.

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Bidirectional Parallel Feature Pyramid Network for Object Detection

ZHENGNING ZHANG[1,5], LIN ZHANG[2], (MEMBER, IEEE), YUE WANG[3], PENGMING FENG[4], (MEMBER, IEEE), AND BAOCHEN SUN[4]

[1]Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China (e-mail: 23880666@qq.com)
[2]Tsinghua Shenzhen International Graduate School, Shenzhen, 518000, China
[3]State Key Laboratory of Space-Ground Integrated Information Technology, CAST, Beijing, 100095, China
[4]Aerospace ShenZhou Smart System Technology Co., Ltd., CAST, Beijing, 100095, China
[5]Space Star Technology Co.Ltd., CAST, Beijing, 100095, China

Corresponding author: Zhengning Zhang (e-mail: 23880666@qq.com).

This work was jointly supported in part by the Shenzhen Science and Technology Program under Grant KQTD20170810150821146 and Civil Aerospace Technology Advance Research Project of China under Grant D040405.

ABSTRACT State-of-the-art Feature Pyramid Networks (FPNs) often focus on extracting features across different levels. In this paper, we propose a novel architecture, Bidirectional Parallel Feature Pyramid Network (BPFPN), to capture multi-scale spatial information from each level of FPN effectively. BPFPN consists of two blocks: Cross-level Channel Attention-Refinement (ClCSAR) Block and Weighted Parallel Feature Aggregation (WPFA) Block. ClCSAR block uses a channel attention mechanism to strengthen the context information of lower-level feature with aid from the upper-level feature. WPFA block exploits discriminating information from variable receptive fields via integrating multi-branch by employing dilated convolutions and using attention mechanisms to capture the salient dependencies over branches. Considering the incremental computation, we also give a lightweight version of BPFPN, namely BPFPN-Lite, integrated with an Efficient WPFA (E-WPFA) to improve detection accuracy while maintaining efficiency. Our proposed network can be easily plugged into existing object detection models and outperforms different feature pyramids methods by $0.2 \sim 2.1$ on the COCO test-dev benchmark without bells and whistles.

INDEX TERMS Object detection,Feature pyramid network,Dilated convolution

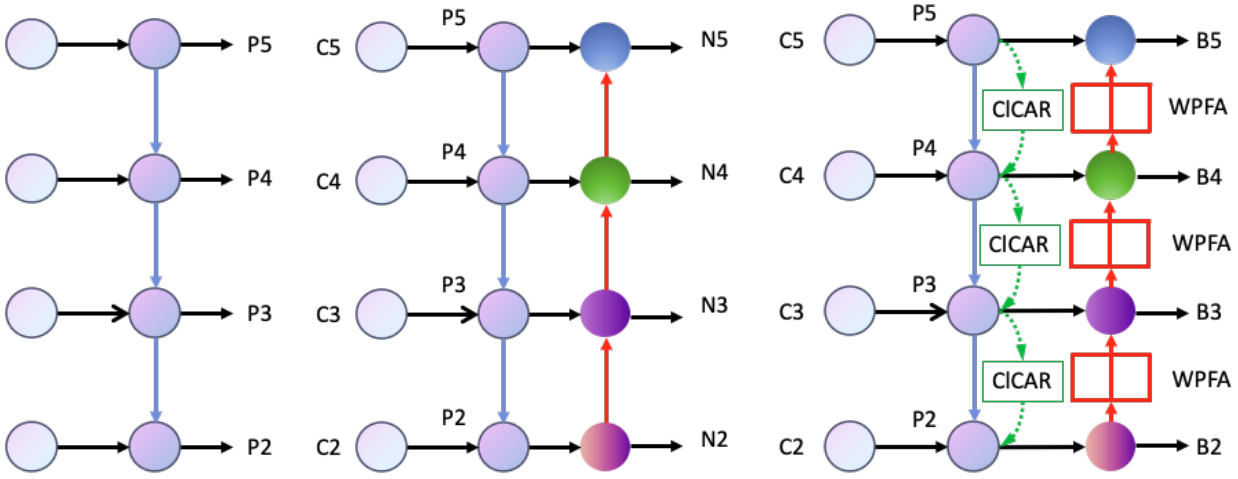
I. INTRODUCTION

Object detection, aiming to classify and locate targets, is one of the most important tasks in computer vision work. Detecting multiple objects across a wide range of scales is one of the major challenges in object detection, the key of which is to extract features with multiple scales more accurately and efficiently, especially in detectors based on Convolution Neural Network (CNN), e.g. remote sensing [1].

Feature Pyramid Network (FPN) [2], as shown in Figure 1(a), taking advantage of generating pyramidal feature representations through a top-down pathway and lateral connections to combine multi-scale features, has become popular in modern object detectors. However, FPN is limited by the single-track information flow. Previous work mainly dealt with this challenge from two aspects: design new network architecture (such as extra cross-scale connections) or fundamental operations (such as new feature upsampling operator). In terms of network architecture, PANet [3] extends pyramid network architecture by introducing an extra bottom-up pathway and achieves better accuracy based on the FPN, as shown in Fig.1(b). ARDFPN [4] uses a set of dilated convolutions in top-down connections to obtain the correlations of regions in the feature pyramid. In terms of fundamental operations, CARAFE [5] proposes a universal and lightweight feature upsampling operator to boost the performance of dense prediction tasks.

However, we notice the performance of existing FPNs are limited by only extracting multi-scale features from different levels and simply merging them, but rarely involve extraction of multi-scale features from single-level and intra-layer feature aggregation. In this work, we move beyond these limitations by expanding the receptive field while maintaining feature with different scales, extracting features in parallel with bidirectional way from each level of FPN, and maintaining compute efficiency. Motivated by this idea, we present a novel Bidirectional Parallel Feature Pyramid Network (BPFPN) for object detection, as shown in Fig.1(c). Our contributions are summarized as following aspects.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3173732, IEEE Access



(a) FPN (b) PANet (c) BPFPN

FIGURE 1. Feature network design (a) FPN [2] introduces top-down connections to fuse multi-scale features; (b) PANet [3] introduces bottom-up connections based on FPN; (c) is our BPFPN with cross-level attention connections and multi-branch bottom-up connections.

- We design a cross-level channel attention-refinement (ClCSAR) mechanism to strengthen the context information of the lower-level features with aid from upper-level features.
- We propose a novel weighted parallel feature aggregation (WPFA) block to capturing multi-scale features from each level of FPN. To the best of our knowledge, we are the first to take advantage of parallel feature aggregation in FPN.
- We develop a lightweight yet effective E-WPFA block based on WPFA to reduce the number of parameters and the computational cost.

We evaluate our approach equipped with variable detectors and backbones on the MS-COCO dataset, experiments show that BPFPN is efficient and generalizes well across detectors.

II. RELATED WORK

A. CNN BASED OBJECT DETECTION

CNN based object detectors firstly employs backbone network to extract features, then two tasks, namely classification and regression are involved. According to network architecture, existing CNN-based detectors can be briefly categorized into two streams: anchor-based and anchor-free detectors. Anchor-based detectors can be generally divided into one-stage methods such as SSD [6] and RetinaNet [7], and multi-stage methods such as Faster R-CNN [8] and Cascade R-CNN [9]. Both of them firstly generate a vast number of anchors on feature maps, then predict the category, refine coordinates of each anchor, and finally output the refined anchors as detection results. One-stage methods directly adopt a unified framework to predict the bounding boxes, while multi-stage methods are driven by a region proposal network (RPN), where the final detection results are generated after extracting region proposals. Nowadays, one-stage methods have attracted much attention by providing high inference speed, while multi-stage methods often hold state-of-the-art on standard detection benchmarks. Anchor-free detectors can be categorized into two types: key-point based methods such as Reppoints [10], and center-based methods such as FCOS [11].

Many algorithms are presented to improve the detection performance and boost speed, including redesign information flow [12], enhance feature fusion [13], anchor refinement [14], attention mechanism [15], and loss function [16]. Existing object detectors are suffering from the problems caused by the multi-scale variation. Feature pyramid network is popularly used in different types of detectors to deal with scale variation.

B. FEATURE PYRAMID NETWORK IN OBJECT DETECTION

Effectively representing and processing multi-scale features is one of the main difficulties and fundamental problems in object detection. In earlier years, many detectors such as SSD [6], and Overfeat [17] utilize multi-level feature maps from backbone networks for prediction. Inspired by the success of residual connection, RON [18] proposes a reverse connection to fuse feature maps from high-level to low-level. Feature Pyramid Network (FPN) [19] is a pioneering work that generates pyramidal feature representations through a top-down pathway and lateral connections, hence combines multi-scale features. FPN is widely exploited by both the state-of-the-art one-stage object detectors and the multi-stage object detectors.

Based on the FPN, PANet [3] extends pyramid network architecture by introducing a coordinate bottom-up pathway and achieves better accuracy. Fu Zhihang et al. propose a previewer block, which previews the objectiveness probability for the potential regression region of each prior box, using the stronger features with larger receptive fields and more

contextual information for better predictions. Qijie Zhao et al. proposed MLFPN to extract multi-level features from the backbone, which are then fed into a Thinned U-shape Module (TUM) to generate a group of multi-scale features. However, MLFPN is complex and time-consuming [20]. Bi-FPN [21] simplifies the PANet by removing nodes with only one input edge and adding a new cross-scale connection to improve model efficiency. Bi-FPN also introduces learnable weights to learn the importance of different input features while repeatedly applying top-down and bottom-up multi-scale feature fusion. DLA [22] further iteratively and hierarchically merge the feature hierarchy. XIAOTONG ZHAO et al. proposed ARDFPN to exploit the inherent correlation of regions in feature pyramid. ARDFPN stacking a building block that aggregates a set of dilated convolutions with the same topology in the top-down connections. CARAFE [5] proposes a universal and lightweight feature upsampling operator to boost the performance of dense prediction tasks. Recursive-FPN [23] incorporates extra feedback connections from FPN into the bottom-up backbone layers. AugFPN [24] focus on adaptive feature fusion mechanism and proposes a cross channel attention module to enhance the multi-scale features with consistent supervision. Pang et al. put forward the balanced feature pyramid, which is used to fuse and balance the features among different layers in Libra R-CNN. DyFPN [25] uses the adaptable selected branch to calculate output features according to a learnable gating operation. Linxiang Zhu et al. proposed ImFPN to fuse different features with the aim of adapting to varying sizes of instances. PRB-FPN is a parallel residual bi-fusion structure, which uses a bottom-up fusion module and a concatenation and re-organization module to detect multi-scale objects at once and recover lost information [26].

Besides above hand-designed architectures, NAS-FPN [19] adopts Neural Architecture Search to design feature network topology automatically. It consists of a combination of top-down and bottom-up connections to fuse features across variable scales. Auto-FPN [27] searches for a better fusion of the multi-level features, resulting in a more lightweight solution when compared with NAS-FPN.

Related to these previous efforts, BPFPN tries to introduce a new parallel pathway structure that could extracting multi-scale features from each level of FPN.

C. DILATED CONVOLUTION

Dilated convolution [28] is originally developed with name atrous convolution, which is proposed for efficient computation of undecimated wavelet transformation. In dilated convolutions, the alignment of the convolution kernel with original weights is expanded by performing convolution at sparsely sampled locations. By increasing the dilation rate, the kernel size are enlarged and their corresponding weights are placed far away from given intervals. Recent studies show that the increasing of the effective receptive field of convolution kernel is essential to feature aggregation with more spatial information [12], while dilated convolution is a practical approach to enlarge the receptive field of convolutions without losing some spatial information compared with the pooling method.

Dilated convolution has been widely used in current computer-vision tasks, such as semantic segmentation to incorporate large-scale context information. DeepLab [29] uses dilated convolutions in the semantic image segmentation tasks and then improved upon in [30], [31], [32]. In order to increase efficiency of ASPP method, Sachin Mehta et al. [33] introduce a new convolution module, efficient spatial pyramid (ESP), which is faster, smaller, lower power and lower latency. Mou et al. [34] propose dense dilated block (DDB) to organize dilated convolutions in a dense form, and summarize the dilated convolutions structure into three different mode: cascade mode, parallel mode, and dense mode. F.Yu and V.Koltun [35] first present a cascaded combination structure of dilated convolution layers with exponentially increasing rates of dilation. SDC (stacked dilated convolution) [36] combines parallel dilated convolutions with different dilation rates as a unified descriptor network for dense matching tasks. DenseASPP [37] connects a set of dilated convolution layers in a dense way, resulting in generating multi-scale features, which cover a larger scale range densely.

In object detection field, Zeming Li et al. [38] designs a specific detection backbone network (DetNet) to maintain the spatial resolution, keep the efficiency, and enlarge the receptive field using dilated convolution. Motivated by DetNet, TridenNet [12] also employs dilated convolution in multi-branch architecture with different dilation rates and shares the same transformation parameters to adapt the receptive fields for objects with different scales.

III. METHOD

A. OVERALL ARCHITECTURE

Denoting $\{C_i, C_{i+1}, C_{i+2}, \ldots, C_{i+n}\}, i, n \in \mathbb{R}$, as feature maps from each level of backbone network, conventional top-down FPN aggregates the corresponding multi-scale features maps $\{P_i, P_{i+1}, P_{i+2}, \ldots P_{i+n}\}$ in a top-down way as:

$$P_i = Conv_{1\times1-s1-d1}(P_i) + Resize(P_{i+1}) \qquad (1)$$

where i denotes the level of the FPN, $Conv_{1\times1-s1-d1}$ denotes a lateral convolutional layer with $kernelsize = 1 \times 1$. $stride = 1$, and $dilationrate = 1$; $Resize$ is an upsampling operation for resolution matching.

By replacing the bottom-up connections of PANet by multi-branch bottom-up connections, the proposed BPFPN has two novel components: 1) top-down Cross-level Channel Attention-Refinement (ClCSAR) block and 2) bottom-up Weighted Parallel Feature Aggregation (WPFA) block.

As shown in Figure 1(c), each ClCSAR block takes a coarser-resolution feature map P_{i+1} to compute cross-level channel attention weights. The upsampled weights are then merged with the corresponding higher-resolution feature map P_i by element-wise multiplication and are then combined with the original P_i. Each WPFA block takes a higher resolution feature map B_i and a coarser map P_{i+1} through

lateral connection and generates a new feature map B_{i+1} as the output feature map of BPFPN.

B. CROSS-LEVEL CHANNEL-SPATIAL ATTENTION-REFINEMENT BLOCK

To model the context relationship between adjacent levels of FPN and enhance the representation ability of feature maps, we propose Cross-level Channel Channel-Spatial Attention-Refinement (ClCSAR) block. As shown in Fig.1, the channel attention mechanism is used to guide the network to learn the channel-wise weights from high-level feature maps and then refine adjacent low-level feature maps. Inspired by CBAM [39], we adopt the Channel Attention (CA) module to compute the cross-level channel attention weights efficiently, as shown in Fig.2. Given two adjacent feature maps P_i and P_{i+1} , we first obtain the channel-wise weights $Weight_{i+1} \in \mathbb{R}^{C\times H_{i+1}\times W_{i+1}}$ from the higher-level but coarser-resolution feature P_{i+1} by a CA module and then upsample $Weight_{i+1}$ to the same size as P_i (nearest neighbor upsampling for simplicity), denoted as $Weight_i \in \mathbb{R}^{C\times H_i\times W_i}$, where C, H, W indicate the channel number, spatial height, and width, respectively. Mathematically, $Weight_i \in \mathbb{R}^{C\times H_i\times W_i}$ is computed as:

$$\begin{aligned} Weight_i &= Upsample(Weight_{i+1}) \\ &= Upsample(CA(P_{i+1})) \\ &= Upsample(\sigma(MLP(AvgPool(P_{i+1})) \\ &+ MLP(MaxPool(P_{i+1})))) \\ &= Upsample(\sigma(W_1(W_0(P_{i+1}^{avg})) \\ &+ W_1(W_0(P_{i+1}^{max})))) \end{aligned} \tag{2}$$

where σ denotes the Sigmoid excitation function, MLP denotes a shared network composed of one multi-layer perception layer and one hidden layer [39], r is the hidden activation function ratio, P_{i+1}^{avg} and P_{i+1}^{max} denote average-pooled features and max-pooled features from average pooling $AvgPool$ and max pooling $MaxPool$, respectively. $W_0 \in \mathbb{R}^{C/r\times C}$ and $W_1 \in \mathbb{R}^{C\times C/r}$ are shared weights of MLP, a Re-LU activation function is followed by W_0. Then, we directly combine P_i with the channel-wise $Weight_i$ by element-wise multiplication and add the result to the original P_i get P_i'. Mathematically,

$$P_i' = Weight_i \odot P_i + P_i \tag{3}$$

Where $\odot$ denotes the element-wise multiplication. By employing ClCSAR block, feature maps with more accurate channel-wise context relationship are obtained, in next section, WPFA block is introduced to further enhance feature extraction.

C. WEIGHTED PARALLEL FEATURE AGGREGATION BLOCK

The WPFA block mainly solves two problems: Firstly, the fixed receptive field within bottom-up connections can hardly satisfy the extraction of multi-scale features from each level of FPN. Secondly, different receptive fields contain different semantic information, hence should not be merged with same weight. The diagram of the WPFA block is shown in Fig.3. Inspired by Deeplab v3 [40], we adopt Atrous Spatial Pyramid Pooling (ASPP) module with different dilation rates to capture multi-scale information. Inspired by EPSANet [41], we introduce Branch Refinement (BR) block to sort the importance of branches by weights and generate more informative output. The BR block consists of two parts: 1) squeeze and excitation module (SEWeight [12]) and Softmax module [41].

As shown in Fig.3, WPFA block is mainly implemented in five steps. Firstly, the multi-scale feature maps are captured by the ASPP module with k different dilated convolution ($dilationrate = d_j, j = 1, 2, \dots k$) branches and a global average pooling branch. The multi-scale feature map generation function is given by:

$$P_{ij}'' = Conv_{3\times3-s1-d_j}(P_i'), j = 1, 2, \dots, k \tag{4}$$

Secondly, the channel-wise attention vector is obtained by BR block to extract the attention of the feature map with different scales. Mathematically, the $c-th$ channel refinement weight of the $j-th$ branch feature map P_{ij}'' can be represented as:

$$\begin{aligned} RW_{ij}^c &= Softmax[SEWeight(P_{ij}''^c)] \\ &= Softmax[\sigma(W_{f1}\delta(W_{f0}(g_c(P_{ij}''^c))))], \\ &j = 1, 2, \dots, k \end{aligned} \tag{5}$$

After which, element-wise multiplication are applied to recalibrate weight of multi-branch attention RW_{ij} and the P_{ij}'' of corresponding branch feature map:

$$Y_{ij} = P_{ij}'' \odot RW_{ij}, j = 1, 2, \dots, k \tag{6}$$

By concatenating the resulting features from all the branches and are passing through two convolution layers, feature map are generated with multiple scales and reduce the spatial size, which can be written as:

$$\begin{aligned} \tilde{P}_i = &Conv_{3\times3-s2-d1}(\\ &Conv_{1\times1-s1-d1}(Cat[Y_{i1}, Y_{i2}, \dots, Y_{ik}])) \end{aligned} \tag{7}$$

Finally, each element of the final feature map $\tilde{P}_i$ and P_{i+1}' are added, and are processed by another convolution layers to generate B_{i+1} for the following levels. This process is iterated until the latest feature map is generated. Mathematically,

$$B_{i+1} = Conv_{3\times3-s1-d1}(\tilde{P}_i + P_{i+1}') \tag{8}$$

By outputting $B_{i+1}, i = 2, \dots, n$, the proposed BPFPN aggregates features in parallel with WPFA block effectively.

D. EFFICIENT WEIGHTED PARALLEL FEATURE AGGREGATION BLOCK

The proposed WPFA block can largely improve the detection accuracy but also brings massive parameters and yield extra computation expenses. Since there is a lot of information

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3173732, IEEE Access

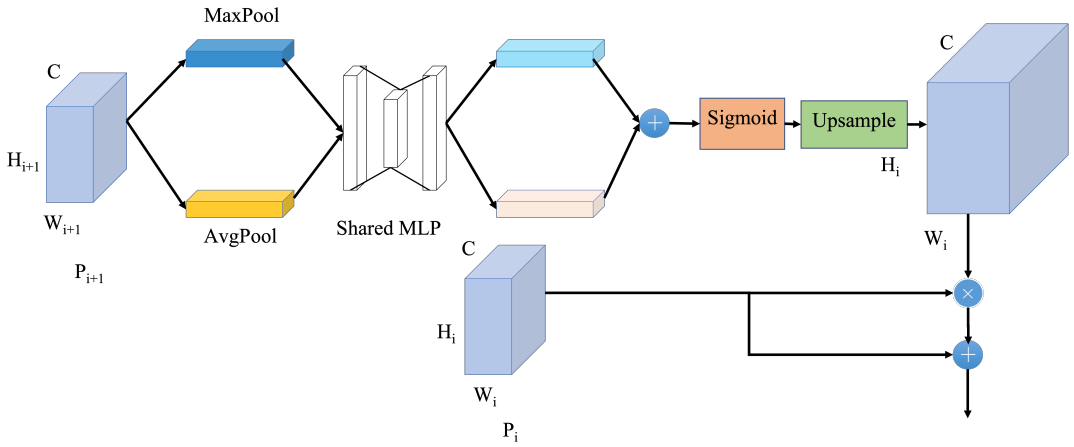


FIGURE 2. Architecture of CICSAR Block. P_i and P_{i+1} are two adjacent feature maps, MLP is a shared network.

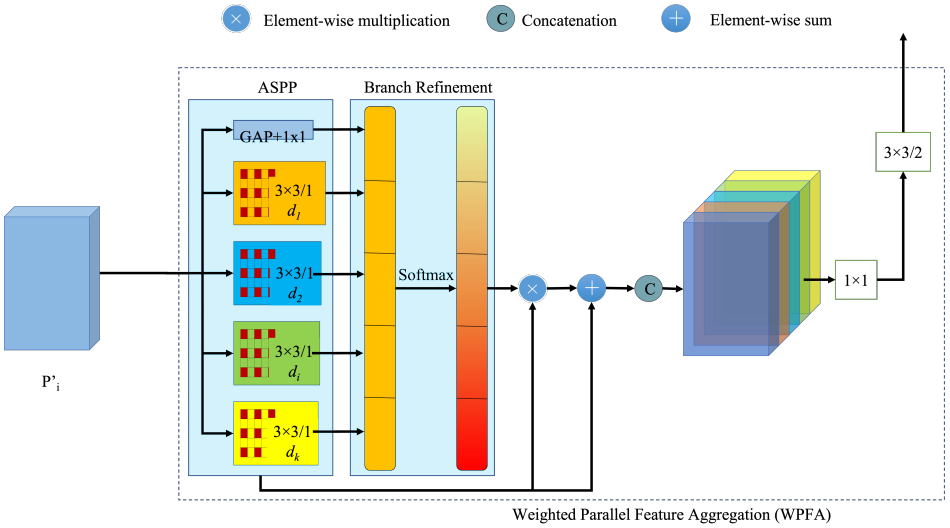


FIGURE 3. Architecture of WPFA Block.

redundancy between different channels of input features, the computational complexity can be reduced by adopting the channel split mechanism method. In this case of the bias between different parts of ASPP, we use a branch refinement module to adjust the weights of different parts. In this case, we propose an Efficient WPFA (E-WPFA) block, as shown in Fig.4, which aims at tackling the problems of WPFA by introducing a channel split mechanism before multi-branch parallel dilated convolutions. The E-WPFA has an extra component: the Channel Split module (CS). CS module split input feature map P'_i into m parts as denoted by $[P'_{i-1}, P'_{i-2}, \ldots, P'_{i-m}]$ along with channel dimensions, i.e., $P'_{i-j} \in \mathbb{R}^{C/s \times H \times W}, j = 1, 2, \ldots, m$. After obtaining the sub-feature, they are fed into dilated convolutions respectively with different rates. In addition, to merge global information elaborately, we employ global average pooling (GAP) on the input feature map and feed the resulting feature to a 1×1 convolution, then upsample the feature to the same spatial dimension as the other SP branch. Then, we generate the corresponding importance coefficient for each sub-feature through a BR module. The output of the BR module can be obtained by:

$$Y'_i = f_i \odot BRWeight_i, i = 0, 1, \cdots, S \tag{9}$$

After BR, all branches are concatenated and pass through a 1×1 convolution. The final feature map of E-WPFAB can be written as:

$$\tilde{F}' = Conv_{3\times3/2}(Conv_{1\times1}(Cat[Y'_0, Y'_1, \cdots, Y'_S])) \tag{10}$$

IV. EXPERIMENTS AND RESULTS

A. DATASET, EVALUATION METRICS, AND EXPERIMENTS SETTING

We evaluate the performance of the proposed BPFPN architecture on the MS-COCO [42] and ShipRSImageNet [1] dataset. MS-COCO dataset contains 80 categories, 115k images for training (train2017), 5k images for validation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3173732, IEEE Access

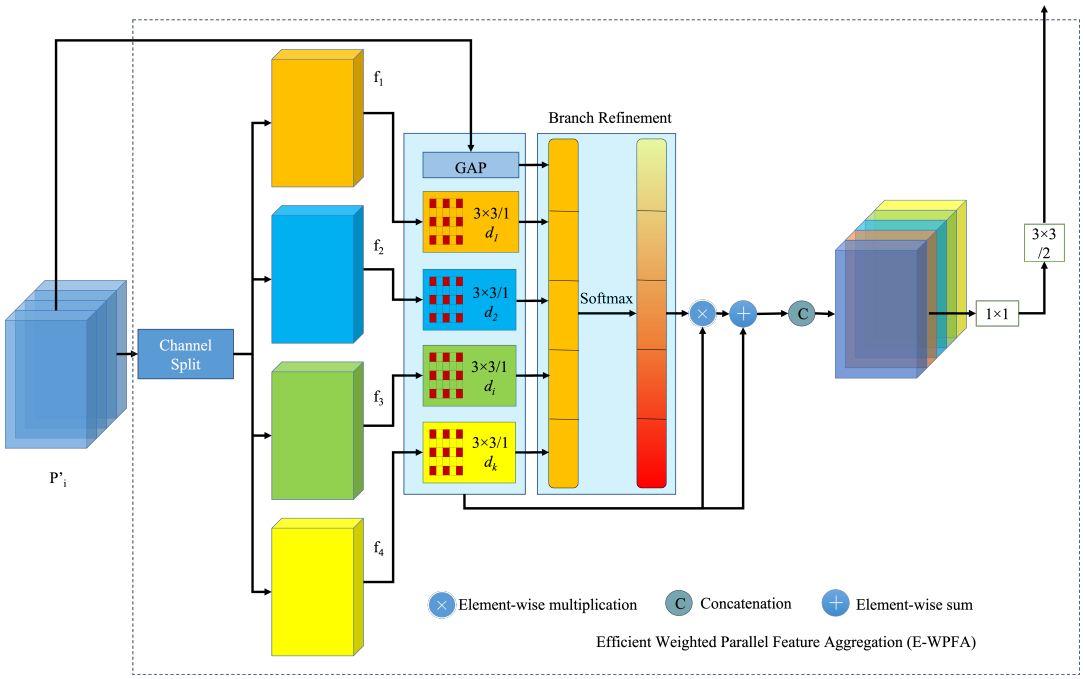


FIGURE 4. Architecture of E-WPFA Block.

(val2017), and 20k images for testing (test-dev). The model is trained on train2017 and evaluated on val2017. ShipRSImageNet is the largest fine-grained remote sensing dataset for ship detection. It contains over 3435 images with 17573 ship instances in 50 categories, providing precise horizontal and orientated bounding boxes annotation. Following common practice, metrics for performance evaluation include the standard average precision AP, AP_{50} , AP_{75}, AP_S, AP_M, AP_L. All the evaluation are implemented with the PyTorch framework [43] and MMdetection toolkit [44] in default settings. For all experiments, we use 2 NVIDIA RTX 3090 GPUs each with 24GB memory, batch size is set as 4, and the SGD optimizer is utilized to train the model. Models are trained 12 epochs and 100 epochs for MS-COCO and ShipRSImageNet dataset, respectively; the initial learning rate is set as 0.01 and decreases by a ratio of 0.001 in the $8th$ and $11th$ epoch. Note that the BPFPN takes level 2-5 features $\{P_2, P_3, P_4, P_5\}$ from the backbone network during all experiments, and the outputs produced are $\{B_2, B_3, B_4, B_5\}$.

B. EVALUATION RESULTS ON COCO

1) Qualitative Evaluation

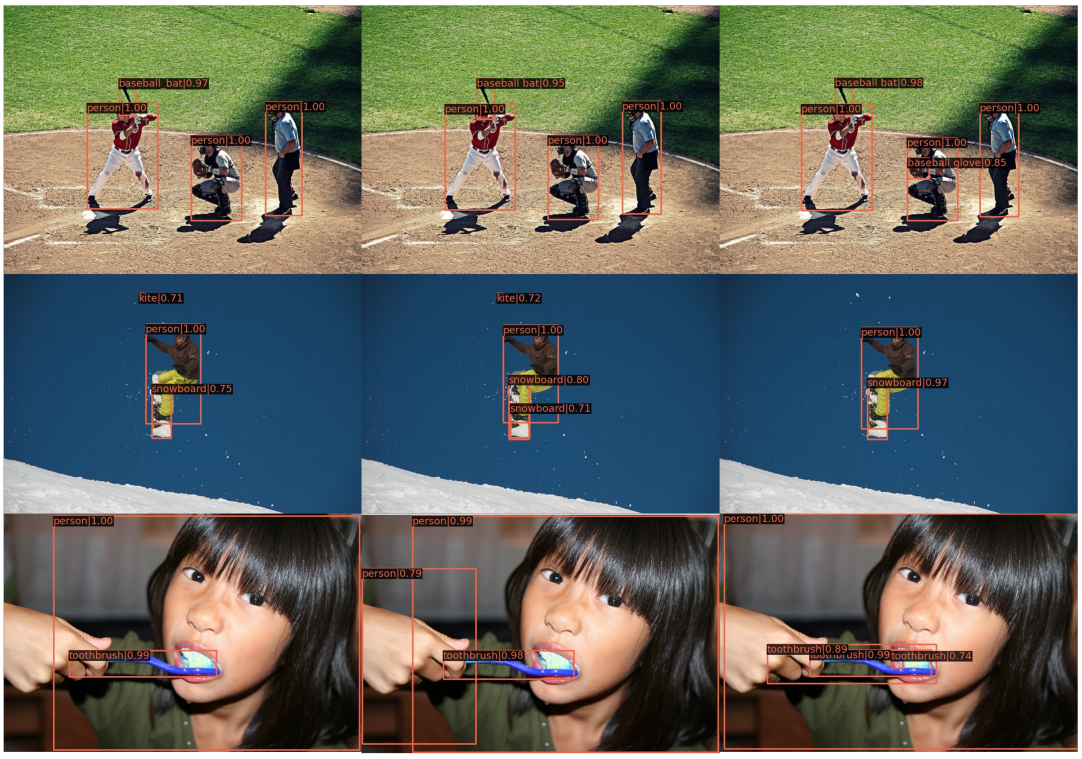
We qualitatively compare the differences between and BPFPN on MS-COCO val2017 [42]. The sample detection results are illustrated in Fig.5. For a fair comparison, we employ ResNet50 as backbone network and compare the detection performance of the same images with score-threshold = 0.7. Fig.5 shows that our BPFPN effectively captures multi-scale features, and can detect objects more accurately with fewer false-positive boxes. Besides, BPFPN can capture the small objects and performs better on ambiguous objects by exploring more discriminating features from variable receptive fields. In the first row of Fig.5, there is a correct BPFPN prediction of 'baseball glove', while neither FPN nor PAFPN detects it. In the second row of Fig.5, it can be seen that BPFPN can distinguish very similar objects such as 'snowflakes' and 'kites' in the sky well, while FPN and PAFPN incorrectly identify them as 'kite'. In the third row of Fig.5, BPFPN correctly detect 'person', but mistakenly identifies one 'toothbrush' as three 'toothbrushes'. We hypothesize that the negative results in the third row of Fig.5 may be due to the fact that BFPPN is too sensitive to small objects.

2) Quantitative Evaluation

We compare the detection performance of proposed BPFPN based on Faster R-CNN framework with existing state-of-the-art FPN architecture. Table 1 reports the experimental results achieved with the above-mentioned competitive methods on MS-COCO 2017 test-dev with various settings. To make more fair comparison, the performance BPFPN are also validated on the MS-COCO val2017 with four different types of popular detectors. As shown in Table 2, BPFPN improves the AP accuracy of the all the anchor-free and anchor-based detectors by a clear margin. The results demonstrate that our proposed modules can effectively capture information with multiple scales on each level, hence greatly facilitates detection performance.

the BPFPN with ResNet-50 offers a highly competitive performance compared to the FPN with the same backbone. In addition, with a more powerful backbone network, BPFPN can still bring significant performance improvements. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3173732, IEEE Access



FPN PAFPN BPFPN

FIGURE 5. Sample detection results from FPN, PAFPN and our BPFPN based method on COCO val2017. The first column lists the results of FPN, The second column lists the results of PAFPN, while the third column lists those of BPFPN.

BPFPN improves the AP accuracy of the Cascade Mask R-CNN using ResNet-50 as the backbone to 43.3%, which outperforms all the anchor-free and anchor-based detectors with the same backbone by a clear margin. The results demonstrate that our proposed modules can effectively capture multi-scale information at each level of the feature pyramid, which in turn greatly facilitates object detection.

More specifically, the BPFPN with ResNet-50 offers a highly competitive performance compared to the FPN with the same backbone, i.e., by 2.6%, 2.1%, 0.7%, 1.1%, and 0.7% AP higher in bounding box AP on the Faster-RCNN [8], Cascade Mask R-CNN [48], RetinaNet [7], Reppoints [10], FCOS [11], VarifocalNet [49], respectively. In addition, with a more powerful backbone network, BPFPN can still bring significant performance improvements. When using 1x schedule ResNet101 or ResNext101-64x4d as the backbone, the proposed BPFPN improves the box AP metric by 1.8% and 0.9%, respectively.

We compare the performance of BFPPN with standard FPN method on the two-stage object detection network Faster R-CNN and the typical one-stage network Retinanet. Table3 presents the experimental results. The results show that the proposed BPFPN improves the box mAP metric by 2.8% and 2.6%, respectively. BPFPN can bring significant performance improvements from small objects.

V. ABLATION STUDY

We perform ablation studies to analyze the performance of individual components in our BPFPN on MS-COCO val2017. Our ablation experiments are conducted on Faster-RCNN with ResNet-50 as the backbone.

A. ABLATION STUDIES FOR EACH COMPONENT

To analyze the importance of each component in BPFPN, we apply ClCSAR Block, ASPP Module, and BR Module separately to the model to verify the effectiveness and accuracy. We also present the improvements brought by a combination of different modules. As shown in Table 4, the combination of ClCSAR and ASPP or ASPP and BR can also improve the baseline by 2.3 and 2.2 AP, which proves that parallel feature aggregation can benefit from cross-level attention and channel refinement. When all components are integrated into the PAFPN, the AP can be improved to 40.0%, which shows that these three modules complement each other. We also perform ablation studies on two pooling methods of the ClCSAR module, and the results show that the max pooling method is the best when only the ClCSAR module is used. When ClCSAR is used with ASPP module, the combination of average pooling and max pooling works best.

ASPP Module is added separately to PAFPN, the baseline performance is improved by 0.3, 2.0 AP, respectively. The results show that the combination of ClCSAR and ASPP

TABLE 1. Comparison with the state-of-the-art feature pyramid methods on COCO test-dev.

Method	*Backbone*	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster R-CNN+FPN* [2]	R50	37.7	58.7	40.8	21.7	40.6	46.7
Faster R-CNN+FPN* [2]	R101	39.7	60.7	43.2	22.5	42.9	49.9
Faster R-CNN+FPN* w/CARAFE [5]	R50	38.1	60.7	41.0	22.8	41.2	46.9
Faster R-CNN+DRFPN [45]	R50	39.6	61.2	42.8	23.1	42.6	49.4
Faster R-CNN+DRFPN [45]	R101	41.1	62.8	44.7	23.9	44.3	52.2
Faster R-CNN+AugFPN [24]	R50	38.8	61.5	42.0	23.3	42.1	47.7
Faster R-CNN+AugFPN [24]	R101	40.6	63.2	44.0	24.0	44.1	51.0
Faster R-CNN+DyFPN [25]	R50	39.0	60.7	42.2	22.4	41.8	49.0
Faster R-CNN+DyFPN [25]	R101	40.8	62.4	44.6	23.4	44.2	51.7
Faster R-CNN+Inception FPN [25]	R50	39.2	60.9	42.5	22.8	42.0	48.9
Faster R-CNN+ImFPN [46]	R50	39.5	60.5	42.9	22.4	42.4	49.8
Faster R-CNN+ARDFPN** [24]	R50	37.2	59.2	40.3	20.5	40.5	49.6
Faster R-CNN+ARDFPN** [24]	R101	38.5	60.7	41.5	20.9	42.0	52.2
M2Det [20]	R101	39.7	60.0	43.3	25.3	42.5	48.3
Faster R-CNN+Inception FPN [25]	R101	40.9	62.4	44.6	23.6	44.1	51.7
NAS-FPN 7@256 [19]	R50	39.0	59.5	42.4	22.4	42.6	47.8
NAS-FPN 7@256 [19]	R101	40.3	61.2	43.8	23.1	43.9	50.1
Libra R-CNN [47]	R50	38.7	59.9	42.0	22.5	41.1	48.8
Libra R-CNN [47]	R101	40.3	61.3	43.9	22.9	43.1	51.0
Faster R-CNN+PAFPN*([3]	R50	38.0	59.0	41.3	22.1	41.2	46.9
Faster R-CNN+PAFPN*([3]	R101	39.2	59.7	42.6	22.4	42.3	49.0
Faster R-CNN+BPFPN	R50	39.8	61.0	43.0	23.0	42.4	49.8
Faster R-CNN+BPFPN	R101	41.7	61.7	45.3	23.7	44.7	52.9
Faster R-CNN+BPFPN-Lite	R50	39.1	60.1	42.3	22.4	41.8	48.9
Faster R-CNN+BPFPN-Lite	R101	40.7	61.9	44.2	23.2	43.8	51.4

The symbol '*' means our re-implementation results.
The symbol '**' means evaluated on the COCO val2017.

TABLE 2. Comparison with other stat-of-the art object detectors on MS-COCO val2017.

Method	*Backbone*	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Anchor-based multi-stage:							
Faster R-CNN+FPN [8]	R50	37.4	58.1	40.4	21.2	41.0	48.1
Faster R-CNN+FPN [8]	R101	39.4	60.1	43.1	22.4	43.7	51.1
Faster R-CNN+FPN [8]	X101	42.1	63	46.3	24.8	46.2	55.3
Cascade Mask R-CNN +FPN [48]	R50	41.2	59.4	45.0	23.9	44.2	54.4
Anchor-based one-stage:							
RetinaNet+FPN [11]	R50	36.5	55.4	39.1	20.4	40.3	48.1
Anchor-free key-point-based:							
Reppoints+FPN [10]	R50	38.1	58.7	40.8	22	41.9	50.1
VarifocalNet+FPN [50]	R50	41.6	59.5	45.0	24.4	45.1	54.2
Anchor-free center-based:							
FCOS+FPN [11]	R50	42.3	61.1	45.4	24.4	45.9	55.8
Faster R-CNN+BPFPN	R50	40.0	60.6	43.3	22.4	43.7	52.7
Faster R-CNN+BPFPN	R101	41.2	61.8	44.9	23.4	45.1	54.8
Faster R-CNN+BPFPN	X101	43.0	63.6	47.0	25.0	47.0	56.2
Faster R-CNN+BPFPN-Lite	R50	40.0	60.9	43.3	23.3	44.3	52.1
Faster R-CNN+BPFPN-Lite	R101	40.5	61.3	44.4	23.3	44.4	53.5
Cascade Mask R-CNN+BPFPN	R50	43.3	61.4	47.2	24.8	46.5	57.9
RetinaNet+BPFPN	R50	37.2	56.7	39.6	21.7	41.5	47.7
Reppoints+BPFPN	R50	39.2	59.9	42.1	23.7	43.3	49.8
VarifocalNet+BPFPN	R50	42.6	60.5	46.2	24.6	46.6	55.6
FCOS+BPFPN	R50	43.0	61.4	46.4	25.5	46.7	56.1

TABLE 3. Comparison with standard FPN on ShipRSImageNet.

Method	*Backbone*	*FPNs*	*mAP*	AP_S
Faster-R-CNN	Resnet-50	FPN	53.3(Baseline)	7.5(Baseline)
Faster-R-CNN	Resnet-101	FPN	54.3(Baseline)	10.5(Baseline)
Ritinanet	Resnet-50	FPN	46.0(Baseline)	5.9(Baseline)
Faster-R-CNN	Resnet-50	BPFPN	54.5(+1.2)	19.5(+12.0)
Faster-R-CNN	Resnet-101	BPFPN	57.1(+2.8)	22.4(+11.9)
Ritinanet	Resnet-50	BPFPN	48.6(+2.6)	23.2(+17.3)

or ASPP and BR can also improve the baseline by 2.3 and 2.2 AP, which proves that parallel feature aggregation can benefit from cross-level attention and channel refinement. When all three components are integrated into the PAFPN, the performance can be improved to 40.0% AP, which shows that these three modules complement each other.

B. ABLATION STUDIES FOR ASPP MODULE DESIGN

The total number of branches and dilation rates in the ASPP module affect its overall ability of multi-scale feature capture. Experiments are designed to study how different dilation rates and the number of branches influence the performance quantitatively. The results are illustrated in Table 5. Since too

TABLE 4. Performance of each module. Results are reported on MS-COCO val2017.

ClCSAR w/Average pooling	*ClCSAR w/Max pooling*	*ClCSAR*	*ASPP*	*BR*	*AP*	*AP50*	*AP75*
PAFPN Baseine					37.5	58.6	40.8
✓					37.7	58.5	40.9
	✓				38.4	59.3	41.8
		✓			37.8	59.5	40.8
			✓		39.5	60.2	42.9
✓			✓		39.6	60.2	43.2
	✓		✓		39.7	60.3	42.9
		✓	✓		39.8	60.3	43.2
			✓	✓	39.7	60.3	43.2
		✓	✓	✓	40.0	60.6	43.3

many branches will lead to too much memory overhead, we set the maximum number of parallel branches of the ASPP module to 5 (not including global pooling branches). The AP show continuous increase when dilation rates increase from 1 to 12. We can observe that larger dilation rates will obtain reasonably high performances.

we report results with several settings: (1) Each ASPP module has four branches, and a dilation rate is an odd number that is relatively prime. (2) Each ASPP module has four branches, and a dilation rate is an even number that is proportionally expanded. (3) The number of branches of each ASPP module is gradually reduced from 4 to 2, and a dilation rate is an even number that is proportionally expanded. (4) The number of branches of each ASPP module is gradually reduced from 5 to 3, and a dilation rate is an even number that is proportionally expanded. As shown in Table 5, settings (2) and (4) use larger dilation rates and obtain reasonably high performances.

C. COMPLEXITY AND EFFICIENCY ANALYSIS

To analyze the model complexity and efficiency, we compare the number of model parameters, floating-point operations per second (FLOPS), and inference time with FPN and PAFPN. We report all the results in Table 6. The results show that BPFPN improves the performance in a large margin but introducing extra parameters. BPFPN-Lite reduces 40.0% computational cost in FLOPs with only 0.7 degradations in AP compared to BPFPN. We can conclude that the E-WPFA Block with channel split mechanism can essentially reduce the amount of computation while preserving high accuracy.

Faster RCNN +FPN [8] serves as our baseline, with 39.82 million learnable parameters and 97.63 GFLOPs. The results show that BPFPN improves the performance in a large margin while introducing extra parameters. The results show that BPFPN improves the performance in a large margin while introducing extra parameters. Our BPFPN-Lite reduces 40.0% computational cost in FLOPs with only 0.7 degradations in AP compared to BPFPN. We can conclude that the E-WPFA Block using a channel split mechanism can essentially reduce the amount of computation while preserving high accuracy.

VI. CONCLUSION

In this paper, we proposed a novel BPFPN to capture spatial information with multiple scales in parallel on each level of the feature pyramid efficiently and effectively. BPFPN comprises two carefully designed blocks (ClCSAR and WPFA) to solve the parallel extraction of multi-scale features from a single level and enhance the feature representations by cross-level channel attention-refinement. We also introduce an E-WPFA Block to improve the accuracy, and then the lightweight version of BPFPN, named BPFPN-Lite, is further proposed to reduce the computational cost while preserving accuracy. Moreover, our proposed BPFPN and BPFPN-Lite does not change the size of the feature pyramid and can be easily plugged into modern object detection networks. Extensive experiments on the COCO dataset demonstrate that BPFPN outperforms existing state-of-the-art feature pyramid methods. Solid improvements over SOTA methods demonstrated the effectiveness of BPFPN. BPFPN and BPFPN-Lite can produce significant improvements upon existing modern object detectors. BPFPN can be used to combine with different backbone networks and replace traditional FPN. We believe that the detector performance can be further improved when combined with a powerful backbone such as the Swin Transformer, which will be addressed in our future work.

REFERENCES

[1] Z. Zhang, L. Zhang, Y. Wang, P. Feng, and R. He, "Shiprsimagenet: A large-scale fine-grained dataset for ship detection in high-resolution optical remote sensing images," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 8458–8472, 2021.

[2] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125, 2017.

[3] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8759–8768, 2018.

[4] X. Zhao, W. Li, Y. Zhang, S. Chang, Z. Feng, and P. Zhang, "Aggregated residual dilation-based feature pyramid network for object detection," IEEE Access, vol. 7, pp. 134014–134027, 2019.

[5] J. Wang, K. Chen, R. Xu, Z. Liu, C. C. Loy, and D. Lin, "Carafe: Content-aware reassembly of features," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3007–3016, 2019.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in European conference on computer vision, pp. 21–37, Springer, 2016.

[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in Proceedings of the IEEE international conference on computer vision, 2017.

TABLE 5. Ablation studies for ASPP design.

Backbone	*Schedule*	*Dilation rates*			*AP*
		P2-P3	*P3-P4*	*P4-P5*	
R50	1x	(1,3,7,1)	(1,3,6,1)	(1,3,5,1)	39.0
R50	1x	(1,3,6,12,1)	(1,3,6,1)	(1,3,1)	39.3
R50	1x	(1,3,5,7,1)	(1,3,5,7,1)	(1,3,5,7,1)	38.9
R50	1x	(1,2,4,8,1)	(1,2,4,8,1)	(1,2,4,8,1)	39.5
R50	1x	(1,2,4,8,1)	(1,2,4,1)	(1,2,1)	39.2
R50	1x	(1,2,4,8,12,1)	(1,2,4,8,1)	(1,2,4,1)	39.5

TABLE 6. Ablation studies of different resource budgets and inference time. The framework of all listed methods is Faster-R-CNN. Using NVIDIA RTX 3090 GPU.

Methods	*Backbone*	*mAP*	*Params (M)*	*GFLOPs*	*Inf time (fps)*
FPN	R50	37.7	39.82	97.63	26.0
FPN	R101	39.7	57.94	143.73	20.4
PAFPN	R50	38.0	45.07	231.85	23.3
PAFPN	R101	39.2	64.06	307.92	20.4
BPFPN	R50	39.8	51.81	413.73	14.8
BPFPN	R101	41.7	70.77	489.8	12.6
BPFPN-Lite	R50	39.1	45.49	242.17	18.5
BPFPN-Lite	R101	40.7	64.48	318.24	16.4

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, Jun 2017.

[9] Z. Cai and N. 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.

[10] Z. Yang, S. Liu, H. Hu, L. Wang, and S. Lin, "Reppoints: Point set representation for object detection," in The IEEE International Conference on Computer Vision (ICCV), Oct 2019.

[11] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," arXiv preprint arXiv:1904.01355, 2019.

[12] Y. Li, Y. Chen, N. Wang, and Z. Zhang, "Scale-aware trident networks for object detection," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6054–6063, 2019.

[13] F. Sun, T. Kong, W. Huang, C. Tan, B. Fang, and H. Liu, "Feature pyramid reconfiguration with consistent loss for object detection," IEEE Transactions on Image Processing, vol. 28, no. 10, pp. 5041–5051, 2019.

[14] T. Kong, F. Sun, H. Liu, Y. Jiang, and J. Shi, "Consistent optimization for single-shot object detection," arXiv preprint arXiv:1901.06563, 2019.

[15] X. Wang, Z. Cai, D. Gao, and N. Vasconcelos, "Towards universal object detection by domain attention," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7289–7298, 2019.

[16] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, "Acquisition of localization confidence for accurate object detection," in Proceedings of the European conference on computer vision (ECCV), pp. 784–799, 2018.

[17] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," arXiv preprint arXiv:1312.6229, 2013.

[18] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, "Ron: Reverse connection with objectness prior networks for object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5936–5944, 2017.

[19] G. Ghiasi, T. Lin, and Q. V. Le, "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.

[20] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling, "M2det: A single-shot object detector based on multi-level feature pyramid network," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 9259–9266, 2019.

[21] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10781–10790, 2020.

[22] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2403–2412, 2018.

[23] S. Qiao, L.-C. Chen, and A. Yuille, "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.

[24] C. Guo, B. Fan, Q. Zhang, S. Xiang, and C. Pan, "Augfpn: Improving multi-scale feature learning for object detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12595–12604, 2020.

[25] M. Zhu, K. Han, C. Yu, and Y. Wang, "Dynamic feature pyramid networks for object detection," arXiv preprint arXiv:2012.00779, 2020.

[26] P.-Y. Chen, J.-W. Hsieh, C.-Y. Wang, H.-Y. M. Liao, and M. Gochoo, "Residual bi-fusion feature pyramid network for accurate single-shot object detection," arXiv preprint arXiv:1911.12051, 2019.

[27] H. Xu, L. Yao, W. Zhang, X. Liang, and Z. Li, "Auto-fpn: Automatic network architecture adaptation for object detection beyond classification," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6649–6658, 2019.

[28] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian, "A real-time algorithm for signal analysis with the help of the wavelet transform," in Wavelets, pp. 286–297, Springer, 1990.

[29] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," arXiv preprint arXiv:1412.7062, 2014.

[30] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in Proceedings of the European conference on computer vision (ECCV), pp. 801–818, 2018.

[31] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," IEEE Transactions on pattern analysis and machine intelligence, vol. 40, no. 4, pp. 834–848, 2017.

[32] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," arXiv preprint arXiv:1706.05587, 2017.

[33] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in Proceedings of the european conference on computer vision (ECCV), pp. 552–568, 2018.

[34] L. Mou, L. Chen, J. Cheng, Z. Gu, Y. Zhao, and J. Liu, "Dense dilated network with probability regularized walk for vessel detection," IEEE transactions on medical imaging, vol. 39, no. 5, pp. 1392–1403, 2019.

[35] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," arXiv preprint arXiv:1511.07122, 2015.

[36] R. Schuster, O. Wasenmuller, C. Unger, and D. Stricker, "Sdc-stacked dilated convolution: A unified descriptor network for dense matching tasks," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2556–2565, 2019.

[37] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "Denseaspp for semantic segmentation in street scenes," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3684–3692, 2018.

[38] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "Detnet: A backbone network for object detection," arXiv preprint arXiv:1804.06215, 2018.

[39] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141, 2018.

[40] H. Li, P. Xiong, J. An, and L. Wang, "Pyramid attention network for semantic segmentation," arXiv preprint arXiv:1805.10180, 2018.

[41] H. Zhang, K. Zu, J. Lu, Y. Zou, and D. Meng, "Epsanet: An efficient pyramid split attention block on convolutional neural network," arXiv preprint arXiv:2105.14447, 2021.

[42] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in European conference on computer vision, pp. 740–755, Springer, 2014.

[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in Advances in Neural Information Processing Systems (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), vol. 32, Curran Associates, Inc., 2019.

[44] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open mmlab detection toolbox and benchmark," arXiv preprint arXiv:1906.07155, 2019.

[45] J. Ma and B. Chen, "Dual refinement feature pyramid networks for object detection," arXiv preprint arXiv:2012.01733, 2020.

[46] L. Zhu, F. Lee, J. Cai, H. Yu, and Q. Chen, "An improved feature pyramid network for object detection," Neurocomputing, vol. 483, pp. 127–139, 2022.

[47] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra r-cnn: Towards balanced learning for object detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 821–830, 2019.

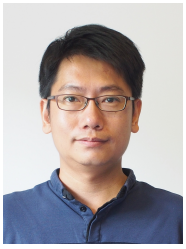
[48] Z. Cai and N. Vasconcelos, "Cascade r-cnn: High quality object detection and instance segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 1–1, 2019.

[49] H. Zhang, Y. Wang, F. Dayoub, and N. Sünderhauf, "Varifocalnet: An iou-aware dense object detector," arXiv preprint arXiv:2008.13367, 2020.

[50] Y. Qin, J. Wen, H. Zheng, X. Huang, J. Yang, N. Song, Y.-M. Zhu, L. Wu, and G.-Z. Yang, "Varifocal-net: A chromosome classification approach using deep convolutional networks," IEEE transactions on medical imaging, vol. 38, no. 11, pp. 2569–2581, 2019.

ZHENGNING ZHANG received the M.Eng. degree in communication engineering from the China Academy of Space Technology (CAST), Beijing, China, in 2009. He is currently working toward the Ph.D. degree in information and communication engineering from the Department of electronic engineering, Tsinghua University, Beijing, China. His research interests include remote sensing image processing, object detection, and machine learning.

LIN ZHANG received the B.Sc., M.Sc., and Ph.D. degrees in Electronic Engineering from Tsinghua University, Beijing, in 1998, 2001, and 2006, respectively. He was a Visiting Professor with the University of California at Berkeley, from 2011 to 2013. He has been teaching the courses in selected topics in communication networks and information theory to senior undergraduate and graduate students at Tsinghua University. He is currently a Professor with the Tsinghua Shenzhen International Graduate School, Tsinghua University. His research interests include efficient protocols for sensor networks, statistical learning and data mining algorithms for sensory data processing, and information theory. Since 2006, he has been implementing wireless sensor networks in a wide range of application scenarios, including underground mine security, precision agriculture, industrial monitoring, and also the 2008 Beijing Olympic Stadium (the Bird's Nest) structural security surveillance Project and a metropolitan area sensing and operating network (MASON) in Shenzhen. Dr. Zhang received the IEEE/ACM SenSys 2010 Best Demo Awards, the IEEE/ACM IPSN 2014 Best Demo Awards, and the IEEE CASE 2013 Best Paper Awards, and the Excellent Teacher Awards from Tsinghua University, in 2004 and 2010.



YUE WANG received the B.Sc. and Ph.D. degrees in Electronic Engineering from the Electronic Engineering Department, Tsinghua University, in 1999 and 2005, respectively. He is currently an Associate Professor with Tsinghua University. His research interests include computer networks, data fusion, and complex networks.

PENGMING FENG (S'13–M'17) was born in Jilin, China. He received the B.Sc. degree in automatic control from the Beijing University of Chemical Technology, Beijing, China, in 2012, the M.Sc. degree in digital communication systems from Loughborough University, Loughborough, U.K., in 2013, and the Ph.D. degree in intelligent signal processing from Newcastle University, Newcastle Upon Tyne, U.K., in 2016. During his Ph.D. degree, he was with the University Defence Research Collaboration sponsored by the U.K. Defence Science and Technology Laboratory and the Engineering and Physical Science Research Council, on the project Signal Processing in Networked Battlespace. He joined China Aerospace Science and Technology Corporation in 2017, as a Doctoral Engineer with the State Key Laboratory of Space-Ground Integrated Information Technology, where he is currently a Senior Doctoral Engineer. His research interests include remote sensing, multiple target tracking, target detection, machine learning, and sparse representation.

BAOCHEN SUN received the M.Eng. degree in Aerospacecraft designing from the Harbin University of Science and Technology (HUST), Harbin, China, in 2005. His research interests include smart city and system engineering.