SELECTING OPTIMAL PROPOSAL NUMBER FOR IMAGE-BASED OBJECT DETECTION

Wenjie Guan[1], *Xiaoqun Zhou*[1], *Ge Li*[1,2], *Yuexian Zou*[1,2*]

[1]ADSPLAB, School of ECE, Peking University, Shenzhen, China
[2]Peng Cheng Laboratory, Shenzhen, China
*Corresponding author: zouyx@pkusz.edu.cn

ABSTRACT

In order to balance the detection time and accuracy, the state-of-the-art region-based detectors use a fixed number of proposals to obtain detection results in the inference phase. However, in surveillance scenes, object population varies in different images, causing the fixed proposal number becomes an undeterminable hyper-parameter, which needs to be correspondingly adjusted to maintain high recall. To solve this problem, we propose two image-level optimal proposal number selection methods called linear proposal number (LPN) selection method and adaptive proposal number (APN) selection method respectively, both aiming at selecting an optimal proposal number for each image to adapt both the images with sparsely and densely distributed objects. In LPN selection method, we introduce a linear weighting hyper-parameter to formulate the relationship between the actual object number and proposals' scores to obtain the optimal proposal number. To avoid setting the hyper-parameter manually, we further propose another APN selection method where the optimal proposal number of each image is selected by exploring the distribution of the proposals' scores. Results obtained from the UA-DETRAC car dataset and self-built bird dataset (BSBDV 2017) show that our proposed methods can largely improve the detection performance in terms of detection time and accuracy without any re-training process.

Index Terms— Object detection, Optimal proposal number, Linear proposal number selection, Adaptive proposal number selection, Surveillance scenes

1. INTRODUCTION

Object detection in surveillance scenes is an active research area since it has a wide range of applications, such as ecological monitoring, traffic surveillance, society security surveillance, etc. Different from the generic object detection, in surveillance scenes, object population is usually distributed inconsistently and changes unexpectedly in different images.

Fig. 1 Examples of object detection in surveillance scenes. Costal wetland bird detection (left) and vehicle detection for traffic surveillance (right), containing densely distributed objects (above) and sparsely distributed objects (bottom).

Fig 1 shows the examples of costal wetland bird detection and vehicle detection where the object population varies greatly in particular monitoring scenes.

One of the most important and successful frameworks for generic object detection is the region-based CNN (R-CNN family) method [3-5]. During inference, an object detection network performs a sequence of convolution operations over an image using deep convolutional neural network (CNN). The network then bifurcates into two branches, including Region Proposals Network (RPN) and RoIs-wise classification network (RCN). First, RPN extracts proposals by generating anchor boxes of specific size and aspect ratios at each region of an image. It then, rank these anchor boxes and adopt non-maximum suppression (NMS) to the selected top-K ranked anchors to generate P higher-quality proposals. After that, RCN generates classification and regression scores of the further selected top-N ($N<P<K$) ranked proposals generated by RPN [7]. Note that these top-N ranked proposals are supposed to cover all the objects.

In general, to balance the detection time and accuracy, N is a fixed number which is set as 300 by default. However, in actual surveillance scenes, N becomes an undeterminable

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). Special acknowledgements are given to Aoto-PKUSZ Joint Research Center of Artificial Intelligence on Scene Cognition & Technology Innovation for its support.

hyper-parameter considering that the object population varies greatly in different images. With a small *N*, it may fail to cover all the objects in an image where the objects are densely distributed, leading to a low recall. In contrast, with a large *N*, it will unfortunately increase the detection time though it could contribute a little to the recall and average precision.

To tackle the problem mentioned above, we present two variable proposal number selection methods based on region-based detectors called linear proposal number (LPN) selection method and adaptive proposal number (APN) selection method, respectively. Both methods aim at selecting an adequate proposal number for each image to adapt both the images with sparsely and densely distributed objects, which enables the generated proposals to cover all the objects in a more efficient way. More importantly, both methods achieve better performance without any re-training process on existing models so they can be widely adopted in any region-based CNN detector.

The remainder of this paper is organized as follows: Section 2 presents the details of our proposed two image-level optimal proposal number selection methods; Section 3 presents intensive experiments and comprehensive analysis; Section 4 concludes our work.

2. PROPOSED METHOD

Proposals are a set of candidate regions in an image that may potentially contain objects. Obviously, as the actual number of objects varies in different images, the number of proposals needs to be correspondingly adjusted to maintain a high recall.

Our goal is to select an optimal proposal number for each image in the inference phase. That is to say, a large proposal number for images with densely distributed objects and a small proposal number for images with sparsely distributed ones. In this section, we elaborate our proposed variable proposal number selection methods called LPN selection method and APN selection method, respectively.

2.1. Linear Proposal Number Selection Method

Given that the optimal proposal number of a single image *M* is of linear correlation with the actual number of objects n in that image, which can be represented as:

$$M = v_1 \times n \tag{1}$$

where v_1 is a proportional coefficient, $v_1 > 0$. Considering that the actual number of objects in an image is not directly available, we need to figure out a way to approximately estimate the *n* of that image.

The region proposal network (RPN) generates classification scores and regression offsets for anchor boxes of specific size and aspect ratios of an image. Then, these anchor boxes are ranked in descending order according to their scores. The selected top-*K* (default $K = 6000$) ranked anchor boxes to which the regression offsets are added to obtain image-level co-ordinates of each anchor. And then, greedy non-maximum suppression (NMS) is applied to these top-*K* ranked anchor boxes, after which we eventually generate *P* region proposals [7]. The total score of the proposals is represented as follows:

$$s = \sum_{i=1}^{P} score_i \tag{2}$$

where $score_i$ is the score of the *i-th* proposal, *P* is the total number of proposals after NMS. Inspired by the core idea of RPN, we formulate the relation between the proposals' scores and the actual number of objects n according to the following considerations. For each input image with the same resolution, the initial distribution of those anchor boxes is the same. Therefore, the total score of proposals *s* varies with the actual number of objects. Obviously, *s* is in positive correlation with *n*. For simplicity, *s* is assumed to be in linear relation with *n*, representing as:

$$s = v_2 \times n. \tag{3}$$

Based on equation (1) and (3), the optimal proposal number *M* is formulated as:

$$M = \frac{v_1}{v_2} \times s = w \times s. \tag{4}$$

Thus, *M* can be simply computed by multiplying linear weighting parameter *w* and proposals' scores *s*. Here, $w > 0$.

2.2. Adaptive Proposal Number Selection Method

In the LPN selection method proposed above, although the hyper-parameter of fixed proposal number is removed, the linear weighting hyper-parameter *w* still needs adjustment according to the object population variation in actual scenes. To avoid setting the hyper-parameter manually, we further propose the adaptive proposal number (APN) selection method.

As mentioned above, all the anchor boxes are ranked in descending order, then the NMS is performed to the top-*K* ranked anchor boxes which are eventually utilized to generate *P* region proposals. To explore the distribution of the scores of the region proposals after NMS, three scatter plots are shown below to illustrate the changing pattern of proposals' scores. Fig 2 shows three image with different object population from BSBDV 2017, together with their corresponding scatter plots of ranked *P* proposals' scores in descending order. It is noted that *P,* the total number of proposals after NMS, is indefinite. From each scatter plot, it could be observed that the ranked scores drop slowly in the tight range close to 1, and then drop much more rapidly around the median value, forming an 'inflection point' in the plot. Furthermore, by comparing the three pairs of images and their corresponding scatter plots, it can be found that the position of the 'inflection point' is closely related to the number of objects in each image. The more objects exist in an image, the more proposals with high scores are generated, then the position of the 'inflection point' moves right along the x-axis. Therefore, the optimal proposal number could be inferred from the 'inflection point'.

To be more clearly, we define the value of difference of the adjacent scores in each interval as the score descending

velocity. Then, the maximum of score descending velocity is computed and its corresponding proposal index is regarded as the optimal proposal number *M*. Meanwhile, it's observed that the 'inflection point' should not be in the tight range close to 1. Thus, a calculation range of scores are set as $score_i \in [0.5, 0.9]$, the set of their corresponding proposals' index are represented as *S*. Therefore, the optimal proposal number *M* in APN selection method is represented as:

$$M = \begin{cases} arg\max_{i \in S}\left(score_{i+\frac{d}{2}} - score_{i-\frac{d}{2}}\right) & if\ \min(score) < 0.9 \\ P & if\ \min(score) \geq 0.9 \end{cases} \quad (5)$$

where *P* represents the total proposal number after NMS, *d* represents the length of each small interval and it is set as 5 in our experiments, *min(score)* represents the minimum of the proposals' scores mentioned above.

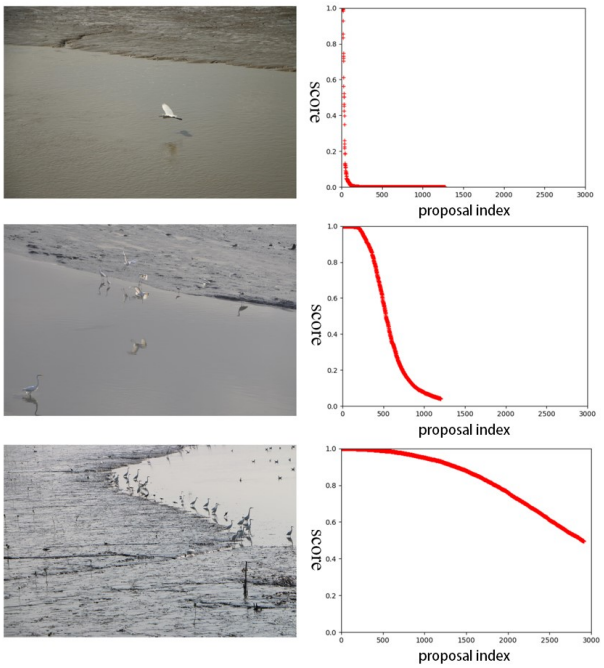


Fig. 2 Images from BSBDV 2017 and their corresponding scatter plots of ranked proposal score in descending order.

3. EXPERIMENTS

In our experiments, we adopt the Faster R-CNN framework with feature pyramid network, simply referred to as FPN [9], as our baseline model. In order to improve the detection performance in surveillance scenes, the LPN selection method and the APN selection method are respectively introduced to RPN in the inference phase. Experiments are conducted over two object detection datasets collected from surveillance scenes, including the UA-DETRAC object detection benchmark [10] and our self-built bird dataset (BSBDV 2017). Average Precision is the evaluation metric which follows the standard PASCAL VOC criteria, i.e., IoU > 0.5 between ground truths and predicted boxes [11].

3.1. Datasets

The Birds dataset of Shenzhen Bay in distant view (BSBDV 2017) is our self-built bird dataset, which is collected in the surveillance scenes of the National Nature Reserve in Shenzhen Bay. We manually annotate 1,772 images with over 10 categories of birds, leading to a total of 7,835 labeled bounding boxes. It is remarkable that the size of birds varies greatly from 18×30 to 1274×632, bringing difficulties to detection. Moreover, as mentioned above, the object population is severely unbalanced and varies greatly in different images. For evaluation, 1,421 images are used for training and the remaining ones are for testing.

The UA-DETRAC [10] is a large car detection benchmark, which contains 1.21 million car instances. The images are of resolution 960×540. To better demonstrate the detection performance in surveillance scenes, we choose 1,500 images which are captured in traffic surveillance scenes. It contains 27,264 instances in total whose size vary from 10×10 to 250×150. Since the UA-DETRAC dataset is collected from surveillance videos, its object population is relatively more balanced in different images.

3.2. Implementation Details

For both car and bird detection tasks, we use the ImageNet [12] pre-trained ResNet-50 model [13] to initialize our backbone network. We resize the input images to 640 and 512 on the shorter side for BSBDV 2017 and UA-DETRAC, respectively. The implementation is based on the publicly available Feature Pyramid Network [9] built on the Caffe platform[14].The whole network is trained end-to-end with Stochastic Gradient Descent (SGD) with learning rate of 0.0001 on a single NVIDIA GeForce GTX TITAN X GPU with 12GB memory. Other settings follow [9].

3.3. Performance Comparison

To illustrate the advantages of our proposed methods, we compare several detection methods in our experiments, including FPN [9] with traditional fixed proposal number (FPN), FPN with LPN selection method (LPN-FPN) and FPN with APN selection method (APN-FPN). Besides, we also use other state-of-the-art detection frameworks for comparison, including YOLOv2 [1], SSD [2], Faster R-CNN [5], R-FCN [8] and our previous work Faster R-CNN+RON [6]. The detection results on BSBDV 2017 and UA-DETRAC are shown in Table1 and Table 2 respectively. Noted that we also vary linear weighting hyper-parameter *w* to figure out its influence on detection performance. Experimentally, we set

Table 1 Detection results on BSBDV 2017

Framework	Backbone Network	Proposals	AP (%)	Time (s)
YOLOv2[1]	Darknet	-	34.6	-
SSD500[2]	VGG-reduce	-	42.0	-
Faster R-CNN[5]	ResNet-101	1200	54.5	0.679
Faster R-CNN +RON[6]	ResNet-101	1200	58.0	0.611
R-FCN [8]	ResNet-50	1200	61.5	0.403
FPN[9]	ResNet-50	300	61.2	0.459
FPN[9]	ResNet-50	600	61.3	0.498
FPN[9]	ResNet-50	1200	66.9	0.617
LPN-FPN	ResNet-50	Linear variation	**67.3**	0.476
APN-FPN	ResNet-50	Adaptive variation	**67.1**	0.467

Table 2 Detection results on UA-DETRAC

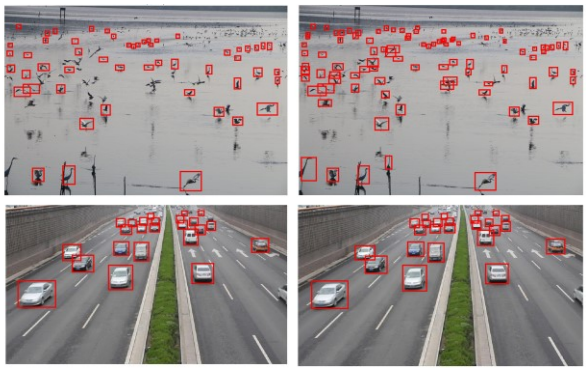
Framework	Backbone Network	Proposals	AP (%)	Time (s)
YOLOv2[1]	Darknet	-	44.3	-
SSD300[2]	VGG-reduce	-	67	-
Faster R-CNN[5]	ResNet-50	1200	58.3	-
Faster R-CNN[5]	ResNet-101	1200	62.1	-
Faster R-CNN+RON[6]	ResNet-101	1200	71.1	-
FPN[9]	ResNet-50	300	79.6	0.192
FPN[9]	ResNet-50	600	86.6	0.228
FPN[9]	ResNet-50	1200	86.6	0.328
LPN-FPN	ResNet-50	Linear variation	**86.8**	0.228
APN-FPN	ResNet-50	Adaptive variation	**86.7**	0.257

w as 0.3 and 0.12 when applying LPN-FPN on BSBDV 2017 and UA-DETRAC respectively.

From the experiment results on BSBDV 2017 (Table 1), it is obvious that our proposed LPN-FPN and APN-FPN achieve the best AP while keeping comparable detection time in contrast to other methods. Here, it is worthy of note that we come over a great problem derived from the FPN baseline that the average detection time increases with the growth of the proposal number. Although the AP can be improved with more proposals, the increased detection time, however, is unendurable under actual surveillance scenes. Specifically, compared with FPN with 300 proposals and with 600 proposals, our proposed LPN-FPN and APN-FPN improve at least 6% and 5.8% AP respectively, meanwhile maintain comparable detection time. Furthermore, compared with FPN with 1200 proposals, although the AP of our proposed LPN-FPN and APN-FPN improve by only 0.4% and 0.2%, the FPS of our proposed methods increase by 29.6% and 32.1%. Therefore, our proposed image-level optimal proposal number selection methods solve the conflict between the detection time and accuracy and show great adaptability over the object detection tasks with unbalanced object population and variable object number in surveillance scenes.

Table 2 shows the experiment results on UA-DETRAC car dataset. Our proposed LPN-FPN and APN-FPN also achieve the best AP. Under the same detection time, the AP of our LPN-FPN and APN-LPN improve by 0.2% and 0.1% compared with FPN with 600 proposals. It is notable that the limited improvement is mainly attributed to the balance and stability of object population in the UA-DETRAC dataset.

One example of detection results is shown in Fig 4. From Fig 3, a comparison of the left and right images indicates that our proposed LPN-FPN significantly improves the detection performance.



(a) FPN (600 proposals) (b) LPN-FPN

Fig. 3 Detection results on BSBDV 2017 (1st row) and UA-DETRAC (2nd row) using FPN (600 proposals) and our proposed LPN-FPN.

4. CONCLUSIONS

In this paper, we have proposed two image-level optimal proposal number selection methods called LPN selection method and APN selection method, and introduce them to region-based detectors in order to improve the detection performance in surveillance scenes. Both of them overcome the problem derived from the region-based methods that the fixed proposal number needs to be set manually, which undoubtedly undermine the detection performance. Instead, image-level optimal proposal number is selected for each image in the inference phase according to different object population. Extensive experiments on two object detection datasets in surveillance scenes have convincingly demonstrated the effectiveness of our proposed approaches in terms of average detection time and accuracy.

5. REFERENCES

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

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

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

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

[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] W. Guan, Y. Zou, and X. Zhou, "MULTI-SCALE OBJECT DETECTION WITH FEATURE FUSION AND REGION OBJECTNESS NETWORK," presented at the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018.

[7] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS — Improving Object Detection with One Line of Code," 2017.

[8] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks," 2016.

[9] T. Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 936-944.

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

[11] 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.

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

[13] 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.

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