IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 March 2023, accepted 26 March 2023, date of publication 3 April 2023, date of current version 6 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3264262

RESEARCH ARTICLE

A Lightweight Detector Based on Attention Mechanism for Fabric Defect Detection

XIN LUO, QING NI, RAN TAO, AND YOUQUN SHI
School of Computer Science and Technology, Donghua University, Shanghai 201620, China
Corresponding author: Qing Ni (13033190657@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB1707700, and in part by the Fundamental Research Funds for the Central Universities under Grant 20D111201.

ABSTRACT Defects on fabric surfaces are difficult to identify owing to unsuitable computing devices, highly complex algorithms, small size, and high degree of integration with the fabric. To this end, this study proposes a lightweight fabric defect-detection network, YOLO-SCD, based on attention mechanism. The introduction of depth-wise separable convolution and the attention mechanism enhanced the capacity of the neck network to extract the defective features and increased the detection speed of the overall network. The extensive experimental results revealed that YOLO-SCD achieved an average accuracy of 82.92%, effective improvement of 8.49% in mAP, and an improvement of 37 fps compared to the original YOLOv4 on a standard fabric defect dataset. By leveraging its swift detection speed and high efficiency, YOLO-SCD excels in both the general fabric defect category and the difficult-to-detect fabric. Overall, it exhibited strong performance in detecting both minor flaws and flaws with high fabric integration. Furthermore, the proposed model was extended to steel datasets with similar characteristics.

INDEX TERMS Fabric defect detection, SoftPool, attention mechanism, depthwise separable convolution, lightweight.

I. INTRODUCTION

Fabric defects are key factors affecting the quality of fabric production and fabric grade. According to market research, the price of fabrics with prominent defects is approximately 50% less than that of defect-free fabrics and cause considerable economic losses to textile enterprises [1]. Although manual inspection is a common defect-detection method in the traditional fabric industry, the conventional methods involve substantial human and management resources and yields low detection efficiency, high error detection rate, and considerable damage to the workers' eyes [2]. Therefore, the development of an automatic inspection method with high detection accuracy and fast detection speed is required to replace the currently employed manual method and improve the production efficiency, reduce workers' labor intensity, and lower production costs of enterprise. With the rapid development of computer technology and machine learning, several inspection methods have started using machine vision and deep learning to replace traditional manual inspection methods and achieve adequate detection results.

To detect defects on the fabric surface, machine vision methods are used for analyzing the texture and defect characteristics of the fabric, followed by identifying and locating the defects using image processing technology. In recent years, domestic and foreign companies have developed machine vision products for fabric defect detection, such as FABRISCAN and webSPECTOR. The detection standards of these devices differ from those regulated in China, and small textile factories cannot afford such detection methods [3].

The existing fabric defect-detection methods proposed by domestic and foreign researchers can be segmented into four major categories: structural analysis [4]—Dhivya and Devi [5] applied the closest node algorithm for multiscale, multidirectional extraction of the contour features of fabric defects. Although this method outperformed support vector machines in experiments, it requires initial pre-processing through conversion and filtering techniques, which accounts

The associate editor coordinating the review of this manuscript and approving it for publication was Poki Chen.

for redundancy. Statistical analysis [6]—Kumari et al. [7] extracted fabric defects based on the similarity estimation method of Sylvester matrix, which can process test images captured under various lighting conditions. However, it detected only three types of defects and its fitting ability was inadequate. Frequency domain analysis [8]—Mak et al. [9] developed a method based on filter-based approach to extract features from images of varying scales. Although it accurately detected the edge defects and holes, it was limited in optimizing the parameter values of the filter and was computationally intensive. Model analysis [10]—Lin et al. [11] used grayscale co-occurrence matrix and redundant contour transform to extract the segmented sub-images texture features and combined it with convolutional neural network classification method to improve the recognition rate of the fabric defects; however, this method is more sensitive to lighting and image noise. These conventional detection methods require predefined thresholds to detect the presence of defects, and the extracted features must be carefully designed. These methods are effective only for a specified defect class and manifest inferior adaptability and insufficient generalization ability under improper imaging conditions.

Deep learning-based target detection algorithms have achieved significant progress in industrial applications for detecting defects in steel and aluminum. Fundamentally, these algorithms are of two types: two-stage detectors and one-stage detectors. The target detection process of the two-stage detector first involves extracting a set of object candidate frames by selective search, followed by the classification of the candidate frames to determine the exact target location. Common two-stage detectors include RCNN [12], Fast R-CNN [13], and Faster R-CNN [14]. Zhou et al. [15] proposed a fabric defect-detection method based on Faster R-CNN by combining a feature pyramid network (FPN), deformable convolutional (DC) network, and distance IoU loss function with an average prediction speed of 17 frames per second (fps), which was less than the detection level of 25 fps required in actual factories. The single-stage detector involved only a single operation of the CNN to obtain classification and position, which exhibited a faster detection speed and achieved end-to-end detection. Compared with two-stage detection networks, one-stage networks do not include a stage for generating candidate frames, and therefore, offer significantly advantageous detection speed. Commonly used single-stage detectors include the YOLO series [16], [17], [18], [19], single-shot multibox detector (SSD) [20], and CenterNet [21]. Zhang et al. [22] proposed a YOLOV2-based method for automatic localization and classification of color fabric defects, which improved the model by optimizing the hyperparameters of the neural network. However, its training set contained only 200 sheets with three categories of defects, which yielded weak generalization and inferior robustness of the model. Luo et al. [23] proposed a fabric defect-detection method based on YOLOv3 combined with deformable convolution, which provided adequate detection results for 17 types of fabric defects. However, the model size was excessively large for practical applications in textile mills. Recently, semantic segmentation has been applied by the computer vision community. The segmentation of an image into groups of pixels with labels and classifications forms the boundary segmentation of existing targets in an image based on pixels. Lu et al. [24] proposed a fabric defect detection method based on a C-RCNN for image segmentation, but the segmentation results were inaccurate if the fabric defects were extremely small or similar.

Although the two-stage detector offers advantages in terms of fabric defect-detection accuracy, it is not as fast as the single-stage detector. This method yields unsuitable image segmentation of fabric defects for fabric defect detection owing to the small size of the defects and the high fusion of defects with the fabric, which restricts the accuracy of detecting fabric defects. Most current studies performed defect detection on only a small number of easily detectable defect types. Thus, the number of defect types that can be detected by the model and its detection accuracy for hard-to-detect defects requires imminent improvement. Moreover, the real-time detection of fabric defects is a challenge due to the large network structure.

Conventional image processing-based defect detection methods fail to appropriately adapt to the complex and continually varying industrial applications. In comparison, deep learning-based target detection algorithms are more robust and can improve the recognition accuracy of defect detection. Therefore, exploring the automatic detection of fabric defects based on deep learning is vital for improving the accuracy of detecting fabric surface defects and enhancing the development of related industries. To this end, this study selects a single-stage detector YOLOv4 as the benchmark model and proposes a new detection network YOLO-SCD that inherits the automatic learning characteristics of YOLOv4 and exhibits improved anti-interference capability, especially for detecting small defects and long strips of defects that are typically difficult to detect. In principle, YOLO-SCD incorporates two targeted techniques—SoftPool and an attention mechanism approach—to enhance the ability of the model for handling hard-to-detect fabric defects and reducing the loss of the model for extracting small fabric defects and other types of feature losses. The introduction of depthwise separable convolution (DSC) significantly reduced the memory of the model and improved the speed of the model in detecting defective images. The fundamental contributions of this research are summarized as follows.

(i) The k-means method was employed to improve the anchor frame used for training and increase the ability to detect fabric faults with large-scale fluctuations. In addition, a new feature pyramid structure with an additional shallow backbone network branch output was suggested, which enabled the model to readily extract the defects of the complex shapes on the fabric surface by fusing shallow and deep features.

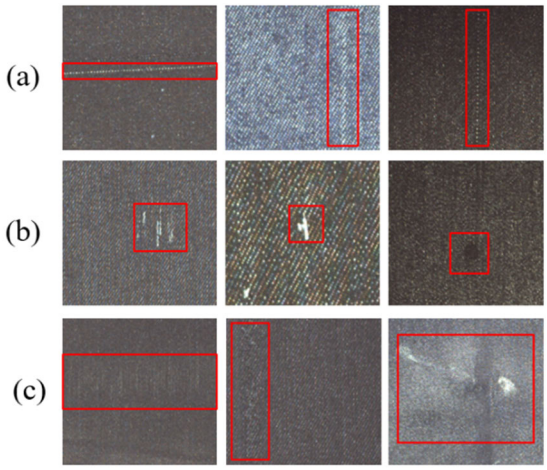


FIGURE 1. Enlarged display of some fabric defect.

(ii) An attention mechanism was introduced in the middle- and high-level components of the network to focus its attention on the defective portion and reduce the loss of the model during the process of defective feature extraction for small defects and other hard-to-detect defects.

(iii) The introduction of DSC significantly reduced the storage space of the model and improved the model detection speed.

The remainder of the paper is organized as follows: the fabric defect features are analyzed in Section II (Related Work), with the introduction of the deep learning models developed for target detection and fabric defect detection in recent years. The detailed structure of YOLO-SCD and certain other improved methods are presented in Section III (Proposed Network). The validation of the proposed method along with certain comparative experiments considering other methods are discussed in Section IV (Experiments and Discussion). The conclusions and future scope of this research are summarized in Section V (Conclusions).

II. RELATED WORK

The fabric defect-detection algorithm includes the extraction of defective features, defect classification, and defect location. In general, fabric surface defects contain numerous categories: *hundred feet*, *broken warp*, *knots*, *hole*, *pulp spots*, *stains*, *abrasion marks*, *three filaments*, *loose warp*, *grain*, etc. The enlarged view of certain fabric defects is displayed in Figure 1, which depicts the following problems related to fabric defects.

1) NONUNIFORM DISTRIBUTION OF DEFECTIVE SHAPE CHARACTERISTICS

Hundred feet and other defects are long and thin lines (Figure 1 (a)), whereas *grain*, *broken warp*, and certain portions of other defects display a dotted appearance (Figure 1 (b)).

2) DEFECTS ACCOUNTED FOR VARIOUS SIZES OF FABRIC AREA

Certain defects such as *stains* and *grains* appear dotted and account for a smaller area of the fabric that is challenging to detect. In comparison, defects such as *pulp spots* and *holes* occupy a larger space in the fabric.

3) HIGH INTEGRATION OF FABRIC DEFECT WITH FABRIC

In Figure 1 (c), the first two defective images illustrate the defect categories of *pulp spot* and *abrasion marks*; its color is similar to the fabric background and cannot be easily distinguished only by the naked eye.

The aforementioned problems of fabric defects, *i.e.*, covering a small proportion of the area with high degree of integration with fabric characteristics, have created challenges for fabric defect detection. The traditional manual detection of fabric defects by the naked eye is difficult and exhibits a high leakage rate with low detection efficiency. To overcome these drawbacks of manual detection, scholars have used a combination of image-processing techniques and manually designed features. Li et al. [25] proposed a pattern-free fabric defect-detection scheme that includes texture feature extraction and detection stages by combining the ''uniform'' MDBP operator and grayscale co-occurrence matrix that initially generates an image vector based on a multidirectional binary pattern to extract the grayscale co-generation matrix, and thereafter, generates the similarity values based on the matrix similarity before arriving at the defect detection result map. Although the detection of defects *via* traditional manual extraction of features is convenient compared to that by the human eye, the process remains tedious with improper imaging conditions.

In recent years, with the widespread industrial application of image processing technology and computer vision technology, new ideas and means have been developed for detecting fabric defects. Lin et al. [26] developed a top–down architecture with horizontal connections, *i.e.*, a feature pyramid, to construct high-level semantic feature maps of all scales. Yang et al. [27] proposed PanNet, which is a top–down and bottom–up bidirectional fusion backbone network based on the feature pyramid, with a ''short-cut'' between the bottom and top layers. Zhang et al. [28] proposed a multifeature aggregation framework for salient object recognition. The feature pyramid can extract features for every scale size of image defects and generate multiscale feature representations. More importantly, strong semantic information can be generated using the feature pyramid network depending on the type of fabric defects, *i.e.*, point-like defects such as *hair grains* and *stains* or linear defects such as *hundred feet* and *broken warps*. The bidirectional backbone fusion method induced by PanNet can generate an effective flow of the fabric defect features between the network layers, because the bottom layer of the network contains additional information on both small and slender defects. Furthermore, multilayer feature fusion can reduce the loss in the fabric

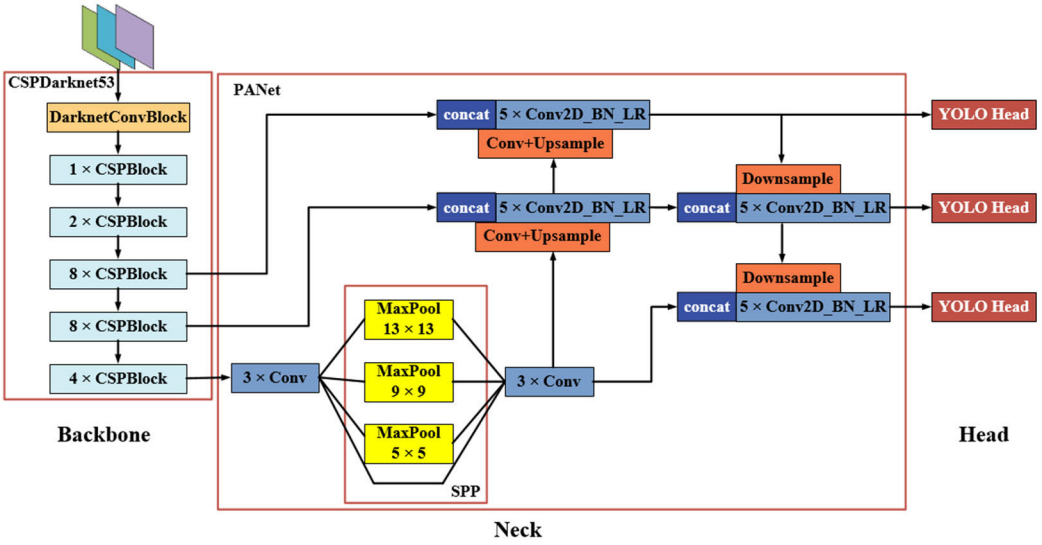


FIGURE 2. The Network structure of YOLOv4.

defect feature extraction process and facilitate the feature extraction of small defects as well as other types of difficult-to-detect defects.

In April 2020, Bochkovskiy et al. [18] proposed YOLOv4 by adding new and improved concepts such as mosaic data enhancement and PanNet to YOLOv3 to achieve the best balance of accuracy and speed for the COCO dataset. Consequently, they derived the end-to-end object location and category output. The continuous improvement of the YOLO network structure from v1 to v4 increased the detection speed and accuracy of complex objects to achieve the most optimal balance between speed and accuracy. The schematic of the original YOLOv4 model structure is illustrated in Figure 2, which was segmented into three components: backbone, neck, and head.

The backbone uses the CSPDarknet53 [29] network for feature extraction, comprising a five-group stacked residual layer structure and a convolutional block with a Mish activation function and batching. The residual structure divides the feature mapping of the base layer into two components and merges them through the cross-stage hierarchy, which reduces the computational bottleneck of the entire model and ensures its high accuracy.

The neck component is used to enhance the features, and the SPP structure [30] along with the FPN+PANet structure [31] is used to multiplex and fuse the features of the three feature layer outputs from the backbone component to enhance the feature representation capability of the model. The feature maps after the backbone network feature extraction are passed into the SPP module, which uses four distinct maximum pooling scales (13×13, 9×9, 5×5, and 1×1) to fuse the four feature maps by Concat operation and enhance the information representation capability of the shallow feature map input. The FPN layer captures the strong semantic features of the image, whereas the PAN records the strong semantic features of the image by self-bottom–up to convey strong localization features. Upon combining these two modules, the target localization can be accomplished. However, using an extensive amount of maximum pooling in SPP affects the accuracy of the fabric defect localization, which deteriorates the fabric defect classification tasks in case the target is similar to the background.

The head section is used to decode the output of the three feature maps by feature enhancement, decode the anchor frame into the input image, compute the loss function for defect prediction, and output the final detection result map, including the confidence level, target location, and corresponding class of the target. Although YOLOv4 delivers adequate detection performance on the large target dataset COCO, the analysis in Section II indicates hard-to-detect characteristics of the fabric. In fabric defects such as *stains* and *grains* that account for a small fabric area and are difficult to detect with the naked eye, the application of YOLOv4 cannot be migrated to the field of fabric defect detection. In this regard, Liu et al. [32] proposed a fabric defect detection method based on YOLOv4 with high detection accuracy, but it divided the fabric defect dataset only into four categories, all of which included easily detectable fabric defect types such as lines and holes, and no targeted research based on minor defects was conducted.

Based on the above analysis, most existing fabric defect detection models were only applied to a small number of easily detectable defects. In the actual production process, dozens of types of fabric defects exist and most defects pertain to the types that are not easily detected. For instance, certain defects were slender and readily deemed as lines, and certain defects were not easily found with high fabric fusion. Therefore, based on the study of the YOLOv4 structure, this study proposes the YOLO-SCD model

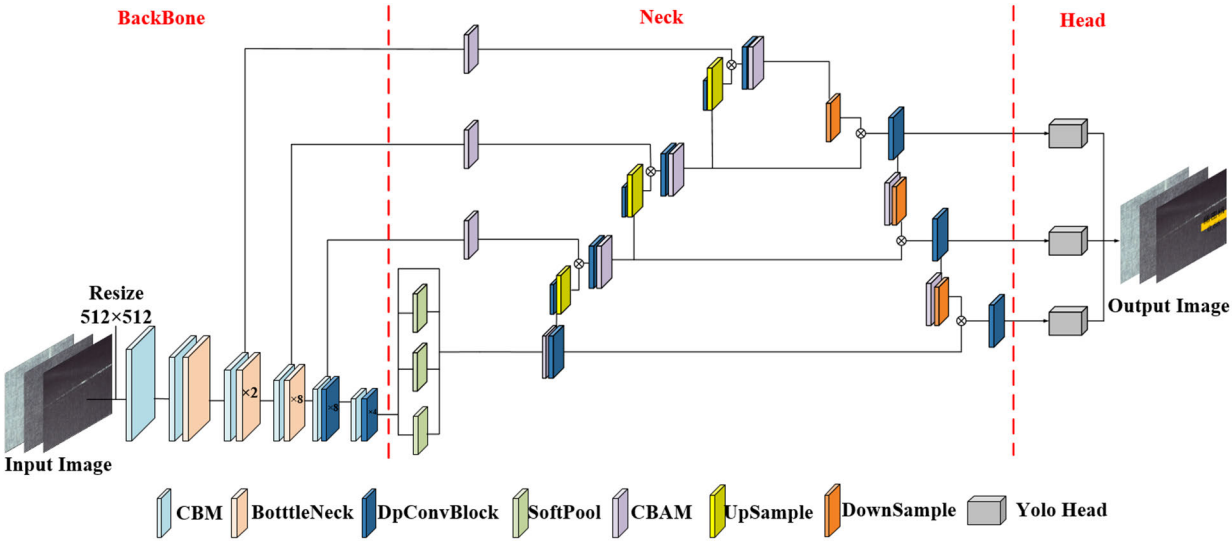


FIGURE 3. Network structure of YOLO-SCD.

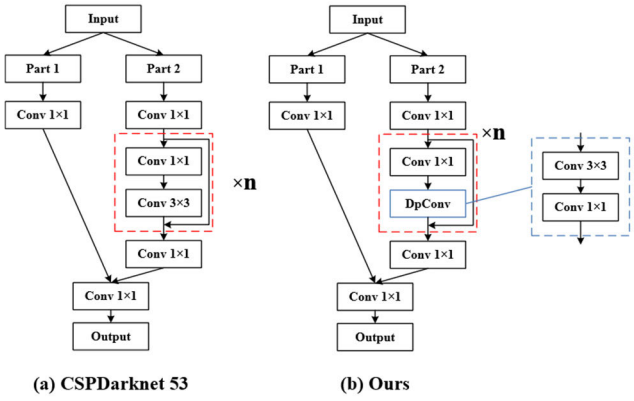


FIGURE 4. Residual structure used in CSPDarknet53 and proposed model.

and designs the corresponding feature fusion module to accurately detect miniscule defects and those highly fused with fabrics.

III. PROPOSED NETWORK

The proposed YOLO-SCD model is introduced in detail herein. First, we introduce the end-to-end structure of the proposed model, including the loss functions used by the network. Second, we present several solutions of the model for fabric defect detection, including SoftPool, attention mechanism, and lightweight procedure of the detection model using DSC.

A. INTRODUCTION OF YOLO-SCD

The structure of the proposed model generally follows that of YOLOv4, including the backbones, neck, and head; the schematic of the overall network structure is illustrated in Figure 3.

1) BACKBONE

Initially, the model resizes the input defect image, compressing it to a resolution of 512 × 512, and feeds the defect image into the feature-extraction network. Thereafter, the backbone portion of the network retains the CSPDarknet53 structure used in YOLOv4, wherein we believe that multiple large residual structures can adequately retain the defect information. In principle, the feature map of the upper layer input is segmented into two portions and merged through the cross-stage hierarchy to reduce the repetitive gradient information along with improving the inference speed. As certain fabric defects cannot be easily distinguished with the naked eye because of their large-scale variations and high fusion with the fabric, a shallow output branch from the backbone network to the PANet network of the neck is added for feature reuse and feature fusion to maximize the retention of the defect features. Consequently, the output feature map size becomes (128 × 128 × 128). To reduce the model size and improve the detection speed, the DSC is introduced into the structure of the last two residual layers of CSPDarknet. As depicted in Figure 4, a part of the input feature map passes through a convolution block, several residual blocks, and another convolution block to extract features. Similarly, the other part is first convolved and then combined with it. Finally, the combined part passes through a transformation layer (convolution block) to derive the final output. The CSPDarknet53 contains 1, 2, 8, 8, and 4 residual layers in each stage. Notably, to achieve the most optimal balance between accuracy and speed, we introduced a 3 × 3 convolution in the last two large residual groups in the model into the DSC and constructed a new bottleneck layer named DpBlock, as depicted in Figure 4(b). Overall, the design of the new residual layer reduced the model size as well as the weight of feature extraction network, thereby reducing the detection speed of the model.

2) NECK

The neck network mainly conducts secondary processing for fabric defect features, such that the network retains more feature information for small defects, using the SPP structure and the FPN+PANet for feature enhancement (refer to Figure 3). The four feature maps obtained as output from the backbone are fed into the SPP structure and pyramid network through the attention mechanism, where the SPP structure primarily extracts the multiscale fabric-defect information by pooling the feature layers inputted from the backbone network through multiple pooling windows. This mechanism effectively expands the perceptual field and can extract richer contextual semantic information, *i.e.*, multiscale fabric defect information. To retain more semantic feature information of the defects, the SPP structure uses a mild feature mapping SoftPool for spatial pyramidal pooling, which is in proportion to the corresponding values of the elements. Additionally, we designed a new PANet structure. As the original structure of the PANet reuses only the high-level feature information of the fabric, we added a shallow pyramid layer to the original three-layer pyramid structure of PANet to retain more information on small fabric defects. Moreover, we embedded a spatial attention mechanism (Figure 3: CBAM) in the up-sampling and feature fusion segment of the pyramid structure to ensure that the model focuses more attention on the defect part, thereby intending to resolve the issue with highly fused fabric defects. Furthermore, we applied DSC for feature extraction and reducing the size of the feature enhancement network. The improved PANet is more suitable for extracting small fabric defects and enhancing the semantic information of the defects.

3) HEAD

The YOLO Head was used for the module to determine whether the fabric defects are identifiable and the type of identified defects corresponds to the three priori frames preset in the feature enhancement networks. In particular, the three tensor dimensions of the input feature—(64, 64, 256), (32, 32, 512), and (16, 16, 1024)—with output dimensions (after Head) of (64 64, 21), (32, 32, 21), (16, 16, 21), and 21 in the last dimension contains the x_offset, y_offset, image width and height, confidence level, and defect classification result. The decoding process of the head first involves adding each grid point with its corresponding x_offset and y_offset to derive the center of the prediction frame. Thereafter, using the combination of the width and height of the prior frame and the feature map, the length and width of the prediction frame were calculated to derive the position of the entire prediction frame. Finally, score sorting and non-great suppression filtering of these prediction frames were performed to plot the final prediction results of the defect classification.

4) LOSS FUNCTION

As discussed in Section II, fabric defects such as *hair grains* and *warp breaks* constitute a small proportion of the entire fabric image. Owing to the high degree of fusion of these defects with the fabric, several types of defects appear similar to each other, resulting in several difficult-to-classify samples. To resolve this issue, the loss function used by YOLO-SCD combines CIOU loss [33] and focal loss [34] to ensure a higher detection accuracy for fabric defect detection. Although CIOU reflects the distance and compliance between the predicted bounding box and the real box, these two boxes do not intersect and prevent the regression gradient from reaching zero. In addition, it considers the aspect ratio fitting between the predicted bounding box and the real box, thereby promoting faster computational convergence of the model. The focal loss can control the weights of the positive and negative samples as well as those of the easy- and difficult-to-classify samples. This solves the imbalance between the number of easy-to-classify fabric defects and difficult-to-classify fabric defects in the actual training and enhances the prediction ability of the network model for difficult-to-classify defects. The loss of the YOLO-SCD model is primarily composed of target position loss ($LOSS_{CIOU}$), classification loss ($LOSS_{cls}$), and focus loss ($LOSS_{focal}$), calculated as follows.

(i) The target position loss ($LOSS_{CIOU}$) is used to evaluate the error between the predicted result box and the real box; a smaller loss indicates a higher degree of overlap between the predicted box and the real box, in addition to higher positioning accuracy of the predicted bounding box.

$$LOSS_{CIOU} = 1 - IOU + \frac{\rho^2(b, b^{gt})}{C^2} + \alpha v \tag{1}$$

$$\alpha = \frac{v}{(1 - IOU) + v} \tag{2}$$

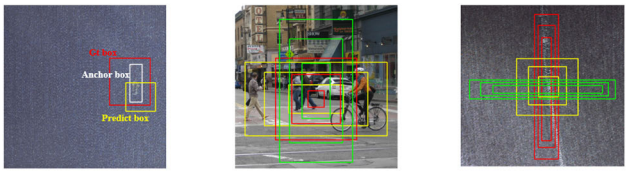
$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{3}$$

where $\rho^2(b, b^{gt})$ denotes the Euclidean distance between the predicted box b and the center-point coordinates of the real box b^{gt}. In (2), α denotes the weight function and v is used to measure the consistency of the aspect ratio; in (3), w^{gt} and h^{gt} denote the width and height of the real box and w and h represent the width and height of the predicted box, respectively.

(ii) Classification loss ($LOSS_{cls}$) compares the predicted and actual results of the categories, expressed as follows:

$$L_{cls} = \sum_{i=0}^{s^2} l_{i,j}^{obj} \sum_{c \in \text{classes}} \bar{p}_i^j(c) \log\left(p_i^j(c)\right) + \left(1 - \bar{p}_i^j(c) \log\left(1 - P_i^j(c)\right)\right) \tag{4}$$

where s^2 denotes the number of grids in the input feature map; $l_{i,j}^{obj}$ indicates the presence of the object in the i-th grid in the j-th prediction frame; $\bar{p}_i^j(c)$ denotes the probability that the target defect class is in the true frame; $p_i^j(c)$ indicates the probability of the target defect class in the prediction frame.



(a) three kinds of box (b) the anchor box of YOLOv4 (c) the anchor box of ours

FIGURE 5. Introduction of anchor frame.

(iii) Focal Loss ($LOSS_{focal}$) is used to calculate the confidence level of the fabric defects, as follows:

$$LOSS_{focal} = -\alpha_t (1 - p_t)^{\gamma} \log (p_t) \tag{5}$$

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{otherwise} \end{cases} \tag{6}$$

where p_t reflects the proximity of the prediction frame to the true frame, *i.e.*, a larger value implies a more accurate classification. α_t and γ are used to solve the positive and negative sample imbalance along with the difficult-to-classify problems in the defective fabric samples, respectively, where $\gamma = 2$.

5) IMPROVEMENT OF ANCHOR FRAME

To better fit the defect aspect ratio in the fabric defect dataset, the anchor box parameter suitable for this experiment must be calculated prior to model training. The anchor box refers to a representative rectangle of pixels with varying scale sizes in the image, which assisted the model to learn the target location, as depicted in Figure 5(a). If each prediction result was adjusted according to the real box, the object size would increase beyond computational efficiency. Accordingly, the location of the real box was unified and initially set to nine scales, according to the defective dataset. as discussed in Section II, the current fabric dataset exhibits a nonuniform distribution of defect sizes, i.e., certain fabric defects display an aspect ratio of 1:1, whereas other defects portray an aspect ratio of 20:1. the anchor frame used in

YOLOV4 is depicted in Figure 5(b). although the target contained three scales, it was not applicable to the characteristic properties of the fabric defects. thus, the values of the original yolov4 anchor frame should be improved.

In this study, k-means clustering was used to optimize the size of the anchor frame based on the true value of the defect size of the defect dataset, and the average IOU was defined as the clustering measure. The average IOU was calculated as follows:

$$\text{AvgIOU} = \arg\max \frac{\sum_{i=1}^{k} \sum_{j=1}^{n_k} I_{IOU}(B, C)}{n} \tag{7}$$

where $I_{IOU}(B, C)$ represents the intersection ratio of the true value to the bounding box of the cluster centers, B denotes the true value, C represents the cluster center, and n denotes the total number of clustered objects. Accordingly, we selected $k = 9$ cluster centers, and the final generated anchor boxes are illustrated in Figure 5(c), with sizes (1, 12), (1, 29), (3, 10), (2, 25), (1, 78), (12, 37), (2, 463), (9, 497), and (145, 121). The size of the anchor frame was set adaptively according to the aspect ratio of the fabric defects, which improved the recognition and localization accuracy of the defect model.

B. OTHER MEANS OF YOLO-SCD TO IMPROVE FABRIC DEFECT DETECTION

The analysis of the anchor frame in Section II and Section III-A5 highlights the presence of numerous fabric defects, including those occupying a relatively small area, with extremely low aspect ratios. Moreover, most *pulp spots*, *warp knots*, and other similar defects are highly integrated with the fabric. Therefore, this study applied specific research techniques to address the limitations of such fabric defects: SoftPool was used to weaken the loss of the defective features through multiple maximum pooling operations; the attention mechanism induces the model to focus more on the defective regions, which positively impacted the feature extraction of the small defects; the introduction of DSC reduces the weight of the model and enables high-speed detection on the mobile side.

1) SOFTPOOL

The SPP structure used in YOLOv4 performs max-pool operations on the feature map at varying scales, with pooling kernels of 1×1, 5×5, $9, \times 9$, and 13×13, after which the four pooling results were fused. The pooling layers with varying kernel sizes can be used to more comprehensively acquire the contextual features.

Although MaxPool can effectively reduce the number of parameters, it loses a significant amount of information during the selection process. In addition, as MaxPool only retained the most evident features to represent the neighborhood, the fabric defects appeared similar to the fabrics, and the small defect information was readily lost after the MaxPool operation. In addition, the pooling operation used a method called AvgPool, which considered the average of the pixel region as the pooled result. To detect the fabric defects, MaxPool in SPP executes the risk of losing defect features owing to the high degree of fusion between the defects and fabrics. Although AvgPool considers all the features in the neighborhood and retains more background information and defect information, it reduces the intensity of the defect features in the region and neglects the evident defect features after averaging.

SoftPool is a new pooling method proposed by Zeiler and Fergus [35], which retains the pixels according to their weights in the feature map, which reasonably solves the problems of MaxPool of easy-to-lose information and the average pooling weakens the target feature intensity, expressed as follows:

$$\begin{cases} w_i = \dfrac{e^{a_i}}{\sum_{j \in R} e^{a_j}} \\ \tilde{a} = \sum_{i \in R} w_i * a_i \end{cases}, \tag{8}$$

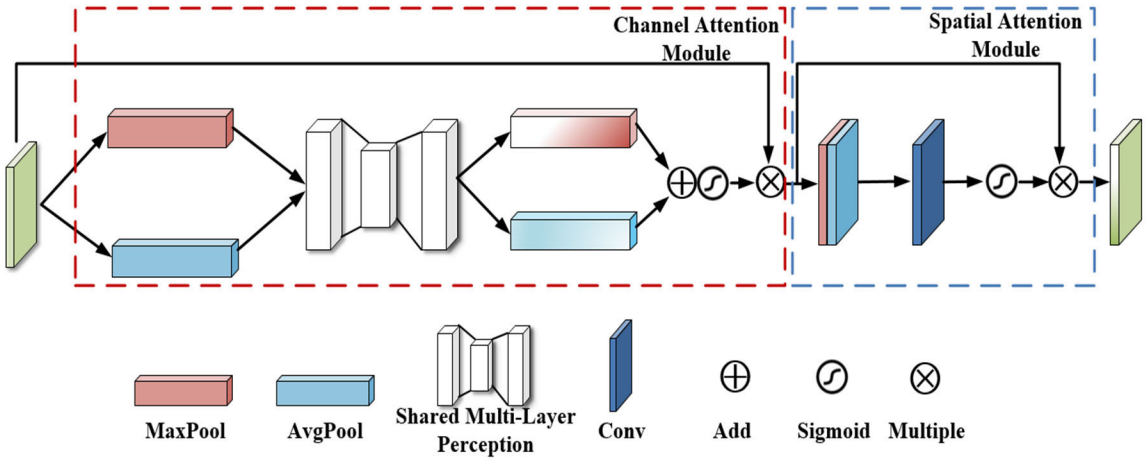


FIGURE 6. Network structure of CBAM.

where w_i represents the weight of the candidate region, a denotes the weight of the activation mapping. The SoftPool is advantageous because the number of features is reduced and the image information is adequately preserved; thus, the small defects are preserved during feature extraction.

2) ATTENTION MECHANISM

The concept of the attention mechanism originates from the human visual mechanism: when humans observe an object, they tend to focus on its most vital part. Driven by human visual mechanisms, scholars have gradually applied attention mechanisms to artificial intelligence, especially in the field of computer vision, and several attention mechanisms-based studies have been conducted to achieve target detection and semantic segmentation [36]. In deep learning, the common implementations of the attention mechanisms include the channel attention mechanisms, channel-space attention combination mechanisms, and efficient channel attention modules. Squeeze-and-excitation networks (SENet) [37], proposed by the WMW team, was the winner of the last ImageNet competition in 2017, and its principle is to generate a (1 × 1 × C) channel descriptor *via* squeeze compression operation for features with spatial dimension w × h. These channel descriptors were subjected to excitation by adaptively adjusting the weights of each channel to extract the key features.

Based on channel attention, Woo et al. [38] proposed a new convolutional attention module: convolutional-block-attention module (CBAM), which utilizes cross-channel information and spatial information to extract the information features that direct the focus of the model. CBAM has been widely used to improve the expressive power of CNNs. In 2020, Wang et al. proposed a new efficient channel attention (ECA) module based on SENet [39], and they proposed a local cross-channel interaction strategy without dimensionality reduction, which effectively avoided the effect of dimensionality reduction on the learning effect of channel attention. The mechanism is an efficient and lightweight module that obtained suitable results on the COCO and ImageNet datasets. In this study, after conducting comparative experiments and analyses of these three attention mechanisms and ultimately selected the CBAM as the attention module in the proposed YOLO-SCD model, which maximized the performance (refer to Section IV-C).

In CBAM, features were initially input to the channel attention module to generate a channel attention map, and the weights of the channels were obtained as output after the sigmoid function. Thereafter, the features were input to the spatial attention module, where the average pooling and maximum pooling operations were first applied along the channel axes and connected to generate a valid feature descriptor, as indicated in the structure diagram in Figure 6. The channel attention mechanism is a plug-and-play module, and its placement at several locations can produce various effects. In this research, CBAM was placed in the feature enhancement section because higher-level semantic information is more beneficial for guiding the model to learn the defect locations and focus more attention on the defective part. We visualized the feature maps of the incoming defective images at each stage of the network. Owing to numerous network layers, we captured an exemplary image with a defect category of holes and selected the shallow network portion of the feature extraction network (CSPDarknet53) and the high-level network portion for the feature visualization output. The results are depicted in Figure 7, wherein the feature map of the shallow network contains more redundant fabric background information, whereas the deeper feature map has more prominent defective fabric portions.

The results indicated that higher-level semantic information contains fewer fabric features with interference. For fabric defect detection, as several categories of defects were highly integrated with the fabric, incorporating the CBAM in the higher-level network position aids the model to focus

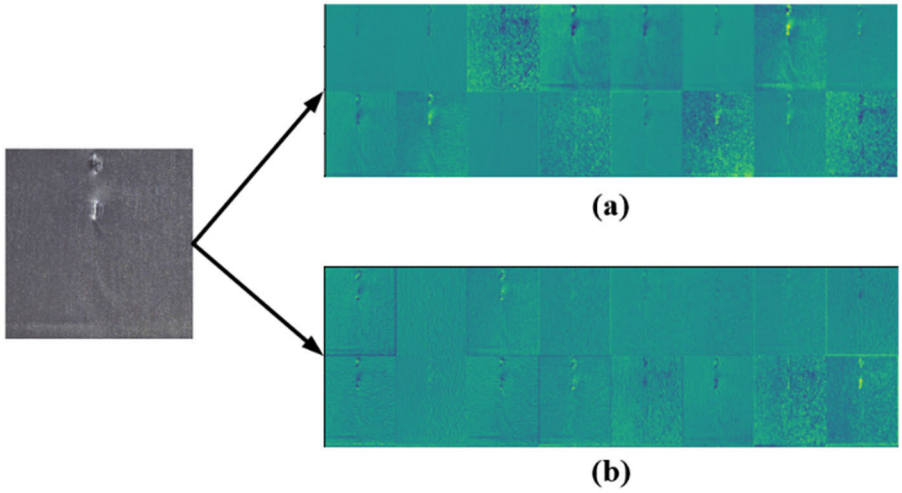


FIGURE 7. Visualization of feature maps in networks: (a) low-level network feature map; (b) high-level network feature map.

more on the defective portion and reduce the information interference.

3) DSC

Howard et al. [40] proposed the use of DSC to effectively reduce the size of models with standard convolution, which can be categorized into two main processes: depth-wise convolution and pointwise convolution. Depth-wise convolution includes a convolution kernel for a single channel, which was convolved by only one convolution kernel and produced exactly the same number of channels of the feature map as that in the input. Considering a color input image of size ($5 \times 5 \times 3$) as an example, depth-wise convolution first passes through the first convolution operation, and owing to the same number of convolution kernels as the channels in the previous layer, a three-channel image is generated with three feature maps after the operation. The channel-by-channel convolution operation independently convolved each channel of the input layer, which failed to effectively utilize the feature information of various channels at a given spatial location. Therefore, a point-by-point convolution is required to combine these feature maps.

The size of the convolution kernel for point-by-point convolution was $1 \times 1 \times$ M, where M denotes the number of channels in the previous layer. To generate a new feature map, the convolution operation combines the feature maps of the previous step in the depth direction with the weightage. As depicted in Figure 8, the input feature map size was $D_w \times D_h \times F$ and the output feature map size was $D_w \times D_h \times N$, where the size of the convolution kernel for the DSC is $D_k \times D_k$, and the computational volume for the depth-wise separable convolution can be determined using Eq. (9), and that for an ordinary convolution is expressed in (10).

$$C_{dep} = D_w \times D_h \times F \times D_k + D_w \times D_h \times F \times N \quad (9)$$

$$C_{com} = D_w \times D_h \times F \times N \times D_k \times D_k \quad (10)$$

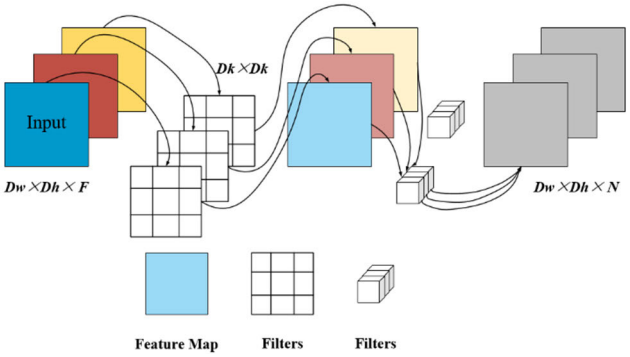


FIGURE 8. Display of DSC.

Considering a color input image of size $5 \times 5 \times 3$ as an example, the computation amount of ordinary convolution was 972, whereas the computation amount of DSC was only 351, representing a reduction of two-thirds of the computation amount. Thus, the depth-wise separable convolution method, which combines width and resolution factors, significantly reduced the computation and model size. As both the CSPDarknet53 and PANet of the YOLOv4 model leveraged numerous ordinary convolutions of 3×3, which increases the model size. Accordingly, we replaced the ordinary convolution in the feature enhancement component with DSC, and simultaneously, selected specific convolutions of the higher-level backbone network for replacement. This is because the images at the bottom layer contain more semantic information, and the use of DSC may yield partial feature loss (Section IV-E).

IV. EXPERIMENTS AND DISCUSSION

We performed data augmentation on the data using popular data augmentation methods. Thereafter, we describe the ablation experiments of the attention mechanism, analyze the soft

pooling, explore the performance variations after introducing DSC, and finally, verify that the final model of the improved YOLOv4 outperformed the unimproved baseline model as well as other widely used models.

A. EXPERIMENTAL DATASET AND MODEL PARAMETER SETTINGS

In this study, the used dataset was obtained from the Aliyun Tianchi Guangdong Industrial Manufacturing Competition (https://tianchi.aliyun.com/competition/entrance/231666/information), containing 5096 defective images with a resolution size of 2446 × 1000.

The experiments were conducted on a computer with an NVIDIA GeForce RTX 2080 Ti GPU with 16 GB RAM, using Python 3.7.4 and TensorFlow 2.2. The learning rate was initially set to 0.01 with a cosine annealing learning-rate adjustment strategy. The backbone of the network was frozen and the network was trained for 50 epochs using these parameters with pre-training weights of 200 epochs on the COCO dataset. Subsequently, the end-to-end network was thawed for 250 epochs and pre-training on the COCO dataset prevents the scattering of the weights. In the experiments, the average precision (mAP) and model inference speed (FPS) were used to evaluate the model performance, calculated as follows:

$$\begin{cases} P = \dfrac{TP}{TP + FP} \\ R = \dfrac{TP}{TP + FN} \\ AP_k = \dfrac{1}{N_k} \displaystyle\sum_{r_k \in R_k} P(r_k) \\ mAP = \dfrac{1}{N} \displaystyle\sum AP_k \end{cases}, \tag{11}$$

$$FPS = \frac{N_{image}}{T_{total}}, \tag{12}$$

where P denotes the precision rate, R indicates the recall rate, TP represents the number of images that correctly detected the cloth defects, FP represents the number of images that predicted non-defective points as defective points, FN indicates the number of defective points that were unidentified, AP represents the average precision of each class, N_k denotes the number of precision and recall for the k-th category, and $P(r_k)$ indicates the precision rate at the k-th category for the recall R_k. FPS denotes the number of images detected per second, which is a direct representation of the detection speed, where T_{total} denotes the total time and N_{image} indicates the total number of frames detected.

B. DATA ENHANCEMENT

Considering the diversity requirements for fabric defect recognition in actual factories, the categories of the defect images were mostly preserved and only certain categories with negligibly low number of defect images were discarded.

This is because a small number of training images was not conducive for model fitting and affected the detection accuracy of the model. Finally, the fabric defect dataset was reclassified into 12 categories: *hundred feet*, *broken warp*, *knotted head*, *hole*, *pulp spot*, *stain*, *abrasion mark*, *three filaments*, *loose warp*, *dense file*, *warp knot*, and *grain*, as illustrated in Figure 9. Among these defects, *hundred feet* and *dense file* were narrow and long, similar to *stitches*, whereas the *broken warp*, *knotted head*, and *warp knot* defects were similar and exhibited point-like distribution. As such, *holes*, *pulp spot*, and *stains* are large defects, accounting for a larger proportion of the fabric area. The number of pictures in the *broken warp* dataset was nonuniformly distributed, and only a small number of defects (*i.e.*, 200) existed in the four categories: *dense file*, *abrasion mark*, *warp knot*, and *grain*. To improve the generalization ability and robustness of the model, the defect categories with less images were selected for data enhancement.

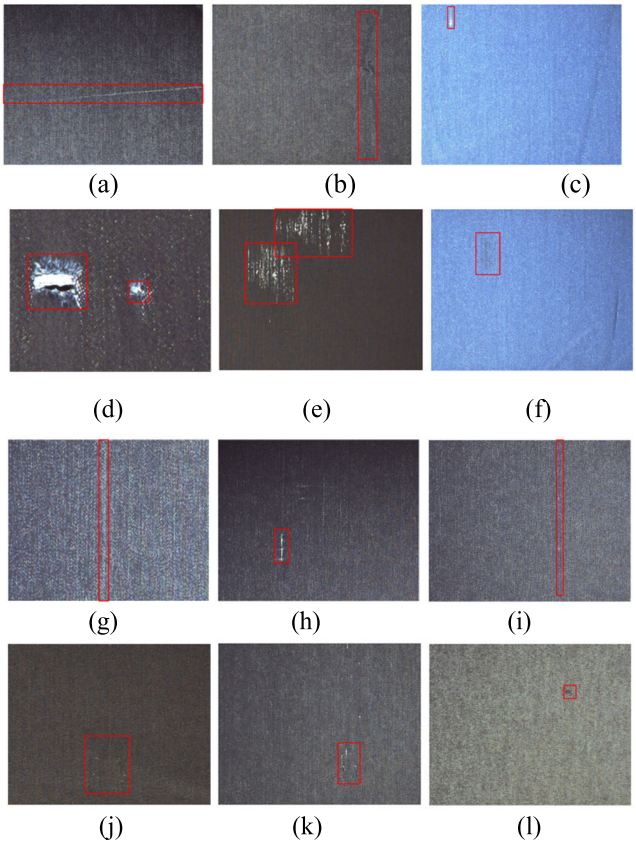


FIGURE 9. Images of the fabric dataset used in this paper.(a) HUNDRED FEET, (b)BROKEN WARP, (c)KNOTTED HEAD, (d)HOLE, (e)PULP SPOT, (f)STAIN, (g)ABRASION MARK, (h)THREE FILAMENTS, (i)LOOSE WARP, (j)DENSE FILE, (k)WARP KNOT, (l)GRAIN.

Common data-enhancement methods include translation, rotation, scaling, and histogram equalization. The first three methods primarily increased the number of defective datasets in various directions by altering the defective positions, whereas the histogram equalization method readjusted the contrast and brightness of the images to highlight the defective regions. Thus, the aforementioned four data-enhancement operations were applied to each defect category

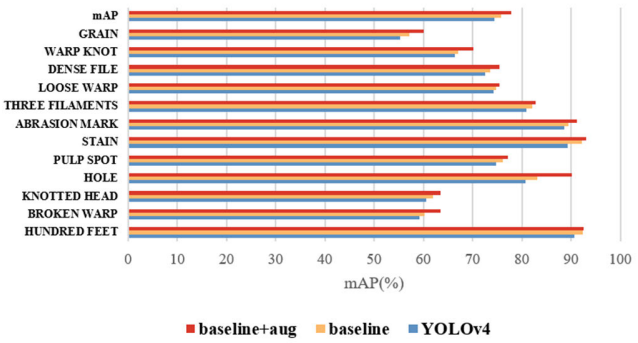


FIGURE 10. Comparison before and after data enhancement (baseline refers to YOLOv4 using improved PANet).

TABLE 1. Comparison of various attention mechanisms.

Model	MS(MB)	mAP(%)
YOLOv4	238	74.43
+SE	245	77.76
+CBAM	246	**78.84**
+ECA	245	78.12

TABLE 2. Comparison of the effects of different insertion positions of CBAM.

Model	Model Size(MB)	mAP(%)
YOLOv4	238	74.43
Backbone+CBAM	246	79.13
Neck+CBAM	246	79.30
Together+CBAM	245	**80.67**

TABLE 3. Comparison of ablation study with Cbam and SoftPool(baseline means YOLOv4 with improved PANet and after data enhancement).

Model	SoftPool	CBAM	mAP(%)
YOLOv4	-	-	74.43
baseline	√	-	76.26
baseline	-	√	80.67
baseline	√	√	**83.31**

with less defects and 7021 defective images were obtained (certain images contained multiple defects). As depicted in Figure 10, the data enhancement improved the most categories of defect images. Notably, defects such as *holes* were improved considerably, with a nearly 10% improvement in mAP relative to YOLOv4, whereas other background-like defects such as *pulp spots* were improved only to a certain extent. Thus, enhancing the contrast and brightness of the images aided the model to identify background-like defects. After obtaining the data-enhanced images, we repartitioned the dataset, and the ratio of the training set to the validation set and test set was 8:1:1.

C. ATTENTION MECHANISM ABLATION EXPERIMENTS

To investigate the effect of the attention mechanism module on the model, we conducted a series of ablation experiments. The results are summarized in Tables 1 and 2, wherein the baseline refers to the addition of a shallow branch output to the original YOLOv4 backbone network and the addition of a pyramidal layer to the PANet. First, we conducted comparative experiments for the three attention mechanism modules (SE, CBAM, and ECA), which were selected as the four branches from the backbone network output to the neck PANet location to compare the impact of these three methods on the model performance. As listed in Table 1, YOLOv4 incorporated with SE, CBAM, or ECA improved the model performance, implying that the introduction of the attention mechanism enables the model to dynamically adjust the feature weights. As this induces a beneficial effect on the model, the application of the CBAM mechanism delivered more improved performance than that of the other two schemes. The SE module compresses the global information to control the complexity of the module. However, the reduction of feature dimensionality causes side-effects in channel attention prediction and inefficiently captures the dependencies between all the channels. ECA avoids dimensionality reduction and efficiently captures cross-channel interactions. Nonetheless, it does not analyze the spatial dimensionality of the feature map. Unlike SE and ECA, CBAM considers the spatial location of features, with a more local scope, accounting for the details of the spatial location of fabric defects for both methods, SE and ECA, and complementing their deficiencies in spatial representation. Conclusively, we decided to embed CBAM in the model for subsequent experiments.

Second, as the attention mechanism module is a plug-and-play module, its insertion position results in varying influences of the model. We embedded the attention mechanism CBAM obtained from the previous comparison experiments, which was most applicable to this model, into the backbone and neck networks to comparatively experiment and obtain the most suitable embedding position for this model. As the backbone network CSPDarknet53 used in YOLOv4 exhibits five residual structures of varying dimensions, the positions between these five residual structures were selected to embed CBAM. In contrast, the neck segment—located higher in the network—was selected to embed the up-sampling and down-sampling positions in the spatial pyramid structure. Furthermore, we performed a comparative experiment by combining the above-mentioned two insertion positions: CBAM is embedded in the neck (higher part of the network) and the branch portion of the backbone network outputs to the neck. As observed in Table 2, incorporating CBAM into a higher level, *i.e.*, neck, yielded superior performance than inserting it into the backbone network model. This finding confirms the discussion in Section III-B2: the higher-level semantic information reduces the interference information such as fabric noise for the model and drives stronger focus

TABLE 4. Comparison of effect of various introductions positions of DSC.

Model	mAP (%)	MS (MB)	FPS (f/s)
baseline	83.31	245	9
Backbone+DSC	80.44	196	30
Neck+DSC	82.92	140	46
Together+DSC	80.25	98	51

TABLE 5. Comparison with state-of-the-art detectors in fabric dataset.

Model	mAP (%)	MS (MB)	FPS (f/s)
Faster-RCNN	79.83	522	5
SSD(VGG16)	65.90	98	15
YOLOv3	72.78	236	7
YOLOv4	74.43	238	9
YOLOv5(x)	76.10	330	28
YOLO-SCD	**82.92**	140	**46**

on the defective regions. As such, combining the two methods yielded more accurate results than integrating the first two methods. Because the feature pyramid structure of the neck extracted features that were considerably reused, incorporating the attention mechanism in the branch portion of the backbone network output to the neck refined the defective features used in the feature pyramid, thereby enabling the utilization of higher-level feature information of the network in PANet. Therefore, we decided to embed CBAM into the output branches of the backbone and neck networks.

D. ABLATION STUDY OF SOFTPOOL

The SPP structure is situated beyond the backbone network, acting as a bridge between the backbone and neck networks. Here, the MaxPool at multiple scales considerably increases the perceptual field of the model and separates the most significant contextual features of the fabric defects. Considering that the SPP structure may lose a certain amount of crucial information of the defects during the MaxPool process, and the SPP structure acts similarly to the attention mechanism, excessive extraction of salient information may result in the loss of information of certain defects with high fabric fusion or small defects during feature extraction. Accordingly, the Maxpool in SPP is replaced here with a relatively moderate SoftPool pooling, and its results are displayed in Table 3.

As observed from Table 3, using SoftPool in the SPP structure displayed superior performance than the MaxPool model, with a 1.83% improvement in its mAP. Although MaxPool can adequately extract the salient features in the fabric, its feature loss for small defects is irreversible. In contrast, SoftPool used softmax for weighted pooling and its gradient (SoftPool pooling) acted complementarily with the attention mechanism.

Thus, the visibility and prominence of the small defective features increased and was beneficial for fabric defect detection, depending on various gradient sizes, thereby maintaining the expressiveness of the features.

E. ABLATION STUDY OF DEPTH-WISE SEPARABLE CONVOLUTION

The YOLOv4 network uses numerous ordinary convolution operations for both the backbone and neck networks. In particular, the neck network uses five convolutions that increases the weight of the model. Consequently, the detection period of a defective image is incompatible with the application requirements of actual production in fabric and textile industries. Therefore, we introduced the DSC to reduce the model size and increase the detection speed. We replaced the 3×3 convolution used in the backbone and neck networks with the DSC to perform ablation experiments, and the results are summarized in Table 4. These results signify that DSC can effectively reduce the network size; however, the detection accuracy of the network is slightly reduced. Upon replacing only the 3×3 convolution used by the neck network, the highest mAP was obtained in comparison to other replacements. This finding indicated the richer underlying semantic information, and the more accurate fabric defect location information provided by the underlying feature map is highly beneficial for the model. Although the model size obtained with this method is not as small in as that derived from complete replacement, the model size was reduced by approximately 42% compared to the original YOLOv4. The detection speed was significantly improved, which was sufficient for actual factory inspection. Therefore, we decided to replace only the convolution of the neck with the DSC.

F. COMPARISON OF DETECTION RESULTS WITH OTHER MODELS

To thoroughly validate the model, we compared the YOLO-SCD with advanced target detectors that satisfy real-time detection requirements for industrial scenarios. To ensure fairness of the experimental results, reduce the computational cost, and ensure training accuracy, the above models were pretrained using the COCO dataset to acquire more compact weights. Specifically, the same training rounds were employed to train the other models using the data-enhanced fabric-defect dataset with the same size of anchor frame. The results of comparatively analyzing the proposed model with Faster-RCNN, SSD (VGG16 network selected by BackBone [41]), YOLOv3, YOLOv4, and YOLOv5 (xLarge) are summarized in Table 5. The results indicated that the developed model retained the high accuracy of YOLOv4 and displayed improved performance at a lower computational cost. Compared with the YOLO series model, the developed model exhibits increased ability to extract small fabric defects with the inclusion of the attention mechanism and Softpool. More importantly, the introduction of DSC considerably reduced the model size. Conversely, both YOLOv3 and YOLOv4 employ numerous normal convolutions that reduce the detection speed of the model, with fps of

TABLE 6. Comparison of single flaw detection results between other methods.

Model	AP(%)				nums	mAP (%)	MS (MB)	FPS (f/s)
	HOLE	STAIN	HUNDRED FEET	ABRASION MARK				
Tianchi top1	-	-	-	-	34	77.3	-	-
Qiang Liu[32]	89.6	86.36	69.96	-	4	86.5	-	21.2
Xin Luo[23]	96.18	85.5	90.97	86.48	17	82.2	-	42
Lu[24]	49.71	23.22	47.77	-	34	29.18	-	-
Bing Wei[42]	-	82.47	-	-	5	75.56	-	-
Ours	92.84	**91.95**	**92.47**	**96.1**	12	82.92	140	46

only 7 and 9, respectively. YOLOv5 employs a FOCUS structure that stacks various feature layers and offers faster detection speed than that of YOLOv3 and YOLOv4. However, its network model weight file is larger. Although YOLO-SCD is not as lightweight as SSD, its detection accuracy is 17.02% higher than that of SSD.

Using the Tianchi dataset, we compared the present results with those reported by other studies investigating fabric-defect detection. As observed from the table, the present experimental results were 5.62% more accurate than those reported by the first-place winner of the Tianchi industrial competition. For validation, we compared the current experimental results with those obtained by Liu et al. [23], [24], [32], who applied the fabric defect dataset of Tianchi. As observed, the present model displayed adequate results for individual defects, and the detection accuracy of large defects such as stain, hundred feet, and other difficult-to-detect defects such as abrasion mark were optimal compared to other studies. Notably, the developed model can detect 12 types of defects, which was less than the 34 types of defects considered in the original Tianchi dataset. However, these 12 defect types are sufficient compared to others who only detected four or five types of defects. Although Liu reported the highest average accuracy in detecting fabric defects, their study classified only four categories of fabric defects, namely, line, float, stain, and hole, which was insufficient for practical industrial applications. Second, we considered a series of targeted measures such as introducing CBAM to increase the adaptability of model toward various defects and improve the existing accuracy and detecting speed. The current results demonstrated the strong detection ability of the proposed model including adaptability to various types of fabric defect detection.

The final performance of the developed model for 12 classes of fabric defect detection is illustrated in Figure 11. As observed, the proposed model achieved substantial improvement in the difficult-to-identify classes (*knot*, *dense file*, and *abrasion mark*) and maintained comparable performance for easy-to-identify classes (*e.g.*, *pulp spots*, *hole*, *hundred feet*).

A graph of the final output of the model for defect detection is plotted in Figure 12, and the heat maps of the model detection are portrayed in Figure 13. As observed in

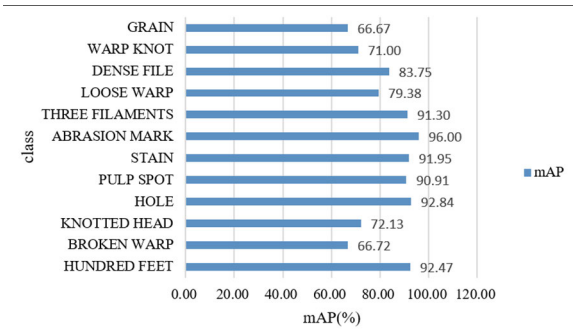


FIGURE 11. Detection mAP result of proposed model.

Figures 12 and 13, the proposed model extracted the features of small defects such as knotted head and grain, in addition to accurately predicting their types. These results signified the applicability of the improved model for detecting fabric defects.

G. EXTENSION TO STEEL PROFILE DATASET

We conducted extended experiments on the Vision-based_SIS_Steel surface defect dataset using the improved model. The Vision-based_SIS_Steel surface defect dataset is from Northeastern University, including six types of steel surface defects, namely, crazing, scratches, inclusion, patches, pitted-surface, and rolled-in_scale. The six types of steel defect images are presented in Figure 14, where pitted-surface and rolled-in_scale appeared as small patchy defects. The majority of these defects were only 6×6 pixels in size. As such, scratches and crazing are long defects, inclusions are similar to stains in fabric defects with irregular surfaces, and patches correspond to pulp spots in fabric defects. We segmented the dataset into 1485 images for training and 166 images for validation and followed the same experimental protocol as specified in Section IV. The corresponding results for the steel dataset are displayed in Figure 15. In particular, the proposed model achieved an accuracy of 89.66% for small pitted-surface defects, over 90% for narrow defects such as inclusions and scratches, and suitable performance for rolled-in_scale defects that blend with the surface. For the steel-surface defect dataset, we obtained an average accuracy of 87.02%. The results demonstrate that the proposed model can infer more advanced features for

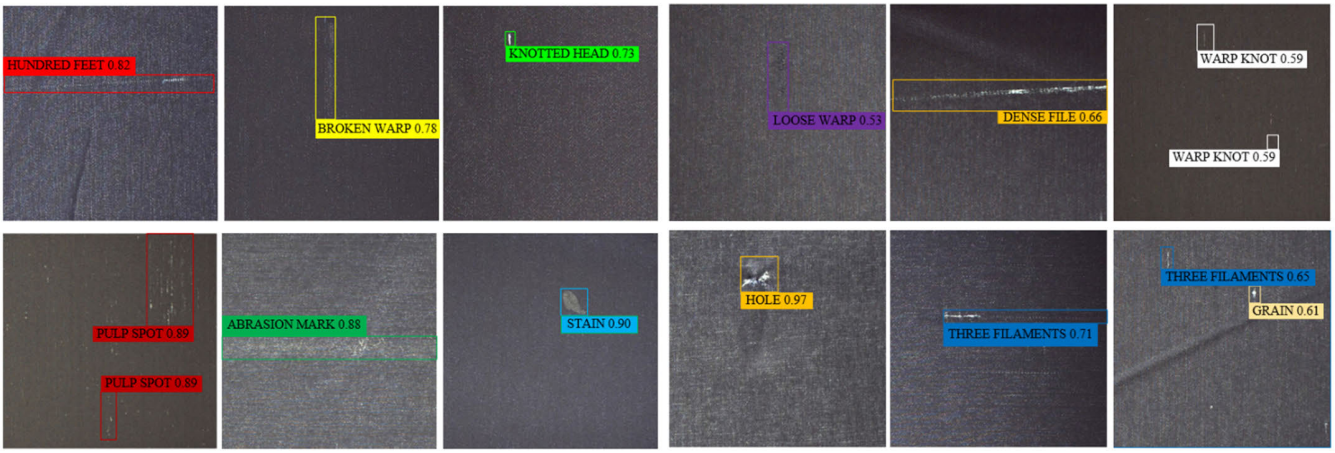


FIGURE 12. Detection result of proposed model.

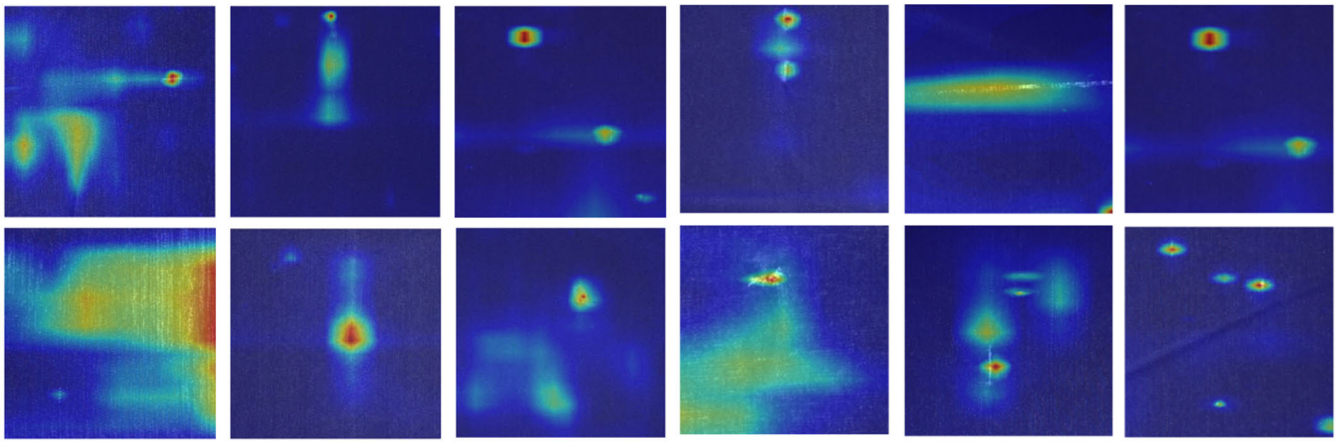


FIGURE 13. Heatmap obtained by model on 12 types of defects.

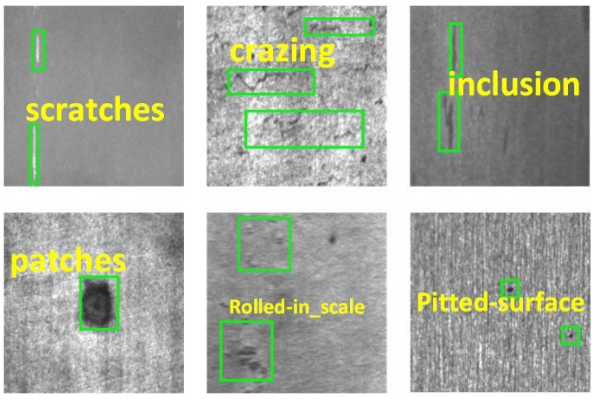


FIGURE 14. Images of Vision-based_SIS_Steel surface defect dataset.

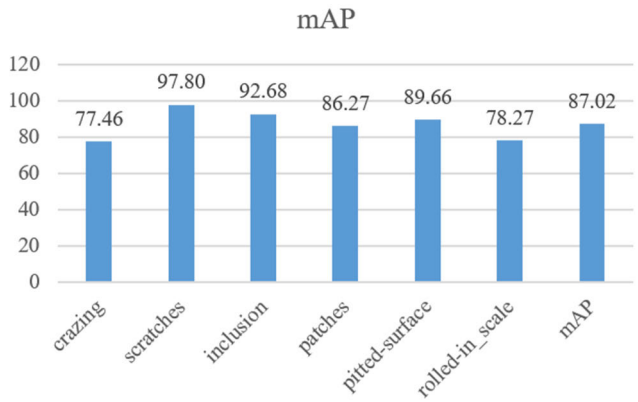


FIGURE 15. Detection results obtained by proposed model on various types of defects in steel dataset.

small-defect detection, thereby signifying its applicability as a reference for other similar situations.

H. ANALYSIS OF DETECTION FAILURE CASES

Although the proposed YOLO-SCD delivered appropriate results in fabric defect detection, it exhibited certain issues with its performance in detecting defect classes from overlapping fabric defects and similar defects. As depicted in Figure 16(a), highly fused defects such as *slurry spots* were correctly detected; however, the class of defects such as *abrasion marks* in the middle of the defective image were

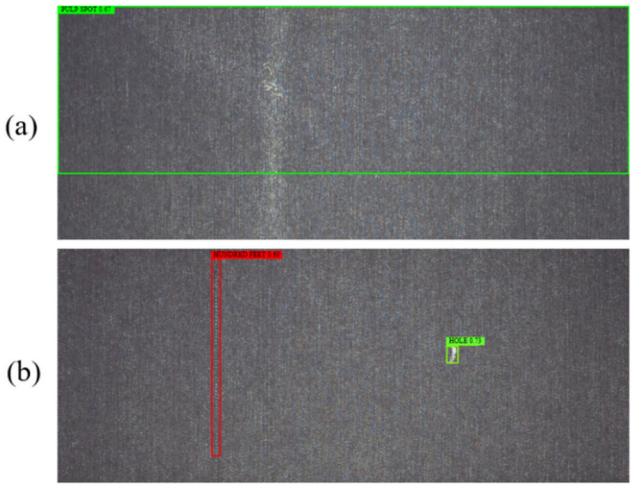


FIGURE 16. Model detection failure case pictures.

not detected and treated as part of the *pulp spot*. We believe that the characteristics of the abrasion mark in Figure 16(a) were extremely similar to those of the pulp spot, and they were misclassified if a single portion exhibited insufficient periodicity in a certain area. As depicted in Figure 16(b), both types of defects, hole and hundred feet, were correctly detected, and hundred feet was classified as difficult-to-see defects to the naked eye, thereby establishing the ability of the proposed model to detect defects that are highly integrated with the fabric. Nonetheless, the detection frame failed to accurately determine the lower edge of hundred feet, presumably because such defects (hundred feet) were excessively thin and long. Consequently, they were treated as lines in the fabric, and the accurate extraction of their features was restricted by their scattered appearance.

V. CONCLUSION

This study proposed a lightweight and efficient YOLO-SCD model to detect fabric defects by leveraging the attention mechanism. The model delivered improved results relative to YOLOv4, which was optimized for difficult-to-detect defects and improved the detection accuracy of small defects. First, we optimized YOLOv4 in terms of anchor frame and loss function to ensure its compatibility with fabric defect detection. Second, we improved the feature enhancement network, including proposing a new PANet structure, introducing soft pooling, and an attention mechanism to enhance the ability of the model to extract features. Finally, we introduced a DSC to reduce the model size and obtain a faster detection speed. Compared with the original YOLOv4, YOLO-SCD achieved an 8.49% improvement in mAP in terms of detection accuracy and considerably increased its detection speed. For instance, the AP values of *abrasion mark*, *stain*, and *broken warp* increased by 7.43%, 2.7%, and 7.51%, respectively. More importantly, the proposed model remained stable for relatively large defects (*e.g.*, *hole*) as well as accurately small defects (*e.g.*, *broken mark*). In particular, tt attained a speed of 46 fps, corresponding to an improvement of 37 increase fps compared with YOLOv4. Based on extensive research and a detailed analysis of fabric defects, the proposed network model is the most suitable for fabric defect detection. Notably, the proposed model may provide a solution for detecting small-size defects or those integrated with the surface.

Future research directions are focused on the following two aspects. First, the present model has been experimented only on a monochromatic background fabric dataset, and future research should improve its detection accuracy for multiple types of fabric defects in a complex pattern background. Second, the proposed model can be applied to other industrial datasets to improve the generalization capability of the model.

REFERENCES

[1] L. Jia, C. Chen, J. Liang, and Z. Hou, "Fabric defect inspection based on lattice segmentation and Gabor filtering," *Neurocomputing*, vol. 238, pp. 84–102, May 2017.
[2] Z. Kang, C. Yuan, and Q. Yang, "The fabric defect detection technology based on wavelet transform and neural network convergence," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Hailar, China, Aug. 2013, pp. 597–601.
[3] R. Meier, J. Uhlmann, and R. Leuenberger, "Uster Fabricscan—Automatic system for inspecting fabric quality," *Melliand China*, no. 3, pp. 48–51, Aug. 1999.
[4] G. Liu and F. Li, "Fabric defect detection based on low-rank decomposition with structural constraints," *Vis. Comput.*, vol. 38, no. 2, pp. 639–653, Feb. 2022.
[5] M. Dhivya and M. Renuka Devi, "Detection of structural defects in fabric parts using a novel edge detection method," *Comput. J.*, vol. 62, no. 7, pp. 1036–1043, Jul. 2019.
[6] M. Yang, P. Wu, J. Liu, and H. Feng, "MemSeg: A semi-supervised method for image surface defect detection using differences and commonalities," 2022, *arXiv:2205.00908*.
[7] R. M. L. N. Kumari, G. A. C. T. Bandara, and M. B. Dissanayake, "Sylvester matrix-based similarity estimation method for automation of defect detection in textile fabrics," *J. Sensors*, vol. 2021, pp. 1–11, Jan. 2021.
[8] S.-Y. Chen, Y.-C. Cheng, W.-L. Yang, and M.-Y. Wang, "Surface defect detection of wet-blue leather using hyperspectral imaging," *IEEE Access*, vol. 9, pp. 127685–127702, 2021.
[9] K. L. Mak, P. Peng, and K. F. C. Yiu, "Fabric defect detection using morphological filters," *Image Vis. Comput.*, vol. 27, no. 10, pp. 1585–1592, 2009.
[10] Y. Huang and Z. Xiang, "RPDNet: Automatic fabric defect detection based on a convolutional neural network and repeated pattern analysis," *Sensors*, vol. 22, no. 16, p. 6226, Aug. 2022.
[11] J. Lin, N. Wang, H. Zhu, X. Zhang, and X. Zheng, "Fabric defect detection based on multi-input neural network," in *Proc. 27th Int. Conf. Mechatronics Mach. Vis. Pract.*, Nov. 2021, pp. 458–463.
[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
[13] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NIPS*, vol. 28, 2016, pp. 91–99.
[15] H. Zhou, B. Jang, Y. Chen, and D. Troendle, "Exploring faster RCNN for fabric defect detection," in *Proc. 3rd Int. Conf. Artif. Intell. Industries*, Sep. 2020, pp. 52–55.
[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
[17] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271, doi: 10.1109/CVPR.2017.690.
[18] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[19] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.
[21] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6569–6578, doi: 10.1109/ICCV.2019.00667.
[22] H.-W. Zhang, L.-J. Zhang, P.-F. Li, and D. Gu, "Yarn-dyed fabric defect detection with YOLOV2 based on deep convolution neural networks," in *Proc. IEEE 7th Data Driven Control Learn. Syst. Conf. (DDCLS)*, May 2018, pp. 170–174.
[23] X. Luo, Z. Cheng, Q. Ni, R. Tao, and Y. Shi, "Defect detection algorithm for fabric based on deformable convolutional network," *Textile Res. J.*, vol. 2022, Dec. 2022, Art. no. 004051752211437.
[24] Z. Lu, Y. Zhang, H. Xu, and H. Chen, "Fabric defect detection via a spatial cloze strategy," *Textile Res. J.*, vol. 2022, Nov. 2022, Art. no. 00405175221135205.
[25] F. Li, L. Yuan, K. Zhang, and W. Li, "A defect detection method for unpatterned fabric based on multidirectional binary patterns and the gray-level co-occurrence matrix," *Textile Res. J.*, vol. 90, nos. 7–8, pp. 776–796, Apr. 2020.
[26] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[27] J. Yang, X. Fu, Y. Hu, Y. Huang, X. Ding, and J. Paisley, "PanNet: A deep network architecture for pan-sharpening," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1753–1761.
[28] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 202–211.
[29] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 390–391. [Online]. Available: https://openaccess.thecvf.com/content_CVPRW_2020/html/w28/Wang_CSPNet_A_New_Backbone_That_Can_Enhance_Learning_Capability_of_CVPRW_2020_paper.html
[30] K. He, X. Zhang, J. Sun, and S. Ren, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Jan. 2015.
[31] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768, doi: 10.1109/CVPR.2018.00913.
[32] Q. Liu, C. Wang, Y. Li, M. Gao, and J. Li, "A fabric defect detection method based on deep learning," *IEEE Access*, vol. 10, pp. 4284–4296, 2022.
[33] 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 *Proc. AAAI*, 2020, pp. 12993–13000.
[34] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.
[35] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," 2013, *arXiv:1301.3557*.
[36] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," *Nature Rev. Neurosci.*, vol. 3, no. 3, p. 201, 2002.
[37] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Jun. 2020.
[38] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19, doi: 10.1007/978-3-030-01234-2_1.
[39] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11534–11542, doi: 10.1109/CVPR42600.2020.01155.
[40] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[41] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[42] B. Wei, L. Gao, X.-S. Tang, and K. Hao, "Multi-class object learning with application to fabric defects detection," *AATCC J. Res.*, vol. 8, no. 1, pp. 165–172, Sep. 2021.

XIN LUO received the M.S. and Ph.D. degrees in engineering from Tokushima University, Japan, in 2004 and 2007, respectively. He is currently an Associate Professor with the School of Computer Science and Technology, Donghua University, Shanghai, China. His major research interests include the analysis, recognition and retrieval of images, videos, and audio. He is a member of CAAI and CCF.

QING NI received the B.S. degree in computer technology from the Anhui University of Finance and Economics, Bengbu, China, in 2021. She is currently pursuing the M.S. degree in computer technology with the School of Computer Science and Technology, Donghua University. Her research interests include computer vision, image recognition, and deep learning.

RAN TAO received the B.S. degree in computer application and the M.S. degree in computer technology from Donghua University, Shanghai, China, in 1998 and 2007, respectively. Since 1998, he has been with Donghua University, where he was a Senior Engineer with the School of Computer Science and Technology, in 2009. From 2014 to 2015, he was a Visiting Scholar with Old Dominion University, Norfolk, VA, USA. He has coauthored three books, more than 30 articles, and four patents. His research interests include information systems, cloud computing, and data mining.

YOUQUN SHI received the M.S. degree in computer application technology and the Ph.D. degree in control theory and control engineering from the China University of Mining and Technology, Jiangsu, China, in 1995 and 1998, respectively.

In 2002, he completed his postdoctoral research with Tongji University, Shanghai, China. In 2006, he joined the School of Computer Science and Technology, Donghua University, Shanghai. From 2014 to 2015, he was a Senior Visiting Scholar with Reutlingen University, Germany, and Purdue University, West Lafayette, IN, USA. His major research interests include network computing, artificial intelligence, and industrial internet technology. He is a member of the Academic Work Committee. He currently serves as the Director for the China Association of Artificial Intelligence, the Industrial Internet Alliance of the China National Textile Industry Council, and the Shanghai Computer Society, and the Vice-Chairperson for the Shanghai Computer Science and Technology Teaching Instruction Committee.