Seeing the Unseen: Geometry-Aware Test-Time Adaptation for LiDAR Segmentation

Yewon Song
Automotive Engineering
Hanyang University
Seoul, Korea
swy1155@hanyang.ac.kr

Soonmin Hwang*
Automotive Engineering
Hanyang University
Seoul, Korea
soonminh@hanyang.ac.kr

***Abstract*—LiDAR semantic segmentation suffers substantial degradation when deployed in unseen environments, where geometric distortions arise from both extrinsic misalignment and intrinsic sensor characteristics. We show that these overlooked factors significantly alter the representation of LiDAR measurements and are major contributors to real-world performance failures. Although domain adaptation has made progress in narrowing domain gaps, existing approaches still struggle when encountering entirely unseen scenes without labels. While test-time adaptation has recently emerged as a promising alternative to address label-free deployment in 2D vision, its application to 3D LiDAR perception remains underexplored, leaving current models vulnerable to distribution shifts. To address this gap we introduce two geometry-aware TTA modules tailored to LiDAR's properties, designed to compensate for one of these key variation sources. Our approach improves mIoU from 16.27 to 18.44 (+13.34%), demonstrating substantial gains in robustness under unseen sensor configurations and environments.**

***Index Terms*—LiDAR Semantic Segmentation, Autonomous Driving, Domain Adaptation, Point Clouds, Test-time Adaptation**

I. Introduction

LiDAR point clouds provide robust distance measurements and rich 3D geometry even in challenging conditions, making LiDAR semantic segmentation (LSS) a key component of scene understanding for autonomous driving and robotics. However, a significant performance gap remains between training and real-world deployment due to distribution shifts in LiDAR data. This challenge has increased interest in unsupervised domain adaptation (UDA), which allows a source-trained model to adapt to a target domain without labels.

In real-world LiDAR scenarios, the domain gap mainly arises from two factors: 1) shifts in depth distribution due to vertical extrinsic variations and 2) changes in point density caused by different intrinsic configurations.

The first factor arises from the disruption of the inherent vertical layering when the sensor height changes. As shown in Fig. 1, the lower beams predominantly capture ground surfaces, the middle beams tend to capture objects such as vehicles and pedestrians, and the upper beams generally observe higher structures such as building facades or the upper parts of vegetation. When the sensor is elevated along the z-axis, the depth distribution becomes dominated by farther points. As a result, the semantic categories learned during training no longer align at test time, producing a severe mismatch between beam index and semantic content. Although vertical viewpoint misalignment is a critical source of geometric distortion, prior work has largely overlooked its impact, often treating sensor height and pitch as fixed and inconsequential factors.

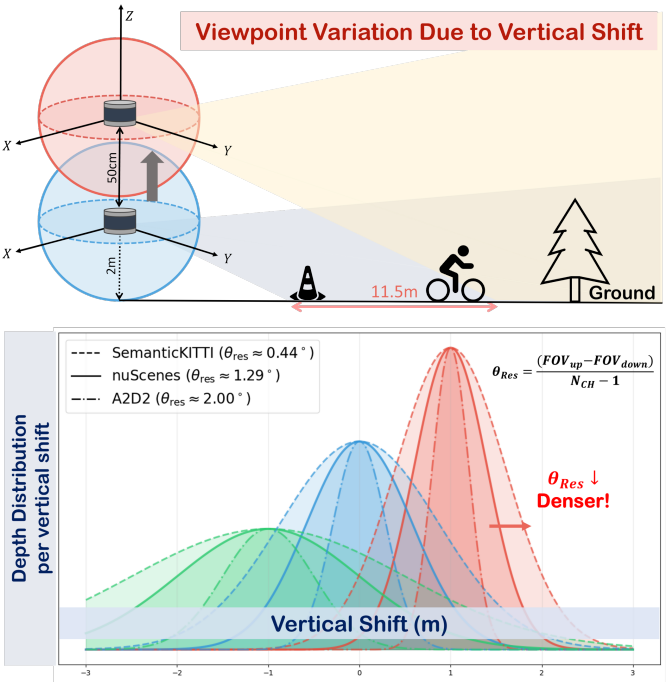

Fig. 1: **Impact of vertical shift and sensor resolution on LiDAR geometry.** (Top) Changing the LiDAR height alters the vertical viewpoint and disrupts the semantic layering of beams. (Bottom) Depth distributions from SemanticKITTI, nuScenes, and A2D2 show how vertical angular resolution (θ_{res}) controls point density. Smaller θ_{res} yields denser sampling.

The second factor originates from discrepancies in point density caused by intrinsic configurations. Different LiDAR sensors may employ varying numbers of vertical channels, distinct vertical fields of view, and different vertical angular resolutions. Vertical angular resolution θ_{res} —defined as the angular spacing between adjacent LiDAR beams—determines how finely the sensor can capture structures at different

heights. It depends on how many channels cover the vertical FoV. More channels make the beams closer together and give denser points. Fewer channels make the sampling much sparser.

For example, nuScenes [1] uses a 32-beam LiDAR with a vertical FoV of about 40°, yielding a vertical resolution of approximately 1.29°. Similarly, SemanticKITTI [2] employs a 64-beam LiDAR with a 26.9° vertical FoV, resulting in a finer resolution of about 0.44°. The A2D2 dataset [3] uses an even sparser 16-beam LiDAR with a vertical FoV of roughly 30°, corresponding to a much coarser resolution of about 2.0°. As illustrated in Fig. 1, a smaller vertical angular resolution leads to a noticeably denser point cloud.

Recently, unsupervised domain adaptation (UDA) methods for LiDAR point clouds have been introduced to address performance degradation in target domains. Most existing approaches [4]–[6] rely on augmentations or pre-processing that approximate target conditions, such as adverse weather simulation or point-density modification. CoSMix [7] and UniMix [8] mitigate the domain gap by mixing source and target features, effectively simulating target-domain geometry. However, both methods assume full prior access to the target domain distribution—a common but restrictive limitation of UDA approaches.

To avoid dependence on explicit target information, recent work has shifted toward domain-generalization (DG) strategies. DGLSS [9] adopts random beam dropping under an assumed lower-spec setting, while DDFE [10] leverages intra-domain density variation to promote cross-dataset consistency. However, these techniques treat sensor properties in isolation and rely heavily on implicit assumptions about the target environment, even though sensor characteristics such as height, FOV, and density are strongly interdependent. As a result, they often become over-regularized, suppressing domain-specific geometric cues that are crucial for reliable adaptation and ultimately limiting their effectiveness in truly unseen environments.

These limitations highlight that simply regularizing the source model is insufficient when the target domain exhibits systematic geometric deviations. In such scenarios, the model must instead interpret the target sensor's geometry at inference time rather than relying on prior assumptions. This naturally motivates a source-free test-time adaptation approach, where the model adjusts itself based solely on the incoming target data.

In this work, we revisit the problem of LiDAR domain shift from the perspective of *extrinsic* and *intrinsic* variation. Our analysis shows that z-axis shift in sensor (extrinsic) and vertical angular resolution (intrinsic) induce systematic distortions in the geometric structure of range-view representations, resulting in severe performance degradation when models are deployed outside their training configuration. To address these challenges, we propose a source-free test-time adaptation framework that explicitly estimates and compensates for both factors using only the first mini-batch of the target domain.

Our contributions are summarized as follows:

- We introduce the first source-free TTA framework that explicitly models and corrects both extrinsic and intrinsic LiDAR variations during inference.
- We identify the fundamental cause of vertical-shift–induced degradation in LiDAR semantic segmentation and introduce VSAM, which estimates target sensor height from the first mini-batch and corrects the disrupted vertical semantic structure.
- We propose VARS, which infers the target sensor's vertical resolution online and re-samples features via elevation-aware interpolation to handle intrinsic density differences across sensors.

II. Related Work

A. Domain Generalization for LSS

Domain generalization (DG) aims to generalize the model to distribution-shifted target domains by learning domain-invariant features at the training stage. For point cloud segmentation, DG methods augment the source domain to simulate unseen target domains, e.g., by randomly subsampling LiDAR scans. 3DLabelProp [11] relies on exploiting the geometry and sequence of LiDAR data to improve its generalization performance by partially accumulated point clouds. LiDOG [12] introduces an additional BEV auxiliary task to learn robust features. Although DG can generalize to multiple domains, its inability to leverage data from application scenarios results in sub-optimal performance.

B. Unsupervised Domain Adaptation for LSS

Domain Adaptation (DA) focuses on reducing the performance gap between the labeled source domain and the unlabeled target domain by exploiting only the raw target data during training. To achieve this, Complete & Label [13] mitigates discrepancies caused by different LiDAR configurations by completing point clouds from both domains onto a shared 3D surface before segmentation. ePointDA [14] performs domain transfer through a Generative Adversarial Network that injects realistic noise into synthetic data. LiDAR-UDA [15] enhances pseudo-label quality through LiDAR beam subsampling and cross-frame ensembling for self-supervised adaptation. Despite these advances, concurrent access to both source and unlabeled target data remains a limitation for real-world deployment.

C. Test-Time Adaptation

Test-Time Adaptation (TTA) aims to adapt models to unseen test distributions by mitigating domain shifts that arise between training and deployment. A seminal work in this area, Tent [16], minimizes prediction entropy by updating batch normalization parameters during inference. Following this idea, numerous methods extend TTA by optimizing a small subset of parameters with different strategies.

For example, EATA [17] selects reliable and non-redundant samples for optimization, while DUA [18] introduces adaptive momentum through a novel normalization layer. NOTE [19] performs instance-wise batch norm updates,

whereas RoTTA [20] and DELTA [21] leverage global statistics. Sharpness-aware optimization has also been adopted, as in SoTTA [22] and SAR [23]. Beyond BN-based updates, several methods adapt the entire network using a mean-teacher framework for stable supervision generate pseudo labels for self-training [24], [25], cluster features to refine adaptation [26]–[28], or utilize augmentations to improve robustness [29].

While most prior efforts target image classification, MemCLR [30] is the first work to apply TTA to 2D object detection, leveraging a mean-teacher structure to align instance-level features. However, the applicability of these image-based TTA approaches to 3D point cloud perception remains largely unexplored.

III. Method

A. Vertical Shift-Aware Adaptation Module

To infer the geometric distribution of the unseen target domain, the proposed Vertical Shift-Aware Adaptation Module (VSAM) analyzes the first mini-batch and estimates its vertical shift. This initial estimation provides a reliable measure of the z-axis displacement characterizing the target domain, allowing the model to capture the underlying geometric shift that will persist across subsequent test samples. VSAM performs this estimation by first determining the LiDAR's relative height from the ground in the target domain.

To obtain this height, VSAM applies RANSAC to the incoming 3D point cloud and robustly fits the ground plane:

$$\pi : ax + by + cz + d = 0, \tag{1}$$

The perpendicular distance from the sensor origin to the plane gives the test-time height:

$$h_{\text{tgt}} = \frac{|d|}{\sqrt{a^2 + b^2 + c^2}}, \qquad \Delta h = h_{\text{tgt}} - h_{\text{src}}. \tag{2}$$

A canonical training-time height h_{src} is computed similarly, and the difference Δh represents the vertical domain shift.

This displacement provides a direct measurement of how much the vertical semantic layering has shifted in the target domain. VSAM compensates for this geometry-induced distortion by vertically warping the intermediate range-view feature map:

$$F'(c, h, w) = F(c,\ h - \alpha\Delta h,\ w)\,, \tag{3}$$

where α converts metric height units to elevation-index units based on the vertical angular resolution.

To further refine this correction, VSAM generates height-dependent FiLM [31] parameters:

$$\gamma_h,\ \beta_h = \text{MLP}(\Delta h), \tag{4}$$

and applies row-wise modulation:

$$\tilde{F}(c, h, w) = \gamma_h\, F'(c, h, w) + \beta_h. \tag{5}$$

Since the shift Δh is estimated only once from the first mini-batch—capturing the global geometric bias of the target domain—it provides a strong source-free prior for subsequent test-time adaptation. Adjusting this mismatch with Δh, the module restores the proper vertical structure and preserves segmentation performance under unseen sensor heights.

B. Vertical Angular Resolution–Aware Sampling

Since the intrinsic vertical angular resolution of the target LiDAR is unknown in the source-free test-time setting, VARS first estimates it directly from the first mini-batch. For each point $p = (x, y, z)$, we compute its elevation angle

$$\phi(p) = \arctan 2\Big(z,\ \sqrt{x^2 + y^2}\Big)\,, \tag{6}$$

sort all angles, and estimate the target sensor's effective vertical resolution as the median spacing:

$$\theta^{\text{tgt}}_{\text{res}} = \text{median}_i\big(\phi_{i+1} - \phi_i\big). \tag{7}$$

From this estimate, VARS constructs a *target elevation grid* $\{\theta^{\text{tgt}}(k)\}$ where k denotes the index of the k-th target elevation row. The spacing between consecutive grid points is set to $\theta^{\text{tgt}}_{\text{res}}$, reflecting how densely the target sensor samples the vertical axis. In contrast, the source-trained model expects features aligned to the *source elevation grid* $\{\theta^{\text{src}}(h)\}$ with resolution $\theta^{\text{src}}_{\text{res}}$, where h indexes the source beam rows.

To reconcile this mismatch, each target elevation angle $\theta^{\text{tgt}}(k)$ is aligned to the source grid through elevation-aware linear interpolation. We first find two adjacent source elevations $\theta^{\text{src}}(h_1)$ and $\theta^{\text{src}}(h_2)$ such that

$$\theta^{\text{src}}(h_1)\ \le\ \theta^{\text{tgt}}(k)\ \le\ \theta^{\text{src}}(h_2), \tag{8}$$

and compute the interpolation weight

$$\lambda = \frac{\theta^{\text{tgt}}(k) - \theta^{\text{src}}(h_1)}{\theta^{\text{src}}(h_2) - \theta^{\text{src}}(h_1)}. \tag{9}$$

Here λ represents how far the target elevation lies between the two source elevations (i.e., a normalized position between rows h_1 and h_2). The resolution-adjusted feature for the k-th target row is then obtained as

$$F^{\text{VARS}}(c, k, w) = (1 - \lambda)\, F(c, h_1, w) + \lambda\, F(c, h_2, w). \tag{10}$$

This interpolation effectively re-samples the source-trained feature map onto the inferred target sampling grid. As a result, the backbone operates with a feature representation whose vertical density is consistent with that of the target sensor. Because VARS relies solely on the first mini-batch and requires no access to source data, it is fully compatible with the source-free test-time adaptation setting.

C. Test-Time Parameter Update

After geometric alignment by VSAM and VARS, the model receives a feature map F_{geo} that reflects the estimated extrinsic and intrinsic properties of the target LiDAR. TTA is then performed by updating a small set of parameters using only the incoming unlabeled target samples.

Following prior source-free TTA methods, we adopt prediction entropy as the self-supervised adaptation signal. Given the segmentation logits $Z = H(F_{\text{geo}})$ and probability map $P = \text{Softmax}(Z)$, the entropy loss is defined as:

$$\mathcal{L}_{\text{ent}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} P_{i,c} \log P_{i,c}, \tag{11}$$

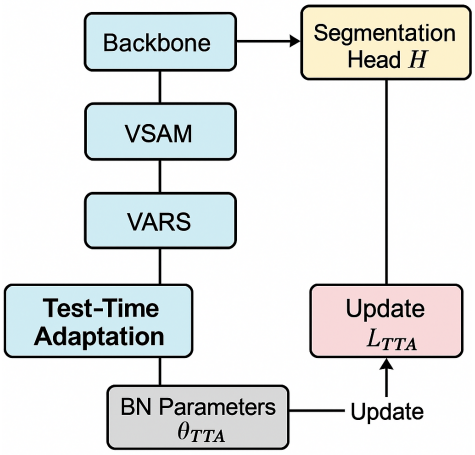

Fig. 2: **Test-time adaptation pipeline.** VSAM corrects extrinsic height shift, VARS aligns intrinsic angular resolution, and the geometry-adjusted features update BN parameters via entropy minimization.

where N is the number of valid pixels and C is the number of classes. Minimizing (11) encourages confident and stable predictions in the target domain.

To preserve the source-trained representation while enabling on-the-fly adaptation, we restrict updates to batch-normalization (BN) affine parameters:

$$\theta_{\mathrm{TTA}} = \{\gamma^{(\ell)}, \beta^{(\ell)}\}_{\ell \in \mathcal{B}},$$

where $\mathcal{B}$ is the set of BN layers. The update rule is:

$$\theta_{\mathrm{TTA}}^{(t+1)} = \theta_{\mathrm{TTA}}^{(t)} - \eta \nabla_{\theta_{\mathrm{TTA}}} \mathcal{L}_{\mathrm{ent}}, \tag{12}$$

with learning rate η.

In addition to updating affine parameters, BN running statistics are incrementally refined using the target batch:

$$\mu_{\mathrm{BN}}^{(t+1)} = (1-\alpha)\,\mu_{\mathrm{BN}}^{(t)} + \alpha\,\hat{\mu}, \tag{13}$$

$$\sigma_{\mathrm{BN}}^{2(t+1)} = (1-\alpha)\,\sigma_{\mathrm{BN}}^{2(t)} + \alpha\,\hat{\sigma}^2, \tag{14}$$

where $(\hat{\mu}, \hat{\sigma}^2)$ are batch statistics and α is a momentum coefficient.

This design ensures that adaptation operates on a geometry-aligned feature representation, enabling stable and effective source-free TTA under diverse target sensor configurations.

D. Overall Architecture

At test time 2, we apply VSAM and VARS sequentially: VSAM first corrects the vertical semantic shift caused by extrinsic height differences, and VARS then aligns the intrinsic vertical resolution by re-sampling features onto the target elevation grid. This ordering reflects the physical relationship between height-induced layer displacement and resolution-induced sampling density.

IV. Experiments

A. Datasets

We focus on [2], [3], and [1], which differ significantly in LiDAR specifications (number of channels, vertical FoV, and sampling patterns). This diversity enables a controlled study of domain shift and sensor-dependent variations.

[2] is a large-scale benchmark for LiDAR semantic segmentation, derived from the KITTI odometry sequences and captured using a Velodyne HDL-64E (64-channel). It provides dense point-wise annotations for over 4.5 billion points, making it a standard reference for evaluating outdoor LSS.

[3] is a multi-modal dataset recorded with multiple cameras and five Velodyne VLP-16 (16-channel) LiDARs. Semantic labels are annotated on 2D images and projected onto 3D, offering consistent multimodal supervision for segmentation.

[1] is a comprehensive autonomous-driving benchmark comprising 1000 scenes collected with a Velodyne HDL-32E (32-channel) LiDAR. It includes full-sweep LiDAR point clouds, object annotations, and semantic labels, covering diverse urban environments, lighting conditions, and weather scenarios, making it a strong testbed for robustness and cross-domain evaluation.

B. Settings

SemanticKITTI [2], nuScenes-lidarseg [1], and A2D2 [3] adopt different label taxonomies with varying granularity, making direct cross-dataset evaluation inconsistent. To establish a unified training and evaluation setup, we identify semantically aligned categories across the three datasets and merge fine-grained or dataset-specific labels into broader groups. Examples include mapping `car/van/bus` into `car` and consolidating `road/street/drivable` into `drivable-surface`, while unmatched labels are assigned to a `ignore` class.

This process yields a unified set of ten common classes: {car, bicycle, motorcycle, truck, other-vehicle, pedestrian, drivable-surface, sidewalk, walkable, vegetation}, enabling consistent cross-dataset evaluation under heterogeneous LiDAR sensors.

C. Implementation Details

For all experiments, we employ EPMF [32] as our baseline segmentation model. EPMF adopts a perception-aware multi-sensor fusion architecture that projects LiDAR points onto camera views to incorporate complementary RGB cues, enabling robust feature learning under heterogeneous sensor configurations.

D. Results

Table II summarizes the cross-domain mIoU when models trained on one dataset are evaluated on others. As expected, each model performs best when tested on its own domain (e.g., KITTI→KITTI, NuScenes→NuScenes, A2D2→A2D2). However, transferring across datasets leads to a severe performance collapse: KITTI-trained models drop from 74.95% in-domain to 13.51% on NuScenes and 11.48% on A2D2,

TABLE I: Quantitative results per class
(Source: Semantic KITTI)

Target	Car	Bicycle	Motorcycle	Truck	Other vehicle	Pedestrian	Drivable Surf.	Sidewalk	Terrain	Vegetation	mIoU (%)
SemanticKITTI	94.35	67.88	47.29	58.94	68.32	67.06	95.86	82.66	75.98	91.15	74.95
A2D2	43.88	39.42	0.00	12.88	0.00	24.82	1.10	0.85	0.22	39.44	16.27
A2D2 (+Ours)	**47.08**	17.93	–	2.32	–	9.54	**64.00**	2.5	0.8	**40.30**	**18.44** (+13.34%)

Notes. Source = training dataset, Target = evaluation dataset.

TABLE II: Performance Gap Across Domains.

Source \ Target	KITTI (64-ch)	NuScenes (32-ch)	A2D2 (16-ch)
	mIoU (%)		
KITTI	74.95	13.51	11.48
NuScenes	20.48	67.82	10.27
A2D2	16.27	20.64	53.56

Notes. KITTI denotes *SemanticKITTI*; A2D2 denotes *Audi A2D2*. "Source" = no adaptation.

while A2D2-trained models similarly degrade when evaluated on KITTI (16.27%). These results highlight the substantial geometric and density discrepancies across LiDAR sensors (64/32/16 beams), making cross-domain generalization highly challenging without targeted adaptation.

Table I summarizes per-class IoU when training on SemanticKITTI and evaluating on both SemanticKITTI and A2D2. As expected, in-domain evaluation (SemanticKITTI→SemanticKITTI) yields consistently high IoUs across most classes, with *Car*, *Drivable Surface*, and *Vegetation* achieving over 90%. However, when transferring to A2D2, performance drops sharply across nearly all categories due to severe cross-sensor and cross-domain discrepancies. Highly structured classes such as *Car* degrade from 94.35% to 43.88%, while texture-sensitive or long-tail categories such as *Motorcycle*, *Other Vehicle*, and *Terrain* collapse to nearly 0%.

Applying our method on the A2D2 target domain yields consistent improvements in several key categories. Notably, *Car* improves from 43.88% to 47.08%, and *Drivable Surface* increases substantially from 1.10% to 64.00%, highlighting the benefit of geometric alignment under mismatched sensor configurations. Although certain underrepresented classes remain challenging (e.g., *Motorcycle* and *Other Vehicle*), the overall mIoU rises from 16.27% to 18.44%, demonstrating that our geometric test-time adaptation effectively mitigates cross-sensor degradation.

V. Conclusion

This work examined how geometric variations—specifically vertical extrinsic shifts and intrinsic differences in sensor resolution—introduce systematic distortions in LiDAR data, ultimately degrading semantic segmentation performance in unseen environments. To address these challenges, we proposed a source-free test-time adaptation framework that explicitly estimates and compensates for both forms of geometric variation using only incoming target data. VSAM corrects vertical semantic misalignment by inferring sensor height, while VARS adapts to resolution differences through elevation-aware feature re-sampling.

By enabling models to reinterpret target-domain geometry at inference time, this study highlights the importance of geometry-aware adaptation for reliable LiDAR perception. Moving forward, we aim to extend this direction beyond cross-environment shifts to encompass broader real-world scenarios, including adverse weather and other challenging conditions where robust test-time adaptation is essential.

VI. Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.RS-2024-00409492).

References

[1] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11621–11631, 2020.

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

[3] J. Geyer, Y. Kassahun, M. Mahmudi, X. Ricou, R. Durgesh, A. S. Chung, L. Hauswald, V. H. Pham, M. Mühlegg, S. Dorn, T. Fernandez, M. Jicke, and A. Tarabay, "A2d2: Audi autonomous driving dataset," 2020. arXiv preprint arXiv:2004.06320.

[4] J. Li, S. Zhang, and M. Yang, "Geodesic correlation alignment for point cloud domain adaptation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[5] Z. Qin, C. You, K. Fu, N. Qiao, C. Li, Z. Chen, and N. Zheng, "Unsupervised domain adaptation for lidar segmentation via surface completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7203–7212, 2021.

[6] M. Jaritz, T.-H. Vu, N. Girard, J. Ponce, and M. Aubry, "Learning sensor simulation for domain adaptation of lidar perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7319–7329, 2020.

[7] C. Saltori, M. Cerman, S. Lathuilière, G. Fiameni, F. Galasso, N. Sebe, E. Ricci, and F. Poiesi, "Cosmix: Compositional semantic mix for domain adaptation in 3d lidar segmentation," in *European Conference on Computer Vision (ECCV)*, 2022.

[8] H. Zhao, J. Zhang, Z. Chen, S. Zhao, and D. 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 (CVPR)*, 2024.

[9] H. Kim, Y. Cho, and J. Kim, "Single domain generalization for lidar semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[10] J. Kim, Y. Zhou, J. A. Ansari, and S. Liu, "Rethinking lidar domain generalization: Single source as multiple lidar sensors," in *European Conference on Computer Vision (ECCV)*, 2024.

[11] J. Sanchez, J.-E. Deschaud, and F. Goulette, "Domain generalization of 3d semantic segmentation in autonomous driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18077–18087, 2023.

[12] C. Saltori, A. Ošep, E. Ricci, and L. Leal-Taixé, "Walking your lidog: A journey through multiple domains for lidar semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 196–206, 2023.

[13] L. Yi, B. Gong, and T. Funkhouser, "Complete & label: A domain adaptation approach to semantic segmentation of lidar point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15363–15373, 2021.

[14] S. Zhao, Y. Wang, B. Li, B. Wu, Y. Gao, P. Xu, T. Darrell, and K. Keutzer, "epointda: An end-to-end simulation-to-real domain adaptation framework for lidar point cloud segmentation," in *AAAI Conference on Artificial Intelligence*, pp. 2443–2451, 2021.

[15] A. Shaban, J. Lee, S. Jung, X. Meng, and B. Boots, "Lidar-uda: Self-ensembling through time for unsupervised lidar domain adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19784–19794, 2023.

[16] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *International Conference on Learning Representations (ICLR)*, 2021.

[17] S. Niu, J. Wu, Y. Zhang, Y. Wang, W. Guo, Z. Zhao, J. Huang, and T. Yao, "Efficient test-time model adaptation without forgetting," in *International Conference on Machine Learning (ICML)*, 2022.

[18] M. J. Mirza, J. Micarek, H. Possegger, and H. Bischof, "The norm must go on: Dynamic unsupervised domain adaptation by adaptive normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[19] T. Gong, J. Jeong, T. Kim, Y. Kim, J. Shin, and S.-J. Hwang, "Note: Robust test-time adaptation against temporal correlation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[20] L. Yuan, B. Xie, and S. Li, "Robust test-time adaptation in dynamic scenarios," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[21] B. Zhao, C. Chen, and S.-T. Xia, "Delta: Degradation-free fully test-time adaptation," *arXiv preprint arXiv:2301.13018*, 2023.

[22] R. Gong, Y. Zhang, X. Wei, Y. Guo, S. Niu, H. Ruan, and X. Wang, "Sotta: Robust test-time adaptation on noisy data streams," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[23] S. Niu, J. Wu, Y. Zhang, Z. Gong, W. Guo, Z. Zhao, J. Huang, and T. Yao, "Towards stable test-time adaptation in dynamic wild world," in *International Conference on Learning Representations (ICLR)*, 2023.

[24] A. Goyal, M. Sun, A. Raghunathan, and J. Z. Kolter, "Test-time adaptation via self-training with nearest neighbor information," *arXiv preprint arXiv:2207.09640*, 2022.

[25] Y. Zeng, Y. Zhang, X. Wei, J. Feng, and X. Wang, "Progressive label refinement for test-time adaptation," *arXiv preprint arXiv:2402.04562*, 2024.

[26] Y. Chen, C. Wang, and H. Li, "Cluster-guided adaptation for test-time learning," in *European Conference on Computer Vision (ECCV)*, 2022.

[27] J. Jung, D. Lee, and H. Nam, "Clutta: Cluster-aware test-time adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[28] H. Wang, S. Liu, and M.-H. Yang, "Feature clustering for reliable test-time adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[29] B. Zhang, Y. Niu, and D. Li, "Augmentation-based test-time robustness," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[30] M. V S, P. Oza, and V. M. Patel, "Memclr: Memory-guided contrastive test-time adaptation for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[31] E. Pérez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with feature-wise linear modulation," in *International Conference on Learning Representations (ICLR)*, 2018.

[32] M. Tan, Z. Zhuang, S. Chen, R. Li, K. Jia, Q. Wang, and Y. Li, "Epmf: Efficient perception-aware multi-sensor fusion for 3d semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 8258–8273, 2024.