FANet: Flexibility and Adaptivity Net for 3D Point Cloud Detection

Jian Ye , Fushan Zuo * , Yuqing Qian

Posted Date: 5 June 2023

doi: 10.20944/preprints202306.0287.v1

Keywords: autonomous driving; object detection; Position Adaptive Convolution; FANet

Article

FANet: Flexibility and Adaptivity Net for 3D Point Cloud Detection

Jian Ye [1], Fushan Zuo* and Yuqing Qian

College of Automobile and Traffic Engineering, Nanjing Forestry University, Nanjing 210037
* Correspondence: zuofushan@163.com; Tel: +86-137-7660-1801

Abstract: 3D object detection is essential for an accurate and reliable autonomous driving system. Currently, the methods used by the state-of-the-art two-stage detectors are not flexible enough and their feature extraction capabilities are very limited to cope effectively with the disorder and irregularity of point clouds. In this paper, we combine the advantages of both PV-RCNN and PAConv (Position Adaptive Convolution) to create a completely new network, FANet, in order to overcome the irregularity and disorder of point clouds. The convolution in our network builds convolutional kernels from a basic weight matrix, whose combined coefficients are learned adaptively by LearnNet from relative points. This network allows for flexible modeling of complex spatial variations and geometric structures in the 3D point cloud, enabling better extraction of point cloud features and producing high-quality 3D proposal boxes. Compared to other methods, FANet is superior in terms of 3D object detection accuracy. Extensive experiments on the KITTI dataset have shown a significant improvement in our approach.

Keywords: autonomous driving; object detection; Position Adaptive Convolution; FANet

1. Introduction

Point cloud data is a major form of 3D scene data, containing more 3D information such as length, width, height, velocity, and reflection angles than images. It has broad application prospects in fields such as autonomous driving, robot navigation, and virtual reality. Currently, 3D object detection algorithms based on point cloud data have become a research hotspot in this field. However, accurate identification of 3D objects remains a major challenge due to the inherent irregularity and disorder of the point cloud.

Researchers have proposed a number of solutions for 3D object detection, which can be broadly classified into two categories[1], namely voxel-based[2] approaches and point-based[3] approaches. The voxel-based approach converts irregular point cloud data into a regular voxel network, encodes the data points falling within the voxel and ex-tracts features through deep learning, and uses 3D convolutional neural networks or voxel-based deep learning to detect objects. This approach can achieve excellent detection accuracy, but inevitably results in causing loss of information and reduced fine-grained localization accuracy, which leads to reduced accuracy in object recognition. Point-based methods can easily achieve larger perceptual fields through point set abstraction, but are computationally more expensive. In addition, some researchers have recently designed convolutional kernels that deal directly with point clouds, which can be used for feature extraction and help in the subsequent generation of 3D proposal boxes. However, in practice, the lack of flexibility of the convolution kernel has led to a lack of accuracy.

In this paper, we present a high-accuracy 3D object detector, the FANet. This detector draws on the method in PAConv for constructing convolution kernels by dynamically combining basic weight matrices stored in a weight bank, and on the PV-RCNN voxel set abstraction module aggregation points and voxel features (These two networks will be mentioned in Methods). The convolution in our network builds convolution kernels from the basic weight matrix, and its combination coefficients are learned adaptively by LearnNet from relative points. On this basis, FANet can obtain multi-scale feature information, including multi-scale voxel semantic information and original point

 doi:10.20944/preprints202306.0287.v1

location information. Therefore, this network can flexibly handle irregularity and disordered point cloud, has strong feature extraction capability, and can accurately generate 3D proposal boxes.

2. Related Work

3D Object Detection with Voxel-based methods[2,4-9]. VoxelNet[10] divides point clouds into a certain number of voxels according to the ratio of length x width x height and divides points in point cloud space into corresponding voxels of location. It uses several individual feature coding layers for local feature extraction for each non-empty voxel. However, ignoring the sparse distribution of point cloud data in space wastes a lot of computation. Yan[11] introduced a new angular loss regression by improving the sparse convolution method, reducing the disadvantages of voxel-based network models in terms of irregularity and disorder. Based on SECOND, SA-SSD[12] method to preserve structure information is proposed. Dense Voxel Fusion[8] is a sequential fusion method that generates multi-scale dense voxel feature representations, improving expressiveness in low point density regions.This method produces more efficient 3D proposal boxes, however important geometric information may be lost due to quantization[13].

3D Object Detection with point-based approach[7,14,15]. Qi has designed an early work called PointNet[3] based on point cloud representation learning, which is simple and effective. Many subsequent point-based 3D object detection networks are developed on PointNet. Transform operations and maximum pooling in PointNet can effectively extract global features of points, but local features cannot be obtained. PointNet++[15] proposed Sampling and Grouping to improve local feature extraction for PointNet. In order to further reduce the inference time, 3D-SSD[16] only uses backbone for downsampling feature extraction to complete the detection task. SE-SSD[17] contains a pair of teacher and student SSDs, contains an efficient IoU-based matching strategy to filter the teacher's soft targets, and formulates a consistency loss to align the student's predictions with it. This approach provides flexible receptive fields for point cloud feature learning[18], however, which makes it difficult to express the complex variability of the point cloud space[19].

PV-RCNN[20], which is a two-stage 3D point cloud object detection, mainly uses a two-step strategy of point-voxel feature aggregation for accurate 3D object detection. This combines the advantages of both point-based and Voxel-based methods to obtain multi-scale feature information, including multi-scale Voxel semantic information and location information of points. The Voxel Set Abstraction Module in PV-RCNN draws on the concept in pointnet++, which is theoretically successful, but in practice has also traded off its design flexibility for effectiveness[21], resulting in poor object detection accuracy[22].

PAConv[23] builds a convolution kernel from a basic weight matrix, whose combined coefficients are learned adaptively by LearnNet from relative points. PAConv is flexible enough to accurately model complex spatial variations and geometric structures in 3D point clouds. However, PAConv itself does not have object detection capability. Therefore, this paper forms a new 3D object detector by embedding PAConv into the Pointnet++ module. The detector offers not only point-based and voxel-based advantages, such as its provision of flexible acceptance domains and efficient 3D proposal boxes, but also good object detection accuracy.

3. Methods

At present, the convolution used by 3D object detection in feature extraction is static[24]. In practical application, this method often reduces the model accuracy and expression ability due to insufficient computing resources [25]. This paper combines dynamic convolution to process irregular and disordered point cloud data through higher flexibility, which enables higher accuracy of 3D object detectors.

Considering the existence of the above problems, this paper proposes a high-accuracy 3D object detector, which can solve the above problems through dynamic convolution. In the 3D space of the point cloud, the relationship between points is complex. PAC can adapt the selection of convolution kernel to the changes around, and improve the accuracy of the 3D object detector. The overall framework of FANet is shown in Figure 1. This network incorporates PAConv into the Voxel Set

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

Abstraction module, which aggregates features from 3D sparse convolutional neural networks and keypoints collected through furthest point sampling, encoding them together. This, combined with high-quality proposals generated through voxel generation, results in the generation of highly accurate 3D proposal boxes.

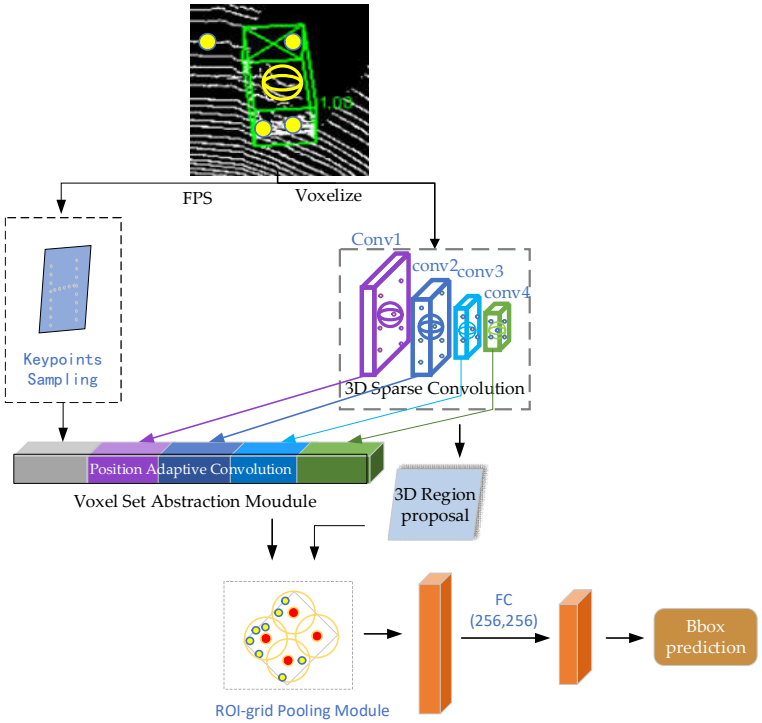


Figure 1. The overall framework of the FANet.

The voxel set abstraction module of PV-RCNN is capable of learning features directly from point cloud keypoints and voxels generated by the 3D sparse neural network, exhibiting a multi-level feature extraction structure. In this paper, we embed PAConv within the voxel set abstraction module to generate a dynamic network. In the dynamic network, dynamic network encoding is performed on each group of local neighborhoods, where the sampled and grouped points are assembled into dynamic kernels as input to PAConv, thereby increasing the feature dimensionality for each point to capture more informative features. Subsequently, the most significant features are selected as the new feature output through max pooling. In this process, the coordinate space dimension does not change, but the feature space dimension becomes higher, in other words, the number of feature channels increases, allowing more feature information to be obtained and facilitating subsequent accurate feature extraction.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

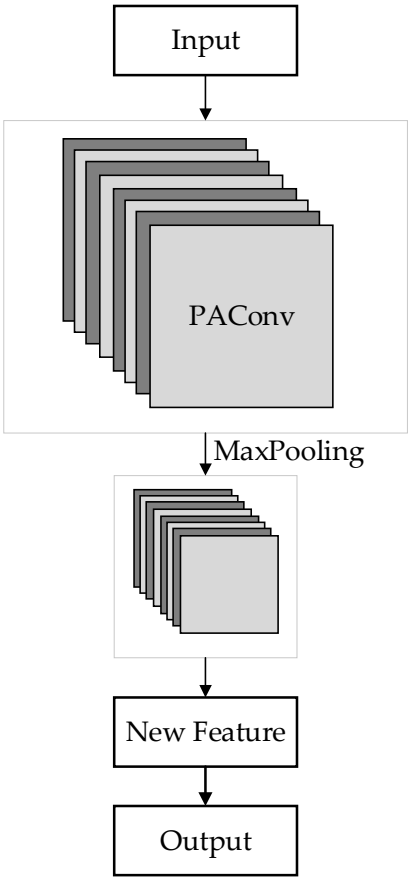


Figure 2. Dynamic Network.

On the one hand, this network generates the gridded features from the original point cloud with 3D sparse convolution, and projects the downsampled 8× feature map to Birds-Eye View (bew), generating 3D prediction frames, two prediction frames for each pixel and each class, 0° and 90°, respectively. The keypoints are sampled uniformly from the surrounding area of the proposal by sectorized proposal-centric keypoint sampling (spc). On the other hand, this network determines the nearest neighbors under one radius of each grid point, and then a PointNet++ module embedded in PAConv is used to integrate the features into the features of the grid points. This particular PointNet++ module has more flexibility to better handle irregular and disordered point cloud data. After the point features of all grid points are obtained by means of multi-scale feature fusion, a 256-dimensional proposal feature is obtained by using a two-layer multilayer perceptron (MLP). The above feature information is fused with multi-scale features by the voxel set abstraction module to obtain new multi-scale features. The new features are refined to derive a more accurate 3D prediction box.

3.1. Position Adaptive Convolution Embedded Network

In the 3D space, the relationship between points is very different from the relationship between points in the 2D plane. In 2D plane space, the features learned by using convolutional neural network can reflect the correlation between points well; but in 3D space, due to the disorderly and irregularity of points, using the convolutional kernel operator for learning features in 2D space will make the correlation poor, which leads to inaccurate detection.

Therefore, we redesigned the convolutional kernel function to enable it to learn point features dynamically. As shown in Figure 3, we first defined a weight library consisting of several weight matrices. Next, we designed a vector of LearnNet learning coefficients to combine the weight matrices based on the location of the points. Finally, we generate dynamic kernels by combining the weight

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

matrices and their associated positional adaptation coefficients. In this way, the kernel is constructed in a data-driven manner, giving our approach greater flexibility than 2D convolution to better handle irregular and disordered point cloud data. In addition, we make the learning process less complex by combining weight matrices rather than forcing kernels to be predicted from the location of points.

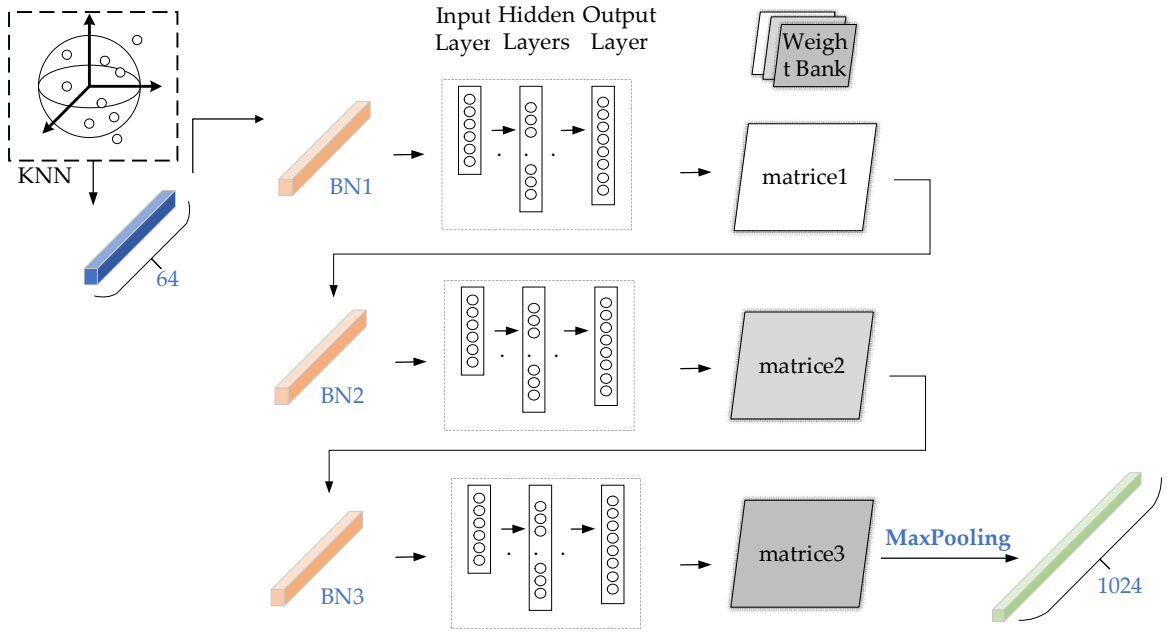


Figure 3. The framework of Position Adaptive Convolution.

Firstly, a weight bank consisting of several weight matrices is defined. Then, the LearnNet is designed. The weighting matrix is combined according to the location of the points. Finally, a dynamic kernel is generated by combining the weighting matrix and its related location adaptive coefficients. Where the weight bank W is defined as Formula 1, $W_T \in R^{C_p * C_q}$ is the weight matrix, C_P is the input channel, C_q is the output channel. T is the number of weight matrices controlling the weights stored in the weight repository W. The larger the value of T, the more diverse the weight matrix assembled by the kernel. However, too many weight matrices can cause redundancy and heavy memory and computing overhead. Experiments[23] have shown that can achieve optimal accuracy when T is 8. The next step is to create a mapping of the discrete kernel of the weight matrix to the continuous 3D space. We use LearnNet to learn coefficients based on the location relationships of the points to combine the weight matrix and produce a dynamic kernel that fits the point cloud. The input to LearnNet is a specific positional relationship between the center point $D_i(x_i, y_i, z_i)$ and the neighboring point $D_j(x_j, y_j, z_j)$of a local region in the point cloud, LearnNet predicts the location adaptive coefficient $A^T_{j,i}$ of each weight matrix w_T, the expression of point location relationship P is formula 2.

$$W = \{W_T | T = 1, \dots, t\} \tag{1}$$

$$P = (D_j - D_i, D_i) \tag{2}$$

Expression of the output vectors Ai,j of LearnNet:

$$A_{i,j} = \text{softmax}\left(\text{relu}(\delta(\text{P}))\right) \tag{3}$$

$$A_{i,j} = \{A^T_{i,j} | T = 1, \dots, t\} \tag{4}$$

In the formula (3), δ is a non-linear function implemented for use with a multilayer sensor; Relu is the activation function; Softmax is a normalization function; The score of the softmax output is in the (0,1) range. This normalization ensures that each weight matrix is selected with probability, with

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

higher scores indicating stronger relationships between the position input and the weight matrix. The higher the score, the stronger the relationship between the location input and the weight matrix. PAConv's dynamic kernel K combines the weight matrix in W with the input characteristic F_P, and then multiplies with the corresponding coefficient $A_{i,J}$ of the LearnNet prediction to get the dynamic kernel K, which is formula 5:

$$K(P) = \sum_{T=1}^{t} A_{j,i}^{T} W_T F_p \tag{5}$$

With the dynamic kernel, the generated adaptive dynamic convolution can learn features more flexibly, so the larger the size of the weight bank, the greater the flexibility and availability of the weight matrix. This kernel assembly strategy allows for flexible modeling of irregular geometries of point clouds.

The weight matrices are randomly initialized and may converge to similar matrices, which does not guarantee the diversity of the weight matrices. Therefore, to avoid this situation, this paper penalizes the correlation between different matrices by a weight regularization function. which ensures the diversity of weights, making the generated kernels also diverse. Which is defined as:

$$\tau_{corr} = \sum_{A_i, A_j \in A, i \neq j} \frac{|\sum A_i A_j|}{\|A_i\|_2 \|A_j\|_2} \tag{6}$$

This approach allows the network to be more flexible in learning features from point clouds, to represent the complex spatial variation of point clouds with relative accuracy, and to have sufficient point cloud geometry information to enable more accurate object detection, resulting in a significant improvement in object detection accuracy over the original method.

Table 1. List of symbols.

symbols	significance
W	weight bank
W_T	weight matrix
C_P	input channel
C_q	output channel
T	number of weight matrices
D_i	center point
D_j	neighboring point
$A^T_{i,j}$	location adaptive coefficient
Relu	activation function
Softmax	normalization function
K	dynamic kernel
F_P	input characteristic
τ_{corr}	weight regularization

4. Experimental Results and Analysis

This experiment is completed under Ubuntu 20.04 operating system. The experimental environment is Intel(R) Core (TM) i7-11700H for the central processor (CPU), NVIDIA GeForce RTX 3070 Laptop, PyTorch version 1.8.1, and Python version 3.9.

4.1. Dataset

Currently, the KITTI[26] dataset, Waymo[27] dataset and nuScenes dataset are commonly used for 3D object detection. The KITTI dataset is one of the most important test sets in the field of auto-driving. It plays an important role in 3D object detection. Therefore, this paper chooses KITTI 3D point cloud dataset as the training and test for the experiment.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

The KITTI dataset contains 7481 training sets and 7518 test samples. The dataset consists of four parts: Calib, Image, Label and Velodyne. Calib is the camera calibration file; Image is a two-dimensional image data file; Label is a data label file (there is no Label in the test set). There are three types of objects labeled by the data: Car, Cyclist and Pedestrian. Velodyne is a point cloud file. For efficiency reasons, Velodyne scans are stored as floating-point binary files, with each point stored in its (x, y, z) coordinates and additional reflection value R.

This network model uses, in order, 80% of the training sample data to train the model and the remaining 20% for validation.

4.2. Evaluation Metrics

In this paper, the above-mentioned optimized network is accurately compared by analyzing *Precision*, *Recall*, *AP(average precision)*, et al. The formula is as follows.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{7}$$

$$Recall = \frac{TP}{TP+FN} \tag{8}$$

where *TP*, *TF*, and *FN* denote true positive, false positive, and false negative, respectively. *AP* and *mAP* are commonly used as evaluation metrics in the field of object detection, reflecting the overall performance of the model. The value of $_{AP}$ is calculated from the area formed between the *Precision*, *Recall*, and the horizontal and vertical axes. The value of *mAP* represents the average of all *AP* values, which is shown in the formula.

$$AP = \sum_{i=1}^{n-1}(r_{i+1} - r_i)P_{inter}(r_i + 1) \tag{9}$$

$$mAP = \frac{\sum_{i=1}^{k} AP_i}{k} \tag{10}$$

4.3. Data and Analysis

To verify the effectiveness of the algorithm, the PV-RCNN before and after optimization will run in the same environment. Also compared with other advanced methods mean Average Precision.

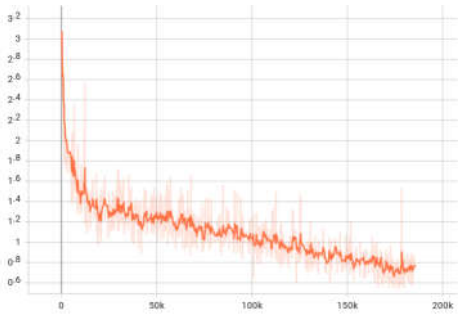


Figure 4. loss of PC-RCNN.

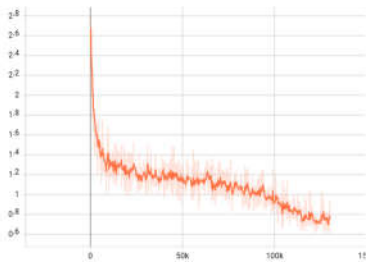
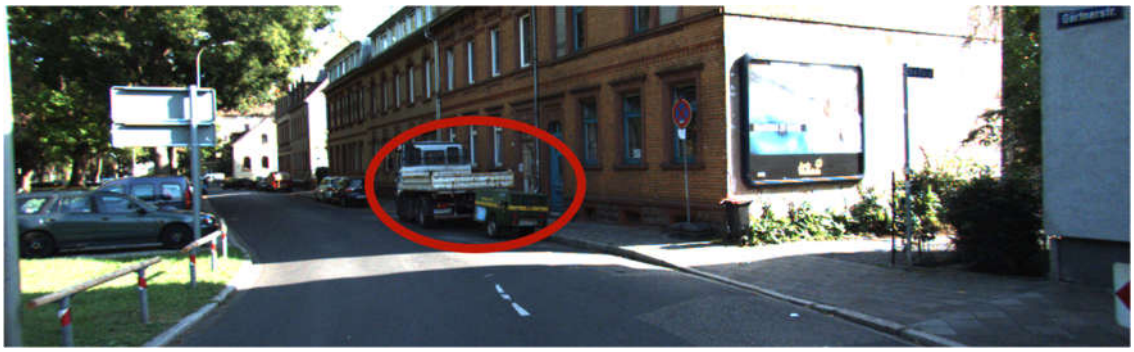


Figure 5. loss of FANet.

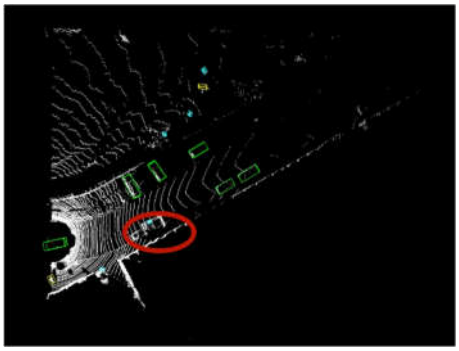
Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

According to Figure. 4 and Figure. 5, FANet has faster network convergence than PV-RCNN. FANet only needs a limited number of iterations to get good results, while PV-RCNN needs more iterations to get the effect as FANet.

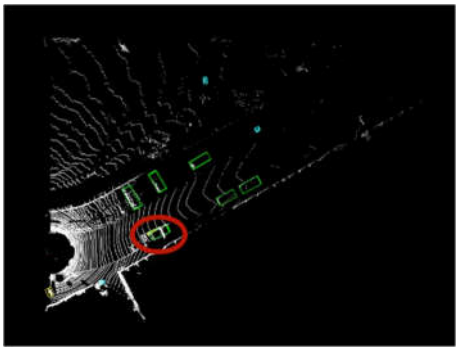
In this paper, to show that our method has a higher accuracy than PV-RCNN, we have selected five random sets of images for comparison. The (a) of each set of images is the camera photo, the (b) of each set of images is the result of the point cloud information identified by PV-RCNN, and the (c) of each set of images is the result of the point cloud information identified by FANet. We use the image information to determine which method is better and more accurate for object detection.



(a) Image information



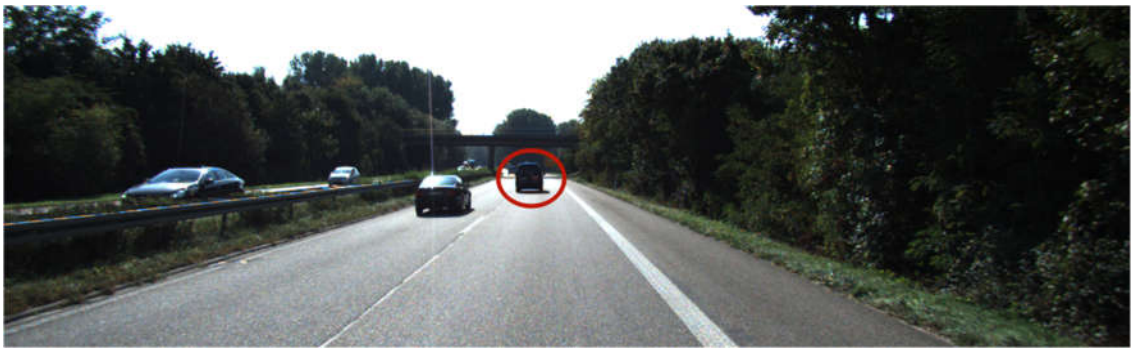
(b) Visualization of PV-RCNN results



(c) Visualization of FANet results

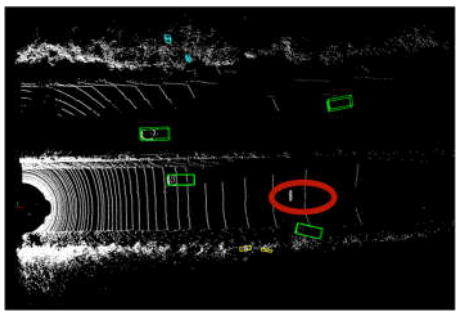
Figure 6. Scene 1.

In the scene 1，according to the image information, FANet can recognize the truck on the right front, while PV-RCNN cannot accurately recognize it.

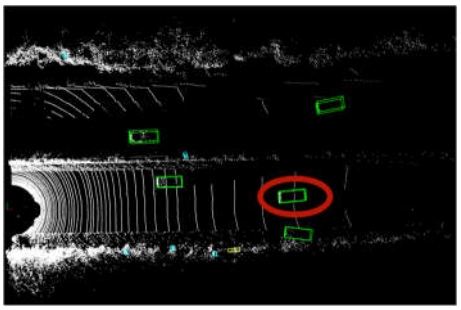


(a) Image information

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1



(b) Visualization of PV-RCNN results.

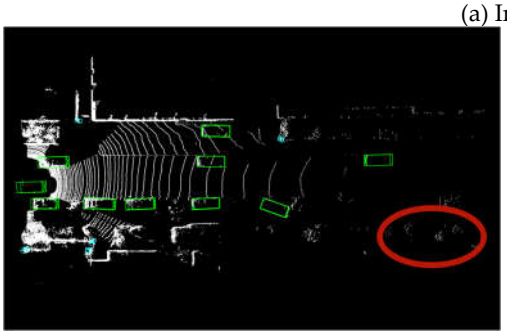


(c) Visualization of FANet results.

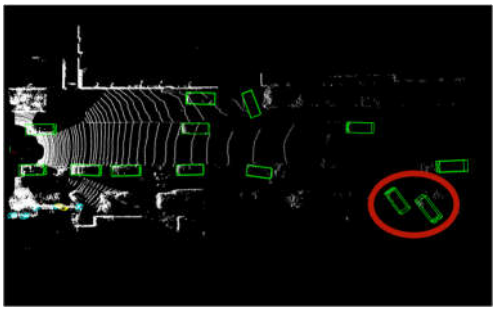
Figure 7. Scene 2.

In the scene 2, according to the image information, FANet can recognize the car in front, while PV-RCNN cannot accurately recognize it.

(a) Image information



(b) Visualization of PV-RCNN results.



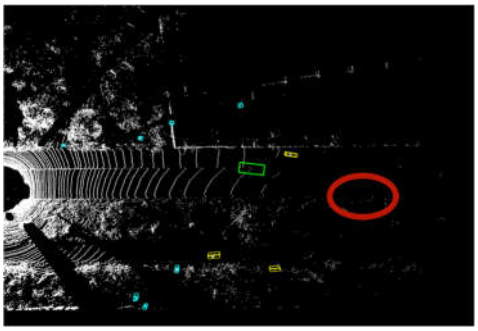
(c) Visualization of FANet results.

Figure 8. Scene 3.

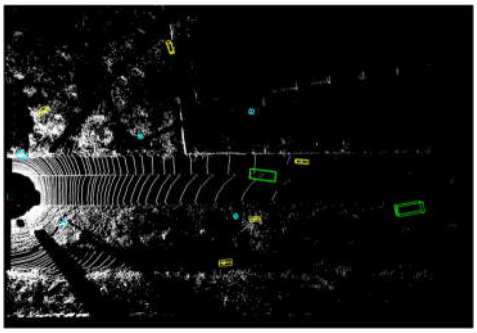
In the scene 3, according to the image information, FANet can recognize the car distant car on the right front, while PV-RCNN cannot accurately recognize it.

(a) Image information

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1



(b) Visualization of PV-RCNN results.

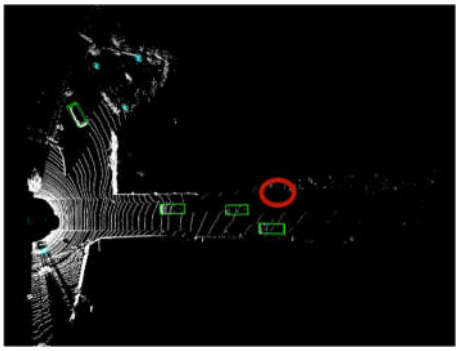


(c) Visualization of FANet results.

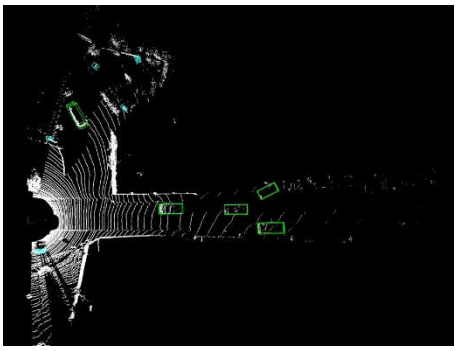
Figure 9. Scene 4.

In the scene 4, according to the image information, FANet can recognize the car distant car in front, while PV-RCNN cannot accurately recognize it.

(a) Image information



(b) Visualization of PV-RCNN results.



(c) Visualization of FANet results.

Figure 10. Scene 5.

In the scene 5, according to the image information, FANet can recognize the car distant car in front, while PV-RCNN cannot accurately recognize it.

In the above five sets of scenes,the image information provide strong evidence that FANet outperforms PV-RCNN in terms of object detection accuracy. Specifically, FANet is able to identify objects in a variety of scenes, demonstrating its superior ability to detect objects in complex and distant scenes. In contrast, PV-RCNN is unable to accurately recognize objects. When we consider both the camera photos and the point cloud data, it becomes even more apparent that FANet is a more accurate 3D object detector than PV-RCNN. The point cloud data allows our method to extract key features from the scene, and our approach enables us to effectively deal with point cloud irregularity and disorder. These features, combined with the ability to accurately recognize objects in complex scenes, lead to significantly improved accuracy compared to PV-RCNN.

In this paper, we record the average accuracy of each type of object recognition with IoU threshold of 0.7(car) and 0.5(Pedestrian and Cyclist). Table 1 - Table 3 shows the comparison of object

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 June 2023 doi:10.20944/preprints202306.0287.v1

detection effect of the PV-RCNN network model before and after optimization for Car, Pedestrian and Cyclist. Table 4 shows how our method compares with state-of-the-art methods with respect to mAP.

Table 2. Comparison of Car class recognition accuracy (Unit: %) of PV-RCNN and FANet.

	Car-3D			Car-BEV		
Method	Easy	Mod	Hard	Easy	Mod	Hard
PV-RCNN	89.66	81.82	78.06	93.00	88.58	88.25
FANet	92.41	82.82	80.30	94.06	90.63	91.30

Table 3. Comparison of Pedestrian class recognition accuracy (Unit: %) of PV-RCNN and FANet.

	Pedestrian-3D			Pedestrian-BEV		
Method	Easy	Mod	Hard	Easy	Mod	Hard
PV-RCNN	63.89	56.35	51.31	66.87	59.79	55.60
FANet	65.53	58.11	52.06	66.21	60.21	55.37

Table 4. Comparison of Cyclist class recognition accuracy (Unit: %) of PV-RCNN and FANet.

	Cyclist-3D			Cyclist-BEV		
Method	Easy	Mod	Hard	Easy	Mod	Hard
PV-RCNN	87.03	68.70	64.38	93.32	75.07	70.49
FANet	90.10	71.27	66.42	89.53	77.34	71.10

Table 5. mAP Comparison of the accuracy (Unit: %) of different algorithms on the KITTI dataset. Results are evaluated by average accuracy and 40 recall positions.

	3D-mAP		
Method	Easy	Mod	Hard
STD	77.89	67.71	62.85
Part-A2	77.75	66.49	61.27
3DSSD	78.30	67.57	62.31
CT3D	77.77	69.77	64.92
VoTR	79.84	70.09	**66.90**
PV-RCNN	80.19	68.96	64.58
PV-RCNN++	80.30	69.41	64.91
FANet	**82.68**	**70.73**	66.26

Table 2, 3 and 4 display the experimental results of the FANet model on the KITTI validation dataset. The accuracy improvements achieved by FANet are noteworthy, with almost all projects surpassing the state-of-the-art PV-RCNN method. In particular, the average accuracy of the Car class has been improved by 3.05%, demonstrating the effectiveness of the proposed new network in improving target detection accuracy. The Pedestrian class showed the highest increase, with a maximum improvement of 1.76% in average precision. Similarly, the average precision of the Cyclist class increased by up to 3.07%. Although PV-RCNN has a slight advantage in pedestrian and cyclist bird's eye view (BEV) detection, the proposed FANet model demonstrates superior overall performance in terms of accuracy improvement. These results validate the effectiveness of the proposed PAConv module in enhancing the performance of 3D object detection models.

Table 5 shows the evaluation of our proposed method on the KITTI benchmark dataset and compares it with the state-of-the-art method, which includes PV-RCNN++[1]. PV-RCNN++ is a new network proposed by Shi based on PV-RCNN. The results indicate that our method achieved superior performance in all three difficulty levels, i.e., Easy, Moderate, and Hard. Specifically, our method demonstrated a maximum increase of 2.49% in accuracy compared to PV-RCNN. Even

compared with PV-RCNN++, this improvement is significant, indicating that our method has a greater advantage in terms of 3D object detection accuracy. In addition, when compared with other existing methods, our approach achieved first place in the Easy and Moderate levels, and second place in the Hard level. The difference in performance between our method and the first-place method in the Hard level was just 0.64%, which is quite small. This demonstrates that our approach is highly competitive and performs at a level that is comparable to the current state-of-the-art methods. Overall, these results confirm that our proposed 3D-MAP method is highly effective in improving the accuracy of 3D object detection, and has a significant advantage over the state-of-the-art PV-RCNN method.

5. Conclusions

In this paper, we propose a high-precision 3D object detector, FANet, which combines the advantages of both PV-RCNN and PAConv.

PV-RCNN is a state-of-the-art 3D object detection method that extracts features from point clouds using a Region Proposal Network (RPN) and Point-Voxel Feature Encoding (PVFE) module, while PAConv is a novel point cloud convolution method that allows for adaptive learning of convolution weights based on the distribution of point clouds.To overcome the limitations of these methods and improve the accuracy of 3D object detection, FANet uses an adaptive learning mechanism that dynamically adjusts the weight of each point based on its relative position in the point cloud. This approach allows FANet to effectively handle the irregularity and disorder of point clouds and extract more accurate and robust features for object detection. To evaluate the effectiveness of FANet, experiments were conducted on the KITTI dataset, which is a widely used benchmark for 3D object detection. The results show that FANet significantly outperforms PV-RCNN in terms of 3D object detection accuracy. Specifically, FANet achieved an increase in the average precision of up to 3.07% for the Cyclist class, 3.05% for the Car class, and 1.76% for the Pedestrian class, compared to PV-RCNN Moreover, FANet also achieved better 3D object detection accuracy compared to other state-of-the-art methods, which demonstrates its competitiveness and effectiveness in the field of 3D object detection. The proposed FANet method provides a significant improvement in the accuracy of 3D object detection and shows the potential of integrating multiple existing methods for further improvement in the field.

While the FANet method shows a significant improvement in 3D object detection accuracy, it still has limitations in certain challenging scenarios, such as adverse weather conditions like rain and fog. To address this issue, we plan to optimize the network for such scenarios in future work.

Author Contributions: Conceptualization, F.Z. and J.Y.; methodology, F.Z. and J.Y.; software, J.Y. and Y.Q.; validation, F.Z. and J.Y.; formal analysis, F.Z. and J.Y.; investigation, Y.Q.; data curation, F.Z.; funding acquisition, F.Z. and J.Y. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shi, S.; Jiang, L.; Deng, J.; Wang, Z.; Guo, C.; Shi, J.; Wang, X.; Li, H.J.a.p.a. PV-RCNN++: Point-voxel feature set abstraction with local vector representation for 3D object detection. **2021**, doi:10.1007/s00371-022-02672-2.
2. Li, B. 3D Fully Convolutional Network for Vehicle Detection in Point Cloud. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, CANADA, Sep 24-28, 2017; pp. 1513-1518.
3. Qi, C.R.; Su, H.; Mo, K.C.; Guibas, L.J.; Ieee. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the 30th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, Jul 21-26, 2017; pp. 77-85.

4. Engelcke, M.; Rao, D.; Wang, D.Z.; Tong, C.H.; Posner, I. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017; pp. 1355-1361.
5. Ye, Y.Y.; Chen, H.J.; Zhang, C.; Hao, X.L.; Zhang, Z.X. SARPNET: Shape attention regional proposal network for liDAR-based 3D object detection. *Neurocomputing* **2020**, *379*, 53-63, doi:10.1016/j.neucom.2019.09.086.
6. Deng, J.J.; Shi, S.S.; Li, P.W.; Zhou, W.G.; Zhang, Y.Y.; Li, H.Q.; Assoc Advancement Artificial, I. Voxel R-CNN: Towards High Performance Voxel-based 3D Object Detection. In Proceedings of the 35th AAAI Conference on Artificial Intelligence / 33rd Conference on Innovative Applications of Artificial Intelligence / 11th Symposium on Educational Advances in Artificial Intelligence, Electr Network, Feb 02-09, 2021; pp. 1201-1209.
7. Yang, Z.T.; Sun, Y.A.; Liu, S.; Shen, X.Y.; Jia, J.Y.; Ieee. STD: Sparse-to-Dense 3D Object Detector for Point Cloud. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, SOUTH KOREA, Oct 27-Nov 02, 2019; pp. 1951-1960.
8. Mahmoud, A.; Hu, J.S.; Waslander, S.L. Dense voxel fusion for 3D object detection. In Proceedings of the Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023; pp. 663-672.
9. Li, Y.; Chen, Y.; Qi, X.; Li, Z.; Sun, J.; Jia, J.J.a.p.a. Unifying voxel-based representation with transformer for 3d object detection. **2022**, doi:10.1109/iros.2013.6696881.
10. Zhou, Y.; Tuzel, O.; Ieee. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In Proceedings of the 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, Jun 18-23, 2018; pp. 4490-4499.
11. Yan, Y.; Mao, Y.X.; Li, B. SECOND: Sparsely Embedded Convolutional Detection. *Sensors* **2018**, *18*, doi:10.3390/s18103337.
12. C., H.; H., Z.; J., H.; X.-S., H.; L., Z. Structure Aware Single-Stage 3D Object Detection from Point Cloud %J Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. **2020**, doi:10.1109/cvpr42600.2020.01189.
13. Mao, J.; Xue, Y.; Niu, M.; Bai, H.; Feng, J.; Liang, X.; Xu, H.; Xu, C.; Ieee. Voxel Transformer for 3D Object Detection. In Proceedings of the 18th IEEE/CVF International Conference on Computer Vision (ICCV), Electr Network, 2021 Oct 11-17, 2021; pp. 3144-3153.
14. Chen, C.; Chen, Z.; Zhang, J.; Tao, D. SASA: Semantics-Augmented Set Abstraction for Point-based 3D Object Detection. **2022**.
15. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet plus plus : Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, Dec 04-09, 2017.
16. Yang, Z.; Sun, Y.; Liu, S.; Jia, J. 3dssd: Point-based 3d single stage object detector. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020; pp. 11040-11048.
17. Zheng, W.; Tang, W.; Jiang, L.; Fu, C.-W.; Ieee Comp, S.O.C. SE-SSD: Self-Ensembling Single-Stage Object Detector From Point Cloud. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Electr Network, 2021 Jun 19-25, 2021; pp. 14489-14498.
18. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum pointnets for 3d object detection from rgb-d data. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018; pp. 918-927.
19. Sheng, H.L.; Cai, S.J.; Liu, Y.; Deng, B.; Huang, J.Q.; Hua, X.S.; Zhao, M.J.; Ieee. Improving 3D Object Detection with Channel-wise Transformer. In Proceedings of the 18th IEEE/CVF International Conference on Computer Vision (ICCV), Electr Network, Oct 11-17, 2021; pp. 2723-2732.
20. 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 Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020; pp. 10529-10538.
21. 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 Transactions on Pattern Analysis and Machine Intelligence* **2021**, *43*, 2647-2664, doi:10.1109/tpami.2020.2977026.
22. Bhattacharyya, P.; Czarnecki, K.J.a.p.a. Deformable PV-RCNN: Improving 3D object detection with learned deformations. **2020**, doi:10.1007/s11263-022-01710-9.
23. Xu, M.T.; Ding, R.Y.; Zhao, H.S.; Qi, X.J.; Ieee Comp, S.O.C. PAConv: Position Adaptive Convolution with Dynamic Kernel Assembling on Point Clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Electr Network, Jun 19-25, 2021; pp. 3172-3181.
24. Thomas, H.; Qi, C.R.; Deschaud, J.-E.; Marcotegui, B.; Goulette, F.; Guibas, L.J.; Ieee. KPConv: Flexible and Deformable Convolution for Point Clouds. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, SOUTH KOREA, 2019. Oct 27-Nov 02, 2019; pp. 6420-6429.

25. Chen, Y.; Dai, X.; Liu, M.; Chen, D.; Yuan, L.; Liu, Z. Dynamic convolution: Attention over convolution kernels. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020; pp. 11030-11039.
26. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R.J.T.I.J.o.R.R. Vision meets robotics: The kitti dataset. **2013**, *32*, 1231-1237, doi:10.1177/0278364913491297.
27. Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020; pp. 2446-2454.