Shelf-Supervised Cross-Modal Pre-Training for 3D Object Detection

Mehar Khurana*, Neehar Peri*, James Hays, Deva Ramanan

Abstract. State-of-the-art 3D object detectors are often trained on massive labeled datasets. However, annotating 3D bounding boxes remains prohibitively expensive and time-consuming, particularly for LiDAR. Instead, recent works demonstrate that self-supervised pre-training with unlabeled data can improve detection accuracy with limited labels. Contemporary methods adapt best-practices for self-supervised learning from the image domain to point clouds (such as contrastive learning). However, publicly available 3D datasets are considerably smaller and less diverse than those used for image-based self-supervised learning, limiting their effectiveness. We do note, however, that such data is naturally collected in a multi-modal fashion, often paired with images. Rather than pre-training with only self-supervised objectives, we argue that it is better to bootstrap point cloud representations using image-based foundation models trained on internet-scale image data. Specifically, we propose a *shelf*-supervised approach (e.g. supervised with off-the-shelf image foundation models) for generating zero-shot 3D bounding boxes from paired RGB and LiDAR data. Pre-training 3D detectors with such pseudo-labels yields significantly better semi-supervised detection accuracy than prior self-supervised pretext tasks. Importantly, we show that image-based shelf-supervision is helpful for training LiDAR-only and multi-modal (RGB + LiDAR) detectors. We demonstrate the effectiveness of our approach on nuScenes, significantly improving over prior work in limited data settings. Our code is available on Github.

Keywords: Shelf-Supervised 3D Object Detection, Vision-Language Models, Autonomous Vehicles, Semi-Supervised Learning

1 Introduction

3D object detection is an integral component of the robot perception stack. To facilitate research in this space, the Autonomous Vehicle (AV) industry has released several large-scale 3D annotated multi-modal datasets [2,53,59]. However, LiDAR-based detectors pre-trained on nuScenes [2] cannot be easily applied to Argoverse [59] due to differences in sensor characteristics across hardware platforms. In practice, training 3D detectors for a new hardware platform requires re-annotating 3D bounding boxes at scale, which can be prohibitively expensive [56] and time consuming [6]. Instead, recent works [26, 52, 61, 62] demonstrate that

* Equal Contribution.

self-supervised pre-training with unlabeled sensor data can improve downstream detection accuracy with limited labels. Notably, AVs *already* capture large-scale unlabeled multi-modal data localized to HD maps during normal testing [58], motivating the exploration of self-supervised pre-training for outdoor scenes.

Status Quo. Self-supervised learning offers a scalable framework to learn from massive unlabeled datasets. Typically, self-supervised methods establish a pretext task that derives supervision directly from the data (e.g. occupancy prediction, contrastive learning, or masked auto-encoding). Once trained, these self-supervised representations can be adapted for downstream tasks by fine-tuning on a limited set of annotations. Given the success of image-based self-supervised learning [13, 14, 45], recent approaches revisit contrastive learning in the context of 3D sensor data [52,67]. However, existing 3D datasets are considerably smaller and less diverse than those used in image-based self-supervised learning. Instead of solely relying on self-supervised objectives for pre-training, we advocate for a more practical approach that embraces recent advances in multi-modal foundational models by bootstrapping point cloud representations with vision-language models (VLMs) trained on internet-scale data. Since off-the-shelf vision-language models *already* encode semantics and object priors, our work focuses on distilling these 2D foundational priors into 3D detectors.

Bootstrapping 3D Representations with 2D Priors. VLMs [23, 25, 28, 69] are trained on internet-scale data and show impressive zero-shot detection performance across many application domains [35]. Therefore, we posit that pre-training with noisy pseudo-labels distilled from vision-language models can outperform contrastive features learned from scratch. However, distilling image-based foundational knowledge to LiDAR remains an open challenge [30, 39, 41]. We address this challenge by explicitly "inflating" 2D instance masks to 3D bounding boxes via unlabelled multi-modal (RGB + LiDAR) training data, using a careful combination of LiDAR cues, HD maps, and shape priors [58]. This allows us to produce accurate 3D pseudo-labels which can then be used to pre-train uni-modal (LiDAR) or multi-modal (RGB + LiDAR) or 3D detectors. One significant advantage of our *shelf*-supervision over self-supervision is that it naturally produces a more aligned pretext task; instead of pre-training 3D detectors with contrastive learning, we learn from 3D pseudo-labels.

Contributions. We present three major contributions.

1. We propose Cross-Modal 3D Detection Distillation (CM3D), a zero-shot approach for generating 3D bounding boxes using off-the-shelf vision-language models.
2. We demonstrate that pre-training detectors with zero-shot 3D bounding box pseudo-labels achieves higher downstream detection accuracy than prior contrastive learning objectives.
3. We conduct extensive experiments to ablate our design choices and demonstrate that our simple approach achieves state-of-the-art semi-supervised 3D detection accuracy on the nuScenes benchmark.

2 Related Works

Unsupervised 3D Object Detection has gained significant interest in recent years as a way to auto-label large datasets without human annotators. Dewan et al. [11] introduced a model-free approach to detect and track the visible portions of objects by leveraging motion cues from LiDAR. More recently, Wong et al. [60] identified unknown instances through supervised segmentation and clustering [37]. However, both [11, 60] only reason about the visible extent of objects and are unable to generate amodal bounding boxes. Cen et. al. [4] used a supervised detector to generate bounding box proposals for unknown categories, and [54] exploited the correspondence between images and point clouds to generate object proposals. Similarly, Wilson et al. [58] employed an HD map and shape priors to "inflate" 2D detections to 3D cuboids. Utilizing priors such as surface normals [10,12,51] and motion cues [9,21,32,33] has been shown to improve unsupervised object detection. You et al. [65] integrated cues from temporal scene changes (e.g. point ephemerality from multiple traversals) to detect mobile objects [15, 16]. Najibi et al. [38] used scene flow and clustering to generate 3D bounding boxes for moving objects. However, this method does not assign semantic labels to cuboids and cannot detect static objects. Similarly, [1, 49] use scene flow to cluster points and generate pseudo-labels. In contrast, CM3D uses 2D VLM detectors to localize, "lift", and classify both static and moving objects.

Self-Supervised Learning for 3D Representations has been applied to many domains, including object-centric point clouds [43, 46, 47], indoor scenes [5,17,18,22,24,68], and outdoor environments [52,55,62]. Current techniques for self-supervised point cloud pre-training can be broadly characterized by their receptive field. Scene-based contrastive learning [40,61] (often used in the context of representation learning for indoor scenes) may not capture the necessary fine-grained details required for recognizing small objects. In contrast, voxel-based contrastive learning (often used in the context of representation learning for outdoor scenes) inherently struggles with a restricted receptive field, limiting its ability to encode larger structures. Region-based representations [44,62] attempt to strike a balance between coarse-grained scene representations and fine-grained voxel representations. However, most regions in outdoor scenes (e.g. ground plane and buildings) lack informative cues, making it difficult to learn robust features. More recently, multi-modal contrastive learning methods [26, 52] demonstrate that paired multi-modal sensors can provide complementary (self-)supervision for pre-training. PointContrast [61] established correspondences between different views of point clouds using a contrastive loss. DepthContrast [67] treated different depth maps as instances and discriminates between them. STRL [19] extracted invariant representations from temporally correlated frames in a 3D point cloud sequence. More recently, SimIPU [26] and CALICO [52] delve into multi-modal contrastive learning by utilizing paired RGB and LiDAR data. GeoMAE [55] frames point-cloud representation learning as a masked auto-encoding task, and uses geometric pretext tasks including occupancy, normals, and curvature estimation. In contrast to prior work, we argue that directly pre-training with noisy pseudo-labels from vision foundation models can be a more effective pre-training strategy for 3D object detection.

Leveraging 2D Foundational Models for 3D Perception is an active area of research. Sautier et al. [48] introduced SLidR, a 2D-to-3D representation distillation method aimed at cross-modal self-supervised learning on large-scale point clouds. Mahmoud et. al. [36] subsequently extended SLidR with a semantically-aware contrastive constraint and a class-balancing loss. SEAL [29] takes inspiration from SLidR and used SAM [20] to generate 3D class-agnostic point cloud segments for contrastive learning. More recently, SA3D [3] extends SAM [20] to segment 3D objects with NeRFs. Anything-3D [50] combines BLIP [23], a pre-trained 2D text-to-image diffusion model, with SAM for single-view conditioned 3D reconstruction. 3D-Box-Segment-Anything utilizes SAM with a pre-trained 3D detector, VoxelNeXt [7], for interactive 3D detection and labeling. SAM3D [66] repurposes SAM to directly segment objects from BEV point clouds. Most recently, UP-VL [39] distilled 2D vision-language features [45] into LiDAR point clouds to generate amodal cuboids. We use foundational models to generate class-specific instance segmentation masks and "inflate" these masks to 3D cuboids using LiDAR cues, HD maps, and shape priors.

3 Cross-Modal 3D Detection Distillation (CM3D)

Given a large unlabelled set of multi-modal (RGB + LiDAR) data, we combine 2D VLMs with 3D domain knowledge (given by 3D shape priors and map constraints) to generate 3D bounding box pseudo-labels for pre-training 3D detectors. We describe pseudo-label generation in Sec. 3.1 and our training pipeline in Sec. 3.3.

3.1 Generating 3D Bounding Box Pseudo-Labels

2D Mask Generation. The first stage of our pipeline requires producing accurate 2D instance masks for a fixed vocabulary of object categories. We prompt a VLM detector (e.g. Detic [69] or GroundingDINO [28]) with class names (e.g. `car, bus, truck`) to generate 2D box proposals. Though some VLM detectors (i.e., Detic) already produce instance segmentation masks, we found it more effective to prompt separate (foundational) models such as SAM [20] with the predicted 2D bounding boxes to generate high-quality segmentation masks.

3D Projection. Next, we project the LiDAR points onto the 2D image plane and group all points within each instance mask. To produce a final 3D bounding box pseudo-label, we need to generate a 3D centroid, 3D orientation, and cuboid dimensions (width, length, height). Our key technical approach, described below and inspired by [58], is to leverage 3D priors in the form of maps (e.g. car cuboids should typically be aligned with map geometry) and shape priors (e.g. fine-grained categories such as trucks tend to have canonical dimensions).

3D Cuboid Generation. We define an initial candidate 3D centroid to be the medoid of points within each mask, expressed as $m = \text{argmin}_{x \in P_M} \sum_{p \in P_M} ||p - x||_2$,

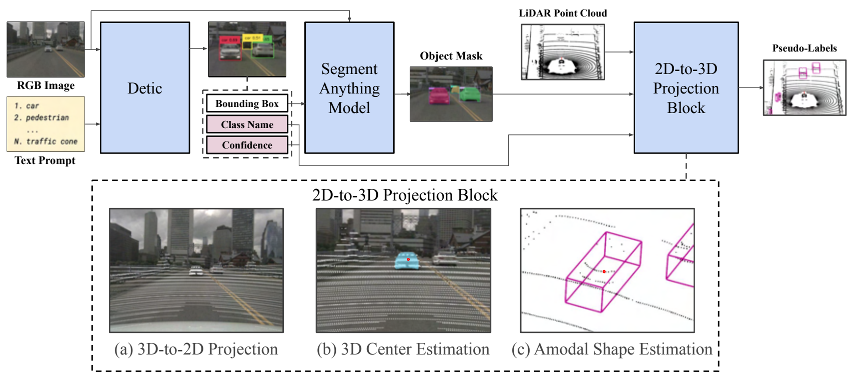


Fig. 1: CM3D Pseudo-Label Pipeline. First, we prompt a vision-language detector (e.g. Detic [69]) with a class name (e.g. `car`) to generate 2D box proposals. Next, we prompt SAM [20] with the predicted 2D bounding boxes to generate high-quality instance segmentation masks. We then generate an oriented 3D cuboid using the set of LiDAR points that project to a given 2D instance mask. Specifically, we define the *center* of the cuboid to be the medoid of the LiDAR points, the *dimensions* (length, width, height) to be a fixed shape prior (similar to an anchor box) as reported by ChatGPT when prompted with the class name, and the *orientation* to be aligned with lane geometry provided from an HD map.

where P_M is the set of LiDAR points that lie within the mask M. Note that this medoid tends to lie on the surface of the object visible to the LiDAR sensor rather than the true object centroid, which implies it will require refinement. To estimate a canonical width-height-length, we simply prompt an LLM (ChatGPT) with each class name and use the returned value for all instances. Although LLMs do not have access to our 3D training data, they have seen many descriptions of object shapes on the web and can provide reasonable prototypical 3D sizes of common objects. Lastly, we estimate box orientation for vehicles using lane geometry from an HD map and assign the direction of the nearest lane to each cuboid. We assign an orientation of zero degrees to all non-vehicles (e.g. `pedestrians`, `barriers` and `traffic cones`).

3D Refinement. Fig. 1 visualizes our overall pseudo-label generation pipeline. Because many components of our pipeline provide only rough 3D estimates (e.g., not all trucks have precisely the same 3D dimension), we find that 3D labels can be refined via self-training, which is discussed in the next section. We several strategies for improvement, including prompt engineering to improve VLM zero-shot 2D detections, LiDAR accumulation to improve medoid estimation, mask erosion to remove noisy LiDAR points near mask boundaries, medoid compensation to reduce object center estimation biases, and late-fusion of independent zero-shot 3D detectors to improve orientation and shape estimation. We find that imperfect pseudo-labels still provide an effective signal for pre-training.

3.2 CM3D Pseudo-Label Refinement

Many components in our CM3D pipeline rely on data-driven priors and can only provide rough 3D estimates. We describe several strategies for improving our 3D psuedo-labels below.

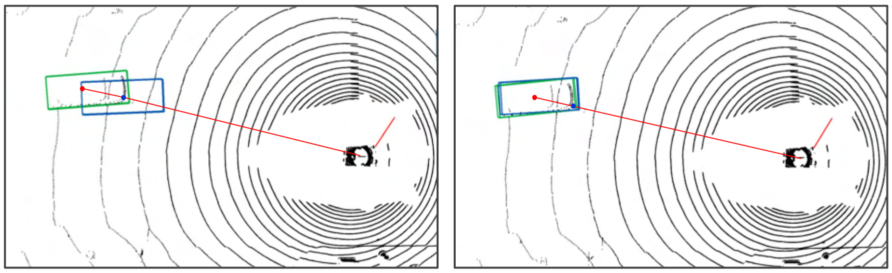
Prompt Engineering. Although VLMs show impressive zero-shot performance, they struggle when the prompted class is different from concepts encountered in their training data [35]. Following prior work [45], we prompt Detic with the standard nuScenes class names and their synonyms (e.g. `{human, adult, person, pedestrian}` for class `pedestrian`, and `{car, sedan, SUV}` for class `car`). Specifically, we use the nuScenes annotator guide to understand how nuScenes defines each class and generate synonyms accordingly. As shown in Fig. 1, Detic predicts class names and 2D bounding boxes for each image, along with confidence scores for each detection. We then perform non-maximum suppression (NMS) to remove redundant predictions across synonyms. Interestingly, Detic is unable to accurately detect classes like `barrier` even with carefully designed prompts, suggesting that prompting with synonyms is insufficient for certain ambiguously defined classes [35].

Mask Erosion. Although instance segmentations from SAM [20] are often accurate, we find that background LiDAR points near object boundaries can significantly impact medoid estimation [64]. We employ mask erosion to remove noisy LiDAR points near mask boundaries. These points are often unreliable because of depth discontinuities and minor errors in sensor calibration.

LiDAR Accumulation. LiDAR sweeps are notoriously sparse at range, making it difficult to distinguish foreground-vs-background. Therefore, the community has adopted the practice of accumulating multiple ego-motion compensated LiDAR sweeps when training 3D detectors [2]. We adopt the same practice in our pseudo-label generation pipeline for two reasons. First, accumulating multiple sweeps makes our medoid estimate more robust to outliers. Second, it biases the medoid towards the surface of the object, making medoid compensation (discussed next) more reliable.

Medoid Compensation. We find that predicted medoids are radially biased toward the ego vehicle because LiDAR points are denser on visible surfaces of objects as perceived from the ego vehicle. To compensate for this bias, we "push" all predictions radially away from the ego vehicle by a distance proportional to the object's size as follows:

Let $\boldsymbol{C}$ be the medoid of the object in the global coordinate frame, $\boldsymbol{E}$ be the center of the ego vehicle with respect to the global coordinate frame, and θ be the heading of the object in the global coordinate frame. We define a vector $\boldsymbol{CE} = \boldsymbol{E} - \boldsymbol{C}$, and α as the global slope angle of this vector, i.e., $\alpha = \arctan\left(\frac{\boldsymbol{CE}_y}{\boldsymbol{CE}_x}\right)$. As shown in Figure 2, we "push" the medoid back by distance $d = \min\left(\left|\frac{w}{2\sin(\alpha-\theta)}\right|, \left|\frac{l}{2\cos(\alpha-\theta)}\right|\right)$. Therefore, our new medoid is $\boldsymbol{C}'_x = \boldsymbol{C}_x - d \cdot \cos(\alpha)$ and $\boldsymbol{C}'_y = \boldsymbol{C}_y - d \cdot \sin(\alpha)$. We find that this simple geometric trick works surprisingly well in practice.



(a) Baseline (a) After Medoid Compensation

Fig. 2: Medoid Compensation. We find that all predicted medoids (shown in blue) tend to be radially biased toward the ego vehicle. This is because the LiDAR pointcloud only captures the visible surface of the car and not its full shape. To compensate for this bias, we "push" all predictions radially away, i.e., along the line connecting the center of the ego vehicle and the object medoid by a distance proportional to the object's size. The corrected medoid is shown in yellow. Empirically, we show that this geometric trick improves mAP by 1.6% and NDS by 2.1%, respectively.

Non-Maximum Suppression. nuScenes uses six RGB cameras to capture a 360° view of the environment, where neighboring cameras capture overlapping regions. Naively generating pseudo-labels across cameras can produce repeated detections for the same instance. Therefore, we perform non-maximum suppression (NMS) in the overlapping regions [57] after medoid compensation to remove duplicate detections.

Late Fusion. Recall, we define the *center* of each predicted cuboid to be the medoid of the LiDAR points within an instance mask, the *dimensions* (length, width, height) as reported by ChatGPT when prompted with the class name, and the *orientation* to be aligned with lane geometry provided by an HD map. Therefore, the quality of our pseudo-label generation pipeline is entirely dependent on the accuracy of our shape and orientation priors. In contrast, SAM3D [66] does not use priors for shape and orientation estimation, but rather directly estimates a rotated cuboid from a BEV perspective point cloud. Although SAM3D does not predict semantics, we find that its rotation and shape estimates are often more accurate than our priors. Therefore, we propose a simple late-fusion strategy to combine the best attributes of both zero-shot predictions.

For a given timestep, we greedily match our zero-shot predictions with SAM3D's predictions using 2D BEV IoU. Spatially matching CM3D and SAM3D predictions yields three categories of detections: matched detections, unmatched CM3D detections (without corresponding SAM3D detections), and unmatched SAM3D detections (without corresponding CM3D detections). We discard unmatched SAM3D detections since these are likely false positives because distinguishing foreground-vs-background with LiDAR-only cues is difficult [42].

Fusion of matched predictions from two independent detectors requires their scores to be comparable, Therefore, we use a class-agnostic implementation of score calibration as defined in [34]. Specifically, we scale the logits for SAM3D using a scaling factor τ (obtained by grid search on a val-set), i.e., confidence value $c = \sigma(logit/\tau)$. We construct a new set of fused detections by selecting the size and orientation from the more confident detection (SAM3D vs. CM3D) after calibration and use the semantic class predicted by CM3D (since CM3D can more accurately predict semantics with RGB images). Finally, we add all unmatched CM3D predictions to the set of fused predictions, unchanged.

3.3 Training with Pseudo-Labels

We use our generated pseudo-labels to pre-train 3D detectors that will be fine-tuned on a limited set of labeled data. We present a block diagram of our training pipeline in Fig 4. Notably, it is trivial to train *any* 3D detector with our pseudo-labels, making our proposed framework widely applicable. Further, although our pseudo-label generator requires paired RGB and LiDAR data for pseudo-label generation, we can easily train a LiDAR-only model like CenterPoint, which requires only LiDAR sweeps at inference.

In this paper, we evaluate CM3D by training two popular detectors, CenterPoint [63] and BEVFusion [31]. Following CALICO's [52] experiment setup, we train BEVFusion with the CenterHead and turn off class-balanced grouping and sampling. Although copy-paste augmentation is widely used when training 3D detectors, we find that performance degrades with our noisy pseudo-labels. We train the CenterHead following standard 3D detection losses. We use the sigmoid focal loss (for recognition) [27] and L1 regression loss (for localization). Concretely, our loss function is as follows: $L = L_{HM} + \lambda L_{REG}$, where $L_{HM} = \sum_{i=0}^{C} SigmoidFocalLoss(X_i, Y_i)$ and $L_{REG} = |X_{BOX} - Y_{BOX}|$, where X_i and Y_i are the i^{th} class' predicted and ground-truth heat maps, while X_{BOX} and Y_{BOX} are the predicted and pseudo-label box attributes.

Self Training. Although directly pre-training with our pseudo-labels significantly improves mAP, it also yields lower NDS compared to prior work. We posit that this is due to the errors in size and orientation estimation, which can both be attributed to our simplistic 3D priors. Prior work suggests that self-training can be effective for semi-supervised learning [39,65]; not only can it discover more objects [39,65], but it may also refine noisy box labels. We find that self-training significantly improves orientation and shape estimates by exploiting the regularity of multi-modal data; similar-looking objects will generate similar pseudo-labels due to the smoothness of the learned detection function. As shown in Fig 3, given K% annotated training data and (1-K%) unlabeled training data with CM3D pseudo-labels, we first pre-train a randomly initialized detector on (1-K%) pseudo-labels, fine-tune on K% annotated data, and use the resulting fine-tuned model to re-label the (1-K%) unlabeled training data. We iterative refine the (1-K%) pseudo-labels through multiple rounds of self-training. Importantly, we randomly initialize the detector after each round of self-training and pseudo-label generation to simplify training. We find that even one round of

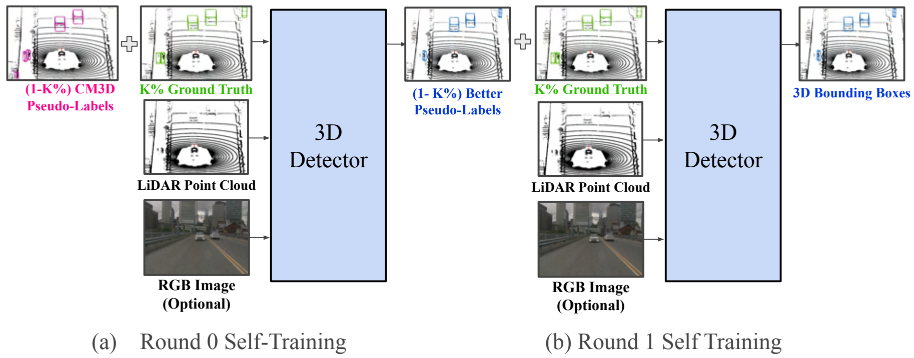


Fig. 3: Self-Training Procedure. We train a detector with ground truth annotations from K% of the train-set an pseudo-labels from (1-K%) of the train-set. We iteratively use the trained detector to predict better pseudo-labels on the (1-K%) subset until the accuracy of the predicted detections plateaus. We visualize one round of self-training above.

self-training significantly improves NDS when fine-tuning detectors with limited annotations. Additional rounds of self-training provide limited improvements.

4 Experiments

In this section, we conduct extensive experiments to evaluate our proposed approach on nuScenes [2]. We follow the suggested protocol in [52, 55, 62] and sample K% of the training data uniformly from the entire training set. We find that CM3D significantly improves over prior works in limited data settings (cf. Table 2).

Implementation Details. When generating pseudo-labels with CM3D, we use all Detic predictions with a confidence greater than 10% and use an IoU threshold of 0.75 for 2D NMS. We use a 3×3 kernel for mask erosion and accumulate the past 3 LiDAR frames for densification. Note that this is different from the usual 10-frame accumulation in nuScenes since 10-frame LiDAR pointcloud accumulation creates long “tails” for moving objects, and leads to inaccurate medoid predictions. For 3D NMS, we use class-specific distance-based thresholds defined in [63].

When training all detectors with pseudo-labels, we employ standard augmentation techniques (except for copy-paste augmentation). Following established practices, we aggregate the past 10 sweeps for LiDAR densification using the provided ego-vehicle poses. We train CenterPoint and BEVFusion using the same hyperparameters prescribed by their respective papers. For CenterPoint, we first train the detector for 20 epochs with CM3D pseudo-labels and fine-tune

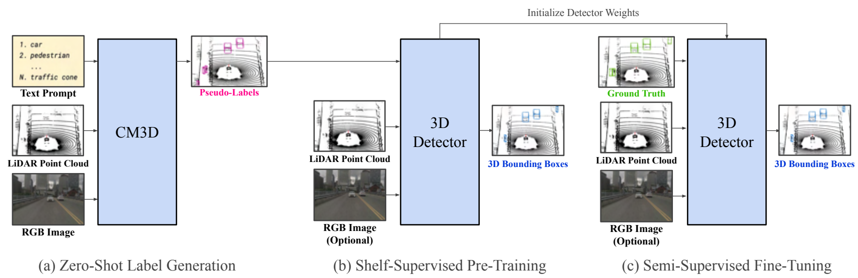


Fig. 4: Pre-Training and Fine-Tuning Pipeline (a) First, we generate zero-shot 3D bounding boxes with CM3D as described in Fig. 1. (b) Next, we pre-train our 3D detector using the generated pseudo-labels. Notably, this process is no different from training 3D detectors on ground-truth labels. (c) Lastly, we fine-tune the detector using a small amount of labeled data.

the detector for 20 epochs using the limited set of annotations. For BEVFusion, we first pre-train the LiDAR-only branch using CM3D pseudo-labels for 20 epochs and fine-tune the LiDAR-only branch for 20 epochs using the limited set of annotations. We train the fusion model (RGB + LiDAR) for 6 epochs using the limited amount of labeled training data. Lastly, we fine-tune all models using self-training. Specifically, we use the fine-tuned model to generate new pseudo-labels on the unlabeled portion of the train-set and retrain the detector on the entire train-set (including the limited set of ground truth labels and pseudo-labels) from scratch. We conduct all experiments on 8 RTX 3090 GPUs. Our code is available on Github.

4.1 Ablation Study of 3D Pseudo-Label Refinement.

We ablate our pseudo-label generation algorithm to determine how each component improves the baseline in Table 1. We find that medoid compensation has the greatest impact on pseudo-label quality, improving mAP by 1.6% and NDS by 2.1%. Tuning the Detic text prompts by including synonyms also improves results significantly. Finally, the 3D distance-based NMS helps remove duplicates present in the overlapping regions of the ring cameras and increases the mAP by 1%.

4.2 Comparison to State-of-the-Art

We first evaluate the zero-shot performance of CM3D to highlight the quality of our pseudo-labels. Next, we evaluate the performance of detectors trained on pseudo-labels. Lastly, we evaluate the performance of detectors pre-trained on CM3D psuedo-labels and fine-tuned on labeled data. Note that models are

Table 1: Ablation on Pseudo-Label Generation. We analyze the impact of each component over the baseline (cf. Fig 1). Importantly, we find that prompt engineering, medoid compensation, and non-maximum suppression have the greatest impact.

Method	mAP ↑	NDS ↑
Baseline	18.6	17.8
+ Prompt Engineering	19.7 (+1.1)	18.4 (+0.6)
+ LiDAR Accumulation	20.0 (+0.3)	18.6 (+0.2)
+ Mask Erosion	20.2 (+0.2)	19.1 (+0.5)
+ Medoid Compensation	21.8 (+1.6)	21.3 (+2.1)
+ Non-Maximal Suppression	22.8 (+1.0)	21.9 (+0.6)
+ Late Fusion	23.0 (+0.2)	22.1 (+0.2)

pre-trained with pseudo-labels from (1 - K%) of the training set and fine-tuned with ground truth labels from K% of the training set. We copy the results of prior work from CALICO and refer readers to [52] for further details. We present salient conclusions from Table 2 below.

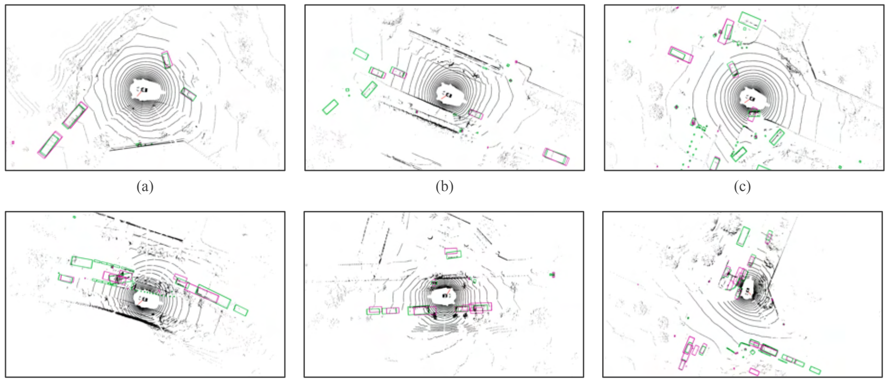
Evaluating Zero-Shot Performance. We evaluate the quality of CM3D's pseudo-labels (0% Training Data) and find that our method achieves 22.3% mAP and 22.1% NDS. We compare CM3D with SAM3D [66], a recently released zero-shot 3D detector. Notably, both CM3D and SAM3D use SAM [20] to group LIDAR points. However, CM3D groups LiDAR points on the image plane and SAM3D [66] groups LiDAR points in the 2D BEV plane. Our multi-modal zero-shot 3D detector significantly outperforms SAM3D's LiDAR-only approach by 20.7%. Importantly, SAM3D was initially designed as a class-agnostic zero-shot detector. We posit that SAM3D's poor performance can be attributed to nuScenes' diverse classes. Unlike many prior methods that generate class-agnostic pseudo-labels [1, 49], our approach is able to generate multi-class predictions because we leverage images during training.

Distilling Multi-Modal Priors into Uni-Modal Models. Although CM3D requires paired RGB images and LiDAR sweeps for pseudo-label generation, we can easily train RGB-only or LiDAR-only student models (e.g. CenterPoint + CM3D). Importantly, these student models don't require paired RGB-LiDAR data at inference (and are therefore compared against other models that only use LiDAR at inference). However, we find that the CenterPoint + CM3D (0% Training Data) student model performs worse than the pseudo-label generator (CM3D). This can be attributed to learning with noisy labels and not having access to the same sensor data as the teacher. However, we find that using multi-modal cues for both training and testing yields higher results. Specifically, BEVFusion + CM3D (0% Training Data) achieves 3.9 mAP and 1.9 NDS better performance than CenterPoint + CM3D (0% Training Data).

Significant Improvements in Low-Data Setting. Pre-training CenterPoint with CM3D pseudo-labels and fine-tuning on labeled data improves over the prior art of PRC [52] by 8.1 mAP and 2.8 NDS with 5% labeled data. Sim-

Table 2: Semi-Supervised 3D Detection on NuScenes. In terms of zero-shot accuracy, pseudo-labels from CM3D outperforms prior art (SAM3D) by 20.7% NDS. With 5% supervised training data, pre-training CenterPoint with CM3D pseudo-labels improves over the prior art of PRC [52] by 8.1 mAP / 2.8 NDS. When comparing to Camera-LiDAR methods, pre-training BEVFusion with CM3D pseudo-labels outperforms prior art (CALICO) by 8.6 mAP / 4.6 NDS. We highlight the best LiDAR-only (L) results in blue, and the best Camera-LiDAR (L+C) results in red.

Training Data	Method	Modality		mAP ↑	NDS ↑
		Train	Test		
0% (Unsupervised)	SAM3D [66]	L	L	1.7	2.4
	CM3D (Ours)	L + C	L + C	23.0	22.1
	CenterPoint [63] + CM3D (Ours)	L + C	L	16.7	21.4
	BEVFusion [31] + CM3D (Ours)	L + C	L + C	20.6	23.3
5%	CenterPoint [63] + Rand. Init.	L	L	33.1	37.4
	PointContrast [61]	L	L	36.7	43.0
	ProposalContrast [62]	L	L	37.0	43.1
	PRC [52]	L	L	38.2	46.0
	CenterPoint [63] + CM3D (Ours)	L + C	L	46.3	48.8
	BEVFusion [31] + Rand. Init.	L + C	L + C	39.0	43.7
	SimIPU [26]	L + C	L + C	39.1	45.8
	PRC [52] + BEVDistill [8]	L + C	L + C	41.0	47.5
	CALICO [52]	L + C	L + C	41.7	47.9
	BEVFusion [31] + CM3D (Ours)	L + C	L + C	51.3	52.5
10%	CenterPoint [63] + Rand. Init.	L	L	41.1	48.0
	PointContrast [61]	L	L	42.3	51.2
	ProposalContrast [62]	L	L	42.1	51.1
	PRC [52]	L	L	44.1	53.1
	CenterPoint [63] + CM3D (Ours)	L + C	L	51.0	56.3
	BEVFusion [31] + Rand. Init.	L + C	L + C	46.2	51.9
	SimIPU [26]	L + C	L + C	47.5	52.4
	PRC [52] + BEVDistill [8]	L + C	L + C	49.7	53.6
	CALICO [52]	L + C	L + C	50.0	53.9
	BEVFusion [31] + CM3D (Ours)	L + C	L + C	53.3	56.5
20%	CenterPoint [63]+ Rand. Init.	L	L	47.1	56.7
	PointContrast [61]	L	L	48.3	57.5
	ProposalContrast [62]	L	L	48.0	57.4
	PRC [52]	L	L	49.5	58.9
	CenterPoint [63] + CM3D (Ours)	L + C	L	54.5	59.0
	BEVFusion [31] + Rand. Init.	L + C	L + C	53.1	58.9
	SimIPU [26]	L + C	L + C	53.4	58.9
	PRC [52] + BEVDistill [8]	L + C	L + C	54.4	59.2
	CALICO [52]	L + C	L + C	54.8	59.5
	BEVFusion [31] + CM3D (Ours)	L + C	L + C	56.2	60.2
50%	CenterPoint [63] (Rand. Init.)	L	L	53.2	61.0
	PointContrast [61]	L	L	53.5	61.4
	ProposalContrast [62]	L	L	53.1	61.0
	PRC [52]	L	L	54.1	62.1
	CenterPoint [63] + CM3D (Ours)	L + C	L	56.9	61.0
	BEVFusion [31] (Rand. Init.)	L + C	L + C	58.5	61.8
	SimIPU [26]	L + C	L + C	58.6	62.0
	PRC [52] + BEVDistill [8]	L + C	L + C	59.6	62.3
	CALICO [52]	L + C	L + C	60.1	62.7
	BEVFusion [31] + CM3D (Ours)	L + C	L + C	59.7	62.5



(a) (b) (c) (d) (e) (f)

Fig. 5: Qualitative Results of Pseudo-Labels. We visualize pseudo-labels (pink) and ground-truth labels (green) across all 10 object classes on the nuScenes val-set. In (**a**), our pseudo-labels accurately estimate location, cuboid size, and orientation, demonstrating the general effectiveness of medoid compensation and map-based orientation estimation. In (**b**), we find that CM3D often misses heavily-occluded objects. This is unsurprising because our method relies on accurate RGB-based detections, which often fail with heavy occlusions. In (**c**), our map-based orientation estimation fails when the predicted object is not oriented in the direction of any lane. For example, the incorrect orientation of the `car` turning into the intersection (not aligned to any nearby lanes) illustrates the limitations of our approach. In both (**d**) and (**f**), we are unable to label several barriers. We attribute these missed detections to the ambiguity of the class name `barrier`. Notably, a `barrier` in nuScenes may not be the same as `barrier` as defined in internet pre-training data [35]. In (**d**), (**e**), and (**f**), we produce duplicate boxes for the same instances, indicating a failure of NMS.

ilarly, pre-training BEVFusion with CM3D pseudo-labels outperforms prior art (CALICO) by 8.6 mAP and 4.6 NDS. This suggests that aligning your pre-training and fine-tuning tasks can yield better performance.

Qualitative Results. We visualize the output of our pseudo-label generation approach in Fig. 5. Although our method generates reasonable predictions in many cases, we find that our method fails in cases of occlusions (where there is no 3D information) and in cases where the VLM predicts a false positive detection with high confidence. See Fig 5 for detailed analysis.

5 Conclusion

In this paper, we propose Cross-Modal 3D Detection Distillation (CM3D), a zero-shot approach for generating 3D bounding boxes using vision-language models. We demonstrate that pre-training detectors with zero-shot 3D bounding box pseudo-labels achieves higher downstream semi-supervised detection than prior

Table 3: nuScenes BEV Map Segmentation Results. Although BEVFusion + CM3D is pre-trained on noisy 3D bounding boxes, it performs better on BEV map segmentation than random initialization! However, our method performs worse than other state-of-the-art methods. This suggests that aligning pre-training and fine-tuning tasks improves semi-supervised performance for the target task (cf. Table 2) at the cost of generalizing to other downstream tasks.

Training Data	Method	mIOU ↑
5%	BEVFusion [31] + Rand. Init.	36.3
	SimIPU [26]	38.5
	PRC [52] + BEVDistill [8]	40.9
	CALICO [52]	**42.0**
	BEVFusion [31] + CM3D (Ours)	38.5
10%	BEVFusion [31] + Rand. Init.	43.8
	SimIPU [26]	45.1
	PRC [52] + BEVDistill [8]	46.4
	CALICO [52]	**47.3**
	BEVFusion [31] + CM3D (Ours)	44.4

contrastive learning objectives. However, we find that our proposed pre-training strategy limits generalization to arbitrary downstream tasks.

Aligning Pre-Training and Fine-Tuning Task Limits Generalizability. Contrastive learning has been widely adopted for self-supervised learning because it encodes task-agnostic representations that can be used for diverse downstream applications. In contrast, our approach uses prior knowledge about the downstream task to design a suitable pretext task. While this works well when the pre-training and fine-tuning tasks are well aligned, it does not provide a significant improvement when this is not the case. We evaluate the generalization of BEVFusion pre-trained on CM3D pseudo-labels for BEV map segmentation. Surprisingly, our pre-training strategy performs better at BEV map segmentation than random initialization! However, our method performs slightly worse than other state-of-the-art self-supervised methods methods. This suggests that aligning our pre-training and fine-tuning task can provide significant improvements (cf. Table 2) at the cost of generalizability to diverse tasks. Future work should explore different shelf-supervised pretext tasks to improve semi-supervised accuracy for diverse tasks in low data settings.

6 Acknowledgements

We would like to thank Jenny Seidenschwarz for her feedback on early drafts. This work was supported in part by NSF GRFP (Grant No. DGE2140739).

References

1. Baur, S., Moosmann, F., Geiger, A.: Liso: Lidar-only self-supervised 3d object detection. arXiv preprint arXiv:2403.07071 (2024)
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020)
3. Cen, J., Zhou, Z., Fang, J., Shen, W., Xie, L., Jiang, D., Zhang, X., Tian, Q., et al.: Segment anything in 3d with nerfs. Advances in Neural Information Processing Systems **36** (2024)
4. Cen, J., Yun, P., Cai, J., Wang, M.Y., Liu, M.: Open-set 3d object detection. In: 2021 International Conference on 3D Vision (3DV). pp. 869–878. IEEE (2021)
5. Chen, R., Liu, Y., Kong, L., Zhu, X., Ma, Y., Li, Y., Hou, Y., Qiao, Y., Wang, W.: Clip2scene: Towards label-efficient 3d scene understanding by clip. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7020–7030 (2023)
6. Chen, W., Edgley, A., Hota, R., Liu, J., Schwartz, E., Yizar, A., Peri, N., Purtilo, J.: Rebound: An open-source 3d bounding box annotation tool for active learning. arXiv preprint arXiv:2303.06250 (2023)
7. Chen, Y., Liu, J., Zhang, X., Qi, X., Jia, J.: Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21674–21683 (2023)
8. Chen, Z., Li, Z., Zhang, S., Fang, L., Jiang, Q., Zhao, F.: Bevdistill: Cross-modal bev distillation for multi-view 3d object detection. arXiv preprint arXiv:2211.09386 (2022)
9. Choudhary, S., Trevor, A.J., Christensen, H.I., Dellaert, F.: Slam with object discovery, modeling and mapping. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 1018–1025. IEEE (2014)
10. Collet, A., Srinivasay, S.S., Hebert, M.: Structure discovery in multi-modal data: a region-based approach. In: 2011 IEEE International Conference on Robotics and Automation. pp. 5695–5702. IEEE (2011)
11. Dewan, A., Caselitz, T., Tipaldi, G.D., Burgard, W.: Motion-based detection and tracking in 3d lidar scans. In: 2016 IEEE international conference on robotics and automation (ICRA). pp. 4508–4513. IEEE (2016)
12. García, G.M., Potapova, E., Werner, T., Zillich, M., Vincze, M., Frintrop, S.: Saliency-based object discovery on rgb-d data with a late-fusion approach. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 1866–1873. IEEE (2015)
13. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al.: Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems **33**, 21271–21284 (2020)
14. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9729–9738 (2020)
15. Herbst, E., Henry, P., Ren, X., Fox, D.: Toward object discovery and modeling via 3-d scene comparison. In: 2011 IEEE international conference on robotics and automation. pp. 2623–2629. IEEE (2011)

16. Herbst, E., Ren, X., Fox, D.: Rgb-d object discovery via multi-scene analysis. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 4850–4856. IEEE (2011)
17. Hou, J., Graham, B., Nießner, M., Xie, S.: Exploring data-efficient 3d scene understanding with contrastive scene contexts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15587–15597 (2021)
18. Hou, J., Xie, S., Graham, B., Dai, A., Nießner, M.: Pri3d: Can 3d priors help 2d representation learning? In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5693–5702 (2021)
19. Huang, S., Xie, Y., Zhu, S.C., Zhu, Y.: Spatio-temporal self-supervised representation learning for 3d point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6535–6545 (2021)
20. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. arXiv preprint arXiv:2304.02643 (2023)
21. Kochanov, D., Ošep, A., Stückler, J., Leibe, B.: Scene flow propagation for semantic mapping and object discovery in dynamic street scenes. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1785–1792. IEEE (2016)
22. Lal, S., Prabhudesai, M., Mediratta, I., Harley, A.W., Fragkiadaki, K.: Coconets: Continuous contrastive 3d scene representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12487–12496 (2021)
23. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: International Conference on Machine Learning. pp. 12888–12900. PMLR (2022)
24. Li, L., Heizmann, M.: A closer look at invariances in self-supervised pre-training for 3d vision. In: European Conference on Computer Vision. pp. 656–673. Springer (2022)
25. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10965–10975 (2022)
26. Li, Z., Chen, Z., Li, A., Fang, L., Jiang, Q., Liu, X., Jiang, J., Zhou, B., Zhao, H.: Simipu: Simple 2d image and 3d point cloud unsupervised pre-training for spatial-aware visual representations. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 1500–1508 (2022)
27. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
28. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023)
29. Liu, Y., Kong, L., Cen, J., Chen, R., Zhang, W., Pan, L., Chen, K., Liu, Z.: Segment any point cloud sequences by distilling vision foundation models. arXiv preprint arXiv:2306.09347 (2023)
30. Liu, Y., Kong, L., Cen, J., Chen, R., Zhang, W., Pan, L., Chen, K., Liu, Z.: Segment any point cloud sequences by distilling vision foundation models. Advances in Neural Information Processing Systems **36** (2024)
31. Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D., Han, S.: Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In: IEEE International Conference on Robotics and Automation (ICRA) (2023)

32. Ma, L., Ghafarianzadeh, M., Coleman, D., Correll, N., Sibley, G.: Simultaneous localization, mapping, and manipulation for unsupervised object discovery. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 1344–1351. IEEE (2015)
33. Ma, L., Sibley, G.: Unsupervised dense object discovery, detection, tracking and reconstruction. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13. pp. 80–95. Springer (2014)
34. Ma, Y., Peri, N., Wei, S., Hua, W., Ramanan, D., Li, Y., Kong, S.: Long-tailed 3d detection via 2d late fusion. arXiv preprint arXiv:2312.10986 (2023)
35. Madan, A., Peri, N., Kong, S., Ramanan, D.: Revisiting few-shot object detection with vision-language models. arXiv preprint arXiv:2312.14494 (2023)
36. Mahmoud, A., Hu, J.S., Kuai, T., Harakeh, A., Paull, L., Waslander, S.L.: Self-supervised image-to-point distillation via semantically tolerant contrastive loss. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7102–7110 (2023)
37. McInnes, L., Healy, J., Astels, S.: hdbscan: Hierarchical density based clustering. The Journal of Open Source Software **2**(11), 205 (2017)
38. Najibi, M., Ji, J., Zhou, Y., Qi, C.R., Yan, X., Ettinger, S., Anguelov, D.: Motion inspired unsupervised perception and prediction in autonomous driving. In: European Conference on Computer Vision. pp. 424–443. Springer (2022)
39. Najibi, M., Ji, J., Zhou, Y., Qi, C.R., Yan, X., Ettinger, S., Anguelov, D.: Unsupervised 3d perception with 2d vision-language distillation for autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8602–8612 (2023)
40. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
41. Ošep, A., Meinhardt, T., Ferroni, F., Peri, N., Ramanan, D., Leal-Taixé, L.: Better call sal: Towards learning to segment anything in lidar. arXiv preprint arXiv:2403.13129 (2024)
42. Peri, N., Dave, A., Ramanan, D., Kong, S.: Towards long-tailed 3d detection. In: Conference on Robot Learning. pp. 1904–1915. PMLR (2023)
43. Poursaeed, O., Jiang, T., Qiao, H., Xu, N., Kim, V.G.: Self-supervised learning of point clouds via orientation estimation. In: 2020 International Conference on 3D Vision (3DV). pp. 1018–1028. IEEE (2020)
44. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
45. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
46. Sanghi, A.: Info3d: Representation learning on 3d objects using mutual information maximization and contrastive learning. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16. pp. 626–642. Springer (2020)
47. Sauder, J., Sievers, B.: Self-supervised deep learning on point clouds by reconstructing space. Advances in Neural Information Processing Systems **32** (2019)
48. Sautier, C., Puy, G., Gidaris, S., Boulch, A., Bursuc, A., Marlet, R.: Image-to-lidar self-supervised distillation for autonomous driving data. In: Proceedings of the

IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9891–9901 (2022)
49. Seidenschwarz, J., Ošep, A., Ferroni, F., Lucey, S., Leal-Taixé, L.: Semoli: What moves together belongs together. arXiv preprint arXiv:2402.19463 (2024)
50. Shen, Q., Yang, X., Wang, X.: Anything-3d: Towards single-view anything reconstruction in the wild. arXiv preprint arXiv:2304.10261 (2023)
51. Shin, J., Triebel, R., Siegwart, R.: Unsupervised discovery of repetitive objects. In: 2010 IEEE International Conference on Robotics and Automation. pp. 5041–5046. IEEE (2010)
52. Sun, J., Zheng, H., Zhang, Q., Prakash, A., Mao, Z.M., Xiao, C.: Calico: Self-supervised camera-lidar contrastive pre-training for bev perception. arXiv preprint arXiv:2306.00349 (2023)
53. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
54. Tian, H., Chen, Y., Dai, J., Zhang, Z., Zhu, X.: Unsupervised object detection with lidar clues. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5962–5972 (2021)
55. Tian, X., Ran, H., Wang, Y., Zhao, H.: Geomae: Masked geometric target prediction for self-supervised point cloud pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13570–13580 (2023)
56. Vedder, K., Peri, N., Chodosh, N., Khatri, I., Eaton, E., Jayaraman, D., Liu, Y., Ramanan, D., Hays, J.: Zeroflow: Fast zero label scene flow via distillation. arXiv preprint arXiv:2305.10424 (2023)
57. Wang, T., Zhu, X., Pang, J., Lin, D.: Fcos3d: Fully convolutional one-stage monocular 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 913–922 (2021)
58. Wilson, B., Kira, Z., Hays, J.: 3d for free: Crossmodal transfer learning using hd maps. arXiv preprint arXiv:2008.10592 (2020)
59. Wilson, B., Qi, W., Agarwal, T., Lambert, J., Singh, J., Khandelwal, S., Pan, B., Kumar, R., Hartnett, A., Pontes, J.K., Ramanan, D., Carr, P., Hays, J.: Argoverse 2: Next generation datasets for self-driving perception and forecasting. In: Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021) (2021)
60. Wong, K., Wang, S., Ren, M., Liang, M., Urtasun, R.: Identifying unknown instances for autonomous driving. In: Conference on Robot Learning. pp. 384–393. PMLR (2020)
61. Xie, S., Gu, J., Guo, D., Qi, C.R., Guibas, L., Litany, O.: Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 574–591. Springer (2020)
62. Yin, J., Zhou, D., Zhang, L., Fang, J., Xu, C.Z., Shen, J., Wang, W.: Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection. In: European Conference on Computer Vision. pp. 17–33. Springer (2022)
63. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3d object detection and tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11784–11793 (2021)

64. Yin, T., Zhou, X., Krähenbühl, P.: Multimodal virtual point 3d detection. Advances in Neural Information Processing Systems **34**, 16494–16507 (2021)
65. You, Y., Luo, K., Phoo, C.P., Chao, W.L., Sun, W., Hariharan, B., Campbell, M., Weinberger, K.Q.: Learning to detect mobile objects from lidar scans without labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1130–1140 (2022)
66. Zhang, D., Liang, D., Yang, H., Zou, Z., Ye, X., Liu, Z., Bai, X.: Sam3d: Zero-shot 3d object detection via segment anything model. arXiv preprint arXiv:2306.02245 (2023)
67. Zhang, Z., Girdhar, R., Joulin, A., Misra, I.: Self-supervised pretraining of 3d features on any point-cloud. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10252–10263 (2021)
68. Zhang, Z., Dong, Y., Liu, Y., Yi, L.: Complete-to-partial 4d distillation for self-supervised point cloud sequence representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17661–17670 (2023)
69. Zhou, X., Girdhar, R., Joulin, A., Krähenbühl, P., Misra, I.: Detecting twenty-thousand classes using image-level supervision. In: ECCV (2022)