Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks
Authors: Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun

Introduction

The ability to hypothesize the location of objects in an image has led to an improvement in the object detection networks. While algorithms like Fast R-CNN help in improving the running time of these networks, the authors propose *Region Proposal Netowrks (RPNs)*, for which the cost for computing is very small. The Fast R-CNN algorithm uses convolutional feature maps for generating region proposals. The authors propose using these feature maps towards regressing region bounds for proposing the regions. The authors build a network unifying the RPNs with Fast R-CNN, referred to as *Faster R-CNN* by building a training scheme which alternates between fine-tuning for object detection and fine-tuning for region proposal. The deep layers in the more recent object detection algorithms helps the RPNs detect the regions better, thus helping improve object detection accuracy, apart from being a cost-effective alternative.

Network Architecture of Faster R-CNN

R-CNN and Fast R-CNN generate region proposals by using the selective search (SS) technique in the RPNs. The authors in this work replace the RPN using selective search with RPNs using CNN, wherein the CNN is shared with the detection network. The authors experiment with both ZFNet and VGGNet for the CNN in the RPNs. The image in the network goes through a series of convolution layers, from which a feature map is extracted, which is further subjected to a sliding window. From each location in the feature map, anchor boxes of three scales (128, 256, 512) and three aspect ratios (1:1, 1:2, 2:1) are used to generate the region proposals. Thus, the RPN network helps check which of the locations contains an object, and these locations are further passed to the detection network for detecting the class of the object in that location.

The detection part in the Faster R-CNN is the same as that in Fast R-CNN. The locations of the regions of interest that are passed from the RPNs are pooled, and the pooled area goes through a CNN and two fully convolutional layers. The interesting part in this network is that the convolutional layers are shared to extract the feature maps, and for the classification of objects. To enable this, the authors use a pragmatic 4-step alternation training technique. In the first step, a pre-trained ImageNet model is fine-tuned for the region proposal task. In the next step, a separate detection network by Fast R-CNN is trained using the proposals spitted out by the previous step. In the third step, to initialize the RPN training, the convolutional layers are fixed and only the layers unique to the RPN are fine-tuned. Finally, keeping the latter layers fixed, the convolutional layers are fine-tuned. Furthermore, it is observed that few RPN proposals highly overlap with each other. To reduce this redundancy, non-maximum suppression (NMS) technique is adopted on the proposal regions. Overall, the authors train Fast R-CNN using a set of RPN proposals, and evaluate different number of proposals at test-time.

Experimentation and Results

The authors of this work did a wide range of experiments to check the performance of Faster R-CNN technique. They evaluate their performance on the PASCAL VOC 2007 dataset which has about 5k train-val images and 5k test images spread over 20 categories. They also tested on the ImageNet pre-trained network, and the VGG-16 model. They evaluate the performance of the Faster R-CNN techniques in these models by using the detection mean Average Precision (mAP) metric. RPN with Fast R-CNN outperforms the Fast R-CNN framework on both Selective Search and EdgeBoxes.

TO investigate the performance of RPNs for region proposal, the authors studied the effect of sharing convolutional layers between RPN and Fast R-CNN detection network. To do this, the authors compare the performance of a Fast R-CNN trained using 2000 SS proposals and ZF net, with the same trained using 300 RPN proposals. While the former has an mAP of 58.7%. the latter achieves an mAP of 56.7%. The authors attribute this loss in performance to the inconsistency in the training and testing proposals. A comparison of results from various combinations of methods and number of region proposals at test-time are recorded in Table 2.

Next, the authors record the performance of RPNs with the VGG-16 model. They record the performance of VGG-16 for both proposal and detection in Table 3. It is observed that this technique outperforms *Selective Search* on both unshared and shared-features variants, with nearly cost-free test set region proposals. In Table 6, the authors record the comparison of results on the PASCAL VOC 2007, and combination of PASCAL VOC 2007 data with PASCAL VOC 2012 data with the Fast R-CNN detector. In place of the detection accuracy, the authors use the Recall-to-IOU technique, which is more appropriate for the diagnosis.

The authors also record experiments on the Microsoft COCO object detection dataset, which includes 80k images in the training set, 20k in the test-dev set, spread over 80 categories. The mAP is averages for IoU ratios of 0.5, 0.05, 0.95, the results of which are recorded in Table 11. It is observed that the Faster R-CNN algorithm outperforms the Fast R-CNN baseline, and the RPN improves the location accuracy. The authors state that the model trained on COCO+VOC set takes about 200ms per image for the region localization.

Strengths & Weaknesses

One of the major strengths of this work is that the RPN network using CNN which was proposed enables the object detection system to run at near real-time frame rates, and that too, at a higher overall object detection accuracy. One of the weaknesses of the RoI pooling layer of Faster R-CNN is that it only uses feature maps of the top convolution layer, which may lead to the failure of feature extraction in low resolution conditions. Also, the confidence score used in Faster R-CNN only represents the same class of objects, but not the same object. [1]

Further Work

Since the introduction of Faster R-CNN, more novel algorithms/models like "fully convolutional networks"[2], YOLO [3] have been introduced which frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities. YOLO, in particular, can be optimized end-to-end since it is a single, unified network. They have proven to outperform R-CNN algorithms.

References

[1] Li, Hailiang Huang, Yongqian Zhang, Zhijun. (2017). An Improved Faster R-CNN for Same Object Retrieval. IEEE Access. PP. 1-1. 10.1109/ACCESS.2017.2729943.

[2] Jonathan Long, Evan Shelhamer, Trevor Darrell;*Fully Convolutional Networks for Semantic Segmentation*; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431-3440

[3] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi; *You Only Look Once: Unified, Real-Time Object Detection*; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788