MULTI-SCALE OBJECT DETECTION WITH FEATURE FUSION AND REGION OBJECTNESS NETWORK

Wenjie Guan, YueXian Zou, Xiaoqun Zhou*

ADSPLAB/Intelligent Lab, School of ECE, Peking University, Shenzhen,518055, China
*Corresponding author: zouyx@pkusz.edu.cn

ABSTRACT

Though tremendous progresses have been made in object detection due to the deep convolutional networks, one of the remaining challenges is the multi-scale object detection(MOD). To improve the performance of MOD task, we take Faster region-based CNN (Faster R-CNN) framework and work on two specific problems: get more accurate localization for small objects and eliminate background region proposals, when there are many small objects exist. Specifically, a feature fusion module is introduced which jointly utilize the high-abstracted semantic knowledge captured in higher layer and details information captured in the lower layer to generate a fine resolution feature maps. As a result, the small objects can be localized more accurately. Besides, a novel Region Objectness Network is developed for generating effective proposals which are more likely to cover the target objects. Extensive experiments have been conducted over UA-DETRAC car datasets, as well as a self-built bird dataset (BSBDV 2017) collected from Shenzhen Bay coastal wetland, which demonstrate the competitive performance and the comparable detection speed of our proposed method.

Index Terms**—**multi-scale object detection, feature fusion, the Region Objectness Network, convolutional neural network

1. INTRODUCTION

Though tremendous progresses have been made in object detection recently due to the deep convolutional neural networks (DCNNs)[1, 2], multi-scale object detection (MOD) is one of the remaining challenging tasks. As shown in Fig.1, costal wetland bird detection and vehicles detection for traffic surveillance are two typical MOD problems in the real world, which contain different scales of objects and have a large proportion of small objects in distant view. Upon such applications, experiments showed that the performance of the state-of-art object detection methods for MOD tasks is unsatisfactory.

One of most important and successful frameworks for generic object detection is the region-based CNN (R-CNN family) method[3-5]. This family of methods divided the object detection process into two tasks, including proposals generation; proposals classification and regression. In this study, we work on Faster R-CNN framework[5]. In principle, Faster R-CNN generates the object proposals by Region Proposals Network (RPN) and implements the classification and regression by RoIs-wise classification network (RCN). Specifically, RPN extracts proposals by generating candidate boxes of specific size and aspect ratio at each region of the image. Then, these proposals are further classified into object categories and background by RCN. With DCNNs, Faster R-CNN has shown high accuracy on mainstream object detection benchmarks [1, 2, 6]. Moreover, a variant of networks derived from Faster R-CNN also achieve state-of-art results in many other computer vision problems beyond object detection [7-10].

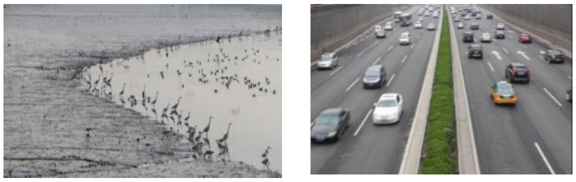


Fig. 1. Examples of multi-scale object detection. Costal wetland bird detection (left) and vehicles detection for traffic surveillance (right).

However, for MOD tasks, the performance of Faster R-CNN is also unsatisfactory. Carefully analysis shows that there are two main problems.

First, in Faster R-CNN method, the high-level feature maps have significantly lower resolution than the original image, which are more effective to capture high-level semantic knowledge but insufficient to capture fine-grained spatial details. Therefore, using the high-level feature maps, it is difficult to get the precise location of small objects. Second, with many small objects for the MOD tasks, areas of the background are greatly larger than that of the foreground. As a result, RPN generates many redundant background candidate boxes (termed as background proposals). However, the redundant background proposals, which do not cover any objects, will cause data imbalance for further training task in RCN [11].

To tackle the problems discussed above, we propose a new MOD method based on Faster R-CNN framework by introducing a feature fusion module and a novel Region Objectness Network. Firstly, to supplement the fine-grained knowledge for small objects in the final feature representation, we introduce a feature fusion module to fuse

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

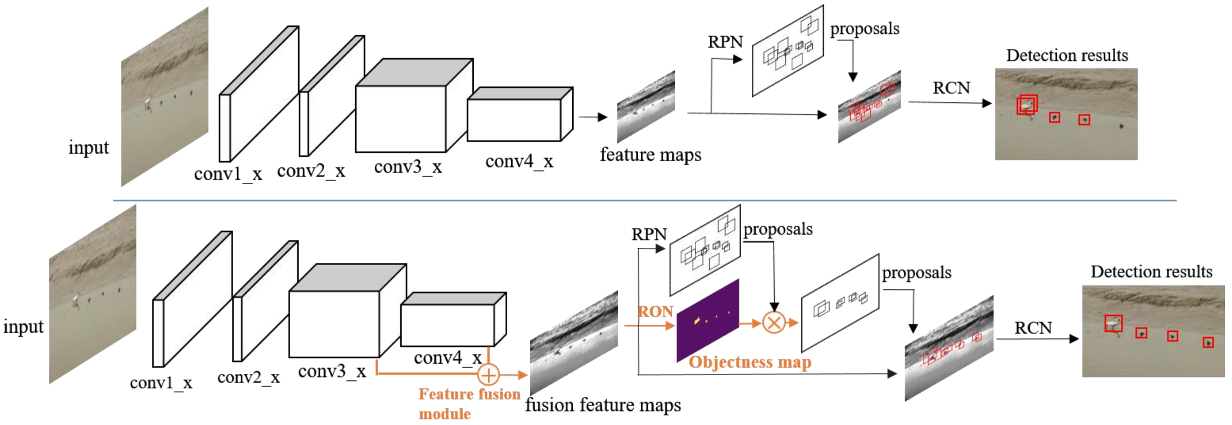


Fig. 2. The overview of Faster R-CNN Framework (top) and our MOD method (bottom).

the strongly semantic representation in the top layers into the fine resolution intermediate layers. Secondly, to eliminate the redundant background proposals, a novel Region Objectness Network is developed. To provide meaningful performance evaluation as well as the need of the research project undertaken by our team, a costal wetland bird dataset, (BSBDV 2017) has been built, which contains more than 1700 costal images taken in different days and different time slots. Meanwhile, the performance evaluation is also conducted using a public UA-DETRAC car dataset for MOD task.

The rest of paper is organized as follow: in Section 2, we present the details of our proposed MOD method; Section 3 presents intensive experiments and experimental analysis; Section.4 concludes our work.

2. PROPOSED MOD METHOD

In this section, we describe our proposed MOD network in details. Firstly, we describe the whole pipeline of the detection network in Section 2.1. Then, we present the feature fusion module in section 2.2. The RON is introduced in section 2.3.

2.1. The pipeline of the detection network

As introduced above, our proposed MOD method is based on Faster R-CNN framework [5]. Fig.2 shows the frameworks of Faster R-CNN and our MOD method, respectively. From Fig.2, it is clear that different from the Faster R-CNN, our MOD method mainly consist of 4 parts, including the feature fusion module, the Region Objectness Network (RON), the Region Proposal Network (RPN) and the RoIs-wise classification network (RCN). Following the protocol using in [12], we also use ResNet-101 as the backbone network, RPN, RON and RCN share the computation from conv1_x to conv4_x. In our study, as discussed in Section 1, to get better feature representation for small scale objects, a feature fusion module is designed. The details will be given in Section 2.2. To eliminate the redundant background proposals, a novel binary objectness map generate by RON is proposed. The details will be introduced in Section 2.3. Finally, the RoIs-wise classification network is applied to classify the generated proposals into object categories and background.

2.2. Feature fusion module

The illustration of our feature fusion module is shown in Fig. 3. In order to get better feature representation for different scale objects (especially for small objects), semantic knowledge in the top layers and fine-grained details in shallower layers are fused. We choose the layer conv3_4 and conv4_6 of the backbone network as the input of the feature fusion module. Both of them are the top layer of its stage. To make conv4_6 layer and conv3_4 layer have the same size, we employ a 2 × 2 deconvolution layer on conv4_6 to up-sampled the feature maps. In our design, the filter of deconvolution layer is fixed as bilinear. Besides, it is noted that the channel dimension of conv3_4 and conv4_6 are different. To solve this issue, we insert a 1 × 1 convolutional layer after conv3_4 to increase channel dimension. After these operations, the feature maps from different levels can be summarized point to point with equivalent weights. In order to further suppress the aliasing effect of the up-sampling process, a 1×1 convolutional layer is appended on the merged map to generate the final fusion feature. In conclusion, as described above, our designed feature fusion module is able to obtain the finer resolution fusion features which provide representation containing the highly abstracted knowledge and fine-grained details of small objects.

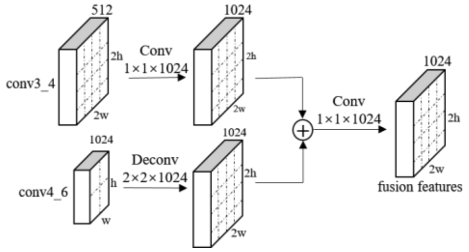


Fig. 3. Illustration of the feature fusion module. Features are combined by element-wise addition.

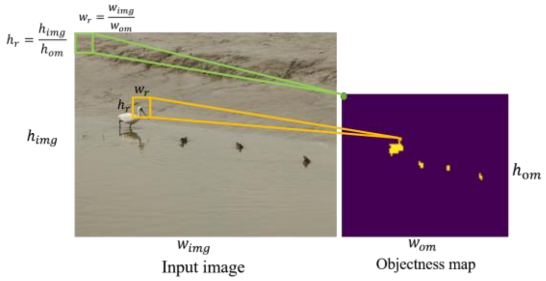


Fig. 4. An example image and its corresponding objectness map. Each pixel of the objectness map corresponds to a governing region with fixed size in the image. Yellow pixels indicate foreground governing region (yellow box) and purple pixels indicate background governing region (green box). The size of the governing region is decided by the ratio of the input image size to the objectness map size.

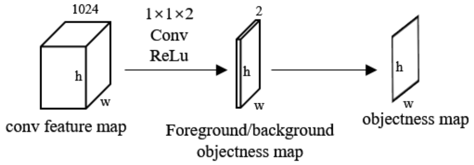


Fig. 5. Illustration the structure of RON.

2.3. Region Objectness Network

In this subsection, we will introduce our proposed novel Region Objectness network (RON), which aims at eliminating the background proposals. We formulate the task as to predict the likelihood of each region in the input image being a foreground object as opposed to background. A RON takes an image of arbitrary size as input and outputs a binary objectness map. As shown in Fig.4., each pixel of the objectness map only corresponds to a region in the image, which is called its *governing region* here. We model this process with a fully convolutional network (FCN) [13], which enable computation sharing with the RPN and the RCN elegantly.

As shown in Fig.5, to generate the objectness map, we append a 1×1×2 convolutional layer after the last shared convolutional feature maps to learn the score which measures the likelihood of the corresponding governing region being either a foreground object or a background one. According to the score, the background objectness maps and foreground objectness map are generated correspondingly. Then, each spatial position on the objectness map is labeled with the higher foreground/background objectness score category.

Obviously, our approach is supervised method. For training the RON, an objectness label (being a foreground object or not) is assigned to each governing region. We assign 1 to a governing region which has an Intersection-over-region (IoR, the ratio between the area of overlap and the area of governing region) overlap higher than 0.7 with any ground-truth box. A governing region whose IoR ratio is lower than 0.3 for all ground-truth boxes would be assign to 0. As shown in Fig. 4, the spatial size of the governing region is decided by the ratio of the input image size to the output objectness map size. It's remarkable that a single ground-truth box may assign positive label to multiple regions.

The loss function used is defined as:

$$L(\{p_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) \qquad (1)$$

where i is the governing region index, the classification loss L_{cls} is the cross-entropy loss over two classes. p_i is the predicted objectness score. The ground-truth label p_i^* is 1 if the governing region is positive, and is 0 if the governing region is negative. Here N_{cls} represent the total number of governing regions in an image.

The RON is trained end-to-end by back-propagation and stochastic gradient descent (SGD). Each mini-batch arises from a single image that contains certain number of positive and negative governing regions. Further implementation details are given in section 3.2.

3. EXPERIMENTS

In order to evaluate the effectiveness of our proposed MOD method, we conduct experiments on two multi-scales object detection datasets, including the UA-DETRAC Object Detection Benchmark [14] and self-built bird dataset (BSBDV 2017). Average Precision is used as the evaluation metric followed by the standard PASCAL VOC criteria, *i.e.*, IoU > 0.5 between ground truths and predicted boxes [1].

3.1. Datasets

3.1.1. UA-DETRAC

The UA-DETRAC Object Detection Benchmark [14] is a large scale car detection benchmark, which contains 1.21 million car instances. The images are of resolution 960×540. In our experiments, we choose 1,500 images from traffic surveillance and one example is given in Fig.1.

3.1.2. BSBDV 2017

The birds dataset of Shenzhen Bay in distant view (BSBDV 2017) is our self-built bird dataset, which aims to provide the community with sufficient bird images for multi-scale object detection research. We manually annotate 1,772 images in BSBDV 2017 for 10 categories of birds and result in 7,835 labeled bounding boxes. The image resolutions in BSBDV 2017 are of 2736×1824 (344 images), 4288×2848 (656 images) and 5472×3648 (772 images) respectively. Evaluating the images in BSBDV 2017, it can be seen that the size of birds varies greatly from 18×30 to 1274×632 which is a great challenge for object detection. In our experiments, 1,421 images are used for training and the remaining ones are for testing. This dataset will be made publicly available.

3.2. Implementation Details

For both car detection and bird detection tasks, we use the ImageNet [6] pretrained ResNet-101 [12] model to initialize our backbone network. As shown in Fig. 2 and described in Section 2.2 and 2.3, the parameters of our designed convolutional layers for feature fusion and RON are initialized with "Xavier" [15]. We respectively resize the input image to 600 and 540 on the shorter side for BSBDV 2017 and UA-DETRAC. The implementation is based on the publicly available Faster R-CNN framework [5] built on the Caffe platform [16].

The proposed MOD network is trained in the end-to-end manner with Stochastic Gradient Descent (SGD), where momentum is 0.9, and weight decay is 0.0005, on a single NVIDIA GeForce GTX TITAN X GPU with 12GB memory. The learning rate is set of 0.0001 for 70k mini-batches, and 0.00001 for the next 30k mini-batch. Each mini-batch involves 1 image and 512 object proposals per image. Other settings are followed by [3].

3.3. Detection accuracy

In our experiments, the state of arts object detection methods including Region-free (SSD [17] and YOLOv2 [18]) and region-based (Faster R-CNN [5]) have been taken as comparison methods. Table 1 and Table 2 provide the experimental results in terms of average precision on UA-DETRAC and BSBDV 2017. From Table 1, we can see that the average precision of our MOD method is 71.1 on UA-DETRAC, which is 27%, 4.1%, 12.8% and 8.8% higher than that of YOLOv2, SSD300, Faster R-CNN (ResNet-50) and Faster R-CNN (ResNet-101), respectively. Similarly, as shown in Table 2, on BSBDV 2017, our MOD method achieves 58.8%, which is 24.2%, 16.8%, 14.5% and 8% higher than that of YOLOv2, SSD300, Faster R-CNN (ResNet-50) and Faster R-CNN (ResNet-101), respectively. One example of detection results is shown in Fig.6. From Fig. 6, a comparison of the left and right images shows that the detection capability of our proposed MOD method is improved significantly. Let's take bird detection as example. For small bird objects in distant view, our method gives more accurate bounding boxes in fitting the objects and much less missing bird objects. Meanwhile, for closer view, our MOD method also gives more accurate detection results and less missing cases compared with Faster R-CNN. Similar conclusions can be drawn for car detection. These experimental results validate the effectiveness of our proposed MOD method.

3.3. Detection speed

We evaluate the average running time of our proposed MOD method and Faster R-CNN for processing 1 image. Without loss of the generality, we take BSBDV 2017 as example and the results are presented in Table 3. Compared with Faster R-CNN, our MOD method asks a slightly smaller running time and much better precision accuracy. These experimental results indirectly demonstrate that our proposed MOD model is able to remove the redundant background proposals but keep more information for object detection under multi-scale object detection scenario.

Table 1 Detection results on UA-DETRAC

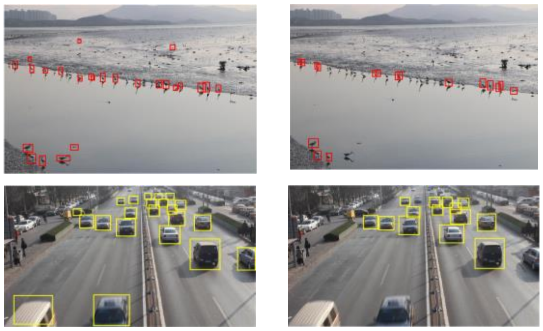
Method	Base Network	proposals	AP (%)
YOLOv2	Darknet	-	44.3
SSD300	VGG-reduce	-	67
Faster R-CNN	ResNet-50	1200	58.3
Faster R-CNN	ResNet-101	1200	62.1
Ours	ResNet-101	1200	**71.1**

Table 2 Detection results on BSBDV 2017

Method	Base Network	proposals	AP (%)
YOLOv2	Darknet	-	34.6
SSD500	VGG-reduce	-	42
Faster R-CNN	ResNet-50	1200	44.3
Faster R-CNN	ResNet-101	1200	50.8
Ours	ResNet-101	1200	**58.8**

Table 3 Detection speed on BSBDV 2017

Method	Base Network	AP (%)	Time (sec)	FPS
Faster R-CNN	ResNet-101	50.8	0.679	1.47
Ours	ResNet-101	**58.8**	**0.611**	**1.64**



(a) our MOD method (b) Faster R-CNN(ResNet-101)

Fig. 6. Detection results of our MOD method (a) and Faster R-CNN (b) on BSBDV 2017(row 1) and UA-DETRAC (row 2).

4. CONCLUSION

In this paper, we proposed a multi-scale object detection method by introducing a novel a feature fusion module and a novel Region Objectness Network aiming at improving the localization performance of small objects and eliminating the redundant proposals. To facilitate this study, a self-built bird dataset (BSBDV 2017) is established which will be available for public. Our proposed MOD method exhibits strong competency in handling multi-scale object detection tasks, where our method achieves 8.8% and 8% higher average precision over that of Faster R-CNN (Resnet-101) on UA-DATRAC and BSBDV 2017, respectively.

5. ACKNOWLEDGMENT

This paper was partially supported by the Shenzhen Science & Technology Fundamental Research Program (No: JCYJ20160330095814461) & Shenzhen Key Laboratory for Intelligent Multimedia and Virtual Reality (ZDSYS201703031405467).

6. REFERENCES

[1] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International journal of computer vision,* vol. 111, no. 1, pp. 98-136, 2015.

[2] T. Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," vol. 8693, pp. 740-755, 2014.

[3] R. Girshick, "Fast R-CNN," *IEEE International Conference on Computer Vision,* pp. 1440-1448, 2015.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," pp. 580-587, 2013.

[5] S. Ren, R. Girshick, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence,* vol. 39, no. 6, pp. 1137-1149, 2017.

[6] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision,* vol. 115, no. 3, pp. 211-252, 2014.

[7] F. Bu, Y. Cai, and Y. Yang, "Multiple Object Tracking Based on Faster-RCNN Detector and KCF Tracker."

[8] A. Fuentes, I. Jun, S. Yoon, and S. P. Dong, "Pedestrian Detection for Driving Assistance Systems based on Faster-RCNN," in *International Symposium on Information Technology Convergence Isitc*, 2016.

[9] W. Jiang and W. Wang, "Face detection and recognition for home service robots with end-to-end deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[10] J. Li *et al.*, "Facial Expression Recognition with Faster R-CNN," *Procedia Computer Science,* vol. 107, no. C, pp. 135-140, 2017.

[11] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," 2017.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Computer Vision and Pattern Recognition*, 2016, pp. 770-778.

[13] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis & Machine Intelligence,* vol. 39, no. 4, pp. 640-651, 2017.

[14] L. Wen *et al.*, "UA-DETRAC: A New Benchmark and Protocol for Multi-Object Detection and Tracking," *Computer Science,* 2015.

[15] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *Journal of Machine Learning Research,* vol. 9, pp. 249-256, 2010.

[16] Y. Jia *et al.*, "Caffe: Convolutional Architecture for Fast Feature Embedding," in *ACM International Conference on Multimedia*, 2014, pp. 675-678.

[17] W. Liu *et al.*, *SSD: Single Shot MultiBox Detector*. Springer International Publishing, 2016, pp. 21-37.

[18] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," 2016.