DetZero: Rethinking Offboard 3D Object Detection with Long-term Sequential Point Clouds

Tao Ma[1,2*] Xuemeng Yang[2] Hongbin Zhou[2] Xin Li[3,2*] Botian Shi[2] Junjie Liu[4*]
Yuchen Yang[5,2*] Zhizheng Liu[6*] Liang He[3] Yu Qiao[2] Yikang Li[2✉]
Hongsheng Li[1,2,7✉]

[1]Multimedia Laboratory, The Chinese University of Hong Kong
[2]Shanghai Artificial Intelligence Laboratory [3]East China Normal University
[4]South China University of Technology [5]Fudan University [6]ETH Zurich [7]CPII
✉ Corresponding author

https://github.com/PJLab-ADG/DetZero

Abstract

Existing offboard 3D detectors always follow a modular pipeline design to take advantage of unlimited sequential point clouds. We have found that the full potential of offboard 3D detectors is not explored mainly due to two reasons: (1) the onboard multi-object tracker cannot generate sufficient complete object trajectories, and (2) the motion state of objects poses an inevitable challenge for the object-centric refining stage in leveraging the long-term temporal context representation. To tackle these problems, we propose a novel paradigm of offboard 3D object detection, named DetZero. Concretely, an offline tracker coupled with a multi-frame detector is proposed to focus on the completeness of generated object tracks. An attention-mechanism refining module is proposed to strengthen contextual information interaction across long-term sequential point clouds for object refining with decomposed regression methods. Extensive experiments on Waymo Open Dataset show our DetZero outperforms all state-of-the-art onboard and offboard 3D detection methods. Notably, DetZero ranks 1st place on Waymo 3D object detection leaderboard[1] with 85.15 mAPH (L2) detection performance. Further experiments validate the application of taking the place of human labels with such high-quality results. Our empirical study leads to rethinking conventions and interesting findings that can guide future research on offboard 3D object detection.

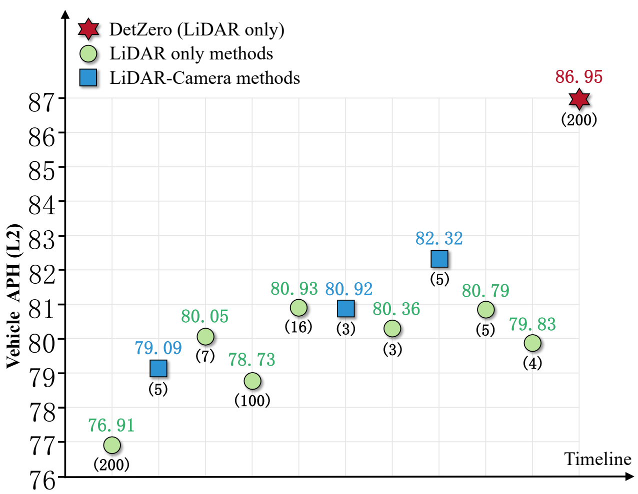

Figure 1. We compare several SOTA methods proposed along the timeline on the Waymo 3D detection leaderboard. Our DetZero obtains the best performance with a remarkable margin on *Vehicle*. We mark the number of input point cloud frames below each method. Please refer to Table 1 for a detailed comparison.

1. Introduction

Autonomous driving has rapidly advanced with promising progress in both industry and academia. A crucial component of this development is offboard 3D object detection, which can utilize entire sequence data from sensors (video or sequential point cloud) with few constraints on model capacity and inference speed. Therefore, some approaches [33, 50] are dedicated to developing high-quality "auto labels", aiming to reduce manual labor in point cloud annotation.

Subsequently, many online detectors [54, 8, 56] are in-

*Work performed during internship at Shanghai Artificial Intelligence Laboratory.

[1]https://waymo.com/open/challenges/2020/3d-detection/.

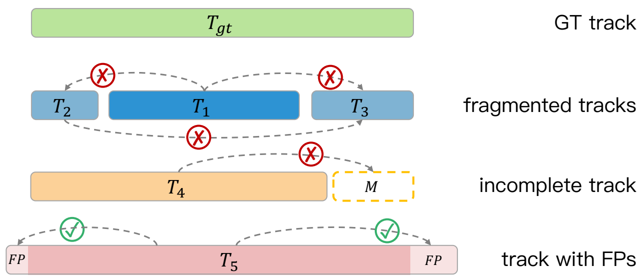

Figure 2. The quality of object tracks affects following refining much. Compared to T^{gt}, T_1, T_2, T_3 are fragmented tracks caused by ID switches, T_4 is an incomplete track with missing boxes M, T_5 contains many false positive (FP) boxes at both the beginning and the end. The dashed lines represent coordinate transform with the pose matrix. During object-centric refining, the full sequential context cannot be utilized across T_1, T_2, T_3; missing boxes M cannot be compensated from existing recalled part of T_4; refined boxes with higher scores are still FP after being transformed to FP's location.

troduced with the majority focusing on developing sophisticated modules to better utilize temporal context. As shown in Fig. 1, these newly proposed methods outperform both online [61, 20, 49, 37, 58, 36, 38, 22, 18, 15] and offboard 3D detectors [33, 50, 48] by a large margin, leaving the impression that current architecture and pipeline of offboard 3D detectors are too weak to learn the complex representation over long-term sequential point clouds. Therefore, in this paper, we revisit state-of-the-art (SOTA) offboard 3D detectors (see Sec. 3.1 for the pipeline) and identify two main factors hindering the full potential: (1) the onboard multi-object tracker can't generate sufficient complete object trajectories, and (2) the motion state of objects poses an inevitable challenge for the object-centric refining stage to leverage the long-term temporal context representation.

Specifically, prevailing online 3D detectors achieve promising performance but easily generate severe fragment trajectories, ID switches, and false positives when coupled with a tracking-by-detection multi-object tracking algorithm. As shown in Fig. 2, this phenomenon may prevent the generation of complete temporal context features. Therefore, we adopt an upstream module comprising a multi-frame 3D detector and offline tracker that ensures the completeness and continuity of object tracking while maintaining high recall. Moreover, the sliding-window-based auto labeling model [33, 50] hinders the exploitation of the commonality of object features, as shown in Fig. 3. We notice that the size of the objects remained consistent over time. By capturing data from various viewpoints, we can enhance the point cloud of an object, allowing for more precise size estimation. Furthermore, the object trajectory is independent of its size, and should always follow the kinematic constraints in continuous time, which is manifested by the smoothness of the trajectory. These characteristics serve as the foundation for leveraging the long-term sequential point clouds in a decomposed paradigm: refine the geometry size, smooth the trajectory position, and update the confidence score.

By focusing on these main issues, we propose a new paradigm of offboard 3D object detection named DetZero. A tenet is underscored here: emphasizing high-recall detection and tracking during the upstream, meticulous high-accuracy refining with long-term temporal context during the downstream. Comprehensive empirical studies and evaluations on the Waymo Open Dataset (WOD) demonstrate that DetZero significantly improves perception by fully utilizing long-term sequential point clouds. Notably, we rank 1st place on WOD 3D object detection leaderboard with 85.15 mAPH (L2). Extensive ablation studies and generalization experiments show that our method performs well with different quality upstream inputs and stricter metrics. Semi-supervised experiments further demonstrate that our method can provide high-quality auto labels for onboard 3D object detection models, which are already on par or even slightly higher than human labels.

The main contributions of our work are summarized as follows:

- We introduce DetZero, a new paradigm of offboard 3D object detection, to activate the potential of long-term sequential point clouds.
- Our proposed multi-frame object detection and offline tracking module generates accurate and complete object tracks, which is crucial for downstream refinement.
- An attention-mechanism based refining module is proposed to leverage the long-term temporal contextual information for objects' attribute predictions.
- We achieve state-of-the-art 3D object detection performance and tracking performance on the challenging WOD with remarkable margins.

2. Related Work

3D Object Detection. Current 3D object detectors usually process the point cloud in different manners: grid-based and point-based. Different grid-split schemes are designed to transform the point cloud into 3D voxels [61, 49, 57, 58], pillars [45, 20] and bird-eye view maps [51] representation. Point-based methods [37, 38, 53, 52, 39, 29] often employ PointNet [31, 32] as a base feature extractor. The hybrid strategy [36, 60, 53, 12] is also utilized to leverage both advantages. Besides, transformer networks make a success to extract point clouds feature by attention mechanism [26, 62, 43, 35, 11, 63], which have shown great potential.

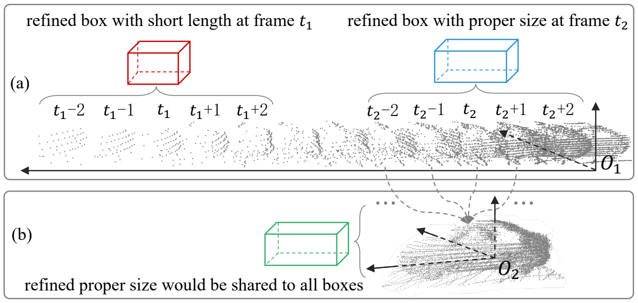

Figure 3. For a dynamic object track, the sliding-window based refining model (a) outputs inaccurate box size at frame t_1 when the object points are sparse, since it can't utilize dense point features at frame t_2. In (b), we merge object points together (from origin O_1 to O_2), then the points from each frame can contribute to precise size prediction.

3D MOT. Most 3D MOT methods [44, 28] still follow the tracking-by-detection paradigm, which benefits from off-the-shelf SOTA 3D detectors. AB3DMOT [46] associates the detected box with tracked trajectories by Kalman filter [17] and Hungarian algorithm. CenterPoint [58] calculates the previous position of the detected box with predicted velocity to get similarity with tracked box and solves matching pair by a greedy algorithm. Pre-processing detection boxes and GIoU-based two-stage data association strategy show effectiveness to improve the performance [28]. Besides, the mismatches are sharply decreased by enlarging the maximum death age to alleviate early termination [44] on WOD.

Sequential Point Clouds Learning. With the emergence of large-scale LiDAR serialized point cloud datasets [42, 6, 9, 5], researchers are increasingly exploring the use of sequential point clouds in real-world scenarios, such as multi-frame object detection [8, 55, 48], point cloud segmentation [3, 27, 16], object tracking [46, 28, 44], and scene flow prediction [24, 10, 47]. Notable works in this area include 3DAL [33] and Auto4D [50], which refine detection boxes at the trajectory level with human input or off-the-shelf detectors. Both of them contain some components that follow the sliding window fashion, which ignores the importance of the long-term characteristics of the tracks. These works inspired our own research efforts in this area.

3. Methodology

We first give a brief review of the entire pipeline of offboard 3D object detection. Meanwhile, the core problem is stated by introducing the input and output representations in Sec. 3.1. Afterward, we present how to boost the potential of the overall pipeline by improving both the upstream object tracks generation in Sec. 3.2, and the downstream attribute-based refining in Sec. 3.3. Details of the network architecture, losses and training strategy are described in Appendix.

3.1. Preliminary

We select 3DAL [33] as our baseline, which is a SOTA offboard 3D detector consisting of four modules to process a sequence of point clouds. Specifically, the first detection module takes as input N frames of point clouds $\left\{\mathcal{P}_i \in \mathbb{R}^{n_i \times (3+C)} \mid i = 1, 2, ..., N\right\}$ (n_i points for each frame, C is the additional feature for each point such as intensity and elongation) and outputs frame-level 3D bounding boxes $\hat{b}_i \in \mathbb{R}^{m_i \times 7}$ and categories. Then, a multi-object baseline tracker links the detected objects across frames as continuous object tracks $\{T_j \in \mathbb{R}^{L_j \times 7}, j \in N_{obj}\}$ (L is track length) with the unique object IDs. For each object track $T_{j,i}$ (j-th object at frame i), the object-specific LiDAR points are extracted by cropping original point clouds within corresponding bounding boxes, which are then merged together by eliminating ego-motion with frame poses $\left\{\mathcal{M}_i = [\mathbf{R}_i | \mathbf{t}_i] \in \mathbb{R}^{3\times 4}\right\}$. The third motion classification module is utilized to determine an object's motion state (static or dynamic) based on its trajectory features. In the final step, the object-centric auto labeling models extract the object's temporal representation separately based on its predicted motion state, to predict precise boxes. The refined boxes are eventually transferred back to each frame with the inverse frame pose. Please refer to the original paper for more details [33].

There are two factors that affect object-specific temporal context learning: *incomplete tracks* from upstream module and *motion-state-based* auto labeling models that ignore common object characteristics. Incomplete object tracks hinder the generation of effective object-specific temporal point cloud data, illustrated in Fig. 2. The sliding-window-based dynamic object refining mechanism fails to use complete temporal contexts, such as the relation between local position and global trajectory and object geometry consistency, as depicted in Fig. 3.

These observations prompt reconsideration of current offboard 3D object detection conventions. As is illustrated in Fig. 4, our evolution focuses on two main aspects: (1) using a multi-frame 3D detector and offline tracker to provide sufficient accurate and complete object tracks, and (2) modernizing the attention-mechanism refining module to reason about object attribute representations in long-term sequential point clouds. Both aspects significantly impact the model's performance, yet were not thoroughly investigated in prior studies.

3.2. Complete Object Tracks Generation

Our upstream object detection and tracking module aims to generate accurate and complete object tracks, which is essential as the entry point to the whole pipeline.

Object Detection. The competitive CenterPoint [58] is

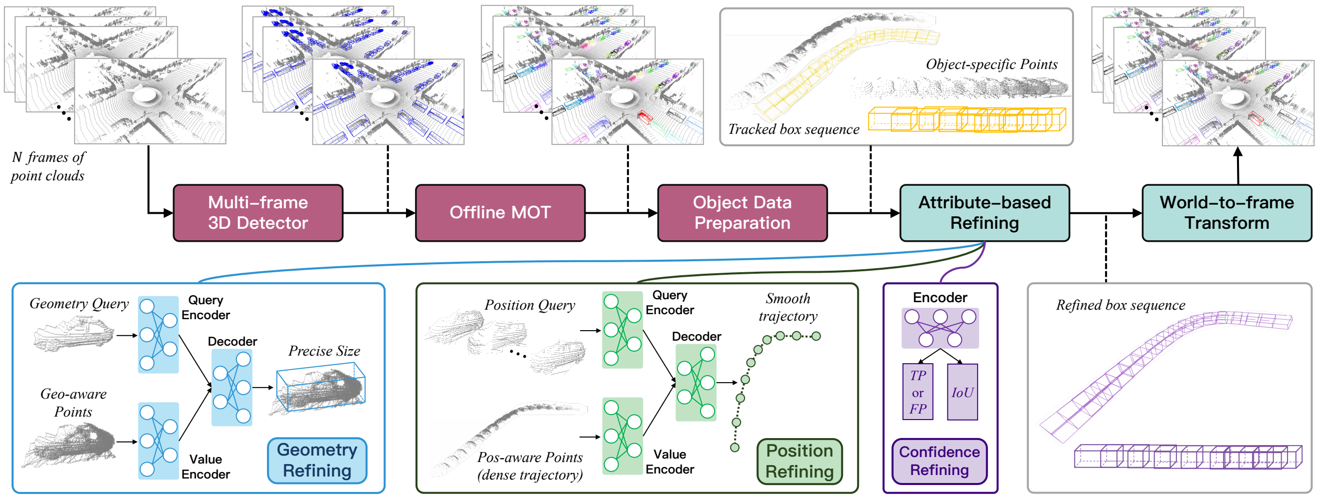

Figure 4. Overview of the framework of our proposed DetZero. The multi-frame detector takes as input N frames of point clouds, the following offline tracker generates accurate and complete object tracks. For each object track, we prepare its object-specific LiDAR points sequence and tracked box sequence. Consequently, we refine the object tracks through three simultaneous steps: refine the geometry size, smooth the motion trajectory and update the confidence score. Afterwards, they are combined together and transformed through world-to-frame poses as the final "auto labels".

adopted as our base detector because the anchor-free design would predict dense and redundant bounding boxes. To provide accurate prediction results as much as possible, we strengthen it in three aspects: (1) a combination of five frames of point cloud serves as the input to maximize the contributions rather than performance diminishes [33, 8]; (2) a point density aware module is designed to leverage the raw point features and voxel feature for precise refinement [13]; (3) to improve the adaption towards complex surroundings, test time augmentation [19] (TTA) for point cloud data and multi-model ensemble (different resolution, network structure and capacity) are utilized to boost the detection performance. Please see the details in Appendix.

Offline Tracking. Recent multi-object trackers [46], taking the tracking-by-detection path, always struggle in redundant detected bounding boxes which focus much on box-level detection metrics. Taking inspiration from [28, 44], our multi-object tracker utilizes a two-stage data association strategy to mitigate the possibility of false matching. Concretely, the detected boxes are partitioned into two distinct groups based on their confidence scores. The pre-existing object tracks initially engage in data association solely with the high-score group, and subsequently, successfully associated boxes are utilized to update the existing tracks. The un-updated tracks are further associated with the low-score group, and the un-associated boxes are deprecated. In addition, the life cycle of an object is allowed to persist immortally until the sequence terminates, after which any redundant boxes that have not been updated are removed. This operation benefits the re-connection of truncated object tracks and effectively prevents ID switches.

Post-processing is also crucial to generate good-quality tracks. We re-execute our tracking method following reverse time order to generate another group of tracks $T_{i,j}^{\text{inv}}$. These tracks are then associated together by a location-aware similarity matching score. Finally, we fuse the paired tracks with the WBF [40] strategy to further ameliorate the missing boxes and stable the motion state, which is called *forward and reverse order tracking fusion*. Besides, we do not run the downstream modules for too short tracks. The boxes of those short tracks, together with the redundant boxes that have not been updated are merged directly into the final auto labels.

Object Data Preparation. Given an object track (identified by the unique object ID), we first slightly scale up the RoI area of the tracked boxes by a parameter α along three dimensions, which compensates for abundant contextual information. Then, points that lie within the regions bounded by these enlarged boxes are taken out. We denote this object-specific LiDAR points sequence as $\{\mathcal{P}_{j,i}\}$ for object j with length L_j at i-th frame of the original sequence, as well as its corresponding tracked box sequence $\{\hat{b}_{j,i}\}$ and confidence scores $\{S_{j,i}\}$.

3.3. Attribute-based Refining Module

Previous object-centric auto labeling methods have employed a state-based strategy to refine the proposals generated by upstream modules. This approach not only results in the propagation of misclassification but also disregards the potential similarities between objects. However, it has been observed that, for rigid objects, the geometric shape of an object does not vary significantly over a continuous period of time, regardless of its motion state. Furthermore, an object's motion state typically exhibits regular patterns and strong consistency with neighboring moments. Based on these observations, we propose a novel approach that decomposes the traditional bounding box regression task into three distinct modules that predict an object's geometry, position, and confidence attributes, respectively.

3.3.1 Geometry Refining Model

Geometry-aware Points Generation. The acquisition of complementary information regarding the appearance and shape of an object can be facilitated by obtaining multiple viewpoints of an object. To obtain a cohesive object track, a local coordinate transform operation, involving translation and rotation, similar to the method presented in [37, 30], is initially applied to align the object points to a local box coordinate at various locations. Subsequently, points from different frames are amalgamated, irrespective of their original origin. Henceforth, we randomly sample a set of points ($\mathcal{P}_j = \{p_1, ..., p_n\} \in \mathbb{R}^{n\times(3+C)}, n = 4096$) for further processing.

Proposal-to-Point Encoding. It is of paramount importance to effectively utilize the geometric information by encoding proposals into object points rather than discarding them after object points extraction [35]. Specifically, for each point p_k of $\mathcal{P}_j$ and its corresponding box $\hat{b}_{j,i}$, we use a point-to-surface approach to compute the projection distance between p_k and the six surfaces of $\hat{b}_{j,i}$, denoted as Δp_k^{sf}. The newly generated point features can be viewed as a better representation of proposal information, which can be expressed as $\left[p_k, \Delta p_k^{\mathrm{sf}_1}, \cdots, \Delta p_k^{\mathrm{sf}_6}\right]$, where $[\cdot]$ denotes the concatenate operation.

Attention-based Geometry Interaction across Views. It has been investigated in 3D object detection that a better initialization of object queries would benefit the convergency of the transformer network [4]. Inspired by this observation, we propose to initialize the geometry query features based on object-specific points. Firstly, we randomly select t samples from the whole object track. Each sample has corresponding 256 randomly-selected points. Each point is also augmented by our proposed proposal-to-point encoding approach, and besides, the corresponding confidence scores. Afterwards, a PointNet-structure encoder ENC_1 is adopted to extract features for each selected sample, which is used to initialize as the geometry queries $\mathbf{Q}^{\mathrm{geo}} \in \mathbb{R}^{t\times D}$. Then we utilize another encoder ENC_2 to take as input $\mathcal{P}_j$ and extract dense point features, which are served as $\mathbf{K}^{\mathrm{geo}}$ and $\mathbf{V}^{\mathrm{geo}} \in \mathbb{R}^{n\times D}$.

The generated geometry queries are first fed into the multi-head self-attention layer, to encode rich contextual relationships among selected samples and feature dependencies for refining geometry information. The following cross attention between geometry queries and the point features aggregates relevant context onto the object candidates, which reasons pairwise differences to compensate the point features of supplementary views for each geometry query. At last, a feed-forward network (FFN) independently decodes t geometry queries into t geometry sizes, which are then averaged as the final predicted size.

For better residual target regression, we map the proposals' size to D-dim embeddings with a linear projection layer. They are element-wisely summed with the query features. Details of the network architecture are shown in Appendix.

3.3.2 Position Refining Model

Position-aware Points Generation. For j-th object, we randomly select the position of a box from its tracked box sequence $\{\hat{b}_{j,i}\}$ as a new local coordinate system, and subsequently, the other boxes are transformed to this coordinate, as well as the corresponding object-specific points $\{\mathcal{P}_{j,i}\}$. Then, a fixed number of points are randomly selected from $\{\mathcal{P}_{j,i}\}$ for each frame.

For each point, in addition to calculating the distance to the proposal's center, we also compute the relative coordinates between each point and eight corners of the corresponding tracked box as $\Delta p_k^{\mathrm{co}} = p_k - p^{\mathrm{co}}$, which results in a 27-dim feature vector. The final position-aware point features can be expressed as $f_k^{\mathrm{pos}} = [p_k, \Delta p_k^{\mathrm{ce}}, \Delta p_k^{\mathrm{co}_1}, \cdots, p_k^{\mathrm{co}_8}]$. To facilitate training, all object tracks are padded to the same length with zeros.

Attention-based Local-to-Global Position Interaction. For an object track, we utilize the same structured query encoder as ENC_1 in Geometry Refining Model (GRM) to generate position queries $\mathbf{Q}^{\mathrm{pos}} \in \mathbb{R}^{L\times D}$ for L frame, whose features consist of position-aware features f^{pos} and confidence scores. Simultaneously, we extract the point features of the entire object track using another encoder that takes f^{pos} as input. These features serve as $\mathbf{K}^{\mathrm{pos}}$ and $\mathbf{V}^{\mathrm{pos}} \in \mathbb{R}^{n^{\mathrm{pos}}\times D}$ for subsequent computation, where n^{pos} is the number of sampled points.

The position queries are first fed into the self-attention module, to capture the relative distance between itself and others. Additionally, we apply a 1D mask near the position of each query to weigh the self-attention. Subsequently, the local position queries $\mathbf{Q}^{\mathrm{pos}}$ and global point trajectory features $\mathbf{K}^{\mathrm{pos}}$, $\mathbf{V}^{\mathrm{pos}}$ are fed into the cross-attention module to model the local-to-global position contextual relations. Finally, we predict the offsets between each ground-truth center and the corresponding initial center under the local coordinate system, as well as the bin-based heading angle.

Method	Rank	Frames	mAPH	*Vehicle* (AP/APH)		*Pedestrian* (AP/APH)		*Cyclist* (AP/APH)	
			L2	L1	L2	L1	L2	L1	L2
DetZero_Ens† (Ours)	1	200	**85.15**	**92.17 / 91.80**	**87.32 / 86.95**	**92.08 / 89.67**	**88.08 / 85.64**	85.49 / 84.70	83.63 / 82.85
3DAL† [33]	–	200	–	85.84 / 85.46	77.24 / 76.91	–	–	–	–
LoGoNet_Ens‡ [21]	2	5	81.96	88.80 / 88.37	82.75 / 82.32	89.63 / 86.74	84.96 / 82.10	84.51 / 83.59	82.36 / 81.46
HRI_ADLAB_HZ*	3	4	81.32	86.77 / 86.40	80.19 / 79.83	88.59 / 86.01	83.84 / 81.27	**85.67 / 84.84**	**83.69 / 82.87**
MT-Net v2[7]	4	5	80.00	87.54 / 87.12	81.20 / 80.79	87.62 / 84.89	82.33 / 79.66	82.80 / 81.74	80.58 / 79.54
BEVFusion_TTA‡ [25]	5	3	79.97	87.96 / 87.58	81.29 / 80.92	87.64 / 85.04	82.19 / 79.65	82.53 / 81.67	80.17 / 79.33
LidarMultiNet_TTA [56]	6	3	79.94	87.64 / 87.26	80.73 / 80.36	87.75 / 85.07	82.48 / 79.86	82.77 / 81.84	80.50 / 79.59
MPPNet_TTA [8]	7	16	79.60	87.77 / 87.37	81.33 / 80.93	87.92 / 85.15	82.86 / 80.14	80.74 / 79.90	78.54 / 77.73
LIVOX Detection*	9	7	78.96	86.81 / 86.45	80.41 / 80.05	87.17 / 84.57	82.16 / 79.59	80.46 / 79.59	78.08 / 77.24
DeepFusion_Ens‡ [23]	12	5	78.41	86.45 / 86.09	79.43 / 79.09	86.14 / 83.77	80.88 / 78.57	80.53 / 79.80	78.29 / 77.58
AFDetV2-Ens [14]	17	1	77.64	85.80 / 85.41	78.41 / 78.34	85.22 / 82.16	79.71 / 76.75	81.20 / 80.30	78.70 / 77.83
INT_Ens [48]	19	100	77.21	85.63 / 85.23	79.12 / 78.73	84.97 / 81.87	79.35 / 76.36	79.76 / 78.65	77.62 / 76.54

Table 1. Performance comparison on the Waymo 3D detection leaderboard. Metrics are standard 3D AP and APH by both L1 and L2 difficulties. Note that all the listed entries use TTA or model ensemble techniques. We use † to denote the offboard 3D detectors. LiDAR-Camera fusion based 3D detectors are marked with ‡, and anonymous submissions are marked with $*$. We report the performance till 2023-03-08 23:59 GMT.

3.3.3 Confidence Refining Model

Our detection and offline tracking module is encouraged to generate sufficient object tracks, which naturally contain boxes that are far from being true positive, even after GRM and Position Refining Model (PRM). To address this issue, we introduce a Confidence Refining Model (CRM), composed of two branches to optimize the confidence scores.

The first classification branch is similar to the traditional second-stage object detector [34], for determining TPs or FPs by updating scores. We assign negative labels to the tracked boxes whose IoU ratios with corresponding ground-truth boxes are lower than τ_l. Tracked boxes with IoU ratio higher than τ_h are treated as positive samples. Other boxes do not contribute to the classification objective.

The second IoU regression branch predicts how much IoU an object should have after being refined [59]. Hence, the regression targets are set as the IoUs between ground-truth boxes and refined ones predicted by previous GRM and PRM.

In the beginning, we process the object-specific points with a similar network structure encoder identical to ENC_1 of GRM. The extracted point cloud features are fused by a simple MLP, and then fed into these two branches for predicting respective scores. During training, we randomly sample pre-divided positive and negative object tracks with $1:1$ ratio in each epoch for better convergence. The final scores are the geometric average of the two branches: $S_j = \sqrt{S^2_{j,\text{cls}} + S^2_{j,\text{iou}}}$.

4. Experiments

In this section, we first introduce the dataset details and evaluation metrics used in our experiments. We then provide a detailed performance comparison between our DetZero and other SOTA 3D detectors in Sec. 4.2. Then, we validate whether such high-quality "auto labels" by DetZero could play the same role as human labels in Sec. 4.3. In Sec. 4.4, we present the ablation studies and analysis for convincing each component of our entire approach. Please refer to Appendix for more detailed experiments and ablation results.

4.1. Dataset

We conduct experiments on the challenging Waymo Open Dataset [41], which is one of the largest dataset containing total 1150 LiDAR scenes with 798 for training, 202 for validation and 150 for testing. The dataset provides 20-second point clouds data for each scene with a sampling frequency at 10Hz, and 3D annotations for 4 object categories in 360 degree field of view. We follow the evaluation protocol with the official metrics, *i.e.*, average precision (AP) and average precision weighted by heading (APH), and report the results on both LEVEL 1 (L1) and LEVEL 2 (L2) difficulty levels. The L1 evaluation includes objects with more than five LiDAR points and L2 evaluation only includes 3D labels with at least one and no more than five LiDAR point. Note that mAPH (L2) is the main metric for ranking in the Waymo 3D detection challenge.

4.2. Comparising with State-of-the-art Detectors

We present a comprehensive comparison of our DetZero with various state-of-the-art 3D detectors.

As shown in Table 1, our DetZero achieves the best results on Waymo 3D detection challenge leaderboard [1] with 85.15 mAPH (L2) detection performance. For comparisons among methods processing long-term sequential point clouds (at least 100 frames), DetZero surpasses 3DAL [33] with 5.93 (L1) and 9.51 (L2) mAPH on *Vehicle*, surpasses INT [48] with 6.16 (L1) and 7.69 (L2) mAPH on *Vehi-*

Method	Frames	*Vehicle* (AP/APH)		*Pedestrian* (AP/APH)	
		L1	L2	L1	L2
MVF [60]	1	62.93 / –	– / –	65.33 / –	– / –
PV-RCNN [36]	1	77.51 / 76.89	68.98 / 68.41	75.01 / 65.65	66.04 / 57.61
CenterPoint [58]	1	76.7 / 76.2	68.8 / 68.3	79.0 / 72.9	71.0 / 65.3
PDV [13]	1	76.85 / 76.33	69.30 / 68.81	74.19 / 65.96	65.85 / 58.28
INT [48]	2	– / –	– / 73.3	– / –	– / 71.9
3D-MAN [54]	16	74.50 / 74.00	67.60 / 67.10	71.10 / 67.70	62.60 / 59.00
CenterFormer [62]	8	78.80 / 78.30	74.30 / 73.80	82.10 / 79.30	77.80 / 75.00
3DAL† [33]	200	84.50 / –	– / –	82.88 / –	– / –
MPPNet [8]	16	82.74 / 82.28	75.41 / 74.96	84.69 / 82.25	77.43 / 75.06
DetZero† (Upstream)	5	83.07 / 82.57	75.72 / 75.24	86.17 / 83.07	79.39 / 76.34
DetZero† (Full)	200	**89.49 / 89.06**	**83.34 / 82.92**	**89.54 / 87.06**	**83.52 / 81.01**

Table 2. Performance comparison on the val set of WOD. Metrics are standard 3D AP and APH by both L1 and L2 difficulties. We use † to denote the entries using TTA or model ensemble techniques.

	3D AP		BEV AP	
	IoU=0.7	IoU=0.8	IoU=0.7	IoU=0.8
Human	86.45	60.49	93.86	86.27
3DAL	85.37	56.93	92.80	87.55
DetZero (Ours)	**90.24**	**67.61**	**95.15**	**90.04**

Table 3. Comparing human labels and auto labels. The results are 3D and BEV AP (L1 difficulty) under 0.7 and 0.8 IoU threshold for *Vehicle* on 5 sequences selected from WOD val set. Please refer to the appendix for more details about the sequences' IDs and human's AP computing method.

Training Data	*Vehicle*		*Pedestrian*	
	AP	APH	AP	APH
100% *train* (Human)	**75.41**	**74.88**	77.51	71.16
10% *train*(Human)	66.88	66.28	67.13	59.66
90% *train* (DetZero)	74.12	73.59	78.57	71.39
10% *train* (Human) + 90% *train* (DetZero)	74.44	73.91	**78.92**	**72.02**

Table 4. Intra-domain semi-supervised learning results.

cle, and 7.65 (L1) and 9.09 (L2) mAPH on *Pedestrian*. DetZero shows great ability to leverage the long-term sequential point clouds for offboard perception. Moreover, compared to state-of-the-art multi-modal fusion 3D detectors [21, 25, 23], DetZero also yields a strong performance gain with at least 3.43 (L1) and 4.63 (L2) mAPH on *Vehicle*, and 2.93 (L1) and 3.54 (L2) mAPH on *Pedestrian*. These results further highlight the great potential of the point cloud sequences explored by DetZero. We also ranked 1st place on Waymo 3D tracking challenge leadboard [2] with 75.05 MOTA (L2) by a 9.97 point margin, please see the detailed performance in Appendix.

Additionally, in Table 2, we provide a comparison between SOTA 3D detectors and our internal components on the validation set of WOD. We outperform other single-frame and multi-frame based methods with a huge margin on both *Vehicle* and *Pedestrian*. Thanks to the high-quality object tracks generated by our upstream module, our full model gets a significant internal improvement: 6.49 (L1) and 7.68 (L2) mAPH for *Vehicle*, 3.99 (L1) and 4.67 (L2) mAPH for *Pedestrian*, more analysis is shown in Sec. 4.4.

4.3. Comparising with Human Labels

It has been shown that humans' capability of recognizing objects in a dynamic 3D scene has minor fluctuations [33]. Human performance is measured by the consistency between the humans' single-frame-based re-labeling and released multi-frame-based ground-truth labels.

We follow their experimental setup to report the mean AP of our DetZero across the 5 selected sequences. In Table 3, we demonstrate superior performance compared to human and 3DAL in particular. With the common 3D AP@0.7 metric, we achieve 3.79 and 4.87 points gains, while the gap is larger in more strict 3D AP@0.8 metric. We obtain similar gains with the BEV AP by ignoring height. To the best of our knowledge, this is the first time that the offboard 3D detector model can outperform the average human labels.

To better study whether such high-quality auto labels could replace human labels for onboard model training, we conduct another intra-domain semi-supervised learning experiment. We choose the single-stage CenterPoint [58] as our student model. Note that the student model takes as input a single frame, and the GT-Paste data augmentation is not used during training. We first randomly select 10% sequences (79 ones) in the WOD training set to train our entire DetZero pipeline. Next, we can generate "auto labels" for the rest 90% sequences (719 ones) in the training set. Afterwards, the student model is trained with different combinations of human labels and "auto labels".

As shown in Table 4, the first two rows give a performance comparison by reducing the human annotations to

Det.	Tra.	GRM	PRM	CRM	*Vehicle* (L1 / L2) IoU=0.7	*Vehicle* (L1 / L2) IoU=0.8	*Pedestrian* (L1 / L2) IoU=0.5	*Pedestrian* (L1 / L2) IoU=0.6
✓					82.57 / 75.09	51.34 / 44.77	83.23 / 76.47	64.12 / 56.49
✓	✓				82.57 / 75.24	51.34 / 44.81	83.07 / 76.34	64.04 / 56.44
✓	✓	✓			84.49 / 77.17	56.71 / 49.60	84.71 / 78.04	68.33 / 60.48
✓	✓		✓		85.48 / 78.55	56.35 / 49.56	84.32 / 77.78	66.53 / 58.99
✓	✓	✓	✓		87.81 / 81.01	64.53 / 57.15	85.94 / 79.48	70.97 / 63.26
✓	✓	✓	✓	✓	**89.06 / 82.92**	**64.94 / 57.84**	**87.06 / 81.01**	**71.61 / 64.03**

Table 5. Effect of each component in our DetZero on WOD val set. Metrics are 3D APH of both L1 and L2 difficulties for *Vehicle* and *Pedestrian* with a standard IoU threshold (0.7 & 0.5) and a higher IoU threshold (0.8 & 0.6).

10%, which decreases the student model's performance by 8.53 and 8.6 points for *Vehicle*, 10.38 and 11.5 points for *Pedestrian*. Surprisingly, when we add other 90% auto labels, the performance increase with 7.56 and 7.63 points for *Vehile* which is close to the first row, 11.79 and 12.36 points for *Pedestrian* which is already higher. Besides, when we remove the 10% human labels (3rd row), the results are predictably slightly lower than the 4th row, still showing 1.06 and 0.23 gain for *Pedestrian*. These results demonstrate that "auto labels" generated by our DetZero are qualified for training online models. We visualized the results and found that "auto labels" contain fewer pedestrian labels than human labels, such as the hard samples at a far distance. Hence, the student model trained with our "auto labels" would output fewer false positives compared to the model trained with 100% human labels. More detailed analyses are in Appendix.

Trk_1	Trk_2	Ref_1	Ref_2	*Vehicle* L1	*Vehicle* L2	*Pedestrian* L1	*Pedestrian* L2
✓		✓		83.74	76.33	83.92	76.94
✓			✓	85.57	78.14	85.47	78.24
	✓	✓		84.92	77.06	84.64	77.52
	✓		✓	**89.06**	**82.92**	**87.06**	**81.01**

Table 6. Evaluate the function of different upstream and downstream modules. We reproduce 3DAL [33] and use the subscripts 1 and 2 to represent their and our models respectively. Metrics are standard 3D APH of both L1 and L2 difficulties for *Vehicle* and *Pedestrian*.

Trk_1	Trk_2	Ref_2	MOTA *Vehicle*	MOTA *Pedestrian*	Recall@track *Vehicle*	Recall@track *Pedestrian*
✓			57.14	61.78	23.96	26.63
	✓		58.41	62.50	40.28	45.19
✓		✓	63.24	65.52	27.41	27.77
	✓	✓	**71.36**	**68.78**	**57.79**	**51.64**

Table 7. Tracking performance comparison on val set of WOD. Metrics are standard 3D MOTA and track-level recall (Recall@track) of L2 difficulty. A ground-truth object track is regarded as a track-level TP only if at least 80% boxes are matched (3D IoU=0.7 for *Vehicle*, 0.5 for *Pedestrian*) with those of a single predicted track.

4.4. Ablation Studies and Analysis

We conduct ablation studies on the WOD validation set to verify all the components of our approach, especially under a fair experimental setting regardless of the techniques for the leaderboard. Additional ablations for the network structures and data augmentations are shown in Appendix.

Effects of each Component. We enable different combinations of our proposed modules to evaluate the performances. In addition to the commonly used standard IoU threshold, we also report the performance under a higher IoU threshold to more accurately assess the disparity between predictions and the ground-truth labels.

In Table 5, compared to the upstream results (2nd row), when IoU equals 0.7, the 3rd row shows that the GRM gains 1.92 (L1) and 1.93 (L2) points for *Vehicle*, and 1.64 and 1.7 points for *Pedestrian*. As a comparison, the 4th row shows that the PRM gains 2.91, 3.31, 1.25 and 1.44 points respectively. When we combine them together, the performance improves a lot, shown by the 5th row. And the CRM also performs well, by re-scoring the samples based on their qualities. When IoU equals 0.8, we get impressive improvements. Specifically, our full downstream refining module boosts the performance by 26.49% and 29.08% for *Vehicle*, by 11.82% and 13.45% for *Pedestrian*. This shows that our entire DetZero tries its best to generate high-quality 3D boxes.

Cross Evaluation. In order to better verify the effect of our proposed principle, we reproduce the baseline tracker [46] and motion state based object auto labeling model [33] and make a cross-evaluation between their modules and ours by using the same detection results (ours). In Table 6, the first row can be viewed as the 3DAL approach [33] and achieve the lowest performance. Based on this baseline performance, using attribute-based refining modules yields 1.83 and 1.81 point gains for *Vehicle*, 1.55 and 1.3

point gains for *Pesdesrian*. And using offline tracking provides 1.18 and 0.73 point gains for *Vehicle*, 0.72 and 0.58 point gains for *Pedestrian*. For this two groups comparison, the attribute-based refiner improves much more than offline tracker, though the object tracks are not that good, our refiner can still leverage the temporal context information. It also shows that complete object tracks are essential to affect the process of long-term sequential point clouds. The reason is revealed in Table 7, our offline tracker yields 16.32 point and 18.56 point gains on Recall@track respectively, while MOTA is slightly higher. Based on the complete tracks, our attribute-based refiner could further boost the performance, as the last row of Table 6 and 7 is shown. These results demonstrate the strong ability of our DetZero.

Compare with prior trackers. We replace our proposed offline tracker with several SOTA trackers but maintain all the other modules in our pipeline, and evaluate the trackers' performance (Recall@track) and the final performance after refining (3D APH). The other trackers lead to degraded final APH performance because our tracker promises the completeness of tracks (Recall@track). Note that the other trackers would update the geometry size and trajectory of objects, while our offline tracker doesn't at this step.

	Recall@track		3D APH	
	Vehicle	*Pedestrian*	*Vehicle*	*Pedestrian*
AB3DMOT [41]	23.96	26.63	78.14	78.24
SimpleTrack [23]	33.86	35.28	80.04	79.01
ImmortalTracker [39]	35.34	39.88	80.56	79.12
Ours	**40.28**	**45.19**	**82.92**	**81.01**

Table 8. Performance (L2) comparison on val set of WOD with different trackers.

Generalization Ability of Refining Module. To better verify the generalization ability of our approach, especially the proposed attribute-based refining module, we take as input three upstream results with different qualities for inference. The *low* group comes from our base detector, while the *mid* group utilizes the techniques mentioned in Sec. 3.2 to generate high-quality results. In *high* group, we leverage the image information to further boost the upstream performance. In Table 9, our downstream refining module obtains significant improvements in all three groups. Besides, on both *Vehicle* and *Pedestrian*, the improvements of L2 are greater than those of L1. These results further show two strong conclusions: (1) our upstream module can recall hard samples, even if they are not over the IoU threshold of true positive, and (2) our downstream refining module takes advantage of temporal context to optimize these hard samples.

	Vehicle		*Pedestrian*	
	L1	L2	L1	L2
upstream (*low*)	77.86	70.00	74.95	67.48
+ Refine	81.43	74.68	78.15	71.17
improvement	+3.57	+4.68	+3.20	+3.69
upstream (*mid*)	82.57	75.24	83.07	76.34
+ Refine	89.06	82.92	87.06	81.01
improvement	+6.49	+7.68	+3.99	+4.67
upstream (*high*)	83.80	76.99	85.77	79.74
+ Refine	89.34	83.57	88.30	82.94
improvement	+5.54	+6.58	+2.53	+3.20

Table 9. Verifying generalization ability of our DetZero on WOD val set. Metrics are standard 3D APH of both L1 and L2 difficulties for *Vehicle* and *Pedestrian*.

5. Conclusion

In this work, we have proposed DetZero, a state-of-the-art offboard 3D detector using long-term sequential point clouds as input. The cores of our success are a multi-frame object detector and offline tracker which generates high-quality complete object tracks, and an attribute-based auto labeling model leveraging the full potential of long-term sequential point clouds. Evaluated on WOD, our method has ranked 1st place, showing remarkable margins over prior art 3D detectors. Moreover, the extensive ablation studies and analysis lead to convincing evaluation and application exploration with such high-quality perception results.

Acknowledgments

This project is funded in part by National Key R&D Program of China Project 2022ZD0161100, by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)'s InnoHK, by General Research Fund of Hong Kong RGC Project 14204021, and by the Science and Technology Commission of Shanghai Municipality (No. 22DZ1100102). Hongsheng Li is a PI of CPII under the InnoHK.

References

[1] Waymo open dataset: 3d detection challenge. https://waymo.com/open/challenges/3d-detection/. Accessed: 2021-01-25. 6

[2] Waymo open dataset: 3d detection challenge. https://waymo.com/open/challenges/2020/3d-tracking/. Accessed: 2021-01-25. 7

[3] Mehmet Aygun, Aljosa Osep, Mark Weber, Maxim Maximov, Cyrill Stachniss, Jens Behley, and Laura Leal-Taixé. 4d panoptic lidar segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5527–5537, 2021. 3

[4] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022. 5

[5] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019. 3

[6] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 3

[7] Shaoxiang Chen, Zequn Jie, Xiaolin Wei, and Lin Ma. Mtnet submission to the waymo 3d detection leaderboard. *arxiv preprint*, 2022. 6

[8] Xuesong Chen, Shaoshuai Shi, Benjin Zhu, Ka Chun Cheung, Hang Xu, and Hongsheng Li. Mppnet: Multi-frame feature intertwining with proxy points for 3d temporal object detection. In *Proceedings of the European conference on computer vision (ECCV)*, September 2022. 1, 3, 4, 6, 7

[9] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 3

[10] Xiuye Gu, Yijie Wang, Chongruo Wu, Yong Jae Lee, and Panqu Wang. Hplflownet: Hierarchical permutohedral lattice flownet for scene flow estimation on large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3254–3263, 2019. 3

[11] Tianrui Guan, Jun Wang, Shiyi Lan, Rohan Chandra, Zuxuan Wu, Larry Davis, and Dinesh Manocha. M3detr: Multi-representation, multi-scale, mutual-relation 3d object detection with transformers, 2021. 2

[12] Chenhang He, Zeng Hui, Jianqiang Huang, Xiansheng Hua, and Lei Zhang. Sa-ssd: Structure aware single-stage 3d object detection from point cloud. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[13] Jordan SK Hu, Tianshu Kuai, and Steven L Waslander. Point density-aware voxels for lidar 3d object detection. In *CVPR*, pages 8469–8478, 2022. 4, 7

[14] Yihan Hu, Zhuangzhuang Ding, Runzhou Ge, Wenxin Shao, Li Huang, Kun Li, and Qiang Liu. Afdetv2: Rethinking the necessity of the second stage for object detection from point clouds. *arxiv preprint*, 2021. 6

[15] Keli Huang, Botian Shi, Xiang Li, Xin Li, Siyuan Huang, and Yikang Li. Multi-modal sensor fusion for auto driving perception: A survey. *arXiv preprint arXiv:2202.02703*, 2022. 2

[16] Juana Valeria Hurtado, Rohit Mohan, Wolfram Burgard, and Abhinav Valada. Mopt: Multi-object panoptic tracking. *arXiv preprint arXiv:2004.08189*, 2020. 3

[17] R. Kalman. A new approach to linear filtering and prediction problems. In *Journal of Basic Engineering*, 1960. 3

[18] Lingdong Kong, Youquan Liu, Xin Li, Runnan Chen, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Robo3d: Towards robust and reliable 3d perception against corruptions. *arXiv preprint arXiv:2303.17597*, 2023. 2

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012. 4

[20] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[21] Xin Li, Tao Ma, Yuenan Hou, Botian Shi, Yuchen Yang, Youquan Liu, Xingjiao Wu, Qin Chen, Yikang Li, Yu Qiao, and Liang He. Logonet: Towards accurate 3d object detection with local-to-global cross-modal fusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 6, 7

[22] Xin Li, Botian Shi, Yuenan Hou, Xingjiao Wu, Tianlong Ma, Yikang Li, and Liang He. Homogeneous multi-modal feature fusion and interaction for 3d object detection. In *European Conference on Computer Vision*, pages 691–707. Springer, 2022. 2

[23] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Bo Wu, Yifeng Lu, Denny Zhou, Quoc V. Le, Alan Yuille, and Mingxing Tan. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022. 6, 7

[24] Xingyu Liu, Charles R Qi, and Leonidas J Guibas. Flownet3d: Learning scene flow in 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 529–537, 2019. 3

[25] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. *arxiv preprint*, 2022. 6, 7

[26] Jiageng Mao, Yujing Xue, Minzhe Niu, Haoyue Bai, Jiashi Feng, Xiaodan Liang, Hang Xu, and Chunjing Xu. Voxel transformer for 3d object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2021. 2

[27] Rodrigo Marcuzzi, Lucas Nunes, Louis Wiesmann, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Contrastive instance association for 4d panoptic segmentation using sequences of 3d lidar scans. *IEEE Robotics and Automation Letters*, 7(2):1550–1557, 2022. 3

[28] Ziqi Pang, Zhichao Li, and Naiyan Wang. Simpletrack: Understanding and rethinking 3d multi-object tracking. *arxiv preprint*, 2021. 3, 4

[29] Charles R. Qi, Or Litany, Kaiming He, and Leonidas J. Guibas. Deep hough voting for 3d object detection in point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 2

[30] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 5

[31] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification

and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2017. 2

[32] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:2212.07289*, 2022. 2

[33] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 1, 2, 3, 4, 6, 7, 8

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arxiv preprint*, 2015. 6

[35] Hualian Sheng, Sijia Cai, Yuan Liu, Bing Deng, Jianqiang Huang, Xian-Sheng Hua, and Min-Jian Zhao. Improving 3d object detection with channel-wise transformer. In *ICCV*, pages 2743–2752, 2021. 2, 5

[36] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 7

[37] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 5

[38] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 2

[39] Weijing Shi and Ragunathan (Raj) Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[40] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *arxiv preprint*, 2019. 4

[41] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, and Benjamin Caine. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 6

[42] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 3

[43] Pei Sun, Mingxing Tan, Weiyue Wang, Chenxi Liu, Fei Xia, Zhaoqi Leng, and Drago Anguelov. Swformer: Sparse window transformer for 3d object detection in point clouds. In *Proceedings of the European conference on computer vision (ECCV)*, September 2022. 2

[44] Qitai Wang, Yuntao Chen, Ziqi Pang, Naiyan Wang, and Zhaoxiang Zhang. Immortal tracker: Tracklet never dies. *arxiv preprint*, 2021. 3, 4

[45] Yue Wang, Alireza Fathi, Abhijit Kundu, David Ross, Caroline Pantofaru, Thomas Funkhouser, and Justin Solomon. Pillar-based object detection for autonomous driving. In *Proceedings of the European conference on computer vision (ECCV)*, September 2020. 2

[46] Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *Proceedings of the IEEE Conference Conference on Intelligent Robots and Systems (IROS)*, pages 10359–10366. IEEE, 2020. 3, 4, 8

[47] Pengxiang Wu, Siheng Chen, and Dimitris N Metaxas. Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11385–11395, 2020. 3

[48] Jianyun Xu, Zhenwei Miao, Da Zhang, Hongyu Pan, Kaixuan Liu, Peihan Hao, Jun Zhu, Zhengyang Sun, Hongmin Li, and Xin Zhan. Int: Towards infinite-frames 3d detection with an efficient framework. In *ECCV*, pages 193–209. Springer, 2022. 2, 3, 6, 7

[49] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. In *Sensors*, 2018. 2

[50] Bin Yang, Min Bai, Ming Liang, Wenyuan Zeng, and Raquel Urtasun. Auto4d: Learning to label 4d objects from sequential point clouds. *arxiv preprint*, 2021. 1, 2, 3

[51] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[52] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[53] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 2

[54] Zetong Yang, Yin Zhou, Zhifeng Chen, and Jiquan Ngiam. 3d-man: 3d multi-frame attention network for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 1, 7

[55] Zetong Yang, Yin Zhou, Zhifeng Chen, and Jiquan Ngiam. 3d-man: 3d multi-frame attention network for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1863–1872, 2021. 3

[56] Dongqiangzi Ye, Zixiang Zhou, Weijia Chen, Yufei Xie, Yu Wang, Panqu Wang, and Hassan Foroosh. Lidarmultinet: Towards a unified multi-task network for lidar perception. *arxiv preprint*, 2022. 1, 6

[57] Maosheng Ye, Shuangjie Xu, and Tongyi Cao. Hvnet: Hybrid voxel network for lidar based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[58] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 2, 3, 7

[59] Wu Zheng, Weiliang Tang, Sijin Chen, Li Jiang, and Chi-Wing Fu. Cia-ssd: Confident iou-aware single-stage object detector from point cloud. *arXiv preprint arXiv:2012.03015*, 2020. 6

[60] Yin Zhou, Pei Sun, Yu Zhang, Dragomir Anguelov, Jiyang Gao, Tom Ouyang, James Guo, Jiquan Ngiam, and Vijay Vasudevan. End-to-end multi-view fusion for 3d object detection in lidar point clouds. In *Conference on Robot Learning*, 2020. 2, 7

[61] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[62] Zixiang Zhou, Xiangchen Zhao, Yu Wang, Panqu Wang, and Hassan Foroosh. Centerformer: Center-based transformer for 3d object detection. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVIII*, pages 496–513. Springer, 2022. 2, 7

[63] Benjin Zhu, Zhe Wang, Shaoshuai Shi, Hang Xu, Lanqing Hong, and Hongsheng Li. Conquer: Query contrast voxel-detr for 3d object detection. *arXiv preprint arXiv:2212.07289*, 2022. 2