Global context aware RCNN for object detection

Wenchao Zhang1 · Chong Fu1,2,3 · Haoyu Xie1 · Mai Zhu1 · Ming Tie4 · Junxin Chen5

Received: 1 December 2020 / Accepted: 19 February 2021 / Published online: 10 March 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract
RoIPool/RoIAlign is an indispensable process for the typical two-stage object detection algorithm, it is used to rescale the object proposal cropped from the feature pyramid to generate a fixed size feature map. However, these cropped feature maps of local receptive fields will heavily lose global context information. To tackle this problem, we propose a novel end-to-end trainable framework, called global context aware (GCA) RCNN, aiming at assisting the neural network in strengthening the spatial correlation between the background and the foreground by fusing global context information. The core component of our GCA framework is a context aware mechanism, in which both global feature pyramid and attention strategies are used for feature extraction and feature refinement, respectively. Specifically, we leverage the dense connection to improve the information flow of the global context at different stages in the top-down process of FPN, and further use the attention mechanism to refine the global context at each level in the feature pyramid. In the end, we also present a lightweight version of our method, which only slightly increases model complexity and computational burden. Experimental results on COCO benchmark dataset demonstrate the significant advantages of our approach.

Keywords Object detection · Context awareness · Attention mechanism · Dense connection

Chong Fu
fuchong@mail.neu.edu.cn
Wenchao Zhang
sylgzwc@163.com
Haoyu Xie
895852154@qq.com
Mai Zhu
zhumai@stumail.neu.edu.cn
Ming Tie
tming7611@sina.com
Junxin Chen
chenjx@bmie.neu.edu.cn

1 School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China
2 Engineering Research Center of Security Technology of Complex Network System, Ministry of Education, Shenyang, China
3 Key Laboratory of Intelligent Computing in Medical Image, Ministry of Education, Northeastern University, Shenyang 110819, China
4 Science and Technology on Space Physics Laboratory, Beijing 100076, China
5 College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110819, China
1 Introduction

Benefiting from the development and application of deep network technology in computer vision community, the performance of a wide range of computer vision tasks such as object detection, semantic segmentation and instance segmentation have greatly been improved. In recent years, many excellent detection frameworks have been proposed. For example, there are one-stage methods with faster speed such as SSD [1] and YOLO [2] and two-stage methods with better detection performance such as faster RCNN [3] and FPN [4]. It is a remarkable fact that most of the currently popular two-stage methods usually use RoIPool/RoIAlign to align regions of interest of different scales to meet the requirement of consistent input size of the neural network. In FPN, it adaptively crops the regions of interest from the feature pyramid corresponding to different spatial scales, and then uniformly resizes them to a fixed spatial scale of $7 \times 7$ through RoIPool/RoIAlign, and after flattening these feature maps, they are further encoded through two fully connected layers, and finally classification and positioning tasks are performed, respectively. However, is it reasonable to use only local receptive field features such as object proposals for classification and positioning? On the one hand, for classification tasks, the target object has a natural inter-dependencies relationship with the background and other foreground objects. For example, in real scenario, cups often appear on the dining table, and there are food, knives, forks and bowls around them. In addition, laptop, keyboard and mouse often appear together on the desk, as shown in the Fig. 1. On the other hand, for positioning tasks, the candidate box coordinates predicted by the detection model are the relative positions in the whole image, so some references in the background can help to locate the target. Therefore, simply using the feature maps of object proposals for detection will bring about the loss of the spatial and category relationship information between these local and global contexts.

To mitigate the drawback mentioned above, in this paper, we propose a context aware mechanism that allows the two-stage object detection network to fuse the global context information with the local informations of the RoIs (Regions of Interest). In two-stage methods, rpn (region proposal network) head (i.e., the first stage) and roi head (i.e., the second stage) both use the image features extracted by the backbone network for prediction tasks, where rpn head is responsible for distinguishing the foreground and background and predicting the regression coefficient of the anchor box, while roi head is responsible for predicting the specific category of roi and obtaining the offset value used to fine-tune the bounding box. Therefore, we enhance its global feature perception capabilities in these two stages, respectively. For rpn head, it uses the overall features of the output from the backbone network for prediction tasks, so we only use global feature statistics to perform feature calibration. But for roi head, it uses partial features cropped from the overall feature map as input, so we need a more sophisticated design for the extraction of its global features. The structure of the our model is depicted in Fig. 2. Specifically, we believe that the feature maps at different stages in the feature pyramid carry global context information of different attributes. Therefore, in order to make full use of this information to help the neural network better complete the object detection task, we fuse the global context information of different stages through dense connection in the roi head, and then leverage our proposed context aware module to

![Fig. 1 Examples of potential relationship between global context and local information, top row: some kitchen utensils such as knives, forks, bowls and cups often appear on the table with food; bottom row: computer, keyboard and mouse often appear together](image)
generate higher-dimensional global descriptors. Simultaneously, like FPN, before decoupling the positioning and classification tasks, we use two shared fully connected layers to further extract features at different stages, and finally we fuse the prediction information of different stages to make the final decision.

To provide evidence for these claims, in Sect. 4, we develop several ablation studies and conduct an extensive evaluation on the COCO dataset [5]. We also present results beyond COCO that indicate that the benefits of our approach are not restricted to a specific dataset. In the end, our method gains \(+1.5\) and \(+0.9\) AP on MS COCO dataset from feature pyramid network (FPN) baselines with ResNet-50 and ResNet-101 backbones, respectively.

In summary, the main contributions of this work are highlighted as follows:

1. We observe that due to the use of RoIAlign, a lot of global context information will be lost in most two-stage object detection networks, and hence we propose GCA RCNN to enhance and refine global context information by using dense connection and attention mechanism, and finally the missing global information is compensated by the fusion of global context and local features.

2. Unlike SENet [6], which uses global context information for feature recalibration at the convolutional level, we extend it to extract global context information on the two-stage object detection pipeline.

3. Our method can be easily deployed in other FPN-based methods, and we also present a lightweight version of our method, which has a small amount of additional computing overhead.

The rest of this paper is organized as follows. In Sect. 2, we briefly review related work on object detection and context awareness. In Sect. 3, we introduced our method in detail from dense global context, context awareness and feature fusion. Experimental details and analysis of the results are elaborated in Sect. 4. Finally, we conclude the paper in Sect. 5.

2 Related work

2.1 Object detection

The vigorous development of object detection technology depends on two factors: the release of large-scale datasets and the emergence of deep networks. Usually, we use common datasets like ImageNet [7] and COCO [5] to train and test our methods. Simultaneously, some datasets focusing on specific scenarios have been established to solve practical problems in our daily lives. Recently, the MOHR dataset [8] was released to study the application of unmanned aerial vehicles (UAVs) scenarios. Moreover, in [9], Zhang et al. established static and dynamic datasets separately for unmanned vending machine scenarios. It is these diverse datasets that are driving the development of deep learning techniques. There are two common ways for
object detection: one-stage and two-stage. Classic one-stage methods such as SSD [1] and YOLO [2, 10, 11] quickly classify and locate targets in an end-to-end manner. Classic two-stage methods such as faster RCNN [3] and FPN [4] first obtain object proposals through the rpn, and then use RoIPool/RoIAlign to align the spatial scales of these object proposals before performing detection. Moreover, most of the subsequent algorithms are based on these two structures for continuous improvement and development. In later study, CornerNet [12], ExtremeNet [13], CenterNet [14] and FCOS [15], etc., use key-point detection technology to optimize the anchor generation process. Wu et al. [16] studied that convolutional neural networks and fully connected networks have different sensitivity to classification tasks and positioning tasks; therefore, it decouples the positioning and classification tasks of FPN to improve detection performance. Wang et al. [17] proposed using image features to guide the generation of anchors. Cascade r-cnn [18] considers that training samples under different IoU (Intersection over Union) conditions have different effects on the performance of target detection network, and then proposes the structure of multi-stage, in which each stage has a different IoU threshold. HTC [19] tries to integrate semantic segmentation into the instance segmentation framework to obtain a better spatial context. Detectors [20] proposed RFP (recursive feature pyramid) and SAC (switchable atrous convolution) to realize looking and thinking twice or more. Although the two-stage methods mentioned above can achieve excellent detection performance, they inevitably leverage RoIPool/RoIAlign in roi head, resulting in the lack of global context, while our method can effectively alleviate this problem by extracting the global features of different stages and fusing them with the local features of object proposals.

2.2 Context awareness

Assembling global context information of the target can enable the neural network to learn more about the relationship between the foreground and the background, so that it can rely on this potential relationship feature to help the neural network highlight and identify the target. There are many ways to obtain contextual information, an alternative way is to use the attention mechanism to obtain global context information [6, 21–24] and use it for feature recalibration. Alongside the methods described above, there is also another way to use contextual information. For the two stage object detection method, the target needs to be aligned to a uniform scale through RoIPool/RoIAlign. The general way is to expand the object proposal by a few pixels when cropping the target from the feature map to obtain more surrounding information [25, 26], while our method can fuse global context information and is not limited to surrounding background features. Context information can also be used in many other ways. Lin et al. [27] proposed CGC (context-gated convolution) to adaptively modify the weight of the convolutional layer. Si et al. [28] proposed DuATM (Dual Attention Matching network) to learn context-aware feature sequences and perform pedestrian re-identification by performing sequence comparison simultaneously. Although the above-mentioned methods verify the importance of the global context, they do not involve the relationship between the cropped local features obtained after RoIPool/RoIAlign and their corresponding global contexts. And most of the methods are used to enhance the attention to global features in the backbone network, whereas our method mainly focuses on the lack of global context in the head network of the two-stage method.

3 Methods

In this Section, we will first introduce the motivation of our design, and then describe the components of our model in detail from three aspects: dense global context, context awareness and feature fusion. Note that since in rpn head does not involve RoIAlign, we will introduce the enhancement of global context information for rpn head in Sect. 4.

3.1 Motivation

In FPN, the feature pyramid is constructed through bottom-up pathway, top-down pathway and lateral connections. As shown in Fig. 3, where bottom-up pathway refers to the process of downsampling the input image 5 times in the backbone network, and for simplicity the output of the deep residual blocks corresponding to \{conv2, conv3, conv4, conv5\} is denoted as \{c2, c3, c4, c5\}, where top-down pathway refers to the upsampling process after convoluting c5 by \(1 \times 1\) convolutional layer; for simplicity, we denote the final feature map set as \{p2, p3, p4, p5\}, where lateral connection refers to the process of fusing the corresponding feature maps between \{c2, c3, c4, c5\} and \{p2, p3, p4, p5\} through the \(1 \times 1\) convolutional layer. In the roi head of FPN, only using the cropped feature map of object proposal to locate and classify the object will greatly lose the global context information, it will weaken the ability of the convolutional neural network to perceive the relationship between background and foreground information. Therefore, we design our method from two aspects: the acquisition of global context information and the fusion of local and global information.
3.2 Dense global context

In order to unify the spatial scale of the global context information at different stages in the feature pyramid, we leverage adaptive average pooling to downsample the spatial scale of \( f_{p2}, f_{p3}, f_{p4}, f_{p5} \) to \((M, N) \times \{1, 1/2, 1/4, 1/8\} \), respectively, since the adaptive average pooling layer can reduce the scale of features while retaining the principal components of global features, it can effectively reduce the occupation of computing resources. Then, we continue to downsample these pooled feature maps using four parallel branches which containing \( f_{p3}, f_{p2}, f_{p1}, f_{p0} \) downsampling blocks, respectively, each downsampling block refers to the composite function of two consecutive operations: a \( 3 \times 3 \) convolution (conv) with stride 2 followed by a ReLU (rectified linear unit) [29] activation function; for simplicity, we denote the downsampling block as \( D \). To this end, the global context feature map set obtained by dense connection is denoted as \( \{g_0, g_1, g_2, g_3\} \). As a consequence, the benefits of the global context captured by multi-branch downsampling blocks can be accumulated through the network and using this dense connection architecture can not only enhance the gradient flow, but also enable efficient feature reuse. The output feature \( g_i \) is defined by

\[
g_i = D^{i-1}(\phi(p_{i+2}), W_i),
\]

where \( D^i \) represents the total use of \( i \) downsampling blocks, \( \phi \) stands for adaptive average pooling function, where \( W_i \) represents the \( i \)-th row of matrix \( W \). After getting the feature set \( \{g_0, g_1, g_2, g_3\} \) of the global context information, we input \( g_i \) and the feature maps of object proposal, respectively, into four parallel context aware modules. Motivated by [30], we define \( [\phi(p_i), W_i] \) as a concatenation of the feature maps in it.

3.3 Context awareness

A diagram illustrating the structure of an context aware module is shown in Fig. 4, the context aware module consists of two sub-modules: attention module and task decoupling module. In [6], the squeeze-and-excitation architecture has been proven to be an effective method for extracting global features, since it can boost meaningful features while suppressing weak ones. And what we are concerned about is the introduction of global context information into local features. For this reason, in the attention module, inspired by [6], we embedding these global context information \( \{g_0, g_1, g_2, g_3\} \) into higher-dimensional features for characterizing the global features. Specifically, we use the global average pooling layer to squeeze the spatial scale of \( \{g_0, g_1, g_2, g_3\} \) that produces a channel-wise statistics by aggregating feature maps across
their spatial dimensions, then we leverage a bottleneck block composed of two fully connected layers with reduction of $r$ to squeeze-and-excitation the dimension of the feature map and learn a non-mutually exclusive relationship between global context information and local information, note that in the bottleneck block, the first fully connected layer uses the ReLU activation function, and the second fully connected layer uses the sigmoid activation function. Besides, in this work, unlike the squeeze-and-excitation block, the output of our attention module is only the extracted 1D global features. In the task decoupling module, we first do the channel-wise multiplication between the global context descriptor obtained by the squeeze-and-excitation module and the $256 \times 7 \times 7$ object proposal feature tensor after RoIAlign resize, and then in order to further combine features of different attributes and enhance generalization, after flattening the object proposal feature map, like FPN, we used two fully connected layers with an output dimension of 1024. It is worth noting that, different from SENet multiplying the global descriptor on each channel of the squeeze-and-excitation block input feature map to perform feature recalibration, we multiply it on each channel of the $256 \times 7 \times 7$ feature tensor obtained by RoIAlign to strengthen the mutual relationship between global context and local receptive field.

### 3.4 Feature fusion

Different from the box head in FPN, the object proposal rescaled by RoIAlign will be classified and located after the subsequent two fully connected layers. In our approach, we leverage four parallel branches to extract global context information at different stages in the feature pyramid, and further leverage these global context informations to fusion with the local receptive fields of object proposals in each branch through the attention mechanism. Simultaneously, like FPN, after each branch, we decouple these fused feature information through two parallel fully connected layers for the classification task and the positioning task. The motivation of feature fusion operation is similar to that of classical multi-head architecture [22]: it can model the features of different attributes through different learnable parameters and significantly enhance the representational power of the network by fusing these features. Thereby, the two parallel 1D features output by each branches are fused through the element-wise sum for the final prediction.

### 4 Experiments

In this Section, we will first introduce our experimental dataset, evaluation criteria and model parameter settings. Then, we conducte ablation experiments on COCO dataset [5] from two aspects, namely the dense connection and global context awareness. Moreover, we verified the generalization of our method on the Cityscapes dataset [31].

#### 4.1 Dataset and metrics

We verify our approch on the large scale detection benchmark COCO dataset with 80 object categories, which are splited into 115k, 5k and 41k images for train, minival and test, respectively. Because the labels of its test-dev split are not publicly available, we use its minival dataset for our ablation study. Simultaneously, we present our final results on the test-dev split (20K images) by uploading our
detection results to the evaluation server. We use the COCO standard metric to evaluate the AP under different IoU [0.5:0.05:0.95] and finally take the average of APs under these thresholds as the result, denoted as mAP@[.5, .95 ].

4.2 Implementation details

Our model is end-to-end trained based on the torchvision detection module [32], using SGD with 0.9 momentum and 0.0005 weight decay for gradient optimization. We train detectors on a single NVIDIA titan xp GPU with the mini-batch size of one image. Unless specified, ResNet-50 pre-trained on ImageNet [7] is taken as the backbone networks. And all newly added convolutional layers are randomly initialized with the “xavier” method [33]. Besides, we flip the input image horizontally with a probability of 0.5. We train detectors for 24 epochs with an initial learning rate of 0.0025 and decrease it by 0.1 after 16 and 22 epochs, respectively. For data augmentation, we randomly shuffle the input image horizontally with a probability of 0.5. And all newly added convolutional layers are randomly initialized with the “xavier” method [33]. Besides, we perform experiments with our method on Cityscapes dataset which comprise a collection of 2975 training, 500 validate and 1525 test 2048 pixel RGB images and labelled with 8 classes. We train our model a total of 64 epochs, respectively. For data augmentation, we randomly sampled from [800, 1024] for reducing overfitting, other parameters are the same as those set in the experiment on COCO dataset.

4.3 Ablation study

4.3.1 Dense connection

Table 1 shows the impact of whether the global context features \{p2,p3,p4,p5\} at different stages in the feature pyramid are densely connected on the performance of our proposed model, it should be noted that we did not use the attention mechanism in this section and the initial adaptive pooling size is (64, 96). Specifically, in order to further encode local receptive field information and global context information, after the 1D global context information captured through global average pooling in attention module and the object proposal 256 \times 7 \times 7 feature map tensor in task decoupling module, we connect a fully connected layer with an output dimension of 512, respectively. Finally, we concatenate the two pieces of 512 1D tensors and then use a fully connected layer with an output dimension of 1024 to learn the potential relationship between local receptive fields and global context information. From Table 1, we can infer that the use of dense connections can enhance the global information flow between different stages, so it can better improve model performance (outperforms FPN baseline by 1.0% on COCO’s standard AP metric and by 2.0% on AP@IoU=0.5). From this vantage, we used dense connections in all subsequent experiments.

4.3.2 Dense connection with different pooling size

Table 2 shows the impact of dense connection of global context information at different stages in the feature pyramid on the performance of the model under different pooling size conditions. Assuming that our initial adaptive pooling size is (128, 192), and thus the pooling sizes corresponding to the four stages \{p2,p3,p4,p5\} from the bottom to the top of the FPN are (128,192) \times \{1/8, 1/4, 1/2, 1\}, with the goal of balancing the memory consumption and accuracy, we analyze the model performance when the initial pooling size is (128,192) \times \{1,1/2, 1/4, 1/8\}, respectively. The comparison in Table 2 shows that performance does not continuously increasing with bigger pooling size, and we find that setting the initial pool size to (64, 96) achieve a good balance between memory consumption and accuracy, so hereafter we used this value in all experiments.

4.3.3 The choices of the attention module

For the sake of better integrating global context information and local receptive field information, we explore

Table 2 Effect of dense connection with different pooling size on COCO val2017 (%)

| Pooling size    | AP  | AP0.5 | AP0.75 |
|-----------------|-----|-------|--------|
| (128,192)       | 37.8| 60.0  | 40.6   |
| (64,96)         | **38.0** | **60.1** | **41.0** |
| (32,48)         | 37.8| 59.7  | 40.9   |
| (16,24)         | 37.7| 59.7  | 40.2   |

| Table 1 Effect of dense connection on COCO val2017 (%) |
|---------------------------------------------------------|
| Method                     | AP  | AP0.5 | AP0.75 |
|---------------------------|-----|-------|--------|
| FPN baseline              | 36.8| 58.0  | 40.0   |
| Dense connection (ours)    | **37.8** | **60.0** | **40.6** |
different attention methods, the results of comparison are shown in Table 3. Specifically, as shown in Fig. 5a, we directly do the channel-wise multiplication between the 256-dimensional global context information output by the squeeze-and-excitation module and the $256 \times 7 \times 7$ feature tensor of the object proposal, and we denote this method as attention on “conv”. Further more, on the basis of attention on “conv”, we add a new branch to the squeeze-and-excitation module and increase the output dimension to 1024, then we try two different structures, one is do the channel-wise multiplication between it and the 1024-dimensional output tensor of the first fully connected layer (attention on “conv+fc1”, as shown in Fig. 5b) and the other is between it and the 1024-dimensional output tensor of the second fully connected layer (attention on “conv+fc2”, as shown in Fig. 5c). In addition, as shown in Fig. 5d, we connected three parallel output branches after the squeeze-and-excitation module, denote as “conv+fc1+fc2”. Simultaneously, we also explored different combinations of attention on single “fc” layer. The results reported in Table 3 indicate that assembling global context information multiple times in the box head network will cause ambiguity, and mapping it to the low-dimensional features of the local information can better learn the relationship between the global context and the local receptive field. By using attention on “conv” we got the best AP value, which exceeded FPN baseline by 1.5% on COCO’s standard AP metric.

### 4.3.4 Attention with different reduction ratio

The comparison in Table 4 shows the effect of different reduction ratios in the squeeze-and-excitation module on the performance of our model. Different reduction ratios allow us to explore different capacity and computational cost of the attention module in the network. We can achieve the best results when $r$ is equal to 8; therefore, we use this value in all of our other experiments.

### 4.3.5 Bottom-up or top-down

In the previous experiment, we obtained global context information by continuously downsampling from top to bottom. In this experiment, we replace all newly added $3 \times 3$ convolutional layers with deconvolutional layers to obtain global context information in a bottom-up manner and keep other network structures unchanged, the results are shown in Table 5.

### 4.3.6 Effect of different convolution kernel sizes

Compared with FPN, our method introduces three additional artificially designed hyperparameters: the initial adaptive pooling size, reduction rate and the convolution kernel size of the newly added convolutional layer in the dense connection architecture. We compared the first two parameters in Tables 2 and 4, respectively. In this section, we analyse the impact of different convolution kernel sizes on model performance. As is known, the size of the convolution kernel determines the range of the receptive field and the number of learnable parameters, which will affect the performance of the model. In this regard, we set the size of these newly added convolution kernels to \{1,3,5\}, respectively, for analysis. It should be noted that for the convolution kernel size of 1, we added a max pooling layer with a stride of 2 to perform scale reduction. Experimentally, it can be seen from Table 6 that the best performance can be obtained when the kernel size is 3, so we use it as the default value in subsequent experiments.

### 4.3.7 Feature recalibration on rpn head

In rpn head, in order to determine whether the anchor box generated on the output feature map at different stages of the backbone network contains objects and obtain the corresponding regression coefficients, it is necessary to use the feature map as input for classification and regression tasks, respectively. For the purpose of enhancing, the global feature information interaction between the input feature map channels, we first use global average pooling to reduce the spatial size of the input feature map, then use a fully connected layer with the same input and output channels for feature refining, and finally it is used for channel-wise multiplication with the input feature map to enhance the feature representation ability, and the results are shown in Table 8.

### 4.3.8 Results on the Cityscapes dataset

Next, we investigate whether the benefits of global context information generalise to datasets beyond COCO. For this purpose, we deployed our algorithm on the Cityscapes dataset.
From Table 7, we observe that our method achieves a better AP value (1.2% improvement) than FPN baseline on Cityscapes datasets, which further illustrates the robustness of our method. Especially, our method is more effective for small objects that are more difficult to detect, and its performance outperforms the FPN baseline by 2.4%. This is because the input image resolution of the Cityscapes dataset is higher than that of the COCO dataset, and thus we can capture more detailed contextual information.

4.4 Main results

4.4.1 Comparison with FPN baselines on COCO validation dataset

In Table 8, we compare the performance of our method with FPN baselines on COCO val2017, where (+rpn head) means to deploy feature calibration on rpn head. From Table 8, we can see that our method can achieve continuous gains on different backbone networks (1.5% improvement with ResNet-50 and 0.9% improvement with ResNet-101). Compared with FPN baselines, our method is better at detecting small and medium targets, which is due to the increased connection between global context and connected layer with ReLU activation function. And ⊗ represents the channel-wise multiplication.
local information. Furthermore, by recalibrating the input features of rpn head to strengthen the relationship between global features, our model’s detection ability for larger objects has also been greatly improved. Meanwhile, from Table 8, we can see that the effect of global feature recalibration is not obvious in rpn head (0.1% improvement with ResNet-50 and 0.3% improvement with ResNet-101), since the rpn head uses the entire feature map of the backbone network output as input. Therefore, in rpn head, not that the global context information is lost, but that the information exchange between channels is lacking. However, in roi head, which using the local feature map after roi align cropping as input will lose a lot of background information. Consequently, using our proposed CGA to compensate for the missing global context information in roi head can make the baseline model obtain a larger performance improvement, which also experimentally validates the rationality of our hypothesis and the effectiveness of our proposed method.

4.4.2 Comparison with state-of-the-art methods on COCO test-dev

Table 9 shows the comparison between our method and the state-of-the-art methods on MS COCO test-dev2017. In Table 9, we also compare the performance of our method with FPN baselines and Double-Head [16] on COCO test-dev, where Double-Head (GCA) means to assemble our method on Double-Head RCNN [16] based on mmdetection [40], note that in this method we only applied our GCA to the fully connected head and we re-implemented Double-Head RCNN using two Titan xp GPUs with one image per GPU (schedule_1x). Our method can achieve continuous gains on different backbone networks (0.9% improvement with ResNet-101) and model (0.5% compared to Double-Head).

4.4.3 Lightweight version

In order to reduce the amount of parameters and model complexity to obtain better precision and speed trade-off, in this experiment, we present a lightweight version of our

| Method                          | Backbone | AP   | AP0.5 | AP0.75 | AP0  | APm  | APj  |
|--------------------------------|----------|------|-------|--------|------|------|------|
| FPN baseline [4]               | ResNet-50| 36.8 | 58.0  | 40.0   | 21.2 | 40.1 | 48.8 |
| GCA (ours)                     | ResNet-50| 38.2 | 59.9  | 41.0   | 22.7 | 42.0 | 49.0 |
| GCA (ours+rpn head)            | ResNet-50| 38.3 | 60.1  | 41.0   | 22.9 | 41.8 | 49.2 |
| FPN baseline [4]               | ResNet-101| 39.1 | 61.0  | 42.4   | 22.2 | 42.5 | 51.0 |
| GCA (ours)                     | ResNet-101| 39.7 | 61.0  | 43.3   | 23.0 | 43.7 | 51.3 |
| GCA (ours+rpn head)            | ResNet-101| 40.0 | 61.5  | 43.4   | 23.5 | 43.9 | 52.4 |

| Method                          | Backbone | AP   | AP0.5 | AP0.75 | Ap0  | Apm  | Apj  |
|--------------------------------|----------|------|-------|--------|------|------|------|
| Deep regionlets [34]            | ResNet-101| 39.3 | 59.8  | –      | 21.7 | 43.7 | 50.9 |
| Mask R-CNN [35]                 | ResNet-101| 39.8 | 62.3  | 43.4   | 22.1 | 43.2 | 51.2 |
| IOU-Net [36]                    | ResNet-101| 40.6 | 59.0  | –      | –   | –   | –   |
| Soft-NMS [37]                   | ResNet-101| 40.9 | 62.8  | –      | 23.3 | 43.6 | 53.3 |
| LTR [38]                        | ResNet-101| 41.0 | 60.8  | 44.5   | 23.2 | 44.5 | 52.5 |
| Fitness NMS [39]                | ResNet-101| 41.8 | 60.9  | 44.9   | 21.5 | 45.0 | 57.5 |
| FPN baseline [4]                | ResNet-101| 39.1 | 60.5  | 42.4   | 22.0 | 42.2 | 49.3 |
| GCA (ours)                      | ResNet-101| 40.0 | 61.6  | 43.5   | 22.8 | 43.2 | 50.3 |
| Double-Head [16]                | ResNet-101| 41.6 | 62.0  | 45.7   | 23.8 | 44.8 | 52.7 |
| Double-Head (GCA)               | ResNet-101| 42.1 | 63.0  | 45.9   | 24.4 | 45.2 | 53.2 |
method. Specifically, for feature recalibration on rpn head, we simply use a $1 \times 1$ convolutional layer to replace the newly added fully connected layer, and for the global information extraction of the roi head, we directly use the feature maps at different stages in the feature pyramid of $p_2, p_3, p_4, p_5$ to obtain their corresponding spatial statistics through global average pooling, and then fuse them by element-wise sum, and after that only one context aware module is retained to refine the fused global features. Finally, we do the channel-wise multiplication between the global context descriptor and the input roi feature. As can be seen from the Table 10, compared to FPN baseline, our lightweight version of GCA can achieve good performance improvements ($+ 0.8\%$), while a similar inference speed is maintained.

### 4.4.4 Qualitative evaluation and future work

In Fig. 6, we show some specific examples of the results of our approach. From the comparison of these results, we can intuitively observe that our method can effectively improve the performance of the object detector and obtain better positioning and classification results. However, we can also observe from the examples in the last row that our method has limited improvement in scenes with similar foreground and background tones, which need to be further explored in our future work.

### 5 Conclusion

In this paper, we propose global context aware (GCA) RCNN to learn the potential relationship between image background and foreground by integrating global context information with local receptive field information of roi and different from the attention mechanism on the convolution level for feature recalibration, we extend it to the network pipeline to strengthen the connection between local and global information. Experiments on the COCO and Cityscapes datasets have verified the effectiveness of our method, and we also hope that our method will be helpful to other scholars.

**Acknowledgements** This work was supported by the National Natural Science Foundation of China (Nos. 61802055 and 61773068) and the Fundamental Research Funds for the Central Universities (No. N2024005-1).

**Compliance with ethical standards**

**Conflict of interest** The authors declare that they have no conflict of interest.
References

1. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) SSD: single shot multibox detector. In: European conference on computer vision, Springer, pp 21–37

2. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779–788

3. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp 91–99

4. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125

5. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: common objects in context. In: European conference on computer vision. Springer, pp 740–755

6. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7132–7141

7. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) ImageNet: a large-scale hierarchical image database. In: CVPR09

8. Zhang H, Sun M, Li Q, Liu L, Liu M, Ji Y (2021) An empirical study of multi-scale object detection in high resolution UAV images. Neurocomputing 421:173–182

9. Zhang H, Li D, Ji Y, Zhou H, Wu W, Liu K (2019) Tornado new retail: a benchmark dataset for smart unmanned vending machines. IEEE Trans Ind Inform 16(12):7722–7731

10. Redmon J, Farhadi A (2018) Yolov3: an incremental improvement. arXiv preprint arXiv:180402767

11. Redmon J, Farhadi A (2017) Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7263–7271

12. Law H, Deng J (2018) Cornernet: detecting objects as paired keypoints. In: Proceedings of the European conference on computer vision (ECCV), pp 734–750

13. Zhou X, Zhuo J, Krahenbühl P (2019) Bottom-up object detection by grouping extreme and center points. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 850–859

14. Zhou X, Wang D, Krahenbühl P (2019) Objects as points. arXiv preprint arXiv:190407850

15. Tian Z, Shen C, Chen H, He T (2019) FCOS: fully convolutional one-stage object detection. In: Proceedings of the IEEE international conference on computer vision, pp 9627–9636

16. Wu Y, Chen Y, Yuan L, Liu Z, Wang L, Li H, Fu Y (2020) Rethinking classification and localization for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10186–10195

17. Wang J, Chen K, Yang S, Loy CC, Lin D (2019) Region proposal by guided anchoring. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2965–2974

18. Cai Z, Vasconcelos N (2018) Cascade R-CNN: delving into high quality object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6154–6162

19. Chen K, Pang J, Wang J, Xiong Y, Li X, Sun S, Feng W, Liu Z, Shi J, Ouyang W, et al. (2019) Hybrid task cascade for instance segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4974–4983

20. Qiao S, Chen LC, Yuille A (2020) Detectors: detecting objects with recursive feature pyramid and switchable atrous convolution. arXiv preprint arXiv:200602334

21. Bello I, Zoph B, Vaswani A, Shlens J, Le QV (2019) Attention augmented convolutional networks. In: Proceedings of the IEEE international conference on computer vision, pp 3286–3295

22. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, pp 5998–6008

23. Woo S, Park J, Lee JY, So Kweon I (2018) Cham: convolutional block attention module. In: Proceedings of the European conference on computer vision, pp 3–19

24. Hu H, Gu J, Zhang Z, Dai J, Wei Y (2018) Relation networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3588–3597

25. Tang X, Du DK, He Z, Liu J (2018) Pyramidbox: a context-assisted single shot face detector. In: Proceedings of the European conference on computer vision (ECCV), pp 797–813

26. Wu J, Kuang Z, Wang L, Zhang W, Wu G (2020) Context-aware RCNN: a baseline for action detection in videos. arXiv preprint arXiv:200709861

27. Lin X, Ma L, Liu W, Chang SF (2020) Context-gated convolution. In: Proceedings of the European conference on computer vision. Springer, Cham, pp 701–718

28. Si J, Zhang H, Li CG, Kuen J, Kong X, Kot AC, Wang G (2018) Dual attention matching network for context-aware feature sequence based person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5363–5372

29. Nair V, Hinton GE (2010) Rectified linear units improve restricted Boltzmann machines. In: ICML

30. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708

31. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3213–3223

32. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L et al. (2019) Pytorch: an imperative style, high-performance deep learning library. In: Advances in neural information processing systems, pp 8024–8035

33. Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the sixteenth international conference on artificial intelligence and statistics, pp 249–256

34. Xu H, Lv X, Wang X, Ren Z, Bodla N, Chellappa R (2018) Deep regionlets for object detection. In: Proceedings of the European conference on computer vision (ECCV), pp 798–814

35. He K, Gkioxari G, Dollar P, Girshick R (2017) Mask R-CNN. In: Proceedings of the IEEE international conference on computer vision (ICCV), pp 2613–2621

36. He K, Gkioxari G, Dollar P, Girshick R (2017) Mask R-CNN. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 7393–7397

37. Bodla N, Singh B, Chellappa R, Davis LS (2017) Soft-NMS—improving object detection with one line of code. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 740–741

38. Tan Z, Nie X, Qian Q, Li N, Li H (2019) Learning to rank with recursive feature pyramid and switchable atrous convolution. arXiv preprint arXiv:190700981

39. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) SSD: single shot multibox detector. In: European conference on computer vision, Springer, pp 21–37
of the IEEE conference on computer vision and pattern recognition, pp 6877–6885

40. Chen K, Wang J, Pang J, Cao Y, Xiong Y, Li X, Sun S, Feng W, Liu Z, Xu J, Zhang Z, Cheng D, Zhu C, Cheng T, Zhao Q, Li B, Lu X, Zhu R, Wu Y, Dai J, Wang J, Shi J, Ouyang W, Loy CC, Lin D (2019) MMDetection: open mmlab detection toolbox and benchmark. arXiv preprint arXiv:190607155

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.