Domain Adaptive LiDAR Point Cloud Segmentation With 3D Spatial Consistency

Aoran Xiao, Dayan Guan, Xiaoqin Zhang, *Senior Member, IEEE*, and Shijian Lu

***Abstract*—Domain adaptive LiDAR point cloud segmentation aims to learn an effective target segmentation model from labelled source data and unlabelled target data, which has attracted increasing attention in recent years due to the difficulty in point-cloud annotation. It remains a very open research challenge as point clouds of different domains often have clear distribution discrepancies with variations in LiDAR sensor configurations, environmental conditions, occlusions, etc. We design a simple yet effective spatial consistency training framework that can learn superior domain-invariant feature representations from unlabelled target point clouds. The framework exploits three types of spatial consistency, namely, geometric-transform consistency, sparsity consistency, and mixing consistency which capture the semantic invariance of point clouds with respect to viewpoint changes, sparsity changes, and local context changes, respectively. With a concise mean teacher learning strategy, our experiments show that the proposed spatial consistency training outperforms the state-of-the-art significantly and consistently across multiple public benchmarks.**

***Index Terms*—LiDAR point clouds, semantic segmentation, domain adaptation, 3D vision, transfer learning, deep learning.**

I. Introduction

Semantic segmentation of 3D LiDAR point clouds is critical in different computer vision tasks such as autonomous driving, remote sensing, robotics, etc. It has achieved great progress thanks to the recent advances in deep neural networks (DNNs). Nevertheless, effective DNN training usually requires large-scale densely annotated point clouds which are extremely laborious to collect. One approach that could alleviate the annotation constraint is to leverage synthetic point clouds that often come with automatically generated labels [1]. However, synthetic point clouds exhibit clear distribution discrepancies as compared with real point clouds [1], [2], and DNN models trained by using synthetic point clouds often experience clear performance drops while applied to real point clouds.

Unsupervised domain adaptation (UDA) can mitigate the distribution discrepancy between a labelled source domain and an unlabelled target domain, which has recently attracted increasing attention for the task of 3D LiDAR point cloud segmentation. Domain adaptive point cloud segmentation has been investigated in three major approaches: 1) self-training that selects confident target predictions as pseudo-labels for network training [2], [4]; 2) cross-domain point cloud translation [1], [5]; and 3) projecting point clouds into 2D space for adaptation [6], [7]. Meanwhile, consistency training [8], [9] has recently emerged as an effective UDA technique in various 2D image recognition tasks. It learns robust and generalizable target representations effectively by enforcing a model's output to remain consistent under the presence of input perturbations.

We investigate the efficacy of consistency training for domain adaptive semantic segmentation of 3D LiDAR point clouds. The primary challenge lies in designing effective consistency strategies that can facilitate the learning of domain adaptive representations for segmenting target point clouds in an unsupervised manner. Intuitively, the semantics of point clouds should remain invariant under the presence of variations in sensor viewpoints, point sampling density, and local context. However, we observe that the predictions of deep models are highly susceptible to the above variations. This can be observed in Fig. 1(c), (d), and (e), where the inter-domain prediction errors (while applying a source-trained model to target point clouds as illustrated in Fig. 1(b)) are exacerbated in different manners when the input point clouds undergo changes in sensor viewpoints, sampling sparsity, and local context, respectively.

Inspired by the above observations, we design SCT, a simple yet effective spatial consistency training framework that can learn effective domain adaptive point cloud representations. SCT introduces spatial perturbations to mimic the aforementioned variation factors and learns by enforcing the prediction of spatially perturbed point clouds to be consistent with that of the original point clouds. We design three types of spatial perturbations: 1) *Geometric transform* that simulates the viewpoint change of LiDAR sensors; 2) *Sparsity variation* that down-samples input point clouds; and 3) *Mixing* that modifies the local context of input point clouds. With the three types of spatial perturbations, we formulate three types of spatial consistency including *geometric-transform consistency*, *sparsity consistency*, and *mixing consistency* which are well tailored to the spatial characteristics of point clouds. With a mean teacher learning strategy [10],

Manuscript received 18 July 2023; revised 25 October 2023; accepted 6 November 2023. Date of publication 28 November 2023; date of current version 21 March 2024. This work was supported by the Ministry of Education Singapore, through the Tier-1 Scheme under Project Number RG18/22. The Associate Editor coordinating the review of this manuscript and approving it for publication was Mr. Jingkuan Song. *(Corresponding authors: Xiaoqin Zhang; Shijian Lu.)*

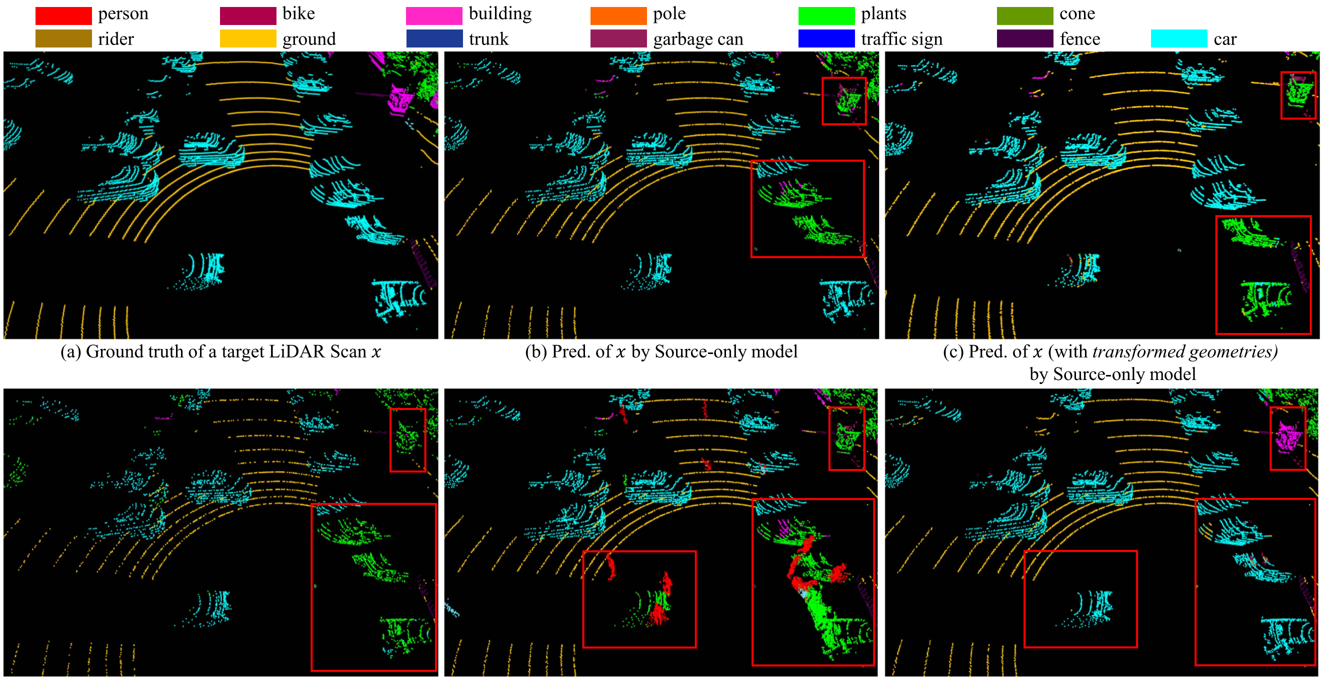
Aoran Xiao and Shijian Lu are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: aoran.xiao@ntu.edu.sg; shijian.lu@ntu.edu.sg).

Dayan Guan is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: dayan.guan@ntu.edu.sg).

Xiaoqin Zhang is with the Key Laboratory of Intelligent Informatics for Safety & Emergency of Zhejiang Province, Wenzhou University, Wenzhou 325035, China (e-mail: zhangxiaoqinnan@gmail.com).

Code is available at https://github.com/xiaoaoran/sct-uda.

Digital Object Identifier 10.1109/TMM.2023.3335879



(a) Ground truth of a target LiDAR Scan x

(b) Pred. of x by Source-only model

(c) Pred. of x (with *transformed geometries)* by Source-only model

(d) Pred. of x (with *changed sparsity)* by Source-only model

(e) Pred. of x (with *changed local contexts)* by Source-only model

(f) Pred. of x by *spatial consistency training* model

Fig. 1. Spatial consistency training helps in domain adaptive LiDAR segmentation: (a) shows the ground-truth segmentation of one target LiDAR scan from SemanticPOSS [3] and the rest shows its segmentation by different models. Specifically, (b) shows the segmentation by a "Source-only" model trained with the source data (i.e., synthetic point clouds in SynLiDAR [1]) whose performance degrades clearly while applied to the target scan from a different domain. The performance degradation exacerbates when the target scan suffers from spatial perturbations such as geometric transformation (rotation, scaling, and flipping) in (c), sparsity changes (point down-sampling) in (d), and local context changes (i.e., the presence of *persons* (in red) around *cars*) in (e). Our *Spatial Consistency Training* exploits the inherent nature of LiDAR point clouds and semantic invariance with respect to spatial perturbations to regularize the domain adaptation process, leading to clearly improved segmentation of the target scan as in (f). The red boxes highlight areas with substantial performance disparities, and LiDAR views in all subfigures are aligned for easy comparison. Best viewed in color.

a simple implementation of the three types of spatial consistency outperforms the state-of-the-art with significant margins as illustrated in Fig. 1(f).

In summary, the contributions of this work are threefold. *First*, we identify that spatial perturbations including geometric transformation, sparsity changes, and local context changes can clearly degrade the cross-domain LiDAR point cloud segmentation. To this end, we design three types of spatial consistency learning strategies tailored for LiDAR point clouds, which help learn domain adaptive representations and enhance unsupervised cross-domain transfer of LiDAR point clouds greatly. *Second*, our proposed spatial consistency training framework, characterized by its elegant simplicity, exceptional effectiveness, and computational efficiency (can train with a single NVIDIA 2080Ti of 11 GB), can serve as a strong baseline and foundation for future studies. We will release code to facilitate this process. *Third*, extensive experiments over two challenging synthetic-to-real benchmarks (i.e., SynLiDAR [1] → SemanticKITTI [10] and SynLiDAR → SemanticPOSS [11]) show that the proposed framework outperforms the state-of-the-art consistently by large margins.

The remainder of this paper is organized as follows. Section II provides a comprehensive review of related studies, including research on point cloud semantic segmentation, domain adaptive LiDAR segmentation, and consistency learning. Section III presents the proposed method in detail including problem definition, spatial consistency training, and fast point-wise matching. Section IV presents experimental results as well as related analysis. Finally, several concluding remarks are drawn in Section V.

II. Related Work

A. Point Cloud Semantic Segmentation

LiDAR point clouds have been widely exploited in various autonomous navigation tasks for 3D scene understanding. This triggers several large-scale LiDAR point-cloud datasets [1], [3], [11], [12], [13] which greatly promote the research in 3D point cloud segmentation [14], [15], [16], [17]. Meanwhile, different deep architectures and learning algorithms have been proposed. One typical approach is to project 3D point clouds into 2D depth images and then adopt standard 2D convolution neural networks for segmentation [18], [19], [20], [21], [22], [23]. This approach is efficient for processing large-scale point clouds but tends to lose geometric information in its 3D-to-2D mapping process. Another approach employs multilayer perceptron for point cloud representation learning [24], [25], [26] but is computationally intensive for large-scale point clouds. Beyond that, several studies [27], [28], [29], [30] quantize point clouds into discrete 3D grids and leverage 3D convolutions [31] or sparse convolutions [27], [32], [33] for learning and segmenting voxelized points. We follow [1], [2] and adopt the state-of-the-art MinkowskiNet [27] which is a sparse convolutional 3D point cloud segmentation network with a fine balance between accuracy and efficiency.

B. Domain Adaptive LiDAR Segmentation

Domain adaptive point cloud segmentation [1], [2], [4], [6], [7], [34], [35] aims for optimal exploitation of previously annotated 'source' point clouds while handling unannotated 'target' point clouds collected in various new domains. It has attracted increasing attention recently due to the challenge in point cloud annotation [36]. Earlier studies [6], [7], [37], [38] project point clouds into depth images and then adopt 2D UDA methods for point cloud segmentation. However, these studies are model-specific and not applicable across deep architectures [39]. Recently, several model-agnostic studies [1], [2], [4] handle the domain discrepancy in the input space directly. For example, [1] translates synthetic point clouds to have similar appearances and sparsity as real point clouds. [5] formulates domain adaptation as a point cloud completion task to minimize density variation across domains. [2], [4] mitigate domain discrepancy by mixing source and target data and creating an intermediate domain with a smaller domain gap. Differently, we design a novel spatial consistency training framework that explores consistency training for domain adaptive 3D LiDAR segmentation.

C. Consistency Training

Consistency training has been widely explored for semi-supervised learning of 2D images, aiming to enhance the robustness of the learnt models while facing various input perturbations. Under this context, different ways of perturbations have been investigated, e.g., by including perturbation noises [40], [41], [42], [43], image augmentation [44], [45], [46], etc. Recently, one line of research [8], [9], [47], [48] extends the concept of consistency training to the unsupervised domain adaptation of 2D images, largely by designing effective image augmentations for reducing domain gaps across datasets. In addition, another line of research extends consistency training to point cloud tasks. For instance, [49] presents a point-level consistency loss for 3D semi-supervised semantic segmentation, while [50] introduced a multi-level consistency framework for domain adaptive 3D object detection. Beyond that, several self-supervised networks [51], [52], [53] adopt contrastive loss for learning consistent predictions over augmented point cloud views. As a comparison, we design three types of spatial consistency for learning domain adaptive point cloud representations for the task of LiDAR point cloud semantic segmentation.

D. Mean-Teacher Structure

The "mean-teacher" architecture is a classical architecture that has been widely adopted in various 2D computer vision tasks such as semi-supervised learning [10], [54] and unsupervised domain adaptation [9]. It involves two networks: a student and a teacher. The student is the main network being trained, while the teacher is a copy of the student with a slower update. During the training, the teacher regularizes the student by ensuring that their predictions on unlabelled data are consistent. Recently, it has been extended into 3D point cloud recognition, including semi-supervised 3D segmentation [55], domain adaptive 3D detection [56] and segmentation [2], etc.

III. Methods

This section presents the proposed method, which consists of four subsections that describe the problem definition for UDA in LiDAR point cloud segmentation, the proposed SCT framework, a fast point-wise matching strategy, and the spatial consistency strategy, respectively.

A. Problem Definition

Under the setting of UDA, we have access to LiDAR point cloud data from a labeled source domain $\mathcal{D}_s = \{x_s^i, y_s^i\}_{i=1}^{N_s}$ and an unlabeled target domain $\mathcal{D}_t = \{x_t^i\}_{i=1}^{N_t}$, where N_s and N_t represent scan numbers of LiDAR point clouds from the source and target domains, respectively. Each LiDAR point cloud $x^i \in \mathbb{R}^{n^i \times 3}$ consists of n^i points with their 3D coordinates while $y_s^i \in \mathbb{R}^{n^i}$ denotes the point-wise labels of the corresponding training sample from the source domain. The goal of domain adaptive point cloud segmentation is to learn a model F based on $\mathcal{D}_s$ and $\mathcal{D}_t$ that can produce accurate predictions $\hat{y}_t$ for new target data from $\mathcal{D}_t$.

B. Overall Framework

The proposed SCT integrates supervised knowledge from the source domain and self-supervised knowledge from the target domain for learning domain-adaptive representations for segmenting target LiDAR point clouds. Fig. 2 shows the overall network framework and Algorithm 1 provides the pseudo-code of the proposed SCT. In the following subsection, we present the *Network Architecture* of SCT, as well as the *Training* and *Inference* of SCT on the task of 3D point cloud semantic segmentation.

Network Architecture We adopted the mean-teacher architecture [10] in the implementation of the proposed SCT. Specifically, the network F consists of a *teacher* network F_T with parameters θ_T, and a *student* network F_S with parameters θ_S. Both are 3D segmentation networks and they share the same network structures.

Training For the labelled source domain, we adopt standard supervised learning to learn semantic structures. Specifically, for a source point cloud scan with corresponding labels $\{x_s^i, y_s^i\}$, we adopt standard cross entropy loss as supervised loss $\mathcal{L}_s$ to optimize the *student* network F_S. The loss is defined as:

$$\mathcal{L}_s = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{1}{n_s^i}\sum_{j=1}^{n_s^i}\mathcal{H}(y_s^{i,j}, p_s^{i,j}(y|x_s^{i,j})) \tag{1}$$

where $p_s^{i,j} \in \mathbb{R}^{1\times C}$ is the output probability distribution of source point j of x_s^i over C classes, i.e., $p_s^{i,j} = softmax(F_S(x_s^{i,j}))$, and $\mathcal{H}$ denotes the entropy.

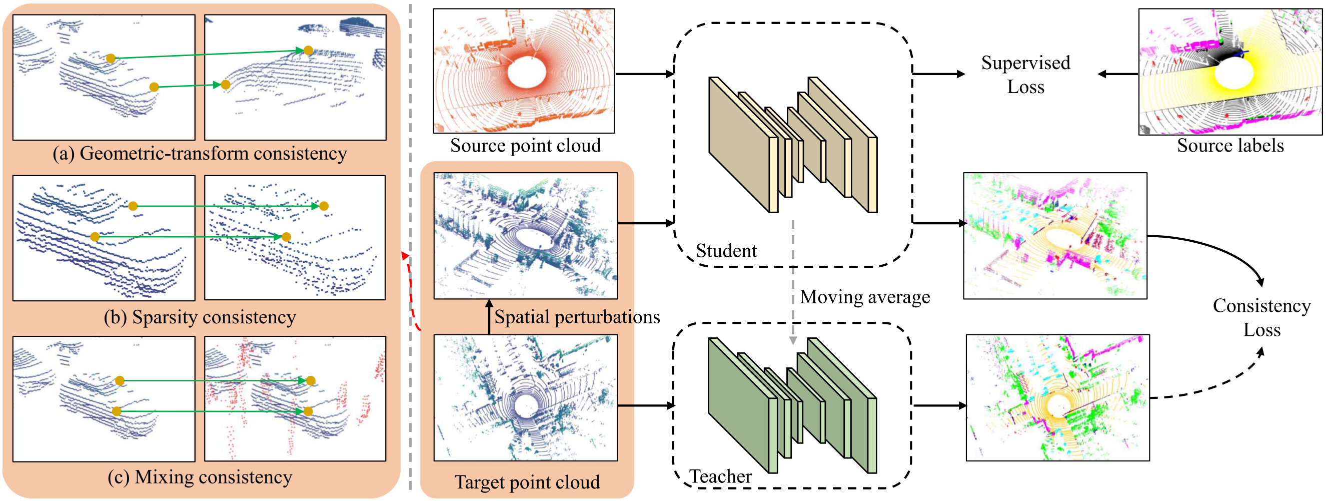


Fig. 2. Pipeline of spatial consistency training (SCT) framework. Leveraging the mean-teacher scheme [10], the student network updates at each iteration by the exponential moving average of itself as the teacher network. The learning enforces the student predictions on spatially perturbed point clouds to be consistent with the teacher predictions on the corresponding raw point clouds under a *Consistency Loss*. We design three types of spatial consistency, namely, *geometric-transform consistency*, *sparsity consistency*, and *mixing consistency*, which are tailored to the spatial characteristics of LiDAR point clouds for enhancing the cross-domain segmentation performance.

For an unlabelled target scan x_t, we first generate a spatially perturbed view $\Omega(x_t)$ by randomly applying one of three types of spatial perturbations as to be described in Section III-D. We then feed $\Omega(x_t)$ to the *student* network F_S to obtain prediction logits $F_S(\Omega(x_t))$. Similarly, we feed x_t to the *teacher* network F_T to obtain prediction logits $F_T(x_t)$. The learning from unlabelled target point clouds can thus be achieved by a cross-entropy loss that enforces the *student*'s predictions to be consistent with the *teacher*'s predictions as follows:

$$\mathcal{L}_t = \frac{1}{N_t}\sum_{i=1}^{N_t}\frac{1}{n_t^i}\sum_{j=1}^{n_t^j}\mathcal{H}(\hat{y}_t^{i,j}, p_t^{i,j}(y|\Omega(x_t^{i,j}))), \tag{2}$$

where $\hat{y}_t^{i,j}$ is the pseudo label generated by the *teacher* model, which is defined as the class with the maximum prediction probability, i.e., $\hat{y}_t^{i,j} = \arg\max(F_T(x_t^{i,j}))$.

Note the *teacher* network does not back-propagate gradients in training. Instead, it is updated iteratively through exponential moving average of the momentum of the student network as follows:

$$\theta_T = \beta\theta_T + (1-\beta)\theta_S \tag{3}$$

where β is the momentum update rate.

The overall objective is a weighted combination of the supervised and unsupervised losses as follows:

$$\mathcal{L} = \mathcal{L}_s + \lambda_t\mathcal{L}_t \tag{4}$$

where λ_t is a balancing weight.

Successful training with the spatial consistency pipeline in Fig. 2 has two prerequisites. *First*, it requires an efficient matching algorithm to match unlabelled target points of two different views, which is a nontrivial task as point clouds are disordered and unstructured data. *Second*, it requires effective spatial consistency strategies, i.e., the design of Ω for learning from unlabelled target data. For the first prerequisite, we design a fast online matching strategy as to be described in Section III-C. For the second prerequisite, we design three types of spatial consistency as to be described in Section III-D.

Inference After training, we employ the *student* network directly for inference. Hence, SCT introduces no additional computational overhead during the inference stage.

C. Fast Point-Wise Matching

3D semantic segmentation of LiDAR point clouds is computationally intensive as each point-cloud scan consists of thousands of points. Existing networks adopt either random sampling [26], [57] or voxelization [27], [28], [29] for reducing the input points. However, both strategies cause point misalignment across two point-cloud views (i.e., x_t and $\Omega(x_t)$). In addition, many point-cloud augmentation strategies such as rotation and scaling alter the 3D coordinates of points, ruling out the possibility of nearest distance search across two point-cloud views. The concatenation of these operations makes efficient point-wise matching complicated and challenging. One solution is to build point-wise correspondence offline [51], but it constricts the variation of training data and also incurs great overhead in computation and storage space.

We develop a simple yet effective approach that can perform efficient point-wise matching across two point-cloud views. Specifically, after loading a LiDAR point-cloud scan as an array, we assign a unique digital identity to each point which is encoded based on the point position in the array and the position of the LiDAR scan in the dataset. While matching points in two views, we search for the intersections of point identities. The resultant indexes enable a direct retrieval of corresponding

Algorithm 1: Pseudocode of SCT in a Pytorch-like style.

```
# i: index of the current target LiDAR scan
# x_t: point cloud of current target scan i
# F_T, F_S: teacher, student segmentation network

import numpy as np
F_T.params = F_S.params# initialize
F_T.params.detach()# no back-propagate gradients for teacher
# Data Preprocessing
pt_num = x_t.shape[0]# point number
ids = np.arange(pt_num)
ids = ids + (i << 32)# assign a unique id for each point
x_t2, ids2 = Omega(x_t,ids)# spatial perturbation
x_t1, ids1 = aug(x_t, ids)# randomly augmentation
x_t2, ids2 = aug(x_t2, ids2)# randomly augmentation

# Fast Point-wise Matching
co_ids = np.intersect1d(ids1, ids2) # ids of co-existed points in two views
sorter = np.argsort(ids1)
m_ids1 = sorter[np.searchsorted(ids1, co_ids, sorter=sorter)]
sorter = np.argsort(ids2)
m_ids2 = sorter[np.searchsorted(ids2, co_ids, sorter=sorter)]

# Forward to Model
pred_t1 = F_T.forward(x_t1)
pred_t2 = F_S.forward(x_t2)
# supervised loss and consistency loss
loss_s = CrossEntropyLoss(pred_t1, labels_t1)
loss_t = CrossEntropyLoss(pred_t2[m_ids2], pred_t1[m_ids1].detach())
loss = loss_s + lambda_t*loss_t

# Update Network
loss.backward()
update(F_S.params)# update student
F_T.param = beta*F_T.param + (1-beta)*F_S.param # ema update teacher
```

point-wise logits from both views, leading to point pairs that can be exploited to compute the spatial consistency loss efficiently. Algorithm 1 provides pseudocode for Fast Point-wise Matching in a Pytorch and Numpy-like style.

D. Spatial Consistency Strategies

We design three types of spatial consistency in SCT for learning domain-invariant point cloud representations that are tolerant to spatial perturbations Ω on target point clouds. More details about the three types of spatial consistency and the corresponding spatial perturbations are to be described in the ensuing three subsections.

1) Geometric-Transform Consistency: The spatial distribution of points in 3D LiDAR point clouds of different domains can vary greatly due to different configurations and viewpoints of LiDAR sensors which can lead to significant differences in geometric structures of point clouds. We introduce geometric perturbations by randomly applying a set of geometric transformations to the target point clouds, such as rotation, scaling, and flipping. The geometric consistency training can thus be achieved by enforcing the model to produce consistent predictions on the spatially transformed and original point clouds as illustrated in Fig. 2(a). This consistency strategy guides the model to learn domain-invariant geometric features and enhances its ability to adapt to different domains.

2) Sparsity Consistency: The sparsity/density of 3D LiDAR point clouds also varies across domains due to different sensor settings (e.g., laser beam number, field of view, etc.) or environments, and such variation can greatly degrade the model's inter-domain recognition performance. We introduce sparsity perturbations by randomly masking a certain portion σ of input points to generate a sparse view of point clouds. Sparsity consistency training can thus be achieved by enforcing the model to produce consistent predictions across the original and the sparsified point clouds as illustrated in Fig. 2(b). Note that during training, we only mask out points by setting their values, including XYZ coordinates and intensity, to zero, while keeping the remaining labels unaltered. This consistency strategy encourages the model to learn sparsity-tolerant but semantic-invariant features, which helps the trained model better adapt across domains with different point-cloud sparsity.

3) Mixing Consistency: Semantic segmentation models often rely on various local contexts in recognition tasks. However, the reliance on such spatial priors in the source domain often misleads the model's recognition in the target domain due to spatial distribution variance across domains. To address this, we introduce spatial context perturbations by randomly mixing points from other LiDAR point-cloud scans which directly modifies the local context of the current LiDAR scan. The consistency with local contexts can thus be achieved by enforcing the prediction consistency between the original and mixed views as illustrated in Fig. 2(c). This consistency induces 3D segmentation models to learn local context-invariant representations, resulting in enhanced recognition ability in the target domain. In the implementation, we adopt the recent PolarMix [4] for context perturbation as PolarMix enriches the local distribution of the mixing data while preserving LiDAR data fidelity.

IV. Experiments

A. Experimental Setup

1) Datasets: We conducted comprehensive experiments to validate the effectiveness of our SCT. The experiments were performed over two challenging synthetic-to-real benchmarks on domain adaptive 3D LiDAR semantic segmentation tasks: SynLiDAR [1] → SemanticKITTI [11] and SynLiDAR → SemanticPOSS [3]. The two benchmarks involve three LiDAR point cloud datasets of road scenes as listed:

- *SynLiDAR* is a large-scale synthetic LiDAR point cloud dataset with 198,396 LiDAR scans and point-level annotations of 32 semantic classes. This large-scale dataset was meticulously collected from nine realistic virtual environments constructed using Unreal Engine 4, including cities, towns, harbors, etc. The data acquisition process involved utilizing a cutting-edge LiDAR simulator capable of generating scans with 64 beam numbers. Following prior studies in [1], [2], we use the officially provided subset with 19,840 scans in our experiments.

- *SemanticKITTI* is a large-scale real LiDAR point cloud dataset collected in Germany. It was collected using a Velodyne HDL-64E LiDAR with 64 laser beams. The dataset consists of 43,552 LiDAR scans with point-wise annotations of 22 semantic classes. We use sequences 00-10 for training, except sequence 08 for validation, following the official split.
- *SemanticPOSS* consists of 2,988 real-world scans with point-level annotations over 14 semantic classes. It was collected on campus using a Pandora LiDAR sensor equipped with 40 laser channels, leading to distinctive spatial distributions that set it apart from the SemanticKITTI dataset. We use sequence 03 for validation and the remaining sequences for training, as per the official benchmark guidelines.

We evaluate our models using per-class Intersection-over-Union (IoU) and mean IoU (mIoU) metrics. Following prior studies [1], [2], we measure IoU and mIoU over 19 classes for SynLiDAR → SemanticKITTI, and 13 shared classes for SynLiDAR → SemanticPOSS.

2) Implementation Details: To ensure fair comparisons, we follow [1], [4] and adopt MinkowskiNet [27] as the backbone model for both teacher and student. MinkowskiNet is a sparse convolutional network with U-Net structure,[1] which stands as the state-of-the-art with superior accuracy and efficiency in 3D point cloud segmentation. We first pre-train the network with cross-entropy loss over source data for 15 epochs by using SGD optimizer with a learning rate of 0.01, momentum of 0.9, and weight decay of 1.e-4, with a batch size of 4. When adapting the model to the target domain, we first initialize the student and teacher models with the pre-trained weights and then train another 5 epochs with SGD optimizer. We set the learning rate to 0.001, momentum to 0.9, and weight decay to 1.e-4, with a batch size of 2 for both source and target data. The hyperparameters λ_t and β are set at 0.1 and 0.99, respectively. As for Ω in different spatial consistency strategies: For geometric-transform consistency, we rotate point clouds along the z-axis between $[-\pi/2, \pi/2]$, scale between $[0.95, 1.05]$, and flip along the x- or y-axis with 50% chance; For sparsity consistency, we randomly mask 50% of points, i.e. $\sigma = 0.5$; For mixing consistency, we implement random 90° scene-level swapping and three times instance-level rotate-pasting for PolarMix [4]. We use TorchSparse library [28] for implementation. All experiments are conducted on one NVIDIA RTX2080Ti with 11 GB GPU memory.

B. Ablation Studies

We conduct comprehensive ablation studies to evaluate the effectiveness of the proposed SCT framework. We report five models over SynLiDAR → SemanticPOSS including:

TABLE I
ABLATION STUDY OF DIFFERENT SPATIAL CONSISTENCY STRATEGIES OVER DOMAIN ADAPTIVE 3D LIDAR SEGMENTATION TASK SYNLIDAR → SEMANTICPOSS

Model	GT-CT	S-CT	M-CT	mIoU
Source-only				20.7
(a)	✓			41.7
(b)	✓	✓		44.6
(c)	✓		✓	43.7
(d)	✓	✓	✓	**46.3**

Geometric-transform consistency training (GT-CT) significantly increases domain generalized 3D segmentation performance. Incorporating sparsity consistency training (S-CT) or mixing consistency training (M-CT) with GT-CT further improves the target segmentation performance clearly. In addition, the combination of all three types of spatial consistency training achieves the best performance, demonstrating the synergic relation among our three designs.
The bold entity highlight data points showcasing the best performance.

1) *Source-only* that is trained using supervised loss $\mathcal{L}_s$ in (1) only, without involving target data in the training process;
2) *Model (a)* that performs geometric-transform consistency training over target data and supervised learning over source data;
3) *Model (b)* that further incorporates sparsity consistency training on top of the model (a);
4) *Model (c)* that incorporates mixing consistency training on top of the model (a); and 5) the full SCT *Model (d)* that combines geometric-transform consistency, sparsity consistency, and mixing consistency in training with target data, as well as supervised learning for source data.

The experimental results are summarized in Table I. As expected, the *Source-only* model trained with SynLiDAR performs poorly due to the clear domain discrepancy. However, we observe a significant improvement in performance with the *Geometric-transform consistency training*, which outperforms the *Source-only* model by a large margin. In addition, incorporating *Sparsity consistency training* and *mixing consistency training* leads to further improvement in the adaptation, demonstrating the effectiveness of our designed spatial consistency strategies. Notably, the full SCT model achieves the best segmentation performance, indicating that the three types of spatial consistency strategies are complementary and synergistic in domain adaptive point cloud segmentation.

C. Comparison With State-of-The-Arts

We compared our spatial consistency training method with a number of state-of-the-art UDA methods. Tables II and III show experimental results over the tasks SynLiDAR → SemanticKITTI and SynLiDAR → SemanticPOSS, respectively. As the two tables show, our method outperforms all state-of-the-art UDA methods clearly and consistently across both tasks, achieving improvements of +3.8 and +5.9 percent points over the state-of-the-art [2], respectively. The superior segmentation performance demonstrates that the proposed spatial consistency training is indeed an advanced method of domain adaptive semantic segmentation for 3D LiDAR point clouds.

We also qualitatively compare our spatial consistency training with the *Source-only* and the state-of-the-art CoSMix [2] over

[1] We use MinkowskiNet_0.5. More details can be found at https://github.com/xiaoaoran/polarmix

TABLE II
EXPERIMENTS ON UNSUPERVISED DOMAIN ADAPTATION WITH SYNLIDAR (AS SOURCE) AND SEMANTICKITTI (AS TARGET)

Method	car	bi.cle	mt.cle	truck	oth-v.	pers.	bi.clst	mt.clst	road	parki.	sidew.	oth-g.	build.	fence	veget.	trunk	terra.	pole	traf.	mIoU
Source-only [1]	42.0	5.0	4.8	0.4	2.5	12.4	43.3	1.8	48.7	4.5	31.0	0.0	18.6	11.5	60.2	30.0	48.3	19.3	3.0	20.4
ADDA [60]	52.5	4.5	11.9	0.3	3.9	9.4	27.9	0.5	52.8	4.9	27.4	0.0	61.0	17.0	57.4	34.5	42.9	23.2	4.5	23.0
Ent-Min [61]	58.3	5.1	14.3	0.6	1.8	14.3	44.5	0.5	50.4	4.3	34.8	0.0	48.3	19.7	67.5	34.8	52.0	33.0	6.1	25.8
ST [62]	62.0	5.0	12.4	1.3	9.2	16.7	44.2	0.4	53.0	2.5	28.4	0.0	57.1	18.7	69.8	35.0	48.7	32.5	6.9	26.5
PCT [1]	53.4	5.4	7.4	0.8	10.9	12.0	43.2	0.3	50.8	3.7	29.4	0.0	48.0	10.4	68.2	33.1	40.0	29.5	6.9	23.9
ST-PCT [1]	70.8	7.3	13.1	1.9	8.4	12.6	44.0	0.6	56.4	4.5	31.8	0.0	66.7	23.7	73.3	34.6	48.4	39.4	11.7	28.9
PolarMix [4]	76.3	**8.4**	17.8	3.9	6.0	26.6	40.8	15.9	70.3	0.0	44.4	0.0	**68.4**	14.7	69.6	38.1	37.1	40.6	10.6	31.0
CoSMix [2]	75.1	6.8	29.4	**27.1**	**11.1**	22.1	25.0	**24.7**	**79.3**	**14.9**	46.7	**0.1**	53.4	13.0	67.7	31.4	32.1	37.9	**13.4**	32.2
SCT (Ours)	**81.9**	3.7	**31.2**	1.6	7.4	**44.6**	**61.1**	3.4	78.2	3.5	**51.2**	0.0	68.0	**31.7**	**74.3**	**45.8**	**51.0**	**41.7**	4.1	**36.0**

SCT outperforms all typical and state-of-the-art methods consistently by large margins.
The bold entities highlight data points showcasing the best performance.

TABLE III
EXPERIMENTS ON DOMAIN ADAPTIVE SEMANTIC SEGMENTATION FROM SYNLIDAR (AS SOURCE) TO SEMANTICPOSS (AS TARGET)

Method	pers.	rider	car	trunk	plants	traf.	pole	garb.	buil.	cone.	fence	bike	grou.	mIoU
Source-Only	3.7	25.1	12.0	10.8	53.4	0.0	19.4	12.9	49.1	3.1	20.3	0.0	59.6	20.7
ADDA [60]	27.5	35.1	18.8	12.4	53.4	2.8	27.0	12.2	64.7	1.3	6.3	6.8	55.3	24.9
Ent-Min [61]	24.2	32.2	21.4	18.9	61.0	2.5	36.3	8.3	56.7	3.1	5.3	4.8	57.1	25.5
ST [62]	23.5	31.8	22.0	18.9	63.2	1.9	**41.6**	13.5	58.2	1.0	9.1	6.8	60.3	27.1
PCT [1]	13.0	35.4	13.7	10.2	53.1	1.4	23.8	12.7	52.9	0.8	13.7	1.1	66.2	22.9
ST-PCT [1]	28.9	34.8	27.8	18.6	63.7	4.9	41.0	16.6	64.1	1.6	12.1	6.6	63.9	29.6
PolarMix [4]	32.6	39.1	25.0	11.9	64.2	5.8	29.6	15.3	44.8	13.3	23.8	10.7	79.0	30.4
CoSMix [2]	55.8	51.4	**36.2**	23.5	**71.3**	22.5	34.2	**28.9**	66.2	20.4	24.9	10.6	78.7	40.4
SCT (Ours)	**57.1**	**54.3**	24.5	**52.3**	62.1	**40.3**	37.6	2.5	**69.7**	**31.7**	**42.7**	**47.6**	**79.5**	**46.3**

SCT outperforms all typical and state-of-the-art methods consistently by large margins.
The bold entities highlight data points showcasing the best performance.

SynLiDAR → SemanticKITTI. As Fig. 3 shows, the *Source-only* produces lots of false predictions due to domain bias. For CoSMix, many confident yet false predictions are selected as pseudo labels which accumulate in the iterative self-training process and finally impair the trained model. Differently, our SCT minimizes the divergence of predictions across point views with respect to different spatial perturbations and learns robust feature representation of the target domain, achieving superior segmentation performance over point clouds in the target domain.

Despite its superior adaptive segmentation performance, SCT still struggles under certain circumstances. One typical scenario happens when a large portion of segmentation failures belongs to long-tail classes that have very limited training samples. Such a lack of training data often degrades representation learning and compromises the adaptability of the learnt model. Besides, the checkpoint selection aiming to optimize mIoU potentially underplays the performance of long-tail classes as well.

It's worth noting that our SCT framework requires minimal computational resources, utilizing only one NVIDIA GTX2080Ti with 11 GB of GPU memory. In contrast, the state-of-the-art CosMix [2] requires much more powerful hardware, utilizing 4×NVIDIA A100 GPUs (each with 40 GB SXM4). We will release our code as a strong baseline repository, lowering the research barrier and facilitating future research in domain adaptive 3D LiDAR segmentation.

Real-to-real adaptation: The proposed SCT can also handle real-to-real adaptation across LiDAR datasets with different numbers of LiDAR beam lines. Following CoSMix (detailed in its appendix), we performed evaluations on SemanticPOSS (40-line)–SemanticKITTI (64-line) where point clouds are captured by LiDAR sensors of different beam line numbers. As Table IV shows, SCT outperforms CoSMix clearly, indicating its robustness and generalization ability in domain-adaptive LiDAR point cloud segmentation.

TABLE IV
ADAPTATION RESULTS ON SEMANTICPOSS → SEMANTICKITTI

Method	mIoU
Source-only	22.5
CoSMix [2]	26.4
SCT (Ours)	**29.4**

The bold entity highlight data points showcasing the best performance.

D. Analysis

We conducted comprehensive experiments to analyse the proposed SCT. The experimental results and findings are detailed in the subsequent subsections.

1) *Varying λ_t*: We examined the effect of parameter λ_t in (4), which balances the supervised loss in the source domain and the unsupervised spatial consistency loss in the target domain. Table V shows experimental results over the task SynLiDAR → SemanticPOSS. We can see that different λ_t produce only moderate variations in mIoU, and all of them outperform the source-only model (i.e., $\lambda_t = 0.0$) significantly. The best mIoU is achieved when

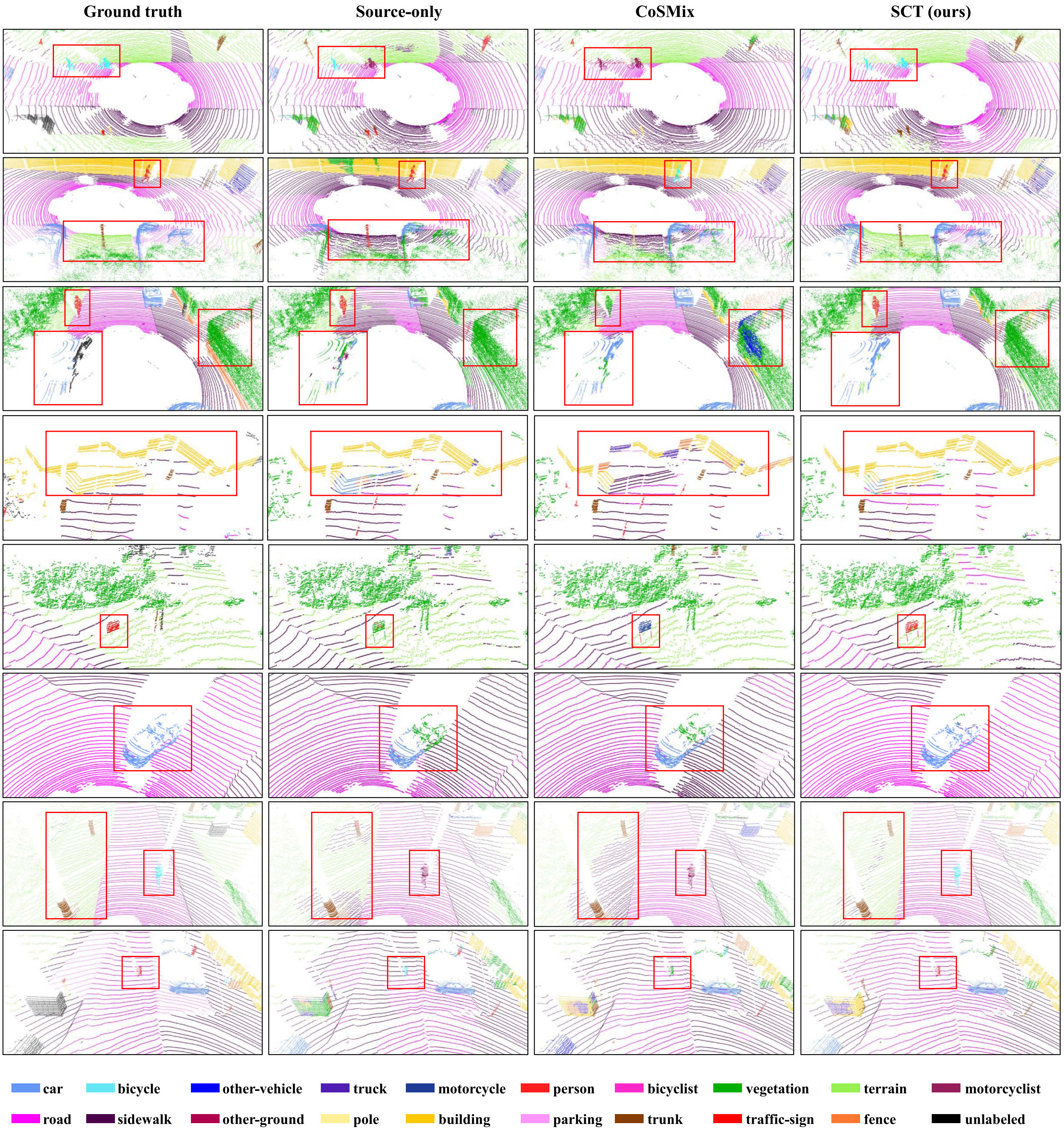


Fig. 3. Qualitative comparison of SCT with the *Source-only* (with no adaptation) and the state-of-the-art CoSMix [2] in domain adaptive 3D LiDAR semantic segmentation. The comparison was conducted over the task 'SynLiDAR → SemanticKITTI'. The 'Ground truth' denotes the ground-truth annotations. The red rectangles highlight regions of interest. Best viewed in color.

TABLE V

PERFORMANCE OF SPATIAL CONSISTENCY TRAINING UNDER DIFFERENT λ_t (THE BALANCE WEIGHT OF SPATIAL CONSISTENCY LOSS AS DEFINED IN (4)) ON THE SYNLIDAR → SEMANTICPOSS UDA TASK

λ_t	0.00	0.05	0.10	0.15	0.20	1.00
mIoU	20.4	44.5	**46.3**	46.1	45.6	44.7

The bold entity highlight data points showcasing the best performance.

$\lambda_t = 0.10$. The experiments show that our proposed SCT is tolerant to the variation in balance weight λ_t.

2) *Varying* β**:** We employ the momentum parameter β to update the teacher model. When β is set to 0, the teacher model is equivalent to the student model with no temporal momentum update. We examine the impact of different values of β in Table VI. We can see that the model performs much better with the exponential moving average

TABLE VI
EVALUATION OF THE PERFORMANCE OF SPATIAL CONSISTENCY TRAINING MODELS WITH VARYING MOMENTUM UPDATE WEIGHT β ON THE SYNLIDAR → SEMANTICPOSS TASK

β	0.0	0.9	0.99	0.999
mIoU	32.3	45.5	**46.3**	45.8

The bold entity highlight data points showcasing the best performance.

TABLE VII
SEGMENTATION PERFORMANCE OF SPATIAL CONSISTENCY TRAINING ON SYNLIDAR → SEMANTICPOSS WITH COMBINATION OF DIFFERENT GEOMETRIC TRANSFORMATIONS

Method	(a)	(b)	(c)
rotation	✓	✓	✓
scaling		✓	✓
flipping			✓
mIoU	45.0	45.9	**46.3**

The bold entity highlight data points showcasing the best performance.

TABLE VIII
RESULTS OF OUR SPARSITY CONSISTENCY WITH DIFFERENT PROPORTIONS OF SPARSITY σ OVER SYNLIDAR → SEMANTICPOSS

σ	0.0	0.3	0.5	0.7
mIoU	43.7	45.3	**46.3**	46.0

The bold entity highlight data points showcasing the best performance.

and it performs the best when β is set to 0.99, indicating that a slowly progressing teacher model is beneficial. At the other end, the teacher model updates too slowly to capture the latest representative network parameters with a very high β, and it updates too fast and leads to less robust and unstable temporal ensembles with a very low β. Both scenarios impair the trained cross-domain segmentation models.

3) *Geometric Transformations:* We study how different geometric transformations affect adaptation performance. Table VII presents experimental results under several typical geometric transformations including *rotation*, *scaling*, and *flipping*. It can be observed that learning under different geometric perturbations improves the adaptation process and more complex geometric perturbations are helpful in enhancing the target performance.

4) *Sparsity Ratio:* We investigated the impact of sparsity ratios σ in sparsity consistency training. As Table VIII shows, incorporating sparsity consistency consistently leads to clear improvements in target segmentation compared to the baseline (i.e., $\sigma = 0$) while the best segmentation performance is achieved when $\sigma = 0.5$. The experiments reveal that setting an appropriate sparsity ratio is important as a large σ provides limited sparsity perturbations while a small σ tends to lose necessary geometric information for point recognition.

5) *Different Consistency Losses***:** We also evaluated the mean squared error (MSE) loss between the teacher's and student's predictions for consistency regularization:

$$\mathcal{L}_{mse} = \frac{1}{N_t}\sum_{i=1}^{N_t}\frac{1}{n_t^i}\sum_{j=1}^{n_t^i}\left(F_T(x_t^{i,j}) - F_S(\Omega(x_t^{i,j}))\right)^2 \tag{5}$$

TABLE IX
PERFORMANCE OF SPATIAL CONSISTENCY TRAINING WITH TWO UNSUPERVISED LOSSES: CROSS-ENTROPY LOSS ($\mathcal{L}_{ce}$) AS DEFINED IN (2), AND MEAN SQUARED ERROR LOSS ($\mathcal{L}_{mse}$) AS DEFINED IN (5)

Consistency Loss	mIoU
N.A.	20.7
$\mathcal{L}_{mse}$	38.2
$\mathcal{L}_{ce}$	**46.3**

Results are shown over synLiDAR → semanticPOSS.
The bold entity highlight data points showcasing the best performance.

Table IX shows experimental results on SynLiDAR → SemanticPOSS, where L_{ce} denotes the cross-entropy loss defined in (2). We can observe that optimizing both unsupervised losses outperforms the Source-only (N.A.) significantly, validating the effectiveness of spatial consistency training. In addition, optimizing the cross-entropy loss leads to significantly better performance, largely because the re-trained student model is supervised with *hard* pseudo-labels, which help to minimize prediction entropy.

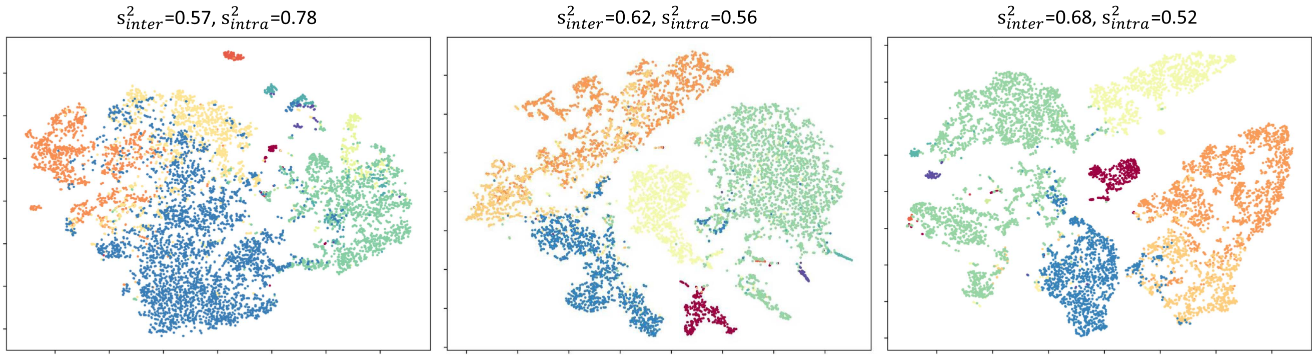
6) *Features Visualization:* To better assess the proposed SCT, we employ t-SNE [61] to visualize point cloud representations of the target domain. Fig. 4 shows the feature visualizations for the source-only model, the state-of-the-art CoSMix [2], and our proposed SCT, respectively. We can observe that the SCT-produced features have clearly better discriminability than those produced by the source-only model, highlighting the outstanding adaptation performance of the proposed SCT. Additionally, SCT also produces more discriminative target features than CoSMix, achieving the largest inter-class variance while maintaining the smallest intra-class variance. This suggests that the upstream class-wise representations from SCT are more discernible, making it a reliable indicator of its effectiveness.

TABLE X
SELECTING PSEUDO LABELS BY THRESHOLDING THEIR PREDICTION PROBABILITIES

δ	0.0	0.5	0.9	0.95	0.99
mIoU	**46.3**	44.8	44.9	44.3	42.2

With different thresholds $\delta \in [0, 1)$, the proposed spatial consistency training learns from different pseudo labels with different segmentation over synLiDAR → semanticPOSS.
The bold entity highlight data points showcasing the best performance.

7) *Pseudolabel Threshold:* We employ all pseudo labels in the spatial consistency training. At the other end, it is possible to adopt thresholding to select confident pseudo labels only in the spatial consistency training. Table X



(a) Source-only (b) CoSMix (c) SCT (ours)

Fig. 4. Feature space visualization with t-SNE [61] on SynLiDAR → SemanticKITTI UDA task. The proposed *SCT* learns more compact feature space for target domain with smaller intra-class variance and larger inter-class variance as compared with the *Source only* and the state-of-the-art *CoSMix* [2]. Different colors denote different classes and best viewed in color.

TABLE XI
Training Resource Usage for CoSMix and Our Method SCT Over SynLiDAR → SemanticKITTI

Method	CoSMix [27]	SCT (Ours)
training time	4.6 hours	**2.2 hours**
GPU usage	4×V100 (4x32GB)	**1x2080Ti(1x11GB)**
mIoU	32.3	**36.0**

The bold entity highlight data points showcasing the best performance.

TABLE XII
Different Sampling Strategies for Sparsity Consistency in SCT, Including Random Sampling ("RS"), Grid Sampling ("GS"), and Distance-Based Sampling ("DS", DS(f)/DS(c) Denoting Higher Sampling Weights Assigned to Farther/Closer Points)

Method	N.A.	RS	GS	DS(f)	DS(c)
mIoU	43.7	**46.3**	**46.3**	46.0	45.2

Results are shown over synLiDAR → semanticPOSS.
The bold entities highlight data points showcasing the best performance.

TABLE XIII
Unsupervised Domain Adaptative Point Cloud Segmentation With the Backbone SPVCNN [28] (On SynLiDAR → SemanticKITTI)

Method	mIoU
Source-Only	23.7
SCT (Ours)	**31.3**

SCT improves UDA consistently with different backbone models.
The bold entity highlight data points showcasing the best performance.

shows relevant experiments, where applying different thresholds δ degrades cross-domain segmentation consistently. We conjecture that the thresholding could produce many confident but false predictions which lead to a deviated solution with error propagation in training. Differently, employing all pseudo labels enables more comprehensive and robust adaptive learning in the target domain.

8) *Training Time Comparsion:* We compare SCT with CoSMix [2] to validate its superior computational efficiency. For CosMix, we use its official code with default configurations and train with four NVIDIA V100 GPUs over the benchmark SynLiDAR → SemanticKITTI. As Table XI shows, SCT (using a single NVIDIA 2080Ti) can be trained much faster than CosMix Ti. Besides, it achieves clearly better mIoU. The experiments highlight the potential of SCT which as a powerful tool could greatly reduce the research barrier and facilitate future research in domain adaptive 3D LiDAR segmentation.

9) *More Analysis for Sampling Strategy:* Targeting a simple, efficient, and effective base technique in point cloud learning, we adopted random sampling (RS) in sparsity consistency design. We tested more sophisticated sampling techniques including grid sampling (GS) and distance-based sampling (DS). Table XII shows experiments on SynLiDAR → SemanticPOSS, where DS(f)/DS(c) means assigning higher sampling weights to farther/closer points. We can see that GS performs similarly to RS while DS(c) performs clearly worse, suggesting that sampling should prioritize nearer and denser points. In addition, all sampling strategies outperform N.A. without using sparsity consistency, validating our finding on maintaining sparsity invariance in cross-domain LiDAR segmentation

10) *SCT with Different Backbone Models:*The proposed SCT is model-agnostic and can work with different backbone models. We verify this by implementing it with another widely adopted 3D segmentation model SPVCNN [28]. As shown in Table XIII, SCT outperforms the Source-only clearly on SynLiDAR → SemanticKITTI, demonstrating its superior robustness and generalization across different backbone models.

11) *Failure Analysis:* Most segmentation failures with SCT are associated with long-tail classes (such as "mt.clst." in Table II and "garb." in Table III) that have very limited

training samples and thereby often suffer from clear overfitting. In addition, our checkpoint selection prioritizes optimizing mIoU which inadvertently sacrifices the performance of long-tail classes. This issue could be alleviated by developing some class-balanced UDA approach that mitigates the long-tail distribution by balancing the data distribution across classes.

V. Conclusion

This paper proposes a novel spatial consistency training framework for addressing the domain shift problem in 3D LiDAR point cloud segmentation. The approach enforces prediction consistency between raw point clouds and their spatially perturbed views, guiding the segmentation network to learn domain-invariant feature representations across domains. Three novel spatial consistency strategies tailored to the data properties of LiDAR point clouds are introduced to facilitate effective consistency training in 3D space, namely geometric-transform consistency, sparsity consistency, and mixing consistency. Comprehensive experimental results demonstrate that the proposed spatial consistency training significantly improves the performance of 3D UDA tasks as compared with the state-of-the-art. In the future, we plan to investigate more effective spatial consistency strategies to further enhance the performance of our framework.

References

[1] A. Xiao, J. Huang, D. Guan, F. Zhan, and S. Lu, “Transfer learning from synthetic to real LiDAR point cloud for semantic segmentation,” in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, pp. 2795–2803.

[2] C. Saltori et al., “Cosmix: Compositional semantic mix for domain adaptation in 3D LiDAR segmentation,” in *Eur. Conf. Comput. Vis.*, 2022, pp. 586–602.

[3] Y. Pan et al., “SemanticPOSS: A point cloud dataset with large quantity of dynamic instances,” in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2020, pp. 687–693.

[4] A. Xiao et al., “Polarmix: A general data augmentation technique for LiDAR point clouds,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 11035–11048.

[5] L. Yi, B. Gong, and T. Funkhouser, “Complete & label: A domain adaptation approach to semantic segmentation of LiDAR point clouds,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15363–15373.

[6] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer, “SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud,” in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 4376–4382.

[7] S. Zhao et al., “EpointDA: An end-to-end simulation-to-real domain adaptation framework for LiDAR point cloud segmentation,” in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 3500–3509.

[8] N. Araslanov and S. Roth, “Self-supervised augmentation consistency for adapting semantic segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15384–15394.

[9] L. Melas-Kyriazi and A. K. Manrai, “Pixmatch: Unsupervised domain adaptation via pixelwise consistency training,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 12435–12445.

[10] A. Tarvainen and H. Valpola, “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 1195–1204.

[11] J. Behley et al., “Semantic KITTI: A dataset for semantic scene understanding of LiDAR sequences,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9297–9307.

[12] W. K. Fong et al., “Panoptic nuscenes: A large-scale benchmark for LiDAR panoptic segmentation and tracking,” *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 3795–3802, Apr. 2022.

[13] R. Loiseau, M. Aubry, and L. Landrieu, “Online segmentation of LiDAR sequences: Dataset and algorithm,” in *Eur. Conf. Comput. Vis.*, 2022, pp. 301–317.

[14] H. Liu, Y. Guo, Y. Ma, Y. Lei, and G. Wen, “Semantic context encoding for accurate 3D point cloud segmentation,” *IEEE Trans. Multimedia*, vol. 23, pp. 2045–2055, 2021.

[15] C. Chen, S. Qian, Q. Fang, and C. Xu, “HAPGN: Hierarchical attentive pooling graph network for point cloud segmentation,” *IEEE Trans. Multimedia*, vol. 23, pp. 2335–2346, 2021.

[16] T. Weng, J. Xiao, F. Yan, and H. Jiang, “Context-aware 3D point cloud semantic segmentation with plane guidance,” *IEEE Trans. Multimedia*, vol. 25, pp. 6653–6664, 2023.

[17] L. Zhao et al., “LIF-Seg: LiDAR and camera image fusion for 3D LiDAR semantic segmentation,” *IEEE Trans. Multimedia*, to be published, doi: 10.1109/TMM.2023.3277281.

[18] B. Wu, A. Wan, X. Yue, and K. Keutzer, “SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1887–1893.

[19] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, “RangeNet++ : Fast and accurate LiDAR semantic segmentation,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4213–4220.

[20] C. Xu et al., “SqueezeSegV3: Spatially-adaptive convolution for efficient point-cloud segmentation,” in *Eur. Conf. Comput. Vis.*, 2020, pp. 1–19.

[21] T. Cortinhal, G. Tzelepis, and E. E. Aksoy, “Salsanext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds,” in *Int. Symp. Vis. Comput.*, 2020, pp. 207–222.

[22] A. Xiao, X. Yang, S. Lu, D. Guan, and J. Huang, “FPS-Net: A convolutional fusion network for large-scale LiDAR point cloud segmentation,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 176, pp. 237–249, 2021.

[23] Y. Zhang et al., “PolarNet: An improved grid representation for online LiDAR point clouds semantic segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9601–9610.

[24] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “PointNet: Deep learning on point sets for 3D classification and segmentation,” in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2017, pp. 652–660.

[25] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “PointNet : Deep hierarchical feature learning on point sets in a metric space,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5105–5114.

[26] Q. Hu et al., “Randla-Net: Efficient semantic segmentation of large-scale point clouds,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11108–11117.

[27] C. Choy, J. Gwak, and S. Savarese, “4D spatio-temporal ConvNets: Minkowski convolutional neural networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3075–3084.

[28] H. Tang et al., “Searching efficient 3D architectures with sparse point-voxel convolution,” in *Proc. Eur. Conf. Computer Vis.*, 2020, pp. 685–702.

[29] X. Zhu et al., “Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 9939–9948.

[30] Y. Hou, X. Zhu, Y. Ma, C. C. Loy, and Y. Li, “Point-to-voxel knowledge distillation for LiDAR semantic segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8479–8488.

[31] Y. Zhou and O. Tuzel, “VoxelNet: End-to-end learning for point cloud based 3D object detection,” in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2018, pp. 4490–4499.

[32] B. Graham and L. V. d. Maaten, “Submanifold sparse convolutional networks,” 2017, *arXiv:1706.01307*.

[33] H. Tang, Z. Liu, X. Li, Y. Lin, and S. Han, “Torchsparse: Efficient point cloud inference engine,” in *Proc. Mach. Learn. Syst.*, 2022, vol. 4, pp. 302–315.

[34] M. Jaritz, T.-H. Vu, R. d. Charette, E. Wirbel, and P. Perez, “xMUDA: Cross-modal unsupervised domain adaptation for 3D semantic segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12605–12614.

[35] A. Xiao et al., “3D semantic segmentation in the wild: Learning generalized models for adverse-condition point clouds,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 9382–9392.

[36] A. Xiao et al., “A survey of label-efficient deep learning for 3D point clouds,” 2023, *arXiv:2305.19812*.

[37] F. Langer, A. Milioto, A. Haag, J. Behley, and C. Stachniss, “Domain transfer for semantic segmentation of LiDAR data using deep neural networks,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 8263–8270.

[38] G. Li, G. Kang, X. Wang, Y. Wei, and Y. Yang, "Adversarially masking synthetic to mimic real: Adaptive noise injection for point cloud segmentation adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 20464–20474.

[39] Y. Guo et al., "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4338–4364, Dec. 2021.

[40] G. French, M. Mackiewicz, and M. Fisher, "Self-ensembling for visual domain adaptation," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–13.

[41] M. Sajjadi, M. Javanmardi, and T. Tasdizen, "Regularization with stochastic transformations and perturbations for deep semi-supervised learning," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 1171–1179.

[42] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 1–11.

[43] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019.

[44] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Unsupervised data augmentation for consistency training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 6256–6268.

[45] D. Berthelot et al., "Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–10.

[46] K. Sohn et al., "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 596–608.

[47] W. Tranheden, V. Olsson, J. Pinto, and L. Svensson, "DACS: Domain adaptation via cross-domain mixed sampling," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 1379–1389.

[48] Y. Xing, D. Guan, J. Huang, and S. Lu, "Domain adaptive video segmentation via temporal pseudo supervision," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 621–639.

[49] L. Jiang et al., "Guided point contrastive learning for semi-supervised point cloud semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6423–6432.

[50] Z. Luo et al., "Unsupervised domain adaptive 3D detection with multi-level consistency," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 8866–8875.

[51] S. Xie et al., "Pointcontrast: Unsupervised pre-training for 3D point cloud understanding," in *Eur. Conf. Computer Vis.*, 2020, pp. 574–591.

[52] J. Hou, B. Graham, M. Nießner, and S. Xie, "Exploring data-efficient 3D scene understanding with contrastive scene contexts," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15587–15597.

[53] S. Huang, Y. Xie, S.-C. Zhu, and Y. Zhu, "Spatio-temporal self-supervised representation learning for 3D point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6535–6545.

[54] Y. Luo, J. Zhu, M. Li, Y. Ren, and B. Zhang, "Smooth neighbors on teacher graphs for semi-supervised learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8896–8905.

[55] L. Kong, J. Ren, L. Pan, and Z. Liu, "Lasermix for semi-supervised LiDAR semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 21705–21715.

[56] Z. Luo et al., "Unsupervised domain adaptive 3D detection with multi-level consistency," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 8866–8875.

[57] H. Thomas et al., "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6411–6420.

[58] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2017, pp. 7167–7176.

[59] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2517–2526.

[60] Y. Zou, Z. Yu, B. Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 289–305.

[61] L. V. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.

Aoran Xiao received the B.Sc. and M.Sc. degrees from Wuhan University, Wuhan, China, in 2016 and 2019, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests include point cloud processing, computer vision, and remote sensing.

Dayan Guan received the Ph.D. degree from the Zhejiang University, Hangzhou, China. He is currently a Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His main research interests include computer vision, pattern recognition, and machine learning. For the past few years, he has dedicated his efforts to advancing the fields of semi-supervised learning and domain adaptation, making noteworthy contributions through the development of innovative techniques.

Xiaoqin Zhang (Senior Member, IEEE) received the B.Sc. degree in electronic information science and technology from Central South University, Changsha, China, in 2005, and the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2010. He is currently a Professor with Wenzhou University, Wenzhou, China. He has authored or coauthored more than 100 papers in international and national journals, and international conferences, including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IJCV, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON COMPUTERS, ICCV, CVPR, NIPS, IJCAI, AAAI, and among others. His research interests include pattern recognition, computer vision, and machine learning.

Shijian Lu received the Ph.D. degree in electrical and computer engineering from the National University of Singapore. He is currently an Associate Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His main research interests include image and video analytics, visual intelligence, and machine learning.