LiDAR Target Detection for Automatic Berthing and De-berthing Scenarios

Liang Yue, Chuang Zhang*, and Muzhuang Guo

Dalian maritime university, Liaoning, Dalian, 116026, China

*Corresponding author's e-mail: zhchuangdmu@163.com

Abstract. Intelligent ships face the problem of accurate and real-time perception of the surrounding environment during the berthing and de-berthing. This paper proposes a Poly YOLO detector based on the YOLOv3 network. Firstly, the detection rate and efficiency of the Poly-YOLO structure is enhanced by introducing the dilated convolution and self-attention module into it; secondly, the LIDAR point cloud data is projected onto the 2D plane, the information of the 2D sparse depth map is enriched to generate the dense depth map using the depth up-sampling method, the data is fed back to the Poly-YOLO detection and recognition network, and the detection is accomplished by using the detection head. The experimental results show that this method can effectively improve the accuracy of the detection of point clouds and ensure real-time performance.

Keywords: Poly YOLO detector; Dilated convolution; Self attention module (SAM); Point cloud;

1 Introduction

With the rapid development of deep learning, automation and intelligent technology are continuously driving the progress of waterway traffic control, and automatic berthing and de-berthing technology is receiving more and more attention. It is not only necessary to achieve high detection accuracy, but also to have efficiency requirements for real-time detection speed. Deep learning based methods have stronger generalization performance and accuracy, and are gradually replacing traditional algorithms as the mainstream ship target detection algorithm.

And in recent years, at present, target detection algorithms based on two-dimensional images have made great progress, but detection algorithms for three-dimensional detection are still under intensive research, and the main sensors used for three-dimensional detection are LiDAR, millimeter-wave radar and depth camera. Among them, the data captured by LIDAR is different from the image, which is a three-dimensional point cloud rather than a two-dimensional image, and according to the different point cloud processing methods, the detection methods can be divided into: target detection methods based on the original point cloud, based on the voxelization of the point cloud, based on the point cloud projection.

© The Author(s) 2024
G. Zhao et al. (eds.), *Proceedings of the 2024 7th International Symposium on Traffic Transportation and Civil Architecture (ISTTCA 2024)*, Advances in Engineering Research 241,
https://doi.org/10.2991/978-94-6463-514-0_33

(1) Typical algorithms based on the original point cloud are PointNet by QI et al [1], Center-Poin6 by YIN et al [2], and 3DSSD by YANG et al [3]. These algorithms have large arithmetic capacity and slow operation speed, with the advantage of being able to maximize the retention of the object's positional information in the three-dimensional space, which is more suitable for instance segmentation-type tasks.

(2) Typical algorithms based on point cloud voxelization are VoxeINet proposed by ZHOU et al [4], Pointpilars proposed by LANG et al [5], and Voxelr cnn proposed by DING et al [6]. These algorithms organize disordered point cloud data into ordered voxel expressions, which effectively improves the speed of network processing compared to the direct processing of the point cloud, but in the face of the uneven distribution of the point cloud, a large number of null voxels are generated, which increase the additional computational effort.

(3) Typical projection-based algorithms include MV3D proposed by CHEN et al [7]. and BirdNet+ proposed by BARREARA et al [8]. Such methods complete target detection by projecting the point cloud into two-dimensional views with different angles, and then utilize the mature two-dimensional target detection network to complete the target detection, which has less computation, faster speed and higher accuracy. Among the projection-based pedestrian detection methods, they can generally be categorized into front-view (FV) projection and bird's eye-view (BEV) projection, etc [9]. In the unmanned field, BEV projection is used because front-view projection has the problem of object occlusion and driving is prone to danger. Complex-YOLO proposed by SIMONY et al [10]. Fewer dimensions mean fewer computations, faster detection speed, but the reduced dimension data is more sparse, the feature extraction network of the original algorithm can't meet the learning enough features.

To address the above problems, an improved Poly-YOLO detector method based on YOLOv3 is proposed, and by introducing dilated convolution and Self-Attention Module (SAM) in the Poly-YOLO structure. The 3D point cloud of LIDAR scans is then projected onto a 2D pixel plane; the sparse depth map obtained from the projection contains insufficient information, so a dense depth map is generated using a depth upsampling method. A new SE-Darknet-53 backbone network in the detector is used to enhance the detection performance of the network and reduce the number of parameters. Finally the results are output through the detection header.

2 Methodology

2.1 Poly-YOLO

In order to investigate a fast and accurate target detector for automated park-and-drive systems, this paper focuses on a single-stage approach and tries to explore its performance enhancement potential. Based on the efficient YOLOv3, Poly-YOLO is proposed and eliminates two weaknesses of YOLOv3: rewritten labels and inefficient distribution of anchors.

For the label rewriting problem, it can only be realized by either increasing the input image resolution size; or increasing the output feature map size. The approach in this paper is to increase the output feature map size. Improvements to the Anchor distribu-

tion problem-The problem posed by k-mean clustering has the following two solutions:

(1) The k-mean clustering process remains unchanged, but to avoid the problem of small objects being assigned to train on top of small output feature maps and large objects being assigned to train on top of large output feature maps, it is specific that firstly, based on the sense field of the output layer of the network, three approximate ranges of scales are defined, and then two thresholds are set to forcibly discretize the three scales to separate them; and then the bboxes are clustered individually three times, and each time the clustering is carried out in the previously specified range is performed by selecting a specific bbox, rather than acting on the entire dataset.

(2) Just one output layer, all objects are predicted in this layer. The k-mean clustering problem can be avoided, but in order to prevent label rewriting, so the output resolution is adjusted upwards, which is perfect at this point. The authors actually use a 1/4 scale output, which is a high resolution output with a very low rewrite probability.

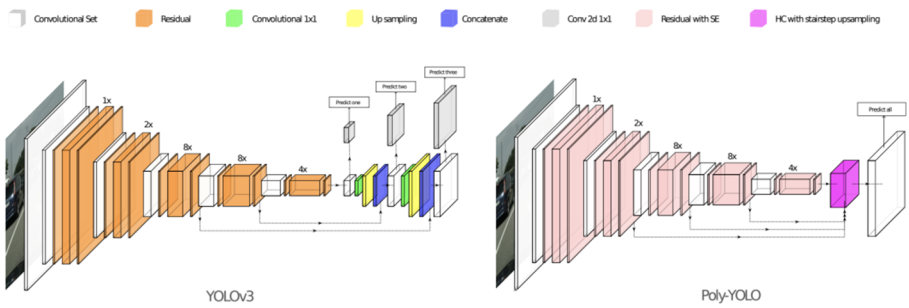

Fig. 1. Structure diagram of YOLOv3 and Poly-YOLO.

It can be found according to the Figure 1:

(1) For the network, in order to reduce the number of parameters, the number of channels is firstly reduced, while in order to improve the performance, SE units are introduced to enhance the features.

(2) The biggest difference from YOLOv3 is that the output layer is one, but also uses multi-scale fusion.

(3) The neck section presents the hypercolumn+stairstep upsampling operation.

2.2 Dilated Convolution and Self-Attention Module (SAM)

Dilated Convolution.

In order to learn discriminative feature maps, we replaced the standard convolution with dilated convolution in the SE-Darknet-53 backbone. Dilation convolution was known in the past as convolution using dilated filters and plays a key role in the átrous algorithm. Later, semantic segmentation was further used to aggregate multi-scale contextual feature maps without loss of image resolution. Mathematically, the convolution operation between two functions can be described as follows:

$$(f*g)(r)=\sum_{m+n=r} f(m)*g(n) \tag{1}$$

Where f is a discrete function with kernel size of m, g is the filter with the size of n, and r indicates the size of receptive field.

Besides, dilated convolution is formulated as follows:

$$(f*_k g)(r)=\sum_{m+kn=r} f(m)*g(n) \tag{2}$$

Where k denotes dilated rate.

As shown in the figure 2, there are at least two key advantages of using the dilated convolution. First, it provides larger receptive field, and allows the model to focus on local feature information. Second, the introduction of dilated convolution rarely incurs additional computational cost, and thus ensures fast convolutional operation and efficient inference.

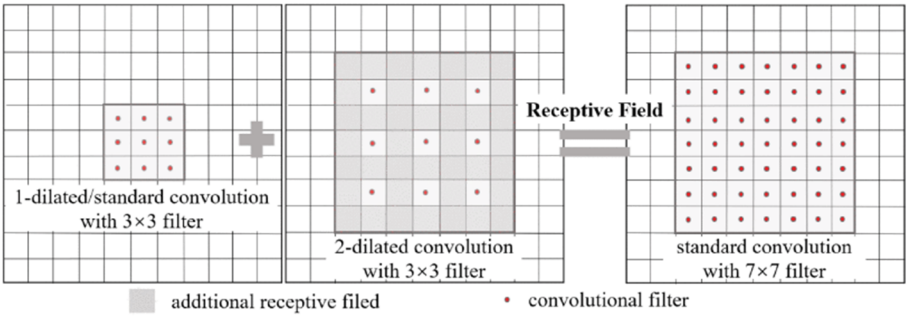

Fig. 2. Overview of Dilated Convolution.

Self-Attention Module (SAM).

In order to highlight useful region and model feature relationships, we further investigate the self-attention mechanism that Transformer successfully employs in machine translation. In Transformer, the input is divided into three components of query (Q), key (K) and value (V), as shown in the figure 3. First the dot products all key queries is calculated and then softmax function is placed on matrix multiplication result to obtain its weight on the value. It is known as self-attentive mechanism.

In this work, we design the Self-Attention Module (SAM) through Transformer and formulate it as a unique attention mechanism for model feature relations. The above figure (b) shows that the input is first divided into q, k and v branches. Subsequently 3 × 3 convolutions with 1, 2 and 5 dilated rates are performed in parallel, and then the activation probabilities are computed using the softmax function. The 1 × 1 convolution is the bottleneck for parameter reduction until the final shortcut is connected. With the help of inflated convolution, SAM can capture global feature relations while focusing on local semantic information.

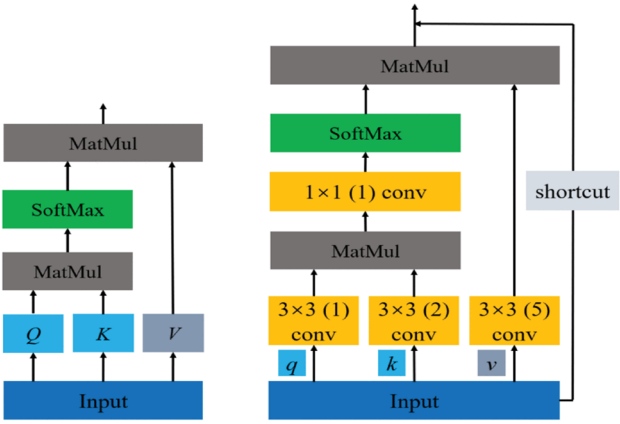

Fig. 3. Architectural Overview of the Self-Attention Module (SAM).

3 Experimental Results and Analysis

3.1 Experimental Data Collection

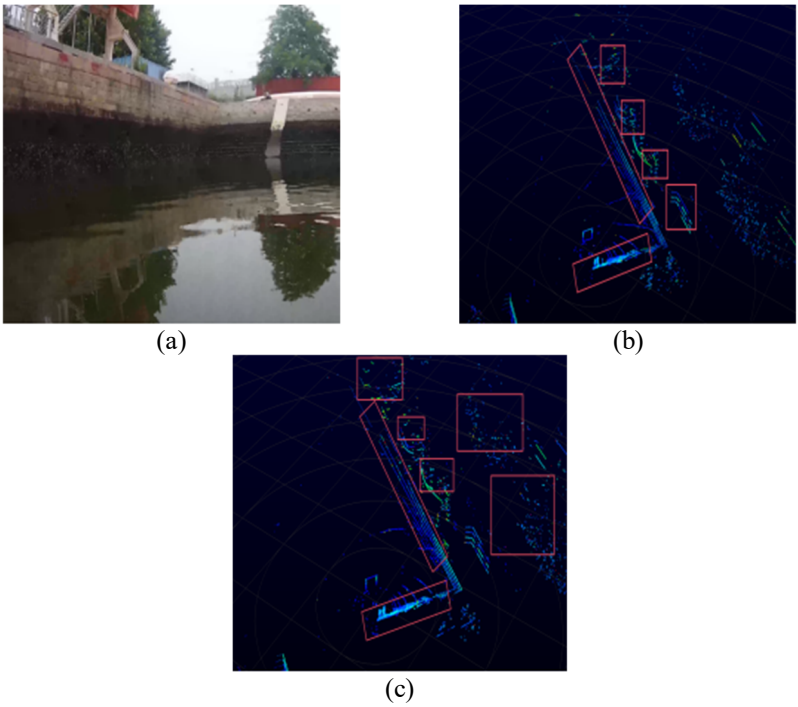

Fig. 4. (a) is the real picture, (b) is the YOLOv3 output result, (c) is the Poly-YOLO output result.

Figure 4 shows the results of the real scene pictures mapped by LIDAR scanning to form a 3D point cloud and deepened by densification after being fed back by YOLOv3 and Poly-YOLO, respectively. Poly-YOLO clearly displays each target in the scene, and the convolution layer of the detection module extracts the feature information at different scales to accurately detect the target, especially our selected scene pictures in the typical frames are clearly represented, such as the point cloud of the shoreline and the target ship. The traditional target detector YOLOv3 usually outputs a rectangular bounding box to describe the detected targets. Poly-YOLO, on the other hand, can more accurately describe the shape of the target by introducing polygonal output, which is especially suitable for irregularly shaped targets, such as ships, buildings, and so on.

(1) Improved clarity and accuracy of target detection: Poly-YOLO clearly shows individual targets in the sea scene in the output, including point clouds of buildings near the shoreline and target ships. This shows that Poly-YOLO's detection module is able to extract feature information at different scales more accurately and recognize target objects more precisely.

(2) Bounding box retention capability: Poly-YOLO has bounding polygons, which help retain the accuracy of bounding box detection. This means that Poly-YOLO is able to capture the shape and boundaries of the target object more accurately, thus improving the quality of target detection.

(3) Effectiveness of SAM: The introduction of SAM enables Poly-YOLO to better process contextual information and learn more discriminative features, which further improves the performance. This shows that Poly-YOLO has better adaptability and generalization ability to better adapt to different scenarios and environments.

4 Conclusion

In this paper, we propose an improved Poly-YOLO detector method based on YOLOv3. In order to enhance the detection rate and efficiency, we introduce the expansion convolution and self-attention module in the Poly-YOLO structure; considering that the sparse depth map obtained from the projection of the 3D point cloud scanned by LIDAR to the 2D pixel plane contains insufficient information, we utilize the depth upsampling method to generate the dense depth map. Finally, the results are output through the detection head. The experimental results show that compared with YOLOv3, the Poly-YOLO proposed in this paper has significant advantages in target detection during berthing, including high accuracy, faster speed, and better boundary preservation capability. In the future, we can further extend our research in the following directions, e.g., by utilising a lighter weight detector, YOLO-lite, and secondly, we can further exploit the performance of other deep learning models to obtain more comprehensive experimental results.

Acknowledgments

Basic Research Projects of Liaoning Provincial Department of Education in 2023(JYTMS20230172)

References

1. Qi C R, Su H, Mo K, et al. 2017 PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. IEEE, DOI:10.1109/CVPR.2017.16.
2. Yin T, Zhou X, Krhenbühl, Philipp 2020 Center-based 3D Object Detection and Tracking. IEEE, DOI:10.48550/arXiv.2006.11275.
3. Yang Z, Sun Y, Liu S, et al. 2020 3DSSD: Point-based 3D Single Stage Object Detector. IEEE, DOI:10.1109/CVPR42600.2020.01105.
4. Zhou Y, Tuzel O 2017 VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. IEEE, DOI:10.48550/arXiv.1711.06396.
5. Deng J, Shi S, Li P, et al. 2021 Voxel R-CNN: Towards High Performance Voxel-based 3D Object Detection. National Conference on Artificial Intelligence. AAAI, DOI: https://doi.org/10.1609/aaai.v35i2.16207.
6. Lang A H, Vora S, Caesar H, et al. 2019 Fast Encoders for Object Detection from Point Clouds. IEEE, DOI:10.1109/CVPR.2019.01298.
7. Chen X, Ma H, Wan J, et al. 2017 Multi-View 3D Object Detection Network for Autonomous Driving. IEEE, DOI:10.1109/CVPR.2017.691.
8. Barrera A, Guindel C, Jorge Beltrán, et al. 2020 BirdNet+: End-to-End 3D Object Detection in LiDAR Bird's Eye View. IEEE, DOI:10.1109/ITSC45102.2020.9294293.
9. Ma Y, Wang T, Bai X, et al. 2022 Vision-Centric BEV Perception: A Survey. arXiv e-prints, DOI:10.48550/arXiv.2208.02797.
10. Simon M, Milz S, Amende K, et al. 2018 Complex-YOLO: An Euler-Region-Proposal for Real-Time 3D Object Detection on Point Clouds. Springer, Cham, ECCV, DOI: 10.1007/978-3-030-11009-3_11.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.