DC-YOLOv3: A novel efficient object detection algorithm

Xu Zhang*, Fang Han, Ping Wang, Wei Jiang, Chen Wang

Science and education center of big data and computer application, Huanghe University of Science and Technology, Zhengzhou, Henan Province, 450000, China

*E-mail: zhangxu_01@hotmail.com

Abstract. Feature pyramids have become an essential component in most modern object detectors, such as Mask RCNN, YOLOv3, RetinaNet. In these detectors, the pyramidal feature representations are commonly used which represent an image with multi-scale feature layers. However, the detectors can’t be used in many real world applications which require real time performance under a computationally limited circumstance. In the paper, we study network architecture in YOLOv3 and modify the classical backbone--darknet53 of YOLOv3 by using a group of convolutions and dilated convolutions (DC). Then, a novel one-stage object detection network framework called DC-YOLOv3 is proposed. A lot of experiments on the Pascal 2017 benchmark prove the effectiveness of our framework. The results illustrate that DC-YOLOv3 achieves comparable results with YOLOv3 while being about 1.32× faster in training time and 1.38× faster in inference time.

1. Introduction

In recent years, object detection has become a focus issue in computer vision, object detection has its own challenge to detect and localize lots of objects in different scales and different locations. Object detection has made a great progress owing to the success of convolutional neural networks (CNN). Many CNN[1-2] based object detection framework and their variants[3-5], have been proposed and remarkably promote the speed and the accuracy for object detection. These developments also accelerate the application of object detection in many fields, including surveillance, autonomous driving, robotics, pedestrian detection, etc.

A typical family of object detection based on deep networks can be categorized into one-stage object detection and two-stage object detection. Generally, two-stage object detection can acquire the highest accuracy, which first generates a limited number of proposal regions using searching algorithms, such as selective search[6], CPMC[7] and RPN[4]. Typical works include R-CNN[1], Fast R-CNN[3], Faster R-CNN[4,8], R-FCN[5], Mask RCNN[9] and Cascade R-CNN[10]. On the contrary, one-stage methods are considered to be simpler and more efficient which predict the object class from the pre-designed candidate boxes. Many recent works including YOLO series[11-14], SSD[15-16], DSOD[17], RetinaNet[18] and CornerNet[19] have been proposed.

In modern object detectors, feature pyramids have both become an essential component for detecting and localizing multiple objects across a wide range of scales and locations. The pyramidal feature representations are commonly used which represent an image with multi-scale feature layers, such as FPN[20], Mask RCNN, YOLOv3[13], RetinaNet[18].

More recently, YOLOF[21] is presented, which studies the influence of FPN’s two benefits in one-stage detectors and considers FPN as a Multiple-in-Multiple-out (MiMo) encoder. Inspired by this, in the paper, based YOLOv3 we present a novel one-stage network structure called DC-YOLOv3.
network. In our network, we study the role of FPN in YOLOv3 and modify the FPN structure of the backbone by using a group of convolutions and dilated convolutions. Many experiments on the Pascal 2017 benchmark prove the effectiveness of our network. Our DC-YOLOv3 achieves comparable results with YOLOv3 while being about 1.32× faster in training time and 1.38× faster in inference time.

2. Related work

2.1. Feature presentation in multiscale

It is always a challenge to detect and localize lots of objects in different scales and different locations. Many works have studied to address the issue. Faster R-CNN and SSD use multiple feature maps at different resolutions to fit various scales objects. FPN builds a feature pyramid by multi-scale features summation. Based on FPN, [22] proposes an additional bottom-up pathway, [23] studies to build stronger feature pyramidal representations by using multiple U-shape modules after a backbone model. [24] presents cascaded pyramid network which can learn the pyramidal feature representation with sufficient context information. [25] proposes to combine features at all scales and generate features at each scale by a global attention operation. [26] proposes “Gated CNN” to introduce a “gate” structure injected by multi-scale feature layers to integrate multiple convolutional layers for object detection. [27] uses dilated convolutions to systematically aggregate multi-scale contextual information without losing resolution. In this paper, we investigate to use dilated convolutions to aggregate multiple different scale feature maps.

2.2. YOLOv3

YOLO series are one-stage detectors, YOLO[11] and YOLOv2[12] only use the last output feature of the backbone. The accuracy can’t reach the demand of some application. More specifically, in YOLOv2 Resnet is used, in Faster R-CNN RPN is used for reference, and in YOLOv3 darknet53[28] is used. Owing to the more depth and the more successful design of darknet53, compare with YOLO and YOLOv2, YOLOv3 is not only deeper but also semantically strong and more accurate.

Despite all this, under a computationally limited non-gpu circumstance, YOLOv3 can’t be used in many real world applications which require real time performance. The main reason is the larger and deeper architecture of darknet53, which includes 53 convolutional layers and multiple residual blocks, with heavy computational cost. In this paper, we will investigate darknet53 and modify it to fit many real world applications.

3. Architecture of DC-YOLOv3

Our method is similar to YOLOv3 while being faster and more efficient. While other works [29] focused on optimization technique or strategy, our method focuses on modify and optimize the backbone to shrink the size of the network without the degradation of the whole performance. Our network is one-stage, which also has two main components: a novel backbone network (often state-of-the-art feature extraction network) and a classification-regression network. The former is modified from darknet53 with dilated convolution blocks, and the latter is same to YOLOv3. Figure 1 illustrates the architecture of darknet53, the designed backbone and DC blocks.

3.1. Design of our novel backbone network

Figure 1(a) illustrates that darknet53 consists of multiple different feature layers, which can extract features from those scales using a similar concept to feature pyramid network. In YOLOv3, the author has demonstrated the strong power of darknet53 by extensive experiments—the features extracted from darknet53 have exhibited super performance. The last three feature layers of darknet53 are used for class prediction and bounding box regression in YOLOv3.

In non-gpu circumstance, YOLOv3 can’t support real time application mainly owing to the deeper and larger architecture of darknet53. Since only the last three features layers are used to predict small,
middle and large scale objects, and dilated convolution [30-32] can be used to enlarge the field of convolutional kernels, we can try to substitute dilated convolution with different dilations for the last several residual blocks.

In darknet53, the last but two layers extract fine-grained feature to predict small scale objects, and the layer is the cornerstone of the last two layers, so it keeps unchanged. The last two layers extract middle-grained and big-grained feature to predict middle and big objects, so we can substitute dilated convolution for them. However, dilated convolution poses gridding problems which lead to the completely missing of local information. As a result, the sample from the input can be sparser, which may not be good for feature extraction, and may degrade the system performance.

To solve the problem, we design two DC blocks (see figure 1(c)) in the framework. Take DC block1 for example, the output of the previous layer (256×52×52) is fed into two branches, one is a convolutional module, and the other is a dilated convolution following a convolutional module. Then concatenate the feature of two branches in channel dimension as the final output of DC block1. Similar with YOLOv3, the feature extracted from DC block1 will be fed into two branches, one is a concatenate block in DC-YOLOv3, and the other is DC block2.

In contrast to Darknet53, DC block1 involves less computation and is much simpler and more efficient. This will be verified in the subsequent experiments.

### 3.2. Extension of the architecture

The idea of our design can easily be extended to other object detection framework. We can substitute the DC blocks for multi scale feature extraction modules. The only thing we need to do is to fine-tune the hyper-parameters in the DC blocks, such as the value of dilation.
In addition, hybrid dilated convolution (HDC)[30] can alleviate the gridding problem in dilated convolution, so design a HDC module, and try to substitute HDC for the relevant layers is also a good idea.

4. Experiments on Pascal VOC 2007 dataset
In this section, we present evaluation results of the proposed method on the Pascal VOC 2007 [33]. In experiments, YOLOv3-darknet53 is used as the baseline method. The training and testing experiments were conducted both on a thinkPad t470 laptop (limited by the computation resource).

In object detection experiments, to accelerate the training speed and get the desired performance, we adopt a pretrained model on MS COCO[34]. Since darknet53 and our backbone network have the same first three layers as shown in Figure 1, we only use the parameters of the first three layers to initialize the corresponding parameters in the mode we will train.

For the Pascal VOC 2007 train set and test set, we use the 5k trainval images and the 5k test images in VOC 2007. The partial hyper-parameters are as follows: the initial learning rate and weight decay are respectively set as 0.001 and 0.92, IoU threshold for assigning ground truth is 0.3. The batch size of training phase is 8 and Adam optimization strategy is adopted.

**Training phase.** We train two networks in the same way, freezing the parameters of the first three layers. From Table 1, contrast to YOLOv3, training time of one epoch and param size in our method are both much fewer. This saves the memory and computation resource obviously. Figure 2 shows the total loss variation with the training time. This indicates that they may have a comparative converge speed and accuracy.

| Methods     | Training time of one epoch(h) | Inference time per image(sec) | param size(M) |
|-------------|--------------------------------|-------------------------------|---------------|
| YOLOv3      | 2.32                           | 1.15                          | 241.087       |
| Our Method  | 1.75                           | 0.83                          | 129.457       |

**Test phase.** We test the two trained network model on 5k test images in Pascal VOC 2007. Our method is slightly inferior to YOLOv3 in accuracy. After about 25 training epochs in the experiments, the mAP of our designed model is 45.21%, while YOLOv3’s is 47.07%. However, from table 1, our method has the obvious advantage in speed. It can be used in the real world applications which require real time performance on a non-gpu computer. Figure 3 shows some object detection results of our method from the test images in Pascal VOC 2007.

5. Conclusion
DC-YOLOv3 achieved its desired goal, which made object detection task work on the non-gpu computers. In addition, DC-YOLOv3 offers the following contributions to the field of object detection.
detection. First, it provides an idea to simplify a multi-scale backbone and easily extends to some network modules. Second, to increase the accuracy the DC-YOLOv3, one can substitute hybrid dilated or well-designed combinatorial dilated convolution for dilated convolution.

Acknowledgments

The work was supported by construction of computer science and technology brand speciality of private colleges of Henan province(ZLG201903), teaching reform and curriculum development project of Huanghe University of Science and Technology(kg2018xm), school level excellent primary-level teaching organization of Huanghe University of Science and Technology (JXZZ201907).

References

[1] Girshick, R., Donahue, J., Darrell, T., Malik, J. (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Computer Vision and Pattern Recognition. Columbia. pp. 580–587.

[2] Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012) Imagenet classification with deep convolutional neural networks. In: advances in Neural Information Processing Systems. Nevada. pp. 1097-1105.

[3] Girshick, R. (2015) Fast r-cnn. In: International Conference on Computer Vision. Santiago. pp. 1440–1448.

[4] Ren, S., He, K., Girshick, R., Sun J. (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. In: advances in Neural Information Processing Systems. Montreal. pp. 91-99.

[5] Dai, J., Li, Y., He, K., Sun, J. (2016) R-FCN: Object detection via region-based fully convolutional networks, In: Advances in Neural Information Processing Systems. Barcelona. pp. 379-387.

[6] Uijlings, J. R., Van De Sande, K. E., Gevers, T., Smeulders, A. W. (2013) Selective search for object recognition, International Journal of Computer Vision, vol. 104, no. 2, pp. 154–171.

[7] Carreira, J., Sminchisescu, C. (2012) Cpmc: Automatic object segmentation using constrained parametric min-cuts, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 34, no. 7, pp. 1312–1328.

[8] Zhang, L., Liang, L., Liang. X, et al. (2016) Is Faster R-CNN Doing Well for Pedestrian Detection? In: European Conference on Computer Vision. Amsterdam.

[9] He,K., Gkioxari,G., Dollar, P., Girshick, R. (2017) Mask r-cnn. In: International Conference on Computer Vision. Venice. pp. 2980–2988.

[10] Cai Z.W., Vasconcelos, Nuno. (2018) Cascade R-CNN: Delving into High Quality Object Detection, In: Computer Vision and Pattern Recognition. Utah. pp. 6154–6162.

[11] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016) You only look once: Unified, real-time object detection. In: Computer Vision and Pattern Recognition. Las Vegas. pp. 779–788.

[12] Redmon. J., Farhadi, A. (2017) Yolo9000: better, faster, stronger. In: Computer Vision and Pattern Recognition. Hawaii. pp. 6517-6525

[13] Redmon, J., Farhadi, A. (2018) YOLOv3: An incremental improvement, arXiv preprint arXiv:1804.02767.

[14] Huang, R., Pedoeem, J., Chen, C. (2018) YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers. In: International Conference on Big Data. Seattle.

[15] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., et al.(2016) SSD: single shot multibox detector. In: European Conference on Computer Vision. Amsterdam.

[16] Zhao, Q., Sheng. T., Wang, Y., et al. (2019) M2Det: A Single-Shot Object Detector Based on Multi-Level Feature Pyramid Network. In: Association for Advancement of Artificial Intelligence, vol,33:9259-9266.
[17] Shen, Z., Liu, Z., Li, J., Jiang, Y-G., et al. (2017) DSOD: Learning deeply supervised object detectors from scratch. In: International Conference on Computer Vision, Venice. pp. 1919-1927.
[18] Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollar, P. (2017) Focal loss for dense object detection. In: International Conference on Computer Vision, Venice. pp. 1, 2, 3, 8
[19] Law, H., Deng, J. (2018) Cornernet: Detecting objects as paired keypoints. In: European Conference on Computer Vision. Munich. pp. 734–750.
[20] Lin, T.-Y., Dollar, Piotr., Girshick, Ross, He, K., et al. (2017) Feature pyramid networks for object detection. In: Computer Vision and Pattern Recognition, Hawaii. pp: 2117–2125.
[21] Chen, Q., Wang, Y., Yang, T., Zhang, X., et al. (2021) You Only Look One-level Feature. In: International Conference on Computer Vision. pp: 13039–13048.
[22] Liu, S., Qi, L., Qin, H., Shi, J., Jia, J. (2018) Path aggregation network for instance segmentation. In: Computer Vision and Pattern Recognition, Utah.
[23] Zhao, Q., Sheng, T., Wang, Y., et al. (2019) M2det: A single-shot object detector based on multi-level feature pyramid network. In: Association for Advancement of Artificial Intelligence, Hawaii, 33, pp.9259-9266
[24] Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J. (2018) Cascaded pyramid network for multi-person pose estimation. In: Computer Vision and Pattern Recognition. Utah.
[25] Kong, T., Sun, F., Huang, W., Liu, H. (2018) Deep feature pyramid reconfiguration for object detection. In: European Conference on Computer Vision. Utah.
[26] Yuan, J., Xiong, H-C., Xiao, Y., et al. (2019) Gated CNN: Integrating Multi-scale Feature Layers for Object Detection. Pattern Recognition, 105(6):107-131.
[27] Yu, F., Koltun, V. (2016) Multi-Scale Context Aggregation by Dilated Convolutions. In: International Conference on Learning Representations. Puerto Rico.
[28] Redmon, J. Darknet: Open source neural networks in c. http://pjreddie.com/darknet/, 2013-2016.
[29] Bochkovskiy, A., Wang, C-Y., Liao, H. (2020) YOLOv4: Optimal Speed and Accuracy of Object Detection. In: Computer Vision and Pattern Recognition. Virtual.
[30] Wang, P., Chen, P., Yuan, Y., et al. (2018) Understanding Convolution for Semantic Segmentation. In: Winter Conference on Applications of Computer Vision. California.
[31] Yu, F., Koltun, V. (2015) Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122.
[32] Chen, L-C., Papandreou, G., Kokkinos, I., Murphy, K., et al. (2016) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv preprint arXiv:1606.00915.
[33] Everingham, M., Eslami, S. A., Van Gool, L., et al. (2015) The pascal visual object classes challenge: A retrospective, International Journal of Computer Vision, vol. 111, no. 1, pp. 98–136.
[34] Lin, T-Y., Maire, M., Belongie, S. et al. (2014) Microsoft COCO: Common objects in context, In: European Conference on Computer Vision, pp. 740–755.