Vision-Language Guidance for LiDAR-based Unsupervised 3D Object Detection

Christian Fruhwirth-Reisinger[1,2]
reisinger@tugraz.at
Wei Lin[3]
lin@ml.jku.at
Dušan Malić[1,2]
dusan.malic@tugraz.at
Horst Bischof[1]
bischof@tugraz.at
Horst Possegger[1,2]
possegger@tugraz.at

[1] Christian Doppler Laboratory for Embedded Machine Learning
[2] Institute of Computer Graphics and Vision
Graz University of Technology
[3] Institute for Machine Learning
Johannes Kepler University Linz

Abstract

Accurate 3D object detection in LiDAR point clouds is crucial for autonomous driving systems. To achieve state-of-the-art performance, the supervised training of detectors requires large amounts of human-annotated data, which is expensive to obtain and restricted to predefined object categories. To mitigate manual labeling efforts, recent unsupervised object detection approaches generate class-agnostic pseudo-labels for moving objects, subsequently serving as supervision signal to bootstrap a detector. Despite promising results, these approaches do not provide class labels or generalize well to static objects. Furthermore, they are mostly restricted to data containing multiple drives from the same scene or images from a precisely calibrated and synchronized camera setup.

To overcome these limitations, we propose a vision-language-guided unsupervised 3D detection approach that operates exclusively on LiDAR point clouds. We transfer CLIP knowledge to classify point clusters of static and moving objects, which we discover by exploiting the inherent spatio-temporal information of LiDAR point clouds for clustering, tracking, as well as box and label refinement. Our approach outperforms state-of-the-art unsupervised 3D object detectors on the Waymo Open Dataset (+23 AP_{3D}) and Argoverse 2 (+7.9 AP_{3D}) and provides class labels not solely based on object size assumptions, marking a significant advancement in the field. Code is available at https://github.com/chreisinger/ViLGOD.

1 Introduction

For safe navigation and efficient path planning, autonomous vehicles critically rely on 3D object detection, *i.e.* they must accurately identify the location, size, and type of objects (*e.g.* vehicle, cyclist, pedestrian) in the surrounding traffic environment. Recent 3D object detectors [12, 23, 40, 41, 42, 55] operate on the single modality of LiDAR point clouds [4,

14] and require supervised training with vast amounts of manually annotated data, which is time-consuming and cost-intensive to obtain at a sufficient quality level. Furthermore, despite their impressive performance, fully supervised 3D detectors lack the flexibility to cope with changing target data caused, for example, by different sensor setups [49] or unseen object classes. Re-annotation of this data would be necessary.

Annotation-efficient solutions, such as semi-supervised approaches [47, 59, 64] that require fewer manually labeled samples, or weakly-supervised methods employing techniques like click supervision [24, 31, 32] already target these issues. However, these methods still require human interaction in the form of either hand-labeled data or a human-in-the-loop setup. Recent unsupervised 3D object detection approaches [2, 29, 34, 50, 62, 66] exhibit impressive performance in automatic labeling, requiring no prior knowledge other than the movement assumption[1] and object size priors [29]. However, such methods suffer from two major restrictions: First, they focus on localization and bounding box estimation, but do not provide class labels. Second, due to the lack of category information, they can merely discover moving objects, thus missing the detection of static foreground objects, which must be obtained in another fashion, *e.g.* via repeated self-training [2, 66]. Furthermore, existing methods mostly require multiple drives from the same scene [29, 62], demanding high-precision mapping equipment, additional camera images that are precisely calibrated and synchronized [50], or precise scene flow estimates [2, 34]. However, an unsupervised, class-aware detector using LiDAR scans from a single drive would be preferable for cost and performance reasons.

This paper addresses these issues by proposing **Vi**sion-**L**anguage **G**uidence for Unsupervised LiDAR-based 3D **O**bject **D**etection - **ViLGOD**. Drawing inspiration from the recent success of vision-language foundation models, we employ CLIP [37] to classify objects from LiDAR point clouds in-the-wild. Specifically, we first propose spatio-temporal clustering over multiple frames incorporating motion cues to retrieve object proposals with high precision. After filtering common-sense background samples, we project the remaining object proposal clusters into 2D image space to generate smooth depth maps from multiple views. This simplification to the image domain allows us to obtain embeddings from the image path of the vision-language model. By matching the visual embeddings against pre-processed text embeddings from the text path, we can acquire corresponding classification scores for the object proposals, independent of their movement status, resulting in zero-shot detections.

The characteristics of LiDAR sensing, however, impose two unique challenges for projecting point clusters (see Fig. 1) to leverage 2D vision-language models, which are not handled by existing approaches that deal only with CAD point clouds [68, 73]: 1) LiDAR scans are 2.5D, *i.e.* only the surface visible to the sensor is measured. This incomplete reconstruction restricts the variety in view points that is needed by [68, 73] to fully exploit the 2D visual embeddings. 2) As the distance to the sensor increases, LiDAR scans become increasingly sparse, making it more difficult to identify the projected objects. To mitigate these problems, we exploit the fact that LiDAR recordings are sequential. We design a simple but effective tracking and propagation module that allows the generation of different temporal views of the same object. This module enables, on the one hand, a more robust classification of objects and, on the other hand, the propagation of class labels. We further exploit temporal object dependencies to create bounding boxes and propagate them within tracks.

Our contributions are four-fold: (1) ViLGOD is the first unsupervised but *class-aware* 3D object detection method for outdoor LiDAR point clouds that provides class labels not

[1] The scene contains moving objects recognizable by point traces throughout a sequence of LiDAR scans.

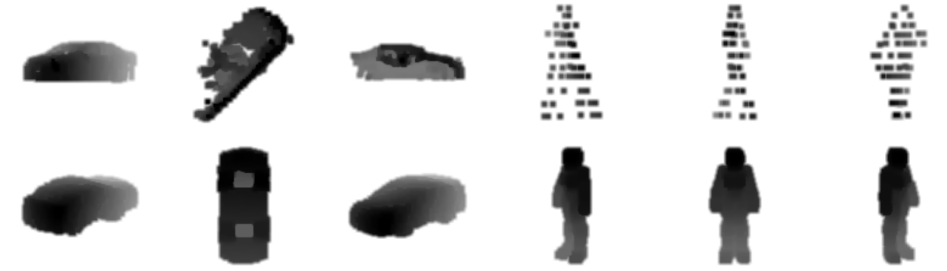

Figure 1: **Comparison of point projections.** We illustrate two projection examples from LiDAR clusters of the WOD [46] (top row) and sampled CAD models [52] (bottom row) evaluated in [73], in three different views. While points sampled from CAD models produce consistently good results, LiDAR point cluster projections are negatively affected by incomplete clusters through self-occlusion (car, top left) and sparsity (pedestrian, top right).

solely based on object size heuristics. (2) ViLGOD operates on the single modality of LiDAR point clouds, and requires neither multiple drives throughout the same scene nor additional camera images. (3) In addition to moving objects, ViLGOD also localizes static objects through CLIP classification, thus provides valuable pseudo labels without the need for repeated self-training cycles. (4) Lastly, our detailed evaluations on the Waymo Open Dataset and Argoverse 2 demonstrate that even in the class-agnostic setup, ViLGOD outperforms the current state-of-the-art unsupervised 3D object detectors.

2 Related Work

Fully supervised LiDAR-based 3D object detection. Current state-of-the-art 3D object detection networks [12, 41, 42, 55, 61, 72] typically rely on supervised learning methods and extensive quantities of human-annotated data [4, 30, 46] to achieve peak performance. Depending on how they handle the sparse and unordered LiDAR point cloud input, these methods can be broadly divided into grid-based [12, 23, 39, 55, 61, 71, 72], point-based [40, 43, 57, 70], and hybrid methods [41, 42, 61]. Instead of human annotations, all of these architectures can be trained with our high-quality pseudo-labels in a self-supervised fashion.

Label-efficient 3D object detection. Weakly-supervised methods learn from a limited amount of annotated data supplemented with auxiliary information, often by indirect supervision through image-level labels, coarse object locations, or scene-level annotations rather than 3D bounding boxes [19, 24, 31, 32, 47, 64]. Semi-supervised methods, on the other hand, leverage a small amount of labeled data in conjunction with a large volume of unlabeled data [5, 36, 56, 59]. Lastly, unsupervised methods strive to learn directly from the raw, unlabeled data, capitalizing on the inherent structure and distribution of the data and geometric properties. These methods frequently employ clustering techniques [2, 60], contrastive learning [17, 26, 27, 60] or masking [3, 22, 33, 53] to derive meaningful representations from the data. Although all these methods have shown the potential to reduce the need for exhaustive manual annotations, they still require supervision in any form.

Unsupervised 3D object detection. Early methods for 3D object detection in LiDAR data [15, 45, 67] introduced the generic pipeline – ground removal, clustering, bounding box fitting, and tracking – which is the foundation for all recent unsupervised methods to acquire initial detections [2, 10, 29, 34, 38, 50, 62, 66]. They then train deep neural networks with the initially generated pseudo labels to steadily improve the performance. However, existing methods are class-agnostic and thus lack the ability to find static objects in the initial label generation phase. Multiple rounds of self-training aim at mitigating this issue. For example, MODEST [62] and DRIFT [29] leverage datasets with multiple drives of the same scene to detect moving objects. Recently, OYSTER [66] leverages track consistency to find reliable pseudo-labels and introduces beam dropping for self-training to enhance detection quality in far ranges. Another line of work leverage 3D scene flow [2, 34] and camera images [50]. In contrast, our ViLGOD already localizes and classifies both static and moving objects without training and does not require multiple drives or additional sensor modalities.

CLIP for 3D understanding. Explorations of transferring CLIP knowledge for 3D understanding start with basic tasks of 3D object classification in zero-shot [18, 54, 65, 68, 73], few-shot [18, 65, 68, 73] and fully-supervised settings [18]. Recent approaches apply CLIP on non-trivial prediction tasks for 3D scene understanding in indoor environments [11, 16, 20, 28, 35, 58]. However, only few approaches transfer CLIP for outdoor scene understanding on LiDAR point clouds: Peng et al. [35] align 3D point cloud features with corresponding camera images via a distillation loss and evaluate on outdoor open-vocabulary semantic segmentation. Chen et al. [8] deploy an annotation-free semantic segmentation pipeline by enforcing consistency between point cloud features and the corresponding image features. All these works rely on multi-modal inputs, utilizing the 2D images as a bridge to connect the point cloud and the language modality. In contrast, our method operates exclusively on LiDAR point clouds and exploits relatively simple techniques [68, 73] for transferring CLIP knowledge to 3D data without requiring additional camera images. However, careful adjustments are needed to deal with clustering errors, increasing sparsity for distant objects, and incomplete objects that all arise from 2.5D LiDAR scans.

3 Vision-Language Guided 3D Object Detection

We aim to detect 3D objects solely from LiDAR point clouds, without training on labeled data. To realize this fully unsupervised, yet class-aware approach, we leverage the spatial and temporal cues inherently available in sequential LiDAR scans in combination with the powerful multimodal capabilities of recent vision-language models, as illustrated in Fig. 2. In particular, we first extract object proposals for both, moving and static objects, by spatio-temporal clustering, filtering and bounding box fitting (Section 3.1). To obtain category estimates for these discovered proposals, we then employ the vanilla image-text foundation model CLIP to classify depth map projections in a multi-view aggregation setup (Section 3.2). In the final step, we leverage the temporal knowledge to refine and propagate bounding boxes and class labels throughout the LiDAR sequence, resulting in improved predictions even for distant objects. We demonstrate that our training-free zero-shot detection results can be leveraged as pseudo labels for any supervised 3D detector (Section 3.3).

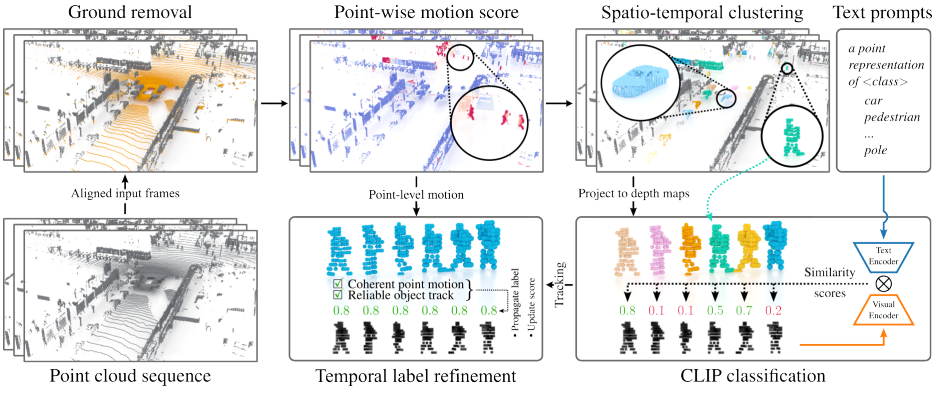

Figure 2: **ViLGOD overview.** After proposal generation (top row), we project 3D point clusters into 2D depth maps, subsequently fed to CLIP for zero-shot recognition. Objects close to the ego-vehicle result in smooth depth maps, which can be correctly classified with high certainty, *e.g.*, the light blue car. Distant objects, however, are more challenging and require temporal label refinement to improve classification results, *e.g.* the green pedestrian. We omit bounding box fitting and refinement visualizations to enhance clarity.

3.1 Unsupervised Object Discovery

We denote a LiDAR point cloud sequence with T frames as $\mathcal{P} = \{\mathcal{P}^t\}_{t=1}^{T}$, where the point cloud in the t-th frame is denoted as $\mathcal{P}^t = \{\boldsymbol{p}_i^t\}_{i=1}^{N^t}$, which represents a set of 3D points $\boldsymbol{p}_i^t \in \mathbb{R}^3$. In order to enable the unsupervised clustering of spatially related objects, we first remove ground points in each LiDAR scan. Specifically, we perform ground segmentation with Patchwork++ [25] that applies plane fitting on concentric zone patches, and outputs a set of ground points $\mathcal{G}^t$. We apply RANSAC [13] to fit a ground plane $\mathcal{O}_g$ with the ground points. To find moving objects, we derive the point-level motion information by identifying ephemeral points [1] in consecutive frames with the persistence point scores (PP-score) [62, 63]. The PP-score is computed as a measure of persistency for a 3D point by counting the number of its local neighbors across adjacent frames. Subsequently, we perform object discovery on non-ground points $\hat{\mathcal{P}}^t = \mathcal{P}^t \setminus \mathcal{G}^t$.

Proposal generation. Given that 3D semantic entities consist of spatially related points, we follow previous work [66] and apply HDBSCAN [6] for clustering. However, to reduce over-segmentation and noise, we first transform n frames of non-ground points $\hat{\mathcal{P}}^{t,\dots,t+n}$ into the reference frame $\hat{\mathcal{P}}^t$, sub-sample each frame by $1/n$ and concatenate the remaining points into a single frame $\hat{\mathcal{P}}_0^t$. In addition, we keep all points of frames where the PP-score indicates motion since moving points most likely belong to objects of interest. As input for clustering, we use the spatial position (x, y, z), the PP-score and the time difference Δt w.r.t. the reference frame for each point $\boldsymbol{p}_i^t$. The additional temporal input features help the clustering algorithm to distinguish 1) moving from static objects and 2) moving from moving objects which occupy the same space at different times (*e.g.* crossing pedestrian trajectories). As a result, we receive cluster segments for $\hat{\mathcal{P}}^t$. In a preliminary filtering step, we eliminate the most probable background objects. These are objects that either do not contain a minimum number of points, or are not located on the ground plane $\mathcal{O}_g$. This filtering results in a reduced set of

segments $\mathcal{S}^t = \{\mathcal{S}_i^t\}_{i=1}^{M^t}$. Afterwards, we apply simple L-Shape fitting [69] to obtain oriented 3D bounding boxes $\mathcal{B}^t = \{\boldsymbol{b}_i^t\}_{i=1}^{M^t}$ for all segments in $\mathcal{S}^t$. We denote the i-th bounding box as $\boldsymbol{b}_i^t = (b_x, b_y, b_z, b_l, b_w, b_h, b_\varphi)^\top \in \mathcal{B}^t$ where b_x, b_y, b_z are the center coordinates, and b_l, b_w, b_h, b_φ the length, width, height and orientation, respectively.

Temporal coherence. In order to distinguish between *moving* and *static* objects, we further exploit the temporal coherence in LiDAR sequences. First, we gather the point-level motion information within each segment $\mathcal{S}_i^t$ to determine the motion status of the corresponding bounding box $\boldsymbol{b}_i^t$. Specifically, if there is a percentile α of points within $\mathcal{S}_i^t$ which have PP-scores above a threshold δ, we consider $\boldsymbol{b}_i^t$ *static*, and *moving* otherwise. Second, we perform multi-target tracking on all bounding boxes throughout the LiDAR sequence with greedy assignment. Tracks are terminated when they have not been matched with any new incoming bounding box after a certain period, and each unassigned bounding box initiates a new track. We determine that a track and its bounding boxes are *static* if 1) all of its bounding boxes overlap with the largest box of the track and 2) none of the bounding boxes was considered *moving* according to the PP-scores. This distinction provides us with groups of moving and static objects, which we assign class labels in the following.

3.2 Vision-Language Guided Object Classification

CLIP preliminary. CLIP [57] is a large-scale vision-language model massively pre-trained via contrastive learning on 400M web image-text pairs, matching web images with their language descriptions. CLIP has a dual-encoder architecture comprising a visual encoder ϕ_v and a text encoder ϕ_t. Given an input query image x and a set of category text prompts $\mathcal{T} = \{t_c\}_{c=1}^{N_C}$, we denote the L2-normalized visual and text features as $z_v = \phi_v(x)$ and $\mathcal{Z}_t = \{\phi_t(t_c)\}_{c=1}^{N_C}$. The zero-shot image classification is performed by selecting the class prompt with the maximum similarity to the visual representation, *i.e.* $\hat{c} = \arg\max_c \phi_v(x)^\top \phi_t(t_c)$.

Transfer CLIP knowledge for 3D recognition. A vision-language foundation model like CLIP cannot be directly applied for recognition tasks on 3D LiDAR point clouds. Therefore, we project the zero-centered 3D points within each bounding box into natural-looking 2D depth maps to mitigate the modality gap between unordered sparse point clouds and grid-based dense image pixels. Specifically, we follow the shape projection proposed for dense CAD point clouds [73], which consists of voxelization, densification, and smoothing, to project a 3D object instance into realistic depth maps. To preserve the 3D information, we generate the depth maps from multiple views after rotating and tilting the points in each bounding box. Examples of projected depth maps are illustrated in Figure 1. We denote the number of views as K, and the set of K depth maps projected from points in the i-th bounding box as $\mathcal{X}_i = \{x_i^k\}_{k=1}^K$. For the category text prompts $t_c \in \mathcal{T}$, we use a 3D-specific prompt template *a point representation of* <*class*>. Then, the zero-shot class label for the k-th view of the i-th object instance is $\hat{c}_i^k = \arg\max_c \phi_v(x_i^k)^\top \phi_t(t_c)$.

Category text refinement. For an improved zero-shot classification with CLIP, we refine the original category names. Particularly, we replace the coarse category name *vehicle* with a set of refined classes such as *car*, *truck*, *bus*, and *van*. Similarly, we replace the abstract category *background* with instantiations of common non-traffic-participant objects such as *traffic light*, *traffic sign*, *fence*, *pole*, etc. Finally, we add relevant synonyms, *e.g. human*

body for *pedestrian*. The detailed text refinement strategy is elaborated in the supplemental material. After performing zero-shot classification on the expanded new category space, we merge the prediction results onto the fewer coarse classes in the original category space.

Multi-view label voting. To improve the prediction accuracy, [73] proposed aggregating the weighted class predictions of all K views projected from a CAD point cloud. Due to the different sensing characteristics, however, CLIP predictions for LiDAR-based projections vary largely depending on the view point. To mitigate this, we vote for the mostly predicted class label $\hat{c}_i$ within K views of an object i and set y_i to the mean prediction score of the $\hat{K}$ views with the same class label, *i.e.* $y_i = \sum_{\hat{k}} y_i^{\hat{k}} / \hat{K}$, where $y_i^{\hat{k}} = \phi_v(x_i^{\hat{k}})^\top \phi_t(t_{\hat{c}_i})$. If the number of votes are equal, we assign the class label with the maximum mean score.

Temporally-coherent label refinement. For unsupervised LiDAR segmentations, the projected depth maps suffer from degraded quality due to clustering errors, sparsity of distant objects, and incomplete objects of the 2.5D scans, as illustrated in the top-right of Figure 2. This leads to erroneous recognition results, especially on distant or incomplete objects. To compensate for this, we leverage the multi-target tracking results from Section 3.1 and apply a refinement strategy to propagate category labels and refined bounding box estimates throughout tracks of moving and static objects: For each track, we propagate the most confident CLIP label along the track if it is reliable w.r.t. the temporal progression of the track. A reliable class prediction means that it appears for at least 60 % of the track. We observed that CLIP prediction scores for smaller and less well-represented classes (*e.g.* pedestrians, cyclists, and background classes) are generally lower than those of vehicles. Thus, we propagate vehicle labels if the predicted score exceeds 0.5 and other labels if they exceed 0.3. Since *moving* objects are most certainly objects of interest in our traffic scenario, we aim to label all of these. If we can not obtain a reliable CLIP prediction, but the object is moving, we assign the class (vehicle, pedestrian, or cyclist) based on the observed object size.

Not only the correct classification of the object is important, but also its size and position. To reliably estimate the bounding boxes even for occluded or incompletely observed objects, we apply a temporal refinement: We first calculate the median box of the M box candidates which contain the most cluster points within a track. For *static* object tracks, we propagate this box estimate at the median position and obtain the orientation as the majority vote among the M boxes. For *moving* object tracks, we follow OYSTER [66] and propagate the box not with the center but along the tracking direction, aligning the box with the closest corner to the ego-vehicle.

3.3 Self-training

Our training-free unsupervised detection approach provides high quality pseudo-labels for the supervised training of any arbitrary 3D object detection architecture. We demonstrate this by leveraging the unsupervised detection results as pseudo ground truth in a supervised learning setting without bells and whistles. In particular, we train Centerpoint [61] with our pseudo-labels in a supervised and class-aware setup. We neither do multiple rounds of training and refinement [9, 29, 62] nor do we require additional augmentations [66].

	Dataset:	Waymo Open Dataset (WOD)						Argoverse 2	
	Motion Category:	Movable		Moving		Static		Movable	
	Average Precision:	BEV	3D	BEV	3D	BEV	3D	BEV	3D
Unsupervised	DBSCAN [6]	2.7	0.8	0.9	0.0	2.7	0.6	5.4	2.0
	RSF [9]	3.0	2.0	8.0	5.5	0.0	0.0	7.4	5.5
	SeMoLi [38] †	-	19.5	-	**57.5**	-	-	-	-
	LISO-CP [7]	29.2	21.1	27.2	20.4	20.8	14.0	-	-
	ViLGOD ‡	**36.3**	**32.3**	**28.0**	24.0	**32.7**	**31.1**	**25.1**	**22.5**
Self-train	OYSTER-CP [66]	21.7	8.4	15.1	6.2	17.6	5.6	38.1	15.0
	LISO-CP [7]	38.0	30.8	35.0	29.6	32.2	25.5	44.8	36.7
	ViLGOD-CP ‡	**56.4**	**53.8**	**54.0**	**52.1**	**46.5**	**45.4**	**46.4**	**44.6**

Table 1: **Class-agnostic evaluation** following the protocols of [7, 38] for WOD [46] (*i.e.* AP for BEV and 3D, difficulty level L2, IoU 0.4) and [7] for Argoverse 2 [51] (*i.e.* AP for BEV and 3D, IoU 0.3). Results for DBSCAN, RSF, OYSTER-CP taken from [7], †: Results from [38]. ‡: Method uses CLIP, unsupervised pre-trained on text-image pairs.

4 Experiments

To demonstrate the capabilities of our ViLGOD, we conduct experiments on two large-scale LiDAR datasets. First, we compare our method to state-of-the-art unsupervised object detectors in a *class-agnostic* setup, where we merge all predicted foreground objects into a single class. Second, we compare our *class-aware* results to class-agnostic approaches with assigned ground truth labels. Finally, we provide a detailed ablation on the separate components of our ViLGOD.

Datasets. We conduct our evaluations on the challenging Waymo Open Dataset (WOD) [46] and Argoverse 2 [51]. WOD contains 1000 publicly available sequences with approximately 200 frames each. It is separated into 798 training and 202 validation sequences. We follow the evaluation protocol of [7, 34], *i.e.* evaluating the area of 100m $\times$ 40m around the ego vehicle and reporting average precision (AP) with an intersection over union (IoU) threshold of 0.4 in 3D and BEV. Following [7, 38], objects that move faster than 1m/s are considered *moving*. Full-range evaluations and additional APH (Average precision and heading) scores are included in the supplemental material. Argoverse 2 contains 700 training and 150 validation sequences with approximately 150 frames each. We follow the evaluation protocol of [7] and evaluate the area of 100m $\times$ 100m around the ego vehicle and report AP with an IoU threshold of 0.3 in BEV. For the sake of comparability, we merge objects with the ability to move into the single class *movable*. In WOD, this affects all relevant object classes; in Argoverse 2, we exclude, for example, *Barrier*, *Traffic cone*, but also *bicycle* since the object is not moveable without a rider (*i.e.* the separate *cyclist* class).

Implementation Details. We build our detection pipeline on top of the OpenPCDet [48] framework (v0.6.0) and conduct all experiments with the provided base models. We utilize Centerpoint [61] for the supervised pseudo-label training. For these experiments, we follow the standard protocol of OpenPCDet and optimize with Adam [21] in a one-cycle policy [44] with a maximum learning rate of 0.003. However, we train for only 10 epochs on 50% of the

	Object Class:	Vehicle		Pedestrian		Cyclist	
	Average Precision:	BEV	3D	BEV	3D	BEV	3D
Unsupervised	*GT class labels*						
	DBSCAN [6]	18.4	4.8	0.2	0.0	0.1	0.0
	RSF [9]	10.9	7.4	0.0	0.0	0.2	0.0
	LISO-CP [2]	**60.7**	44.0	2.9	0.9	1.0	0.4
	Predicted class labels						
	ViLGOD	49.0	**44.8**	**16.8**	**14.1**	**7.6**	**7.4**
Self-train	*GT class labels*						
	OYSTER-CP [66]	56.2	20.4	0.0	0.0	0.0	0.0
	LISO-CP [2]	**69.5**	54.3	5.5	3.7	2.2	1.6
	Predicted class labels						
	ViLGOD-CP ‡	64.4	**62.4**	**38.8**	**35.9**	**7.5**	**7.5**

Table 2: **Class-aware evaluation on WOD** [46] following the protocol of [2] (*i.e.* AP scores for BEV and 3D, difficulty level L2, IoU 0.4). Results for DBSCAN, RSF, OYSTER-CP taken from [2]. ‡: Method uses CLIP, unsupervised pre-trained on text-image pairs.

training data and do not sample from a pseudo-label database. We ran all our experiments on 4 NVIDIA® RTX™ A6000 GPUs. Further implementation details and parameters can be found in the supplemental material.

Class-agnostic Results. The unsupervised 3D object detection results on the WOD validation set are shown in Table 1. The direct comparison among all unsupervised methods shows their object discovery capabilities: Our vision-language guidance allows ViLGOD to locate both moving *and* static objects in a single pass without requiring any re-training cycles. By leveraging the temporal coherence, we are also able to obtain accurate 3D bounding box estimates as indicated by the small gap between AP BEV to AP 3D. Thus, our ViLGOD excels in retrieving object candidates that can be used as pseudo labels to train a detector. To demonstrate this, we use these object proposals to train a Centerpoint [61] detector from scratch (denoted ViLGOD-CP). The results for this *self-training* in Table 1 show that our object proposals lead to a substantially improved detection performance, despite training Centerpoint for only 10 epochs (without augmenting samples from the pseudo-label database).

Class-aware Results. Table 2 shows the results for our zero-shot detections (*i.e.* class-aware predictions) in comparison to existing class-agnostic approaches with assigned ground truth (GT) labels. The consistently high AP 3D scores show that our ViLGOD provides accurate object proposals that are well suited for training a detector. In particular, our ViLGOD enables, for the first time, the training of a *class-aware* detector in an efficient manner: Without any manual human intervention and without time-consuming repeated self-training cycles. Notably, our approach leads to remarkable improvements in detecting the vulnerable road user classes (*i.e.* pedestrians and cyclists).

	Movable		Vehicle	Pedestrian	Cyclist
	BEV	3D	BEV	BEV	BEV
Baseline	19.9	18.3	32.0	4.6	1.3
+ Filtering	22.2	19.0	34.9	4.9	1.6
+ Corner alignment	25.1	20.5	35.9	8.0	1.4
+ Class label refinement	30.1	25.2	39.0	16.3	6.4
+ Bounding box refinement	36.3	32.3	49.0	16.8	7.6

Table 3: **Ablation study** following the protocols of [2, 34] on the WOD [46] (*i.e.* AP for BEV and 3D, difficulty level L2, IoU 0.4). The baseline includes spatio-temporal clustering, CLIP [37] classification, and L-shape bounding box fitting [69]. The ablations are split into filtering (see Section 3.1) and temporally coherent refinement steps, *i.e.*, corner alignment, class label refinement, bouncing box refinement (see Section 3.2).

Ablation Study. We conduct a detailed ablation study to show the contribution of each step of our approach. Table 3 lists the zero-shot detection results (pseudo-labels) on the WOD validation set. In addition to class-agnostic scores, we provide class-aware results, as our method provides zero-shot class-label predictions, allowing for better analysis.

The baseline represents a simple combination of spatio-temporal clustering, CLIP [37] classification, and L-shape box fitting [69]. As shown by the results, all steps contribute to the effectiveness of our unsupervised detection approach ViLGOD, allowing us to surpass the current state-of-the-art in unsupervised (class-agnostic) 3D object detection. The simple preliminary filtering not only increases the performance by 2.3 AP_{BEV} / 0.7 AP_{3D} but also speeds up the entire detection process by significantly reducing the number of remaining cluster segments. The corner-aligned box fitting [56] for *moving* objects particularly affects the detection quality of pedestrians. With an overall increase in performance by 2.9 AP_{BEV} / 1.5 AP_{3D} for *movable* objects, pedestrians benefit the most with an increase of 63.3% from 4.9 AP_{BEV} to 8.0 AP_{BEV}. The most significant step, however, is the class label refinement. It increases the performance for all object classes, doubling the performance on *pedestrians*, and enables the detection of at least some *cyclists*. Finally, the propagation of adjusted bounding boxes is especially advantageous for *vehicles*. Incorporating this final refinement step improves the overall performance by 6.2 AP_{BEV} / 7.1 AP_{3D} and thus represents the greatest overall improvement in absolute terms. Note that the largest improvement stems from the *vehicle* class, which represents the largest proportion of the classes in the WOD.

5 Conclusion

We proposed ViLGOD, the first fully unsupervised, yet class-aware 3D object detection method for LiDAR data. We combine the strong representation capabilities of vision-language models with unsupervised object discovery for both static and moving objects. This enables zero-shot detections, which result in reliable pseudo labels when propagated throughout LiDAR sequences. These pseudo-labels can be directly utilized to train a 3D object detector in a supervised manner, without the need for multiple self-training iterations. Our evaluations demonstrate the potential of this fully unsupervised data exploration strategy to significantly reduce the manual annotation costs needed to obtain sufficient amounts of data to train current state-of-the-art detectors.

Acknowledgements

The financial support by the Austrian Federal Ministry for Digital and Economic Affairs, the National Foundation for Research, Technology and Development and the Christian Doppler Research Association is gratefully acknowledged.

References

[1] Dan Barnes, Will Maddern, Geoffrey Pascoe, and Ingmar Posner. Driven to Distraction: Self-Supervised Distractor Learning for Robust Monocular Visual Odometry in Urban Environments. In *Proc. ICRA*, 2018.

[2] Stefan Baur, Frank Moosmann, and Andreas Geiger. LISO: Lidar-only Self-Supervised 3D Object Detection. *arXiv CoRR*, abs/2403.07071, 2024.

[3] Alexandre Boulch, Corentin Sautier, Björn Michele, Gilles Puy, and Renaud Marlet. ALSO: Automotive Lidar Self-supervision by Occupancy estimation. In *Proc. CVPR*, 2023.

[4] 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 *Proc. CVPR*, 2020.

[5] Benjamin Caine, Rebecca Roelofs, Vijay Vasudevan, Jiquan Ngiam, Yuning Chai, Z. Chen, and Jonathon Shlens. Pseudo-labeling for Scalable 3D Object Detection. *arXiv CoRR*, abs/2103.02093, 2021.

[6] Ricardo J. G. B. Campello, Davoud Moulavi, Arthur Zimek, and Jörg Sander. Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection. In *Proc. PAKDD*, 2013.

[7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. NeurIPS*, 2020.

[8] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. CLIP2Scene: Towards Label-efficient 3D Scene Understanding by CLIP. In *Proc. CVPR*, 2023.

[9] David Deng and Avideh Zakhor. RSF: Optimizing Rigid Scene Flow From 3D Point Clouds Without Labels. In *Proc. WACV*, 2023.

[10] Ayush Dewan, Tim Caselitz, Gian Diego Tipaldi, and Wolfram Burgard. Motion-based detection and tracking in 3D LiDAR scans. In *Proc. ICRA*, 2016.

[11] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. PLA: Language-Driven Open-Vocabulary 3D Scene Understanding. In *Proc. CVPR*, 2023.

[12] Lue Fan, Ziqi Pang, Tianyuan Zhang, Yu-Xiong Wang, Hang Zhao, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Embracing Single Stride 3D Object Detector with Sparse Transformer. In *Proc. CVPR*, 2022.

[13] Martin A. Fischler and Robert C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM*, 24(6):381–395, 1981.

[14] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Proc. CVPR*, 2012.

[15] M. Himmelsbach, Felix v. Hundelshausen, and H.-J. Wuensche. Fast segmentation of 3D point clouds for ground vehicles. In *Proc. IV*, 2010.

[16] Rui Huang, Xuran Pan, Henry Zheng, Haojun Jiang, Zhifeng Xie, Cheng Wu, Shiji Song, and Gao Huang. Joint representation learning for text and 3D point cloud. *Pattern Recognition*, 147(C):110086, 2024.

[17] Siyuan Huang, Yichen Xie, Song-Chun Zhu, and Yixin Zhu. Spatio-Temporal Self-Supervised Representation Learning for 3D Point Clouds. In *Proc. ICCV*, 2021.

[18] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proc. ICCV*, 2023.

[19] Xiangru Huang, Yue Wang, Vitor Campagnolo Guizilini, Rares Andrei Ambrus, Adrien Gaidon, and Justin Solomon. Representation Learning for Object Detection from Unlabeled Point Cloud Sequences. In *Proc. CoRL*, 2022.

[20] Xiaoshui Huang, Sheng Li, Wentao Qu, Tong He, Yifan Zuo, and Wanli Ouyang. EPCL: Frozen CLIP Transformer is An Efficient Point Cloud Encoder. In *Proc. AAAI*, 2024.

[21] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proc. ICLR*, 2015.

[22] Georg Krispel, David Schinagl, Christian Fruhwirth-Reisinger, Horst Possegger, and Horst Bischof. MAELi - Masked Autoencoder for Large-Scale LiDAR Point Clouds. In *Proc. WACV*, 2024.

[23] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. PointPillars: Fast Encoders for Object Detection from Point Clouds. In *Proc. CVPR*, 2019.

[24] Jungwook Lee, Sean Walsh, Ali Harakeh, and Steven L. Waslander. Leveraging Pre-Trained 3D Object Detection Models for Fast Ground Truth Generation. In *Proc. ITSC*, 2018.

[25] Seungjae Lee, Hyungtae Lim, and Hyun Myung. Patchwork++: Fast and robust ground segmentation solving partial under-segmentation using 3D point cloud. In *Proc. IROS*, 2022.

[26] Hanxue Liang, Chenhan Jiang, Dapeng Feng, Xin Chen, Hang Xu, Xiaodan Liang, Wei Zhang, Zhenguo Li, and Luc Van Gool. Exploring Geometry-Aware Contrast and Clustering Harmonization for Self-Supervised 3D Object Detection. In *Proc. ICCV*, 2021.

[27] Kangcheng Liu, Aoran Xiao, Xiaoqin Zhang, Shijian Lu, and Ling Shao. FAC: 3D Representation Learning via Foreground Aware Feature Contrast. In *Proc. CVPR*, 2023.

[28] Yuheng Lu, Chenfeng Xu, Xiaobao Wei, Xiaodong Xie, Masayoshi Tomizuka, Kurt Keutzer, and Shanghang Zhang. Open-Vocabulary Point-Cloud Object Detection without 3D Annotation. In *Proc. CVPR*, 2023.

[29] Katie Z Luo, Zhenzhen Liu, Xiangyu Chen, Yurong You, Sagie Benaim, Cheng Perng Phoo, Mark Campbell, Wen Sun, Bharath Hariharan, and Kilian Q Weinberger. Reward Finetuning for Faster and More Accurate Unsupervised Object Discovery. In *Proc. NeurIPS*, 2023.

[30] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, Jie Yu, and Chunjing Xu. One Million Scenes for Autonomous Driving: ONCE Dataset. In *Proc. NeurIPS*, 2021.

[31] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Luc Van Gool, and Dengxin Dai. Weakly Supervised 3D Object Detection from Lidar Point Cloud. In *Proc. ECCV*, 2020.

[32] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Yunde Jia, and Luc Van Gool. Towards a Weakly Supervised Framework for 3D Point Cloud Object Detection and Annotation. *IEEE TPAMI*, 44(8):4454–4468, 2022.

[33] Chen Min, Xinli Xu, Dawei Zhao, Liang Xiao, Yiming Nie, and Bin Dai. Voxel-MAE: Masked Autoencoders for Pre-training Large-scale Point Clouds. *arXiv CoRR*, abs/2206.09900, 2022.

[34] Mahyar Najibi, Jingwei Ji, Yin Zhou, Charles R. Qi, Xinchen Yan, Scott Ettinger, and Dragomir Anguelov. Motion Inspired Unsupervised Perception and Prediction in Autonomous Driving. In *Proc. CVPR*, 2022.

[35] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D Scene Understanding with Open Vocabularies. In *Proc. CVPR*, 2023.

[36] Charles R. Qi, Yin Zhou, Mahyar Najibi, Pei Sun, Khoa Vo, Boyang Deng, and Dragomir Anguelov. Offboard 3D Object Detection From Point Cloud Sequences. In *Proc. CVPR*, 2021.

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proc. ICML*, 2021.

[38] Jenny Seidenschwarz, Aljoša Ošep, Francesco Ferroni, Simon Lucey, and Laura Leal-Taixé. SeMoLi: What Moves Together Belongs Together. In *Proc. CVPR*, 2024.

[39] Guangsheng Shi, Ruifeng Li, and Chao Ma. PillarNet: High-Performance Pillar-based 3D Object Detection. In *Proc. ECCV*, 2022.

[40] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. PointRCNN: 3D Object Proposal Generation and Detection from Point Cloud. In *Proc. CVPR*, 2019.

[41] 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 *Proc. CVPR*, 2020.

[42] 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. *IJCV*, 131(6):531–551, 2023.

[43] Weijing Shi and Ragunathan Rajkumar. Point-GNN: Graph Neural Network for 3D Object Detection in a Point Cloud. In *Proc. CVPR*, 2020.

[44] Leslie N. Smith. A Disciplined Approach to Neural Network Hyper-Parameters: Part 1 – Learning Rate, Batch Size, Momentum, and Weight Decay. *arXiv CoRR*, abs/1803.09820, 2018.

[45] Muhammad Sualeh and Gon-Woo Kim. Dynamic Multi-LiDAR Based Multiple Object Detection and Tracking. *Sensors*, 19(10):1474, 2019.

[46] 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 *Proc. CVPR*, 2020.

[47] Yew Siang Tang and Gim Hee Lee. Transferable Semi-supervised 3D Object Detection from RGB-D Data. In *Proc. ICCV*, 2019.

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

[49] Yan Wang, Xiangyu Chen, Yurong You, Li Erran, Bharath Hariharan, Mark Campbell, Kilian Q. Weinberger, and Wei-Lun Chao. Train in germany, test in the usa: Making 3d object detectors generalize. In *Proc. CVPR*, 2020.

[50] Yuqi Wang, Yuntao Chen, and Zhaoxiang Zhang. 4D Unsupervised Object Discovery. In *Proc. NeurIPS*, 2022.

[51] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next Generation Datasets for Self-driving Perception and Forecasting. In *Proc. NeurIPS*, 2021.

[52] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proc. CVPR*, pages 1912–1920, 2015.

[53] Runsen Xu, Tai Wang, Wenwei Zhang, Runjian Chen, Jinkun Cao, Jiangmiao Pang, and Dahua Lin. MV-JAR: Masked Voxel Jigsaw and Reconstruction for LiDAR-Based Self-Supervised Pre-Training. In *Proc. CVPR*, 2023.

[54] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning unified representation of language, image and point cloud for 3d understanding. In *Proc. CVPR*, 2023.

[55] Yan Yan, Yuxing Mao, and Bo Li. SECOND: Sparsely Embedded Convolutional Detection. *Sensors*, 18(10):3337, 2018.

[56] Bin Yang, Min Bai, Ming Liang, Wenyuan Zeng, and Raquel Urtasun. Auto4D: Learning to Label 4D Objects from Sequential Point Clouds. *arXiv CoRR*, abs/2101.06586, 2021.

[57] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3DSSD: Point-based 3D Single Stage Object Detector. In *Proc. CVPR*, 2020.

[58] Yuan Yao, Yuanhan Zhang, Zhenfei Yin, Jiebo Luo, Wanli Ouyang, and Xiaoshui Huang. 3D Point Cloud Pre-training with Knowledge Distillation from 2D Images. *arXiv CoRR*, abs/2212.08974, 2022.

[59] Junbo Yin, Jin Fang, Dingfu Zhou, Liangjun Zhang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. Semi-supervised 3D Object Detection with Proficient Teachers. In *Proc. ECCV*, 2022.

[60] Junbo Yin, Dingfu Zhou, Liangjun Zhang, Jin Fang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. ProposalContrast: Unsupervised Pre-training for LiDAR-based 3D Object Detection. In *Proc. ECCV*, 2022.

[61] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3D Object Detection and Tracking. In *Proc. CVPR*, 2021.

[62] Yurong You, Katie Z Luo, Cheng Perng Phoo, Wei-Lun Chao, Wen Sun, Bharath Hariharan, Mark Campbell, and Kilian Q. Weinberger. Learning to Detect Mobile Objects from LiDAR Scans Without Labels. In *Proc. CVPR*, 2022.

[63] Yurong You, Cheng Perng Phoo, Katie Z Luo, Travis Zhang, Wei-Lun Chao, Bharath Hariharan, Mark Campbell, and Kilian Q. Weinberger. Unsupervised Adaptation from Repeated Traversals for Autonomous Driving. In *Proc. NeurIPS*, 2022.

[64] Sergey Zakharov, Wadim Kehl, Arjun Bhargava, and Adrien Gaidon. Autolabeling 3D Objects with Differentiable Rendering of SDF Shape Priors. In *Proc. CVPR*, 2020.

[65] Yihan Zeng, Chenhan Jiang, Jiageng Mao, Jianhua Han, Chaoqiang Ye, Qingqiu Huang, Dit-Yan Yeung, Zhen Yang, Xiaodan Liang, and Hang Xu. Contrastive Language-Image-Point Pretraining from Real-World Point Cloud Data. In *Proc. CVPR*, 2023.

[66] 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 *Proc. CVPR*, 2023.

[67] Quanshi Zhang, Xuan Song, Xiaowei Shao, Huijing Zhao, and Ryosuke Shibasaki. Unsupervised 3D category discovery and point labeling from a large urban environment. In *Proc. ICRA*, 2013.

[68] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. PointCLIP: Point Cloud Understanding by CLIP. In *Proc. CVPR*, 2022.

[69] Xiao Zhang, Wenda Xu, Chiyu Dong, and John M. Dolan. Efficient L-shape fitting for vehicle detection using laser scanners. In *Proc. IV*, 2017.

[70] 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 *Proc. CVPR*, 2022.

[71] Yin Zhou and Oncel Tuzel. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In *Proc. CVPR*, 2018.

[72] Zixiang Zhou, Xiangchen Zhao, Yu Wang, Panqu Wang, and Hassan Foroosh. CenterFormer: Center-based Transformer for 3D Object Detection. In *Proc. ECCV*, 2022.

[73] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyao Zeng, Shanghang Zhang, and Peng Gao. PointCLIP V2: Prompting CLIP and GPT for Powerful 3D Open-world Learning. In *Proc. CVPR*, 2023.