Point Cloud Object Detection Based on PointNet of Attention Mechanism

Tao Liu Zijia Dai Meiman He
ShanghaiTech University
{liutao, diazj, hemm}2022@shanghaitech.edu.cn

Abstract

LiDAR point cloud data is sparse and unordered. Usually, researchers convert these data into 3D voxel grids or images, but this will produce huge data and artifacts. In this paper, we use Kdtree accelerated DBSCAN clustering, skillfully use the sparsity of point cloud data, and can divide complex clusters without setting the number of prior categories, while saving computing costs. We input each type of point cloud we obtained into the PointNet classification network. In the network, we designed two kinds of attention mechanisms that operate along the channel number dimension and the point cloud number dimension. These two kinds of attention mechanisms are not affected by data disorder and can focus on the extraction of important features. Finally, we use clustering method to successfully detect pedestrian, bicycle and other targets from KITTI dataset without converting point cloud data, and use attention mechanism to improve the classification accuracy.

1. Introduction

LiDAR 3D semantic segmentation is the basic perception task of automatic driving. But in the outdoor scene segmentation problem, because the same object has differences in different lighting, location and other situations, these will lead to poor segmentation results. In addition, since LiDAR point cloud data is not in a conventional format, most researchers usually convert these data into conventional three-dimensional voxel meshes, etc. However, this data representation conversion will generate large-scale data and introduce artifacts. In addition, given the sparsity and disorder of LiDAR point cloud data, many 3D semantic segmentation methods [7, 8] cannot directly call the semantic segmentation of outdoor LiDAR point cloud. In this paper, to solve the above problems, we first use Kdtree accelerated DBSCAN clustering for non surface points, which uses the non-uniformity of LiDAR point cloud distribution in the scan band to identify the dense areas in the feature space, and then extract all point clouds within the radius of the longest diagonal of the 3D object detection box. After the points in the detection box are suppressed by non maximum values, they are input into our classification network for classification operations. Through the method in this paper, the input of the neural network is each type of point cloud after clustering, which does not need to be converted into voxel graph, and can save the calculation cost. The neural network we selected is the classification network part of Point Net. Because the convolution calculation has a local receptive field, and the pixels with the same label also have differences. So inspired by the channel and spatial attention mechanism in the image field, this paper designs an attention mechanism that operates along the channel number dimension and the point cloud number dimension. This attention mechanism is not affected by the disorder of data. At the same time, the attention weight is generated without convolution to optimize the expression of features and inhibit the extraction of unnecessary features.

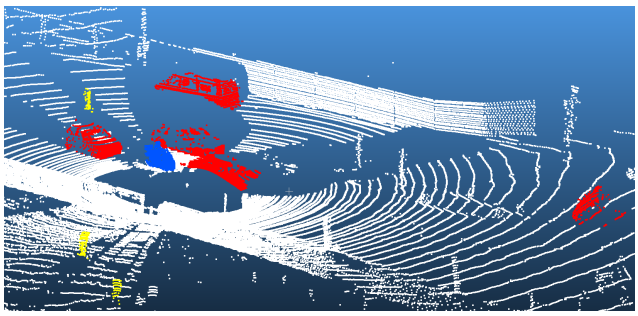

Figure 1. **Road scene object detection.** The yellow, red and blue point clouds in the figure are pedestrian, bicycle, car and other targets that we have detected in the KITTI dataset of the road scene.

Our main contributions are summarized as follows:

• We use Kdtree accelerated DBSCAN clustering for non surface points in LiDA data to extract each type of point cloud and input it into the improved Point Net classification network, which greatly saves computing costs without converting data. At the same time, this clustering algorithm skillfully uses the sparseness of point cloud data to divide

various complex clusters without setting the number of prior categories, It is better and faster than using Point-Net for segmentation.

• We added channel attention and point cloud attention modules. For the special structure of point cloud data, we pooled them along the two dimensions of channel number and point cloud number, respectively. Without using convolution, we used MLP to generate attention weights, making feature extraction follow the part that needs more attention. These two attention modules can inhibit unnecessary feature extraction from different angles, and improve the classification performance of Point Net networks.

2. Related Work

LiDAR Semantic Segmentation LiDARpoint cloud semantic segmentation usually needs to convert large-scale sparse point cloud into 3D voxel map, 2D bird's eye view (BEV) or distance view. [5, 10] is the pioneer of applying 3D convolutional neural network to voxelized shapes. Considering the sparsity of data and the computational cost of 3D convolution, more methods [1, 13] began to apply 3D sparse CNN to the voxel feature map of LiDAR point cloud segmentation. Following the trend of LiDAR point cloud detection method [4, 11], PolarNet [12] projected the point cloud into the 2D polar BEV diagram to balance the point distribution during voxelization. [6] Try to render a 3D point cloud or shape into a 2D image, and then apply a 2D convection network to classify it. [2,3] Feature based DNN first converts 3D data into vectors by extracting traditional shape features, and then uses fully connected networks to classify shapes. However, this is limited by the representation ability of extracted features. Therefore, in this paper, we will not convert LiDAR point clouds, but use DBSCAN clustering to extract point clouds in small-scale and dense target detection boxes, and then input the target point clouds into the depth neural network for classification.

Deep Learning on Unordered Sets From the perspective of data structure, point cloud data has the characteristics of non structure and disorder. But now a lot of work is on the ordered input of images or sequences, which also leads to the failure of the processing of ordered space on point clouds. A recent study by Oriol Vinyals et al. [9] explored this issue. They use read write networks with attention mechanisms to consume unordered input sets. However, their work focuses on generic sets and NLP applications, so they lack the role of geometry in sets. Therefore, in view of the disorder of the point cloud in this paper, the attention mechanism we designed is pooling along the dimension of the number of characteristic channels, and different input orders have no impact on the neural network model we use, while conducting spatial transformation on the input point cloud, so that the point cloud also has rotation invariance.

3. Method

3.1. RANSAC ground segmentation

We use KITT automatic driving data set, which is the data set of computer vision algorithm evaluation in the current international large vehicle environment. There are people, cars, bicycles, etc. in the road scenes included in this data set. First, we divide the KITT dataset on the ground to extract the non ground data we need. We use RANSAC algorithm for ground segmentation, which realizes ground segmentation by repeatedly selecting a set of random subsets in the data. According to the point cloud data, we set the minimum number of points in the ground data, and then use the random subset to fit out the model with the largest number of points, that is, the ground point set. Then we extract non ground point clouds from KITT for subsequent clustering operations.

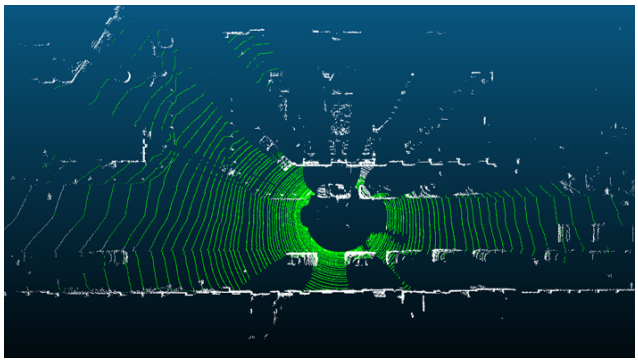

Figure 2. **Ground segmentation** Green is the divided ground point cloud data by RANSAC.

3.2. Kdtree accelerated DBSCAN clustering

DBSCAN algorithm is applicable to point cloud clustering. Considering that 3D point cloud data is large, we use Kdtree to retrieve adjacent points to accelerate DBSCAN algorithm in order to improve efficiency. In view of the sparsity of our point cloud data, we can use DBSCAN to divide the dense area into the same cluster. The algorithm first takes the points whose number of adjacent points is greater than the threshold as the core sample points, and then takes the points whose distance between core sample points is less than the threshold as a class for clustering operation.

The advantage of this algorithm is that it can well divide clusters of arbitrary shape, and does not set the number of clusters as a priori information. At the same time, it does not have a strong dependence on the access order. Therefore, DBSCAN method is very suitable for the data disorder of point clouds and the unknown of automatic driving scenes.

However, the complexity of the algorithm depends on the neighborhood point search, so in order to avoid time-

consuming, we use the Kdtree index method to query the neighborhood points. In order to find the nearest neighbor effectively, Kdtree adopts the idea of divide and rule, that is, the whole space is divided into several small parts, and then relevant search operations are carried out in the parts of a specific space.

After that, we perform non maximum suppression on the points in the detection box after clustering by DBSCAN algorithm, and then input each type of point cloud into the subsequent network for classification.

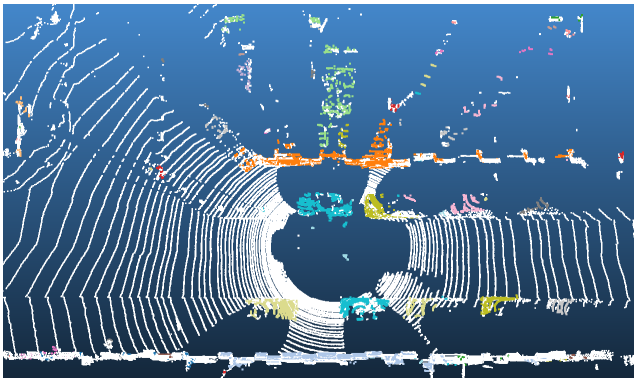

Figure 3. **Point cloud clustering.** After clustering, each color represents a cluster, and we will delete the class with less points before entering the network.

3.3. Attention mechanism

Attention mechanism can help the network pay more attention to the key position when extracting features, and inhibit the extraction of unnecessary features. Because point cloud data has the characteristics of disorder and sparsity, according to the dimension NxC of point cloud data, we designed two attention mechanisms for point clouds, where N is the number of point clouds, and C is equal to 3 when input, indicating coordinate information.

Channel Attention Mechanism For input data, we will first pool along the dimension of channel number C. To facilitate the aggregation of features from different angles, we use maximum pooling and average pooling to simultaneously operate point cloud data. Then the pooled data is trained through a single hidden layer MLP with shared parameters, because the unstructured point cloud data cannot use convolution operation to train weights. The MLP trained data will be added to generate attention weight, and then activated by activation function. Finally, it will multiply with the input data to play the role of channel attention mechanism. Channel Attention Mechanism A_c can be expressed as:

$$A_c = \text{sigmoid}\left(MLP\left(F_{avg}^{C}\right) + MLP\left(F_{\max}^{C}\right)\right) \quad (1)$$

$$A_c = \text{sigmoid}\left(w0\left(F_{avg}^{C}\right) + w0\left(F_{\max}^{C}\right)\right) \quad (2)$$

F_{avg}^{C}represents average pooling along the dimension of channel number C, $F_{\max}^{C}$ maximizes pooling along the dimension of channel number C. The MLP in the channel attention mechanism is single-layer, and w0 represents the shared weight of MLP. Finally, we activate it through the sigmoid function.The structure is shown in Figure 4.

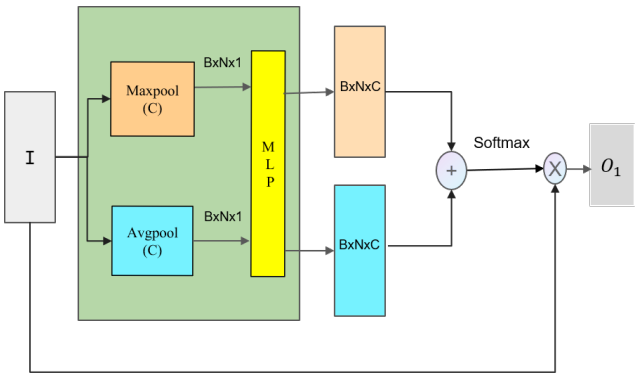

Figure 4. Channel Attention Mechanism

Point Cloud Attention Mechanism Different from the channel attention mechanism, the point cloud attention mechanism is first pooled along the dimension of point cloud number N.Similarly, we will use maximum pooling and average pooling to obtain the characteristics of aggregation from different angles. Then we use the dual hidden layer MLP with shared parameters for training. The first hidden layer reduces the number of point cloud channels C, and the second layer recovers it. Finally, the maximum pooling and average pooling are added to generate attention weights. The activation function is multiplied with the input after activation. Point Cloud Attention Mechanism A_n can be expressed as:

$$A_n = \text{sigmoid}\left(MLP\left(F_{\text{avg}}^{N}\right) + MLP\left(F_{\max}^{N}\right)\right) \quad (3)$$

$$A_n = \text{sigmoid}\left(w1\left(w2\left(F_{\text{avg}}^{N}\right)\right) + w1\left(w2\left(F_{\max}^{N}\right)\right)\right) \quad (4)$$

F_{avg}^{N} represents average pooling along the dimension of point cloud number N, $F_{\max}^{N}$ is the maximum pooling along the dimension of point cloud number N. The channel attention mechanism adopts two-layer MLP, where w1 and w2 represent the shared weight of MLP. Finally, we activate it through the sigmoid function.The specific structure is shown in Figure 5.

3.4. Network architecture

Our network architecture is improved based on PointNet. PointNet consists of classification network and segmentation network in the process of 3D semantic segmentation. In Section 3.2, we have extracted the clustered point cloud data through the DBSCAN algorithm, so our improvement

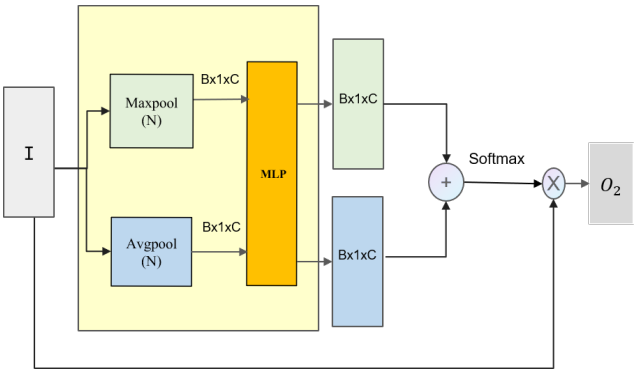

Figure 5. Point Cloud Attention Mechanism

is based on the PointNet classification network.We take each type of point cloud data after clustering as input,in order to enable the network to extract feature information more specifically and inhibit the extraction of unnecessary features, we make the point cloud data rotate spatially to the angle conducive to classification through two attention mechanisms after data input, and then multiply the affine matrix obtained by T-NET learning. Then the shared MLP is used for feature extraction, and the extracted multidimensional features are aligned by multiplying with the affine matrix. Then use MLP to extract multidimensional features again, and use Maxpool to obtain 1024 dimensional global features. There are four categories of our classification targets, namely, pedestrians, cars, bicycles and others. So finally, we reduce the feature dimension of global features through three full connection layers to four dimensions, that is, the score of each category to achieve the classification goal.

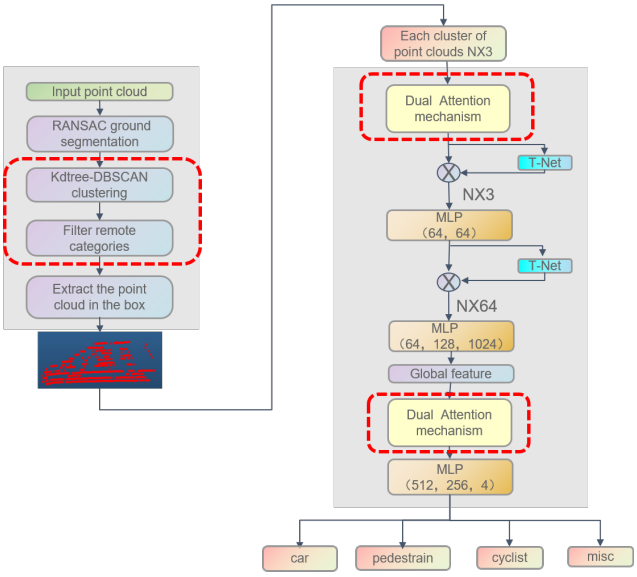

Figure 6. Network architecture

4. Experiments

Our experiment is divided into two parts. In the first part, we no longer use the network to segment each type of point cloud, but use the clustering idea to obtain four types of point cloud data. In the second section, we added attention mechanism on the basis of PointNet classification network, and classified the clustered point clouds to complete the target detection task in the automatic driving scene.

Point cloud clustering We first segment the KITTI autopilot dataset on the ground by RANSAC, and then classify the non ground data, using the DBSCAN clustering method suitable for point clouds. Through the experimental results, we can see that this clustering method makes good use of the sparsity of point cloud data and clusters the non ground data.

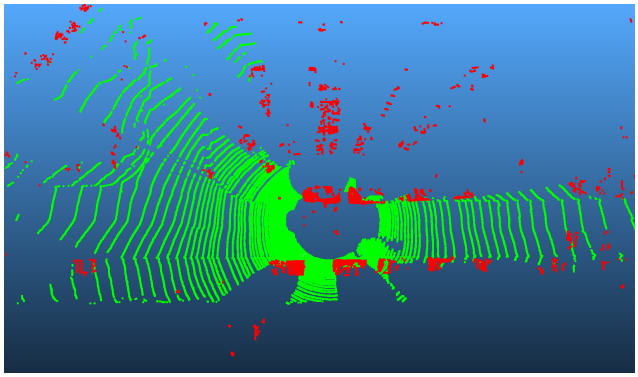

Figure 7. **Point cloud clustering** Red is the segmented object by DBSCAN clustering method, and some walls are removed through vector filtering

However, because the complexity of DBSCAN clustering depends on neighborhood point search, we use the improved Kdtree index method to query neighborhood points. Table 1 shows that the speed of clustering using Kdtree is nearly 20 times faster without changing the effect.

clustering method	time
DBSCAN	10.1s
Kdtree-DBSCAN	0.5s

Table 1. Time comparison before and after Kdtree acceleration

Network classification We input the clustered point clouds into our network for classification. Our original network is the classification part of PointNet. Considering the characteristics of point cloud data and in order to pay more attention to the extraction of important features, we add the designed attention mechanism before network input and before global feature dimensionality reduction. The finally clustered point clouds are successfully divided into four cat-

egories by the network: people, cars, bicycles and noise. The visualization of the results is shown in the figure.

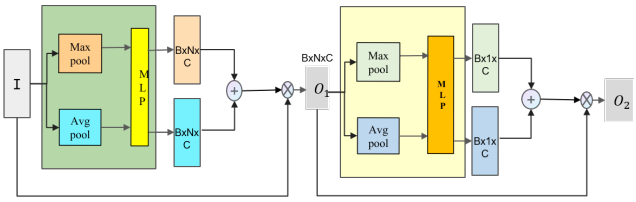

Figure 8. $A_c - A_c$ attention

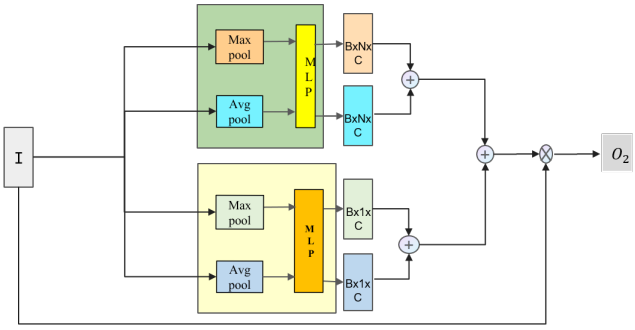

Figure 9. $A_n + A_c$ attention

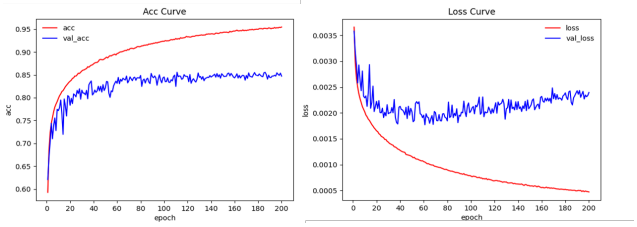

Figure 10. Accuracy and loss on KITTI training set

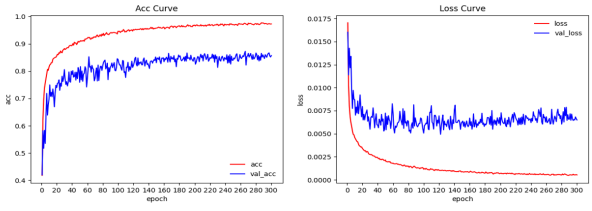

Figure 11. Accuracy and loss on Modelnet40 training set

Attention mechanism 1	KITTI	Modelnet40
no attention	0.8565	0.8620
A_n attention	0.8569	0.8576
A_c attention	0.8562	0.8594
$A_c - A_c$ attention	0.8621	0.8676
$A_n - A_c$ attention	0.8576	0.8661
$A_n + A_c$ attention	**0.876**	**0.8742**

Table 2. **Network classification accuracy.** This table is the accuracy statistics of various attention mechanisms after point cloud input. $A_c - A_c$ attention means that the channel attention mechanism is accessed first followed by the point cloud attention mechanism, $A_n - A_c$ means that the two are opposite, and $A_n + A_c$ attention means that the two attention mechanisms are accessed in parallel and the final weight is added.

Attention mechanism 2	KITTI	Modelnet40
no attention	0.8565	0.8620
A_n attention	0.8588	0.8676
A_c attention	0.8570	0.8776
$A_c - A_c$ attention	0.8656	0.8676
$A_n - A_c$ attention	0.8687	0.8716
$A_n + A_c$ attention	**0.8836**	**0.8796**

Table 3. **Network classification accuracy.** This table has the same attention mechanism as Table 1, but the difference is that various attention mechanisms are added after the global attention generated after MLP.

From the final experimental data, we can see that the overall accuracy of adding the attention mechanism to the global features is higher than that after the input. We tested five connection schemes for two attention mechanisms, channel and point cloud. From Table 2 and Table 3, the $A_n + A_c$ attention in parallel has the highest accuracy. Compared with the network without attention mechanism, the accuracy of our parallel attention mechanism in KITTI dataset can be improved by 2.71%, and the parallel attention mechanism on Modelnet40 dataset can be improved by 1.76% at most. In general, the classification time of the network we designed can be improved by nearly 20 times, and the accuracy can be improved by nearly 3 percentage points at most.The visualization of the final result is shown in Figure 12.

5. Conclusion

In this paper, we propose a clustering based target detection network, which is mainly composed of a part of PointNet classification network that adds attention mechanism. We found that DBSCAN algorithm can make good use of the sparsity of point cloud data to complete clustering, which provides a new perspective for semantic segmen-

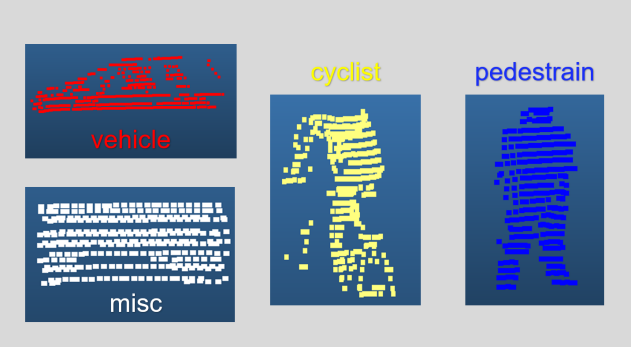

Figure 12. Classification visualization

tation of 3D point cloud. In addition, inspired by the spatial attention mechanism in the image field, in order to suppress unnecessary features, we designed two kinds of attention mechanisms for point cloud data, which are pooled along the two dimensions of channel number and point cloud number respectively. Without using convolution, MLP is used to generate attention weights, so that feature extraction follows the part that needs more attention. According to the different combination order of the two attention mechanisms, the same or better results as those of the prior art can be obtained on the standard benchmark.

References

[1] Ran Cheng, Ryan Razani, Ehsan Taghavi, Enxu Li, and Bingbing Liu. 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12547–12556, 2021. 2

[2] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019. 2

[3] Di Feng, Yiyang Zhou, Chenfeng Xu, Masayoshi Tomizuka, and Wei Zhan. A simple and efficient multi-task network for 3d object detection and road understanding. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7067–7074. IEEE, 2021. 2

[4] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12697–12705, 2019. 2

[5] Ke Li, Bharath Hariharan, and Jitendra Malik. Iterative instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3659–3667, 2016. 2

[6] Zhichao Li, Feng Wang, and Naiyan Wang. Lidar r-cnn: An efficient and universal 3d object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7546–7555, 2021. 2

[7] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1

[8] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 1

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[10] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 2

[11] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7652–7660, 2018. 2

[12] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9601–9610, 2020. 2

[13] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9939–9948, 2021. 2