Unsupervised Domain Adaptive 3D Detection with Multi-Level Consistency

Zhipeng Luo[1,2*] Zhongang Cai[1,2,3*] Changqing Zhou[2,4*] Gongjie Zhang[4] Haiyu Zhao[2,3]
Shuai Yi[2,3] Shijian Lu[4†] Hongsheng Li[5] Shanghang Zhang[6] Ziwei Liu[1]
[1] S-Lab, Nanyang Technological University [2] Sensetime Research [3] Shanghai AI Laboratory
[4] Nanyang Technological University [5] Chinese University of Hong Kong [6] UC Berkeley
{zhipeng001, zhou0365}@e.ntu.edu.sg, {caizhongang, zhaohaiyu, yishuai}@sensetime.com
{shijian.lu}@ntu.edu.sg, hsli@ee.cuhk.edu.hk, shz@eecs.berkeley.edu, zwliu.hust@gmail.com

Abstract

Deep learning-based 3D object detection has achieved unprecedented success with the advent of large-scale autonomous driving datasets. However, drastic performance degradation remains a critical challenge for cross-domain deployment. In addition, existing 3D domain adaptive detection methods often assume prior access to the target domain annotations, which is rarely feasible in the real world. To address this challenge, we study a more realistic setting, unsupervised 3D domain adaptive detection, which only utilizes source domain annotations. ***1)*** *We first comprehensively investigate the major underlying factors of the domain gap in 3D detection. Our key insight is that geometric mismatch is the key factor of domain shift.* ***2)*** *Then, we propose a novel and unified framework,* ***Multi-Level Consistency Network (MLC-Net)****, which employs a teacher-student paradigm to generate adaptive and reliable pseudo-targets. MLC-Net exploits point-, instance- and neural statistics-level consistency to facilitate cross-domain transfer. Extensive experiments demonstrate that MLC-Net outperforms existing state-of-the-art methods (including those using additional target domain information) on standard benchmarks. Notably, our approach is detector-agnostic, which achieves consistent gains on both single- and two-stage 3D detectors. Code will be released.*

1. Introduction

With the prevalent use of LiDARs for autonomous vehicles and mobile robots, 3D object detection on point clouds has drawn increasing research attention. Large-scale 3D object detection datasets [11, 35, 3] in recent years has empowered deep learning-based models [32, 42, 41, 21, 31, 43, 25, 34, 33, 50, 45] to achieve remarkable success. However, deep learning models trained on one dataset (source domain) often suffer tremendous performance degradation when evaluated on another dataset (target domain). We investigate the bounding box scale mismatch problem (e.g., vehicle size in the U.S. is noticeably larger than that in Germany), which is found to be a major contributor to the domain gap, in alignment with previous work [38]. This is unique to 3D detection: compared to 2D bounding boxes that can have a large variety of size, depending on the distance of the object from the camera, 3D bounding boxes have a more consistent size in the same dataset, regardless of the objects' location relative to the LiDAR sensor. Hence, the detector tends to memorize a narrow and dataset-specific distribution of bounding box size from the source domain (Figure 2).

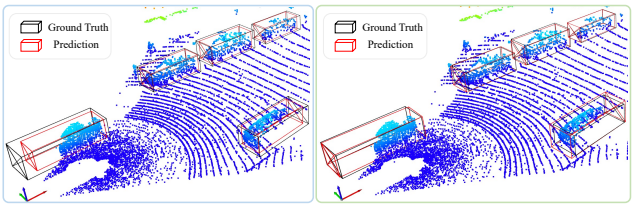

Figure 1: Visualization of detection results for domain adaptation from KITTI to Waymo dataset. **Left**: Predictions of baseline model trained on KITTI dataset and directly tested on Waymo dataset. The model can classify and localize the objects, but produces inaccurate box scale due to geometric mismatch. The predicted boxes are therefore noticeably smaller than the ground truth. **Right**: Predictions of our domain-adaptive MLC-Net, which demonstrates accurate bounding box scale even though MLC-Net is trained without access to any target domain annotations. Best viewed in color.

Unfortunately, existing works are inadequate to address the domain gap with a realistic setup. Recent methods on domain adaptive 3D detection either require some labeled data from the target domain for finetuning or utilize some additional statistics (such as the mean size) of the target domain [38]. However, such knowledge of the target do-

*Equal contribution
†Corresponding author

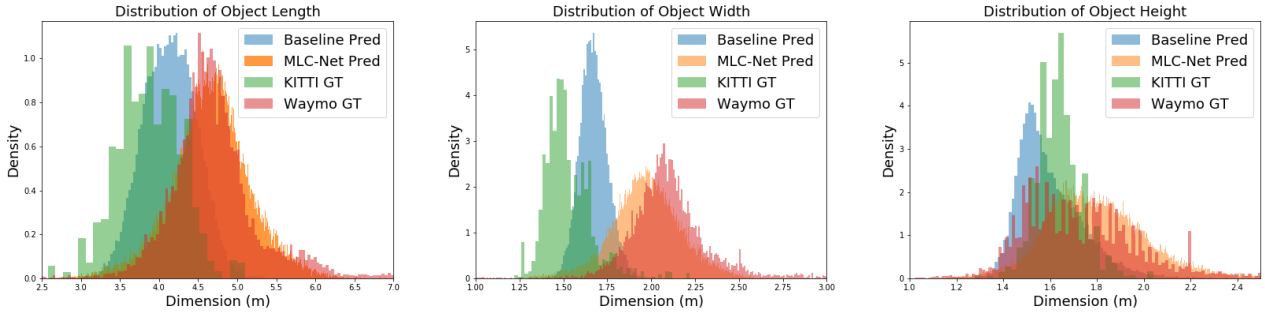

Figure 2: A study on the domain shift for 3D detection. Here we take KITTI as the source dataset and Waymo as the target dataset. Our key insights include: 1) distribution of object dimensions varies drastically across datasets, indicating geometric mismatch can be a key factor for the domain gap; 2) directly applying a model trained on KITTI to Waymo (referred to as the baseline in the figure) is ineffective: the model continues to predict box dimensions close to the source domain; 3) our MLC-Net is effective in addressing the geometric mismatch, and the distributions of its predictions on the target domain accurately align with the ground truth. Best viewed in color.

main is not always available. In addition, popular 2D unsupervised domain adaptation methods that leverage feature alignment techniques [8, 29, 48, 15, 6, 14, 40, 19, 22, 17, 46, 18, 47, 37] to mitigate domain shift are not readily transferable to 3D detection. While these methods are effective in handling domain gaps due to lighting, color, and texture variations, such information is unavailable in point clouds. Instead, point clouds pose unique challenges such as the geometric mismatch discussed above.

Therefore, we propose MLC-Net for unsupervised domain adaptive 3D detection. MLC-Net is designed to tackle two major challenges. First, to create meaningful scale-adaptive targets to facilitate the learning, MLC-Net employs the mean teacher [36] learning paradigm. The teacher model is essentially a temporal ensemble of student models: the parameters of the teacher model are updated by an exponential moving average window on student models of preceding iterations. Our analyses show that the mean teacher produces accurate and stable supervision for the student model without any prior knowledge of the target domain. To the best of our knowledge, we are the first to introduce the mean teacher paradigm in unsupervised domain adaptive 3D detection. Second, to design scale-related consistency losses and construct useful correspondences of teacher-student predictions to initiate gradient flow, we design MLC-Net to enforce consistency at three levels. **1)** Point-level. As point clouds are unstructured, point-based region proposals or equivalents [32, 42] are common. Hence, we sample the same subset of points and share them between the teacher and student. We retain the indices of the points that allow 3D augmentation methods to be applied without losing the correspondences. **2)** Instance-level. Matching region proposals can be erroneous, especially at the initial stage when the quality of region proposals is substandard. Hence, we resort to transferring teacher region proposals to students to circumvent the matching process. **3)** Neural statistics-level. As the teacher model only accesses the target domain input, the mismatch between the batch statistics hinders effective learning. We thus transfer the student's statistics, which is gathered from both the source and the target domain, to the teacher to achieve a more stable training behavior.

MLC-Net shows remarkable compatibility with popular mainstream 3D detectors, allowing us to implement it on both two-stage [32] and single-stage [42] detectors. Moreover, we verify our design through rigorous experiments across multiple widely used 3D object detection datasets [11, 35, 3]. Our method outperforms baselines by convincing margins, even surprisingly surpassing existing methods that utilize additional information. In summary, our main **contributions** are:

- We formulate and study unsupervised domain adaptive 3D detection, a pragmatic, yet underexplored task that requires no annotations of the target domain. We comprehensively investigate the major underlying factors of the domain gap in 3D detection and find geometric mismatch is the key factor.
- We propose a concise yet effective mean-teacher paradigm that leverages three levels of consistency to facilitate cross-domain transfer, achieving a significant performance boost that is consistent across multiple popular public datasets.
- We validate our hypothesis on the unique challenges associated with point clouds and verify our proposed approach with comprehensive evaluations, which we hope would lay a strong foundation for future research.

2. Related Works

LiDAR-based 3D Detection. LiDAR-based 3D detection methods mainly come from two categories, namely grid-based methods and point-based methods. Grid-based approaches convert the whole point cloud scene to voxels of fixed size and process the input with 2D or 3D CNN.

MV3D [7] first projects point clouds to bird-eye view images to generate proposals. PointPilar [21] performs voxelization on point clouds and converts the representation to 2D. VoxelNet [49] obtains voxel representations by applying PointNet [27] to points and processes the features with 3D convolution. SECOND [41] applies 3D sparse convolution [12] to improve the efficiency. PV-RCNN [31] proposes to combine voxelization and point-based set abstraction to obtain more discriminative features. On the other hand, point-based methods directly extract features from raw point cloud input. F-PointNet [26] applies PointNet [27] to perform 3D detection based on 2D bounding boxes. PointRCNN [32] proposes a two-stage framework to generate box bounding proposals from the whole point clouds and refine them with feature pooling. 3DSSD [42] proposes to use F-FPS for better point sampling to achieve single-stage detection. In this work, we conduct focused discussion with PointRCNN as the base model but we show our method is also compatible to single-stage detector (3DSSD) in Supplementary Material.

Point Cloud Domain Adaptation. While extensive researches have been conducted on domain adaptation tasks with 2D image data, the 3D point cloud domain adaptation field has relatively small literature. PointDAN [28] proposes to jointly align local and global features using discrepancy loss and adversarial training for point cloud classification. Achituve et. al. [1] introduces an additional self-supervised reconstruction task to improve the classification performance on the target domain. Yi et. al. [44] designs a sparse voxel completion network to perform point cloud completion for domain adaptive semantic segmentation. Jaritz et. al. [20] leverages multi-modal information by projecting point cloud to 2D images and train models jointly. For object detection, [38] identifies the major domain gap of object size mismatch among autonomous driving datasets and proposes to mitigate the gap by leveraging target domain object scale statistics. SF-UDA [30] computes motion coherence over consecutive frames to select the best scale for the target domain. Our proposed method works under a similar setup to [38] but does not require target domain geometric statistics.

Mean Teacher. The mean teacher framework [36] is first proposed for semi-supervised learning. Many variants [9, 2, 39] have been proposed to further improve its performance. Furthermore, the framework has also been applied to other fields such as domain adaptation [10, 4] and self-supervised learning [16, 13, 24] where labeled data is scarce or unavailable. Specifically, the mean teacher framework incorporates one trainable student model and a non-trainable teacher model whose weights are obtained from the exponential moving average of the student model's weights. The student model is optimized based on the consistency loss between the student and teacher network predictions. In particular, although [4] also employs the mean teacher paradigm for the detection task by aligning region-level features, point cloud detection models are substantially different from 2D detectors and our proposed method differs by incorporating multi-level consistency.

3. Our Approach

In this section, we formulate the 3D point cloud domain adaptive detection problem (Section 3.1), and provide an overview of our MLC-Net (Section 3.2), followed by the details of our mean-teacher paradigm (Section 3.3). Finally, we explain the details of the point-level (Section 3.4), instance-level (Section 3.5), and statistics-level (Section 3.6) consistency of our MLC-Net.

3.1. Problem Definition

Under the unsupervised domain adaptation setting, we have access to point cloud data from one labeled source domain $\mathbb{D}_s = \{x_s^i, y_s^i\}_{i=1}^{N_s}$ and one unlabeled target domain $\mathbb{D}_t = \{x_t^i\}_{i=1}^{N_t}$, where N_s and N_t are the number of samples from the source and target domains, respectively. Each point cloud scene $x^i \in \mathbb{R}^{n\times 3}$ consists of n points with their 3D coordinates while y^i denotes the label of the corresponding training sample from the source domain. y is in the form of object class k and 3D bounding box parameterized by the center location of the bounding box (c_x, c_y, c_z), the size in each dimension (d_x, d_y, d_z), and the orientation η. The goal of the domain adaptive detection task is to train a model F based on $\mathbb{D}_s$ and $\mathbb{D}_t$ and maximize the performance on $\mathbb{D}_t$.

3.2. Framework Overview

We illustrate MLC-Net in Figure 3. The labeled source input x_s is used for standard supervised training of the student model F with loss L_{source}. For each unlabeled target domain example x_t, we perturb it by applying random augmentation h to obtain $\hat{x_t}$. The perturbed and original point cloud inputs are passed to the student model and teacher model respectively to get their point-level box proposals $\hat{R}_t$ and R_t where point-level consistency is applied. Subsequently, teacher proposals are augmented with h and passed to the student model for box refinement, to obtain $\hat{S}_t$. Together with teacher's instance-level predictions S_t, the instance-level consistency is applied. The overall consistency loss $L_{consist}$ is computed as:

$$L_{consist} = L_{pt,cls} + L_{pt,box} + L_{ins,cls} + L_{ins,box} \quad (1)$$

where pt, ins, cls and box stand for point-level, instance-level, classification and box regression respectively. These loss components are elaborated in Section 3.4 and 3.5. In each iteration, the student model is updated through gradient descent with the total loss L, which is a weighted sum

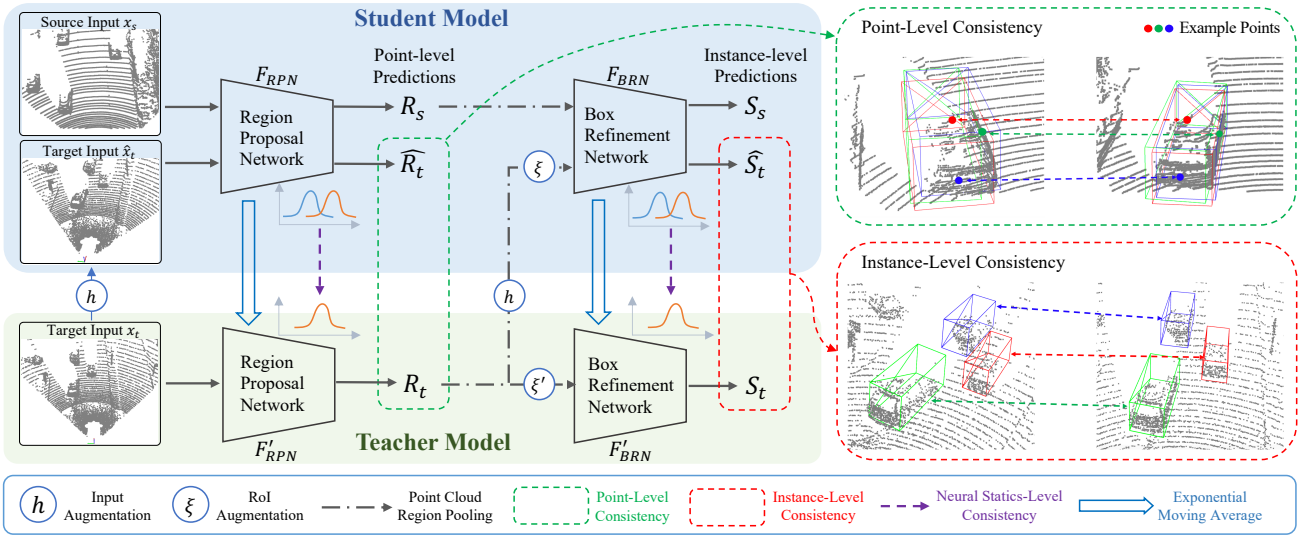

Figure 3: The network architecture of our proposed MLC-Net. MLC-Net leverages the mean-teacher [36] paradigm where the teacher is the exponential moving average (hence the name mean-teacher) of the student model and is updated at every iteration. This mean-teacher design provides high-quality pseudo labels to facilitate smooth learning of the student model. Towards the goal, we design consistency enforced at three levels. First, at point-level, 3D proposals are associated based on point correspondences, which are established by sampling the same set of points from the target domain for both the student and teacher models; second, at instance-level, the teacher 3D proposals are passed to the student Box Refinement Network, and the correspondences between 3D box predictions from two models are naturally maintained; third, at neural statistics-level, we discover non-learnable parameters in batch normalization layers demonstrate significant domain shift, and thus align the teacher's parameters with the student's. We highlight the efficacy of MLC-Net and further discuss our design motivations in Section 3. Best viewed in color.

of L_{source} and $L_{consist}$:

$$L = \lambda L_{source} + L_{consist} \tag{2}$$

where λ is the weight coefficient. The learnable parameters of the student model are then used to update the corresponding teacher model parameters, where the details can be found in Section 3.3. In addition, we enforce non-learnable parameters to be aligned between the teacher and the student via neural statistics-consistency (Section 3.6).

MLC-Net achieves two major design goals towards effective unsupervised 3D domain adaptive detection. **First**, to generate accurate and robust pseudo targets without any access to the target domain annotation or statistical information. MLC-Net leverages a mean teacher paradigm where the teacher model can be regarded as a temporal ensemble of student models, allowing it to produce high-quality predictions and guide the learning of the student. **Second**, to design effective consistency losses at point-, instance- and neural statistics-level that enhance adaptability to scale variation, and construct the teacher-student correspondences that allow the back-propagated gradient to flow through the correct routes. Although we conduct most analysis on PointRCNN as the representative of two-stage 3D detectors, we highlight that our method is generic and can be easily extended to single-stage detection models such as 3DSSD with modest modifications (see Supplementary Material).

3.3. Mean Teacher

Motivated by the success of the mean teacher paradigm [36] in semi-supervised learning and self-supervised learning, we apply it to our point cloud domain adaptive detection task as illustrated in Figure 3. The framework consists of a student model F and a teacher model F' with the same network architecture but different weights θ and θ' , respectively. The weights of the teacher model are updated by taking the exponential moving average of the student model weights:

$$\theta' = m\theta' + (1 - m)\theta \tag{3}$$

where m is known as the momentum which is usually a number close to 1, e.g. 0.99. Figure 5 shows that the teacher model constantly provides effective supervision to the student model via high-quality pseudo targets. Hence, by enforcing the consistency between the student and the teacher, the student learns domain-invariant representations to adapt to the unlabeled target domain, guided by the pseudo labels. We show in Table 5 that the mean teacher significantly improves model performance compared to baseline.

3.4. Point-Level Consistency

The point-level consistency loss is calculated between the first-stage box proposals of the student and teacher models. One of the key challenges for formulating consistency

is to find the correspondence between the student and the teacher. Unlike image pixels that are arranged in regular lattices, points reside in continuous 3D space which lacks structure [27]. Hence, constructing point correspondences can be problematic (Table 3). Instead, we circumvent the difficulty by feeding the teacher and the student two identical sets of points at the very beginning and trace the point indices to maintain correspondences.

Specifically, for each target domain example, we sample M points from the point cloud scene to obtain the teacher input x_t and apply random augmentation h on a replicated set to obtain $\hat{x_t}$ with $\hat{x_t} = h(x_t)$. h consists of random global scaling of the point cloud scenes and can be regarded as applying displacements on individual points, without disrupting the point correspondences. As a result, each point $p \in x_t$ corresponds to a point $\hat{p} \in \hat{x_t}$, and this relationship holds for the point-level predictions of the region proposal network F_{RPN}. We denote the first stage prediction as $R = F_{RPN}(x)$. Note that the point correspondences are transferred to box proposals as each point generates one box proposal. R consists of class prediction R^c and box regression R^b. For the class predictions, we define the consistency loss as the Kullback-Leibler (KL) divergence between each point pair from x_t and $\hat{x_t}$:

$$L_{pt,cls} = \frac{1}{|x_t|} \sum D_{KL}(\hat{R}_t^c || R_t^c) \tag{4}$$

where $|x_t|$ stands for the number of points in x_t.

More importantly, we enforce consistency between bounding box regression predictions to address geometric mismatch. For the bounding box predictions, we only compute the consistency over points belonging to the objects because the background points do not generate meaningful bounding boxes. We obtain a set of points $\mathbb{P}_{pos}$ which fall inside the bounding boxes of the final predictions of both the student and teacher networks with $\mathbb{P}_{pos} = \{(p \in NMS(\hat{S}_t)) \cap (p \in NMS(S_t))\}$, where $\hat{S}_t$ and S_t are the refined bounding box predictions after second stage (see Section 3.5). We then compute the point-level box consistency loss as:

$$L_{pt,box} = \frac{1}{|\mathbb{P}_{pos}|} \sum_{p^i \in \mathbb{P}_{pos}} d(\hat{R}_t^{c(i)}, h(R_t^{c(i)})) \tag{5}$$

where d is the smooth $L1$ loss and h is the random augmentation applied to the input x_t. We apply the same augmentation to the teacher bounding box predictions to align with the scale of the student point cloud scene before computing the consistency.

3.5. Instance-Level Consistency

In the second stage, NMS is performed on R to obtain N high-confidence region proposals denoted as G for each point cloud scene. We highlight that the association between region proposals from the student and teacher models are lost in the NMS process due to the differences between $\hat{R}_t$ and R_t. To match the instance-level predictions for consistency computation, a common method is to perform greedy matching based on IoU between teacher and student region proposals. However, such matching is not robust due to the large number of noisy predictions, which leads to ineffective learning as shown experimentally in Table 3. Hence, we adopt a simple approach by replicating the teacher region proposals to the student model and applying the input augmentation h to match the scale of the student model. Subsequently, we disturb the region proposals by applying random RoI augmentation ξ for the sets of region proposals before they are used for feature pooling. The motivation of this operation is to force the models to output consistent predictions given non-identical region proposals and prevent convergence to trivial solutions. Formally, the above process can be described as $\hat{\boldsymbol{f}}_{\boldsymbol{t}} = pool(\xi(h(G_t)))$ and $\boldsymbol{f_t} = pool(\xi'(G_t))$ for the student and teacher models, respectively, where $\boldsymbol{f}$ denotes the instance-level features obtained from feature pooling as described in [32]. The pooled features are then passed to the box refinement network F_{BRN} for box refinement to obtain the second stage predictions $S = F_{BRN}(\boldsymbol{f})$. Similar to the first stage prediction R, S consists of the class prediction S^c as well as the bounding box prediction S^b. We define the instance-level class consistency as the difference between $\hat{S}_t^c$ and S_t^c:

$$L_{ins,cls} = \frac{1}{|G_t|} \sum D_{KL}(\hat{S}_t^c || S_t^c) \tag{6}$$

where $|G_t|$ denotes the number of region proposals. On the other hand, to compute the instance-level box consistency loss, we first obtain a set of positive predictions $\mathbb{S}_{pos} = \{(\hat{S}_t^c > \varepsilon) \cap (S_t^c > \varepsilon)\}$ by selecting bounding boxes with classification predictions larger than a probability threshold ε. We then apply h to S_t^b to match the scale and compute the instance-level box consistency loss based on the discrepancy between $\hat{S}_t^b$ and S_t^b for the selected predictions:

$$L_{ins,box} = \frac{1}{|\mathbb{S}_{pos}|} \sum_{S_t^i \in \mathbb{S}_{pos}} d(\hat{S}_t^{b(i)}, S_t^{b(i)}) \tag{7}$$

3.6. Neural Statistics-Level Consistency

As pointed out in [23, 5] that the mismatch in batch normalization statistics between teacher and student models could lead to suboptimal model performance, in our case, while the student model takes both source domain data x_s and target domain data $\hat{x_t}$ as input, the teacher model only has access to the target data x_t. The distribution shift lying between source and target data could lead to mismatched batch statistics between the batch normalization (BN) layers of the student and teacher models. This mismatch could

cause misaligned normalization and in turn, leads to an unstable training process with degraded performance or even divergence. We provide an in-depth analysis regarding this matter in Section 4.4.

To mitigate this issue, we propose to use the running statistics of the student model BN layers for the teacher model during the training process. Specifically, for each BN layer in the student model, the batch mean μ and variance σ are used to update the running statistics at every iteration:

$$\mu' = (1-\alpha)\mu' + \alpha\mu \tag{8}$$
$$\sigma' = (1-\alpha)\sigma' + \alpha\sigma \tag{9}$$

where μ' and σ' are the running mean of μ and σ and α is the BN momentum that controls the speed of batch statistics updating the running statistics. For the teacher model, we use μ' and σ' instead of the batch statistics for all the BN layers to normalize the layer inputs. We argue that this modification closes the gap caused by domain mismatch and leads to more stable training behavior. We empirically demonstrate the effectiveness by comparing the performance under different BN settings in Section 4.3.

4. Experiments

We first introduce the popular autonomous driving datasets including KITTI [11], Waymo Open Dataset [35], and nuScenes [3] used in the experiments (Section 4.1). We then benchmark MLC-Net across datasets where MLC-Net achieves consistent performance boost in Section 4.2. Moreover, we ablate MLC-Net to give a comprehensive assessment of its submodules and justify our design choices in Section 4.3. Finally, we further investigate the challenges of unsupervised domain adaptive 3D detection and show MLC-Net successfully addresses them. We further analyse the problems in 3D domain adaptive detection and our solutions in Section 4.4. Due to space constraint, we include the implementation details in the Supplementary Material.

4.1. Datasets

We follow [38] to evaluate MLC-Net on various source-target combinations with the following datasets.

KITTI. KITTI [11] is a popular autonomous driving dataset that consists of 3,712 training samples and 3,769 validation samples. The 3D bounding box annotations are only provided for objects within the Field of View (FoV) of the front camera. Therefore, points outside of the FoV are ignored during training and evaluation. We use the official KITTI evaluation metrics for evaluation where the objects are categorized into three levels (Easy, Moderate, and Hard) and the mean average precision is evaluated.

Waymo Open Dataset. The Waymo Open Dataset (referred to as Waymo) [35] is a large-scale benchmark that contains 122,000 training samples and 30,407 validation samples. We subsample 1/10 of the training and validation set. To align the input convention, we apply the same front camera FoV as the KITTI dataset. The official Waymo evaluation metrics including mean average precision (AP) and mean average precision weighted by heading (APH) are used to benchmark the performance for objects of two difficulty levels (L1 and L2).

nuScenes. The nuScenes [3] dataset consists of 28,130 training samples and 6,019 validation samples. We subsample the training dataset by 50% and use the entire validation set. We also apply the same FoV on the input as other datasets. We adopt the official evaluation metrics of translation, scale, and orientation errors, with the addition of the commonly used average precision based on 3D IoU with a threshold of 0.7 to reflect the overall detection accuracy.

4.2. Benchmarking Results

As an emerging research area, the cross-domain point cloud detection topic has relatively small literature. To the best of our knowledge, [38] is the most relevant work that has a similar setting as our study. We compare our method with two normalization methods proposed in [38], namely Output Transformation (OT) and Statistical Normalization (SN), where the former transforms the predictions by an offset and the latter trains the detector with scale-normalized input. Moreover, we also compare with the adversarial feature alignment method, which is commonly used on image-based tasks, by adapting DA-Faster [8] to our PointRCNN [32] base model. We also provide Direct Transfer and Wide-Range Augmentation as baselines. Figure 1 displays a qualitative comparison of the detection results before and after domain adaptation with our proposed method. More results can be found in the Supplementary Material.

Table 1 demonstrates the cross-domain detection performance on four source-target domain pairs, MLC-Net outperforms all unsupervised baselines by convincing margins. We highlight that our method adapts scale for each instance instead of applying a global shift, allowing us to surpass state-of-the-art methods that utilize target domain object scale statistics.

4.3. Ablation Study

To evaluate the effectiveness of the components of MLC-Net, we conduct ablation studies on KITTI $\rightarrow$ Waymo transfer with PointRCNN as the base model.

Effectiveness of Point/Instance-Level Consistency. We study the effects of different components of the proposed consistency loss. Table 2 reports the experimental results when different combinations of loss components are applied. It is observed that for both point-level consistency and instance-level consistency, the box consistency clearly has a larger contribution as compared to the class consis-

Table 1: Performance of MLC-Net on four source-target pairs in comparison with various baselines and state-of-the-art methods. MLC-Net outperforms all baselines and even surpasses SOTA methods that utilize target domain annotation information (indicated by †). Direct transfer: the model trained on the source domain is directly tested on the target domain. Wide-Range Aug: baseline method with random scaling augmentation of a wide range which potentially includes the target domain scales. It is thus validated the drastic performance degradation cannot be fully mitigated by simple data augmentation. DA-Faster[8]: a representative method based on adversarial feature alignment, a common technique used in 2D domain adaptation. # indicates the implementation is adapted from 2D to 3D. However, feature alignment is unable to solve the geometric mismatch, which we argue is unique to 3D detection. The state-of-the-art work [38] proposes to perform output transformation (OT) to scale predictions and statistical normalization (SN) for scale-adjusted training examples. Both OT and SN require known target domain statistics. MLC-Net, albeit being fully unsupervised, even surpasses these methods on key metrics: APH/L2 (Waymo), AP^{3D} (nuScenes), and AP Moderate (KITTI).

KITTI → Waymo					**Waymo → KITTI**			
Methods	AP/L1	APH/L1	AP/L2	APH/L2	Methods	Easy	Moderate	Hard
Direct Transfer	9.17	8.99	7.94	7.78	Direct Transfer	20.22	21.43	20.49
Wide-Range Aug	18.61	18.18	16.77	16.40	Wide-Range Aug	30.23	31.49	32.85
DA-Faster [8]#	6.96	6.87	6.42	6.33	DA-Faster [8]#	4.42	5.55	5.53
OT [38]†	26.48	25.84	23.85	23.29	OT [38]†	39.78	37.82	39.55
SN [38]†	30.69	30.06	27.23	26.67	SN [38]†	61.93	58.07	**58.44**
Ours	**38.21**	**37.74**	**34.46**	**34.04**	Ours	**69.35**	**59.44**	56.29
KITTI → nuScenes					**nuScenes → KITTI**			
Methods	ATE	ASE	AOE	AP^{3D}	Methods	Easy	Moderate	Hard
Direct Transfer	0.207	0.248	0.212	13.01	Direct Transfer	49.13	39.56	35.51
Wide-Range Aug	0.200	0.228	0.211	16.01	Wide-Range Aug	58.71	45.37	43.03
DA-Faster [8]#	0.247	0.253	0.292	10.77	DA-Faster [8]#	52.25	40.62	35.90
OT [38]†	0.207	0.220	0.212	14.67	OT [38]†	23.13	27.26	29.10
SN [38]†	0.227	**0.168**	0.368	23.15	SN [38]†	44.81	45.15	47.60
Ours	**0.197**	0.179	**0.197**	**23.47**	Ours	**71.26**	**55.42**	**48.99**

Table 2: Ablation study of point-level and instance-level consistency loss components. Results show loss components are highly complementary; the joint use of all four losses at two levels achieves the best performance. More importantly, we find that the bounding box regression loss, which is directly associated with bounding box scale, benefits the performance more than the classification loss. This further validates our stance that geometric mismatch is a key domain gap for 3D detection.

$L_{pt,cls}$	$L_{pt,box}$	$L_{ins,cls}$	$L_{ins,box}$	AP/L1	APH/L1	AP/L2	APH/L2
				18.61	18.18	16.77	16.40
✓				20.34	19.91	18.07	17.70
	✓			30.34	29.69	27.08	26.49
✓	✓			31.00	30.39	27.64	27.09
		✓		21.12	20.87	18.79	18.57
			✓	33.21	32.44	29.95	29.26
		✓	✓	34.95	34.53	31.43	31.05
✓	✓	✓	✓	**38.21**	**37.74**	**34.46**	**34.04**

Table 3: Ablation study of point-level and instance-level matching methods. Nearest Point: a baseline for point match where a point in the student input is matched to the nearest point in the teacher input using Euclidean distance. Max IoU Box: a baseline for box matching where a student box prediction is matched to the teacher pseudo label with the largest IoU. Ours: input point clouds or region proposals of the student are replicated from the teacher. We highlight that our matching method ensures accurate one-to-one correspondence, which is critical to effective teacher-student learning.

Matching Method	AP/L1	APH/L1	AP/L2	APH/L2
Nearest Point	2.93	2.86	2.65	2.58
Max IoU Box	26.95	26.66	24.18	23.92
Ours	**38.21**	**37.74**	**34.46**	**34.04**

tency. This observation indicates that the scale difference is a major source of the domain gap between source and target domains with different object size distributions, which is also in line with the previous work [38]. It also shows that our proposed box consistency regularization method effectively mitigates this gap. In addition, all losses are complementary to one another: the best result is achieved when all four of them are used.

Furthermore, we compare MLC-Net with two alternative approaches for point and box matching respectively in Table 3. Compared to these baseline approaches, MLC-Net replicates the input point clouds and the region proposals before they are passed to the student and teacher models to eradicate any noise which may arise from inaccurate matching. The results highlight the importance of correspondence in constructing meaningful consistency losses for effective unsupervised learning.

Effectiveness of Neural Statistics-Level Consistency. We also experiment on the effectiveness of neural statistics-level consistency by comparing the performance when such alignment is enabled and disabled. From Table 4 we can see that when neural statistics-level consistency is disabled, the model performance severely drops. As analyzed in Section 3.6, when neural statistics-level consistency is not in place, the teacher model BN layers normalize the input fea-

Table 4: Ablation study of neural statistics-level consistency indicates that MLC-Net effectively closes the domain gap due to neural statistics mismatch. Disabled: no consistency is enforced. Separate: the student model performs BN separately for source and target domain inputs to align with the teacher model. Enabled: our proposed neural statistics-level alignment.

Setting	AP/L1	APH/L1	AP/L2	APH/L2
Disabled	2.79	2.74	2.54	2.49
Separate	29.88	29.45	26.85	26.48
Enabled	**38.21**	**37.74**	**34.46**	**34.04**

Table 5: Ablation study of the exponential moving average (EMA) update scheme in mean teacher paradigm. The performance significantly degrades when the exponential moving average update is disabled, highlighting the importance of the mean teacher design in producing meaningful targets.

EMA	AP/L1	APH/L1	AP/L2	APH/L2
Disabled	8.95	8.66	8.35	8.08
Enabled	**38.21**	**37.74**	**34.46**	**34.04**

tures using batch statistics that are obtained from only target data, while the student model performs BN with statistics from both source and target domains. This misalignment creates a significant gap. As a result, the consistency computation between the student and teacher predictions is invalidated. We also compare with the approach that the student model performs separate BN for source and target data. In this case, although the normalization for target input is performed with target statistics for both models, the mismatched normalization of source and target inputs leads to suboptimal performance as compared to MLC-Net.

Effectiveness of Mean Teacher. The teacher model is essentially a temporal ensemble of student models at different time stamps. We study the effectiveness of the mean teacher paradigm by comparing the performance when the exponential moving average update is enabled or disabled. Table 5 shows that it is important to employ the moving average update mechanism for the teacher to generate meaningful supervisions to guide the student model, and the removal of such mechanism leads to performance deterioration.

4.4. Further Analysis

Analysis of Distribution Shift. We highlight that the geometric mismatch is a significant issue for cross-domain deployment of 3D detection models. In Figure 2, the object dimension (length, width, and height) distributions are drastically different across domains with a relatively small overlap. The baseline, trained on the source domain, is not able to generalize to the target domain as the distribution of its dimension prediction is still close to that of the source domain. In contrast, MLC-Net is able to adapt to the new domain by predicting highly similar geometric distribution as the target domain.

Analysis of Neural Statistics Mismatch. Figure 4 shows that inputs from different domains have very different distributions of batch statistics, which explains the tremendous performance drop when our proposed neural statistics-level consistency is not applied to align the statistics (Table 4).

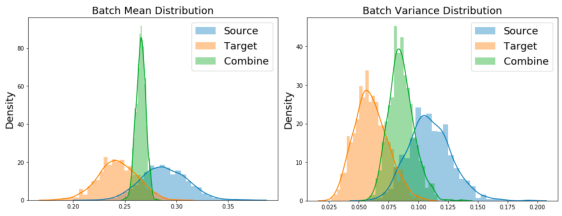

Figure 4: Neural statistics mismatch across domains. We plot the distributions of batch mean and batch variance. Significant misalignment in batch statistics between source and target domains is observed, which highlights the necessity of neural statistics-level consistency.

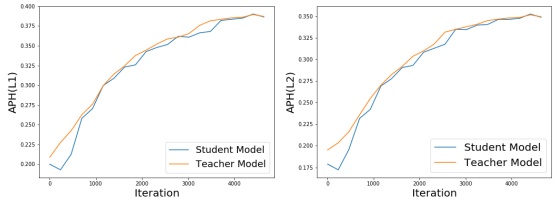

Figure 5: Teacher and student model performance against iteration. Not only does the teacher model constantly outperform the student, its performance curve is also smoother. Hence, the teacher model, which can be regarded as a temporal ensemble of the student model, is able to produce more stable and accurate pseudo labels to supervise the student model.

Analysis of Teacher/Student Paradigm. In Figure 5, the teacher model in MLC-Net demonstrates stronger performance during the training process until convergence. Moreover, the teacher model exhibits a smoother learning curve. This validates the effectiveness of our mean-teacher paradigm to create accurate and reliable supervision for robust optimization of the student model.

5. Conclusion

We study unsupervised 3D domain adaptive detection that requires no target domain annotation or annotation-related statistics. We validate that geometric mismatch is a major contributor to the domain shift and propose MLC-Net that leverages a teacher-student paradigm for robust and reliable pseudo label generation via point-, instance- and neural statistics-level consistency to enforce effective transfer. MLC-Net outperforms all the baselines by convincing margins, and even surpasses methods that require additional target information.

Acknowledgements This study is supported by NTU NAP, and under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s).

References

[1] Idan Achituve, Haggai Maron, and Gal Chechik. Self-supervised learning for domain adaptation on point clouds. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 123–133, 2021. 3

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019. 3

[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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1, 2, 6

[4] Qi Cai, Yingwei Pan, Chong-Wah Ngo, Xinmei Tian, Lingyu Duan, and Ting Yao. Exploring object relation in mean teacher for cross-domain detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11457–11466, 2019. 3

[5] Zhaowei Cai, Avinash Ravichandran, Subhransu Maji, Charless Fowlkes, Zhuowen Tu, and Stefano Soatto. Exponential moving average normalization for self-supervised and semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 194–203, 2021. 5

[6] Chaoqi Chen, Zebiao Zheng, Xinghao Ding, Yue Huang, and Qi Dou. Harmonizing transferability and discriminability for adapting object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8869–8878, 2020. 2

[7] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017. 3

[8] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3339–3348, 2018. 2, 6, 7

[9] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018. 3

[10] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*, 2017. 3

[11] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 1, 2, 6

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

[13] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 3

[14] Dayan Guan, Jiaxing Huang, Shijian Lu, and Aoran Xiao. Scale variance minimization for unsupervised domain adaptation in image segmentation. *Pattern Recognition*, 112:107764, 2021. 2

[15] Dayan Guan, Jiaxing Huang, Aoran Xiao, Shijian Lu, and Yanpeng Cao. Uncertainty-aware unsupervised domain adaptation in object detection. *IEEE Transactions on Multimedia*, 2021. 2

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 3

[17] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Fsdr: Frequency space domain randomization for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6891–6902, 2021. 2

[18] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Rda: Robust domain adaptation via fourier adversarial attacking. *arXiv preprint arXiv:2106.02874*, 2021. 2

[19] Jiaxing Huang, Shijian Lu, Dayan Guan, and Xiaobing Zhang. Contextual-relation consistent domain adaptation for semantic segmentation. In *European conference on computer vision*, pages 705–722. Springer, 2020. 2

[20] Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Emilie Wirbel, and Patrick Pérez. xmuda: Cross-modal unsupervised domain adaptation for 3d semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12605–12614, 2020. 3

[21] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019. 1, 3

[22] Congcong Li, Dawei Du, Libo Zhang, Longyin Wen, Tiejian Luo, Yanjun Wu, and Pengfei Zhu. Spatial attention pyramid network for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 481–497. Springer, 2020. 2

[23] Zeming Li, Songtao Liu, and Jian Sun. Momentum2 teacher: Momentum teacher with momentum statistics for self-supervised learning. *arXiv preprint arXiv:2101.07525*, 2021. 5

[24] Songtao Liu, Zeming Li, and Jian Sun. Self-emd: Self-supervised object detection without imagenet. *arXiv preprint arXiv:2011.13677*, 2020. 3

[25] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 1

[26] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018. 3

[27] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 3, 5

[28] 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. *arXiv preprint arXiv:1911.02744*, 2019. 3

[29] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6956–6965, 2019. 2

[30] Cristiano Saltori, Stéphane Lathuiliére, Nicu Sebe, Elisa Ricci, and Fabio Galasso. Sf-uda 3d: Source-free unsupervised domain adaptation for lidar-based 3d object detection. *arXiv preprint arXiv:2010.08243*, 2020. 3

[31] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020. 1, 3

[32] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019. 1, 2, 3, 5, 6

[33] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1

[34] Vishwanath A Sindagi, Yin Zhou, and Oncel Tuzel. Mvx-net: Multimodal voxelnet for 3d object detection. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7276–7282. IEEE, 2019. 1

[35] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020. 1, 2, 6

[36] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv preprint arXiv:1703.01780*, 2017. 2, 3, 4

[37] Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, and Ming-Hsuan Yang. Cross-domain few-shot classification via learned feature-wise transformation. *arXiv preprint arXiv:2001.08735*, 2020. 2

[38] Yan Wang, Xiangyu Chen, Yurong You, Li Erran Li, Bharath Hariharan, Mark Campbell, Kilian Q Weinberger, and Wei-Lun Chao. Train in germany, test in the usa: Making 3d object detectors generalize. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11713–11723, 2020. 1, 3, 6, 7

[39] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*, 2019. 3

[40] Chang-Dong Xu, Xing-Ran Zhao, Xin Jin, and Xiu-Shen Wei. Exploring categorical regularization for domain adaptive object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11724–11733, 2020. 2

[41] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 1, 3

[42] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11040–11048, 2020. 1, 2, 3

[43] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1951–1960, 2019. 1

[44] Li Yi, Boqing Gong, and Thomas Funkhouser. Complete & label: A domain adaptation approach to semantic segmentation of lidar point clouds. *arXiv preprint arXiv:2007.08488*, 2020. 3

[45] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. *arXiv preprint arXiv:2006.11275*, 2020. 1

[46] Fangneng Zhan, Chuhui Xue, and Shijian Lu. Ga-dan: Geometry-aware domain adaptation network for scene text detection and recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9105–9115, 2019. 2

[47] An Zhao, Mingyu Ding, Zhiwu Lu, Tao Xiang, Yulei Niu, Jiechao Guan, and Ji-Rong Wen. Domain-adaptive few-shot learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1390–1399, 2021. 2

[48] Yangtao Zheng, Di Huang, Songtao Liu, and Yunhong Wang. Cross-domain object detection through coarse-to-fine feature adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13766–13775, 2020. 2

[49] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018. 3

[50] Xinge Zhu, Yuexin Ma, Tai Wang, Yan Xu, Jianping Shi, and Dahua Lin. Ssn: Shape signature networks for multi-class object detection from point clouds. In *Proceedings of the European Conference on Computer Vision*, 2020. 1