts, and (2) point drop, resulting from beam attenuation, beam missing, or potential occlusions. This motivates us to use random jittering and point drop as the primary and generic augmentation strategies. Unlike previous methods that only adopt limited range and magnitude, we define a broader augmentation space to fully simulate various levels of weather disturbances. Specifically, we define jitter std in a range of $[j_{min}, j_{max}]$ for random jittering and drop ratio range of $[d_{min}, d_{max}]$ for point drop, and uniformly sample the perturbation magnitudes during training. For other subsidiary augmentations, we follow previous works and employ random rotation, random scaling, random flipping, random noise perturbation, scan mix (Kong et al., 2023c; Xiao et al., 2022a).

3.5 Semantic Confusion Prior Latent Learning

Semantic confusion, which refers to the network's inherent uncertainty in distinguishing between classes, as reflected in the predicted probability distribution. To decouple semantic confusion and semantic shift in the augmented point cloud, we mine prior knowledge of semantic confusion from the original point cloud predictions.

Specifically, we first introduce an autoencoder for prior latent learning, which follows VQ-VAE (Van Den Oord et al., 2017). Through quantized latent variables and a reconstruction process, it can effectively learn meaningful representations with sufficient representational power. The input for encoder

$\mathbb{E}$ is obtained by first concatenating $f(x)$ (processed through softmax) and x, then splitting the result class-wise according to the label y to prevent inter-class interactions, and finally concatenating the resulting submatrices along the batch dimension. This can be expressed as:

$$z_e = \mathbb{E}(\underbrace{[f(x^1) \oplus x^1, f(x^2) \oplus x^2, ..., f(x^C) \oplus x^C]}_{n \times (C+3)}), \tag{5}$$

where $f(x^i) \oplus x^i$ represents the concatenation of prediction and coordinates for points with class i, and n is the number of valid voxels. The $[\cdot, \cdot]$ notation denotes the concatenation operation along the batch dimension. The output for decoder $\mathbb{D}$ is reconstructed to $[f(x^1); f(x^2); ...; f(x^C)]$. Eq.4 is used to optimize for this process without backpropagating the gradients to f.

Compared to the $K \times D$ codebook in VQ-VAE, we use class-specific sub-codebooks of size $C \times k \times D$, where k is the number of latent variables in each sub-codebook, and D is the dimension of the latent variables. Each latent variable in the sub-codebook can be considered as modeling a specific local semantic distribution pattern under corresponding class. The encoder $\mathbb{E}$ models the $p(z|f(x))$, which represents the prior distribution of the latent variables z given the predicted probabilities $f(x)$. The decoder $\mathbb{D}$ models the $p(f(x)|z)$, which represents the posterior distribution of reconstructing the predicted probabilities $f(x)$ conditioned on the latent variables z. Through this process, the autoencoder can model the network's semantic confusion online during the training process.

3.6 Semantic Shift Region Localization

Excessive augmentation can cause semantic shift in certain regions, leading to abnormal prediction results, as shown in Fig. 2 (b). After modeling the semantic confusion prior, we can assess whether the network's predictions for the augmented point cloud conform to the normal semantic distribution patterns, thus treating the localization of semantic shift regions as an anomaly detection problem.

During latent learning process, we dynamically track the representation distribution of each quantized latent variable. For each e_i, we maintain statistics of its corresponding latent embedding distribution before quantization and store variance σ_i^2 of each channel, which is updated using exponential moving average. The representation distribution of each e_i can be expressed as:

$$r(e_i) = \mathcal{N}(e_i, \text{diag}(\sigma_{i,1}^2, \sigma_{i,2}^2, \ldots, \sigma_{i,D}^2)), \tag{6}$$

Next, we use the frozen prior latent encoder $\mathbb{E}$ to map predictions of augmented point cloud to latent embedding space. Since the encoder models the class-wise $p(z|f(x))$, embeddings from regions affected by semantic shift will not fall into the representation distribution of their corresponding sub-codebooks. We locate the semantic shift regions by checking whether the embeddings lie within the representation distribution of their nearest latent variable in corresponding sub-codebooks.

After distinguishing all the embeddings, we remap them back to the prediction space, i.e., $f(\hat{x})$, to determine the semantic consistency regions (SCR) and semantic shift regions (SSR) in the prediction space. We use two masks to represent these two regions:

$$M_{SCR} = \mathbb{1}(z_e \in r(NN_{sub}(z_e))), \tag{7}$$

$$M_{SSR} = \mathbb{1}(z_e \notin r(NN_{sub}(z_e))), \tag{8}$$

where z_e represents the latent embeddings of the augmented point cloud predictions, $NN_{sub}(z_e)$ denotes the nearest latent variable of z_e, and $\mathbb{1}(\cdot)$ is the indicator function.

3.7 Optimization Strategies

After determining the SCR and SSR, we assign different optimization strategies. For SCR, we use the original labels:

$$\widetilde{\mathcal{L}}_{ce} = \ell_{ce}(f(\hat{x}) \odot M_{SCR}, y \odot M_{SCR}) = \hat{\mathcal{L}}_{ce} \odot M_{SCR}, \tag{9}$$

where $\odot$ denotes element-wise multiplication. For SSR, we propose a latent variable-based distillation loss to provide appropriate supervisory signals. Instead of querying the nearest neighbor from sub-codebook of the corresponding class, we obtain the closest semantic confusion pattern prior for this region by querying from global codebook. Then, we use this prior to distill the latent embeddings of SSR:

$$\mathcal{L}_{distill} = ||z_e - sg[NN_{global}(z_e)]||_2^2, \tag{10}$$

Table 1: Comparison results of s [A] → [C]. $*$ denotes the reproduced result with the same backbone.

Methods	car	bi.cle	mt.cle	truck	oth-v.	pers.	bi.clst	mt.clst	road	parki.	sidew.	other-g.	build.	fence	veget.	trunk	terr.	pole	traf.	D-fog	L-fog	Rain	Snow	mIoU	gain
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	51.9	54.6	57.9	53.7	54.7	-
Baseline	67.1	**5.0**	28.1	38.5	14.6	45.8	8.3	13.8	40.1	16.1	26.1	3.3	71.6	52.7	53.8	33.9	39.2	25.3	12.7	30.7	30.1	29.7	25.3	31.4	+0.0
PointDR* (Xiao et al., 2023)	69.2	1.0	8.9	**41.9**	7.6	48.9	17.0	36.2	57.8	15.9	32.3	4.0	75.7	46.4	54.0	36.2	43.9	23.7	24.2	37.3	33.5	35.5	26.9	33.9	+2.5
DGUIL (Kim et al., 2023)	78.2	2.5	33.0	29.7	6.1	49.8	0.8	40.9	67.3	7.2	38.0	2.2	**79.8**	**54.4**	**64.1**	36.8	52.3	31.0	40.0	36.3	34.5	35.5	33.3	37.6	+6.2
WADG (Du et al., 2024)	72.0	0.0	32.9	37.0	1.9	37.7	6.8	52.9	59.9	10.7	31.8	2.2	76.0	48.8	62.7	34.0	49.3	23.6	20.4	39.5	32.5	31.7	29.4	34.8	+3.4
DGLSS (He et al., 2024)	69.6	0.8	42.8	34.4	8.9	41.9	12.8	44.5	52.0	14.5	30.8	**6.0**	77.8	51.1	57.6	38.9	43.2	29.7	30.6	34.2	34.8	36.2	32.1	36.2	+4.8
LiDARWeather (Park et al., 2024)	86.1	4.8	13.8	39.7	**26.6**	**55.4**	8.5	50.4	63.7	14.9	37.9	5.5	75.2	52.7	60.4	39.7	44.9	30.1	40.8	36.0	37.5	37.6	33.1	39.5	+8.1
NTN (Park et al., 2025)	83.3	3.7	31.3	36.2	18.2	53.3	6.8	**55.9**	67.2	**18.1**	37.2	5.4	72.1	41.8	58.0	36.0	46.0	28.2	39.8	35.3	35.1	35.7	32.4	38.9	+7.5
A3Point (ours)	**88.3**	4.1	**57.5**	29.0	7.6	45.3	**24.0**	46.4	**69.2**	16.9	**38.4**	3.3	74.8	48.2	63.1	**42.9**	**49.2**	**32.6**	**41.8**	**41.1**	**38.5**	**38.2**	**37.2**	**41.3**	+9.9

Table 2: Comparison results of [B] → [C]. $*$ denotes the reproduced result with the same backbone.

Methods	car	bi.cle	mt.cle	truck	oth-v.	pers.	bi.clst	mt.clst	road	parki.	sidew.	other-g.	build.	fence	veget.	trunk	terr.	pole	traf.	D-fog	L-fog	Rain	Snow	mIoU	gain
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	51.9	54.6	57.9	53.7	54.7	-
Baseline	33.8	1.7	3.3	15.5	0.2	25.5	1.6	3.4	15.3	9.2	16.8	0.1	33.4	21.9	39.5	18.7	**44.0**	8.8	0.8	15.2	16.0	16.8	12.8	15.5	+0.0
PointDR* (Xiao et al., 2023)	41.1	2.8	3.4	18.1	0.2	31.3	**2.8**	3.3	34.4	**10.2**	19.7	1.0	52.7	22.0	48.5	21.3	38.3	19.2	5.6	19.1	20.3	25.3	19.0	19.8	+4.3
WADG (Du et al., 2024)	33.8	1.1	2.9	17.0	0.2	26.8	1.0	4.3	53.9	5.0	20.6	2.2	**64.3**	27.1	53.8	27.0	37.0	**28.6**	8.6	21.6	23.4	27.2	21.4	21.9	+6.4
LiDARWeather (Park et al., 2024)	39.3	2.9	0.9	19.4	0.8	27.7	2.2	3.8	42.5	9.4	21.6	0.3	51.9	33.5	47.4	23.1	33.3	23.2	6.8	19.0	21.2	23.1	17.3	20.5	+5.0
NTN (Park et al., 2025)	48.4	1.5	2.4	19.4	0.2	29.1	3.2	8.9	43.5	6.7	20.5	0.0	52.2	30.1	49.8	20.0	32.9	24.7	7.5	-	-	-	-	21.1	+5.6
A3Point (ours)	**76.7**	**4.0**	**5.0**	**29.6**	**1.3**	**35.1**	1.7	**9.5**	**55.4**	3.9	**24.0**	**3.5**	61.7	**34.5**	**60.1**	**34.1**	33.3	28.1	**14.8**	**26.8**	**26.6**	**31.9**	**28.6**	**27.2**	+11.7

where $NN_{global}(z_e)$ denotes the global nearest neighbor latent variable. Eq.10 is similar to previous commitment loss while we do not update the encoder $\mathbb{E}$, but backpropagate the gradient to the network f. Our total training loss is:

$$\mathcal{L}_{total} = \mathcal{L}_{ce} + \widetilde{\mathcal{L}}_{ce} + \lambda \mathcal{L}_{distill}, \tag{11}$$

where λ is a hyperparameter balancing loss terms. In this way, the semantic confusion prior learned from the original point cloud can provide meaningful supervisory signals for semantic shift regions, enhancing the model's performance on augmented data while preserving semantic consistency. For further implementation details, discussions and algorithm flow, please refer to Appendix B-E.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We use four datasets: SemanticKITTI (Behley et al., 2019), SynLiDAR (Xiao et al., 2022b), SemanticKITTI-C (Kong et al., 2023b), and SemanticSTF (Xiao et al., 2023). ❶ SemanticKITTI: 19,130 training scans (sequences 00-10, except 08), collected in urban environments under standard weather conditions. ❷ SynLiDAR: 198,396 synthetic scans (19 billion points) generated using Unreal Engine 4. ❸ SemanticKITTI-C: Corrupted version of SemanticKITTI generated via simulation. ❹ SemanticSTF: 2,076 LiDAR scans from STF (Bijelic et al., 2020) under adverse weather (snow, dense fog, light fog, rain), split into 1,326 training, 250 validation, and 500 testing scans. SemanticKITTI and SynLiDAR serve as source domains, while SemanticKITTI-C and SemanticSTF assess robustness as target domains. We denote SemanticKITTI as [A], SynLiDAR as [B], SemanticSTF as [C], and SemanticKITTI-C as [D] for brevity.

Evaluation Metrics. We adopt MinkowskiNet-18/32width (Choy et al., 2019) as the base model and also evaluate SPVCNN (Tang et al., 2020). Performance is measured by Intersection over Union (IoU) per class and mean IoU (mIoU) across classes, including breakdowns by weather conditions.

Implementation Details. The segmentation network is trained with SGD (learning rate 0.24, weight decay 0.0001), and the autoencoder with Adam (learning rate 0.001). The sub-codebook size k is set to 32. These networks are updated alternately. For augmentation, we set the drop ratio to [0.2, 0.8] and jitter standard deviation to [0.01, 0.05]. Training runs for 50 epochs with a batch size of 4 on an RTX 3090 (24 GB). The balancing coefficient λ is set to 0.1 to maintain gradient stability.

4.2 COMPARISON WITH EXISTING METHODS

Overall Quantiative Results. Tab. 1-2 compare A3Point with state-of-the-art domain generalization methods on the $[A] \rightarrow [C]$ and $[B] \rightarrow [C]$ benchmarks. The baseline is trained only with $\mathcal{L}_{ce}$. A3Point outperforms all methods, achieving 9.9% and 11.7% mIoU improvements over the baseline.

Table 3: Different architectures & benchmarks.

Method	SPVCNN		Minkowski	
	baseline	w/ A3Point	baseline	w/ A3Point
$[A] \to [C]$	28.1	40.1 (**+12.0**)	31.4	41.3 (**+9.9**)
$[B] \to [C]$	17.3	25.8 (**+8.5**)	15.5	27.2 (**+11.7**)
$[A] \to [D]$	52.5	54.7 (**+2.2**)	53.0	58.9 (**+5.9**)

Table 4: Ablation study of A3Point components.

None	EAS ($\hat{\mathcal{L}}_{ce}$)	SCR ($\widetilde{\mathcal{L}}_{ce}$)	SSR ($\mathcal{L}_{distll}$)	$[A] \to [C]$	$[B] \to [C]$
✓				31.4	15.5
	✓			38.7	24.9
	✓	✓		40.2	26.5
	✓	✓	✓	**41.3**	**27.2**

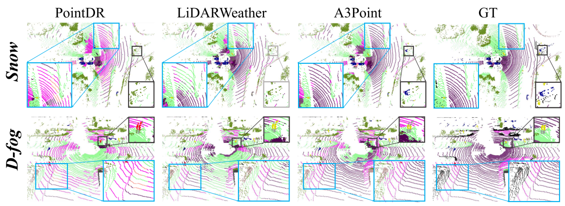


Figure 4: Qualitative results on $[A] \to [C]$. Significant improvements are marked with boxes.

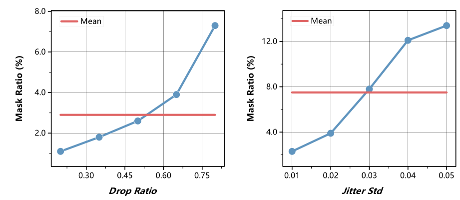


Figure 5: Mask ratio of SSR under different augmentation levels.

Class-wise analysis shows that A3Point performs particularly well on safety-critical classes such as *motorcycle*, *bicyclist*, *traffic sign*, and *car*, which are typically challenging due to unique geometries and susceptibility to adverse conditions. The local pattern encoding in VQ-VAE and region-specific optimization enhance robust feature capture for these classes under diverse weather conditions.

Weather-level Comparison. A3Point demonstrates superior robustness across all weather conditions, with substantial leads even in the most severe cases like dense fog and heavy snow. The extensive augmentation space and adaptive latent-space distillation enable A3Point to handle a wide spectrum of weather disturbances while preserving predictions in less corrupted regions, resulting in better performance than other methods in milder conditions.

Qualitative Results. Fig.5 presents a visual comparison of A3Point with previous methods under challenging weather conditions like snow and dense fog. A3Point shows a superior ability to accurately segment major scene components such as *sidewalk*, *road*, and *terrain*, which are often obscured or distorted in adverse weather. Moreover, it exhibits better segmentation of complex instances like *car* and *traffic sign*, which are critical for safe navigation. The local pattern encoding helps capture the unique geometries of these objects, while the decoupling of semantic confusion and shift enhances the discriminative performance of these classes.

4.3 Ablation Study

See Appendix F-K for more analyses and visualizations.

Additional Results. To validate the effectiveness and generalizability of A3Point, we conduct experiments across different architectures and benchmarks (Tab. 3). Results show consistent gains. With SPVCNN, A3Point achieves +12.0%, +8.5%, and +2.2% mIoU on $[A] \to [C]$, $[B] \to [C]$, and $[A] \to [D]$, respectively. Using Minkowski, the improvements are +9.9%, +11.7%, and +5.9% mIoU. Notably, improvements are larger on the real adverse weather benchmark $[C]$ than on the synthetic corruption benchmark $[D]$, indicating superior robustness to real-world disturbances. The consistent gains across architectures highlight the approach's architecture-agnostic nature, making it broadly applicable to various LiDAR segmentation networks.

Effectiveness of Components. We conduct an ablation study on the [A]/[B]→[C] benchmark to evaluate the impact of A3Point's components (Tab. 4). For [A]→[C], the baseline without augmentations achieves 31.4% mIoU. Introducing the enhanced augmentation space (EAS) improves performance to 38.7%, highlighting the importance of diverse augmentations for domain robustness. Applying semantic shift region localization and masking to only optimize semantic consistency regions (SCR) with $\widetilde{\mathcal{L}}_{ce}$ further improves mIoU to 40.2%. Finally, incorporating latent variable-based distillation loss $\mathcal{L}_{distill}$ for semantic shift regions (SSR) achieves the best result of 41.3% mIoU. A similar trend is observed for [B]→[C], where the baseline starts at 15.5% mIoU. EAS significantly boosts performance to 24.9%, bridging the large gap between synthetic and real adverse weather data. Adding SCR improves mIoU to 26.5%, and the full model with SSR reaches 27.2%. These results demonstrate the complementary nature of our components across domain generalization scenarios. By leveraging learned semantic confusion priors to guide semantic shift optimization, the model adapts to novel disturbances while preserving semantic consistency.

Table 5: Ablation study of augmentation level. We define different levels (light, moderate, and heavy), same as Fig.1.

Method	None	light	moderate	heavy	random
$[A] \rightarrow [A]$	62.7	61.2	59.7	55.5	60.4
w/ A3Point	**62.6**	62.4	61.0	58.7	62.5
$[A] \rightarrow [C]$	31.4	37.6	38.0	37.1	38.7
w/ A3Point	31.8	38.5	40.4	40.5	**41.3**
$[B] \rightarrow [C]$	15.5	19.3	23.6	24.1	24.9
w/ A3Point	15.6	20.7	25.1	26.4	**27.2**

Table 6: Performance comparison of different strategies for modeling semantic confusion prior.

Strategy	$[A] \rightarrow [C]$	$[B] \rightarrow [C]$
None	38.7	24.9
GT-based	39.1	25.0
Offline	40.7	26.4
Online (ours)	**41.3**	**27.2**

Analysis of Augmentation Level. We analyze the impact of augmentation levels on network performance using the [A] validation set ([A]→[A]) and cross-domain benchmarks [A]/[B]→[C] (Tab. 5). On [A]→[A], baseline performance drops from 62.7% to 55.5% mIoU as augmentation increases from none to heavy, showing the adverse effect of severe augmentations. In contrast, A3Point maintains higher performance, with only a slight decrease from 62.6% to 58.7%, demonstrating its ability to mitigate the negative impact of aggressive augmentations by handling semantic shift regions. For [A]→[C], augmentations improve baseline performance from 31.4% to 38.7% mIoU, confirming their role in enhancing robustness. However, a slight drop from 38.0% to 37.1% from moderate to heavy augmentations suggests emerging semantic shift issues. A3Point avoids this drop, achieving 40.5% mIoU with heavy augmentations and 41.3% mIoU with random augmentations. Similar trends appear in [B]→[C], where A3Point achieves larger relative gains as augmentation intensity increases. The performance gap over the baseline grows from 1.4% with light augmentations to 2.3% with random augmentations. These findings confirm that A3Point effectively handles semantic shift, allowing the use of aggressive augmentations without harming performance. Its ability to balance source and target performance makes it well-suited for real-world applications.

Analysis of Semantic Confusion Prior. Tab. 6 compares different strategies for modeling semantic confusion prior on [A]/[B]→[C] benchmarks. For [A]→[C], the GT-based method, using one-hot ground truth labels, models only class-wise shape distribution but lacks inter-class confusion knowledge, yielding a marginal 0.4% mIoU gain. The offline method, which uses a pre-trained model's predictions on the source domain, improves mIoU by 2.0% but struggles to capture evolving confusion patterns during training. Similar trends occur in [B]→[C], where the GT-based method provides minimal gains (25.0% vs. 24.9%), and the offline method achieves better but limited improvement (26.4% vs. 24.9%). In contrast, our online modeling approach continuously updates the prior information, effectively decoupling semantic confusion in augmented point clouds. This dynamic strategy achieves the best performance: 41.3% mIoU on [A]→[C] and 27.2% mIoU on [B]→[C], outperforming the baseline by 2.6% and 2.3%, respectively. By continuously adapting to evolving confusion patterns, A3point mitigates semantic shifts and consistently outperforms static priors, demonstrating the importance of dynamic prior modeling in our framework.

Analysis of Semantic Shift Region. Fig.5 shows how the SSR mask ratio varies with augmentation level. As the augmentation intensity increases, the mask ratio of the localized SSR grows accordingly. This observation aligns with our expectation that stronger perturbations lead to more significant semantic shift in the augmented point clouds, resulting in a larger proportion of the input being identified as belonging to the SSR. The upward trend in the curve suggests that our proposed SSR localization module effectively captures the regions most affected by the augmentation-induced semantic shift, enabling the network to apply appropriate optimization strategies in these areas to mitigate the impact of the shift and enhance the model's robustness to adverse weather conditions.

5 Conclusion

This paper presents A3Point, an adaptive augmentation-aware latent learning framework for robust LiDAR semantic segmentation under adverse weather. A3Point effectively leverages a large augmentation space while mitigating semantic shift through a two-step strategy: (1) semantic confusion prior latent learning, which encodes local confusion patterns through discrete latent representations, and (2) semantic shift region localization, which detects anomalies in augmented point clouds to separate semantic consistency from semantic shift regions, enabling targeted optimization. Experiments on domain generalization benchmarks demonstrate its effectiveness, particularly in generalizing from synthetic normal weather to real adverse weather.

REFERENCES

Eren Erdal Aksoy, Saimir Baci, and Selcuk Cavdar. Salsanet: Fast road and vehicle segmentation in lidar point clouds for autonomous driving. In *2020 IEEE intelligent vehicles symposium (IV)*, pp. 926–932. IEEE, 2020.

Angelika Ando, Spyros Gidaris, Andrei Bursuc, Gilles Puy, Alexandre Boulch, and Renaud Marlet. Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5240–5250, 2023.

Yingbin Bai, Erkun Yang, Zhaoqing Wang, Yuxuan Du, Bo Han, Cheng Deng, Dadong Wang, and Tongliang Liu. Rsa: reducing semantic shift from aggressive augmentations for self-supervised learning. *Advances in Neural Information Processing Systems*, 35:21128–21141, 2022.

Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. 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.

Mario Bijelic, Tobias Gruber, and Werner Ritter. A benchmark for lidar sensors in fog: Is detection breaking down? In *2018 IEEE intelligent vehicles symposium (IV)*, pp. 760–767. IEEE, 2018.

Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11682–11692, 2020.

Hao Chen, Ze Wang, Xiang Li, Ximeng Sun, Fangyi Chen, Jiang Liu, Jindong Wang, Bhiksha Raj, Zicheng Liu, and Emad Barsoum. Softvq-vae: Efficient 1-dimensional continuous tokenizer. *arXiv preprint arXiv:2412.10958*, 2024.

Jaesung Choe, Chunghyun Park, Francois Rameau, Jaesik Park, and In So Kweon. Pointmixer: Mlp-mixer for point cloud understanding. In *European Conference on Computer Vision*, pp. 620–640. Springer, 2022.

Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3075–3084, 2019.

Jeffrey De Fauw, Sander Dieleman, and Karen Simonyan. Hierarchical autoregressive image models with auxiliary decoders. *arXiv preprint arXiv:1903.04933*, 2019.

Jing Du, John Zelek, and Jonathan Li. Weather-aware autopilot: Domain generalization for point cloud semantic segmentation in diverse weather scenarios. *ISPRS Journal of Photogrammetry and Remote Sensing*, 218:204–219, 2024.

Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14504–14513, 2021.

A Filgueira, H González-Jorge, Susana Lagüela, L Díaz-Vilariño, and Pedro Arias. Quantifying the influence of rain in lidar performance. *Measurement*, 95:143–148, 2017.

Whye Kit Fong, Rohit Mohan, Juana Valeria Hurtado, Lubing Zhou, Holger Caesar, Oscar Beijbom, and Abhinav Valada. Panoptic nuscenes: A large-scale benchmark for lidar panoptic segmentation and tracking. *IEEE Robotics and Automation Letters*, 7(2):3795–3802, 2022.

Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9224–9232, 2018.

Martin Hahner, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Fog simulation on real lidar point clouds for 3d object detection in adverse weather. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 15283–15292, 2021.

Martin Hahner, Christos Sakaridis, Mario Bijelic, Felix Heide, Fisher Yu, Dengxin Dai, and Luc Van Gool. 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.

Pei He, Licheng Jiao, Lingling Li, Xu Liu, Fang Liu, Wenping Ma, Shuyuan Yang, and Ronghua Shang. Domain generalization-aware uncertainty introspective learning for 3d point clouds segmentation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 651–660, 2024.

Robin Heinzler, Philipp Schindler, Jürgen Seekircher, Werner Ritter, and Wilhelm Stork. Weather influence and classification with automotive lidar sensors. In *2019 IEEE intelligent vehicles symposium (IV)*, pp. 1527–1534. IEEE, 2019.

Hyeonseong Kim, Yoonsu Kang, Changgyoon Oh, and Kuk-Jin Yoon. Single domain generalization for lidar semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17587–17598, 2023.

Jaeyeul Kim, Jungwan Woo, Jeonghoon Kim, and Sunghoon Im. Rethinking lidar domain generalization: Single source as multiple density domains. In *European Conference on Computer Vision*, pp. 310–327. Springer, 2024.

Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Lingdong Kong, Youquan Liu, Runnan Chen, Yuexin Ma, Xinge Zhu, Yikang Li, Yuenan Hou, Yu Qiao, and Ziwei Liu. Rethinking range view representation for lidar segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 228–240, 2023a.

Lingdong Kong, Youquan Liu, Xin Li, Runnan Chen, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Robo3d: Towards robust and reliable 3d perception against corruptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19994–20006, 2023b.

Lingdong Kong, Jiawei Ren, Liang Pan, and Ziwei Liu. Lasermix for semi-supervised lidar semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21705–21715, 2023c.

Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical transformer for lidar-based 3d recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17545–17555, 2023.

Miaoyu Li, Yachao Zhang, Xu Ma, Yanyun Qu, and Yun Fu. Bev-dg: Cross-modal learning under bird's-eye view for domain generalization of 3d semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11632–11642, 2023.

Ying Li, Lingfei Ma, Zilong Zhong, Fei Liu, Michael A Chapman, Dongpu Cao, and Jonathan Li. Deep learning for lidar point clouds in autonomous driving: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3412–3432, 2020.

You Li and Javier Ibanez-Guzman. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Processing Magazine*, 37(4): 50–61, 2020.

Hao Liu, Wilson Yan, and Pieter Abbeel. Language quantized autoencoders: Towards unsupervised text-image alignment. *Advances in neural information processing systems*, 36:4382–4395, 2023.

Zhipeng Luo, Zhongang Cai, Changqing Zhou, Gongjie Zhang, Haiyu Zhao, Shuai Yi, Shijian Lu, Hongsheng Li, Shanghang Zhang, and Ziwei Liu. Unsupervised domain adaptive 3d detection with multi-level consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8866–8875, 2021.

Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 4213–4220. IEEE, 2019.

Alexey Nekrasov, Jonas Schult, Or Litany, Bastian Leibe, and Francis Engelmann. Mix3d: Out-of-context data augmentation for 3d scenes. In *2021 international conference on 3d vision (3dv)*, pp. 116–125. IEEE, 2021.

Junsung Park, Kyungmin Kim, and Hyunjung Shim. Rethinking data augmentation for robust lidar semantic segmentation in adverse weather. *arXiv preprint arXiv:2407.02286*, 2024.

Junsung Park, Hwijeong Lee, Inha Kang, and Hyunjung Shim. No thing, nothing: Highlighting safety-critical classes for robust lidar semantic segmentation in adverse weather. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6690–6699, 2025.

Thierry Peynot, James Underwood, and Steven Scheding. Towards reliable perception for unmanned ground vehicles in challenging conditions. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1170–1176. IEEE, 2009.

Gilles Puy, Alexandre Boulch, and Renaud Marlet. Using a waffle iron for automotive point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3379–3389, 2023.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pp. 1530–1538. PMLR, 2015.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

Julian Ryde and Nick Hillier. Performance of laser and radar ranging devices in adverse environmental conditions. *Journal of Field Robotics*, 26(9):712–727, 2009.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10765–10775, 2021.

Yuhta Takida, Yukara Ikemiya, Takashi Shibuya, Kazuki Shimada, Woosung Choi, Chieh-Hsin Lai, Naoki Murata, Toshimitsu Uesaka, Kengo Uchida, Wei-Hsiang Liao, et al. Hq-vae: Hierarchical discrete representation learning with variational bayes. *arXiv preprint arXiv:2401.00365*, 2023.

Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *European conference on computer vision*, pp. 685–702. Springer, 2020.

Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6411–6420, 2019.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pp. 1096–1103, 2008.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010.

Yifei Wang, Zhengyang Geng, Feng Jiang, Chuming Li, Yisen Wang, Jiansheng Yang, and Zhouchen Lin. Residual relaxation for multi-view representation learning. *Advances in Neural Information Processing Systems*, 34:12104–12115, 2021.

Aoran Xiao, Xiaofei Yang, Shijian Lu, Dayan Guan, and Jiaxing Huang. Fps-net: A convolutional fusion network for large-scale lidar point cloud segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 176:237–249, 2021.

Aoran Xiao, Jiaxing Huang, Dayan Guan, Kaiwen Cui, Shijian Lu, and Ling Shao. Polarmix: A general data augmentation technique for lidar point clouds. *Advances in Neural Information Processing Systems*, 35:11035–11048, 2022a.

Aoran Xiao, Jiaxing Huang, Dayan Guan, Fangneng Zhan, and Shijian Lu. Transfer learning from synthetic to real lidar point cloud for semantic segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 2795–2803, 2022b.

Aoran Xiao, Jiaxing Huang, Weihao Xuan, Ruijie Ren, Kangcheng Liu, Dayan Guan, Abdulmotaleb El Saddik, Shijian Lu, and Eric P Xing. 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.

Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d: Self-training for unsupervised domain adaptation on 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10368–10378, 2021.

Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

Jianlong Yuan, Yifan Liu, Chunhua Shen, Zhibin Wang, and Hao Li. A simple baseline for semi-supervised semantic segmentation with strong data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8229–8238, 2021.

Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9601–9610, 2020.

Haimei Zhao, Jing Zhang, Zhuo Chen, Shanshan Zhao, and Dacheng Tao. 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.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16259–16268, 2021.

Lin Zhao, Hui Zhou, Xinge Zhu, Xiao Song, Hongsheng Li, and Wenbing Tao. Lif-seg: Lidar and camera image fusion for 3d lidar semantic segmentation. *IEEE Transactions on Multimedia*, 26: 1158–1168, 2023.

Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9939–9948, 2021.

A Appendix

This appendix provides additional details and analyses to complement our main paper. We include implementation details, ablation studies, and additional visualization results. Specifically, the appendix is organized as follows:

- **Section B** introduces the autoencoder implementation details introduced in Semantic Confusion Prior Latent Learning.
- **Section C** presents our anomaly detection strategy in Semantic Shift Region Localization
- **Section D** discusses the differences between discrete and continuous encoding.
- **Section E** provide a pseudo algorithm of A3Point.
- **Section F** studies the impact of Other Augmentation Techniques.
- **Section G** provides a hyperparameter analysis of A3Point.
- **Section H** offers a visual analysis about Semantic Shift Region.
- **Section K** provides more qualitative results.

B Implementation of SCP Latent Learning

In this section, we provide the implementation details of our semantic confusion prior (SCP) latent learning module, which is based on the vector quantized variational autoencoder (VQ-VAE) architecture (Van Den Oord et al., 2017). The goal is to learn a discrete latent representation that captures the semantic confusion patterns in the original point cloud predictions.

B.1 VQ-VAE Architecture

The VQ-VAE consists of an encoder, a decoder, and a vector quantization layer. The encoder network maps the input point cloud features x concatenated with the predicted probabilities $f(x)$ to a latent representation z_e. We use sparse 3D convolutions (Tang et al., 2020) in the encoder to process the sparse point cloud data efficiently. The architecture is as follows:

- The encoder has 4 sparse conv3d downsampling blocks with channel dimensions [16, 32, 64, 128]. Each block consists of a stride-2 sparse conv3d, batch norm, and LeakyReLU. This downsamples the point cloud by 16x.
- An additional sparse conv3d, batch norm and LeakyReLU, followed by 2 residual blocks to further process the features.
- A final sparse conv3d to map to the latent dimension (64).

The decoder network reconstructs the predicted probabilities $f(x)$ from the quantized latent representation z_q. It follows a mirrored architecture to the encoder:

- A sparse conv3d, batch norm and LeakyReLU to map from the latent dimension to the initial feature dimension (128).
- 2 residual blocks for further processing.
- 4 sparse transposed conv3d upsampling blocks, each consists of a stride-2 transposed sparse conv3d, batch norm and LeakyReLU. This upsamples the features by 16x back to the original resolution.
- A final sparse conv3d and tanh activation to reconstruct the predicted probabilities.

B.2 Per-Class Vector Quantization

To avoid inter-class confusion, we perform vector quantization independently for each semantic class. The VQ codebook $\mathcal{C}$ contains learnable embeddings for each class, with dimensions $C \times k \times D$, where C is the number of classes, k is the number of embeddings per class, and D is the latent dimension.

During the forward pass, we split the encoder output z_e into class-specific latents based on the class labels. For each class, the latents are quantized to the nearest embedding in its corresponding sub-codebook using L2 distance. The distances to embeddings of other classes are masked out to prevent selecting incorrect classes.

B.3 Commitment Loss and Embedding Loss

The learning objective combines a commitment loss and an embedding loss, following the original VQ-VAE:

- The commitment loss encourages the encoder output z_e to commit to the selected embeddings z_q, and is defined as:
$$\mathcal{L}_{\text{commit}} = \|z_e - \text{sg}(z_q)\|_2^2 \tag{12}$$
- The embedding loss brings the selected embeddings z_q close to the encoder output z_e, and is defined as:
$$\mathcal{L}_{\text{embed}} = \|\text{sg}(z_e) - z_q\|_2^2 \tag{13}$$

where sg stands for the stop-gradient operation. The total VQ loss is:

$$\mathcal{L}_{\text{VQ}} = \mathcal{L}_{\text{commit}} + \beta \mathcal{L}_{\text{embed}} \tag{14}$$

where β is a weighing coefficient (set to 0.25).

B.4 Reconstruction Loss

To ensure the quantized representation can reconstruct the input predicted probabilities, we employ a mean squared error reconstruction loss:

$$\mathcal{L}_{\text{recon}} = \|f(x) - \mathcal{D}(z_q)\|_2^2 \tag{15}$$

The final loss is a weighted combination of the reconstruction loss and the VQ losses:

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{VQ}} \tag{16}$$

In summary, our SCP latent learning module utilizes a VQ-VAE architecture with sparse convolutions to learn a discrete latent representation of semantic confusion patterns in a class-wise manner. The model is trained end-to-end using a combination of reconstruction loss and VQ losses. This allows capturing informative priors on the semantic confusion from the original point cloud predictions.

C Implementation of SSR Localization

In this section, we describe the implementation details of our semantic shift region (SSR) localization module. The goal is to identify regions in the augmented point cloud where the network's predictions deviate from the learned semantic confusion priors, indicating potential semantic shifts caused by the augmentations.

C.1 Tracking Latent Representation Distributions

During the training of the SCR latent learning module, we track the distribution of latent representations associated with each quantized latent variable (i.e., each embedding in the codebook). Specifically, for each latent variable e_i, we calculate the variance σ_i^2 of its corresponding latent representations before quantization. The variance is updated using an exponential moving average (EMA) with a momentum factor γ (set to 0.9):

$$\sigma_i^2 \leftarrow \gamma \sigma_i^2 + (1 - \gamma)\text{Var}(z_e | z_q = e_i) \tag{17}$$

where $\text{Var}(\cdot)$ denotes the variance operation. This allows us to estimate the typical distribution of latent representations for each latent variable.

C.2 Locating Semantic Shift Regions

To locate the semantic shift regions, we first pass the augmented point cloud through the trained encoder $\mathbb{E}$ to obtain its latent representations z_e. We then compare each latent representation to the distribution of its nearest latent variable in the corresponding sub-codebook.

Specifically, for each latent representation z_e^j, we find its nearest latent variable $e_{NN(j)}$ in the sub-codebook of the corresponding class. We consider z_e^j to be an outlier (i.e., belonging to a semantic shift region) if its distance from $e_{NN(j)}$ exceeds a threshold based on the tracked variance $\sigma^2_{NN(j)}$:

$$\text{isOutlier}(z_e^j) = \mathbb{1}\left(\|z_e^j - e_{NN(j)}\|_2 > t\sqrt{\sigma^2_{NN(j)}}\right) \tag{18}$$

where $\mathbb{1}(\cdot)$ is the indicator function and t is a hyperparameter controlling the threshold (set to 3). The intuition is that if a latent representation deviates significantly from the typical distribution of its nearest latent variable, it likely corresponds to a semantic shift region.

C.3 Generating SSR and SCR Masks

After identifying the outlier latent representations, we project them back to the point cloud space to generate masks for the semantic shift regions (SSR) and semantic consistency regions (SCR).

We first create a binary mask M_{outlier} in the latent representation space, where $M^j_{\text{outlier}} = \text{isOutlier}(z_e^j)$. We then use the transpose of the point cloud downsampling operation (used in the encoder) to upsample the mask back to the original point cloud resolution. This gives us the SSR mask M_{SSR}.

The SCR mask is simply the complement of the SSR mask:

$$M_{\text{SCR}} = \mathbb{1} - M_{\text{SSR}} \tag{19}$$

C.4 Handling Sparse Outliers

In practice, the outlier latent representations may be sparse and scattered. To ensure the SSR mask covers semantically meaningful regions, we perform a dilation operation on the SSR mask. Specifically, for each point (x, y, z, c) in the SSR mask, we set its neighboring points within a certain radius r to also belong to the SSR. This helps to connect nearby outlier points and form contiguous semantic shift regions.

The dilation operation can be efficiently implemented by first upsampling the outlier mask to a dense 3D grid, performing dilation on the grid, and then downsampling the dilated mask back to the point cloud using nearest-neighbor interpolation.

In summary, our SSR localization module identifies semantic shift regions in the augmented point cloud by comparing the latent representations to the learned semantic confusion priors. It generates SSR and SCR masks to guide the subsequent training with appropriate losses for each region. The module is computationally efficient and can be integrated into the training pipeline without significantly increasing the training time.

D Discussion: Discrete vs. Continuous Encoding

A key aspect of the proposed Semantic Confusion Prior (SCP) is **explicitly obtaining the distribution form of the representation** for subsequent localization and optimization. This section discusses the differences between discrete and continuous encoding methods and explains why discrete encoding is more suitable for our framework.

D.1 COMPARISON OF ENCODING PARADIGMS

Discrete Encoding (e.g., VQ-VAE). VQ-VAE (Van Den Oord et al., 2017) learns a discrete latent space by: 1) Mapping the input to a latent space via an encoder. 2) Quantizing the latent space into a codebook, effectively clustering representative features.

Continuous Encoding Methods. Continuous representation methods can be categorized into two types: **(1) Fixed Prior Distribution Paradigms (e.g., VAE (Kingma et al., 2013), Flow-based Models (Rezende & Mohamed, 2015; Kingma & Dhariwal, 2018))** These methods constrain the latent space to a fixed prior distribution (e.g., Gaussian) to achieve implicit global distribution alignment. However, this alignment lacks interpretability and cannot naturally decouple different semantic classes, often requiring separate encoders and decoders per class. **(2) Unconstrained Latent Representations (e.g., AE (Rumelhart et al., 1986), DAE (Vincent et al., 2008), SDAE (Vincent et al., 2010))** These methods learn continuous latent representations without a predefined prior, making it difficult to directly extract structured distribution information. As a result, clustering techniques may still be needed to impose structure, similar to VQ-VAE's discrete encoding.

D.2 WHY DISCRETE ENCODING IS NECESSARY?

While discrete encoding limits the number of latent patterns to k, this does not hinder our framework's effectiveness. Instead, it provides several advantages:

Better Interpretability: Each latent code represents a specific semantic confusion pattern, making it easier to analyze and interpret the learned representations.

Class-wise Separation: VQ-VAE allows natural clustering of latent representations per class, which is crucial for our semantic shift localization.

Robustness and Generalization: A well-designed codebook ensures that only the most representative and meaningful patterns are learned, improving generalization to unseen data.

E PSEUDO ALGORITHM OF A3POINT.

We further provide detailed algorithmic description in Alg. 1.

F IMPACT OF OTHER AUGMENTATION TECHNIQUES

To further analyze the role of various augmentation techniques, we conduct experiments to evaluate their impact on both source and target domain performance.

F.1 EXPERIMENTAL SETUP

We follow the same training setup as in Section 3.3.2, varying the intensity of secondary augmentations such as scaling, flipping, and noise perturbation, while keeping point deformation and point loss augmentations unchanged. We define three augmentation levels: - **Light**: Default augmentation settings used in our main experiments. - **Moderate**: Increasing the magnitude of scaling and flipping while maintaining the original distribution. - **Heavy**: Aggressively increasing scaling factors and noise perturbation levels.

F.2 RESULTS AND ANALYSIS

Tab. 7 presents the results on the [A]→[C] and [B]→[C] benchmarks.

From the results, we observe that: 1. **Excessive secondary augmentations degrade both source and target performance.** Increasing the augmentation magnitude from Mild to Heavy reduces mIoU by 3.8% on [A]→[C] and 2.7% on [B]→[C]. 2. **Moderate augmentations provide limited benefit.** Compared to the Mild setting, Moderate augmentation slightly reduces performance, indicating that secondary augmentations do not contribute significantly to domain adaptation. 3. **Source**

Algorithm 1 Pseudo Algorithm of A3Point

Inputs: Source domain $D_S = \{(x_i^S, y_i^S)\}_{i=1}^{N_S}$
Define: Network f_θ, Autoencoder $\mathbb{E}, \mathbb{D}$, Codebook $\mathcal{C}$, Augmentation function $\mathcal{A}$, Learning rates α, β, δ, Loss weight λ
Output: Trained model f_θ
for each batch (x_i^S, y_i^S) in D_S **do**
    # Step 1: Augmented Training
    Apply augmentations: $(\hat{x}_i^S, \hat{y}_i^S) = \mathcal{A}(x_i^S, y_i^S)$
    Compute supervised loss: $\mathcal{L}_{ce} = \ell_{ce}(f_\theta(x_i^S), y_i^S)$ ▷ Eq. (1)
    Compute augmented loss: $\hat{\mathcal{L}}_{ce} = \ell_{ce}(f_\theta(\hat{x}_i^S), \hat{y}_i^S)$ ▷ Eq. (2)
    # Step 2: Semantic Confusion Prior Learning
    Compute latent embedding: $z_e = \mathbb{E}([f_\theta(x_i^S) \oplus x_i^S])$
    Quantize latent code: $z_q = \text{quantize}(z_e)$ using codebook $\mathcal{C}$ ▷ Eq. (3)
    Reconstruct prediction: $\bar{x}_i^S = \mathbb{D}(z_q)$
    Update VQ-VAE loss: $\mathcal{L}_{vq} = ||x_i^S - \bar{x}_i^S||_2^2 + ||\text{sg}(z_e) - z_q||_2^2 + ||z_e - \text{sg}(z_q)||_2^2$ ▷ Eq. (4)
    # Step 3: Semantic Shift Region Localization
    Track variance statistics for latent codes σ^2 using EMA
    Map augmented predictions to latent space: $z_e^{aug} = \mathbb{E}([f_\theta(\hat{x}_i^S) \oplus \hat{x}_i^S])$
    Identify semantic shift regions (SSR) via anomaly detection:
        - Semantic consistency mask: $M_{SCR} = \mathbb{1}(z_e^{aug} \in r(NN_{sub}(z_e^{aug})))$ ▷ Eq. (7)
        - Semantic shift mask: $M_{SSR} = \mathbb{1}(z_e^{aug} \notin r(NN_{sub}(z_e^{aug})))$ ▷ Eq. (8)
    # Step 4: Optimization
    Compute loss for SCR: $\widetilde{\mathcal{L}}_{ce} = \ell_{ce}(f_\theta(\hat{x}_i^S) \odot M_{SCR}, y_i^S \odot M_{SCR})$ ▷ Eq. (9)
    Compute distillation loss for SSR: $\mathcal{L}_{distill} = ||z_e^{aug} - \text{sg}[NN_{global}(z_e^{aug})]||_2^2$ ▷ Eq. (10)
    Update model: $\theta \leftarrow \theta - \delta \nabla_\theta(\mathcal{L}_{ce} + \widetilde{\mathcal{L}}_{ce} + \lambda \mathcal{L}_{distill})$ ▷ Eq. (11)
end for

Augmentation Level	[A]→[C]	[B]→[C]	[A]→[A]
Light	**41.3**	**27.2**	**62.5**
Moderate	40.1	26.0	60.3
Heavy	37.5	24.5	55.8

Table 7: Impact of different augmentation levels on domain generalization performance.

domain performance deteriorates with stronger augmentations. On [A]→[A], mIoU drops significantly from 62.6% to 55.8%, suggesting that excessive scaling and noise perturbation disrupt the original data distribution and hinder learning.

These findings confirm that point deformation and loss (jittering, point drop) are the primary drivers of domain adaptation, while other augmentations are actually domain-agnostic factors and should be applied conservatively to avoid performance degradation.

G Influence of Parameters Setting

In this section, we discuss the key hyperparameters of our A3Point framework and their impact on the performance and behavior of the model. All experiments are conducted on SemanticKITTI→ semanticSTF.

G.1 Latent Space Dimensions

The dimensions of the latent space in the SCR latent learning module, determined by the number of embeddings per class (k) and the embedding dimension (D), affect the expressiveness and granularity of the learned semantic confusion patterns.

In our implementation, we set $k = 32$ and $D = 64$ (Table 8). These values provide a good trade-off between the representational power of the latent space and the computational efficiency of the

k	D	mIoU (%)
16	64	39.9
32	64	41.3
64	64	41.4
32	32	40.8
32	64	41.3
32	128	41.2

Table 8: Ablation study on latent space dimensions.

module. Increasing k allows the model to capture more diverse semantic confusion patterns within each class, but it also increases the memory footprint and the risk of overfitting. Similarly, increasing D enhances the capacity of each embedding to encode more complex patterns but also increases the computational overhead.

G.2 SSR Localization Threshold

The threshold t used in the SSR localization module determines the sensitivity of detecting semantic shift regions.

	$t = 2$	$t = 3$	$t = 4$
mIoU (%)	41.0	41.3	40.9

Table 9: Ablation study on SSR localization threshold.

In our experiments, we set $t = 3$, which effectively identifies the regions with substantial semantic shifts while minimizing false positives. A higher value of t results in a more conservative approach, where only the most significant deviations from the learned semantic confusion patterns are considered as semantic shifts. Conversely, a lower value of t makes the module more sensitive, potentially identifying more regions as semantic shifts.

G.3 Loss Weighting Coefficient

The loss weighting coefficient λ balances the contributions of the cross-entropy loss and the distillation loss in the overall training objective.

	$\lambda = 0.02$	$\lambda = 0.1$	$\lambda = 0.5$
mIoU (%)	40.5	41.3	40.9

Table 10: Ablation study on loss weighting coefficient.

In our experiments, we found that setting $\lambda = 0.1$ achieves a good balance between the two loss terms. A higher value of λ gives more importance to the distillation loss, encouraging the model to focus more on aligning the predictions in the semantic shift regions with the learned semantic confusion patterns. On the other hand, a lower value of λ emphasizes the cross-entropy loss, prioritizing the overall segmentation accuracy.

H Visual Analysis of SSR

To better understand how our method identifies semantic shift regions (SSR), we conduct a detailed visual analysis comparing the error maps with the detected SSR, as shown in Fig. 6. The error map highlights regions where the model's predictions differ from ground truth labels, while SSR indicates areas identified by our semantic shift detection mechanism.

From the visualization results, we observe a strong correlation between the SSR and regions prone to prediction errors. Specifically: (1) Our SSR detection effectively captures areas where adverse weather conditions cause significant semantic ambiguity, particularly at object boundaries and distant regions where point cloud density decreases. (2) The SSR often corresponds to challenging

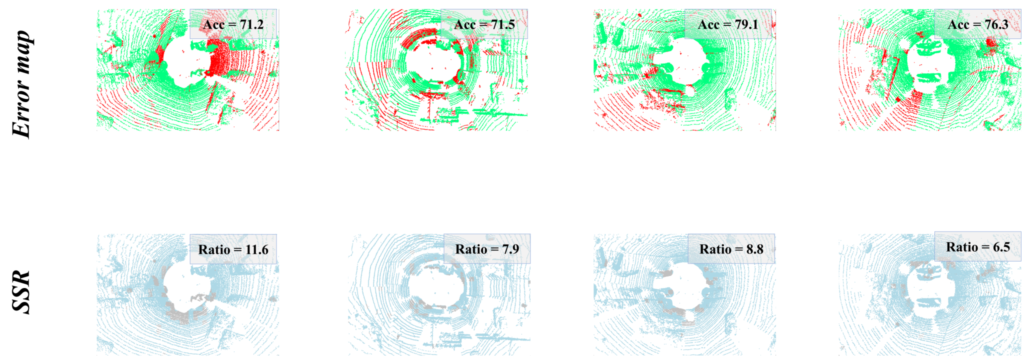


Figure 6: Qualitative results on comparison between error map with semantic shift region.

scenarios such as intersections between different semantic categories (e.g., road-sidewalk boundaries) and areas with complex geometric structures. (3) The semantic consistency regions (non-SSR areas) generally align well with regions where the model maintains accurate predictions, validating our approach's ability to identify reliable predictions. This visual analysis demonstrates that our SSR detection mechanism provides meaningful guidance for applying different optimization strategies during training.

I MORE QUALITATIVE RESULTS

To further demonstrate the effectiveness of our proposed method, we present additional qualitative results comparing our approach with baseline methods on the challenging SemanticKITTI→SemanticSTF domain generalization task, as shown in Fig. 7. These qualitative results further validate the effectiveness of our semantic confusion prior learning and semantic shift region localization strategies in improving domain generalization performance for LiDAR semantic segmentation.

J MORE QUALITATIVE RESULTS

To further demonstrate the effectiveness of our proposed method, we present additional qualitative results comparing our approach with baseline methods on the challenging SemanticKITTI→SemanticSTF domain generalization task, as shown in Fig. 7. These qualitative results further validate the effectiveness of our semantic confusion prior learning and semantic shift region localization strategies in improving domain generalization performance for LiDAR semantic segmentation.

K LLMS AND SOCIETY IMPACT

Use of Large Language Models (LLMs). In preparing this manuscript, we used LLMs solely for language polishing and writing assistance (e.g., clarity, grammar, and style). LLMs were not used to generate research ideas, experimental results, code, or analyses, and no proprietary data or sensitive information were provided to LLMs beyond the manuscript text. All technical content, experiments, and conclusions were produced and verified by the authors.

Within this paper, we present an approach for domain-generalized LiDAR semantic segmentation under adverse weather. Our contributions focus on robustness to distribution shifts via augmentation-aware latent learning and semantic shift localization. At present, we are not aware of direct negative societal implications arising specifically from the proposed methodology.

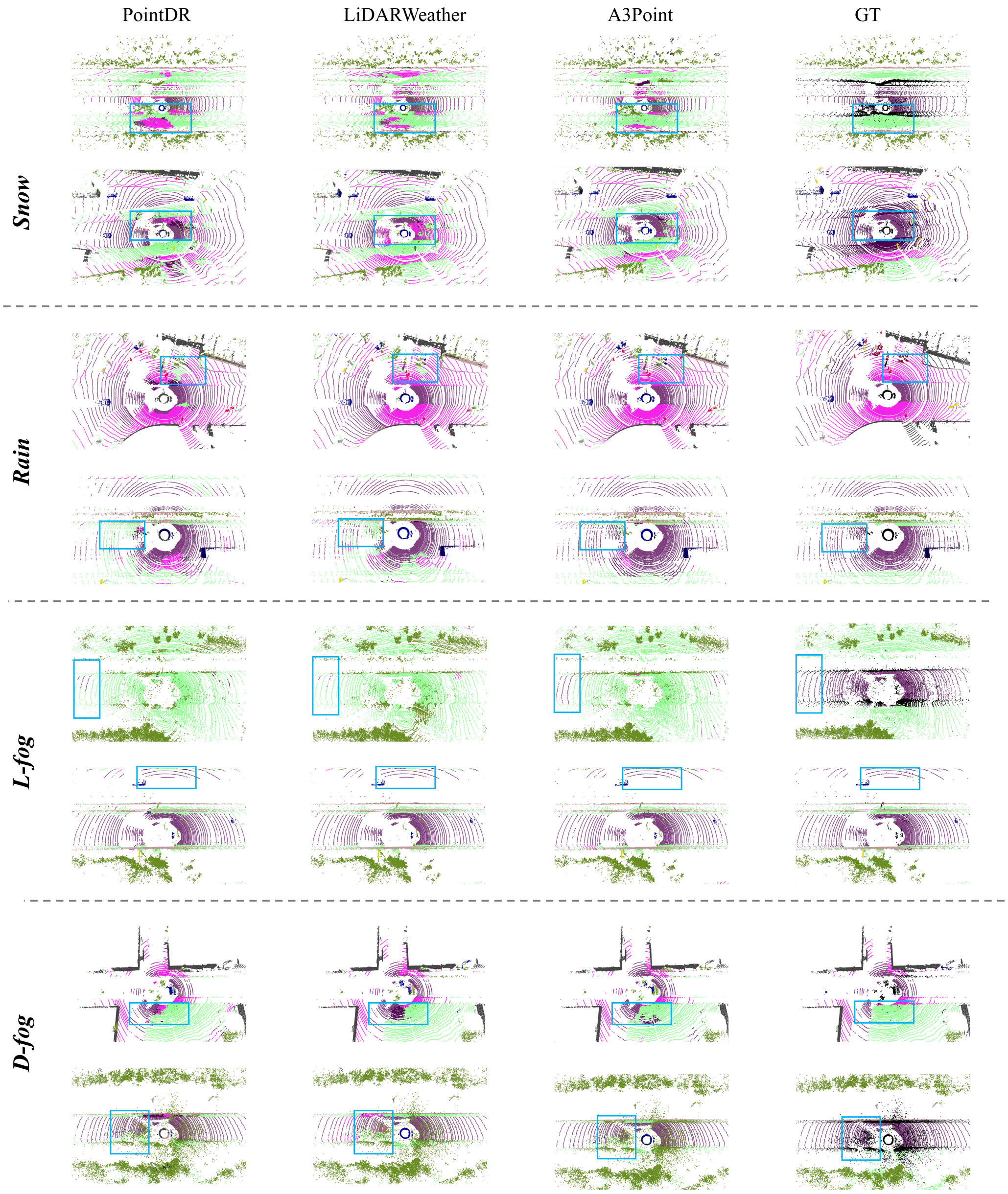


Figure 7: Qualitative results on [A] → [C], where improvements are marked with boxes.