Perspective-Invariant 3D Object Detection

Ao Liang[*,1,2,3,4] Lingdong Kong[*,1,5] Dongyue Lu[*,1] Youquan Liu[6]
Jian Fang[4] Huaici Zhao[4,✉] Wei Tsang Ooi[1,✉]

[1]National University of Singapore [2]University of Chinese Academy of Sciences
[3]Key Laboratory of Opto-Electronic Information Processing, Chinese Academy of Sciences
[4]Shenyang Institute of Automation, Chinese Academy of Sciences [5]CNRS@CREATE [6]Fudan University

Project Page: *Link* **GitHub: *Link*** **Dataset: *Link***

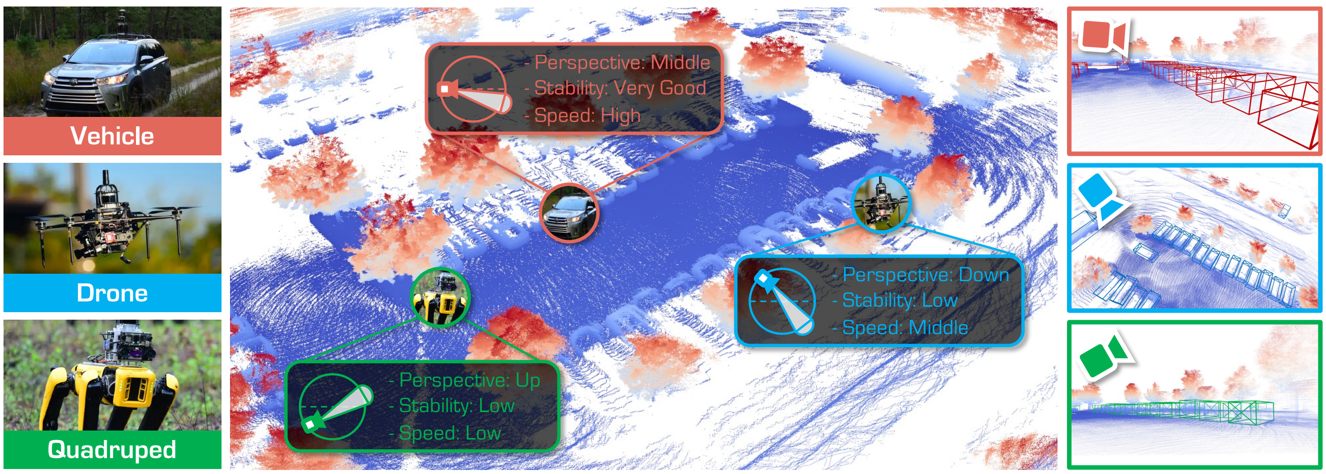

Figure 1. Motivation of Perspective invariant 3D object DETection (**Pi3DET**). We focus the practical yet challenging task of 3D object detection from heterogeneous robot platforms: **Vehicle**, **Drone**, and **Quadruped**. To achieve strong generalization, we contribute: **1)** The first **dataset** for multi-platform 3D detection, comprising more than **51**K LiDAR frames with over **250k** meticulously annotated 3D bounding boxes; **2)** An adaptation **framework**, effectively transfers capabilities from vehicles to other platforms by integrating geometric and feature-level representations; **3)** A comprehensive **benchmark** study of state-of-the-art 3D detectors on cross-platform scenarios.

Abstract

*With the rise of robotics, LiDAR-based 3D object detection has garnered significant attention in both academia and industry. However, existing datasets and methods predominantly focus on vehicle-mounted platforms, leaving other autonomous platforms underexplored. To bridge this gap, we introduce **Pi3DET**, the first benchmark featuring LiDAR data and 3D bounding box annotations collected from multiple platforms: vehicle, quadruped, and drone, thereby facilitating research in 3D object detection for non-vehicle platforms as well as cross-platform 3D detection. Based on Pi3DET, we propose a novel cross-platform adaptation framework that transfers knowledge from the well-studied vehicle platform to other platforms. This framework achieves perspective-invariant 3D detection through robust alignment at both geometric and feature levels. Additionally, we establish a benchmark to evaluate the resilience and robustness of current 3D detectors in cross-platform scenarios, providing valuable insights for developing adaptive 3D perception systems. Extensive experiments validate the effectiveness of our approach on challenging cross-platform tasks, demonstrating substantial gains over existing adaptation methods. We hope this work paves the way for generalizable and unified 3D perception systems across diverse and complex environments. Our Pi3DET dataset, cross-platform benchmark suite, and annotation toolkit have been made publicly available.*

(∗) Ao, Lingdong, and Dongyue contributed equally to this work.

1. Introduction

LiDAR-based 3D object detection provides detailed spatial and geometric information about objects of interest, attracting significant research attention [1, 39, 48, 115]. Despite this trend, existing datasets [8, 22, 54, 76] and methods [31, 33, 43, 69, 72, 102, 113] predominantly target autonomous vehicles, leaving other platforms underexplored.

Table 1. **Summary of LiDAR-based 3D object detection datasets**. We compare **key aspects** from [1]robot platforms, [2]scale, [3]sensor setups, [4]temporal (Temp.), [5]multi-conditions, *etc*. To our knowledge, **Pi3DET** stands out as the first work to feature multi-platform 3D detection from **Vehicle**, **Drone**, and **Quadruped**, with fine-grained 3D bounding box annotations, conditions, and practical use cases.

Dataset	Venue	Platform: Vehicle	Platform: Drone	Platform: Quadruped	# of Frames	LiDAR Setup	Temp.	Freq. (Hz)	Condition: Day	Condition: Night	Other Sensors Supported
KITTI [22]	CVPR'12	✓	✗	✗	$14,999$	1×64	No	-	✓	✗	RGB, IMU, Stereo
ApolloScape [28]	TPAMI'18	✓	✗	✗	$143,906$	1×64	Yes	2	✓	✓	RGB, IMU, Radar
Waymo Open [76]	CVPR'19	✓	✗	✗	$198,000$	$1 \times 64, 4 \times 16$	Yes	10	✓	✓	RGB, IMU, Radar
nuScenes [8]	CVPR'20	✓	✗	✗	$35,149$	1×32	Yes	2	✓	✓	RGB, IMU, Radar
ONCE [54]	arXiv'21	✓	✗	✗	$\sim$ 1M	1×40	No	2	✓	✓	RGB, IMU
Argoverse 2 [85]	NeurIPS'21	✓	✗	✗	$\sim$ 6M	2×32	Yes	10	✓	✗	RGB, IMU
aiMotive [57]	ICLRW'23	✓	✗	✗	$26,583$	1×64	Yes	10	✓	✓	RGB, IMU
Zenseact Open [2]	ICCV'23	✓	✗	✗	$\sim$ 100K	$1 \times 128, 4 \times 16$	Yes	1	✓	✓	RGB, IMU
MAN TruckScenes [21]	NeurIPS'24	✓	✗	✗	$\sim$ 30K	6×64	Yes	2	✓	✓	RGB, IMU, Radar
AeroCollab3D [78]	TGRS'24	✗	✓	✗	$3,200$	N/A	No	-	✓	✗	RGB, IMU
Pi3DET (M3ED)	**Ours**	✓	✓	✓	$\mathbf{51,545}$	$\mathbf{1 \times 64}$	**Yes**	**10**	✓	✓	**RGB**, **IMU**, **Stereo**, **Event**

With rapid advancements in robotics, autonomous systems such as quadrupeds and drones are becoming increasingly vital for diverse real-world applications [3, 5, 9, 26, 37, 50, 61, 78, 91]. Equipping these emerging platforms with accurate 3D perception capabilities comparable to those of autonomous vehicles is therefore highly significant [6, 23, 34, 38, 48, 89]. Currently, research into non-vehicle platforms remains sparse [14, 41, 53, 64, 78], revealing a critical gap in cross-platform 3D object detection studies.

A major barrier impeding progress in multi-platform detection is the lack of annotated multi-platform LiDAR datasets. Current benchmarks almost exclusively focus on vehicles [8, 22, 74, 76, 108]. Although some drone datasets exist [9, 78], they often lack comprehensive 3D annotations and sufficient platform diversity. Chaney *et al*. introduce M3ED [9], a dataset compiled from multiple platforms. However, the lack of annotated 3D bounding boxes currently limits its direct applicability for 3D detection tasks. Training platform-specific models independently is both resource-intensive and impractical for real-world deployment, especially in resource-constrained scenarios. Cross-platform adaptation, transferring knowledge from well-studied vehicle datasets to other platforms like drones and quadrupeds, emerges as a promising alternative. Existing domain adaptation techniques [116], however, primarily tackle cross-dataset shifts and neglect intrinsic geometric discrepancies caused by differences in platform dynamics and sensor viewpoints.

To address these limitations, we introduce **Pi3DET**, the **first** publicly available multi-platform 3D detection dataset. Our dataset consists of $\mathbf{51{,}545}$ LiDAR frames with over $\mathbf{250{,}000}$ meticulously annotated 3D bounding boxes spanning **Vehicle**, **Drone**, and **Quadruped**. Our dataset is constructed using an automated labeling pipeline, supplemented by extensive manual refinement totaling approximately $\mathbf{500}$ hours. As detailed in Tab. 1, Pi3DET contains $\mathbf{25}$ sequences covering diverse environments under varying day and night conditions (examples in Appendix A.3). Analyses of Pi3DET highlight **three crucial discrepancies across platforms**: differences in ego-motion characteristics, variations in point-cloud distributions, and distinct bounding box properties, underscoring the necessity for specialized adaptation methods and techniques.

Motivated by these insights, we propose **Pi3DET-Net**, a novel cross-platform adaptation framework. Our approach consists of two stages. In the *Pre-Adaptation (PA)* stage, we learn global transformations and extract geometric cues from the source platform. In the *Knowledge Adaptation (KA)* stage, we propagate the acquired knowledge and align features between the source and target platforms to improve cross-platform generalization. In particular, our method effectively bridges the platform gap among heterogeneous robotic systems at both the **geometric** and **feature** levels:

■ **Geometry-Level.** We develop *Random Platform Jitter (RPJ)* to augment source data with simulated ego-motion disturbances, enhancing robustness to platform-specific motion variations. Moreover, *Virtual Platform Pose (VPP)* projects target platform point clouds into a source-like coordinate frame, mitigating viewpoint discrepancies.

■ **Feature-Level.** Our *Geometry-Aware Transformation Descriptor (GTD)* encodes platform-specific geometric properties (*e.g.*, sensor elevation distributions), guiding effective feature alignment. The proposed *KL Probabilistic Feature Alignment (PFA)* leverages variational inference to minimize domain-specific distribution gaps, thereby facilitating accurate platform-specific pose adaptation.

Extensive experiments on KITTI [22], nuScenes [8], and our **Pi3DET** validate our effectiveness. Specifically, Pi3DET-Net achieves mAP gains of $+11.84\%$ and $+12.03\%$ in Vehicle $\rightarrow$ Drone and Vehicle $\rightarrow$ Quadruped adaptations, respectively. Additionally, cross-dataset experiments show an average improvement of $+25.27\%$ mAP over source-only methods in the nuScenes $\rightarrow$ KITTI scenario. We further establish a **comprehensive benchmark** on Pi3DET with 18 state-of-the-art detectors, identifying insights to enhance resilience against platform variations. When combined with these detectors, our method consistently boosts performance, underscoring its architecture-agnostic nature and wide applicability. In summary, the contributions of this work are:

- We introduce **Pi3DET**, a diverse and large-scale multi-platform 3D object detection dataset, serving as a solid foundation for cross-platform 3D detection research.
- We propose a novel cross-platform 3D object detection framework, Pi3DET-Net, to effectively transfer 3D detection capabilities from vehicles to other platforms by integrating geometric and feature-level representations.
- We establish an extensive benchmark, providing crucial insights for future development of generalizable 3D detection systems across heterogeneous robot platforms. To our knowledge, this is the first work in this line of research.

2. Related Work

Datasets & Benchmarks for 3D Detection. LiDAR-based 3D detection aims to estimate an object's 3D position and geometric dimensions [56, 63, 81]. Typical detectors are classified by their approach to process point cloud data: grid-based (using voxels [15, 42, 46, 55], range grids [18, 79, 112] and BEV grids [49, 75, 84], pillars [43, 67, 83], or cylindrical partitions [11, 65, 120]), point-based (directly learning features from raw points [62, 99, 100, 113]), or hybrid point-grid [47, 70, 72, 73], which often delivers state-of-the-art results but at higher computational cost. Datasets such as KITTI [22], nuScenes [8], Waymo Open [76], and others [6, 28, 54, 85, 88] have driven progress in accuracy [49, 70], robustness [17, 25, 32, 36, 74], and efficiency [95, 101]. Yet, most research targets vehicle-mounted sensors, leaving quadrupeds and drones underexplored despite similar LiDAR payloads. To address this gap, we present **Pi3DET**, the **first** publicly available dataset incorporating heterogeneous data from multi-platform setups for 3D object detection.

Cross-Dataset 3D Detection. Prior work transfers knowledge often in cross-dataset settings. ST3D [96] and ST3D++ [98] introduced a three-stage approach (pretraining, pseudo-labeling, and self-training) to improve generalization on target data. Further work refines pseudo-label accuracy [10, 80, 110, 111, 114] and self-training guidance [104, 119], or leverages unified training sets [16, 105] and knowledge distillation [27, 29, 97]. However, most ignore the more challenging cross-platform scenario. While Wozniak *et al.* [86] highlight its importance, they lack a suitable dataset for vehicle-to-other-platform experiments. In contrast, we analyze platform-level shifts and propose the first method tailored for cross-platform transfers. Building on **Pi3DET**, we validate its effectiveness on genuine multi-platform data.

Auto-Labeling 3D Object Detection. Accurate point cloud annotations are crucial for 3D detection, yet labeling a single point cloud can take over 100 seconds [117]. To reduce this burden, researchers have explored semi-automated [51, 87] and fully-automated [109] approaches, including active learning [20, 24, 66, 103], weak supervision [45, 58, 59, 107], and pseudo-label refinement [7, 11, 12, 19, 44, 80, 94]. Recent works integrate vision–language models [52, 90, 92, 107, 117, 118] for greater efficiency. However, these methods primarily target vehicle-mounted platforms. In contrast, we design **Pi3DET-Net** to address multi-platform auto-labeling, including quadruped and drones, to advance 3D object detection in broader operational scenarios.

3. Pi3DET: Dataset & Benchmark

3.1. Motivation

While existing LiDAR-based 3D detection datasets predominantly focus on vehicle data, their utility diminishes for other platforms (*e.g.*, drones and quadrupeds) due to diverging operational perspectives. To address this limitation, we introduce **Pi3DET** (**P**erspective **i**nvariant **3**D object **DET**ection), the first multi-platform dataset for LiDAR-based 3D object detection. Built upon M3ED [9], Pi3DET provides annotated LiDAR sequences across **Vehicle**, **Drone**, and **Quadruped**, specifically designed to advance research in multi-platform 3D object detection.

3.2. Dataset Statistics

Our **Pi3DET** benchmark spans 25 sequences collected from vehicle, quadruped, and drone platforms, annotated at 10 Hz. Compared to other datasets in Tab. 1, Pi3DET provides **51,545 frames** and more than **250,000 box annotations** across two object categories (*Vehicle* and *Pedestrian*), covering day/night conditions in urban, suburban, and rural areas. We combine an automated labeling pipeline with extensive manual refinement, requiring about **500 hours** of human effort. For additional details on the annotation process, dataset statistics, and examples, please refer to Appendix A.

3.3. Perspective Discrepancies Analysis

To quantify cross-platform gaps, we first formalize the problem setup and analyze **geometric discrepancies** across three platforms. We define a point cloud as $\mathcal{P}^\beta = \{\mathbf{p}_i\}_{i=1}^{N^\beta}$, and a single point[1] from the set as $\mathbf{p} = (p^x, p^y, p^z) \in \mathbb{R}^3$, β denotes the platform, including vehicles, drones, and quadrupeds, and N^β is the number of point clouds for platform β. The 3D bounding boxes are denoted by $\mathcal{B}^\beta = \{\mathbf{b}_j\}_{j=1}^{M^\beta}$. We denote one bounding box from this set as $\mathbf{b} = (c^x, c^y, c^z, l, w, h, \varphi) \in \mathbb{R}^7$. Here, $\mathbf{c} = (c^x, c^y, c^z)$ represents the bounding box center, (l, w, h) the dimensions, φ the heading angle, and M^β is the number of bounding box. Additionally, the ego pose is given by a transformation $\mathbf{T} \in \mathrm{SE}(3)$, decomposed into a rotation matrix $\mathbf{R} \in \mathrm{SO}(3)$ (parameterized by Euler angle ϕ, θ, and ψ for roll, pitch, yaw) and a translation vector $\mathbf{t} = [t^x, t^y, t^z]$. We further

[1]For simplicity, we use $\mathbf{p}$ to represent a point from a point cloud, rather than explicitly referencing each individual sample from the point set. The same applies to the 3D bounding boxes.

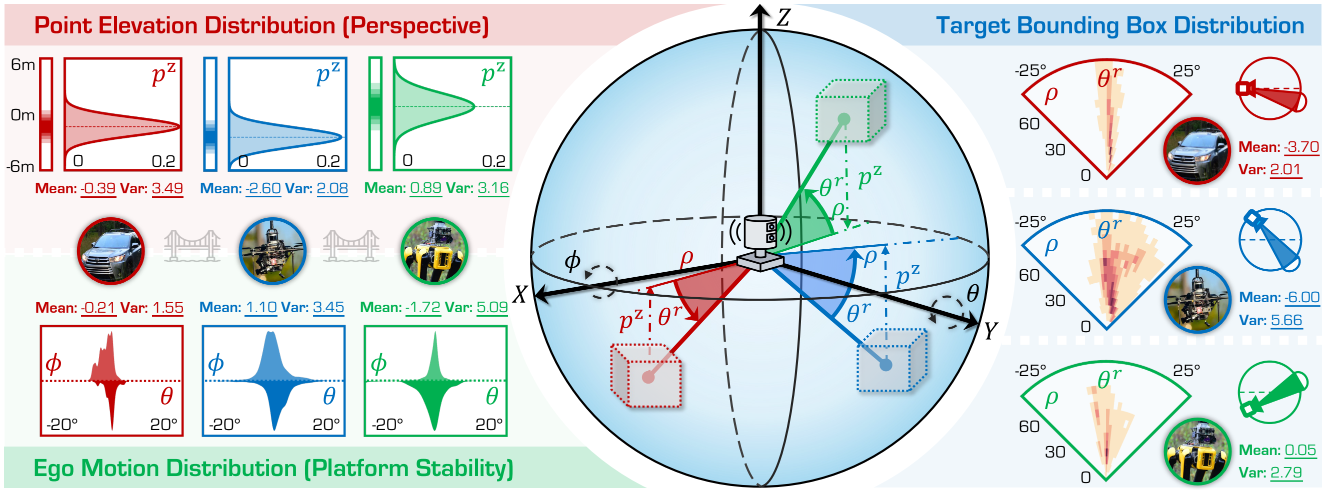

Figure 2. Analysis of perspective differences across three robot platforms. We present the statistics of point elevation distribution (**upper-left**), ego motion distribution (**bottom-left**), and target bounding box distribution (**right**), along with means and variances for each platform's data. We use different colors to denote different platforms for simplicity, *i.e.*, **Vehicle**, **Drone**, and **Quadruped**. Best viewed in colors.

define the distance between the target bounding box and the ego platform in bird's-eye view (BEV) as ρ, and denote the relative pitch from the bounding box to the ego platform in the ego coordinate system as θ^r. As shown in Fig. 2, we identify **three** critical cross-platform discrepancies.

Ego Motion Distributions. Vehicle-mounted LiDAR sensors exhibit stable motion with minimal roll/pitch variance ($\phi, \theta < 5°$). In contrast, drones and quadrupeds suffer significant ego jitter due to dynamic locomotion and aerodynamics, inducing roll/pitch fluctuations up to 20°, shown in the bottom-left part in Fig. 2. This instability introduces high-frequency perturbations in point cloud geometry.

Point Elevation Distributions. Beyond the roll and pitch jitter caused by ego motion, the overall distribution of the elevation p^z of the input point cloud varies significantly among the platforms due to their different intrinsic heights. As shown in the upper-left in Fig. 2, for vehicles, most points lie slightly below their own height ($p^z < t^z$). In contrast, on quadrupeds, the points cluster above the height of platform ($p^z > t^z$), while for drones, the points are distributed substantially lower than the drone's altitude ($p^z << t^z$).

Target Bounding Box Distributions. Variations in platform height influence the relative orientation of the detected object. The right part of Fig. 2 shows the relationship between targets' relative pitch angles θ^r and BEV distances ρ. Comparatively, drones observe objects with larger downward pitch angles and large variances, indicating that targets are positioned lower relative to the ego platform with a more uneven distribution. In contrast, quadrupeds exhibit larger upward pitch angles, suggesting that objects are relatively higher in their view. Vehicles, benefiting from stable motion, display the smallest variance in pitch angle distribution.

These discrepancies make single-platform models ineffective for cross-platform deployment. Training separate models for each platform is resource-intensive and impractical for real-world scalability. Instead, we aim to propose a unified cross-platform adaptation framework that trains on large-scale readily available source platform data ($\mathcal{S}$, *e.g.*, vehicle) and generalizes to target platform data ($\mathcal{T}$) without target labels, addressing geometric shifts through perspective-invariant learning.

4. Methodology

As illustrated in Fig. 3, we propose a two-stage **Pi3DET-Net** consisting of *Pre-Adaption (PA)* and *Knowledge-Adaption (KA)* for cross-platform adaptation. For geometric alignment (Sec. 4.1), Random Platform Jitter facilitates robustness against ego-motion variations, while Virtual Platform Pose aligns viewpoints. For feature alignment (Sec. 4.2), KL Probabilistic Feature Alignment aligns target features with the source space, and a Geometry-Aware Transformation Descriptor corrects global transformations across platforms. The training pipeline is illustrated in Sec. 4.3.

4.1. Cross-Platform Geometry Alignment

As outlined in Sec. 3.3, platform-induced point cloud discrepancies arise from varying ego motions, point elevations, and target bounding box distributions. To mitigate these, we propose two complementary strategies. First, we apply Random Platform Jitter during PA on the source platform, enhancing robustness to pose jitter. Second, we use a Virtual Platform Pose in KA on the target platform to achieve effective scene alignment. Together, these approaches enable smoother geometric adaptation from source to target.

Random Platform Jitter (RPJ). To emulate the roll and pitch jitters observed on quadruped and drone platforms, we introduce Random Platform Jitter during PA on the

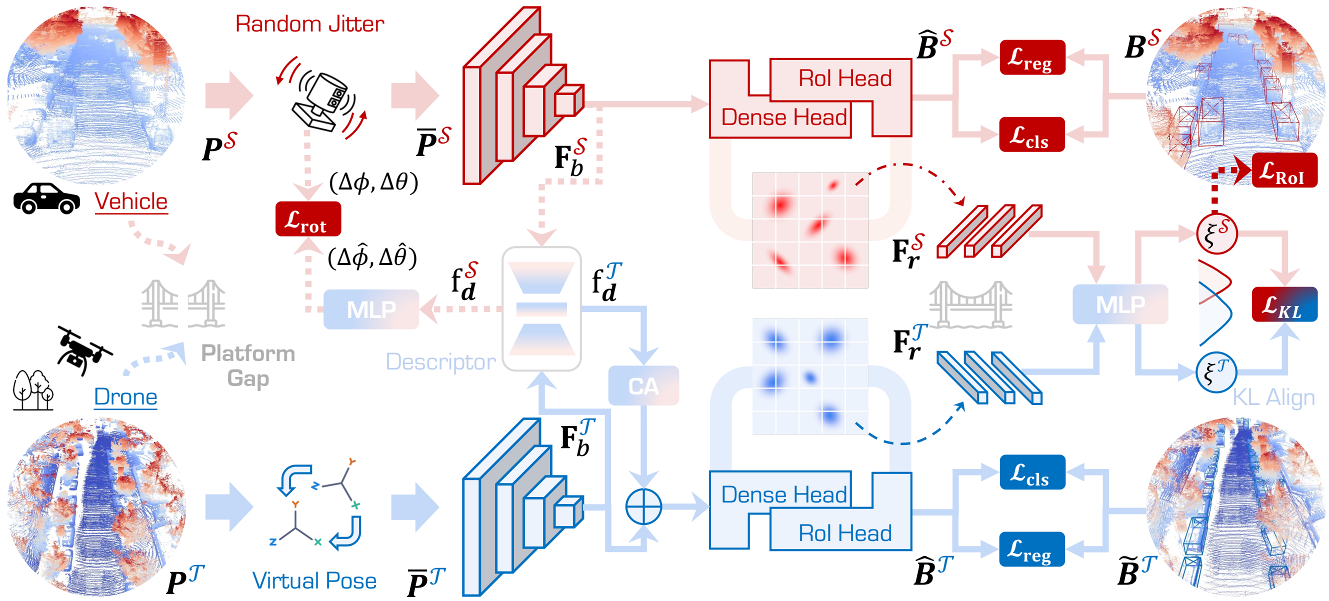

Figure 3. **Framework Overview.** The proposed **Pi3DET-Net** consists of two main stages: *Pre-Adaption (PA)* and *Knowledge-Adaption (KA)*, aiming at bridging the gap across heterogeneous robot platforms through alignment at both geometric (Sec. 4.1) and feature levels (Sec. 4.2). On the geometric side, PA employs *Random Platform Jitter* to enhance robustness against ego-motion variations, while KA uses *Virtual Platform Pose* to simulate source-like viewpoints to achieve bidirectional geometric alignment across platforms. On the feature side, Pi3DET-Net further incorporates *KL Probabilistic Feature Alignment* to align target features with the source space, along with a *Geometry-Aware Transformation Descriptor* to correct global transformations across platforms.

source platform. Specifically, we sample two angles $\Delta\phi$ and $\Delta\theta$ from a uniform distribution for roll and pitch, and define a composite rotation $\mathbf{R}(\Delta\phi, \Delta\theta)$. For point $\mathbf{p} \in \mathcal{P}^{\mathcal{S}}$, bounding-box $\mathbf{b} \in \mathcal{B}^{\mathcal{S}}$ and its center $\mathbf{c}$, we have:

$$\bar{\mathbf{p}} = \mathbf{R}(\Delta\phi, \Delta\theta)\,\mathbf{p}\,, \quad \bar{\mathbf{c}} = \mathbf{R}(\Delta\phi, \Delta\theta)\,\mathbf{c}\,. \tag{1}$$

Here, the box dimensions are unchanged, and the heading angle is preserved. The transformed point cloud $\bar{\mathcal{P}}^{\mathcal{S}}$ is then input into the backbone for feature extraction. Exposing the model to these rotated point cloud inputs tends to enhance the robustness to roll-pitch variations on target platforms.

Virtual Platform Pose (VPP). We establish a virtual pose on the target platform during KA to mimic the source viewpoint and reduce the platform geometry gap. Since input point cloud and bounding box distributions diverge, we define a virtual pose $\bar{\mathbf{T}}$ from the actual ego pose $\mathbf{T}$. We set roll and pitch to zero ($\bar{\phi} = 0, \bar{\theta} = 0$), keep the actual yaw ($\bar{\psi} = \psi$), and preserve planar coordinates ($\bar{t}^x = t^x, \bar{t}^y = t^y$), fixing the height at $\bar{t}^z = t^z_{\text{vehicle}}$. Given a point cloud $\mathbf{p} \in \mathcal{P}^{\mathcal{T}}$ from target platform, along with the bounding box $\mathbf{b} \in \mathcal{B}^{\mathcal{T}}$ and its center $\mathbf{c}$, we express them in homogeneous coordinates $\mathbf{P}, \mathbf{C}$, and then transform them to the following:

$$\bar{\mathbf{P}} = \bar{\mathbf{T}}\,\mathbf{T}^{-1}\,\mathbf{P}\,, \quad \bar{\mathbf{C}} = \bar{\mathbf{T}}\,\mathbf{T}^{-1}\,\mathbf{C}\,. \tag{2}$$

Here, dimensions remain unchanged, while the heading φ is offset by $\Delta(\bar{\psi}, \psi)$. The resulting point cloud $\bar{\mathcal{P}}^{\mathcal{T}}$ is used for feature extraction. Transforming both point clouds and bounding boxes to this virtual coordinate frame mitigates platform gaps and improves cross-platform adaptations.

4.2. Cross-Platform Feature Alignment

To address domain shifts across platforms, we leverage both probabilistic modeling and global geometric cues to align cross-platform features. As illustrated in Fig. 3, our feature alignment consists of two key components: **1)** a transformation descriptor that learns global geometric invariance; and **2)** a probabilistic feature alignment guided by KL divergence.

Geometry-Aware Transformation Descriptor (GTD). As discussed in Sec. 3.3, differing ego-motion distributions cause global shifts in source and target point clouds. We address these by learning a geometry-aware descriptor on the source platform, then applying it to correct transformations on the target. During PA, we apply global max-pooling to the backbone's feature $\mathbf{F}_b^{\mathcal{S}}$ to obtain a compact vector, which is encoded by a hierarchical convolutional module into a large-scale geometric descriptor $\mathbf{f}_d^{\mathcal{S}}$. A small regression MLP then predicts the artificially introduced random jitter angles $(\Delta\hat{\theta}, \Delta\hat{\phi})$ from this descriptor, optimizing the following rotation loss:

$$\mathcal{L}_{\text{rot}} = \|\Delta\hat{\phi} - \Delta\phi\|^2 + \|\Delta\hat{\theta} - \Delta\theta\|^2\,. \tag{3}$$

Notably, minimizing $\mathcal{L}_{\text{rot}}$ equips the network with platform-agnostic transformation cues. This descriptor, learned on the source platform, corrects global offsets on the target platform during KA, ensuring robust cross-platform performance.

Table 2. **Comparisons of 3D detection methods for vehicle→drone/quadruped tasks.** We report the average precision (AP) in "BEV / 3D" at the IoU thresholds of 0.7 and 0.5, respectively. Symbol ‡ denotes algorithms *w.o.* ROS [96]. All scores are given in percentage (%). "-" denotes the code is not available. The Best and Second Best scores under each metric are highlighted in Red and Blue, respectively.

#	Method	Vehicle → Quadruped				Vehicle → Drone				Average	
		PV-RCNN [70]		Voxel RCNN [15]		PV-RCNN [70]		Voxel RCNN [15]			
		AP@0.7	AP@0.5	AP@0.7	AP@0.5	AP@0.7	AP@0.5	AP@0.7	AP@0.5	AP@0.7	AP@0.5
nuScenes [8]	Source Platform	43.40 / 33.55	44.86 / 42.84	43.25 / 33.74	45.62 / 43.32	50.91 / 35.26	57.73 / 50.24	50.15 / 29.41	57.10 / 49.10	46.93 / 32.99	51.33 / 46.34
	ST3D [96]	55.40 / 42.02	59.59 / 54.75	44.54 / 35.96	45.81 / 44.38	65.05 / 40.01	68.93 / 64.09	54.62 / 33.79	58.45 / 52.89	54.90 / 37.95	58.20 / 54.03
	ST3D‡ [96]	55.68 / 44.50	59.32 / 55.32	45.01 / 37.13	46.73 / 45.45	65.40 / 43.63	69.24 / 64.88	55.23 / 36.51	59.30 / 54.23	55.33 / 40.44	58.65 / 54.97
	ST3D++ [98]	55.76 / 43.51	59.93 / 55.28	45.56 / 36.97	47.28 / 45.84	60.91 / 40.09	68.96 / 59.96	57.02 / 37.52	61.30 / 55.43	54.81 / 39.52	59.37 / 54.13
	ST3D++‡ [98]	54.96 / 40.81	60.47 / 54.65	45.69 / 36.76	48.30 / 46.05	65.50 / 43.46	68.99 / 64.62	55.92 / 39.46	59.93 / 55.19	55.52 / 40.12	59.42 / 55.13
	REDB [13]	52.43 / 41.34	57.12 / 54.18	- / -	- / -	65.31 / 39.19	68.74 / 64.13	- / -	- / -	- / -	- / -
	MS3D++ [80]	56.24 / 43.20	60.88 / 56.13	51.50 / 40.14	56.03 / 53.86	66.99 / 43.76	69.87 / 65.85	62.68 / 38.26	68.34 / 61.09	59.35 / 41.34	63.78 / 59.23
	Pi3DET-Net	56.80 / 46.36	61.54 / 57.20	54.85 / 42.38	57.41 / 55.54	65.43 / 45.94	69.24 / 65.87	65.63 / 44.62	72.05 / 63.83	60.68 / 44.83	65.06 / 60.61
	Target Platform	54.15 / 40.24	58.63 / 54.96	54.90 / 39.74	56.46 / 55.19	67.67 / 46.11	70.04 / 66.14	68.52 / 46.53	70.67 / 61.42	61.31 / 43.16	63.95 / 59.43
Pi3DET (Vehicle)	Source Platform	38.61 / 26.84	40.64 / 39.22	43.95 / 31.24	48.22 / 44.17	57.29 / 36.62	58.92 / 56.19	52.85 / 37.96	61.10 / 52.47	48.17 / 33.16	52.22 / 48.01
	ST3D [96]	49.29 / 38.69	51.02 / 49.71	47.70 / 37.91	48.07 / 47.59	60.17 / 33.01	62.84 / 54.51	53.79 / 40.18	65.29 / 53.40	52.74 / 37.45	56.81 / 51.30
	ST3D‡ [96]	47.89 / 38.07	49.50 / 48.23	47.01 / 41.85	54.01 / 53.46	60.67 / 33.27	62.98 / 54.61	53.85 / 40.02	62.70 / 53.08	52.35 / 38.30	57.30 / 52.34
	ST3D++ [98]	46.05 / 37.22	49.33 / 47.84	48.52 / 37.84	55.82 / 48.53	60.04 / 33.98	62.71 / 54.13	53.71 / 39.94	62.43 / 53.20	52.08 / 37.24	57.57 / 50.92
	ST3D++‡ [98]	45.14 / 35.70	46.94 / 45.37	47.52 / 37.13	54.37 / 47.63	64.15 / 34.20	63.81 / 55.44	53.64 / 40.27	62.43 / 53.10	52.61 / 36.83	56.89 / 50.38
	REDB [13]	46.74 / 38.47	50.29 / 49.54	- / -	- / -	61.57 / 34.05	63.22 / 54.07	- / -	- / -	- / -	- / -
	MS3D++ [80]	53.66 / 40.66	55.21 / 53.78	53.65 / 41.93	54.69 / 54.00	66.05 / 41.17	67.80 / 63.26	53.85 / 40.91	62.87 / 53.44	56.80 / 41.17	60.14 / 56.12
	Pi3DET-Net	56.19 / 44.28	60.35 / 56.20	55.54 / 45.18	59.48 / 58.90	66.26 / 44.47	68.25 / 63.36	67.87 / 46.83	69.95 / 66.26	61.47 / 45.19	64.51 / 61.18
	Target Platform	54.15 / 40.24	58.63 / 54.96	54.90 / 39.74	56.46 / 55.19	67.67 / 46.11	70.04 / 66.14	68.52 / 46.53	70.67 / 61.42	61.31 / 43.16	63.95 / 59.43
-	Combined All	58.21 / 46.27	62.18 / 59.67	60.96 / 48.15	63.04 / 61.04	68.44 / 48.19	71.11 / 68.24	68.90 / 48.88	72.55 / 69.18	64.13 / 47.87	67.22 / 64.53

KL Probabilistic Feature Alignment (PFA). We aim to reduce cross-platform discrepancies by matching the Region-of-Interest (RoI) feature distributions of source and target platforms during KA.

Specifically, we approximate each platform's RoI features before the detection head with a probabilistic method, ensuring robust distribution alignment. For source-platform RoI feature $\mathbf{F}_r^{\mathcal{S}}$, a probabilistic encoder $p(\xi^{\mathcal{S}}|\mathbf{F}_r^{\mathcal{S}}) = \mathcal{N}\big(\boldsymbol{\mu}(\mathbf{F}_r^{\mathcal{S}}), \boldsymbol{\sigma}^2(\mathbf{F}_r^{\mathcal{S}})\big)$ maps this feature into a Gaussian distribution, which predicts $\boldsymbol{\mu}(\mathbf{F}_r^{\mathcal{S}})$ and $\boldsymbol{\sigma}^2(\mathbf{F}_r^{\mathcal{S}})$ with MLPs. Using the reparameterization trick [30], latent samples $\xi^{\mathcal{S}} = \boldsymbol{\mu}(\mathbf{F}_r^{\mathcal{S}}) + \boldsymbol{\sigma}(\mathbf{F}_r^{\mathcal{S}}) \odot \boldsymbol{\epsilon}$ are generated ($\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$). Analogous encoding applies to the target-platform RoI feature $\mathbf{F}_r^{\mathcal{T}}$, producing latent samples $\xi^{\mathcal{T}}$ accordingly.

Since the true distribution of latent features is unknown, we can only estimate it from latent samples on both platforms. By comparing these samples via the KL term, we have:

$$\mathcal{L}_{\mathrm{KL}} = D_{\mathrm{KL}}\Big[p(\xi^{\mathcal{S}} \mid \mathbf{F}_r^{\mathcal{S}}) \,\Big\|\, p(\xi^{\mathcal{T}} \mid \mathbf{F}_r^{\mathcal{T}})\Big] . \quad (4)$$

The model pushes the target platform's features toward the source manifold. Crucially, this nonadversarial approach provides a stable alignment in the absence of direct target supervision. As investigated by [60], the KL objective not only prevents out-of-distribution samples but also offers a mode-seeking alignment, ultimately improving target performance. For the source platform, we also train a classification head $q(\mathbf{g}|\xi)$ to discriminate foreground from background:

$$\mathcal{L}_{\mathrm{RoI}} = \mathbb{E}_{\xi^{\mathcal{S}} \sim p(\xi^{\mathcal{S}}|F_r^{\mathcal{S}})}\big[-\log q(\mathbf{g}^{\mathcal{S}} \mid \xi^{\mathcal{S}})\big] , \quad (5)$$

where $\mathbf{g}^{\mathcal{S}}$ is the classification task ground truth. This loss ensures the latent representation $\xi^{\mathcal{S}}$ captures semantic features in the source platform for effective alignment through $\mathcal{L}_{\mathrm{KL}}$.

4.3. Objective & Optimization

The overall framework aims to learn global transformations and semantic cues during *Pre-Adaptation*, then propagate and align target data during *Knowledge-Adaptation*.

Pre-Adaptation (PA). In the source platform, our goal is to extract and internalize the necessary knowledge while enhancing geometric robustness through Random Platform Jitter, addressing platform-specific discrepancies through the rotation loss $\mathcal{L}_{\mathrm{rot}}$. and learning RoI-based semantic features via $\mathcal{L}_{\mathrm{RoI}}$. We also apply a standard detection loss composed of a classification loss and a bounding-box regression loss:

$$\mathcal{L}_{\mathrm{det}} = \mathcal{L}_{\mathrm{cls}}(\hat{\mathcal{B}}^{\mathcal{S}}, \mathcal{B}^{\mathcal{S}}) + \mathcal{L}_{\mathrm{reg}}(\hat{\mathcal{B}}^{\mathcal{S}}, \mathcal{B}^{\mathcal{S}}) , \quad (6)$$

where $\hat{\mathcal{B}}^{\mathcal{S}}$ denotes the predicted bounding box. The overall pre-adaptation objective is: $\mathcal{L}_{\mathrm{PA}} = \mathcal{L}_{\mathrm{det}} + \lambda_{\mathrm{rot}}\mathcal{L}_{\mathrm{rot}} + \lambda_{\mathrm{RoI}}\mathcal{L}_{\mathrm{RoI}}$, where λ_{rot} and λ_{RoI} are weights used to balance the losses. This step trains a robust 3D detector while imparting global geometric awareness for adaptation.

Knowledge-Adaptation (KA). After PA, we first use the source-platform knowledge to generate pseudo-annotations $\tilde{\mathcal{B}}^{\mathcal{T}}$ on target data, then train jointly on both platforms:

- **Source Platform:** To preserve source performance, we disable $\mathcal{L}_{\mathrm{rot}}$ and optimize only detection and RoI classification, *i.e.*, $\mathcal{L}_{\mathrm{KA}}^{\mathcal{S}} = \mathcal{L}_{\mathrm{det}}^{\mathcal{S}} + \lambda_{\mathrm{RoI}}\mathcal{L}_{\mathrm{RoI}}^{\mathcal{S}}$.
- **Target Platform:** We encode the learned global descriptor $\mathbf{f}_d^{\mathcal{T}}$ with channel attention (*i.e.*, CA in Fig. 3) and add it to the backbone features as a residual offset, enforce a detection loss, and align RoI features via KL. This process can be formulated as: $\mathcal{L}_{\mathrm{KA}}^{\mathcal{T}} = \mathcal{L}_{\mathrm{det}}^{\mathcal{T}} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}}$, where λ_{KL} is used to balance the KL loss.

The combined objective is $\mathcal{L}_{\mathrm{KA}} = \mathcal{L}_{\mathrm{KA}}^{\mathcal{T}} + \mathcal{L}_{\mathrm{KA}}^{\mathcal{S}}$. By decoupling geometry learning (during PA) from feature correction

Table 3. **Study on cross-platform 3D detection between drone and quadruped platforms**. We report the average precision (AP) in "BEV / 3D" at the IoU thresholds of 0.7 and 0.5, respectively.

#	Method	PV-RCNN [70] AP@0.7	PV-RCNN [70] AP@0.5	Voxel RCNN [15] AP@0.7	Voxel RCNN [15] AP@0.5
Quad → Drone	Source Platform	27.43 / 11.08	36.97 / 27.92	33.22 / 20.20	41.17 / 33.29
	ST3D‡ [96]	33.85 / 18.45	44.35 / 35.83	35.21 / 22.87	36.05 / 35.52
	ST3D++‡ [98]	32.92 / 17.76	40.91 / 32.97	43.30 / 28.86	44.69 / 43.24
	REDB [27]	37.24 / 20.89	44.43 / 37.29	44.27 / 30.55	46.69 / 44.29
	MS3D++ [80]	39.74 / 22.31	47.59 / 41.61	45.84 / 32.21	48.27 / 45.87
	Pi3DET-Net	43.11 / 25.16	52.87 / 47.55	49.27 / 36.24	54.58 / 49.63
	Target Platform	67.67 / 46.11	70.04 / 66.14	68.52 / 46.53	70.67 / 61.42
Drone → Quad	Source Platform	27.23 / 20.36	30.27 / 28.92	32.18 / 23.35	33.94 / 32.70
	ST3D‡ [96]	46.06 / 35.14	51.17 / 49.53	49.04 / 36.94	55.73 / 49.73
	ST3D++‡ [98]	49.09 / 37.57	55.30 / 50.90	48.74 / 38.22	55.19 / 48.94
	REDB [27]	47.29 / 35.67	53.21 / 49.76	49.36 / 38.11	55.96 / 50.21
	MS3D++ [80]	48.24 / 34.12	52.43 / 48.66	49.76 / 37.55	56.17 / 49.97
	Pi3DET-Net	51.24 / 38.94	57.31 / 52.90	52.64 / 38.88	57.57 / 51.83
	Target Platform	54.15 / 40.24	58.63 / 54.96	54.90 / 39.74	56.46 / 55.19

(during KA), the geometry-aware transformation descriptor remains focused on platform-induced differences. Meanwhile, RoI feature alignment pulls target features toward the source distribution, narrowing the cross-platform gap and enabling accurate 3D detection on target platforms.

5. Experiments

5.1. Experimental Settings

Datasets. We evaluate cross-platform and cross-dataset 3D detection using three benchmarks: nuScenes [8], KITTI [22], and our Pi3DET. nuScenes [8] provides 35,149 frames from day and night urban scenes, KITTI [22] provides 14,999 daytime frames, and Pi3DET comprises 51,545 frames spanning urban, suburban, and rural environments. For additional dataset details, please refer to Appendix A.

Benchmark Setup. We design six cross-platform adaptation benchmarks and two cross-dataset adaptation benchmarks to cover a wide range of scenarios and to demonstrate the generalizability of our method. Due to space limits, please refer to Appendix B.6 for the complete benchmark settings.

Baselines. We use PV-RCNN [69] and Voxel-RCNN [15] as our detection backbones. Our comparisons include several related cross-domain detection methods ST3D [96], ST3D++ [96], and MS3D++ [80], as well as three baseline training strategies: training on *"source data only"*, training on *"target data only"*, and training on *"both source and target data"*. For more details, please refer to Appendix B.6.

Implementation Details. Our experiments follow the setting of ST3D++ [98], and are implemented using OpenPCDet [77], with experiments run on two NVIDIA Titan RTX GPUs. We follow the KITTI evaluation protocol by reporting average precision (AP) in both bird's-eye view (BEV) and 3D over 40 recall positions. The hyperparameters are set as $\lambda_{\text{rot}} = 0.1$, $\lambda_{\text{RoI}} = 0.2$, and $\lambda_{\text{KL}} = 10^{-4}$. For more details, please refer to Appendix B.3.

Table 4. **Cross-dataset 3D detection benchmark**. Experiments are conducted on the nuScenes [8] → KITTI [22] task. We report the average precision (AP) in "BEV / 3D" at the IoU thresholds of 0.7, 0.5, and 0.5 for Car, Pedestrian, and Cyclist classes, respectively. The reported AP is for moderate cases. All scores are given in percentage (%). Symbol † denotes method *w.o.* RPJ, since no pitch or roll jitter occurs when both the source and target platforms are vehicles. w.temp indicates the use of temporal information, and w.SN denotes the incorporation of statistic normalization [82].

Method	Car AP@0.7	Pedestrian AP@0.5	Cyclist AP@0.5	Average
Source Dataset	51.80 / 17.90	39.95 / 34.57	17.70 / 11.08	36.48 / 21.18
SN [82]	40.30 / 21.23	38.91 / 34.36	11.11 / 5.67	30.17 / 20.42
ST3D [96]	75.90 / 54.10	44.00 / 42.60	29.58 / 21.21	49.83 / 39.30
ST3D [96] w.SN	79.02 / 62.55	43.12 / 40.54	16.60 / 11.33	46.25 / 38.14
ST3D [96] w.temp	81.06 / 66.98	34.65 / 31.76	27.32 / 20.52	47.68 / 39.75
ST3D++ [98]	80.50 / 62.40	47.20 / 43.96	30.87 / 23.93	52.86 / 43.43
ST3D++ [98] w.SN	78.87 / 65.56	47.94 / 45.57	13.57 / 12.64	46.79 / 41.26
ST3D++ [98] w.temp	80.91 / 68.23	30.48 / 27.86	29.88 / 25.57	47.09 / 40.55
REDB [13]	74.23 / 51.31	25.95 / 18.38	13.82 / 8.64	38.00 / 26.11
DTS [27]	81.40 / 66.60	- / -	- / -	- / -
CMDA [10]	82.13 / 68.95	- / -	- / -	- / -
PLR [114]	73.65 / 66.84	42.69 / 35.47	17.38 / 15.95	44.57 / 39.42
Pi3DET-Net†	82.86 / 70.20	46.23 / 43.44	31.14 / 25.72	57.51 / 46.45
Target Dataset	83.29 / 73.45	46.64 / 41.33	62.92 / 60.32	62.92 / 60.32

5.2. Comparative Study

We analyze the performance of Pi3DET-Net across various cross-platform and cross-dataset adaptation tasks.

Adaptation with Vehicle as Source. Tab. 2 presents the cross-platform adaptation results for vehicle → quadruped/drone tasks. In these experiments, source data are taken from nuScenes [8] and Pi3DET, while all target data come from Pi3DET. Overall, Pi3DET-Net consistently outperforms the baselines. For instance, on the vehicle → quadruped task using nuScenes as source, our method with PV-RCNN achieves a 12.81% gain in AP_{3D}@0.7 compared to the source-only baseline, validating the effectiveness of our approach. Notably, our method even outperforms target-only training, likely due to the smaller target dataset size.

Adaptation with Drone and Quadruped as Source. Tab. 3 presents cross-platform detection results between the quadruped and drone platforms. Under our approach, both PV-RCNN and Voxel-RCNN achieve the best performance across all evaluated metrics. For instance, in the drone → quadruped task, our method with PV-RCNN improves AP_{3D}@0.7 by 18.58% relative to the source-only baseline, nearly matching the target-only performance.

Cross-Dataset Adaptation. To demonstrate the broad applicability of Pi3DET-Net, we evaluate on the cross-dataset task from nuScenes to KITTI. Following [98], we adopt SECOND-IoU [93] as the backbone. Tab. 4 presents the results, which show that Pi3DET-Net achieves state-of-the-art performance on both Car and Cyclist. For Car targets, our AP_{3D}@0.7 is only 3.25% lower than that of the target-only baseline. Additionally, we design a separate cross-dataset adaptation task from nuScenes to Pi3DET on the vehicle platform, detailed analysis is provided in Appendix C.3.

Table 5. **Ablation study of components in Pi3DET-Net**. Experiments are conducted on the vehicle → drone/quadruped tasks. We report the average precision (AP) in "BEV / 3D" at the IoU thresholds of 0.7 and 0.5, respectively. All scores are given in %.

RPJ	VPP	PFA	GTD	Vehicle → Drone AP@0.7	Vehicle → Drone AP@0.5	Vehicle → Quadruped AP@0.7	Vehicle → Quadruped AP@0.5
✗	✗	✗	✗	52.85 / 37.96	61.10 / 52.47	43.95 / 31.24	48.22 / 44.17
✓	✗	✗	✗	60.20 / 39.93	64.76 / 59.52	45.36 / 33.01	49.26 / 47.03
✗	✓	✗	✗	59.83 / 39.26	63.55 / 59.47	44.43 / 32.23	51.59 / 49.47
✓	✓	✗	✗	64.52 / 41.50	66.84 / 60.68	48.45 / 36.10	53.83 / 51.52
✓	✓	✓	✗	67.87 / 46.83	69.95 / 66.26	55.72 / 44.77	59.48 / 58.90
✓	✓	✓	✓	68.48 / 47.75	69.87 / 67.82	55.54 / 45.18	62.02 / 60.29

Table 6. **Cross-platform 3D detection benchmark**. We report the average precision (AP) in "BEV / 3D" at the IoU thresholds of 0.7. All scores are given in percentage (%). "-C" and "-A" denote detectors with the Anchor-based or Center-based detection head.

#	Method	Vehicle AP@0.7	Quadruped AP@0.7	Drone AP@0.7	Average
Grid	PointPillar [40]	51.85 / 44.34	36.24 / 14.51	49.53 / 27.02	45.87 / 28.62
	CenterPoint [102]	51.90 / 42.12	37.74 / 14.68	53.14 / 29.29	47.59 / 28.70
	Part A* [71]	54.88 / 48.23	45.47 / 20.10	56.72 / 34.44	52.36 / 34.26
	Transfusion-L [4]	49.27 / 38.21	36.29 / 14.43	51.27 / 24.63	45.61 / 25.76
	HEDNet [106]	46.73 / 37.60	34.30 / 14.51	49.31 / 20.89	43.45 / 24.33
	SAFNet [35]	42.60 / 34.88	33.47 / 13.65	49.93 / 24.70	42.00 / 24.41
	Part A*+ Ours	53.81 / 47.56	44.31 / 23.73	59.53 / 38.31	52.55 / 36.53
Point	PointRCNN [68]	49.38 / 43.03	41.35 / 23.69	52.59 / 38.67	47.77 / 35.13
	3DSSD [100]	46.58 / 39.88	42.47 / 23.89	51.54 / 37.78	46.86 / 33.85
	IA-SSD [113]	44.00 / 34.91	48.11 / 24.89	59.69 / 35.79	50.60 / 31.86
	DBQ-SSD [99]	41.28 / 33.19	44.27 / 21.85	54.65 / 32.08	46.73 / 29.04
	PointRCNN + Ours	51.19 / 48.09	42.18 / 26.07	57.54 / 41.70	50.30 / 38.62
Grid-Point	PV-RCNN [69]	63.32 / 56.58	45.22 / 22.94	60.11 / 39.68	56.22 / 39.73
	PV-RCNN++ [72]	64.05 / 57.01	47.54 / 22.35	60.54 / 40.10	57.38 / 39.82
	PV-RCNN++-C [72]	57.94 / 50.56	40.75 / 20.78	53.46 / 40.00	50.72 / 37.11
	VoxelRCNN-A [15]	63.00 / 56.98	46.78 / 23.30	64.46 /42.76	58.08 / 41.01
	VoxelRCNN [15]	58.39 / 51.11	48.30 / 21.61	60.29 / 39.15	55.66 / 37.29
	PV-RCNN++ + Ours	63.47 / 56.60	57.08 / 31.09	68.52 / 47.92	63.02 / 45.20

5.3. Ablation Study

In this section, we use Voxel-RCNN [15] as the backbone detector to validate the effectiveness of individual components in Pi3DET-Net for cross-platform tasks.

Random Platform Jitter. As shown in Tab. 5, adding RPJ leads to performance improvements across all metrics. For instance, in the vehicle → drone task, the addition of RPJ boosts AP_{BEV}@0.7 by 7.35% relative to the source-only baseline. These results confirm that simulating ego-motion noise through RPJ effectively augments the source data, thereby enhancing the model's robustness to the jitters observed on non-vehicle platforms.

Virtual Platform Pose. We also evaluate the impact of Virtual Platform Pose (VPP) in Tab. 5. The results clearly show that VPP enhances Pi3DET-Net's performance, achieving a 7% improvement in AP_{3D}@0.5 relative to the source-only baseline in the Vehicle → Drone task. Notably, when RPJ and VP are combined, they yield greater improvements, see an enhancement of 9.67% in AP_{BEV}@0.7. These findings underscore the importance of both geometric alignment strategies in improving cross-platform detection performance.

KL Probabilistic Feature Alignment. PFA is designed to narrow the cross-platform gap during the Knowledge-Adaption stage. As shown in Tab. 5, incorporating PFA leads to significant performance gains on cross-platform tasks. By approximating the RoI features with probabilistic encoders and aligning their distributions using a KL divergence loss, PFA ensures that the target features are gradually pulled toward the source feature manifold. This alignment is crucial for reducing domain discrepancies and improving the overall detection accuracy on the target platform.

Geometry-Aware Transformation Descriptor. GTD is designed to capture global transformation cues on the source platform during the PA stage and correct global offsets on the target platform during the KA stage. As demonstrated in Tab. 5, incorporating GTD leads to significant performance gains. By learning geometric intrinsic that reflect sensor-specific characteristics such as sensor height and pitch distribution, GTD helps the network to predict and correct spatial misalignments between platforms.

In Appendix C.3, we provide a detailed analysis of the impact of varying the jitter angles introduced by RPJ across different platforms, where we investigate how different levels of simulated ego-motion affect detection performance.

5.4. Multi-Platform 3D Detection Benchmark

We establish a benchmark on Pi3DET to evaluate the cross-platform performance of 18 commonly-used 3D detectors by training all models on the vehicle set and testing them on vehicle, quadruped, and drone data (see Tab. 6 and Appendix C.2). Detectors are categorized into grid-based, point-based, and grid-point-based. Although grid-point-based methods excel on vehicles, their performance declines on quadruped and drone platforms, where point-based detectors achieve more balanced results, demonstrating enhanced viewpoint robustness. Furthermore, we apply our RPJ to the top-performing detectors on the vehicle platform. While this augmentation slightly degrades performance on vehicles due to the introduction of unseen noises, it significantly boosts results on the other two platforms. Overall, our findings underscore that effective geometry alignment and robust point-based architectures are crucial for developing unified 3D detectors across diverse platforms.

6. Conclusion

In this work, we introduced **Pi3DET**, a large-scale dataset for cross-platform 3D detection that includes diverse samples from vehicle, drone, and quadruped platforms. We proposed a novel adaptation approach that transfers the knowledge of vehicle detectors to other platforms by aligning geometric and feature representations. Extensive experiments show that our method is superior in both cross-platform and cross-dataset 3D object detection. We also establish a cross-platform benchmark on current 3D detectors and provide insights to improve resilience to platform variations, which benefits the research on unified 3D detection systems operating reliably across diverse autonomous platforms.

Acknowledgments

This work is under the programme DesCartes and is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. This work is also supported by the Apple Scholars in AI/ML Ph.D. Fellowship program.

The author, Ao Liang, gratefully acknowledges the financial support from the China Scholarship Council.

Additionally, the authors would like to sincerely thank the Program Chairs, Area Chairs, and Reviewers for the time and effort devoted during the review process.

References

[1] Rashid Abbasi, Ali Kashif Bashir, Hasan J Alyamani, Farhan Amin, Jaehyeok Doh, and Jianwen Chen. Lidar point cloud compression, processing and learning for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 24(1):962–979, 2022.

[2] Mina Alibeigi, William Ljungbergh, Adam Tonderski, Georg Hess, Adam Lilja, Carl Lindstrom, Daria Motorniuk, Junsheng Fu, Jenny Widahl, and Christoffer Petersson. Zenseact open dataset: A large-scale and diverse multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[3] Angelika Ando, Spyros Gidaris, Andrei Bursuc, Gilles Puy, Alexandre Boulch, and Renaud Marlet. Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5240–5250, 2023.

[4] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1090–1099, 2022.

[5] Hengwei Bian, Lingdong Kong, Haozhe Xie, Liang Pan, Yu Qiao, and Ziwei Liu. Dynamiccity: Large-scale 4d occupancy generation from dynamic scenes. In *International Conference on Learning Representations*, 2025.

[6] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11682–11692, 2020.

[7] Alexandre Boulch, Corentin Sautier, Björn Michele, Gilles Puy, and Renaud Marlet. Also: Automotive lidar self-supervision by occupancy estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13455–13465, 2023.

[8] 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.

[9] Kenneth Chaney, Fernando Cladera, Ziyun Wang, Anthony Bisulco, M Ani Hsieh, Christopher Korpela, Vijay Kumar, Camillo J Taylor, and Kostas Daniilidis. M3ed: Multi-robot, multi-sensor, multi-environment event dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop*, pages 4016–4023, 2023.

[10] Gyusam Chang, Wonseok Roh, Sujin Jang, Dongwook Lee, Daehyun Ji, Gyeongrok Oh, Jinsun Park, Jinkyu Kim, and Sangpil Kim. Cmda: Cross-modal and domain adversarial adaptation for lidar-based 3d object detection. In *AAAI Conference on Artificial Intelligence*, pages 972–980, 2024.

[11] Qi Chen, Lin Sun, Ernest Cheung, and Alan L Yuille. Every view counts: Cross-view consistency in 3d object detection with hybrid-cylindrical-spherical voxelization. *Advances in Neural Information Processing Systems*, 33:21224–21235, 2020.

[12] Runnan Chen, Youquan Liu, Lingdong Kong, Nenglun Chen, Xinge Zhu, Yuexin Ma, Tongliang Liu, and Wenping Wang. Towards label-free scene understanding by vision foundation models. In *Advances in Neural Information Processing Systems*, pages 75896–75910, 2023.

[13] Zhuoxiao Chen, Yadan Luo, Zheng Wang, Mahsa Baktashmotlagh, and Zi Huang. Revisiting domain-adaptive 3d object detection by reliable, diverse and class-balanced pseudo-labeling. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3714–3726, 2023.

[14] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. https://github.com/open-mmlab/mmdetection3d, 2020.

[15] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *AAAI Conference on Artificial Intelligence*, pages 1201–1209, 2021.

[16] Jinhao Deng, Wei Ye, Hai Wu, Xun Huang, Qiming Xia, Xin Li, Jin Fang, Wei Li, Chenglu Wen, and Cheng Wang. Cmd: A cross mechanism domain adaptation dataset for 3d object detection. In *European Conference on Computer Vision*, pages 219–236. Springer, 2024.

[17] Yinpeng Dong, Caixin Kang, Jinlai Zhang, Zijian Zhu, Yikai Wang, Xiao Yang, Hang Su, Xingxing Wei, and Jun Zhu. Benchmarking robustness of 3d object detection to common corruptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1022–1032, 2023.

[18] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Rangedet: In defense of range view for lidar-based 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2918–2927, 2021.

[19] Lue Fan, Yuxue Yang, Yiming Mao, Feng Wang, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Once detected,

never lost: Surpassing human performance in offline lidar based 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19820–19829, 2023.

[20] Di Feng, Xiao Wei, Lars Rosenbaum, Atsuto Maki, and Klaus Dietmayer. Deep active learning for efficient training of a lidar 3d object detector. In *IEEE Intelligent Vehicles Symposium*, pages 667–674, 2019.

[21] Felix Fent, Fabian Kuttenreich, Florian Ruch, Farija Rizwin, Stefan Juergens, Lorenz Lechermann, Christian Nissler, Andrea Perl, Ulrich Voll, Min Yan, et al. Man truckscenes: A multimodal dataset for autonomous trucking in diverse conditions. *Advances in Neural Information Processing Systems*, 37:62062–62082, 2025.

[22] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[23] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, et al. A2d2: Audi autonomous driving dataset. *arXiv preprint arXiv:2004.06320*, 2020.

[24] Ahmed Ghita, Bjørk Antoniussen, Walter Zimmer, Ross Greer, Christian Creß, Andreas Møgelmose, Mohan M Trivedi, and Alois C Knoll. Activeanno3d–an active learning framework for multi-modal 3d object detection. *arXiv preprint arXiv:2402.03235*, 2024.

[25] Xiaoshuai Hao, Mengchuan Wei, Yifan Yang, Haimei Zhao, Hui Zhang, Yi Zhou, Qiang Wang, Weiming Li, Lingdong Kong, and Jing Zhang. Is your hd map constructor reliable under sensor corruptions? In *Advances in Neural Information Processing Systems*, pages 22441–22482, 2024.

[26] Fangzhou Hong, Lingdong Kong, Hui Zhou, Xinge Zhu, Hongsheng Li, and Ziwei Liu. Unified 3d and 4d panoptic segmentation via dynamic shifting networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5): 3480–3495, 2024.

[27] Qianjiang Hu, Daizong Liu, and Wei Hu. Density-insensitive unsupervised domain adaption on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17556–17566, 2023.

[28] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(10):2702–2719, 2020.

[29] Yuzhe Ji, Yijie Chen, Liuqing Yang, Ding Rui, Meng Yang, and Xinhu Zheng. Vexkd: The versatile integration of cross-modal fusion and knowledge distillation for 3d perception. *Advances in Neural Information Processing Systems*, 37: 125608–125634, 2025.

[30] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[31] Lingdong Kong, Youquan Liu, Runnan Chen, Yuexin Ma, Xinge Zhu, Yikang Li, Yuenan Hou, Yu Qiao, and Ziwei Liu. Rethinking range view representation for lidar segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 228–240, 2023.

[32] Lingdong Kong, Youquan Liu, Xin Li, Runnan Chen, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Robo3d: Towards robust and reliable 3d perception against corruptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19994–20006, 2023.

[33] Lingdong Kong, Jiawei Ren, Liang Pan, and Ziwei Liu. Lasermix for semi-supervised lidar semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21705–21715, 2023.

[34] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit Cottereau, and Wei Tsang Ooi. Robodepth: Robust out-of-distribution depth estimation under corruptions. In *Advances in Neural Information Processing Systems*, pages 21298–21342, 2023.

[35] Lingtong Kong, Bo Li, Yike Xiong, Hao Zhang, Hong Gu, and Jinwei Chen. Safnet: Selective alignment fusion network for efficient hdr imaging. In *European Conference on Computer Vision*, pages 256–273. Springer, 2024.

[36] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Yaru Niu, Wei Tsang Ooi, Benoit R. Cottereau, Lai Xing Ng, Yuexin Ma, Wenwei Zhang, Liang Pan, Kai Chen, Ziwei Liu, Weichao Qiu, Wei Zhang, Xu Cao, Hao Lu, Ying-Cong Chen, Caixin Kang, Xinning Zhou, Chengyang Ying, Wentao Shang, Xingxing Wei, Yinpeng Dong, Bo Yang, Shengyin Jiang, Zeliang Ma, Dengyi Ji, Haiwen Li, Xingliang Huang, Yu Tian, Genghua Kou, Fan Jia, Yingfei Liu, Tiancai Wang, Ying Li, Xiaoshuai Hao, Yifan Yang, Hui Zhang, Mengchuan Wei, Yi Zhou, Haimei Zhao, Jing Zhang, Jinke Li, Xiao He, Xiaoqiang Cheng, Bingyang Zhang, Lirong Zhao, Dianlei Ding, Fangsheng Liu, Yixiang Yan, Hongming Wang, Nanfei Ye, Lun Luo, Yubo Tian, Yiwei Zuo, Zhe Cao, Yi Ren, Yunfan Li, Wenjie Liu, Xun Wu, Yifan Mao, Ming Li, Jian Liu, Jiayang Liu, Zihan Qin, Cunxi Chu, Jialei Xu, Wenbo Zhao, Junjun Jiang, Xianming Liu, Ziyan Wang, Chiwei Li, Shilong Li, Chendong Yuan, Songyue Yang, Wentao Liu, Peng Chen, Bin Zhou, Yubo Wang, Chi Zhang, Jianhang Sun, Hai Chen, Xiao Yang, Lizhong Wang, Dongyi Fu, Yongchun Lin, Huitong Yang, Haoang Li, Yadan Luo, Xianjing Cheng, and Yong Xu. The robodrive challenge: Drive anytime anywhere in any condition. *arXiv preprint arXiv:2405.08816*, 2024.

[37] Lingdong Kong, Dongyue Lu, Xiang Xu, Lai Xing Ng, Wei Tsang Ooi, and Benoit R. Cottereau. Eventfly: Event camera perception from ground to the sky. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1472–1484, 2025.

[38] Lingdong Kong, Xiang Xu, Jun Cen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. Calib3d: Calibrating model preferences for reliable 3d scene understanding. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1965–1978, 2025.

[39] Lingdong Kong, Xiang Xu, Jiawei Ren, Wenwei Zhang, Liang Pan, Kai Chen, Wei Tsang Ooi, and Ziwei Liu. Multi-modal data-efficient 3d scene understanding for autonomous

driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3748–3765, 2025.

[40] 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.

[41] Justin Lazarow, David Griffiths, Gefen Kohavi, Francisco Crespo, and Afshin Dehghan. Cubify anything: Scaling indoor 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22225–22233, 2025.

[42] Jae-Keun Lee, Jin-Hee Lee, Joohyun Lee, Soon Kwon, and Heechul Jung. Re-voxeldet: Rethinking neck and head architectures for high-performance voxel-based 3d detection. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7503–7512, 2024.

[43] Jinyu Li, Chenxu Luo, and Xiaodong Yang. Pillarnext: Rethinking network designs for 3d object detection in lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17567–17576, 2023.

[44] Li Li, Hubert PH Shum, and Toby P Breckon. Less is more: Reducing task and model complexity for 3d point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9361–9371, 2023.

[45] Li Li, Hubert PH Shum, and Toby P. Breckon. Rapid-seg: Range-aware pointwise distance distribution networks for 3d lidar segmentation. In *European Conference on Computer Vision*, pages 222–241. Springer, 2024.

[46] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. *Advances in Neural Information Processing Systems*, 35:18442–18455, 2022.

[47] Yanwei Li, Xiaojuan Qi, Yukang Chen, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Voxel field fusion for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1120–1129, 2022.

[48] Ye Li, Lingdong Kong, Hanjiang Hu, Xiaohao Xu, and Xiaonan Huang. Is your lidar placement optimized for 3d scene understanding? In *Advances in Neural Information Processing Systems*, pages 34980–35017, 2024.

[49] Zhenxin Li, Shiyi Lan, Jose M Alvarez, and Zuxuan Wu. Bevnext: Reviving dense bev frameworks for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20113–20123, 2024.

[50] Youquan Liu, Runnan Chen, Xin Li, Lingdong Kong, Yuchen Yang, Zhaoyang Xia, Yeqi Bai, Xinge Zhu, Yuexin Ma, Yikang Li, et al. Uniseg: A unified multi-modal lidar segmentation network and the openpcseg codebase. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21662–21673, 2023.

[51] Youquan Liu, Lingdong Kong, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. Segment any point cloud sequences by distilling vision foundation models. In *Advances in Neural Information Processing Systems*, pages 37193–37229, 2023.

[52] Youquan Liu, Lingdong Kong, Xiaoyang Wu, Runnan Chen, Xin Li, Liang Pan, Ziwei Liu, and Yuexin Ma. Multi-space alignments towards universal lidar segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14648–14661, 2024.

[53] Zhijian Liu, Alexander Amini, Sibo Zhu, Sertac Karaman, Song Han, and Daniela L Rus. Efficient and robust lidar-based end-to-end navigation. In *IEEE International Conference on Robotics and Automation*, pages 13247–13254, 2021.

[54] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, et al. One million scenes for autonomous driving: Once dataset. *arXiv preprint arXiv:2106.11037*, 2021.

[55] Jiageng Mao, Yujing Xue, Minzhe Niu, Haoyue Bai, Jiashi Feng, Xiaodan Liang, Hang Xu, and Chunjing Xu. Voxel transformer for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3164–3173, 2021.

[56] Jiageng Mao, Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. 3d object detection for autonomous driving: A comprehensive survey. *International Journal of Computer Vision*, 131(8):1909–1963, 2023.

[57] Tamás Matuszka, Iván Barton, Ádám Butykai, Péter Hajas, Dávid Kiss, Domonkos Kovács, Sándor Kunsági-Máté, Péter Lengyel, Gábor Németh, Levente Pető, et al. aimotive dataset: A multimodal dataset for robust autonomous driving with long-range perception. *arXiv preprint arXiv:2211.09445*, 2022.

[58] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Luc Van Gool, and Dengxin Dai. Weakly supervised 3d object detection from lidar point cloud. In *European Conference on Computer Vision*, pages 515–531. Springer, 2020.

[59] Björn Michele, Alexandre Boulch, Tuan-Hung Vu, Gilles Puy, Renaud Marlet, and Nicolas Courty. Train till you drop: Towards stable and robust source-free unsupervised 3d domain adaptation. In *European Conference on Computer Vision*, pages 1–19. Springer, 2024.

[60] A Tuan Nguyen, Toan Tran, Yarin Gal, Philip HS Torr, and Atılım Güneş Baydin. Kl guided domain adaptation. *arXiv preprint arXiv:2106.07780*, 2021.

[61] Gilles Puy, Alexandre Boulch, and Renaud Marlet. Using a waffle iron for automotive point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3379–3389, 2023.

[62] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems*, 30:5105–5114, 2017.

[63] Rui Qian, Xin Lai, and Xirong Li. 3d object detection for autonomous driving: A survey. *Pattern Recognition*, 130: 108796, 2022.

[64] Chao Qin, Haoyang Ye, Christian E Pranata, Jun Han, Shuyang Zhang, and Ming Liu. Lins: A lidar-inertial state estimator for robust and efficient navigation. In *IEEE International Conference on Robotics and Automation*, pages 8899–8906, 2020.

[65] Meytal Rapoport-Lavie and Dan Raviv. It's all around you: Range-guided cylindrical network for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2992–3001, 2021.

[66] Nermin Samet, Oriane Siméoni, Gilles Puy, Georgy Ponimatkin, Renaud Marlet, and Vincent Lepetit. You never get a second chance to make a good first impression: Seeding active learning for 3d semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18445–18457, 2023.

[67] Guangsheng Shi, Ruifeng Li, and Chao Ma. Pillarnet: Real-time and high-performance pillar-based 3d object detection. In *European Conference on Computer Vision*, pages 35–52. Springer, 2022.

[68] 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.

[69] 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.

[70] 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.

[71] 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*, 43(8):2647–2664, 2020.

[72] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *International Journal of Computer Vision*, 131(2):531–551, 2023.

[73] Nan Song, Tianyuan Jiang, and Jian Yao. Jpv-net: Joint point-voxel representations for accurate 3d object detection. In *AAAI Conference on Artificial Intelligence*, pages 2271–2279, 2022.

[74] Ziying Song, Lin Liu, Feiyang Jia, Yadan Luo, Caiyan Jia, Guoxin Zhang, Lei Yang, and Li Wang. Robustness-aware 3d object detection in autonomous driving: A review and outlook. *IEEE Transactions on Intelligent Transportation Systems*, 25(11):15407–15436, 2024.

[75] Ziying Song, Lei Yang, Shaoqing Xu, Lin Liu, Dongyang Xu, Caiyan Jia, Feiyang Jia, and Li Wang. Graphbev: Towards robust bev feature alignment for multi-modal 3d object detection. In *European Conference on Computer Vision*, pages 347–366. Springer, 2024.

[76] 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.

[77] OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020.

[78] Pengju Tian, Zhirui Wang, Peirui Cheng, Yuchao Wang, Zhechao Wang, Liangjin Zhao, Menglong Yan, Xue Yang, and Xian Sun. Ucdnet: Multi-uav collaborative 3d object detection network by reliable feature mapping. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–16, 2025.

[79] Zhi Tian, Xiangxiang Chu, Xiaoming Wang, Xiaolin Wei, and Chunhua Shen. Fully convolutional one-stage 3d object detection on lidar range images. *Advances in Neural Information Processing Systems*, 35:34899–34911, 2022.

[80] Darren Tsai, Julie Stephany Berrio, Mao Shan, Eduardo Nebot, and Stewart Worrall. Ms3d++: Ensemble of experts for multi-source unsupervised domain adaptation in 3d object detection. *IEEE Transactions on Intelligent Vehicles*, pages 1–16, 2024.

[81] Xuan Wang, Kaiqiang Li, and Abdellah Chehri. Multi-sensor fusion technology for 3d object detection in autonomous driving: A review. *IEEE Transactions on Intelligent Transportation Systems*, 25(2):1148–1165, 2024.

[82] 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.

[83] Yue Wang, Alireza Fathi, Abhijit Kundu, David A Ross, Caroline Pantofaru, Tom Funkhouser, and Justin Solomon. Pillar-based object detection for autonomous driving. In *European Conference on Computer Vision*, pages 18–34. Springer, 2020.

[84] Yingjie Wang, Jiajun Deng, Yuenan Hou, Yao Li, Yu Zhang, Jianmin Ji, Wanli Ouyang, and Yanyong Zhang. Club: cluster meets bev for lidar-based 3d object detection. *Advances in Neural Information Processing Systems*, 36:40438–40449, 2024.

[85] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. *arXiv preprint arXiv:2301.00493*, 2023.

[86] Maciej K Wozniak, Viktor Kårefjärd, Mattias Hansson, Marko Thiel, and Patric Jensfelt. Applying 3d object detection from self-driving cars to mobile robots: A survey and experiments. In *IEEE International Conference on Autonomous Robot Systems and Competitions*, pages 3–9, 2023.

[87] Aotian Wu, Pan He, Xiao Li, Ke Chen, Sanjay Ranka, and Anand Rangarajan. An efficient semi-automated scheme

for infrastructure lidar annotation. *IEEE Transactions on Intelligent Transportation Systems*, 25(7):8237–8247, 2024.

[88] Shaoyuan Xie, Lingdong Kong, Yuhao Dong, Chonghao Sima, Wenwei Zhang, Qi Alfred Chen, Ziwei Liu, and Liang Pan. Are vlms ready for autonomous driving? an empirical study from the reliability, data, and metric perspectives. *arXiv preprint arXiv:2501.04003*, 2025.

[89] Shaoyuan Xie, Lingdong Kong, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Benchmarking and improving bird's eye view perception robustness in autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3878–3894, 2025.

[90] Xiang Xu, Lingdong Kong, Hui Shuai, Wenwei Zhang, Liang Pan, Kai Chen, Ziwei Liu, and Qingshan Liu. 4d contrastive superflows are dense 3d representation learners. In *European Conference on Computer Vision*, pages 58–80. Springer, 2024.

[91] Xiang Xu, Lingdong Kong, Hui Shuai, and Qingshan Liu. Frnet: Frustum-range networks for scalable lidar segmentation. *IEEE Transactions on Image Processing*, 34:2173–2186, 2025.

[92] Xiang Xu, Lingdong Kong, Hui Shuai, Liang Pan, Ziwei Liu, and Qingshan Liu. Limoe: Mixture of lidar representation learners from automotive scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27368–27379, 2025.

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

[94] Anqi Joyce Yang, Sergio Casas, Nikita Dvornik, Sean Segal, Yuwen Xiong, Jordan Sir Kwang Hu, Carter Fang, and Raquel Urtasun. Labelformer: Object trajectory refinement for offboard perception from lidar point clouds. In *Conference on Robot Learning*, pages 3364–3383. PMLR, 2023.

[95] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7652–7660, 2018.

[96] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d: Self-training for unsupervised domain adaptation on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10368–10378, 2021.

[97] Jihan Yang, Shaoshuai Shi, Runyu Ding, Zhe Wang, and Xiaojuan Qi. Towards efficient 3d object detection with knowledge distillation. *Advances in Neural Information Processing Systems*, 35:21300–21313, 2022.

[98] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. 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.

[99] Jinrong Yang, Lin Song, Songtao Liu, Weixin Mao, Zeming Li, Xiaoping Li, Hongbin Sun, Jian Sun, and Nanning Zheng. Dbq-ssd: Dynamic ball query for efficient 3d object detection. *arXiv preprint arXiv:2207.10909*, 2022.

[100] 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.

[101] Maosheng Ye, Shuangjie Xu, and Tongyi Cao. Hvnet: Hybrid voxel network for lidar based 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1631–1640, 2020.

[102] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11784–11793, 2021.

[103] Jiakang Yuan, Bo Zhang, Xiangchao Yan, Tao Chen, Botian Shi, Yikang Li, and Yu Qiao. Bi3d: Bi-domain active learning for cross-domain 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15599–15608, 2023.

[104] Jiakang Yuan, Bo Zhang, Kaixiong Gong, Xiangyu Yue, Botian Shi, Yu Qiao, and Tao Chen. Reg-tta3d: Better regression makes better test-time adaptive 3d object detection. In *European Conference on Computer Vision*, pages 197–213. Springer, 2024.

[105] Bo Zhang, Jiakang Yuan, Botian Shi, Tao Chen, Yikang Li, and Yu Qiao. Uni3d: A unified baseline for multi-dataset 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9253–9262, 2023.

[106] Gang Zhang, Chen Junnan, Guohuan Gao, Jianmin Li, and Xiaolin Hu. Hednet: A hierarchical encoder-decoder network for 3d object detection in point clouds. *Advances in Neural Information Processing Systems*, 36:53076–53089, 2023.

[107] Guowen Zhang, Junsong Fan, Liyi Chen, Zhaoxiang Zhang, Zhen Lei, and Lei Zhang. General geometry-aware weakly supervised 3d object detection. In *European Conference on Computer Vision*, pages 290–309. Springer, 2024.

[108] Hongcheng Zhang, Liu Liang, Pengxin Zeng, Xiao Song, and Zhe Wang. Sparselif: High-performance sparse lidar-camera fusion for 3d object detection. In *European Conference on Computer Vision*, pages 109–128. Springer, 2024.

[109] Lunjun Zhang, Anqi Joyce Yang, Yuwen Xiong, Sergio Casas, Bin Yang, Mengye Ren, and Raquel Urtasun. Towards unsupervised object detection from lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9317–9328, 2023.

[110] Ruixiao Zhang, Yihong Wu, Juheon Lee, Xiaohao Cai, and Adam Prugel-Bennett. Detect closer surfaces that can be seen: New modeling and evaluation in cross-domain 3d object detection. In *European Conference on Artificial Intelligence*, pages 65–72, 2024.

[111] Weichen Zhang, Wen Li, and Dong Xu. Srdan: Scale-aware and range-aware domain adaptation network for cross-dataset 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6769–6779, 2021.

[112] Xinyu Zhang, Li Wang, Guoxin Zhang, Tianwei Lan, Haoming Zhang, Lijun Zhao, Jun Li, Lei Zhu, and Huaping Liu. Ri-fusion: 3d object detection using enhanced point features with range-image fusion for autonomous driving. *IEEE*

Transactions on Instrumentation and Measurement, 72:1–13, 2022.

[113] Yifan Zhang, Qingyong Hu, Guoquan Xu, Yanxin Ma, Jianwei Wan, and Yulan Guo. Not all points are equal: Learning highly efficient point-based detectors for 3d lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18953–18962, 2022.

[114] Zhanwei Zhang, Minghao Chen, Shuai Xiao, Liang Peng, Hengjia Li, Binbin Lin, Ping Li, Wenxiao Wang, Boxi Wu, and Deng Cai. Pseudo label refinery for unsupervised domain adaptation on cross-dataset 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15291–15300, 2024.

[115] Xin Zheng and Jianke Zhu. Efficient lidar odometry for autonomous driving. *IEEE Robotics and Automation Letters*, 6(4):8458–8465, 2021.

[116] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2023.

[117] Yijie Zhou, Likun Cai, Xianhui Cheng, Zhongxue Gan, Xiangyang Xue, and Wenchao Ding. Openannotate3d: Open-vocabulary auto-labeling system for multi-modal 3d data. In *IEEE International Conference on Robotics and Automation*, pages 9086–9092, 2024.

[118] Yijie Zhou, Likun Cai, Xianhui Cheng, Qiming Zhang, Xiangyang Xue, Wenchao Ding, and Jian Pu. Openannotate2: Multi-modal auto-annotating for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, pages 1–13, 2024.

[119] Xinge Zhu, Jiangmiao Pang, Ceyuan Yang, Jianping Shi, and Dahua Lin. Adapting object detectors via selective cross-domain alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 687–696, 2019.

[120] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Wei Li, Yuexin Ma, Hongsheng Li, Ruigang Yang, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar-based perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6807–6822, 2021.