USING EDGECONV TO IMPROVE 3D OBJECT DETECTION FROM RGB-D DATA

Weisheng Lin [1], Yiping Chen [1, *], Cheng Wang [1], Jonathan Li [1, 2]

[1] Fujian Key Laboratory of Sensing and Computing, School of Informatics, Xiamen University, 422 Siming Road South, Xiamen 361005, China; (23320171153124@stu.xmu.edu.cn, chenyiping@xmu.edu.cn, cwang@xmu.edu.cn, junli@xmu.edu.cn)
[2] Mobile Mapping Lab, Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada;

KEY WORDS: 3D Object Detection, Point Clouds, RGB-D, Frustum, Graph Convolution, Deep Learning

ABSTRACT:

In this paper, we proposed a novel 3D deep learning model for object localization and object bounding boxes estimation. To increase the detection efficiency of small objects in the large scale scenes, the local neighbourhood geometric structure information of objects has been taken into the Edgeconv model, which can operate the original point clouds. We evaluated the 3D bounding box with high resolution in the RGB-D dataset and acquired stable effectiveness even under the sparse points and the strong occlusion. The experimental results indicate that our method achieved the higher mean average precision and better IOU of bounding boxes in SUN RGB-D dataset and KITTI benchmark.

1. INTRODUCTION

Recently, great achievements have been made in 2D object detection and instance segmentation. Compare the 2D image understanding task, the 3D understanding was more challenge and have signification on the fields such as autonomous driving and augmented reality. With the popularization of various 3D sensors, how to represent objects in space is an open question. We focus on the 3D point clouds captured from RGB-D and using deep neural networks for 3D object detection and estimating the oriented 3D bounding boxes.

The most common 3D data representation formats are point clouds, voxel and mesh. Point clouds record the position of the object surface and other information which can be augmented such as multispectral and colours, etc. 3D voxel and Mesh data can be obtained by transformation of point clouds through various of algorithms. Previous many works proposed that the projection (H. Su et al. 2015 and C. R. Qi et al. 2016) from point clouds to image and then leverage convolutional networks to detect objects. The invariance and patterns of 3D point clouds are often obscure in those algorithms.

PointNet (C. R. Qi et al. 2017) is a major turning point which presents a deep neural network that takes raw point clouds as input and get a good effect. The operation successfully worked on unorder point sets by deep learning encourage many researchers to turn their work to deal with the raw 3D data. Since then, the field of directly consume irregular point clouds become active and deep learning approaches have been developed. Thanks to the availability of large-scale dataset and available GPU computing resources, the researcher can target more complex and tricky problem when detecting objects of hundreds of categories from point clouds. Many researchers have started toward effort concentrating multisource from different sensors for 3D shape representation. The main challenge is how to combine two kinds of input and learned better 3D shape representation robustly and to ensure the highly accurate of objects detection.

In contrast to previous work that treats RGB-D data as 2D maps for CNNs, the F-PointNet (Qi et al. 2018) is more 3D-centric as lift depth maps to 3D point clouds and process them in 3D space. In order to reduce the search space, F-PointNet extract the frustum of an object by extruding 2D bounding boxes from mature image detection primarily. While the architecture deeply limits its performance by 2D object detection for 2D bounding box and 3D instance segmentation for the 3D bounding box. PointNet lacked the local structure limits its ability to recognize fine-grained patterns and generalizability to complex scenes. We can find that we missed some object due to the 3D instance segmentation improperly. Therefore, we apply the graph Convolution to enforce the effect of instance segmentation for 3D object detection. Our main contribution is proposing a method for instance segmentation which considers the local neighbourhood geometric information and use it to 3D object detection for solving multiple instances overlap.

2. RELATED WORK

2.1 3D Object Detection from RGB-D Data

Various methods have been proposed for 3D object detection from RGB-D data. We review the related methods and discuss in the following two categories.

2D based methods: (B. Li et al. 2016) project LIDAR point clouds to the front view, which is used as an input to a fully convolutional neural network to directly generate dense 3D bounding boxes. (X. Chen et al. 2016) takes monocular RGB images and shape priors or occlusion patterns to infer 3D bounding boxes. MV3D (X. Chen et al. 2017) extends the image based region proposal network (S. Ren et al. 2015) to 3D by corresponding every pixel in the bird’s eye view feature map

[*] Corresponding author

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-2-W13-835-2019 | © Authors 2019. CC BY 4.0 License.

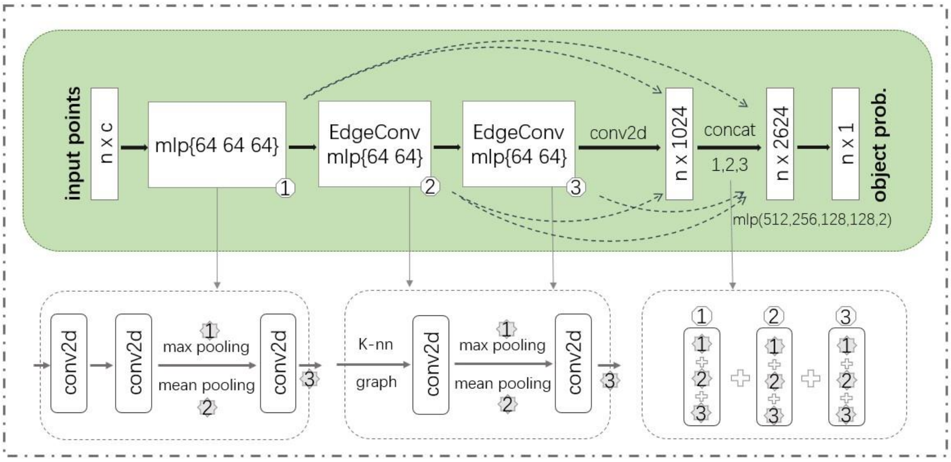

Figure 1. **3D instance segmentation for object detection.**

to multiple prior 3D anchors. However, the method drops behind in detecting small objects and cannot easily adapt to scenes with multiple objects in a vertical direction.

3D based methods: (S. Song et al. 2014) learns to classify 3D bounding box proposals generated by a 3D sliding window using synthetically-generated 3D features. (M. Engelcke et al. 2017) extends (D. Z. Wang et al. 2015) by replacing SVM with 3D CNN on voxelized 3D grids. (G. Riegler et al. 2016) is using 3D convolution for object detection by representing 3D data. Computation cost for those methods is usually quite high due to the expensive cost of 3D convolutions and large 3D search space. Recently, (C. R. Qi et al. 2017) proposes a flexible and effective solution with deep 3D feature learning (PointNet) predicts an (oriented and amodal) 3D bounding box for the object from the points in the frustum.

2.2 Deep Learning on Point Clouds

A lot of work has been done on photometric data by deep learning methods. However, the vast successful deep learning architecture are ineffective for 3D point clouds because of the unstructured and irregular data. In order to enable convolutional network plays an important role in 3D data, most researchers resort the 3D data to regular represent data such as 3D voxel grid before feeding the networks at the beginning. The (Zhou. Y et al. 2017) has successfully applied the convolution network on voxelization data. The author proposed the representation that enables a geometric 3D shape as a probability distribution of binary variables works on a 3D voxel grid, using a Convolutional Deep Belief Network. However, due to the 3D voxel have some disadvantage like unnecessarily large representation and loss spatial information, various researcher transfers their work on directly using deep architectures to unstructured point set. PointNet is a successful model that takes raw point as input and get a good effect, but it loss the local structure for more complex scene. (Yu Wang et al. 2018) proposed graph convolution operation which captures local geometric structure while maintaining permutation invariance. We improved the 3D architecture by reducing the layers of the Edgeconv and eliminating the spatial transform to identify fine-grained point clouds of the Frustum, which is got a good effect.

3. 3D DETECTION WITH FRUSTUM EDGECONV

Our method mainly works on the 3D Instance segmentation parts of F-PointNet. Since occlusion often appear in dense areas, 3D deep learning for instance segmentation is required to distinguish different objects. We improve the performance of instance segmentation and it more robust to detect small objects in the large-scale scenes.

3.1 Frustum Proposals

Similar to F-PointNet method, we build the candidate region of the target which can quickly be found through 2D target detection firstly. Then each candidate region is transformed into frustum by a given camera projection matrix. We normalize the frustums by rotating them toward to the center view due to frustum may have different directions. For each object, we use the ground truth 2D bounding box to extract correspond frustum for training our neural networks. We chose a certain number of point clouds for each frustum ($n \times c$ with n points and c channels of XYZ, intensity, etc.) and randomly sample to certain number when higher a certain number.

3.2 Object Detection

3D instance segmentation predicts a probability score for each point cloud that indicates how likely the point belongs to the object of interest. After getting the one instance from each frustum, we transform the point clouds to local coordinate to boost the translation invariance of the algorithm. Note that objects may be obscured when object partially overlap and how to distinguish each point belonged to a different instance. It is a difficult problem and a key challenge in this case. We normalized its coordinate after obtaining the object instance

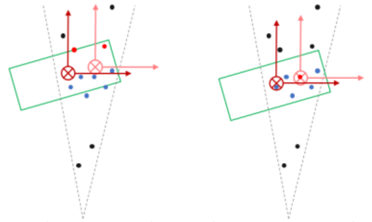
(a) False segmentation (b) True segmentation

Figure 2. **Influence of instance segmentation.** Artificial points are shown to illustrate, the red point is false segmentation, blue point is true segmentation point. (a) False segmentation and predict 3D bounding box. (b) True segmentation and predict 3D bounding box.

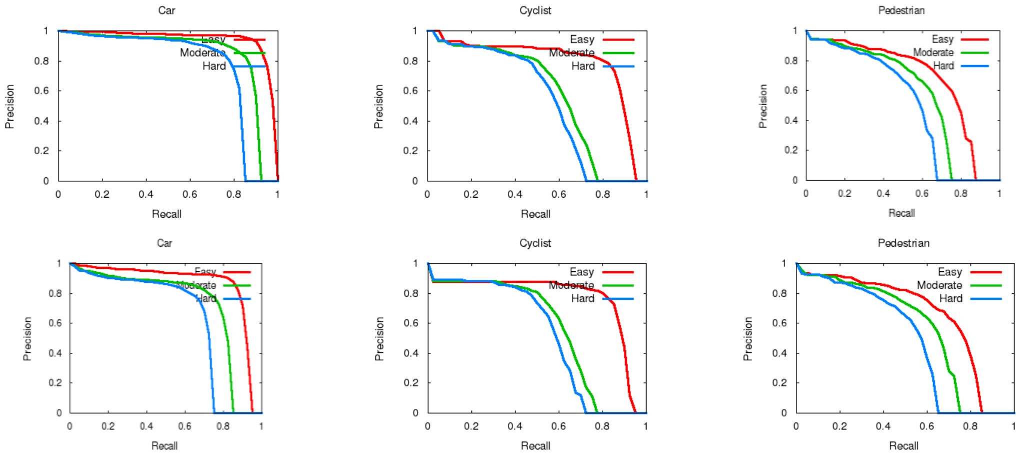

Figure 3. **Precision recall (PR) curves for 3D object detection on KITTI *val* set.**

point clouds and separated different terms to optimize the 3D bounding box parameterization. Subsequently, the 3D mask coordinate has been influenced by the instance segmentation and further effect predicted the true center and angle of the entire object. Fig. 2 shows one example of the segmentation result influences on the final 3D bounding box estimation illustrated.

In order to decrease the influence of instance segmentation for learning the true center of the object, we use our neural network to improve the 3D object detection performance. We choose a certain amount of point of each point by KNN and use EdgeConv to learn neighbouring edge features. Consider the speed and runtime, we adjust our neural network for the different dataset. Specifically, we choose the 20 neighbour points for each set when learning feature from KITTI (SUN-RGBD) dataset. Figure 1 shows the architecture of instance segmentation for getting the point clouds of the object. We remove the one-hot when learning a feature of the object for instance segmentation. For amodal 3D box estimation, we use PointNet to regress its center position, size and heading angle.

4. EXPERIMENTS

In the experiment, we trained our network on an NVIDIA GeForce GTX TITAN Z using ADAM optimizer with an initial learning rate 0.001, the batch size of 16. First, we compare with state-of-art methods for 3D detection on SUN-RGBD (S. Song et al. 2015) and KITTI (A. Geiger et al. 2012). Second, we show some results of our method and have deep discussion of its strengths.

4.1 Evaluation on KITTI

KITTI The object detection benchmark in KITTI contains both 2D and 3D annotations of vehicles, pedestrians, and cyclists in urban driving scenarios by the wide-angle camera and Velodyne HDL-64E LiDAR. The training set contains 7,481 frames and an undisclosed test set contains 7,581 frames. In our own experiments (except those for test sets), we follow (X. Chen et al 2016) to split the official training set to a train set of 3,717 frames and a validation set of 3769 frames such that frames in train/valid sets belong to different video clips. We choose the ground truth 2D bounding box to get the frustum point clouds and evaluate different methods. We report model performance on the validation set for all three object categories.

Benchmark	Easy	Moderate	Hard
Pedestrian(v2)	70.00	61.32	53.59
Cyclist(v2)	77.15	56.49	53.37
Pedestrian(v1)	65.05	55.69	49.10
Cyclist(v1)	75.74	56.50	52.77
Pedestrian(our)	67.84	59.10	51.60
Cyclist(our)	**78.17**	**57.21**	**53.53**

Table 1. Performance on KITTI *val* set for pedestrians and cyclists.

Tab. 1 shows the performance of our method on the KITTI *val* set. Our method gets the higher AP than F-PointNet(v1) on Pedestrain and Cyclist. And achieve the highest AP compare with F-PointNet(v1) and F-PointNet(v2). The point clouds of cyclist are sparser but our method get the best results. Therefore, our method is robust for the small object.

Method	Easy	Moderate	Hard
Mono3D	2.53	2.31	2.31
3DOP	6.55	5.07	4.10
VeloFCN	15.20	13.66	15.98
MV3D	71.29	62.68	56.56
F-PointNet(v1)	83.26	69.28	62.56
F-PointNet(v2)	83.76	70.92	63.65
our	**84.64**	**71.15**	**63.68**

Table 2. 3D object detection AP on KITTI *val* set (cars only)

Tab. 2 shows the different methods performance on KITTI *val* set. our method has the best performance of different level 3D detection task for the car. What's more, our parameter quantity has lower than F-PointNet(v2).

Fig. 3 reports the 3D AP curves of our method and F-PointNet on KITTI *val* set. the top is our method and bottom are F-PointNet. We can see the tasks with different levels of difficulty for three class. For the car we get the best performance and have high recall than F-PointNet.

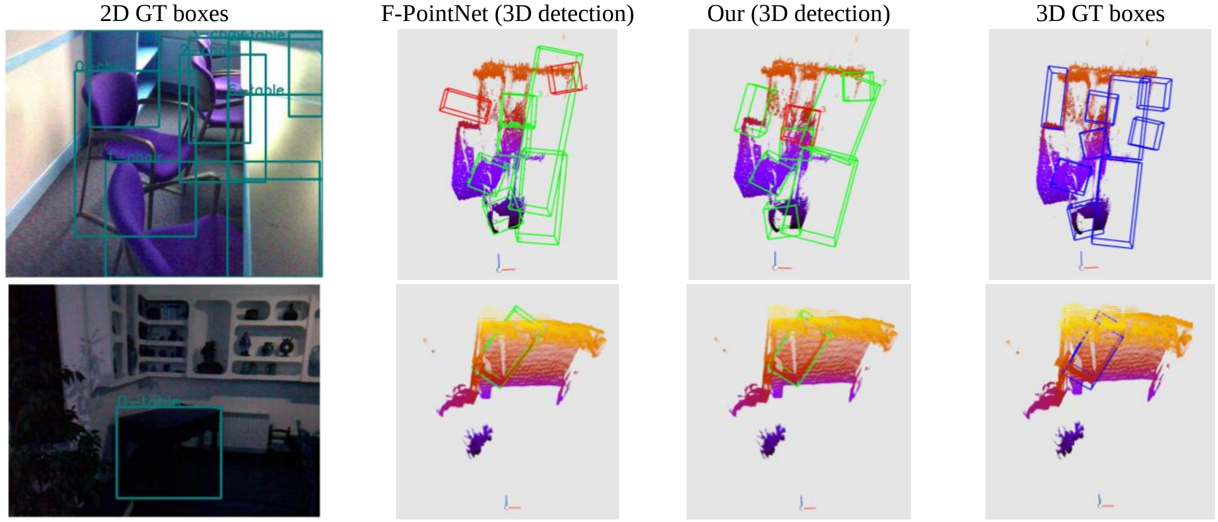

Figure 4. **Visualization of results on SUN-RGBD val set.**

4.2 Evaluation on SUN-RGBD

SUN-RGBD The data set consists of 10,355 RGB-D images captured from various depth sensors for indoor scenes (bedrooms, dining rooms, etc.). The training and testing sets contain 5285 and 5050 images, respectively. We follow the same train/val splits as (S. Song et al. 2015) for experiments. As to strong occlusion and tight arrangement of the objects in indoor scenes, it is more challenge for detection. We get several results for object detection. Due to F-PointNet without release the there own 2D detection and the result also deep rely on the 2D box bounding to get the proposed region. We choose the ground truth of 2D bounding box to get the frustum point clouds and evaluate different methods on it. In this task, we show the performance of our architecture for small object detection and strong occlusion of object detection.

Fig. 4 visualizes the results on SUN-RGBD val set. *First column:* 2D ground true bounding boxes. *Second column:* the predicted 3D bounding boxes results from F-PointNet. *Third column:* our method detection results and predicted 3D bounding boxes. *Fourth column:* 3D ground true bounding boxes. Green boxes are true positive and red boxes are false positives. Compared to F-PointNet, we can find that our algorithm has the success detection of the bookshelf and has higher accuracy segmentation ability for the table (successful separate two close tables). The second example shows our method have more fine-grained segmentation results (our method has a more accurate box size that F-PointNet).

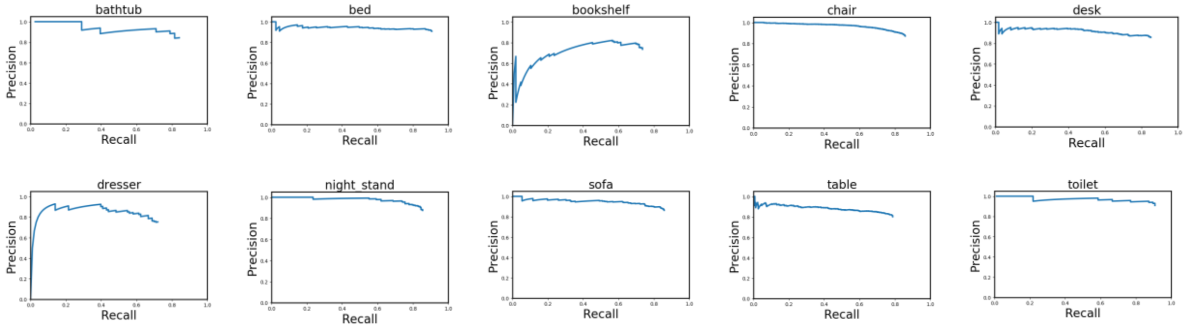

Figure 5. **Precision recall (PR) curves for 3D object detection on SUN-RGBD val set.**

	bathtub	bed	bookshelf	chair	desk	dresser	Night stand	sofa	table	toilet	map
F-PointNet	82.1	88.5	51.1	78.3	72.9	66.2	80.9	79.0	62.0	85.3	74.6
our	79.9	86.1	60.0	83.2	78.6	64.0	83.4	81.8	70.0	88.6	77.6

Table 3. 3D object detection AP on SUN-RGBD val set.

Tab. 3 reports the different category 3D average precision and mAP. Compared with state-of-the-arts F-PointNet our method is 3.0% to better in mAP.

Fig. 5 shows the PR curves of our method on SUN-RGBD val set. Our method has high recall for different class.

5. CONCLUSIONS

We have presented a remarkably accurate 3D instance segmentation method which has better IOU of 3D bounding box estimation. Even at strong occlusion, we also get better result than previous methods on the two datasets we used. Our method has lower complexity and gets a good result. To some extent, we have solved the segmentation result of output blending when multiple instances occurred. Experiments on the SUN-RGBD dataset shows the high accuracy on the 3D localization and category classification tasks. To a certain degree, we improve the 3D object detection even when have strong overlapping.

ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China under Grants 61601392 and 41871380.

REFERENCES

A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving the kitti vision benchmark suite. In Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

B. Li, T. Zhang, and T. Xia, “Vehicle detection from 3d lidar using fully convolutional network,” in Proceedings of Robotics: Science and Systems, AnnArbor, Michigan, June 2016.

C. R. Qi, H. Su, M. Nie ßner, A. Dai, M. Yan, and L. Guibas. Volumetric and multi-view cnns for object classification on 3d data. In Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.

C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, 2017.

D. Z. Wang and I. Posner. Voting for voting in online point cloud object detection. Proceedings of the Robotics: Science and Systems, Rome, Italy, 1317, 2015.

G. Riegler, A. O. Ulusoys, and A. Geiger. Octnet: Learning deep 3d representations at high resolutions. arXiv preprint arXiv:1611.05009, 2016.

H. Su, S. Maji, E. Kalogerakis, and E. G. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In Proc. ICCV, 2015.

M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In Robotics and Automation (ICRA), 2017 IEEE International Conference on, pages 1355–1361. IEEE, 2017.

Qi, Charles R and Liu, Wei and Wu, Chenxia and Su, Hao and Guibas, Leonidas J. Frustum PointNets for 3D Object Detection from RGB-D Data. Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, 2018.

S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.

S. Song and J. Xiao. Sliding shapes for 3d object detection in depth images. In Computer Vision–ECCV 2014, pages 634–651. Springer, 2014.

S. Song, S. P. Lichtenberg, and J. Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 567–576, 2015.

X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun. Monocular 3d object detection for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2147–2156, 2016.

X. Chen, H. Ma, J. Wan, B. Li, and T. Xia. Multi-view 3d object detection network for autonomous driving. In IEEE CVPR, 2017.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. 2018. Dynamic Graph CNN for Learning on Point Clouds. arXiv preprint arXiv:1801.07829 (2018).

Zhou Y, Tuzel O. Voxelnet: End-to-end learning for point cloud based 3d object detection[J]. arXiv preprint arXiv:1711.06396, 2017.

https://doi.org/10.5194/isprs-archives-XLII-2-W13-835-2019