OBJECT DETECTION VIA FEATURE FUSION BASED SINGLE NETWORK

Jian Li, Jianjun Qian, Jian Yang

School of Computer Science and Engineering
Nanjing University of Science & Technology

ABSTRACT

This paper presents a novel network, coined single unified fully convolutional network (SingleNet), for object detection. The proposed method mainly combines two ideas: (1) Our approach aggregates hierarchical features and map them into a uniform space. So, it can further enhance the feature representation ability to degrade the recognition error; (2) To approximate the ground-truth box, we design a set of dense boxes over different aspect ratios and scales per feature map pixel to regress bounding box. It's easy to improve the location performance using dense boxes scheme. Additionally, SingleNet is convenient to train and can be integrated into detection system. Experimental results (mAP: 0.776) on VOC 2007 test demonstrate the advantages of the proposed method over state-of-the-art methods.

Index Terms— Object detection, Single network, Hierarchical feature, Dense box, Feature fusion

1. INTRODUCTION

Compared to image classification, object localization and recognition is one of the most fundamental and challenging problem in the field of computer vision. Traditional object detection methods mainly adopt hand-engineered features like SIFT [1] or HOG [2] [3]. Recently, Deep Convolutional Networks have been successfully used in object detection. All the CNN based object detection methods can be roughly divided into two categories. The first one mainly based on Region Proposal Network (RPN) and employs two stage detection scheme like Faster R-CNN [8]. RPN is trained end-to-end and generate high-quality region proposals which then are further refined by Fast R-CNN detector. S. Ren.et.al [8] argue that the region-wise features pooled from proposal boxes can more faithfully cover the features of the region and lead to more accurate detection. Another set of methods like SSD [6] get rid of RPN and directly predict bounding boxes and confidences. W. Liu.et.al think that SSD is easy to train and straightforward to be integrated into detection systems. However, without feature resampling step, one-stage detection methods [6] can not obtain satisfied performance for small object detection.

Recently, Fully Convolution Network (FCN) [15] is proposed for segmentation by combining appearance information from shallow layer and semantic information from deep layer. Based on FCN, W. Liu adds global context to fully convolutional network by using the average feature for a layer to augment the features at each location [22]. T. Kong.et.al [11] develop a novel HyperNet by incorporating multi-resolution layer combinations into Faster R-CNN trunk network for detection. Inspired by SSD, Z. Cai.et.al [16] employ the multiple output layers to produce the RPN for matching objects of different scales. Distinctly, above methods are all based on Faster R-CNN framework to generate region proposal from RPN.

Motivated by YOLO [13] and SSD [14], we present a novel one-stage SingleNet framework for object detection. The proposed method just contains a single fully convolutional network trunk using VGG-net [17] without RPN. Based on above analysis, we believe that high layer has coarse, semantic information for classification while low layer has fine, high-resolution location information for detection. Therefore, we design a feature fusion network by aggregating hierarchical feature maps from low level output layers and high level ones. The feature maps are thus compressed into a uniform space. In this way, we can further enhance the feature presentation ability in one-stage prediction. Additionally, we introduce a set of dense boxes, which contains different aspect ratios and scales, over fused feature map location to improve the location performance.

2. RELATED WORK

As you know, the pioneering work R-CNN [4] generates features with deep convolutional networks to classify object proposals and obtains better results than classical methods. However, the computation time of R-CNN is expensive. To address this problem, the spatial pyramid pooling networks (SPP-net) is proposed to speed up the R-CNN by sharing computation. SPP-net is also a multi-stage pipeline that involves extracting features, fine-tuning a network, SVMs training and bounding-box regression. Fast R-CNN [7] has significantly improved the accuracy and efficiency of R-CNN and SPP-net. In Fast R-CNN, a convolutional feature map is generated via convolution and max pooling operations. For each object proposal, a fixed-length feature vector is extracted from ROI pooling layer and fed into a fully connected layers. The fully connected layer finally branch into two sibling layers for classification and

bounding box regression. Multi-task loss enables single-stage detector end-to-end training on shared convolutional features.

Nevertheless, one common issue of above detection methods is that they rely on Selective Search (SS) [23] which is time-consuming to generate sparse proposals. Instead of region proposal algorithms to hypothesize object location, Faster R-CNN [8] gives a Region Proposal Network (RPN) to improve the performance of region proposals generation. RPN shares full-image convolutional features with the down-stream detection network. In this way, region proposals generation can be considered as a nearly cost-free step. However Faster R-CNN still apply a costly per-region sub-network hundreds of times when Faster R-CNN dealing with region proposal produced by RPN. R-FCN [9] design a fully convolutional region-based detector with almost all computation shared on the entire image. The position-sensitive score maps is thus proposed to balance translation-invariance for classification and translation-variance for detection. In addition, several architectures PVANET [10], HyperNet [11], FPN [12] MSCNN [16] analogous to Faster R-CNN framework are proposed to decrease the runtime and improve the performance.

Faster R-CNN framework follows the pipeline of "CNN feature extraction plus region proposal and ROI classification". Instead of proposal generation and feature resampling stages, YOLO [13] uses a single convolutional network directly regress spatially separated bounding boxes and associated class probabilities. J. Redmon.et.al divide the entire image into a 7x7 grid, each grid cell is responsible for 2 bounding boxes and confidence scores. However, it's difficult for coarse grid to achieve satisfied results when facing multi-scale object. SSD [14] alleviated this problem by using multiple scale default boxes on multiple layers to regress and predict object, which can improve accuracy for PASCAL VOC from 63.4% mAP for YOLO to 74.3% mAP. Compared to Faster R-CNN, SSD achieves a significant improvement in speed and accuracy from 7FPS with mAP 73.2% to 59FPS with mAP 74.3% on Titan GPU.

3. SINGLENET FRAMEWORK

This section mainly describes the proposed SingleNet in detail.

3.1. SingleNet Pipeline

Our SingleNet framework is illustrated in Fig. 1. From Fig. 1, we can see that the fully convolutional network is employed to generate base network feature. Hierarchical feature maps are then fused by un-pooling and element-wise sum operation. At last, bounding box detection and classification are directly implemented in different fusion feature map location using different scales and aspect rations dense boxes. Specifically, we choose VGG16 as our truncated network and use convolutional layers in fc6 and fc7. Several extra convolutional layers are also added to the end of the truncated network for constructing our fully convolutional network. This fully convolutional network is considered as base network, which is used to extract the low level feature and the high level one. Then, we introduce the fusion network to fuse the feature maps with different size from up to bottom. The Fusion network is composed of convolutional layer, L2-normalization layer, up-sample layer and element-wise sum layer. Given an image with arbitrary size, our method can obtain several multi-scale fusion feature maps. Finally, we adopt a set of convolutional filters on these fusion feature maps to directly produce a fixed set of detection predictions containing rectangular bounding box and corresponding confidence.

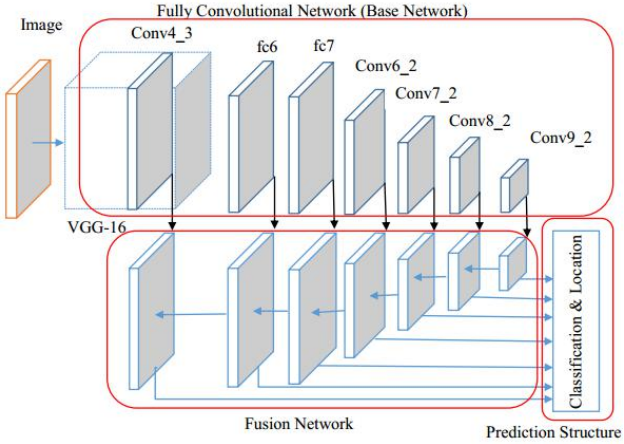

Fig.1. The pipeline of SingleNet framework consisting of base network, fusion network and prediction structure.

3.2. Feature Fusion

Here, we apply the convolutional layers of base network to achieve feature maps. These feature maps has different resolution and semantics. Instead of using these feature maps to regress bounding boxes directly and associated class confidences, we propose a fusion network to fuse these feature maps from top to down. The top feature maps with strong semantic information are up-sampled to match the scale of down feature maps with higher resolution. This scheme will enhance the local feature representation power and make it contains more global context. Inspired by neuroscience, we believe that the feature map quality is improved by above fusion method and can help to improve the detection performance.

Fig.2 shows the fusion block can generate top-down feature map. Our fusion strategy is carried out uniformly for different layers. Here, we just use a 1x1 convolutional kernel to normalize feature maps and make them has same channel dimension. Then, we un-pool and crop the top feature map to match the down feature map. Finally, these two feature maps are merged by element-wise sum. The

fusion feature map is also un-pooled and cropped for next down feature map and the process is iterated.

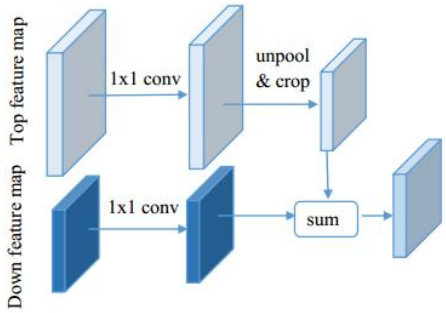

Fig.2. Fusion block illustrating the two top and down feature maps fusion by convolution, unpool&crop and element-wise sum.

3.3. Prediction structure

In this study, we introduce prediction structure in the fusion network to complete the prediction task in one stage. Like SSD [14], we design a set of different aspect ratios of dense boxes for each pixel in each feature map to discretize the space of possible output box shapes. Fig. 3 shows that prediction architecture uses convolutional filter and flatten operation to produce confidence and corresponding bounding box offset. Bounding box are regressed from dense box to a nearby ground-truth box.

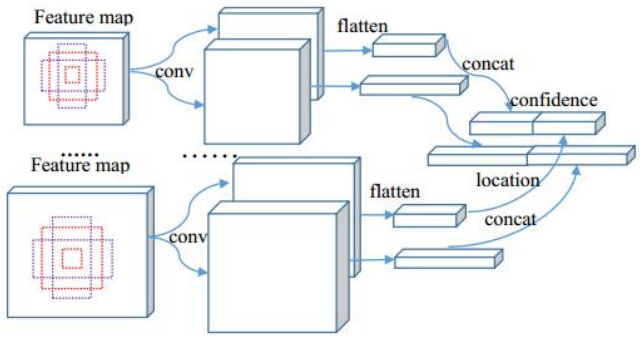

Fig.3. Prediction structure predicting confidence and bounding box for each pixel in each feature map.

3.4. Training methodology

Loss function

In this study, we employ the multi-task loss [7] [8] to describe the errors since it can help to complete the training task in a single stage. Our multi-task loss function is defined as:

$$L(\{p_i\},\{p_i^*\},\{t_i\},\{t_i^*\}) = \frac{1}{N}\left(\sum_i L_{conf}(p_i, p_i^*) + \alpha \sum_i p^* L_{loc}(t_i, t_i^*)\right)$$

Where N is the number of matched dense boxes, L_{conf} is the softmax loss over multiple classes, and L_{loc} is the smooth L1 loss between parameterizations of predicted box and ground-truth box using dense box. $p_i^* = \{0,1\}$, and when $p_i^* = 1$, the localization loss is activated. α is a balancing weight.

Dense box design

Similar to SSD [14], the scale of the dense boxes for each feature map is computed as:

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m-1}(k-1), k \in [1, m]$$

Here, m is the number of feature maps. $s_{\min}$ is set to be 0.2 and $s_{\max}$ is set to be 0.9, the aspect ratios of each pixel in each feature map is $\{\frac{1}{3}, \frac{1}{2}, 1, 2, 3\}$.

Training

We firstly pre-train VGG16 on the imagenet (1000-class competition) dataset [18], then convert the model to perform detection by modifying fc6 and fc7 to convolutional layers and removing all dropout layers and fc8 layer. Finally, we add extra convolutional layers to the end of modified VGG16 to construct a fully convolutional network. For comparison with other frameworks, we set the input image 300x300 as the architecture shows in Figure 1, we use conv4_3, fc6, fc7, conv6_2, conv7_2, conv8_2, conv9_2 to fusion future. We set batch size 64, momentum 0.9, weight decay 0.0005 and train our end-to-end network using SGD with learning rate 0.001. The code is built on Caffe [19] and is made publicly available at https://github.com/lijiannuist/SingleNet.

4. EXPERIMENTAL EVALUATION

We evaluated the proposed method on PASCAL VOC [20]. The images from the union set of VOC2012 trainval and VOC2007 trainval (16551 images) are used for training. VOC2007 test set (4952 images) are used for testing. Table 1 shows that our SingleNet obtains better results and surpasses Faster R-CNN 4.4% mAP and SSD 3.3% mAP respectively. For inference time, SingleNet can run at 12 FPS on K80 GPU, while SSD can run at 13 FPS a little quickly without feature fusion network.

Additionally, we adopt the detection analysis tool [21] to analyze the performance of our SingleNet from the perspective of different object characteristics. We consider six object characteristics including occlusion, truncation, area size, aspect ratio, visibility of parts, viewpoint and examine their effects across several categories. Fig. 4 shows that SingleNet is robust to truncation, aspect ratio, visibility of parts and viewpoint, owning to the deep convolutional network and multi-scale dense box. Bounding box size and occlusion are still important characteristics. However, feature fusion in our SingleNet framework can enhance the

feature representation ability to alleviate the impact of these two types. In Fig.5, We illustrate that SingleNet gets high recall on various object categories and prediction accuracy is also described by large white area. Without region proposal network, our SingleNet can still regress the high-quality object shape in one stage and make less localization error, which is explained by the proportion of loc area. Compared to SSD [14], a small proportion for Sim area shows that feature fusion in SingleNet improves the feature discriminative power.

5. CONCLUSION

In this paper, we present a unified single fully convolutional network named SingleNet for object detection. Feature fusion network combined high-resolution features of low layer and semantic information of high layer together as the augment feature. In addition, a set of dense boxes over different aspect ratios and scales per feature map location are used to directly regress bounding box. The proposed SingleNet can obtain 0.776 mAP on VOC 2007 and run 12FPS on K80 GPU.

6. ACKNOWLEDGEMENT

This work was partially supported by the National Science Fund under Grant Nos.61502235, 91420201 and 61472187. The Key Project of Chinese Ministry of Education under Grant No.313030, the 973 Program No.2014CB349303, Fundamental Research Funds for the Central Universities No.30920140121005, and Program for Changjiang Scholars and Innovative Research Team in University No.IRT13072.

Method	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Faster rcnn	**73.2**	**76.5**	**79.0**	**70.9**	**65.5**	**52.1**	**83.1**	**84.7**	**86.4**	**52.0**	**81.9**	**65.7**	**84.8**	**84.6**	**77.5**	**76.7**	**38.8**	**73.6**	**73.9**	**83.0**	**72.6**
SSD	**74.3**	**75.5**	**80.2**	**72.3**	**66.3**	**47.6**	**83.0**	**84.2**	**86.1**	**54.7**	**78.3**	**73.9**	**84.5**	**85.3**	**82.6**	**76.2**	**48.6**	**73.9**	**76.0**	**83.4**	**74.0**
SingleNet	**77.6**	**79.2**	**85.7**	**75.9**	**68.9**	**51.2**	**85.9**	**85.3**	**86.5**	**62.6**	**83.6**	**76.3**	**86.6**	**87.1**	**86.3**	**78.4**	**53.9**	**78.1**	**77.3**	**87.4**	**77.3**

Table1. PASCAL VOC 2007 test detection results. The minimum dimension of input images of faster rcnn is 600. SSD and SingleNet takes 300x300 images as input. The train data is the union set of VOC 2007 trainval and VOC 2012 trainval.

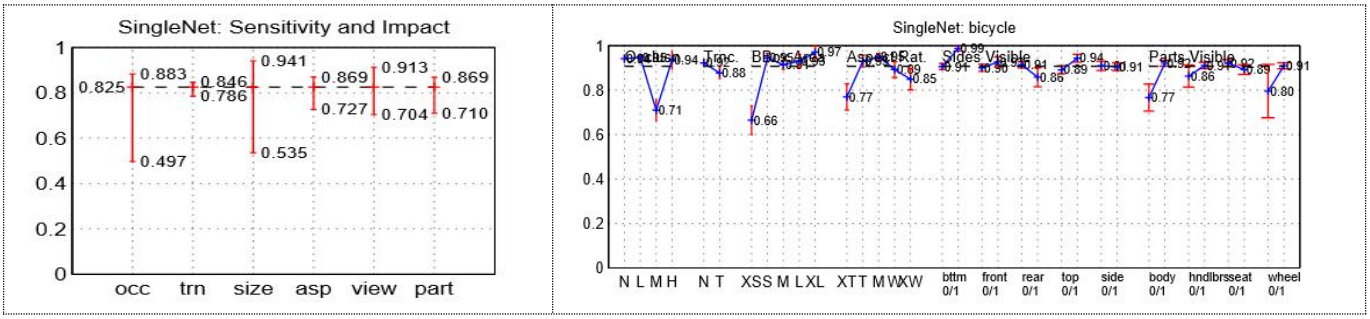

Fig4. Sensitivity and Impact of Object Characteristics of occlusion, Truncation, area size, aspect ratio, visibility of parts and viewpoint. The left figure show the average AP performance of the highest performing and lowest performing subsets within each characteristic. The right figure shows the analysis of characteristics for bicycle category. AP ('+') with standard error bars (red). Black dashed lines is overall AP.

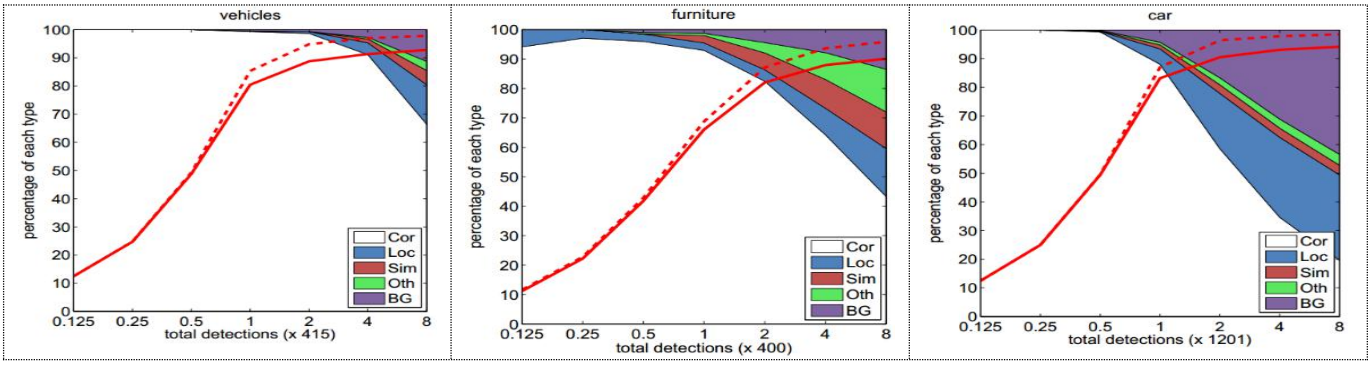

Fig.5 The performances of SingleNet on vehicles, furniture, and car. The solid red line in first row reflects the relation between recall with strong criteria (0.5 jaccard overlap) and the number of detections. The dash red line uses the weak criteria (0.1 jaccard overlap). The cumulative fraction is also showed for correct detection (Cor) or false positive like poor location (Loc), confusion with similar categories (Sim), with others (Oth), or with background (BG).

6. REFERENCES

[1] D. Lowe, "Distinctive image features from scale-invariant keypoints," in *IJCV*, 2004.

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

[3] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatiely trained part based models," in *TPAMI*, 2010.

[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. V. d. 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-H Kim, S. Hong, B. Roh, Y. Cheon, and M. Park, "PVANET: Deep but Lightweight Neural Networks for Real-time Object Detection," arXiv preprint arXiv:1608.08021.

[11] T. Kong, A. Yao, Y. Chen, F. Sun, "HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection," arXiv preprint arXiv:1604.00600.

[12] Y Lin, P. Dollar, R. Girshick, K. He, B. Hariharan , and S. Belongie, "Feature Pyramid networks for Object Detection," arXiv preprint arXiv:1612.03144.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," arXiv preprint arXiv:1506.02640, 2015.

[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot multibox detector," arXiv preprint arXiv:1512.02325, 2015.

[15] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," In *CVPR*, 2015.

[16] Z. Cai, Q. Fan, R. S. F and N. Vasconcelos, "A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection," arXiv preprint arXiv:1607.07155.

[17] Simonyan, K. Zisserman, "A Very deep convolutional networks for large-scale image recognition," CoRR abs/1409.1556 2014.

[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," in *IJCV*, 2015.

[19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," In: MM, ACM 2014.

[20] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) Challenge," in *IJCV*, 2010.

[21] Hoiem, D. Chodpathumwan, Y. Dai, Q. Dai, "Diagnosing error in object detectors," in *ECCV* 2012.

[22] W. Liu, A. Rabinovich, A. C. Berg, "ParseNet: Looking wider to see better," in *ICLR* 2016.

[23] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A.W. Smeulders, "Selective search for object recognition," in *IJCV*, 2013.