Ensemble R-FCN for Object Detection

Jian Li, Jianjun Qian and Yuhui Zheng

[1]Computer Science and Engineering
Nanjing University of Science & Technology, Nanjing 210094, China
[2]School of Computer and Software,
Nanjing University of Information Science & Technology, Nanjing 210044, China
lijiannuist@gmail.com, csjqian@njust.edu.cn, zhengyh@vip.126.com

Abstract. This paper presents an Ensemble R-FCN framework for object detection. Specifically, we mainly make three contributions to our detection framework: (1) we augment the training images for R-FCN when facing the limited training samples and small object. (2) We further introduce several enhancement schemes to improve the performance of the single R-FCN. (3) An ensemble R-FCN is proposed to make our detection system more robust by combining different feature extractors and multi-scale inference. Experimental results demonstrate the advantages of the proposed method. Especially, our method achieved the performance of AP score 0.829 which ranked No.1 among over 360 teams in Ucar Self-drving deep learning Competition.

Keywords: Object detection, R-FCN, Self-driving, Deep learning.

1 Introduction

Object detection is a hot theme in the field of computer vision since it has great potential applications in self-driving. The traditional object detection methods mainly use the hand-designed features (SIFT[1] or HOG[2]) to describe the image. Recently, deep convolutional networks as the powerful tool for feature learning is widely used in object detection and image classification. In fact, the CNN based object detection methods can be divided into two categories: Region Proposal Network (RPN) based methods and one-stage detection schemes. In Faster R-CNN, the main contribution is that they introduce the RPN into Fast R-CNN [7]. We can use Fast R-CNN to refine the high-quality region proposals generated via RPN for achieving better results. S. Ren et al [8] argue that the region-wise features, which are pooled from proposals, can faithfully cover the region character. In this way, the object detection performance can be significantly improved. The most representative method of the second category is Single Shot MultiBox Detector (SSD) [6]. SSD get rid of RPN and directly predict bounding boxes and confidences. W. Liu et al think that SSD is easy to train and can be straightforward integrated into detection systems. However, it's difficult to obtain the satisfied performance of the small object detection task for one-stage detection methods without feature resampling.

Recently, region-based detector R-FCN [9] is proposed for achieving competitive performance and inference speed. This method introduce position-sensitive score maps to boost classification and object detection. Additionally, R-FCN [9] can use different fully convolutional network like resnet101 or resnet152 [10] as the backbone for feature extraction. A robust and efficient feature extractor can decrease the effects of vehicle occlusion, confusion between pedestrian and cyclist. In addition, the training images are limited in most real-world applications, which will lead the deep model over-fitting.

To address this problem and further improve the detection performance, we first give the data augmentation scheme and then propose an ensemble model by combing multi-scale inference and more feature extractors. Finally, we evaluated the proposed method on the Ucar self-driving deep learning competition and achieved the performance of AP score 0.829 which ranked No.1 among over 360 teams in total.

2 Related work

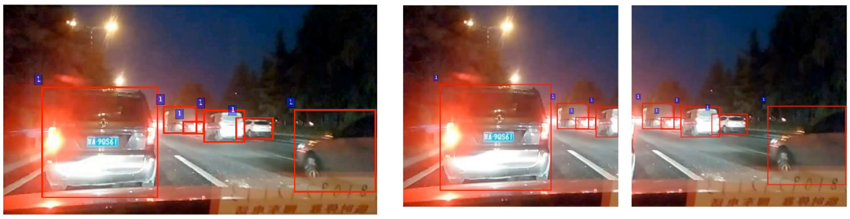
The pioneering work (R-CNN) extract the object proposals via selective search [5] and employ the deep convolutional networks to generate features for object proposals classification. To avoid repeatedly computing the convolutional features, SPPnet just computed the feature maps of the entire image once. Then the feature maps are pooled in arbitrary regions to obtain the fixed-length outputs. However, SPPnet is also a multi-stage pipeline including feature extraction, network fine-tuning, SVMs training and bounding-box regression. This scheme will spend expensive computation load. To improve training and testing speed while also increasing detection accuracy, Fast R-CNN[7] presents a ROI pooling layer convert the features inside the region proposal into a small feature map. Compared with R-CNN and SPPnet, Fast R-CNN can complete the training process in one stage by using multi-task loss.

However, above mentioned methods highly depended on selective search to generate sparse proposals. To avoid this scheme, the region proposal network (RPN), which shares full-image convolutional features with the down-stream detection network, is proposed to complete the task. Faster R-CNN actually combines the cost-free region proposals scheme RPN and detection into a unified network [8]. It's a pity that faster R-CNN still apply a costly per-region sub-network hundreds of times when Fast R-CNN processing region proposal produced by RPN. In [9], the authors presented region based fully convolutional networks (R-FCN) to speed the detection time. R-FCN employ the position-sensitive score maps on the top of fully convolutional network to balance translation-invariance in image classification and translation-invariance for object detection. Therefore, R-FCN is similar with Faster R-CNN, and both of them follow the pipeline of "CNN feature extraction plus ROI pooling and ROI classification".

3 Ensemble R-FCN

3.1 Data augmentation

As well known, Data augmentation is essential to combat over-fitting in deep models. To avoid this problem, we enlarge the training data by using label-preserving transformations. In this way, the transformed images are produced from the original images with little computation [11]. Given a road image (as shown in Fig. 1 (a)), we crop 2/3 of the image from left to right (as shown in Fig. 1 (b)). Meanwhile, we also crop 2/3 of the image from right to left (as shown in Fig. 1 (c)). Then, we resize the extended images as the same as the original image. Additionally, the bounding boxes should be recomputed for the two new images. We remove the truncated bounding boxes whose areas are less than 1/5.



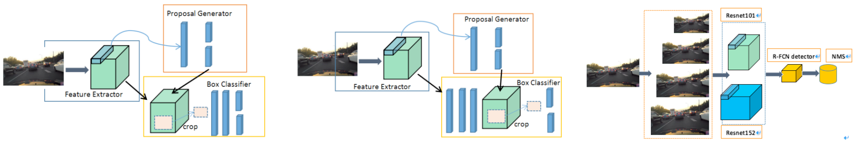
(a) original image (b) left image (c) right image

Fig. 1. (a) Real road images which are taken by vehicle recorder, Annotation is showed by the red bounding box and the top left number. (b) left truncated image; (c) right truncated image. (b)(c) also generate new bounding boxes around the cropping line.

3.2 Detector

In this section, we first give the comparisons between Faster R-CNN and R-FCN, and then introduce our ensemble R-FCN.

Faster R-CNN (Fig.2 a) integrates the region proposal network(RPN) into Fast R-CNN [7]. RPN is trained end-to-end and can generate high-quality region proposals. We can then crop features from the feature map according to the region proposals. The final feature will lead to more accurate detection. Region-based Fully Convolutional Networks (R-FCN) is similar with Faster R-CNN. However, for minimizing the amount of computation, R-FCNcrops features from the last feature layer before prediction (as shown in Fig. 2b). A position-sensitive mechanism is proposed to keep translation variance for localization representations. Compared with Faster R-CNN, R-FCN can achieve better performance (83.6% mAP on the VOC 2007 test).



(a) Faster R-CNN (b) R-FCN (c) Ensemble R-FCN

Fig. 2. (a)The architecture of Faster R-CNN; (b)The architecture of R-FCN; (c)The architecture of our Ensemble R-FCN.

To further improve the detection performance, we presented a simple and effective detection system named Ensemble R-FCN. Specifically, we employ resnet101 and resnet152 to extract the feature maps of different scale images since they describe the different texture grain feature. In this paper, we provide four different scale images. So, there are eight stack feature maps after feature extraction. For each feature maps, we employ the R-FCN to obtain the candidate object windows. Subsequently, all the feature maps were reshaped to the same size and the corresponding candidate windows also were zoomed. In this way, we can aggregate all the candidate windows into the original image. Fig. 2(c) shows the architecture of our Ensemble R-FCN.

3.3 Enhancement methods

Here, we mainly design the abundant anchors to achieve the more accurate bounding box for different kinds of object. Anchor, which is centered at the sliding window, is actually used for parameterizing proposals and ground truth bounding box. Each sliding position has k anchors with different scales and aspect ratio. A WxH convolutional feature map will contain WxHxK anchors.

Non-maximum suppression (NMS) selects the detection window with highest confidence score and eliminates the detection windows when the intersection over union (IOU) ratio is higher than the threshold. We choose different thresholds and evaluate them based on Faster R-CNN (VGG1024). We set NMS threshold 0.45, which can obtain satisfied detection performance.

4 Experimental Evaluation

In this section, we will evaluate the proposed method on Ucar Self-drving deep learning Competition datasets. The training set contains 10000 images and each image has annotation including object bounding box and object category. The annotation provided by Ucar is in format of Json and encodes vehicle, pedestrian, cyclist and traffic lights using number (1, 2, 3, 20). Figure1 shows the real road image and object annotation including red bounding box and top left blue number. Actually, these images are taken by vehicle recorder and has complex environment including pixel blur, different intensity of light, vehicle occlusion, confusion between cyclist and pedestrian. Season1 offered 2000 validation images for examining and improving our detector. Season2 provided 3000 testing images for evaluating teams' detection system and give the final rank.

Each training images has the same size 360x640 and contains different numbers of above four categories object. We analyze all training images and calculate the number of each category. Vehicle has highest proportion among all bounding boxes and its ratio is 0.84, and other categories take a small proportion.

In this experiment, the computing environment is GPU K80 with 12GB. We adopt hyper-parameters: momentum 0.9, weight decay 0.0005, and train our network using SGD with initial learning rate 0.001 and 0.0001 after 10w iterations. Here, we evaluate four baseline models: Faster R-CNN (VGG1024), Faster R-CNN (VGG16), R-FCN (resnet101), R-FCN (resnet152) on augmented training set. By analyzing the results of each detector in Table1, we choose R-FCN as our base detector. Based on Faster R-CNN (VGG1024), we evaluate our anchor design to match the size of four categories (vehicle, pedestrian, cyclist, traffic light). The reason why we choose VGG1024 for feature extraction is that we can search parameters with fewer training time. Table 2 shows detection performance when chooses anchors with different scale and aspect ratio. Finally, we design 5x5 anchors with aspect ratio (0.3,0.5,1,2,3) and scale (2,4,8,16,32).

Table 1. Performance of four detectors: Faster R-CNN(VGG1024), Faster R-CNN(VGG16), R-FCN(resnet101), R-FCN(resnet152).

Average	Faster R-CNN	Faster R-CNN	R-FCN	R-FCN
Precision	(VGG1024)	(VGG16)	(resnet101)	(resnet152)
vehicle	0.794	0.865	0.867	0.875
cyclist	0.264	0.409	0.476	0.531
pedestrian	0.577	0.643	0.710	0.737
Traffic light	0.149	0.253	0.443	0.449
Weighted AP	0.720	0.8019	0.819	0.8305

Table 2. Anchor design based on Faster R-CNN (VGG1024)

Anchor number	3x3	3x6	5x5
Aspect ratio	(0.5,1,2)	(0.5,1,2)	(0.3,0.5,1,2,3)
Anchor scale	(8,16,32)	(2,4,6,8,16,32)	(2,4,8,16,32)
AP	0.753	0.756	0.761

In this competition, we use two feature extractors for multi-scale testing image. We provide four scales (648x1152, 720x1280, 792x1408, 864x1536) of the test image for the proposed Ensemble R-FCN. Then Ensemble R-FCN integrated resnet101 and resnet152 for feature extraction. Finally, our Ensemble R-FCN obtains the AP score 0.829 which ranked No.1 among over 360 teams in total. Figure 3 shows detection examples using our Ensemble R-FCN.

5 Conclusion

In this paper, we developed an ensemble R-FCN method for object detection. To avoid over-fitting, we try our best to augment training data of Ensemble R-FCN. In addition, we further modify the anchor module and NMS in R-FCN to adapt to multi-scale object detection. At last, different feature extractors and multi-scale Inference strategy are integrated into R-FCN to make our detection system more robust. With the help of augmenting data and averaging model, the excellent performance demonstrates the advantages of our Ensemble R-FCN.

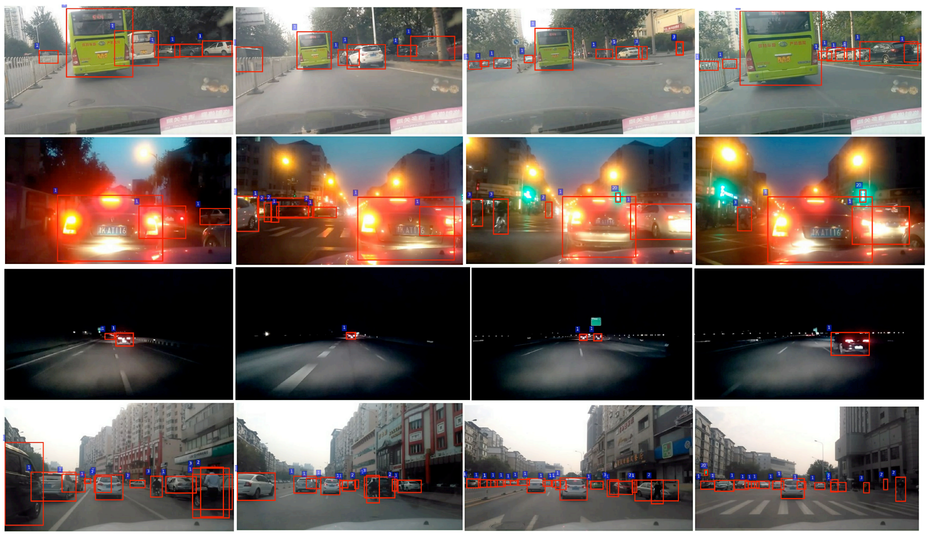


Fig .3 Detection examples on real road image using Ensemble R-FCN.

References

[1] D. Lowe, Distinctive image features from scale-invariant key-points, in IJCV, 2004.

[2] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, in CVPR, 2005.

[3] Hoiem, D. Chodpathumwan, Y. Dai, Q. Dai. Diagnosing error in object detectors. In: ECCV 2012. (2012)

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.

[5] K. E. Van de Sande, J. R. Uijlings, T. Gevers, and A. W. Smeulders. Segmentation as selective search for object recognition. In ICCV, 2011.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In ECCV, 2014.

[7] R. Girshick. Fast R-CNN In ICCV, 2015.

[8] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015.

[9] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. arXiv preprint arXiv:1605.06409, 2016.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR, 2016.

[11] D. J Li, J Li, B. N, S. Q Sun. Deconvolution Single Shot MultiBox Detector for Supermarket commodity detection and classification. In ICDIP 2017.