CMD: A Cross Mechanism Domain Adaptation Dataset for 3D Object Detection

Jinhao Deng[1,2,*], Wei Ye[1,2,*], Hai Wu[1,2], Xun Huang[1,2], Qiming Xia[1,2], Xin Li[4], Jin Fang[3], Wei Li[3,✉], Chenglu Wen[1,2,✉], and Cheng Wang[1,2]

[1] Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, Xiamen, P.R China
[2] Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, Xiamen, P.R. China.
[3] Inceptio, Shanghai, P.R China
[4] Section of Visual Computing and Interactive Media, Texas A&M University, Texas, USA

Abstract. Point cloud data, representing the precise 3D layout of the scene, quickly drives the research of 3D object detection. However, the challenge arises due to the rapid iteration of 3D sensors, which leads to significantly different distributions in point clouds. This, in turn, results in subpar performance of 3D cross-sensor object detection. This paper introduces a **C**ross **M**echanism **D**ataset, named **CMD**, to support research tackling this challenge. CMD is **the first** domain adaptation dataset, comprehensively encompassing **diverse mechanical sensors** and various scenes for 3D object detection. In terms of sensors, CMD includes 32-beam LiDAR, 128-beam LiDAR, solid-state LiDAR, 4D millimeter-wave radar, and cameras, all of which are well-synchronized and calibrated. Regarding the scenes, CMD consists of 50 sequences collocated from different scenarios, ranging from campuses to highways. Furthermore, we validated the effectiveness of various domain adaptation methods in mitigating sensor-based domain differences. We also proposed a **DIG** method to reduce domain disparities from the perspectives of **D**ensity, **I**ntensity, and **G**eometry, which effectively bridges the domain gap between different sensors. The experimental results on the CMD dataset show that our proposed DIG method outperforms the state-of-the-art techniques, demonstrating the effectiveness of our baseline method. The dataset and the corresponding code are available at https://github.com/im-djh/CMD.

Keywords: Dataset · 3D Object Detection · Domain Adaptation

1 Introduction

As a crucial component in robotics and autonomous driving systems, 3D object detection has garnered increasing attention from researchers. Due to the inherent

* Equal contribution
✉ Corresponding authors: W. Li (liweimcc@gmail.com); C. Wen (clwen@xmu.edu.cn)

depth of information, point cloud data enjoys a unique advantage in the domain of 3D object detection. Notably, researchers have developed several exceptional 3D object detection datasets and benchmarks (e.g., KITTI [13], nuScenes [5], Waymo [30], ONCE [21], etc.). Leveraging these high-quality datasets, numerous outstanding 3D object detection approaches [3, 9, 18, 29, 39, 41, 49] emerged, significantly facilitating the 3D object detection research. However, these prevalent datasets typically comprise mechanical spinning LiDAR only w.r.t. 3D point cloud sensors. The widely equipped low-cost automotive-grade sensors, *e.g.* solid-state LiDAR, in mass-produced vehicles [27] [25] are heavily overlooked even missing in those datasets.

The truth of outdatedness of existing datasets in terms of **sensors** behind is that point cloud sensors are undergoing really rapid advancements. Commonly used sensors now include: (1) mechanical spinning LiDAR with low-beam (e.g., 32 beams) or high-beam (e.g., 128 beams); (2) solid-state LiDAR; and (3) the recently acclaimed 4D millimeter-wave radar. These sensors differ either in the number of beams they use or in their world modeling patterns. We refer to these sensor differences as various **mechanisms**. The data acquired by different mechanisms may exhibit *domain disparities* in terms of *density*, *intensity*, and *geometry* (see Fig. 1(a), (b) and (c)). As a result, 3D detectors trained with data from one sensor often incur substantial accuracy degradation when directly applied to another sensor (see "Direct" in Fig. 1(d), (e) and (f)). For example, when migrating from mechanical spinning LiDAR to a more inexpensive and compact solid-state LiDAR, the cost of annotating new sensor data to re-train detectors is prohibitive. Cross-mechanism domain adaptation is a more efficient yet promising solution with rigid demand to transfer not only detectors/models but also all related assets to new sensors.

Meanwhile, prevalent domain adaptation research [19, 26, 35, 36, 45, 46] finds it hard to quantitatively analyze the pivotal sensor factor. This challenge arises because they have been directed at cross-dataset settings (e.g. domain adaptation between KITTI, Waymo, and nuScenes), covering differences in sensors, geographic location, climate, and so on. Therefore, they have not shown optimal performance when applied to cross-mechanism domain adaptation problems (see Fig. 1(d)(e)(f)). The reason lies in the fact that there is a lack of datasets that contain comprehensive sensors to fully decouple the domain disparities caused by locations and sensors. Several inspiring datasets, encompassing various common sensors (e.g., PandaSet [44], KRadar [22], VoD [23], LiDAR-CS [11]), often fall short in providing comprehensive modality coverage. This limitation, in turn, restricts the design of methods to bridge the sensor-based domain gap.

To address this problem, this paper introduces a **C**ross **M**echanism domain adaptation **D**ataset (**CMD**) for 3D object detection. To the best of our knowledge, CMD is **the first** domain-adaptation dataset in 3D object detection that contains a comprehensive suite of sensors, including: (1) *32-beam low-resolution and 128-beam high-resolution mechanical spinning (MS) LiDAR*, (2) *automotive-grade solid-state LiDAR*, (3) *4D millimeter-wave radar*, and (4) *camera*. This combination provides the most extensive modality coverage. All sensors are time-

synchronized with high accuracy under 1 ms, ensuring that different modalities can capture and model the same scene accurately, even with highly dynamic objects. Furthermore, the CMD comprises 50 sequences. Each sequence has a time span of 20 seconds, with each sensor capturing 10 frames per second, totaling 10,000 frames of data per sensor. In addition, we meticulously annotated 3D objects of 13 categories based on a multi-sensor collaborative annotation system subjected to multiple rounds of human inspection.

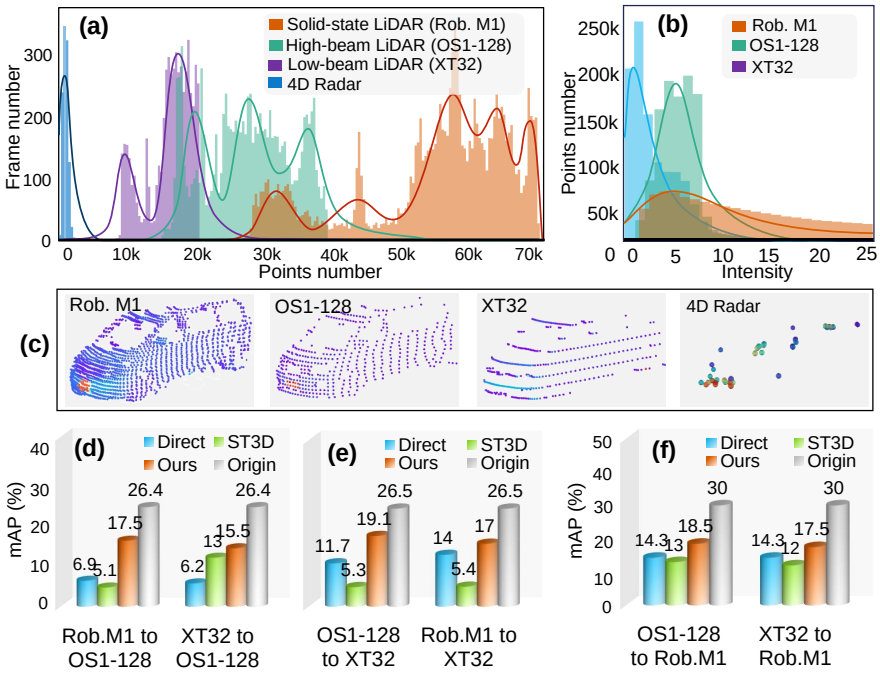

Fig. 1: (a) The points number distribution of different sensors. (b) The intensity distribution of different sensors. (c) Point clouds of the same instance for different sensors. The results show that data acquired by different sensors exhibit substantial differences in point density, intensity, and geometry. Subfigure (d), (e), and (f) show the cross-mechanism detection results of different domain adaptation methods on our CMD. Our DIG outperforms the traditional ST3D by a large margin.

Using the CMD, we investigate the cross-mechanism domain adaptation issues and summarize the gaps into three primary components: **D**ensity gap, **I**ntensity gap, and **G**eometry gap. Subsequently, we introduce a simple yet effective baseline method, named **DIG**, to address each of these gaps systematically. (1) *Density gap*: We propose the Beam-Distance Down-sampling (BDS) that enforces a similar data pattern in terms of beam numbers and density distribution along different detection distances. (2) *Intensity gap*: We introduce the Box-Cox log Normalization (BCN) that initially transforms point clouds from

different modalities into near-normal distributions and then normalizes them by logarithms. (3) *Geometry gap*: We design the Geometry-Aware label Mixing (GAM) that mixes the pseudo-labels from the target domain to narrow down the geometry difference in training. By implementing these three components, DIG achieves optimal results in bridging the identified domain gaps. Our main contributions are as follows.

- We present the Cross Mechanism Dataset (CMD), the first domain adaptation dataset that contains a comprehensive suite of LiDAR/4D Radar sensors. CMD is a key piece of the puzzle of cross-mechanism detection as existing public datasets can not adequately evaluate detection performance drops caused by cross-domain disparities due to limited sensor variety.
- We construct comprehensive 3D cross-sensor detection baselines by benchmarking the results of SOTA domain adaptation algorithms on CMD.
- We provide an in-depth analysis of the key factors leading to cross-sensor domain disparities, *i.e.* Density, Intensity, and Geometry gaps. Based on this, we propose a novel DIG method, which shows plausible performance on cross-sensor detection.

2 Related Work

2.1 Datasets for 3D Object Detection

With increasing research on 3D object detection [7, 8, 17, 40, 42, 43, 47], more datasets have emerged. Widely used datasets such as KITTI [13], Waymo [30], nuScenes [5], and ONCE [21] support conventional 3D object detection research and facilitate domain adaptation across different geographic locations. However, focusing solely on geographical domain differences is inadequate for addressing the domain adaptation problem in detection algorithms; sensor variability is also crucial. To address this, datasets like PandaSet [44], K-Radar [22], TJ4DRadSet [52], VoD [23], Lyft L5 [15], and Cirrus [37] offer diverse sensors, scan beams, locations, and weather conditions. Despite their contributions, these datasets are limited in sensor diversity within individual collections. In contrast, our CMD includes extensive image and point cloud data annotated for detection and tracking, encompassing five different sensor types, thus providing a comprehensive evaluation. Statistical comparisons with other 3D object detection datasets are shown in Table 2.

2.2 Domain Adaption in 3D Object Detection

To mitigate the poor generalization of 3D object detectors to unknown data, recent methods have focused on adapting domain gaps [10, 14, 32–34, 48, 50, 51]. Among these methods, some attempt to introduce a teacher-student framework. ST3D [45] and ST3D++ [46] design label assignment strategy and pseudo-labels denoising to enable adaptation between domains. MLC-Net [19] employs consistency for cross-domain transfer. Another statistics-based approach adapts distribution gaps, with SN [35] correcting car size distribution gaps. GBA [26]

adapts label distribution gap via a Gradual Batch Alternation training strategy. And, SSDA3D [36] adapting point cloud distributions through Inter-domain Alignment module. Further methods address sensor gaps, with LiDAR Distillation [38] and DTS [16] proposing progressive and density-insensitive frameworks to mitigate beam-induce gaps from different sensors, and CL3D [24] using spatial geometry alignment to adapt geometric gaps between sensors.

In general, current domain adaptation research methods still primarily focus on geographical information and sensor beam migration. Due to dataset limitations, domain adaptation methods for multiple types of sensors have yet to be seen. Therefore, it is necessary to propose a novel, well-annotated dataset including various types of sensors to advance this research field.

3 Cross Mechanism Dataset

3.1 Sensor Setup

Sensor specifications. Our CMD comprises data from seven sensors of five types: a 128-beams mechanical scanning (MS) LiDAR, a 32-beams MS LiDAR, a solid-state LiDAR, a 4D millimeter-wave radar, and three cameras. Detailed specifications for our sensors are shown in Table 1. The sensors in our dataset, each with unique beam configurations or modeling modes, are collectively referred to as ***mechanisms***. As shown in Table 2, in contrast to other datasets, our CMD encompasses a diverse array of commonly employed sensors, establishing it as the most comprehensive dataset presently available.

Sensor	Type	HFOV(°)	VFOV(°)	Resolution	FPS
OS128	128 beams MS LiDAR	360	$[-22.5, 22.5]$	128 * 1024	10
XT32	32 beams MS LiDAR	360	$[-16, 15]$	32 * 2048	10
M1	SS LiDAR	$[-60, 60]$	$[-12.5, 12.5]$	126 * 625	10
Radar	4D Radar	$[-75, 75]$	$[-15, 15]$	>600	10
HIK Cam.	Camera	$[-31.2, 31.2]$	$[-27.7, 27.7]$	1080 * 1920	10

Table 1: Sensor specifications for CMD. MS and SS denote mechanical spinning and solid-state respectively. FPS refers to frames per second.

Sensor layout. In our CMD, all sensors are oriented in the same direction as the vehicle's forward direction. Such a configuration is instrumental in ensuring that the most critical information is captured within the area of greatest overlap of FOV across different sensors. To achieve this alignment, a rigid bracket has been designed, aligning all sensors along a vertically aligned axis. We have deliberately placed all sensors as close to each other as possible to optimize FOV overlap. In addition to this, a camera system comprising three evenly spaced cameras has been integrated. This system spans a 150° horizontal field of view (HFOV), as depicted in Fig. 2(c). Such a configuration is pivotal

for ensuring data consistency across different mechanisms. The overall FOV is shown in Fig. 2(b).

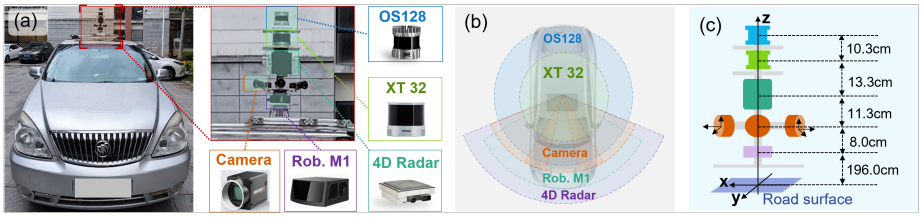

Fig. 2: The illustration of sensor layout. (a) demonstrates the specific layout of each sensor; (b) showcases the field of view angles for each sensor; (c) demonstrates the positions of each sensor along the z-axis.

3.2 Synchronization and Calibration

Synchronization and trigger. To achieve synchronization among different sensors when modeling the world, we utilized the Precision Time Protocol Version 2 [1] to establish a time synchronization system with an error margin within 1 ms. Specifically, we employed the CoolShark AUTO 66 unit as the PTP grandmaster clock, while other sensors and the host served as PTP slave clocks. Once synchronized, M1 is set to generate a frame every 100ms, and both OS128 and XT32 generate frames simultaneously using phase-locking. Meanwhile, the radar is triggered by a network signal sent from the host. Due to the differing principles of data acquisition between cameras and LiDAR, the cameras are triggered by OS128. Whenever OS128 scans to 0°, it sends out a trigger signal. This signal undergoes a delay process via a microcontroller before being fed into the cameras. For each camera, $T_{delay} = \frac{\theta_{cam}}{2\pi} - \frac{1}{2}T_{exposure}$ where $T_{exposure}$ means the exposure time for camera and θ_{cam} means the angle at which the camera is positioned. By this, the time when OS128 scans the central of the camera's HFOV, is the central of its exposure period, which ensures a lower frame time offset between the cameras and the LiDARs. This low-latency synchronization ensures that the same frame of data captured by different sensors effectively reflects their respective mechanism differences.

Calibration. Achieving high-quality data in a multi-sensor setup relies heavily on calibration. (1) For the LiDARs (i.e. OS128, M1, XT32) and radar, we calibrate their extrinsic parameters using Generalized-ICP [28]. Several corner reflectors in the space are used to obtain more precise corner points for radar calibration. (2) For the cameras, we calibrate their intrinsic parameters and extrinsic parameters with MATLAB Toolkit [12] and OpenCV [4]. The results of calibration are shown in Fig. 4. Subsequent to calibration, we established a well-defined coordinate system, as illustrated in Fig. 2(c). We transfer XT32 and

radar to the coordinate system of OS128 and move this coordinate system vertically down to the ground to get the ego system. The ego system has the x-axis pointing forward, the y-axis to the left, and the z-axis upward. The camera coordinate system, in 2D, aligns the x-axis with image width and y-axis with height, starting from the top-left corner.

3.3 Annotation

Annotation area. With precise time synchronization and calibration, we can jointly annotate data from multiple sensors, improving accuracy, especially for distant objects. We annotate objects within $160m$ and within the HFOV of M1. We believe that the data within this range is highly representative.

Annotation rules. There are 13 annotated categories, namely: Car, Van, Bus, Truck, Semi-Trailer towing vehicle, Special Vehicles, Motorcycle, Bicycle, Tricycle, Adult Pedestrian, Children Pedestrian, Animal, and Barrier. We labeled each object as a 9-DoF 3D bounding box $(x, y, z, l, w, h, \theta_x, \theta_y, \theta_z)$. Where x, y, z represents the center coordinates, l, w, h denotes length, width height, and $\theta_x, \theta_y, \theta_z$ are the rotation angles around the x,y,z axis. Additionally, we provide a motion state and tracking ID for all objects. Occlusion situations are also provided. More annotation specifications are provided in the appendix.

Dataset	Mechanism	MS LiDAR(beams) Low	MS LiDAR(beams) High	SS LiDAR	4D Radar	Cam.	Anno. Frams	Tracking	Illumination	Real
KITTI Det. [13]	Single	✗	64	✗	✗	✓	15K	✗	✓	✓
Waymo [30]		✗	64	✗	✗	✓	230K	✓	✓	✓
nuScenes [5]		32	✗	✗	✗	✓	40K	✓	✓	✓
Argoverse [6]		32*2	✗	✗	✗	✓	44K	✓	✓	✓
Lyft L5 [15]		40*2	64	✗	✗	✓	30K	✓	✗	✓
K-Radar [22]	Multi	✗	64, 128	✗	✓	✓	35K	✓	✓	✓
Lidar-CS [11]		16, 32	64, 128	✓	✗	✗	14K	✗	✗	✗
PandaSet [44]		✗	64	✓	✗	✓	8.2K	✓	✓	✓
TJ4DRadSet [52]		✗	64	✗	✓	✓	7.8K	✓	✓	✓
VoD [23]		✗	64	✗	✓	✓	8.7K	✗	✗	✓
CMD		32	128	✓	✓	✓	10K	✓	✓	✓

Table 2: Comparison with existing datasets. "Mechanism" means whether the dataset contains multiple mechanisms of 3D sensor. "Tracking" signifies the relevance of annotations for tracking. "Illumination" indicate the inclusion of various light conditions. "Real" refers to whether the dataset is real or synthetic.

3.4 Dataset Analytic

Diversity of scenes. In CMD, scenes are carefully selected for wide coverage, primarily encompassing urban areas, suburbs, campuses, bridges, and tunnels with different illuminations (see Fig. 3(a)). We select 50 high-quality sequences, each spanning 20 seconds, equating to 200 frames per sensor, culminating in a dataset of 40,000 frames of point cloud data and 30,000 frames of image

data. This comprehensive collection offers a rich resource for research on cross-mechanism domain adaptation. These 50 sequences are evenly divided into 30, 10 and 10 for training, validation, and testing according to different scenarios. This arrangement ensures a balanced distribution of data across environments, crucial for training reliable and generalizable 3D object detection models.

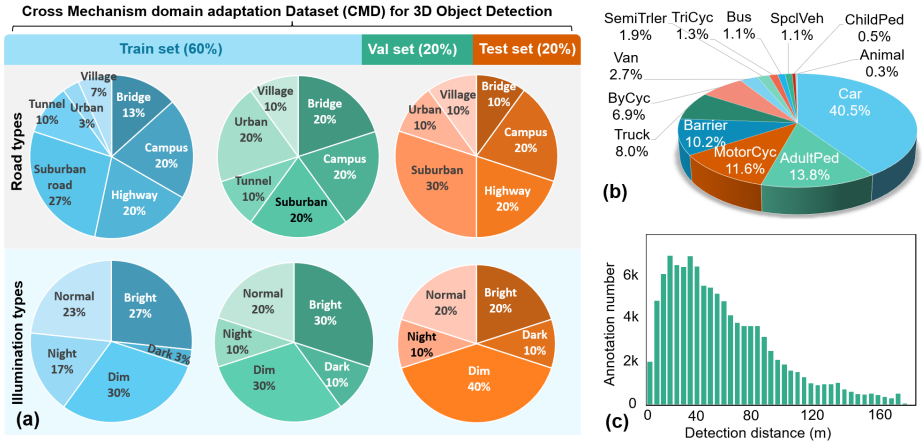

Fig. 3: The scene and annotation details. (a) The upper section showcases the distribution of road types, while the lower section demonstrates the distribution across various illumination types; (b) presents the distribution of objects' numbers for different categories; (c) presents the distribution of objects' numbers along different distances.

Distribution of annotations. We annotated approximately 230,000 3D bounding boxes based on the rules described in Section 3.3. The distribution of objects' numbers for different categories is presented in Fig. 3(b). The distribution of objects' numbers along different distances is shown in Fig. 3(c). Among these categories, the most concerned "Car" has the highest annotation count, reaching approximately 92,000 instances.

Comparisons between mechanisms. The average number of points for these four sensors is quite different. From most to least, they are M1, OS128, XT32, and 4D radar. For a better understanding, we show the visualization results in Fig. 4. Point clouds from sensors with different mechanisms exhibit substantial differences. (1) OS128 and XT32 both adopt a mechanism spinning style, clearly modeling the rigid vehicles with minimal vertical distortion. The main difference between them is density. (2) Due to the use of a zigzag scan pattern, M1 tends to introduce a bit of horizontal displacement. (3) Point clouds from radar sensors are generally sparse and inaccurate, posing challenges in accurately distinguishing the geometric shapes of objects.

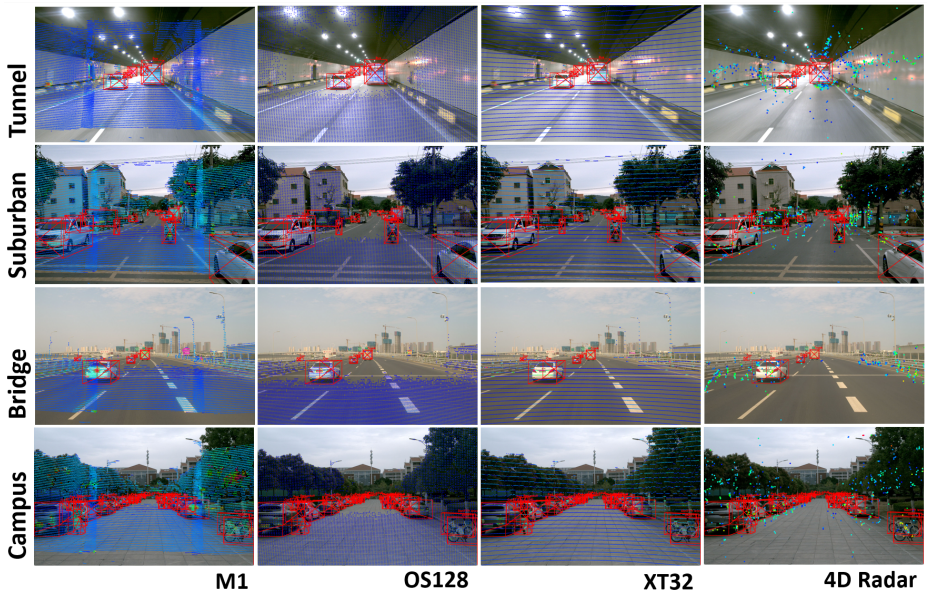

Fig. 4: Different mechanisms and their sample data. The left side of the image denotes the various scenes of data collection, while the bottom indicates the types of sensors involved.

3.5 DIG Baseline Method

In this section, we propose DIG (Density-Intensity-Geometry), a streamlined and potent baseline for cross-mechanism domain adaptation. It consists of three key components: Box-Cox Intensity Log Normalization (BCN), Beam-Distance Down Sampling (BDS), and Geometry-Aware Label Mixing (GAM).

Box-Cox intensity log Normalization (BCN). The distribution of intensity for point clouds may significantly differ across various mechanisms. See Fig. 1(b). Recent studies have either opted to discard [45] [38] intensity or straightforwardly normalized it to [0,1] [36]. However, it is observed that intensity plays a crucial role in detecting foreground objects across diverse domains. We contend that effectively incorporating intensity can lead to enhanced performance in domain adaptation. Motivated by this insight, we introduce the BCN to mitigate disparities in intensity. For both intensities (denoted as i) for source and target domain point clouds, we initially take the logarithm to address biases stemming from different sensor definitions of intensity ranges. Subsequently, we apply the Box-Cox transformation to enhance their normality and homoscedasticity, respectively. This process can be formulated as:

$$i(\lambda) = \frac{log(i+1)^{\lambda} - 1}{\lambda}, \tag{1}$$

where λ refers to the box-cox transfer parameter [2]. It's different for source and target domains, and remains constant respectively. After being normalized to

$[0, 1]$, the transformed intensity information exhibits a good level of similarity between the source and target domains, thus narrowing the domain gap.

Beam-Distance based down Sampling (BDS). As shown in Fig. 1(a), the density distributions of points vary remarkably among different mechanisms. We observe that, within a relatively short distance, domain adaptation for different mechanisms is more significantly impacted by variations in point density. However, when it comes to longer distances, the point cloud is excessively sparse, which has a greater impact on object detection accuracy. To deal with this problem, we introduce the BDS that assigns different sampling probabilities based on both beam and distances. Denote $P_{keep}(p)$ as the probability of a point (p) being retained, and it can be formulated as:

$$P_{keep}(p) = 1 - \mathbf{1}\{beam(p) \notin \mathcal{B}\} exp(-\frac{1}{K}||p||), \tag{2}$$

where K controls the rate of probability increase, and we set it to 35 empirically. $\mathcal{B}$ means the set of target beam indexes that are closest to the source domain. Function $beam(p)$ demotes the index of the beam that point p belongs to. If this beam information is not directly available, an alternative approach involves calculating the angle of each point, denoted as $\theta(p) = arctan(\frac{z}{\sqrt{x^2+y^2}})$. This angle can then be used to apply a clustering algorithm, such as K-means [20], to approximate the beam index for each point.

Geometry-Aware label Mixing (GAM). Point clouds captured from the same object by diverse sensors frequently exhibit notable geometric variances. To tackle this challenge, we introduce the Geometry-Aware Mix (GAM) module. In our approach, we initially utilize a model trained on the source domain to generate pseudo-labels within the target domain. Subsequently, instances from the target domain with high-confidence pseudo-labels are integrated back into the source domain for further training. Inspired by GBA [26], our second training phase progressively reduces source domain labels while maintaining a constant number of target domain pseudo-labels. This technique aims to gradually adapt the model to the geometric characteristics of the target domain. Through this method, the model becomes more adept at learning and adapting to the unique data characteristics of the target domain, thereby improving its ability for cross-mechanism domain adaptation.

4 Experimental Result

4.1 Experiment Setup

All of our experiments were conducted using four Nvidia 3090 GPUs and the open-source code repository OpenPCDet [31]. The Voxel-RCNN [9] served as the detector for most experiments. The batch size was set to 32, and the learning rate was 0.003. Similar to KITTI [13], the grid range was defined as [0m,70.4m], [-70.4m, 70.4m], and [-2m, 4m] along the x, y, and z axis respectively. Point cloud data were cropped to HFOV $[-60°, 60°]$ with horizontal distances smaller than 70.4m. Voxel size was set to [0.1,0.1,0.15] along the x, y, and z axis respectively.

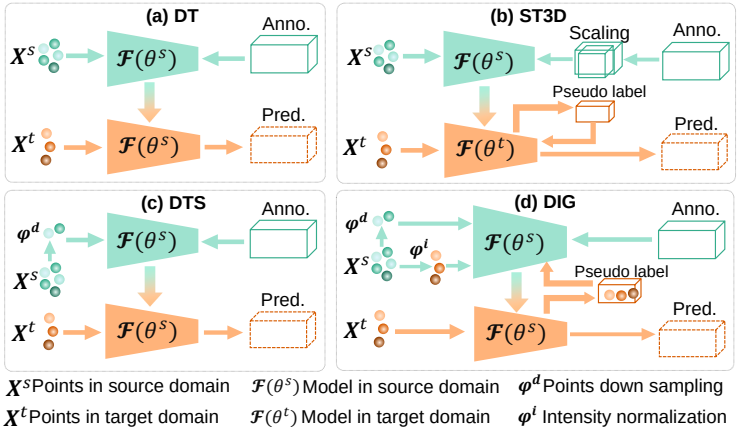

Fig. 5: Similarities and differences among baselines. ST3D [45] leverage random object scaling, generating and denoising pseudo-labels. DTS [16] down sample point clouds before training source-detector. Our DIG jointly downs ample point clouds, transforms intensity, and makes use of pseudo-labels.

4.2 Evaluation Metric

Mean Average Precision (mAP). mAP is a commonly used object detection evaluation metric. Here when calculating mAP, we use 3D IoU thresholds of 0.5, 0.5, 0.25, and 0.25 for Car, Truck, Pedestrian, and Cyclist, respectively. Following the ONCE [21] dataset, we calculated an orientation-aware AP on 50 precision steps. The AP among different detection distances (e.g., overall and 0 to 30m) are also computed for detailed analysis. Formally, the AP is defined as:

$$AP = 100 \int_0^1 max\{p(r'|r' \geq r)\}\, dr, \tag{3}$$

where $p(r)$ refers to the precision-recall curve that calculated by 50 recall positions. The mAP is the average of AP across all categories.

Mean Closed Gap (mCG). Directly applying AP or mAP as evaluation metrics for domain adaptation is not intuitive since different sensors do not share the same domain gap. Closed gap [35] is used for measuring the effectiveness of a method on a single domain adaptation task. CG can be formulated as:

$$CG = 100 \frac{mAP_{model} - mAP_{DT}}{mAP_{Oracle} - mAP_{DT}}, \tag{4}$$

where DT means direct transfer using a model trained sorely on source domain data. Oracle implies results obtained by the full training on target domain. To quantify the universality of domain adaptation methods, we define a new metric named mean closed gap (mCG), formulated as:

$$mCG = \sum_{s \in \mathbf{M}} \sum_{t \in \mathbf{M}} \mathbf{1}(s \neq t) CG_{s \to t}, \tag{5}$$

where **M** refers to the set of sensors, including OS128, XT32, and M1 in CMD.

Method	mCG	CG					
		M1⇒OS128	XT32⇒OS128	OS128⇒M1	XT32⇒M1	OS128⇒XT32	M1⇒XT32
ST3D [45]	-16.72	-9.27	34.00	-7.95	-5.07	-43.39	-68.61
ST3D++ [46]	09.23	12.65	54.49	02.99	17.23	07.56	-39.54
DTS [16]	29.22	48.28	22.92	09.02	07.09	34.50	**53.51**
DIG(Ours)	**42.89**	**54.41**	**55.09**	**33.69**	**25.22**	**49.87**	39.06

Table 3: Overall mean closed gap for baseline methods.

Discussion. Currently, commonly used CG is tailored for a single task, usually only considering the "Car" category [16], and cannot reflect the comprehensive performance of methods across various domain adaptation tasks. In contrast, mCG includes all MS LiDARs with varying beam numbers and SS LiDARs (i.e., OS128, XT32, M1), measuring performance from six experimental groups, thus providing the most comprehensive evaluation of cross-mechanism domain adaptation. See the appendix for 4D radar results.

4.3 Results and Benchmark

Domain adaptation baselines. We selected four baseline methods. Fig. 5 illustrates the similarities and differences between them. (1) Direct transfer (DT) means directly applying the source model to the target dataset. (2) ST3D [45] and ST3D++ [46] share a similar teacher-student architecture. The source model is trained with random object scaling, then used to generate pseudo-labels. The difference is that ST3D++ comes with a more effective denoising method to obtain more high-quality pseudo-labels. (3) DTS [16] introduced Random Beam Re-Sampling (RBRS) to enhance the robustness to varying beam densities. We reproduced the RBRS on CMD, providing representative baseline results. (4) Our proposed DIG.

Overall comparison results. Table 3 shows the overall results. Since the ST3D and ST3D++ are mostly designed for cross-geographical domain adaptation that highly relies on pseudo-label quality, the two detectors show lower performance in our cross-mechanism dataset. DTS takes density into consideration, thus outperforming ST3D++ by 19.99. Our DIG outperformed all previous methods by a large margin. Its mCG reached 42.89, surpassing ST3D, ST3D++, and DTS by 59.61, 33.66, and 13.67 respectively, as our method considers all the typical domain disparities in the cross-mechanism domain adaptation problem.

Performance discussion. A more detailed composition of mCG is shown in Table 4. We can get several conclusions from these results. (1) DIG exhibits notable superiority in the categories of pedestrians and bicycles, both across all distances and within a detection range of 30 meters. The similarity in these categories is the fewer points within the targets. We believe that sparse points make intensity the key factor for adapting to different domains. DIG's BCN module

Task	Method	CG	mAP	Car	Truck	Ped	Cyc
OS128	Oracle	100	26.44	36.50/69.34	17.45/38.51	18.79/38.99	33.0/69.22
XT32 ⇓ OS128	DT	0	06.29	11.76/20.59	04.61/08.23	01.54/02.15	07.25/08.90
	ST3D [45]	34.00	13.14	22.78/47.39	10.41/15.10	06.83/16.01	12.55/33.05
	ST3D++ [46]	54.49	17.27	**27.87**/**61.02**	**13.50**/22.61	08.57/19.76	18.95/49.32
	DTS [16]	22.92	10.91	14.26/49.43	06.72/**27.67**	09.29/20.94	13.37/41.21
	DIG(Ours)	**55.09**	**17.39**	22.73/60.59	13.60/30.16	**11.56**/**25.67**	**21.70**/**57.46**
M1 ⇓ OS128	DT	0	06.92	13.23/30.21	05.08/07.86	01.98/03.61	07.38/09.98
	ST3D [45]	-9.27	05.11	10.24/39.11	04.06/11.32	00.00/00.00	06.14/21.18
	ST3D++ [46]	12.56	09.39	13.46/46.93	06.61/23.16	05.85/13.68	11.65/35.06
	DTS [16]	48.25	16.34	21.60/**58.99**	**12.67**/**34.14**	09.57/23.49	21.51/58.70
	DIG(Ours)	**54.41**	**17.54**	**23.03**/57.08	09.71/25.95	**09.71**/**25.95**	**25.00**/**66.13**
M1	Oracle	100	30.09	42.01/70.64	19.96/40.01	17.54/36.99	40.84/68.49
XT32 ⇓ M1	DT	0	13.32	20.33/42.82	**09.47**/19.84	04.81/11.47	18.67/36.06
	ST3D [45]	-5.07	12.47	30.34/55.41	04.20/03.82	00.19/00.22	15.13/25.95
	ST3D++ [46]	17.23	16.21	**32.92**/61.07	05.24/06.04	02.18/04.17	24.48/49.39
	DTS [16]	07.09	14.51	23.26/56.94	08.78/**28.02**	**06.99**/**14.72**	19.01/48.44
	DIG(Ours)	**25.22**	**17.55**	27.19/**64.70**	09.26/22.79	05.72/12.57	**28.04**/**57.34**
OS128 ⇓ M1	DT	0	14.36	27.07/56.54	09.91/24.25	01.13/03.81	19.32/33.49
	ST3D [45]	-7.95	13.11	28.71/60.37	08.26/07.25	06.37/17.48	09.08/12.72
	ST3D++ [46]	02.99	14.83	**32.98**/**68.45**	06.63/08.45	07.28/19.46	12.43/17.89
	DTS [16]	09.02	15.78	22.11/54.01	11.84/**30.26**	11.88/28.58	17.27/40.40
	DIG(Ours)	**33.69**	**19.66**	26.85/62.89	**11.86**/28.11	**13.69**/**32.24**	**26.19**/**60.40**
XT32	Oracle	100	26.59	36.84/72.47	19.26/41.01	16.23/32.74	34.01/66.59
M1 ⇓ XT32	DT	0	14.07	22.10/59.86	10.44/21.48	06.29/15.24	17.44/45.69
	ST3D [45]	-68.61	05.48	11.39/38.05	02.92/06.78	00.00/00.00	02.92/06.78
	ST3D++ [46]	-39.54	09.12	14.56/47.94	07.47/22.07	03.15/05.02	11.28/32.18
	DTS [16]	**53.51**	**20.77**	**30.62**/68.03	**15.67**/33.01	09.48/21.63	**27.33**/57.48
	DIG(Ours)	39.06	18.96	25.95/**68.78**	14.77/**34.65**	**11.45**/**26.47**	23.66/**60.92**
OS128 ⇓ XT32	DT	0	11.77	22.25/58.20	08.45/19.16	01.66/04.50	14.71/41.05
	ST3D [45]	-43.99	05.34	11.46/45.41	01.72/05.87	00.00/00.00	08.18/22.82
	ST3D++ [46]	07.56	12.89	17.81/56.99	05.31/19.98	10.69/24.22	17.75/48.29
	DTS [16]	34.50	16.09	18.87/59.78	**13.63**/**27.70**	11.26/25.49	20.59/58.19
	DIG(Ours)	**49.87**	**19.16**	**27.20**/**63.89**	10.75/26.79	**11.76**/**27.16**	**26.91**/**64.22**

Table 4: Detailed experimental results for all cross-mechanism settings. For each category, we provide the Average Precision (AP) results for all distances and within a range of 30 meters.

aligns intensity well, which explains its strong performance in these cases. (2) The results indicate a significant decrease in mAP while using OS128 as the target domain. This decrease is primarily attributed to its unique intensity range of [0, 512], unlike the [0, 256] range of other sensors. The pre-trained model finds it challenging to handle unfamiliar intensities. Unlike the other three methods, which either discarded the intensity information or did not include a dedicated module for it, we addressed this issue with BCN, resulting in a significant performance improvement. (3) All these methods perform unsatisfactorily for domain adaptation from XT32 to M1. Even DIG, the best one among them, gets CGs of 25.22 on it. This observation means that M1 and XT32 may suffer from larger domain gaps, highlighting the need for further investigation in the future. (4) ST3D++ gets a CG of 54.49 for XT32 to OS128, which is mainly due to the

great performance on APs for 0-30m. As they are both MS LiDAR, they suffer more from differences in density. Pseudo-labels from XT32 might look like harder cases for OS128. Refining model parameters with these cases can dramatically improve the performance. (5) It indicates that DTS plays a crucial role when there is a substantial difference in the number of beams. CG on tasks that contain XT32 gets nice performance improvement. Specifically, CG can reach 34.51 and 53.51 for OS128 to XT32 and M1 to XT32 respectively.

modules			M1 ⇒ OS128		OS128 ⇒ XT32	
BDS	BCN	GAM	mAP	CG	mAP	CG
✓			09.22	11.78	14.69	19.70
	✓		17.14	52.53	16.47	31.71
✓	✓		17.47	54.04	18.95	48.45
✓	✓	✓	**17.54**	**54.41**	**19.16**	**49.87**

Table 5: Ablation study for DIG baseline method.

Ablation study for DIG. We conduct ablation studies to analyze the effectiveness of each component in the DIG baseline method. Two representative tasks are selected. $M1 \Rightarrow OS128$ and $OS128 \Rightarrow XT32$ respectively refer to domain adaptation for sensors with different scanning patterns and different beams/densities. As shown in Table 5, For both tasks, BCN is anticipated to be the most effective module, contributing 52.53 and 31.71 CG on each respective task. This is attributed to the substantial intensity differences between the source and target domains in both tasks. BDS also exhibits commendable performance in both tasks, achieving 11.78 and 19.70 CG, respectively. Although less conspicuous, GAM proves to be effective for both tasks.

5 Conclusion

We presented a dataset specifically designed for cross-mechanism domain adaptation, incorporating mechanical LiDARs with both high-res and low-res beams, solid-state LiDAR, and 4D millimeter-wave radar. These sensors undergo precise time synchronization, allowing them to simultaneously generate point clouds of the same external environment. Data from diverse scenes were selected and meticulously annotated. We believe the proposed dataset would hugely facilitate research of cross-mechanism 3D detection, as it currently stands as the most comprehensive point cloud 3D detection dataset in terms of sensor types. We also provide a novel DIG method as well as complete experimental results of recent methods on our dataset. They can be used as reliable baselines to further benefit the research communities.

In the future, we will continue to explore more usage of CMD, including utilizing data from multiple mechanisms as the source domain to enhance domain adaptation effects, and employing CMD as a tool for validating the performance of multi-modal 3D object detection.

References

1. Ieee standard for a precision clock synchronization protocol for networked measurement and control systems. IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002) pp. 1–269 (2008). https://doi.org/10.1109/IEEESTD.2008.4579760
2. Atkinson, A.C., Riani, M., Corbellini, A.: The box–cox transformation: Review and extensions (2021)
3. Bai, X., Hu, Z., Zhu, X., Huang, Q., Chen, Y., Fu, H., Tai, C.L.: Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR) (2022)
4. Bradski, G.: The opencv library. Dr. Dobb's Journal: Software Tools for the Professional Programmer **25**(11), 120–123 (2000)
5. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2020). https://doi.org/10.1109/cvpr42600.2020.01164, http://dx.doi.org/10.1109/cvpr42600.2020.01164
6. Chang, M.F., Lambert, J., Sangkloy, P., Singh, J., Bak, S., Hartnett, A., Wang, D., Carr, P., Lucey, S., Ramanan, D., et al.: Argoverse: 3d tracking and forecasting with rich maps. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8748–8757 (2019)
7. Chen, Y., Yu, Z., Chen, Y., Lan, S., Anandkumar, A., Jia, J., Alvarez, J.M.: Focalformer3d: Focusing on hard instance for 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8394–8405 (2023)
8. Chen, Y., Liu, J., Zhang, X., Qi, X., Jia, J.: Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21674–21683 (2023)
9. Deng, J., Shi, S., Li, P., gang Zhou, W., Zhang, Y., Li, H.: Voxel r-cnn: Towards high performance voxel-based 3d object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence (2021)
10. Ding, G., Zhang, M., Li, E., Hao, Q.: Jst: Joint self-training for unsupervised domain adaptation on 2d&3d object detection. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 477–483. IEEE (2022)
11. Fang, J., Zhou, D., Zhao, J., Tang, C., Xu, C.Z., Zhang, L.: Lidar-cs dataset: Lidar point cloud dataset with cross-sensors for 3d object detection (Jan 2023)
12. Fetić, A., Jurić, D., Osmanković, D.: The procedure of a camera calibration using camera calibration toolbox for matlab. In: 2012 Proceedings of the 35th International Convention MIPRO. pp. 1752–1757. IEEE (2012)
13. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (Jun 2012). https://doi.org/10.1109/cvpr.2012.6248074, http://dx.doi.org/10.1109/cvpr.2012.6248074
14. Hegde, D., Sindagi, V., Kilic, V., Cooper, A.B., Foster, M., Patel, V.: Uncertainty-aware mean teacher for source-free unsupervised domain adaptive 3d object detection. arXiv preprint arXiv:2109.14651 (2021)
15. Houston, J., Zuidhof, G., Bergamini, L., Ye, Y., Chen, L., Jain, A., Omari, S., Iglovikov, V., Ondruska, P.: One thousand and one hours: Self-driving motion prediction dataset. In: Conference on Robot Learning. pp. 409–418. PMLR (2021)

16. Hu, Q., Liu, D., Hu, W.: Density-insensitive unsupervised domain adaption on 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17556–17566 (2023)
17. Huang, X., Wu, H., Li, X., Fan, X., Wen, C., Wang, C.: Sunshine to rainstorm: Cross-weather knowledge distillation for robust 3d object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 2409–2416 (2024)
18. Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D., Han, S.: Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. ArXiv (2022)
19. Luo, Z., Cai, Z., Zhou, C., Zhang, G., Zhao, H., Yi, S., Lu, S., Li, H., Zhang, S., Liu, Z.: Unsupervised domain adaptive 3d detection with multi-level consistency. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8866–8875 (2021)
20. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. vol. 1, pp. 281–297. Oakland, CA, USA (1967)
21. Mao, J., Niu, M., Jiang, C., Liang, H., Liang, X., Li, Y., Ye, C., Zhang, W., Li, Z., Yu, J., Xu, H., Xu, C.: One million scenes for autonomous driving: Once dataset. Cornell University - arXiv,Cornell University - arXiv (Jun 2021)
22. Paek, D.H., Kong, S.H., Wijaya, K.: K-radar: 4d radar object detection dataset and benchmark for autonomous driving in various weather conditions (Jun 2022)
23. Palffy, A., Pool, E., Baratam, S., Kooij, J., Gavrila, D.: Multi-class road user detection with 3+1d radar in the view-of-delft dataset
24. Peng, X., Zhu, X., Ma, Y.: Cl3d: Unsupervised domain adaptation for cross-lidar 3d detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 2047–2055 (2023)
25. Raj, T., Hanim Hashim, F., Baseri Huddin, A., Ibrahim, M.F., Hussain, A.: A survey on lidar scanning mechanisms. Electronics **9**(5), 741 (2020)
26. Rochan, M., Chen, X., Grandhi, A., Corral-Soto, E.R., Liu, B.: Domain adaptation in 3d object detection with gradual batch alternation training. arXiv preprint arXiv:2210.10180 (2022)
27. Roriz, R., Cabral, J., Gomes, T.: Automotive lidar technology: A survey. IEEE Transactions on Intelligent Transportation Systems **23**(7), 6282–6297 (2021)
28. Segal, A., Haehnel, D., Thrun, S.: Generalized-icp. In: Robotics: science and systems. vol. 2, p. 435. Seattle, WA (2009)
29. Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H.: Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR). pp. 10526 – 10535 (2020)
30. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in perception for autonomous driving: Waymo open dataset. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2020). https://doi.org/10.1109/cvpr42600.2020.00252, http://dx.doi.org/10.1109/cvpr42600.2020.00252
31. Team, O.D.: Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet (2020)
32. Tsai, D., Berrio, J.S., Shan, M., Nebot, E., Worrall, S.: Ms3d: Leveraging multiple detectors for unsupervised domain adaptation in 3d object detection. arXiv preprint arXiv:2304.02431 (2023)

33. Tsai, D., Berrio, J.S., Shan, M., Nebot, E., Worrall, S.: Viewer-centred surface completion for unsupervised domain adaptation in 3d object detection. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 9346–9353. IEEE (2023)
34. Tsai, D., Berrio, J.S., Shan, M., Worrall, S., Nebot, E.: See eye to eye: A lidar-agnostic 3d detection framework for unsupervised multi-target domain adaptation. IEEE Robotics and Automation Letters **7**(3), 7904–7911 (2022)
35. Wang, Y., Chen, X., You, Y., Li, L.E., Hariharan, B., Campbell, M., Weinberger, K.Q., Chao, W.L.: 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. pp. 11713–11723 (2020)
36. Wang, Y., Yin, J., Li, W., Frossard, P., Yang, R., Shen, J.: Ssda3d: Semi-supervised domain adaptation for 3d object detection from point cloud. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 2707–2715 (2023)
37. Wang, Z., Ding, S., Li, Y., Fenn, J., Roychowdhury, S., Wallin, A., Martin, L., Ryvola, S., Sapiro, G., Qiu, Q.: Cirrus: A long-range bi-pattern lidar dataset. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 5744–5750. IEEE (2021)
38. Wei, Y., Wei, Z., Rao, Y., Li, J., Zhou, J., Lu, J.: Lidar distillation: Bridging the beam-induced domain gap for 3d object detection. In: European Conference on Computer Vision. pp. 179–195. Springer (2022)
39. Wu, H., Deng, J., Wen, C., Li, X., Wang, C.: Casa: A cascade attention network for 3d object detection from lidar point clouds. IEEE Transactions on Geoscience and Remote Sensing (2022)
40. Wu, H., Wen, C., Li, W., Yang, R., Wang, C.: Learning transformation-equivariant features for 3d object detection (2022)
41. Wu, H., Wen, C., Shi, S., Li, X., Wang, C.: Virtual sparse convolution for multimodal 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21653–21662 (2023)
42. Xia, Q., Chen, Y., Cai, G., Chen, G., Xie, D., Su, J., Wang, Z.: 3-d hanet: A flexible 3-d heatmap auxiliary network for object detection. IEEE Transactions on Geoscience and Remote Sensing **61**, 1–13 (2023)
43. Xia, Q., Deng, J., Wen, C., Wu, H., Shi, S., Li, X., Wang, C.: Coin: Contrastive instance feature mining for outdoor 3d object detection with very limited annotations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6254–6263 (2023)
44. Xiao, P., Shao, Z., Hao, S., Zhang, Z., Chai, X., Jiao, J., Li, Z., Wu, J., Sun, K., Jiang, K., Wang, Y., Yang, D.: Pandaset: Advanced sensor suite dataset for autonomous driving. In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC) (Sep 2021). https://doi.org/10.1109/itsc48978.2021.9565009, http://dx.doi.org/10.1109/itsc48978.2021.9565009
45. Yang, J., Shi, S., Wang, Z., Li, H., Qi, X.: St3d: Self-training for unsupervised domain adaptation on 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10368–10378 (2021)
46. Yang, J., Shi, S., Wang, Z., Li, H., Qi, X.: St3d++: Denoised self-training for unsupervised domain adaptation on 3d object detection. IEEE transactions on pattern analysis and machine intelligence **45**(5), 6354–6371 (2022)
47. Yang, Z., Sun, Y., Liu, S., Jia, J.: 3dssd: Point-based 3d single stage object detector. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR). pp. 11040–11048 (2020)

48. Yihan, Z., Wang, C., Wang, Y., Xu, H., Ye, C., Yang, Z., Ma, C.: Learning transferable features for point cloud detection via 3d contrastive co-training. Advances in Neural Information Processing Systems **34**, 21493–21504 (2021)
49. Yin, T., Zhou, X., Krähenbühl, P.: Center-based 3d object detection and tracking. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR) (2021)
50. You, Y., Diaz-Ruiz, C.A., Wang, Y., Chao, W.L., Hariharan, B., Campbell, M., Weinbergert, K.Q.: Exploiting playbacks in unsupervised domain adaptation for 3d object detection in self-driving cars. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 5070–5077. IEEE (2022)
51. You, Y., Phoo, C.P., Luo, K., Zhang, T., Chao, W.L., Hariharan, B., Campbell, M., Weinberger, K.Q.: Unsupervised adaptation from repeated traversals for autonomous driving. Advances in Neural Information Processing Systems **35**, 27716–27729 (2022)
52. Zheng, L., Ma, Z., Zhu, X., Tan, B., Li, S., Long, K., Sun, W., Chen, S., Zhang, L., Wan, M., Huang, L., Bai, J.: Tj4dradset: A 4d radar dataset for autonomous driving (Apr 2022)