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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

A Small-sized Object Detection Oriented Multi-scale Feature Fusion Approach with Application to Defect Detection

Nianyin Zeng*, Peishu Wu, Zidong Wang, *Fellow, IEEE*, Han Li, Weibo Liu, and Xiaohui Liu

***Abstract*—Object detection is a well-known task in the field of computer vision, especially the small target detection problem which has aroused great academic attentions. In order to improve the detection performance of small objects, in this paper, a novel enhanced multi-scale feature fusion method is proposed, namely atrous spatial pyramid pooling-balanced-feature pyramid network (ABFPN). In particular, the atrous convolution operators with different dilation rates are employed to make full use of context information, where the skip connection is applied to achieve sufficient feature fusions. In addition, there is a balanced module to integrate and enhance features in different levels. Performance of the proposed ABFPN is evaluated on three public benchmark datasets, and experimental results demonstrate that it is a reliable and efficient feature fusion method. Furthermore, in order to validate the applicational potential in small objects, the developed ABFPN is utilized to detect surface tiny defects of the printed circuit board (PCB), which acts as the neck part of an improved PCB defect detection (IPDD) framework. While designing the IPDD, several powerful strategies are also employed to further improve the overall performance, which are evaluated via extensive ablation studies. Experiments on a public PCB defect detection database have demonstrated the superiority of the designed IPDD framework against other seven state-of-the-art methods, which further validates the practicality of the proposed ABFPN.**

***Index Terms*—Object detection, defect detection, feature fusion, atrous spatial pyramid pooling, printed circuit board.**

I. Introduction

Computer vision is a simulation of biological vision using computers and related equipment. Recently, computer vision has attracted enormous attention in various fields such as industrial production, agriculture, and medical health. It is known that computer vision tasks can be divided into four categories which are image classification, object detection, semantic segmentation and instance segmentation [7]. Thanks to its wide application potential in image processing and pattern recognition, object detection has received an ever-increasing research interest from both academic and industrial communities during the past few decades. With the rapid development of deep learning techniques, object detection algorithms can be divided into two groups, which are the one-stage object detection algorithms and the two-stage ones. The one-stage object detection algorithms can directly obtain the category probability as well as position coordinate values of objects, e.g., the you only look once (YOLO) models, the single shot multibox detector (SSD), and the corner network [1], [27], [37], [40]–[42]. The two-stage ones need to obtain the region proposals with rough location information, and then classify the candidate regions into different groups. Some representative two-stage object detection algorithms are the region convolutional neural network (RCNN) [13], the fast RCNN [14], the faster RCNN [43], the mask RCNN [15], and the spatial pyramid pooling network [16].

Owing to their strong abilities in defect detection and fault diagnosis, object detection algorithms have been successfully applied to a wide range of areas such as transportation, electrical and electronic engineering, biomedical engineering and so on [2], [12], [22], [23], [51], [58]. It should be pointed out that the size of the object plays a critical role in object detection, especially in industrial applications. In fact, the performance of the conventional object detection algorithms is poor by using low-level features (e.g., edge information) for small object detection. Additionally, it is difficult to extract high-level semantic features of small objects. As such, it is challenging to accurately position and classify small objects by using conventional object detection algorithms.

During the past few years, tremendous efforts have been devoted to small object detection [20], [30], [32], [33], [35], [36]. To summarize, the recently developed small object detection methods can be divided into three types: 1) using context information; 2) applying feature fusion; and 3) generating enhanced features. For example, a fully end-to-end object detector has been proposed in [20], where an object relation module has been designed to integrate the context information of the features. In [33], a feature pyramid network (FPN) has been proposed to merge the feature maps at different stages. Recently, a path aggregation network has been introduced in [36] by designing a bottom-up path enhancement branch, which could integrate the information from high-level features and low-level ones in a sufficient manner. To deal with the inconsistency among different feature scales, an adaptive spatial feature fusion method has been proposed in [35] by learning the weighting parameters. Very recently, a trident network has been presented in [32] for detecting objects in distinct sizes,

This work was supported in part by the National Natural Science Foundation of China under Grant 62073271, the International Science and Technology Cooperation Project of Fujian Province of China under Grant 2019I0003 and the Independent Innovation Foundation of AECC under Grant ZZCX-2018-017.

N. Zeng, P. Wu, and H. Li are with the Department of Instrumental and Electrical Engineering, Xiamen University, Fujian 361005, China. Email: zny@xmu.edu.cn

Z. Wang, W. Liu and X. Liu are with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom. Email: Zidong.Wang@brunel.ac.uk

*Corresponding author.

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

where the atrous convolution method with multiple dilation rates has been employed to generate different receptive fields in parallel.

Unfortunately, the aforementioned small object detection methods still have some limitations, which do not fully mine latent information such as more accurate location information and stronger semantic features from the feature maps. For instance, most context information-based object detection methods only concatenate high-level and low-level features in a simple manner, however, such fusion stage with rough stacking may cause the increase of redundant information like noise information, which may decrease the detection performance. In this case, existing small object detection methods may not be suitable for complex small object detection tasks in real-world applications such as surface defect detection for the PCB [9], tiny target detection for remote sensing images [31], and long-distance motion target detection [6]. A seemingly natural idea is to develop an advanced small object detection framework by making full use of context information and enhanced feature fusion together.

In this paper, a novel feature fusion method, an atrous spatial pyramid pooling (ASPP) balanced FPN (ABFPN), is put forward for small object detection. The developed ABFPN makes full use of the advantages of the aforementioned three types of small object detection methods. Specifically, a skip-ASPP module is developed to enhance feature fusion and expand the receptive field, where the ASPP with different dilation rate D is set in a skip-connection manner [7]. Besides, a balanced module consisting of three blocks (i.e., the resize & average block, the space nonlocal block and the residual block) is applied to learn the semantic and detailed information more effectively. The features fused by the balanced module can have balanced information from each feature map with different resolution, which can avoid the semantic information in non-adjacent layers being weakened with lateral connections. Notice that the FPN is selected as the basis of the proposed ABFPN due to its capability in dealing with multi-scale changes through the integration of low-level and high-level features. It should be emphasized that the proposed ABFPN method is a competitive feature fusion approach, which can be embedded in any existing object detection frameworks.

As a typical small object detection task, PCB surface defect detection is very important in electrical and electronic engineering. Generally speaking, the surface defects in PCB can be classified into six categories, which are missing holes, mouse bite, open circuit, short circuit, spur and spurious copper [9]. In the public datasets, it is found that the PCB surface defects normally lie in a concealed area, and some of them even exist in the tiny wiring part, which greatly increases the difficulty of surface defect detection.

Motivated by above discussions, there is a need to develop an advanced object detection framework for PCB surface defect detection. In this paper, an improved PCB defect detection (IPDD) framework is put forward for defect detection, where the proposed ABFPN is embedded as the feature fusion method in the IPDD framework. In summary, the main contributions of this paper are outlined as follows:

1) A novel feature fusion method, the ABFPN, is proposed for small object detection, where a skip-ASPP module with diverse dilation rates is designed to enlarge the receptive field. A balanced module is deployed to extract latent features for feature fusion. Experimental results demonstrate the effectiveness of the ABFPN on benchmark datasets.
2) An IPDD framework is put forward for PCB surface defect detection, where the developed ABFPN is embedded as the feature fusion method in the IPDD framework. Ablation study is conducted to verify the effectiveness of the IPDD framework.
3) The proposed IPDD framework is successfully applied to a public PCB tiny defect detection task. Experimental results demonstrate the superiority of the IPDD framework over seven state-of-the-art methods (including the improved YOLOv3 (Impro YOLOv3), the improved faster RCNN (Impro faster RCNN), the fully convolutional one-stage object detection algorithm (FCOS), the PaddlePaddle-YOLO (PP-Yolo), the tiny defect detection network (TDD-Net), the efficient multi-scale training method (sniper) and the deformable detection transformer (deformable DETR)) in terms of detection precision and recall.

The remainder of this paper is organized as follows. The proposed enhanced feature fusion method ABFPN and applied robustness enhancement strategies are elaborated in section II. Comprehensive benchmark evaluations of the ABFPN are performed in section III, with in-depth analysis of adopted strategies. In section IV, the proposed ABFPN is further used to develop the IPDD framework, which is applied to the PCB surface tiny defect detection task. Finally, conclusions and an outlook of future works are presented in section V.

II. Methodology

In this section, the structure of a typical object detection framework is first illustrated. Then, the developed ABFPN is presented where the skip-ASPP module and the balanced module are analyzed with details, which is a multi-scale feature fusion approach for small-sized object detection tasks. Meanwhile, some robustness enhancement strategies are introduced for further improving the overall performance.

A. The structure of a typical object detection framework

In a typical object detection network, there are generally four basic components which are the input layer, the backbone, the neck and the detection head [1]. The architecture of the typical object detection network is shown in Fig. 1.

In general, the input of the object detection framework requires data augmentation to boost the robustness of the training model, especially for industrial applications. Some commonly used data augmentation techniques include spatial transformations (such as random scaling, cropping and flipping) and color distortions (e.g., changing transparency, brightness and saturation). The backbone part is set for extracting features from the input layer. Some widely used models include the visual geometry group [44], the residual network (ResNet) [17], and the dark network [42]. The neck

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

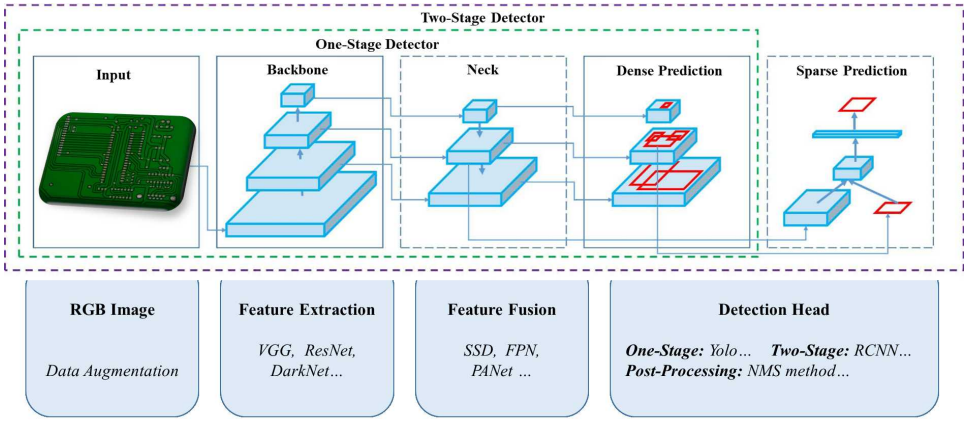


Fig. 1. A general object detection framework.

part is of vital importance in object detection. To be specific, feature fusion is carried out in the neck part to reprocess the extracted features and study the latent features according to different requirements. For example, the SSD proposed in [37] is applied for up and down sampling. The FPN can be used for path aggregation [33]. The last component of the object detection framework is the detection head which is utilized for localization and classification. It should be mentioned that there is always a post-processing module in the detection head, which usually refers to the non-maximum suppression (NMS) method [25] and its improved versions like the soft NMS method [3] and the weighted NMS method [26].

B. ABFPN: an enhanced feature fusion approach

The diagram of the proposed ABFPN is depicted in the red dashed box of Fig. 2, where the proposed ABFPN is the neck part of the object detection framework. In the ABPFN, there are two designed modules (which are the skip-ASPP module and the balanced module) for feature fusion.

In Fig. 2, C_1 denotes the feature map obtained by down-sampling the input image; $C = \{C_2, C_3, C_4, C_5\}$ denotes the feature map obtained by the corresponding residual block in the backbone at each stage. In this context, C_5 is the output feature map of the last residual block at the final stage of the backbone, which is the input of the skip-ASPP module.

Comparing with the traditional convolution operator, the atrous convolution operator could obtain a larger receptive field without increasing the number of kernel parameters. In this paper, D in the D-ASPP block stands for the dilation rate. Notice that the larger the dilation rate, the larger the corresponding receptive field. As a result, 5 different D-ASPP blocks are employed in the developed ABFPN, which enables the model to capture multi-scale context information. In the simulation, the values of D in five D-ASPP blocks are set to be 3, 6, 12, 18 and 24, respectively, which are the same as DenseASPP [50]. It should be pointed out that [50] adopts dense connection, which works well in networks with deeper layers. While in the proposed skip-ASPP module as a part of the ABFPN, skip connection has been employed, which could also reduce the computational complexity so as to speed up the convergence and inference.

The skip-connection is applied in the skip-ASPP module to enhance the interaction of the pre-output as well as post output features of each D-ASPP block and enhance the feature fusion. The work principle of the whole skip-ASPP module is formulated as follows:

$$out_i = \begin{cases} C_5 \oplus S_i\left(C_5\right), & \text{if} \quad i = 1 \\ out_{i-1} \oplus S_i\left(out_{i-1}\right), & \text{if} \quad i = 2,3,4 \\ S_i\left(out_{i-1}\right), & \text{if} \quad i = 5 \end{cases} \tag{1}$$

where $S_i\left(\cdot\right)\left(i = 1,2,3,4,5\right)$ stands for the operation of corresponding D-ASPP blocks; each D-ASPP block contains 1×1 and 3×3 atrous convolution operator with $dilation\_rate = 3, 6, 12, 18, 24$ respectively; $\oplus$ is the concatenate operation; and $out_i\left(i = 1,2,3,4,5\right)$ is the obtained result in each D-ASPP as marked in Fig. 2.

The final output of the skip-ASPP module is calculated by:

$$Out = S_1\left(C_5\right) \oplus S_2\left(out_1\right) \oplus S_3\left(out_2\right) \oplus S_4\left(out_3\right) \oplus S_5\left(out_4\right). \tag{2}$$

As shown in Fig. 2, the final output of the skip-ASPP module is then added with C_5 in the element-wise manner after the 1×1 convolution operator to obtain the feature map P_5. Similar to the conventional FPN, $P = \{P_2, P_3, P_4, P_5\}$ shares a concatenated path from P_5, which is combined with $C = \{C_2, C_3, C_4, C_5\}$ through lateral connection. In particular, the upsampled P_5, P_4 and P_3 are merged with the corresponding feature maps C_4, C_3 and C_2 in the element-wise manner. Note that the 1×1 convolution operation is performed on $\{C_2, C_3, C_4\}$ to reduce the channel dimension before merging with feature maps. After that, the obtained feature maps P (including P_2, P_3, P_4 and P_5) are fed into the balanced module. In order to balance detailed and semantic information on small target detection tasks and improve the overall detection performance, the utilized balanced module contains three blocks, which are the resize & average block, the space nonlocal block and the residual block [39], [48]. In work [39] proposed Libra RCNN solves the problem of imbalance image sampling and feature selection, especially the balanced operation of different layers. Inspired by this,

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

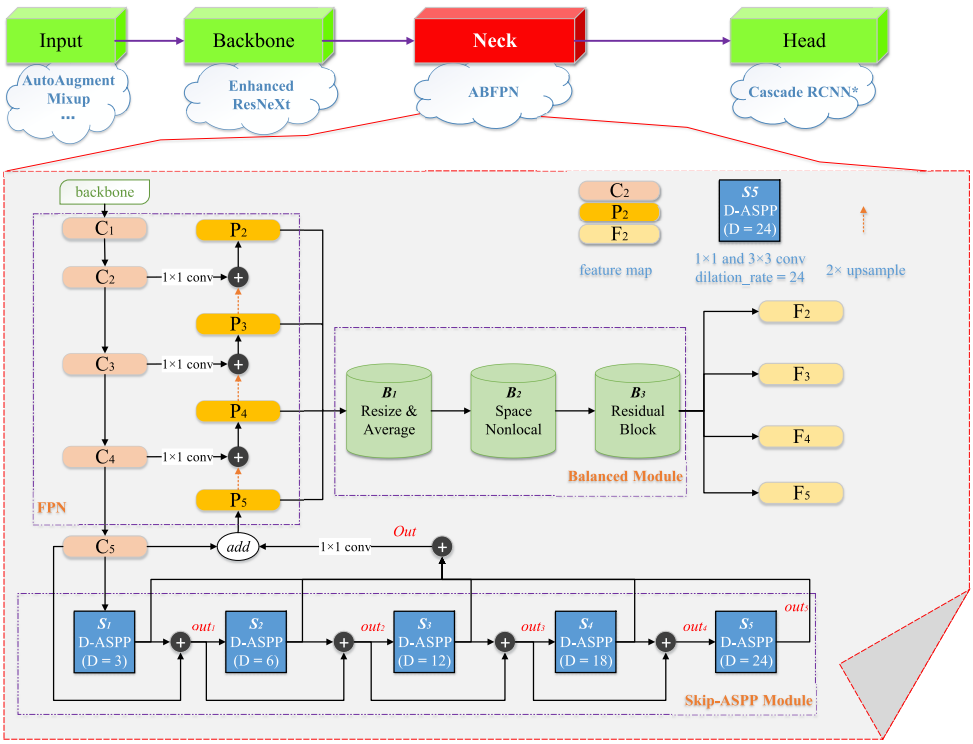


Fig. 2. Diagram of the enhanced feature fusion method ASPP-Balanced-FPN (ABFPN).

utilized "balanced module" in the ABFPN handles the further refined feature maps from FPN and skip-ASPP, which enables detection models achieve the balance of enhanced feature fusion and obtain more sufficient image context and receptive field information. To be specific, the resize & average block B_1 is designed for gathering multi-level features in P by resizing and averaging P_2, P_3 and P_5 to the same size as P_4. The output of B_1 is:

$$x = \frac{\sum \left[pool\left(P_2, 4\right), pool\left(P_3, 2\right), intp\left(P_5, 2\right) \right]}{3} \tag{3}$$

where $pool\left(P_2, 4\right)$ and $pool\left(P_3, 2\right)$ represent the max pooling operation with stride equaling to 4 and 2 for P_2 and P_3, respectively; $intp\left(P_5, 2\right)$ denotes the nearest neighbor interpolation for P_5 with multiplier factors of height and width equaling to 2.

In general, once the scale of convolutional kernels is determined, the generated receptive field will be restricted to some local regions of the feature map. To overcome the limitation of local information, the space nonlocal module B_2 is employed to gather global information of the feature map. Based on the output of B_1, the non-local output y_i is obtained by:

$$y_i = \frac{\sum_{\forall j} f\left(x_i, x_j\right) c\left(x_j\right)}{\sum_{\forall j} f\left(x_i, x_j\right)} \tag{4}$$

where $x_i \in x$ indicates the information of the current focused location; x_j represents the global information of the output of B_1; $c\left(\cdot\right)$ is the 1×1 convolution operator; and $f\left(\cdot\right)$ is the Embedded Gaussian function used to calculate the similarity of x_i and x_j. $f\left(\cdot\right)$ is defined by:

$$f\left(x_i, x_j\right) = e^{\theta\left(x_i\right)^T \cdot \phi\left(x_j\right)} \tag{5}$$

where $\theta\left(\cdot\right)$ and $\phi\left(\cdot\right)$ both stand for 1×1 convolution operator. According to [48], the output of block B_2 is:

$$z_i = c\left(y_i\right) + x_i \tag{6}$$

where $c\left(\cdot\right)$ is the 1×1 convolution operator.

The residual block B_3 scatters refined features from the output of B_2 in a multi-level manner through a residual path. To be specific, the operation of block B_3 can be expressed by the following formula:

$$F_k = P_k + intp\left(z_k, 0.5^{k-4}\right), \qquad (k = 2, 3, 4, 5) \tag{7}$$

where $intp\left(\cdot\right)$ resizes the output of the space nonlocal block z_k to be identical with the corresponding feature maps $P_k (k = 2, 3, 4, 5)$. Finally, via a skip-connection, the output feature maps $F = \{F_2, F_3, F_4, F_5\}$ of the entire ABFPN approach are obtained.

C. Robustness enhancement strategies for object detection

It is worth pointing out that the proposed ABFPN serves as the neck part in an object detection framework. The developed ABFPN aims to sufficiently merge abundant context information with hope to achieve satisfactory detection accuracy for small-size objects. To further improve the generalization ability and detection accuracy of the framework, some existing robustness enhancement strategies are employed in other components of the object detection framework.

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

In this paper, two well-known data augmentation techniques, the AutoAugmetImage method [8] and the Mixup method [52], are applied in the input part of the model training process. Both of them can enhance the model performance, and to be specific, AutoAugmetImage can automatically select the optimal combination of enhancement strategies for different datasets, which customizes a data-specific augmentation scheme. Whereas Mixup method can enrich the database via randomly mixing two samples, including their labels. By this way, influence of samples with wrong label can be greatly reduced so that the model robustness is improved.

In the backbone part, the ResNet proposed in [17] has become a popular network structure. In this paper, the ResNeXt structure [49] is adopted as the backbone, which includes stacked bottleneck paths with the same topology and one shortcut pooling path. It should be highlighted that each bottleneck path contains a squeeze-and-excitation (SE) attention mechanism, which is denoted as the attention bottleneck path in this paper. Unimportant channel features are suppressed via an SE operator in each path, and the SE operator essentially consists of one global average pooling layer and two fully connected layers with sigmoid function [21]. Moreover, the deformable convolution operator [54] is employed as a substitution of the traditional convolution operator so that the receptive field can be adaptively adjusted according to size, posture and other geometric changes of the objects. Furthermore, a stride equaling to 2 is shifted from the first 1×1 convolution operator to the 3×3 one in each attention bottleneck path. In addition, a stride equaling to 2 is shifted from the 1×1 convolution operator to the 2×2 average pooling operator in the shortcut pooling path. The operation of shifting the position with a stride size of 2 could prevent the loss of a large amount of feature information. The diagram of the enhanced ResNeXt block is displayed in Fig. 3.

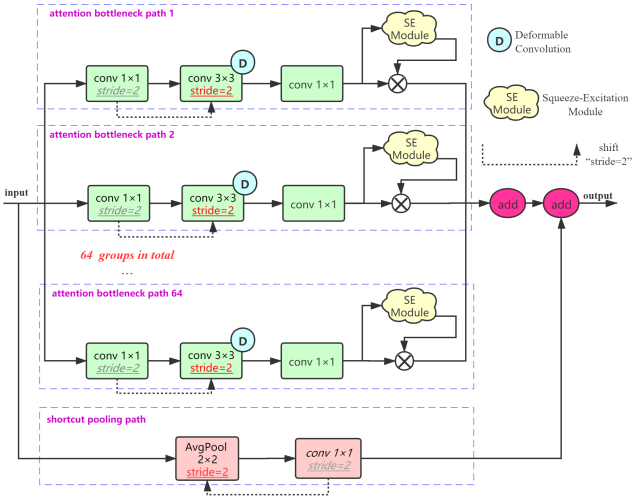


Fig. 3. Illustration of the enhanced ResNeXt block.

The cascade RCNN introduced in [4] is selected as the detection head in this paper, which is denoted by cascade RCNN* in Fig. 2. Specifically, the complete intersection over union (CIoU) loss proposed in [57] is applied to evaluate the predicted bounding box. The DIoU-NMS serves as the post-processing method [57]. The CIoU loss and DIoU-NMS are utilized in the post-processing stage of the head part in the object detection framework. It is remarkable that the employment of CIoU loss and DIoU-NMS considers 1) the overlap areas between the predicted box and ground truth; 2) the distance between the center points of the predicted box and ground truth; and 3) the aspect ratio of the bounding box, which would lead to a more reliable prediction result than traditional methods.

III. Evaluations of the Proposed ASPP-Balanced-FPN on Benchmark Datasets

In this section, sufficient ablation studies are conducted on three public benchmark datasets for verifying the performance of the proposed ABFPN, which are the COCO [34], the VOC [11] and the VisDrone detetction dataset [56]. A brief introduction of adopted datasets and experimental settings are presented. Meanwhile, the faster RCNN with ResNet50 [43] is selected as the baseline of detection method to verify the effectiveness and generalization ability of the proposed ABFPN along with the utilized robustness enhancement strategies. A series of ablation studies are performed under the same condition for evaluation.

A. Experiment settings and datasets

In this work, three well-known benchmark datasets in object detection, the MS COCO2017, the Pascal VOC07+12 and the VisDrone2019 detection dataset, are applied for performance evaluation. The COCO2017 dataset is a large-scale image dataset consisting of $330,000$ images of which more than $200,000$ are labeled. In COCO2017, there are 1.5 million object instances belonging to 80 categories. The Pascal VOC07+12 dataset contains two mutually exclusive image datasets (i.e., VOC2007 and VOC2012), which covers 20 kinds of objects, and the number of instances in Pascal VOC07+12 is over $20,000$. The VisDrone2019 dataset contains 10 classes and 54,200 instances of remotely sensed objects collected by drones, which covers complex scenes under different weather and lighting conditions, and the detected targets are relatively small in size, which makes the detection more challenging.

In the experiment on the COCO2017 dataset, the number of training and testing samples are $118,287$ and $5,000$, respectively. For the VOC07+12 dataset, $16,551$ images are used for training, and $4,952$ images are utilized for testing. The training and validation sets of the VisDrone2019 detection dataset have 7018 and 1609 images, respectively. All models are trained on the PaddlePaddle 1.8.4 framework with a single GPU TeslaV100 (16 GB memory). Detailed information of experimental settings on three datasets is presented in Table I.

B. Experimental results

As aforementioned, evaluations of the proposed ABFPN and several designed strategies are performed mainly in the form of ablation study on three benchmarks, where the two-stage network faster RCNN is selected as the baseline method.

TABLE I
EXPERIMENTAL SETTINGS

	COCO	**VOC**	**VisDrone**
Iterations	709716	270000	120000
Batch size	2	2	2
Initial learning rate	0.0025	0.02	0.02
Learning rate decay iters	[473144, 650573]	[180000, 240000]	[90000, 110000]
Learning rate decay factor	0.1	0.1	0.1
Optimizer	SGD+Momentum	SGD+Momentum	SGD+Momentum
Regularization method	L2 weight decay	L2 weight decay	L2 weight decay

1) Validation on COCO2017: Table II presents the experimental results of the ablation study on the COCO2017 dataset, where the evaluation metrics include average precision and its extensions. To be specific, AP is average precision over IoU at [0.5:0.95:0.05] (from 0.5 to 0.95 with the interval of 0.05). AP@50 is AP over IoU at 0.5. AP_s, AP_m and AP_l refer to average detection precision on small, medium-scale and large-scale objects, respectively. As shown in Table II, the AP of the proposed ABFPN is 0.9% larger than that of the FPN. On the AP_s and AP_m metrics, the results of the ABFPN are 3.7% and 0.7% larger than that of the FPN, respectively. While the AP_l result of the ABFPN is slightly smaller than that of the FPN, and the AP@50 of both methods are the same. Furthermore, experimental results of the model (that combines the ABFPN with the robustness enhancement strategies) are better than that of the faster RCNN with the neck of the FPN on all metrics. Specifically, the AP, AP@50, AP_s, AP_m and AP_l of the faster RCNN with the ABFPN and strategies is larger than that with the FPN by 4.7%, 4.4%, 6.6%, 4.3% and 4.1% respectively.

According to the experimental results, the proposed ABFPN is a reliable feature fusion method, which greatly increases the detection precision of small-size objects. Though the proposed ABFPN performs not well on the indicator AP_l, which may be caused by over-fitting because the ABFPN concentrates on the latent context information. By introducing a series of robustness enhancement strategies, the deficiency of the ABFPN on the indicator AP_l is overcome. Other indicators have been significantly increased as well, indicating an improved overall performance. As such, the combination of the ABFPN and robustness enhancement strategies performs better than the ABFPN-based faster RCNN and the traditional faster RCNN based on experimental results on the COCO2017 dataset.

2) Validation on VOC07+12: Experimental results on the VOC07+12 testing set are displayed in Table III. The popular metric mAP $(0.50, 11point)$ is employed on the VOC07+12 dataset, where mAP $(0.50, 11point)$ stands for the mean average precision values of 11 points with IoU greater than 0.5 and recall in the range of $[0 : 1 : 0.1]$ (from 0 to 1 with the interval of 0.1).

In Table III, it can be clearly observed that the mAP of the ABFPN is 84.06%, which is nearly 1% larger than that of the standard FPN. After introducing the robustness enhancement strategies, the mAP value of the modified ABFPN-based object detection framework is further increased to 85.59%, which indicates that the applied robustness enhancement strategies indeed improve the overall performance of the framework.

Furthermore, the performance comparison of the faster RCNN [43], the hierarchical shot detector (HSD) [5], the Perona Malik [24], the intertwiner network (InterNet) [28], the refinement detector (RefineDet) [53], the Blitz Network (BlitzNet) [10], the early exit evolutionary architecture network (EEEA-Net) [46] and our method on the VOC07+12 dataset is shown in Table IV. Notice that the data of the utilized methods are directly obtained from the corresponding literature, which is marked in Table IV. Experimental results demonstrate the effectiveness of the proposed ABFPN for small-size objects detection comparing with some state-of-the-art algorithms. It is noteworthy that the comparison algorithms used are architecturally designed to be suitable for application to small target detection tasks, hence the results are totally comparable. Specifically, the proposed method achieves the best result in terms of mAP.

3) Validation on VisDrone2019: In this part, the results of the ABFPN-based object detection framework with the robustness enhancement strategies on the VisDrone2019 detection dataset are with the ablation studies in Table V. It is worth mentioning that the detection on VisDrone2019 dataset is a difficult small-sized target detection task, and the chosen evaluation metrics are the same as used on COCO2017 dataset. As can be seen from Table V, the ABFPN can also guarantee a 1% improvement in average precision on complex detection tasks compared to the FPN, and the better performance is especially noticeable on smaller size targets. When related strategies are further introduced, the improvement in the five metrics AP, AP@50, AP_s, AP_m and AP_l is 2.5%, 3.2%, 2.3%, 3.5% and 2.7%, respectively, compared to the original FPN.

The validation of the ablation experiments on the above three public challenging datasets demonstrates the effectiveness of the proposed ABFPN and related strategies, which are particularly suitable for small-sized detection tasks; meanwhile, generalization ability of the ABFPN is also proven on multiple databases. To further validate the practicality of the ABFPN, in next section, it is applied to detect tiny surface defects of PCB.

IV. APPLICATION IN PCB DEFECT DETECTION

In this section, an IPDD framework is designed to detect tiny surface defects in PCB, where the proposed ABFPN is incorporated with the aforementioned robustness enhancement strategies. To verify its effectiveness and practicality, 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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

TABLE II
ABLATION STUDY ON THE MS COCO2017 DATASET

$Algorithms$	$AP(\%)$	$AP@50(\%)$	$AP_s(\%)$	$AP_m(\%)$	$AP_l(\%)$
Faster RCNN with the FPN	33.9	56.9	17.8	37.7	45.8
Faster RCNN with the ABFPN	34.8	56.9	21.5	38.4	44.1
Faster RCNN with the ABFPN and strategies	**38.6**	**61.3**	**24.4**	**42.0**	**49.9**

TABLE III
ABLATION STUDY ON THE PASCAL VOC07+12 TESTING DATASET

$Algorithms$	$mAP(0.50, 11point)(\%)$
Faster RCNN with the FPN	83.07
Faster RCNN with the ABFPN	84.06
Faster RCNN with the ABFPN and strategies	**85.59**

TABLE IV
DETECTION EVALUATION RESULTS OF DIFFERENT ALGORITHMS ON THE PASCAL VOC07+12 TESTING DATASET

$Algorithms$	$mAP(0.50, 11point)(\%)$
Faster RCNN [43]	73.2
HSD [5]	83.0
Perona Malik [24]	74.37
BlitzNet [10]	81.5
EEEA-Net [46]	81.8
InterNet [28]	82.7
RefineDet [53]	83.8
Ours	**85.59**

developed IPDD framework is tested on the public PCB defect dataset.

A. The improved PCB defect detection framework

The proposed IPDD framework consists of the input layer, the backbone, the neck, and the detection head. The diagram of the IPDD framework is displayed in Fig. 4. The enhancement strategies used in each part of the IPDD framework are described in Section II-C. It is worth emphasizing that the enhanced ResNeXt structure (including 152 layers with 50 blocks) is selected as the backbone, which is denoted as Enhanced-ResNeXt-152. Meanwhile, the proposed ABFPN is chosen as the neck part, and the cascade RCNN* is selected as the detection head.

In object detection, localization and classification are the most significant tasks, by which the object bounding box and the corresponding category are determined correctly. In Fig. 4, the localization and classification are highlighted within a blue box. In the proposed IPDD framework, the head part employs a region proposal network (RPN) to obtain regions of interest (RoI). In addition, the RPN is applied to distinguish the foreground (i.e., the PCB surface defects) and the background.

As stated previously, the feature maps $F = \{F_2, F_3, F_4, F_5\}$ are the final output of the neck part and are also the input of the detection head. Then, multiple proposals with different sizes and aspect ratios are generated at each position of the feature map. Each proposal is matched with a corresponding ground truth and performed by the IoU threshold filtering operation, which could thus distinguish positive and negative samples. The bounding box loss function L_{rpn_bbox} in the RPN is expressed by:

$$L_{rpn\_bbox} = \begin{cases} M\left[0.5 * (loc_p - loc_t)^2\right], & \text{if} \quad dif < \sigma \\ M\left[\sigma * |loc_p - loc_t| - 0.5 * \sigma^2\right], & \text{otherwise} \end{cases} \tag{8}$$

where M is the average operation; loc_p and loc_t represent the location of bbox (short for bounding box) predicted by the RPN and the target bbox, respectively; $dif = |loc_p - loc_t|$ is the absolute value of the difference between loc_p and loc_t; σ is the threshold parameter, which is set to 3 in this simulation. Besides, the classification loss function of the RPN is:

$$L_{rpn\_cls} = M\left\{-s^j_{cls} \cdot l_j + log\left(\sum_{i=0}^{K} \exp\left(s^i_{cls}\right)\right)\right\}, \quad (j = 1, 2, ..., K) \tag{9}$$

where s_{cls} denotes the prediction score, l is the real label and K represents the total number of categories.

It should be highlighted that the RPN only accomplishes the rough proposals, which need further refinements. In fact, a single PCB image may probably contain more than one defect. As such, it is of vital significance to further identify each type precisely from the proposals. Both the feature map F and the generated RoI are performed a series of cascade operations, denoted by the RoI align and the Bbox head blocks as shown in Fig. 4.

Three cascade levels are re-sampled to increase the IoU value of the proposals stage by stage. The "RoI align" blocks adjust features of the candidate areas to a fixed size through the pooling operation. The "Bbox head" blocks obtain the prediction bounding box B_{pre} and classification score S_{cls}. Each cascade stage is trained by using the positive and negative samples with different IoUs, and the output of previous stage serves as the input of next stage. If the IoU of the generated RoI increases, the next cascade stage will focus on a certain area in the updated proposal so as to improve the detection accuracy.

For loss functions of the detection head, the classification loss function L_{head_cls} adopts the cross entropy loss function as shown in Eq. 9, and the CIoU loss mentioned in Section II-C is used for the bounding box loss L_{head_bbox}. The bounding box loss L_{head_bbox} of the head is calculated by:

$$L_{head\_bbox} = M\left\{1 - IoU + dist\left(b_p, b_t\right) + \alpha\nu\right\} \tag{10}$$

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

TABLE V
ABLATION STUDY ON THE VISDRONE2019 DETECTION DATASET

Algorithms	*AP*(%)	*AP*@50(%)	AP_s(%)	AP_m(%)	AP_l(%)
Faster RCNN with the FPN	14.6	26.5	7.8	22.5	27.5
Faster RCNN with the ABFPN	15.6	27.4	8.6	23.8	26.8
Faster RCNN with the ABFPN and strategies	**17.1**	**29.7**	**10.1**	**26.0**	**30.2**

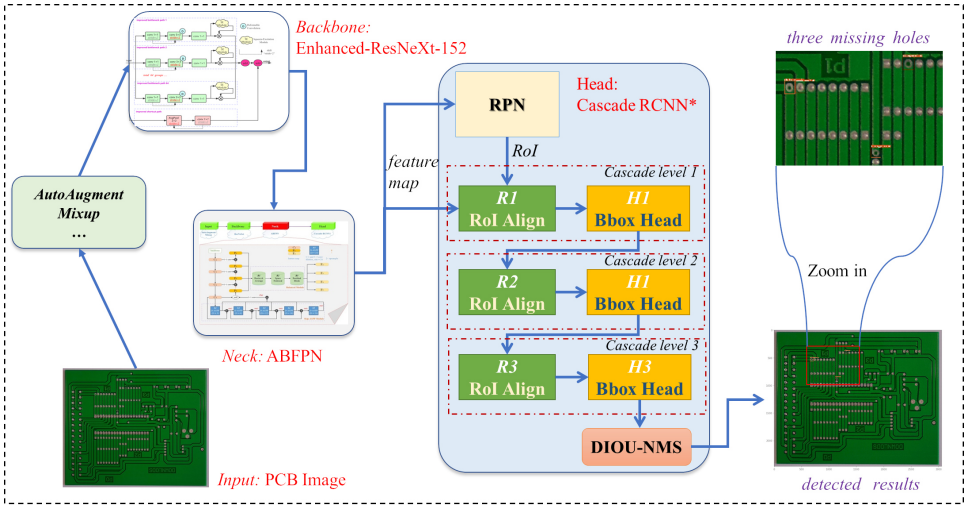


Fig. 4. The diagram of the proposed IPDD framework.

where b_p and b_t represent the predicted box and the real bounding box, respectively; α and ν are two influence factors with respect to the aspect ratio of b_p and b_t. $dist\left(\cdot\right)$ calculates the distance between b_p and b_t, which is defined by:

$$dist\left(b_p, b_t\right) = \frac{\rho^2\left(b_p, b_t\right)}{c^2} \tag{11}$$

where c is the diagonal distance of the smallest bounding rectangle, which can cover both b_p and b_t; $\rho\left(\cdot\right)$ stands for the Euclidean distance.

The total loss function of the cascade RCNN* is given by:

$$L_{total} = L_{rpn\_cls} + L_{rpn\_bbox} + \sum_{i=1}^{3}\left(L^i_{head\_cls} + L^i_{head\_bbox}\right) \tag{12}$$

where L_{head_cls} is the cross entropy loss function as shown in Eq. 9.

The DIoU-NMS method is applied for further refining the prediction results to preserve the best bounding box, as there may be other redundant PCB tiny defects. It is worth mentioning that the DIoU-NMS method considers not only the IoU value but also the distance between center points of two bounding boxes. The DIoU-NMS method provides a score as reference and the process of the method is:

$$score_i = \begin{cases} score_i, & \text{if} \quad \left|IoU - dist\left(b_M, b_i\right)\right| < \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where ϵ is the threshold of the DIoU-NMS method, which is set to be 0.5 in this work; b_M is the bounding box with the highest confidence value, and b_i stands for nearby boxes. If the score is set to be 0, the corresponding box will be redundant for a certain defect, which will be filtered out. Otherwise, a small value of $\left|IoU - dist\left(b_M, b_i\right)\right|$ implies that the obtained box may belong to another defect, which should not be eliminated arbitrarily.

The pseudocode of the proposed IPDD framework is provided in Algorithm 1.

B. Evaluation results and discussions of the IPDD framework

To evaluate the performance of the proposed IPDD framework, the PKU public PCB defect detection dataset has been adopted [9]. Some existing defect detection algorithms have been utilized for performance evaluation, including the Impro YOLOv3 [29], the FCOS [47], the PP-Yolo [38], the Impro faster RCNN [19], the TDD-Net [9], the deformable DETR [55] and the sniper [45]. Among the utilized methods, the Impro faster RCNN, the TDD-Net, the deformable DETR and the sniper are two-stage methods, which are similar to our IPDD framework.

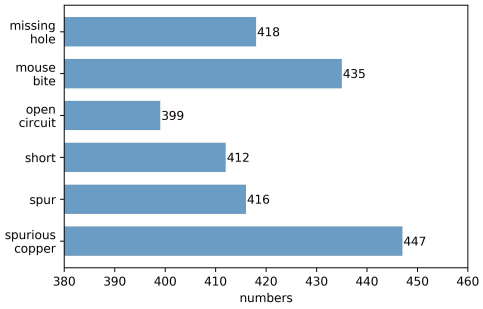
The utilized dataset contains 693 images with 6 different types of defects (including missing hole, mouse bite, open circuit, short, spur and spurious copper). The dataset is visualized in Fig. 5, where the number of each defect type is plotted in Fig. 5 (a). $area\_ratios$ is the proportion of ground truth bounding box to entire image, which also reflects the relative size of objects for detection. In Fig. 5 (b), it is clear that almost all defects only occupy a tiny area in an image, which makes it challenging to achieve accurate positioning and classification results.

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

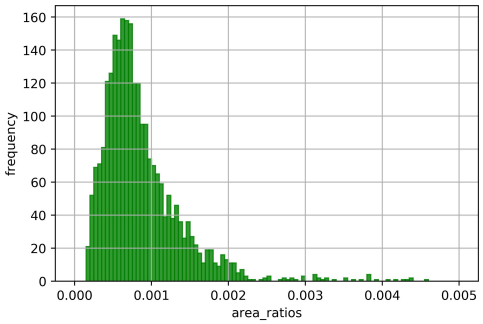
Algorithm 1 Pseudocode of the proposed IPDD framework.

Require:
RGB images with PCB surface defects
Ensure:
The predicted bounding boxes and corresponding classification results of PCB defects

1: Use the AutoAugmetImage and mixup techniques for data augmentation;
2: the Backbone part Enhanced-ResNeXt-152 returns feature maps $C = \{C_2, C_3, C_4, C_5\}$;
3: the Neck part ABFPN outputs feature maps $F = \{F_2, F_3, F_4, F_5\}$ based on Eq. 1 - Eq. 7;
4: Enter the region proposal network (RPN) in the head part to generate regions of interest (RoI);
5: Calculate the loss of the RPN, including L_{rpn_bbox} by Eq. 8 and L_{rpn_cls} by Eq. 9;
6: **For** i from 1 to 3:
Perform RoI align feature extraction;
Obtain the updated bounding box B_{pre} and classification score S_{cls};
Calculate L_{head_bbox} and L_{head_cls} referring to Eq. 10 and Eq. 11;
Endfor
7: Calculate total loss L_{total} of the head part according to Eq. 12;
8: Apply the DIoU-NMS method for further refinement;
9: Get the final prediction bounding boxes and corresponding classification scores.



(a) The number of each PCB defect type



(b) The frequency of various area_ratios

Fig. 5. Partial visualization information of the PCB defect detection dataset.

In the simulation, the proposed IPDD framework is trained with $50,000$ iterations, and the initial learning rate is 0.00125. The decay factor is 0.1 in the iteration interval of $[42000, 48000]$. 593 PCB images are randomly selected as the training samples whereas the rest 100 pictures are used for testing. Other experimental settings and environment remain the same as presented in Section III.

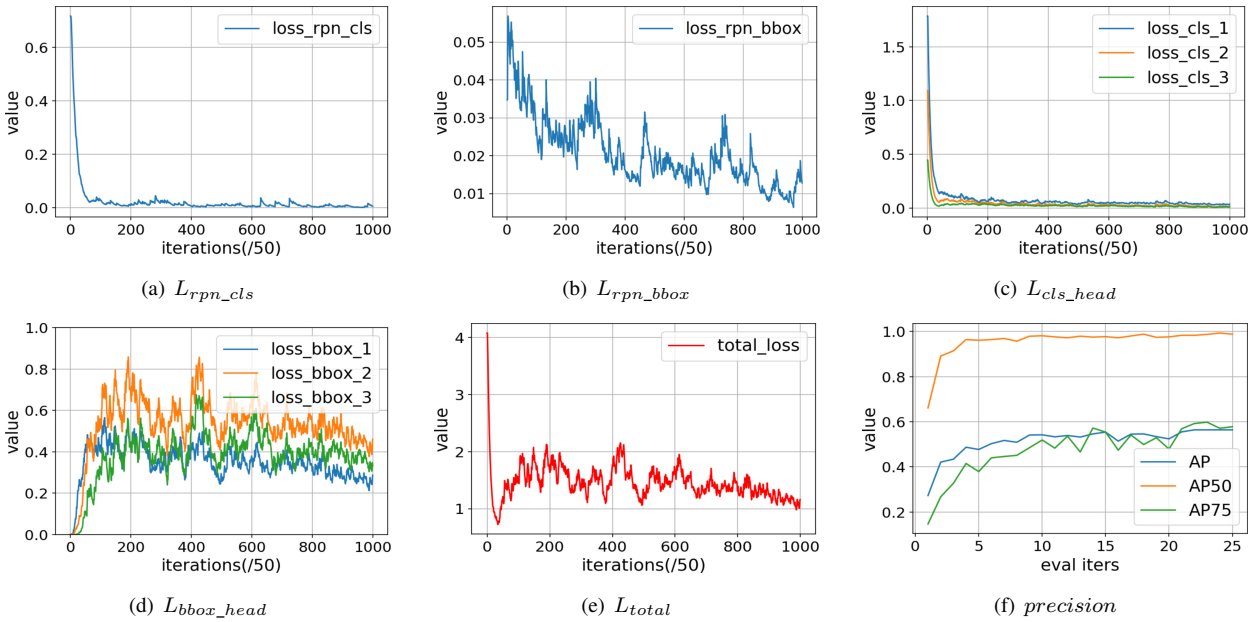
1) Algorithm verification and comparison: The change curves of five loss functions (i.e., L_{rpn_cls}, L_{rpn_bbox}, L_{head_cls}, L_{head_bbox} and L_{total}) are shown in Fig. 6, where each loss value is calculated every 50 iterations. It is observed that when iteration passes nearly $880 \times 50 = 44000$, the oscillation of L_{total} is restricted in a small range, which can be deemed to reach the stable state. Besides, Fig. 6 (f) presents the change of AP, where AP@50 and AP@75 denote AP over IoU at 0.5 and 0.75, respectively. The precision value is sampled every 2000 iteration, and when evaluation times reach 22, i.e., the number of iterations is $22 \times 2000 = 44000$, the curves tend to be converged when AP, AP@50 and AP@75 are 56.4%, 98.8%, and 57.8%, respectively.

Table VI displays the comparison results of the proposed IPDD framework and the other seven state-of-the-art detection methods. It is noteworthy that Impro YOLOv3, FCOS, deformable DETR and sniper are all the detection methods with excellent performance in small-sized object detection tasks, and TDD-Net is a specific method proposed for PCB small defect detection. Evaluation metrics are the same as presented in Table II with two extra ones which are AP@75 and average recall (AR) rate. The larger AR rate, the more positive samples are classified correctly. As shown in Table VI, the proposed IPDD framework achieves the best results on all performance indicators, which demonstrates the effectiveness of the IPDD framework for PCB defect detection. In particular, the IPDD framework outperforms the sub-optimal method *sniper* on all evaluation metrics of AP, AP@50, AP@75, AP_s, AP_m, AP_l and AR. Compared with Impro YOLOv3 (which ranks second on AP_s), the indicator AP_s is improved by 1.7% when using the IPDD framework, which indicates the superiority of the proposed IPDD framework on detecting small defects.

In addition, TDD-Net, a dedicated algorithm proposed for PCB tiny defect detection, is selected in this paper as a comparison method for visualization and subsequent error analysis. For an intuitive view, experimental results of the proposed IPDD framework and the TDD-Net are visualized in Fig. 7. The first two columns are results obtained by the IPDD framework, where images are enlarged for a clear view. Similarly, the last two columns are results obtained by the TDD-Net. It should be highlighted that for the *mouse_bite* defect shown in line 3, the TDD-Net outputs a redundant prediction bounding box. By using the proposed IPDD framework, the positioning is more accurate than that of the TDD-Net with a higher confidence value that equals to 0.99, which shows that the proposed IPDD framework demonstrates better overall performance than TDD-Net in terms of both localization and classification of small objects.

Furthermore, to comprehensively evaluate the detection performance of the proposed IPDD framework on each type of defect, the precision-recall (PR) and score-recall (SR) curves are employed for evaluation. Experimental results are shown

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement



(a) L_{rpn_cls} (b) L_{rpn_bbox} (c) L_{cls_head}

(d) L_{bbox_head} (e) L_{total} (f) *precision*

Fig. 6. Iteration curves of loss functions and precision.

TABLE VI
COMPARISONS OF DIFFERENT METHODS FOR PCB DEFECT DETECTION

Algorithms	$AP(\%)$	$AP@50(\%)$	$AP@75(\%)$	$AP_s(\%)$	$AP_m(\%)$	$AP_l(\%)$	$AR(\%)$
Impro YOLOv3 [29]	43.6	94.8	30.4	45.9	44.4	31.7	51.9
Impro faster RCNN [19]	48.6	93.9	42.5	28.6	49.1	41.4	55.0
FCOS [47]	48.7	94.8	43.1	41.3	49.8	35.0	55.8
PP-Yolo [38]	47.4	95.1	38.4	27.5	48.3	45.3	63.2
Deformable DETR [55]	49.2	96.1	42.4	44.7	49.0	49.2	58.9
TDD-Net [9]	49.3	95.2	43.7	32.1	50.1	36.2	56.6
Sniper [45]	51.4	97.9	47.2	45.7	51.4	54.1	60.5
IPDD framework (In this paper)	**56.4**	**98.8**	**57.8**	**47.6**	**56.6**	**57.2**	**63.3**

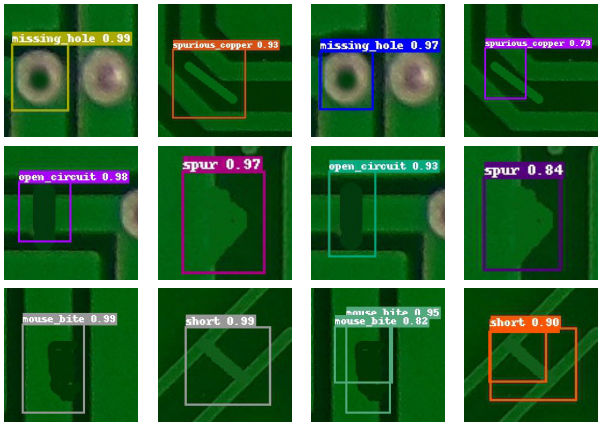


Fig. 7. Comparison of visualization results of our IPDD framework (left two columns) and TDD-Net (right two columns).

in Fig. 8, where the IoU threshold is fixed to 0.5. The PR curve reflects a trade-off between classification accuracy and capability to cover positive samples (i.e., recall). The SR curve shows the confidence scores under different recall values.

Generally, the value of precision and confidence scores will monotonically decline as recall increases. Thus, an effective and practical model is supposed to enable the precision and confidence scores to maintain stable even when recall is increased. As a result, the larger area enclosed by PR, SR curves and coordinate axes, the better performance of the model. Fig. 8 shows that the IPDD framework is able to keep the value of precision and confidence score at a high level with growing recall, which validates the robustness and reliability of the IPDD framework on PCB defect detection.

2) Ablation study and error analysis: To further validate the effectiveness of our proposed IPDD framework, an ablation study has been conducted in this paper, where two variant IPDD frameworks (i.e., IPDD-Nv1 and IPDD-Nv2) are adopted. To be specific, the IPDD framework employs both the designed ABFPN and other robustness enhancement strategies. In the variant IPDD framework, IPDD-Nv1, the input layer is the original one without using data argumentation techniques, the backbone is the conventional ResNeXt-152, the neck part is the proposed ABFPN, and the cascade RCNN is used as the detection head. The only difference between IPDD-Nv2 and IPDD-Nv1 is that the neck part in IPDD-Nv2 is a conventional FPN.

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

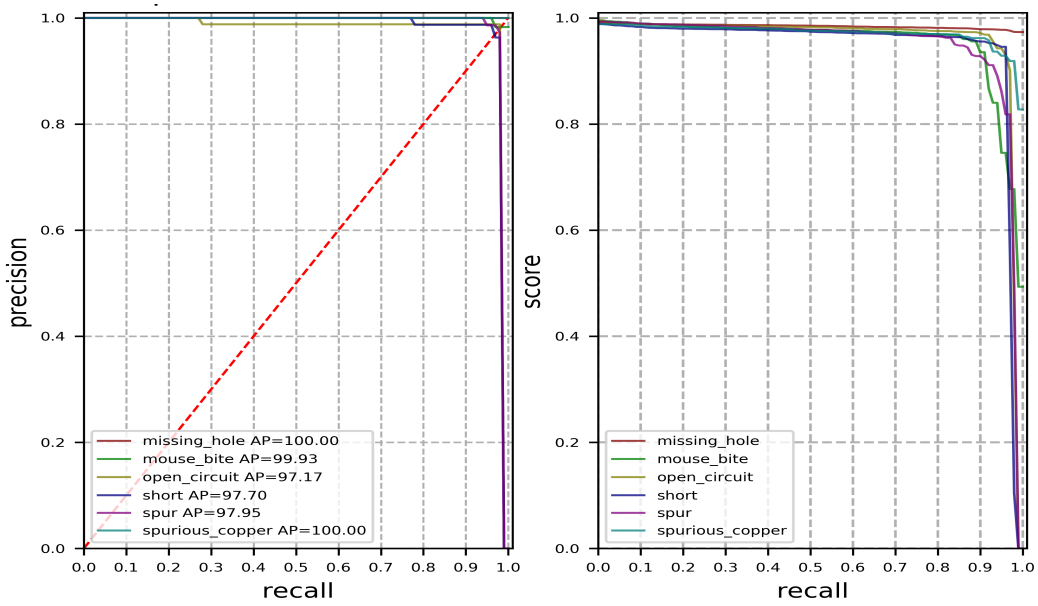


Fig. 8. Precision-recall and score-recall curves of each defect type (IoU=0.5).

The ablation study results are shown in Table VII. It can be seen in Table VII that IPDD-Nv1 outperforms IPDD-Nv2 on all indicators, particularly on AP_s. The AP_s of IPDD-Nv1 is 3.1% larger than that of the IPDD-Nv2, which indicates the competitiveness of the designed ABFPN (that can be seen as an outstanding feature fusion method). By further introducing robustness enhancement strategies, it is found that except for AP_s, the IPDD framework has increased by 0.6%, 0.6%, 3.6%, 0.6%, 2.9% and 0.7% respectively on AP, AP@50, AP@75, AP_m, AP_l and AR when compared with IPDD-Nv1. On the AP_s metric, the value of AP_s in the proposed IPDD framework is slightly smaller than that of the IPDD-Nv1, due mainly to the reason that the applied strategies focus on objects with middle-size or large-size. As such, the proposed IPDD framework could achieve satisfactory overall detection performance. Improvements on other six indicators have demonstrated that other introduced strategies can effectively enhance robustness of the model.

Fig. 9 is the scatter plot of the precision and recall, including eight PCB defect detection methods and the two variant IPDD frameworks. Based on the relationship between precision and recall, the point in the upper right corner indicates that the model is robust. As can be seen in Fig. 9, the proposed IPDD framework is the best out of 10 methods. It should also be noticed that the variant IPDD-Nv1 which only employs the proposed ABFPN ranks second, which implies that the introduced ABFPN is competitive in small object detection.

Additionally, the PR curve is used for error analysis [18]. Fig. 10 (a) and (b) show the PR curves of the TDD-Net and the IPDD framework, where seven colored areas are marked. To be specific, C_{75} and C_{50} stand for the area enclosed by the PR curve and coordinate axes at IoU = 0.75 and IoU = 0.5, respectively. Compared with the TDD-Net, the proposed IPDD framework has an improvement on the AP by 3.6% on C_{50} and 14.1% on C_{75}, which indicates the effectiveness

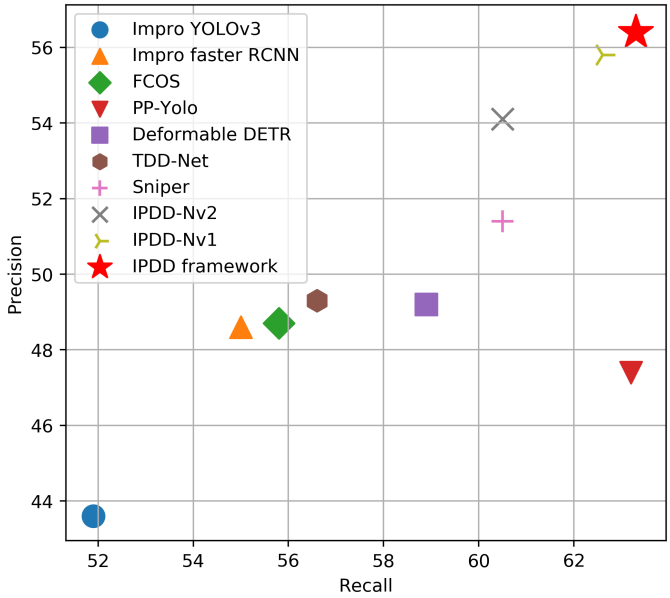


Fig. 9. Scatter plot of the precision-recall relationship of each algorithm.

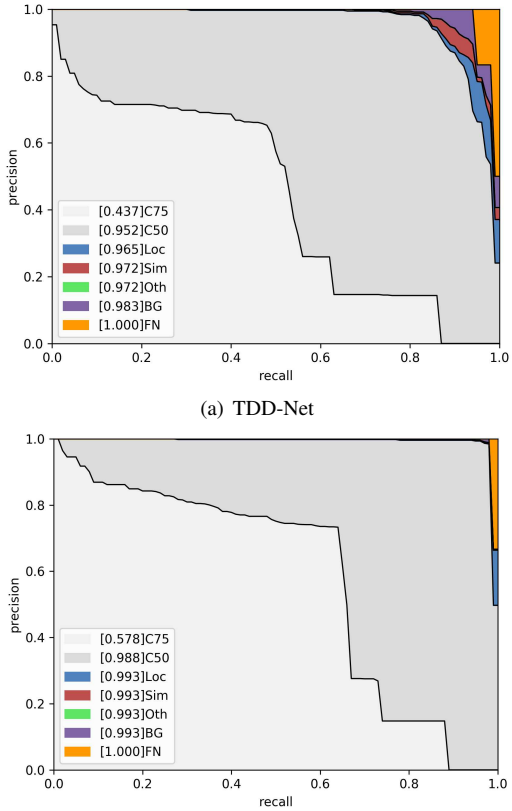
and superiority of the proposed ABFPN in positioning. After removing location errors, the obtained new area is denoted by the indicator Loc. Notice that AP of the IPDD framework on Loc is further increased from 98.8% to 99.3%, whereas AP of TDD-Net on Loc is changed from 95.2% to 96.5%, which indicates that inaccurate localization is a common reason that causes the low detection performance. To conclude, the proposed IPDD framework performs better than the TDD-Net.

The indicator Oth is the value of AP after eliminating all misclassification results; and further when all false positive samples are removed, the AP value is characterized by BG.

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: DOI10.1109/TIM.2022.3153997, IEEE Transactions on Instrumentation and Measurement

TABLE VII
ABLATION STUDY RESULTS OF THE IPDD FRAMEWORK

Algorithms	*AP*	*AP*@50	*AP*@75	AP_s	AP_m	AP_l	*AR*
IPDD-Nv2	54.1	98.0	53.2	45.7	54.3	54.2	60.5
IPDD-Nv1	55.8	98.2	54.2	**48.8**	56.0	54.3	62.6
IPDD framework	**56.4**	**98.8**	**57.8**	47.6	**56.6**	**57.2**	**63.3**



(a) TDD-Net

(b) IPDD framework

Fig. 10. Error analysis via precision-recall curves.

It is found that both Oth and BG remain unchanged in Fig.10 (b), which shows that the IPDD framework could achieve precise classifications. The results on AP regarding Oth and BG in the TDD-Net demonstrate that the classification accuracy of the TDD-Net is worse than that of the proposed IPDD framework. The last indicator FN is the AP value after eliminating all kinds of mistakes. Based on the above discussions, the proposed IPDD framework demonstrates remarkable classification accuracy, and the main reason for inaccurate detection is imperfect positioning performance.

V. CONCLUSION

In this paper, an IPDD framework has been put forward for PCB surface defect detection, where an ABFPN has been designed as the neck part of the IPDD framework for feature fusion. In the developed ABFPN, the atrous convolution operator with different dilation rates has been utilized to enlarge the receptive field. The skip connection has been adopted for the atrous convolution operators, which could enhance the interactions among features in different levels. In addition, a balanced module has been introduced in the ABFPN for studying the semantic information of the obtained features. The performance of the ABFPN has been evaluated on three public datasets, and the ablation studies prove the effectiveness of the ABFPN especially for small-sized objects. The designed IPDD framework has been successfully applied to small object detection with application to PCB surface defect detection. Several robustness enhancement strategies have been employed in the IPDD framework to further improve the overall detection performance. Experimental results have demonstrated the superiority of the proposed IPDD framework over seven state-of-the-art methods in terms of both localization and classification.

In the future, we aim to 1) apply the proposed IPDD framework to other small object detection tasks such as defect detection of industrial components and object detection in pastoral landscapes; 2) investigate a precise localization method to improve the positioning performance of the IPDD framework; and 3) utilize evolutionary computation algorithms to tune the hyperparameters of the proposed IPDD framework.

REFERENCES

[1] A. Bochkovskiy, C. Wang and H. Liao, YOLOv4: optimal speed and accuracy of object detection, *arXiv preprint:2004.10934*, 2020.
[2] Y. Bao, K. Song, J. Liu, Y. Wang, Y. Yan, H. Yu and X. Li, Triplet-graph reasoning network for few-shot metal generic surface defect segmentation, *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021.
[3] N. Bodla, B. Singh, R. Chellappa and L. Davis, Soft-NMS–improving object detection with one line of code, In: *Proceedings of the 16th IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, Oct. 2017, pp. 5562–5570.
[4] Z. Cai and N. Vasconcelos, Cascade R-CNN: high quality object detection and instance segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 1483–1498, 2021.
[5] J. Cao, Y. Pang, J. Han and X. Li, Hierarchical shot detector, In: *Proceedings of the 17th IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea, Oct. 2019, pp. 9704–9713.
[6] E. Chen, O. Haik and Y. Yitzhaky, Online spatio-temporal action detection in long-distance imaging affected by the atmosphere, *IEEE Access*, vol. 9, pp. 24531–24545, 2021.
[7] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy and A. 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, pp. 834–848, 2018.
[8] E. Cubuk, B. Zoph, D. Mane, V. Vasudevan and Q. Le, AutoAugment: learning augmentation strategies from data, In: *Proceedings of the 32th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, U.S., Jun. 2019, pp. 113–123.
[9] R. Ding, L. Dai, G. Li and H. Liu, TDD-net: a tiny defect detection network for printed circuit boards, *CAAI Transactions on Intelligence Technology*, vol. 4, pp. 110–116, 2019.
[10] N. Dvornik, K. Shmelkov, J. Mairal and C. Schmid, BlitzNet: a real-time deep network for scene understanding, In: Proceedings of the 16th IEEE International Conference on Computer Vision (ICCV), Venice, Italy, Oct. 2017, pp. 4174-4182.

[11] M. Everingham, L. Gool, C. Williams, J. Winn, A. Zisserman and C. Zitnick, The pascal visual object classes (VOC) challenge, *International Journal of Computer Vision*, vol. 88, pp. 303–338, 2010.

[12] H. Geng, H. Liu, L. Ma and X. Yi, Multi-sensor filtering fusion meets censored measurements under a constrained network environment: advances, challenges and prospects, *International Journal of Systems Science*, vol. 52, no. 16, pp. 3410–3436, 2021.

[13] R. Girshick, J. Donahue, T. Darrell and J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, In: *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, U.S., Jun. 2014, pp. 580–587.

[14] R. Girshick, Fast R-CNN, In: *Proceedings of the 15th IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[15] K. He, G. Gkioxari, P. Dollar and R. Girshick, Mask R-CNN, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 386–397, 2020.

[16] 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, pp. 1904–1916, 2015.

[17] K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, In: *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, U.S., Jun. 2016, pp. 770–778.

[18] D. Hoiem, Y. Cho and Q. Dai, Diagnosing error in object detectors, In: *Proceedings of the 12th European Conference on Computer Vision (ECCV)*, Firenze, Italy, Oct. 2012, pp. 340–353.

[19] B. Hu and J. Wang, Detection of PCB surface defects with improved faster-rcnn and feature pyramid network, *IEEE Access*, vol. 8, pp. 108335–108345, 2020.

[20] H. Hu, J. Gu, Z. Zhang, J. Dai and Y. Wei, Relation networks for object detection, In: *Proceedings of the 31th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, U.S., Jun. 2018, pp. 3588–3597.

[21] J. Hu, L. Shen, S. Albanie, G. Sun and E. Wu, Squeeze-and-excitation networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 2011–2023, 2020.

[22] J. Hu, H. Zhang, H. Liu and X. Yu, A survey on sliding mode control for networked control systems, *International Journal of Systems Science*, vol. 52, no. 6, pp. 1129–1147, 2021.

[23] Y. Ju, X. Tian, H. Liu and L. Ma, Fault detection of networked dynamical systems: a survey of trends and techniques, *International Journal of Systems Science*, vol. 52, no. 16, pp. 3390–3409, 2021.

[24] S. Mishra, A. Shah, A. Bansal, J. Choi, A. Shrivastava, A. Sharma and D. Jacobs, Learning visual representations for transfer learning by suppressing texture, *arXiv preprint:2011.01901*, 2020.

[25] A. Neubeck and L. Van Gool, Efficient non-maximum suppression, In: *Proceedings of the 18th International Conference on Pattern Recognition (ICPR)*, Hong Kong, China, Aug. 2006, pp. 850–855.

[26] C. Ning, H. Zhou, Y. Song and J. Tang, Inception single shot multiBox detector for object detection, In: *Proceedings of the 17th IEEE International Conference on Multimedia & Expo Workshops (ICME)*, Hong Kong, China, Jul. 2017, pp. 549–554.

[27] H. Law and J. Deng, CornerNet: detecting objects as paired keypoints, In: *Proceedings of the 15th European Conference on Computer Vision (ECCV)*, Munich, Germany, Sep. 2018, pp. 734–750.

[28] H. Li, B. Dai, S. Shi, W. Ouyang and X. Wang, Feature intertwiner for object detection, *arXiv preprint:1903.11851*, 2019.

[29] J. Li, J. Gu, Z. Huang and J. Wen, Application research of improved YOLO v3 algorithm in PCB electronic component detection, *Applied Sciences*, vol. 9, pp. 3738–3750, 2019.

[30] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng and S. Yan, Perceptual generative adversarial networks for small object detection, In: *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, U.S., Jul. 2017, pp. 1951–1959.

[31] J. Li and Z. Liu, Self-measurements of point-spread function for remote sensing optical imaging instruments, *IEEE Transactions on Instrumentation and Measurement*, vol. 69, pp. 3679–3686, 2020.

[32] Y. Li, Y. Chen, N. Wang and Z. Zhang, Scale-aware trident networks for object detection, In: *Proceedings of the 17th IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea, Oct. 2019, pp. 6053–6062.

[33] T. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan and S. Belongie, Feature pyramid networks for object detection, In: *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, U.S., Jul. 2017, pp. 936–944.

[34] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar and C. Zitnick, Microsoft COCO: common objects in context, In: *Proceedings of the 13th European Conference on Computer Vision (ECCV)*, Zurich, Switzerland, Sep. 2014, pp. 740–755.

[35] S. Liu, D. Huang and Y. Wang, Learning spatial fusion for single-shot object detection, *arXiv preprint:1911.09516*, 2019.

[36] S. Liu, L. Qi, H. Qin, J. Shi and J. Jia, Path aggregation network for instance segmentation, In: *Proceedings of the 31th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, U.S., Jun. 2018, pp. 8759–8768.

[37] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. CBerg, SSD: single shot multibox detector, In: *Proceedings of the 14th European Conference on Computer Vision (ECCV)*, Amsterdam, Netherlands, Oct. 2016, pp. 21–37.

[38] X. Long, K. Deng, G. Wang, Y. Zhang, Q. Dang, Y. Gao, H. Shen, J. Ren, S. Han, E. Ding and S. Wen, PP-YOLO: an effective and efficient implementation of object detector, *arXiv preprint:2007.12099*, 2020.

[39] 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 32th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, U.S., Jun. 2019, pp. 821–830.

[40] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, You only look once: unified, real-time object detection, In: *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, U.S., Jul. 2016, pp. 779–788.

[41] J. Redmon and A. Farhadi, YOLO9000: better, faster, stronger, In: *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, U.S., Jul. 2017, pp. 6517–6525.

[42] J. Redmon and A. Farhadi, YOLOv3: An incremental improvement, In: *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, U.S., Jul. 2017, pp. 1–6.

[43] 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*, vol. 39, pp. 1137–1149, 2017.

[44] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, In: *Proceedings of the 3th International Conference on Learning Representations (ICLR)*, San Diego, U.S., May. 2015, pp. 1–14.

[45] B. Singh, M. Najibi and L. Davis, Sniper: efficient multi-scale training, In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS)*, Montreal, Canada, Dec. 2018, pp. 9333–9343.

[46] C. Termritthikun, Y. Jamtsho, J. Ieamsaardb, P. Muneesawang and I. Lee, EEEA-Net: an early exit evolutionary neural architecture search, *Engineering Applications of Artificial Intelligence*, vol. 104, in press, DOI: 10.1016/j.engappai.2021.104397.

[47] Z. Tian, C. Shen, H. Chen and T. He, FCOS: fully convolutional one-stage object detection, In: *Proceedings of the 17th IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea, Oct. 2019, pp. 9626–9635.

[48] X. Wang, R. Girshick, A. Gupta and K. He, Non-local neural networks, In: *Proceedings of the 31th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, U.S., Jun. 2018, pp. 7794–7803.

[49] S. Xie, R. Girshick, P. Dollar, Z. Tu and K. He, Aggregated residual transformations for deep neural networks, In: *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, U.S., Jul. 2017, pp. 5987–5995.

[50] M. Yang, K. Yu, C. Zhang, Z. Li and K. Yang, DenseASPP for semantic segmentation in street scenes, In: *Proceedings of the 31th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, U.S., Jun. 2018, pp. 3684–3692.

[51] W. Yue, Z. Wang, J. Zhang and X. Liu, An overview of recommendation techniques and their applications in healthcare, *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 4, pp. 701–717, 2021.

[52] H. Zhang, M. Cisse, Y. Dauphin and D. Lopez-Paz, Mixup: beyond empirical risk minimization, *arXiv preprint:1710.09412*, 2018.

[53] S. Zhang, L. Wen, X. Bian, Z. Lei and S. Li, Single-shot refinement neural network for object detection, In: *Proceedings of the 31th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, U.S., Jun. 2018, pp. 4203–4212.

[54] X. Zhu, H. Hu, S. Lin and J. Dai, Deformable ConvNets v2: more deformable, better results, In: *Proceedings of the 32th IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, U.S., Jun. 2019, pp. 9300–9308.

[55] X. Zhu, W. Su, L. Lu, B. Li, X. Wang and J. Dai, Deformable DETR: deformable transformers for end-to-end object detection, In: *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, Vienna, Austria, May. 2021, Oral.

[56] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu and H. Ling, Detection and tracking meet drones challenge, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, in press, DOI:10.1109/TPAMI.2021.3119563.

[57] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye and D. Ren, Distance-IoU loss: faster and better Learning for bounding box regression, In: *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, U.S., Feb. 2020, pp. 12993–13000.

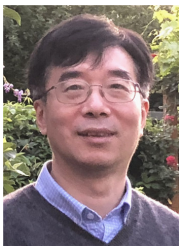
[58] L. Zou, Z. Wang, J. Hu, Y. Liu and X. Liu, Communication-protocol-based analysis and synthesis of networked systems: Progress, prospects and challenges, *International Journal of Systems Science*, vol. 52, no. 14, pp. 3013–3034, 2021.

Nianyin Zeng was born in Fujian Province, China, in 1986. He received the B.Eng. degree in electrical engineering and automation in 2008 and the Ph. D. degree in electrical engineering in 2013, both from Fuzhou University. From October 2012 to March 2013, he was a RA in the Department of Electrical and Electronic Engineering, the University of Hong Kong. From September 2017 to August 2018, he as an ISEF Fellow founded by the Korea Foundation for Advance Studies and also a Visiting Professor at the Korea Advance Institute of Science and Technology.

Currently, he is an Associate Professor with the Department of Instrumental & Electrical Engineering of Xiamen University. His current research interests include intelligent data analysis, computational intelligent, time-series modeling and applications. He is the author or co-author of several technical papers and also a very active reviewer for many international journals and conferences.

Dr. Zeng is currently serving as Associate Editors for Neurocomputing, Evolutionary Intelligence, and Frontiers in Medical Technology, and also Editorial Board members for Computers in Biology and Medicine, Biomedical Engineering Online, and Mathematical Problems in Engineering.

Peishu Wu received the bachelor degree in Measurement and Control Technology and Instrumentation from Tianjin University of Science and Technology, Tianjin, China, in 2020. He is currently pursuing the master degree in Measuring and Testing Technologies and Instruments at Xiamen University, Xiamen, China. His research interests include computer vision and deep learning techniques.



Zidong Wang (SM'03-F'14) was born in Jiangsu, China, in 1966. He received the B.Sc. degree in mathematics in 1986 from Suzhou University, Suzhou, China, and the M.Sc. degree in applied mathematics in 1990 and the Ph.D. degree in electrical engineering in 1994, both from Nanjing University of Science and Technology, Nanjing, China.

He is currently Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, U.K. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the UK. Prof. Wang's research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published more than 600 papers in international journals. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for *International Journal of Systems Science*, the Editor-in-Chief for *Neurocomputing*, the Editor-in-Chief for *Systems Science & Control Engineering*, and an Associate Editor for 12 international journals including IEEE Transactions on Automatic Control, IEEE Transactions on Control Systems Technology, IEEE Transactions on Neural Networks, IEEE Transactions on Signal Processing, and IEEE Transactions on Systems, Man, and Cybernetics-Part C. He is a Member of the Academia Europaea, a Member of the European Academy of Sciences and Arts, an Academician of the International Academy for Systems and Cybernetic Sciences, a Fellow of the IEEE, a Fellow of the Royal Statistical Society and a member of program committee for many international conferences.

Han Li received the bachelor degree in Measurement and Control Technology and Instrumentation from Xiamen University, Xiamen, China, in 2018. He is currently working towards the Ph. D. degree in Measuring and Testing Technologies and Instruments at Xiamen University, Xiamen, China. His research interests include intelligent optimization algorithms and deep learning techniques.

Weibo Liu received the B.S. degree in electrical engineering from the Department of Electrical Engineering & Electronics, University of Liverpool, Liverpool, UK, in 2015, and the Ph.D. degree in computer science from Brunel University London, Uxbridge, UK, in 2019. He is currently a Lecturer with the Department of Computer Science at Brunel University London, Uxbridge, UK. His research interests include big data analysis and deep learning techniques.

Xiaohui Liu received the B.Eng. Degree in Computing from Hohai University, Nanjing, China, in 1982 and the Ph.D. degree in Computer Science from Heriot-Watt University, Edinburgh, UK, in 1988. He is currently a Professor of Computing at Brunel University London, Uxbridge, UK, where he conducts research in artificial intelligence and intelligent data analysis, with applications in diverse areas including biomedicine and engineering.