CESNet: Cross-dimensional information extraction and channel sharing

Qian Long[1], Gaihua Wang[1,2*], Kehong Li[1*]

[1] Tianjin University of Science and Technology, Tianjin, 300457, China
[2] Hubei Key Laboratory of Optical Information and Pattern Recognition, Wuhan Institute of Technology, Wuhan, 430205, China
wanggh@tust.edu.cn
1061423701@qq.com

Abstract. To improve detection accuracy, it proposes cross-dimensional information extraction and channel sharing (CESNet). The cross-dimensional information extraction(CE) module uses max pooling and average pooling to strengthen important features in different dimensions, and then interacts across channels to focus on regions of interest. Channel sharing(CS) module of involution, group convolution and efficient channel attention for deep convolutional neural networks(ECA-Net). It can reduce the loss of semantic information caused by channel reduction during feature fusion. Experiments show that the proposed method can work on different networks. Among them, the accuracy of CESNet reaches 34.1% in box AP on the COCO dataset. And the detection performance of our network is better than other networks.

Keywords: Deep learning, Object detection, CE module, CS module.

1 INTRODUCTION

Object detection is a prerequisite for advanced vision tasks and has been applied in different tasks, such as intelligent video surveillance [1], content-based image detection [2], robot navigation [3], and augmented implementation[4]. However, traditional object detection [5-7] needs manual extraction, design, and training. And it is difficult to obtain robust features.

With the development of deep learning, object detection based on convolutional neural networks (CNN) has been widely used. It consists of three parts: backbone network, feature pyramid and detection head. The backbone network is used for the feature extraction of images. The SSD [8] algorithm uses the VGG [9] as the backbone to extract features, which effectively reduces the parameters of the network. However, the network depth is only 19 layers, it does not sufficiently extract the features and the detection accuracy of the algorithm is not high. YOLO-V3 [10] proposes the Darknet53, which can balance the accuracy of the algorithm and the number of parameters. Resnet [11] network proposes that the deeper the network, the more sufficient the image feature extraction is. Therefore, most algorithms use Resnet50 or Resnet101 to extract features. However, feature extraction of Resnet requires large capacity, slow running speed, and is not suitable for mobile terminals. In order to transplant the object detection to the mobile terminal, Ghostnet [12] proposes to de-redundant features to achieve network lightweight.

Feature pyramid is used to address the multi-scale object detection problem. Lin et al. [13] propose the FPN structure, which combines fine-grained spatial information from shallow feature maps and deep semantic information. PANet [14] fuses feature maps by upsampling and downsampling to reconstruct pyramids with enhanced spatial information. NAS-FPN [15] proposes to fuse feature maps of different scales by using neural network structure search. BiFPN[16] allows the network to learn the importance of different input features by weighted fusion of features. Recursive-FPN [17] feeds the fused feature map to the backbone network for feature fusion again.

The detection head is used to achieve object classification and position regression. According to the detection head, the object detection algorithm is mainly divided into the one-stage algorithm and the two-stage algorithm. The two-stage algorithm first generates regions containing objects, and then classifies and regresses the candidate regions. A typical two-stage algorithm such as Faster-RCNN [18], proposes an RPN network to replace the traditional method, generates regional proposals, and then collects the input feature maps and proposals through Roi Pooling. After synthesising this information, the proposal feature maps are extracted and sent to subsequent fully connected layers, which can determine the exact location of the object category and detection frame. The one-stage algorithm directly gives the final detection result, and there is no obvious step for generating candidate boxes. FCOS [19] is a one-stage object detection algorithm that predicts objects pixel by pixel. This algorithm completely avoids complex calculations related to anchors by eliminating pre-defined anchors, thereby reducing model complexity. Since the two-stage algorithm first generates candidate regions, and then classifies and refines the candidate regions, it has higher detection accuracy and slower detection speed than the one-stage algorithm.

To achieve both high detection speed and high detection accuracy in object detection models, many researchers have conducted in-depth studies on one-stage object detection models. Zhang et al. [20] proposed an IOU-aware Classification Score and Varifocal Loss, which improves the accuracy of FCOS by 2%. Zhou et al. [21] proposed replacing sparse pseudo-boxes with dense predictions as a unified and intuitive form of pseudo-label, and introduced a region selection technique, leading to further improvements in the accuracy of FCOS. Liu et al. [22] proposed Joint Confidence Estimation and Task-Separation Assignment to address the issue of inaccurate selection and assignment in one-stage detectors, resulting in improved accuracy.

Zhang et al. proposed ATSS [23], which is a one-stage algorithm. It uses Resnet to extract features, and reduces feature maps of different sizes to the same channels, then uses FPN for feature fusion. The deeper network not only increases the complexity of the network but also causes semantic information redundancy. The reduction of channel dimension also causes information loss. In this paper, we propose CESNet based on ATSS. The main contributions of the algorithm are as follows:

(1) It proposes a CE module, which uses average pooling and max pooling to extract weights of different dimensions, and then fuses across channels. The method can strengthen the network's attention to important features and regions of interest.

(2) It proposes a CS module which consists of involution, group convolution and ECA-Net [24]. Involution can enhance channel independence and spatial invariance. By channel grouping, group convolution can reduce the number of parameters. Through the extraction of feature weights, ECA-Net strengthens important features.

(3) The experimental results show that it has little effect on the complexity of the network, but significantly improves the network accuracy.

2 RELATED WORK

2.1 Attention Mechanism

In recent years, the attention mechanism has been widely used to enhance the ability of feature extraction. It is divided into channel attention mechanisms, spatial attention mechanisms and hybrid attention mechanisms. The channel attention mechanism aims to show the correlation between different channels, strengthening the important features and suppressing useless information. Spatial attention aims to improve the feature representation of key regions, enhance specific target regions of interest while weakening irrelevant background regions. The hybrid attention mechanism combines the channel and the spatial attention mechanism.

In 2017, SENet [25] performs the Squeeze and Excitation operation to obtain the weights of different channels, and then multiply by the original feature map to get the final outputs. In 2018, CABM [26] adopted the parallel pooling of avg \& max, which proves that using average pooling and max pooling in parallel is better than using single pooling for detection. In 2019, SK-Net [27] introduced multiple parallel convolution kernel branches with different receptive fields to learn feature map weights at different scales, enabling the network to pick out more appropriate multi-scale features. In 2020, ECA-Net [24] uses one-dimensional convolution to replace the fully connected layer, and it realises the information interaction across channels. In 2021, Triplet [28] establishes the dependencies between dimensions through residual transformation and adopts three-branch cross-dimension interaction to achieve attention alignment.

In this paper, we propose a CE module. It first performs average pooling and max pooling for the two dimensions, respectively. Then it adopts dual-branch cross-dimensional interaction, which alleviates channel and space dependencies.

2.2 Involution

Convolution is widely used in convolutional neural networks due to its spatial invariance and channel independence. However, the features between channels are independent and the parameters are not shared. There is a lot of redundancy in the features. Extracting rich semantic spatial information requires large convolution kernels, which increases a lot of parameters and the complexity of the network. To solve these problems, Li et al. [29] proposed an involution, which makes the involution spatial independence and channel invariance by reversing the convolution. Compared with the convolutional neural network, it eliminates redundancy by sharing parameters between different channels and does not bring many parameters due to the increase of the convolution kernel.

Therefore, this paper proposes a CS module to replace the original convolution. The input obtains the channel independence and spatial invariance of the vector through the involution operation. Then it uses the group convolution to reduce the number of parameters and uses the 1*1 convolution to reduce the channel dimension. Finally, channel attention is enhanced through ECA-Net.

3 OUR APPROACH

It proposes CESNet based on the ATSS [23] network. The network structure is shown in Fig.1.

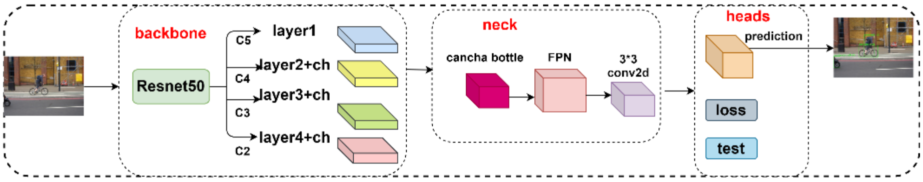


Fig. 1. Illustration of the proposed CESNet.

CESNet consists of a backbone, neck and heads. The backbone adopts Resnet50 with attention, which adds the hybrid attention to layer2, layer3 and layer4. The specific structure of the backbone is shown in Table 1. The Neck uses a CS module, then uses FPN for feature fusion, and finally uses 3*3 convolution to eliminate redundancy. Heads are used for object detection to achieve object classification and regression. It adopts FocalLoss, GIoULoss and CrossEntropyLoss for classification loss, regression loss and confidence loss.

Table 1. The structure of Resnet50 and our backbone.

Backbone	Resnet50	Resnet50+attention
Conv1	7*7, 64, stide=2 3*3, max, pool, stride=2	7*7, 64, stide=2 3*3, max, pool, stride=2
Layer1	[[1*1, 64], [3*3, 64], [1*1, 256] * 3]	[[1*1, 64], [3*3, 64], [1*1, 256] * 3]
Layer2	[[1*1,128], [3*3,128], [1*1,512]] *4	[[3*3,128], [attention], [1*1,512]] *1 [[1*1,128], [3*3,128], [1*1,512]] *3
Layer3	[[1*1,256], [3*3,256], [1*1,1024]] *6	[[1*1,256], [3*3,256], [attention], [1*1,1024]] *1 [[1*1,256], [3*3,256], [1*1,1024]] *5
Layer4	[[1*1,512], [3*3,512], [1*1,2048]] *3	[[1*1,512], [3*3,512], [attention], [1*1,2048]] *1 [[1*1,512], [3*3,512], [1*1,2048]] *2
output	layer1, layer2, layer3, layer4	layer1, layer2, layer3, layer4

3.1 CE MODULE

The CE module is shown in Fig.2. Let *X* donate the input feature map, its size is [*B*, *C*, *H*, *W*], where *B*, *C*, *H*, *W* indicates the batch size, channel size, spatial height, and width, respectively.

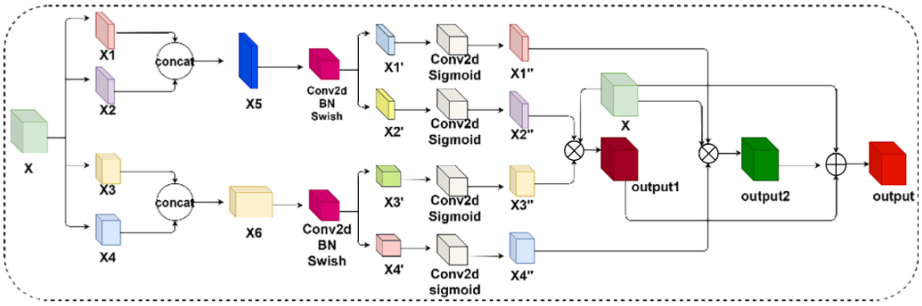


Fig. 2. The proposed CE module.

First, the weights of different dimensions are extracted by max pooling and average pooling respectively. The expression of pooling is shown in Equations 1, 2, 3 and 4.

$$X_1 = F_{AVGH}X,\ X_1 \in R^{[B,\ C,\ 1,\ W]} \tag{1}$$

$$X_2 = F_{MAXH}X,\ X_1 \in R^{[B,\ C,\ 1,\ W]} \tag{2}$$

$$X_3 = F_{AVGW}X,\ X_1 \in R^{[B,\ C,\ W,\ 1]} \tag{3}$$

$$X_4 = F_{MAX\ W}X,\ X_1 \in R^{[B,\ C,\ W,\ 1]} \tag{4}$$

Where F_{AVGH} indicates average pooling for spatial height dimensions, F_{MAXH} indicates max pooling for spatial height dimensions, F_{AVGW} indicates average pooling for width dimensions and F_{MAXW} indicates max pooling for width dimensions. Then X_1, X_2 and X_3, X_4 are spliced in different dimensions to obtain X_5 and X_6, Where *Concat* indicates concatenating in the same dimension.

After that, it uses 1*1 convolution to compress channels for X_5 and X_6 respectively and uses Batch Normalization and F_{Swich} to encode vertical and horizontal information. Then split them to get X_1', X_2', X_3' and X_4'. The expression is shown in Equations 5 and 6.

$$X_1',\ X_2' = F_{split}F_{Swish}F_{BN}W_aX_5,\ X_1',\ X_2' \in R^{[B,\ C,\ 1,\ W]} \tag{5}$$

$$X_3',\ X_4' = F_{split}F_{Swish}F_{BN}W_aX_6,\ X_3',\ X_4' \in R^{[B,\ C,\ H,\ 1]} \tag{6}$$

Where W_a indicates 1*1 convolution, FBN indicates Batch Normalization, F_{Swish} indicates the Swish activation function, and F_{split} indicates the split operation. Then the dimension weights X_1'', X_2'', X_3'' and X_4'' are obtained by 1*1 convolution and Sigmoid activation function.

Finally, in order to capture the cross-channel information and increase the rich semantic features, the different dimension weights are multiplied by the input. The expression is shown in Equations 7, 8 and 9.

$$output1 = X''\,2 \odot X''\,3 \odot X,\ output1 \in R^{[B,\ C,\ H,\ W]} \tag{7}$$

$$output2 = X''\,1 \odot X''\,4 \odot X,\ output2 \in R^{[B,\ C,\ H,\ W]} \tag{8}$$

$$output = output1 + output2 + X,\ output \in R^{[B,\ C,\ H,\ W]} \tag{9}$$

Where $\odot$ indicates multiplication.

3.2 CS MODULE

The CS module is shown in Fig.3. Let X donate the input feature map, its size is [*B*, *C*, *H*, *W*].

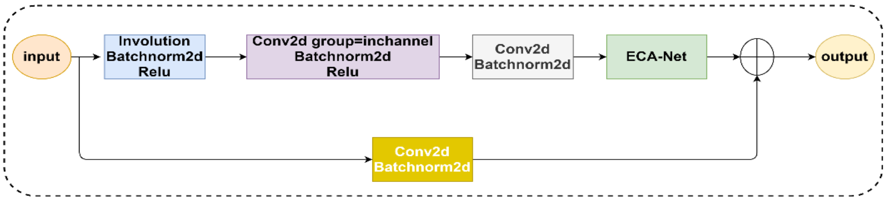


Fig. 3. The proposed CS module.

In order to increase the channel sharing of feature vectors. First, it operates on the input using involutions without changing the dimensionality of the input. Next, it uses group convolution, which not only reduces the number of parameters, but also further extracts features, and the output vector size is [*B*, *C*, *H*, *W*]. To reduce the number of channels of the input vector, it uses 1*1 convolution. And the output vector size is [*B*, 256, *H*, *W*]. Then we use ECA-Net to strengthen the network's attention to important features. The skip connection is added to the main branch. The final output is [*B*, 256, *H*, *W*].

4 EXPERIMENTS

This part of the experiment is to validate our proposed module and compare our proposed network. Before that, we introduce the datasets, experimental environment and experimental strategy used in the experiment.

Datasets: The PASCAL VOC datasets use PASCAL VOC 2007 and PASCAL VOC 2012. They have a total of 21 categories, 16551 training images and 16492 testing images. The MS COCO2017 dataset has a total of 80 categories and 118,287 images. It covers the most common objects in life and is a rich object detection dataset.

Experimental environment: CPU: Intel Xeon E5-2683 V3@2.00GHz; RAM: 32 GB; Graphics card: Nvidia GTX 1080Ti; Hard disk: 500GB. Software: MMdetection2.6; PyTorch1.6.0; Torchvision=0.7.0; CUDA10.0; CUDNN7.4.

Experimental strategy: It crops all the images to 512*512 for training, uses the SGD optimizer, and sets the learning rate to 0.001, momentum to 0.9, and weight decay to 0.0001. The learning rate adopts a step adjustment strategy, and the iteration period is 12 epochs. PASCAL VOC datasets adopt mAP as the evaluation index. MS COCO2017 dataset adopts average precision (Average-Precision, AP), AP_{50}, AP_{75}, AP_S, AP_M, and AP_L to evaluate the detection accuracy.

4.1 Ablation study

In this section, ablation experiments will be performed on the PASCAL VOC datasets and MS COCO 2017 dataset. The CE module experiments and CE module experiments test the influence of the CE module and CS module on different networks.

CE module experiments

Although Resnet101 has higher accuracy, it requires more time and memory for training. Considering our equipment, all backbones use Resnet50. And the neck adopts FPN. In order to verify the effectiveness of the CE module, we conducted comparative experiments on four different networks. The experimental results are shown in Table 2 and Table 3.

As shown in Table 2. The AP of FCOS increased by 0.5%, and it improved significantly on AP_M and AP_L, which increased by 0.8% and 1.2%, respectively. The AP of VFNet increased from 33.6% to 34.7%. Although its AP_L decreased by 0.4%, its small objects improvement effect was obvious, AP_S increased by 1.4%, and AP_M increased by 1.6%. The AP of FoveaBox has increased from 28.5% to 29.2%, and its effect on large objects is obvious. AP_M increases from 32.2% to 33.4%, and AP_L increases from 43.8% to 45.1%. The AP of ATSS increased by 0.8%, and AP_M increased especially, from 35.3% to 37.3%. This illustrates the effectiveness of our proposed attention mechanism. It can help the backbone network to fully extract the semantic features of the image and improve the detection accuracy of the algorithm. The structure extracts weights through average pooling and max pooling, which can maximize the network's attention to important features and weaken the attention to non-important features. The mutual interaction of height and width further enhances the spatial attention to the object's area of interest.

Table 2. The influence on MS COCO2017 dataset of CE module different networks. × indicates that there is no attention mechanism. √ indicates that there is a CE module.

Model	Attention	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
FCOS[19]	×	28.7	45.3	29.9	10.2	32.1	44.1
FCOS[19]	√	**29.2**	**46.1**	**30.4**	**10.5**	**32.9**	**45.3**
VFNet[20]	×	33.6	49.9	34.9	13.4	36.8	**51.9**
VFNet[20]	√	**34.7**	**50.4**	**37.1**	**14.8**	**38.4**	51.5
FoveaBox[30]	×	28.5	46.4	29.9	10.3	32.2	43.8
FoveaBox[30]	√	**29.2**	**47.1**	**30.4**	**10.5**	**33.4**	**45.1**
ATSS[23]	×	32.1	48.7	34.1	13.2	35.3	**50.1**
ATSS[23]	√	**32.9**	**49**	**35.1**	**14.1**	**37.3**	48.9

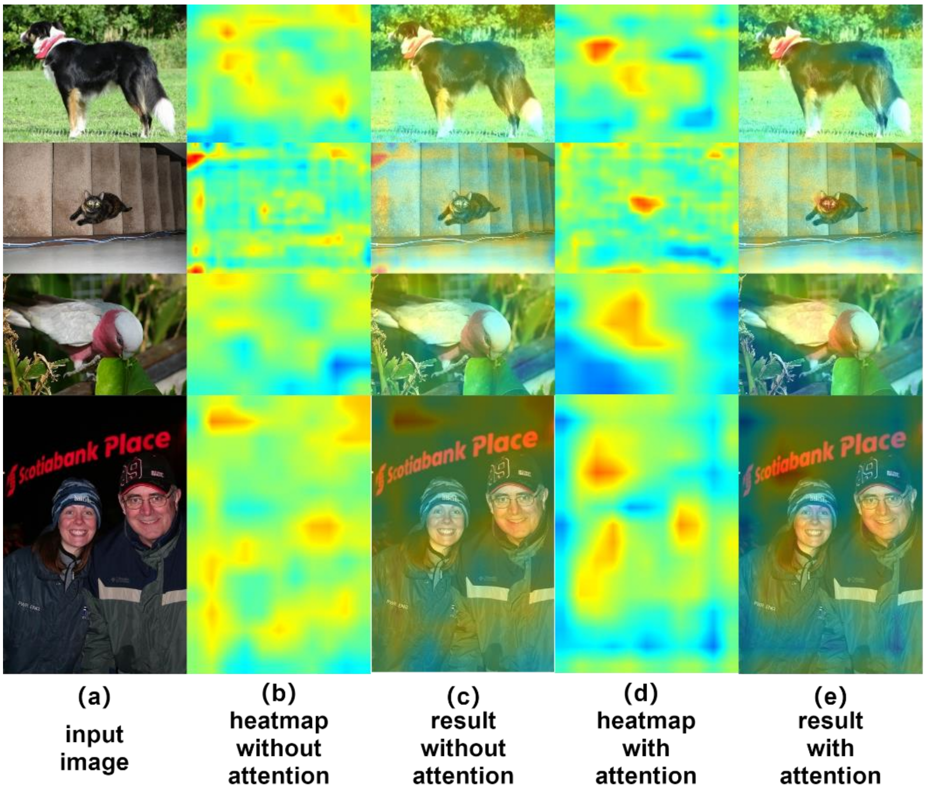
For PASCAL VOC datasets, as shown in Table 3. The mAP of ATSS improves from 76.4% to 77.3%. FoveaBox's mAP improved by 0.6%. The improvement effect of FCOS is obvious, and its mAP has increased from 73.1% to 74.8%. The mAP of VFNet is improved from 77.1% to 77.4%. And it also has a great improvement for different categories, such as bicycle, bottle, cat, and dog. Although it degrades in some categories, such as aeroplane and tvmonitor, it is mainly due to the lack of samples or incomplete objects in these categories. This proves that the attention mechanism can also improve the detection accuracy of the network on PASCAL VOC datasets, further confirming the effectiveness of the structure.

Table 3. The influence on PASCAL VOC datasets of CE module on different networks. × indicates that there is no attention mechanism. √ indicates that there is a CE module.

Class	ATSS[23]		FoveaBox[30]		FCOS[19]		VFNet[20]	
	×	√	×	√	×	√	×	√
aeroplane	79.7	79	79.5	78.6	78.7	77.6	83	81
bicycle	81.3	**83.4**	80.9	**82.4**	78.6	**80.2**	82.1	**83**
bird	79.3	**80.4**	77.8	**78**	79.9	78.4	78.9	**79.6**
boat	67.7	**68.6**	64.8	**66.9**	66.2	66	69.6	69.2
bottle	60.6	**61.6**	61.9	**64.1**	58.1	**59**	62.5	62
bus	81.3	80.9	81.9	81.3	79.4	**81.5**	82.5	**83.7**
car	85.9	**86.1**	84.4	84.4	83.6	**84.1**	85.8	85
cat	86.8	**88**	88.6	87.5	86.8	**87.5**	86.7	88
chair	61.8	59.4	59.5	**59.6**	58.1	**58.9**	59.6	**60.7**
cow	80.8	**81.5**	80.1	**83**	74.8	**79**	82.5	**84.2**
diningtable	66.4	**68.3**	66.2	**70.8**	61.1	**63.9**	65.7	**70**
dog	84.8	**86.8**	85.7	85.2	83.9	**85.6**	84.8	**85.6**
horse	84.3	**85.1**	82.5	**84.7**	71.6	**77.5**	85.5	85.5
motorbike	81.6	**84.2**	81.7	80.9	73.6	**77**	82.2	80.2
person	81.7	**82.2**	81.4	**81.5**	79.3	**79.6**	81.9	81.7
pottedplant	46.2	**48.8**	49.2	**52.6**	47.4	**51.1**	50.5	48.8
sheep	77.7	**82.3**	79.2	**79.4**	76.7	**77.7**	82	81
sofa	76.7	74.5	69	**71.6**	69.6	**72.9**	72.3	**75.1**
train	85.2	83.9	82.3	78.3	78.5	**81.5**	84.8	**86.3**
tvmonitor	78.4	**80.5**	76.7	74.5	76.7	76.6	78.4	77.5
mAP /%	76.4	**77.3**	75.7	**76.3**	73.1	**74.8**	77.1	**77.4**

It compares the effect of the CE module on feature visualization. In Fig.4, column (a) represents the input image, and columns (b) and (d) represent the heat without the CE module and with the CE module, respectively. Columns (c) and (e) represent the heat map acting on the input image.

Fig. 4. Feature visualization on MS COCO2017 dataset.



From the two columns (b) and (d), we can clearly find that when the attention mechanism is not added, the high spots in the heat map are scattered. After adding the attention mechanism, the highlights in the heat map focus on the target of the original image. This shows that the CE module can help the network strengthen the extraction of important features and pay more attention to the region of interest in the image.

CS module experiments

In order to study the effect of the CS module on detection accuracy. It conducts CS module ablation experiments on 4 different networks. The experimental results are shown in Table 4 and Table 5.

As shown in Table 4. The mAP of ATSS improved by 1.3%. FoveaBox's mAP increased from 75.7% to 76.6%. The mAP of FCOS increased by 1.5% from 73.1%. The mAP of VFNet increased from 77.1% to 77.9%. And it has great improvement for common types such as bicycle, bird, bus, car, cat, dog and person. This proves that the CS module can also improve the detection accuracy of the network on the PASCAL VOC dataset, further confirming the effectiveness of the structure.

Table 4. The influence on PASCAL VOC datasets of CS module on different networks. Conv indicates that there is a 1*1 convolution. Re indicates that there is a CS module.

Class	ATSS[23]		FoveaBox[30]		FCOS[19]		VFNet[20]	
	Conv	Re	Conv	Re	Conv	Re	Conv	Re
aeroplane	79.7	**80**	79.5	79.5	78.7	**79.2**	83	80
bicycle	81.3	**84.4**	80.9	**84.7**	78.6	78.5	82.1	**82.7**
bird	79.3	**81.5**	77.8	**80.7**	79.9	**81.6**	78.9	**80.8**
boat	67.7	**71.6**	64.8	**66.5**	66.2	**70.2**	69.6	**70.5**
bottle	60.6	**66.1**	61.9	**63.9**	58.1	**59.2**	62.5	**64**
bus	81.3	**82.4**	81.9	80.6	79.4	**80.1**	82.5	**83.5**
car	85.9	**86.2**	84.4	**85.9**	83.6	**84.8**	85.8	**86.3**
cat	86.8	**87.7**	88.6	87.7	86.8	**87.2**	86.7	**88.7**
chair	61.8	60.7	59.5	**59.7**	58.1	**59.1**	59.6	**60.5**
cow	80.8	79.4	80.1	**80.8**	74.8	**78**	82.5	**83.9**
diningtable	66.4	**68**	66.2	**67.9**	61.1	**63.3**	65.7	**69.4**
dog	84.8	**85.4**	85.7	**86.2**	83.9	**84.1**	84.8	**86.8**
horse	84.3	**84.8**	82.5	**83**	71.6	**78**	85.5	**85.7**
motorbike	81.6	**83.5**	81.7	80.6	73.6	**75**	82.2	81.5
person	81.7	**82.4**	81.4	**81.9**	79.3	**80.2**	81.9	**82.4**
pottedplant	46.2	**51.5**	49.2	**52.2**	47.4	**48.9**	50.5	**50.9**
sheep	77.7	**79.4**	79.2	**80.5**	76.7	76.3	82	**82.3**
sofa	76.7	76	69	**71**	69.6	**72.5**	72.3	**75.4**
train	85.2	83.2	82.3	81.8	78.5	**78.9**	84.8	**85.6**
tvmonitor	78.4	**78.9**	76.7	**78**	76.7	**77.9**	78.4	77.3
mAP /%	76.4	**77.7**	75.7	**76.6**	73.1	**74.6**	77.1	**77.9**

As Table 5 shows. The AP of FCOS increased by 0.5%, AP$_S$, AP$_M$ and AP$_L$ increased by 0.5%, 0.6% and 0.8%, respectively. The AP of VFNet increases from 33.6% to 34.1%, AP$_S$ increases by 1% and AP$_M$ increases by 0.9%. The AP of FoveaBox is improved by 0.9%, and it has a significant improvement for objects of different sizes, AP$_S$ is improved by 0.8%, AP$_M$ is improved by 1%, and AP$_L$ is improved by 1.5%. The AP of ATSS increased by 0.7%, and AP$_M$ increased from 35.3% to 36.4%. This shows that the residual-like structure can indeed improve the accuracy of the network. Directly reducing the channel dimension through the 1*1 convolution layer will result in a large loss of extracted features. Involution's channel sharing effectively reduces the loss of semantic information caused by the decline of network channels. The skip connection of the residual-like structure also increases the depth of the network, and the ECA-Net also strengthens the attention to important features and reduces redundancy.

Table 5. The influence on PASCAL VOC datasets of CS module on different networks. Conv indicates that there is a 1*1 convolution. Re indicates that there is a CS module. Fovea-Box[30] is a completely anchor-free object detection architecture.

Model	Neck	AP	AP_{50}	AP_{75}	AP_{S}	AP_{M}	AP_{L}
FCOS[19]	Conv	28.7	45.3	29.9	10.2	32.1	44.1
FCOS[19]	Re	**29.2**	**46**	**30.7**	**10.7**	**32.7**	**44.9**
VFNet[20]	Conv	33.6	49.9	34.9	13.4	36.8	**51.9**
VFNet[20]	Re	**34.1**	**50.1**	**36.3**	**14.4**	**37.7**	51.1
Fovea-Box[30]	Conv	28.5	46.4	29.9	10.3	32.2	43.8
Fovea-Box[30]	Re	**29.4**	**47.4**	**30.8**	**11.1**	**33.2**	**45.3**
ATSS[23]	Conv	32.1	48.7	34.1	13.2	35.3	**50.1**
ATSS[23]	Re	**32.8**	**49.4**	**35**	**13.5**	**36.4**	50.0

4.2 Compare with classic networks

It compares the impact of each module on the original network. From Table 6, the AP of the original network is only 32.1%. The AP of the network reaches 32.9% with the CE module. After adding the CS module, the AP of the network reaches 32.8%. After adding the CE module and CS module at the same time, the AP of the network reaches 34.1%, and its AP_{75} of, AP_{S} and AP_{M} are all the highest. This shows that the combination of two different modules can also effectively improve the accuracy of the network.

Table 6. The effect of different modules on the network. Attention indicates that there is a CE module. Re indicates that there is a CS module.

Model	Module	AP	AP_{50}	AP_{75}	AP_{S}	AP_{M}	AP_{L}
ATSS[23]	Res50	32.1	48.7	34.1	13.2	35.3	**50.1**
	Res50+ attention	32.9	49.0	35.1	14.1	37.3	48.9
	Res50+ Re	32.8	**49.4**	35.0	13.5	36.4	50.0
	Res50+ attention + Re	**34.1**	**52.3**	**35.3**	**15.5**	**38.4**	49.9

It tests the influence of different modules on the network complexity. As shown in Table 7. The original network parameters are 32.16M and the Flops is 51.75G. Its mAP in PASCAL VOC datasets is 76.4%. When the CE module is added, the amount of network parameters increases by 0.03M, the Flops increase by 0.01G, and the mAP increases by 0.9%. When the CS module is added, the amount of network parameters increases by 2.35M, the Flops increase by 1.82G, and the mAP increases by 1.3%. When these two modules are added at the same time, the amount of network parameters increases by 2.38M, the Flops increase by 1.83G, and the mAP increases by 1.5%.

Although these two modules increase the complexity of the network, the influence of the amount of parameters and the Flops on the network is acceptable.

Table 7. The influence of different modules on network parameters, calculation amount and mAP. attention indicates that there is a CE module. Re indicates that there is a CS module.

Model	Module	Parameter	Flops	mAP
ATSS[23]	Res50	32.16M	51.75G	76.4
	Res50+ attention	32.19M	51.76G	77.3
	Res50+ Re	34.51M	53.57G	77.7
	Res50+ attention + Re	34.54M	53.58G	**77.9**

It compares the proposed network with other classic networks. As shown in Table 8. Our network has a high AP of 34.1%, which is higher than all other networks in accuracy. And AP$_{50}$, AP$_{75}$, AP$_{S}$ and AP$_{M}$ are all the highest. Although its AP$_{L}$ is lower than that of ARSL, its accuracy is higher than other networks. Its AP$_{S}$ is as high as 15.5%, which shows that it has a good detection effect on small objects. SSD have the lowest Flops, which shows that they require the least amount of computational resources and can be well deployed in places with poor computational resources. Although our model is 55% higher in Flops, it improves by 33% in accuracy, which indicates that it can better detect the objects and reduce the security risk in real applications. Meanwhile, 53.58G also belongs to one of the lower computational resources, which can be widely applied.

Table 8. Comparison of the proposed method with other classic networks on the MS COCO2017 dataset. Retinanet[31] is a single-stage target detection model that proposes Focal Loss to solve the positive and negative category imbalance.

Model	AP	AP$_{50}$	AP$_{75}$	AP$_{S}$	AP$_{M}$	AP$_{L}$	Flops
Faster-RCNN[18]	28.3	45.0	30.4	12.0	31.0	41.7	187G
FCOS[19]	28.7	45.3	29.9	10.2	32.1	44.1	180G
SSD[8]	25.6	44.0	26.2	9.1	29	38.9	**34.36G**
Retinanet[31]	30.5	47.3	32.2	10.8	35.2	48.4	215G
Yolov3[10]	24.6	42.6	25.1	8.2	26.8	36.5	52.09G
FoveaBox[30]	28.5	46.4	29.9	10.3	32.2	43.8	185G
ATSS[23]	32.1	48.7	34.1	13.2	35.3	50.1	51.75G
VFNet[20]	33.6	49.9	34.9	13.4	36.8	51.9	173G
Dense Teacher[21]	33.8	50.4	35.2	14.0	35.6	49.6	193G
ARSL[22]	33.9	51.6	35.0	12.9	37.1	**52.7**	188G
The proposed method	**34.1**	**52.3**	**35.3**	**15.5**	**38.4**	49.9	53.58G

Fig.5 shows the detection effect of different networks. Faster-RCNN has obvious redundancy in detection. The first, second and third pictures detect many redundant objects, and the second picture does not accurately return the position of the car. FCOS has missed detection. The TV and the person on the right are not detected in the first picture, and the laptop is also missing in the third picture. The first and third images of FoveaBox have redundant detections, and the fourth image mistakenly detects a cat as a bear. ATSS also falsely detected the fourth image as a bear. Our proposed network can not only detect the target accurately but also has a low false detection and redundancy rate. Compared with other networks, its detection accuracy is better than these networks.

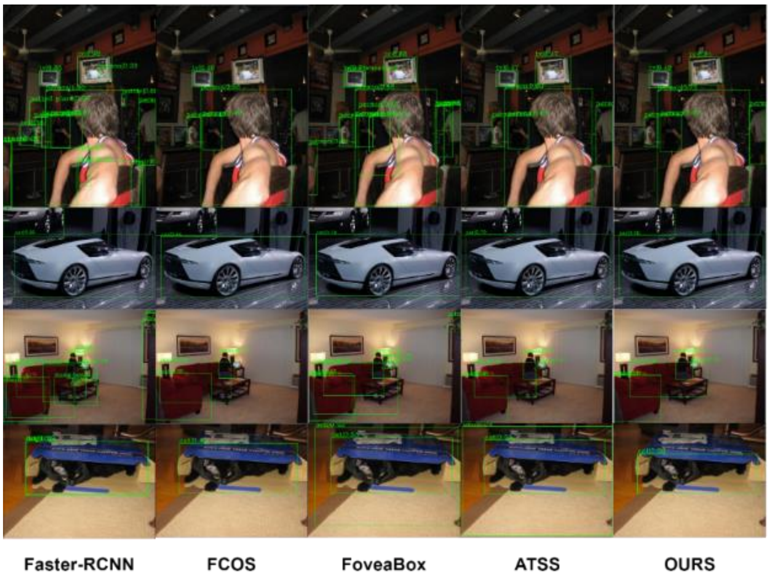


Fig. 5. Comparison of detection effects of different networks. All images have the confidence threshold set to 0.3.

Fig.6 shows the detection effect of our network on different occasions. It accurately detects when there is only a single object in the picture, such as bicycles, cats and buses, despite the different sizes of the objects. When the shadow of the dog and the dog appear at the same time, the network can accurately detect the dog regardless of its shadow. The network was also able to detect dogs when the object was affected by background light. When the objects in the picture are incomplete, the network can detect people based on their legs and birds based on their heads. And when there are multiple objects in the picture, it can detect people and ships separately. For dense crowds, it can not only detect obvious people but also detect occluded objects. It is not difficult to see that the network proposed in this paper can accurately complete the detection task, and can also achieve good results in disturbed scenes.

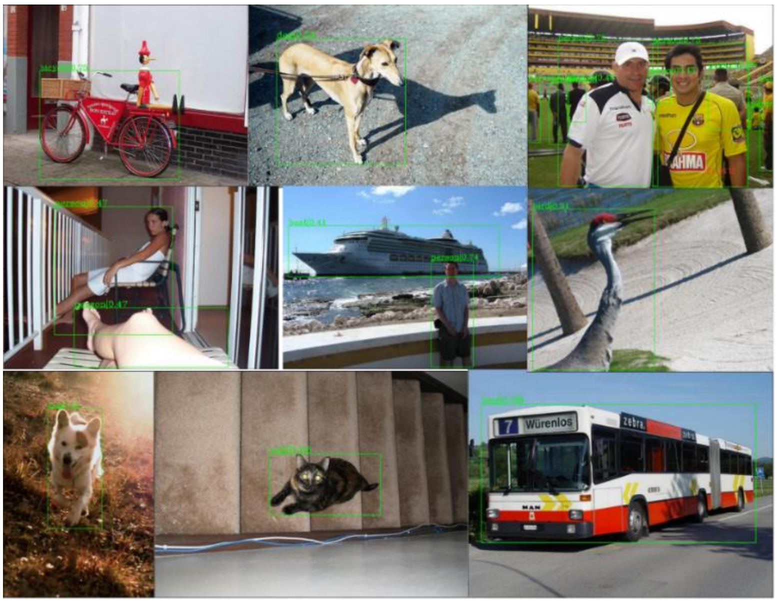


Fig. 6. Qualitative results of the proposed method. This model achieves 34.1% in AP. All images have the confidence threshold set to 0.3.

5 CONCLUSION

In this paper, we propose the CESNet. The core modules of the network are as follows: CE module and CS module. The CE module shows the correlation between different channels, strengthens important features and suppresses nonimportant features. The CS module effectively reduces the semantic loss caused by feature vector channel dimensionality reduction.

Ablation experiments verify the effectiveness of our proposed modules. These modules have little effect on the complexity of the network. Under the same configuration, our algorithm improves 2.0% AP, 3.6% AP_{50}, 1.2% AP_{75}, 2.3% AP_S and 3.1% AP_M, respectively. In future work, we will explore the effects of spatial attention and channel attention on the CE module, respectively. We will also compare the performance of the CS module and the residual structure. In addition, our proposed module is a plug-and-play module, which can be quickly added to many single-stage object detection methods and improve the accuracy effect very quickly.

Acknowledgements. This work is supported by the Open Foundation Project of Hubei Key Laboratory of Optical Information and Pattern Recognition, Wuhan Institute of Technology. Grant No. 202306

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Mabrouk, A.B., Zagrouba, E.: Abnormal behaviour recognition for intelligent video surveillance systems: A review. Expert Systems with Applications 91, 480–491 (2018)
2. Kim, C.: Content-based image copy detection. Signal Processing: Image Communication 18(3), 169–184 (2003)
3. DeSouza, G.N., Kak, A.C.: Vision for mobile robot navigation: A survey. IEEE transactions on pattern analysis and machine intelligence 24(2), 237–267 (2002)
4. Mookherjee, D., Reichelstein, S.: Implementation via augmented revelation mechanisms. The Review of Economic Studies 57(3), 453–475 (1990)
5. Zhao, Z.Q., Zheng, P., Xu, S.t., Wu, X.: Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems 30(11), 3212–3232 (2019)
6. Koskowich, B.J., Rahnemoonfai, M., Starek, M.: Virtualot—a framework enabling real-time coordinate transformation & occlusion sensitive tracking using uas products, deep learning object detection & traditional object tracking techniques. In: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. pp. 6416–6419. IEEE (2018)
7. Cortès, U., Sànchez-Marrè, M., Ceccaroni, L., R-Roda, I., Poch, M.: Artificial intelligence and environmental decision support systems. Applied intelligence 13, 77–91 (2000)
8. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 21–37. Springer (2016)
9. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
10. Impiombato, D., Giarrusso, S., Mineo, T., Catalano, O., Gargano, C., La Rosa, G., Russo, F., Sottile, G., Billotta, S., Bonanno, G., et al.: You only look once: Unified, real-time object detection. Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip 794, 185–192 (2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
12. Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., Xu, C.: Ghostnet: More features from cheap operations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1580–1589 (2020)
13. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
14. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8759–8768 (2018)
15. Ghiasi, G., Lin, T.Y., Le, Q.V.: Nas-fpn: Learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7036–7045 (2019)
16. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10781–10790 (2020)
17. Qiao, S., Chen, L.C., Yuille, A.: Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10213–10224 (2021)

18. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015)
19. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9626–9635 (2019). https://doi.org/10.1109/ICCV.2019.00972
20. Zhang, H., Wang, Y., Dayoub, F., Sunderhauf, N.: Varifocalnet: An iou-aware dense object detector. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8514–8523 (2021)
21. Zhou, H., Ge, Z., Liu, S., Mao, W., Li, Z., Yu, H., Sun, J.: Dense teacher: Dense pseudo-labels for semi-supervised object detection. In: European Conference on Computer Vision. pp. 35–50. Springer (2022)
22. Liu, C., Zhang, W., Lin, X., Zhang, W., Tan, X., Han, J., Li, X., Ding, E., Wang, J.: Ambiguity-resistant semi-supervised learning for dense object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15579–15588 (2023)
23. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9759–9768 (2020)
24. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: Efficient channel attention for deep convolutional neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11534–11542 (2020)
25. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7132–7141 (2018)
26. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018)
27. Wu, W., Zhang, Y., Wang, D., Lei, Y.: Sk-net: Deep learning on point cloud via end-to-end discovery of spatial keypoints. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 6422–6429 (2020)
28. Misra, D., Nalamada, T., Arasanipalai, A.U., Hou, Q.: Rotate to attend: Convolutional triplet attention module. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 3139–3148 (2021)
29. Li, D., Hu, J., Wang, C., Li, X., She, Q., Zhu, L., Zhang, T., Chen, Q.: Involution: Inverting the inherence of convolution for visual recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12321–12330 (2021)
30. Kong, T., Sun, F., Liu, H., Jiang, Y., Li, L., Shi, J.: Foveabox: Beyound anchor-based object detection. IEEE Transactions on Image Processing 29, 7389–7398 (2020)
31. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)