Object Detection with Extended Attention and Spatial Information

1st Yingda Guan
Institute of Microelectronics
Tsinghua University
Beijing, China
gyd17@mails.tsinghua.edu.cn

2nd Zuochang Ye
Institute of Microelectronics
Tsinghua University
Beijing, China
zuochang@mail.tsinghua.edu.cn

3nd Yan Wang
Institute of Microelectronics
Tsinghua University
Beijing, China
wangyan@mail.tsinghua.edu.cn

***Abstract*—Scale variation is one of the most challenging problems in general object detection. Although current approaches have achieved significant progress by exploiting the multi-level information, they pay little attention to how to fuse feature maps and construct the feature pyramid more effectively. In this paper, we propose two novel modules to enhance the characteristics of object detection. First, a Pair-wise Attention Module (PAM) is proposed to introduce the two-way attention mechanism, which can emphasize informative features and filter less useful ones adaptively when fusing feature. Second, a Pyramid Reconfigure Module (PRM) is proposed to promote cross-level spatial information communication by the split-align-reconstruct operation. Then the feature among different levels can be complemented and enhanced with each other. The effectiveness of our proposed modules is evaluated on the COCO benchmark, and experimental results show that our approach achieves state-of-the-art results.**

I. Introduction

General object detection is one of the most fundamental and extensively-applied tasks in computer vision [1]. Compared with face detection and pedestrian detection, it is more challenging because it needs to detect at a wider range of categories and geometries. Despite the significant improvements bought by Deep Convolutional Neural Network (DCNN), it still struggles in many problems. Scale variation across object instances is one of the major challenges [2] [3].

In the early period, the majority of object detection methods based on DCNN [2] [4] [5] [6] [7], use the top-most layer of the network to detect objects at different scales, as shown Fig. 1(a). However, the information in this layer may be too coarse spatially to allow precise localization, especially for small objects [3]. On the contrary, earlier layers have more fine-grained details but are also much less sensitive to semantics. Intuitively, exploiting multi-level features can get the best of both worlds. Many methods have been proposed based on this principle. Although the methods vary widely, they can be roughly divided into three types. Some approaches detect with combined features of multiple DCNN layers [8] [9] [10]. They combine finer features from lower layers and coarse semantic features from higher layers by concatenation. But simply incorporating features from different levels does not yield significant improvements due to overfitting caused by high dimensionality. Different from the above, there are many methods of detecting on multiple levels [11] [12] [13] [14],

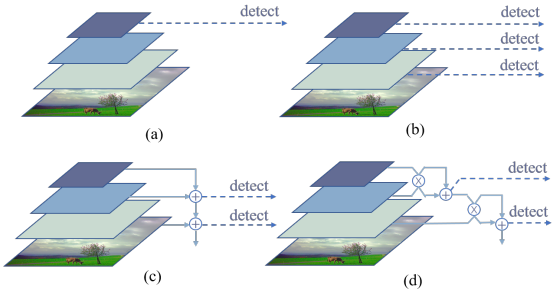

Fig. 1. Comparison of different methods. (a) Detect with the top-most feature map, which performs poorly in small object detection, due to low resolution. (b) Recent detection systems construct feature pyramid with hierarchical feature maps, but the low-level feature lacks semantic information. (c) One solution is introducing a reverse fusion module to enhance the low-level feature maps. (d) Our method adopts the attention mechanism and reconfigures the feature pyramid when fusing features at different levels.

shown Fig. 1(b). These methods allocate objects of different sizes to different layers, high-level feature maps with stronger semantic information are used to detect large objects, while low-level features with richer spatial details to detect small objects. However, without sufficient semantics and contextual information, the earlier feature maps cannot capture small objects as we expected. By analyzing the characteristics of the above two approaches, the latest methods [3] [15] [16] [17] [18] supplement an existing bottom-up pathway with a Reverse Fusion Block(RFB), as illustrated in Fig. 1(c), which is specifically composed of a top-down branch and a lateral connection. Bottom-up features at intermediate depths with finer details, after lateral processing, are combined with the top-down features carrying semantics, and this combination is further transmitted down to lower layers by reverse fusion module.

Building a feature pyramid with a reverse fusion module is the best way to solve multi-scale problems in the mentioned methods. We go beyond this standard structure with new issues. Generally speaking, the above-mentioned methods have two following limitations. Firstly, in the RFB, most methods adopt simple addition or concatenation when combining two-level feature maps. While in common backbone(e.g. ResNet [19]), the adjacent levels are separated by a series of convolu-

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

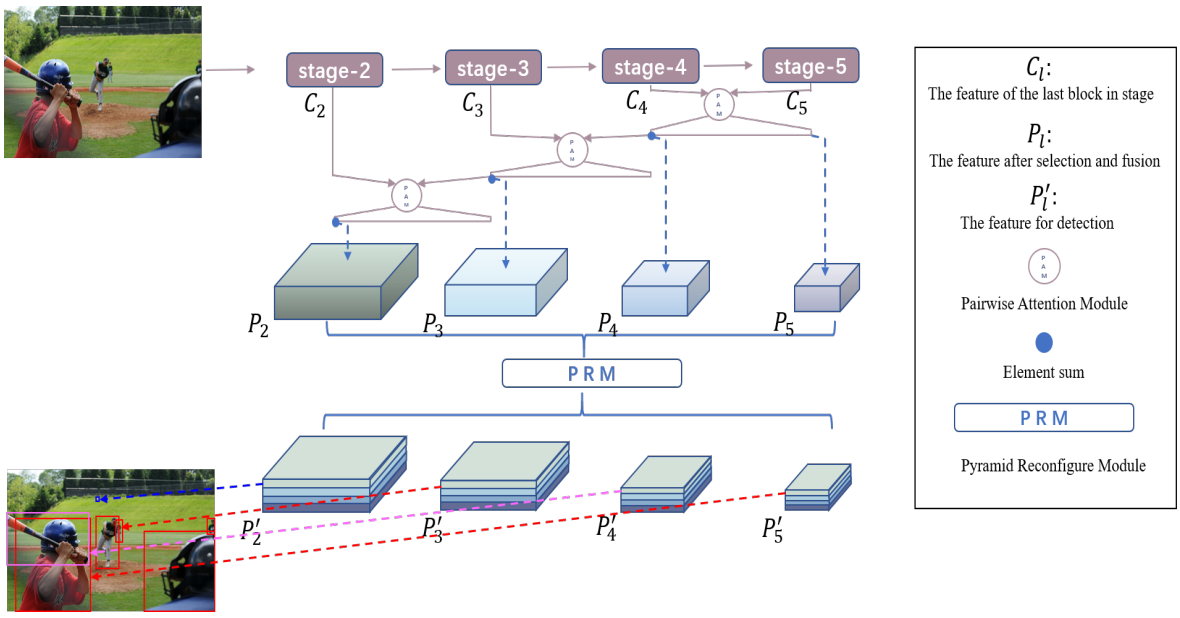

Fig. 2. An overview of the proposed method. Our method utilizes the backbone to extract features from the input image and takes the output of different stages from the backbone to construct the feature pyramid $\boldsymbol{C}$=$\{C_2,C_3,C_4C_5\}$. During the reverse fusion process, different level feature maps $\boldsymbol{P}$=$\{P_2,P_3,P_4P_5\}$ can be fused by an effective Pair-wise Attention Module(**PAM**) without destroying the features' representative abilities. Feature maps with rich semantics are generated recursively by this way. Then the Pyramid Reconfigure Module(**PRM**) module adopts the operation of split-rescale-aggregate to combine high-level semantics and low-level details. The characteristics of different levels complement each other, with multi-scale information for detection. Finally, $\mathbf{P'}$=$\{P'_2,P'_3,P'_4P'_5\}$ produces dense bounding boxes and category scores.

tion layers, pooling and activation functions. The distribution of features at different levels is quite diverse, and the naive fusion method will damage the original feature. We believe that the information which is more important to the detection should be adaptively highlighted, so the attention mechanism is very necessary during fusing. Secondly, if we compare neural networks to human eyes when we detect an object, we must not only rely on the scale and details of the object itself but also the surrounding environmental information. For example, let's suppose that there is a cup of small size on a given image, and the low-level features may accurately locate it, but it is not easy to determine if it's a cup or other cylindrical things due to the lack of object-level information. However, there is very little communication among different levels in the previous method. The feature at different levels is assigned to detect the different subset of objects according to the scale, which contradicts the judgment logic of human eyes.

The goal of this paper is to construct a more effective feature pyramid for detecting multi-scale objects. The pair-wise attention mechanism and pyramid reconfiguration are introduced in our method as shown in Fig. 1(d). And the overview of the method is illustrated in Fig. 2. Similar as the traditional methods, we take ResNet [19] as backbone pretrained on ImageNet [20] to extract feature at different levels. Given the multi-level feature, we design the Pair-wise Attention Module(PAM) to help feature from the adjacent level to fuse. In this module, we emphasize the importance of mutual supervision and filtering. To the best of our knowledge, though the channel-wise self-supervising method is popularly used, the pair-wise attention mechanism is rarely mentioned in the detection task. And we conduct extensive experiments to prove that this method is more efficient in heightening features, compared with the self-attention during the process of feature fusion.

Moreover, a Pyramid Reconfigure Module(PRM) is proposed to promote information transfer across the channels among the feature maps. We accept that the feature at different levels should be mutually promoted and supplemented. Even if we use the high-level feature maps to detect large objects, we need the underlying details in the low-level feature. Similarly, we rely more on high-level semantics when we apply low-level features to detect small objects. In order to combine semantics and fine-grained appearance in the PRM, we adopt the split-rescale-reconstruct method to enrich every single level with information from other levels. At this point, we have obtained feature maps with characteristics of multi-level and multi-scale, then the detector can detect more comprehensively and accurately. It is worth noting no additional parameters are introduced in this process.

Since FPN [3] and mask-RCNN [21] are powerful structures for detection, we implement them as the baseline in our experiments to investigate the impact of our method. We do the ablation studies on the COCO benchmark [22] to demonstrate the effectiveness of the Pair-wise Attention Module and Pyramid Reconfigure Module from various aspects. Experimental results show that our approach achieves state-of-the-art results. The main contributions of this paper are

summarized as follows:

- First, we develop a Pair-wise Attention Module(PAM) to introduce mutual attention guiding the feature from adjacent levels to fuse adaptively.
- Second, we propose a Pyramid Reconfigure Module(PRM) to enhance the cross-channel information communication between different levels.
- Our method achieves state-of-the-art results on the COCO dataset benchmark.

II. Related Work

Detection is one of the most important and fundamental tasks in the field of computer vision. Researchers have proposed many methods to improve detection accuracy. And different network structures have been proposed to exploit the potential of multi-level feature maps' representation power.

A. Object Detection

Breakthrough progress has been made in object detection since the popularity of DCNN. Although there are many variations based on DCNN, all methods can be roughly divided into two groups:***(i) proposal-based*** and ***(ii)proposal-free methods***. For the former, Fast-rcnn [4], spp-net [2]made proposals through Selective Search, exposing computation as a bottleneck. Faster-RCNN [5] proposed Region Proposal Network(RPN), which utilizes DCNN to compute proposals as a precedent. The method reduced the computational complexity and boosts the detection accuracy greatly. However, methods mentioned above exploited the top-most layer to make a prediction which is not sensitive to small objects. FPN [3] extends the existing bottom-up path in DCNN with a reverse fusion module. It ensures all layers are semantically strong, including high-resolution level. R-FCN [6]shared the computation after ROI-pooling to run faster and utilized position sensitive ROI-pooling to enrich the feature with more localization representation. Mask-RCNN [21] adds an extra object mask branch based on box detection path in parallel. Meanwhile, ROI-Align layer is proposed to avoid quantification of the proposals' boundaries. It can restrain misalignment compared with the ROI-pooling in Faster-RCNN [5]. Cascade-RCNN [23] consists of a series of detectors, trained with increasing IOU thresholds. Then the detector cannot be easily misled by false positives and leads to more accurate localization. The proposal-based approach is not efficient because proposals need to be computed first by DCNN then regressed. Representative methods in proposal-free like YOLO [7],SSD [11] merged two stages into one, which can be applied in real-time tasks. YOLO [7] predicted bounding boxes and class probabilities directly from full images in one evaluation with a single neural network. SSD [11] detected at multilayer, high-level is responsible for large objects, lower for small. Subsequently, lots of work [15] [16] [17] based on SSD [11] have been proposed. They adopt the reverse fusion module to strengthen the low-level features. Also, they utilized other techniques to achieve comparable localization accuracy to the proposal-based methods [16], solve the unbalance of positive and negative samples [17] [24].

B. Detection with Multi-Level Feature

Many research results have shown that making better use of multi-level features is of vital importance to accurate visual recognition. To the best of our knowledge, the methods of multi-layer feature fusion are mainly divided into the following three types.

Hypercolumns [8], HyperNet [9], ION [10] detect with combined features of multiple DCNN layers by the skip-layer connection. Hypercolumns [8] exploited the hypercolumn descriptor for every pixel, which means concatenating activations of all DCNN units above that pixel as a vector. Some layers were skipped since adjacent layers are strongly correlated. ION [10] used skip pooling to extract information at multiple levels, adopted L2 normalization prior to combining them. HyperNet [9] aggregates hierarchical feature maps and compresses them into a uniform space first, then generate region proposals. HyperNet is more computation efficiency compared with ION because all features can be precomputed before region proposal generation and the detection module. To combine multi-level feature maps at the same resolution, upsampling and downsampling are implemented by deconvolution and max-pooling respectively.

SSD [11], MSCNN [12] detected at multi-layers directly. In these methods, detection is performed at multiple output layers, so that receptive fields can match objects of different scales. The deeper feature with the large receptive field is responsible for big objects, while the shallower with more details for small objects. However, the semantics of the low-level feature is too weak to detect small objects. RFB-Net [13] proposed Receptive Field Block, which adopted multi-branch pooling with varying kernels corresponding to receptive field of different sizes. And further assembled Receptive Field Block to the top of SSD, enhancing discriminability and robustness of features at all levels. DSOD [14] proposed a dense structure for prediction which can learn half and reuse half. In each scale, half of the feature maps are learned from the previous scale with convolution, while the remaining half feature maps are directly down-sampled, then concatenate them to fuse feature.

For the first method mentioned above, multi-layer features are directly aggregated into one for detection, which leads to overfitting problem due to high dimensions and affects the feature representation of the model. For the second, the high-resolution features struggle in insufficient semantics when making the final prediction directly. In view of the disadvantages of the two methods, assembling reverse fusion block to the existing hierarchical multi-layer features is adopted by many latest methods [3] [15] [16] [17]. The Reverse Fusion Block(RFB) consists of a top-down branch and a horizontal connection to the existing bottom-up branch. Bottom-up features at the intermediate depth, after lateral processing, are fused with the top-down feature carrying semantic information, and this fusion is then continuously transmitted

down through reverse fusion path. Although there are many variations in this approach, they may differ in the design of the RFB. In the top-down branch, since the resolutions of the two adjacent levels are different, FPN [3] carried out up-sampling on low-resolution features by interpolation, and DSSD [15], RON [17], RefineDet [16] used deconvolution. In the lateral processing, FPN and RetinaNet [24] used 1x1 convolution, RON and RefineDet used 3x3 convolution, and DSSD uses additional BatchNormalization(BN) [25] and ReLU [26] besides 3x3 convolution. When fusing, FPN, RON, RefineDet adopted element-wise addition, and DSSD used the element-wise product. RFB strengthens the low-level features through iterative fusion and transmission significantly. It is recognized that constructing a feature pyramid with a reverse fusion is one of the most powerful measures against multi-scale problems.

III. Proposed Method

The overall architecture of our method is shown in Fig. 2. Similar to FPN [3], we adopt the effective ResNet [19] as the backbone and use a multi-level feature pyramid network to explore the effect of Pair-wise Attention Module and Pyramid Reconfigure Module. We first briefly review the structure of FPN, then present the details of our method.

A. Feature Pyramid Network

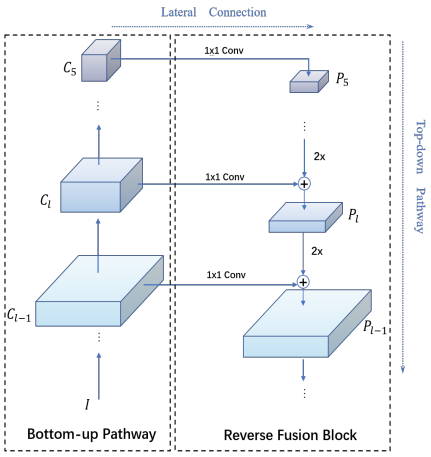

Fig. 3. Illustration of FPN. Based on the bottom-up pathway, FPN adopts a novel reverse fusion module, semantics is transferred recursively in the top-down path until the lowest level. The lateral connection fuses feature maps from different levels together through the addition.

As shown in Fig. 3, FPN network is composed of the bottom-up pathway and reverse fusion block, which consists of a top-down pathway and lateral connection. In the bottom-up pathway, supposed that the input image is *I*, FPN uses the output of the last residual block of each stage in ResNet to create the feature pyramid. In the reverse fusion block, high-level feature maps can get the same resolution as the current by bilinear interpolation during the top-down path. Feature maps at adjacent levels are unified into 256 channels by 1x1 convolution in lateral connection. Then the feature for detection P=$\{P_2,P_3,P_4P_5\}$ is obtained by add operations. This process can be expressed by (1).

$$\begin{aligned} P_{l-1} &= \mathbf{F_u}(P_l) + \alpha_{l-1}C_{l-1} \\ l \in [2,5], P_5 &= \alpha_5 C_5 \end{aligned} \tag{1}$$

where α means weights in 1x1 convolution, F_u means upsampling by bilinear interpolation, l means the l^{th} level in P and C. We can find that P_l is the linear combination of the feature at the current level and other higher-level features in (2).

$$P_l = \sum_{i=l}^{5} w_i C_i \tag{2}$$

where w_l is the generated final weights for l^{th} layer output after similar polynomial expansions. The linear combination with a deeper feature hierarchy can enrich the shallower layer recursively. However, its representation power is not enough for the complex task of object detection which often lives on a non-linear function of input [27].

B. Pair-wise Attention Module

Motivation Given the hierarchy feature at different levels, how to enrich low-level features with more semantics from higher levels is essential. FPN adopts the add operation simply. Though there are many nonlinear convolutions between the adjacent level. The representation of features has relative independence. Combining characteristics from different levels should be skillful. So we proposed the two-way choice mechanism to emphasize informative features and filter less useful ones.

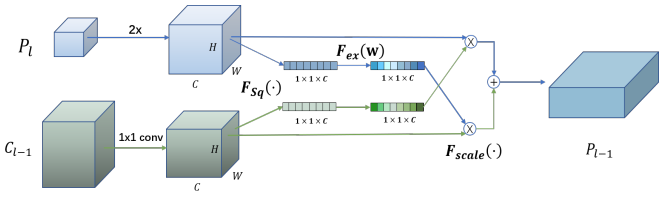

Fig. 4. Illustration of Pairwise Attention Module. We introduce the Squeeze-and-Excitation method to extract descriptors of different-level features, then cross-multiply them with the original feature maps to obtain pair-wised attention.

Pair-wise Attention Module(**PAM**) brings attention when fusing features from different levels as shown in Fig. 4, low-level features can learn how to chose and high-level features can learn how to share. The design of this module is based on the Squeeze-and-Excitation block [28] and we increase skillfully designed mutual supervision mechanisms. Given the feature pyramid, $\boldsymbol{C}$=$\{C_2,C_3,C_4,C_5\}$, we take C_l and C_{l-1} as an example to present the pairwise mechanism because the resolution of C_l is twice smaller than C_{l-1}, we firstly upsample P_l by bilinear method to make sure that feature maps from two levels have the same size. Then we use global average pooling to squeeze global information to channel-wise

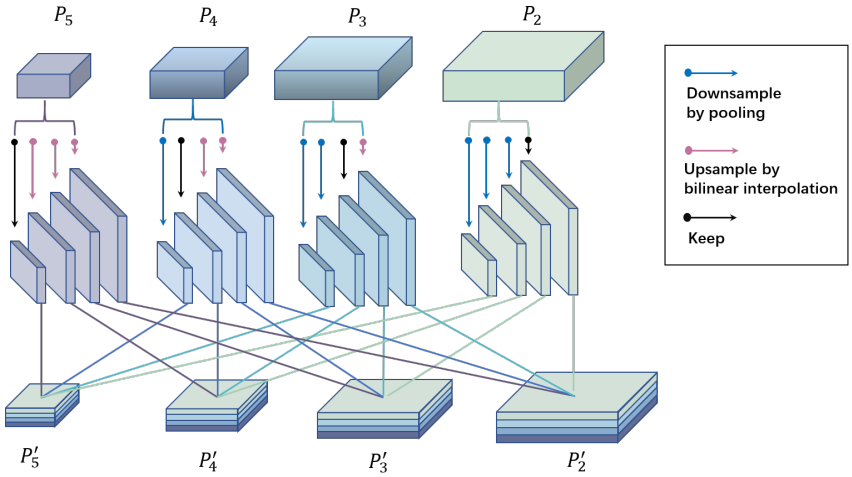

Fig. 5. Illustration of Pyramid Reconfigure Module. In the process of reverse fusion, high-level semantic information is always transmitted downward, but it is challenging to get the low-level details. PRM selects parts of channels from different levels as representatives, combines them to exchange information. Then the different characteristics among different levels can be complemented and enhanced with each other.

descriptor z_l and z_{l-1} for the l^{th} level and the *l-1*thlevel at the squeeze step.

$$z_{l-1} = \mathbf{F_{sq}}(C_{l-1}) = \frac{1}{H_{l-1} \times W_{l-1}} \sum_{i=1}^{H_{l-1}} \sum_{j=1}^{W_{l-1}} C_{l-1}(i,j)$$
$$z_l = \mathbf{F_{sq}}(C_l) = \frac{1}{H_l \times W_l} \sum_{i=1}^{H_l} \sum_{j=1}^{W_l} C_l(i,j) \tag{3}$$

Where H_l and W_l refer to the height and width of the feature map in the l^{th} level. Since the global average pooling is channel-wise, the dimension of z_l is the same as the number of channels of C_l.

To make use of the information aggregated in the squeeze operation, the following excitation step aims at fully capturing channel-wise dependencies respectively, as in (4).

$$s_{l-1} = \mathbf{F_{ex}}(z_{l-1}) = \sigma(W_{l-1}^{\ 2}(\delta(W_{l-1}^{\ 1} z_{l-1}))$$
$$s_l = \mathbf{F_{ex}}(z_l) = \sigma(W_l^2(\delta(W_l^1 z_l)) \tag{4}$$

Where δ refers to ReLU [26] function, σ refers to sigmoid Function, A parameterized gate mechanism is formed by a bottleneck block with two fully-connect layer, $W^1 \in \mathbb{R}^{C \times \frac{C}{r}}$ and $W^2 \in R^{\frac{C}{r} \times C}$, we choose 16 as the reduction ratio r in dimensionality-reduction layer. After squeeze and excitation operation, We get s_{l-1} and s_l as a collection of per-channel modulation weights for two levels. Then we reconfigure the feature maps P_{l-1} using them as in (5).

$$P_{l-1} = \mathbf{F_{scale}}(s_l, C_{l-1}) + \mathbf{F_{scale}}(s_{l-1}, C_l)$$
$$= s_l C_{l-1} + s_{l-1} C_l \tag{5}$$

This operation can be regarded as a pairwise-attention function on features of different levels. The relationships between them are not supposed to be confined to the local receptive field by the convolution filters. The mechanism allows information from the global receptive field of the network to be used by other levels. And the feature we get from this operation is more powerful to be employed in fusing cross levels.

C. Pyramid Reconfigure Module

Motivation After the feature selection and fusion in the PAM, the feature for detection has obtained strong expressive abilities. In the process of detecting network training, high-level features are used to detect large objects and shallow features for small objects. In this way, the dependence between layers is destroyed. We demonstrated that the information with different characteristics among different levels should be complemented and enhanced with each other. Combining and reconfigure of high-level semantics and low-level details will effectively improve the detection accuracy of multi-scale objects.

Given the feature set *P*, we propose a novel PRM module as shown in Fig. 6, which does not introduce extra parameters. First, we split the *l-1*th level feature map P_l into four slices. To ensure that the features are aligned during the following fusion operation, we need to adjust the slices to the scales of each level in *P*. If the slice is the same size as the target feature, we keep the original, if smaller, we upsample the slice feature by bilinear interpolation, if larger, we adopt max pooling operation. (e.g. If the target size is four times larger than the current slices, bilinear interpolation will be done twice). After rescaling, the slices can be denoted as $\boldsymbol{S}$=$\{S_l^2, S_l^3, S_l^4 S_l^5\}$, which has the same resolution with $\{P_2, P_3, P_4 P_5\}$.Then we combine the slice of each level with the same resolution into a new feature map as in (6).

$$P'_l = \mathbf{F_{concat}}(S_l^2, S_l^3, S_l^4, S_l^5) \tag{6}$$

Where F_{concat} means concatenation, and P'_l is the feature for detection. Based on the original feature map P_l, we adopt rescaling, reconfigure and align to combine semantics and details at different levels.

TABLE I
EXPERIMENTS RESULTS

Method	Backbone	Avg. Precision, IoU			Avg. Precision, Area			delta, IoU	delta, Area
		mAP	*AP50*	*AP75*	*APs*	*APm*	*APl*	*mAP*	*APs*
Faster-RCNN	VGG	*23.5*	*43.9*	*22.6*	*8.1*	*25.1*	*34.7*	-	-
R-FCN	ResNet-50	*27.1*	*49.0*	*26.9*	*10.4*	*29.7*	*39.2*	-	-
CoupleNet	ResNet-101	*34.4*	*54.8*	*37.2*	*13.4*	*38.1*	*50.8*	-	-
FPN	ResNet-50	*36.4*	*59.0*	*39.2*	*20.3*	*38.8*	*46.4*	-	-
Ours+FPN	ResNet-50	***37.6***	*59.9*	*40.7*	*21.9*	*41.2*	*48.7*	***+1.2***	***+1.6***
mask-RCNN	ResNet-50	*36.7*	*58.7*	*40.3*	*21.0*	*39.7*	*48.8*	-	-
Ours+mask-RCNN	ResNet-50	*37.8*	*60.2*	*40.7*	*22.0*	*40.7*	*50.2*	***+1.1***	***+1.0***
FPN	ResNet-101	*38.8*	*61.1*	*41.9*	*21.3*	*41.8*	*49.8*	-	-
Ours+FPN	ResNet-101	***39.6***	*62.2*	*42.9*	*22.7*	*43.4*	*52.3*	***+0.8***	***+1.4***
mask-RCNN	ResNet-101	*38.9*	*60.6*	*42.6*	*21.4*	*42.4*	*52.2*	-	-
Ours+mask-RCNN	ResNet-101	***39.9***	*62.1*	*43.6*	*22.9*	*44.0*	*52.3*	***+1.0***	***+1.5***
FPN	ResNeXt-101-32x4d	*40.1*	*62.6*	*43.9*	*23.1*	*44.4*	*53.2*	-	-
Ours+FPN	ResNeXt-101-32x4d	***40.9***	*63.2*	*44.6*	*23.8*	*45.1*	*54.4*	***+0.8***	***+0.7***
mask-RCNN	ResNeXt-101-32x4d	*41.1*	*63.4*	*45.2*	*23.9*	*45.6*	*54.6*	-	-
Ours+mask-RCNN	ResNeXt-101-32x4d	***42.0***	*64.2*	*46.7*	*24.9*	*46.8*	*55.7*	***+0.9***	***+1.0***

IV. EXPERIMENTS

In this section, we present experimental results on the MS-COCO benchmark. The dataset consists of 80 categories and objects which vary in size and can be divided into small, medium and large. We use the 115k images for training, and 5k images(minival) for validation, 20k images(test-dev) for testing. To compare with the other state-of-arts models, we report COCO mAP on test-dev which has no public labels and requires the use of the evaluation server. And we report the results of ablation studies on minival for convenience. Our experiments include three parts:(1) Implement Details, (2) Comparisons with state-of-the-art models, (3) Ablation studies.

A. Implement Details

Training details. All architectures in Table 1 are trained end-to-end. The input image is resized such that its shorter side has 800 pixels, and the height and width should be divided by 16. We trained the network on 8 GPUs and there are 2 images in a mini-batch per GPU. The iter size is set to 4. We used a weight decay of 0.0005 and a momentum of 0.9. The base learning rate is 0.02, and we adopt a warm-up mode, which is when training starts, We reduce the base learning rate by the multiplying warm-up factor(e.g. 1/3), and the learning rate will increase linearly to the base value after warm-up iters(e.g.500). We train 12 epochs. We choose 15 anchors which involve 5 sizes and 3 ratios for each image finally. After the RPN stage, if the score of the proposal is higher than 0.5, we denote it as a positive example, otherwise negative. Then we sampled 512 proposals by a positive-negative ratio of 1:3. These proposals are used to train the Fast stage. We use the ResNet pre-trained on Imagenet as the backbone, ResNet50, ResNet101 and ResNext-101 [29] for comparison experiments. In the ablation experiments, ResNet50 is used by default.

Testing Details During the test process, we rescale the image to the same size as that in training. After the RPN stage, we sort the proposals by score, and choose top 2000 proposals, perform the NMS with a threshold at 0.5, and take the top 1000 proposals as the input of the second stage according to score. After Fast stage, we select the boxes whose score is higher than 0.01 and perform the NMS with a threshold at 0.7 to get the final detection bounding boxes.

B. Comparisons with State-of-the-art Models

Table I compares our model with other state-of-the-art methods on COCO test-dev. FPN and mask-RCNN are recognized as the most effective methods among different detection algorithms. Therefore we adopt two algorithms, with different sizes of backbone as the baseline to establish the effectiveness of our method. We mainly focus on two metrics, *mAP* and *APs*, where *mAP* can reflect the comprehensive performance of the model, and *APs* can measure the accuracy of small objects that is hard to detect. For FPN method, our model achieves 37.6 *mAP* with the backbone of ResNet50, 39.6 *mAP* for ResNet101 and 40.9 *mAP* for ResNeXt-101, which outperforms FPN 36.4 *mAP*, 38.8 *mAP* and 40.1 *mAP*. For mask-RCNN method, our model achieves the state-of-the-art results, 37.8 *mAP* with the backbone of ResNet50, 39.9 *mAP* for ResNet101 and 42.0 *mAP* for ResNet101. It is worth mentioning that, our method can improve the accuracy of small objects significantly regardless of the strong baseline. Deepening the network and exploiting multi-layer information can make features more representative. We can confirm that how to fully utilized multi-level features is more relevant from the experimental results. Our method integrates multi-level features more effectively instead of extracting new features, the performance can be promoted with only a few additional parameters consequently.

C. Ablation Studies

In this section, we conduct the ablation studies on Pair-wise Attention Module and Pyramid Reconfigure Module on the COCO minival dataset. The FPN method with the backbone of ResNet-50 is used by default in all ablation studies.

Component Analysis We can achieve 37.2 *mAP* when only use the PAM, with the 0.8 *mAP* improvement over 36.4 *mAP*

of FPN. This result proves that pair-wise attention is vital when different levels fuse. Nevertheless, when we only use PRM, there is almost no performance improvement. Because information from different levels is destroyed when we traditionally join features. It is not enough to reconfigure the feature pyramid. When PAM and PRM are applied simultaneously, we can get a 1.2 *mAP* improvement. All experiment results are shown in Table II.

TABLE II
COMPONENT ANALYSIS

Module		AP Metrics					
PAM	*PRM*	*AP*	*AP50*	*AP75*	*APs*	*APm*	*APl*
		36.4	*59.0*	*39.2*	*20.3*	*38.8*	*46.4*
√		*37.2*	*60.1*	*40.8*	*21.5*	*39.6*	*47.2*
	√	*36.5*	*59.2*	*38.9*	*20.2*	*38.8*	*46.5*
√	√	*37.6*	*59.9*	*40.7*	*21.9*	*41.2*	*48.7*

Why Pair-wise Attention in PAM? Attention is widely practiced in DCNN. SE-block can provide channel-wise attention simply but effectively. We do extended experiments to evidence that pair-wise attention is more robust than self-attention in the multi-level feature fusion. As shown in the Fig. 6, we use the two methods to join the SE module. In (a), we add an SE block to each level before fusing, and after fusing in (b).

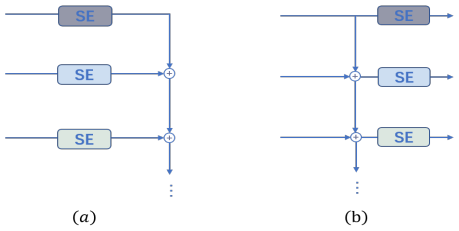

Fig. 6. SE.

Tabel III shows the experiment results, and we find that both methods have much less improvement in performance than PAM, indicating that we need mutual supervision between adjacent levels, instead of self-attention in one single level. The reason for Method (a) performing better than (b) is we introduce attention within the reverse fusion block in (a), which can participate in the information transfer between different levels efficiently.

TABLE III
PAIR-WISE ATTENTION

Method	*a*	*b*	*PAM*	*a+PRM*	*b+PRM*	*PAM+PRM*
mAP	*36.62*	*36.38*	*37.20*	*36.54*	*36.44*	*37.23*
APs	*20.41*	*20.12*	*21.5*	*20.39*	*20.24*	*21.93*

How to slice in PRM? In PRM, we consider that the characteristics carried by different channels are equivalent when reconstructing the pyramid, hence the *S* are sliced most simply as shown in Fig. 5, then the slices are rescaled and combined. To prove this point, we try different slicing methods, such as adjusting the order of slices, taking one channel every four channels to concatenate into one slice, etc. The experimental results are found to be the same as that obtained by the original method. This shows that in the same level, the information of different channels may vary slightly, but when combined with other levels, any subset of channels can represent the characteristics of the feature in current level.

V. CONCLUSION

In this paper, we mainly address the problem of scale variation in general object detection. Based on the idea of the feature pyramid and reverse fusion, we design a novel Pair-wise Attention Module(PAM) to introduce a mutual guiding mechanism between adjacent levels. Pyramid Reconfigure Module(PRM) is used to enhance the communication of cross-channel characteristics in different levels with no extra parameters. Overall, our model achieves state-of-the-art performance on the COCO dataset benchmark.

REFERENCES

[1] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *arXiv preprint arXiv:1809.02165*, 2018.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[3] T.-Y. 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*, 2017, pp. 2117–2125.

[4] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[6] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[8] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 447–456.

[9] T. Kong, A. Yao, Y. Chen, and F. Sun, "Hypernet: Towards accurate region proposal generation and joint object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 845–853.

[10] S. Bell, C. Lawrence Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2874–2883.

[11] 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*. Springer, 2016, pp. 21–37.

[12] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *european conference on computer vision*. Springer, 2016, pp. 354–370.

[13] S. Liu, D. Huang *et al.*, "Receptive field block net for accurate and fast object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 385–400.

[14] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "Dsod: Learning deeply supervised object detectors from scratch," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1919–1927.

[15] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: Deconvolutional single shot detector," *arXiv preprint arXiv:1701.06659*, 2017.

[16] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4203–4212.

[17] 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*, 2017, pp. 5936–5944.

[18] S. Woo, S. Hwang, and I. S. Kweon, "Stairnet: Top-down semantic aggregation for accurate one shot detection," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1093–1102.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[20] 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*. Springer, 2014, pp. 740–755.

[21] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[22] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[23] 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*, 2018, pp. 6154–6162.

[24] 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, pp. 2980–2988.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[26] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 315–323.

[27] T. Kong, F. Sun, C. Tan, H. Liu, and W. Huang, "Deep feature pyramid reconfiguration for object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 169–185.

[28] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8697–8710.

[29] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.