A Study on Surface Defect Detection Algorithm of Strip Steel Based on YOLOv8n

Haozhe Sun [1*], Fengzhi Dai [1], Junjin Chen [2]

[1] *Tianjin University of Science and Technology, Tianjin, China*

[2] *SMC (Beijing) Manufacturing Co., LTD., Beijing, China*

*E-mail: * 2921407938@qq.com*

www.tust.edu.cn

Abstract

Hot rolled steel strip has been extensively applied in industrial production and processing due to its outstanding properties. Nevertheless, during the production procedure, as a result of technological constraints, defects will inevitably occur on the surface of the steel strip, significantly influencing the performance and safety of the steel strip. Hence, how to detect the surface defects of steel strips has turned into the key point. In this paper, an enhanced YOLOv8n network model is proposed to make it applicable for the surface defect detection tasks of hot rolled steel strips. The improved model introduces Dynamic Snake Convolution and Efficient Multi-Scale Attention Module. The average precision of the improved model is 6.8 percentage points higher than that of the original model.

Keywords: Deep learning, Surface defect detection, YOLOv8n, Dynamic Snake Convolution, Efficient Multi-Scale Attention Module

1. Introduction

Steel production technology serves as a crucial indicator for gauging the development level of a country's iron and steel industry. Among various types of steel, strip steel is extensively utilized in the automotive, construction, and aviation sectors because of its high dimensional accuracy, excellent surface quality, ease of processing, material-saving properties, and other advantages, and the demands for its product quality are also higher. Nevertheless, due to the influence of multiple factors such as raw materials, the rolling process, and system control, the surface of hot rolled steel strips frequently presents pitting, scratches, inclusions, rolling oxide scale, cracks, and other defects. These defects will, to varying degrees, impact the main performance indicators of the steel plate, such as wear resistance, fatigue resistance, corrosion resistance, and electromagnetic characteristics. Therefore, how to detect the surface defects of steel strips and analyze their causes to enhance product quality has become the key. The commonly used detection methods are mainly divided into three categories: manual detection, theoretical mechanism detection and machine vision detection [1].

Before 1970, strip surface defect detection mainly relied on manual visual observation. The advantage of this approach lies in its lower cost. However, due to the large scale of metal products production, surface defects are often small, especially prone to missed detection, false detection, and the speed is slow. The physical mechanism detection methods commonly used are: infrared detection, eddy current detection and magnetic leakage detection methods. The accuracy of such methods is generally greatly affected by the environment. With the development of science and technology, machine vision technology has been proposed and applied to defect detection tasks. Machine vision detection can be divided into traditional image processing detection methods and deep learning detection methods. The main principle of traditional image processing detection method is to identify defects by feature extraction of the preprocessed image. These methods rely on human-designed features. Moreover, the types of defects are complex, the differences between samples of different categories are large, and the production environment is complex, which leads to the poor adaptability and generalization ability of the method in the actual production link, and the stability of the detection results is not high. Visual inspection methods based on deep learning can adaptively learn defect features, thereby accurately and rapidly locating defects. The performance of the model tends to enhance as the number of data samples grows. Even when the amount of data is insufficient, image enhancement can also be employed to expand the sample. The trained model possesses strong universality, high robustness and fast detection speed, and is applicable to defect detection issues in industrial production. In this paper, the original model of YOLOv8n is used to improve it to enhance the ability to capture the feature information of small targets. After improving the model, the experiment is carried out, and the results are analyzed.

The rest of this paper is organized as follows. The second section introduces the structure of YOLOv8n network model. In the third part, the structure and principle of Dynamic Snake Convolution and Efficient Multi-Scale Attention Module are introduced. The fourth section gives the process of the experiment to verify the effectiveness of the improved deep learning model. The fifth part summarizes the main contents of this paper.

2. Structure and Working Principle of YOLOv8n

YOLO (You Only Look Once) [2] series object detection algorithm is a single-stage object detection algorithm. With the continuous change of YOLO model version, its speed and accuracy have been significantly improved. YOLOv8 is a widely used and newer version. In order to adapt to different application scenarios and hardware resources, it is divided into YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l and YOLOv8x versions. These versions mainly improve the detection performance by increasing the depth and width of the network. The main body of YOLOv8 network is divided into three modules: Backbone network, Neck network and Head network. Among them, the backbone network was responsible for feature extraction and passed to the neck network. The neck network fused and enhanced the extracted features. The head network uses the feature information derived from the previous network to predict the image. The YOLOv8 network structure is shown in Fig. 1.

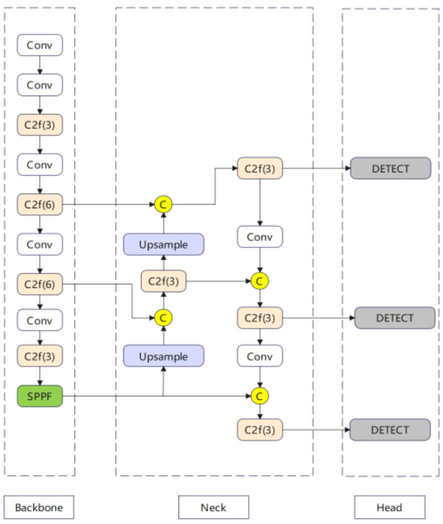

Fig. 1 Schematic diagram of YOLOv8 network structure

The backbone network of YOLOv8 adopts the CSPDarkNet [3] structure, which is constituted by convolutional modules CBS and C2f (CSPLayer With 2Conv), as well as Spatial Pyramid Pooling Fast (SPPF). CBS is a composite convolution module and serves as the most fundamental part of the entire backbone network. It is primarily composed of a two-dimensional convolution layer Conv, a two-dimensional batch normalization layer BatchNorm, and an activation function SiLU. In the C2f module, the input feature map initially traverses a convolution module on the trunk for feature extraction, and subsequently through n bottleneck modules to extract higher-level abstract feature information. These bottleneck modules employ the Split method to achieve cross-layer connection, while adding more parallel gradient flow branches. Thus, the expressiveness and performance of the entire network model are enhanced, and the ability of the model to capture complex features is improved without sacrificing computational efficiency, which contributes to enhancing the accuracy of the model in target detection tasks. The SPPF fast spatial pyramid pool module is situated at the last layer of the backbone network, enabling the feature map to acquire a more abundant receptive field, thereby strengthening the expression capacity of the network, as well as accelerating the operation speed and enhancing the efficiency of target detection. The neck network of YOLOv8 uses the structure of PAFPN, which is improved by the fusion of FPN and PAN. Its main purpose is to solve the problem of insufficient feature pyramid network in multi-scale detection tasks in object detection. The head network of YOLOv8 uses the Decoupled-Head structure, which uses two parallel branches to extract category features and location features respectively, and then uses a layer of CBS convolution and two-dimensional convolution to complete the classification and localization tasks.

3. Improve YOLOv8n Steel Surface Defect Detection Algorithm

3.1 Dynamic snake convolution

Due to the presence of minor flaws like fine cracks and rolled oxide scale in the NEU-DET [4] dataset, along with the insignificant difference between them and the background gray value, the detection accuracy is relatively low. To enhance the detection accuracy of the model for such minute defects, this paper adopts dynamic snake convolution to replace the original C2f module in YOLOv8n. Dynamic Snake Convolution (DSC) [5] is a technique originated from clinical medicine. The advent of this technology significantly contributes to the feature extraction work of cardiovascular targets such as slender, tortuous, and irregular ones. In the surface defects of steel strips investigated in this paper, the defects like fine cracks with poor detection results share similar structures, thereby this module is introduced to improve the detection accuracy.

Given the standard 2D convolution coordinates designated as K, the central coordinate is $Ki = (xi, yi)$. A 3×3 kernel K with a dilation of 1 is expressed as:

$$K = \{(x-1, y-1), (x-1, y), \cdots, (x+1, y+1)\} \quad (1)$$

To enable the convolution kernel to better focus on defect targets with complex geometric features, an offset Δ is introduced. Generally, the fixed convolution layer applies the same convolution operation to the feature maps of different levels, that is, the position of the sampled pixels is fixed. For instance, the convolution kernels of all CBS modules in the original YOLOv8n model are 3×3, and the feature information sampled by this approach will incorporate many background features. By adding a direction parameter to each convolution element, the deformable convolution kernel with a base of 3×3 can adaptively modify its shape during the training process to better match the characteristics of the input data. The characteristics of the deformable convolution kernel are presented in Fig. 2. The convolution kernel will extend from the green point area to the surrounding to change its shape, forming a new convolution kernel dominated by blue points. The middle arrow part is the introduced offset Δ.

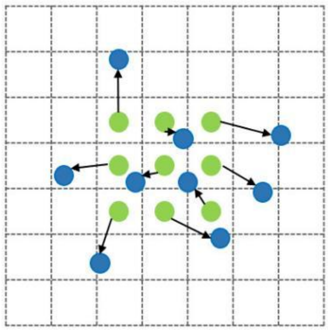

Fig. 2 Schematic diagram of deformable convolution

In DSC, the standard convolution kernel is elongated along the X-axis and Y-axis respectively. As presented in Fig. 3, the size of the convolution kernel subsequent to elongation is set at 9×9. Taking the X-axis direction as an exemplar, the specific location of each grid in K is expressed as $K_{i\pm c} = (x_{i\pm c}, y_{i\pm c})$, where $c = \{0,1,2,3,4\}$ indicates the horizontal distance from the central grid. The determination of each grid position $K_{i\pm c}$ within the convolution kernel K is a cumulative procedure. Commencing from the central position K_i, the position deviating from the central grid is contingent upon the position of the preceding grid: K_{i+1} has an offset Δ relative to K_i, and$\Delta = \{\delta | \delta \epsilon [-1,1]\}$.

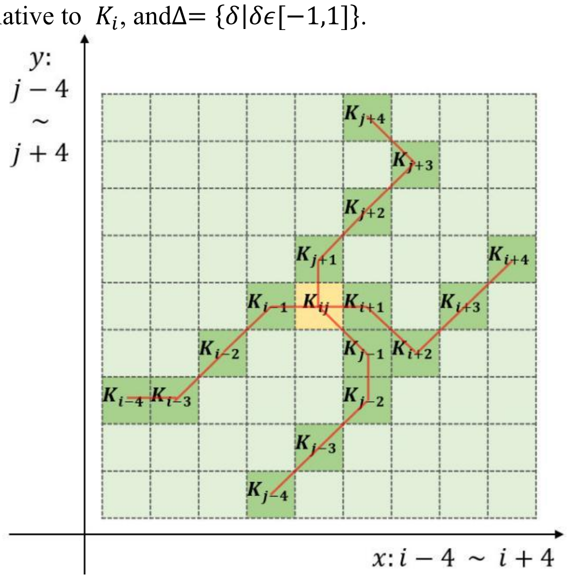

Fig. 3 Stretched convolution kernel

To guarantee that the receptive field does not deviate excessively from the target feature while the model adaptively acquires the defect feature offset, an iterative tactic is implemented to successively select the position that can be chosen at the next offset for each target to be processed, thereby ensuring the continuity of the convolution kernel's attention and preventing the receptive field from shifting too far due to the deformation of the convolution kernel. As depicted in Fig. 4, taking the X-axis direction as an example, the next offset position K_{i-1} for K_i is identified first, and its location is determined before proceeding to find the position of K_{i-2}.

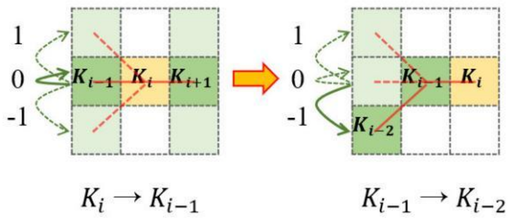

Fig. 4 Schematic diagram of K_i offset sequentially in the x-axis direction

The convolution kernel becomes Eq. (2) in the X-axis direction:

$$K_{i\pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = (x_i + c, y_i + \sum_i^{i+c} \Delta y) \\ (x_{i-c}, y_{i-c}) = (x_i - c, y_i + \sum_{i-c}^{i} \Delta y) \end{cases} \tag{2}$$

The convolution kernel becomes Eq. (3) in the Y-axis direction:

$$K_{j\pm c} = \begin{cases} \left(x_{j+c}, y_{j+c}\right) = (x_j + \sum_j^{j+c} \Delta x, y_j + c) \\ \left(x_{j-c}, y_{j-c}\right) = (x_j + \sum_{j-c}^{j} \Delta x, y_j - c) \end{cases} \tag{3}$$

DSConv is designed to better adapt to the slender tubular structure based on the dynamic structures so as to better perceive the key features.

3.2 Efficient multi-scale attention module

Efficient Multi-Scale Attention(EMA) [6] is a novel attention mechanism specifically designed for computer vision tasks that aims to reduce computational overhead while preserving key information in each channel. The EMA module reconstructs part of the channels into batch dimensions and groups the channel dimensions into multiple sub-features so that the spatial semantic features are evenly distributed within each feature group. In addition to encoding the global information to recalibe the channel weights in each parallel branch, the output features of the two parallel branches are further aggregated by cross-dimensional interaction to capture pixel-level pairwise relationships.

A schematic of the Efficient Multi-Scale Attention (EMA) module is shown in Fig. 5. "g" denotes the number of groups into which the input channels are divided. X Avg Pool" and "Y Avg Pool" represent the one-dimensional horizontal and vertical global pooling operations, respectively. In the EMA module, the input is first grouped and then processed through different branches: one branch for one-dimensional global pooling and the other for feature extraction via a 3x3 convolution. The output features of the two branches are then modulated by a sigmoid function and a normalization operation, and finally merged by a cross-dimensional interaction module to capture the pixel-wise pairwise relationship. After the final sigmoid conditioning, the feature maps are output to enhance or weaken the original input features to obtain the final output.

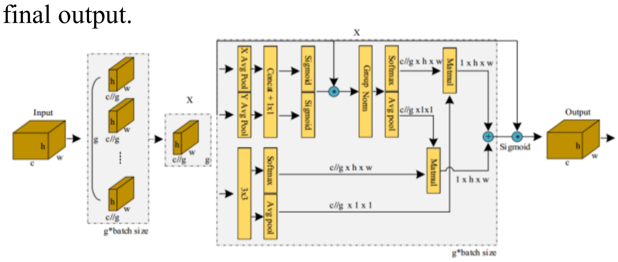

Fig. 5 Schematic diagram of EMA network structure

4. Experimental Setups

4.1 Processing of the dataset

In order to detect the common defects on the surface of steel strips, this paper uses the public data set for training. The data set is NEU-DET, the hot rolled steel strip surface defect detection data set organized by Professor Song Kecheng's team in Northeastern University. The dataset includes six defect types: crazing, inclusion, patches, pitted_surface, rolled-in scale, and scratches. There are 300 images of each defect, with a total of 1800 grayscale images. The original resolution of each image is 200×200 pixels.

The original data set of steel strip surface defects is in VOC data format, namely, the label file is in xml format. To facilitate the training of YOLO, a one-stage object detection model with high detection accuracy and fast detection speed for small objects, this paper converts the label file of the original data set into the txt label file used by YOLO through writing Python script code.

The original dataset only contains 1800 grayscale images, and there are few samples of each type of defect. In the actual training of deep learning network, it may cause overfitting phenomenon, resulting in that the model cannot fully learn the features of each type of defect, resulting in the reduction of defect detection accuracy. In order to solve the problem of sparse data sets, this paper expands the original steel strip defect data set, and uses image flipping, image rotation, adjusting image brightness and contrast to enhance the image, and expands the steel strip surface defect data set to 5400 images according to the ratio of 1:3.

After the expansion of the dataset, the corresponding label of each image also needs to be changed. Since the label content of the dataset is the defect category number and the normalized rectangular box information, the original label information is set as Eq. (4) here:

$$f = (i, x, y, w, h) \tag{4}$$

Where i is the defect category number, (x, y) is the center point coordinate of the defect rectangular box, and (w, h) is the width and height of the normalized defect rectangular box. Then the label information after horizontal flipping is shown in Eq. (5):

$$f' = (i, 1 - x, y, w, h) \tag{5}$$

The label information after vertical flipping is shown in Eq (6):

$$f'' = (i, x, 1 - y, w, h) \tag{6}$$

The 5400 images in the augmented dataset were divided into training set, validation set and test set according to the ratio of 8:1:1. That is, 4320 images in the training set, 540 images in the validation set, and 540 images in the test set.

4.2 Experimental environment and parameter setting

The network was built using the PyTorch deep learning framework on Windows 10 system. The processor used was the 10th Gen Intel(R) Core(TM) i9-10900K running at 3.70GHz. The graphics card employed was the NVIDIA GeForce RTX 2080Ti. The Python version utilized was 3.11, the PyTorch version utilized was 2.0.1, with a CUDA version of 11.8.

The training process consisted of 300 epochs, with the optimizer set to SGD. The initial learning rate was 0.01, workers was 4, and the batch size was 32. Set the model accuracy no longer rising within 50 rounds to trigger the early stop mechanism.

4.3 Evaluation index

mAP, Params, FLOPs were used as evaluation indicators.

Precision, The accuracy of the reaction model can be calculated by:

$$Precision = TP/TP + FP \tag{7}$$

where TP indicates that the object to be detected is correctly detected, and FP indicates that the non-object to be detected is correctly detected.

Recall, Represents the proportion of samples that are correctly predicted to belong to a class by the model among all samples that actually belong to a class, and is calculated as follows:

$$Recall = TP/TP + FN \tag{8}$$

where FN indicates that the object to be detected is incorrectly detected.

mAP@0.5, Represents the average precision value under different Intersection over Union (IoU) thresholds, calculated as:

$$mAP = \frac{1}{n}\sum_{i=0}^{n} AP(i) \tag{9}$$

where c is the total number of categories of images, i is the number of detections, and AP is the average recognition precision of a single category. This metric provides a comprehensive evaluation of the algorithm's performance. mAP@0.5 represents the average precision case with an IoU threshold of 0.5, and mAP@0.5:0.95 represents the average precision with an IoU threshold ranging from 0.5 to 0.95.

Params, is the sum of trainable parameters in a model, usually including the weights of convolutional layers, the weights of fully connected layers, and the bias term, and is calculated as follows:

$$Params = C_o \times (k_w \times k_h \times C_i + 1) \quad (10)$$

FLOPs is the number of floating-point operations required for the model to perform one forward propagation, which represents the computational burden of the model in a single inference process, and can be used to measure the complexity of the algorithm, calculated as:

$$FLOPs = 2 \times (C_i \times k_w \times k_h - 1) \times H \times W \times C_o \quad (11)$$

In the formula, H represents the height of the feature map, W represents the width of the feature map, and if the convolution kernel has a bias term, the $+1$ term in the parentheses is removed.

In this paper, mAP@0.5, Params and FLOPs are used as the main evaluation metrics of model detection accuracy.

4.4 Experimental result

Through the analysis of DSC and EMA modules mentioned above, all C2f modules in the original structure of YOLOv8 are replaced with DSC modules, and EMA attention mechanism is added to the 16th, 20th and 24th layers of the original model. The improved model structure diagram is shown in . The experimental results of the original YOLOv8n model and the experimental results of the improved YOLOv8_DSC_EMA model are compared to verify the influence of the improved network using DSConv and EMA modules on the surface defect detection results of steel strips, as shown in Table 1.

It can be seen from Table 1 that the average precision of YOLOv8_DSC_EMA model reaches 80.3%, which is 6.8 percentage points higher than that of YOLOv8n model. This shows that by replacing the deep convolution module of the original YOLOv8n with dynamic snake-like convolution, the extraction ability of feature information in the deep network is improved, especially for the detection accuracy of small target defects with tubular structure such as small cracks. In terms of the number of parameters, since the calculation process of snake convolution is more complex than that of C2f module, the number of parameters will increase and the amount of calculation will also increase slightly. Therefore, EMA is added to lightweight the model while retaining the key information of each channel. It is used for recalibration of channel weights, which enhances the ability of feature representation. Through comparison, it is found that the time used for image preprocessing and inference is little different from the original YOLOv8n network, but mAP@0.5 is greatly improved.

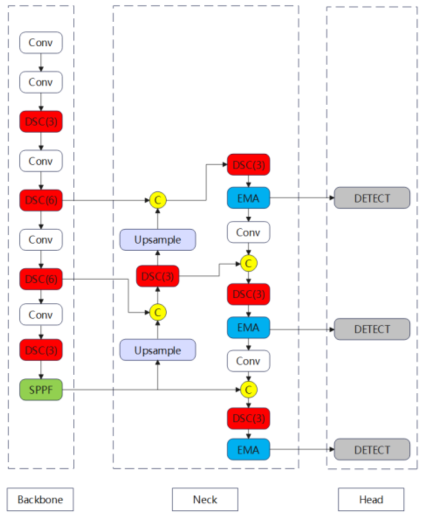

Fig. 6 Schematic diagram of YOLOv8_DSC_EMA network structure

Table 1 Comparison of model experiments for improving convolutional modules

Model	mAP @0.5	Params /M	FLOPs /G
YOLOv8n	0.735	3.01	8.1
YOLOv8_ DSC_EMA	0.803	3.91	9.9

The Precision-Recall curves obtained by training the original YOLOv8n model and YOLOv8_DSC_EMA model are shown in Fig. 7 and Fig. 8, respectively. By comparing the two groups of curves, it is found that the accuracy of the improved model has increased significantly.

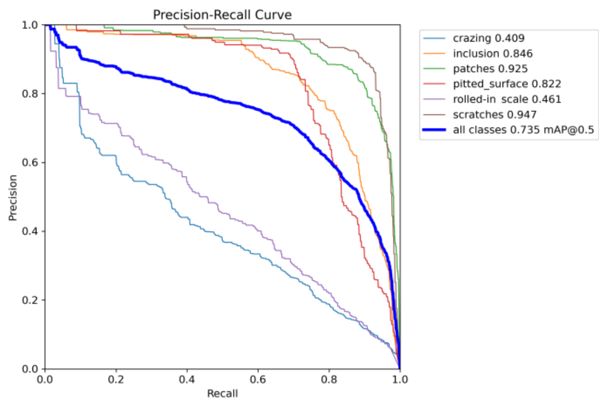

Fig. 7 Precision-Recall curve of the YOLOv8n model

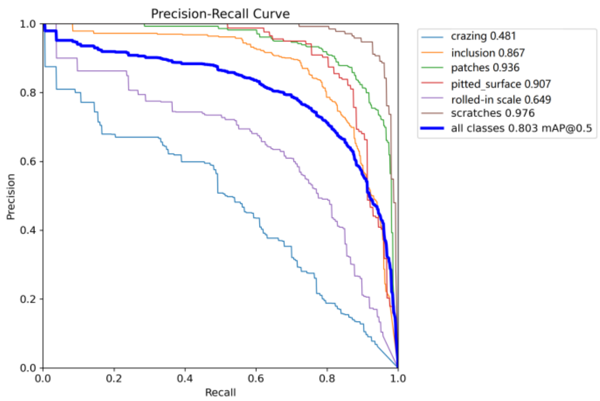

Fig. 8 Precision-Recall curve of the YOLOv8_DSC_EMA model

The comparison between the detection effect of the improved model and the detection effect of the pre-improved model is shown in Fig. 9.

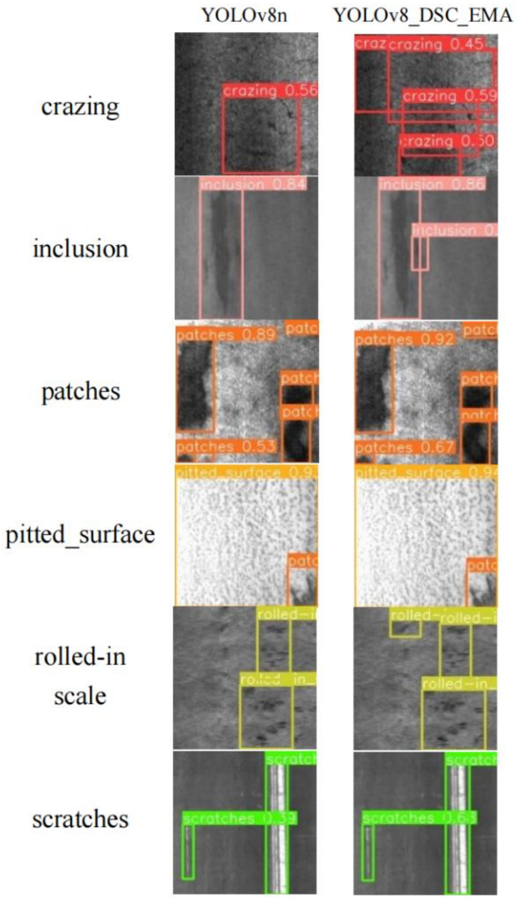

Fig. 9 Comparison chart of various defect detection and testing effects

Through the comparison figure, it can be seen that after the improvement, the average precision of the model is improved obviously, the confidence of the defect recognition in the detection result figure is improved to a certain extent, and the positioning of the recognition box is more accurate, which verifies the effectiveness of the improvement.

5. Conclusion

With the continuous innovation of deep learning theories and methods, a large number of deep learning algorithms have been applied to industrial production in the field of object detection. Based on the data set of steel strip surface defects collected by Professor Song Kechen's team at Northeastern University, this paper designs an improved surface defect detection algorithm based on YOLOv8n model, which has achieved good results in the detection accuracy and real-time performance of steel strip surface defects. Firstly, from the perspective of steel strip production process, combined with the defect types in the data set, the formation cause and visual characteristics of each defect are analyzed, and the YOLO series target detection method is selected. In order to facilitate model training, the labels of the dataset are converted from VOC format to YOLO format, and then the dataset is expanded, the 1800 images of the original dataset are expanded to 5400 images, and the image coordinate mapping formula is calculated and the corresponding labels are modified. Complete the preprocessing of the dataset. YOLOv8n was used as the baseline model for experiments, and the experimental results were analyzed. In order to solve the problem of low accuracy of defect detection such as small target and small edge gray change, DSC is used to replace the deep convolution calculation method of the backbone network and the EMA attention mechanism is introduced to lightweight the model, which can realize that the calculation amount is not greatly improved under the premise of ensuring the accuracy of the model, saving calculation cost and improving calculation speed. The experimental results indicate that the enhanced model achieves an average accuracy of 80.3%, representing a 6.8% improvement over the original model.

Although the improved method proposed in this paper has brought about a considerable improvement in accuracy, there is a definite increase in computational effort. In future research endeavors, attempts at lightweight enhancement can be made to reduce its computational load and enhance the detection speed of the network. Additionally, its accuracy still holds some potential for improvement, and a new detection head could be incorporated in the shallow layer to further augment the detection capability of small target defects.

References

1. Hong-Tao Z, Fa-Jie D, Ke-Qin D. Study on On-Line Surface Defect Detection Vision System for Steel Strip. *Chinese Journal of Sensors and Actuators*, 2007.
2. Redmon J , Divvala S , Girshick R ,et al. You Only Look Once: Unified, Real-Time Object Detection. *Computer Vision & Pattern Recognition.IEEE*, 2016.
3. Bochkovskiy A , Wang C Y , Liao H Y M .YOLOv4: Optimal Speed and Accuracy of Object Detection. *ArXiv*, 2020.
4. Song K C, Yan Y H. A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects. *Applied Surface Science*, 2013, Volume 285, Part B, 858-864.

5. Qi Y, He Y, Qi X, et al. Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation. *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2023: 6070-6079.
6. Ouyang D , He S , Zhan J ,et al. Efficient Multi-Scale Attention Module with Cross-Spatial Learning. *ArXiv*, 2023

Authors Introduction

Mr. Haozhe Sun

He received his B.S. degree from College of Electronic Information and Automation, Tianjin University of Science and Technoogy, China in 2023. He is currently a Master course student in Tianjin University of Science and Technology. His research area is about deep learning and image processing.

Dr. Fengzhi Dai

He received M.E. and Doctor of Engineering (PhD) from the Beijing Institute of Technology, China in 1998 and Oita University, Japan in 2004 respectively. His main research interests are artificial intelligence, pattern recognition and robotics. He worked in National Institute of Technology, Matsue College, Japan from 2003 to 2009. Since October 2009, he has been the staff in College of Electronic Information and Automation, Tianjin University of Science and Technology, China.

Mr. Junjin Chen

He received his bachelor's degree in Mechanical Design, Manufacturing and Automation from Beijing Union University in 2014. His research interests are industrial automation and robotics. From 2005 to 2007, he worked at Beijing Aeronautical Manufacturing Engineering Research Institute. Since September 2007, he has been working at SMC (Beijing) Manufacturing Co., LTD., China.