An Optimization of Deep Sensor Fusion Based on Generalized Intersection over Union

Lianxiao Meng[1,2], Lin Yang[2], Gaigai Tang[1,2], Shuangyin Ren[2(✉)], and Wu Yang[1]

[1] Information Security Research Center of Harbin Engineering University, Harbin, China
[2] National Key Laboratory of Science and Technology on Information System Security, Institute of System Engineering, Chinese Academy of Military Science, Beijing, China
renshuangyin@126.com

Abstract. 3D object detection is a major topic in unmanned driving and robotics, which is suffering from the low accuracy recently. We found that the loss function of 3D object detection network is the main cause to the low accuracy. For this, we proposed an optimized realization of deep sensor fusion network model (DSFN) based on Generalized Intersection over Union (GIoU). In DSFN, the designed backbone network is used to fuse point cloud features and image features, making full use of heterogeneous sensor information. Specifically, we introduced the GIoU as the loss function of the backbone network. We evaluated our model on KITTI dataset which is resulted from a LIDAR-camera setup. Compared with similar models, our model shows a higher accuracy.

Keywords: Deep Sensor Fusion Network (DSFN) · 3D bounding box estimation · Loss function · Deep learning · Camera and Lidar

1 Introduction

In recent years, artificial intelligence (AI) has developed rapidly, where the unmanned driving (UD) has been paid more attention by major enterprises, scholars and even general public. There are two quite different ways to achieve the goal of UD: one is a progressive method adopted by traditional enterprises, starting from the existing assisted driving system, and gradually increasing automatic steering to actively prevent collisions and other functions, to achieve conditional UD, and finally to complete UD when the costs and related technologies reach certain requirements. The other is represented by high-tech IT enterprises, they choose "one step" way to directly reach the ultimate goal of driverless driving. But the technical route chosen by the latter is more challenging and risky.

This research is supported by the National Natural Science Foundation of China (Grant No.61931017 and No.61831007).

M. Qiu (Ed.): ICA3PP 2020, LNCS 12453, pp. 552–562, 2020.
https://doi.org/10.1007/978-3-030-60239-0_37

Therefore, innovative algorithms and efficient and robust systems are needed to support it. Under this demand, object detection and positioning is particularly important, because it is equivalent to the intelligent unmanned system can "see" all directions, and provides a lot of useful information for the decision-making planning of UD system.

Nowadays, the latest development of convolutional neural networks has enabled 2D detection in complex environments. However, there is an urgent demand for precise detection in 3D environment, leading to an open challenge of 3D object detection. Thus, in this context, We propose an optimized realization of DSFN by using $GIoU$ (see Fig. 1) based on PointFusion [1], that can give a $6-DoF$ pose and the 3D bounding box dimensions by combining point cloud and RGB information, along with identifying objects of interest in the scene.

Bounding box regression is one of the most basic components in many 2D/3D computer vision tasks. The tasks of target location, multi-object detection, target tracking and instance level segmentation all depend on the associated bounding box regression. The main trend of using deep neural network to improve application performance is to propose a better framework backbone [2] or a better strategy to extract reliable local characteristics [3].

Main contribution of our model as follows:

- The model we based on is validated on the KITTI 3D object detection dataset [4] with two indicators, namely, object classification accuracy and 3D bounding box accuracy. From many research works, we found that the object classification task had achieved a good enough performance, so we paid more attention on 3D bounding box accuracy while ensuring the object classification accuracy. For achieving a better 3D bounding box regression, we optimized the loss function by using IoU and $GIoU$ [5] in backbone network.

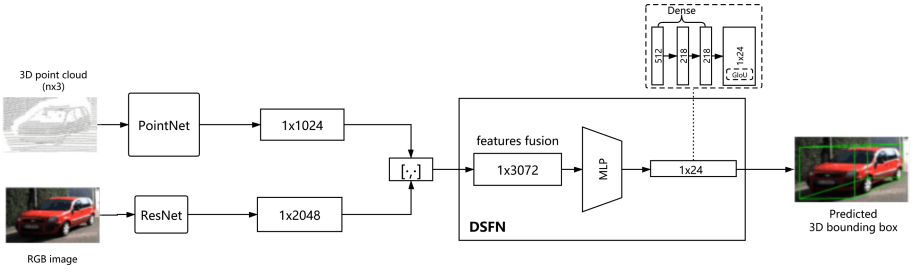

Fig. 1. An overview of the dense DSFN architecture. DSFN has two feature extractors: a PointNet variant that processes raw point cloud data, and a CNN (ResNet) that extracts visual features from an input image. Our fusion backbone network that directly regresses the box corner locations.

The rest of this article is as follows. The Sect. 2 introduces some achievements and related work. After discussing the DSFN model in Sect. 3, the specifications

about the datasets with the implementation and experimentation details are discussed in Sect. 4. The obtained performance results are also analyzed. Finally, we make a summary and set further goals of future work in Sect. 5.

2 Related Work

2.1 Object Detection Accuracy Measures

Intersection over Union (IoU) is a standard for measuring the accuracy of corresponding objects in a specific data set. It can be used to measure any task that gets a prediction range in the output. In lots of task detection and 2D/3D bounding box projects [6,7], IoU is the most commonly used to determine true positives and false positives in a set of prediction, which will be given a decision threshold when be selected. Similarly, our experiment also uses IoU as a measure of the performance of our 3D bounding box, and the threshold is set to 0.5. Actually, due to not sensitive to the scales of the target object, IoU, as the core evaluation index, can be used as the direct representation of the regression loss of bounding box. However, there are few experiments that use IoU as loss function directly, so we have verified the effect of idea that IoU as loss function in our experiments.

2.2 Bounding Box Representations and Losses

In object detection, learning bounding box parameters is crucial. Various kinds of bounding box representations and losses are proposed in the literatures. Redmon et al. in YOLO v1 [8] propose a direct regression on the bounding box parameters with a small tweak to predict square root of the bounding box size to remedy scale sensitivity. Girshick et al. [9] in R-CNN parameterize the bounding box representation by predicting location and size offsets from a prior bounding box calculated using a selective search algorithm [10]. Most popular object detectors [11–13] utilize some combination of the bounding box representations and losses mentioned above. These considerable efforts have yielded significant improvement in object detection. As their loss of bounding box regression is not a direct representation of the core evaluation indicators, there may be opportunities to further improve localization.

In 2019, based on IoU, [5] introduced $GIoU$ as a new loss and new measure to make up for the lack of IoU as the loss function, and verified its advantages in 2D environment. However, as far as we know, the performance of this new concept in 3D environment has not been explored much yet, so we made a attempt in our experiment and has gotten the optimized effect just like it in 2D experiment.

3 Deep Sensor Fusion Network Model

DSFN is expected to use image features extracted by a standard CNN, ResNet 50, and corresponding point cloud features generated by changed PointNet subnetworks as inputs to combine these functions and output a 3D boundary frames of the target objects. The backbone network of DSFN is showed as follows:

- A changed PointNet network to extract point cloud features.
- Resnet50 to extract image appearance features.
- The fusion network takes the fusion information of the two mentioned above as input, and then outputs 3D boundary prediction.

3.1 Fusion Subcomponent

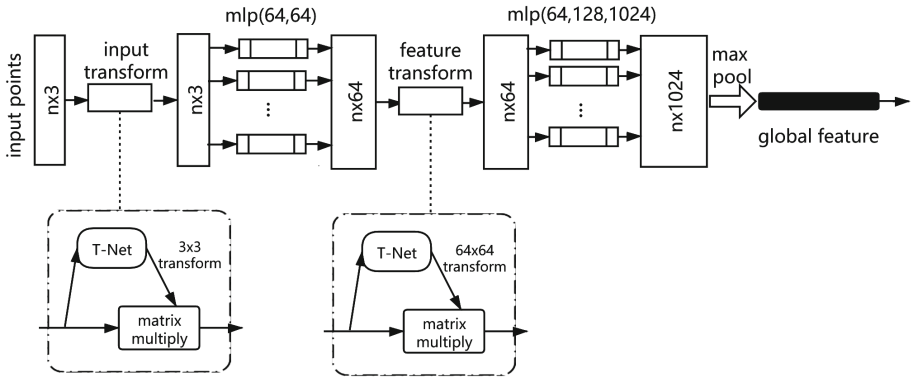

Fig. 2. Pointnet architecture: PointNet takes the lead in using the symmetric function (max pooling) to realize the permutation invariance in the processing of unordered 3D point sets. The modified model inherits this point, and then in order to improve the estimation performance of 3D bounding box, the original batch normalization layer is removed.

For the part of processing point cloud, we used a changed PointNet architecture, which was revised on the basis of Qi et al.'s model [14]. PointNet takes the lead in using the symmetric function (max pooling) to realize the permutation invariance in the processing of unordered 3D point sets. The modified model inherits this point, and then in order to improve the estimation performance of 3D bounding box, the original batch normalization layer is removed. But theme architecture has no fundamental change (see Fig. 2). The model takes the raw point cloud and learns a spatial coding of each point and the aggregated global point cloud features. Then these features are used for classification and semantic segmentation. PointNet has many ideal properties: it directly processes the original points without the lossy operation like voxelization or projection, and it is linearly proportional to the number of input points.

Resnet50 [15] (see Fig. 3) is used in the part of image feature extraction. The traditional convolution network or the all connected network have some problems, such as information loss, when information is transmitted. At the same time, it will cause the gradient to disappear or the gradient to explode, which makes the deep network unable to be trained. Resnet50 solves this problem to a certain extent by bypassing the input information directly to the output to

protect the integrity of the information. The whole network only needs to learn the difference between the input and output to simplify the learning objectives and difficulties.

Our fusion model takes raw point cloud and image features preprocessed by the above two methods as input (we try the painless fusion function based on the previous experience, and find that the series of two feature vectors can obtain better performance), and then directly outputs the 3D positions of the eight corners of the target bounding box. The fusion network is implemented with three dense layers, where the dense network does well in regression problem.

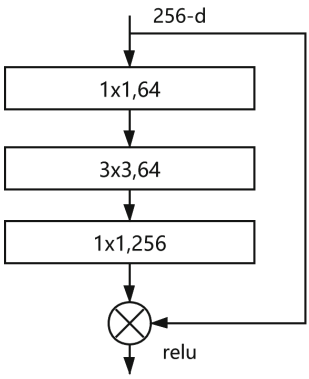

Fig. 3. ResNet50 architecture: Resnet50 solves some problems that will be caused by traditional convolution network to a certain extent by bypassing the input information directly to the output to protect the integrity of the information.

3.2 Loss Function

SmoothL1 Loss The loss function in the global fusion model we refer to is:

$$L = \sum_{i} smoothL1(X_i^*, X_i) + L_{stn} \tag{1}$$

where X_i^* are the ground-truth box corners, X_i are the predicted corner locations and L_{stn} is the spatial transformation regularization loss introduced in [14] to enforce the orthogonality of the learned spatial transform matrix. A major drawback of the global fusion network is that the variance of the regression target X_i is directly dependent on the particular scenario. For autonomous driving, the system may be expected to detect objects from 1 m to over 100 m. This variance places a burden on the network and results in suboptimal performance.

In order to solve this problem, we make two adjustments to the loss function in the model, namely, L_{IoU} and L_{GIoU}.

IoU Loss IoU is intersection over union, and it is the most commonly used index in object detection. In the anchor-based method, its function is not only to determine the positive and negative samples, but also to evaluate the deviation between the output box and the ground truth.

$$L_{IoU} = 1 - IoU = 1 - \frac{I(X)}{U(X)} \tag{2}$$

where $I(X)$ represents the intersection of the ground truth of target object and the 3D bounding box, and $U(X)$ is the union. As a loss function, IoU can directly reflect the detection effect of predicted box and ground truth. In addition, another good feature is scale invariance, that is, it is not sensitive to scale. In the region task, it can meet the nonnegative, identity, symmetry and triangle inequality of prediction results.

With the IoU loss is applied into the fusion model, it results a better regression effect than $SmoothL1$ loss, but also shows some defects. One is that if two boxes with no overlap, according to the definition, $IoU = 0$, it cannot reflect the deviation between them (degree of coincidence). At the same time, when $IoU = 0$, the loss $= 1$, there is no gradient return, which stops the model from training.

GIoU (Generalized Intersection over Union) Loss In CVPR2019, [5] proposed the idea of $GIoU$. IoU is a concept of ratio, and it is not sensitive to the scale of the target. But it shows an obvious defect that doesn't take the situation that two boxes without overlap into consideration. $GIoU$ can effective counters this situation through a more precise definition of the deviation between two boxes.

$$L_{GIoU} = 1 - GIoU = 1 - (IoU - \frac{A_c - U(X)}{A_c}) \tag{3}$$

Similar to IoU, $GIoU$ is also a deviation measure. As a loss function, it meets the basic requirements of loss function: $GIoU$ is not sensitive to scale. $GIoU$ is the lower bound of IoU. For surrounding any group of ground truth and 3D bounding box, where A_c is the volume of the smallest box. In the case of infinite coincidence of two frames, $IoU = GIoU$. The value of IoU is within [0,1], but the value of $GIoU$ has symmetric interval, and the value range is within $[-1,1]$. The maximum value is 1 when the two are coincident, and the minimum value is -1 when the two are not intersected and infinite, so $GIoU$ is a very good deviation measure. Different from IoU only focusing on overlapping areas, $GIoU$ not only focuses on overlapping areas, but also other non overlapping areas, which can better reflect the degree of coincidence between the two boxes.

4 Experiments

4.1 Dataset

KITTI The KITTI dataset [4] contains 2D and 3D labels of cars, pedestrians and cyclists in urban driving scenarios. The sensor configuration includes a wide-angle camera and velodyne $hdl{-}64e$ lidar. The official training collection contains

7481 images. In order to ensure the validity and credibility of the experimental results comparison, we follow the dataset processing in the comparison model, and divide the official training dataset into training set, development set and verification set. The size of each set is also consistent (see Table 1).

Table 1. Train-Dev-Test Split

	Train data	Dev.data	Test data
No.of examples	6750	365	366

4.2 Pre-processing

The velodyne setup on the station wagon is used to produce the points clouds. For details, it's a rotating 3D laser scanner that generates data points at a rate of 10 HZ, 64 beams, with 0.09° angular resolution, 2 cm distance accuracy, collecting 1.3 million point/second, with a horizontal and vertical field of view of 360° and 26.8 respectively. There are lots of points so that we need to trim down the input size for correspondence, feasibility and relevance. As a result, we filter the point clouds falling in the camera view angle and randomly sample 2048 points from them. Then, the points are fed through a Spatial Transformation Network in order to canonicalize the input space. Further, the ground truth labels are transformed to the velodyne coordinate for tractability in prediction.

4.3 Network Specifications

This subsection generally lists the network specifications resulted from extensive experimentations. The model we used has 1,808,207 trainable parameters, where most of them belong to the PointNet [14] architecture. Moreover, in the fusion network, we finally choose a simple and effective architecture which is consisted of 3 hidden layers having 512, 128 and 128 units, respectively, the fusion layer gives the box-corner locations as output.

4.4 Results

First of all, we completely restored Global Fusion network which is the component of Pointfusion [1]. SmoothL1 as the loss function of Global Fusion network has been introduced in 3.2. the model was trained to give the loss curves, presented in the figure below (see Fig. 4).

The output of Global Fusion has two aspects: classification and 3D bounding box regression. In the process of recurrence experiment, we found that the classification accuracy reached 96.17%, which was hardly to be improved qualitatively, leading to the truth that 96.17% shows a good enough performance indeed. However, there is a lot of room for improvement in the performance

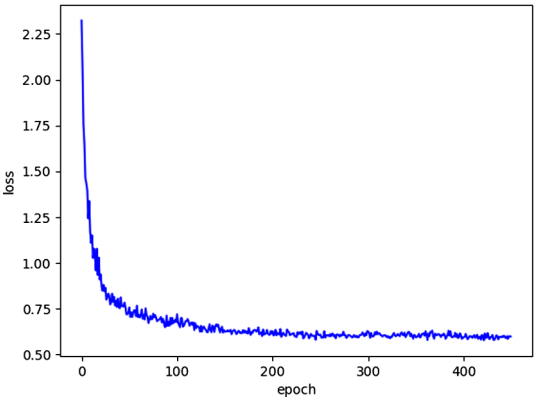

Fig. 4. Box output loss vs Epochs for training and development with SmoothL1

of 3D bounding box regression. So in our experiment, we focused on the 3D bounding box regression on the premise of ensuring classification accuracy.

IoU is the core index to measure 3D bounding box regression. The loss curves of applying *IoU* as loss function instead of SmoothL1 in DSFN is shown as follows (see Fig. 5):

GIoU is a new concept based on *IoU* in 2019. An example has be given in [5] that the performance in 2D environment is indeed better than *IoU*. According to the research, *GIoU* has not been widely used at present, especially in 3D bounding box regression. The performance of the DSFN using *GIoU* as loss function is shown as follows. (see Fig. 5):

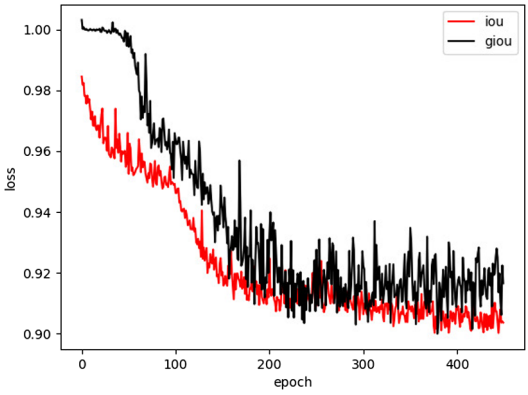

Fig. 5. Box output loss vs Epochs for training and development with L_{IoU} and L_{GIoU}

0 is the threshold of whether the target is framed in 3D bounding box. From many experimental results, we found that not all the values of IoU are bigger than 0. Thus, we decided to take the ratio of the number of samples with IOU bigger than 0 to the total number of test-set as the measurement of accuracy, and then calculated the average value of all the IOU bigger than 0 as the measurement of precision. Table 2 offers more information about different loss function's performance on the test set.

Table 2. Different Loss Function's Performance

	SmoothL1	L_{IoU}	L_{GIoU}
Test set	366	366	366
Number of simples (IoU>0)	201	85	343
Accuracy	54.9%	23.2%	**93.7%**
Precision (average IoU)	0.30	**0.49**	0.13

From the Fig. 4, we find that the model with SmoothL1 trends to converge roughly when the epoch exceeds 300. Moreover, there is a comparison between the model with L_{IoU} and L_{GIoU} as shown in Fig. 5. We can observe that the loss of L_{GIoU} with a faster convergence than L_{IoU}, while compared to the SmoothL1 show the same performance.

From Table 2 we find that model with loss of L_{IoU} shows a better precision than loss of SmoothL1, which has a precision improvement of 0.19. It can be explained by that L_{IoU} represents the expected aim at evaluation of model at the phase of training, which is more significant than SmoothL1. However, there is a accuracy descend of 31.7% of L_{IoU} when compared to SmoothL1, which is caused by that L_{IoU} only focuses on the case that intersection of two bounding boxes is bigger than 0, but ignores the case that the intersection is 0. For L_{GIoU}, we observe that there is a large improvement of accuracy to 93.7%, which is due to that L_{GIoU} take both cases of intersection of two bounding boxes into consideration. Nevertheless, an unsatisfactory effect occurs on the precision, we consider this maybe because L_{GIoU} is incomplete for the specific intersection position of 3D bounding box, which is a problem that does not exist in 2D verification before. This provides a research point for our follow-up work.

Further, figures (see Fig. 6 Fig. 7) demonstrate some correct result predicted by the model with L_{GIoU}(Red: ground truth; Ink Blue: 3D bounding box).

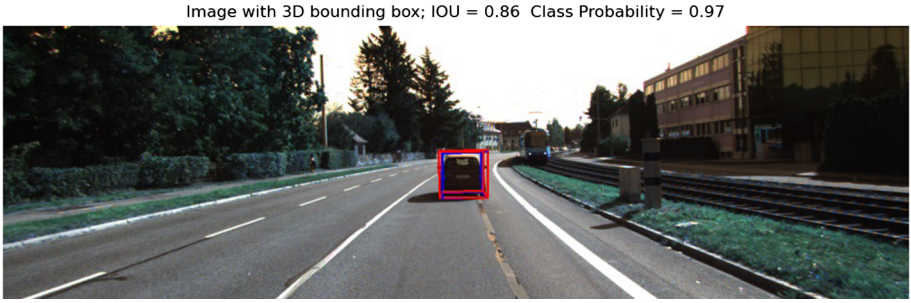

Fig. 6. Correct prediction. (Color figure online)

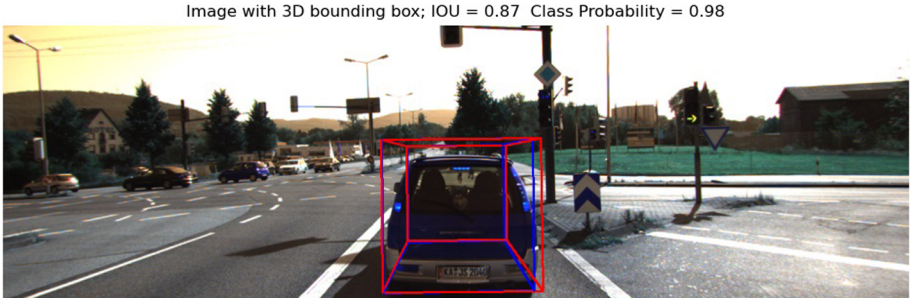

Fig. 7. Correct prediction. (Color figure online)

5 Conclution

In order to improve the performance of the bounding box regression in 3D environment, we optimized DSFN model based on Global Fusion model. Our model focuses on the optimization of loss function in training process. In experiments, we use IoU and $GIoU$ instead of SmoothL1 as the loss function of DSFN respectively. The results are obtained that IoU outperforms SmoothL1 on bounding box regression precision while $GIoU$ shows the best performance on bounding box regression accuracy.

References

1. Xu, D., Anguelov, D., Jain, A.: Pointfusion: deep sensor fusion for 3D bounding box estimation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, 18–22 June 2018. IEEE Computer Society, pp. 244–253 (2018)
2. Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. CoRR, vol. abs/1708.02002 (2017). http://arxiv.org/abs/1708.02002
3. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN (2017)
4. KITTI. http://www.cvlibs.net/datasets/kitti/raw_data.php

5. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I.D., Savarese, S.: Generalized intersection over union: a metric and a loss for bounding box regression. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, 16–20 June 2019. Computer Vision Foundation/IEEE, pp. 658–666 (2019). https://arxiv.org/pdf/1902.09630.pdf
6. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4
7. Lin, Tsung-Yi., et al.: Microsoft COCO: common objects in context. In: Fleet, David, Pajdla, Tomas, Schiele, Bernt, Tuytelaars, Tinne (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
8. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016. IEEE Computer Society, pp. 779–788 (2016). https://doi.org/10.1109/CVPR.2016.91
9. Girshick, R.B., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, 23–28 June 2014. IEEE Computer Society, pp. 580–587 (2014). https://doi.org/10.1109/CVPR.2014.81
10. Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T.: Selective search for object recognition. Int. J. Comput. Vis. **104**(2), 154–171 (2013). https://doi.org/10.1007/s11263-013-0620-5
11. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. CoRR, vol. abs/1804.02767 (2018). http://arxiv.org/abs/1804.02767
12. Redmon, J.: Yolo9000: better, faster, stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017. IEEE Computer Society, pp. 6517–6525 (2017). https://doi.org/10.1109/CVPR.2017.690
13. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: Lee, D.D., Sugiyama, M., von Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, 5–10 December 2016, Barcelona, Spain, pp. 379–387 (2016). http://papers.nips.cc/paper/6465-r-fcn-object-detection-via-region-based-fully-convolutional-networks
14. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: deep learning on point sets for 3D classification and segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017. IEEE Computer Society, pp. 77–85 (2017). https://doi.org/10.1109/CVPR.2017.16
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016. IEEE Computer Society, pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90