Feature Enhancement and Reconstruction for Small Object Detection

Chong-Jian Zhang[1,2], Song-Lu Chen[1,2], Qi Liu[1,2], Zhi-Yong Huang[1,2],
Feng Chen[2,3], and Xu-Cheng Yin[1,2](✉)

[1] University of Science and Technology Beijing, Beijing 100083, China
{chongjianzhang,qiliu7,huang.zhiyong}@xs.ustb.edu.cn,
{songluchen,xuchengyin}@ustb.edu.cn
[2] USTB-EEasyTech Joint Lab of Artificial Intelligence, Beijing 100083, China
[3] EEasy Technology Company Ltd., Zhuhai 519000, China
cfeng@eeasytech.com

Abstract. Due to the small size and noise interference, small object detection is still a challenging task. The previous work can not effectively reduce noise interference and extract representative features of the small object. Although the upsampling network can alleviate the loss of features by enlarging feature maps, it can not enhance semantics and will introduce more noises. To solve the above problems, we propose CAU (Content-Aware Upsampling) to enhance feature representation and semantics of the small object. Moreover, we propose CSA (Content-Shuffle Attention) to reconstruct robust features and reduce noise interference using feature shuffling and attention. Extensive experiments verify that our proposed method can improve small object detection by 2.2% on the traffic sign dataset TT-100K and 0.8% on the object detection dataset MS COCO compared with the baseline model.

Keywords: Small object detection · Content-aware upsampling · Content-shuffle attention

1 Introduction

Small object detection has a wide range of applications, such as traffic sign detection, face recognition, and remote sensing image analysis. However, due to the small size and noise interference, generic object detectors are not effectively applicable to small object detection. A common practice to detect small objects is to enlarge the feature map using upsampling. Traditional upsampling methods include nearest-neighbor interpolation and bilinear interpolation, which can enlarge the image resolution but introduce more noises. Deep-learning-based upsampling method DUpsampling [23] can enlarge the feature map based on the relationship between pixels. However, the downsampling operation in the network can cause the loss of object information, especially for the small object,

D.-T. Dang-Nguyen et al. (Eds.): MMM 2023, LNCS 13833, pp. 16–27, 2023.
https://doi.org/10.1007/978-3-031-27077-2_2

thus leading to missing detection of the small object. CARAFE [25] is an upsampling method via content sensing and feature recombination, which can reduce the information loss of small objects via context modeling. However, it does not consider multi-scale features during feature recombination and is not conducive to detecting small objects. To solve the above problems, we propose an upsampling module CAU to reduce the loss of object information by aggregating the global context information and extracting multi-scale features, thus improving small object detection. Moreover, small objects are easily affected by background noises. The attention mechanism can suppress noise interference by focusing on essential areas and ignoring irrelevant information. SENet [10] and ECANet [26] can capture the relationship between channels with channel attention, suppressing background noises using global context modeling. However, the channel attention can not capture local information around the object, affecting small object detection. Bilinear attention mechanism GSoP-Net [8] and Fang et al. [6] propose to capture local feature interactions within each channel via spatial attention. Although the spatial attention can effectively utilize local relationships, the feature interaction brings heavy computational complexity. To solve the above problems, we propose a CSA module to reconstruct features via feature shuffling and attention, which combines channel attention and feature shuffling for robust representation. CSA can improve small object detection with little computation added. In this paper, CAU is added to the detection neck, and CSA is added to the detection head. By adding a few parameters and calculations, small object detection can be improved significantly. Our main contributions can be summarized as follows.

1. To reduce the loss of object information, we propose an upsampling module CAU, which can enhance feature representation via global context aggregation and multi-scale feature extraction.
2. To reduce background noise interference, we propose an attention module CSA, which significantly improves small object detection via robust feature reconstruction.
3. Our proposed method achieves 66.1% and 21.9% on the small object of TT-100K and MS COCO, respectively, 2.2% and 0.8% higher than the baseline model by adding a few parameters and calculations.

2 Related Work

2.1 Upsampling Method

Upsampling is used to restore the resolution of the image or feature map to the original resolution. Traditional upsampling methods include nearest-neighbor upsampling, bilinear upsampling, and bicubic upsampling, which only improve the image resolution according to its image signal. However, these methods bring many side effects, such as increased noise and computational complexity. To solve the above problems, deep-learning-based upsampling methods are proposed. Sub-pixel layer [20] is an end-to-end learnable layer that generates and recombines multiple channels to perform upsampling. DUpsampling [23] performs downsampling operations for low-level feature representations and then concatenate them with high-level features to complete feature fusion and upsampling. CARAFE [25] performs feature upsampling by recombining features in the region centered at each location by weighted combination, which can aggregate contextual information of large receptive fields. However, small objects will be missed due to the lack of multi-scale features in feature recombination. We propose an upsampling module that can aggregate global context and extract multi-scale features to improve small object detection.

2.2 Multi-scale Feature Extraction

SPPNet [9] and GoogLeNet [21] propose a parallel branch to extract features at different spatial scales based on its receptive field, i.e., spatial pyramid. Liu et al. propose RFBNet [16] with dilated convolution and fuse three branches to improve the poor representation of small objects. Zhao et al. [30] use global average pooling to extract multi-scale information and achieve competitive results in semantic segmentation. Chen et al. [3] design multiple parallel dilated convolution modules with different sampling rate to extract multi-scale features. We propose to extract global information by adding a multi-scale feature extraction module.

2.3 Attention Mechanism

The attention mechanism can focus on the essential area of the object and suppress irrelevant information. Channel attention includes SENet [10] and ECANet [26], which focuses on the relationship between channels and automatically learns the importance of different channel features. Bilinear attention mechanism GSoP-Net [8] and Fang et al. [6] propose to enhance the local pairwise feature interaction in each channel while retaining spatial information, improving feature representation via local relationship. SANet [28] introduces the random channel mixing operation, which uses spatial and channel attention mechanisms in parallel. Ying et al. [27] propose a multi-scale global attention mechanism to alleviate the loss of small object information caused by downsampling operations. We propose to combine feature reconstruction and attention mechanism to enhance object feature representation.

3 Method

3.1 Network Architecture

The overall network architecture is shown in Fig. 1. We choose the lightweight YOLOv5 as our baseline model. YOLOv5 uses CSPDarknet53 [18] as the backbone network for feature extraction. The detection neck adopts a Path Aggregation Network (PANet) [15] for feature fusion, where the low-resolution features are upsampled by nearest-neighbor upsampling to be concatenated with the high-resolution features. In this paper, we introduce an upsampling module CAU to replace the nearest-neighbor upsampling module in the detection neck. CAU can reduce information loss by global context aggregation and multi-scale feature extraction. Moreover, we propose an attention module CSA in the detection head to reconstruct features for robust representation. CAU and CSA can improve small object detection with a few parameters and calculations added.

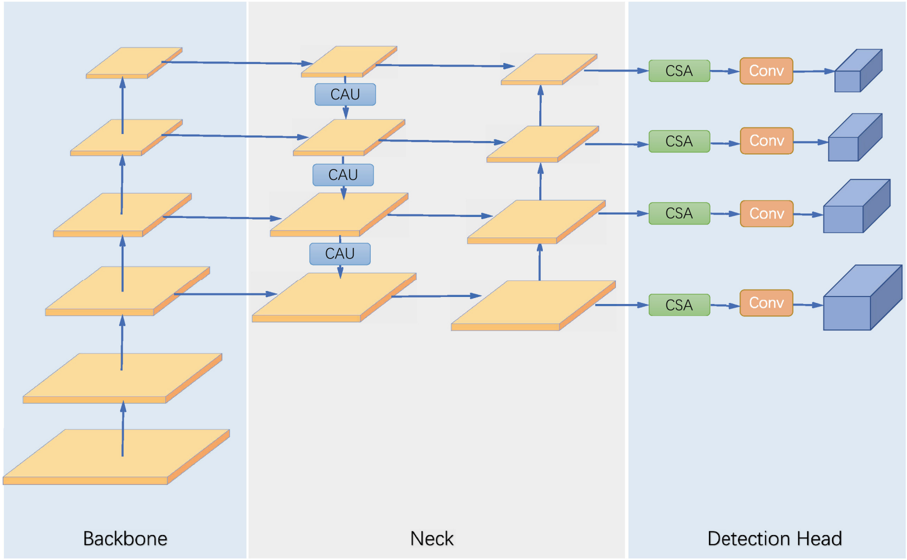

Fig. 1. Overall architecture.

3.2 Content-Aware Upsampling (CAU)

As shown in Fig. 2, CAU is divided into two branches: the upsampling kernel prediction branch and the multi-scale feature recombination branch. The former branch can automatically predict the upsampled kernel corresponding to the object location. The latter branch can extract multi-scale features for robust representation. We predict the upsampling kernel and then use the multi-scale feature recombination module to complete the upsampling operation. Given an input feature map F of H × W × C and an upsampling rate r, the size of the output feature map F' will be rH × rW × C. For any object location $l' = (i', j')$ of the output F', there is a corresponding source location $l = (i, j)$ at the input F, where $i = \lfloor i'/r \rfloor$, $j = \lfloor j'/r \rfloor$.

Upsampling Kernel Prediction. We use a content-aware method to predict the upsampled kernel. Each position on the feature map F corresponds to the

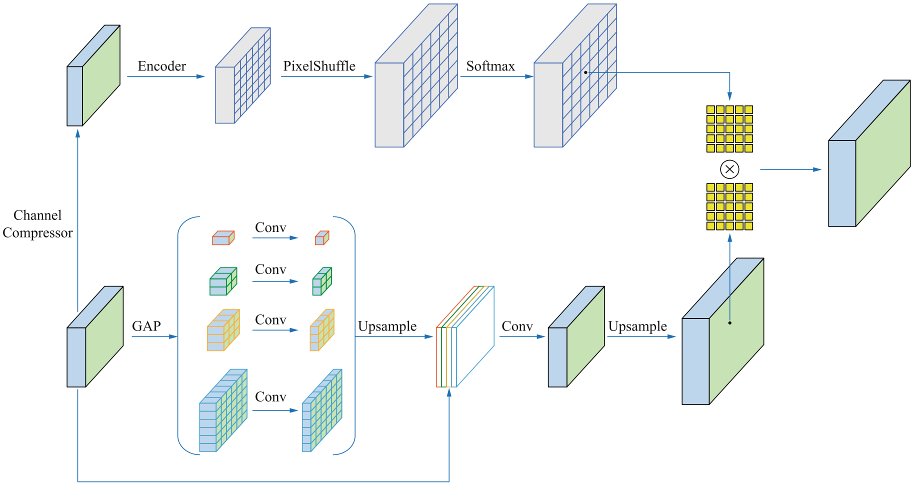

Fig. 2. The Content-Aware Upsampling (CAU) module. The top is the upsampling kernel prediction branch, and the bottom is the multi-scale feature recombination branch.

r^2 object position in the feature map F'. Each object position needs a $K_u \times K_u$ upsampling kernel. For the input feature map F, we first use a 1×1 convolution to compress its channels to reduce the computational burden. We use the convolutional layer of kernel size K_e to generate the upsampled kernel, where $K_e = K_u - 2$. The size of the output upsampled kernel is H × W × $r^2 K_u^2$. Then pixelshuffle [20] is to activate the corresponding subpixels periodically during convolution according to different subpixel positions of the low-resolution feature map to complete the construction of the high-resolution feature map. Finally, the size of the final output upsampled kernel is rH × rW × K_u^2, and the softmax function is applied to each $K_u \times K_u$ for normalization. As shown in Eq. (1), the upsampling kernels at different positions are predicted adaptively.

$$K_{l'} = \psi(N(F_l, K_e)) \tag{1}$$

where the kernel prediction module ψ predicts a location-wise kernel $K_{l'}$ for each location l' based on the neighbor of F_l. Here we denote N(F_l, K_e) as the $K_e \times K_e$ sub-region of F centered at the location l, i.e., the neighbor of F_l.

Multi-scale Feature Recombination. For the input feature map F, we divide the feature map into different sub-regions through adaptive pooling operations, obtaining 1×1, 2×2, 3×3, and 6×6 feature maps. Then we perform 1×1 convolution on each feature map to reduce the number of channels to $\frac{1}{4}$ of F. The sub-region features are then upsampled and concatenated with the input features, which aggregate global context information of different scales to improve small object detection,the output feature map f. As shown in Eq. (2), for each position l' in the output feature map, we map it to the feature map f_l and dot product the region centered by $K_u \times K_u$ with $K_{l'}$ to obtain the output value.

$$F'_{l'} = \phi(N(f_l, K_u), K_{l'}) \tag{2}$$

where ϕ is the multi-scale content-aware reassembly module that reassembles the neighbor of f_l with the kernel $K_{l'}$.

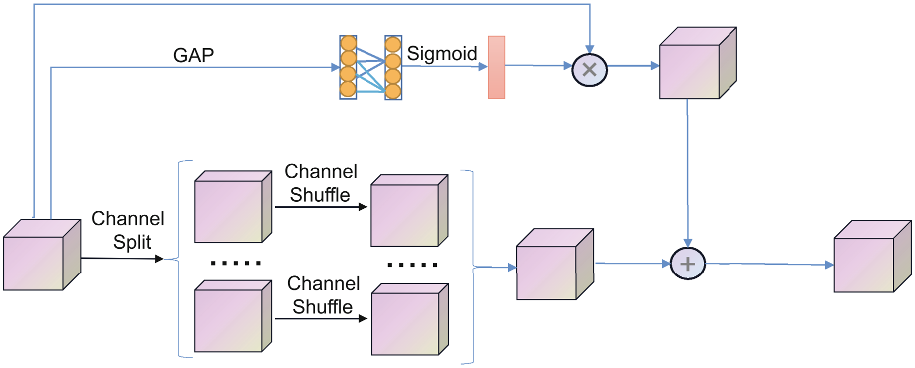

Fig. 3. The Channel Shuffle Attention (CSA) module.

3.3 Channel Shuffle Attention (CSA)

As shown in Fig. 3, we introduce the CSA module, which combines feature reconstruction and attention mechanism. Through feature reconstruction, the network can effectively suppress background noise interference; By adding the attention mechanism, the network can focus on critical objects and filter out useless information. CSA is mainly divided into two branches. In the bottom feature reconstruction branch, CSA divides the channels of the input feature X into multiple groups and performs corresponding interception operations. Then channel shuffle is used for each group to enhance the feature robustness. This branch can disrupt the internal relationship between features and force the network to learn more subtle features. In the top attention branch, global average pooling is carried out on the input feature X to aggregate global spatial information. As shown in Eq. (3), the 1-dimensional convolution operation is carried out to capture the local cross-channel information interaction, where we use the sigmoid activation function to calculate the weight of each channel. This way, we can extract the dependencies between channels, improving small object detection. As shown in Eq. (4), the output feature M' is obtained by multiplying the weights with the original input features. As shown in Eq. (5), the final feature M'' is obtained via adding the top and bottom output.

$$W = \sigma(Conv1D(GAP(X))) \tag{3}$$

$$M' = W \cdot X \tag{4}$$

$$M'' = \Phi(M', X') \tag{5}$$

4 Experiments

4.1 Datasets

MS COCO. [14] Microsoft COCO (MS COCO) is a widely used dataset for object detection, including 118k training images, 5k validation images, and 40k test images. MS COCO has 80 categories, approximately containing 41% small objects, 34% medium objects, and 24% large objects. MS COCO is challenging because small objects account for a large proportion with complex background. In this paper, we set the input image size as 640×640.

Tsinghua-Tencent 100K (TT-100K). [32] TT-100K is a traffic sign detection dataset, containing 100k high-resolution (2048×2048) images and 30k traffic signs under various weather and lighting conditions. Same as the previous work [5], we remove the categories with a few samples and only keep 45 categories of more than 200 categories. To reduce computation and memory overflow, we crop the original image to the size of 1280×1280. TT-100K can be divided into three scales, i.e., the area lower than 32×32 as the small-sized object, the area between 32×32 and 96×96 as the middle-sized object, and the area greater than 96×96 as the large-sized object.

4.2 Evaluation Metrics

Same as the evaluation metrics in the MS COCO competition [14], we use Average Precision (AP) to evaluate the detection performance, which is a comprehensive metric of precision and recall. AP is calculated as Eq. (6), where p represents precision, and r represents recall. As shown in Eq. (7), mean Average Precision (mAP) denotes average AP of multiple categories, where n represents the number of categories. Moreover, we use Floating Point Operations (FLOPs) to measure the computation complexity, which represents the total number of calculation operations of a detection model.

$$AP = \int_0^1 p(r)dr \tag{6}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{7}$$

4.3 Implementation Details

We conduct all the experiments with Pytorch 1.11.0 and CUDA 11.3. All the networks are trained with 4 NVIDIA GeForce 2080Ti GPUs. We use the SGD optimizer to train the models for 300 epochs. The initial learning rate is set to 0.01, and the cosine annealing strategy is used to reduce the learning rate. The weight decay, batch size, and momentum are set to 0.0005, 8, and 0.937, respectively.

4.4 Ablation Study

As shown in Table 1, we perform ablation experiments on the TT-100K dataset to verify the proposed CAU and CSA modules. We use YOLOv5l6 as the baseline detection model. Experiments prove that CAU and CSA can improve the detection performance of all sizes with little computation added, achieving 1.7% mAP improvement compared with the baseline model. Specially, the proposed method can sinificantly improve small object detection, 2.2% higher than YOLOv5l6. CAU can improve the AP of small objects by 1.9%, which proves its ability to reduce information loss.

Table 1. Ablaton study on TT-100K

Method	AP_s	AP_m	AP_l	AP_{50}	mAP	params	GFLOPs
YOLOv5l6	63.9	81.3	87.1	95.5	76.5	76.5M	**110.8**
YOLOv5l6+CAU	65.8	81.9	88.2	95.9	77.5	80.5M	116.6
YOLOv5l6+CAU+CSA	**66.1**	**82.8**	**88.5**	**96.4**	**78.2**	84.9M	121.9

4.5 Comparative Results

As shown in Table 2, we compare our method with other state-of-the-art detectors on the TT-100K dataset, including prestigious one-stage SSD [17], RetinaNet [13], YOLO [1,18] and two-stage Faster R-CNN [19], FPN [12]. Our proposed method can achieve the best detection performance for all evaluation metrics. Especially for the small object, our method can acquire a large performance gain. Moreover, Table 3 shows the detection results on MS COCO. Due to the

Table 2. Comparison with state-of-the-art methods on TT-100K.

Method	AP_s	AP_m	AP_l	AP_{50}	mAP
Faster R-CNN [19]	50.0	82.0	88.0	90.3	72.1
Cascade R-CNN [2]	55.7	85.4	90.4	92.5	74.9
FPN [12]	40.6	63.7	63.0	43.5	64.1
SSD [17]	25.3	67.8	81.5	83.5	59.8
RetinaNet [13]	60.9	79.5	81.2	92.4	70.9
Efficientdet-D0 [22]	-	74.4	83.6	85.4	66.7
YOLOv3 [18]	60.4	76.7	81.1	91.6	70.2
YOLOv4 [1]	62.8	79.8	85.4	94.7	74.9
YOLOv5l6	63.9	81.3	87.1	95.5	76.5
Ours	**66.1**	**82.8**	**88.5**	**96.4**	**78.2**

limited computing resources, we use YOLOv5s as the baseline model[1], replacing the upsampling module with CAU in the detection neck and adding CSA to the detection head. Our method achieves the best detection performance, proving its effectiveness and generality.

Table 3. Comparison with state-of-the-art methods on MS COCO.

Method	AP_s	AP_m	AP_l	AP_{50}	mAP
Faster R-CNN [19]	15.7	35.8	44.3	55.2	34.1
Cascade R-CNN [2]	19.6	38.9	48.0	55.8	36.8
Deformable R-FCN [4]	19.4	40.1	52.5	58.0	37.5
CoupleNet [31]	11.6	36.3	50.1	53.5	33.1
CornerNet [11]	17.0	39.0	50.5	53.7	37.8
RefineDet [29]	16.5	38.8	51.5	57.0	36.1
DSSD513 [7]	13.0	35.4	51.1	53.3	33.2
RetainNet [13]	18.5	37.2	45.4	53.4	35.1
DeNet [24]	12.3	36.1	50.8	53.4	33.8
YOLOv3 [18]	20.5	39.3	45.9	54.8	35.1
YOLOv5s	21.1	42.3	47.5	56.7	37.1
Ours	**21.9**	**42.9**	**47.8**	**57.1**	**37.5**

4.6 Qualitative Results

In Fig. 4, we visualize the feature map after using CAU and CSA. Compared with the original detection neck, CAU can highlight the features of small traffic signs, which proves CAU can enhance feature representation. Compared with the original detection head, CSA can effectively suppress background noise interference and extract robust features. This way, our method can improve small object detection.

Figure 5 shows some detection examples of TT-100K and MS COCO. Our method can accurately detect objects, especially for small objects.

[1] Please refer to https://github.com/ultralytics/yolov5. For TT-100K, we use the large YOLOv5l6 as the baseline model. For MS COCO, we use the small YOLOv5s as the baseline model. Except for CSA and CAU, our model is the same as the official model. The parameters of YOLOv5l6 are about 10× of YOLOv5s.

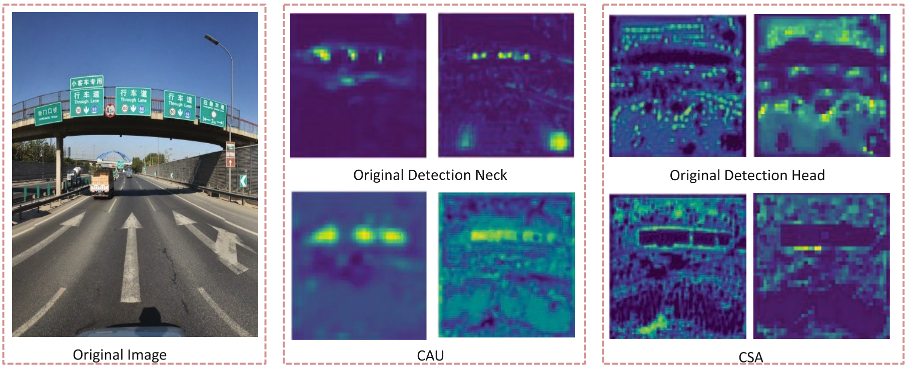

Fig. 4. Feature visualization. CAU can enhance object features. CSA can suppress background noises for robust feature extraction.

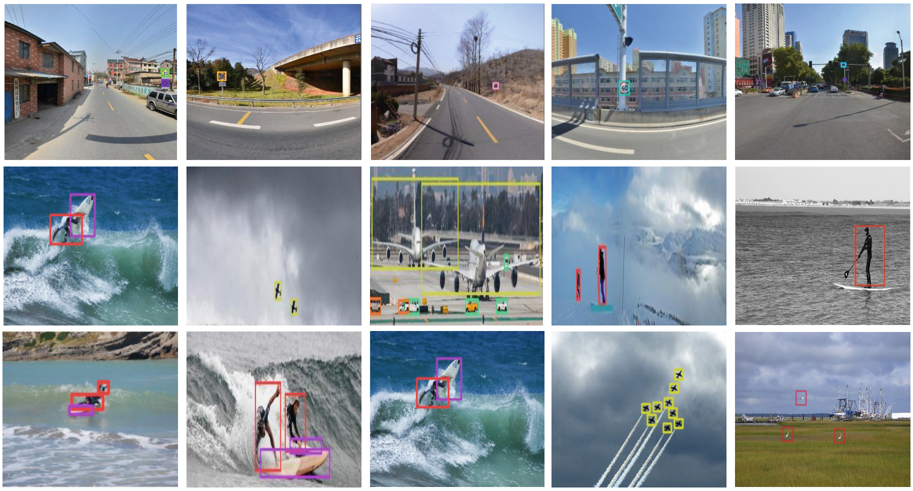

Fig. 5. Qualitative results on TT-100K (1st row) and MS COCO (2nd & 3rd rows)

5 Conclusion

This paper proposes a feature enhancement module CAU and a feature reconstruction module CSA for small object detection. CAU can enhance feature representation with less information loss, and CSA can suppress background noise interference to extract robust features. Comprehensive experiments prove that our method can improve object detection, especially for small objects. In the future, we will validate the generality of the proposed modules with more networks.

Acknowledgment. The research is supported by National Key Research and Development Program of China (2020AAA0109701), National Natural Science Foundation of China (62076024, 62006018).

References

1. Bochkovskiy, A., Wang, C.-Y., Mark Liao, H.-Y.: YOLOv4: optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
2. Cai, Z., Vasconcelos, N.: Cascade R-CNN: delving into high quality object detection. In: CVPR (2018)
3. Chen, L.-C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
4. Dai, J., et al.: Deformable convolutional networks. In: ICCV, pp. 764–773 (2017)
5. Deng, C., Wang, M., Liu, L., Liu, Y., Jiang, Y.: Extended feature pyramid network for small object detection. IEEE Trans. Multimedia **24**, 1968–1979 (2021)
6. Fang, P., Zhou, J., Kumar Roy, S., Petersson, L., Harandi, M.: Bilinear attention networks for person retrieval. In: ICCV, pp. 8029–8038 (2019)
7. Fu, C.-Y., Liu, W., Ranga, A., Tyagi, A., Berg, A.C.: DSSD: deconvolutional single shot detector. arXiv preprint arXiv:1701.06659 (2017)
8. Gao, Z., Xie, J., Wang, Q., Li, P.: Global second-order pooling convolutional networks. In: CVPR, pp. 3024–3033 (2019)
9. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
10. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR, pp. 7132–7141 (2018)
11. Law, H., Deng, J.: CornerNet: detecting objects as paired keypoints. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 765–781. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_45
12. Lin, T.-Y., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. In: CVPR, pp. 936–944 (2017)
13. Lin, T.-Y., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV, pp. 2999–3007 (2017)
14. 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
15. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: CVPR, pp. 8759–8768 (2018)
16. Liu, S., Huang, D., Wang, Y.: Receptive field block net for accurate and fast object detection. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11215, pp. 404–419. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01252-6_24
17. 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
18. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767, 2018

19. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017)
20. Shi, W., et al.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR, pp. 1874–1883 (2016)
21. Szegedy, C., et al.: Going deeper with convolutions. In: CVPR, pp. 1–9 (2015)
22. Tan, M., Pang, R., Le, Q.V.: EfficientDet: scalable and efficient object detection. In: CVPR, pp. 10778–10787 (2020)
23. Tian, Z., He, T., Shen, C., Yan, Y.: Decoders matter for semantic segmentation: data-dependent decoding enables flexible feature aggregation. In: CVPR, pp. 3126–3135 (2019)
24. Tychsen-Smith, L., Petersson, L.: Denet: scalable real-time object detection with directed sparse sampling. In: ICCV, pp. 428–436 (2017)
25. Wang, J., et al.: CARAFE: content-aware reassembly of features. In: ICCV, pp. 3007–3016 (2019)
26. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: ECA-Net: efficient channel attention for deep convolutional neural networks. In: CVPR, pp. 11531–11539 (2020)
27. Ying, X., et al.: Multi-attention object detection model in remote sensing images based on multi-scale. IEEE Access **7**, 94508–94519 (2019)
28. Zhang, Q.-L., Yang, Y.-B.: SA-Net: shuffle attention for deep convolutional neural networks. In: ICASSP, pp. 2235–2239 (2021)
29. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. In: CVPR, pp. 4203–4212 (2018)
30. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR, pp. 6230–6239 (2017)
31. Zhu, Y., Zhao, C., Wang, J., Zhao, X., Wu, Y., Lu, H.: CoupleNet: coupling global structure with local parts for object detection. In: ICCV, pp. 4146–4154 (2017)
32. Zhu, Z., Liang, D., Zhang, S.-H., Huang, X., Li, B., Hu, S.-M.: Traffic-sign detection and classification in the wild. In: CVPR, pp. 2110–2118 (2016)