Construct to Associate: Cooperative Context Learning for Domain Adaptive Point Cloud Segmentation

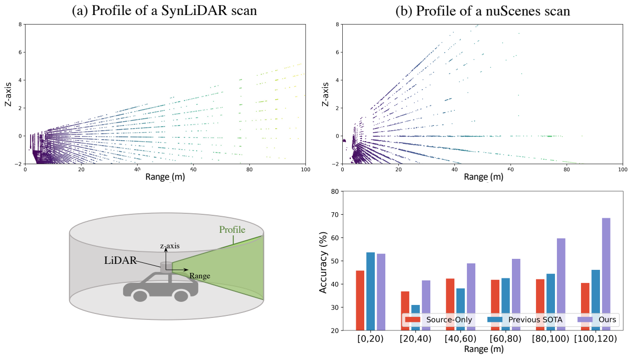
Guangrui Li
ReLER, AAII, University of Technology Sydney
guangrui.li@outlook.com

Abstract

This paper tackles the domain adaptation problem in point cloud semantic segmentation, which performs adaptation from a fully labeled domain (source domain) to an unlabeled target domain. Due to the unordered property of point clouds, LiDAR scans typically show varying geometric structures across different regions, in terms of density, noises, etc, hence leading to increased dynamics on context. However, such characteristics are not consistent across domains due to the difference in sensors, environments, etc, thus hampering the effective scene comprehension across domains. To solve this, we propose Cooperative Context Learning that performs context modeling and modulation from different aspects but in a cooperative manner. Specifically, we first devise context embeddings to discover and model contextual relationships with close neighbors in a learnable manner. Then with the context embeddings from two domains, we introduce a set of learnable prototypes to attend and associate them under the attention paradigm. As a result, these prototypes naturally establish long-range dependency across regions and domains, thereby encouraging the transfer of context knowledge and easing the adaptation. Moreover, the attention in turn attunes and guides the local context modeling and urges them to focus on the domain-invariant context knowledge, thus promoting the adaptation in a cooperative manner. Experiments on representative benchmarks verify that our method attains the new state-of-the-art.

1. Introduction

Point cloud semantic segmentation aims to classify points in a LiDAR scan into the predefined semantic classes. Recently, tremendous efforts [5, 14, 31–33, 48, 51, 58] have been devoted to this task for its fundamental role in environmental perception, *e.g.*, autonomous driving, virtual reality, *etc*. Albeit the recent progresses, most current solutions still assume an ideal environment with massive point-wise annotations accessible. However, such an assumption cannot be always satisfied in real world as we cannot foresee and annotate all the scenes encountered, especially with the expensive annotation cost. Thus, a model may fail to cope with novel scenes under the distribution shift induced by various factors, *e.g.*, sensor configurations, weather. To solve this, domain adaptation [8, 24, 26, 30, 34, 37] is considered a feasible solution which seeks to adapt the model trained on one labeled domain to a novel domain without annotation.



(a) Profile of a SynLiDAR scan

(b) Profile of a nuScenes scan

(c) LiDAR scanning process

(d) Accuracy along the range

Figure 1. (a)(b): Profiles of LiDAR scans from two datasets that gather over 360°. (d): Accuracy of SynLiDAR→nuScenes. Along the increasing range, two datasets exhibit apparent discrepancies in the scene geometry, *e.g.*, density, the angle of beams, thus leading to the context gap. Previous solutions mainly improve the adaptation at the regions with similar geometric structures (range $<$ 20m) but become less effective at distant regions. Instead, our method yields better results consistently with explicit context modeling and modulation across domains.

Domain adaptation for semantic segmentation has been widely investigated on 2D images [4, 11, 13, 15, 22, 55], while only a few attention [23, 35, 59] has been paid to the 3D scenario. A critical difference between 2D images and point clouds lies in the organization of the scene, *i.e.*, 2D images are organized with rigid grids while point clouds are in an unordered manner. Thus, LiDAR scans collected from different sensors or environments typically show different characteristics on the scene geometry. As shown in

Fig. 1 (a)(b), two datasets show different characteristics in scene structures, *e.g*., angle of beams, sparsity, *etc*. Such a discrepancy naturally results in the distribution shift in the geometric context, *i.e*., the geometric relationships with surrounding neighbors, posing a new challenge to the domain adaptation problem However, previous 2D solutions typically consider the scene context with the appearance feature of images, hence being hardly applicable to the 3D scenario.

Recently, increasing attention has been paid to the domain adaptation problem for point cloud segmentation. The initial researches identify the dropout noises as the main obstacle for performing adaptation, where the noises are caused by varying reflectivity of different materials. To solve this, Wu *et al*. [45] proposes to perform noise rendering with the collected noise frequencies in the projected range view. Zhao *et al*. [59] proposes to enhance the noise rendering process with CycleGAN [60]. A more recent work, CoSMix [35], proposes an augmentation technique that mixes the LiDAR scans from two domains in a semantic-aware manner. However, these solutions commonly adopt the input transformation that treats each scene as a whole but neglects the spatial variation of contexts inside the scenes. Thus, without explicit context modeling, it is questionable if they can well handle regions under severe shifts in scene context, especially in the absence of target supervision. For example, in Fig. 1, previous solutions mainly benefit the adaptation of regions less suffered from the context gap (range $<$ 20m), while barely improving over the source-only baseline under severer shifts. Instead, with explicit context modeling and modulation, our method attains consistent improvement over all ranges, including the part more susceptible to the context gap.

In this paper, we propose Cooperative Context Learning (CCL) to mitigate the context gap through effective context modeling and modulations in a cooperative manner. Specifically, in the local scope, we devise context embeddings to model contextual relationships with close neighbors using an MLP(Multi-Layer Perception) that encodes the relative distance and excludes noise samples. Then at the global level, we introduce a set of learnable prototypes to attend the context embeddings from both domains under the attention paradigm (prototype as the key and context embedding as the query), thus associating the contexts from different regions and domains. Therefore, these prototypes naturally encourage the transfer of context knowledge and hence mitigate the cross-domain variation in the context distribution. More importantly, the global attention in turn attunes the local context modeling, urging them to focus on domain-invariant context patterns. As such, a synergy forms between context embedding and attention where they work cooperatively to promote the adaptation, *i.e*., the former lays the foundation for global modulations while the latter in turn guides and promotes the local context modeling. Extensive experiments and ablations are performed to validate the effectiveness of the proposed method.

The contributions of this work can be summarized as: 1) We observe the discrepancy in context as a critical obstacle for cross-domain point cloud segmentation; 2) We propose Cooperative Context Learning to mitigate the context gap with effective context modeling and modulation cooperatively; 3) Extensive experiments are conducted to verify the effectiveness of the proposed approach, and it attains new state-of-the-art on representative benchmarks.

2. Related Works

Point Cloud Semantic Segmentation aims to classify each point in the LiDAR scan into one of the predefined semantic categories. Due to its unordered property, there are three pathways to preprocess the point clouds for better representation learning. The initial researches propose to process the raw point clouds directly with MLP (Multi-Layer Perception) [10, 32, 33], GCN (Graph Neural Network) [44, 53], or newly designed modules [40, 47, 52, 57]. However, the computational cost for these solutions commonly scales up with the number of points, making it cumbersome for large-scale processing due to its higher latency. Voxel-based methods [12, 19, 20, 39, 54] proposed to divide the 3D spaces into voxels evenly, then apply sparse convolutions to process the voxels. Some researchers [56, 61] also explores different partition strategies to handle the variational density. Also, the heavy computation cost of 3D CNN hampers their applicability to the real-world applications. 3) Projection-based solution is another routine focusing on transforming 3D point clouds into 2D grids so that 2D convolutions can be utilized directly. Various architectures [7, 18, 28, 45, 46, 50] in this direction have been proposed to cope with the projected 2D images, and have been proven the effectiveness and efficiency. Moreover, projection-based methods also receive increasing attention on other tasks [25, 38] for their lower computation cost. In this paper, we choose the projection-based architecture as the backbone to perform our adaptation task as they strike a better balance between performance and efficiency.

Domain Adaptive Semantic Segmentation seeks to derive domain-invariant representations for the dense prediction tasks. Recently, extensive efforts have been paid to the 2D scenario and achieved significant progress. However, the 3D scenario is still less investigated.

In terms of 2D domain adaptive segmentation, there are several ways to mitigate the domain gap. One way handles the gap on the learned features directly [17, 41, 43]. The seminal work [41, 43] leverages the domain adversarial training to facilitate the domain alignment at the output level. Another line of works [62, 63] attempts to realize the effective knowledge transfer with self-training, where confident target samples are assigned with pseudo labels and

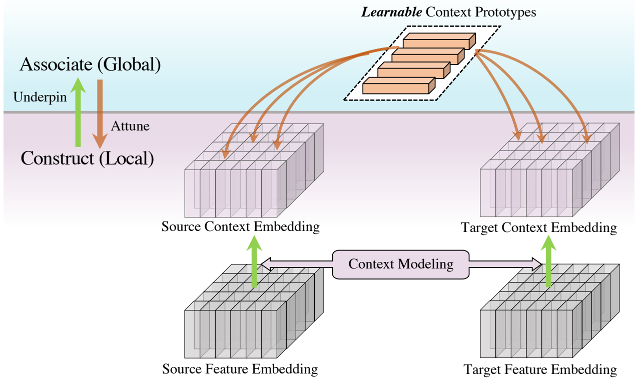


Figure 2. Overview of the "Construct to Associate" scheme. The Construct step builds up context embeddings to model contextual relationships for both domains. Then the Associate step associates and align the context distribution across domains through learnable context prototypes. As such, the "contruct" step models the context distribution, laying the basis of context alignment, while the "associate" step refines the context modeling process via effective cross-domain association and alignment.

optimized with the supervision loss jointly. Li *et al.* [22] proposes to model the contextual distribution explicitly and associate them across domains, thereby minimizing the context gap. However, these methods still focus on context learning on appearance features, thus hampering the scalability and applicability to the 3D scenario.

There are several works dealing with the 3D segmentation adaptation scenario [45, 47, 59]. Wu *et al.* [45] and ePointDA [59] identify the dropout noises as the main obstacle for adaptation and perform noise rendering with the collected noise frequency (the former) or CycleGAN (the latter). SqueezeSegV2 [46] proposes to derive the intensity channels for the synthetic LiDAR scans and perform the domain alignment with geodesic correlation alignment [29]. Saltori *et al.* [35] propose a data augmentation technique to alleviate the distribution shift, *i.e.*, scans from two domains are mixed in a semantic-aware sampling. ASM [23] proposes to adversarially inject noises in the target domain, thereby mitigating the gap induced by noises in the synthetic-to-real adaptation. However, these solutions commonly employ input transformations that treat each scene as a whole but neglect the spatial variation of context inside each scene. Thus, without explicit context modeling, it is still questionable if they can well transform and adapt the regions susceptible to the context shift, especially in the absence of target supervision. Instead, this paper aims to explicitly model and modulate the context distribution across domains, thus better mitigating the gap induced by the shifted geometric context.

3. Methodology

In Fig. 2, we provide an overview of the proposed "Construct to Associate" scheme, where context modeling and modulation are performed at different levels in a cooperative manner, *i.e.*, context embedding for local context modeling and context attention for global modulation. The local context modeling underpins the context modulation in the global scope, while the global attention in turn attunes the modeling of local context relationships, thus cooperating with each other to promote the adaptation. In the following sections, we first provide necessary preliminaries (§ 3.1) for domain adaptive point cloud segmentation. Then, we elaborate on the proposed context learning scheme for local and global scopes in § 3.2 and § 3.3 accordingly. Finally, we detail the training objectives in § 3.4.

3.1. Notations and Preliminaries

In domain adaptive point cloud segmentation, we are given samples from two domains, *i.e.*, annotated source scans $\mathcal{S} = \{(\boldsymbol{P}_i^s, \boldsymbol{Y}_i^s)\}_{i=1}^{N^s}$ and unlabeled target scans $\mathcal{T} = \{(\boldsymbol{P}_i^t)\}_{i=1}^{N^t}$. Here $\boldsymbol{P}_i \in \mathbb{R}^{n_i \times 4}$ denotes the set of points with coordinates (x, y, z) and intensity, and $\boldsymbol{Y}_i \in \mathbb{R}^{n_i}$ denotes the ground-truth annotation for the point cloud, and n_i is the number of points in the i-th scan. As a pre-processing, we project the raw points $\boldsymbol{P}$ into 2D images $\boldsymbol{I} \in \mathbb{R}^{H \times W \times 5}$, which is presented below. The labels are transformed to $\bar{\boldsymbol{Y}} \in \mathbb{R}^{H \times W}$ accordingly.

Spherical Projection. Like previous range-view-based solutions [45, 46], we transform the raw point clouds into 2D rigid images with spherical projection for the sake of efficiency. Concretely, for a point with coordinate (x, y, z), we project it into a 2D image with coordinates (p, q):

$$\left[\begin{array}{c} p \\ q \end{array} \right] = \left[\begin{array}{c} \frac{1}{2}(1 - arctan2^{1}(y, x)/\pi) \cdot W \\ (1 - (arcsin(z \cdot r^{-1}) + f_{up}) \cdot f^{-1}) \cdot H \end{array} \right], \tag{1}$$

where $r = \sqrt{x^2 + y^2 + z^2}$ is the range of this point. $f = f_{up} + f_{down}$ is the vertical field-of-view of the LiDAR sensor. In the projected 2d grids with the resolution of $H \times W$, each point consists of five channels, *i.e.*, range (r), intensity, and the Cartesian coordinates (x, y, z). Notably, even with rigid grids, the property of geometric context is still preserved in terms of density, noise, *etc*.

3.2. Construct: Context Embedding.

We first describe how to discover and model the contextual relationships within the local scope, which act as the prerequisite for the subsequent global context learning. As illustrated in Fig. 3 (left), the context embedding aggregate the neighbor sample (dash borders) with weights that drawn from the contextual relationships.

[1] The arctan2 function in the Numpy library (www.numpy.org).

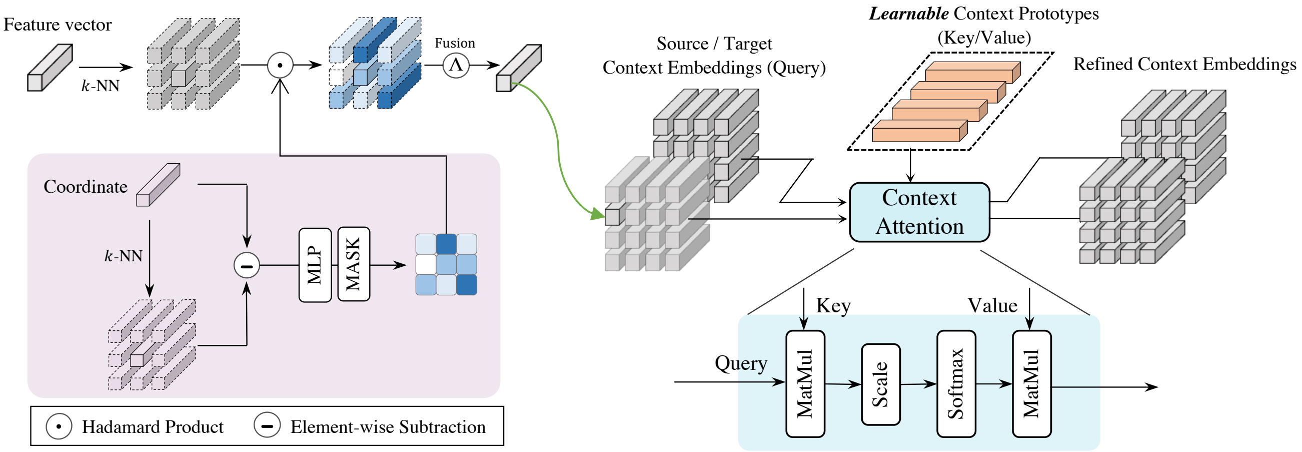


Figure 3. Schematic of the Cooperative Context Learning. **Context Embedding ("Construct")** (left) aggregates the neighbor samples (dashed borders) using weights derived from the geometric context, where an MLP encodes the relative distance and then excludes noise samples. **Context Attention ("Associate")** leverages a set of learnable prototypes to associate the context embeddings, which attend and be optimized with both domains, hence building up the long-range dependency across regions and domains. As such, these prototypes enable the cross-domain transfer of context knowledge, thereby mitigating the cross-domain variation. Moreover, as the learnable property of context embeddings, they can be further modulated with the guidance of attention, thus promoting the adaptation cooperatively.

Concretely, the context embedding considers the contextual relationships from two aspects, *i.e*., the density and noise samples, and integrates them into two consecutive steps. First, like by [40, 58], we treat the relative distances with neighbors as an important clue for geometric context and feed them into an MLP to get the weights. As such, the geometric relationships can be expressed in a learnable manner, offering vital flexibility for the subsequent modulation on them. Second, we leverage a noise mask to zero out the weights of noise samples. The noise mask is a binary mask indicating if a pixel in the range image is filled by samples in point clouds, where the unfilled parts (noises) are missing points during collection due to environmental factors, *e.g*., glasses. The rationale here is that they are observed to distract as outliers in the weight learning.

To be more precise, for the i th sample with coordinate $c_i \in \mathbb{R}^{1\times 3}$ and feature vector $f_i \in \mathbb{R}^{1\times c}$, the context encoding with neighbor samples is formulated as:

$$r_{ij} = \varphi(c_i - c_j)M_j, j \in \mathcal{Q}(i), \tag{2}$$

where φ is a two-layer MLP with non-linearity and $\mathcal{Q}(i)$ is the set of indices for the k-nearest neighbors (k-NN) given index i. M is a binary mask that separates normal samples from noise ones. Benefiting from the spherical projection that transforms point clouds into rigid grids, the neighbor acquisition can be efficiently achieved with the `im2col` operation, rather than the tedious neighbor search.

Then the context embedding can be formulated as:

$$f'_i = \Lambda(\{f_j r_{ij} | j \in \mathcal{Q}(i)\}), \tag{3}$$

where Λ is the fusion function in terms of concatenation, average pooling, or max pooling. And empirical experiments show that concatenation performs best (Sec.4.3).

3.3. Associate: Context Attention

With context embedding, we can model the contextual relationships within local scopes, while leaving cross domain context alignment untouched. To solve this, the self-attention mechanism [42] can be a plausible solution that promote the alignment with cross-domain interactions. However, it is intractable to derive point-to-point affinities with the large amount of points. Hence, we introduce context prototypes as proxies to bridge the context distribution across regions and domains, *i.e*., regarding regions inside and across domains.

As depicted in Fig. 3 (right), we introduce a set of *learnable* prototypes that are shared and optimized with both domains, which attend the context embeddings with the dot-product attention, *i.e*., the prototypes act as the Key and Value and the embeddings are the Query. Formally, given context embedding F' with shape of $h \times w \times d$ and prototypes $E \in \mathbb{R}^{n_{con}\times d}$, the attention further refine the context embedding as:

$$\texttt{Query} = F'W_q, \texttt{Key} = EW_k, \texttt{Value} = EW_v, \tag{4}$$

$$F'' = F' + W_m(softmax(\frac{F'W_q(EW_k)}{\sqrt{d_h}})(EW_v)), \tag{5}$$

where $W_q/W_k/W_v \in \mathbb{R}^{d\times d_l}$ are linear layers that project the inputs into the identical dimension space, and W_m is a linear layer that maps the dimension back to d. A residual connection is added for training stability.

Table 1. Experiments results of SynLiDAR [49] → SemKITTI [1] with SqueezeSegV3-21 [50] as the backbone.

Methods	car	bicycle	mt.cle	truck	oth.-veh.	person	bicyclist	road	parking	sidewalk	building	fence	veget.	trunk	terrain	pole	traff.	**mIoU** (%)
2D methods																		
Source Only	13.7	9.5	2.1	3.0	3.1	9.6	8.1	55.2	6.9	26.8	37.4	3.4	41.3	11.4	30.1	23.1	4.9	17.0
CBST [62]	15.7	6.0	2.5	1.9	3.3	6.9	12.0	56.0	2.2	36.3	36.9	5.7	37.9	19.8	42.1	23.6	3.8	18.4
AdaptSeg [41]	22.1	9.7	3.7	2.3	6.4	9.3	12.9	65.2	5.4	31.6	39.9	6.1	41.7	24.5	42.7	24.0	2.9	20.6
CCM [22]	10.2	8.1	5.1	1.9	3.3	9.7	14.4	52.1	2.7	30.6	39.3	3.9	38.8	18.8	26.9	24.2	4.2	17.3
PLCA [17]	14.6	7.5	3.0	2.6	5.7	10.1	16.3	66.6	1.3	30.9	36.3	3.0	37.2	15.5	37.5	24.7	3.6	18.6
PLCA + CCL	16.7	18.7	7.4	2.9	6.4	13.8	33.4	58.6	4.3	30.6	45.7	5.1	60.1	25.8	26.9	27.0	4.5	22.8 (+4.2)
MMD [36]	23.6	5.4	2.9	2.6	6.4	7.7	10.3	60.0	7.2	28.7	50.9	8.9	51.0	20.4	36.5	24.8	3.4	20.7
MMD + CCL	31.6	13.0	4.4	3.8	5.2	12.1	20.9	61.5	6.7	29.1	55.1	13.3	61.9	27.6	36.7	29.8	6.8	24.6 (+3.9)
3D methods																		
SqzV2 [46]	23.3	5.3	2.4	2.8	6.5	6.6	8.0	63.4	5.6	29.6	38.5	6.3	44.0	24.2	37.5	22.5	3.3	19.4
ASM [23]	19.7	13.8	9.7	2.1	4.1	8.0	8.2	64.5	8.0	36.0	54.6	6.7	58.0	24.7	35.8	29.1	4.2	22.8
LiDARNet [16]	26.3	6.1	3.0	2.1	4.5	7.8	14.9	60.6	8.8	30.9	38.1	5.1	33.2	19.9	35.3	22.9	4.2	19.0
LiDARNet + CCL	30.6	10.3	6.6	3.6	9.7	10.0	22.5	64.3	6.4	33.3	44.3	9.8	43.1	22.5	41.0	25.3	8.1	23.0 (+4.0)
CoSMix [35]	17.3	4.8	3.0	1.6	1.9	5.4	5.5	55.3	1.9	33.7	67.6	7.3	66.1	29.2	43.0	34.4	3.0	22.4
CoxMix + CCL	12.1	5.2	2.1	0.2	7.1	15.3	3.6	75.2	5.2	43.2	66.6	21.0	67.1	26.4	47.3	21.5	3.1	24.9 (+2.5)
SqzV1 [45]	36.7	14.7	8.6	1.9	12.3	12.6	27.0	66.8	9.6	35.1	43.4	8.1	46.0	27.2	42.3	27.1	5.4	25.0
SqzV1 + CCL	52.5	16.3	8.8	3.2	10.6	15.9	28.2	65.4	9.4	33.9	54.4	8.3	63.7	29.4	35.0	31.6	11.7	28.2 (+3.2)

With the long-range dependency across regions and domains, context attention can further promote the context learning regarding both global and local aspects. At the global level, the optimization with both domains urges these prototypes to preserve the context knowledge that are comparably domain invariant while discarding the uncommon ones, hence mitigating the gap via the exchange of domain-invariant context. The latter investigation (Fig. 5) further verifies this in which prototypes show consistent preference towards context patterns across domains. Then for local context learning, context attention offers important guidance for further attuning the context embeddings. As the contextual relationships are expressed in a learnable manner, they can be further modulated and attuned to better extract domain-invariant contexts in terms of both domains, thus realizing a synergy between the local and global context learning and promoting the adaptation cooperatively.

For the sake of efficiency, we replace a standard 3x3 convolution layer with the proposed module (including the part of context embedding). Empirical experiments reveal that performing the replacement at the first encoder block (five in total) is adequate to mitigate the gap with negligible computation overhead (Sec. 4.3 and Table 5 (g)).

3.4. Training Objective

During optimization, we impose three objectives to the model, *i.e.*, cross-entropy loss ($\mathcal{L}_{ce}$), Lovasz-Softmax loss [2] ($\mathcal{L}_{lov}$), and the domain alignment loss ($\mathcal{L}_{da}$):

$$\min_{\theta} \mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{lov} + \mathcal{L}_{da}, \tag{6}$$

Table 2. Overview of used datasets. FOV: vertical field of view.

Dataset	Beams	FOV	# Training	# Validation
SynLiDAR [49]	64	$[-25°, 3°]$	19840	–
SemKITTI [1]	64	$[-25°, 3°]$	19130	4071
nuScenes [3]	32	$[-30°, 10°]$	28130	6019

where θ denotes the parameters of the model and the cross-entropy loss is calculated in a point-wise manner:

$$\mathcal{L}_{ce} = -\sum_{h,w}^{H,W} \log[\theta(I^s)(h, w, \bar{Y}_{h,w})]. \tag{7}$$

Note that $\bar{Y}_{h,w}$ is the label for the point at position (h, w) of a projected LiDAR image.

The training and inference of the model follow the common practice [45,46], which categorizes each point into one of the predefined semantic categories. For a comprehensive evaluation, we incorporate CCL with several representative domain adaptation paradigms and thus do not specify the domain alignment loss here.

4. Experiments

4.1. Setup

Datasets. We validate the proposed CCL on two benchmarks with three datasets, *i.e.*, SynLiDAR→SemKITTI and SynLiDAR→ nuScenes, as presented in Table 2.

SynLiDAR (Syn.) [49] is a synthetic dataset that covers a variety of scenes *i.e.*, urban cities, sub-urban towns, and harbors. We use its official subset for training efficiency.

SemKITTI (Sem.) [1] is a large-scale point cloud dataset collected from the real world. Following common prac-

Table 3. Experiments results of SynLiDAR [49] → nuScenes [3] with SalsaNext [7] as the backbone.

Methods	bicyle	bus	car	oth.-veh.	mt.cle	pedes.	truck	road	oth-grd.	sidew.	terrain	manmd.	veget.	**mIoU** (%)
Source Only	0.3	0.2	8.8	0.3	2.1	5.2	14.2	64.1	1.4	18.0	8.2	27.8	16.5	12.9
+ CCL	0.5	0.1	9.2	0.7	2.3	6.8	10.7	62.2	0.4	20.6	5.4	36.1	19.4	13.4 (+0.5)
CBST [62]	0.2	1.2	13.6	0.6	1.2	3.0	10.8	66.3	0.8	13.6	2.1	21.8	23.4	12.2
+ CCL	0.1	0.8	10.8	1.6	0.9	12.0	16.3	56.9	2.6	20.1	5.6	34.1	35.6	15.1 (+2.9)
AdaptSeg [41]	0.4	1.7	5.9	0.6	2.1	7.1	12.4	58.9	1.6	16.4	10.7	26.8	23.7	13.0
+ CCL	0.5	1.5	19.8	0.3	3.3	6.5	16.7	71.9	2.5	24.3	22.5	37.6	28.1	18.1 (+5.1)
PLCA [17]	0.1	0.7	1.4	0.1	0.5	1.5	8.9	59.7	0.2	14.1	6.9	34.9	17.7	11.3
+ CCL	0.4	1.3	14.6	1.1	2.0	10.3	16.8	63.3	2.4	19.3	12.1	36.1	20.2	15.4 (+4.1)
MMD [36]	0.4	0.2	9.2	0.6	1.8	6.7	16.3	67.9	1.9	21.1	8.4	34.4	24.1	14.8
+ CCL	0.6	2.9	13.6	0.4	2.9	9.4	12.3	70.5	3.3	20.9	8.3	39.5	21.0	15.8 (+1.0)
SqzV1 [45]	0.6	0.8	14.1	0.3	2.6	9.9	13.9	65.8	2.4	17.6	11.5	30.7	21.5	14.7
+ CCL	0.5	2.1	17.0	0.9	1.6	12.0	14.5	69.6	4.7	20.5	10.1	30.3	28.0	16.3 (+1.6)
SqzV2 [46]	0.4	0.3	4.1	0.4	1.7	9.3	10.2	60.5	2.4	18.4	6.8	33.9	23.1	13.2
+ CCL	0.6	0.7	25.5	0.4	2.5	8.0	15.4	71.8	0.7	23.6	23.0	38.8	26.7	18.3 (+5.1)
LiDARNet [16]	0.4	0.5	3.4	0.5	1.5	6.8	10.9	61.5	1.9	18.7	11.6	24.8	23.5	12.8
+ CCL	0.9	1.3	18.3	1.1	2.3	10.3	15.1	70.8	2.0	21.1	15.2	31.3	26.3	17.3 (+4.5)
CoSMix [35]	0.1	2.9	0.4	0.4	0.3	6.2	1.7	67.2	14.5	12.5	10.2	37.8	19.7	13.2
+ CCL	0.3	2.7	1.2	1.1	1.2	8.9	11.2	70.8	8.3	13.3	6.0	40.7	26.6	14.8 (+1.6)

Table 4. Experiments results of SynLiDAR → SemanticKITTI with MinkowskiNet [6] as the backbone.

Method	mIoU
CosMix [35]	32.2
SALUDA [27]	30.2
CosMix + CCL (Ours)	**34.5**

tice [14, 50, 61], we choose sequences 00-10 for training except sequence 08, which is used for validation.

nuScenes-lidarseg (Nus.) [3] is another real-world LiDAR dataset with 40000 frames in total, which mainly collected from two cities, *i.e*., Singapore and Boston.

For both transfers, we merge part of the semantic classes to enable the mapping across datasets, after which 17 and 13 classes for selected for SynLiDAR → SemKITTI and SynLiDAR → nuScenes, respectively.

Evaluation. Following common practice [14, 61], we use mean intersection over union (mIoU) as the evaluation metric. The main results are averaged over 3 random runs.

Implementation. We run the experiments on two benchmarks with two backbones, *i.e*., SqueezeSegV3-21 [50] and SalsaNext [7]. k is set to 8 and n_{con} is 8. The MLP in Eq. 2 is with the format of `fc-bn-relu-fc`, and the hidden dimension is 3. We choose the momentum SGD optimizer to train the model, where the momentum is 0.9 and weight decay is 1×10^{-4}. The optimization schedule follows [50] with a warm-up epoch and decays in an exponential manner. The initial learning rate is set to 4×10^{-3} and 2×10^{-3} for SynLiDAR→SemKITTI and SynLiDAR→nuScenes, respectively. The batch size is set to 24 and the model is optimized for 50 epochs in total.

4.2. Comparisons with Previous Methods

We compare our method with representative domain adaptation segmentation solutions regarding both 2D images and point clouds. For 2D solutions, we choose representation one-stage methods, *i.e*., CBST [62], PLCA [17], AdaptSeg [41], and adapt their code to the 3D scenario. For 3D solutions, we compare our method with SqueezeSegV1 [45], SqueezeSegV2 [46], LiDARNet [21], and CoSMix [35]. For a fair comparison, all the conducted experiments use the identical backbone and supervision loss, *i.e*., $\mathcal{L}_{ce} + \mathcal{L}_{lov}$. In Table 1, Table 3, and Table 4, we integrate the proposed module into these solutions and report the results, where we can make the following observations:

1) Our method can bring consistent improvements over previous solutions, *e.g*., + 4.2 % than PLCA in Table 3, and + 2.3% than CosMix in Table 4, indicating a complementary effect to them. This generally proves the benefit of effective context modeling and alignment in domain adaptation.

2) Compared with source-only, 2D solutions attain negligible improvement. This implies that methods focus on appearance feature cannot scale to point clouds with rich geometries. Besides, CCL leads to limited gains than source-only. This is because the CCL cannot perform cross-domain association and alignment in the absence of target data.

3) Integrating with previous 3D solutions leads to better adaptation results consistently, including methods already considering the context gap, *e.g*., +3.2% than SqzV1 in Table 1. This validates that CCL can better help the model scale to variational geometric contexts across domains.

4.3. Ablation Studies and Analysis

In this section, we perform comprehensive ablations to validate the necessity of each proposed design. Here we choose the variant using AdaptSeg + CCL in the ablations for its competitive performances on both benchmarks.

Contribution of each component. First, in Table 5 (a), we evaluate the contribution of the proposed components, *i.e*., MLP-encoding and masking in context embed-

Table 5. Ablation studies on SynLiDAR (Syn.) → SemKITTI (Sem.) and SynLiDAR (Syn.) → nuScenes (Nus.)

ConEmb		ConAtt	Transfer	
MLP-enc.	Masking		Syn.→Sem.	Syn.→Nus.
			20.6	13.0
		✓	21.4	14.5
✓	✓		21.8	16.5
✓		✓	22.0	16.1
✓	✓	✓	23.3	18.1

(a) Contribution of the proposed modules: MLP-encoding and Masking in Context Embedding (ConEmb), and Context Attention (ConAtt).

Module	Transfer	
	Syn.→Sem.	Syn.→Nus.
KPConv [40]	20.3	14.1
Point Transformer [58]	21.2	14.4
Meta-Kernel [9]	21.8	14.7
Domain-specific proto.	22.3	16.0
Domain-shared proto.	**23.3**	**18.1**

(b) Comparison with other modules for context modeling. Proto.= Prototypes.

Position	Syn.→Sem.	Syn.→Nus.
1st Enc.	**23.3**	**18.1**
3rd Enc.	22.2	16.9
5th Enc.	21.5	14.4
End of Dec.	21.0	14.3

(c) Comparison on different insert positions, where Enc. = encoder and Dec. = decoder.

n_{con}	Syn.→Sem.	Syn.→Nus.
4	23.1	17.8
8	**23.3**	**18.1**
12	22.8	18.0
16	23.1	16.9

(d) Sensitivity to n_{con}, the number of context prototypes.

k (grid)	Syn.→Sem.	Syn.→Nus.
8 (3x3)	23.3	18.1
24 (5x5)	27.2	18.2
48 (7x7)	**29.5**	**19.9**
80 (9x9)	27.8	17.4

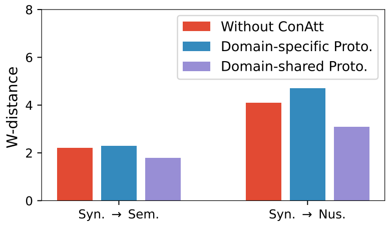
(e) Comparison on different numbers of neighbors in context embeddings.

Fusion	Transfer	
	Syn.→Sem.	Syn.→Nus.
MaxPool	23.4	18.0
AvgPool	22.6	16.8
Concat.	23.3	18.1

(f) Ablation on the fusion process in context embedding, *i.e.*, max pooling, average pooling, and concatenation (concat.).

Backbone	FLOPS (G)	Param. (M)
SqzV3	25.2	9.3
+ CCL	24.7 (-0.5)	9.2 (-0.1)
SalsaNext	3.9	6.8
+ CCL	3.8 (-0.1)	6.7 (-0.1)

(g) Analysis of the model capacity with or without the proposed module, in terms of FLOPS and parameters (Param.).



(h) The cross-domain distance on relative encoding with different attention strategies.

ding (ConEmb), and context attention (ConAtt). First, we can observe that all components contribute to the domain adaptation, and removing neither of them leads to an apparent performance drop. Second, combing context embedding and context attention can lead to better results than using one of them. This is because they two focus on local and global contexts accordingly and are complementary to each other, thus integrating them can better promote the adaptation. Especially, under the scenario with a larger context gap (SynLiDAR→nuScenes), we can notice that the masking process brings an obvious improvement, *i.e.*, + 2.0 % on nuScenes. The reason here is that the sparser LiDAR scans contain more dropout noises and the domain alignment is more vulnerable to them, while our masking strategy can effectively mitigate such distractions.

Comparison with other context learning modules. In Table 5 (b), we compare our method with other modules for the context learning of point clouds. The results show that our method outperforms other solutions by a noticeable margin. We conjecture the reason is that they do not consider the cross-domain variation, thus cannot brings obvious benefits for the domain alignment. To further investigate this, we change the domain-sharing prototypes to domain-specific prototypes that are optimized separately and report the result. The decreased performance further justifies the benefit of the domain-sharing mechanism on prototypes.

Locations for inserting the proposed module. In Table 5 (c), we investigate the optimal position for placing the proposed module, *i.e.*, 1st, 3rd, 5th encoder block (five in total), and end of the decoder. As we can observe, placing at the first encoder block achieves the best result. This reason is that the fined-grained contextual relationships are better preserved at the shallower layers and diminish at deeper.

Fusion strategy for context embedding. In Table. 5 (f), we exploit different fusion strategies for aggregation with neighbors, and the concatenation achieves the best. However, using other variants also maintains our superiority against other solutions, indicating our method is not sensitive to the fusion process.

Analysis on model capacity. In Table. 5 (g), we present the analysis of model capacity in terms of FLOPS and parameters. Apparently, our method does not induce extra computation costs, even less. This indicates that the gain in performance should be ascribed to effective context learning, rather than increasing model capacity.

Sensitivity to hyper-parameters. Here we examine the sensitivity analysis to the hyper-parameters. First, in Table 5 (d), we evaluate the sensitivity to the number of pro-

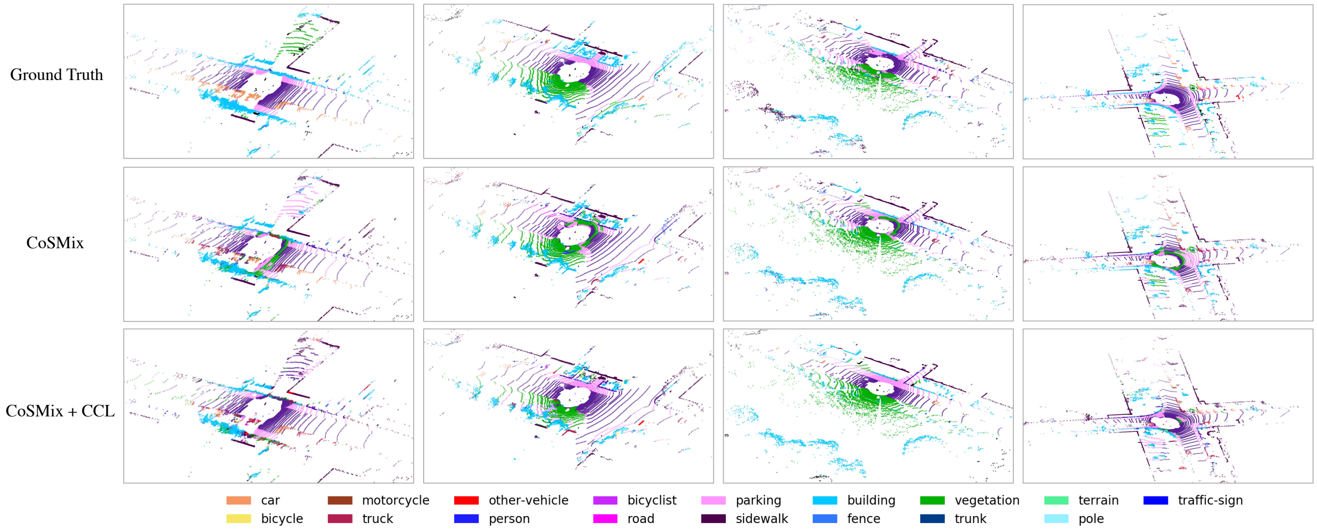


Figure 4. Visualization of the segmentation results on the validation set of SemKITTI [1]. Best viewed in color.

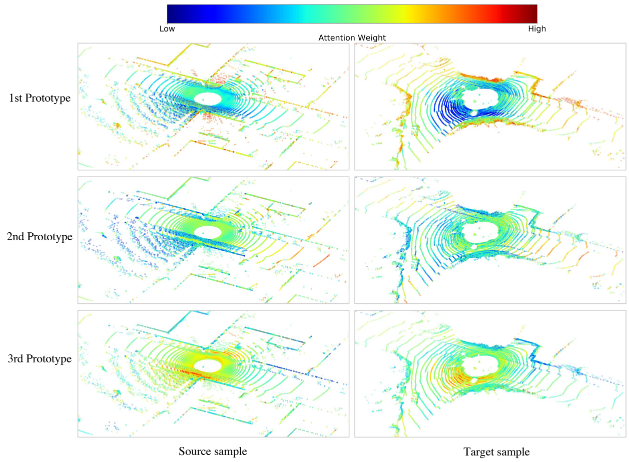


Figure 5. Heatmap visualization for the source and target samples with the first three prototypes. The prototypes implicitly differentiate into different context patterns that are consistently held across domains. Best viewed in color.

totypes (n_{con}). With a large range (from 4 to 16), the performance suffers very limited fluctuations, verifying the robustness to it. Second, we examine the sensitivity to the number of neighbors (k) in Table 5 (e). As we could observe, increasing k first leads to apparent improvement and then decreases slightly, showing that enlarging the receptive field can boost the performance but too large may harm. The reason we set k to 8 is to be in line with the receptive field of the replaced part, *i.e.*, the 3x3 convolution layer.

How does CCL promote the adaptation. Here we investigate the effect of CCL in the adaptation process both *quantitatively* and *qualitatively*:

1) In Table 5 (h), we compare the cross-domain Wasserstein distances on the weights (r in Eq. 2) in context embeddings. Notably, using domain-specific prototypes that breaks the cross-domain interaction, cannot mitigate the domain gap. This reveals that the CCL can guide the context embedding to overcome the gap through cross-domain interactions.
2) In Fig. 5, we visualize the heatmap of the context prototypes, where we can observe different prototypes focus on different context patterns but in a domain-consistent manner. This proves that these prototypes indeed preserve domain-invariant context knowledge to some extent.

Visualizations. In Fig. 4, we visualize and compare our method with CoSMix and the ground truth. Apparently, CCL derives better segmentation results with effective context learning, *e.g.*, CCL can better identify class “road” and “vegetation” which CoSMix is easily confused with.

5. Conclusion

In this paper, we aim to overcome the discrepancy induced by the variational geometric context for domain adaptive point cloud segmentation. To arrive at it, we propose a new context learning paradigm, Cooperative Context Learning, which models and modulates the contextual representation from different aspects but in a cooperative manner. Specifically, CCL first constructs the context embeddings within the local scope, then associates them with context attention globally, where the attention in turn attunes the context modeling and therefore narrows the gap cooperatively. Consequently, the cross-domain variation in context distribution can be effectively mitigated through the global exchange of context knowledge and modulation on the local context modeling. Extensive experiments are conducted to verify the effectiveness of the proposed method.

References

[1] 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 *ICCV*, 2019. 5, 8

[2] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *CVPR*, 2018. 5

[3] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *CVPR*, 2020. 5, 6

[4] Wei-Lun Chang, Hui-Po Wang, Wen-Hsiao Peng, and Wei-Chen Chiu. All about structure: Adapting structural information across domains for boosting semantic segmentation. In *CVPR*, 2019. 1

[5] Jaesung Choe, Chunghyun Park, Francois Rameau, Jaesik Park, and In So Kweon. Pointmixer: Mlp-mixer for point cloud understanding. *arXiv preprint arXiv:2111.11187*, 2021. 1

[6] 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*, pages 3075–3084, 2019. 6

[7] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds for autonomous driving, 2020. 2, 6

[8] Runyu Ding, Jihan Yang, Li Jiang, and Xiaojuan Qi. Doda: Data-oriented sim-to-real domain adaptation for 3d semantic segmentation. In *ECCV*, 2022. 1

[9] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and ZhaoXiang Zhang. Rangedet: In defense of range view for lidar-based 3d object detection. In *ICCV*, October 2021. 7

[10] 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 *CVPR*, 2021. 2

[11] Rui Gong, Yuhua Chen, Danda Pani Paudel, Yawei Li, Ajad Chhatkuli, Wen Li, Dengxin Dai, and Luc Van Gool. Cluster, split, fuse, and update: Meta-learning for open compound domain adaptive semantic segmentation. In *CVPR*, 2021. 1

[12] Tong He, Chunhua Shen, and Anton van den Hengel. DyCo3d: Robust instance segmentation of 3d point clouds through dynamic convolution. In *CVPR*, 2021. 2

[13] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022. 1

[14] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Learning semantic segmentation of large-scale point clouds with random sampling. *IEEE TPAMI*, 2021. 1, 6

[15] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Cross-view regularization for domain adaptive panoptic segmentation. In *CVPR*, 2021. 1

[16] Peng Jiang and Srikanth Saripalli. Lidarnet: A boundary-aware domain adaptation model for point cloud semantic segmentation. *ICRA*, 2021. 5, 6

[17] Guoliang Kang, Yunchao Wei, Yi Yang, Yueting Zhuang, and Alexander G Hauptmann. Pixel-level cycle association: A new perspective for domain adaptive semantic segmentation. In *NeurIPS*, 2020. 2, 5, 6

[18] Deyvid Kochanov, Fatemeh Karimi Nejadasl, and Olaf Booij. Kprnet: Improving projection-based lidar semantic segmentation. *ECCV workshop*, 2020. 2

[19] 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*, pages 17545–17555, 2023. 2

[20] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, 2019. 2

[21] F. Langer, A. Milioto, A. Haag, J. Behley, and C. Stachniss. Domain Transfer for Semantic Segmentation of LiDAR Data using Deep Neural Networks. In *IROS*, 2020. 6

[22] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. Content-consistent matching for domain adaptive semantic segmentation. In *European Conference on Computer Vision*, pages 440–456. Springer, 2020. 1, 3, 5

[23] Guangrui Li, Guoliang Kang, Xiaohan Wang, Yunchao Wei, and Yi Yang. Adversarially masking synthetic to mimic real: Adaptive noise injection for point cloud segmentation adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20464–20474, June 2023. 1, 3, 5

[24] Guangrui Li, Yifan Sun, Zongxin Yang, and Yi Yang. Decompose to generalize: Species-generalized animal pose estimation. In *The Eleventh International Conference on Learning Representations*, 2023. 1

[25] Zhidong Liang, Zehan Zhang, Ming Zhang, Xian Zhao, and Shiliang Pu. Rangeioudet: Range image based real-time 3d object detector optimized by intersection over union. In *CVPR*, 2021. 2

[26] Mingsheng Long, ZHANGJIE CAO, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018. 1

[27] Bjoern Michele, Alexandre Boulch, Gilles Puy, Tuan-Hung Vu, Renaud Marlet, and Nicolas Courty. Saluda: Surface-based automotive lidar unsupervised domain adaptation. In *3DV*, 2024. 6

[28] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. RangeNet++: Fast and Accurate LiDAR Semantic Segmentation. In *IROS*, 2019. 2

[29] Pietro Morerio, Jacopo Cavazza, and Vittorio Murino. Minimal-entropy correlation alignment for unsupervised deep domain adaptation. *ICLR*, 2018. 3

[30] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 2010. 1

[31] Chunghyun Park, Yoonwoo Jeong, Minsu Cho, and Jaesik Park. Fast point transformer. In *CVPR*, 2022. 1

[32] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CVPR*, 2017. 1, 2

[33] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 2017. 1, 2

[34] Can Qin, Haoxuan You, Lichen Wang, C-C Jay Kuo, and Yun Fu. Pointdan: A multi-scale 3d domain adaption network for point cloud representation. *NeurIPS*, 2019. 1

[35] Cristiano Saltori, Fabio Galasso, Giuseppe Fiameni, Nicu Sebe, Elisa Ricci, and Fabio Poiesi. Cosmix: Compositional semantic mix for domain adaptation in 3d lidar segmentation. *ECCV*, 2022. 1, 2, 3, 5, 6

[36] Dino Sejdinovic, Bharath Sriperumbudur, Arthur Gretton, and Kenji Fukumizu. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. *The annals of statistics*, pages 2263–2291, 2013. 5, 6

[37] Yuefan Shen, Yanchao Yang, Mi Yan, He Wang, Youyi Zheng, and Leonidas J. Guibas. Domain adaptation on point clouds via geometry-aware implicits. In *CVPR*, 2022. 1

[38] Pei Sun, Weiyue Wang, Yuning Chai, Gamaleldin Elsayed, Alex Bewley, Xiao Zhang, Cristian Sminchisescu, and Dragomir Anguelov. Rsn: Range sparse net for efficient, accurate lidar 3d object detection. In *CVPR*, 2021. 2

[39] 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 *ECCV*, 2020. 2

[40] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. *ICCV*, 2019. 2, 4, 7

[41] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker. Learning to adapt structured output space for semantic segmentation. In *CVPR*, 2018. 2, 5, 6

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017. 4

[43] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Mathieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019. 2

[44] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM TOG*, 2019. 2

[45] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In *ICRA*, 2018. 2, 3, 5, 6

[46] Bichen Wu, Xuanyu Zhou, Sicheng Zhao, Xiangyu Yue, and Kurt Keutzer. Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In *ICRA*, 2019. 2, 3, 5, 6

[47] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. *CVPR*, 2019. 2, 3

[48] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. In *NeurIPS*, 2022. 1

[49] Aoran Xiao, Jiaxing Huang, Dayan Guan, Fangneng Zhan, and Shijian Lu. Synlidar: Learning from synthetic lidar sequential point cloud for semantic segmentation. *AAAI*, 2022. 5, 6

[50] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. In *ECCV*, 2020. 2, 5, 6

[51] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16024–16033, 2021. 1

[52] Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. In *CVPR*, 2021. 2

[53] Qiangeng Xu, Xudong Sun, Cho-Ying Wu, Panqu Wang, and Ulrich Neumann. Grid-gcn for fast and scalable point cloud learning. *CVPR*, 2020. 2

[54] Xu Yan, Jiantao Gao, Chaoda Zheng, Chao Zheng, Ruimao Zhang, Shuguang Cui, and Zhen Li. 2dpass: 2d priors assisted semantic segmentation on lidar point clouds. In *European Conference on Computer Vision*, pages 677–695. Springer, 2022. 2

[55] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *CVPR*, 2020. 1

[56] 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 *CVPR*, 2020. 2

[57] Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In *ICCV*, 2019. 2

[58] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *ICCV*, 2021. 1, 4, 7

[59] Sicheng Zhao, Yezhen Wang, Bo Li, Bichen Wu, Yang Gao, Pengfei Xu, Trevor Darrell, and Kurt Keutzer. epointda: An end-to-end simulation-to-real domain adaptation framework for lidar point cloud segmentation. In *AAAI*, 2021. 1, 2, 3

[60] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 2

[61] 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 *CVPR*, 2021. 2, 6

[62] Yang Zou, Zhiding Yu, B.V.K. Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 2, 5, 6

[63] Yang Zou, Zhiding Yu, Xiaofeng Liu, B.V.K. Vijaya Kumar, and Jinsong Wang. Confidence regularized self-training. In *ICCV*, 2019. 2