Sensors and Materials, Vol. 36, No. 11 (2024) 4881–4902
MYU Tokyo

S & M 3837

Improved Reparameterization You-Only-Look-Once v5 Model for Strip-steel Surface Defect Detection

Sijie Qiu,[1] Chi-Hsin Yang,[1*] Long Wu,[1] Wenqi Song,[2] and Jian-Zhou Pan[3,4]

[1]School of Mechanical and Electric Engineering, Sanming University, Sanming, Fujian Province 365004, China
[2]School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350000, China
[3]Fujian Sansteel (Group) Co. Ltd., Sanming, Fujian Province 365004, China
[4]School of Material Science and Engineering, University of Science and Technology Beijing, Beijing 100083, China

(Received April 30, 2024; accepted October 8, 2024)

Keywords: surface defect detection, RepVGG–Light module, bidirectional feature pyramid network (BiFPN), normalized Gaussian-Wasserstein distance (NGWD)

In this study, we propose a reparameterization You-Only-Look-Once v5 (YOLOv5) algorithm model for strip-steel surface defect detection to address low precision and poor timeliness in traditional methods. The proposed model introduces a re-parameterized VGG Light module, an enhanced bidirectional feature pyramid network feature structure, and a bounding box regression loss function fused with a normalized Gaussian-Wasserstein distance metric to improve small-target-defect detection accuracy. The experimental findings reveal a mean average precision (*mAP*) of 82.1% on the NEU-DET dataset, representing a notable improvement of 4.1% over the baseline YOLOv5s algorithm. Furthermore, the proposed algorithm model demonstrates superior detection accuracy compared with other prevalent object detection models and effectively mitigates challenges such as false detections and missed detections of small targets. Notably, it achieves an impressive detection speed of 68 FPS, affirming its efficacy in real-time applications.

1. Introduction

Strip steel is a critical component in the industrial and economic landscape, with widespread use in the automobile industry, home appliances, mobile phone electronics, and construction. There is an increasing need for strip steel with superior surface quality that meets stringent performance standards and demonstrates exceptional durability. However, environmental conditions, variations in raw material quality, and manufacturing processes can contribute to the generation of surface defects during production. These defects can significantly impact the steel's wear resistance, corrosion resistance, and fatigue strength while also posing potential risks during practical usage scenarios.

*Corresponding author: e-mail: 20190207@fjsmu.edu.cn
https://doi.org/10.18494/SAM5107

ISSN 0914-4935

With the evolution of computer technology, machine-vision-based defect detection has become widely prevalent in industrial applications.[1,2] Recent advancements in deep learning algorithms for object detection have been particularly notable owing to the rapid progress in convolutional neural network (CNN) architectures. The existing literature classifies deep learning object detection algorithms into two primary categories: two-stage and single-stage methods. Two-stage methods encompass faster region-based CNNs (F-RCNNs)[3,4] and improved F-RCNNs,[5] while single-stage methods primarily include the You-Only-Look-Once (YOLO) series[6–13] and the single-shot multibox detector (SSD).[14–16]

The YOLO series object detection algorithm has gained widespread adoption in various fields owing to its fast and accurate detection results. According to Ref. 7, an enhanced YOLOXs model was developed for identifying subhealth regions on rape plants during the bolting stage in agriculture. Moreover, multiple enhanced versions of YOLOv5 models have been developed for detecting helmets,[8] lithium battery poles,[9] road damage,[10] and obstacles in crab ponds.[11] Additionally, improved YOLOX models are specifically designed for face mask recognition of masked people and traffic sign detections.[12,13]

A single-stage target detection algorithm model, such as the YOLO series, unifies the tasks of target classification and localization, resulting in rapid detection speed that aligns with the stringent demands of high-speed industrial detection applications. The detection of steel surface defects has been extensively researched using various iterations of YOLO algorithms. In Ref. 17, an enhanced YOLOv3 model was proposed, while in Refs. 18 to 21, the issue of steel-surface defect detection was addressed by introducing several YOLOv5-based models with different modified submodules. More recent advancements have been made with improved YOLOX algorithms.[22,23] The aforementioned effort is directed towards refining feature extraction to address the low detection accuracy associated with the single-stage algorithm. The goal is to achieve a more optimal balance between detection accuracy and speed. However, it remains challenging to simultaneously meet the requirements for both speed and accuracy in industrial defect detection.

Motivated by the results of prior analysis, an improved model associated with machine learning technology, named the reparameterization strip-steel surface defect detection (RSSDD) YOLO algorithm model, is introduced to address the real-time and accuracy requirements of strip-steel surface defect detection. Among the four primary models in the YOLOv5 series (YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x), we opt for the lightweight model and real-time performance of the YOLOv5s network structure. The main aim is to further enhance detection accuracy while ensuring swift processing. Consequently, the key contributions and innovations of our study are evident in several aspects.

(1) In this study, a re-parameterized VGG Light (RepVGG–Light) module based on the original RepVGG module[24] has been developed to replace the C3_1 module in the Backbone. This modification results in the establishment of an enhanced Backbone feature extraction network that maintains a multibranch structure during training, thereby amplifying the model's representational power. Furthermore, a single branch structure is utilized during the inference stage to expedite inference speed and strike a balance between detection accuracy and speed.

(2) We develop an enhanced BiFPN-FE structure to improve the Neck module of the original YOLOv5s model. By incorporating weighted connections, we effectively integrates in this design shallow and deep features, maximizing the utilization of the Backbone extracted features to improve the model's multiscale prediction capability to detect surface defects on strip steel.

(3) The bounding box regression loss function, which integrates the normalized Gaussian-Wasserstein distance (NGWD) metric as the WD-IoU loss function, was proposed. It is combined with the CIoU loss function in the original YOLOv5s model to optimize the regression loss for predicted bounding box. The refined algorithm model presented in this study shows promise in significantly enhancing detection accuracy, particularly for small target defects.

This paper is structured into the following sections. Section 2 offers a concise exposition of the original YOLOXv5s model architecture and delineates the functionalities of its submodules. In Sect. 3, we introduce the RSSDD YOLO model architecture and propose three strategies for improving the original YOLOXv5s model, including the implementation of a RepVGG–Light_N module and the elaboration of the BiFPN-FE network structure as well as the characteristics of the WD-IoU loss function. Section 4 encompasses the training and efficiency verification of the proposed RSSDD YOLO model using the NEU-DET dataset.[25] Finally, a summary is provided.

2. Architecture of YOLOv5s Model

As illustrated in Fig. 1, the architecture of the YOLOv5s model is compartmentalized into four primary parts, that is, the Input, the Backbone, the feature fusion Neck, and the detection network Head. Submodules within YOLOv5s are further detailed in Fig. 1, with comprehensive

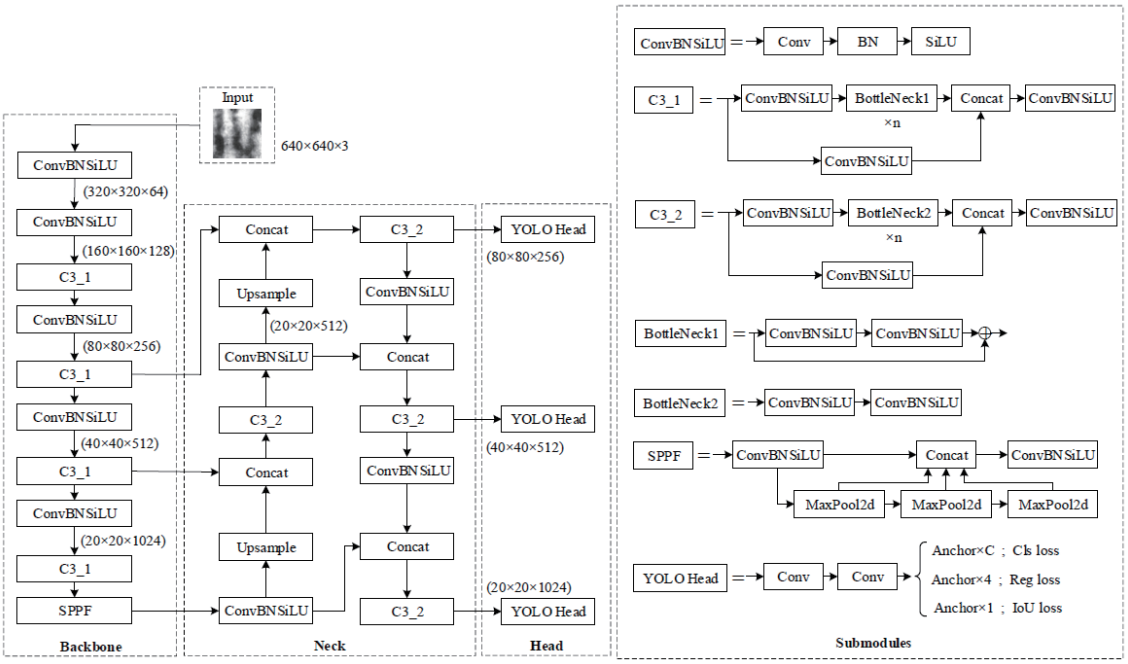


Fig. 1. Architecture of YOLOv5s model.

definitions and operational guidelines provided in Ref. 26. The functions of each primary part are as follows.

(1) Input: the input image is preprocessed so that the size of the picture becomes 640 × 640 × 3 and the image data is normalized.
(2) Backbone: the Backbone module is composed of ConvBNSiLU, C3_1, and spatial pyramid pooling fast (SPPF) modules. The ConvBNSiLU and C3_1 modules are responsible for the hierarchical extraction and abstraction of image features from the input data. On the other hand, the SPPF module integrates feature information across multiple scales, thereby establishing a robust foundation for subsequent object detection tasks. The input image is processed through Backbone convolution, resulting in size changes from 640 × 640 × 3 to 320 × 320 × 64, and then to 160 × 160 × 128. It undergoes processing by the C3_1 module to maintain its size unchanged. The image is further reduced to 80 × 80 × 256 and remains the same after processing with the C3_1 module again. Then, it is compressed to 40 × 40 × 512 before being downsized to 20 × 20 × 1024 and processed by the C3_1 module. Finally, local and global features are integrated within the SPPF module to provide improved network capabilities.
(3) Neck: the Neck module employs a hybrid architecture that combines top-down feature pyramid networks (FPNs)[27] and a bottom-up path aggregation network (PANet),[28] enabling improved integration of semantic and localization features to extract more comprehensive feature information.
(4) Head: the Head consists of three detection layers, each predicting targets of different scales (80 × 80 × 256, 40 × 40 × 512, and 20 × 20 × 1024), culminating in the generation of detection results through loss function calculation.

The loss function of YOLOXv5s is composed of three key components: the positional regression's loss function, L_{IoU}, the cross-entropy loss function for classification, L_{cls}, and the bounding box regression's loss function, L_{reg}. Both L_{cls} and L_{reg} make use of the binary cross-entropy loss.[5] Furthermore, L_{IoU} incorporates the CIoU loss function.[29] The L_{CIoU} function is precisely defined as

$$Loss_{CIoU} = 1 - IoU + \frac{\rho^2\left(A_{pred}, A_{gt}\right)}{D^2} + \alpha v, \tag{1}$$

$$IoU = \frac{\left(A_{pred} \cap A_{gt}\right)}{\left(A_{pred} \cup A_{gt}\right)}, \tag{2}$$

$$\alpha = \frac{v}{\left(1 - IoU\right) + v}, \tag{3}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w_{gt}}{h_{gt}} - \arctan\frac{w_{pred}}{h_{pred}}\right)^2. \tag{4}$$

The *IoU* (intersection over union) is the ratio of the area of overlap between the predicted bounding box A_{pred} and the ground truth bounding box A_{gt} to the area of their union. $\rho(A_{pred}, A_{gt})$ represents the Euclidean distance between the center points of A_{pred} and A_{gt}, while D represents the diagonal length of the smallest external rectangle that encompasses both boxes. The variables w_{gt} and h_{gt} indicate the width and height of the ground truth frame, whereas w_{pred} and h_{pred} represent those of the predicted frame. Additionally, α is a positive trade-off parameter used in optimization algorithms, while v is a measure of the aspect ratio consistency between w_{gt}/h_{gt} and w_{pred}/h_{pred}.

3. RSSDD YOLO Algorithm Model

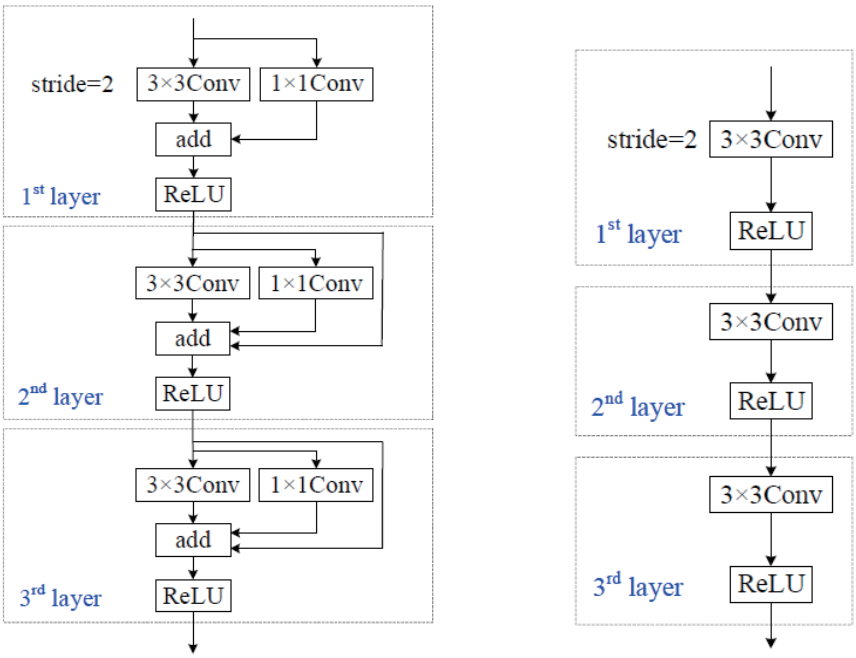
The RSSDD YOLO algorithm model proposed in this study represents a refinement of the YOLOv5s network. Firstly, the operation of a RepVGG–Light_N module is performed to replace the C3_1 module in the Backbone of YOLOv5s, thereby reducing the parameter count and accelerating the model reasoning speed while upholding detection accuracy. Secondly, the feature-strengthening BiFPN-FE structure is integrated to augment the network's feature fusion capability. Lastly, the bounding box loss function WD-IoU is employed in the feature prediction section to optimize predicted bounding boxes and enhance the small target detection capability.

3.1 RepVGG–Light_N module

The C3_1 module in the YOLOv5_s Backbone leverages a multibranch and layered approach to augment model parameters; this potentially leads to overfitting. Consequently, additional data may be required for training and model fine-tuning. While training with a larger dataset can enhance classification accuracy, it also results in prolonged training times, which diminishes memory bandwidth utilization on hardware such as GPUs. To ensure real-time defect detection and algorithmic accuracy enhancement, we propose substituting the original Backbone's C3_1 module with the RepVGG module proposed in Refs. 30 and 31. RepVGG has demonstrated promising potential in image classification by amplifying accurately while curbing computational complexity through structural reparameterization.

The RepVGG module's training and inference networks are depicted in Fig. 2, with the training network in Fig. 2(a) comprising three layers, each incorporating 3 × 3 convolution branches, 1 × 1 convolution branches, and identity residual branches. Notably, the initial layer omits the inclusion of an identity residual branch. Leveraging convolution kernels of diverse sizes within a multibranch structure enables the RepVGG module to attain varying receptive fields and amalgamate feature information from these fields to augment feature extraction capabilities. Furthermore, the integration of multiple residual branch structures endows the network with numerous gradient flow paths, thereby mitigating potential issues related to vanishing gradients in deep-level networks while concurrently enhancing overall network efficiency for feature extraction purposes.

The multibranch architecture of the RepVGG module enhances network detection accuracy, albeit at the cost of a substantial increase in the count of network training parameters. To



(a) (b)

Fig. 2. (Color online) Structure of training network and inference network of RepVGG module. (a) Training network and (b) inference network.

mitigate this drawback, during the Inference stage, the RepVGG module employs the structural reparameterization fusion strategy[30] to amalgamate the original three-way branch in the training network into a single branch, resulting in a significant reduction in the count of parameters for the RepVGG module's inference network, as depicted in Fig. 2(b). Additionally, existing software acceleration libraries have undergone extensive optimization for 3 × 3 we can achieve convolution operations. By transforming the inference network of the RepVGG module into a single-path model using 3 × 3 convolutions, we can realize superior performance optimization using existing software increasing libraries, thereby increasing the inference speed for the RepVGG module.

To enhance the object detection efficiency of the original YOLOXv5_s model without unduly inflating the network's parameter count, we introduce an improved RepVGG–Light_N module. Here, N denotes the number of sequentially connected RepVGG modules, as depicted in Fig. 3. Prior to input into the RepVGG module, a 1 × 1 convolutional layer is employed to downsample the number of input feature channels by half, while another 1 × 1 convolutional layer is utilized post-RepVGG processing to restore the original channel count.

In this study, the C3_1 module in the original YOLOXv5_s model's Backbone has been replaced with the proposed RepVGG–Light_N module to enhance detection accuracy while maintaining the real-time performance of the entire network. The refined Backbone network, as illustrated in Fig. 4, forms an essential part of the RSSDD YOLO algorithm model.

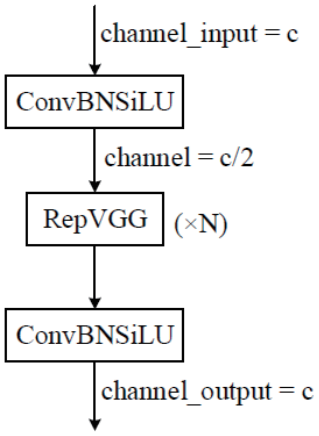


Fig. 3. Structure of RepVGG–Light_N module.

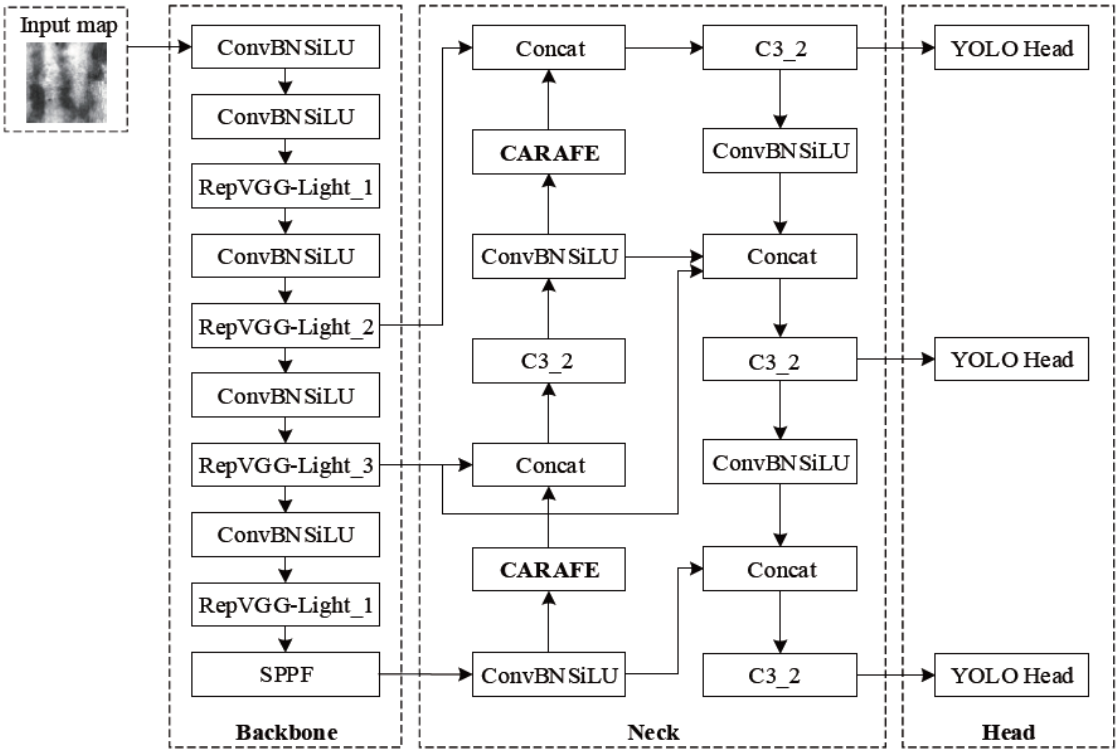


Fig. 4. Architecture of RSSDD YOLO model.

3.2 Structure of BiFPN-FE module

In the YOLOv5s algorithm model, the shallow network features abundant detailed information but lacks strong semantic information owing to fewer convolutional operations. On the other hand, deep network features can generate richer semantic information through multiple convolution operations, but its capability to capture tiny details is somewhat weakened.[27]

Therefore, it is crucial to integrate both deep and shallow features to enhance the model's performance.

The original FPN structure integrates high-level semantic information with precise spatial position details at lower levels using a top-down feature fusion method to generate multiscale feature maps.[32] However, the extended transmission path for spatial location data impedes the effective integration of high- and low-level information. As shown in Fig. 5(a), the PANet introduces a bottom-up pathway within the FPN framework, shortening the transmission paths and enhancing network accuracy. Nevertheless, when fusing features from different input types in both FPN and PANet, there is a lack of differentiated weighting configuration or discernment of distinct input feature significance during simple addition operations.[33]

To address this issue, we introduce a simple and efficient BiFPN structure in this study, as depicted in Fig. 5(b). The BiFPN is based on the PANet architecture and utilizes the fast normalized weighted optimization strategy and incorporates trainable adjusted weight ω_i, i = 1, 2, ..., 9 to learn and define the importance of different input features. This allows for a more rational merging of information from various features across deep and shallow layers within the network.

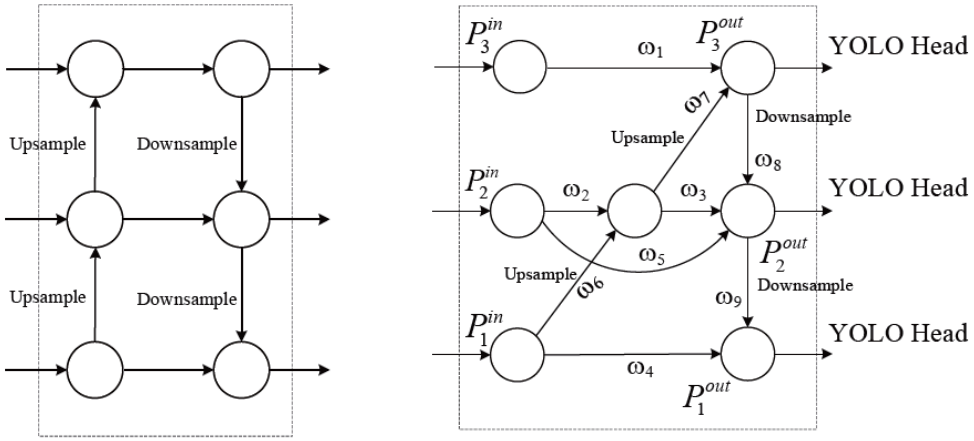
To better integrate the BiFPN structure into the YOLOv5s model and tailor it for strip-steel surface defect detection, in this study, we made three improvements to the BiFPN and devised a feature-enhanced BIFPN-FE module.

(1) Modification of the feature fusion layer structure.

As depicted in Fig. 5(b), an additional cross-stage connection path is incorporated between the input and output nodes of the same scale, based on the PANet structure of the YOLOv5s model, enabling the integration of more features without increasing computational cost. Nodes with minimal contribution are eliminated to enhance feature fusion efficiency. This modification reduces two feature fusion layers compared with the original BiFPN, aligning with the three feature layers of the output detector in the Head.

(2) Modification of the feature fusion operation.

The feature fusion operation has been enhanced with the integration of trainable and



(a) (b)

Fig. 5. Structures of PANet and BiFPN. (a) PANet and (b) BiFPN.

learnable weights across all fusion paths. This facilitates feature graph fusion through a series-parallel mode of channel paths, thereby superseding the original PANet's arithmetic method of summation-by-channel. In this way, the multiscale features are extracted from the input image. In Fig. 5(b), the BiFPN implements rapid normalized fusion,

$$P_i^{out} = \sum_j \frac{\omega_j}{\delta + \Sigma_k \omega_k} \cdot P_j^{in}, i = 1, 2, 3, \tag{5}$$

where P_j^{in} and P_i^{out} are the input and output nodes, respectively. ω_j, ω_k are nonnegative learnable normalized weights in each path and $\delta = 0.0001$. The rectified linear unit is adopted to ensure the data stability. Each normalized weight falls between 0 and 1.

While direct summation of channels may reduce computational complexity, the series-parallel operation of channel paths excels in integrating feature information from diverse branches and ensuring stable defect detection performance. The feature maps extracted by the BiFPN-FE module significantly enhance the final classification and prediction capabilities of the YOLOv5s network.

(3) Modification of the up-sampling operation of the Neck module.

In YOLOv5s, the up-sampling operation utilizes nearest-neighbor and bilinear interpolations. This approach offers the benefits of reduced computational complexity, a straightforward algorithm, and rapid processing. However, the nearest-neighbor interpolation method only considers the gray-level binary values of the closest pixels to the sampling point as input, neglecting the values of other adjacent pixels. Consequently, the original YOLOv5s' up-sampling operation may result in a loss of image information and decreased object detection accuracy.(34)

In contrast to the nearest-neighbor interpolation, which focuses solely on the immediate pixel region around the sample point, the content-aware reassembly of features (CARAFE) module enables up-sampling across a broader receiving range by aggregating continuous information from neighboring regions.(35) By employing adaptive and optimized recombination cores at different locations, the CARAFE module achieves finer feature maps with minimal detail loss and more suitable sampling for strip-steel surface defect characteristics.

The CARAFE module serves as a reassembly operator employing content-aware kernels and comprises two sequential steps: the prediction of a reassembly kernel for each target location based on its content, followed by the reassembling of features using the predicted kernels. The operational processes are delineated by the following formula:

$$W_{l'} = \psi\left(N\left(X_l, k_{encoder}\right)\right), \tag{6}$$

$$X'_{l'} = \phi\left(N\left(X_l, k_{up}\right), W_{l'}\right), \tag{7}$$

where ψ denotes the kernel prediction module and ϕ represents the content-aware reassembly module. Here, X signifies the original feature map, while X' refers to the new feature map

generated after up-sampling. The variable l denotes the original target position, and l' indicates the target position post-up-sampling. $N(X_l, k)$ symbolizes the $k \times k$ subregion of X centered at position l, essentially representing the domain of X_l. $W_{l'}$ pertains to the kernel prediction module ψ that predicts an appropriate kernel for each location l' using contextual information extracted from X_l.

Furthermore, in Eqs. (6) and (7), k_{up} represents the reassembly kernel size, whereas $k_{encoder}$ is a convolution layer with a specific kernel size. An empirical formula such as $k_{encoder} = k_{up} - 2$ has been observed to strike an optimal balance between performance and efficiency in this context. For further details on CARAFE's operations, refer to Ref. 35.

The diagram in Fig. 4 illustrates the enhanced RSSDD YOLO algorithm model, focusing on the Neck component.

3.3 Bounding box regression loss function WD-IoU

The NEU-DET dataset[25] utilized in this study is a publicly available collection specifically curated for the analysis of surface defects on steel material. Using the steel data sets, we calculated the ratio of surface defect area to image area for each picture, as detailed in Table 1. The findings revealed that approximately 24.2% of total defects have a defect area ≤5%, while about 44.8% of total defects have a defect area ≤10%. This underscores the significance of small surface defects and small- to medium-sized imperfections, such as inclusions, patches, roll scraps, and scratches, within steel data.

Wang *et al.*[36] and Sun *et al.*[37] emphasized the high susceptibility of the intersection over union (IoU) metric to small-target bounding box displacement, as illustrated in Fig. 6. When the

Table 1
Ratio of area for defect in NEU-DET dataset.

Ratio of defect area to image area	No.
≤ 1%	51
1% < *Ratio of Area* ≤ 5%	963
5% < *Ratio of Area* ≤ 10%	863
> 10%	2312

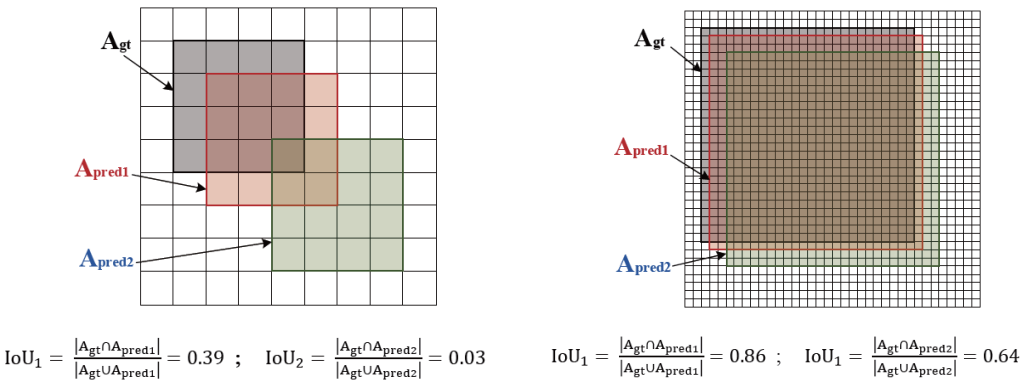


Fig. 6. (Color online) IoU changes for small targets and large targets. (a) Large targets and (b) small targets.

predicted bounding boxes in Figs. 6(a) and 6(b) are shifted by an equal number of pixels, a more pronounced change can be seen in the IoU values in Fig. 6(a), indicating that IoU computation has a greater impact on small-target displacements. Here, A_{gt} represents the true bounding box of the target, while A_{pred1} and A_{pred2} denote two different cases of predicted bounding boxes for the targets. Moreover, it is important to note that because of its reliance on the overlap ratio for position bounding box loss calculation, the CIoU loss function used in the original YOLOv5s model does not effectively capture distinguishing characteristics of small targets.

Drawing from the surface defect data of diverse steel materials in NEU-DET,[25] as delineated in Table 1, and which encompass defect targets of varying dimensions, we introduce a calculation method for the Wasserstein distance IoU (WD-IoU) loss function $L_{WD_{Iou}}$ of the predicted bounding box. This approach is tailored to concurrently accommodate targets of diverse sizes, thereby enhancing the regression accuracy of the RSSDD YOLO algorithm model and optimizing the efficiency of detecting and identifying defect targets.

The WD-IoU loss function $L_{WD_{Iou}}$ is computed as follows.

$$L_{WD_{Iou}} = 1 - WD_{IoU} \tag{8}$$

$$WD_{IoU} = \lambda_1 \cdot NGWD\left(\mathcal{N}_{gt}, \mathcal{N}_{pred}\right) + \lambda_2 \cdot Dis_{IoU} - Asp \tag{9}$$

In Eq. (9), the WD_{IoU} metric is constrained within the range of [0, 1] and $\lambda_1, \lambda_2 \in [0,1]$ denote the scale coefficients, which are tuned by the NEU-DET dataset.[25] The selection process for λ_1 and λ_2 is further described in Sect. 4. $L_{WD_{Iou}}$ is primarily composed of three constituent parts; the detailed calculation procedures are outlined for each component below.

(1) $NGWD\left(\mathcal{N}_{gt}, \mathcal{N}_{pred}\right)$.

The rectangular box, defined by the vector $R = [cx, cy, h, w]^T$ representing its center coordinates (cx, cy), height h, and width w in the image map, can be accurately characterized as a two-dimensional Gaussian distribution. This model accurately captures the varying weights of pixels within the box, with the central pixel (cx, cy) exhibiting maximum weights with h and w and gradually decreasing towards the boundary on two axes. Mathematically, this representation aligns with the two-dimensional Gaussian distribution $\mathcal{N}(\mu, \Sigma)$,

$$\mu = \begin{bmatrix} cx & 0 \\ 0 & cy \end{bmatrix}, \Sigma = \begin{bmatrix} h^2/4 & 0 \\ 0 & w^2/4 \end{bmatrix}, \tag{10}$$

where μ and Σ represent the covariance matrix of the mean vector and the Gaussian distribution.

The Wasserstein distance is a metric derived from optimal transport theory.[36,37] The Gaussian-Wasserstein distance (GWD) between two 2-dimensional Gaussian distributions $m_1 = \mathcal{N}(\mu_1, \Sigma_1)$ and $m_2 = \mathcal{N}(\mu_2, \Sigma_2)$ can be formally defined as

$$W_2^2\left(m_1, m_2\right) = \left\|\mu_1 - \mu_2\right\|_2^2 + \left\|\Sigma_1^{1/2} - \Sigma_2^{1/2}\right\|_2^2. \tag{11}$$

Thereby, the vector R representations of the ground truth bounding box and the predicted bounding box are denoted as $R_{gt} = [cx_{gt}, cy_{gt}, w_{gt}/2, h_{gt}/2]^T$ and $R_{pred} = [cx_{pred}, cy_{pred}, w_{pred}/2, h_{pred}/2]^T$, respectively. Here, $\left(cx_{\text{gt}}, cy_{\text{gt}}\right)$ represent the center coordinates of the ground truth bounding box, while w_{gt} and h_{gt} denote its width and height. Similarly, $\left(cx_{\text{pred}}, cy_{\text{pred}}\right)$ indicate the center point coordinates of the predicted bounding box, with w_{pred} and h_{pred} representing its width and height. Following Eq. (10), $\mathcal{N}_{gt}\left(\mu_{gt}, \Sigma_{gt}\right)$ is obtained for modeling the multivariate Gaussian distribution of the ground truth bounding box, as well as $\mathcal{N}_{pred}\left(\mu_{pred}, \Sigma_{pred}\right)$ for the predicted bounding box. Subsequently, in accordance with Eq. (11), we calculate the GWD $W_2^2\left(\mathcal{N}_{\text{gt}}, \mathcal{N}_{\text{pred}}\right) \in [0, \infty)$ between these distributions to quantify their dissimilarity. The NGWD is further adjusted using an exponential function:

$$NGWD\left(\mathcal{N}_{gt}, \mathcal{N}_{pred}\right) = \exp\left(-\frac{1}{\gamma}\sqrt{W_2^2\left(\mathcal{N}_{gt}, \mathcal{N}_{pred}\right)}\right) \in [0,1], \tag{12}$$

where γ is a constant determined by the average absolute magnitude of the target in the dataset.

(2) Ratio of height difference to width difference: *Asp*.

According to Ref. 38, an effective bounding box regression loss should take into account the overlapping area of the predicted bounding box and the ground truth bounding box, as well as the distance between their center points and aspect ratio. The NGWD introduced in Eq. (12) addresses both the overlapping area and the center distance between two boundary boxes simultaneously. Therefore, for incorporating the aspect ratio, we adopt the formula for calculating the ratio of height difference to width difference of the ground truth and predicted bounding boxes from the EIoU loss function.[29]

$$Asp = \frac{\rho^2\left(w_{pred}, w_{gt}\right)}{C_w^2} + \frac{\rho^2\left(h_{pred}, h_{gt}\right)}{C_h^2} \tag{13}$$

The variables w_{pred} and h_{pred} represent the width and height of the predicted bounding box, while w_{gt} and h_{gt} denote the width and height of the ground truth bounding box. The terms $\rho^2(w_{pred}, w_{gt})$ and $\rho^2(h_{pred}, h_{gt})$ signify the squared differences between the widths of the predicted and ground truth bounding boxes, as well as their heights, respectively. C_w and C_h respectively stand for the width and height of the minimum enclosing rectangle encompassing both the predicted and ground truth bounding boxes.

(3) Evaluation of positional regression for large predicted bounding boxes.

The concentration of the foreground in the center of the bounding box for small targets and the background being predominantly distributed along the edge of the bounding box necessitates the utilization of NGWD to represent the overlap area between the ground truth and predicted bounding boxes. This approach also accounts for their distance from the center point, facilitating a gradual reduction in distribution weight from the center to the edge, aligning with characteristics specific to small targets. It is important to note that large targets

may not necessarily adhere to this distribution pattern. Given that large targets constitute a significant proportion of our dataset, as shown in Table 1, further consideration is required for elements such as real overlapping areas between two bounding boxes and distances between their respective center points. The relevant calculation formula is provided below.

$$Dis_{iou} = \frac{\left|A_{pred} \cap A_{gt}\right|}{\left|A_{pred} \cup A_{gt}\right|} - \frac{\rho^2\left(A_{pred}, A_{gt}\right)}{D^2} \tag{14}$$

The first term in Eq. (14) is the IoU ratio, and the term $\rho(A_{pred}, A_{gt})$ refers to the Euclidean distance between the centroid of the predicted bounding box A_{pred} and the centroid of the ground truth bounding box A_{gt}. In this context, D signifies the diagonal length of the minimum enclosing rectangle formed by A_{pred} and A_{gt}. D is frequently employed in object detection applications to assess spatial alignment between predicted bounding boxes and their corresponding ground truth annotations.

4. Experimental Results and Analysis

4.1 Experimental dataset

The NEU-DET dataset[25] utilized in this study is a publicly available collection specifically curated for the purpose of analyzing surface defects on steel sections. It encompasses six distinct defect categories: cracks (Cr), inclusions (In), patches (Pa), pittings (Ps), rolling scraps (Rs), and scratches (Sc). Visual examples of these six defect types are shown in Fig. 7. Each category consists of a total of 300 high-resolution images measuring 200 × 200 pixels. Each type of defect

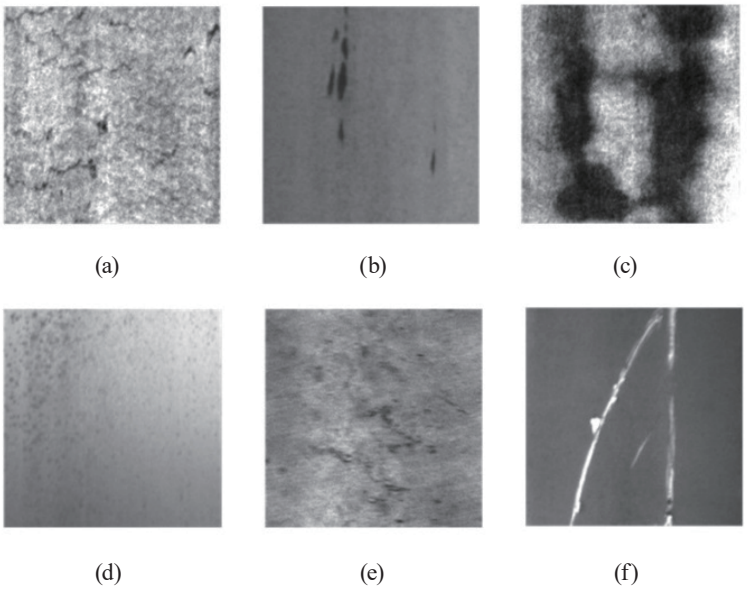


Fig. 7. Six defect categories of NEU-DET datasets (a) Cr, (b) In, (c) Pa, (d) Ps, (e) Rs, and (f) Sc.

was randomly divided into training and test sets at a ratio of 9:1, resulting in 1620 pieces allocated to the training set with 270 pieces for each type; and 180 test sets, with 30 samples for each category.

4.2 Experimental configuration setup and training strategy

The experimental setup utilized the Ubuntu 20.04 LTS operating system with 16 GB of memory, an AMD Ryzen 5 5600X CPU, and an NVIDIA GeForce RTX3060 GPU with 12 GB of VRAM. PyTorch version 1.10.1 and CUDA version 11.2 were employed for software implementation, while Python version 3.7 served as the primary programming language.

The training strategy remained consistent across all experiments, utilizing a batch size of 16 and fixed input image size of 640 × 640 pixels. The training process involved a total of 120 epochs, commencing with an initial learning rate set at 0.01 and a momentum value of 0.937 using stochastic gradient descent (SGD) as the optimizer with a regression coefficient set to optimize the model at a value of 0.0005.

4.3 Evaluation index

The identification of various types of defect on strip-steel surfaces is a critical task that requires the model to detect them quickly and accurately. In the experimental evaluation, the mean average precision (*mAP*), which is commonly used in object detection to reflect the overall accuracy of the model, the parameter count of the model, the giga floating point operations per second (GFLOPS), and the frames per second (FPS) were employed as key metrics to gauge the accuracy and real-time performance of the RSSDD YOLO algorithm model for strip-steel surface defect detection.

The equations for calculating *mAP* and *FPS* are as follows:

$$mAP = \frac{1}{n}\sum_{i=0}^{n} AP_i, \tag{15}$$

$$FPS = \frac{Frame\_Num}{Elapsed\_Time}, \tag{16}$$

where *AP* represents the average precision across all IoU categories. *Frame_Num* denotes the total number of frames to be processed, and *Elapsed_Time* signifies the total duration of model inference. A higher *FPS* indicates a higher model detection speed and greater throughput in terms of images processed per second.

4.4 Experimental analysis of BiFPN-FE module

Feature fusion is a critical strategy for enhancing the performance of model detection algorithms. In this study, we introduce the CARAFE module within the BiFPN double-path weighted feature pyramid network, as depicted in the Neck in Fig. 4, to further augment feature

incharacterization and elevate model detection accuracy. Furthermore, in terms of the original YOLOv5s model, the experimental analysis in the section delves into (1) assessing the impact of channel dimension series operation and channel summation operation on the feature fusion capability of the BiFPN module, (2) evaluating how the CARAFE up-sampling method affects the feature extraction prowess of the model, and (3) examining the synergistic effect resulting from combining the CARAFE within the BiFPN module. The experimental findings are detailed in Table 2.

By analyzing the results from Table 2, we can identify the following key points.

(1) The BiFPN utilizing channel dimension concatenation (Concat) operation demonstrates a 2.6% increase in *mAP* compared with the original network with PANet, whereas the BiFPN using channel summation operation only shows a 1.1% enhancement.
(2) Following the incorporation of the CARAFE up-sampling module, both the PANet feature fusion mode and BiFPN under the Concat operation exhibit a significant improvement in *mAP*, indicating that the CARAFE module effectively enhances the model's feature extraction capability and integrates well with the BiFPN module. Specifically, the *mAP* value of the BiFPN module utilizing Concat combined with CARAFE is 81.2%, marking a 3.2% improvement over the original Yolov5s model algorithm, whereas combining the channel summation operation with CARAFE results in only a 1.9% increase.
(3) Despite causing a reduction in the model detection speed, the CARAFE up-sampling module still meets real-time detection requirements. Experimental findings validate the effectiveness of employing Concat operation combined with CARAFE within the BiFPN structure for feature fusion, leading to its designation as a feature-enhanced BiFPN-FE module.

4.5 Experimental investigation of the WD-IoU bounding box regression loss function

The WD-IoU loss function $L_{WD_{Iou}}$ in Eqs. (8) and (9) includes two constants $\lambda_1, \lambda_2 \in [0,1]$. When ($\lambda_1$, λ_2) = (0, 1), $L_{WD_{Iou}}$ decreases to $L_{E_{Iou}}$.[29] These constants must be carefully selected to ensure that $WD_{IoU} \in [0,1]$, and the optimal performance of the RSSDD YOLO model with respect to detection and identification tasks can be achieved. The selection of λ_1 and λ_2 involves evaluating the *mAP* value derived from the model's detection results and conducting experimental analysis on the detection data to determine the appropriate combination of (λ_1, λ_2).

As illustrated in Fig. 8, the *Y* axis represents the value of λ_1 while the *X* axis corresponds to the value of λ_2. The numbers within each grid denote the *mAP* values obtained from detection results generated by the RSSDD YOLO model under various combinations of (λ_1, λ_2). According

Table 2
Experimental results of different types of feature fusion.

Type of feature fusion	*mAP* (%)	Parameter count (M)	GFLOPS (G)	*FPS*
PANet	78.0	7.02	15.8	111
PANet (+CARAFE)	78.8 (+0.8)	7.16	16.1	68
BiFPN (Add)	79.1 (+1.1)	7.17	16.4	105
BiFPN (Add+CARAFE)	79.9 (+1.9)	7.31	16.7	64
BiFPN (Concat)	80.6 (+2.6)	7.09	16.0	108
BiFPN (Concat+CARAFE)	81.2 (+3.2)	7.23	16.3	68

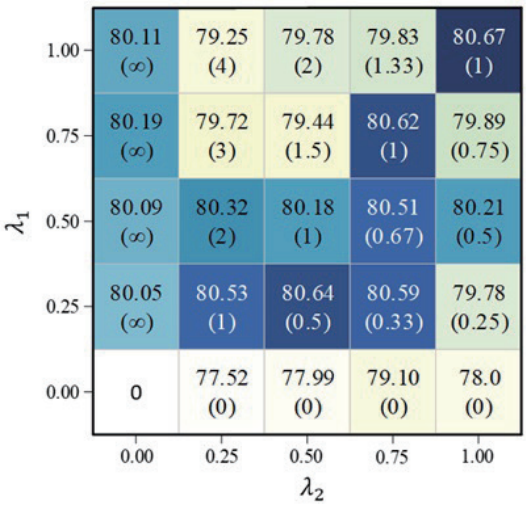


Fig. 8. (Color online) Experimental results for selection of λ_1 and λ_2.

to Table 1, small- and medium-sized targets with a defect-area-to-image-area proportion of less than 5% account for approximately 24.2% of all targets, and those with a proportion below 10% make up about 44.8%. Consequently, the ratio of targets with an area proportion below 5% to other targets is roughly 0.319, while for those with a proportion below 10%, it stands at approximately 0.812.

As depicted in Fig. 8, the incorporation of NGWD ($\lambda_1 \neq 0$) leads to a higher *mAP* value for the RSSDD YOLO model than in the case of CIoU alone ($\lambda_1 = 0$, $\lambda_2 \neq 0$). When the ratio of λ_1/λ_2 is approximately 0.319 or 0.812, the *mAP* exceeds 80.50%. Therefore, because of the findings in Fig. 8, we further investigate the selected values of (λ_1, λ_2). The strategy for value selection involves fixing one of them to correspond to a larger *mAP* value in Fig. 8 while simultaneously satisfying $\lambda_1/\lambda_2 = 0.319$ or $\lambda_1/\lambda_2 = 0.812$. Table 3 presents the *mAP* values of the RSSDD YOLO model corresponding to the ratio of λ_1/λ_2.

From the results presented in Table 3, it is evident that the model achieves a *mAP* value of 80.55% when $\lambda_1 = 0.25$ and $\lambda_2 = 0.784$, whereas with $\lambda_1 = 0.609$ and $\lambda_2 = 0.75$, the *mAP* value is 80.74%. Notably, a higher *mAP* value is observed when the ratio of $\lambda_1/\lambda_2 = 0.812$ than when $\lambda_1/\lambda_2 = 0.319$. These findings suggest that leveraging the ratio of λ_1/λ_2 as an approximation for discerning targets with area ratios below 10% versus other targets has significant potential for enhancing the model's detection accuracy. Subsequently, in our analysis of the experimental results, we selected $\lambda_1 = 0.609$ and $\lambda_2 = 0.75$ to compute the WD-IoU loss function $L_{WD_{Iou}}$.

Table 4 presents the experimental results of the RSSDD YOLO model, showcasing performance metrics obtained using three distinct loss functions, CIoU, NGWD, and WD-IoU. The AP_{50} metric represents the average precision at an IoU threshold of 0.5, while the AP_{75} indicates the average precision at an IoU threshold of 0.75. Additionally, *AP* denotes the mean average precision across various IoU thresholds from 0.5 to 0.95, with AP_S, AP_M, and AP_L denoting the mean average precisions for small, medium, and large targets, respectively. The subsequent results can be inferred from the data in Table 4.

Table 3
Experimental results with various selections of (λ_1, λ_2).

Experiment (1) with $\lambda_1/\lambda_2 = 0.319$.			Experiment (2) with $\lambda_1/\lambda_2 = 0.319$.		
λ_1	$\lambda_2 = \lambda_1/0.319$	*mAP* (%)	$\lambda_1 = 0.319\ \lambda_2$	λ_2	*mAP* (%)
0.25	0.784	**80.55**	0.08	0.25	79.28
0.50	∉[0,1]	none	0.16	0.50	80.45
0.75	∉[0,1]	none	0.239	0.75	80.51
1	∉[0,1]	none	0.319	1	79.80
Experiment (3) with $\lambda_1/\lambda_2 = 0.812$.			Experiment (4) with $\lambda_1/\lambda_2 = 0.812$.		
λ_1	$\lambda_2 = \lambda_1/0.812$	*mAP* (%)	$\lambda_1 = 0.812\ \lambda_2$	λ_2	*mAP* (%)
0.25	0.308	80.62	0.203	0.25	80.56
0.50	0.616	80.67	0.406	0.50	80.40
0.75	0.924	80.42	0.609	0.75	**80.74**
1	∉[0,1]	none	0.812	1	80.71

Table 4
Detection results obtained with different loss functions.

Loss function	*AP* (%)						*AP*	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
	Cr	In	Pa	Ps	Rs	Sc						
CIoU	48.7	78.4	**95.0**	87.1	62.8	96.0	45.0	78.0	52.2	31.9	38.8	65.3
NGWD	50.8	80.2	94.7	92.2	66.8	95.9	46.1	80.1	53.2	34.7	**39.9**	64.0
WD-IoU	**50.9**	**82.5**	94.0	**92.6**	**67.4**	**96.8**	**46.5**	**80.7**	**53.4**	**36.9**	39.2	**65.8**

(1) Compared with the CIoU loss function, the utilization of the NGWD loss function in the model results in a 2.1% increase in the AP_{50} value, which indicates an increased detection accuracy for small- and medium-area objects, with improvements of 2.8% and 1.1% in AP_S and AP_M values, respectively.
(2) The CIoU loss function within the model yields superior detection performance for large-area objects, leading to a 1.3% higher AP_L value compared with that achieved by applying the NGWD loss function.
(3) Under identical conditions, employing the WD-IoU loss function leads to the optimal detection performance for the model, where an AP_{50} of 80.7% was achieved. Furthermore, it attains the peak detection accuracy for small- and large-area targets among all three loss functions compared with using the CIoU loss function, and an AP_S of 36.9% and an AP_L of 65.8% were obtained.
(4) The use of the WD-IoU loss function results in minimal disparity in detection accuracy between small-area and large-area objects of only 28.9%, which is 4.5% lower than when using the CIoU loss function and 0.4% lower than when using the NGWD loss function.
(5) Experimental findings indicate that the WD-IoU loss function effectively facilitates the precise regression of target bounding boxes while enhancing the overall defect detection performance across various sizes.

4.6 Ablation experiments

The RSSDD YOLO model is an evolution of the original YOLOv5s model, integrating improvements in the Backbone, the feature fusion module, and the bounding box regression loss

function. A series of ablation experiments have been meticulously designed to evaluate the impact of these enhancements on the performance of the RSSDD YOLO model and to explore potential interactions between them, using the original YOLOv5s model as a reference point. The results of these experiments are detailed in Table 5 for comprehensive analysis.

(1) Upon the exclusive utilization of the BiFPN-FE module, there was an increase in the parameter count. However, compared with the original YOLOv5s model, a 3.2% rise in the *mAP* value was observed. This indicates that the BiFPN-FE module effectively augments the feature extraction capability and proficiently learns crucial features, thereby enhancing the detection performance of the model.

(2) Substituting the CIoU loss function with the WD-IoU loss function in the original YOLOv5s model ensures the efficient fitting of the ground truth bounding box for both small- and large-target predicted boxes during model training. Without altering the parameter count of the model or floating-point operations, there is a 2.7% increase in *mAP*.

(3) The introduction of the RepVGG–Light_N module resulted in a 2.6% increase in *mAP* compared with the original network. This signifies that the module adeptly extracts context information related to defects and enhances the overall detection effectiveness within the model. Furthermore, it can be simplified into a single-branch structure during the inference stage to improve the model detection speed.

(4) According to the results of experiment 8, when juxtaposed with the original YOLOv5s model, our proposed RSSDD YOLO model yields a 4.1% increase in the *mAP* value while achieving a detection speed of 68 FPS. It demonstrates superior detection accuracy with fewer parameters and shorter floating-point operation times than its predecessor while maintaining exceptional real-time detection capabilities.

Furthermore, the empirical results obtained from experiments 5 to 7 have demonstrated that the various enhancement strategies proposed in this study can be effectively integrated. The inclusion of the BiFPN-FE module leads to a higher average detection accuracy for the RSSDD YOLO model than when the BiFPN-FE module is not included. This indicates that the BiFPN-FE module makes a significant contribution to enhancing the detection accuracy of the original model. This finding aligns with experimental data showing that incorporating only the BiFPN-FE module results in the largest increase in the *mAP* value.

The observed improvement can be attributed to several factors including the capability of the BiFPN-FE module to adaptively learn from salient features, assimilate richer semantic

Table 5
Results of ablation experiments.

No.	RepVGG–Light_N	BiFPN-FE	WD-IoU	*mAP* (%)	Parameter count (M)	GFLOPS (G)	*FPS*
1				78.0	7.0	15.8	111
2	√			80.6 (+2.6)	6.7	15.0	114
3		√		81.2 (+3.2)	7.2	16.3	68
4			√	80.7 (+2.7)	7.0	15.8	112
5	√	√		81.9 (+3.9)	6.9	15.5	66
6	√		√	81.1 (+3.1)	6.7	15.0	116
7		√	√	81.5 (+3.5)	7.2	16.3	70
8	√	√	√	82.1 (+4.1)	6.9	15.5	68

information, and aggregate contextual details surrounding image features. As a result, the BiFPN-FE module enables a more targeted focus on defect regions within the network architecture and proves more adept at addressing defect detection tasks than its predecessor.

In addition, to comprehensively evaluate the performance of the RSSDD YOLO model algorithm proposed in this study, we compared the detection capabilities of the RSSDD YOLO model and the original YOLOv5s model. As depicted in Fig. 9, it is evident that the RSSDD YOLO model excels at detecting various defects, such as Cr, In, Rs, and Sc, which are undetectable by the original YOLOv5s model. Furthermore, the RSSDD YOLO model demonstrates its capability to successfully identify small target defects overlooked by the original YOLOv5s model in In, Rs, and Sc samples. These results indicate that the RSSDD YOLO model algorithm exhibits superior detail capture capability compared with the original YOLOv5s model. Additionally, while multiple defects were identified by the original YOLOv5s model in the right half of the Pa sample, only one defect was accurately identified by the RSSDD YOLO model. Thus, its advantage in global information extraction is highlighted.

4.7 Contrast experiments

To evaluate the progress and effectiveness of the RSSDD YOLO model proposed in the study, comparative experiments were conducted with SSD, YOLOv3, and YOLOv4, as well as original algorithms such as YOLOv5s, YOLOX, and YOLOv7 on the NEU-DET dataset. The experimental results are presented in Table 6 for assessment.

Upon a comprehensive analysis of the data presented in Table 6, several key insights were gleaned.

(1) The *mAP* of the RSSDD YOLO model reached an impressive 82.1%, surpassing those of other mainstream single-stage object detection models. Notably, the RSSDD YOLO model demonstrated superior accuracy in detecting Pa, Rs, and Sc defects compared with alternative algorithm models.

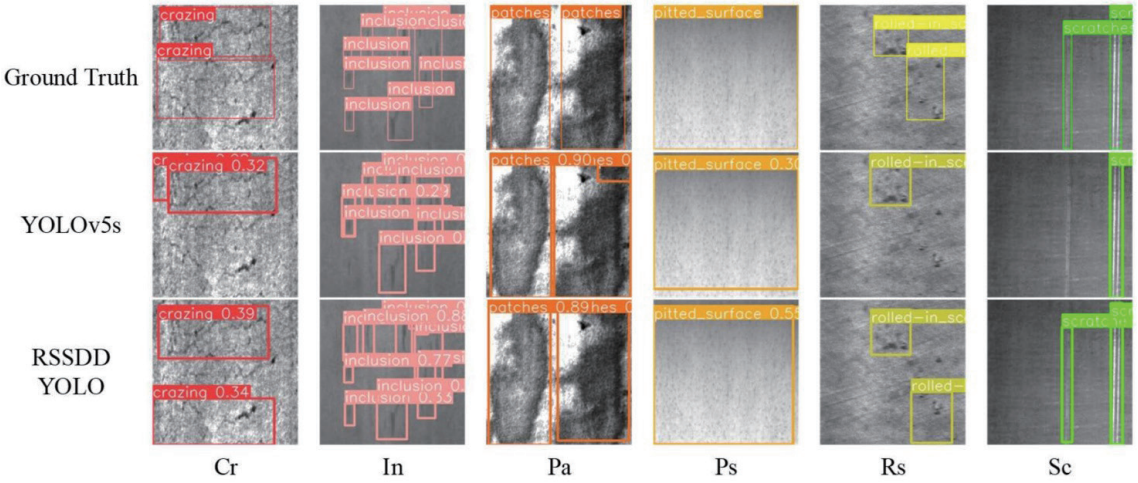


Fig. 9. (Color online) Detection results obtained using RSSDD YOLO and YOLOv5s models.

Table 6
Result of comparative experiments with different object detection models.

Method		*AP* (%)						*mAP*(%)	Parameter count (M)	GFLOPS (G)	*FPS*
		Cr	In	Pa	Ps	Rs	Sc				
1	SSD	35.2	**91.3**	92.2	53.3	50.3	34.1	59.4	13.7	15.4	50
2	YOLOv3-SPP	38.9	65.4	71.7	88.2	49.4	67.5	63.5	62.6	117.1	41
3	YOLOv4	38.4	74.4	94.3	90.6	57.6	88.7	74.0	22.5	52.0	45
4	YOLOv5s	48.7	78.4	95.0	87.1	62.8	96.0	78.0	7.0	15.8	111
5	YOLOXs	51.2	80.9	93.3	89.4	64.2	95.6	79.1	8.9	26.8	84
6	YOLOv7-tiny	39.6	77.4	95.5	**93.5**	52.7	87.5	74.4	6.0	13.1	141
7	YOLOv7s	**60.2**	81.8	96.4	88.5	63.3	94.7	80.8	6.2	103.2	57
8	RSSDD YOLO	58.9	82.0	**97.2**	92.4	**66.1**	**96.2**	**82.1**	6.9	15.5	68

(2) Furthermore, for the comparative evaluation of the scale of various detection models, the parameter count and GFLOPS serve as pivotal metrics. In contrast with SSD, YOLOv3-SPP, YOLOv4, YOLOv5s, and YOLOXs, the RSSDD YOLO model reduced the parameter count by 6.8, 55.7, 15.6, 0.1, and 2 M, respectively. When juxtaposed against the lightweight models such as YOLOv7-tiny and YOLOv7s, the increase was merely 0.9 and 0.7 M, respectively, signifying that the RSSDD YOLO algorithm model ensures heightened detection accuracy while augmenting the parameter count.

(3) To effectively gauge the object detection speed of each model, the *FPS*, which is provided in Eq. (16), serves as a crucial metric. The RSSDD YOLO model's detection speed is lower by 43 FPS than that of YOLOv5s, 16 FPS than that of YOLOXs, and 73 FPS than that of the lightweight YOLOv7-tiny model. However, the RSSDD YOLO model still maintains a higher detection speed than the other types of detection model algorithm.

To summarize, the RSSDD YOLO model not only improves the detection accuracy, but also ensures real-time algorithm execution, rendering it suitable for deployment in defect detection environments necessitating high precision and real-time performance.

5. Conclusions

In the context of the formidable challenges associated with achieving both high precision and high speed in detecting surface defects on strip steel within a steel production plant, we introduced an improved RSSDD YOLO algorithm model based on the YOLOv5s model. We proposed three improvement strategies.

(1) A RepVGG–Light_N module was introduced to replace the C3_1 module in the Backbone, thereby augmenting the feature extraction capability while preserving reasoning speed, equilibrium detection accuracy, and detection speed.

(2) An enhanced BiFPN-FE structure for feature fusion was designed to improve the multiscale prediction capability for strip-steel surface defects.

(3) A bounding box regression loss function of WD-IoU was proposed to increase the detection accuracy for small target defects.

The experimental results validated the efficacy of the RSSDD YOLO algorithm model, which exhibited superior detection accuracy compared with other mainstream object detection models and effectively mitigated issues such as the false detection and missed detection of small targets. Furthermore, the algorithm provided real-time detection results.

Acknowledgments

This work was carried out as part of the Major Science and Technology Projects of Fujian Province (Grants no. 2022-HZ-026025, 2023-H-6036, and 2023-T-5001), the Program for Innovative Research Team in Science and Technology in Fujian Province University, the Production and Research Collaboration with Innovative in Key Scientific and Technological Project of Sanming City (Grant no. 2022-G-17), and the Operational Funding of the Advanced Talents for Scientific Research (Grant no. 19YG04) of Sanming University. The authors also acknowledge the support from the University of Science and Technology Beijing and the School of Mechanical and Electric Engineering, Sanming University.

References

1 B. Tang, L. Chen, W. Sun, and Z. Lin: IET Image Proc. **17** (2023) 303. https://doi.org/10.1049/ipr2.12647
2 Q. Luo, X. Fang, L. Liu, and C. Yang: IEEE Trans. Instrum. Meas. **69** (2020) 626. https://doi.org/10.1109/TIM.2019.2963555
3 S. Ren, K. He, R. Girshick, and J. Sun: IEEE Trans. Pattern. Anal. Mach. Intell. **39** (2017) 1137. https://doi.org/10.1109/TPAMI.2016.2577031
4 M. Lokanath, K. S. Kumar, and E. S. Keerthi: IOP Conf. Ser. Mater. Sci. Eng. **263** (2017) 052028. https://doi.org/10.1088/1757-899X/263/5/052028
5 J. Z. Pan, C. H. Yang, L. Wu, and W. H. Tang: Sens. Mater. **35** (2023) 4653. https://doi.org/10.18494/SAM4589
6 B. Rahmat, Y. V. Via, and A. Wasian: Advances in Intelligent Systems and Computing, E. Joelianto, A. Turnip, A. Widyotriatmo, Eds. (Springer, Singapore, 2021) **1291** (2021) 81. https://doi.org/10.1007/978-981-33-4062-6_8
7 X. Gong, X. Zhang, and R. Zhang: Comput. Electron. Agric. **203** (2022) 107461. https://doi.org/10.1016/j.compag.2022.107461
8 L. Zhao, T. Tohti, and A. Hamdulla: Signal Image Video Process **17** (2023) 4435. https://doi.org/10.1007/s11760-023-02677-x
9 Y. Yang, Y. Zhou, and N. U. Din: Appl. Sci. **13** (2023) 2402. https://doi.org/10.3390/app13042402
10 Z. Diao, X. Huang, and H. Liu: Int. J. Intell. Syst. **2023** (2023) 8879622. https://doi.org/10.1155/2023/8879622
11 Y. Sun, J. Sun, B. Yuan: Trans. Chinese Soc. Agric. Eng. **39** (2023) 152. https://doi.org/10.11975/j.issn.1002-6819.202308043
12 J. Chen, X. Zhang, Y. Tang: The J. Supercomput. **80** (2024) 2948. https://doi.org/10.1007/s11227-023-05585-6
13 T. Xu, L. Ren, T.W. Shi, Y. Gao, J.B. Ding, and R.C. Jin: Inf. Technol. Control **52** (2023) 966. https://doi.org/10.5755/j01.itc.52.4.34039
14 W. Liu, D. Anguelov, D. Erhan: Proc. 2016 Eur. Conf. Comput. Vis. (ECCV, 2016) **9905** (2016) 21. https://doi.org/10.1007/978-3-319-46448-0
15 Z. B. Yin, F. Y. Liu, and H. Geng: PLOS ONE **19** (2024) e0296314. https://doi.org/10.1371/journal.pone.0296314
16 K. Yan and Z. Zhang: IEEE Access **9** (2021) 150925. https://doi.org/10.1109/ACCESS.2021.3125703
17 J. Cheng, X. Duan, and W. Zhu: Comput. Eng. Appl. **57** (2021) 252. https://doi.org/10.3778/j.issn.1002-8331.2104-0324
18 Y. Cao, M. Wu, and L. Xu: J. Graphics **44** (2023) 335. https://kns.cnki.net/kcms/detail/10.1034.T.20221012.1820.004.html
19 J. Shi, J. Yang, and Y. Zhang: Electron. **11** (2022) 3735. https://doi.org/10.3390/electronics11223735
20 Y. Jiang: Math. Biosci. Eng. **20** (2023) 19858. https://doi.org/10.3934/mbe.2023879
21 Y. Xu, Z. Ding, and W. Li: J. Electr. Comput. Eng. **2023** (2023) 5399616. https://doi.org/10.1155/2023/5399616

22 G. Li and A. A. Mokhtarzadeh: J. Phys. Conf. Ser. **2467** (2023) 012005. https://doi.org/10.1088/1742-6596/2467/1/012005
23 C. Li, A. Xu, and Q. Zhang: IEEE Access **12** (2024) 37643. https://doi.org/10.1109/ACCESS.2024.3374869
24 X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun: Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR, 2021) 13733. https://doi.org/10.1109/CVPR46437.2021.01352
25 K. Song and Y. Yan: Appl. Surface Sci. **285** (2013) 858. https://doi.org/10.1016/j.apsusc.2013.09.002
26 G. Jocher, K. Nishimura, and T. Mineeva: Available online: https://github.com/ultralytics/yolov5/releases/tag/v6.0 (accessed on 1 May 2023).
27 T.Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie: Proc. 2017 IEEE Conf. Comput. Vis Pattern Recognit. (CVPR, 2017). https://doi.org/10.1109/CVPR.2017.106
28 S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia: Proc. 2018 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR, 2018) 8759. https://doi.org/10.1109/CVPR.2018.00913
29 Y. Sun, J. Wang, and H. Wang: IEEE Access **12** (2024) 37363. https://doi.org/10.1109/ACCESS.2024.3359433
30 X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun: Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR, 2021) 13733. https://doi.org/10.1109/CVPR46437.2021.01352
31 Q. Zhang, J. Shu, C. Chen, Z. Teng, Z. Gu, F. Li, and J. Kan: Front. Med. **10** (2023) 1233724. https://doi.org/10.3389/fmed.2023.1233724
32 Q. Chen, Y. Wang, and T. Yang: Proc 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR, 2021). https://doi.org/10.1109/CVPR46437.2021.01284
33 M. Tan, R. Pang, and Q.V. Le: Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR, 2020). https://doi.org/10.1109/CVPR42600.2020.01079
34 J. Lin, M. Jiang, Y. Pang, H. Wang, Z. Chen, C. Yan, Q. Liu, and Y. Wang: Proc. 2022 Int. Conf. Sustain. Comput. Data Commun. Syst. (ICSCDS, 2022) 1455. https://doi.org/10.1002/cpe.6320
35 J. Wang, K. Chen, R. Xu, Z. Liu, C. C. Loy, and D. Lin: Proc. 2019 IEEE/CVF Int. Conf. Comput. Vis. (ICCV, 2019). https://doi.org/10.1109/ICCV.2019.00310
36 J. Wang, C. Xu, W. Yang, and L. Yu: Comput. Vis. Pattern Recognit. (2021) arXiv:2110.13389. https://doi.org/10.48550/arXiv.2110.13389
37 K. Sun, Z. Wu, M. Wang, J. Shang, Z. Liu, D. Zhao, and X. Luo: J. Mar. Sci. Eng. **12** (2024) 333. https://doi.org/10.3390/jmse12020333
38 K. Su, L. Cao, B. Zhao, N. Li, D. Wu, and X. Han: Neural Comput. Appl. **36** (2024) 3049. https://doi.org/10.1007/s00521-023-09133-4