DENSE GLOBAL CONTEXT AWARE RCNN FOR OBJECT DETECTION CONFERENCE SUBMISSIONS

Anonymous authors
Paper under double-blind review

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, in this paper, we propose a novel end-to-end trainable framework, called Dense Global Context Aware (DGCA) 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 DGCA 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 fuse the global context information of different stages in the top-down process of FPN, and further leverage the attention mechanism to perform global context aware. Thus, the implicit relationship between object proposal and global features can be captured by neural networks to improve detection performance. Experimental results on COCO benchmark dataset demonstrate the significant advantages of our approach.

1 INTRODUCTION

Benefiting from the development and application of deep network technology in computer vision community, the performance of wide range of computer vision tasks such as target detection, semantic segmentation and instance segmentation have been greatly improved. In recent years, many excellent detection frameworks have been proposed. For example, there are one-stage methods with faster speed such as SSD (Liu et al., 2016) and YOLO (Redmon et al., 2016), and two-stage methods with better detection performance such as Faster RCNN (Ren et al., 2015) and FPN (Lin et al., 2017). Most of the currently popular two-stage 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 crop the regions of interest from the feature pyramid corresponding to different spatial scales, and then uniformly resizes them to a fixed spatial scale of 7×7 through RoIPool, and after flattening these feature maps, they are further encoded through two fully connected layers, and finally classification and positioning tasks are performed respectively. But is it reasonable to use only features of local receptive field like object proposal for classification and positioning? On the one hand, for classification tasks, the target object has a potential relationship with other objects in the background. For example, in real scenario, cups often appear on the dining table, and there are food, knives, forks and bowls around them, and laptop, keyboard, 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 decetion network to fuse the global context information with the local informations of the RoIs(Regions of Interest). 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, and then leverage our proposed context aware module to 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.

To provide evidence for these claims, in section 4 we develop several ablation studies and conduct an extensive evaluation on the COCO dataset (Lin et al., 2014). We also present results beyond COCO that indicate that the benefits of our approach are not restricted to a specific dataset. Our method gains +1.4 and +0.6 AP on MS COCO dataset from Feature Pyramid Network (FPN) baselines with ResNet-50 and ResNet-101 backbones.

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

1. We propose DGCA RCNN to extract and refine global context information by using dense connection and attention mechanism, and fuse the global features of different stages in the feature pyramid to calibrate the local features.

2. Unlike SENet (Hu et al., 2018b), which uses global context information for feature recalibration at the convolutional level, we extent it to assemble global context information on the two-stage target detection pipeline.

3. Our method can be easily deployed in other FPN based methods and can continuously improve their performance.

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

Figure 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.

2 Related Work

2.1 Object Detection

There are two common ways for object detection: one-stage and two-stage. Classic one-stage methods such as SSD (Liu et al., 2016),YOLO (Redmon et al., 2016; Redmon & Farhadi, 2017; 2018), etc. quickly classify and locate targets in an end-to-end manner. Classic two-stage methods such as Faster RCNN (Ren et al., 2015), FPN (Lin et al., 2017), etc. first obtain object proposals through the RPN(Region Proposal Network), and then use RoIPool/RoIAlign to align the spatial scales of these object proposals before performing detection. Most of the subsequent algorithms are also based on these two structures for continuous improvement and development. CornerNet (Law & Deng, 2018), ExtremeNet (Zhou et al., 2019b), CenterNet (Zhou et al., 2019a), and FCOS (Tian et al., 2019), etc., take advantage of the nature of the anchor box can formed by keypoint and optimize the

detection process of one-stage by improving the method of anchor generation. (Wu et al., 2020b) studies 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. Cascade r-cnn (Cai & Vasconcelos, 2018) 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 muti stage, in which each stage has a different IoU threshold. HTC (Chen et al., 2019a) tries to integrate semantic segmentation into the instance segmentation framework to obtain a better spatial context. DetectoRS (Qiao et al., 2020) proposed RFP (Recursive Feature Pyramid) and SAC (Switchable Atrous Convolution) to realize looking and thinking twice or more. (Wang et al., 2019) proposed to guide the generation of anchor through image features.

2.2 Context Aware

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 (Bello et al., 2019; Vaswani et al., 2017; Woo et al., 2018; Hu et al., 2018b;a) 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 task, 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 (Tang et al., 2018; Wu et al., 2020a). Context information can also be used in many ways, (Lin et al., 2019) proposed CGC(Context-Gated Convolution) to adaptively modify the weight of the convolutional layer. (Si et al., 2018) proposed DuATM (Dual Attention Matching network) to learn context-aware feature sequences, and perform pedestrian re-identification by performing sequence comparison simultaneously.

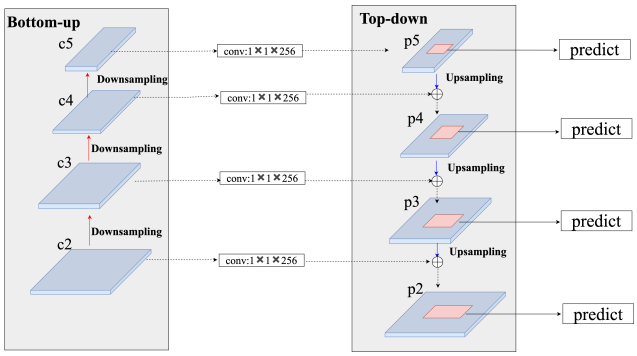

Figure 2: Network architecture of Feature Pyramid Network(FPN). In this figure, the upward red arrow represents the down-sampling process, the downward blue arrow represents the up-sampling process, the black dashed line represents the data flow, and $\oplus$ represents the element-wise sum.

3 Method

3.1 Motivation

In FPN, the feature pyramid is constructed through Bottom-up pathway, Top-down pathway and lateral connections, as shown in Fig. 2. Where Bottom-up pathway refers to the process of down-sampling the input image 5 times in the backbone network, and the output of the residual blocks corresponding to $\{conv2, conv3, conv4, conv5\}$ is denoted as $\{c2, c3, c4, c5\}$. Where Top-down pathway refers to the up-sampling process after convoluting c5 by 1×1 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 1×1 convolutional layer. In FPN, only using the cropped feature map of object proposal

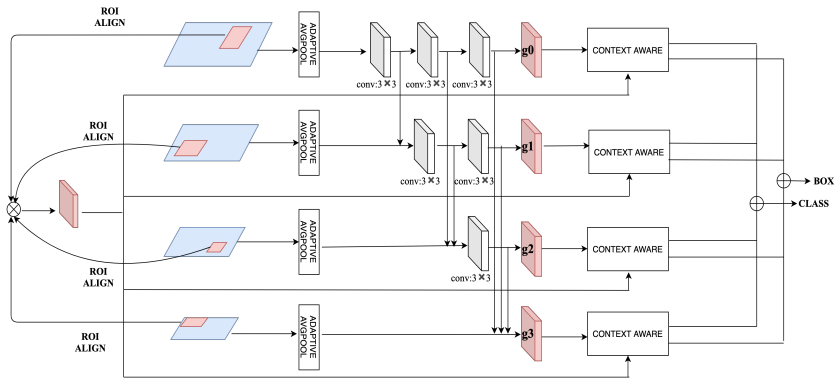

Figure 3: Network architecture of DGCA RCNN. In this figure, $\{p2, p3, p4, p5\}$ represents the feature pyramid used to generate RoI from the corresponding feature map and the channel dimensions are all 256, $\{g0, g1, g2, g3\}$ represents the global context feature map obtained by down-sampling $\{p2, p3, p4, p5\}$ through the corresponding 3×3 convolutional layer, $\otimes$ represents the concatenation function and $\oplus$ represents the element-wise sum.

to locate and classify the object will greatly lose the global context information, which will lead to the loss of the convolutional neural network's ability to perceive the relationship between the background and the 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

The structure of the dense global context module is depticted in Fig. 3. 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 $\{p2, p3, p4, p5\}$ to $(M, N) \times \{1, 1/2, 1/4, 1/8\}$ respectively. Then we continue to downsample these pooled feature maps using four parallel branches which containing $\{3, 2, 1, 0\}$ downsampling blocks respectively, each downsampling block refers to the composite function of two consecutive operations: a 3×3 convolution (conv) with stride 2 followed by a ReLU(rectified linear unit) (Nair & Hinton, 2010) 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 $\{g0, g1, g2, g3\}$. As a consequence, the benefits of the global context captured by multi-branch downsampling blocks can be accumulated through the network. The output feature g_i is defined by

$$g_i = D^{3-i}([\phi(p_{i+2}), \boldsymbol{W}_i]), \tag{1}$$

$$\boldsymbol{W} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} D^i(\phi(p_2)) & D^1[\phi(p_3), D^1(\phi(p_2))] & [g_0, g_1, g_2] \end{pmatrix}^T, \tag{2}$$

where D^i represents the total use of i downsampling blocks, ϕ stands for adaptive average pooling function, and where $\boldsymbol{W}_i$ represents the i-th row of matrix $\boldsymbol{W}$. After getting the feature set $\{g0, g1, g2, g3\}$ 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 (Huang et al., 2017), we define $[\phi(p_i), \boldsymbol{W}_i]$ as a concatenation of the feature-maps in it.

3.3 Context aware

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. In the attention module, inspired by (Hu et al., 2018b), we embedding these global context information $\{g0, g1, g2, g3\}$

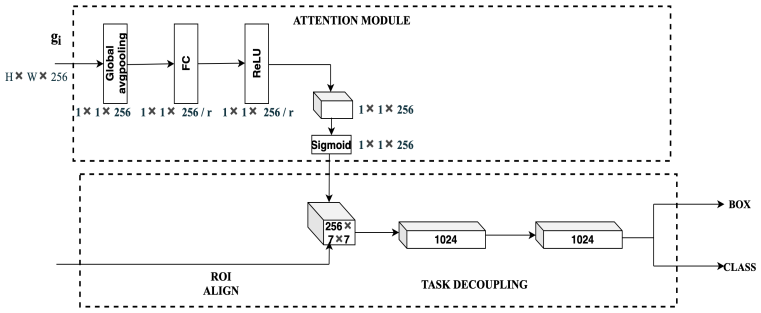

Figure 4: Network architecture of context aware module. It contains two sub-modules: attention and task decoupling, the attention module is in charge of mappings global context information to high-dimensional space and the task decoupling module outputs the prediction results of classification and positioning.

into higher-dimensional features for characterizing the global features. Specifically, we leverage the global average pooling layer to squeeze the spatial scale of $\{g0, g1, g2, g3\}$ 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. In the task decoupling module, we first do the channel-wise multiplication between the global context descriptor obtained by the squeeze-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 notice, 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 SENet, after each branch, we separately encode these fused feature information through two parallel fully connected layers, and then decouple the classification task and the positioning task. Finally, the one-dimensional features output by these four branches are fused through the element-wise sum method and then the object proposals are classified and located respectively.

4 Experiments

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/test. 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].

Table 1: Effect of dense connection on COCO val2017 (%)

Method	AP	$AP_{0.5}$	$AP_{0.75}$
FPN baseline	36.8	58.0	40.0
Without Dense connection(ours)	37.7	59.6	40.3
Dense connection(ours)	37.8	60.0	40.6

4.2 Implementation Details

Our model is end-to-end trained based on the torchvision detection module (Paszke et al., 2019), using SGD with 0.9 momentum, 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 pretrained on imagenet (Deng et al., 2009) is taken as the backbone networks on this dataset. Following the common practice, the size of the input image is adjusted to 800 for the short side and less or equal to 1333 for the long side. We train detectors for 22 epoches with an initial learning rate of 0.0025, and decrease it by 0.1 after 16 and 22 epoches, respectively. For data augmentation, we randomly flip the input image horizontally with a probability of 0.5. And all newly added convolutional layers are randomly initialized with the "xavier" method (Glorot & Bengio, 2010).

4.3 Ablation study

Dense connection: Table 1 shows the impact of whether $\{p2, p3, p4, p5\}$ these global context features 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 experiment 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 two branches of the object proposal $256 \times 7 \times 7$ feature map tensor and the one-dimensional global context information captured through global average pooling, we connect a fully connected layer with an output dimension of 512 respectively. Finally, we concatenate the two pieces of information together, and then connect 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 see that with or without dense connection, the AP value of our proposed method is improved relative to the FPN baseline, which also illustrates the importance of global context information. However, using dense connection can integrate global context information at different stages, so the model performance is better improved(outperforms FPN basline by 1.0% on COCO's standard AP metric and by 2.0% on AP@IoU=0.5). We used dense connections in all subsequent experiments.

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), 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 are $(128, 192) \times \{1, 1/2, 1/4, 1/8\}$ respectively. The comparison in Table 2 shows that performance dose not continuously increasing with bigger pooling size, we found that setting the initial pool size to (64, 96) achieved a good balance between memory consumption and accuracy, we used this value in all subsequent experiments.

The choices of the attention module: For the sake of better integrating global context information and local receptive field information, we explore different attention methods, the results of comparison are shown in Table 3. specifically, we directly multiply the 256-dimensional global context information output by the squeeze-excitation module with 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 added a new branch to the squeeze-excitation module and increase the output dimension to 1024, then multiply it respectively with the 1024-dimensional output of the first(attention on *"conv+fc1"*) and second fully connected layers (attention on *"conv+fc2"*) after the $256 \times 7 \times 7$

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

Pooling Size	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$
(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 3: Effect of different choices of the attention module on COCO val2017 (%)

Attention	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$
conv	38.2	59.9	41.0
fc1	37.5	58.8	40.6
fc2	37.6	59.4	40.1
conv+fc1	37.7	59.5	40.3
conv+fc2	37.7	59.2	40.6
conv+fc1+fc2	37.9	59.4	40.9

feature map tensor of the object proposal. In addition, we connected three parallel output branches after the squeeze-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 1.4% on COCO's standard AP metric.

Attention with different reduction ratio: The comparison in Table 4 shows the effect of different reduction ratios in the squeeze-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.

Bottom-up or Top-down: In the previous experiment, we obtained global context information by continuously down-sampling from top to bottom. In this experiment, we replace all newly added 3×3 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.

Additional datasets: We next investigate whether the benefits of global context information generalise to datasets beyond COCO. We perform experiments with our method on Cityscapes dataset which comprise a collection of 2975 training, 500 validate and 1525 test 2048×1024 pixel RGB images, and labelled with 8 classes. We train our model a total of 64 epochs, the learning rate is initially set to 0.0025 and drops by a factor of 10 after 48 epochs. We set the initial pooling size to (128, 256). The shorter edges of the images are randomly sampled from [800, 1024] for reducing overfitting, other parameters are the same as those set in the experiment on coco dataset. From Table 6 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.

Comparison with Baselines and State-of-the-art Methods on COCO: In Table 7, we compare the performance of our method with FPN baselines and Double-Head (Wu et al., 2020b) on

Table 4: Effect of different reduction ratios r on COCO val2017 (%)

ratio	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$
4	37.8	59.7	40.5
8	38.2	59.9	41.0
16	37.9	59.6	40.6

Table 5: Effect of Bottom-up and Top-down on COCO val2017 (%)

Method	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$
Bottom-up	37.4	58.7	40.2
Top-down	38.2	59.9	41.0

Table 6: Comparisons with FPN baseline on Cityscapes datasets with ResNet-50 backbone (%)

Method	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$	**AP**$_s$	**AP**$_m$	**AP**$_l$
FPN baseline	36.2	63.6	34.8	10.6	30.9	51.3
DGCA(ours)	37.4	63.8	38.2	13.0	31.9	52.2

COCO val 2017, where Double-Head(with DGCA) means to assemble our method on Double-Head RCNN (Wu et al., 2020b) based on mmdetection (Chen et al., 2019b), note that in this method we only applied our method 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(1.4% improvement with ResNet-50 and 0.6% improvement with ResNet-101) and model(0.7% compared to Double-Head). Table 8 shows the comparison between our method with the state-of-the-art methods on MS COCO 2017 test-dev. Compared with FPN baseline, our method is better at detecting small and medium targets, which is due to the increased connection between global context and local information.

Table 7: Object detection results (bounding box AP) on COCO val2017.

Method	**Backbone**	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$	**AP**$_s$	**AP**$_m$	**AP**$_l$
FPN baseline (Lin et al., 2017)	ResNet-50	36.8	58.7	40.4	21.2	40.1	48.8
DGCA(ours)	ResNet-50	38.2	59.9	41.0	22.7	42.0	49.0
FPN baseline (Lin et al., 2017)	ResNet-101	39.1	61.0	42.4	22.2	42.5	51.0
DGCA(ours)	ResNet-101	39.7	61.0	43.3	23.0	43.7	51.3
Double-Head (Wu et al., 2020b)	ResNet-50	39.5	59.8	43.0	23.2	43.2	51.1
Double-Head(with DGCA)	ResNet-50	40.2	61.0	44.0	24.1	43.8	52.6

Table 8: Object detection results (bounding box AP) on COCO test-dev.

Method	**Backbone**	**AP**	**AP**$_{0.5}$	**AP**$_{0.75}$	**AP**$_s$	**AP**$_m$	**AP**$_l$
FPN (Lin et al., 2017)	ResNet-101	37.3	59.6	40.3	19.8	40.2	48.8
Mask RCNN (He et al., 2017)	ResNet-101	38.2	60.3	41.7	20.1	41.1	50.2
DGCA(ours)	ResNet-101	40.0	61.6	43.5	22.8	43.2	50.3
Double-Head (Wu et al., 2020b)	ResNet-50	39.8	60.2	43.4	23.0	42.7	49.8
Double-Head(with DGCA)	ResNet-50	40.3	61.1	43.9	23.7	43.0	50.7

5 CONCLUSION

In this paper, we propose Dense Global Context Aware (DGCA) 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.

REFERENCES

Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3286–3295, 2019.

Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154–6162, 2018.

Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4974–4983, 2019a.

Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019b.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3588–3597, 2018a.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018b.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 734–750, 2018.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

Xudong Lin, Lin Ma, Wei Liu, and Shih-Fu Chang. Context-gated convolution. 2019.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pp. 21–37. Springer, 2016.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv preprint arXiv:2006.02334*, 2020.

Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, 2017.

Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pp. 91–99, 2015.

Jianlou Si, Honggang Zhang, Chun-Guang Li, Jason Kuen, Xiangfei Kong, Alex C Kot, and Gang Wang. 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, 2018.

Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. Pyramidbox: A context-assisted single shot face detector. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 797–813, 2018.

Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 9627–9636, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Jiaqi Wang, Kai Chen, Shuo Yang, Chen Change Loy, and Dahua Lin. Region proposal by guided anchoring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2965–2974, 2019.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

Jianchao Wu, Zhanghui Kuang, Limin Wang, Wayne Zhang, and Gangshan Wu. Context-aware rcnn: A baseline for action detection in videos. *arXiv preprint arXiv:2007.09861*, 2020a.

Yue Wu, Yinpeng Chen, Lu Yuan, Zicheng Liu, Lijuan Wang, Hongzhi Li, and Yun Fu. Rethinking classification and localization for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10186–10195, 2020b.

Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019a.

Xingyi Zhou, Jiacheng Zhuo, and Philipp Krahenbuhl. 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, 2019b.