Received 6 June 2024, accepted 29 June 2024, date of publication 2 July 2024, date of current version 10 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3422244

RESEARCH ARTICLE

Surface Defect Detection of Steel Plate Based on SKS-YOLO

SHIYANG ZHOU, SIMING AO, ZHIYING YANG, AND HUAIGUANG LIU
Key Laboratory of Metallurgical Equipment and Control Technology, Ministry of Education, Wuhan University of Science and Technology, Wuhan 430081, China
Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan 430081, China
Precision Manufacturing Institute, Wuhan University of Science and Technology, Wuhan 430081, China

Corresponding author: Shiyang Zhou (zhoushiyang@wust.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 51805386.

ABSTRACT During the production process of steel plate, surface defect detection is crucial for high-quality products. For the existing defect detection method based on machine vision, there are various types of problems, such as large model calculations, low detection accuracy and difficulties of recognizing small defect targets. To reduce and solve these issues, the paper proposes a new defect detection model, simplified kernel and squeeze on a you only look once network (SKS-YOLO), which can achieve rapid and effective defect detection on steel plate. Firstly, it adopts EfficientNetv2 as the backbone, significantly reducing model calculations and accelerating training speed while maintaining accuracy. Subsequently, the atrous spatial pyramid pooling (ASPP) module is utilized to obtain a larger receptive field, extracting more feature information from surface defects. The integration of the squeeze excitation network (SE-Net) attention mechanism enhances capabilities of feature extraction furtherly. Then, the K-means algorithm is applied to cluster and obtain more suitable anchor frames for defect targets. It not only increases the number of positive samples, but also expedites model convergence. Finally, the loss function of simplified intersection over union (SIoU) is used to enhance the ability of model to locate and detect surface defect targets. The experimental results show that the mean average precision (mAP) is 89.40% at a detection speed of 55 frames per second (FPS), which is better than the state-of-the-art (SOTA) detection models.

INDEX TERMS Surface defect detection, YOLO, attention mechanism, anchor frames, loss function.

I. INTRODUCTION

During the production of industry, surface defects detection is an indispensable process to control product quality. The quality of product directly impacts production efficiency, and reduces a waste of raw materials. Moreover, product quality is closely tied to the factory's reputation and market share. Surface defects detection has consequently become a significant research field. Initially, surface defects inspection relied on manual methods. However, due to the limited energy and attention of human, this approach not only yields low accuracy but also consumes considerable time [1]. With the development of machine learning and artificial intelligence, the image-based surface defect inspection is widely applied to various industrial scenarios, instead of manual inspection. Current methods for detecting surface defects can be broadly categorized into traditional methods and deep learning methods. There are several differences between these two methods. Traditional methods extract defect features through image processing, image analysis, and so on. In contrast, the deep learning methods learn defect features from a specified number of defect samples and then automatically extracts these features [2]. Firstly, Traditional defect detection methods extract image features through pre-processing techniques, such as histogram equalization, grayscale binarization and filtering and denoising. Subsequently, the classification and detection of defects are accomplished by using morphology, Fourier transforms, Gabor transforms and various machine learning techniques. For example, Prasitmeeboon et al. [3] employed a combination of color histogram and support vector machine (SVM) for detecting particle board defects and utilized thresholding

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

and smoothing techniques to localize the faults precisely. Chang et al. [4] conducted defect detection to use a combination of polar coordinate transform, Hough circle transform, weighted Sobel filter, and SVM. Wang and Zuo [5] used Fourier transform and Hough transform to reconstruct the magnet surface image. They obtained defect information by comparing the grayscale difference between the reconstructed image and the original image for detecting defects. Wen et al. [6] proposed a method which fuses 2D gray information and 3D depth information for solving the detection in the complex areas of steel plate surface. Wang et al. [7] published a strip surface defect detection algorithm based on a straightforward guide template, which accurately locates defects from pseudo-defects and random arrangements of gray levels in the background. Hou et al. [8] found a multi-kernel vector machine method based on second-order cone programming optimization to enhance accuracy and reduce running time. All of these conventional detection methods require feature extraction and redefined thresholds to identify the presence of defects. However, they are effective for a specified defect class, exhibiting inferior adaptability and insufficient generalization ability.

With the rapid development of deep learning techniques, numerous outstanding target detection algorithms have emerged. Comparing with traditional methods, deep learning approaches learn data-driven parameters to extract features automatically, and feed them into subsequent networks for classification and localization. These methods decrease the need for the complex process of manual designing algorithms, and demonstrate remarkable robustness and accuracy. Generally speaking, these methods can be classified into two types: two-stage target detection method and one-stage target detection method. The two-stage target detection method involves classifying each proposed region of interest using convolutional neural network (CNN) to determine its corresponding object category. There are lots of examples, such as region-CNN (R-CNN) [9], Fast R-CNN [10], Faster R-CNN [11], etc. These methods improve detection accuracy at the expense of detection speed. Zhao et al. [12] enhanced the traditional Faster R-CNN by reconstructing the network structure through multi-scale feature fusion and deformable convolution network, which has finished 75.2% mAP. Cha et al. [13] utilized Faster R-CNN for the detection of concrete cracks and steel corrosion defects. Su et al. [14] devised a complementary attention network for leveraging the benefits of spatial location features and channel features to suppress background noise features. They integrated this network into Faster R-CNN for the detection of solar cell in electroluminescence images. Zhang et al. [15] raised a strip steel surface defect detection algorithm based on Faster R-CNN so as to tackle the issues of low automation, slow detection speed and low accuracy. Although the two-stage target detection method yields satisfactory results in terms of detection accuracy, it is proved to be a huge challenge because of efficiency concerns, particularly in real-time industrial defect detection. Consequently, the one-stage target detection method has garnered more attention. The one-stage target detection method treats target detection as a regression problem, utilizing CNN to determine the position by variety of the bounding box. For examples, YOLO [16], [17], [18], [19], single shot multibox detector (SSD) [20], RetinaNet [21], etc., which simplify network designs by employing a single network for target classification and localization. These approaches meaningfully increase detection speed. Yin et al. [22] used YOLOv3 to detect sewer pipe defects and achieved 85.37% mAP. Zhang et al. [23] enhanced the original YOLOv3 by citing a new transfer learning method for detecting concrete bridge defects, and brought about a 13% performance improvement. Yu et al. [24] proposed an efficient stepped pyramidal network which is characterized by fusing multi-scale features to improve the accuracy of small object detection. Wang and Cheung [25] incorporated count loss which can detect defects in the additive manufacturing process. Zhang et al. [26] combined the coordinate attention mechanism and the context feature enhancement module with YOLOv5 to maximize the performance of small target detection.

For the small set of defect samples and small size defects, above deep learning-based approaches cause unsatisfactory results. Additionally, real-time detection of steel plate defects encounters lots of challenges that large network structure. As a result, there is an urgent requirement to strengthen capacity of model, which can detect a broader range of defects and increase accuracy for small defects. Based on above analysis, we propose SKS-YOLO method include the module of ASPP [27] and SE-Net [28], especially for detecting small and long defects that are typically challenging. The main contributions of the paper can be summarized as follows.

1) To reduce the computational load of the model, we utilized the more lightweight EfficientNetv2 [29] to extract defect features. To tackle the fixed network receptive field, we employ ASPP module to expand the receptive field of feature extraction network. It allows the model to extract defect information more effectively within the suitable receptive field.
2) The SE-Net attention module can boost the performance of the network by learning the weight of each channel adaptively. The K-means method is executed to refine the anchor frames for training, enhancing the ability of model to detect steel plate defects with large-scale fluctuations. Besides, the SIoU loss function [30] was carried out to raise training efficiency of model and stability of bounding box predictions.

The paper is organized as follows: Section I explains the motivation and contribution of our study, Section II discusses related works, Section III describes the basic framework and methodology, and Section IV explains the experiments and summarizes the results. Finally, the conclusion is argued in Section V.

II. RELATED WORK

Machine vision has developed quickly in recent years, particularly in the fields of image classification, face recognition, industrial manufacturing and object detection [31]. It shows the advantages of both stability and efficiency [32], [33], [34], [35]. Zhang et al. [36] used a lightweight YOLOv5 method to detect surface defect of strip steel, reaching the trade-off between accuracy and speed. While the detection of large targets is gradually enhancing, there is still much works to be done for the detection of small targets [37]. Li et al. [38] put forward YOLOv5 to solve the numerous types, small-sized and unbalanced samples of fabric defects. Li et al. [39] incorporated attention mechanisms and receptive fields in YOLOv5 to solve the poor detection of small targets. Xin and Zhang [40] presented an improved bidirectional feature pyramid network to heighten the feature extraction ability and tackle incomplete feature fusion in the YOLOv5. Zhang and Wen [41] solved insufficient detection capability, long model inference time and low recognition accuracy for small targets and long strip defects. Guo et al. [42] displayed confused defect categories, substantial defect scale changes and poor detection results for small defects. Shi et al. [43] put forward an improved YOLOv5 algorithm to enhance the accuracy and efficiency of defect detection on steel surfaces, especially for small targets. Zhang et al. [36] demonstrated a lightweight YOLOv5 to solve the problems such as target loss, false alarms, large computation and the imbalance between detection accuracy and speed. Qu et al. [44] tackled the troubles encountered in the detection of tiny targets.

YOLOv5s is a lightweight deep learning model that comprises of backbone, neck, and head networks. The backbone network extracts feature by Conv, Focus, C3, and spatial pyramid pooling (SPP) modules. The Conv module performs convolution operations and strengthens model convergence through normalization operations, improving target detection precision. The Focus module conducts slice operations on images to weaken computational overhead and increase speed. The C3 module contains three standard convolutional layers and multiple Bottleneck modules to fuse features from different scales and accelerate network runtime. The SPP module down-samples the input image in parallel through multiple maximum pooling layers of different sizes. The receptive field is expanded by the aggregated information, retaining more context features and edge information. The neck network serves as a feature fusion network through combining feature pyramid network and path aggregation network to accomplish multi-scale feature fusion of defect images. The complete intersection over union (CIoU) loss function and the non-maximum suppression algorithm are used to detect in the prediction network.

The YOLOv5s loss function consists of three parts: confidence loss L_{conf}, classification loss L_{cls} and location loss L_{box}, is shown as follows:

The L_{conf} is as follows:

$$L = L_{conf} + L_{cls} + L_{box} \tag{1}$$

$$\begin{aligned} L_{conf} = &-\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{obj}[\overline{C}_i^j \log C_i^j \\ &+ (1-\overline{C}_i^j)\log(1-C_i^j)] \\ &- \sum\nolimits_{i=0}^{S^2}\sum\nolimits_{j=0}^{B} I_{ij}^{noobj}[\overline{C}_i^j \log C_i^j \\ &+ (1-\overline{C}_i^j)\log(1-C_i^j)] \end{aligned} \tag{2}$$

where, S^2 is the number of grids; B is the number of anchor frames; I_{ij}^{obj} indicates whether there is a target at i and j, and it is 1 if there is a target, otherwise is 0; I_{ij}^{noobj} indicates whether the prediction box contains the target, taking 0 if it does, otherwise is 1; C_i^j is predicted confidence and $\bar{C}_i^j$ is ground truth confidence.

The L_{cls} is as follows:

$$\begin{aligned} L_{cls} = &-\sum_{i=0}^{S^2} I_{ij}^{obj}\{\overline{P}_i^j(c)\log[P_i^j(c)] \\ &+ [1-\overline{P}_i^j(c)]\log[1-P_i^j(c)]\} \end{aligned} \tag{3}$$

where, c is the type of the detection target; $P_i^j(c)$ is the predicted probability; $\overline{P}_i^j(c)$ is the ground truth probability that the target belongs to category c.

The L_{box} is as follows:

$$L_{box} = 1 - \mathrm{IoU} + \frac{\rho^2\left(B, B^{gt}\right)}{m^2} + \alpha\upsilon \tag{4}$$

where, IoU represents the intersection ratio of the predicted bounding box and the ground truth; (B, B^{gt}) is the center point coordinates of the predicted bounding box and the ground truth; $\rho^2(B, B^{gt})$ denotes the Euclidean distance; m signifies the diagonal distance of the minimum circumscribed rectangle containing both the predicted bounding box and the ground truth; α is the weight coefficient; υ is the length-width ratio consistency parameter.

$$\alpha = \frac{\upsilon}{1 - \mathrm{IoU} + \upsilon} \tag{5}$$

$$\upsilon = \frac{4}{\pi^2}\left(\arctan\frac{\omega_{gt}}{h_{gt}} - \arctan\frac{\omega}{h}\right)^2 \tag{6}$$

where, ω_{gt} and h_{gt} are width and height of the ground truth; ω and h are width and height of the predicted bounding box, respectively.

III. METHODOLOGY

In this section, we provide a detailed description of the proposed SKS-YOLO method. The network structure is depicted in Fig. 1. It mainly includes backbone, neck and head network. We adopt EfficientNetv2 as the backbone network for feature extraction to lessen the computational complexity. The feature extraction module of ASPP is incorporated into the backbone network to obtain varying receptive fields for various sizes of defects, especially for small defects. We integrate SE-Net modules to the backbone and neck network

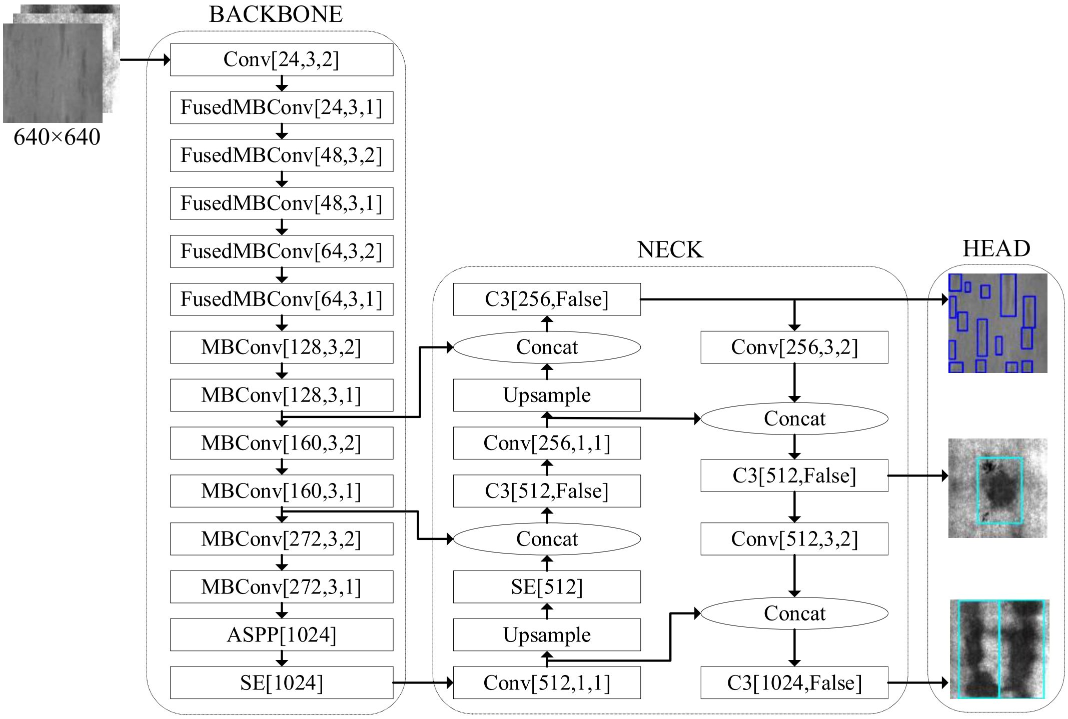


FIGURE 1. Network structure of the proposed SKS-YOLO.

further to strengthen concentration on important feature channels and capture correlations between different defects for the network. It can improve discrimination ability of small defects.

A. ALGORITHM OF SKS-YOLO

In the proposed SKS-YOLO method, feature extraction part is employed to extract the most crucial and representative information from raw defect images. And feature fusion is used to aggregate multi-scale features at different resolutions. Feature pyramid network (FPN) is used to transmit deep semantic features to enhance semantic expression at multiple scales. Path aggregation network (PAN) and global average pooling (GAP) is used to transmit shallow position information to enhance the location ability on multiple scales. The algorithm is shown in Table 1, where, Conv (input, stride, channel); {C3, C4, C5} are features of the backbone; {P3, P4, P5} are the levels of the FPN; {N3, N4, N5} are the levels of the PAN; {N3_H, N4_H, N5_H} are the features of the head.

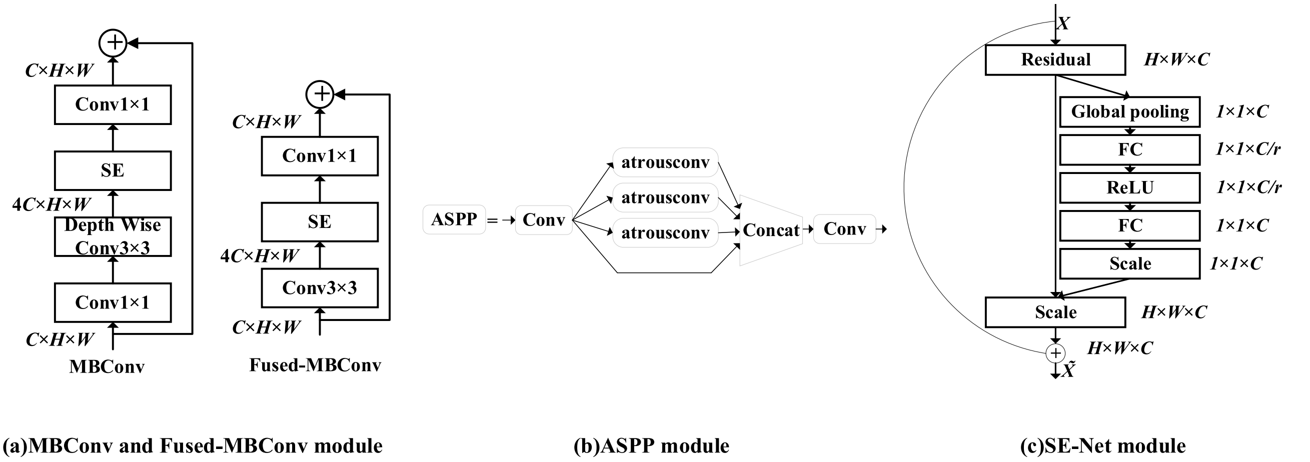
B. EfficientNetv2 MODULE

As the cross stage partial DarkNet53 (CSP-DarkNet53) of feature extraction network in the original YOLOv5s stack a mass of residual blocks [35]. It increases the number of parameters and leads to a slower detection speed. Most lightweight feature extraction networks rely on depth-wise separable convolution to decrease parameters, it causes low detection accuracy. In the paper, EfficientNetv2 is chosen as the feature extraction module in backbone network to achieve faster convergency speed with compromising accuracy.

TABLE 1. Algorithm of the proposed SKS-YOLO.

```
Algorithm 1  SKS-YOLO
Input: I is an 640×640 surface defect image
Output: Feature map {N3_H, N4_H, N5_H}
C3 = Conv (I, 1, 128)
C4 = Conv (I, 1, 160)
C5_1 = Conv (I, 1, 272)
C5_2 = Conv (Concat (Conv (Conv (GAP (C5_1)))))
C5 = Multiply (C5_2, Conv (Sigmoid (GAP(C5_2))))
P3 = Conv (C3, 1, 128)
P4 = Conv (C4, 1, 160)
P5 = Conv (C5, 1, 512)
P4_U = Upsample (P5)
P4_S = Multiply (P4_U, Conv (Sigmoid (GAP (P4_U))))
P4_F = Concat (P4, P4_S)
P3_U = Upsample (P4_F)
N3 = Concat (P4, P3_U)
N4 = Concat (P4_F, N3)
N5 = Concat (P5, N4)
N3_H = Concat(Conv (N3))
N4_H = Concat (Conv (N4))
N5_H = Concat (Conv (N5))
return {N3_H, N4_H, N5_H}
```

In EfficientNetv2, the memory access overhead is minimized by choosing for a smaller expansion ratio and eliminating the last module of EfficientNetv1 with a step



(a)MBConv and Fused-MBConv module

(b)ASPP module

(c)SE-Net module

FIGURE 2. Schematics diagram of different modules.

size of 1. Besides, a progressive training strategy is utilized to acquire basic capabilities for the network dramatically. A smaller training size and milder regularization method are adopted in the early stages of training. Subsequently, the regularization method is strengthened as the image size increases gradually. The progressive learning strategy is formalized into a formula which makes it possible to determine the training size and regularization intensity at different training stages. It trains the model for N/Msteps with image size S_i and regularization R_i.It is expressed as follows:

$$\begin{cases} S_i = S_0 + (S_e - S_0)\,\frac{i}{M-1} \\ R_i = R_0 + (R_e - R_0)\,\frac{i}{M-1} \end{cases} \tag{7}$$

where, S_0 is the initial image size; S_e is the final image size; R_0 is the initial regularization parameter; R_e is the final regularization parameter; N is the total number of steps; M is the total number of training sessions; i is the index of the current phase, ranging from 0 to M-1.

Therefore, we adopt the mobile inverted residual bottleneck convolutional (MBConv) module and the fused mobile inverted residual bottleneck convolutional (Fused-MBConv) module. The MBConv module divides the feature map of the upper input into two parts and merges them through cross-stage hierarchical merging. It speeds up detection and reduces repeated gradient information. The Fused-MBConv module has enhanced performance for the MBConv structure. It transforms a common convolution with a 1×1, 3×3 kernel containing batch normalization (BN) and sigmoid linear unit (SiLU) activation functions into a convolution with 3×3 kernel. The structures of MBConv, Fused-MBConv are illustrated in Fig. 2.

Suppose the entire training process consists of N steps, with a target training size (final training scale) denoted as S_e and a regularization list (final regularization strength) represented by $\Phi_e = \{\Phi_e^k\}$. where k signifies various regularization methods. The initial training size S_0 and initial regularization strength $\Phi_0 = \{\Phi_0^k\}$ are initialized. The entire training process is divided into M stages. For the i stage ($1\leq\ i\ \leq\ M$), the training size of model is S_i and the regularization strength is $\Phi_i = \{\Phi_i^k\}$. Linear interpolation is employed between different stages for incremental adjustments. By means of incorporating the Fused-MBConv module into the shallow layer of the network, enhancing progressive learning and adjusting (to adjust) the regularization method based on the training image size dynamically. In the end, the detection speed and accuracy are strengthened. EfficientNetv2 as a new backbone network is constructed to reduce the computational complexity of the model on the basis of these improvements.

The experiment is conducted with different lightweight networks on the NEU-DET dataset [45], and the experimental results are presented in Table 2, where, PAR represents the number of model parameters, GFLOPS represents giga floating-point operations per second. It demonstrates that GhostNetv2 [35] reduces parameters but decreases speed that compared with CSP-DarkNet53. In contrast, ShuffleNetv2 [35] significantly diminishes parameters and achieves faster speed during actual operations. EfficientNetv2 is superior to the CSP-DarkNet53 and ShuffleNetv2 in detection accuracy by 1.1% and 4.5%, respectively. The speed only decreases 3 FPS compared with CSP-DarkNet53. It is sufficient for meeting the real-time requirements of industrial defect detection.

TABLE 2. Comparison of different feature extraction module in backbone network.

Backbone	mAP (%)	PAR (10^6)	GFLOPS	FPS
CSP-DarkNet53	85.6	7.28	17.2	66
EfficientNetv2	86.7	5.60	5.6	63
GhostNetv2	81.8	5.83	7.2	50
ShuffleNetv2	82.3	3.80	8	71

C. ASPP MODULE

The SPP module in YOLOv5s is used to extract global features by applying maximum pooling operations to the feature maps of input through filter kernels with sizes of 5, 9 and 13. Although it addresses the multi-scale problem in target detection, it tends to lose some features related to small defect targets. Considering that atrous convolution can provide a larger receptive field than traditional convolution, the ASPP module on dilated convolution is adopted in the paper. The ASPP module utilizes 3 × 3 dilated convolution kernels with expansion factors of 6, 12 and 18, respectively. It is designed to further extract features related to small defects on the steel plate, emphasizing the key characteristics of the target. The structure of the ASPP module is depicted in Fig. 2.

D. SE-NET MODULE

By incorporating the attention mechanism, the model can gain the capability of learning automatically and focus on essential information from the input selectively. It enhances the performance and generalization ability of the model. The SE-Net attention module is integrated into the neck network, enhancing the model's perception of defect location information and improving detection accuracy, especially for small defect targets. The SE-Net attention module is illustrated in Fig. 2.

The SE-Net module adopts both squeeze and excitation operations. In the squeeze stage, it compresses the output feature map of the convolutional layer into a feature vector through a global average pooling operation. In the excitation stage, a weight vector for each channel is learned by adopting a fully connected layer and a nonlinear activation function. This weight vector is applied to each channel on the original feature map, giving different weight for the features of different channels effectively. By incorporating squeeze and excitation operations, the performance of network is enhanced by learning the weight of each channel adaptively. At the same time, the SE-Net module learns the weight of each channel automatically, which can adjust the significance of features in each channel for subsequent network layers dynamically. Consequently, the network can allocate more attention to crucial feature channels, thus improving the discrimination ability for small defect targets and enhancing the overall detection performance.

E. ANCHOR FRAME OPTIMIZATION

The anchor box is crucial for target detection, During the training process, the network optimizes anchor box iteratively to establish associations with pixel information on feature maps. It can be guided by ground truth. The feature extraction network divides the input image into feature maps of sizes 13 × 13, 26 × 26, and 52 × 52. Subsequently, these feature maps are used to scan each image in detection. If the pixel information within these grid cells closely matches that observed target, the system will identify these grid cells. After that, the bounding box is adopted for localization and visualization precisely, which is used to guide the generation of bounding boxes. Consequently, anchor boxes can reduce the number of negative samples significantly to enhance the accuracy of detection. The schematics diagram of anchor box, ground truth and predicted bounding box are shown in Fig. 3. The relationship between anchor box, ground truth, and predicted bounding box are expressed as follows:

$$\begin{aligned} t_x^p &= \frac{x_p - x_a}{w_a}, t_y^p = \frac{y_p - y_a}{h_a} \\ t_w^p &= \log(\frac{w_p}{w_a}), t_h^p = \log(\frac{h_p}{h_a}) \\ t_x^g &= \frac{x_g - x_a}{w_a}, t_y^g = \frac{y_g - y_a}{h_a} \\ t_w^g &= \log(\frac{w_g}{w_a}), t_h^g = \log(\frac{h_g}{h_a}) \end{aligned} \tag{8}$$

where, (t_x^p, t_y^p) represents the offset of the center point of predicted bounding box relative to the center point of anchor box; (t_w^p, t_h^p) denotes the scaling coefficients for the width and height of predicted bounding box relative to the center point of anchor box; (t_x^g, t_y^g) indicates the offset of the center point of ground truth relative to the center point of anchor box; (t_w^g, t_h^g) signifies the scaling factors for the width and height of the ground truth relative to the center point of anchor box.

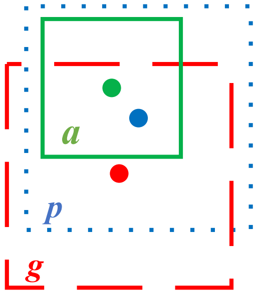


FIGURE 3. Schematics diagram of anchor box (green dashed-line box), ground truth (red solid-line box) and predicted bounding box (blue dotted-line box).

The first two sub-formulas in Eq. (8) adjust the anchor box by incorporating the offset of the network prediction to obtain the predicted bounding box. The impact of defects that are excessively long or wide can be alleviated by the normalization and logarithmic transformation. The latter two sub-formulas in Eq. (8) regulate the network parameters by calculating the loss caused by the offset between the ground truth and the predicted bounding box. It's important to consider that the setting of the anchor box can impact the prediction results. The 1-IoU index [17] and K-means algorithm is utilized to cluster the dataset of surface defects by genetic algorithm. The steps are shown as follows.

1) Select the number of clusters K.
2) Choose initial center of the cluster.
3) Assign each sample to the nearest cluster.
4) Update the cluster center by the mean of samples in each cluster.

5) Repeat the above two steps until the cluster center remains unchanged or changes little to meet the given termination condition.

The index of best possible recall (BPR) that shown in Eq. (9) is used for comparing the optimized anchor box and the original anchor box. The closer the value of BPR is to 1, the better the anchor box is.

$$BPR = gt_{call} \big/ gt_{total} \tag{9}$$

where, gt_{call} reflects the total number of anchor boxes exceeding a predefined threshold; gt_{total} represents the overall quantity of all anchor boxes.

Comparison results between optimized anchor box and original anchor box is shown in Fig. 4. The BPR is 0.9817, 0.9992 for YOLOv5s and SKS-YOLO, respectively. It is shown that the optimization of anchor box can improve the detection accuracy, and the predicted bounding box is nearly the same as the ground truth.

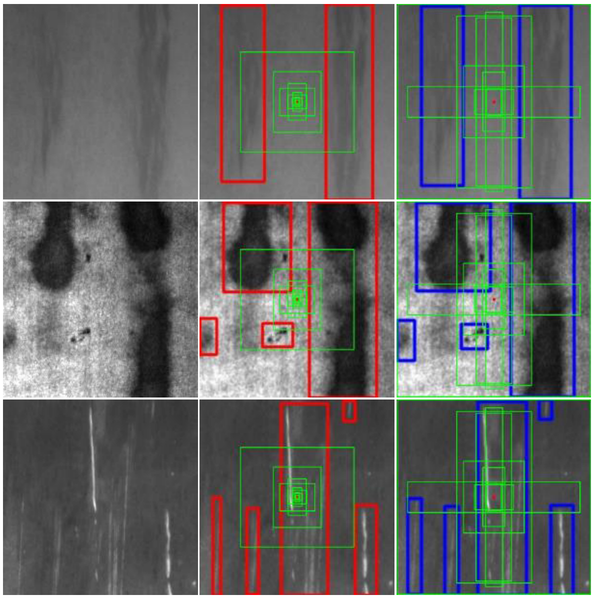

(a) (b) (c)

FIGURE 4. Comparison between optimized anchor box and original anchor box: the first column represents the original image, the second column represents the original anchor box and ground truth, the third column represents the optimized anchor box and predicted bounding box.

F. LOSS FUNCTION OPTIMIZATION

The loss function of CIoU is extremely sensitive to the offset between the predicted bounding box and the ground truth. A small offset will give rise to a significant change in the CIoU, which is not able to reflect the quality of the target detection box precisely. The distance between the center point of the predicted bounding box and the ground truth should be considered in CIoU, it may lead to the instability of the boundary prediction. The paper further considers the vector angle between the ground truth and the predicted bounding box based on the loss function of SIoU. It includes four parts: angle loss, distance loss, shape loss and IoU loss, which is shown in Fig. 5. The penalty term can reduce the offset of the bounding box and raise the stability of the bounding box prediction.

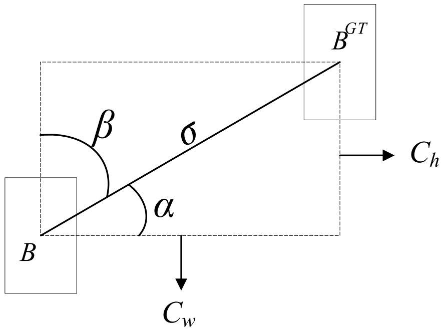


FIGURE 5. Schematics diagram of SIoU calculation.

The angle loss formula is:

$$\Lambda = 1 - 2 * sin^2 \left(arcsin(x) - \frac{\pi}{4} \right) \tag{10}$$

where, $x = c_h \big/ \sigma = sin(\alpha)$; σ is the distance between the center point of the predicted bounding box and the ground truth; c_h is the difference of height between the center point of the predicted bounding box and the ground truth.

The distance loss is calculated as follows:

$$\Delta = \sum_{t=x,y} \left(1 - e^{-\gamma \rho_t}\right) \tag{11}$$

where, $\rho_x = (\frac{b_{c_x}^{gt} - b_{c_x}}{c_w})^2$, $\rho_y = (\frac{b_{c_y}^{gt} - b_{c_y}}{c_w})^2$, $\gamma = 2 - \Lambda$.

When α tends to 0°, the contribution of distance loss will decrease. When α tends to 45°, the contribution of distance loss will increase.

The shape loss formula is:

$$\Omega = \sum_{t=w,h} \left(1 - e^{-\omega_t}\right)^{\theta} \tag{12}$$

where, $\omega_w = \frac{|w - w^{gt}|}{max(w, w^{gt})}$, $\omega_h = \frac{|h - h^{gt}|}{max(h, h^{gt})}$, (w, h) and (w^{gt}, h^{gt}) are the width and height of the predicted bounding box and the ground truth, respectively.

The IoU loss is calculated as follows:

$$\mathrm{IoU} = \frac{\left|B \cap B^{GT}\right|}{\left|B \cup B^{GT}\right|} \tag{13}$$

In summary, the SIoU loss is calculated as follows:

$$L_{\mathrm{SIoU}} = 1 - \mathrm{IoU} + \frac{\Delta + \Omega}{2} \tag{14}$$

IV. EXPERIMENTS AND RESULT ANALYSIS

A. EXPERIMENTAL SETUP

1) DATASET DESCRIPTION

three typical surface defects images (Inclusions, Patches, and Scratches) from the NEU-DET dataset and defect-free image are selected in the following experiments. Each type of defect comprises of 300 images with a size of 200 × 200, as shown in Fig. 6. The dataset is randomly divided into a training set and a test set in a 9:1 ratio. The input image size is 640 × 640, the training rounds is 300 and a batch size is 16.

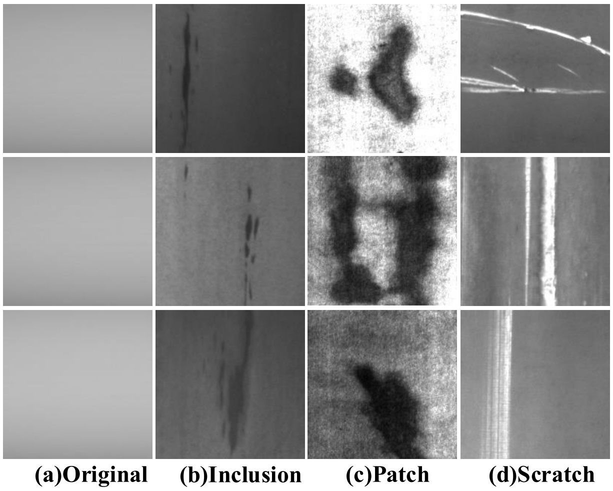


FIGURE 6. Examples of surface images of steel sheet.

2) IMPLEMENTATION DETAILS

the experiments are performed on a work computer with an Intel(R) Core (TM) i7-12700H CPU @2.30GHz CPU, an NVIDIA GeForce RTX3070Ti GPU (with 8 GB memory), CUDA 11.3, and cuDNN 8.9.2 on Windows 11 64-bit with PyTorch1.10.

3) EVALUATION METRICS

The precision (P), recall (R), average precision (AP) and mAP are used as the metrics to evaluate the proposed method comprehensively. The GFLOPS are used as the metric to evaluate the computational complexity. The FPS is used as the metric to evaluate the efficiency. The above metrics are defined as follows.

$$P = \frac{N_{TP}}{N_{TP} + N_{FP}}, R = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{15}$$

where, N_{TP} and N_{FN} represent the number of defects that are detected correctly or not respectively. N_{FP} represents the number of areas that are misclassified.

The mAP is used to evaluate the overall detection performance of the proposed method, which is the average of the AP of all the categories.

The AP is calculated as follows:

$$\mathrm{AP} = \int_0^1 P(R)\, dR \tag{16}$$

The mAP is calculated as follows:

$$\mathrm{mAP} = \frac{\sum_{i=1}^{N} (AP)_i}{N} \tag{17}$$

where, N is the number of samples in the dataset.

The FPS is calculated as follows:

$$\mathrm{FPS} = FrameNum \big/ ElapsedTime \tag{18}$$

where, $FrameNum$ is the total number of test image; $ElapsedTime$ is the total run time.

B. CONVERGENCE ANALYSIS AND ABLATION STUDY

The training epoch is 300 rounds, and loss curve is shown in Fig. 7. Box_loss represents the difference between the predicted bounding box and the ground truth. Class_loss denotes the classification loss. It is used to judge whether the model can identify the defect target and classify it into the correct category precisely. Object_loss represents a confidence loss, which is capable of detecting whether there is an object in the grid and calculating the confidence of the network. When the epoch is 300, the loss values of the SKS-YOLO no longer decrease to indicate the network has converged and stabilized.

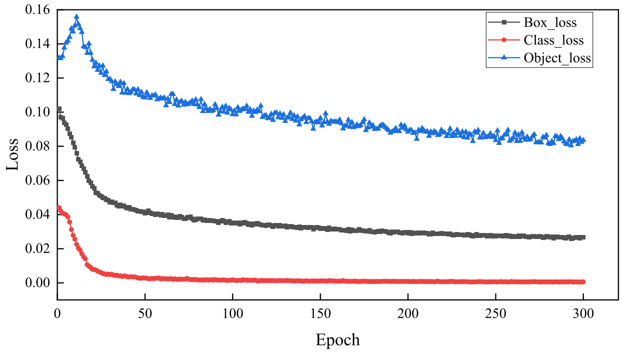


FIGURE 7. The convergence curve of SKS-YOLO.

The ablation study consists of the following: affections of the EfficientNetv2 module, affections of the ASPP module, affections of the SE-Net module, affections of anchor frame optimization module, and affections of the SIoU loss function in the proposed SKS-YOLO model.

1) AFFECTIONS OF EfficientNetv2

EfficientNetv2 module is used to replace the CSP-DarkNet53 module in the original YOLOv5s backbone network. It is used for the fusion of features from the shallow and deep layers to the front layer. Compared with CSP-DarkNet53, the feature fusion of EfficientNetv2 is more comprehensive. The experimental results in Table 3 (1^{st} and 2^{nd} row) show that mAP is increased from 85.6% to 86.1% and GFLOPS decreased by 67.4%. It indicates that EfficientNetv2 can improve the detection accuracy and reduce the calculation load in steel defect detection.

2) AFFECTIONS OF ASPP

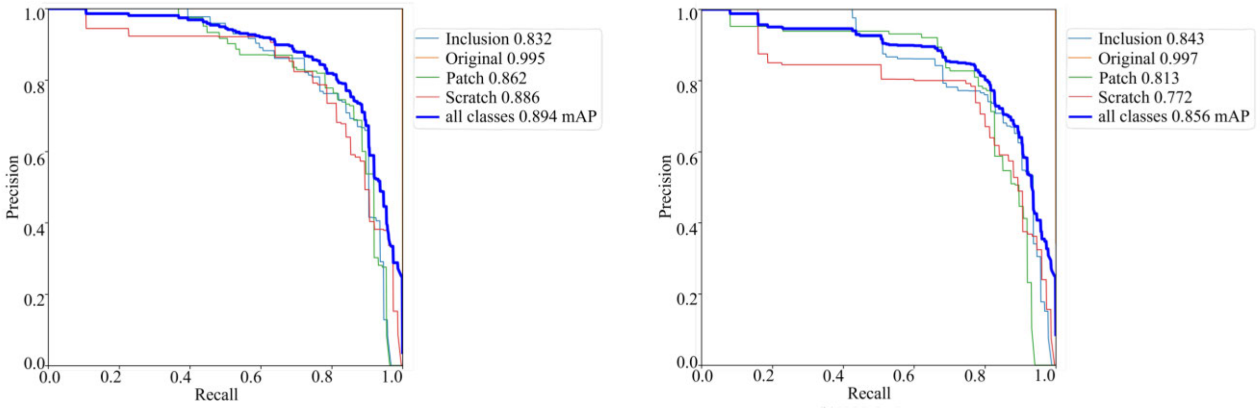
from the Table 3 (6^{th} and 7^{th} row), we can see that the mAP is increased from 86.2% to 89.4%. It indicates that ASPP module can improve the ability of feature extraction for small defect targets.

TABLE 3. Comparison of ablation experiment.

No.	EfficientNetv2	ASPP	SE-Net	Anchors	SIoU	AP (%)				mAP (%)	GFLOPS
						Original	Inclusion	Patch	Scratch		
1						99.7	84.3	81.3	77.2	85.6	17.2
2	√					99.6	83.4	82.7	81.1	86.7	5.6
3	√		√			99.5	83.5	87.0	78.8	87.2	6.3
4	√	√	√			99.6	80.3	85.4	81.7	86.7	6.3
5	√	√	√	√		99.6	84.7	86.4	83.3	88.5	6.3
6	√		√	√	√	99.6	78.8	82.3	84.1	86.2	6.3
7	√	√	√	√	√	99.6	83.2	86.2	88.6	89.4	6.3

TABLE 4. Comparison of different methods.

Method	AP (%)				mAP (%)	FPS
	Original	Inclusion	Patch	Scratch		
Faster R-CNN	100.0	65.7	74.9	79.8	80.1	11
SSD	100.0	55.0	71.9	66.3	73.3	30
RetinaNet	100.0	27.6	67.0	31.2	56.5	25
CenterNet	100.0	70.8	71.2	83.5	81.4	20
EfficientDet	100.0	55.9	75.6	45.9	69.3	15
YOLOv5s	99.7	84.3	81.3	77.2	85.6	66
SKS-YOLO	99.5	83.2	86.2	88.6	89.4	55



(a)SKS-YOLO

(b)YOLOv5s

FIGURE 8. Comparison of P-R curves of the proposed SKS-YOLO and YOLOv5s.

3) AFFECTIONS OF SE-NET

Table 3 (2nd and 3rd row) shows that SE-Net module can boost the performance of the network by learning the weight of each channel adaptively, and the mAP is increased from 86.7% to 87.2%. This indicates that it helps SKS-YOLO perform better in steel defect detection. Compared with YOLOv5s directly assigning attention to the feature channel, this module calculates the difference between the features to give attention to the feature channel and focus on the detection of small targets. Thus, it has higher detection accuracy.

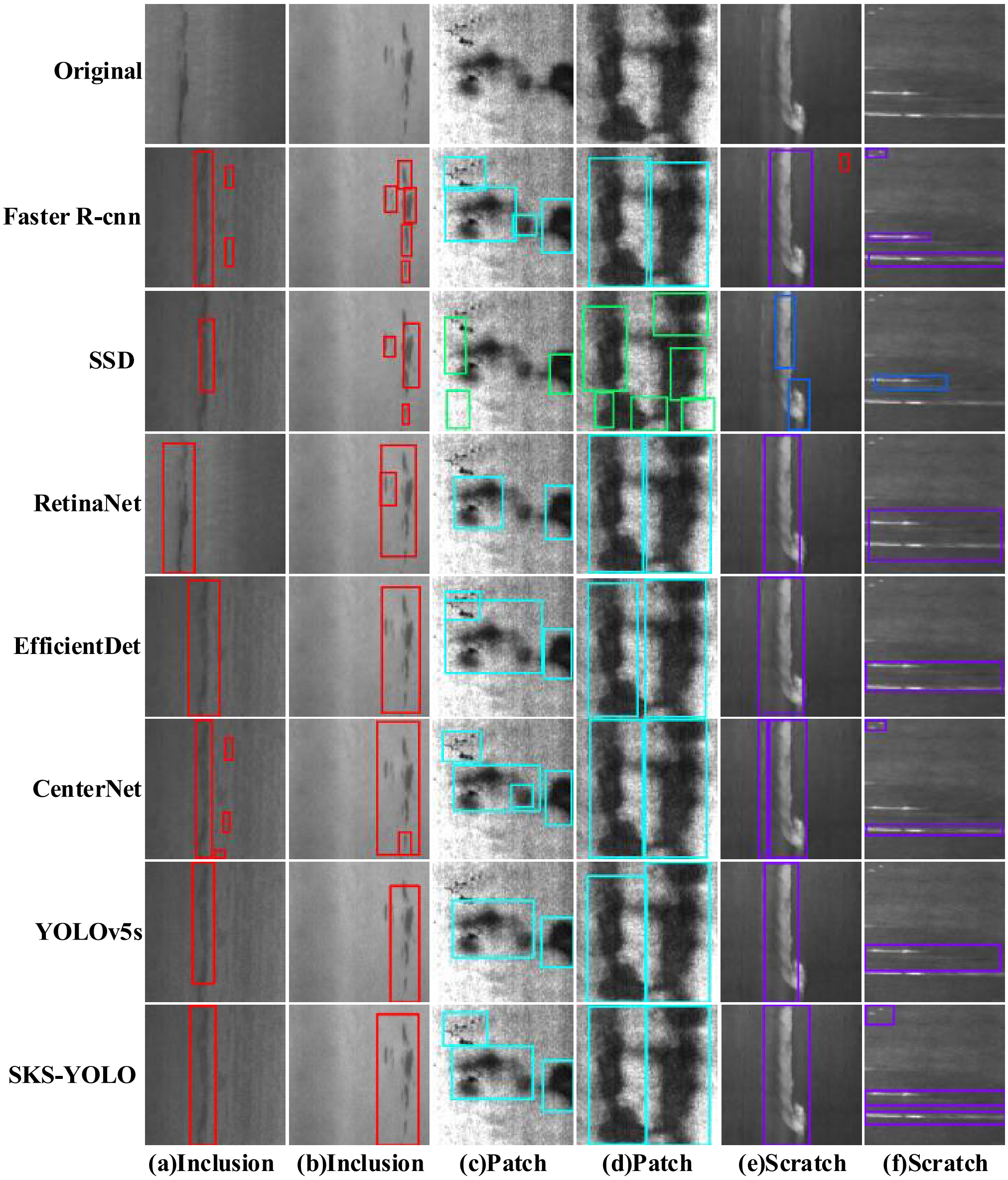


FIGURE 9. Comparison of the defect image detection results by different methods.

4) AFFECTIONS OF ANCHOR FRAME

it does not set many hyper-parameters of these anchor boxes and possesses good detection results on defects with small and narrow shapes. Table 3 (4th and 5th row) shows that it can improve the detection accuracy of the model for various defects greatly, the mAP is increased from 86.7% to 88.5%.

5) AFFECTIONS OF THE SIoU

the results are shown in Table 3 (5th and 7th row) denotes the mAP is increased from 88.5% to 89.4%. It can be observed that SIoU slightly improves the detection performance of SKS-YOLO.

C. COMPARISON WITH SOTA WORKS

In order to verify the detection performance of the proposed SKS-YOLO method, it is compared with some detection methods, including Faster R-CNN, SSD, RetinaNet, CenterNet [46], EfficientDet [47] and YOLOv5s. The experimental results are presented in Table 4, Fig. 8 and Fig. 9. As shown in Fig. 8, the mAP of SKS-YOLO is 89.4%, which is 3.8% higher than YOLOv5s (85.6%). SKS-YOLO significantly increases the detection precision of Patches (4.9%) and Scratch (11.4%). Compared with YOLOv5s has some problems of mismatch and missing for small defect targets, SKS-YOLO has better performance in detecting small defect targets. It indicates the added attention mechanism is conducive to locate defect targets. The improved network has better detection accuracy and exhibits better robustness. It is evident from Table 4 that mAP of the SKS-YOLO is 89.4%, which is 3.8% higher than YOLOv5s. SKS-YOLO achieved the largest AP on Patch and Scratch defects, with 86.2% and 88.6%, respectively. Compared with Faster R-CNN, SSD, RetinaNet, CenterNet, EfficientDet, it is increased by 9.3%, 16.1%, 32.9%, 8%, 20.1%, respectively.

V. CONCLUSION

In this article, a new detection model, SKS-YOLO based on YOLOv5s is proposed for the detection of steel plate defects. This deep network can tackle challenges such as large defect spans, poor detection of small defects, and low accuracy in the existing defect detection method on the surface of steel plates. Compared with the various detection models, both the accuracy and efficiency of SKS-YOLO are improved. The module of EfficientNetv2, ASPP and SE-Net greatly improve the detection performance of SKS-YOLO. The experimental results show that the proposed SKS-YOLO model achieves 89.4% mAP for the defect detection task.

The proposed SKS-YOLO model requires that the available surface defects are representative of those likely to be seen, and that representative normal images are included. It may not be able to reliably identify completely different types of defects that it has not seen before. At the same time, the anchor frame optimization of SKS-YOLO takes the K-means algorithm as the basic processing unit, but how to adaptively determine the number of clusters K is not discussed in the paper. We plan to put forward a multiscale detection and segmentation method, which is expected to automatically determine the K to be detected. In addition, SKS-YOLO can be optimized to further enhance detection efficiency and accuracy.

In the future, we will focus on two directions as follows:

1) In the surface defect detection of steel plate, normal image and defect image need to be labeled and classified, which is laborious and time-consuming, while the deep learning model needs large-scale samples to obtain an excellent detection model. Therefore, we are going to apply transfer learning and unsupervised learning to design more effective models to distinguish and detect various kinds of defects. Besides, we are also looking for more efficiency attention module to speed up the feature map generation process to enable real-time online defect localization and detection.

2) For steel sheet and other industrial products, such as AMOLED screens, wood, ceramic tile and leather, normal surface generally shows homogeneous texture, defects are local anomalies on the surface, which are dissimilar to the texture at other locations. The defects are considered as aberrant or anomalous arrayed pixels in the image, comparing with defect-free region. Therefore, the application of other industrial products in complex working conditions to verify the applicability of proposed SKS-YOLO model.

ACKNOWLEDGMENT

The authors would like to thank editors and anonymous reviewers for their constructive suggestions to improve the manuscript.

REFERENCES

[1] Y. Liu, C. Zhang, and X. Dong, "A survey of real-time surface defect inspection methods based on deep learning," *Artif. Intell. Rev.*, vol. 56, no. 10, pp. 12131–12170, Oct. 2023, doi: 10.1007/s10462-023-10475-7.

[2] Y. Chen, Y. Ding, F. Zhao, E. Zhang, Z. Wu, and L. Shao, "Surface defect detection methods for industrial products: A review," *Appl. Sci.*, vol. 11, no. 16, p. 7657, Aug. 2021, doi: 10.3390/app11167657.

[3] P. Prasitmeeboon and H. Yau, "Defect detection of particleboards by visual analysis and machine learning," in *Proc. 5th Int. Conf. Eng., Appl. Sci. Technol. (ICEAST)*, Jul. 2019, pp. 1–4, doi: 10.1109/ICEAST.2019.8802526.

[4] C.-F. Chang, J.-L. Wu, K.-J. Chen, and M.-C. Hsu, "A hybrid defect detection method for compact camera lens," *Adv. Mech. Eng.*, vol. 9, no. 8, Aug. 2017, Art. no. 168781401772294, doi: 10.1177/1687814017722949.

[5] F.-L. Wang and B. Zuo, "Detection of surface cutting defect on magnet using Fourier image reconstruction," *J. Central South Univ.*, vol. 23, no. 5, pp. 1123–1131, May 2016, doi: 10.1007/s11771-016-0362-y.

[6] X. Wen, K. Song, L. Huang, M. Niu, and Y. Yan, "Complex surface ROI detection for steel plate fusing the gray image and 3D depth information," *Optik*, vol. 198, Dec. 2019, Art. no. 163313, doi: 10.1016/j.ijleo.2019.163313.

[7] H. Wang, J. Zhang, Y. Tian, H. Chen, H. Sun, and K. Liu, "A simple guidance template-based defect detection method for strip steel surfaces," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 2798–2809, May 2019, doi: 10.1109/TII.2018.2887145.

[8] J. Hou, K. Xia, F. Yang, and B. Zu, "Strip steel surface defects recognition based on socp optimized multiple kernel RVM," *Math. Problems Eng.*, vol. 2018, no. 1, pp. 1–8, Mar. 2018, doi: 10.1155/2018/9298017.

[9] A. Y. Virasova, D. I. Klimov, O. E. Khromov, I. R. Gubaidullin, and V. V. Oreshko, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Radioengineering*, vol. 85, no. 9, pp. 115–126, Sep. 2021, doi: 10.18127/j00338486-202109-11.

[10] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[12] W. Zhao, F. Chen, H. Huang, D. Li, and W. Cheng, "A new steel defect detection algorithm based on deep learning," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–13, Mar. 2021, doi: 10.1155/2021/5592878.

[13] Y. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Buyuközturk, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 33, no. 9, pp. 731–747, Sep. 2018, doi: 10.1111/mice.12334.

[14] B. Su, H. Chen, P. Chen, G. Bian, K. Liu, and W. Liu, "Deep learning-based solar-cell manufacturing defect detection with complementary attention network," *IEEE Trans. Ind. Informat.*, vol. 17, no. 6, pp. 4084–4095, Jun. 2021, doi: 10.1109/TII.2020.3008021.

[15] C. Zhang, J. Cui, and W. Liu, "Multilayer feature extraction of AGCN on surface defect detection of steel plates," *Comput. Intel. Neurosci.*, vol. 2022, no. 1, pp. 1–13, Oct. 2022, doi: 10.1155/2022/2549683.

[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. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, vol. 1, no. 1, pp. 6517–6525, doi: 10.1109/CVPR.2017.690.

[18] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[19] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[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. 14th Eur. Conf.* Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 21–37.

[21] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," 2017, *arXiv:1708.02002*.

[22] X. Yin, Y. Chen, A. Bouferguene, H. Zaman, M. Al-Hussein, and L. Kurach, "A deep learning-based framework for an automated defect detection system for sewer pipes," *Autom. Construct.*, vol. 109, Jan. 2020, Art. no. 102967, doi: 10.1016/j.autcon.2019.102967.

[23] C. Zhang, C. C. Chang, and M. Jamshidi, "Bridge damage detection using a single-stage detector and field inspection images," 2018, *arXiv:1812.10590*.

[24] X. Yu, W. Lyu, D. Zhou, C. Wang, and W. Xu, "ES-Net: Efficient scale-aware network for tiny defect detection," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022, doi: 10.1109/TIM.2022.3168897.

[25] R. Wang and C. F. Cheung, "CenterNet-based defect detection for additive manufacturing," *Expert Syst. Appl.*, vol. 188, Feb. 2022, Art. no. 116000, doi: 10.1016/j.eswa.2021.116000.

[26] T. Y. Zhang, J. Li, J. Chai, Z. Q. Zhao, and W. D. Tian, "Improved YOLOv5 network with attention and context for small object detection," in *Proc. ICIC*, 2022, pp. 341–352.

[27] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "DenseASPP for semantic segmentation in street scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3684–3692, doi: 10.1109/CVPR.2018.00388.

[28] J. Hu, S. Li, and G. Sun, "Squeeze-and-excitation networks," 2018, *arXiv.1709.01507*.

[29] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," 2021, *arXiv:2104.00298*.

[30] Z. Gevorgyan, "SIoU loss: More powerful learning for bounding box regression," 2022, *arXiv:2205.12740*.

[31] X. Zihao, W. Hongyuan, Q. Pengyu, D. Weidong, Z. Ji, and C. Fuhua, "Printed surface defect detection model based on positive samples," *Comput., Mater. Continua*, vol. 72, no. 3, pp. 5925–5938, 2022, doi: 10.32604/cmc.2022.026943.

[32] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, Feb. 2020, doi: 10.1007/s11263-019-01247-4.

[33] P. K. R. Maddikunta, Q.-V. Pham, P. B, N. Deepa, K. Dev, T. R. Gadekallu, R. Ruby, and M. Liyanage, "Industry 5.0: A survey on enabling technologies and potential applications," *J. Ind. Inf. Integr.*, vol. 26, Mar. 2022, Art. no. 100257, doi: 10.1016/j.jii.2021.100257.

[34] Y. Xie, W. Hu, S. Xie, and L. He, "Surface defect detection algorithm based on feature-enhanced Yolo," *Cognit. Comput.*, vol. 15, no. 2, pp. 565–579, Mar. 2023, doi: 10.1007/s12559-022-10061-z.

[35] H. Wang, X. Yang, B. Zhou, Z. Shi, D. Zhan, R. Huang, J. Lin, Z. Wu, and D. Long, "Strip surface defect detection algorithm based on YOLOv5," *Materials*, vol. 16, no. 7, p. 2811, Mar. 2023, doi: 10.3390/ma16072811.

[36] Y. Zhang, S. Shen, and S. Xu, "Strip steel surface defect detection based on lightweight YOLOv5," *Frontiers Neurorobotics*, vol. 17, Oct. 2023, Art. no. 1263739.

[37] J. Zhang, Y. Meng, and Z. Chen, "A small target detection method based on deep learning with considerate feature and effectively expanded sample size," *IEEE Access*, vol. 9, pp. 96559–96572, 2021, doi: 10.1109/ACCESS.2021.3095405.

[38] F. Li, K. Xiao, Z. Hu, and G. Zhang, "Fabric defect detection algorithm based on improved YOLOv5," *Vis. Comput.*, vol. 40, no. 4, pp. 2309–2324, Apr. 2024, doi: 10.1007/s00371-023-02918-7.

[39] Y. Li, Y. Fan, S. Wang, J. Bai, and K. Li, "Application of YOLOv5 based on attention mechanism and receptive field in identifying defects of thangka images," *IEEE Access*, vol. 10, pp. 81597–81611, 2022, doi: 10.1109/ACCESS.2022.3195176.

[40] H. Xin and K. Zhang, "Surface defect detection with channel-spatial attention modules and bi-directional feature pyramid," *IEEE Access*, vol. 11, pp. 88960–88970, 2023, doi: 10.1109/ACCESS.2023.3303897.

[41] R. Zhang and C. Wen, "SOD-YOLO: A small target defect detection algorithm for wind turbine blades based on improved YOLOv5," *Adv. Theory Simulations*, vol. 5, no. 7, Jul. 2022, Art. no. 2100631, doi: 10.1002/adts.202100631.

[42] Z. Guo, C. Wang, G. Yang, Z. Huang, and G. Li, "MSFT-YOLO: Improved YOLOv5 based on transformer for detecting defects of steel surface," *Sensors*, vol. 22, no. 9, p. 3467, May 2022, doi: 10.3390/s22093467.

[43] J. Shi, J. Yang, and Y. Zhang, "Research on steel surface defect detection based on YOLOv5 with attention mechanism," *Electronics*, vol. 11, no. 22, p. 3735, Nov. 2022, doi: 10.3390/electronics11223735.

[44] Y. Qu, B. Wan, C. Wang, H. Ju, J. Yu, Y. Kong, and X. Chen, "Optimization algorithm for steel surface defect detection based on PP-YOLOE," *Electronics*, vol. 12, no. 19, p. 4161, Oct. 2023, doi: 10.3390/electronics12194161.

[45] K. Song and Y. Yan, "A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects," *Appl. Surf. Sci.*, vol. 285, pp. 858–864, Nov. 2013, doi: 10.1016/j.apsusc.2013.09.002.

[46] X. Zhou, D. Wang, and P. Krähenbuhl, "Objects as points," 2019, *arXiv:1904.07850*.

[47] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787, doi: 10.1109/cvpr42600.2020.01079.

SHIYANG ZHOU received the Ph.D. degree from Huazhong University of Science and Technology, Wuhan, China, in 2017. He is currently an Associate Professor with the School of Machinery and Automation, Wuhan University of Science and Technology. His current research interests include machine vision and laser cleaning.

SIMING AO received the B.S. degree in mechatronic engineering from Zhengzhou Institute of Science and Technology, Zhengzhou, Henan, China, in 2022. He is currently pursuing the master's degree in mechanical engineering with Wuhan University of Science and Technology, China. His current research interest includes machine vision inspection in industrial production.

ZHIYING YANG received the B.S. degree in mechatronic engineering from the Science and Technology College, Nanchang Hangkong University, Nanchang, Jiangxi, China, in 2023. She is currently pursuing the master's degree in mechanical engineering with Wuhan University of Science and Technology, China. Her current research interests include signal analysis and visual inspection.

HUAIGUANG LIU received the Ph.D. degree in mechanical and electronic engineering from Huazhong University of Science and Technology, in 2011. He is currently an Associate Professor with the School of Machinery and Automation, Wuhan University of Science and Technology. His research interests include machine vision, intelligent detection, and noncontact precision measurements.