Improve YOLOv3 using dilated spatial pyramid module for multi-scale object detection

Xiaoguo Zhang, Ye Gao, Huiqing Wang and Qing Wang

Abstract
Effectively and efficiently recognizing multi-scale objects is one of the key challenges of utilizing deep convolutional neural network to the object detection field. YOLOv3 (You only look once v3) is the state-of-the-art object detector with good performance in both aspects of accuracy and speed; however, the scale variation is still the challenging problem which needs to be improved. Considering that the detection performances of multi-scale objects are related to the receptive fields of the network, in this work, we propose a novel dilated spatial pyramid module to integrate multi-scale information to effectively deal with scale variation problem. Firstly, the input of dilated spatial pyramid is fed into multiple parallel branches with different dilation rates to generate feature maps with different receptive fields. Then, the input of dilated spatial pyramid and outputs of different branches are concatenated to integrate multi-scale information. Moreover, dilated spatial pyramid is integrated with YOLOv3 in front of the first detection header to present dilated spatial pyramid-You only look once model. Experiment results on PASCAL VOC2007 demonstrate that dilated spatial pyramid-You only look once model outperforms other state-of-the-art methods in mean average precision, while it still keeps a satisfying real-time detection speed. For 416 × 416 input, dilated spatial pyramid-You only look once model achieves 82.2% mean average precision at 56 frames per second, 3.9% higher than YOLOv3 with only slight speed drops.

Keywords
Real-time object detection, YOLOv3, scale variation, dilated spatial pyramid, receptive fields

Date received: 22 September 2019; accepted: 14 May 2020

Introduction
Real-time multi-scale object detection is one of the most challenging tasks in computer vision. Generally, the traditional object detection algorithms are composed of three stages: select candidate regions on the given image, extract features from these regions, and finally classify each region with trained classifier. The performance of such kind of algorithms generally depends on the expression ability of features extracted by designers.

In recent years, with the development of big data technology and the improvement of computing performance, deep convolutional neural networks (DCNNs) have achieved significant advances in object detection. Existing object detection algorithms based on DCNN can be roughly divided into two categories: (1) two-stage methods, mainly including R-CNN, Fast R-CNN, Faster R-CNN and R-FCN and (2) one-stage methods, mainly including YOLOv3.
You only look once (YOLO)\textsuperscript{5} and Single Shot MultiBox Detector (SSD).\textsuperscript{6}

The two-stage methods firstly generate a series of region proposals and then do feature extraction with CNN for classification and bounding box regression. Although the two-stage methods have achieved competitive performance, they are too slow for real-time applications due to its intensive computation cost.

The one-stage methods consider object detection as a single regression problem and are able to realize real-time detection due to its high computational efficiency, while its accuracy is usually lower than that of those two-stage methods. YOLO\textsuperscript{5} and SSD\textsuperscript{6} are two representative one-stage methods. The YOLO algorithm has gone through three stages of development: (1) YOLO\textsuperscript{5} divides the input image into $s \times s$ grid cells, but each grid cell can only predict one kind of objects and, therefore, YOLO has difficulty on dense and small object detection; (2) YOLO\textsuperscript{2} improves the base network of YOLO and adopts anchor mechanism and multi-scale training method, consequently, both accuracy and speed have been improved; and (3) YOLO\textsuperscript{3} proposes a new network Darknet-53 and predicts objects at three different scales. Considering the algorithm performance, for $320 \times 320$ input, YOLO\textsuperscript{3} achieves 28.2\% mean average precision (mAP) in 22 ms on MS COCO data set, being as accurate as SSD but with a three-time faster speed.

Although YOLO\textsuperscript{3} has very good performance on small object detection with multi-scale predictions, it performs relatively worse on medium and larger size objects.\textsuperscript{8} Much has been done to improve the accuracy of the YOLO algorithm. In 2018, Li and Yang\textsuperscript{8} replaced the standard convolution with depth-wise separable convolution and introduced feature pyramid network (FPN)\textsuperscript{10} into the detection module in YOLO\textsubscript{2} to reduce parameters and improve feature extraction ability, and the improved YOLO\textsubscript{2} algorithm outperforms YOLO\textsubscript{2} on small object detection. In 2019, Zhang et al.\textsuperscript{11} added three residual blocks to the bottom of residual network of original YOLO\textsubscript{3} and designed six multi-scale convolutional feature maps for prediction to formulate DF-YOLO\textsubscript{3} model, thus improving the vehicles detection accuracy in complex scenes. To make full use of multi-scale feature maps, Huang and Wang\textsuperscript{12} proposed DC-spatial pyramid pooling (SPP)-YOLO model, which adopted dense connection and an improved SPP\textsuperscript{13} module to YOLO\textsubscript{2}. Xu et al.\textsuperscript{14} proposed Attention-YOLO, in which channel and spatial attention mechanism is added to the feature extraction network of YOLO\textsubscript{3}; as a result, Attention-YOLO achieves about 0.6\% AP\textsubscript{50} higher than YOLO\textsubscript{3} on MS COCO data set.

Although the above improved YOLO algorithms improve the accuracy, the inference speed decreases obviously at the same time, and recognizing objects at multiple scales is still the challenging problem for YOLO\textsubscript{3} which is waiting to be improved.

There are a variety of ways to remedy the scale variation problem for object detector. The direct but inefficient method is to extract features and predict different scales of images, such as image pyramid,\textsuperscript{15} which is widely used in hand-engineered feature-based methods.\textsuperscript{16,17} Instead of taking multiple scales of images as input, one kind of methods deals with scale variation problem by exploiting multiple layers in CNN. SSD\textsuperscript{6} used a pyramidal feature hierarchy structure to perform object detection at different layers and finally combined predictions from multiple feature maps with different resolutions to deal with objects of various sizes. FPN\textsuperscript{10} utilized a top-down architecture with lateral connections to build high-level semantic feature maps at all scales, which has shown significant improvements over several baselines. Feature Fusion Single Shot Multibox Detector (FSSD)\textsuperscript{18} concatenated features from different layers with different scales and then generated a new pyramidal feature hierarchy structure to predict objects by down-sampling the concatenation layer for several times; as a result, FSSD improved the accuracy significantly over SSD with only a slight decrease on speed. PANet\textsuperscript{19} added bottom-up path augmentation to the FPN, which shortened the information path between low-level feature maps and high-level feature maps to enhance the feature fusion structure. Instead of conducting feature fusion between different layers with different resolutions, some other methods resample the feature maps at the same scale to obtain multi-scale information. He et al.\textsuperscript{13} proposed a feature extraction module named SPP, which resampled the feature maps by using different pooling rates, and test results showed that the objects at different scales can be accurately classified by resampling the same feature maps. Liu et al.\textsuperscript{20} proposed a receptive filed block (RFB) module consisting of multi-branch convolution layers with different dilation rates and kernels, and by simply replacing the top convolution layers of SSD with RFB, the performance get significantly improved. Sachin Mehta et al.\textsuperscript{21} proposed an efficient spatial pyramid consisting of spatial pyramid of dilated convolutions and point-wise convolutions for semantic segmentation.

YOLO\textsubscript{3} adopts the method that exploiting multiple layers for detection and thus the performance of small object detection is improved, but it still performs relatively worse on medium and larger objects. Inspired by the method that resampling the feature maps at the same scale, a novel dilated spatial pyramid (DSP) module is proposed in this article to integrate multi-scale information by resampling the feature maps to deal with scale variation problem.

The main contribution of this article is as follows:

1. A novel DSP module consisting of multiple parallel branches with different dilation rates is proposed to learn multi-scale information from large effective fields to deal with scale variation problem.
2. DSP-YOLO is presented by integrating DSP module to YOLOv3 in front of the first detection header and it gains significant improvement in accuracy, while it is still able to maintain comparable speed with YOLOv3.

3. Experiments are designed to explore the optimal number of branches and fusion method of DSP and evaluate the performance among DSP-YOLO and other state-of-the-art object detectors.

The experiment results demonstrate that our DSP-YOLO model outperforms other state-of-the-art object detection methods in accuracy on PASCAL VOC2007, while it is still able to keep a satisfying real-time detection speed.

**Proposed methods**

The structure of original YOLOv3 is shown in Figure 1. Through extracting features from three different scales and combing low-resolution and high-resolution features via a top-down pathway and lateral connections, YOLOv3 improves the performance of small object detection ability, but it still performs relatively worse on medium and large size objects.8

Theoretically, the detection performance on objects with different scales is influenced by the effective receptive field of the network. Large effective receptive field could be used to enhance the performance of larger scale object detection, but it compromises the performance on small objects at the same time.22 Thus, exploiting multi-scale
receptive fields is one of the ways to improve the ability of multi-scale object detection.

Large effective receptive fields can be achieved by conducting down-sample operation, increasing convolutional layers, applying dilated convolution (atrous convolution), and enlarging the size of the convolutional kernel. Suppose the down-sample rate is $s$, then the receptive field is increased by $s$ times, while the resolution of the feature map is reduced by $s$ times, which may result in information loss. Increasing both convolutional layers and the kernel size of convolutional filters increases the receptive field at the cost of increasing the number of parameters and deep network may cause the problem of vanishing gradient during training process. Dilated convolution expands the convolutional kernel by inserting holes between each pixel, enlarging the effective receptive filed without additional parameters and computations cost. As shown in Figure 2, dilation rate $m$ means inserting $m - 1$ holes between each pixel and the dilated convolution is equal to standard convolution when dilation rate is set as 1.

Suppose the kernel size of the dilated convolution is $k$, dilation rate is $r$, then the actual kernel size $k'$ is

$$k' = k + (k - 1)(r - 1) = kr - r + 1$$

Suppose stride is 1, a $k \times k$ dilated convolution can obtain the same receptive fields as the standard convolution with kernel size of $(kr - r + 1)$.

Dilated convolutional layers with different dilation rates will generate different receptive fields, so the choice of dilation rate depends on the scale of the object to be measured. To deal with scale variation problem, the receptive fields are expected to balance between small- and large-scale objects. Inspired by SPP, a novel DSP module consisting of multiple parallel branches with different dilation rates is proposed in this article. As shown in Figure 3, in DSP structure, the multiple parallel branches resample the convolutional feature maps using a $3 \times 3$ convolution with different dilation rates simultaneously to generate feature maps with different receptive fields. The $k$th branch uses dilation rate $k$ and the padding parameter $p = 3(k - 1)/2$ to keep the resolution of the input feature map invariable after dilated convolution while increasing the receptive fields. Then, we proposed the following two methods to fuse feature maps with different receptive fields to obtain multi-scale information:

1. The first method is concatenation. As shown in Figure 3(a), the input of DSP $a$ module and outputs
of different dilated convolutional layers are concatenated in the channel dimension of feature maps to obtain richer semantic information.

2. The second method is element-wise addition operation. As shown in Figure 3(b), the outputs of different branches are concatenated in the channel dimension of feature maps. The following \(\frac{1}{C_2} \times C_1\) standard convolutional layer is used to reduce the number of the channel of feature maps, since element-wise addition is performed on two feature maps channel by channel, requiring the input and output are of the same dimensions. To improve the gradient flow inside the network, the element-wise addition is performed between the input of DSP \(b\) module and output of \(\frac{1}{C_2} \times C_1\) standard convolutional layer.

As shown in Figure 4(a), the DSP \(a\) module is integrated with YOLOv3 after the third convolutional layer in the first convolutional set (the gray block in Figure 1) and \(1 \times 1\) standard convolutional layer is added following DSP \(a\) module to reduce dimensions, leaving the \(3 \times 3\) convolutional layer with smaller input dimensions to reduce computation. The DSP \(b\) module is integrated with YOLOv3 between the third and fourth convolutional layers in the first convolutional set (the gray block in Figure 1) to present DSP-YOLO \(b\) model. The corresponding structure is shown in Figure 4(b).

Experiments

The performance of the DSP-YOLO is compared with state-of-the-art object detection methods on PASCAL VOC 2007 data set, which has 20 categories, and mAP and frames per second (FPS) are used to evaluate the performance of our approach.

Regarding the experiment on VOC2007 test, the union of VOC2007 trainval and VOC2012 trainval is used as the training data and the VOC2007 test (4952 images) as the test data.

Experiment condition

The proposed model is trained on GPU, and the detailed experimental configurations are listed in Table 1.

Training strategy

The DSP-YOLO is based on Darknet53, which is pretrained on the ImageNet, and it is trained in a batch size of 64 on a Titan Xp GPU with a starting learning rate of \(10^{-3}\), which is divided by 10 at 40,000 and 45,000 iterations. The max training iteration is 60,000, the weight decay is 0.0005, and momentum is 0.9. The training strategies mostly follow YOLOv3, including multi-scale training, data augmentation, convolutional with anchor boxes, and loss function. For a fair comparison, the YOLOv3 and YOLOv3-spp are trained in the same way as that of DSP-YOLO.

Ablation study on VOC2007

The number of branches in DSP module. To find the optimal number of branches in DSP, DSP-YOLO \(a\) with different number of branches is evaluated on PASCAL VOC2007, and the performance evaluated with \(320 \times 320\) input is recorded in Table 2. According to the results in Table 2, with three branches, DSP-YOLO \(a\) achieves 79.8\%, 4.2\% higher than the original YOLOv3, demonstrating that the DSP structure indeed improves the performance of YOLOv3 though with slight speed drop. With five branches, DSP-YOLO \(a\) achieves 79.7\% mAP, 0.3\% lower than that with four branches, indicating that five branches do not bring further improvement over four branches. Considering the accuracy and speed, the branches of DSP module is selected as 4.

Fusion method of feature maps in DSP module. To evaluate the performances of different fusion methods of feature maps in DSP module, DSP-YOLO \(a\) and DSP-YOLO \(b\) are evaluated on PASCAL VOC2007, and the performances

![Figure 4. The structure of convolutional set before the first detection header of DSP-YOLO. (a) DSP \(a\) is integrated after the third convolutional layer in the first convolutional set and \(1 \times 1\) convolutional layer is added to reduce dimensions; (b) DSP \(b\) is integrated between the third and fourth convolutional layers in the first convolutional set. DSP: dilated spatial pyramid; YOLO: You only look once.](image)

| Table 1. Experimental configurations. |
|---------------------------------------|
| **Hardware and software** | **Profile** |
| CPU | Intel(R) Core(TM) i7-8700 CPU @ 3.20 GHz |
| GPU | NVIDIA Titan Xp |
| Deep learning framework | Darknet |
| Operating system | Ubuntu 16.04 |
evaluated with 320 × 320 input are recorded in Table 3. According to the results in Table 3, DSP-YOLO a achieves 80.0% mAP, little higher than that of DSP-YOLO b, indicating that concatenation is more effective for feature maps fusion in DSP module. Considering the accuracy and speed, the DSP-YOLO a with 4 branches is selected and called DSP-YOLO.

Experiments on PASCAL VOC 2007

Table 4 shows the PASCAL VOC2007 test detection results. Since Redmon and Farhadi did not present the results of the original YOLOv3 and YOLOv3-spp on PASCAL VOC 2007 test in this article,8 the YOLOv3 and YOLOv3-spp are trained in the same training strategies as DSP-YOLO based on this article to get the benchmark results.

With low-dimension input images (e.g. 320 × 320), DSP-YOLO achieves 80.0% mAP, 1.0% lower than that of DSSD513.26 However, with high-dimension input (e.g. 608 × 608), DSP-YOLO can produce 83.1% mAP, much better than other improved YOLO models, like DC-SPP-YOLO12 and other state-of-the-art methods like R-FCN,4 DSSD513,26 STDN513,27 and so on.

For 416 × 416 input, compared to original YOLOv3, DSP-YOLO improves detection performance in almost every category, improving the ability for multi-scale detection, which proves the effectiveness of the proposed DSP module.

Inference time

Table 5 presents the comparison test results among the proposed DSP-YOLO and the state-of-the-art object detection methods in mAP and FPS on PASCAL VOC2007 test. With NVIDIA Titan Xp and CUDA 9.1, DSP-YOLO
achieves 82.2% mAP at 56 FPS with input 416 × 416, being as accurate as RFB Net512 but much more faster. For high-dimension input (e.g. 608 × 608), DSP-YOLO achieves 83.1% mAP at 31 FPS, surpassing other state-of-the-art object detection models including two-stage and one-stage methods and other improved YOLOv3 algorithms, while it still maintains real-time detection speed.

### Table 5. Comparison of object detection methods on PASCAL VOC2007 test.

| Method       | Base network | mAP(%) | FPS | GPU   |
|--------------|--------------|--------|-----|-------|
| Faster R-CNN3| VGG16        | 73.2   | 7   | Titan X |
| R-FCN4       | ResNet-101   | 80.5   | 9   | Titan X |
| RefineDet32028| VGG16        | 80.0   | 40.3| Titan X |
| RefineDet51228| VGG16        | 81.8   | 24.1| Titan X |
| RFB Net30028 | VGG16        | 80.5   | 83  | Titan X |
| RFB Net51220 | VGG16        | 82.2   | 38  | Titan X |
| STDN51327   | DenseNet-169 | 80.9   | 28.6| Titan Xp |
| SSD3006     | VGG16        | 74.3   | 46  | Titan X |
| SSD5126     | VGG16        | 76.8   | 19  | Titan X |
| DSSD3216    | ResNet-101   | 78.6   | 9.5 | Titan X |
| DSSD51326   | ResNet-101   | 81.5   | 5.5 | Titan X |
| YOLOv2 416  | Darknet19    | 76.8   | 67  | Titan X |
| YOLOv3 416  | Darknet53    | 78.3   | 57.7| Titan Xp |
| YOLOv3-spp 416| Darknet53    | 78.4   | 56.7| Titan Xp |
| DC-SPP-YOLO 41612| Darknet19 | 78.4   | 56.3| GTX 1080 Ti |
| DC-SPP-YOLO 54412| Darknet19 | 79.6   | 38.9| GTX 1080 Ti |
| Attention-YOLO-B14| Darknet53 | 81.9   | 26  | Titan X |
| DSP-YOLO 416| Darknet53    | 82.2   | 56  | Titan Xp |
| DSP-YOLO 544| Darknet53    | 82.8   | 36  | Titan Xp |
| DSP-YOLO 608| Darknet53    | 83.1   | 31  | Titan Xp |

| mAP: mean average precision; FPS: frames per second; DSP: dilated spatial pyramid; YOLO: You only look once; SPP: spatial pyramid pooling; SSD: Single Shot MultiBox Detector. |

### Visualization

In Figure 5, we show several detection examples of YOLOv3 and DSP-YOLO on PASCAL VOC2007 test. Compared to original YOLOv3, the proposed DSP-YOLO model improves in two scenarios. The first scenario is for detection on multi-scale objects. YOLOv3 has relatively poor performance on medium and larger object detection and has the problem of false detection (the false detection “train” and “aeroplane” in Figure 5(a)), repeated detection (Figure 5(c)) and missed detection (Figure 5(c), (e), and (g)), but DSP-YOLO shows significant improvement, as shown in Figure 5(b), (d), (f), and (h). And comparing Figure 5(i) with Figure 5(j), DSP-YOLO also works well on small object detection. The second scenario is for detection on occluded object. As shown in Figure 5(a) to

---

![Figure 5.](image-url)  
Comparison of detection results of YOLOv3 and DSP-YOLO. The first column shows the results of YOLOv3 and the second column is the result of DSP-YOLO. DSP: dilated spatial pyramid; YOLO: You only look once.
Acknowledgements

The authors would like to thank the support from Jiangsu Overseas Visiting Scholar Program for University Prominent Young & Middle-aged Teachers and Presidents.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work has supported by the projects of the National Key Research and Development Plan of China [Grant Number: 2016YFB0502103] and the Natural Science Foundation of Jiangsu Province of China [Grant Number: BK20160696].

ORCID iD

Xiaoguo Zhang https://orcid.org/0000-0001-9961-6443

References

1. Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE conference on computer vision and pattern recognition, CVPR 2014 (eds C Schmid, S Soatto and C Tomasi), Columbus, OH, USA, 23–28 June 2014, pp. 580–587. Piscataway, NJ: IEEE.
2. Girshick R. Fast R-CNN. In: IEEE international conference on computer vision (eds K Grauman, K Grauman, K Grauman and A Zisserman), Santiago, Chile, 7–13 December 2015, pp. 1440–1448. Piscataway, NJ: IEEE.
3. Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. Adv Neur In 2015; 39(6): 91–99.
4. Dai J, Li Y, He K, et al. R-FCN: object detection via region-based fully convolutional networks. In: Advances in neural information processing systems 29: Annual conference on neural information processing systems 2016 (eds DD Lee, M Sugiyama, UV Luxburg, I Guyon and R Garnett), Barcelona, Spain, 5–10 December 2016, pp. 379–387. Cambridge, MA: MIT Press.
5. Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection. In: 2016 IEEE conference on computer vision and pattern recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016, pp. 779–788. Piscataway, NJ: IEEE.
6. Liu W, Anguelov D, Erhan D, et al. SSD: single shot multi-box detector. In: European conference on computer vision (eds B Leibe, J Matas, N Sebe and M Welling), Amsterdam, The Netherlands, 11–14 October 2016, pp. 21–37. Berlin, German: Springer.
7. Redmon J and Farhadi A. YOLO9000: better, faster, stronger. In: 2017 IEEE conference on computer vision and pattern recognition, CVPR 2017 (eds Y Liu, JM Rehg, CJ Taylor and Y Wu), Honolulu, HI, USA, 21–26 July 2017, pp. 6517–6525. Piscataway, NJ: IEEE.
8. Redmon J and Farhadi A. YOLOv3: an incremental improvement. arXiv preprint arXiv: 1804.02767, 2018.
9. Li R and Yang J. Improved YOLOv2 object detection model. In: 2018 6th International conference on multimedia computing and systems (ICMCS) (eds M Essaaidi and Y Zaz), Rabat, Morocco, 10–12 May 2018, pp. 1–6. Piscataway, NJ: IEEE.
10. Lin TY, Dollár P, Girshick R, et al. Feature pyramid networks for object detection. In: IEEE conference on computer vision and pattern recognition CVPR 2017 (eds Y Liu, JM Rehg, CJ Taylor and Y Wu), vol. 1, Honolulu, HI, USA, 21–26 July 2017, pp. 936–944. Piscataway, NJ: IEEE.
11. Zhang F, Yang F, and Li C. Fast vehicle detection method based on improved YOLOv3. Com Eng App 2019; 55: 12–20.
12. Huang Z and Wang J. DC-SPP-YOLO: dense connection and spatial pyramid pooling based YOLO for object detection. arXiv preprint arXiv:1903.08589, 2019.
13. He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE T Pattern Anal 2015; 37: 1904–1916.
14. Xu C, Wang X, and Yang Y. Attention-YOLO: YOLO detection algorithm that introduces attention mechanism. ComEngApp 2019; 55(6): 13–23.
15. Adelson EH, Anderson CH, Bergen JR, et al. Pyramid methods in image processing. RCA Engineer 1984; 29: 33–41.
16. Dalal N and Triggs B. Histograms of oriented gradients for human detection. In: IEEE computer society conference on computer vision and pattern recognition CVPR 2005 (eds C Schmid, S Soatto and C Tomasi), San Diego, CA, USA, 20–25 June 2005, pp. 4278–4284. Piscataway, NJ: IEEE.
17. Lowe DG. Distinctive image features from scale-invariant keypoints. Int J Comput Vis 2004; 60(2): 91–110.

Conclusions

To improve the multi-scale object detection performance of YOLOv3, in this article, a novel DSP module consisting of multiple parallel branches with different dilation rates is proposed and integrated with YOLOv3 in front of the first detection header to present DSP-YOLO model. Experiments show that our DSP-YOLO model outperforms YOLOv3 on PASCAL VOC2007 for not only its medium and larger object detection ability but also its detection ability on occluded and small objects, and at the same time, it is still able to maintain the comparable speed as that of YOLOv3. What’s more, for 608 × 608 input, DSP-YOLO achieves 83.1% mAP, surpassing other state-of-the-art object detectors on PASCAL VOC2007 while keeping the real-time speed. In addition, although the DSP module is only applied to YOLOv3 in this article, it can act as a generic solution for scale-invariant representations and can be integrated with other object detectors like SSD.

In the future work, DSP-YOLO will be applied to the autonomous driving filed to deal with the car and people detection in both close and far scene.
18. Li Z and Zhou F. FSSD: feature fusion single shot multibox detector. *arXiv preprint arXiv: 1712.00960*, 2017.
19. Liu S, Qi L, Qin H, et al. Path aggregation network for instance segmentation. In: *IEEE conference on computer vision and pattern recognition CVPR 2018* (eds D Forsyth, I Laptev, D Ramanan and A Oliva), Salt Lake City, Utah, USA, 18–22 June 2018, pp. 8759–8768. Piscataway, NJ: IEEE.
20. Liu S, Di H, and Wang Y. Receptive field block net for accurate and fast object detection. *arXiv preprint arXiv: 1711.07767*, 2017.
21. Mehta S, Rastegari M, Caspi A, et al. ESPNet: efficient spatial pyramid of dilated convolutions for semantic segmentation. In: *European conference on computer vision ECCV* (eds V Ferrari, M Hebert, C Sminchisescu and Y Weiss), Munich, Germany, 8–14 September 2018, pp. 552–568. Berlin, Germany: Springer.
22. Li Y, Chen Y, Wang N, et al. Scale-aware trident networks for object detection. *arXiv preprint arXiv: 1901.01892*, 2019.
23. Yu F and Koltun V. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv: 1511.07122*, 2016.
24. Everingham M, Van Gool L, Williams CKI, et al. The pascal visual object classes (VOC) challenge. *Int J Comput Vis* 2010; 88: 303–338.
25. Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. *Int J Comput Vis* 2015; 115: 211–252.
26. Fu CY, Liu W, Ranga A, et al. DSSD: deconvolutional single shot detector. *arXiv preprint arXiv: 1701.06659*, 2017.
27. Zhou P, Ni B, Geng C, et al. Scale-transferrable object detection. In: *IEEE conference on computer vision and pattern recognition CVPR 2018* (eds D Forsyth, I Laptev, D Ramanan and A Oliva), Salt Lake City, Utah, USA, 18–22 June 2018, pp. 528–537. Piscataway, NJ: IEEE.
28. Zhang S, Wen L, Bian X, et al. Single-shot refinement neural network for object detection. In: *IEEE conference on computer vision and pattern recognition CVPR 2018* (eds D Forsyth, I Laptev, D Ramanan and A Oliva), Salt Lake City, Utah, USA, 18–22 June 2018, pp. 4203–4212. Piscataway, NJ: IEEE.