Towards Stable 3D Object Detection

Jiabao Wang[1⋆], Qiang Meng[2⋆], Guochao Liu[2], Liujiang Yan[2], Ke Wang[2], Ming-Ming Cheng[1,3], and Qibin Hou[1,3†]

[1] VCIP, College of Computer Science, Nankai University
[2] KargoBot Inc., China
[3] NKIARI, Shenzhen Futian
https://github.com/jbwang1997/StabilityIndex

Abstract. In autonomous driving, the temporal stability of 3D object detection greatly impacts the driving safety. However, the detection stability cannot be accessed by existing metrics such as mAP and MOTA, and consequently is less explored by the community. To bridge this gap, this work proposes Stability Index (SI), a new metric that can comprehensively evaluate the stability of 3D detectors in terms of confidence, box localization, extent, and heading. By benchmarking state-of-the-art object detectors on the Waymo Open Dataset, SI reveals interesting properties of object stability that have not been previously discovered by other metrics. To help models improve their stability, we further introduce a general and effective training strategy, called Prediction Consistency Learning (PCL). PCL essentially encourages the prediction consistency of the same objects under different timestamps and augmentations, leading to enhanced detection stability. Furthermore, we examine the effectiveness of PCL with the widely-used CenterPoint, and achieve a remarkable SI of 86.00 for vehicle class, surpassing the baseline by 5.48. We hope our work could serve as a reliable baseline and draw the community's attention to this crucial issue in 3D object detection.

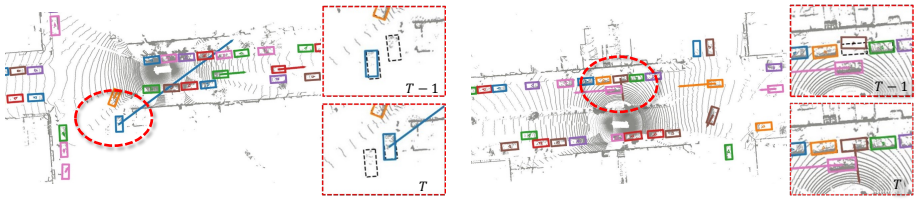
Keywords: 3D Object Detection · Temporal Stability

1 Introduction

3D object detection aims to perceive objects of interest within the surrounding environment, utilizing data from diverse sources such as point clouds [12, 18, 33, 42, 45, 49], camera images [21, 39], multi-sensors [8, 22, 26], *etc*. Serving as a foundational component in autonomous driving, this task has attracted great attention from both academia and industry. Numerous performant detectors [4, 16, 20, 38, 39, 43, 48, 50] have been proposed recently, significantly advancing the development of 3D object detection.

Counterintuitively, it is rather common for highly performant detectors to exhibit instability. Sensor noise, model sensitivity, slight scene changes, and non-

⋆ Equal contribution.
† Corresponding author.



(a) confidence variations *(b) geometric variations*

Fig. 1: Visualizations of potential safety threats caused by detection instability. On the left, confidence fluctuations lead to flickering boxes, which results in inaccurate object association and induces an abnormal velocity estimation. On the right, an intent of merging into traffic is erroneously forecast because of the shaking boxes, though the vehicle is stationary in fact. Here, dashed boxes represent the ground truths. Detection results are predicted by [42], and object tracking is conducted with SimpleTrack [25].

deterministic operators, all contribute to detection instability. Despite great advancements, current state-of-the-art detectors predominantly emphasize improving *single-shot* detection accuracy, while often neglecting such *temporal* stability.

Detection stability encompasses more than mere robustness; it extends to the broader context of ensuring human safety in autonomous driving. As exemplified in Fig. 1, unstable detections, on both confidence scores and bounding boxes, can result in abnormal velocity estimated by tracking. These erroneous estimations may trigger false judgement on the behaviors of surrounding agents, potentially misleading the ego-vehicle to make improper or even hazardous decisions. In addition, systematically complementing poor detection stability requires extra modules (*e.g.*, Kalman filters [3,5,41] with carefully and usually manually tuned parameters). This not only increases system complexity and latency, but also necessitates tedious engineering efforts. As a conclusion, enhancing detection stability is a crucial step towards safe and reliable autonomous driving.

To the best of our efforts, we find no prior work dedicating on detection stability for 3D object detectors. One primary reason is the absence of an appropriate metric to quantify such stability. Current metrics in measuring detection accuracy, such as mAP [14], usually overlook temporal information, which is fundamental for stability assessment. On the other hand, metrics designed for temporal object tracking (*e.g.*, MOTA and MOTP [2]) are tailored to evaluate how well objects are tracked over time. Trackers are designed to be robust with respect to detection noises. A well-implemented tracking algorithm will certainly hide instabilities of upstream detectors. As illustrated in Fig. 1(b), although the detector produced inconsistent yaw and positions across two frames, trackers can still associate the two boxes and fuse the well-behaving box extent information.

We argue a new metric is needed for detection stability. For this purpose, we initiate a comprehensive analysis of the task, identifying that an effective metric should exhibit four core properties: *1) Comprehensiveness*: The metric must take all detected attributes into account. *2) Homogeneity*: All attributes should be uniformly integrated into the metric. *3) Symmetry*: The metric should be consistent regardless of the input order. *4) Marginal Unimodality*: The metric

value will never increase as the stability of any element deteriorates. Based on our analysis, we accordingly propose a novel metric called Stability Index (SI), which evaluates stability by quantifying the temporal consistency in terms of the confidence score, box location, extent, and heading. Through our meticulously designed schemes, the proposed SI fully complies with all the aforementioned requirements, as demonstrated by our rigorous theoretical proofs.

On the large-scale Waymo Open Dataset (WOD) [35], we thoroughly benchmark various popular 3D object detectors and observe that there is no evident correlation between existing metrics (*e.g.*, mAP and MOTA) and our proposed stability metric SI. Furthermore, our experiments reveal that some effective tricks in object detection, like using more augmentations and multi-frame strategies, fail to yield many improvements in terms of stability.

To this end, we additionally introduce a framework called Prediction Consistency Learning (PCL), which in essence penalizes prediction errors' discrepancies from the same objects under different timestamps and augmentations. It's noteworthy that our PCL is a general framework applicable to all detectors, and it introduces no additional cost during inference. Without bells and whistles, PCL boosts the SI of CenterPoint [45] from 80.52 to an impressive 86.00 SI on the vehicle class, surpassing all state-of-the-art detectors.

We summarize our contributions as follows:

1. For the first time, we provide a comprehensive analysis of detection stability. Subsequently, we introduce the Stability Index (SI) metric, which uniformly evaluates and positively indicates the stability of all detection elements. Rigorous theoretical proofs are further presented to validate the efficacy of SI.
2. A general framework termed Prediction Consistency Learning (PCL) is proposed to boost detection stability. Extensive experiments on the Waymo Open Dataset unearth several intriguing insights of object stability as well as demonstrate the effectiveness of the PCL.

2 Related Work

2.1 3D Object Detection

3D object detection, a fundamental building block in autonomous driving, focuses on accurately locating objects within a three-dimensional space. Prior works in this domain can be broadly categorized based on input modalities.

Most existing LiDAR-based works [12,13,15,18,19,33,42,45,49,50] transform non-uniform point clouds into regular 2D pillars or 3D voxels, and employ convolutions for efficient processing in later stages. Beyond voxel-based methods, the task can also be accomplished using alternative point cloud representations, including the range-view [4,7,16,24,36], point-view [9,20,27,28,31,32,34,43,44,46,48], and their combinations [29,36]. Several studies [23,35,38] introduce recently popular Transformer architectures and achieve remarkable detection accuracies.

Transformer architectures also demonstrate great success in transforming camera images into bird's-eye-view features. Such algorithmic breakthrough paved

the way for vision [21,39] and fusion [8,22,26] based 3D detection for self-driving vehicles. It's noteworthy that our proposed metric and method are agnostic to input modalities, thus applicable to all 3D object detection methods.

2.2 Related Metrics

Properly measuring performances is crucial for any machine learning task, let alone 3D object detection. The KITTI [17] dataset plays a pioneering role in evaluating autonomous driving tasks, employing the well-established average precision (AP) as metric. Waymo Open Dataset [35] further extends the metric into APH by accounting for heading errors. In contrast, nuScenes [6] questions the suitability of IOU-based metrics for vision-only methods, which usually come with large localization errors. Therefore, a new metric called NDS is proposed to assess error-prone predictions by utilizing a thresholded 2D center distance.

Multi-object tracking (MOT), the downstream task of object detection, stands as another critical component for autonomous driving. Bernardin and Stiefelhagen [2] introduces metrics of MOTA and MOTP, where MOTA combines errors including False Negatives, False Positives, and Identity Switches, while MOTP focuses on how good sequences overlap with ground truths. Weng *et al.* [40] points out that both metrics do not take scores into account, and extends them into AMOTA and AMOTP by averaging scores across different recall levels. In general, detection metrics disregard temporal relationships of detected boxes, whereas tracking metrics mainly focus on whether objects are correctly associated across frames. Previous methods fall short in capturing detection stability across frames, which serves as the key motivation behind this work.

3 Methodology

In this section, we first comprehensively analyze the stability in 3D object detection. Based on our analysis, we introduce a novel metric called Stability Index(SI) and prove its key properties. In the end, we introduce our Prediction Consistency Learning (PCL) to enhance detection stability.

3.1 Notations

A valid prediction P from 3D object detectors comprises a confidence score c and a 3D bounding box defined as $B = (x, y, z, l, w, h, \theta)$. Here, (x, y, z) are the coordinates of the box center, while (l, w, h) denote the box extent, and θ represents the yaw angle. *Elements* and *attributes* are used interchangeably to refer to the box properties.

Given two boxes B_1 and B_2, we define a transformation function $T_{B_1 \to B_2}(\cdot)$ which represents the mapping from B_1 to B_2. Consequently, we can apply this customized transformation to an arbitrary box B, resulting in $\hat{B} = T_{B_1 \to B_2}(B)$,

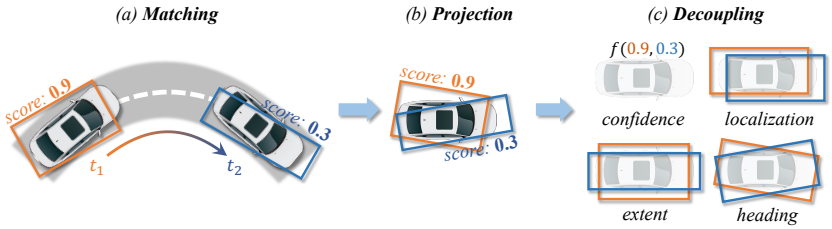


Fig. 2: The procedure of computing Stability Index. The orange and blue boxes represent the best matches between the predictions and the ground-truths searched by the Hungarian algorithm. These boxes are subsequently associated across frames using their object ID labels. After projecting predictions into a pre-built pivot box, SI decouples them into element-wise computations, which are then aggregated for the final assessment of detection stability.

where

$$\begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix} = \begin{pmatrix} \cos(\theta_2 - \theta_1) & \sin(\theta_2 - \theta_1) & 0 \\ -\sin(\theta_2 - \theta_1) & \cos(\theta_2 - \theta_1) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x - x_1 \\ y - y_1 \\ z - z_1 \end{pmatrix} + \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix},$$
$$\begin{pmatrix} \hat{l} \\ \hat{w} \\ \hat{h} \end{pmatrix} = \begin{pmatrix} l_2/l_1 \times l \\ w_2/w_1 \times w \\ h_2/h_1 \times h \end{pmatrix}, \qquad \hat{\theta} = \theta + (\theta_2 - \theta_1).$$

In essence, this operation transforms discrepancies between B_1 and B into those between B_2 and $\hat{B}$.

3.2 Analysis of Detection Stability

In the context of autonomous driving, variations in any of the predicted attributes from detectors may result in hazardous situations. For instance, fluctuations in box locations and heading may lead to inaccurate velocity estimations, potentially leading to unsafe interaction decisions. Unstable confidence scores may cause flickering predictions and hinder the autonomous driving system from accurately tracking objects. Moreover, erratic predictions of a nearby vehicle's size may prompt the ego-vehicle to take improper evasive maneuvering. In summary, the stability of all detection elements must be comprehensively taken into account to ensure the safety of autonomous driving.

A naive approach for assessing stability is to sum variations of all these elements, essentially extending Zhang and Wang [47] in 2D video detection. However, these variations should not be directly added as these detection attributes represent different physical properties of the object. Moreover, element variations are agnostic of the object properties and therefore fail to capture the hazard levels caused by unstable predictions. For example, large jitters on the yaw angle can lead to rapid changes in object behaviors for large-volume objects (*e.g.*, vehicles). In contrast, pedestrians suffer more from the instability of center

offsets and box dimensions rather than headings. Therefore, how to standardize these elements into a single and consistent unit remains a challenging problem.

One possible way to unify physical units of the box-related elements is to adopt the Intersection-over-Union (IoU) as used in the mAP metric. To achieve this, we begin by assessing detection stability at the smallest unit, involving a single object at two timestamps t_1, t_2 as illustrated in Fig. 2. Denote the ground-truth boxes as B_1^g, B_2^g and the predictions are $P_i = \{c_i, B_i\}, i \in \{1, 2\}$. The IoU between the two predicted 3D boxes cannot be directly computed due to object movement. In contrast, our pre-defined operation enables the measurement by projecting boxes onto one of the ground-truths. For example, we can project B_1 into the second ground-truth as $\hat{B}_1 = T_{B_1^g \to B_2^g}(B_1)$ and compute $\text{IoU}(\hat{B}_1, B_2)$. This IoU can reflect the detection stability to some extent. Nevertheless, this measurement has two significant flaws (proved by Properties 1 and 2 in the supplementary): (1) IoU varies with the order of frames, *i.e.*, $\text{IoU}(\hat{B}_1, B_2) \neq \text{IoU}(B_1, \hat{B}_2)$. (2) IoU is not marginal unimodal. In other words, enhancing the stability of an element can, at times, lead to a poorer IoU value. Both flaws prohibit IoU from serving as an effective assessment of detection stability.

Through the detailed analysis of stability and exploration of potential solutions, we identify four key properties that an effective metric should meet:

- ***Comprehensiveness***: The metric should comprehensively reflect influences from all relevant detection elements.
- ***Homogeneity***: Influences caused by all elements should be well-processed into unified physical units.
- ***Symmetry***: The metric values should be consistent when applied to both forward and reverse inputs.
- ***Marginal Unimodality***: For each element with others fixed, the metric should be unimodal w.r.t. its stability.

3.3 Stability Index

While the IoU is a promising starting point, meeting the four properties demands careful designs to effectively integrate the confidence score and address the asymmetry and non-unimodality design flows. To this end, we introduce schemes of projection with pivot boxes, element decoupling, and stability aggregation, as illustrated in Fig. 2. Ultimately, we assess the stability of object pairs in consecutive frames and denote the metric as Stability Index (SI).

Projection with pivot box. Since projections onto either of ground-truths can introduce the asymmetry issue, we therefore propose to cast predictions onto an intermediary pivot box $B^p = (0, 0, 0, l^p, w^p, h^p, 0)$. Here, we leverage geometric averages $l^p = \sqrt{l_1^g l_2^g}$, $w^p = \sqrt{w_1^g w_2^g}$, $h^p = \sqrt{h_1^g h_2^g}$ to ensure that the pivot box's dimensions closely match those of the ground-truths B_1^g, B_2^g. This is crucial for accurate stability measurements, as objects of different sizes are affected by fluctuations to varying degrees. Finally, we have $\hat{B}_1 = T_{B_1^g \to B^p}(B_1), \hat{B}_2 = T_{B_2^g \to B^p}(B_2)$ as indicated in Fig. 2(b).

Element decoupling. For the marginal unimodality, the metric must exhibit the following two characteristics when all elements are fixed except for one arbitrary element: (1) The metric reaches the peak value if and only if the element is stable. (2) The metric value is monotonically non-decreasing as the stability of an element deteriorates in any continuous direction. We recognize that IoU fails to meet these characteristics due to the mutual interference between elements, and therefore propose to decouple them into four distinct parts as shown in Fig. 2(c). For instance, to measure the localization stability, we make elements except for box centers in $\hat{B}_1, \hat{B}_2$ to be identical. Specifically, we replace them with those from the pivot box, resulting in $\hat{B}_i^{loc} = (\hat{x}_i, \hat{y}_i, \hat{z}_i, l^p, w^p, h^p, 0), i \in \{1, 2\}$. Similarly, we can have $\hat{B}_i^{ext}, \hat{B}_i^{hdg}, i \in \{1, 2\}$ for the box extent and heading. Then, we assess the stability in box localization and extent by the two equations:

$$SI_l = \mathrm{IoU}(\hat{B}_1^{loc}, \hat{B}_2^{loc}), \quad SI_e = \mathrm{IoU}(\hat{B}_1^{ext}, \hat{B}_2^{ext}). \tag{1}$$

Directly employing $\mathrm{IoU}(\hat{B}_1^{hdg}, \hat{B}_2^{hdg})$, however, violates the unimodality if the angle difference between $\hat{\theta}_1$ and $\hat{\theta}_2$ exceeds $\pi/4$ (proved in Lemma 3 in our supplementary). Therefore, we regard this case as a failure and explicitly set the metric to be 0. The stability in box heading finally is

$$SI_h = \begin{cases} 0, & \text{if } |\hat{\theta}_1 - \hat{\theta}_2| \geq \pi/4, \\ \mathrm{IoU}(\hat{B}_1^{hdg}, \hat{B}_2^{hdg}), & \text{otherwise.} \end{cases} \tag{2}$$

The stability in confidence can be captured by the difference between the scores c_1, c_2, *i.e.*, using $1 - |c_1 - c_2|$. A remaining issue is that this function is vulnerable to intrinsic confidence scales of object detectors. For example, if all scores are divided by a scaling factor, the detection performance and stability should remain unaffected. However, the value of $1 - |c_1 - c_2|$ would increase, leading to an inaccurate measurement of stability. To address this issue, we calculate 99% and 1% percentile of all confidences as $c^{0.99}$ and $c^{0.01}$. The confidence stability is then calibrated by

$$SI_c = \max\left(0, 1 - |c_1 - c_2| / (c^{0.99} - c^{0.01})\right). \tag{3}$$

Stability aggregation. In the last step, we aggregate stability from all components using the following formulation:

$$SI = SI_c \times (SI_l + SI_e + SI_h)/3. \tag{4}$$

Here, $SI_c \in [0, 1]$ is treated as the weight of the box stability. SI_l, SI_e, SI_h can be averaged thanks to the same unit of IoU. In the end, SI successfully satisfies the four properties of a valid stability evaluator according to Lemmas 1 and 2. Detailed analyses and theoretical proofs are available in our supplementary.

Lemma 1. *SI is a symmetric metric which uniformly assesses all elements' influences on the detection stability.*

Lemma 2. *SI is marginal unimodality w.r.t. all elements. The maximum value of 1 is reached if and only if the detection is perfectly stable across frames.*

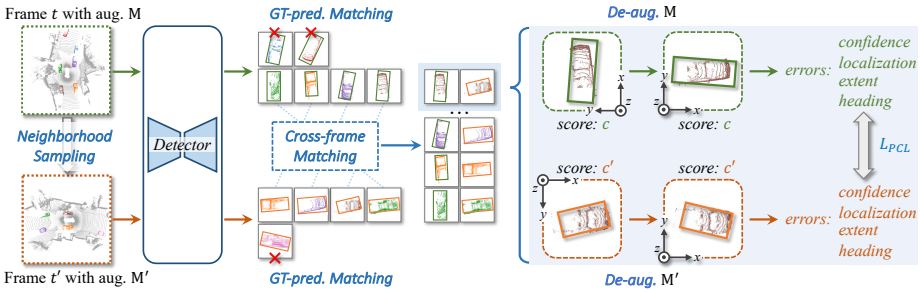


Fig. 3: The pipeline of the proposed Prediction Consistency Learning (PCL). In each iteration, PCL samples a pair of frames at neighboring timestamps t and t', and applies augmentations $\mathbf{M}$ and $\mathbf{M}'$ to the paired samples. GT-prediction matching and cross-frame matching then collaboratively associate the detector's predictions from the same objects between the two frames. After the de-augmentation procedure, PCL calculates the prediction errors in terms of confidence, localization, extent, and heading, which are defined in the object self-coordinate system. Finally, PCL penalizes the error disparities among all prediction pairs to enforce the temporal consistency. In the figure, pred. and aug. represent prediction and augmentation, respectively.

Our previous discussions focus on the smallest set, consisting of a single object at two consecutive timestamps. To assess SI for large-scale benchmarks, we begin by pairing each ground-truth with a prediction using the Hungarian algorithm. With the labeled object IDs, we segment the evaluation into calculating SI for numerous smallest sets. The final result is simply the average of all values. More details like the handling of corner cases are presented in the supplementary.

3.4 Prediction Consistency Learning

Beyond the design of the metric, we further attempt to boost the detection stability of 3D object detectors. For this purpose, we introduce a general and effective training strategy named Prediction Consistency Learning (PCL), as illustrated in Fig. 3. Our PCL is built on the core idea of encouraging prediction consistency across frames under various augmentations and timestamps. It consists of four key stages: neighborhood sampling, prediction pairing, de-augmentation, and prediction consistency loss.

Neighborhood sampling. For each frame F with timestamp t, we begin by uniformly sampling an integer Δt from the range $[-n, n]$, where n is a pre-defined parameter. Subsequently, we get the frame F' at timestamp $t + \Delta t$ and bundle F, F' as a pair-wise input for the network. The frames are further augmented separately by random flipping, rotation, and scaling. We record the augmentations into matrices $\mathbf{M}$ and $\mathbf{M}'$, where $\mathbf{M}$ can be described as follows:

$$\mathbf{M} = \begin{pmatrix} i_x & 0 & 0 \\ 0 & i_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot s. \tag{5}$$

Here, i_x and i_y indicate whether the frames are x and y direction flipped, with -1 meaning the corresponding flipping occurs and 1 otherwise. α is the angle applied by random rotation, and s denotes the factor for random scaling.

Prediction pairing. After the detector generating predictions from paired samples, our next step is to gather the corresponding predictions for comparisons. We first perform the GT-prediction matching to assign each ground-truth box with the best-matched prediction, which can be accomplished by the Hungarian algorithm or any other rational method. Subsequently, cross-frame matching associates predictions between two frames by corresponding object IDs and creates prediction pairs for later comparisons.

De-augmentation. Data augmentation used during training can largely alter the patterns of detection errors, impeding fair comparisons of predictions in each pair. For example, random scaling can scale up the errors in box locations and extents, while random flipping may change the error direction. Therefore, we apply a de-augmentation step on each prediction to eliminate the influences of augmentations. For a prediction $P = \{c, x, y, z, l, w, h, \theta\}$, we recover it into $\bar{P} = \{\bar{c}, \bar{x}, \bar{y}, \bar{z}, \bar{l}, \bar{w}, \bar{h}, \bar{\theta}\}$ with the corresponding $\mathbf{M}$:

$$\begin{cases} \bar{c} = c, \\ (\bar{x}, \bar{y}, \bar{z})^T = \mathbf{M}^{-1}(x, y, z)^T, \\ (\bar{l}, \bar{w}, \bar{h})^T = (l, w, h)^T / s, \\ \bar{\theta} = i_x \cdot i_y \cdot (\theta - \alpha). \end{cases} \tag{6}$$

Prediction consistency loss. Before introducing the consistency loss, we first compute prediction errors for a de-augmented prediction $\bar{P}$ with respect to the ground-truth box B^g. We define the error for confidence as $e_c = 1 - \bar{c}$. Prediction errors in box localization, extent, and heading are computed in the object's ego-coordinate system. Specifically, the error for box center is calculated by

$$\mathbf{e}_l = \begin{pmatrix} \cos\theta^g & \sin\theta^g & 0 \\ -\sin\theta^g & \cos\theta^g & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \bar{x} - x^g \\ \bar{y} - y^g \\ \bar{z} - z^g \end{pmatrix}. \tag{7}$$

The prediction error for the box extent is formulated as

$$\mathbf{e}_e = \left(\bar{l}/l^g, \bar{w}/w^g, \bar{h}/h^g\right)^T. \tag{8}$$

In the end, the error $\mathbf{e}_h$ for box heading is encoded into trigonometric vectors:

$$\mathbf{e}_h = \left(\sin(\bar{\theta} - \theta^g), \cos(\bar{\theta} - \theta^g)\right)^T. \tag{9}$$

Our final step is to encourage each prediction pair to reveal similar patterns in terms of prediction errors. Thereby, we collect the pair-wise errors $\{e_{c,i}, e'_{c,i}\}$, $\{\mathbf{e}_{l,i},\ \mathbf{e}'_{l,i}\}$, $\{\mathbf{e}_{e,i},\ \mathbf{e}'_{e,i}\}$, and $\{\mathbf{e}_{h,i}, \mathbf{e}'_{h,i}\}$ for $i \in \{1, 2, \cdots, N\}$, where N is the number of successfully associated objects between frames F and F'. In the end, our prediction consistency loss is:

$$\begin{aligned} L_{PCL} = & \frac{1}{N} \sum_{i=1}^{N} \big(w_1 \cdot \mathrm{MSE}(e_{c,i}, e'_{c,i}) + w_2 \cdot L_1(\mathbf{e}_{l,i}, \mathbf{e}'_{l,i}) \\ & + w_3 \cdot L_1(\mathbf{e}_{e,i}, \mathbf{e}'_{e,i}) + w_4 \cdot L_1(\mathbf{e}_{h,i}, \mathbf{e}'_{h,i}) \big). \end{aligned} \tag{10}$$

Table 1: Benchmarks on Waymo Open Dataset. Models are sorted based on mAPH on the class vehicle. We use two different intensities of colors to highlight the highest and second-highest results in each column. "†" denotes the model is not LiDAR-based only. CenterPoint*[45] represents the pillar version of CenterPoint.

Methods	Vehicle(%)						Pedestrain(%)						Cyclist(%)					
	mAPH	SI	SI_c	SI_l	SI_e	SI_h	mAPH	SI	SI_c	SI_l	SI_e	SI_h	mAPH	SI	SI_c	SI_l	SI_e	SI_h
Second [42]	72.60	81.37	90.2	84.2	92.0	92.2	59.81	63.07	83.9	69.6	87.8	67.6	61.95	67.21	81.1	76.1	88.3	83.8
CenterPoint*[45]	72.82	80.61	89.0	85.4	91.0	92.8	65.28	64.57	83.2	74.4	87.4	68.9	65.87	68.06	80.8	77.7	87.0	85.9
Pointpillar [18]	72.84	80.84	89.6	84.4	92.3	91.6	54.64	62.03	84.7	72.1	88.8	57.9	59.51	66.14	82.2	74.9	88.0	77.4
CenterPoint [45]	73.73	80.52	89.0	85.3	90.7	92.9	69.50	68.40	85.7	73.3	88.6	75.0	71.04	68.40	80.3	78.5	87.4	89.8
PartA2Net [33]	75.02	82.86	91.4	85.4	91.7	91.7	66.16	65.08	84.6	73.6	86.7	67.0	67.90	72.73	85.9	79.3	87.0	84.3
PV R-CNN [29]	75.92	83.73	91.9	86.4	92.3	91.7	66.28	66.17	86.0	73.5	87.4	66.6	68.38	73.53	86.8	78.9	88.4	83.2
Voxel R-CNN [12]	77.19	84.26	92.0	86.7	92.1	93.3	74.21	69.50	86.9	75.3	88.1	73.6	71.68	73.23	84.4	80.1	87.7	89.3
VoxelNext [10]	77.84	84.82	92.9	86.3	91.6	94.2	76.24	74.74	92.7	75.7	88.0	75.8	75.59	76.48	90.0	79.2	84.9	87.8
PV R-CNN$^+$[30]	77.88	84.49	92.1	87.2	92.4	93.2	73.99	69.27	86.8	75.3	88.1	73.2	71.84	73.05	84.2	80.3	87.7	89.2
DSVT [38]	78.82	84.90	92.5	86.9	91.5	94.8	76.81	74.58	91.9	76.5	88.7	75.9	75.44	76.20	88.2	80.5	86.1	89.9
TransFusion† [1]	79.00	82.32	89.3	86.8	92.7	95.7	76.52	69.11	84.5	75.4	89.9	78.8	70.11	70.35	80.6	79.5	90.6	91.1

Here w_1, w_2, w_3, and w_4 are weights to balance different parts in our loss, which being 1 if not specified. MSE and L_1 are losses of mean square error and L_1 distance, respectively. Both original detection losses and our prediction consistency loss are leveraged to train the object detectors.

4 Experiments

4.1 Benchmark on the Waymo Open Dataset

Implementation details. We replicate commonly used LiDAR-based and fusion-based 3D detectors on top of OpenPCDet [37] and MMDetection3D [11]. All detectors are trained on Waymo Open Dataset (WOD) [35] with default configurations. Our training uses the full version of the training set, consisting of 798 sequences with 158,361 samples. We evaluate these models with the LEVEL 1 mAP weighted by Heading accuracy (mAPH) and the proposed SI on the validation set, which contains 202 sequences with 40,077 samples. Besides the SI, we further present its sub-indicators of stability on confidence (SI_c), localization (SI_l), extent (SI_e), and heading (SI_h).

Relation between SI and mAPH. Tab. 1 presents model results on categories of vehicle, pedestrian, and cyclist. Models are sorted by the mAPH on the class vehicle, and we highlight the two best performing models in each column. From the results, we find that there is no evident correlation between detection accuracy and model stability. For instance, TransFusion has the highest mAPH on the class vehicle while its SI is much lower than the LiDAR-based counterparts with similar detection metrics. That could be because the fusion model improves detection accuracy by additional information from camera images. The visual information, however, is indirect in inferring precise 3D locations, thereby increasing the detection uncertainty. On the other hand, CenterPoint achieves

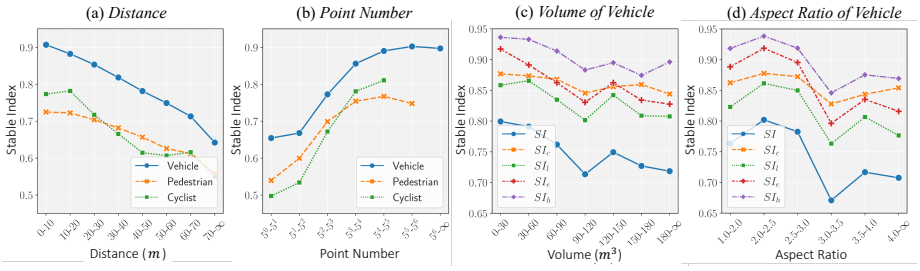


Fig. 4: Relationships between object properties and detection stability.

Table 2: Effects of multi-frame strategy on the detection stability.

Methods	Number of frames	Vehicle(%)		Pedestrian(%)		Cyclist(%)	
		mAPH	SI	mAPH	SI	mAPH	SI
CenterPoint	1	73.73	80.52	69.50	68.40	71.04	68.40
	2	75.04	80.86	75.17	70.40	71.23	69.39
	4	75.85	81.74	75.38	71.69	71.68	69.93
PV R-CNN	1	78.33	85.17	75.75	70.15	72.47	73.31
	2	79.62	86.39	80.37	73.79	73.66	76.78
	4	80.51	87.50	81.12	75.32	74.77	76.34

73.73 mAPH for vehicle detection, higher than Second and PointPillar. But it has the lowest SI of 80.52 among all detectors. These results negate definitive positive relations between the two metrics.

Influence of object properties. Fig. 4 shows how various object properties affect detection stability. We group objects based on specified properties and detect them with CenterPoint. Fig. 4(a) presents a negative relationship between detection stability and object distance, where longer distances correspond to harder objects to learn in general. For all classes, SI increases with the number of object points and becomes saturated when the point number reaches 5^3, as demonstrated in Fig. 4(b). Fig. 4 (c) and (d) further explore the effects of object volumes and length-to-width ratios for vehicles. We find that small vehicles tend to have more stable detection. Vehicles with length-to-width ratios between 2 and 3 exhibit relatively high SI values. This may be attributed to the prevalence of such vehicles in real-world scenarios. Vehicles with larger length-to-width ratios, such as trucks/trams/buses, are relatively scarcer in the dataset, and require larger receptive field requirements, making them more unstable in detection.

Effects of multi-frame strategy. Merging several consecutive point clouds as one input is a commonly used strategy to address the sparsity in LiDAR data. Tab. 2 reveals that this scheme not only improves model accuracy but also benefits detection stability. Taking the vehicle for example, using four frames results in notable improvements in the detection accuracy of CenterPoint and PV R-CNN, reaching 75.85 and 80.51 mAPH, which surpass the baseline by 2.12 and 2.18 mAPH. Meanwhile, the values of SI for CenterPoint and PV R-

Table 3: The effects of the proposed PCL. "-" is the base model and "w/o PCL" represents the model fine-tuned without prediction consistency loss.

Methods	Vehicle(%)		Pedestrian(%)		Cyclist(%)	
	mAPH	SI	mAPH	SI	mAPH	SI
-	73.73	80.52	69.50	68.40	71.04	68.40
w/o PCL	73.70	80.93	69.55	68.35	71.27	68.20
PCL ($n = 0$)	75.57	85.42	70.18	71.87	70.86	68.80
PCL ($n = 4$)	75.26	85.83	69.56	72.76	70.65	69.22
PCL ($n = 8$)	75.04	85.94	68.82	72.87	70.31	69.32
PCL ($n = 12$)	74.64	85.93	68.50	72.95	70.85	69.33
PCL ($n = 16$)	74.54	86.00	67.82	73.14	70.25	69.16

CNN are increased by 1.22 and 2.33, respectively. This trend is consistent for all classes, illustrating the general effectiveness of the strategy in boosting detection performances of both accuracy and stability.

Summary. Our experiments verify that the proposed SI is a complementary metric to detection accuracy. The metric value varies a lot for different model types and demonstrates several interesting patterns w.r.t. object properties. We also examine two common-used schemes including data augmentation (in the supplementary) and the multi-frame strategy. Increasing the degree of data augmentation has a minor impact on detection stability. Though using multi-frames is proven to be beneficial, it places heavy computational overhead during encoding data into voxel features. In contrast, our proposed PCL introduces no additional computations during inference, while significantly improving detection stability as illustrated by the later experiments.

4.2 Experiments on PCL

Implementation details. We employ the widely-used CenterPoint as our base model, which is trained with the default setting in OpenPCDet. Specifically, we train the model for 36 epochs with the Adam optimizer. The one-cycle policy with an initial learning rate 0.003 is used. The learning rate gradually increases to 0.03 in the first 40% epochs and then gradually decreases in the rest of training.

Instead of end-to-end training, we choose to fine-tune the base model with PCL equipped for a few epochs. Training configuration mirrors that of the end-to-end one, with the exception that the epoch number is reduced to 5 and the learning rate is divided by 10. It's noteworthy that the scheme not only highly reduces training cost, but also shows how effectively PCL can take effect.

Effectiveness of PCL. We compare the performances of models fine-tuned with and without PCL, as shown in Tab. 3. It can be observed that directly fine-tuning the model has a marginal impact on both model accuracy and stability. In contrast, when using PCL without cross-frame information involved (*i.e.*, $n = 0$), we already achieve SI values of 84.54, 70.95, and 68.80 for vehicle, pedestrian, and cyclist, respectively. These results reveal significant enhancements, with gains of +4.49, +3.52, and +0.60 compared to the baseline. For the mAP, we find an

Table 4: Results (%) on vehicle class with different components in PCL. “C", “L", “E", and “H" denote applying the loss parts relative to confidence, localization, extent, and heading, respectively.

Components				mAPH	SI	SI_c	SI_l	SI_e	SI_h
C	L	E	H						
				73.70	80.93	88.90	85.90	91.50	93.64
✓				75.64	84.04	92.28	85.80	91.44	93.65
	✓			74.15	81.80	89.34	87.16	91.79	93.73
		✓		73.86	81.73	89.14	86.13	93.37	93.67
			✓	73.44	80.89	88.88	85.64	91.47	93.85
✓	✓	✓	✓	75.57	85.42	92.81	86.78	93.31	93.90

interesting phenomenon: the mAP of three classes changes by +1.87, +0.63, and -0.39. That leads to two valuable conclusions: (1) Our PCL not only enhances stability but also improves the overall detection accuracy, particularly for the vehicle class. (2) Regardless of how the mAP changes, the SI is consistently improved, reinforcing that these two metrics assess different model attributes.

Effects of the interval between frame pairs. A key hyper-parameter of PCL is the maximum interval n between a pair of frames, as described in the neighborhood sampling. The larger n becomes, the longer the spans between two frames contrasted by PCL will be. The results in Tab. 3 show opposing trends in the detection accuracy and stability with n changes. Take the class vehicle as an example. When $n = 0$, we have the highest mAP of 75.57 and the lowest SI of 85.42 among all PCL models. The mAP eventually drops to 74.54 as n grows. On the contrary, the model stability gradually rises to 86.00 SI with n being 16. This may be because object morphology can change considerably when the frame interval n grows. Forcefully aligning them can bring damage to model accuracy. However, such alignment promotes consistent predictions for the same objects, which subsequently leads to stable detection.

Effects of loss components in PCL. Our introduced consistency loss comprises components of confidence, localization, extent, and heading. We examine the model performances with various combinations of these loss components and report the results in Tab. 4. These results show that each part of the loss in PCL can boost the model stability from the corresponding aspect, which confirms the effectiveness of each component in PCL. The detection accuracy and stability are the highest with all loss parts involved.

The loss component related to confidence score yields the highest improvements in the final SI. This may be because that the classification loss for training detectors primarily focuses on whether an object is correctly classified, leaving sufficient room for enhancing consistency. In contrast, box parameters already have a latent potential for consistent predictions as they all use ground-truth labels as the targets. Enforcing consistency on these parameters is not as influential as it is on the confidence score. Furthermore, we observe that the loss associated with the heading component leads to the least improvement, indicating that maintaining consistency in heading is a challenging task.

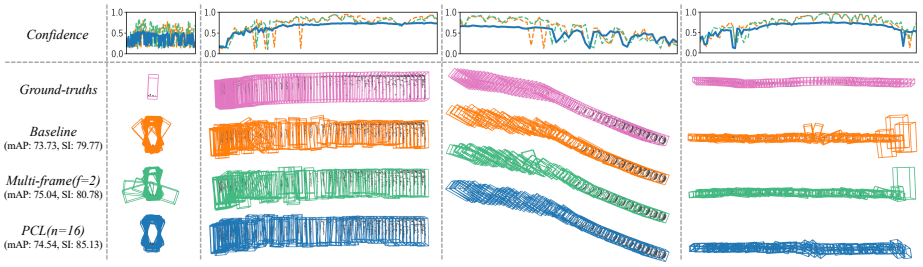


Fig. 5: Visualizations of ground-truths (in pink) and predictions of CenterPoint models trained by the baseline (in orange), multi-frame strategy (in green), and PCL strategy (in blue). Predicted confidences (top row) and 3D boxes (bottom row) are all presented.

Visualization. In Fig. 5, we present visualizations of a few ground-truth data and detection results from three distinct models: the baseline CenterPoint, CenterPoint with 2 frames as input, and our PCL model with $n = 16$. In the first row, we plot the trends in confidence scores with time changes and find the PCL model exhibits superior capability in suppressing confidence score fluctuations compared to the other two models. For the predicted boxes, PCL also has more stable results than other models. It's noteworthy that our PCL model, despite having a lower mAP compared to the multi-frame version of CenterPoint, significantly outperforms it in terms of SI. This further verifies that detection accuracy and stability capture independent aspects of model performance. These phenomena all demonstrate the effectiveness of PCL in enhancing detection stability.

5 Conclusions and Limitations

In this work, we comprehensively study a critical but overlooked issue in object detection, *i.e.*, detection stability. For evaluation of such stability, we carefully design a well-proved metric named Stability Index (SI). The prediction consistency learning framework is further proposed to enhance model stability. Our extensive experiments have verified the rationality of SI and the effectiveness of the proposed framework. We hope our work can serve as a reliable baseline and draw the community's attention to this crucial issue in 3D object detection.

To motivate future work, we outline a few limitations based on our current comprehension: (1) The proposed SI focuses solely on the default detection elements for the purpose of generalization. However, some detectors yield additional predictions such as velocity and attribute. Integrating the stability of these extra elements into the metric, while maintaining the properties of SI, is a practically valuable direction. (2) In the pursuit of a general baseline approach, we restrict the design of PCL to be compatible with existing object detectors, avoiding the introduction of extra computations during inference to ensure broad applicability. Future works may surpass these constraints to explore possibilities for enhanced performance.

References

1. Bai, X., Hu, Z., Zhu, X., Huang, Q., Chen, Y., Fu, H., Tai, C.L.: Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1090–1099 (2022)
2. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. EURASIP Journal on Image and Video Processing **2008**, 1–10 (2008)
3. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: 2016 IEEE international conference on image processing (ICIP). pp. 3464–3468. IEEE (2016)
4. Bewley, A., Sun, P., Mensink, T., Anguelov, D., Sminchisescu, C.: Range conditioned dilated convolutions for scale invariant 3d object detection. arXiv preprint arXiv:2005.09927 (2020)
5. Bishop, G., Welch, G., et al.: An introduction to the kalman filter. Proc of SIGGRAPH, Course **8**(27599-23175), 41 (2001)
6. 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: CVPR. pp. 11621–11631 (2020)
7. Chai, Y., Sun, P., Ngiam, J., Wang, W., Caine, B., Vasudevan, V., Zhang, X., Anguelov, D.: To the point: Efficient 3d object detection in the range image with graph convolution kernels. In: CVPR. pp. 16000–16009 (2021)
8. Chen, X., Ma, H., Wan, J., Li, B., Xia, T.: Multi-view 3d object detection network for autonomous driving. In: CVPR. pp. 1907–1915 (2017)
9. Chen, Y., Liu, S., Shen, X., Jia, J.: Fast point r-cnn. In: ICCV. pp. 9775–9784 (2019)
10. Chen, Y., Liu, J., Zhang, X., Qi, X., Jia, J.: Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In: CVPR. pp. 21674–21683 (2023)
11. Contributors, M.: MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. `https://github.com/open-mmlab/mmdetection3d` (2020)
12. Deng, J., Shi, S., Li, P., Zhou, W., Zhang, Y., Li, H.: Voxel r-cnn: Towards high performance voxel-based 3d object detection. In: AAAI. vol. 35, pp. 1201–1209 (2021)
13. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
14. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html (2005)
15. Fan, L., Pang, Z., Zhang, T., Wang, Y.X., Zhao, H., Wang, F., Wang, N., Zhang, Z.: Embracing single stride 3d object detector with sparse transformer. In: CVPR. pp. 8458–8468 (2022)
16. Fan, L., Xiong, X., Wang, F., Wang, N., Zhang, Z.: Rangedet: In defense of range view for lidar-based 3d object detection. In: ICCV. pp. 2918–2927 (2021)
17. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)

18. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: CVPR. pp. 12697–12705 (2019)
19. Li, J., Luo, C., Yang, X.: Pillarnext: Rethinking network designs for 3d object detection in lidar point clouds. In: CVPR. pp. 17567–17576 (2023)
20. Li, Z., Wang, F., Wang, N.: Lidar r-cnn: An efficient and universal 3d object detector. In: CVPR. pp. 7546–7555 (2021)
21. Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Qiao, Y., Dai, J.: Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In: ECCV. pp. 1–18. Springer (2022)
22. Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D.L., Han, S.: Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In: IEEE International Conference on Robotics and Automation. pp. 2774–2781. IEEE (2023)
23. Liu, Z., Yang, X., Tang, H., Yang, S., Han, S.: Flatformer: Flattened window attention for efficient point cloud transformer. In: CVPR. pp. 1200–1211 (2023)
24. Meyer, G.P., Laddha, A., Kee, E., Vallespi-Gonzalez, C., Wellington, C.K.: Lasernet: An efficient probabilistic 3d object detector for autonomous driving. In: CVPR. pp. 12677–12686 (2019)
25. Pang, Z., Li, Z., Wang, N.: Simpletrack: Understanding and rethinking 3d multi-object tracking. arXiv preprint arXiv:2111.09621 (2021)
26. Prakash, A., Chitta, K., Geiger, A.: Multi-modal fusion transformer for end-to-end autonomous driving. In: CVPR. pp. 7077–7087 (2021)
27. Qi, C.R., Litany, O., He, K., Guibas, L.J.: Deep hough voting for 3d object detection in point clouds. In: ICCV. pp. 9277–9286 (2019)
28. Qi, C.R., Liu, W., Wu, C., Su, H., Guibas, L.J.: Frustum pointnets for 3d object detection from rgb-d data. In: CVPR. pp. 918–927 (2018)
29. Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H.: Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: CVPR. pp. 10529–10538 (2020)
30. Shi, S., Jiang, L., Deng, J., Wang, Z., Guo, C., Shi, J., Wang, X., Li, H.: Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. arXiv preprint arXiv:2102.00463 (2021)
31. Shi, S., Wang, X., Li, H.: Pointrcnn: 3d object proposal generation and detection from point cloud. In: CVPR. pp. 770–779 (2019)
32. Shi, S., Wang, Z., Shi, J., Wang, X., Li, H.: From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. IEEE TPAMI **43**(8), 2647–2664 (2020)
33. Shi, S., Wang, Z., Wang, X., Li, H.: Part-aˆ 2 net: 3d part-aware and aggregation neural network for object detection from point cloud. arXiv preprint arXiv:1907.03670 **2**(3) (2019)
34. Shi, W., Rajkumar, R.: Point-gnn: Graph neural network for 3d object detection in a point cloud. In: CVPR. pp. 1711–1719 (2020)
35. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: CVPR. pp. 2446–2454 (2020)
36. Sun, P., Wang, W., Chai, Y., Elsayed, G., Bewley, A., Zhang, X., Sminchisescu, C., Anguelov, D.: Rsn: Range sparse net for efficient, accurate lidar 3d object detection. In: CVPR. pp. 5725–5734 (2021)
37. Team, O.D.: Openpcdet: An open-source toolbox for 3d object detection from point clouds. `https://github.com/open-mmlab/OpenPCDet` (2020)

38. Wang, H., Shi, C., Shi, S., Lei, M., Wang, S., He, D., Schiele, B., Wang, L.: Dsvt: Dynamic sparse voxel transformer with rotated sets. In: CVPR. pp. 13520–13529 (2023)
39. Wang, Y., Guizilini, V.C., Zhang, T., Wang, Y., Zhao, H., Solomon, J.: Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In: Conference on Robot Learning. pp. 180–191. PMLR (2022)
40. Weng, X., Wang, J., Held, D., Kitani, K.: 3d multi-object tracking: A baseline and new evaluation metrics. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 10359–10366. IEEE (2020)
41. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE international conference on image processing (ICIP). pp. 3645–3649. IEEE (2017)
42. Yan, Y., Mao, Y., Li, B.: Second: Sparsely embedded convolutional detection. Sensors **18**(10), 3337 (2018)
43. Yang, Z., Sun, Y., Liu, S., Jia, J.: 3dssd: Point-based 3d single stage object detector. In: CVPR. pp. 11040–11048 (2020)
44. Yang, Z., Sun, Y., Liu, S., Shen, X., Jia, J.: Ipod: Intensive point-based object detector for point cloud. arXiv preprint arXiv:1812.05276 (2018)
45. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3d object detection and tracking. In: CVPR. pp. 11784–11793 (2021)
46. Zhang, H., Wang, Y., Dayoub, F., Sunderhauf, N.: Varifocalnet: An iou-aware dense object detector. In: CVPR. pp. 8514–8523 (2021)
47. Zhang, H., Wang, N.: On the stability of video detection and tracking. arXiv preprint arXiv:1611.06467 (2016)
48. Zhang, Y., Hu, Q., Xu, G., Ma, Y., Wan, J., Guo, Y.: Not all points are equal: Learning highly efficient point-based detectors for 3d lidar point clouds. In: CVPR. pp. 18953–18962 (2022)
49. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In: CVPR. pp. 4490–4499 (2018)
50. Zhou, Z., Zhao, X., Wang, Y., Wang, P., Foroosh, H.: Centerformer: Center-based transformer for 3d object detection. In: European Conference on Computer Vision. pp. 496–513. Springer (2022)