Artificial Intelligence Review (2023) 56:10651–10677
https://doi.org/10.1007/s10462-023-10438-y

An efficient lightweight convolutional neural network for industrial surface defect detection

Dehua Zhang[1] · Xinyuan Hao[1] · Dechen Wang[1] · Chunbin Qin[1] · Bo Zhao[2] · Linlin Liang[3] · Wei Liu[4]

Accepted: 10 February 2023 / Published online: 1 March 2023

Abstract
Since surface defect detection is significant to ensure the utility, integrality, and security of productions, and it has become a key issue to control the quality of industrial products, which arouses interests of researchers. However, deploying deep convolutional neural networks (DCNNs) on embedded devices is very difficult due to limited storage space and computational resources. In this paper, an efficient lightweight convolutional neural network (CNN) model is designed for surface defect detection of industrial productions in the perspective of image processing via deep learning. By combining the inverse residual architecture with coordinate attention (CA) mechanism, a coordinate attention mobile (CAM) backbone network is constructed for feature extraction. Then, in order to solve the small object detection problem, the multi-scale strategy is developed by introducing the CA into the cross-layer information flow to improve the quality of feature extraction and augment the representation ability on multi-scale features. Hereafter, the multi-scale feature is integrated to design a novel bidirectional weighted feature pyramid network (BWFPN) to improve the model detection accuracy without increasing much computational burden. From the comparative experimental results on open source datasets, the effectiveness of the developed lightweight CNN is evaluated, and the detection accuracy attains on par with the state-of-the-art (SOTA) model with less parameters and calculation.

Keywords Lightweight convolutional neural networks · Surface defect detection · Attention mechanism · Feature pyramid networks

1 Introduction

Surface defects have significant adverse effects on the quality of industrial products. Therefore, surface defect detection has attracted more and more attention in recent years and made positive progress for quality control in industrial applications (Chen et al. 2021).

Xinyuan Hao, Dechen Wang, Chunbin Qin, Bo Zhao, Linlin Liang and Wei Liu have contributed equally to this work.

✉ Bo Zhao
zhaobo@bnu.edu.cn

Extended author information available on the last page of the article

However, the detection of surface defects is easily influenced by many environmental factors such as illumination, background and materials. These factors significantly increase the difficulty of surface defect detection. Furthermore, defects are complex in the natural environment, such as crazing, inclusion, patches, pit surface, rolled in scales and scratches.

As we all know, small object detection is one of the most challenging tasks in object detection (Chen et al. 2022; Song et al. 2022). And there are often slight defects in industrial products, which make detection difficult (Yang et al. 2020). The inability to accurately identify slight defects will directly affect the quality of production control in the intelligent manufacturing industry. Traditionally, most surface defect inspection tasks in the manufacturing industry are still performed manually. Unfortunately, manual defect inspection has the disadvantages of subjective instability and time-consuming. Compared with manual detection, automatic defect detection technology has obvious advantages. It can not only adapt to inappropriate environments, but also work for a long time with high precision and high efficiency. Studying defect detection technology can reduce production costs, improve production efficiency and product quality, and lay a solid foundation for the intelligent transformation of the manufacturing industry (Chen et al. 2021). With the rapid development of image processing technology in recent years, it is predictable that the manual work involved in inspection will be replaced by automation, which can not only reduce the labor cost, but also improve efficiency (Yan et al. 2020).

The main idea of traditional image processing technology is to describe surface defects by carefully designed hand-crafted features. The commonly used hand-crafted features containing histogram of the oriented gradient (HOG) (Song et al. 2013), gray level co-occurrence matrix (GLCM) (Huang and Lee 2009), local binary patterns (LBP) (Liu et al. 2016a), and other statistical features. However, these image processing methods can not be directly applied in reality, since the defect recognition usually requires complex threshold settings and is sensitive to environmental factors such as lighting conditions and background.

The traditional machine learning vision model depends on the specific vision inspection task obtained by manually analyzing and extracting defect features, and then makes decisions by using rule-based experience or learning-based classifier. Support vector machine (SVM) and decision tree all use this method (Ghorai et al. 2013), in which the system performance largely depends on the accurate representation of specific feature types. However, modern manufacturing technology requires a more robust product line, which limits the use of traditional machine learning models. For complex surface defect features, the traditional machine learning models need a long development cycle to adapt to different tasks.

In recent years, deep learning-based image processing methods have been extensively studied,which can automatically extract features from images and achieve good performance in image-related tasks such as classification, segmentation, and localization. With the rapid development of deep learning and the great success of convolutional neural networks (CNNs) in feature extraction (Wang et al. 2019), the object detection algorithm of DCNNs has the advantages of real-time, robustness and generalization, which shows great potential in automatic detection of surface defects (Cha et al. 2018; Zheng et al. 2021b). However, the existing algorithm model still has much room for improvement in surface defect detection. For example, the classical two-stage algorithm Faster R-CNN (Ren et al. 2017) has parameter redundancy, poor robustness and low efficiency. The classical single-stage algorithms SSD (Liu et al. 2016b) and YOLOv3 (Redmon and Farhadi 2018) have greatly improved their efficiency, but its robustness and accuracy are still not satisfactory. The SOTA object detection algorithms YOLOX (Zheng et al. 2021a) and EfficientDet (Tan

et al. 2020) gives consideration to both efficiency and accuracy, but its emphasis on universality leads to insufficient robustness in specific detection tasks. It is worth emphasizing here that the classic object detector has a large number of missed detections, and the detection precision is low, while the SOTA object detection models pay more attention to universality, and most of the model parameters and calculations are too high.

Traditional CNNs usually need a large number of parameters and floating point operations (FLOPs) to achieve a satisfactory accuracy, More importantly, *deploying DCNNs on embedded devices is very difficult due to limited storage space and computational resources*. For defect detection models, fewer parameters and less computation are more important. Generally, building a more complex network with more complex layers allows the network to learn good features from the input data. That is, a model with the more total number of parameters and computation will perform better in terms of accuracy. Therefore, this paper is devoted to developing a defect detection model with less model Params and FLOPs, while having good performance.

At present, some lightweight CNNs have been successfully applied in various aspects (Hui et al. 2018; Zhao et al. 2020; Shen et al. 2022), but there are few studies in industrial defect detection. In this study, an efficient lightweight CNN for detecting surface defects has been developed for the defect detection field. It provides a novel lightweight CNN model, which can locate and show surface defects from video streams in real time. The main contributions of this study are summarized as follows,

(1) A lightweight CNN model based on combination of multi-scale detection and attention mechanisms is designed for real-time, complex and real industrial scenarios. Two open source and challenging dataset are chosen to verify the effectiveness of the proposed method, and the results prove that the framework outperforms other classical models in terms of accuracy and attains performance on par with SOTA models in such tasks, with less model parameters and computational overhead.
(2) A CAM backbone network based on the combination of inverse residual block (IRB) and CA mechanism is proposed for preliminary feature extraction. The CAM backbone network is able to extract features efficiently with a small number of parameters and computational overhead through residual connectivity and depth-separable convolution techniques.
(3) The multi-scale strategy is proposed for the small object (defect) detection problem, which can highly improve the accuracy and robustness of the proposed model. Extraction of feature quantity affects straightway the distinguish ability ratio of disfigurement detection system. As feature maps at lower layer have more detailed information with higher resolution, while the deeper layer outputs have more semantic meanings with smaller resolution, cross-layer feature fusion can utilize rich semantic features and spatial features of images which can provide more information and make better predictions from different scales aiming at small object detection problem.
(4) A novel BWFPN is then designed to effectively represent and process multi-scale features which can fuse more features but without increasing too much cost. The designed network ties together depthwise separable convolution, cross-scale connection, and weighted feature fusion approach in a particularly way that makes it powerful in effectively reducing the redundancy of parameters and improving the efficiency of feature fusion. In addition, the CA is applied in the cross-layer information flow of BWPFN, which can augment the representations of the objects of interest and can make better use of multi-scale feature information at a small computational overhead.

The rest of the paper is organized as follows. In Sect. 2, the paper introduces the relevant work for surface defect detection. Next, the proposed network architecture for surface defect detection is designed in details in Sect. 3. Then, experimental analysis is presented in Sect. 4, and conclusions are summarized in Sect. 5.

2 Related work

In this section, the relevant techniques of surface defect detection based on image processing is studied, including traditional detection methods and deep learning-based detection methods, which are discussed in detail.

2.1 Traditional detection methods

Traditional detection approaches mainly include image processing based on hand-crafted features and shallow machine learning technology. Traditional image processing techniques that use hand-crafted features to extract, describe and detect defects can be divided into four categories: conventional statistical, spectral, model-based and machine learning methods (Luo et al. 2020). In detail, statistical approaches realize defect detection by evaluating the regular and periodic distribution of pixel intensities, mainly including local binary pattern (LBP) operator (Ojala et al. 1996), threshold-based methods (Amir et al. 2022), gray-level statistic methods (Hasan et al. 2016) and edge-based detection methods (Wu and Li 2021). Despite extensive research by early researchers on conventional statistical methods, many methods fail to provide reliable and correct results for defects with illumination varies or slight intensity transitions. The spectral methods find better solutions in the transform domain that are less sensitive to noise and intensity changes than the direct processing method in the pixel domain. The spectral methods mainly contain fourier transform (Aiger and Talbot 2010), gabor filters (Xie et al. 2010), wavelet transform (Ghorai et al. 2013) and optimized finite impulse response (FIR) filter (Kumar and Pang 2002). The spectral-based detection methods separate the defect objects from the background by finding a special transform domain, however, these methods methods lack local information and have bottlenecks on representing miscellaneous defects. The model-based method can better detect various defects by projecting the original texture distribution of the image block onto the low-dimensional distribution through the structural special model enhanced by parameter learning. Some classical models for surface defect detection include markov random field model (Cross and Jain 1983), weibull model (Wu et al. 2018) and active contour model (Wang et al. 2017). As a powerful branch of model-based method, machine learning has been widely proposed for surface defect detection. These methods treat the defect detection task as a binary (defective or non-defective) classification problem, such as support vector machine (SVM) (Li et al. 2022), decision trees (Alex et al. 2022) and shallow neural networks (Tan et al. 2015).

2.2 Deep learning-based detection methods

CNN has been widely used in surface defect detection in recent years because of its excellent ability of feature extraction and direct processing of two dimensions images (Luo et al. 2021; Li et al. 2021; Xie et al. 2021). Cha et al. proposed a deep convolution neural network (DCNN) method for detecting surface cracks of concrete and steel without manually

designing defect features (Cha et al. 2017). The framework can withstand to some extent the disturbances caused by the widely changing real environment. Then, this team also designed an automatic visual inspection method based on faster region-based CNN (Faster R-CNN) to ensure that multiple types of defects can be detected simultaneously in quasi real time (Cha et al. 2018). Lin et al. detected the surface defects of hot rolled strip by modifying the backbone network VGG16 of R-CNN to ResNet50 to generate feature map, and the experimental results show that the detection method based on deep learning was more effective than traditional methods and could detect the surface defects of strip more accurately (Lin et al. 2019). What's more, Song et al. combined DCNN with skeleton extraction to realize accurate detection of weak scratches on metal surface, and the experimental results show that it is robust to background noise (Song et al. 2019). Li et al. improved YOLO network with the idea of complete convolution, and then used YOLO variant to detect the surface defects of flat steel, and the accuracy of the network reached 99% (Li et al. 2018). However, most of the above methods only consider the classification accuracy but ignore the extraction and location of defect position. More importantly, most DCNNs usually need a large number of parameters and FLOPs to achieve satisfactory detection accuracy. Due to limited storage space and computing resources, it is very difficult to deploy DCNNs on embedded devices.

At present, some researchers have carried out research on lightweight CNN in applied research. For example, Hui et al. proposed a lightweight CNN for optical flow estimation with a model size only one-thirtieth of the SOTA models for optical flow estimation. The results show that the proposed lightweight model attains performs on par with the large SOTA model (Hui et al. 2018). Zhao et al. proposed a lightweight emotion recognition (LER) model to handle the latency problem under natural conditions. Compared with the VGG13, the LER model achieves higher accuracy and reduces the number of parameters by 97 times (Zhao et al. 2020). Pan et al. proposed a lightweight network called MCNA for gas identification, which combines a multi-scale deep convolutional network with a self-attention mechanism. The MCNA requires much fewer parameters and computation costs than previous deep learning networks, but it still achieves the same high gas identification accuracy (Pan et al. 2022).

However, the research on practical lightweight CNN model in the field of defect detection is still insufficient. Compared with current detection schemes, our schemes have lower computational overhead but higher efficiency and precision. Now our schemes are designed in details, and the corresponding properties of the relevant technologies adopted are analyzed in the next section.

3 Proposed surface defect detection network

The small object detection problem especially surface defect detection is one of the most challenging technical problems in object detection area in real-world applications. As we mentioned above in the first section, the industrial surface defects have complex features, therefore it is very difficult to detect surface defects efficiently and accurately. So, a novel efficient surface defect detection network is developed in this section, including the proposed CAM backbone network based on IRB and CA, a detection strategy combining multi-scale and attention mechanism and a novel BWFPN. The complete network architecture is shown in Fig. 1.

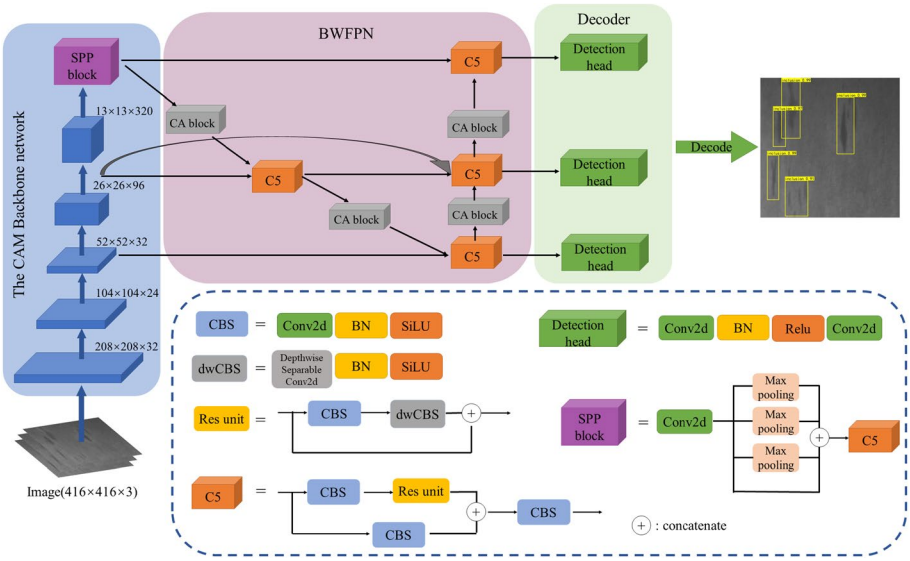


Fig. 1 The architecture of proposed network

3.1 Backbone network

3.1.1 CA mechanism

Although a novel feature fusion algorithm is established in part (Sect. 3.3), not all the feature information of the target carry important meaning. Based on the inspiration of animal visual attention, researchers found that most of the information in the image is useless, and only selecting the relevant parts for calculation, which is named attention mechanism, can make the calculation process more efficient (Wang et al. 2021a; Li et al. 2020). The main goal of attention mechanism is to select the information that is more critical to the task goal from plenty of information.

CA is a lightweight attention mechanism, which was proposed by (Hou et al. 2021). CA can be regarded as a computing unit, and its purpose is to enhance the expressive ability of the learned features in the network. Inspired by this, in our method, the CA block is introduced to proposed network to augment the representations of the objects of interest. The CA block first aggregates vertically (Y) and horizontally (X) input features into two separate direction-aware feature maps using two one-dimensional global pooling operations, and then encodes the two feature maps with specific direction information into two attention maps respectively, and each attention map captures the long-range dependencies of the input feature map along a spatial direction. The location information can be saved in the generated attention map, and two attention map are applied to input feature maps by multiplication to emphasize the representation of attention regions. Its implementation can be found in Fig. 2, where avg pool stands for average pooling, Conv2d (1×1) represents a convolution operation with a convolution kernel size of 1, BN stands for batch normalization, the hard-swish (Hswish) (Howard et al. 2019) and Sigmoid activation functions are represented as follows,

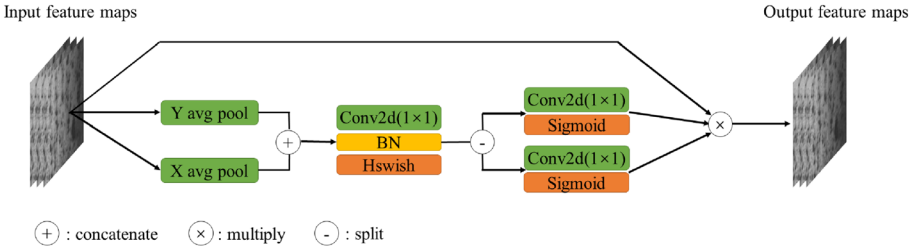


Fig. 2 The CA block used in proposed network

$$Hswish(x) = x \cdot \frac{Relu6(x+3)}{6}, \tag{1}$$

$$Sigmoid(x) = \frac{1}{1+e^{-x}}, \tag{2}$$

and the Relu6 function will be described in Sect. 3.1.2. Although the Sigmoid activation function can improve detection accuracy, it has an exponential operation, which results in more computational costs. This is even more significant on mobile devices. The introduction of the Hswish activation function can effectively reduce the computational overhead of the model.

3.1.2 The CAM backbone network

Compared with DCNNs, the models of MobileNet series are excellent lightweight backbone network with smaller volume and less computation. The accuracy of MobileNet network has surpassed that of some DCNNs with the optimization of network architecture. MobileNet series has become the most popular object detection methods in practical applications with its high efficiency and good detection accuracy (Wang et al. 2021b).

Depthwise separable convolutions (Howard et al. 2017) are a key building block for the efficient MobileNet architectures. The basic idea is to replace convolutional operator with two separate layers, the first layer is depthwise convolution, and the second layer is point-wise convolution. Depthwise convolution applying a single convolutional filter per input channel, and pointwise convolution is a 1×1 convolution, which is responsible for constructing new features by calculating the linear combination of input channels. Conventional convolution takes a $P_{in} \times H \times W$ input feature map, and applies convolutional kernel K to produce a $P_{out} \times H \times W$ output feature map. It has the computational cost of $P_{in} \times K \times K \times P_{out}$. In depthwise separable convolution, the computational cost equals: $P_{in} \times K \times K + P_{out} \times 1 \times 1 \times P_{in}$, which is the sum of depthwise and pointwise convolution. According to experience, depthwise separable convolution is almost as effective as conventional convolution in most cases.

The inverse residual architecture was first proposed by (Sandler et al. 2018), which are constructed by the idea of bottleneck (expansion–convolution–compression). Unlike residual block, the design of inverse residual architecture is considerably more memory efficient, and works slightly better. The idea of the inverse residual architecture is to use convolution to extract features after expanding the channels, and then compress the channels.

Inspired by the above, in this paper, we propose the CAM backbone network by combining CA in the inverse residual architecture. Specifically, the inverse residual architecture

we adopt first use 1×1 conventional convolution operation to expand the channel, batch normalizes (BN) the resulting feature maps and activates them using the Relu6 function. The Relu6 function can be expressed as follows,

$$Relu6(x) = min(6, max(0, x)). \tag{3}$$

The Relu6 activation function is chosen because it has a lower computational complexity than the Relu function, and it can avoid the problem that the weight range caused by the Relu function is too large when the mobile network performs weight quantization. Secondly, we adopt group convolution (convolution-BN-activate) to extract features one by one on the dilated channels to generate feature maps. Then, these feature maps will be input into the CA block to enhance the effective representation of the region of interest. Finally, we use point-wise convolution (convolution-BN) to compress the channels. A more visual representation can be found in Fig. 3. The detailed configuration parameters of the CAM backbone network can be found in Sect. 4.2.1.

3.1.3 SPP module

Spatial pyramid pooling (SPP) was first proposed by He et al. (2015) for converting feature map tensor into fixed length feature vectors. CNN takes images of arbitrary size in a sliding window fashion, but fully connected layers can only accept fixed size inputs. SPP divides the feature map into a fixed number of local spatial cells and pools all the elements within each cell to enable the CNN model to receive input images of different sizes without cropping or resizing the images.

The SSP module also helps with object detection as it can extract multi-scale features with different receptive fields from the same convolutional layer. The SPP module concatenates the input feature maps at four different scales, 1×1, 5×5, 9×9 and 13×13 with stride 1. Multi-scale feature maps pooled from the same layer are concatenated to fuse multi-scale local region features for object detection. Therefore, we insert the SPP module after the final layers of the CAM backbone network. This helps to improve detection accuracy with less computational overhead.

3.2 Multi-scale detection strategy

Small object detection is one of the most challenging tasks in object detection (Chen et al. 2022; Song et al. 2022). Because there are many small defects in the surface defects of

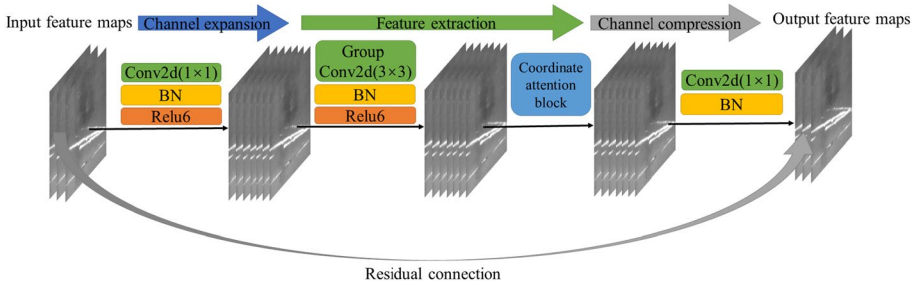


Fig. 3 The proposed inverse residual block

industrial products, enhancing the detection accuracy of small objects is a problem that the defect detection network must deal with.

The backbone of CNN extracts semantic information by increasing the network layers while gradually reducing the spatial scales of the network. Generally, the earlier layers with higher spatial resolution in the backbone network have less semantic information and more fine-grained information, while the deeper layers with smaller spatial resolution have more semantic information. Cross-layer feature fusion integrates fine-grained information from earlier layers and more meaningful semantic features from deeper layers, which helps to detect small objects (Lin et al. 2017). Meanwhile, because the size of the input images has obvious influence on the performance of the detection model, and the robustness of the detection model to the size of the object can be improved to a certain extent by training by inputting images of different sizes. Therefore, multi-scale strategy is one of the best techniques to improve the precision of the model.

Given the above rationale, the multi-scale strategy is designed for the small object detection problem based on the proposed CAM backbone network which can highly improve the accuracy and robustness of the proposed model. Feature maps at lower layer have more detailed information with higher resolution, while the deeper layer outputs have more semantic meanings with smaller resolution. Cross-layer feature fusion utilizes rich semantic features and spatial features of images. In specific, we choose three output feature layers of IRBs with output channels of 32, 96 and 320 in CAM backbone network as the input of multi-scale detection. These multi-scale features span a wider range of CNN layers and provide more information, so as to better predict objects of different scales.

3.3 The BWFPN

How to effectively represent and process multi-scale features has become one of the main difficulties in object detection after applying the multi-scale strategy. As one of the pioneering works, feature pyramid network (FPN) (Lin et al. 2017) proposes a top-down pathway to combine multi-scale features. Following this idea, path aggregation network (PAN) (Liu et al. 2018) adds an extra bottom-up path aggregation network on top of FPN, and achieves the best result in various proposed feature pyramids. Bidirectional feature pyramid network (BiFPN) (Tan et al. 2020) proposes efficient bidirectional cross-scale connection and weighted feature fusion to optimize PAN, so that it can fuse features more efficiently.

Multi-scale feature fusion aims to aggregate features at different backbone layers. Formally, given a list of multi-scale features $\vec{P^{in}} = (P_1^{in}, P_2^{in}, \ldots, P_l^{in})$, where P_l^{in} represents the input feature of the l-th level, and l represents a feature level with resolution of $1/2^l$ of the input images. For instance, if the input resolution is 416×416, then P_3^{in} represents feature level 3 with resolution 52×52 ($416/2^3 = 52$), while P_5^{in} represents feature level 5 with resolution 13×13. The goal of a feature pyramid is to find a transformation f that can effectively aggregate features at different layers and output a list of new features, that is $\vec{P^{out}} = f(\vec{P^{in}})$.

FPN introduces a top-down pathway to fuse multi-scale features, which effectively improves the detection precision and the detection effect of small objects, as shown in Fig. 4a. But the conventional top-down pathway is inherently limited by the single-pass information flow. PAN adds an extra bottom-up path aggregation network after FPN to address this issue, as shown in Fig. 4b. BiFPN further optimizes PAN with the idea of cross-scale connection: First, remove the node with only one input edge, because if a node has only one input edge, its contribution to the feature network aiming at fusing different

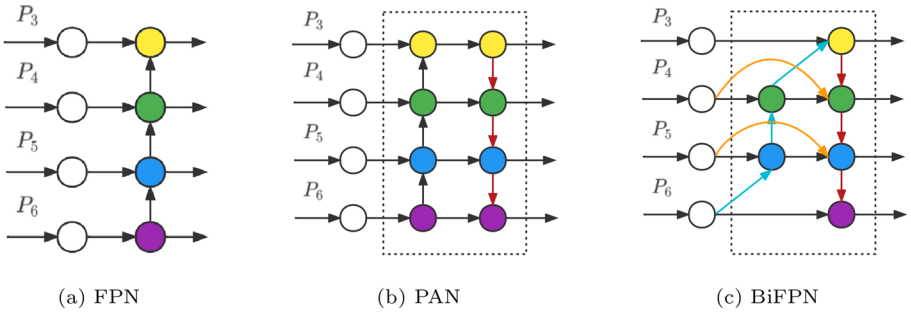


Fig. 4 Design of different feature pyramid structures. **a** FPN fuses multi-scale features by introduce a top-down pathway; **b** PAN adds an additional bottom-up pathway on top of FPN; **c** BiFPN removes nodes that only have one input edge and adds an extra shortcut connection from the input node to output node if they are at the same level

features will be small. Second, an extra edge is added from the input node and the output node when the input node and the output node are at the same level, so as to fuse more features without increasing too much cost. Third, each bidirectional path is regarded as a feature network layer, and this layer can be repeated many times to achieve higher-level feature fusion. Figure 4c shows the specific structure of BiFPN.

The contribution of different input features to output features in different resolutions is usually unequal. BiFPN lets the network to learn the importance of each input feature by adding an extra weight to each input. Inspired by this, we adopt the fast normalized fusion method for weighted fusion, which is defined as follows,

$$F = \sum_i \frac{w_i}{\eta + \sum_j w_j} \cdot X_i, \tag{4}$$

where w_i and w_j are learnable weights, η is a small fixed value to avoid numerical instability, X indicates each input feature, and F indicates fused weighted feature. For instance, in Fig. 4c, two fused features at level 4 can be described as follows,

$$P_5^{td} = C\left(\frac{w_1 \cdot P_5^{in} + w_2 \cdot R(P_6^{in})}{w_1 + w_2 + \eta}\right), \tag{5}$$

$$P_5^{out} = C\left(\frac{w_1' \cdot P_5^{in} + w_2' \cdot P_5^{td} + w_3' \cdot R(P_4^{out})}{w_1' + w_2' + w_3' + \eta}\right), \tag{6}$$

where P_l^{td} is the intermediate feature at level l, and P_l^{out} is the output feature at level l. R usually represents upsampling or downsampling operation, and C is usually a convolutional op for feature processing.

In summary, inspired by BiFPN and (Bochkovskiy et al. 2020), a novel BWFPN is then designed to effectively represent and process multi-scale features which can fuse more features but without increasing too much cost. The designed network ties together depthwise separable convolution, residual connection, cross-scale connection, attention mechanism and weighted feature fusion approach in a particularly way that makes it powerful in effectively reducing the redundancy of parameters and improving the

efficiency of feature fusion. The nodes of the mentioned BWFPN are constructed by a module called C5, which basic building block is the CBS (convolution, BN and SiLU activation function) unit and residual unit. The SiLU function has no upper bound, a lower bound, is smooth and non-monotonic. Although its operation is larger compared to the Relu function, we choose the SiLU function as the activation function in order to improve the effectiveness of BWFPN. The representation of the SiLU function is as follows,

$$SiLU(x) = x \cdot Sigmoid(x). \tag{7}$$

The residual unit inputs the input feature maps into a CBS unit and a dwCBS (depth-wise separable convolution, BN and SiLU activation function) unit, and concatenates the obtained result with the initial input feature maps.

In addition, we input the results of each upsampling or downsampling in BWFPN into the CA block to augment the representations of the objects of interest. The introduction of attention mechanisms into the multi-scale information flow is a effective way of processing multi-scale features. A simple framework representation of the proposed BWFPN can be found in Fig. 5, and its concrete implementation can be found in Fig. 1.

4 Experiments and results

In this section, experiments are conducted to demonstrate the effectiveness of the developed surface defect detection method, and the results and analysis are presented. Firstly, two kinds of open source and challenging datasets are selected and evaluation indicators are introduced to evaluate the performance of this method. Secondly, the ablation experiments of the improved method are carried out to verify the effectiveness of the proposed method. Finally, 9 other classical models and SOTA models are compared, and the experimental results and analysis are given.

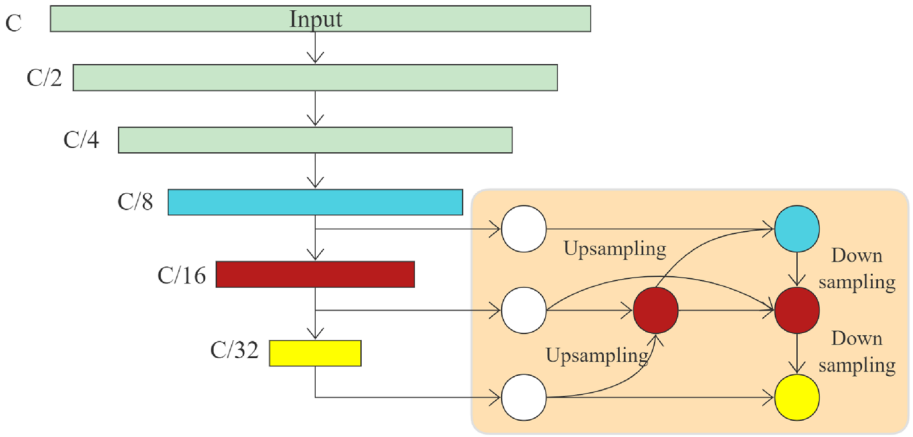


Fig. 5 The architecture of the proposed novel BWFPN

4.1 Dataset and evaluation metrics

4.1.1 Dataset preparation

In the experiment, we chose two open source and challenging datasets: NEU-DET dataset provided by Northeastern University and PCB defect dataset provided by HRI laboratory of Peking University, to verify the effectiveness of proposed method. The NEU-DET dataset contains 6 types of steel surface defects, including crazing (Cr), inclusion (In), patches (Pa), pitted surface (Ps), rolled-in scales (Rs), and scratches (Sc). Each defect has 300 images of 200×200 pixels, and each image contains one type of defects mentioned above. The PCB dataset contains 690 images with 6 types of defects, including missing hole (MH), mouse bite (MB), open circuit (OC), short (SH), spur (SP), spurious copper (SC). Each defect has 115 images with pixels ranging from 3034×2464 to 2904×1521. Due to its high resolution and small defect region, the PCB dataset has high requirements on the small object detection ability of the model. The defects in the image are marked with a rectangular frame called ground truth box. Additionally, the coordinate information of the ground truth box is recorded in the annotation files. All the datasets contain more than 5000 ground truth boxes. Figure 6 presents the examples of defect images with ground truth bounding boxes in the NEU-DET dataset.

4.1.2 Evaluation metrics

There are several metrics commonly adopted to evaluate the performance of object detection methods. True Positives (TP) means positive predictions that match with the ground truth. False negative (FN) means that negative predictions that do not match with ground truth. False positive (FP) means positive predictions that do mot match with ground truth. The quantifiable indicators for precision (P) and recall (R) are defined as follows,

$$P = \frac{TP}{TP + FP}, \tag{8}$$

$$R = \frac{TP}{TP + FN}. \tag{9}$$

Precision is generally used to evaluate the global accuracy of the model, reflecting the proportion of true positive samples among the predicted positive samples determined by

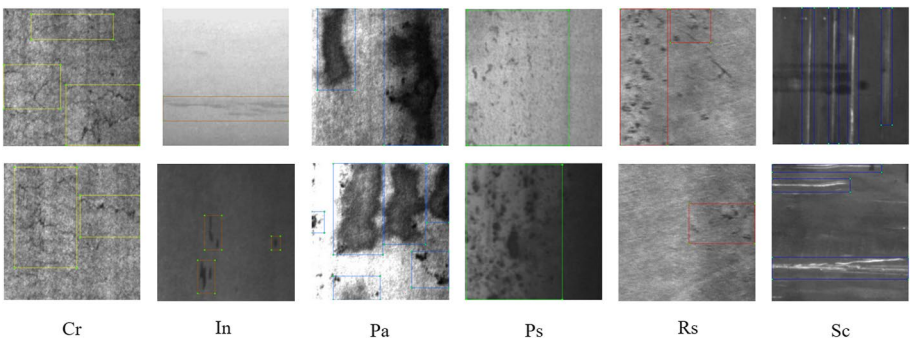


Fig. 6 Partial presentation of the NEU-DET dataset

the classifier. The recall rate reflects the proportion of true positive samples among the labelled positive samples.

It is not appropriate to use the above two metrics to evaluate the performance of the model, because when the accuracy of model A is higher than that of model B, its recall rate is likely to be lower than that of model B. That is, there is a certain contradiction between the P and the R. The precision-recall curves illustrate the trade off between precision and recall for different thresholds. Average precision (AP) stands for the area under the curve, which measures the detection performance for a single class. Mean average precision (mAP) is the global score as the mean of the average precision for each class.

In this paper, all mAP scores are under the condition that the intersection over union (IoU) threshold is greater than 0.5. The IoU can be used to measure quantify the overlapping level between prediction and ground truth bounding box, which is defined as follows,

$$IoU_{A,B} = \frac{|A \cap B|}{|A \cup B|}, \tag{10}$$

where $|*|$ represents the cardinality of the set. If IoU is above a certain threshold (such as 0.5 or 0.75), the prediction result is considered correct.

The number of total parameters (Params) and FLOPs in a model reflects the size and complexity of a model. For example, for a convolutional layer, its parameters can be calculated as follows,

$$Params_{conv} = C_{out} \times (K_h \times K_w \times C_{in} + 1), \tag{11}$$

where K_h and K_w represent the width and height of the convolution kernel, C_{out} and C_{in} represent number of input channels and number of output channels, respectively. The FLOPs of a convolutional layer can be calculated as follows,

$$FLOPs_{conv} = F_h \times F_w \times \left(C_{in} \times \frac{2k^2 - 1}{g} + 1 \right) \times C_{out}, \tag{12}$$

where F_h and F_w represent the width and height of the input feature map, k represents the size of convolution kernel, and g represents group size for convolution. For the object detection model, the unit of measurement of Params is generally megaParams (M), and the unit of FLOPs is generally gigaFLOPS (G).

It should be noted that there is no corresponding relationship between Params and FLOPS. A model with a large amount of Params may have a low amount of FLOPs, and a model with a small amount of Params may also have a high amount of FLOPs. For example, DenseNet is a model with fewer Params, but due to its densely connected structure, it has a large amount of FLOPs within a module.

4.2 Implementation of the proposed method

4.2.1 Experimental environment and configuration

All the experiments are conducted by a PC with Intel Core i9-11900k @3.50GHz central processing unit, NVIDIA RTX3090 graphic processing unit, and 32GB RAM under the windows system. Python 3.7 is used to compile the source files, and PyTorch 1.10.1 is adopted for the open source deep learning framework. The batch size is set to 64. We run experiments under windows platform. We use the pre-training weight from the VOC07+12

dataset for training to ensure uniformity of the initial weights of the model and accelerate model convergence. The initial learning rate is set to 0.001, and the minimum learning rate is 0.01 times the initial learning rate. Cosine annealing (Loshchilov and Hutter 2016) is adopted to adaptively control the change of learning rate, and the T_{max} is set to 5. The optimizer we chose is Adam, and the weight decay and the momentum is set to 0.0005 and 0.93 respectively. The images in the NEU-DET dataset are resized to 416×416 before they are input to the training networks, and the images in PCB dataset are resized to 640×640 due to their higher raw pixels. The NEU-DET dataset is randomly divided as training set: validation set: test set = 0.72:0.08:0.2, and the PCB dataset is randomly divided as training set: validation set: test set = 0.63:0.07:0.3.

Mosaic data augmentation (Bochkovskiy et al. 2020) is used to enrich the dataset. Mosaic data augmentation will randomly select four images in the dataset, then rotate, scale, transform the gamut of pictures, and place them in four directions. However, the images generated by this method are quite different from the images in the real scene, which will reduce the stability of the model (Zheng et al. 2021a). Therefore, we choose to use mosaic data augmentation in the first 70% epochs of training.

The label smoothing is proposed to solve the problems in network model such as weak generalization ability, over fitting and over believing in the category of prediction. Label smoothing convert hard label into soft label for training, which can make model more robust and prevent the model from over fitting to a certain extent. In this work, we use label smoothing ($\epsilon = 0.005$) to improve the performance of the model.

The loss function of the network we choose consists of regression loss, object loss and classification loss (Redmon and Farhadi 2018). Among them, the regression loss represents the error between the bounding box predicted by the network and the ground truth bounding box, and we choose to use GIoU (Generalized IoU) loss (Rezatofighi et al. 2019), and its proportion weight is set to 0.05. The object loss represents the confidence error, which refers to whether the bounding box contains the target object, and we choose to use the cross entropy loss. The proportion weight of the object loss is set as the square of the input size divided by the square of 416. The classification loss represents the error between the model's judgment of the class of the target object and its actual class, and we choose to use the cross-entropy loss. The proportion weight of the classification loss is set to 0.5× (number of classes to detect/80)

4.2.2 Parameter settings of backbone network

The detailed parameters of the proposed CAM backbone network are shown in Table 1, where C represents the number of output channels of this layer, S indicates the stride of convolution operation in IRB, N is the number of IRBs in this layer, E represents expansion factor of IRBs in this layer, and Att indicates whether to include a CA block in this layer, with 0 for no and 1 for yes.

4.2.3 Anchor boxes setting

Anchor box is a predefined bounding box with a certain aspect ratio, and is mainly used to improve the speed and accuracy in the object detection model. In our work, to learn the right number and size of anchor boxes from the dataset, We choose to set 3 anchor boxes for each detection at 3 scales. The 9 anchor boxes that are artificially set: (10, 13), (16,

Table 1 The parameters of the MobileNetv2 backbone network

Layer	Operator	*C*	*S*	*N*	*E*	*Att*
0	conv2d	32	1	–	–	–
1	IRB	16	1	1	1	1
2	IRB	24	2	2	4	0
3	IRB	32	2	2	2.5	1
4	IRB	64	2	1	2.5	0
5	IRB	96	1	2	2.5	1
6	IRB	160	2	2	2.5	0
7	IRB	320	1	1	2.5	1

Table 2 The performance of different backbones on the PCB dataset

Backbone	Neck	Params	FLOPs@640	mAP (%)
MobileNetv1	BWFPN	8.54 M	16.52 G	86.60
MobileNetv2	BWFPN	5.28 M	9.52 G	91.43
MobileNetv3	BWFPN	5.93 M	7.99 G	89.17
GhostNet	BWFPN	5.63 M	6.73 G	87.24
CAM	BWFPN	4.00 M	6.34 G	91.95

30) and (33, 23) for small objects, (30, 61), (62, 45) and (59, 119) for medium objects and (116, 90), (156, 198) and (373, 326) for large objects.

4.3 Ablation study and analysis

To demonstrate the contributions of different parts of the proposed network, ablation experiments are conducted. In this section, ablation studies on CAM backbone network, BWFPN and CA are successively reported.

4.3.1 Ablation study for CAM backbone network

In this experiment, we evaluate the effectiveness of CAM backbone network by replacing different lightweight backbones on the proposed framework: MobileNetv1 (Howard et al. 2017), MobileNetv2 (Sandler et al. 2018), MobileNetv3 (Howard et al. 2019) and GhostNet (Han et al. 2020). These networks we selected are excellent lightweight backbone networks for vision tasks. The Params and FLOPs of the network are calculated with an input size of 640×640 (@640) to reflect the complexity of the backbone. We conduct experiments on the PCB dataset, and the results are shown in Table 2. The Neck in the table refers to the part of the network used to process the multi-scale features extracted from the backbone network. As reported in Table 2, the CAM backbone network gets the highest mAP score on the PCB dataset among the four backbones, while the Params and FLOPs are lower than other four backbones. It can be concluded from Table 2 that CAM backbone

Table 3 Comparison of different feature pyramid network

	Backbone	Neck	Params	FLOPs@640	Dataset	mAP (%)
(a)	CAM	FPN	22.52 M	40.47 G	NEU	75.51
					PCB	82.89
(b)	CAM	PAN	36.17 M	58.36 G	NEU	76.70
					PCB	83.47
(c)	CAM	BWFPN	4.00 M	6.34 G	NEU	78.64
					PCB	91.95

network achieves the best performance with lower Params and FLOPs. The superiority of the CAM backbone network is thus proved.

4.3.2 Ablation study for BWFPN

BWFPN fuses and extracts the multi-scale features obtained from the backbone network, which is of great significance for improving the performance of the model and needs to be verified. The FPN, PAN and proposed BWFPN are used to test and verify on the two datasets respectively, and the results are shown in Table 3. Among them, the specific implementation of FPN is introduced in (Redmon and Farhadi 2018), and the specific implementation of PAN is introduced in (Bochkovskiy et al. 2020). The Params and FLOPs in the table reflect the amount of parameters and computation of the entire network. As reported in Table 3, the Params of (a) and (b) are much higher than those of (c), and the FLOPs of (a) and (b) is slightly higher than that of (c). (c) Achieves the best performance with the lowest Params and FLOPs on two datasets, which proves the validity of the proposed BWFPN.

4.3.3 Ablation study for CA

Attention mechanisms have important implications for improving the performance of networks with lighter backbones and need to be emphasized and validated. We specifically conducted the experiments to examine the effects of different attention mechanism on performance based on the PCB dataset. We choose three other lightweight classical attention mechanisms, including squeeze-and-excitation (SE) attention (Hu et al. 2020), efficient channel attention (ECA) (Wang et al. 2020) and convolutional block attention module (CBAM) (Woo et al. 2018). The four attention mechanisms are respectively adopted: (1) attention is used in the backbone, (2) attention is used in the neck, (3) attention is used in the backbone and neck, and the results are shown in the Table 4. As reported in Table 4, among all lightweight attention modules, the CA achieves the highest mAP score and has the smallest extra computational overhead. The ECA brings almost no extra model parameters, but its improvement in detection performance is minimal. It can be seen that adding attention block to the neck is usually more effective than adding attention block to the backbone. This is because the feature information in the cross-layer information flow generated by the multi-scale detection strategy is more

Table 4 Comparison of different attention mechanisms on the PCB dataset

Attention	Backbone	Neck	Params	FLOPs	mAP (%)
×			3.953 M	6.342 G	87.42
SE	✓		4.088 M	6.352 G	88.12
		✓	3.974 M	6.345 G	88.30
	✓	✓	4.108 M	6.355 G	89.37
CBAM	✓		4.018 M	6.375 G	89.26
		✓	3.995 M	6.347 G	89.20
	✓	✓	4.059 M	6.380 G	89.78
ECA	✓		3.953 M	6.351 G	88.28
		✓	3.953 M	6.345 G	88.43
	✓	✓	3.953 M	6.355 G	88.97
CA	✓		3.982 M	6.345 G	89.44
		✓	3.972 M	6.343 G	89.83
	✓	✓	4.000 M	6.347 G	91.95

important, which confirms the effectiveness of the proposed method combining multi-scale detection and attention mechanism.

4.4 Comparative experiments and analysis

This section conducts a series of comparative experiments between the proposed model and the SOTA model on the dataset mentioned in Sect. 4.1.1 to verify the effectiveness of the proposed model, and gives the corresponding results and analysis.

4.4.1 Experiments on the NEU-DET dataset

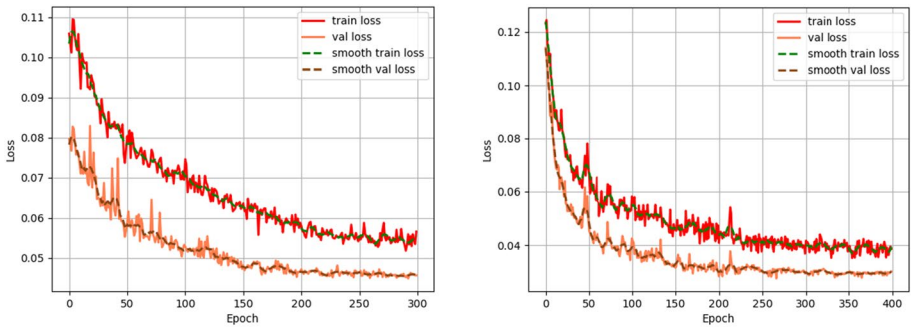
The training and testing are further carried out to verify the proposed network model, and Fig. 7a and b show the change curves of the proposed model training loss function values on the NEU-DET dataset and the PCB dataset, respectively. It can be seen that on the NEU-DET dataset, our model converges in 250 epochs, and on the PCB dataset, our model converges in 400 epochs, and the loss function value decreases according to the cosine fluctuation. It takes more epochs to complete the convergence on the PCB dataset because the input resolution of the PCB dataset is higher, and the detection of small objects is more difficult.

On the basis of verifying the validity of the proposed model, we further verify the advance of the proposed model through comparative experiments. Nine object detectors, i.e., SSD (Liu et al. 2016b), Faster R-CNN (Ren et al. 2017), CenterNet (Duan et al. 2019), RetinaNet (Lin et al. 2020), YOLOv3 (Redmon and Farhadi 2018), YOLOv4 (Bochkovskiy et al. 2020), YOLOv5, YOLOX (Zheng et al. 2021a) and EfficientDet (Tan et al. 2020) are chosen, and the corresponding comparative test results are shown in Table 5.

In all models, the FLOPs except EfficientDet are calculated with an input size of 640 × 640, because the EfficientDet model adopts a strategy which is used different input sizes according to the corresponding model in the detection, and its specific input size is represented in the table by @ symbol. As reported in Table 6, Overall, the detection accuracy of classical object detectors is lower, and the Params and FLOPs are higher. Most of the SOTA models have high Params and FLOPs, so they are not suitable for practical detection

Table 5 The comparison results of the proposed model and other models on the NEU-DET dataset

Network	Backbone	Params	FLOPs@640	mAP (%)
faster R-CNN	VGG	136.79 M	402.03 G	74.16
	ResNet50	28.32 M	946.53 G	75.56
CenterNet	ResNet50	32.66 M	109.33 G	74.78
RetinaNet	ResNet50	36.43 M	164.94 G	63.24
SSD300	VGG	24.28 M	275.60 G	75.32
SSD300-Lite	MobileNetv2	4.20 M	*6.19 G*	71.26
YOLOv3	DarkNet	61.55 M	155.16 G	76.71
YOLOv4	CSPDarkNet	63.96 M	141.52 G	77.28
YOLOv5-s	CSPDarkNet	7.07 M	16.42 G	78.47
YOLOv5-m	CSPDarkNet	21.07 M	50.46 G	**79.57**
YOLOv5-l	CSPDarkNet	46.65 M	114.32 G	79.07
YOLOv5-x	CSPDarkNet	87.27 M	217.43 G	78.18
YOLOX-s	CSPDarkNet	8.94 M	26.64 G	78.61
YOLOX-m	CSPDarkNet	25.28 M	73.51 G	76.45
YOLOX-l	CSPDarkNet	54.15 M	155.33 G	78.21
YOLOX-x	CSPDarkNet	99.00 M	281.53 G	77.39
EfficientDet-D0@512	EfficientNet-b0	*3.83 M*	*4.62 G*	66.37
EfficientDet-D1@640	EfficientNet-b1	6.55 M	11.23 G	67.18
EfficientDet-D2@768	EfficientNet-b2	8.01 M	20.15 G	69.20
EfficientDet-D3@896	EfficientNet-b3	11.91 M	46.19 G	70.28
EfficientDet-D4@1024	EfficientNet-b4	20.55 M	103.72 G	70.47
EfficientDet-D5@1152	EfficientNet-b5	33.43 M	257.52 G	71.28
Proposed method	CAM	**4.00 M**	**6.34 G**	*78.64*



(a) The training loss and validation loss curve changes on the NEU-DET dataset.

(b) The training loss and validation loss curve changes on the PCB dataset.

Fig. 7 The training loss and validation loss curve changes of the proposed model

applications due to the cost of detection hardware. In the SOTA models, only Params and FLOPs of EfficientDet-D0 are lower than that of the proposed model, but its input size is 512, and its mAP score is much lower than that of the proposed model. Due to the adoption of the lightweight backbone network MobileNetv2, SSD300-Lite has similar Params

and FLOPs to the proposed model, however, its mAP score is much lower than that of the proposed model. As the lightweight models of YOLOv5 and YOLOX, YOLOv5-s and YOLOX-s show the superiority of the SOTA model, but due to their emphasis on generality, the indicators in the industrial surface defect detection task are not as good as the proposed model. Although YOLOv5-m achieves the highest mAP score on the NEU-DET dataset, which is 0.8% higher than the proposed model, its Params are five times that of the proposed model, and the FLOPs are eight times that of the proposed model.

Notably, we observe that EfficientDet and RetinaNet have lower scratches class AP scores, only around 30% to 40%. According to the analysis, this is due to the use of the focal loss in both models. The gray-scale of the defective regions in class Cr, In, Pa, Ps and Rs pictures is lower than the background regions, while the gray-scale of the defective regions in class Sc pictures is higher than that of the background regions, and there is a lot of interference in the background. The focal loss was proposed to solve the problem of imbalance between positive (objects) and negative (background) samples. In the training process, the focal loss misjudged the positive and negative samples in class Sc, and adjusted its weights accordingly, which eventually led to low AP.

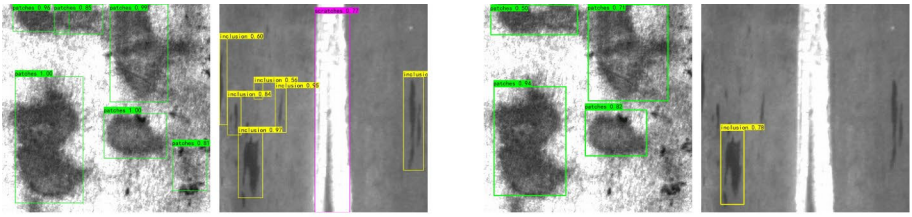
And we also randomly selected 2 detection results pictures of the above 9 models and compared them with the 2 detection results pictures of the model proposed in this paper. The results can be found in Fig. 8. The label on the anchor box shows the class and confidence that the model determines the object in this region. As shown in Fig. 8, the proposed model has almost the same detection effect as the SOTA model, but uses fewer parameters and less computation power.

4.4.2 Experiments on the PCB dataset

Since the defect images of the PCB dataset have high pixels and small defect region, all these have put forward higher demand for the small object detection capability of the model. Therefore, to further verify the performance of the proposed model to scale changes, the PCB dataset is employed to evaluate the usefulness of our proposed model.

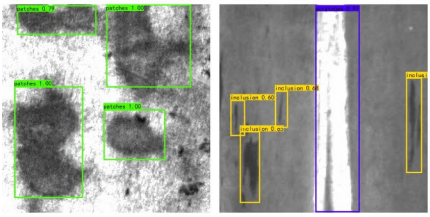
The corresponding experimental results are shown as in Table 6 and Fig. 10. Classic object detection models have poor resistance to scale transformation and are almost unable to detect defects in PCB datasets. The mAP scores of Faster R-CNN (ResNet), SSD (VGG), RetinaNet and CenterNet on this dataset are 5.77%, 12.37%, 6.24% and 26.41%. Although the EfficientDet-D0 has less Params and FLOPs, its mAP scores on the PCB dataset is 8.7%. The EfficientDet-D1 has slightly higher Params and FLOPs than the proposed model, and its input image size is 640×640, but its mAP score is only 39.82%, which is 52.13% lower than the proposed model. This proves that the architecture of the proposed model is more reasonable and efficient for the small defect detection task. The YOLOX-l gets the highest mAP score at 640 input size on the PCB dataset, which is 3.24% higher than the proposed model, but at the cost of 2450% extra FLOPs and 1353% extra Params compared with the proposed model.

Based on the multi-scale input strategy and multi-scale training strategy, we try to use different larger input sizes to conduct experiments and tests on YOLOX-l, YOLOX-x, YOLOv5-l, YOLOv5-x and the proposed model. The selected input sizes are 640, 768, 896, 1024 and 1152, respectively. 1152×1152 is the maximum input size that YOLOv5-x and YOLOX-x can achieve on a single RTX3090 graphics card under the FP16 mixed-precision training strategy. Due to the advantage of low FLOPs, the proposed model additionally selects three input sizes of 1536, 1920 and 2432. Figure 9 is plotted with the

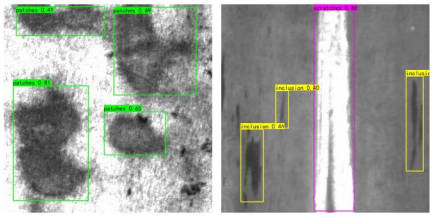


(a) Faster R-CNN.

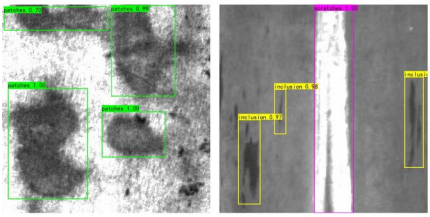
(b) RetinaNet.



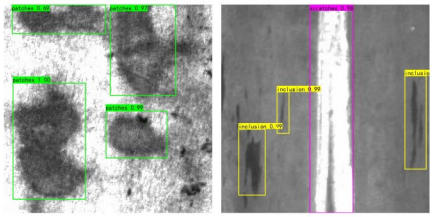
(c) SSD.



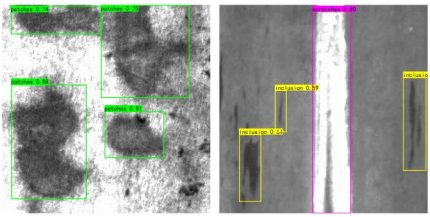
(d) CenterNet.



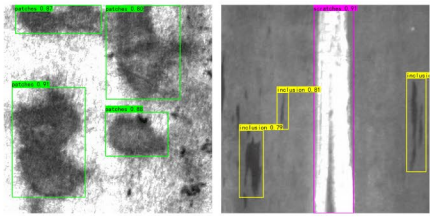
(e) YOLOv3.



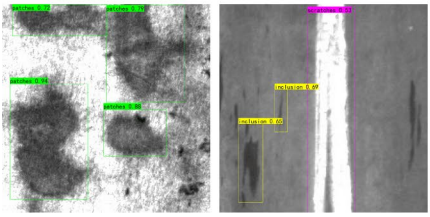
(f) YOLOv4.



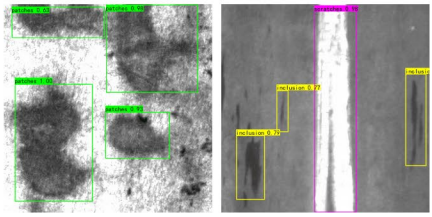
(g) YOLOv5-x.



(h) YOLOX-*l*.



(i) EfficientDet-D5.



(j) Ours.

Fig. 8 Visualization of part of the defect image detection results

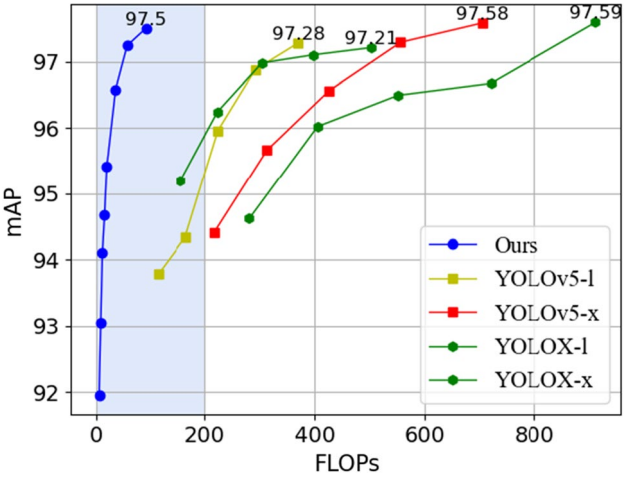


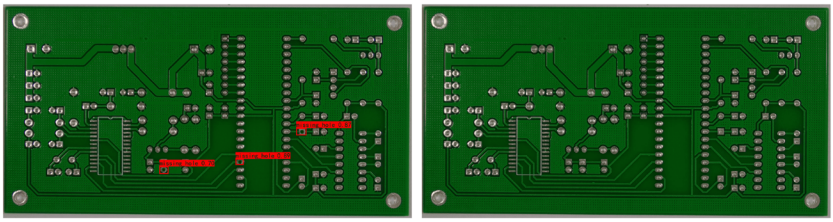
Fig. 9 Performance comparison between SOTA model and the proposed model under different input sizes

Table 6 Comparison of test results on the PCB dataset between the proposed network other networks

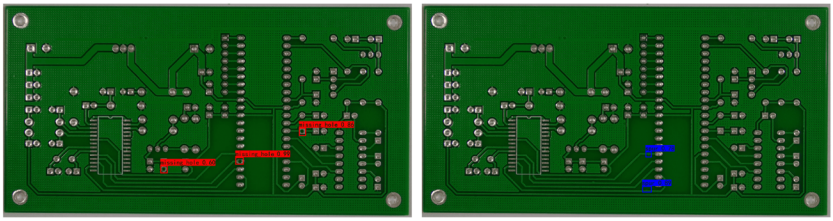
Model	Input size	mAP (%)
YOLOv3	640^2	82.79
YOLOv4	640^2	93.87
YOLOv5-s	640^2	87.75
YOLOv5-m	640^2	91.73
YOLOv5-l	640^2	93.78
YOLOv5-x	640^2	94.41
YOLOX-s	640^2	86.81
YOLOX-m	640^2	89.35
YOLOX-l	640^2	95.19
YOLOX-x	640^2	94.62
EfficientDet-D1	640^2	39.82
EfficientDet-D2	768^2	68.17
EfficientDet-D3	896^2	74.09
EfficientDet-D4	1024^2	87.12
EfficientDet-D5	1152^2	97.03
Proposed model	640^2	91.95

calculation amount of the model under different input sizes as the horizontal axis and the mAP scores on the vertical axis. As can be seen from the figure, when increasing the input size to 2432, the proposed model attains performance on par with the SOTA model in terms of mAP scores, but in this case the FLOPs of the proposed model are much lower than the SOTA model.

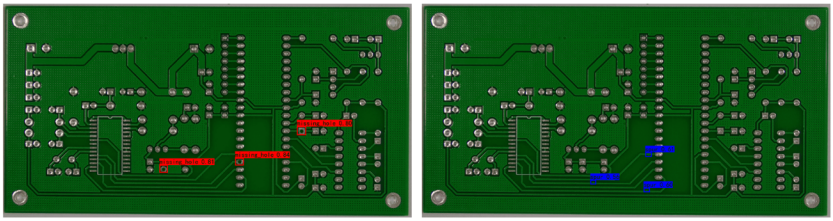
Figure 10 shows the comparison between the proposed model and the SOTA models on 2 randomly selected images of detection results. For multi-model SOTA models such as YOLOv5 and YOLOX, we all choose the model with the highest mAP score to display. As



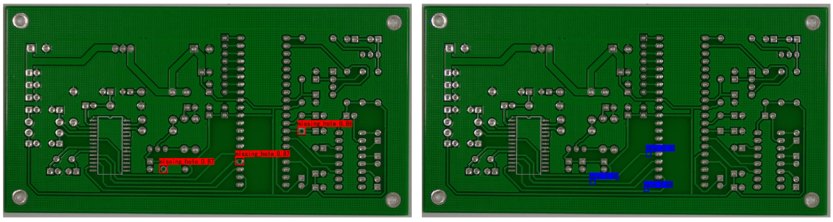
(a) YOLOv3.



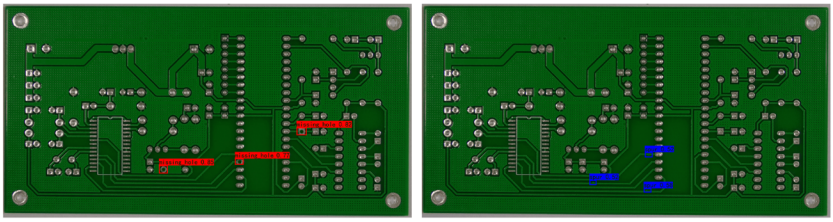
(b) YOLOv4.



(c) YOLOv5-x.



(d) YOLOX-*l*.



(e) Ours.

Fig. 10 Visualization of the detection results comparison of defects on PCB dataset using different object detection models

can be seen from the figure, the detection ability of the proposed model for small objects is comparable to the large SOTA models, and the proposed model is much lower than the large SOTA models in terms of model size and computational complexity.

5 Conclusions

Aiming at the problem that the existing convolutional neural networks are difficult to be applied to industrial surface defect detection due to the high amount of parameters and computation, an efficient lightweight surface defect detection model for industrial product defect detection is proposed in this study. Firstly, based on the inverse residual structure and CA mechanism, a lightweight backbone network named as CA mobile backbone network is designed for preliminary feature extraction. Secondly, the multi-scale strategy is carried out on the features extracted from backbone network on three scales for the small object detection problem, which can highly improve the accuracy and robustness of the proposed model. Finally, a BWFPN is constructed for feature fusion which combines depthwise separable convolution, cross-scale connection, weighted feature fusion approach and multi-scale feature fusion strategy based on attention mechanism. Moreover, the effectiveness of the method is verified by experiments on two open source and challenging dataset. The results of extensive ablation study and comparative experiments show that each module of the model proposed in this paper is efficient, and the detection accuracy of the SOTA model is comparable to that of the SOTA model on both datasets but with much less feature parameters and computation.

Although the lightweight model proposed in this paper has a significant advantage in terms of number of parameters and computational overhead, there are a small number of missed detections in the detection of complex defect images, and there is still room for improvement in the detection accuracy of the proposed model. In future work, we will work on designing more efficient lightweight detection models with higher accuracy, and on the other hand, we will explore the application of our proposed method in the actual industrial process.

Funding This work was supported in part by the [National Natural Science Foundation of China] (Grant Numbers [62001359] and [61973330]), in part by [Foundation of Excellent Young-Backbone Teacher of Colleges and Universities in Henan Province] (Grant Number [2019GGJS182]), in part by [Key Scientific Research Project of Henan Colleges and Universities] (Grant Numbers [20A120005] and [21B120001]) and in part by [Postgraduate Cultivating Innovation and Quality Improvement Action Plan of Henan University] (Grant Number [SYLYC202219]).

Declarations

Conflict of interest All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

References

Aiger D, Talbot H (2010) The phase only transform for unsupervised surface defect detection. In: 2010 IEEE conference on computer vision and pattern recognition (CVPR). pp 295–302. https://doi.org/10.1109/CVPR.2010.5540198

Alex S, Dhanaraj KJ, Deepthi PP (2022) Private and energy-efficient decision tree-based disease detection for resource-constrained medical users in mobile healthcare network. IEEE Access 10:17098–17112. https://doi.org/10.1109/ACCESS.2022.3149771

Amir NIM, Dziyauddin RA, Mohamed N, et al (2022) Real-time threshold-based fall detection system using wearable iot. In: 2022 4th international conference on smart sensors and application (ICSSA). pp 173–178. https://doi.org/10.1109/ICSSA54161.2022.9870974

Bochkovskiy A, Wang CY, Liao H (2020) YOLOv4: optimal speed and accuracy of object detection. Preprint at http://arxiv.org/abs/2004.10934, https://doi.org/10.48550/arXiv.2004.10934

Cha YJ, Choi W, Büyüköztürk O (2017) Deep learning-based crack damage detection using convolutional neural networks. Comput-Aided Civil Infrastruct Eng 32(5):361–378. https://doi.org/10.1111/mice.12263

Cha YJ, Choi W, Suh G et al (2018) Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Comput-Aided Civil Infrastruct Eng 33(9):731–747. https://doi.org/10.1111/mice.12334

Chen Y, Ding Y, Zhao F et al (2021) Surface defect detection methods for industrial products: a review. Appl Sci 11(16):7657. https://doi.org/10.3390/app11167657

Chen G, Wang H, Chen K et al (2022) A survey of the four pillars for small object detection: multiscale representation, contextual information, super-resolution, and region proposal. IEEE Trans Syst Man Cybern Syst 52(2):936–953. https://doi.org/10.1109/TSMC.2020.3005231

Cross GR, Jain AK (1983) Markov random field texture models. IEEE Trans Pattern Anal Mach Intell 5(1):25–39. https://doi.org/10.1109/TPAMI.1983.4767341

Duan K, Bai S, Xie L, et al (2019) Centernet: Keypoint triplets for object detection. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV). pp 6568–6577. https://doi.org/10.1109/ICCV.2019.0066

Ghorai S, Mukherjee A, Gangadaran M et al (2013) Automatic defect detection on hot-rolled flat steel products. IEEE Trans Instrum Meas 62(3):612–621. https://doi.org/10.1109/TIM.2012.2218677

Han K, Wang Y, Tian Q, et al (2020) Ghostnet: More features from cheap operations. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 1577–1586. https://doi.org/10.1109/CVPR42600.2020.00165

Hasan AM, Meziane F, Jalab HA (2016) Performance of grey level statistic features versus gabor wavelet for screening mri brain tumors: a comparative study. In: 2016 6th international conference on information communication and management (ICICM). pp 136–140. https://doi.org/10.1109/INFOCOMAN.2016.7784230

He K, Zhang X, Ren S et al (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916. https://doi.org/10.1109/TPAMI.2015.2389824

Hou Q, Zhou D, Feng J (2021) Coordinate attention for efficient mobile network design. In: 2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 13708–13717. https://doi.org/10.1109/CVPR46437.2021.01350

Howard AG, Zhu M, Chen B, et al (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. Preprint at http://arxiv.org/abs/1704.04861, https://doi.org/10.48550/arXiv.1704.04861

Howard A, Sandler M, Chen B, et al (2019) Searching for mobilenetv3. In: 2019 IEEE/CVF international conference on computer vision (ICCV). pp 1314–1324. https://doi.org/10.1109/ICCV.2019.00140

Hu J, Shen L, Albanie S et al (2020) Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell 42(8):2011–2023. https://doi.org/10.1109/TPAMI.2019.2913372

Huang PW, Lee CH (2009) Automatic classification for pathological prostate images based on fractal analysis. IEEE Trans Med Imaging 28(7):1037–1050. https://doi.org/10.1109/TMI.2009.2012704

Hui TW, Tang X, Loy CC (2018) Liteflownet: A lightweight convolutional neural network for optical flow estimation. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 8981–8989. https://doi.org/10.1109/CVPR.2018.00936

Kumar A, Pang G (2002) Defect detection in textured materials using optimized filters. IEEE Trans Syst Man Cybern B (Cybern) 32(5):553–570. https://doi.org/10.1109/TSMCB.2002.1033176

Li J, Su Z, Geng J et al (2018) Real-time detection of steel strip surface defects based on improved yolo detection network. IFAC-Papers OnLine 51(21):76–81. https://doi.org/10.1016/j.ifacol.2018.09.412

Li J, Pu Y, Tang J et al (2020) Deepatt: a hybrid category attention neural network for identifying functional effects of dna sequences. Brief Bioinform 22(3):159. https://doi.org/10.1093/bib/bbaa159

Li F, Li F, Xi Q (2021) Defectnet: toward fast and effective defect detection. IEEE Trans Instrum Meas 70:1–9. https://doi.org/10.1109/TIM.2021.3067221

Li F, Li Z, Liu J, et al (2022) Recognition method of two types of insulation joints based on wavelet transform and svm. In: 2022 global conference on robotics, artificial intelligence and information technology (GCRAIT). pp 736–741. https://doi.org/10.1109/GCRAIT55928.2022.00158

Lin TY, Dollár P, Girshick R, et al (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). pp 936–944. https://doi.org/10.1109/CVPR.2017.106

Lin WY, Lin CY, Chen GS, et al (2019) Steel surface defects detection based on deep learning. In: Proceedings of the international conference on applied human factors and ergonomics (AHFE). pp 141–149. https://doi.org/10.1007/978-3-319-94484-5_15

Lin TY, Goyal P, Girshick R et al (2020) Focal loss for dense object detection. IEEE Trans Pattern Anal Mach Intell 42(2):318–327. https://doi.org/10.1109/TPAMI.2018.2858826

Liu L, Lao S, Fieguth PW et al (2016a) Median robust extended local binary pattern for texture classification. IEEE Trans Image Process 25(3):1368–1381. https://doi.org/10.1109/TIP.2016.2522378

Liu W, Anguelov D, Erhan D, et al (2016b) Ssd: single shot multibox detector. In: Proceedings of the IEEE European conference on computer vision (ECCV). pp 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

Liu S, Qi L, Qin H, et al (2018) Path aggregation network for instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 8759–8768. https://doi.org/10.1109/CVPR.2018.00913

Loshchilov I, Hutter F (2016) SGDR: stochastic gradient descent with restarts. Preprint at http://arxiv.org/abs/1608.03983, https://doi.org/10.48550/arXiv.1608.03983

Luo Q, Fang X, Liu L et al (2020) Automated visual defect detection for flat steel surface: a survey. IEEE Trans Instrum Meas 69(3):626–644. https://doi.org/10.1109/TIM.2019.2963555

Luo J, Yang Z, Li S et al (2021) Fpcb surface defect detection: a decoupled two-stage object detection framework. IEEE Trans Instrum Meas 70:1–11. https://doi.org/10.1109/TIM.2021.3092510

Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on feature distributions. Pattern Recogn 29(1):51–59. https://doi.org/10.1016/0031-3203(95)00067-4

Pan J, Yang A, Wang D et al (2022) Lightweight neural network for gas identification based on semiconductor sensor. IEEE Trans Instrum Meas 71:1–8. https://doi.org/10.1109/TIM.2021.3135503

Redmon J, Farhadi A (2018) YOLOv3: an incremental improvement. Preprint at http://arxiv.org/abs/1804.02767, https://doi.org/10.48550/arXiv.1804.02767

Ren S, He K, Girshick R et al (2017) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

Rezatofighi H, Tsoi N, Gwak J, et al (2019) Generalized intersection over union: A metric and a loss for bounding box regression. In: 2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 658–666. https://doi.org/10.1109/CVPR.2019.00075

Sandler M, Howard A, Zhu M, et al (2018) Mobilenetv2: inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 4510–4520. https://doi.org/10.1109/CVPR.2018.00474

Shen J, Liu N, Xu C et al (2022) Finger vein recognition algorithm based on lightweight deep convolutional neural network. IEEE Trans Instrum Meas 71:1–13. https://doi.org/10.1109/TIM.2021.3132332

Song Y, Cai W, Zhou Y et al (2013) Feature-based image patch approximation for lung tissue classification. IEEE Trans Med Imaging 32(4):797–808

Song L, Lin W, Yang YG et al (2019) Weak micro-scratch detection based on deep convolutional neural network. IEEE Access 7:27547–27554. https://doi.org/10.1109/ACCESS.2019.2894863

Song Z, Zhang Y, Liu Y et al (2022) Msfyolo: feature fusion-based detection for small objects. IEEE Lat Am Trans 20(5):823–830. https://doi.org/10.1109/TLA.2022.9693567

Tan SC, Watada J, Ibrahim Z et al (2015) Evolutionary fuzzy artmap neural networks for classification of semiconductor defects. IEEE Trans Neural Netw Learn Syst 26(5):933–950. https://doi.org/10.1109/TNNLS.2014.2329097

Tan M, Pang R, Le QV (2020) EfficientDet: scalable and efficient object detection. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 10778–10787. https://doi.org/10.1109/CVPR42600.2020.01079

Wang L, Chang Y, Wang H et al (2017) An active contour model based on local fitted images for image segmentation. Inf Sci 418–419:61–73. https://doi.org/10.1016/j.ins.2017.06.042

Wang H, Liu C, Yu L, et al (2019) Research on target detection and recognition algorithm based on deep learning. In: Proceedings of the Chinese control conference (CCC). pp 8483–8487. https://doi.org/10.23919/ChiCC.2019.8865560

Wang Q, Wu B, Zhu P, et al (2020) Eca-net: efficient channel attention for deep convolutional neural networks. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 11531–11539. https://doi.org/10.1109/CVPR42600.2020.01155
Wang D, Zhang Z, Jiang Y et al (2021a) Dm3loc: multi-label mrna subcellular localization prediction and analysis based on multi-head self-attention mechanism. Nucleic Acids Res 49(8):e46. https://doi.org/10.1093/nar/gkab016
Wang H, Liu C, Zhao Z et al (2021b) Application of deep convolutional neural networks for discriminating benign, borderline, and malignant serous ovarian tumors from ultrasound images. Front Oncol 11:770683. https://doi.org/10.3389/fonc.2021.770683
Woo S, Park J, Lee JY et al (2018) Convolutional block attention module, vol 11211. Springer, Cham, pp 3–19. https://doi.org/10.1007/978-3-030-01234-2_1
Wu MS, Li CY (2021) Edge-based realtime image object detection for uav missions. In: 2021 30th wireless and optical communications conference (WOCC). pp 293–294. https://doi.org/10.1109/WOCC53213.2021.9602868
Wu T, Luo J, Fang J et al (2018) Unsupervised object-based change detection via a weibull mixture model-based binarization for high-resolution remote sensing images. IEEE Geosci Remote Sens Lett 15(1):63–67. https://doi.org/10.1109/LGRS.2017.2773118
Xie S, Shan S, Chen X et al (2010) Fusing local patterns of gabor magnitude and phase for face recognition. IEEE Trans Image Process 19(5):1349–1361. https://doi.org/10.1109/tip.2010.2041397
Xie L, Xiang X, Xu H et al (2021) Ffcnn: a deep neural network for surface defect detection of magnetic tile. IEEE Trans Ind Electron 68(4):3506–3516. https://doi.org/10.1109/TIE.2020.2982115
Yan Y, Kaneko S, Asano H (2020) Accumulated and aggregated shifting of intensity for defect detection on micro 3d textured surfaces. Pattern Recogn 98(107):057. https://doi.org/10.1016/j.patcog.2019.107057
Yang J, Li S, Wang Z et al (2020) Using deep learning to detect defects in manufacturing: a comprehensive survey and current challenges. Materials 13(24):5755. https://doi.org/10.3390/ma13245755
Zhao G, Yang H, Yu M (2020) Expression recognition method based on a lightweight convolutional neural network. IEEE Access 8:38528–38537. https://doi.org/10.1109/ACCESS.2020.2964752
Zheng G, Songtao L, Feng W, et al (2021a) YOLOx: exceeding yolo series in 2021. Preprint at http://arxiv.org/abs/2107.08430, https://doi.org/10.48550/arXiv.2107.08430
Zheng Z, Zhao J, Li Y (2021b) Research on detecting bearing-cover defects based on improved yolov3. IEEE Access 9:10304–10315. https://doi.org/10.1109/ACCESS.2021.3050484

Authors and Affiliations

Dehua Zhang[1] · Xinyuan Hao[1] · Dechen Wang[1] · Chunbin Qin[1] · Bo Zhao[2] · Linlin Liang[3] · Wei Liu[4]

Dehua Zhang
dhuazhang@vip.henu.edu.cn

Xinyuan Hao
hao_xinyuan@foxmail.com

Dechen Wang
wdechen666@163.com

Chunbin Qin
qcb@henu.edu.cn

Linlin Liang
llliang@xidian.edu.cn

Wei Liu
lw3171796@163.com

[1] School of Artificial Intelligence, Henan University, Zhengzhou 450046, Henan, China

[2] School of Systems Science, Beijing Normal University, Beijing, Beijing 100875, China

[3] School of Cyber Engineering, Xidian University, Xi'an 710126, Shanxi, China

[4] College of Electromechanic Engineering, Nanyang Normal University, Nanyang 473061, Henan, China