D^{3}CTTA: Domain-Dependent Decorrelation for Continual Test-Time Adaption of 3D LiDAR Segmentation

Jichun Zhao[1], Haiyong Jiang[1]*, Haoxuan Song[1], Jun Xiao[1]*, Dong Gong[2]
[1]University of Chinese Academy of Sciences, [2]University of New South Wales
zhaojichun23@mails.ucas.ac.cn, haiyong.jiang@ucas.ac.cn,
songhaoxuan24@mails.ucas.ac.cn, xiaojun@ucas.ac.cn, dong.gong@unsw.edu.au

Abstract

Adapting pre-trained LiDAR segmentation models to dynamic domain shifts during testing is of paramount importance for the safety of autonomous driving. Most existing methods neglect the influence of domain changes and point density in continual test-time adaption (CTTA), relying on backpropagation and large batch sizes for stability. We approach this problem with three insights: 1) Point clouds at different distances usually have different densities resulting in distribution disparities; 2) The feature distribution of different domains varies, and domain-aware parameters can alleviate domain gaps; 3) Features are highly correlated and make segmentation of different labels confusing. To this end, this work presents D^{3}CTTA, an online backpropagation-free framework for 3D continual test-time adaption for LiDAR segmentation. D^{3}CTTA consists of a distance-aware prototype learning module to integrate LiDAR-based geometry prior and a domain-dependent decorrelation module to reduce feature correlations among different domains and different categories. Extensive experiments on three benchmarks showcase that our method achieves a state-of-the-art performance compared to both backpropagation-based methods and backpropagation-free methods. Code is available at https://github.com/ZhaoJichun1/D3CTTA.

1. Introduction

Leveraging simulation data to train segmentation models for autonomous driving becomes a favorable choice for its scalable label collection and wide coverage of diverse corner cases [33, 38]. However, real-world testing scenes have a very different data distribution from the synthetic ones, raising a critical problem for the robustness and safety of autonomous driving. Moreover, a driving car can encounter dynamic and diverse domain changes over time due to different sensor configurations, environmental changes, and viewpoint shifts. To address these problems, we aim at online continual test-time adaptation (CTTA) for LiDAR segmentation without access to the source domain.

The origin of CTTA can be traced to test-time adaption (TTA) [14, 40, 47, 48] that adapts a source-trained model to a single specific domain at test time. Therefore, TTA-based methods are designed for adapting to a single test domain rather than handling continuous domain shifts. The seminal work on CTTA, CoTTA[41], utilizes the mean teacher framework for continual parameter updating and stochastic restoration to mitigate cumulative errors. Subsequent works [1, 2, 9, 23] follow the mean teacher framework and focus on improving pseudo-label quality and stabilizing model updates across domains. Particularly, GIPSO [33] addresses 3D TTA by utilizing a highly time-consuming geometric feature encoder for label propagation. While effective, these methods require computationally expensive backpropagation and some even need large batch sizes for stable parameter updates, limiting their practicality for autonomous driving scenes. Furthermore, most of these methods are primarily designed for 2D classification tasks and merely leverage the unique density distributional properties of LiDAR data, particularly when point cloud density and geometric structures vary significantly across domains.

To address these issues, we propose a domain-aware backpropagation-free approach for 3D LiDAR CTTA, termed D^{3}CTTA. The first idea is to leverage density variations within a LiDAR scan to enhance domain adaptation. Traditional prototype-based methods [14] construct a prototype for each category. Considering varying point cloud densities usually result in distributional disparities [5, 11, 12, 43], representing them with different prototypes according to densities is a better choice. The density of LiDAR points is correlated to their distance to the LiDAR sensor, therefore we can construct distance-aware prototypes for better discriminativity. Moreover, the semantic consistency among neighboring regions also provides a geometric clue to select reliable predictions for model adaption.

*Joint Corresponding Authors. Haiyong is the Project Lead.

Another observation is that point-wise features are highly correlated, resulting in susceptible classifications to noisy input, as shown in the left and middle of Fig. 3 (a). Compared to the backpropagation-based CTTA, removing feature correlations can be conducted online without any optimization and is very efficient. Considering the data distribution of different testing domains varies a lot, we make the decorrelation and classification domain-specific, further alleviating domain confusion.

To fulfill the above ideas, we present a domain-dependent decorrelation method for 3D CTTA. The overview of the method is shown in Fig. 1. D^{3}CTTA consists of two core modules: a Distance-Aware Prototype Learning (DAPL) module and a Domain-Specific Decorrelation (DSD) module. The DAPL module constructs distance-aware prototypes by accounting for the influence of varying point cloud density on learned point cloud features. The DSD module conducts domain-specific decorrelation with reliable predictions from the DAPL module and finally predicts the LiDAR segmentation.

To summarize, our contributions lie in the following aspects.

- A novel framework for online 3D CTTA without backpropagation.
- The distance-aware prototype learning module to incorporate the LiDAR geometry prior to CTTA.
- The domain-specific decorrelation module to reduce point-wise feature correlations for less category confusion.
- State-of-the-art performance on three CTTA benchmarks, Synth4D to SemanticKITTI-C, Synth4D to nuScenes-C, and repeating domains from Synth4D to SemanticKITTI-C, demonstrating its effectiveness for 3D CTTA.

2. Related work

2.1. Test-Time Adaptation

Test-time adaptation aims to adapt source-pretrained models to target domains during inference. A typical line of works [10, 14, 33, 39, 45, 47, 48] relies on pseudo labels to guide the adaptation. Pseudo-labels are not equally certain and important, so some methods [14, 48] select confident samples for prototype construction, while ProDA [47] weights samples based on their distance to prototypes. Recent works [33, 51] on 3D segmentation only leverage geometric prior for reliable pseudo-label construction.

Batch normalization (BN) updates [13, 22, 29, 40, 42] form another kind of approach for TTA. TENT [40] updates BN layers by minimizing entropy and ensures fast adaptation by recalculating batch statistics on the fly, while DIGA [42] proposes updating BN statistics using exponential moving averages to adapt smoothly. TTN [22] adjusts BN layers based on domain shift sensitivity, dynamically balancing source and target domain statistics. Another strategy for TTA is to use entropy minimization [7, 21, 29, 40, 46, 49] to encourage more confident predictions on target domain samples. For example, TENT [40], AugMix [7], and MEMO [46] directly minimize prediction entropy to encourage confident outputs. These methods collectively provide diverse pathways to improve TTA, balancing efficient adaptation through pseudo-labeling, BN updates, and robust entropy minimization, ensuring models remain reliable across changing target domains. In this work, we focus on continual TTA for LiDAR segmentation.

2.2. Continual Test-Time Adaptation

CTTA is a special case of TTA with continuously changing target domains during adaption. The pioneering work CoTTA [41] introduces a mean teacher framework using augmentation-average predictions to mitigate cumulative errors and a stochastic restoration mechanism to prevent catastrophic forgetting. To improve prediction accuracy and stability, some methods enhance the mean teacher framework with a symmetric cross-entropy loss [6], regularization on the posterior distribution [1], and across-modal label fusion [2]. Another category of methods [9, 36, 44] addresses CTTA through data augmentation, additional data memories, and adaptive sampling. Additionally, some approaches [20, 23, 27] tackle domain-specific challenges by dynamically detecting domain changes and learning domain-specific models by low-rank experts [20] and modeling resetting [27]. ViDA [23] exploits high-rank embedding for domain-specific knowledge and low-rank embeddings for domain-shared knowledge.

2.3. Domain Adaption for LiDAR Segmentation

Point cloud semantic segmentation has been widely studied, with various methods proposed to tackle different challenges in 3D scene understanding [50]. However, autonomous driving presents unique domain adaptation challenges due to diverse domain shifts, such as varying numbers of LiDAR beams and changing weather conditions. These domain changes have a significant impact on the robustness of 3D perception and make it urging to properly adapt the trained model. Data mixing is a widely adopted domain adaption method and is also suitable for LiDAR segmentation [18, 19, 32]. Adversarial methods have also been explored for LiDAR segmentation [3, 4], typically leveraging adversarial training frameworks to generate fake target-domain data, thereby aligning the source domain distribution with the target domain. Furthermore, the semantic continuity in the geometry structures and temporal motion patterns consists of two important clues to construct better pseudo labels [24, 30, 34, 35]. Another obstacle lies in the changes in LiDAR beam densities. Therefore, density-aware features and data augmentation strate-

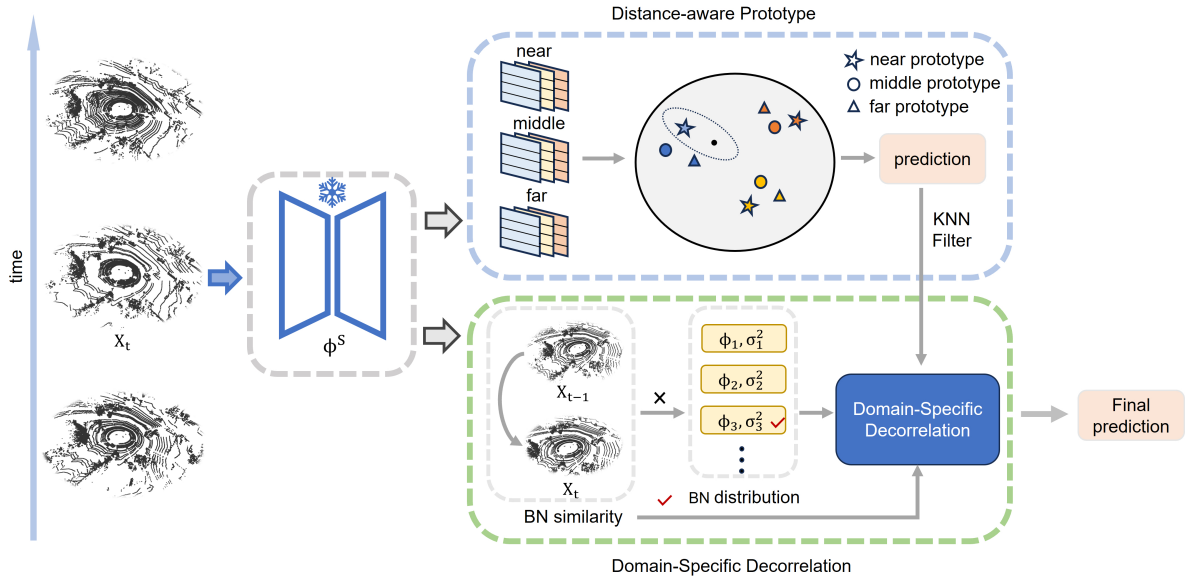


Figure 1. Framework of our backpropagation-free D^{3}CTTA. The Distance-Aware Prototype Module learns prototypes for point clouds at varying distances and refines pseudo-labels through KNN filtering. The Domain-Specific Decorrelation Module reduces feature confusion by applying decorrelations. It distinguishes the domain of the current sample by evaluating the similarity of batch normalization (BN) between consecutive frames and domain distributions, then updates distinct decorrelation matrices and prototypes for each domain.

gies [12, 15, 16, 35] are good strategies to sidestep domain gaps caused by the density of LiDAR beams. In this work, we focus on domain-specific adaptions in an online setting.

3. Methodology

This work aims at online continual test-time adaption of a pre-trained source model for 3D LiDAR segmentation in different target domains. The online CTTA takes as input one unlabeled LiDAR scan $\boldsymbol{X}$ from a target domain $D = \{D_{T_1}, D_{T_2}, \ldots, D_{T_n}\}$ and updates the source model $\mathcal{F}_s$ to a target model $\mathcal{F}_t$ without training. The challenges of the problem lie in two aspects. First, the data distribution of target domains may diverge from the source training domain and can change over time due to factors like sensor variations, environmental changes, or shifts in viewpoints. Second, the adaption process does not have access to source domain data and allows only a one-time pass through each target domain.

To address this problem, we propose a domain-specific adaption method called D^{3}CTTA, consisting of a Distance-Aware Prototype Learning module and a Domain-Specific Decorrelation module. The overview is shown in Fig. 1. First, the method takes as input a LiDAR scan and predicts pseudo labels with the source model. The DAPL module selects reliable predicted pseudo labels with a Space-Aware Filter and continuously updates classification prototypes with reliable pseudo labels (see Sec. 3.2). Afterward, the DSD module takes the reliable pseudo labels and constructs unique decorrelation matrices and prototypes for each domain (see Sec. 3.3). This module enables us to optimize feature distributions effectively without computationally intensive backpropagation while minimizing the mutual influence of distributional differences across domains, ultimately achieving more accurate predictions. At last, we take the prediction of the DSD module as the final result.

3.1. Preliminary: Prototype-based TTA

Pseudo-prototypical classifier [14] is a typical method for optimization-free TTA. And prototypes are typically used to capture data and class characteristics [8, 25]. Assuming the pre-trained segmentation model can be divided into the last linear layer $g_{\boldsymbol{w}}$ for point-wise classification and a feature extractor $f_{\boldsymbol{\theta}}$ for the rest network. The method replaces $g_{\boldsymbol{w}}$ with a prototype-based classifier and adjusts prototype features according to the support set $\mathbb{S}_k^t$ for each time step t and category k. The support set is initiated as $\mathbb{S}_k^0 = \{\frac{\boldsymbol{w}_k}{||\boldsymbol{w}_k||}\}$ and updated by appending L2 normalized point features with reliable pseudo labels, where $\boldsymbol{w}_k$ are the parameters of $g_{\boldsymbol{w}}$ for a category. Afterward, the prototype is updated with the L2 normalized centroid of $\mathbb{S}_k^t$:

$$\boldsymbol{P}_k^t = \frac{1}{|\mathbb{S}_k^t|} \sum_{\boldsymbol{z} \in \mathbb{S}_k^t} \boldsymbol{z}, \tag{1}$$

where $\mathbb{S}_k^t$ collects normalized point features with reliable pseudo label k at each testing step. A point from LiDAR scan $\boldsymbol{X}$ with feature $\boldsymbol{z} \in f_{\boldsymbol{\theta}}(\boldsymbol{X})$ is then classified by $\operatorname{argmax}_k \boldsymbol{z} \cdot \boldsymbol{P}_k^t$. The probability distribution for each point from LiDAR scan $\boldsymbol{X}$ with feature $\boldsymbol{z} \in f_{\boldsymbol{\theta}}(\boldsymbol{X})$ is defined as $\frac{\exp(\boldsymbol{z} \cdot \boldsymbol{P}_k^t)}{\sum_j \exp(\boldsymbol{z} \cdot \boldsymbol{P}_j^t)}$. Reliable pseudo labels are selected by taking points with an Entropy smaller than a preset threshold.

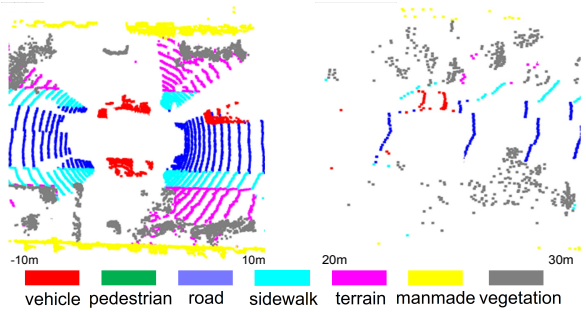


Figure 2. Visualization results of point clouds at varying distances within the same scene. Near-range point clouds (left) and far-range point clouds (right) show a clear difference in density.

3.2. Distance-aware Prototype Learning

Prototype-based TTA constructs a prototype for each category. In this work, we take it further by constructing distance-aware prototypes. Given the close relationship between point cloud density and sensor distance, we observe that point clouds with varying densities exhibit distinguishable distributional disparities when the model faces challenges in representing cross-domain data, as shown in Fig. 2. To address this, we propose a distance-based segmentation strategy, dividing the point cloud into m distinct regions according to density levels. Specifically, we define a set of thresholds $\phi = \{\phi_1, \phi_2, \ldots, \phi_m\}$ with $\phi_1 < \phi_2 < \cdots < \phi_m$ that segment the point cloud based on sensor distance. For a point $\boldsymbol{q} \in \mathbb{R}^3$, if its xy-coordinate $(\boldsymbol{q}_x, \boldsymbol{q}_y)$ satisfies $\phi_j \leq \sqrt{\boldsymbol{q}_x^2 + \boldsymbol{q}_y^2} < \phi_{j+1}$, the point belongs to region j and we denote it as $\Theta_j(\boldsymbol{q})$=1 (otherwise $\Theta_j(\boldsymbol{q})$=0). Each region has its specific prototypes, allowing the model to capture region-aware variations. We update the prototype $\boldsymbol{P}_{i,j}^*$ for each class i in region j over time using an Exponential Moving Average (EMA) [25] with a momentum α as follows:

$$\boldsymbol{P}_{i,j}^{t+1} = \alpha \cdot \boldsymbol{P}_{i,j}^{t} + (1-\alpha) \cdot \frac{\sum_{\boldsymbol{q}_k \in \boldsymbol{X}, \boldsymbol{z}_k \in f_{\boldsymbol{\theta}}(\boldsymbol{X})} \Theta_j(\boldsymbol{q}_k) \cdot \boldsymbol{z}_k}{\sum_{\boldsymbol{q}_k \in \boldsymbol{X}} \Theta_j(\boldsymbol{q}_k)}. \tag{2}$$

Due to domain shifts, predictions are often unreliable, so we have to refine pseudo-labels further. While conventional approaches rely on prediction confidence or entropy for refinement, our experiments show that these metrics can sometimes be biased in cases of large domain shifts. Therefore, we leverage the spatial consistency of K-Nearest Neighbors (KNN) to assess prediction stability across neighboring points. Specifically, we define $\Lambda(\boldsymbol{p})$ as the ratio of neighboring points that share the same pseudo-label as the current point $\boldsymbol{p}$. Then points with consistency $\Lambda(\cdot)$ greater than a threshold are assumed to have reliable pseudo labels, improving the model's adaptability to domain shifts.

3.3. Domain-Specific Decorrelation for Adaption

Despite improved predictions with the DAPL module, the extracted point-wise features from $f_{\boldsymbol{\theta}}$ are still highly correlated among different categories due to the domain gap, especially in a training-free setting. As shown in Fig. 3a(left), we calculate the Pearson correlation matrix between prototypes of different classes and observe a high level of correlations across class features, indicating a tendency for class confusion.

Therefore, we decorrelate extracted features to counterstrike the correlation problem. Following RanPAC [26], we first apply a randomly initialized fixed random projection matrix $\boldsymbol{W} \in \mathbb{R}^{d \times d'}$ to map features into a higher-dimensional space, where d is the original feature dimension and d' is a higher dimension, and then perform decorrelation on both the features and the prototypes to construct better representations. For sample $\boldsymbol{X}^t$ at time step t, we compute the cumulative Gram matrix $\boldsymbol{G}^t$ and the prototypical features $\boldsymbol{C}^t$ as follows:

$$\boldsymbol{G}^t = \sum_{k=1}^{t} \boldsymbol{h}^k \otimes \boldsymbol{h}^k, \quad \boldsymbol{C}^t = \sum_{k=1}^{t} \boldsymbol{h}^k \otimes \boldsymbol{y}^k, \tag{3}$$

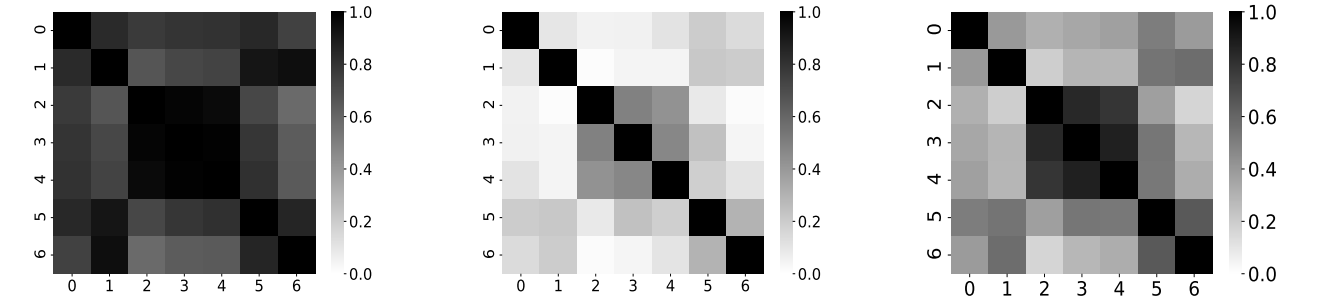
where $\otimes$ means outer products, $\boldsymbol{h}^t = \phi(f_{\boldsymbol{\theta}}(\boldsymbol{X}^t) \cdot \boldsymbol{W})$ represents the projected high-dimensional feature, and ϕ is a nonlinearity function. Note that the prototype C here differs from the prototype P in the DAPL module. P is directly computed from the raw features and is used for pseudo-label optimization, whereas C is derived from decorrelated features to obtain the final prediction. The final prediction $\hat{\boldsymbol{y}}$ for this sample is given by:

$$\hat{\boldsymbol{y}} = \boldsymbol{h}^t (\boldsymbol{G}^t + \lambda \cdot \boldsymbol{I})^{-1} \boldsymbol{C}^t, \tag{4}$$

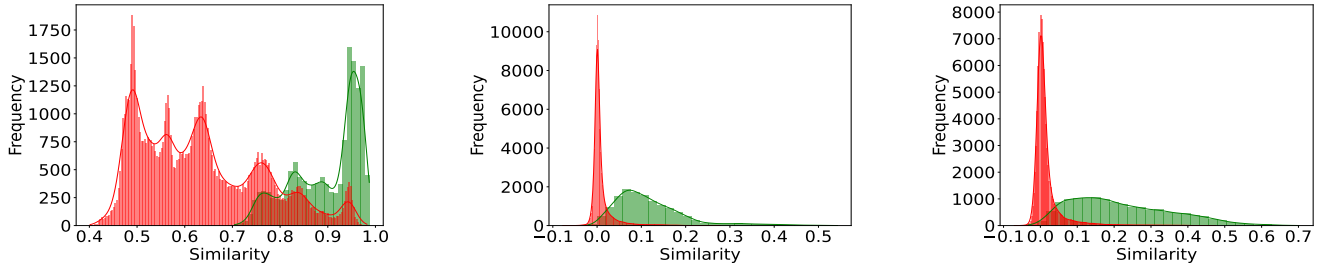
where λ is a regularization to avoid ill-posed Gram matrix. For further details, refer to the derivation in RanPAC.

As shown in Fig. 3a(middle), the decorrelation process significantly reduces the degree of class feature confusion. Additionally, in Fig. 3b, We calculate the cosine similarity distribution between each sample feature and its corresponding class prototype (red) as well as the prototypes of the other $C-1$ classes (green). After decorrelation, the overlap area between these distributions decreases notably, highlighting the reduction in cross-class feature confusion.

To achieve a more precise and stable computation of the decorrelation matrix, we use a cumulative summation approach, rather than an EMA-based accumulation, to calculate the Gram matrix. However, different domains exhibit feature shifts among each other, resulting in differences in their respective decorrelation matrices (see Fig. 4). In light of this cross-domain confusion, we employ separate Gram matrix $\boldsymbol{G}_d^t$ and prototypical features $\boldsymbol{C}_d^t$ for each domain, aiming for more accurate predictions. We distinguish domains using BN statistical parameters similar to previous



(a) Prototype Pearson CCs



(b) Similarity distribution between features and prototypes

Figure 3. Correlations and similarity distributions. (a) Pearson correlation coefficients for 7 class prototypes of "cross sensor" domain. (b) Similarity distribution between all point-wise features and prototypes from the "fog" domain scenes: green indicates similarity with their corresponding class prototype, and red with prototypes from other classes. In both (a) and (b), from left to right: results without decorrelation, results with domain-specific decorrelation (a separate decorrelation matrix for each domain), and results with a global decorrelation matrix (calculated across all domains) applied to all prototypes and features.

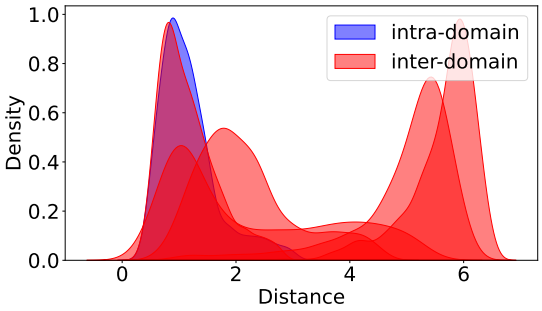


Figure 4. Distance distribution between features and prototypes within the same domain (blue) and across different domains (red) for the "vehicle" class. The intra-domain uses "cross-sensor".

methods [27, 37]. Specifically, we use BN statistical parameters of the last layer with the most downsampling ratio, denoted as $\widehat{\boldsymbol{\mu}}^t, (\widehat{\boldsymbol{\sigma}}^t)^2$ and maintain domain-specific BN parameters $\boldsymbol{\mu}_d^t, (\boldsymbol{\sigma}_d^t)^2$ for each domain d for domain detection. Since autonomous driving involves sequential data, adjacent frames within the same domain tend to exhibit high similarity, allowing us to detect domain shifts by computing the similarity between consecutive frames:

$$c = \text{sim}(\widehat{\boldsymbol{\mu}}^t, \widehat{\boldsymbol{\mu}}^{t-1}), \tag{5}$$

where $\text{sim}(\cdot,\cdot)$ calculates the cosine similarity. If c exceeds a preset threshold τ_c, we assume the domain remains unchanged, and continue updating the domain's decorrelation parameters $(\boldsymbol{G}_d^t, \boldsymbol{C}_d^t)$ and the domain's mean and variance:

$$\boldsymbol{\mu}_d^t = \boldsymbol{\mu}_d^{t-1} + \frac{\widehat{\boldsymbol{\mu}}^t - \boldsymbol{\mu}_d^{t-1}}{t}, \tag{6}$$

$$(\boldsymbol{\sigma}_d^t)^2 = (\boldsymbol{\sigma}_d^{t-1})^2 + \frac{(\widehat{\boldsymbol{\sigma}}^t)^2 - (\boldsymbol{\sigma}_d^{t-1})^2}{t}, \tag{7}$$

with initial values $\boldsymbol{\mu}_d^0 = \widehat{\boldsymbol{\mu}}^0$, $(\boldsymbol{\sigma}_d^0)^2 = (\widehat{\boldsymbol{\sigma}}^0)^2$and $\widehat{\boldsymbol{\mu}}^t$, $(\widehat{\boldsymbol{\sigma}}^t)^2$ are the mean and variance value of BN parameters for the current input.

If the similarity c falls below threshold τ_c, we assume a domain change occurs. To determine if this new domain has been encountered before, we calculate the Wasserstein distance[31] between the BN parameters $(\widehat{\boldsymbol{\mu}}^t, (\widehat{\boldsymbol{\sigma}}^t)^2)$ of the current sample and those of known domains $\{\boldsymbol{\mu}_d^t, (\boldsymbol{\sigma}_d^t)^2\}_{d\in\{1,\cdots,n\}}$, as it effectively reflects the similarity between distributions and offers relatively stable computation. If the minimum Wasserstein distance is below a certain threshold, we assign the sample to the matching domain and use point features for pseudo labels from the DAPL module to update the decorrelation parameters $(\boldsymbol{G}_d^t, \boldsymbol{C}_d^t)$. Otherwise, we initialize a new domain. The domain-specific Gram matrix (c.f. Fig. 3a-middle) provides a significantly more effective decorrelation compared to the global Gram matrix (c.f. Fig. 3a-right).

Table 1. Quantitative comparison of continual test-time adaptation from Synth4D to SemanticKITTI-C. We report the mIoU for each domain as well as the average mIoU across all methods. BP for backpropagation.

Method	DA Type	BP	Domain							mIoU
			Beam	Sensor	Crosstalk	Fog	Echo	Motion	Snow	
PSLabel	TTA	✓	26.32	23.31	25.69	18.42	17.52	17.61	16.22	20.73
TENT[40]	TTA	✓	23.82	14.75	12.40	9.17	4.86	10.21	10.34	12.22
GIPSO[33]	3D TTA	✓	31.49	29.36	26.69	22.83	29.06	16.55	22.89	25.55
EATA[28]	CTTA	✓	28.34	33.01	31.66	25.06	33.72	30.34	21.95	29.15
ViDA[23]	CTTA	✓	31.10	32.72	24.33	20.74	27.52	14.16	20.70	24.47
Source	-	✗	30.92	32.02	23.82	21.12	27.48	13.84	20.52	24.25
T3A[14]	TTA	✗	**31.89**	30.87	23.10	20.97	27.09	13.09	21.02	24.00
Ours	3D CTTA	✗	29.47	**34.56**	**32.36**	**26.45**	**35.34**	**32.67**	**24.92**	**30.82**

Table 2. Quantitative comparison of continual test-time adaptation from Synth4D to nuScenes-C. We report the mIoU for each domain as well as the average mIoU across all methods.

Method	DA Type	BP	Domain							mIoU
			Beam	Sensor	Crosstalk	Fog	Echo	Motion	Snow	
PSLabel	TTA	✓	22.87	20.09	**21.52**	18.00	21.97	20.78	21.18	20.92
TENT	TTA	✓	14.29	6.04	3.87	14.66	10.05	9.37	11.71	10.00
GIPSO	3D TTA	✓	26.01	23.39	18.83	18.24	31.95	23.13	20.05	23.09
EATA	CTTA	✓	25.71	22.90	19.60	19.84	30.64	24.50	23.14	23.76
ViDA	CTTA	✓	**29.12**	25.00	12.78	16.96	**33.59**	12.60	20.30	21.48
Source	-	✗	26.10	**25.01**	12.29	16.75	33.33	11.78	20.03	20.75
T3A	TTA	✗	27.60	23.85	10.74	16.22	32.04	10.40	19.24	20.01
Ours	3D CTTA	✗	26.10	23.15	19.17	**20.11**	31.12	**26.12**	**23.26**	**24.15**

4. Experimental Results

4.1. Task Settings

Datasets and Metrics. We evaluate our method on two benchmarks: Synth4D-nuScenes-C and Synth4D-SemanticKITTI-C. Both benchmarks use Synth4D[33] as the source domain, containing 20K simulated LiDAR point clouds. The target domains, nuScenes-C and SemanticKITTI-C[17], are derived from the real-world 32-beam LiDAR dataset (nuScenes) and 64-beam LiDAR dataset (SemanticKITTI) with different kinds of synthetic corruption to simulate urban driving environments. For the corruption types, we use beam missing, cross-sensor noise, crosstalk, fog, incomplete echo, motion blur, and snow at moderate levels, applying each sequentially as a distinct domain for adaptation, thereby constructing a continuously evolving target domain. To prevent duplicate scenes across different corrupted domains, we divide each dataset into seven equal parts and present them in a fixed order.

Following [33], we adopt mean Intersection over Union (mIoU) as the evaluation metric.

Baselines. We compare our approach with existing TTA and CTTA methods on both benchmarks. 'Source' is the model pre-trained on the source domain without any adaptation. For TTA, we select several approaches: a pseudo-labeling method that selects predictions with entropy below a certain threshold as pseudo-labels to guide model parameter updates using cross-entropy loss, an entropy minimization method called Tent [40], a prototype-based method known as T3A [14], and GIPSO [33] leveraging geometric features and temporal consistency. For CTTA, we include EATA [28], which optimizes BN parameters through a weighted Fisher regularizer, and ViDA [23], an adapter-based approach. Except for the T3A, all the other methods require backpropagation of the source model.

Implementation Details. Following GIPSO, we use MinkowskiNet pre-trained on the Synth4D dataset as the source model. For the DAPL module, we divide the point cloud into three segments by distance range $[0, 50]$ meters and set the EMA momentum to 0.95, K to 20, and the consistency threshold to 0.8. For domain-specific decorrelation for adaption, point-wise features are projected from 96 to 1024 dimensions, with ReLU as the nonlinearity function $\phi(\cdot)$. For cross domain detection, cosine similarity $\text{sim}(\cdot, \cdot)$ is used with a threshold of 0.85. All methods adhere to a strict one-pass protocol.

4.2. Comparisons on Cross-domain CTTA

We compare with the competing methods as shown in Tab. 1. Our method achieves state-of-the-art (SOTA) performance in average mIoU across all domains and mIoU on most individual domains. It improves by 27.1% over the

Table 3. Performance comparison for the repeated domain setup from Synth4D to SemanticKITTI-C. We repeat the target domain sequence three times sequentially and report the average mIoU as well as the mIoU on each domain for each round.

Time	t ⟶		
Round	1	2	3
PSLabel	28.43	27.69	24.74
Tent	17.69	12.86	12.84
GIPSO	26.58	27.21	26.77
EATA	29.35	28.99	28.34
ViDA	25.65	25.60	25.55
Source	25.65	25.65	25.65
T3A	24.98	24.98	24.98
Ours	**31.14**	**31.19**	**31.27**

Table 4. Ablation study on Synth4D-SemanticKITTI-C. We sequentially report the average mIoU across all domains when adding each module: decorrelation, domain detection, prototype learning, distance-aware prototype, and KNN filter.

	Decorrelation	**Domain**	**Prototype**	**Distance**	**KNN**	**Mean**
Exp1	-	-	-	-	-	24.25
Exp2	✓	-	-	-	-	30.17
Exp3	✓	✓	-	-	-	30.32
Exp4	✓	✓	✓	-	-	30.17
Exp5	✓	✓	✓	✓	-	30.44
Exp6	✓	✓	-	-	✓	30.42
Exp7	✓	✓	✓	✓	✓	30.82

source model and 5.7% over the previous SOTA method, both achieved without backpropagation. Methods like PSLabel and TENT perform poorly, likely due to their simplistic reliance on prediction probabilities for pseudo-label selection, which struggles with large domain gaps. TENT's entropy minimization approach appears to further amplify this issue, causing misalignment of the model's updates. In contrast, methods with more refined selection or regularization, such as T3A, GIPSO, and ViDA, achieve moderate improvements. Our approach stands out by offering substantial performance gains across most domains, particularly where the source model performed poorly. For example, our method achieves significant improvements in "crosstalk" (34.56%) and "fog" (26.45%) domains, highlighting its robustness in challenging conditions. While our method experiences a minor performance drop in the "beam missing" domain compared to some baselines, it still consistently surpasses the source model and most previous methods in terms of overall performance.

Tab. 2 presents the results on Synth4D-nuScenes-C, where our method again achieves SOTA average mIoU, exceeding the previous SOTA method by 1.6% and the source model by 16.4%, with SOTA results in three domains. Compared to SemanticKITTI-C, the point clouds in nuScenes are sparser, increasing task difficulty. Consequently, our backpropagation-free approach shows a slightly less pronounced improvement on this benchmark

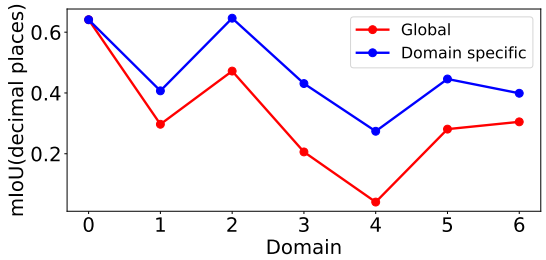


Figure 5. mIoU curves for domain-specific decorrelation (blue) and global decorrelation (red), evaluated on the Synth4D to SemanticKITTI-C setting.

and experiences performance drops in the "cross sensor" and "incomplete echo" domains. Nonetheless, our method consistently demonstrates strong overall effectiveness across most domains and in average performance.

These results underscore our method's capability to adapt across diverse and challenging domain shifts, demonstrating significant improvements in 3D CTTA tasks. More results on the light corruption level are presented in the Appendix. A.1. Results under reverse order are shown in the Appendix. A.2

Performance on Repeated Domains. Tab. 3 presents the performance of different methods on domains that reoccur in a sequence. In this setup, we use three domains, including "cross sensor", "crosstalk", and "fog", repeating them cyclically. Our method achieves SOTA performance under these conditions, with an improvement of 21.4%, 21.6%, 21.9% over the source model and 6.1%, 7.6%, 10.3% over the previous SOTA method in each respective round. Notably, PSLabel, TENT, and EATA experience significant performance drops in the later rounds, suggesting that as they adapt to new domains, the models tend to forget past domains. T3A maintains consistent performance across all rounds, likely due to its conservative prototype update strategy and the lack of model parameter updates. ViDA, by introducing low-rank and high-rank adapters to capture domain-shared and domain-specific knowledge, exhibits only minor degradation in performance. GIPSO shows the most improvement per round, though its overall performance is not better than some competing methods. By retaining unique decorrelation parameters and prototypes for each domain, our approach not only achieves optimal performance but also shows steady improvement over time, highlighting that our approach can effectively retain and utilize prior knowledge. More results on additional repetition rounds (c.f. Appendix. A.3) further demonstrate the improvements of our method.

4.3. Ablations

Effectiveness of Different Modules. We conduct an ablation study on Synth4D-SemanticKITTI-C to evaluate the contribution of each component, including the decorrelation

Table 5. Pseudo label accuracy and mIoU using entropy, probability, and KNN filters. The baseline refers to results without any filtering.

Method	baseline	Ent(30%)	Ent(50%)	Ent(70%)	Prob(30%)	Prob(50%)	Prob(70%)	KNN
Accuracy	54.45	**66.36**	63.83	60.58	66.35	63.81	60.54	60.77
mIoU	30.44	29.38	29.85	30.36	29.34	29.78	30.34	**30.82**

module, domain detection, the DAPL module consisting of prototype and distance partitioning, and KNN filtering, as shown in the Tab. 4. The decorrelation module significantly improves model performance for 24.4%, indicating the importance of this module for the CTTA task. When we introduce domain detection to differentiate parameters across domains, model performance improves further, validating our previous observations. As shown in Fig. 5, Exp2 and Exp3 perform similarly on the first domain, but from the second domain onward, the decorrelation matrix and prototypes of Exp2, which mix information from different domains, experience a noticeable drop in performance. This phenomenon becomes more pronounced with repeated domains. We conducted 13 rounds of testing, and the results show that after adding the domain detection module, the improvement increased from 0.1 to 0.43, detailed data can be found in the Appendix. A.3. And our domain detection module achieves an accuracy of 84.28%, where accuracy is defined as the mean maximum overlap rate per domain. The accuracy is based solely on the corruption-defined domains we established, while feature shift plays a more critical role in performance, influenced by factors such as scene variations. Thus, it serves only as a rough reference for domain detection. The ablation study on the threshold of the domain detection module is presented in Appendix. B.2. Both Exp2 and Exp3 directly use the predictions of the source model as pseudo labels. In Exp4, adding the DAPL prototype slightly decreases performance. We suspect that this is due to excessive incorrect labels causing bias in the prototype update direction, as distance partitioning and pseudo-label filtering are not yet applied. However, as we sequentially add distance partitioning and KNN filtering, the model performance improves further, resulting in an overall increase of 2.1% compared to the initial Exp2 configuration. When the DAPL module is removed, as shown in Exp6, the performance drops from 30.82 to 30.44, highlighting the module's significance.

Effectiveness of the KNN Filter. We test the results of different strategies for selecting pseudo-labels, including entropy, probability, and KNN, examining the resulting pseudo-label accuracy and final mIoU of the model's predictions, as shown in Tab 5. For each of the entropy and probability-based strategies, we select pseudo-labels with thresholds of top 30%, 50%, and 70%, respectively. We observe that all filtering methods significantly improve pseudo-label accuracy. However, both entropy and probability-based selection led to a decrease in model performance. The KNN filter, though providing a relatively small improvement in pseudo-label accuracy, achieves the best overall model performance, underscoring the importance of spatial information in 3D CTTA.

Further examining results for entropy and probability filters reveals that as we apply stricter thresholds, pseudo-label accuracy improves, yet mIoU decreases. This suggests that when a substantial domain gap exists, the distribution of predictions may diverge from the true distribution. As stricter selection thresholds favor more confident predictions, the pseudo-labels increasingly align with the biased distribution center, thereby resulting in performance degradation. Further ablation studies on K values are discussed in Appendix B.1.

5. Conclusion and Discussions

We present a novel domain-aware backpropagation-free approach for 3D CTTA of LiDAR segmentation. Our method, D^{3}CTTA, integrates two key modules: the distance-aware prototype learning module and the domain-specific decorrelation module. With these two modules, we significantly improve the robustness and accuracy of LiDAR segmentation across dynamic and diverse real-world environments. This approach provides a practical solution for adapting models to continuously changing conditions without the need for backpropagation, making it highly suitable for autonomous driving applications. Experimental results across several benchmark settings demonstrate that our method outperforms state-of-the-art approaches.

Limitations and Future Work. While our method achieves state-of-the-art performance, it still degrades in certain domains. Additionally, Geometric features significantly enhance segmentation, yet our approach relies only on a simple distance prior. In future work, leveraging more precise geometric features could further enhance performance. Another challenge lies in constructing unbiased pseudo-labels, which remains an open area for exploration. Our findings indicate that, in the presence of bias, pseudo-label accuracy does not directly correlate with the model's final performance. For instance, selecting the top 30% entropy-based predictions and filtering out incorrect labels with ground truth yielded accurate labels, but did not lead to any performance gain. Conversely, randomly selecting an equivalent number of labels from the ground truth allowed the model to nearly match supervised performance. In this work, we found that directly using KNN is relatively effective for label selection, yet more refined and precise filtering techniques are still needed for further improvement.

Acknowledgement

We thank all the anonymous reviewers for their insightful comments. This work was partially supported by the National Key R&D Program of China (2024YFC3308000), the National Natural Science Foundation of China (62271467, 62476262, 62206263, 62306297, 62306296), Beijing Nova Program, Beijing Natural Science Foundation (4242053, L242096), and China Postdoctoral Science Foundation (2022T150639).

References

[1] Dhanajit Brahma and Piyush Rai. A probabilistic framework for lifelong test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3582–3591, 2023. 1, 2

[2] Haozhi Cao, Yuecong Xu, Jianfei Yang, Pengyu Yin, Shenghai Yuan, and Lihua Xie. Multi-modal continual test-time adaptation for 3d semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18809–18819, 2023. 1, 2

[3] Eduardo R Corral-Soto, Amir Nabatchian, Martin Gerdzhev, and Liu Bingbing. Lidar few-shot domain adaptation via integrated cyclegan and 3d object detector with joint learning delay. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 13099–13105. IEEE, 2021. 2

[4] Eduardo R Corral-Soto, Mrigank Rochan, Yannis Y He, Xingxin Chen, Shubhra Aich, and Liu Bingbing. Domain adaptation in lidar semantic segmentation via hybrid learning with alternating skip connections. In *2023 IEEE Intelligent Vehicles Symposium (IV)*, pages 1–7. IEEE, 2023. 2

[5] Eduardo R Corral-Soto, Alaap Grandhi, Yannis Y He, Mrigank Rochan, and Bingbing Liu. Effects of range-based lidar point cloud density manipulation on 3d object detection. In *2024 IEEE Intelligent Vehicles Symposium (IV)*, pages 3166–3172. IEEE, 2024. 1

[6] Mario Döbler, Robert A Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7704–7714, 2023. 2

[7] François Fleuret et al. Test time adaptation through perturbation robustness. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021. 2

[8] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1705–1714, 2019. 3

[9] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation. *Advances in Neural Information Processing Systems*, 35:27253–27266, 2022. 1, 2

[10] Sachin Goyal, Mingjie Sun, Aditi Raghunathan, and J Zico Kolter. Test time adaptation via conjugate pseudo-labels. *Advances in Neural Information Processing Systems*, 35:6204–6218, 2022. 2

[11] Jordan SK Hu, Tianshu Kuai, and Steven L Waslander. Point density-aware voxels for lidar 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8469–8478, 2022. 1

[12] Qianjiang Hu, Daizong Liu, and Wei Hu. Density-insensitive unsupervised domain adaption on 3d object detection. In *CVPR*, pages 17556–17566. IEEE, 2023. 1, 3

[13] Xuefeng Hu, Gokhan Uzunbas, Sirius Chen, Rui Wang, Ashish Shah, Ram Nevatia, and Ser-Nam Lim. Mixnorm: Test-time adaptation through online normalization estimation. *arXiv preprint arXiv:2110.11478*, 2021. 2

[14] Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 34:2427–2440, 2021. 1, 2, 3, 6

[15] Hyeonseong Kim, Yoonsu Kang, Changgyoon Oh, and Kuk-Jin Yoon. Single domain generalization for lidar semantic segmentation. In *CVPR*, pages 17587–17598. IEEE, 2023. 3

[16] Jaeyeul Kim, Jungwan Woo, Jeonghoon Kim, and Sunghoon Im. Rethinking lidar domain generalization: Single source as multiple density domains, 2024. 3

[17] 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*, pages 19994–20006, 2023. 6

[18] Lingdong Kong, Niamul Quader, and Venice Erin Liong. Conda: Unsupervised domain adaptation for lidar segmentation via regularized domain concatenation. In *IEEE International Conference on Robotics and Automation, ICRA 2023,*, pages 9338–9345. IEEE, 2023. 2

[19] Lingdong Kong, Jiawei Ren, Liang Pan, and Ziwei Liu. Lasermix for semi-supervised lidar semantic segmentation. In *CVPR*, pages 21706–21716. IEEE, 2023. 2

[20] Daeun Lee, Jaehong Yoon, and Sung Ju Hwang. BECoTTA: Input-dependent online blending of experts for continual test-time adaptation. In *Forty-first International Conference on Machine Learning*, 2024. 2

[21] Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[22] Hyesu Lim, Byeonggeun Kim, Jaegul Choo, and Sungha Choi. TTN: A domain-shift aware batch normalization in test-time adaptation. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[23] Jiaming Liu, Senqiao Yang, Peidong Jia, Renrui Zhang, Ming Lu, Yandong Guo, Wei Xue, and Shanghang Zhang. ViDA: Homeostatic visual domain adapter for continual test time adaptation. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 6

[24] Minghua Liu, Yin Zhou, Charles R Qi, Boqing Gong, Hao Su, and Dragomir Anguelov. Less: Label-efficient semantic

segmentation for lidar point clouds. In *ECCV*, pages 70–89. Springer, 2022. 2

[25] Haodong Lu, Dong Gong, Shuo Wang, Jason Xue, Lina Yao, and Kristen Moore. Learning with mixture of prototypes for out-of-distribution detection. In *The Twelfth International Conference on Learning Representations*, 2024. 3, 4

[26] Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems*, 36, 2024. 4

[27] Fahim Faisal Niloy, Sk Miraj Ahmed, Dripta S Raychaudhuri, Samet Oymak, and Amit K Roy-Chowdhury. Effective restoration of source knowledge in continual test time adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2091–2100, 2024. 2, 5

[28] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*, pages 16888–16905. PMLR, 2022. 6, 1

[29] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[30] Xidong Peng, Xinge Zhu, and Yuexin Ma. CL3D: unsupervised domain adaptation for cross-lidar 3d detection. In *AAAI*, pages 2047–2055. AAAI Press, 2023. 2

[31] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40:99–121, 2000. 5

[32] Cristiano Saltori, Fabio Galasso, Giuseppe Fiameni, Nicu Sebe, Elisa Ricci, and Fabio Poiesi. Cosmix: Compositional semantic mix for domain adaptation in 3d lidar segmentation. In *ECCV*, pages 586–602. Springer, 2022. 2

[33] Cristiano Saltori, Evgeny Krivosheev, Stéphane Lathuiliére, Nicu Sebe, Fabio Galasso, Giuseppe Fiameni, Elisa Ricci, and Fabio Poiesi. Gipso: Geometrically informed propagation for online adaptation in 3d lidar segmentation. In *European Conference on Computer Vision*, pages 567–585. Springer, 2022. 1, 2, 6

[34] Jules Sanchez, Jean-Emmanuel Deschaud, and François Goulette. Domain generalization of 3d semantic segmentation in autonomous driving. In *ICCV*, pages 18031–18041. IEEE, 2023. 2

[35] Amirreza Shaban, Joonho Lee, Sanghun Jung, Xiangyun Meng, and Byron Boots. Lidar-uda: Self-ensembling through time for unsupervised lidar domain adaptation. In *ICCV*, pages 19727–19737. IEEE, 2023. 2, 3

[36] Damian Sójka, Sebastian Cygert, Bartłomiej Twardowski, and Tomasz Trzciński. Ar-tta: A simple method for real-world continual test-time adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3491–3495, 2023. 2

[37] Junha Song, K Park, Inkyu Shin, Sanghyun Woo, and In So Kweon. Cd-tta: Compound domain test-time adaptation for semantic segmentation. *arXiv preprint arXiv:2212.08356*, 2022. 5

[38] Zhihang Song, Zimin He, Xingyu Li, Qiming Ma, Ruibo Ming, Zhiqi Mao, Huaxin Pei, Lihui Peng, Jianming Hu, Danya Yao, and Yi Zhang. Synthetic datasets for autonomous driving: A survey. *IEEE Trans. Intell. Veh.*, 9(1): 1847–1864, 2024. 1

[39] Haopeng Sun, Lumin Xu, Sheng Jin, Ping Luo, Chen Qian, and Wentao Liu. PROGRAM: PROtotype GRAph model based pseudo-label learning for test-time adaptation. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[40] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. 1, 2, 6

[41] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211, 2022. 1, 2

[42] Wei Wang, Zhun Zhong, Weijie Wang, Xi Chen, Charles Ling, Boyu Wang, and Nicu Sebe. Dynamically instance-guided adaptation: A backward-free approach for test-time domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24090–24099, 2023. 2

[43] Yi Wei, Zibu Wei, Yongming Rao, Jiaxin Li, Jie Zhou, and Jiwen Lu. Lidar distillation: Bridging the beam-induced domain gap for 3d object detection. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXXIX*, pages 179–195. Springer, 2022. 1

[44] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15922–15932, 2023. 2

[45] Longbin Zeng, Jiayi Han, Liang Du, and Weiyang Ding. Rethinking precision of pseudo label: Test-time adaptation via complementary learning. *Pattern Recognition Letters*, 177: 96–102, 2024. 2

[46] Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *Advances in neural information processing systems*, 35:38629–38642, 2022. 2

[47] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12414–12424, 2021. 1, 2

[48] Yifan Zhang, Xue Wang, Kexin Jin, Kun Yuan, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Adanpc: Exploring non-parametric classifier for test-time adaptation. In *International Conference on Machine Learning*, pages 41647–41676. PMLR, 2023. 1, 2

[49] Bowen Zhao, Chen Chen, and Shu-Tao Xia. DELTA: DEGRADATION-FREE FULLY TEST-TIME ADAPTA-

TION. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[50] Yao Yao Xiaoqing Feng Juntao Gao Zhijiang Qiu, Lin Zhang. A survey on semantic segmentation in 3d point cloud scenes. In *Journal of Image and Graphics*, pages 1–17, 2024. 2

[51] Tianpei Zou, Sanqing Qu, Zhijun Li, Alois Knoll, Lianghua He, Guang Chen, and Changjun Jiang. HGL: hierarchical geometry learning for test-time adaptation in 3d point cloud segmentation. *ECCV*, abs/2407.12387, 2024. 2