Voxel RCNN-HA: A Point Cloud Multi-Object Detection Algorithm with Hybrid Anchors for Autonomous Driving

Hai Wang, *Senior Member, IEEE,* Le Tao, Yiming Peng, Zhiyu Chen, *IEEE,* and Yong Zhang

***Abstract*— 3D object detection using lidar becomes essential for subsequent vehicle decision-making and planning as part of an intelligent vehicle perception system. Voxel RCNN is a two-stage voxel-based 3D object detection algorithm that is fast and accurate. However, the detection accuracy for specific categories is insufficient in complex traffic scenarios, thus we propose the Voxel RCNN-HA algorithm. First, in light of the shortcomings of Voxel RCNN in detecting pedestrians, a hybrid detection head is proposed to balance the advantages and disadvantages of anchor-based and anchor-free algorithms and significantly improve pedestrian detection performance while maintaining vehicle accuracy. Second, Self-Attention is introduced in the second stage of the algorithm and a Voxel Region of Interest (RoI) Self-Attention pooling module is developed to obtain both local and global features in RoI, which addresses the issue that the original Voxel RoI pooling module is challenging to obtain global features of large objects. On the One millioN sCEnes (ONCE) dataset, the proposed Voxel RCNN-HA achieves 66.79% mean Average Precision (mAP) and 11.7 Frames Per Second (FPS), outperforms both Voxel RCNN and CenterPoints in terms of detection accuracy. Additionally, experiments on Waymo Open dataset and Custom-Rslidar dataset further validate the effectiveness and generalization of the proposed method.**

***Index Terms*— 3D Object Detection, Lidar, Anchor-based, Anchor-free, Self-Attention**

I. INTRODUCTION

Intelligent vehicles gather perceive surrounding environment via camera, lidar, and other sensors, then followed by subsequent decision-making and planning to ensure the normal operation of the vehicles. With the advances of sensor technology and the continuous improvement of computing power, object detection algorithms based on deep learning have made significant progresses in recent years [1]. The algorithms based on deep learning can make the intelligent vehicle accurately and quickly perceive the surrounding obstacles, such as vehicles, pedestrians, cyclists and so on. With the efforts both academia and industry, image-based 2D object detection[2] and segmentation[3] have matured to the point where they can achieve the performances comparable to that of human eyes. However, the 2D detection algorithms based on camera are easily affected by environmental factors such as illumination, and cannot adapt to complex and changeable autonomous driving scenarios[4]. Additionally, compared with 2D object detection, 3D object detection based on lidar can provide accurate information about the localization, dimensions, and heading angle of objects relative to the ego vehicle[5, 6]. Besides, the depth information of 3D space can be directly obtained by lidar which is less affected by environmental factors. With the continuous reduction of the cost of lidar and the development of chip technology, the 3D object detection algorithms based on lidar have advanced rapidly and began to be used in vehicles[7]. Voxel RCNN[8] is a classical and efficient two-stage anchor-based 3D object detection algorithm, which achieves a good balance between speed and accuracy, but the algorithm is challenging to detect specific categories such as pedestrians, buses and trucks well in complex autonomous driving scenarios. Therefore, in this paper, we propose an anchor-based and anchor-free hybrid detection head and Voxel RoI Self-Attention pooling module to alleviate this drawback.

According to whether anchors are used to enumerate the objects, the object detection algorithms can be categorized into anchor-based and anchor-free algorithms. The anchor-free algorithms perform better for small objects like pedestrian while the anchor-based algorithms perform better for huge objects like buses and trucks. The phenomenon is attributed to the inherent characteristic of the two kind of algorithms. The anchor-based algorithms propose to set axis-aligned anchors at each pixel of the 2D feature map as prior knowledge. The anchor-based algorithms are easy to train, and the network always converges quickly. Especially, the axis-aligned anchors can fit the vehicle well via neural network because the vehicles in driving scene are usually vertical or parallel to ego vehicle. However, the orientation of pedestrians on the roadside is typically changeable in the scale of 360 degrees and cannot be enumerated with limited axis-aligned anchors. Meanwhile, the addition of anchors with more angles will lower the inference speed of algorithm and worsen the imbalance between positive and negative samples to affect training process. Additionally, anchor-based algorithms are easier to match with large objects which have fixed orientation than small objects with changeable orientation, resulting in a good performance for vehicles but poor performance for pedestrians. In contrast to anchor-based algorithms, anchor-free algorithms predict the object's center point using the heatmap and then directly predict localization, dimensions and heading angle using the center point's feature[9]. The anchor-free algorithms attend to

This work was supported in part by the National Natural Science Foundation of China (52225212, U20A20333, 52072160), Key Research and Development Program of Jiangsu Province (BE2020083-3).

(Corresponding Author: Hai Wang);

H. W. L.T, Y. P. and Z. C. are with the School of Automotive and Traffic Engineering of Jiangsu University, Zhenjiang, 212013, China (e-mail: wanghai1019@163.com, 2212104060@stmail.ujs.edu.cn, 3200401204@ stmail.ujs.edu.cn, 1445536148@qq.com).

Y. Z., is with the School of Automotive and Traffic Engineering of Nanjing Forestry University, Nanjing, 212001, China (zy.js@163.com)

represent objects as key-points, and this representation is very friendly to rotational objects or objects with changeable orientations. However, several drawbacks exist in the anchor-free algorithms. The network converges slowly on datasets with few training samples, such as ONCE[10] and KITTI[11]. The anchor-free algorithms struggle to regress accurate localization and dimensions of large objects well using the center point feature. That's because lidar can only obtain the surficial information of object by emitting laser beam, which inevitably cause the lost of feature in center area of objects. As a result, anchor-free algorithms cannot accurately predict the localization for larger objects, such as bus and trucks. There will also be deviation when predicting the object's dimensions using the center point feature. And the dimension of pedestrians is small in autonomous driving scenarios and typically occupies only one grid on the feature map, so the grid-level feature is sufficient to predict the pedestrians' localization, dimension and heading angle well. From the above analysis, it is clearly that anchor-based algorithms performs better for large objects and anchor-free algorithms performs better for small objects. Therefore, we propose an anchor-based and anchor-free hybrid detection head based on the advantages and disadvantages of anchor-based and anchor-free algorithms. The anchor-based detection head is used for four types of objects with larger dimensions, including cars, buses, trucks, and cyclists, and the anchor-free detection head is utilized for pedestrians. This hybrid detection head achieves a good balance to detect large and small objects.

In order to refine the localization, dimensions and heading angle of the proposals, the second stage network obtains the feature of the proposals generated in the first stage via the RoI pooling module. The two-stage network has a clear effect on improving the detection accuracy of the object detection algorithm. In the refinement stage, PV-RCNN[12] and Voxel RCNN[8] seed 6×6×6 uniform grid points along the length, width and height of the candidate boxes, then encode the key point features or voxel features around each grid point onto the grid points. And the grid-level features are used to refine the localization, dimensions and heading angle of proposals through the fully connected layers. For small objects, each grid point encodes the key points or voxel features in its vicinity can obtain the object's global features. While for large objects such as trucks and buses, each grid point can only encode the features of part of the object. However, global features are critical for the refinement of the objects. CT3D[13] encodes the original point cloud and obtains the global features of the proposals in two-stage network with the Self-Attention mechanism. However, as is mentioned in Lidar RCNN[14], encoding features from the original point cloud will cause ambiguity in the proposals, which loses the dimension information of the proposals, and is not conducive to the refinement of the proposals. Based on the above reasons, we propose the Voxel RoI Self-Attention Pooling module based on Voxel RCNN[8] and CT3D[13] algorithms. On the basis of grid points encoding the surrounding voxels and obtaining the local features of objects, Self-Attention mechanism is used to expand the reception field of each grid point to obtain proposal-level features, encoding both the local and global features of the proposals for each grid point, which achieves the fine-grained refinement of proposals. The proposed Voxel RoI Self-Attention Pooling module highly improves the detection performance of large objects, including both buses and trucks.

In summary, we propose the Voxel RCNN-HA algorithm based on the Voxel RCNN algorithm, addressing its poor performance in detecting pedestrians, trucks, and buses. A hybrid detection head is proposed to balance the advantages and disadvantages of both anchor-based and anchor-free algorithms. A Voxel RoI Self-Attention pooling module is utilized to obtain the local and global features of proposals, significantly improving the detection performance of pedestrians, trucks, and buses. The following are the paper's primary contributions:

1. To address the shortcoming of the Voxel RCNN algorithm's poor detection performance for pedestrians, we propose an anchor-based and anchor-free hybrid detection head that combines the advantages of anchor-based algorithms for vehicles and anchor-free algorithms for pedestrians.
2. To address the shortcoming of the Voxel RCNN algorithm in terms of detection accuracy for large objects, a Voxel RoI Self-Attention pooling module is proposed. This module overcomes the shortcoming that the original Voxel RoI pooling module is challenging to obtain the global feature of large objects.
3. The proposed Voxel RCNN-HA algorithm enhances pedestrian detection performance while maintaining vehicle accuracy. On the ONCE dataset, Voxel RCNN-HA outperforms Voxel RCNN in terms of detection accuracy by 3.78%. Additionally, experiments on Waymo Open dataset and Custom-Rslidar dataset further validate the effectiveness and generalization of the proposed Voxel RCNN-HA.

II. Related Works

The 3D object detection algorithm based on lidar can be classified as anchor-based and anchor-free algorithm according to whether anchors are as prior knowledge. And 3D object detection algorithms can also be categorized into one-stage algorithms and two-stage algorithms depending on whether Region Convolutional Neural Networks (RCNN) are used.

Anchor-based algorithms and anchor-free algorithms

The concept of anchors are proposed by Faster R-CNN[15] which is a pioneering work of anchor-based algorithms. Different with image-based detection algorithms, the dimensions of anchors utilized by 3D algorithms need to be extended along z-axis. The classical 3D algorithm of SECOND[16] proposes to orientation prediction branch to refine the anchors, which is an efficient solution to reduce the false prediction of orientation. Considered the proposal recall by setting axis-aligned anchors is challenging to handle complex scenes, the STD[17] algorithm introduces spherical anchors at each point, which can significantly improve the recall of proposals. Part-A2[18] compares the anchor-based and anchor-free strategies anchor concludes that the former can obtain a higher recall of proposals. However, the higher recall brought by anchor-based algorithms is at the expense of higher memory and computational cost. Meanwhile, the axis-aligned anchors are difficult to fit the objects with changeable orientations, such as pedestrians.

The efficiency of network is significant factor to self-driving system, so some works proposes to eliminate the utilization of anchors to construct the anchor-free 3D algorithms. Anchor-free algorithms regard the task of object detection as keypoint regression. CenterPoints[9] and AFDet[19, 20] series represent each object in 3D space as a center point in BEV space. And the attributes of category, localization and dimension are directly regressed from the center-wise features. The algorithms significantly simplify the pipeline of 3D detection. CenterNet3D[21] algorithm further boosts the accuracy of localization of bounding boxes by designing a corner prediction branch for auxiliary training. However, the center-wise features are challenging to recover the real dimensions of objects in 3D space, especially for objects with large dimensions, such as bus and trucks.

One-stage algorithms and two-stage algorithms

Inspired by 2-D detection algorithms, two-stage 3D detection algorithms based on lidar generate proposals at first stage and subsequently refine the proposals at second stage. The PointRCNN[22] algorithm proposes to directly process the raw point cloud to generate coarse proposals and aggregates the features in proposal region for a refinement. However, the huge computational complexity of PointRCNN cannot be used to self-driving scenes. Part-A2[18] proposes a RoI-aware pooling module to aggregate the feature of proposal region, which fully considers the position information of each point in the proposals. Instead, LiDAR RCNN[14] proposes to aggregate the features of raw points in RoI region by using virtual points and boundary offsets. The PV-RCNN[12] proposes a voxel set abstraction (VSA) module that aggregates voxel features and raw points features. Considered the low inference speed of PV-RCNN, Voxel RCNN designs a Voxel-RoI pooling module to aggregate features directly from voxels, which achieves comparable performance with SOTA point-based algorithms at premise of high inference speed. The CT3D[13] algorithm proposes a channel-wise transformer to process the raw points inside the proposal region, which is highly efficient for proposal refinement. The researches in academia about two-stage 3D algorithms now are aimed to construct complex refinement stage and have made a lot of progress. However, the existed two-stage 3D algorithms cannot balance the detection for all kinds of categories well.

For one-stage lidar detector, VoxelNet[23] is a pioneering work to encode point cloud data into 3D voxels and use 3-D convolutions for feature extraction. Based on VoxelNet, SECOND[16] introduces sparse convolutions to replace the general 3D convolutions, which significantly reduces the inference time and improves the detection accuracy. In order to further speed up the 3D detection algorithms, PointPillars[24] encodes point cloud data into BEV pillars which is friendly to introduce the technology in image field and is more simple for industrial deployment. The one-stage lidar detectors are usually fast at inference speed but the detection accuracy are not enough to satisfy the demand of self-driving.

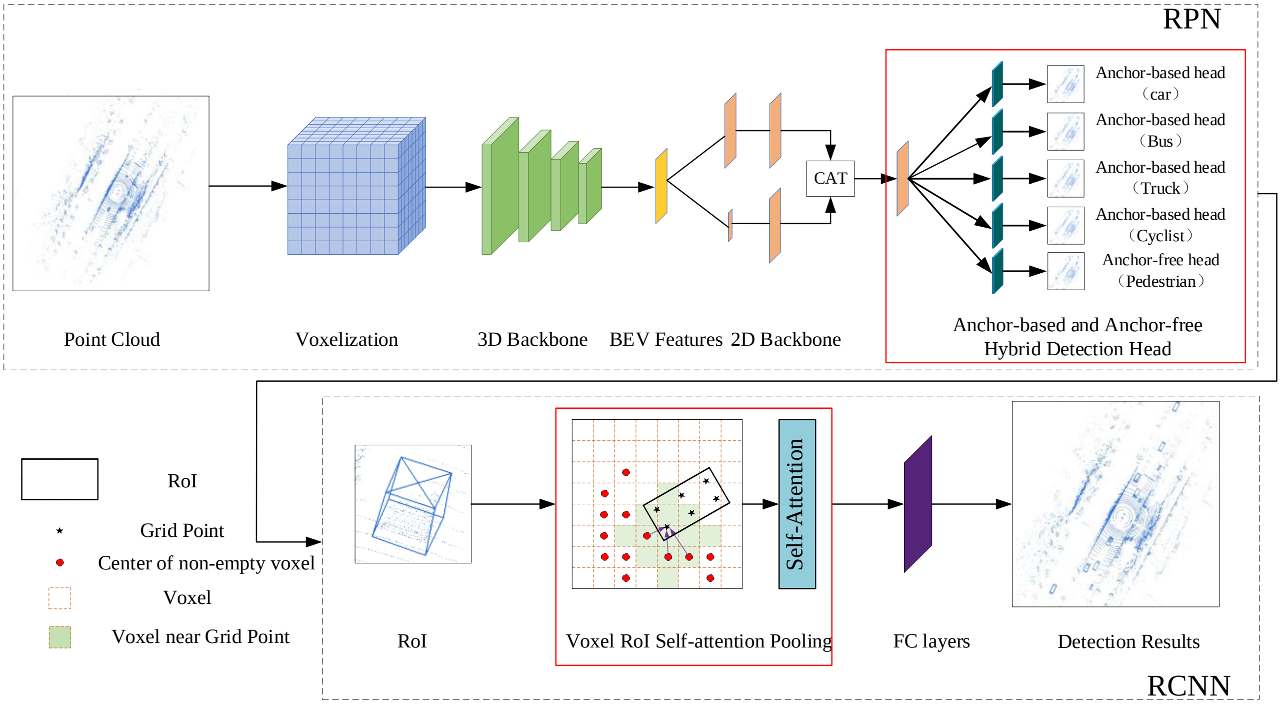


Fig.1 The network of Voxel RCNN-HA. The network of Voxel RCNN-HA consists of RPN and RCNN. The pipeline of RPN is similar to the Voxel RCNN. Instead of using coupled anchor-based detection head, Voxel RCNN-HA utilizes the decoupled anchor-based and anchor-free hybrid head. The RCNN stage consists of Voxel RoI Self-attention Pooling module and FC layers.

III. Methodology

This chapter demonstrates the Voxel RCNN-HA algorithm flow. Voxel RCNN-HA proposes an anchor-based and anchor-free hybrid detection head in the first stage and a Voxel RoI Self-Attention pooling module in the second stage based on Voxel RCNN. Section *A* describes pipelines of the Voxel RCNN and Voxel RCNN-HA respectively; Section *B* introduces anchor-based and anchor-free hybrid detection heads; Section *C* explains the structure of the Voxel RoI Self-Attention pooling module; and Section D presents the loss function of Voxel RCNN-HA in detail.

A. The procedure of Voxel RCNN and Voxel RCNN-HA

Voxel RCNN is a two-stage anchor-based algorithm for 3D object detection. The first stage involves the voxelization to the point cloud along the X, Y, and Z axes, followed by a 3D backbone network composed of sparse convolutions and submanifold convolutions[25, 26]. Submanifold convolution is utilized to extract features from non-empty voxels, while sparse convolution is aimed to expand the reception field. The 3D backbone network, significantly improves the processing speed of point clouds. By 1×, 2×, 4×, and 8× down-sampling of voxels with a 3D backbone network, multi-scale 3D voxel features can be obtained, and then are projected to the bird eye view to form a 2D pseudo-image. A 2D backbone network composed of 2D convolutions is utilized to perform feature fusion to obtain the feature which can satisfy object detection of different categories. Two axis-aligned anchor boxes with angles of 0 and 90 degrees are placed on each pixel of the two-dimensional feature map for each category. Then, the proposals with high quality can be obtained by detection head which regards the axis-aligned anchors and output of 2D backbone as input. In the second stage, 6×6×6 grid points are seeded uniformly along each dimension of the length, width and height of each proposal, and multi-scale voxel features are aggregated into each grid point to obtain grid-level features. Following the Voxel RoI pooling module[8], each proposal obtains the pooled feature for refinement, and the offsets for proposal refinement are performed by fully connected layers.

However, the detection performance of the Voxel RCNN algorithm is insufficient to meet the complex traffic conditions. Specifically, the detection performance for specific categories, such as pedestrians, trucks, buses, should be further improved. As a result, the Voxel RCNN-HA are proposed as a variant of Voxel RCNN. The network structure of Voxel RCNN-HA is depicted in Figure 1, where the red rectangles represent the proposed part. In the first stage of Voxel RCNN-HA, an anchor-based and anchor-free hybrid detection head is proposed; each detection head is responsible for a specific category, with four anchor-based detection heads for detecting cars, buses, trucks, and cyclists, respectively, and an anchor-free detection head for recognizing pedestrians. The design can ensure the detection performance of both pedestrians and vehicles at the same time. The second stage consists of the Voxel RoI Self-Attention pooling module and fully connected layers. The Voxel RoI Self-Attention pooling module captures local and global features for each grid point, significantly improving the detection performance of large objects such as trucks and buses.

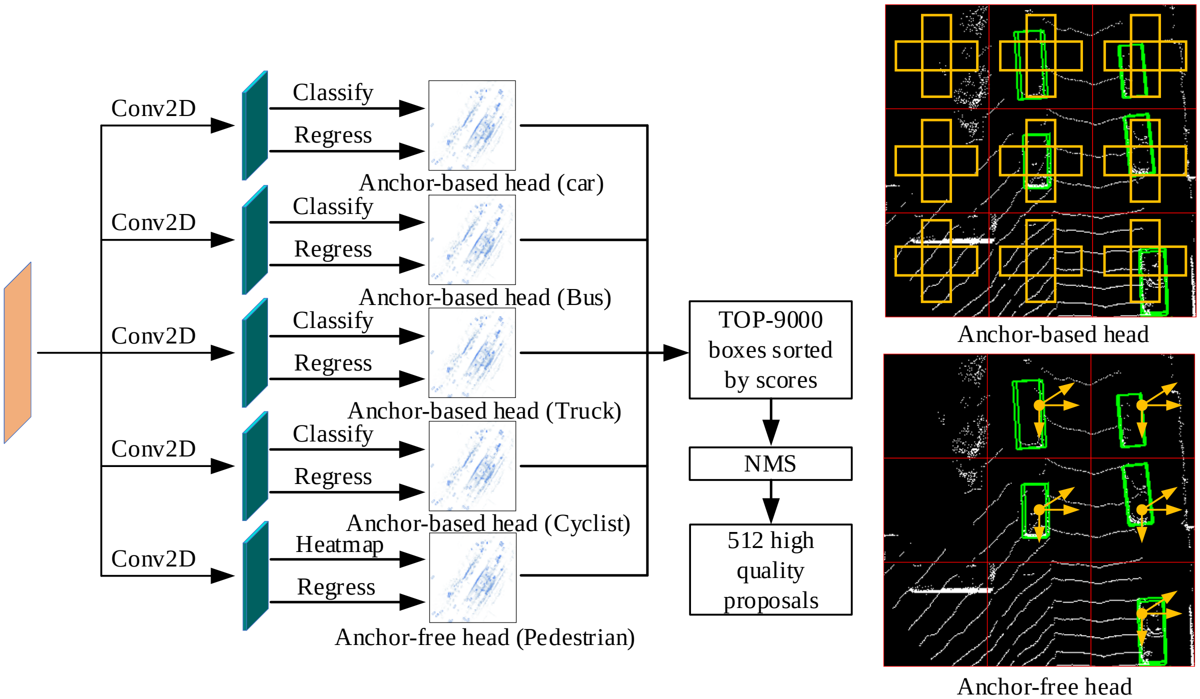


Fig.2 The structure of anchor-based and anchor-free hybrid detection head

B. Anchor-based and anchor-free hybrid detection head

The anchor-based algorithm enumerates the possible localizations of objects by setting axis-aligned anchor boxes. However, because the heading angle of a pedestrian is flexible and changeable, the anchor-based algorithm performs poorly for the detection of pedestrians. The anchor-free algorithm aggregates the global features of point cloud into the center points to regress the objects, which is robust to objects with changeable orientations. However, the point cloud exists only on the object's surface, and the center point's features are lost severely for large objects such as vehicles. Given the poor detection performance of pedestrians with anchor-based algorithms and the low detection accuracy of vehicles with anchor-free algorithms, an efficient anchor-based and ancho-free hybrid detection head is proposed in the first stage. The hybrid detection head can improve pedestrian detection performance while ensuring the vehicle's detection accuracy, which makes full use of the advantages of both anchor-based and anchor-free algorithms.

As illustrated in Figure 2, the anchor-based and anchor-free hybrid detection head modules take the output of 2D backbone as input and generate five new feature maps using five two-dimensional convolutions. These five new feature maps are sent to the hybrid detection heads, of which the first four detection heads are anchor-based detection heads used to predict cars, buses, trucks, and cyclists respectively, the residual detection head is an anchor-free detection head, which is responsible for detecting pedestrians. For each anchor-based detection head, the axis-aligned anchors with an angle of 0 and 90 degrees are placed at each pixel of the feature map. And there are two branches, one is used to classify the anchors to identify foreground and background, the other is utilized to regress the localization, dimension, and heading angle. Proposals are obtained based on the predictions by hybrid detection head and the prior anchors, which are pretty close to the real objects. The anchor-free detection head first predicts heatmaps for all categories which are utilized for decoding the center points and categories of objects. Then the residual attributes of objects such as dimensions, orientations are regressed from the features on center points. Finally, the output boxes of the hybrid detection head are combined and sorted according to the classification scores to obtain proposals with high quality. In this work, we select 9000 boxes with the highest classification scores for the Non-Maximum Suppression (NMS) module, which generates 512 high-quality proposal boxes for the second stage.

C. *Voxel RoI Self-Attention pooling module*

The original Voxel RoI pooling module uniformly seeds 6×6×6 grid points along the dimension of length, width, and height, and each grid point encodes the surrounding voxel features. With this encoding method, each grid point can only obtain the object's global feature for small objects. However, for objects with large dimensions, like trucks, and buses, whose length and height are approximately 10 meters and 3 meters. As a result, each grid point can only encode the object's local features, which are incapable of accurately refining the proposals. Additionally, small angle deviation of large objects will lead to smaller the intersection of union (IoU) between the prediction and ground truth boxes. To address this issue, the algorithm introduces the Self-Attention mechanism to capture the relationship between each grid point, and each grid point can encode the feature of other grid points. Each grid point can obtain both local and global features of the objects via the voxel RoI Self-Attention pooling module, allowing for more fine-grained refinement of the large object's localization, dimension, and heading angle.

Figure 3 illustrates the structure of the Voxel RoI Self-Attention pooling module. Using the high-quality proposals generated in the first stage and the multi-scale voxel features from 3D backbone as input, the voxel features surrounding the grid points are aggregated onto the grid points, and each grid point can obtain the grid-level feature, we also name it as local feature F_{local} in this work. The Self-Attention mechanism is further utilized to extract grid-level features to obtain proposal-level features. Compared with normal self-attention mechanism, the channel-wise self-attention mechanism can well learn the importance of each grid-level feature to obtain comprehensive proposal features, which can contribute the final refinement a lot. Specifically, the local feature F_{local} from grid points is embedding to high dimension by matrix W_q, W_k, and W_v will get the features *Q*, *K*, and *V,* respectively. The inner product of features *Q* and *K* is used to determine the relationship between grid point feature and other grid point features. Weights calculated by Softmax function are used to filter informative features from *V*, obtaining the global feature F_{global} of the object for each grid point. Finally, local feature F_{local} and global feature F_{global} are combined to form the proposal-level feature $F_{proposal}$ for the refinement of the proposals.

$$Q = F_{local} \cdot W_q, \;\; K = F_{local} \cdot W_k \;, \;\; V = F_{local} \cdot W_v \quad (1)$$

$$F_{global} = self\text{-}attention(Q, K, V)$$

$$= Softmax\left(\frac{Q \cdot K^T}{\sqrt{d}}\right) \cdot V \quad (2)$$

$$F_{proposal} = F_{local} + F_{global} \quad (3)$$

where *d* is the channels of feature *Q* and *K*.

D. *Loss Function*

The loss function L_{TOTAL} of Voxel RCNN-HA algorithm consists of two stages: first-stage proposals generation network loss L_{RPN} and second-stage proposals refinement network loss L_{RCNN}.

$$L_{TOTAL} = L_{RPN} + L_{RCNN} \quad (4)$$

where the proposals generation network loss, L_{RPN}, is comprised of anchor-free detection head loss L_{AF} and anchor-based detection head loss L_{AB}. Proposals refinement network loss L_{RCNN} is comprised of confidence prediction loss L_c and proposal localization refinement loss L_{locref}.

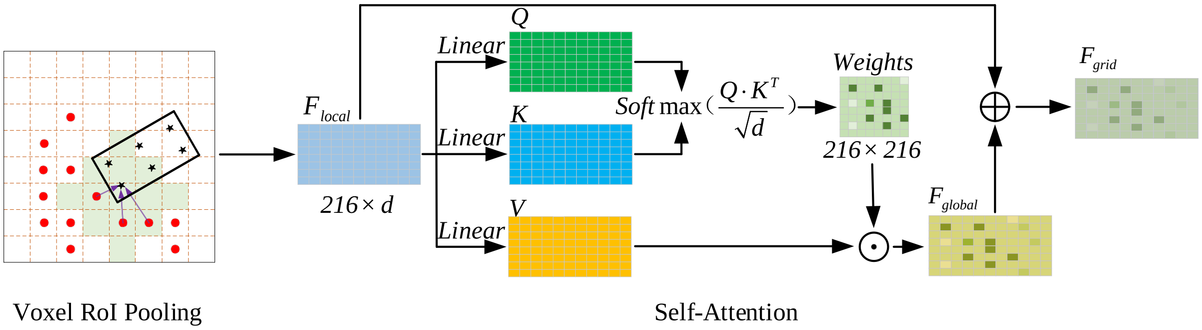


Fig.3 The structure of Voxel RoI Self-Attention pooling module

$$L_{RPN} = \lambda_1 L_{AF} + L_{AB} \tag{5}$$
$$L_{RCNN} = L_c + \lambda_2 L_{locref} \tag{6}$$
$$L_{AF} = L_{HM} + \lambda_3 L_{locaf} \tag{7}$$
$$L_{AB} = L_{cls} + \lambda_4 L_{locab} \tag{8}$$

where anchor-free detection head loss L_{AF} is composed of heatmap loss L_{HM} and anchor-free proposal localization loss L_{locaf}. While anchor-based detection head loss L_{AB} is composed of classification loss L_{cls} and anchor-based proposal localization loss L_{locab}. Voxel RCNN-HA employs Gaussian focal loss[27] to calculate the heatmap loss L_{HM}, focal loss to calculate the classification loss L_{cls}, binary cross entropy loss to calculate the confidence prediction loss L_c. It also uses smooth L1 loss[15] to calculate anchor-free candidate box location loss L_{locaf}, anchor-based candidate box location loss L_{locab} , and candidate box location refinement loss L_{locref}. In formula (5-8), λ_1, λ_2, λ_3, and λ_4 are loss balance coefficients.

IV. Experiments and Results

A. Dataset

ONCE dataset is a public dataset from HuaWei corporation, which was collected in China using a HeSai Pandar 40-beam lidar and seven cameras to capture automatic driving scenes at various periods, road conditions, and weather conditions. Compared to the KITTI dataset, the ONCE dataset covers more complex scenes. As illustrated in Figure 4, the ONCE dataset contains a greater number of objects per frame. Figure 5 illustrates a frame sample from the KITTI and ONCE datasets. The KITTI dataset contains a single forward-looking perspective, whereas the ONCE dataset contains a 360-degree scene. As a result, the ONCE dataset is a better fit for the perception requirements of intelligent vehicles operating in automated driving scenarios. ONCE contains over one million unlabeled samples and over sixteen thousand labeled samples that can be used for unsupervised, semi-supervised, and supervised learning. This algorithm is classified as supervised learning; it divides 16000 labeled samples into three groups: 5000 for training, 3000 for validation, and 8000 for testing. According to the evaluation script officially provided by ONCE dataset, the detection accuracy of five types of objects, including cars, trucks, buses, cyclists, and pedestrians, can be evaluated. Additionally, cars, trucks, and buses can be combined to evaluate the detection accuracy of three types of objects: vehicles, cyclists, and pedestrians.

Waymo Open Dataset is the largest public dataset of 3D detection for autonomous driving. There are totally 798 training sequence with around 160k LiDAR samples, and 202 validation sequences with 40k LiDAR samples. The We evaluate the proposed Voxel RCNN-HA on this large-scale dataset to further validate the effectiveness and generalization of our improvement.

Custom-Rslidar Dataset is a dataset built by our group. The dataset is collected in China using a Robosense 128-beam lidar on real automatic driving scenes covering various periods, road conditions, and weather conditions. There are totally 5.4k Lidar samples with annotations which include 4 categories of small car, large car, pedestrian and cyclist. And the train set is 4k and validate set is 1.4k in this work. We evaluate the proposed model on the Custom-Rslidar Dataset to provide quantitative analysis in real world to further prove the effectiveness of our improvement.

TABLE I
Results of Voxel RCNN-HA with Different λ_1

λ_1	Vehicle	Pedestrian	Cyclist	mAP
1	79.60	46.64	67.58	64.61
0.5	80.91	48.13	69.04	66.03
0.2	81.24	49.22	69.92	66.79
0.1	78.98	49.66	63.96	64.20

B. Experiment details and settings

On ONCE dataset, the Voxel RCNN-HA algorithm is compared with Voxel RCNN[8] and other classical 3D object detection algorithms in order to demonstrate its effectiveness. Among them, the mAP of PointPainting[28], PointRCNN[22], PointPillars[24], SECOND[16], PV-RCNN[12], CenterPoints[9] is provided by ONCE benchmark [1]. Voxel RCNN and Voxel RCNN-HA are trained and tested on a workstation equipped with two Nvidia RTX 2080 Ti GPUs, i7-9700k CPU, and 64GB memory, and it takes about 28 hours to train the network. The platform is ubuntu 18.04, deep-learning framework is pytorch 1.4, programming language is python 3.7, batch size is 4, epoch is 80, the initial learning rate is 0.003, the learning rate policy is cosine annealing[29], and the optimizer is adamW[30]. Using the lidar of ego vehicle as the view point, the detection range is 75.2 meters forward, back, left and right, respectively, 3 meters upward and 5 meters downward. The whole scenario is divided into uniform voxels and the size of each voxel is 0.1×0.1×0.2 meters.

When training the Voxel RCNN-HA, loss balance coefficients λ_1 has great effect on the detection accuracy. As is shown in TABLE I, larger or smaller λ_1 result in a significant decrease in mAP. λ_1 is used to balance the anchor-free and anchor-based detection head. Larger λ_1 makes the gradient of anchor-free branch descent larger, smaller λ_1 makes the loss are concentrated on anchor-based branch, no matter larger or smaller λ_1 makes the network not

TABLE II
COMPARISON OF VOXEL RCNN-HA AND OTHER ALGORITHMS ON ONCE DATASET

Method	Anchor-based	Anchor-free	Vehicle	Pedestrian	Cyclist	mAP	FPS/Hz
PointPainting	✓		66.17	44.84	62.34	57.78	-
PointRCNN		✓	52.09	4.28	29.84	28.74	1.3
PointPillars	✓		68.57	17.63	46.81	44.34	**27.1**
SECOND	✓		71.19	26.44	58.04	51.89	26.6
PV-RCNN	✓		77.77	23.50	59.37	53.55	5.9
CenterPoints		✓	66.79	**49.90**	63.45	60.05	14.2
Voxel RCNN	✓		80.13	42.03	66.88	63.01	15.3
Ours	✓	✓	**81.24**	49.22	**69.92**	**66.79**	11.7

TABLE III
ABLATION STUDY RESULTS OF EACH MODULE OF VOXEL RCNN-HA ON ONCE DATASET

Method	Car	Bus	Truck	Pedestrian	Cyclist	mAP
Baseline	80.68	72.61	45.29	42.03	66.88	61.5
Baseline + Hybrid Detection Head	81.09	74.66	45.98	49.07	68.83	63.93
Improvement	**0.41**	**2.05**	**0.69**	**7.04**	**1.95**	**2.43**
Baseline + Hybrid Detection Head + voxel RoI Self-Attention pooling	81.37	78.67	47.48	49.22	69.92	65.33
Improvement	**0.28**	**4.01**	**1.50**	**0.15**	**1.09**	**1.4**

TABLE IV
ABLATION STUDY RESULTS OF EACH MODULE OF VOXEL RCNN-HA ON WAYMO OPEN DATASET

Method	Vehicle		Pedestrian		Cyclist	
	L1	L2	L1	L2	L1	L2
Baseline	75.11	67.04	61.35	54.24	64.68	62.37
Baseline + Hybrid Detection Head	75.87	67.48	66.44	60.11	65.89	62.91
Improvement	**0.76**	**0.44**	**5.09**	**5.87**	**1.21**	**0.54**
Baseline + Hybrid Detection Head + voxel RoI Self-Attention pooling	76.78	68.11	66.96	60.54	66.34	63.13
Improvement	**0.91**	**0.63**	**0.52**	**0.43**	**0.45**	**0.22**

All models are trained with 20% frames from the training set and are evaluated on the full validation set of the Waymo Open Dataset, and the evaluation metric is the mAPH in terms of LEVEL 1 (L1) and LEVEL 2 (L2) difficulties.

TABLE V
ABLATION STUDY RESULTS OF EACH MODULE OF VOXEL RCNN-HA ON CUSTOM-RSLIDAR DATASET

Method	Small car	Large car	Pedestrian	Cyclist	mAP
Baseline	70.99	80.66	44.07	65.02	65.19
Baseline + Hybrid Detection Head	72.86	81.95	50.43	66.32	67.13
Improvement	**1.87**	**1.29**	**6.36**	**1.3**	**1.94**
Baseline + Hybrid Detection Head + voxel RoI Self-Attention pooling	73.45	84.36	51.24	67.13	68.96
Improvement	**0.59**	**2.41**	**0.81**	**0.81**	**1.83**

[1] https://once-for-auto-driving.github.io/benchmark.html#benchmark

converge well. Through our experiment, we set λ_1 as 0.2, which has best detection performance. And the settings of λ_2, λ_3, and λ_4 is followed by Voxel-RCNN and Centerpoints, which is 1, 2, 2.

On Waymo Open dataset, we performed the ablation experiments to prove the effectiveness of our improvements. We utilize the anchor-based head for vehicle and cyclist and anchor-free head for pedestrian. The 3D threshold is set as 0.7 for vehicle detection and 0.5 for pedestrian/cyclist detection. For the Waymo Open dataset, the detection range is set as [75.2m, 75.2m] for both X and Y axes, and [-2m, 4m] for the Z axis, while the voxel size is set as (0.1m, 0.1m, 0.15m).

On Custom-Rslidar Dataset, we performed the ablation experiments to provide quantitative analysis in real world. We utilize the anchor-based head for small car, large car and cyclist, anchor-free head for pedestrian. The 3D threshold is set as 0.7 for small car and large car detection, and 0.5 for pedestrian/cyclist detection. For the Custom-Rslidar Dataset, the detection range is set as [-98.4m, 98.4m] for both X and Y axes, and [-3m, 5m] for the Z axis, while the voxel size is set as (0.1m, 0.1m, 0.2m).

C. *Experiment results*

TABLE II compares the detection accuracy and speed of Voxel RCNN-HA and other algorithms on the ONCE dataset. The mAPs for PointPainting, PointRCNN, PointPillars, SECOND, PV-RCNN, and CenterPoints are all taken from the ONCE dataset's official benchmark. The speed of all algorithms is trained by their default configurations and calculated on NVIDIA RTX 2080Ti GPU by us. As shown in TABLE II, anchor-based algorithms such as SECOND, PVRCNN, and VoxelRCNN outperform anchor-free algorithms for vehicles. In contrast, anchor-free algorithms such as CenterPoints outperform anchor-based algorithms for pedestrians. The Voxel RCNN-HA improves the accuracy of vehicles, pedestrians, cyclists, and mAP by 1.11%, 7.19%, 3.04%, and 3.78%, respectively, compared to original Voxel RCNN. The mAP of Voxel RCNN-HA is 9.01%, 38.05%, 22.45%, 14.9%, 13.24%, and 6.74% higher than that of PointPainting, PointRCNN, PointPillars, SECOND, PV-RCNN, CenterPoints. By comparing Voxel RCNN-HA to other algorithms, it is possible to conclude that the improvements proposed by Voxel RCNN-HA are efficient, which can improve pedestrian detection accuracy while maintaining vehicle detection accuracy, demonstrating the algorithm's effectiveness.

D. *Ablation Study*

To demonstrate the effectiveness and generalization of the improvements in Voxel RCNN-HA, the ablation study is performed on ONCE dataset, Waymo Open dataset, and Custom-Rslidar, respectively, which is based on the Voxel RCNN as a baseline. TABLE III represents the results on ONCE dataset.

After adding the anchor-based and anchor-free hybrid detection head, compared with the baseline, the improvements to pedestrians detection is significant which is improved by 7.04%, and there are also varying degrees of performance gains in other categories. Table IV shows the ablation results on Waymo Open dataset, it is clear to find that the hybrid head brings 5.09% and 5.87% improvements of mAPH on pedestrian in terms of L1 and L2 respectively. Meanwhile, the results on Custom-Rslidar dataset from Table V further prove the effectiveness of the strategy, which brings 6.36% improvement of AP. The experiments on the three datasets fully proves the advantages of the hybrid detection head. In addition, the performance promotion to objects of different categories reflects the effectiveness of decoupled-category strategy, which alleviates the mutual interference between features to some extent.

The model with hybrid detection head is regard as the new baseline. Then the Voxel RCNN-HA algorithm employs the Voxel RoI Self-Attention pooling module on the new baseline in the second stage. From Table III, the strategy obtains significant promotion on buses and trucks which are improved by 4.01% and 1.50%, respectively. As for Waymo Open dataset, our method also achieves superior performance to the three categories in terms of mAPH than baseline, where the gain for vehicle is 0.91% and 0.63% in terms of L1 and L2 respectively, which represents the generalization of the strategy. From table V, it is intuitive that the proposed method also achieves better

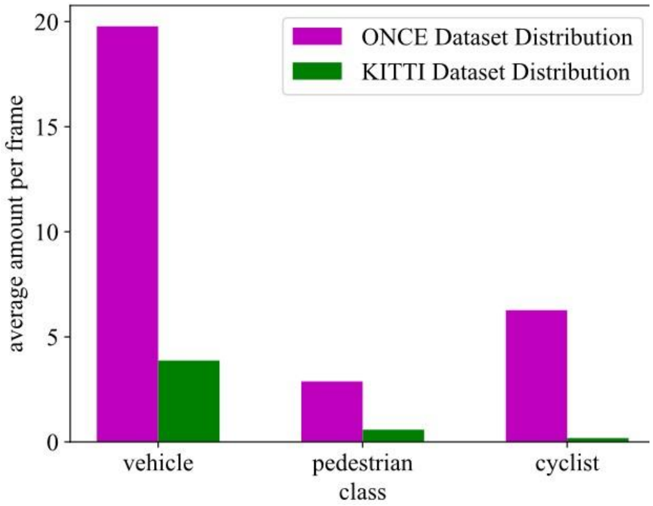
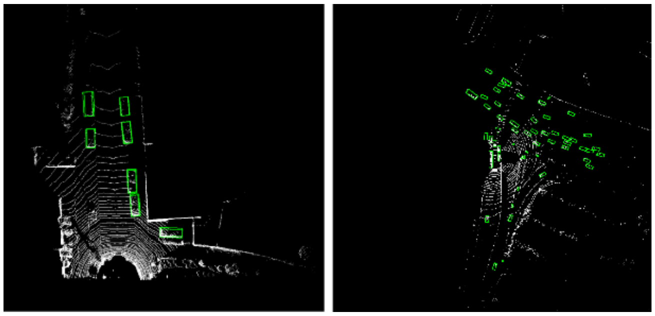


Fig.4 Distribution of ONCE dataset and KITTI dataset



(a) KITTI dataset (b) ONCE dataset

Fig.5 Visualization of KITTI dataset and ONCE dataset

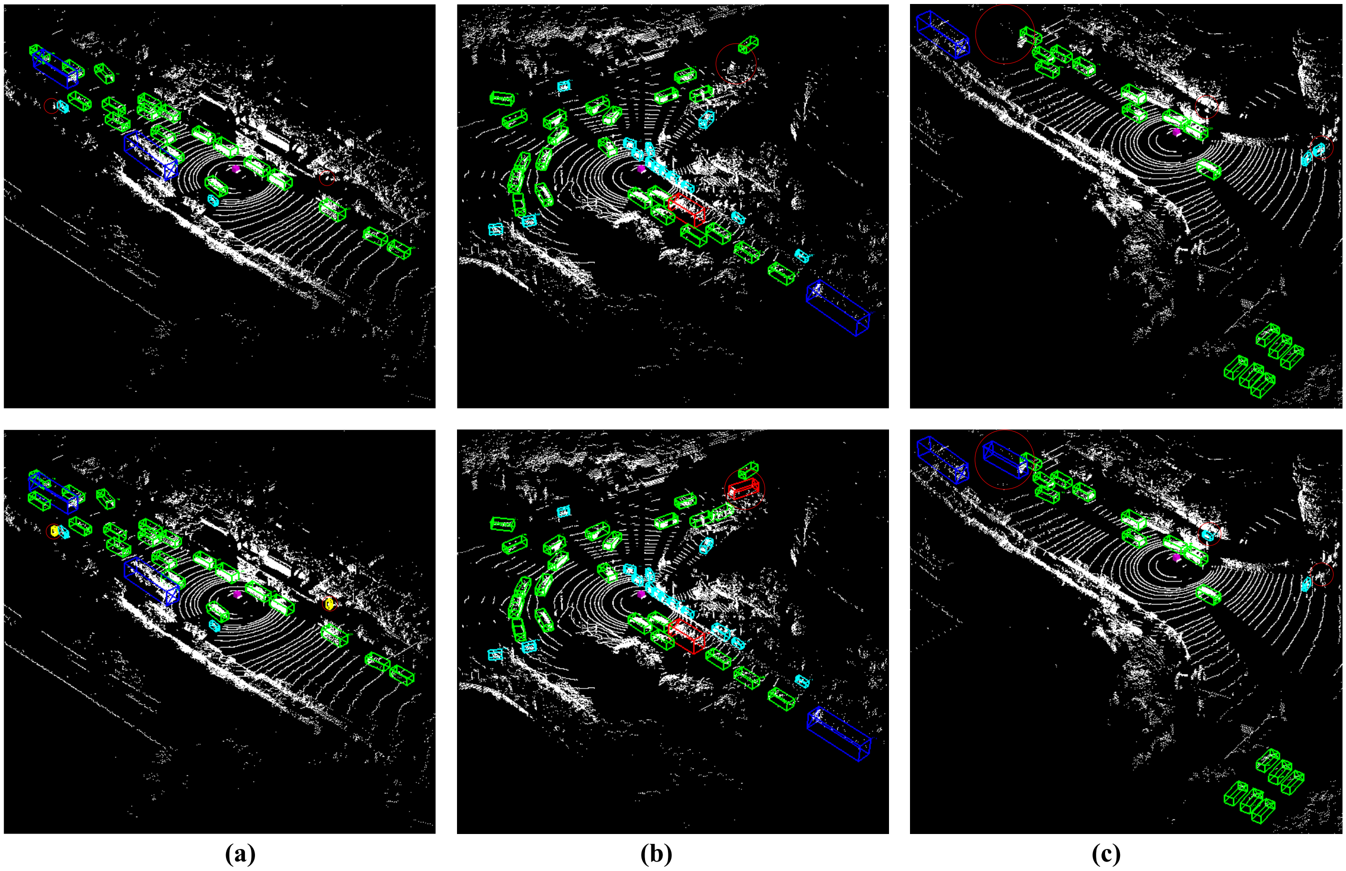


Fig.6 Comparison of detection performance between Voxel RCNN and Voxel RCNN-HA
The green, red, dark blue, light blue, and yellow 3D boxes represent cars, trucks, buses, cyclists, and pedestrians. The red circle represents the differences between Voxel RCNN-HA and Voxel RCNN.

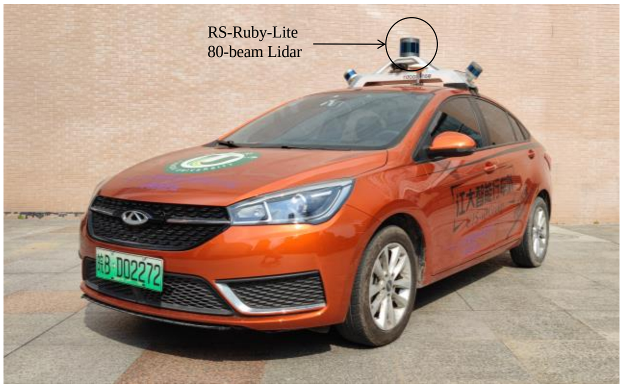


Fig. 7 The figure of intelligent vehicle platform and installation location of lidar

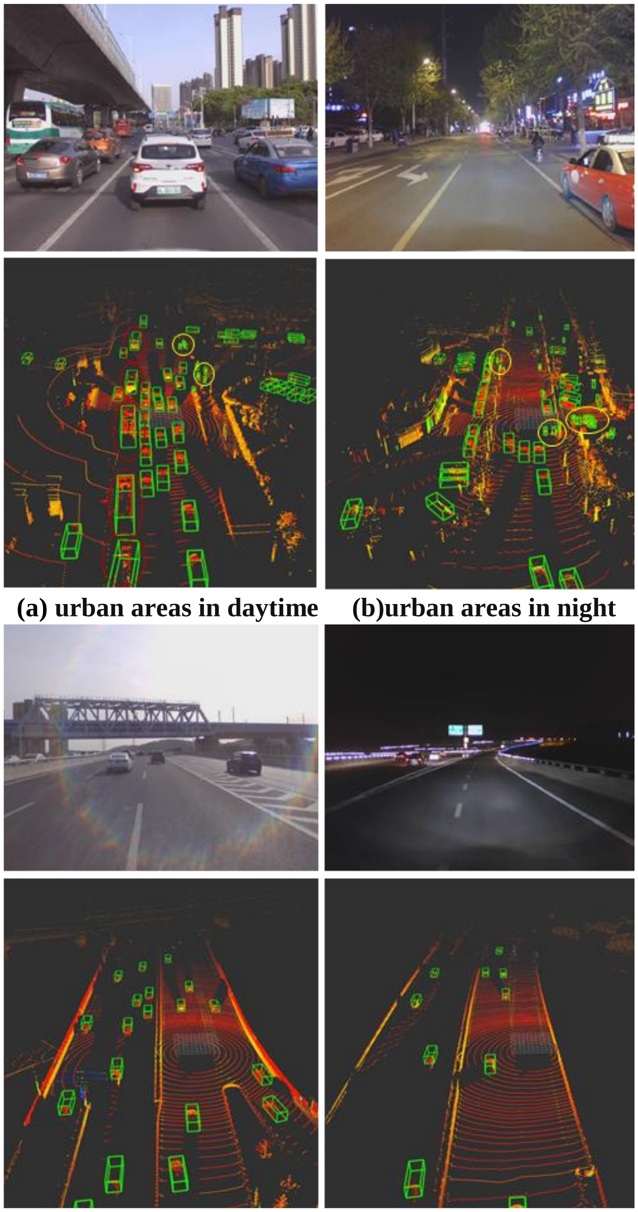
performance on Custom-Rslidar dataset than baseline, where the maximum gain is 2.41% to large car in terms of AP, which further validates the generalization ability of the strategy in real world. The results on various datasets fully validate the effectiveness and generalization of the Voxel RoI Self-Attention pooling module, which can well balance the feature extraction of both local and global features.

E. Visualization of ONCE dataset

To demonstrate the effectiveness of the Voxel RCNN-HA algorithm clearly, Figure 6 shows a comparison of detection results visualization. The first row represents the result of Voxel RCNN, while the second row represents the result of Voxel RCNN-HA. The green, red, dark blue, light blue, and yellow 3D boxes represent cars, trucks, buses, cyclists, and pedestrians. The red circle represents the differences between Voxel RCNN-HA and Voxel RCNN. The visualization results may convey a more intuitive representation of the improvement in detection performance of Voxel RCNN-HA over Voxel RCNN. As illustrated in Figure 6 (a)-(c), Voxel RCNN-HA improves the detection performance of pedestrians, trucks, and buses, respectively.

F. Real Road Experiment

To demonstrate the effectiveness of Voxel RCNN-HA in real traffic scenes, the algorithm is deployed on the Chery Arize 5E intelligent vehicle platform, and real-world road experiments are conducted. The intelligent vehicle platform and lidar are depicted in Figure 7. On the roof of the experiment platform, a RoboSense 80-beam lidar, two 16-beam lidars, and two MOKOSE cameras are installed, and the platform has an industrial computer equipped with an NVIDIA RTX 2080Ti GPU. Only an 80-beam RoboSense lidar, one camera, and an industrial computer are used in this experiment; the types are RS-Ruby-lite, MOKOSE IP70, and Advantech mic7700. Experiments are conducted in urban areas and on elevated roads during the day and night to validate the algorithm's effectiveness. Figure 8 illustrates the experiment's performance.



(a) urban areas in daytime (b)urban areas in night

(c) elevated roads in daytime (d)elevated roads in night

Fig. 8 Real road experiment results of Voxel RCNN-HA

The first row depicts the camera's front view; the second row depicts the lidar's detection results; the green boxes represent the detection results; the red circles represent detected buses, and the yellow circles represent detected pedestrians. Real world road experiments conducted at various periods and under varying road conditions demonstrate the algorithm's effectiveness. Our proposed Voxel RCNN-HA algorithm performs well for cars, buses, trucks, cyclists, and pedestrians.

V. Conclusions

Aiming at the poor performance of Voxel RCNN algorithm for pedestrians, trucks and buses on complex traffic conditions, we propose Voxel RCNN-HA algorithm based on Voxel RCNN. We propose an anchor-based and anchor-free hybrid detection head in the first stage of the algorithm. Combining the advantages of anchor-based and anchor-free algorithms significantly improves pedestrian detection accuracy while maintaining vehicle detection accuracy. Additionally, in the second stage, we propose a Voxel RoI Self-Attention pooling module that improves the detection performance of large objects such as trucks and buses. On the Huawei ONCE dataset, experiments demonstrate that this algorithm outperforms Voxel RCNN, PV-RCNN, and CenterPoints. Additionally, when the algorithm deployed on the intelligent vehicle platform ROS system still performs well in the real world experiment, demonstrating the algorithm's effectiveness. However, due to the characteristics of lidar, few point clouds are returned when confronted with small objects or objects at a great distance. As a result, the 3D object detection algorithm for small and long-distance objects remains a significant challenge. In the future, we will combine Voxel RCNN-HA with camera images to augment the point cloud with semantic information from the image, thereby improving detection results for small objects and objects at long distance and severe weather[31].

References

[1] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems,* vol. 30, no. 11, pp. 3212-3232, 2019.

[2] Y. Cai, T. Luan, H. Gao, H. Wang, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "YOLOv4-5D: An effective and efficient object detector for autonomous driving," *IEEE Trans Instrum Meas,* vol. 70, pp. 1-13, 2021.

[3] H. Wang, Y. Chen, Y. Cai, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "SFNet-N: An improved SFNet algorithm for semantic segmentation of low-light autonomous driving road scenes," *IEEE Trans Intell Transp Syst,* vol. 23, no. 11, pp. 21405-21417, 2022.

[4] Y. Li, F. Feng, Y. Cai, et. al, "Localization for intelligent vehicles in underground car parks based on semantic information," *IEEE Transactions on Intelligent Transportation Systems*. DOI: 10.1109/TITS.2023.3320088, 2023.

[5] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep Learning for 3d Point Clouds: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence," 2020.

[6] M. M. Rahman, Y. Tan, J. Xue, and K. Lu, "Notice of violation of IEEE publication principles: Recent advances in 3D object detection in the era of deep neural networks: A survey," *IEEE Transactions on image processing,* vol. 29, pp. 2947-2962, 2019.

[7] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3d object detection methods for autonomous driving applications," *IEEE Trans Intell Transp Syst,* vol. 20, no. 10, pp. 3782-3795, 2019.

[8] J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, "Voxel r-cnn: Towards high performance voxel-based 3d object detection," in Proceedings of the AAAI Conference on Artificial Intelligence, 2021, pp. 1201-1209.

[9] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d

object detection and tracking," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 11784-11793.

[10] J. Mao, M. Niu, C. Jiang, H. Liang, J. Chen, X. Liang, Y. Li, C. Ye, W. Zhang, and Z. Li, "One million scenes for autonomous driving: Once dataset," *arXiv preprint arXiv:210611037*, 2021.

[11] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite." pp. 3354-3361.

[12] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10529-10538.

[13] H. Sheng, S. Cai, Y. Liu, B. Deng, J. Huang, X.-S. Hua, and M.-J. Zhao, "Improving 3d object detection with channel-wise transformer," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 2743-2752.

[14] Z. Li, F. Wang, and N. Wang, "Lidar r-cnn: An efficient and universal 3d object detector," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 7546-7555.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems,* vol. 28, 2015.

[16] Y. Yan, Y. Mao, and B. Li, "SECOND: Sparsely Embedded Convolutional Detection," *Sensors (Basel),* vol. 18, no. 10, Oct 6, 2018.

[17] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "Std: Sparse-to-dense 3d object detector for point cloud," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1951-1960.

[18] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, "From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network," *IEEE Trans Pattern Anal Mach Intell,* vol. 43, no. 8, pp. 2647-2664, 2020.

[19] R. Ge, Z. Ding, Y. Hu, Y. Wang, S. Chen, L. Huang, and Y. Li, "Afdet: Anchor free one stage 3d object detection," *arXiv preprint arXiv:200612671*, 2020.

[20] Y. Hu, Z. Ding, R. Ge, W. Shao, L. Huang, K. Li, and Q. Liu, "AFDetV2: Rethinking the Necessity of the Second Stage for Object Detection from Point Clouds," *arXiv preprint arXiv:211209205*, 2021.

[21] G. Wang, J. Wu, B. Tian, S. Teng, L. Chen, and D. Cao, "CenterNet3D: An anchor free object detector for point cloud," *IEEE Trans Intell Transp Syst*, 2021.

[22] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 770-779.

[23] Y. Zhou, and O. 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, 2018, pp. 4490-4499.

[24] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 12697-12705.

[25] B. Graham, "Sparse 3D convolutional neural networks," *arXiv preprint arXiv:150502890*, 2015.

[26] B. Graham, and L. van der Maaten, "Submanifold sparse convolutional networks," *arXiv preprint arXiv:170601307*, 2017.

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980-2988.

[28] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection." pp. 4604-4612.

[29] L. N. Smith, and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates." pp. 369-386.

[30] I. Loshchilov, and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:171105101*, 2017.

[31] C. Zhang, H. Wang, Y. Cai, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "Robust-FusionNet: Deep multimodal sensor fusion for 3-D object detection under severe weather conditions," *IEEE Trans Instrum Meas,* vol. 71, pp. 1-13, 2022.