Multi-flow Sub-network and Multiple Connections for Single Shot Detection

Ye Li[1], Huicheng Zheng[1,2,3(✉)], and Lvran Chen[1]

[1] School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
zhenghch@mail.sysu.edu.cn
[2] Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou, China
[3] Guangdong Key Laboratory of Information Security Technology, Guangzhou, China

Abstract. One-stage object detection methods are usually more computationally efficient than two-stage methods, which makes it more likely to be applied in practice. However, one-stage methods often suffer from lower detection accuracies, especially when the objects to be detected are small. In this paper, we propose a multi-flow sub-network and multiple connections for single shot detection (MSSD), which is built upon a one-stage strategy to inherit the computational efficiency and improve the detection accuracy. The multi-flow sub-network in MSSD aims to extract high quality feature maps with high spatial resolution, sufficient non-linear transformation, and multiple receptive fields, which facilitates detection of small objects in particular. In addition, MSSD uses multiple connections, including up-sampling, down-sampling, and resolution-invariant connections, to combine feature maps of different layers, which helps the model capture fine-grained details and improve feature representation. Extensive experiments on PASCAL VOC and MS COCO demonstrate that MSSD achieves competitive detection accuracy with high computational efficiency compared to state-of-the-art methods. MSSD with input size of 320×320 achieves 80.6% mAP on VOC2007 at 45 FPS and 29.7% mAP on COCO, both with a Nvidia Titan X GPU.

Keywords: Object detection · Single shot detection
Feature representation enhancement

1 Introduction

In recent years, many outstanding object detection methods based on deep learning have been proposed. They are mainly divided into two categories: two-stage methods and one-stage methods. The two-stage methods usually achieve better detection performance, while the one-stage methods are usually more computationally efficient. However, when an object detection method is to be applied in

J.-H. Lai et al. (Eds.): PRCV 2018, LNCS 11257, pp. 168–179, 2018.
https://doi.org/10.1007/978-3-030-03335-4_15

practice, the detection accuracy and computational efficiency must be considered together.

It is a feasible idea to design an advanced one-stage method which has good accuracy while maintain the advantage of in computationally efficiency. Some advanced one-stage methods, such as DSSD [5], RetinaNet [6], and BPN [7] sacrifice computational efficiency when improving the accuracy. In order to improve the accuracy while maintaining the computational efficiency, this paper analyzes the deficiencies of the one-stage methods. Many experimental results show that one-stage methods are weak in small object detection and feature representation. To address these issues, we propose a single shot detector with multi-flow sub-network and multiple connections (MSSD). The main motivations and corresponding operations of MSSD are as follows.

First, this paper tries to solve the difficulty in small object detection. Since low-level features are important for small object detection, as mentioned in [9], this paper proposes a multi-flow sub-network module to optimize the low-level feature representation by obtaining deeply non-linear transformation and different receptive fields. Then, this paper tries to enhance the feature representation of the model. A common method is to employ a complex backbone network such as ResNet101 [10], but this will lead to low computational efficiency. This paper enhances feature representation by reusing different feature maps through multiple connections, which has little affect on computational efficiency. Thanks to the multi-flow sub-network and multiple connections, MSSD achieves state-of-art results with a lightweight backbone network, such as ResNet18 [10], while maintaining the real-time computational speed. Different from SSD, we introduce shortcut connections to the extra feature layers to strengthen feature propagation and further reduce the number of detected feature maps to improve the generalization of the network.

The contributions of this paper can be summarized as follows:

1. A multi-flow sub-network module is proposed to obtain high quality feature maps with high spatial resolution, sufficient non-linear transformation, and multiple receptive fields, which is beneficial for object detection, especially for small instances.
2. A multiple connection module is proposed to enhance feature representation by encouraging feature reuse rather than using complex backbone networks.
3. The extra feature layers are modified to strengthen feature propagation and improve the network generalization.
4. MSSD achieves the state-of-the-art results on PASCAL VOC 2007, 2012 [1] and MS COCO [2].

2 Related Work

Object Detection. Early object detection methods like those based on DPM [11] and HOG [12] employ hand-crafted features, and the detection system consist of three modules: region selection, feature extraction, and classification. With the development of deep convolutional networks, deep learning based methods

have attracted great attention. These methods can be roughly divided into two categories, two-stage methods and one-stage methods.

Two-stage methods, such as RCNN [13], Fast RCNN [14], and Faster RCNN [15], consist of two parts, where the first one generates candidate object proposals, and the second one classifies the candidate regions and determines its accurate location using convolutional neural networks. Such methods are superior in accuracy, but difficult to achieve real-time performance. Methods like Mask-RCNN [3], R-FCN [24], and CoupleNet [4] achieve state-of-the-art accuracies with complex backbone networks. However, the resulting huge computational cost restricts their applications in practice.

One-stage methods, represented by YOLO [16] and SSD [8], convert the object detection problem into a regression problem. Such methods implement end-to-end training and detection, and do not require the generation of candidate regions, which ensures their high computational efficiency. However, the accuracy of one-stage methods trails that of two-stage methods. Some methods like DSSD [5] and RetinaNet [6] use complex backbone network to achieve high accuracy comparable to two-stage methods but sacrifice computational speed.

Receptive Fields. There are several methods to improve feature representation by constructing feature map with different receptive fields. [20] uses a multi-scale input layer to construct an image pyramid to achieve multiple levels of receptive field sizes. GoogLeNet [21] uses filters of different sizes to obtain feature map with different receptive fields. The deformable-net [22] replaces the original fixed position sample with the offset sample, so that the sample point position can change with the image content. In addition, DICSSD [18] and RBFnet [19] use dilated convolution [17] to obtain feature map with different receptive fields. The multi-flow sub-network of MSSD also employ dilated convolution. However, compared with DICSSD which only uses dilated convolution directly on all detected feature maps, MSSD combines dilated convolution, group convolution and bottle-net into a sub-network module and performs much better. Unlike RBFnet whose module is complicated and motivated by biological vision, the multi-flow sub-network of MSSD has simple structure (each branch is the same in topology) and is proposed to solve the difficulty of small object detection.

Short-Path Methods. Among the various connection methods currently used, some methods only use resolution-invariant connections such as DenseNet [23]; some methods only use up-sampling and resolution-invariant connections, such as DSSD; and some methods only use down-sampling connections, such as ResNet [8]. In contrast to them, the multiple connections of MSSD includes up-sampling, down-sampling, and resolution-invariant connections. To the best of our knowledge, in the one-stage object detection methods, MSSD is the first one to combine such multiple connections.

3 MSSD

The pipeline of the MSSD proposed in this paper is shown in Fig. 1, which consists of five parts. The first is a backbone network. The second is the extra feature layers (conv8_1–conv10_1) with shortcut connections marked in yellow. The third part is two dilated convolutional layers (conv6, conv7), used to connect the backbone network and the extra feature layers. The fourth is a multiple connection module that makes full use of the five detected feature maps. The fifth is the multi-flow sub-network module proposed for addressing the problem of small object detection.

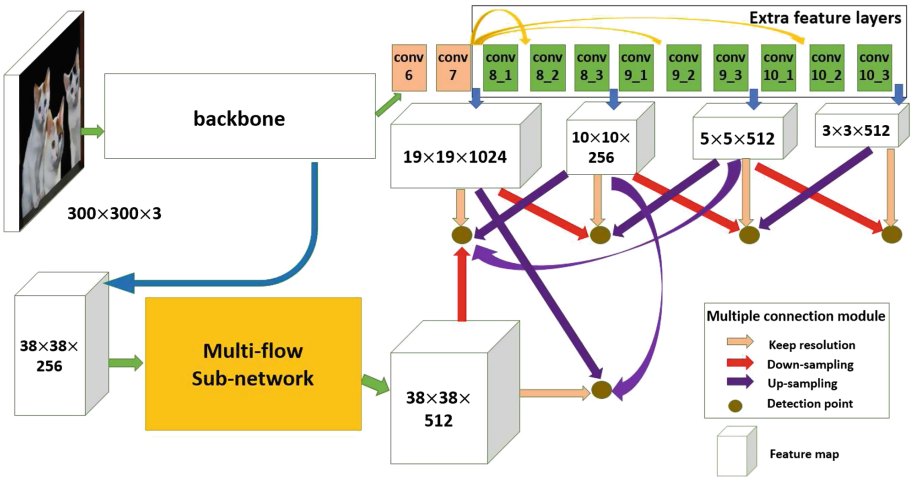

Fig. 1. The network structure of MSSD. The backbone is a pre-trained ResNet18 whose average pooling layer and fc layer are removed. B1 to B4 are layers of ResNet18 and the size of the feature map after B3 is $38 \times 38 \times 256$. The yellow connections between different extra feature layers are three convolutional layers which are used to connect different feature maps. (Color figure online)

3.1 The Multi-flow Sub-network Module

The multi-flow sub-network module aims to solve the problem in small object detection. The poor performance of small object detection is mainly due to the fact that the spatial resolutions of the high-level feature maps are too low and the receptive fields are too large.

In high-level feature maps, the model tends to focus on large objects, ignoring small objects. Although the low-level feature maps have high spatial resolution, they have insufficient nonlinear transformation due to a limited number of convolutional layers and nonlinear active layers they passed. Therefore, it is also difficult to detect small objects. In addition, each feature map used for detection has a fixed receptive field size, which is undoubtedly not the best choice for detecting objects with different sizes and shapes.

The multi-flow sub-network module is shown in Fig. 2. The module consists of multiple branches, each with four convolutional layers (each convolutional layer followed by a batch normalization layer and ReLU layer). The conv1 of each branch is a dilated convolutional layer [17] with different parameters, so that different branches can obtain feature maps of different receptive fields. This paper refers to the bottleneck architecture in GoogLeNet [21], so conv2 and conv4 are designed as convolutional layers with 1×1 kernel sizes. In addition, conv3 uses group convolution. These two operations allow MSSD to obtain feature maps with good feature representations without much increase in computation. Each of the branches is the same except for conv1, but the parameters between these same structures are not shared. At the end of the module, the feature maps extracted from the multi-flow sub-network module are concatenated with the original feature map and a feature map with high spatial resolution, sufficiently complex nonlinear transformation, multiple receptive fields and context information is obtained.

In addition, since the lowest-level feature maps (which produce more than 70% default boxes) are significant to objects detection (especially to small object detection), the multi-flow sub-network module is only used to process the lowest-level detected feature map.

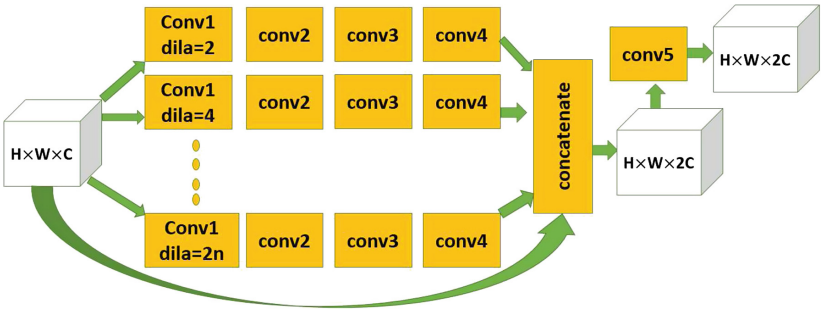

Fig. 2. Structure of the multi-flow sub-network

3.2 The Multiple Connection Module

Many methods like CoupleNet [4], R-FCN [24], and DSSD [5] enhance feature representation by using complex backbone networks, but sacrifice computational efficiency. In this paper, in order to enhance feature representation, five feature maps extracted from the model are reused by multiple connections such as up-sampling convolution, down-sampling convolution, and resolution-invariant convolution. Without sacrificing computational speed, 5 high quality feature maps were extracted for detecting. The specific operation of the multiple connection module is shown in Fig. 3.

Each detection point may obtain different number of feature maps obtained through different connection methods. These different feature maps are complementary. In this paper, the feature maps obtained from up-sampling convolution

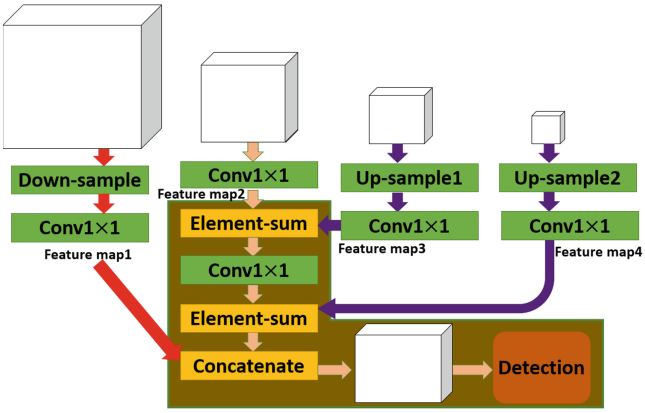

Fig. 3. Detection point

and resolution-invariant convolution are combined together through element-sum operation. The new feature map will be concatenated with the feature map obtained through down-sampling convolution. Then, the high quality feature maps are obtained. For example, if the detection point get *Feature map1*, *Feature map2* and *Feature map3*, *Feature map2* and *Feature map3* are firstly combined together through element-sum operation. Then the new feature map will be concatenated with *Feature map1* to get the final feature map. If the detection point only get *Feature map1* and *Feature map2*, *Feature map2* will be concatenated with *Feature map1* to get the final feature map. This kind of well-designed structure makes the information of the feature maps fully utilized, and helps the model to capture more fine-grained details. Finally, these 5 high quality feature maps are used to calculate the final detection results by following SSD, more details of which can be found in [8].

3.3 The New Extra Feature Layers Module

The original SSD method is very classic, which employs 6 layers to produce 6 detected feature maps. More specifically SSD divides the target objects in the image into six parts according to their sizes. Each part corresponds to a size range, and then the network extract six feature maps that are responsible for different sizes of objects. However, such kind of design is not very generalized. When different data sets are trained and tested, it is necessary to repartition the six detection ranges, often resulting in some inconveniences.

Therefore, when designing the new extra feature layers of MSSD, this paper removes the original con11_1 and conv11_2 that generate the 1×1 feature map, which reduces the number of detected feature maps and enhances the generalization. In addition, in order to improve the expression ability of the model, conv8_3, conv9_3, and conv10_3 are introduced in the new extra feature layers.

What's more, we introduce three shortcut connections in the new extra feature layers to strengthen feature propagation.

4 Experiments

In order to verify the reliability and effectiveness of the proposed object detection method, experiments are conducted on two benchmarks: PASCAL VOC and MS COCO. We follow nearly the same training policy as SSD [8], including loss function (e.g., smooth L1 loss for localization and softmax loss for classification), matching strategy, data augmentation and hard negative mining, while learning rate scheduling is slightly changed.

The details of the MSSD network structure are as follows: The MSSD uses the pre-trained ResNet18 as the backbone network. The multi-flow sub-network module uses four branches. The conv1 of each branch network uses dilated convolution and the corresponding dilation parameters are $\{2, 4, 6, 8\}$, respectively. The group parameter of the group convolution layer for each branch is 4. In addition, all experiments are carried out on one Nvidia Titan X GPU.

4.1 PASCAL VOC

There are two types of experiments conducted on PASCAL VOC, one is trained on the union set of 2007 *trainval* and 2012 *trainval*, tested on 2007 *test* set. The other one is trained on union set of 2007 *trainval* and 2012 *trainval* and 2007 *test*, tested on the 2012 *test* set. In the experiments of PASCAL VOC, the training setting of MSSD is basically the same as that of SSD [8]. We use the SGD algorithm to train the network. The initial learning rate is 10^{-3}, the momentum is 0.9, the weight decay is 0.0005, and the batch size is 32. Due to resource limitations, batch size is 16 when training MSSD512. The number of MSSD training iterations is 120k. When the number of iterations are 80k, 90k, 100k, and 110k, the learning rates are reduced to 5×10^{-4}, 1×10^{-4}, 5×10^{-5}, and 1×10^{-5}, respectively. The results of two types of experiments are shown in Table 1 and 2, respectively.

Compared with all the one-stage methods and two-stage methods in Table 1, MSSD512 achieves the best detection accuracy while maintaining the real-time computational speed. MSSD achieves better performance than the baseline SSD in detection accuracy and computational speed. MSSD300_v and MSSD512_v achieve comparable performance with MSSD300 and MSSD512, respectively, which further verifies the effectiveness of the contributions we proposed.

In order to ensure the reliability and stability of MSSD, this paper also shows the test results of MSSD in PASCAL VOC2012. From Table 2, we can see that MSSD still achieved excellent performance. MSSD300 exceeds SSD300 2.5% in detection accuracy and achieves better performance than OHME++ which employs a larger input size.

Table 1. PASCAL VOC 2007 detection results. All methods are trained on VOC 2007 *trainval* sets and VOC 2012 *trainval* sets, and tested on VOC 2007 *test* set with a Nvidia Titan X GPU. Only the batch size of MSSD512 is 16 during training.

Method	Backbone	Input size	mAP	FPS
Faster RCNN [15]	VGG16	$\sim 1000 \times 600$	73.2	7
R-FCN [24]	ResNet101	$\sim 1000 \times 600$	80.5	9
CoupleNet [4]	ResNet101	$\sim 1000 \times 600$	81.7	8.7
SSD300 [8]	VGG16	300×300	77.2	46
SSD512 [8]	VGG16	512×300	79.8	19
RSSD300 [25]	VGG16	300×300	78.5	35
YOLOv2 [26]	Darknet-19	544×544	78.6	40
FSSD300 [27]	VGG16	300×300	78.8	–
DSOD300 [28]	DS/64-192-48-1	300×300	77.7	–
DSSD321 [5]	ResNet101	321×321	78.6	9.5
DICSSD300 [18]	VGG16	300×300	78.1	40.8
RefineDet320 [29]	VGG16	320×320	80.0	40.3
RefineDet512 [29]	VGG16	512×512	81.8	24.1
BPN320 [7]	VGG16	320×320	80.3	32.4
BPN512 [7]	VGG16	512×512	**81.9**	18.9
MSSD300_v	VGG16	300×300	80.0	43.0
MSSD512_v	VGG16	512×512	81.6	20.0
MSSD300	ResNet18	300×300	80.3	**55.7**
MSSD320	ResNet18	320×320	80.6	45.4
MSSD512	ResNet18	512×512	**81.9**	21.4

Table 2. PASCAL VOC 2012 detection results. All methods are trained on union set of PASCAL VOC 2007 *trainval* and PASCAL VOC 2012 *trainval* and PASCAL VOC2007 *test*, and tested on PASCAL VOC 2012 *test* set. For more details about our results, please see http://host.robots.ox.ac.uk:8080/anonymous/5BMTAL.html

Method	Backbone	Input size	mAP
Faster RCNN [15]	VGG16	$\sim 1000 \times 600$	70.4
R-FCN [24]	ResNet101	$\sim 1000 \times 600$	77.6
SSD300 [8]	VGG16	300×300	75.8
DSSD321 [5]	ResNet101	321×321	76.3
RefineDet320 [29]	VGG16	320×320	78.1
MSSD300	ResNet18	300×300	**78.3**

4.2 MS COCO

In order to further verify the effectiveness of MSSD, especially to assess the performance of MSSD in small object detection, this paper also conducted experiments on MS COCO. MSSD is trained on *trainval35k* (2014 *train* + 2014 *val35k*) and the training policy of MSSD is also almost the same as that of SSD [8]. We train the network using SGD with momentum 0.9, weight decay 0.0005 and batch size 32. The number of MSSD training epochs is 120. In the top 5 epochs, we apply the "warmup" technique to gradually increase learning rate from 1×10^{-6} to 1×10^{-3}. When the number of epochs are 80 and 100, the learning rate are reduced to 1×10^{-4} and 1×10^{-5}, respectively.

Table 3. MS COCO 2017 test-dev detection results. Our MSSD are trained on *trainval35k*.

Method	Backbone	Data	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster RCNN [15]	VGG16	trainval	21.9	42.7	–	–	–	–
OHME++ [30]	VGG16	trainval	25.5	45.9	26.1	7.4	27.7	40.3
YOLOv2 [26]	DarkNet-19	trainval35k	21.6	44.0	19.2	5.0	22.4	35.5
SSD300 [8]	VGG16	trainval35k	25.1	43.1	25.8	6.6	25.9	41.4
DSSD321 [5]	ResNet101	trainval35k	28.0	46.1	29.2	7.4	28.1	**47.6**
RefineDet320 [29]	VGG16	trainval35k	29.4	49.2	**31.3**	10.0	32.0	44.4
MSSD300	ResNet18	trainval35k	29.1	49.6	30.1	11.2	31.2	43.4
MSSD320	ResNet18	trainval35k	**29.7**	**50.4**	30.8	**12.8**	**32.1**	42.8

Table 3 shows that MSSD300 exceeds SSD300 3.9% in AP. MSSD320 achieves state-of-the-art detection accuracy. In the small object detection, all the methods in Table 3 are exceeded by MSSD320. This fully proves that the multi-flow network module for small object detection is very effective. Compared to most of other state-of-the-art one-stage methods (such as RefineDet) and two-stage methods (such as OHME++), MSSD achieves higher detection accuracy in the same condition.

4.3 Ablation Study

In order to verify the role of the three innovations of MSSD, a series of confirmatory experiments were also conducted in this paper. All confirmatory experiments were trained on the union set of PASCAL VOC 2007 *trainval* and 2012 *trainval*, and tested on the 2007 *test* set. The input image size for all experiments was 300×300. The experimental results are shown in Table 4.

The primitive model v1 is SSD with a ResNet18 backbone network. At this time, the mAP is 76.9. If the multi-flow sub-network module is added to the model v1, the model v2 is obtained, and the mAP of v2 is 78.6. The multiple

Table 4. Results of the confirmatory experiments.

Component	MSSD300			
	v1	v2	v3	v4
Multi-flow sub-network		√	√	√
Multiple connection module			√	√
New extra feature layers				√
mAP	76.9	78.6	79.8	80.3

connection module is introduced into the model v2, and the model v3 is obtained. The mAP of v3 is 79.8. Finally, we remove the conv11_1 and conv11_2 of the extra feature layers of v3 and introduce shortcut connections and three convolution layers to obtain new extra feature layers. At this point the model is v4, and the mAP becomes 80.3.

Table 4 shows that each key component in this paper can bring about improvements in the detection performance. In addition, as the number of key components increases, the model performs better, which further confirms the reliability and effectiveness of MSSD.

4.4 Visualization

In order to understand the detection effect of MSSD more intuitively, this section presents some of the results of MSSD testing on PASCAL VOC 2007, as shown in Fig. 4.

Fig. 4. Detection examples on PASCAL VOC 2007 *test* set with MSSD512 model.

5 Conclusion

This paper analyzes deficiencies of the existing object detection methods and proposes a multi-flow sub-network and multiple connections for single shot detection

(MSSD). MSSD maintains real-time computational speed and achieves better detection accuracy than state-of-the-art methods. Compared with the existing object detection method, MSSD achieved the state-of-the-art detection accuracy with a smaller input size and a higher computational speed. MSSD has successfully achieved the original intention of this paper. It helps object detection method to be applied in practice better, and also contributes to the solution of the difficulty in small object detection and weak feature representation, which are commonly found in one-stage methods. MSSD has achieved good performance on PASCAL VOC and MS COCO. In the future, we may consider combining relevant knowledge in the field of transfer learning and further migrate more information.

Acknowledgements. This work was supported by National Natural Science Foundation of China (U1611461), Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase, No. U1501501), and Science and Technology Program of Guangzhou (No. 201803030029).

References

1. Everingham, M., Gool, L.V., Williams, C.K., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010)
2. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
3. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: International Conference on Computer Vision, pp. 2980–2988. IEEE, Venice (2017)
4. Zhu, Y., Zhao, C., Wang, J., Zhao, X., Wu, Y., Lu, H.: CoupleNet: coupling global structure with local parts for object detection. In: International Conference on Computer Vision, pp. 4146–4154. IEEE, Venice (2017)
5. Fu, C.Y., Liu, W., Ranga, A., Tyagi, A., Berg, A.C.: DSSD: deconvolutional single shot detector. arXiv preprint arXiv:1701.06659 (2017)
6. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: International Conference on Computer Vision, Venice, pp. 2999–3007. IEEE (2017)
7. Wu, X., Zhang, D., Zhu, J., Steven C.H.: Single-shot bidirectional pyramid networks for high-quality object detection. arXiv preprint arXiv:1803.08208 (2018)
8. Liu, W., et al.: SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
9. Hu, P., Ramanan, D.: Finding tiny faces. In: IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, pp. 1522–1530. IEEE (2017)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, pp. 770–778. IEEE (2016)
11. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, pp. 1–8. IEEE (2008)

12. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Conference on Computer Vision and Pattern Recognition, San Diego, pp. 886–893. IEEE (2005)
13. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, Columbus, pp. 580–587. IEEE (2014)
14. Girshick, R.: Fast R-CNN. In: International Conference on Computer Vision, Santiago, pp. 1440–1448. IEEE (2015)
15. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: International Conference on Neural Information Processing Systems, Montreal, pp. 91–99. MIT (2015)
16. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, pp. 779–788. IEEE (2016)
17. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
18. Xiang, W., Zhang, D.Q., Athitsos, V., Yu, H.: Context-aware single-shot detector. arXiv preprint arXiv:1707.08682 (2017)
19. Liu, S., Huang, D., Wang, Y.: Receptive field block net for accurate and fast object detection. arXiv preprint arXiv:1711.07767 (2017)
20. Fu, H., Cheng, J., Xu, Y., Wong, D.W.K., Liu, J., Cao, X.: Joint optic disc and cup segmentation based on multi-label deep network and polar transformation. IEEE Trans. Med. Imaging **37**(7), 1597–1605 (2018)
21. Szegedy, C., et al.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition, Boston, pp. 1–9. IEEE (2015)
22. Dai, J., et al.: Deformable convolutional networks. In: IEEE International Conference on Computer Vision, Venice, pp. 764–773. IEEE (2017)
23. Huang, G., Liu, Z., Maaten, L.V.D., Weinberger, K.Q.: Densely connected convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, pp. 2261–2269. IEEE (2017)
24. Dai, J., Li, Y., He, K., Sun, J., et al.: R-FCN: object detection via region-based fully convolutional networks. In: International Conference on Neural Information Processing Systems, Barcelona, pp. 379–387. MIT (2016)
25. Jeong, J., Park, H., Kwak, N.: Enhancement of SSD by concatenating feature maps for object detection. arXiv preprint arXiv:1705.09587 (2017)
26. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, pp. 6517–6525. IEEE (2016)
27. Li, Z., Zhou, F.: FSSD: feature fusion single shot multibox detector. arXiv preprint arXiv:1712.00960 (2017)
28. Shen, Z., Liu, Z., Li, J., Jiang, Y.G., Chen, Y., Xue, X.: DSOD: learning deeply supervised object detectors from scratch. In: IEEE International Conference on Computer Vision, Venice, pp. 1937–1945. IEEE (2017)
29. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.: Single-shot refinement neural network for object detection. arXiv preprint arXiv:1711.06897 (2017)
30. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, pp. 761–769. IEEE (2016)