International Journal of Science and Engineering Applications
Volume 13-Issue 11, 11 – 16, 2024, ISSN:- 2319 - 7560
DOI: 10.7753/IJSEA1311.1003

Improved RT-DETR Approach for Steel Surface Defect Identification

Hao Luo
School of Electric Information and Electrical Engineering Yangtze University Jingzhou, China

Yimeng Xia
School of Electric Information and Electrical Engineering Yangtze University Jingzhou, China

Abstract: To improve the accuracy of surface defect detection on steel while maintaining detection speed, this study proposes an enhanced RT-DETR detection model called FTD-DETR. First, images were obtained from a publicly available steel surface defect dataset, and data were partitioned and augmented, resulting in a steel surface defect dataset containing 2,000 images. The ResNet18 model, known for its low computational complexity and high detection accuracy, was chosen as the backbone feature extraction network. Then, a Faster-EMA module was introduced to update the basic blocks in ResNet18, enhancing the feature extraction speed of the model and improving inter-layer feature interaction. Finally, the AIFI module of RT-DETR was replaced with a Transformer with Deformable Attention encoder structure. This multi-head self-attention mechanism combined with dynamic attention further increases feature representation while reducing computational complexity. Experimental results show that FTD-DETR achieves a precision of 83.6%, recall of 67.7%, and mean average precision (mAP) of 79.3%. Compared to the baseline model RT-DETR, FTD-DETR significantly reduces parameters, floating-point operations, and memory usage while maintaining high accuracy. It features low complexity, high accuracy, and fast detection speed, providing technical support for steel surface defect detection.

Keywords: Steel Surface Crack Detection ; Vision Transformer ; EMA ; yolo ; RT-DETR

1. INTRODUCTION

Steel, as a widely used material in fields such as mechanical manufacturing, construction engineering, and transportation, is highly valued for its high strength, durability, and broad range of applications. Therefore, surface defect detection in steel is an important quality control task [1]. China, as one of the largest steel producers in the world, manufactures a wide variety of steel products, which are extensively used in mechanical manufacturing, construction engineering, and transportation. Steel is highly regarded for its strength, durability, and versatility. However, surface defects in steel are diverse and vary depending on different production processes and environments. During steel processing, surface defect detection is a key step in ensuring product quality, directly impacting customer choice and the competitiveness of products in the market. In various production environments, steel surfaces are prone to defects such as scratches and dents, which can affect their performance [2].However, especially in large-scale production, manual inspection often fails to identify all defects in a timely manner. Additionally, due to a lack of effective detection knowledge, some problems go unnoticed [3] . Thus, there is a need for a method that can automatically detect steel surface defects and provide early warnings. Traditional methods for detecting steel surface defects mainly rely on manual inspection, but due to the complexity of the production environment and the variability in lighting and surface conditions, manual detection is time-consuming, labor-intensive, and prone to errors, making it difficult to meet the needs of automated production lines. Subsequent optical detection technologies [4], such as X-rays, infrared imaging, and laser scanning, have improved detection speed and reduced human errors. However, these devices are expensive, have high operating costs, and are limited in their ability to handle complex defects, restricting their widespread application.

Currently, automated methods for steel surface defect detection [5] can be broadly categorized into three types: traditional image processing-based detection methods, deep learning-based detection methods [6] , and the use of 3D point cloud scanning. The former [7] relies on steps such as data augmentation, image preprocessing, edge detection, and feature extraction. These methods depend on well-defined rules or algorithms (e.g., grayscale thresholding, Canny edge detection) to identify and classify steel surface defects. While they are effective for detecting simple defects, their accuracy in identifying more complex surface textures or varied defect shapes is limited, and their generalization ability is poor.The latter approach utilizes 3D scanning technologies [8] (such as LiDAR and photogrammetry) combined with machine learning to further enhance the accuracy of surface defect detection. 3D point clouds can accurately capture the geometric features of the surface, making them particularly effective for detecting small defects, such as cracks or surface irregularities. However, these methods involve high computational complexity, which makes it difficult to meet the real-time requirements of automated systems.

With the development of deep learning, Anvar A et al. [9] proposed in 2020 an improved convolutional neural network (CNN) architecture called ShuffleDefectNet. This network combines the lightweight design of ShuffleNet with specific layer structures suitable for defect detection tasks. By using data augmentation and transfer learning techniques, the detection performance for different types of metal surface defects, such as cracks, scratches, and pits, was improved. Hu B et al. [10] proposed an enhanced version of the classic Faster R-CNN detection algorithm, integrating FPN (Feature Pyramid Network) technology to enhance multi-scale feature extraction capabilities, enabling the model to more accurately identify PCB defects of varying sizes and types, such as breaks, short circuits, and missing components. The improved Faster R-CNN with FPN demonstrated significant improvements in detection precision and recall compared to the original model.Xiao L et al. [11] proposed an improved Mask R-CNN model called IPCNN. This model first utilizes a deep residual neural network to extract features from image pyramids, generating multi-level pyramid features. These features are processed by the Region Proposal Network (RPN) to generate defect bounding boxes and classifications. Finally, within the generated defect bounding boxes, a fully convolutional network (FCN) generates corresponding defect masks. Xia B et al. [12] introduced the SSIM-NET model, which combines SSIM (used to measure image similarity) with the lightweight convolutional neural

network MobileNet-V3. SSIM is first used to compare the input image with a template image, preliminarily locating potential defect areas. Then, MobileNet-V3 acts as a feature extractor, further classifying and detecting the located areas, improving overall efficiency.Yang L et al. [13] aimed to improve model performance by modifying YOLOv5, adopting the lightweight MobileNetV2 as the backbone network and introducing the CBAM attention module to optimize detection accuracy. The improved YOLOv5 not only reduces model parameters and computation but also significantly increases inference speed, improving detection efficiency while maintaining high accuracy. Wang Y et al. [14] optimized YOLO-V7 by incorporating a de-weighted BiFPN structure to enhance feature fusion, thereby reducing information loss during the convolution process and improving detection accuracy. Additionally, the ECA attention mechanism was introduced in the backbone network to strengthen important feature channel representation. The original bounding box loss function was replaced with the SIoU loss function, redefining the penalty term to account for the angle between required regression vectors. The optimized YOLO-V7 significantly increased detection efficiency and accuracy.Song X et al. [15] proposed a multi-directional optimization model based on YOLOv8. This model enhances the feature learning capability of the CSP Bottleneck and C2F modules by introducing deformable convolutions (DCN). It adopts a bidirectional feature pyramid network (BiFPN) for feature fusion and adds the BiFormer attention mechanism to adaptively allocate attention, effectively identifying potential defects. Additionally, the loss function was adjusted to Wise-IoUv3 (WIoUv3) to address overfitting issues with low-quality bounding boxes. With the application of transformers in object detection, Tang B et al. [16] proposed a steel surface defect detection method based on the Swin Transformer architecture. This research aimed to develop an efficient end-to-end model that leverages the hierarchical representation capability of Swin Transformer to improve feature extraction and fusion, thus enhancing defect detection accuracy. Experimental results demonstrated the method's excellent performance in identifying various steel surface defects, highlighting its potential in industrial quality inspection applications.Zhang L et al. [17] recognized the advantages of DETR (Detection Transformer) in the field of image object detection and optimized the DETR model for feature extraction and detection performance, improving the recognition accuracy of casting defects. The improved framework excelled in handling complex defect morphologies, particularly in detecting irregularly shaped defects on casting surfaces.

In conclusion, this paper designs a steel surface crack detection model, FTD-DETR, based on RT-DETR [18] . To address the issue of a small dataset, data augmentation is applied to expand the dataset. After comparing different feature extraction networks, ResNet18 [19] was selected as the baseline backbone network. Faster-EMA is utilized to adjust the basic blocks, further improving feature extraction speed while enhancing feature interaction. To resolve issues with feature detail information at mid and lower levels, the AIFI (Anchor-Free Instance-aware Feature Interaction) module is replaced with Transformer-DAttention, which enhances RT-DETR's global perception capability, multi-scale feature processing, and overall performance. Finally, experimental results confirm that the FTD-DETR model can effectively handle steel surface defect detection tasks.

2. Method

2.1 RT-DETR Model

RT-DETR is an end-to-end object detection model designed for real-time applications, based on the Transformer framework. It is specifically optimized for handling multi-scale features. By decoupling interactions between features at the same scale and integrating cross-scale features, RT-DETR significantly reduces the computational complexity of the original DETR model. It retains efficient multi-scale information extraction capabilities while surpassing many similar models, such as the YOLO series, in both inference speed and detection accuracy. The model simplifies traditional post-processing workflows, ensuring zero-latency inference and stable output.The core of RT-DETR consists of a backbone network, a hybrid encoder, and a decoder with auxiliary prediction heads. The feature extraction component is based on the selected backbone network architecture, utilizing features from the last three stages as inputs for the encoder. The hybrid encoder contains the AIFI (Anchor-Free Instance-aware Feature Interaction) and CCFM (Cross-scale Context Fusion Module) modules: AIFI focuses on encoding the highest-level features (S5), while the CCFM module integrates multi-scale features through both bottom-up and top-down feature fusion paths, producing rich image representations.In the decoding phase, RT-DETR introduces an IoU-aware query module, which selects key image features from the encoder's output as initial object queries and iteratively optimizes them to generate precise bounding boxes and confidence scores. As shown in the network architecture diagram (Figure 1), this design greatly enhances detection efficiency and accuracy, especially in applications requiring high real-time performance. These improvements make RT-DETR a high-performance model, significantly reducing computational burdens while maintaining precision, making it well-suited for various real-time object detection tasks.

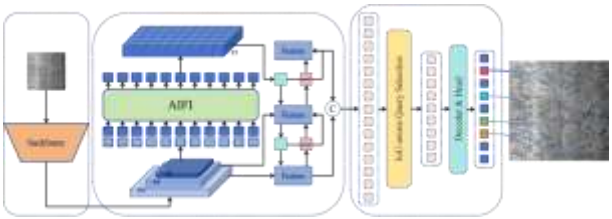

Fig1. The structure of RT-DETR model.

2.2 Improved Model Design for RT-DETR

Although RT-DETR is a high-performance model that significantly reduces computational load while maintaining accuracy, making it suitable for various real-time object detection tasks, the choice of backbone network directly affects feature extraction and computational complexity. In this paper, a series of lightweight networks such as ResNet18, Mobilenetv3 [20] , Fasternet [21] , Efficientnet [22] , HGNetV2, and VanillaNet13 [23] were selected for experiments, as shown in Table 1. These experiments comprehensively evaluated the model's parameter count, computational complexity, and accuracy in detection tasks. The results, displayed in Table 1, indicate that ResNet18 delivers the most balanced performance.

International Journal of Science and Engineering Applications
Volume 13-Issue 11, 11 – 16, 2024, ISSN:- 2319 - 7560
DOI: 10.7753/IJSEA1311.1003

Table 1. Comparison of training results of different

Backbone	Parameters/Mb	FLOPs/G	mAP/%
Mobilenetv3	37.20	24.7	60.3
Fasternet	41.25	28.5	69.4
Efficientnet	56.77	33.2	70.3
Resnet-18	75.85	57.0	74.5
VanillaNet	105.73	165.9	73.2
HGNetV2	125.16	108.0	75.5

2.2.1 Resnet-18

ResNet-18 (Residual Network 18) is a classic convolutional neural network (CNN) architecture composed of 18 convolutional layers. Its key innovation lies in the introduction of "residual blocks" (as shown in Figure 2). These blocks use skip connections to directly pass input information to later layers, addressing the common vanishing gradient problem in deep networks. This allows deeper networks to be trained effectively. The design of ResNet-18 enables the network to learn more efficient feature representations while avoiding performance degradation caused by increased network depth.Specifically, ResNet-18 consists of 5 convolutional stages, with each stage containing multiple residual blocks. These blocks perform convolution operations using 3×3 kernels, and after applying the activation function, the output from the previous layer is added. Compared to deeper ResNet versions, such as ResNet-50 or ResNet-101, ResNet-18 has fewer parameters, making it more computationally efficient while maintaining high accuracy. This makes it particularly suitable for resource-constrained applications.

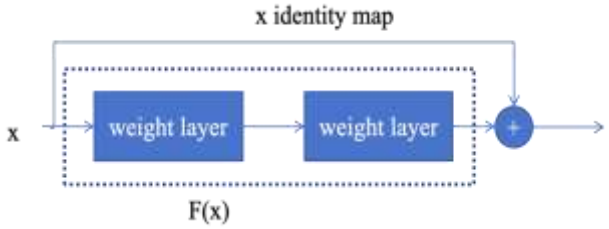

Fig2. Residual learning mechanism

2.2.2 EMA

In steel surface defect detection, the presence of occlusions or complex textures can affect detection accuracy, leading to false positives and missed detections. To reduce interference from irrelevant features and enhance the model's feature extraction capabilities, we introduced the EMA attention mechanism into the model. This mechanism retains information from each channel while reducing computational costs, allowing the model to focus more on the target defect areas, thereby improving detection performance. The structure of the EMA attention module is shown in Figure 3.

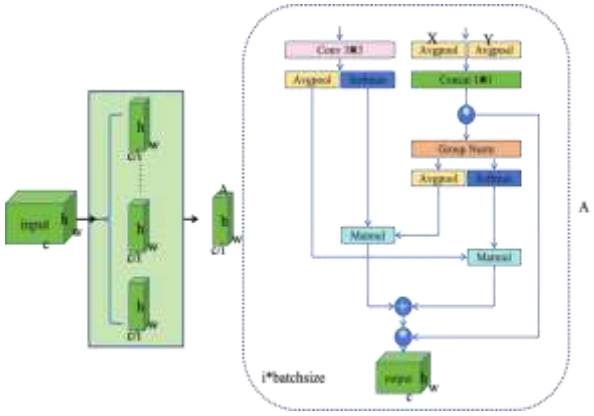

Fig3. The structure of EMA

In the steel surface defect detection task, the input features are divided into i sub-features along the channel dimension, and the attention weights learned by the model enhance the feature representation related to defect areas within each sub-feature. To capture cross-channel dependencies and reduce computational load, cross-channel information interaction is modeled along the channel direction, processed through three parallel paths: horizontal global average pooling, vertical global pooling, and convolution.The features from the first two paths are connected along the height of the image and share a 1×1 convolution, with the output feature vectors activated by a Sigmoid function. The third path captures local cross-channel interactions using a 3×3 convolution to expand the feature space.In the cross-spatial learning part, 2D global average pooling is used to encode global spatial information for both the 1×1 and 3×3 branches, and the output is processed using a Softmax function. The smallest branch output is reshaped to the corresponding dimensions. The two generated spatial attention weights are aggregated through a cross-spatial interaction module, establishing long-range dependencies and capturing pixel-level pairwise relationships of steel surface defects. This highlights the global context information of all pixels.After fusing multi-scale information, the output feature map is activated by a Sigmoid function, enhancing the model's focus on defect regions on the steel surface. This results in richer feature aggregation and improved accuracy in defect detection.

2.2.3 FasterNet Block

In response to the issue of slow inference speed on edge devices due to the large number of model parameters in current steel surface defect detection tasks, this paper introduces improvements to the ResNet-18 module. Specifically, the FasterBlock-EMA module from the FasterNet network is used to replace the BasicBlock module in ResNet-18. This modification effectively reduces both computational complexity and the number of parameters in the detection task, significantly improving detection speed. With its lightweight design, the model is particularly well-suited for efficient inference on resource-constrained edge devices.

The FasterNet Block is the core component of FasterNet, and its design is inspired by GhostNet, addressing the redundancy issue in feature convolution channels. However, unlike GhostNet, FasterNet does not use DWConv (Depthwise Separable Convolution); instead, it introduces a new Partial Convolution (PConv), as shown in Figure 4. PConv applies regular convolution only to a portion of the input channels to extract spatial features, while the remaining channels remain unchanged. This approach effectively reduces computational load and memory usage, significantly improving computational efficiency without notably sacrificing feature representation

International Journal of Science and Engineering Applications
Volume 13-Issue 11, 11 – 16, 2024, ISSN:- 2319 - 7560
DOI: 10.7753/IJSEA1311.1003

capability.For continuous feature access, the first or last channel is treated as a representative of the entire feature map for computation, which further reduces computational complexity.

FasterNet is an efficient neural network designed specifically for object detection tasks, optimized for both speed and accuracy. Its core concept is to enhance feature representation capability and expand the receptive field coverage, all while maintaining a lightweight architecture and high inference speed. The FasterNet network structure consists of four stages, as shown in Figure 4. Each stage is responsible for extracting features at different scales, with the primary differences being the size of the convolution kernels.The Embedding module performs the initial feature extraction using regular convolutions with a stride of 4. The Merging layer uses convolutions with a stride of 2 for spatial downsampling and channel expansion, progressively reducing the spatial resolution of the feature maps while increasing the number of channels. This stepwise compression of spatial resolution and expansion of channel dimensions is crucial for efficient detection at multiple scales.

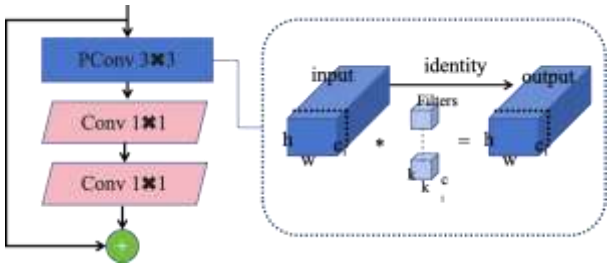

Fig4. The structure of Fasternet

2.3 Transformer_DAattention

In the RT-DETR network, the newly introduced AIFI module (Anchor-Free Instance-aware Feature Interaction) offers several advantages, particularly its efficient local feature extraction capability. By implementing object detection in an anchor-free manner, it eliminates the complex anchor design found in traditional detectors, simplifying the overall model architecture. Moreover, AIFI enhances object instance perception through its feature interaction mechanism, making it highly effective at handling objects of varying shapes and scales, achieving good detection speed and accuracy.However, AIFI also has some limitations. Since its focus is primarily on local feature extraction, it is less effective at capturing long-range global dependencies, which could limit the model's global awareness in complex scenes. Additionally, the feature interaction mechanism in AIFI is somewhat inadequate in handling multi-scale features, which may result in underperformance when dealing with tasks that involve large variations in object size. In contrast, models equipped with multi-head self-attention mechanisms tend to perform better in such scenarios.Overall, while the AIFI module excels in certain contexts, it has room for improvement in terms of global information capture and multi-scale processing.

To address the challenges of automated detection for steel surfaces in complex environments, this paper replaces the AIFI module with Transformer with Deformable Attention. The Transformer Encoder, utilizing the self-attention mechanism, is more effective in capturing global contextual information and can manage long-range feature dependencies. This improvement helps the model perform better in detecting complex objects and backgrounds.Additionally, Deformable Attention (DAttention) within the Transformer Encoder dynamically adjusts the weights between different tokens based on input features or context, making the attention mechanism more flexible. The inclusion of residual connections and feedforward neural networks further prevents gradient vanishing and information loss, enhancing both the stability and accuracy of the model. Moreover, the adaptive nature of the attention mechanism allows it to dynamically allocate attention weights according to task requirements, boosting performance in complex scenes.Thus, replacing AIFI with Transformer-DAttention significantly improves RT-DETR's global perception capabilities, multi-scale feature processing, and overall performance. The revised network architecture is shown in Figure 5.

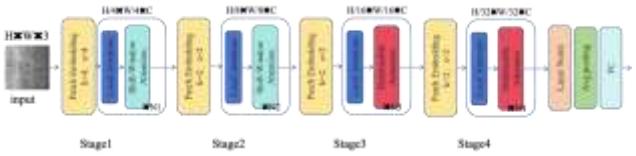

Fig5. The structure of Transformer-DAttention

In the proposed model, Patch Embedding consists of a Layer Norm and a convolution layer, which functions similarly to the token transformation process in Vision Transformers. This embedding process ensures the input features are appropriately transformed for subsequent stages.In stage 1 and stage 2, the design includes the Swin Transformer's paired W-MSA (Window-based Multi-Head Self-Attention) and SW-MSA (Shifted Window Multi-Head Self-Attention) mechanisms. These components help enhance the model's efficiency in capturing local and global dependencies.In stage 3 and stage 4, a combination of W-MSA and MDHA (Multi-Head Deformable Attention) modules is used. The MDHA is the core Deformable Attention Module, which allows the model to dynamically focus on the most relevant parts of the input while handling varying object scales and deformations effectively. This structure is illustrated in Figure 6.

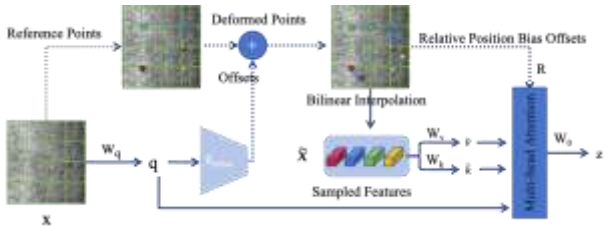

Fig6. Deformable attention module

The Deformable Attention module enhances the flexibility and accuracy of feature extraction by dynamically adjusting the sampling positions. First, it generates sampling offsets based on the input feature map, allowing the sampling positions to adapt to different shapes and scales across various regions. Then, the module performs non-uniform sampling at these dynamically adjusted locations and applies learned attention weights to aggregate the sampled features, ensuring the capture of important contextual information. After the weighted aggregation, the output features become more expressive, effectively addressing the challenges of detecting complex and irregular objects.

3. EXPERIMENTS

3.1 Data Set

The dataset used in this study is based on the surface defect database released by Northeastern University (NEU), which includes six typical defect types on steel surfaces: rolling scale (Rs), blister (Pa), crack (Cr), pitting (Ps), inclusions (In), and scratch (Sc). This database contains a total of 1,800 grayscale images, with 300 samples for each defect type. To enhance the

International Journal of Science and Engineering Applications
Volume 13-Issue 11, 11 – 16, 2024, ISSN:- 2319 - 7560
DOI: 10.7753/IJSEA1311.1003

robustness of the model and prevent overfitting, data augmentation techniques were applied to the original dataset, as illustrated in Figure 7. These techniques include rotation, flipping, contrast adjustment, and noise addition. The augmented dataset comprises a total of 2,000 images, which were split into training, testing, and validation sets in a ratio of 7:2:1. The six types of surface defects were annotated using the LabelImg tool, with each image potentially containing multiple defects.

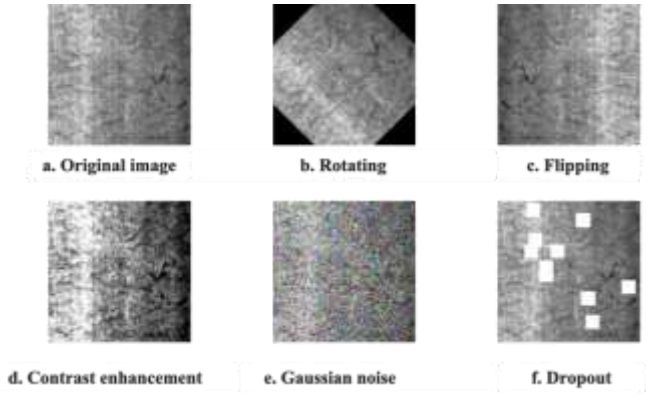

Fig7. Steel surface defect data set enhancement results

3.2 Experimental Environment and Parameter Settings

The experiments in this paper were conducted using the PyTorch 1.13.1 deep learning framework, with the operating system being Windows 11. The hardware environment includes an Intel i7-12700 processor, 80 GB of RAM, and an NVIDIA GeForce RTX 3070 graphics card with 8 GB of VRAM. To accelerate the training process, CUDA 11.6 was used for GPU acceleration.

The specific settings for the experimental parameters are as follows: after preprocessing, the images processed by the model are consistently sized at 640×640. The batch size is set to 16, and the number of iterations is 300. The optimizer chosen is AdamW, with an initial learning rate set at 0.0001 and a weight decay coefficient also set at 0.0001.

3.3 Experimental Results

3.3.1 Comparison of Detection Results for Different Defects Experiments

The detection results of the FTD-DETR model for six typical surface defects in steel are shown in Table 2. The results indicate that the detection performance for inclusion is the best, with precision, recall, and mAP reaching 87.7%, 83.9%, and 89.6%, respectively. In contrast, the detection accuracy for crazing is the lowest, with precision, recall, and mean average precision (mAP) at 58.8%, 40.4%, and 42.7%, respectively. The difference in performance can be attributed to the fact that inclusion exhibits more pronounced and consistent features among steel surface defects, while the morphology of crazing is complex and variable, making it easily confused with other defects, resulting in lower detection performance. On the other hand, the dataset for inclusion is significantly more abundant than that for crazing, allowing the model to learn more effectively and improve its generalization capabilities.

Table 2. Comparison of detection results of different defects by FTD-DETR model

Defect name	P/%	R/%	mAP/%
Crazing	58.8	40.4	42.7
Inclusion	87.7	83.9	89.6
Patches	83.6	76.7	79.6
Pitted Surface	89.1	70.7	77.3
Rolled-in Scale	75.5	56.9	66.9
Scratches	80.6	77.7	84.6

3.3.2 The comparison results of different models.

To compare and validate the performance of the FTD-DETR network model in detecting steel surface defects, four different models, including the original RT-DETR, Yolov5, Yolov7, and Yolov8, were selected under the same experimental conditions. The detailed comparison results can be found in Table 3.

Table 3. Comparison results of different models

Model	P/%	R/%	mAP/%	FLOPs/G
Yolov5	69.5	71.4	73.4	16.5
Yolov7	75.9	66.8	71.6	105.2
Yolov8	74.1	66.1	73.1	28.7
RT-DETR-r18	74.5	64.0	74.5	57.0
FTD-DETR	83.6	67.7	79.3	51.7

As shown in Table 3, FTD-DETR demonstrates excellent performance in steel surface defect detection, with precision, recall, and mAP values of 83.6%, 67.7%, and 79.3%, respectively. Compared to other detection models, FTD-DETR shows an improvement in mAP by 5.9, 7.7, 6.2, and 4.8 percentage points, indicating its higher accuracy and reliability in detecting steel surface defects. Additionally, FTD-DETR has relatively low memory usage, showcasing better computational efficiency, which reflects the model's efficient resource utilization. Overall, FTD-DETR delivers balanced and outstanding performance in terms of detection accuracy and resource efficiency, making it particularly suitable for steel surface defect detection tasks.

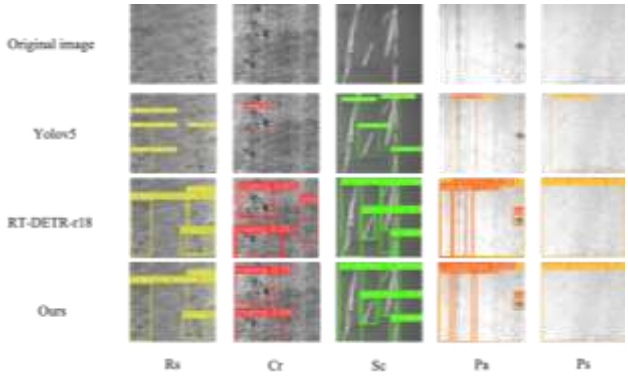

Fig8. Comparison results of different models

Figure 8 visually compares the results of RT-DETR, Yolov5, and RIC-DETR on the test dataset. Different types of defects are marked with rectangular boxes in distinct colors, and the corresponding confidence scores are labeled for each detection.

4. CONCLUSION

In summary, the proposed improved model, FTD-DETR, demonstrated excellent performance in the task of steel surface defect detection. By selecting ResNet18 as the backbone for feature extraction and integrating the Faster-EMA module to replace ResNet18's basic block, feature extraction efficiency was enhanced. Additionally, the standard Transformer encoder and dynamic attention mechanism were employed to replace the original modules. FTD-DETR performed exceptionally well in terms of precision, recall, and mean average precision (mAP). Experimental results show that this model not only maintains high accuracy but also significantly reduces the number of parameters, computational complexity, and memory usage. With its low complexity and fast detection speed, FTD-DETR is suitable for efficient steel surface defect detection in real-world scenarios, providing a reliable technical solution for industrial applications.

Future work could focus on expanding the dataset size and enriching the variety of defect types to improve the model's generalization ability and adaptability to more complex real-world applications. Additionally, further optimization of the multi-scale feature fusion mechanism could enhance the detection of defects of various sizes and shapes, particularly small defects, thereby improving overall detection accuracy.

5. REFERENCES

[1] Ban H Y, Shi G, Shi Y J, et al. Research progress on the mechanical property of high strength structural steels[J]. Advanced materials research, 2011, 250: 640-648.

[2] Schneller W, Leitner M, Pomberger S, et al. Fatigue strength assessment of additively manufactured metallic structures considering bulk and surface layer characteristics[J]. Additive Manufacturing, 2021, 40: 101930.

[3] Campbell L E, Connor R J, Whitehead J M, et al. Human factors affecting visual inspection of fatigue cracking in steel bridges[J]. Structure and Infrastructure Engineering, 2021, 17(11): 1447-1458.

[4] Mordia R, Verma A K. Visual techniques for defects detection in steel products: A comparative study[J]. Engineering Failure Analysis, 2022, 134: 106047.

[5] Sun X, Gu J, Tang S, et al. Research progress of visual inspection technology of steel products—a review[J]. Applied Sciences, 2018, 8(11): 2195.

[6] Tang B, Chen L, Sun W, et al. Review of surface defect detection of steel products based on machine vision[J]. IET Image Processing, 2023, 17(2): 303-322.

[7] Jing L, Tingting D, Dan S, et al. A review on surface defect detection[J]. Journal of Frontiers of Computer Science & Technology, 2014, 8(9): 1041.

[8] Mariniuc A M, Cojocaru D, Abagiu M M. Building Surface Defect Detection Using Machine Learning and 3D Scanning Techniques in the Construction Domain[J]. Buildings, 2024, 14(3): 669.

[9] Anvar A, Cho Y I. Automatic metallic surface defect detection using shuffledefectnet[J]. Journal of The Korea Society of Computer and Information, 2020, 25(3): 19-26.

[10] Hu B, Wang J. Detection of PCB surface defects with improved faster-RCNN and feature pyramid network[J]. Ieee Access, 2020, 8: 108335-108345.

[11] Xiao L, Wu B, Hu Y. Surface defect detection using image pyramid[J]. IEEE Sensors Journal, 2020, 20(13): 7181-7188.

[12] Xia B, Cao J, Wang C. SSIM-NET: Real-time PCB defect detection based on SSIM and MobileNet-V3[C]//2019 2nd World conference on mechanical engineering and intelligent manufacturing (WCMEIM). IEEE, 2019: 756-759.

[13] Yang L, Huang X, Ren Y, et al. Steel plate surface defect detection based on dataset enhancement and lightweight convolution neural network[J]. Machines, 2022, 10(7): 523.（yolov5）

[14] Wang Y, Wang H, Xin Z. Efficient detection model of steel strip surface defects based on YOLO-V7[J]. Ieee Access, 2022, 10: 133936-133944.

[15] Song X, Cao S, Zhang J, et al. Steel Surface Defect Detection Algorithm Based on YOLOv8[J]. Electronics, 2024, 13(5): 988.

[16] Tang B, Song Z K, Sun W, et al. An end-to-end steel surface defect detection approach via Swin transformer[J]. IET Image Processing, 2023, 17(5): 1334-1345.

[17] Zhang L, Yan S, Hong J, et al. An improved defect recognition framework for casting based on DETR algorithm[J]. Journal of Iron and Steel Research International, 2023, 30(5): 949-959.

[18] Zhao Y, Lv W, Xu S, et al. Detrs beat yolos on real-time object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 16965-16974.

[19] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[20] Howard A G. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[21] Chen J, Kao S, He H, et al. Run, don't walk: chasing higher FLOPS for faster neural networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 12021-12031.

[22] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]//International conference on machine learning. PMLR, 2019: 6105-6114.

[23] Chen H, Wang Y, Guo J, et al. Vanillanet: the power of minimalism in deep learning[J]. Advances in Neural Information Processing Systems, 2024, 36.