Applied Computer Science 2026, vol. 22, no. 2, pp. 193–207
https://doi.org/10.35784/acs_9140

Submitted: 2026-02-03 | Revised: 2026-03-23 | Accepted: 2026-04-10

Keywords: rail surface defect segmentation, semantic segmentation, direction-sensitive feature fusion, multi-scale context aggregation, spatial-channel attention

Qike WU [1,2], *Sharafiz ABDUL RAHIM* [2*], *Sai Hong TANG* [2], *Muhammad Azim AZIZI* [2], *Li ZHANG* [2]

[1] Hainan Tropical Ocean University, China, gs68576@student.upm.edu.my

[2] Universiti Putra Malaysia, Malaysia, saihong@upm.edu.my, muhdazim@upm.edu.my, gs65663@student.upm.edu.my

[*] Corresponding author: sharafiz@upm.edu.my

SFAB-Net: Semantic segmentation network for railway track surface defects based on Spatial Fusion and Adaptive Bottleneck feature enhancement

Abstract

Under the long-term action of train loads and complex environmental conditions, the surfaces of railway tracks are prone to defects such as cracks, spalling, and pitting, which seriously threaten the safety of railway operations. Semantic segmentation can achieve pixel-level positioning and morphological characterization of defects. However, existing methods still struggle to model strongly directional structures and multi-scale defects while maintaining a balance between accuracy and efficiency in rail-surface inspection. To address the above issues, this paper proposes a lightweight semantic segmentation network for railway track surface defects (SFAB-Net) based on spatial fusion and adaptive bottleneck feature enhancement. This network effectively characterizes the features of slender cracks along the rail direction using the direction-sensitive Spatial-Fusion module and combines them with the simplified spatial pyramid pooling module to achieve multi-scale context aggregation. In the decoding stage, an adaptive feature reconstruction mechanism and spatial-channel joint attention are introduced to enhance multi-scale feature fusion and suppress background interference. Experimental results on the NEU-DET dataset and a self-built rail surface image dataset show that SFAB-Net outperforms several representative methods in segmentation accuracy and robustness, and has strong potential for engineering applications.

1. INTRODUCTION

Railways serve as crucial infrastructure for large-scale passenger and freight transport. The track structure is subjected to sustained train loads and complex environmental conditions, inevitably leading to surface defects such as spalling, cracking, and pitting corrosion. If these defects are not promptly detected and addressed, they pose significant risks to train operation safety and equipment longevity. Recent review papers indicate a shift in the railway industry from manual inspection and traditional non-destructive testing to automated detection systems that leverage machine vision and deep learning. Rail-surface defect detection is regarded as a pivotal component of such systems (Kumar & Harsha, 2025; Pappaterra et al., 2024). With the increasing use of on-board, wayside, and laboratory detection equipment, achieving high-precision, stable, and deployable defect identification in extensive track image datasets has become a shared concern in both academia and industry.

1.1. Existing techniques for rail surface defect segmentation

In surface defect analysis, deep learning methods can generally be divided into three categories: image-level classification, object detection, and semantic segmentation. Image-level classification only determines whether a sample contains defects, but it cannot provide precise information about defect location or morphology. Object detection predicts defects by bounding boxes, yet this representation is often insufficient for rail surface defects with slender structures, irregular contours, or small adjacent regions, because rectangular boxes cannot accurately describe their geometric boundaries (Demir et al., 2023; Frydrych et al., 2025; Li et al., 2025). By contrast, semantic segmentation assigns a category to each pixel and can therefore

provide defect distribution, boundary morphology, and area information simultaneously. For this reason, semantic segmentation has attracted increasing attention in industrial inspection tasks involving steel, metal sheets, and similar materials. Several recent studies have reported high-precision segmentation networks for steel defects, PCB defects, and metal-surface abnormalities, showing encouraging performance for fine-grained industrial deployment (Chen & Min, 2025; Guclu & Akin, 2025; Guo et al., 2025; Zhang et al., 2025).

In railway track surface inspection, however, existing studies have still focused mainly on detection-based frameworks. Zheng et al. (2021) proposed a deep convolutional neural network for multi-object detection of rail-surface and fastener defects, thereby improving both detection accuracy and real-time performance. Ming et al. (2023) combined a 3D line-scan camera with a deep network to detect rail-surface spalling and pitting defects, thereby reducing false alarms caused by illumination variation. Mao et al. (2024) improved CenterNet by optimizing the backbone and multi-branch architecture and proposed a one-stage model for rail-surface defect detection, achieving favorable localization performance in complex backgrounds. In addition, frameworks such as Faster R-CNN and Mask R-CNN have also been introduced into rail-surface and fastener inspection to support joint recognition of multiple components (Yilmazer & Karakose, 2024). Nevertheless, most of these methods remain limited to box-level or instance-level prediction. They often fail to provide the precise boundaries and topological details required for crack tips, tiny pitting corrosion, and multiple closely spaced defect regions, which restrict subsequent size measurements and failure analysis.

To address these limitations, a small number of semantic segmentation methods have recently been developed for rail surfaces. Pan et al. (2025) designed an encoder-decoder rail surface defect segmentation network and introduced an attention mechanism to improve segmentation performance under complex backgrounds. Si et al. (2024) proposed Rail-STrans, a rail surface defect segmentation method based on an improved Swin Transformer. By using hierarchical window self-attention as the backbone and enhancing feature fusion and contextual modeling, their method improved the stability of pixel-level segmentation for rail defects with slender morphology, unclear boundaries, and large scale differences under complex and low-contrast conditions. In related industrial scenarios such as steel plates and metal surfaces, many studies have incorporated dilated convolution, pyramid pooling, multi-layer feature fusion, and lightweight design into classical segmentation architectures such as U-Net and DeepLab, thereby improving segmentation accuracy and efficiency to varying degrees (Guclu & Akin, 2025; Chen & Min, 2025; Zhang et al., 2025). Although recent railway-domain studies have advanced pixel-level defect segmentation, their design emphases differ from that of the present work. Pan et al. (2025) mainly improved segmentation performance under complex backgrounds through an attention-enhanced encoder-decoder framework, whereas Si et al. (2024) relied on an improved Swin Transformer backbone to strengthen hierarchical contextual modeling for rail defects with unclear boundaries and large-scale variation. In contrast, the contribution of SFAB-Net does not lie in introducing a heavier Transformer architecture or simply enhancing a generic encoder-decoder pipeline. Instead, our method is specifically designed for the structural characteristics and deployment requirements of rail-surface inspection. It explicitly models direction-sensitive crack morphology through the Spatial-Fusion module, strengthens lightweight multi-scale context aggregation, and further improves decoder-side feature reconstruction through adaptive bottleneck feature enhancement. Therefore, the novelty of this work lies in a task-oriented lightweight segmentation design that jointly addresses directional structure modeling, multi-scale defect representation, and efficient feature recovery for practical railway inspection scenarios.

Nevertheless, from the perspective of rail-specific structure modeling and deployment efficiency, three challenges still remain when these methods are applied to rail-surface inspection. First, rail-surface cracks and rolling-contact-fatigue defects often exhibit strong longitudinal directionality and are frequently accompanied by transverse scratches and block-like spalling. Conventional isotropic 3×3 convolutions and general dilated convolutions cannot explicitly model such directional structures, which may lead to crack fracture, adhesion, or incomplete boundary recovery. Second, existing multi-scale modeling strategies mainly rely on generic modules such as feature pyramid networks or atrous spatial pyramid pooling, and thus still struggle to represent both large-area spalling and extremely fine cracks within a unified framework. As a result, their sensitivity to strong scale variation remains limited. Third, practical railway inspection systems require a favorable trade-off between segmentation accuracy and computational cost. Excessive lightweight design may reduce parameter count and inference burden, but it often weakens the representational capacity for small defects and complex morphologies. Accordingly, unlike recent railway-domain segmentation methods that primarily focus on generic attention enhancement or Transformer-based contextual modeling, the present work emphasizes a lightweight, task-oriented design for direction-sensitive rail defect representation and adaptive decoder reconstruction.

1.2. Our method and contributions

In view of the above issues, this study proposes SFAB-Net, a lightweight semantic segmentation network for railway track surface defects based on spatial fusion and adaptive bottleneck feature enhancement. Built upon the U-Net encoder-decoder framework, the proposed method aims to improve the representation of direction-sensitive crack structures, strengthen multi-scale contextual perception, and refine feature reconstruction during decoding, while preserving computational efficiency for practical deployment. Specifically, in the encoder, a direction-sensitive Spatial-Fusion module is introduced to more effectively capture elongated cracks along the rail direction and transverse structural variation. A simplified spatial pyramid pooling module is then employed to aggregate multi-scale contextual information without introducing excessive complexity. In the decoder, an adaptive feature reconstruction mechanism based on CNNAdapter is adopted to reduce the scale inconsistency and redundancy caused by direct skip-feature concatenation. In addition, a spatial-channel squeeze-and-excitation module is used to suppress background interference and enhance defect-related responses. Depthwise separable convolution is applied throughout the network to further reduce model complexity (He et al., 2015; Howard et al., 2017; Roy et al., 2018).

The main contributions of this work are summarized as follows:

1. We propose a direction-sensitive feature encoding strategy for rail defect morphology, which explicitly captures elongated longitudinal cracks and transverse structural variations. Unlike conventional isotropic convolution-based encoding, this design is better suited to the structural characteristics of rail-surface defects.
2. We design an adaptive feature reconstruction mechanism for the decoder stage, which enhances encoder-decoder multi-scale fusion while reducing redundancy and scale inconsistency introduced by direct skip concatenation. This improves the recovery of complex defect boundaries and fragmented regions.
3. We develop a lightweight segmentation network that balances accuracy and efficiency. By combining multi-scale context aggregation with spatial-channel attention, the proposed model enhances critical defect regions, suppresses background interference, and improves engineering deployability under limited computational resources.
4. We validate the proposed method on both the publicly available NEU-DET dataset and a self-built rail surface image dataset. Experimental results demonstrate that SFAB-Net outperforms several representative segmentation methods across multiple evaluation metrics.

2. OVERALL NETWORK ARCHITECTURE

As shown in Figure 1, SFAB-Net is based on the symmetric encoder–decoder architecture of U-Net. First, we preprocess the rail surface images and crop them into fine-grained image slices to serve as network input.

During the encoding stage, the network replaces the traditional downsampling operation with the Spatial-Fusion module. This module models local texture features and horizontal and vertical spatial-structure information in parallel through multiple depthwise convolution branches, thereby effectively extracting direction-sensitive features. Next, the Simplified Spatial Pyramid Pooling Fast (SimSPPF) module (He et al., 2015) is introduced to aggregate multi-scale contextual information without reducing feature resolution, enhancing the network's ability to detect defects at different scales.

In the decoding stage, feature maps first restore spatial resolution via bilinear upsampling, then form skip connections with the corresponding encoder features. To alleviate scale inconsistency between encoding and decoding features, the CNNAdapter module performs multi-scale residual refinement and fusion on the skip-connected features. Finally, the spatial-channel squeeze-and-excitation (scSE) module (Roy et al., 2018) adaptively recalibrates features, suppressing background interference and enhancing the response in the defect region.

Depthwise separable convolutions (Howard et al., 2017) are applied throughout the network to reduce model complexity. Finally, the output head generates the pixel-level segmentation results for railway track surface defects.

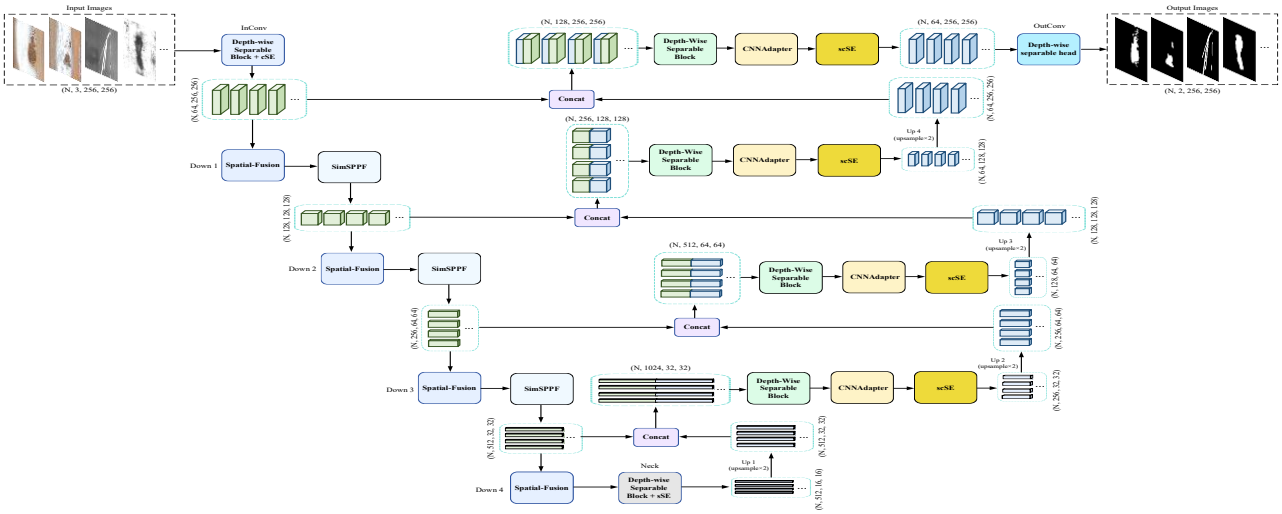

Fig. 1. Overall Architecture of the SFAB-Net

3. TECHNICAL DETAILS

3.1. Image preprocessing

In real railway track scenes, noncritical elements such as ballast, nuts, screws, and track sidebands are often present. This study focuses on the semantic segmentation of railway track surface defects. Accordingly, we first isolate and extract images of the track surface. We then apply fine-grained image slicing to divide each surface image into multiple small regions and perform pixel-level annotation on regions that contain defects. The preprocessing workflow is shown in Figure 2.

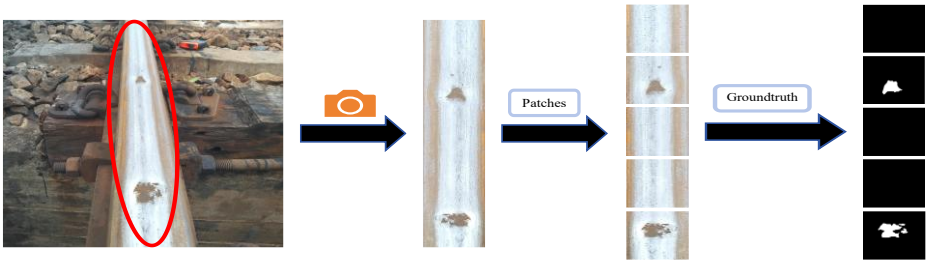

Fig. 2. Image preprocessing process

3.2. Depthwise separable block

To reduce model parameters and computational cost, SFAB-Net uses depthwise separable convolution (DWConv) (Howard et al., 2017) instead of standard convolution as the basic operation in the backbone.

As shown in Figure 3, let the input feature tensor be denoted by $X \in R^{C_{in} \times H \times W}$, where C_{in} denotes the number of input channels, and H and W represent the height and width of the feature map, respectively. A depthwise separable convolution consists of two successive operations: a depthwise convolution followed by a pointwise convolution.

First, a $k \times k$ depthwise convolution is applied independently to each input channel

$$U_c = W_c^{dw} * X_c \, , c = 1,2 \ldots \, , C_{in}. \tag{1}$$

Where: X_c denotes the feature map of the $c - th$ input channel, $*$ denotes the 2D convolution operator, and $W_c^{dw} \in \mathbb{R}^{k \times k}$ is the corresponding depthwise kernel for that channel. Stacking the outputs over all channels yields $U_c \in \mathbb{R}^{C_{in} \times H' \times W'}$。 Where H' and W' are determined by the convolution stride and padding.

Then, a 1×1 pointwise convolution performs a linear combination across channels to produce the channel-mixed output:

$$Y = U * W^{pw} \tag{2}$$

Where: $W^{pw} \in \mathbb{R}^{C_{out} \times C_{in} \times 1 \times 1}$ is the 1×1 pointwise kernel, C_{out} and C_{in} is the number of output and input channels, resulting in $Y \in \mathbb{R}^{C_{out} \times H' \times W'}$.

Ignoring bias terms, the parameter count of a standard $k \times k$ convolution is $P_{std} = C_{in} C_{out} k^2$; whereas that of a depthwise separable convolution is $P_{ds} = C_{in} k^2 + C_{in} C_{out}$; Their ratio is given by:

$$\frac{P_{ds}}{P_{std}} = \frac{C_{in} k^2 + C_{in} C_{out}}{C_{in} C_{out} k^2} = \frac{1}{C_{out}} + \frac{1}{k^2} \tag{3}$$

Where: P_{ds} and P_{std} denote the number of parameters in depthwise separable convolution and standard convolution, respectively. and k is the kernel size.

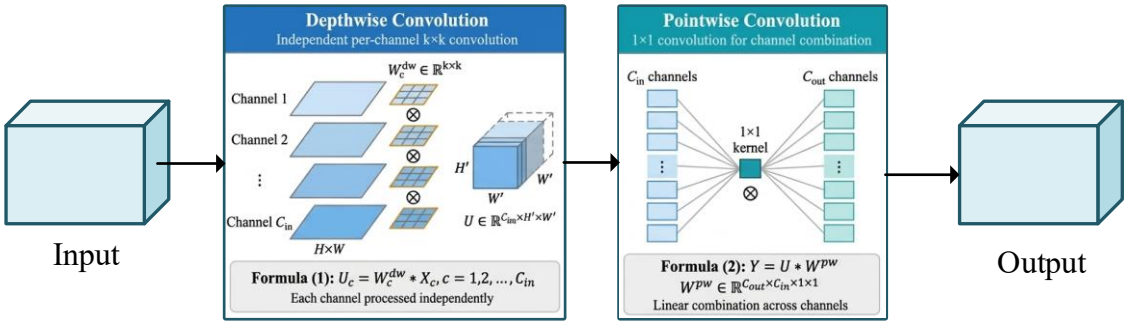

Fig. 3. Depthwise separable convolution layer

3.3. Channel squeeze-and-excitation attention

(1) Encoder stage

To strengthen defect-relevant channel responses while suppressing redundant features, channel squeeze-and-excitation (cSE) attention is introduced at several key layers in this study (Roy et al., 2018).

As shown in Figure 4, given an input feature map $X \in \mathbb{R}^{C \times H \times W}$, the module first applies global average pooling over the spatial dimensions to obtain a channel descriptor, then uses two 1×1 mappings to generate channel-wise weights, and finally recalibrates the input X on a per-channel basis, as follows:

$$\alpha = \sigma\left(W_2 \delta\left(W_1 GAP(X)\right)\right) \tag{4}$$

$$X' = X \odot \alpha \tag{5}$$

Where: $GAP(\cdot)$ denotes global average pooling, which compresses X into a channel statistic in $\mathbb{R}^{C \times 1 \times 1}$; W_1 and W_2 are two 1×1 convolutional mappings that perform channel reduction and expansion, respectively (with a typical squeeze ratio r, *i.e.*, $C \to C/r \to C$); $\delta(\cdot)$ is the ReLU activation; $\sigma(\cdot)$ is the Sigmoid function; $\odot$ denotes element-wise multiplication with channel-wise broadcasting; $\alpha \in \mathbb{R}^{C \times 1 \times 1}$ is the channel weight vector; and X' is the recalibrated output feature map.

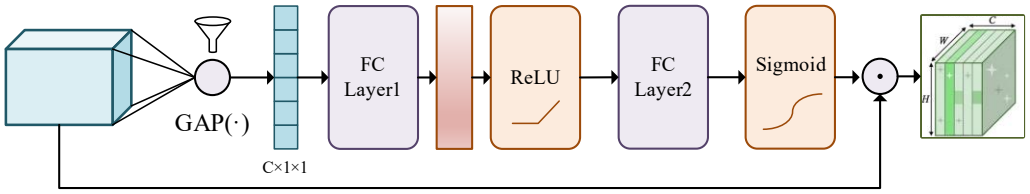

Fig. 4. Channel attention module

(2) Decoder stage

To enhance responses over defect regions while suppressing background interference during decoding, a spatial attention branch is incorporated on top of cSE.

As illustrated in Figure 5, the spatial branch (sSE) extracts spatial cues using depthwise convolution and then applies a 1×1convolution to squeeze features into a single-channel spatial weight map M, which is subsequently used to reweight the input X at each pixel location:

$$M = \sigma(Conv_{1\times1}(DWConv_{3\times3}(X)) \tag{6}$$

$$X_{sSE} = X \odot M \tag{7}$$

Where: $DWConv_{3\times3}(\cdot)$ denotes depthwise convolution, $Conv_{1\times1}(\cdot)$ denotes pointwise convolution, $\odot$ represents element-wise multiplication with channel-wise broadcasting.

The output of the channel branch (cSE), denoted as X_{cSE}, is computed in the same manner as in Eq. (5), i.e., channel weights are generated via global average pooling followed by two 1×1 mappings, and the input X is recalibrated channel-wise.

The scSE fusion (Roy et al., 2018) is implemented by element-wise summation of the two branch outputs:

$$X_{scSE} = X_{sSE} + X_{cSE} \tag{8}$$

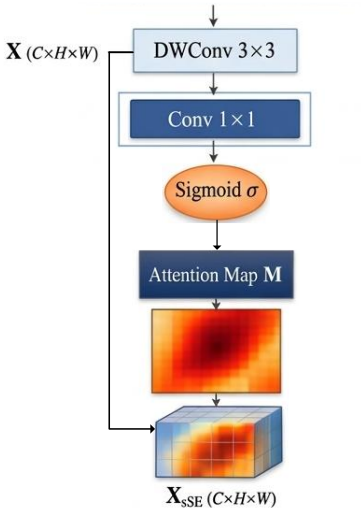

Fig. 5. Channel-spatial attention module

3.4. Spatial-fusion downsampling and direction-sensitive fusion

To simultaneously achieve 2 × downsampling and directional information modeling in the encoder, we employ a Spatial-Fusion module that partitions the input feature map via even-odd subsampling. Three depthwise convolution branches are then introduced to capture local texture cues and horizontal and vertical structural patterns. Finally, a pointwise convolution is used for channel aggregation and compression. As shown in Figure 6, given an input feature map $X \in \mathbb{R}^{C\times H\times W}$, four sub-feature maps are first obtained by even-odd sampling with stride 2:

$$\mathrm{X}_{00} = \mathrm{X}[:, 0{:}\,\mathrm{H}{:}\,2, 0{:}\,\mathrm{W}{:}\,2] \tag{9}$$

$$\mathrm{X}_{01} = \mathrm{X}[:, 0{:}\,\mathrm{H}{:}\,2, 1{:}\,\mathrm{W}{:}\,2] \tag{10}$$

$$\mathrm{X}_{10} = \mathrm{X}[:, 1{:}\,\mathrm{H}{:}\,2, 0{:}\,\mathrm{W}{:}\,2] \tag{11}$$

$$\mathrm{X}_{11} = \mathrm{X}[:, 1{:}\,\mathrm{H}{:}\,2, 1{:}\,\mathrm{W}{:}\,2] \tag{12}$$

Where: $X_{ab} \in \mathbb{R}^{C\times\frac{H}{2}\times\frac{W}{2}}$

Next, direction-aware context is encoded using three branches with depthwise convolutions of different shapes:

$$U_{hw} = f_{3\times3}(X_{00}), U_w = f_{1\times L}(X_{10}) \tag{13}$$

$$U_h = f_{L\times1}(X_{01}) \tag{14}$$

Where: f denotes a depthwise convolution with kernel size $m \times n$, and L is the length of the band-shaped kernel (in this work, L =11). Meanwhile, X_{11} is kept as an unconvolved identity branch to preserve fine-grained details and reduce additional computation.

Finally, the four feature streams are concatenated along the channel dimension and aggregated by a pointwise convolution:

$$X_{sf} = PWConv([X_{11}, U_{hw}, U_w, U_h]) \tag{15}$$

Where: $[\cdot]$ denotes channel-wise concatenation, and $PWConv(\cdot)$ is a 1×1convolution that maps the concatenated channels from $4C$ to the target channel size C'. Consequently $X_{sf} \in \mathbb{R}^{C'\times\frac{H}{2}\times\frac{W}{2}}$ is the output of the Spatial-Fusion module.

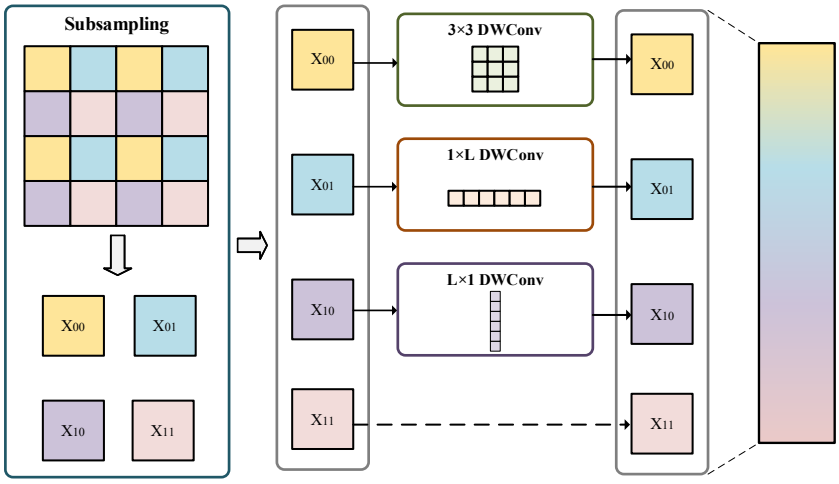

Fig. 6. Spatial-Fusion downsampling and Direction-Sensitive fusion process

3.5. SimSPPF multi-scale context aggregation

To enlarge the effective receptive field and incorporate multi-scale contextual information without further reducing feature resolution, as show in Figure 7, a SimSPPF module is inserted after the Spatial-Fusion module in the encoder. SimSPPF first applies a 1×1 convolution to compress the input channels, then performs three cascaded max-pooling operations. The resulting multi-scale features are concatenated along the channel dimension and fused back to the target channel size using another 1×1 convolution, given an input feature map $X \in \mathbb{R}^{C\times H\times W}$, channel compression is first performed as:

$$X_r = \delta(BN(Conv_{1\times1}(X))) \in \mathbb{R}^{\frac{C}{2}\times H\times W} \tag{16}$$

Where: $Conv_{1\times1}(\cdot)$ denotes a 1×1convolution, $BN(\cdot)$ denotes batch normalization.

Then, three cascaded max-pooling operations are applied while preserving the spatial resolution：

$$P_1 = MP_{k_1}(X_r) \tag{17}$$

$$P_2 = MP_{k_2}(P_1) \tag{18}$$

$$P_3 = MP_{k_3}(P_2) \tag{19}$$

Where: $MP_{k_1}(\cdot)$ denotes max pooling with kernel size k_i, using stride 1, and padding is applied to keep the output size at padding $H \times W$. In our implementation, $k_1 = 5, k_2 = 9, k_3 = 13$.

Finally, X_r and the pooled features are concatenated along the channel dimension and fused as:

$$X_{sppf} = \delta\left(BN\left(Conv_{1\times1}([X_r, P_1, P_2, P_3])\right)\right) \tag{20}$$

Where: $[\cdot]$ denotes channel-wise concatenation, and X_{sppf} is the output feature of the module.

This cascaded pooling strategy aggregates multi-scale context progressively without introducing additional downsampling, thereby improving the model's stable representation of defects at different scales (e.g., slender cracks and block-like spalling) while maintaining low parameter and computational overhead.

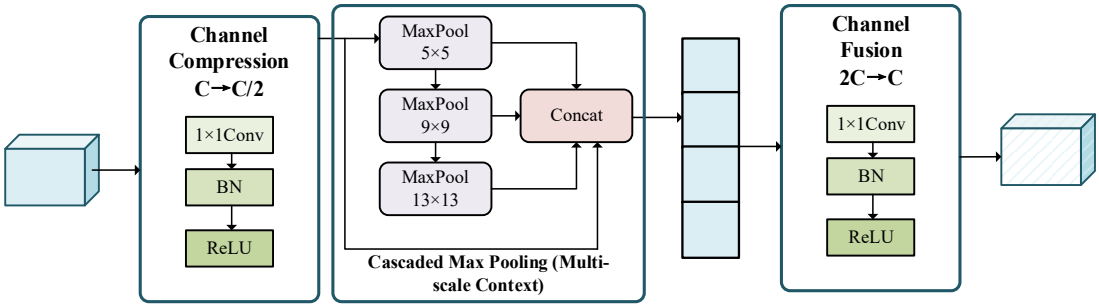

Fig. 7. SimSPPF Multi-Scale context aggregation module

4. EXPERIMENT

4.1. Dataset

Although rail-surface defects are not identical to defects on hot-rolled steel strips, they share certain characteristics. Therefore, we selected a subset from the NEU-DET (Song et al., 2014) dataset that includes four defect types visually similar to rail-surface defects: scratches, inclusion, patches, and crazing. We then performed pixel-level manual annotation on these four defect types.

We further constructed a self-built rail-surface image dataset using defective rail-surface images captured in practical acquisition scenarios. The images were collected using an iPhone 15 Pro Max camera under real rail-surface inspection conditions. The collected images contain representative rail-surface defects such as cracks, spalling, and pitting-like damage under complex backgrounds and texture disturbances. To support semantic segmentation, all images were manually annotated at the pixel level using LabelMe to generate ground-truth masks corresponding to the defect regions. After annotation, the dataset was randomly divided into training, validation, and test sets in an 8:1:1 ratio. As summarized in Table 1, the self-built dataset contains 447 annotated images in total, including 356 training images, 46 validation images, and 45 test images. Finally, all images and their corresponding masks were cropped to 256 × 256 pixels for network training and evaluation.

The category and quantity distribution of the NEU-DET dataset and the self-built rail surface image dataset are shown in Table 1.

Tab. 1. Data categories and quantity distribution

Category	Crazing	Patches	Inclusion	Scratches	Self-built
Training Nums	861	837	853	849	356
Validation Nums	89	91	87	88	46
Testing Nums	87	88	89	87	45

4.2. Hyperparameter settings

The experiments were conducted on a workstation equipped with an Intel i7-11700K CPU, an NVIDIA RTX 4090 GPU, and 64 GB of memory. The hyperparameter settings for model training are presented in Table 2.

Tab. 2. Hyperparameter settings

Hyperparameters	value
Batch size	4
Number of training epochs	300
Initial learning rate	1×10^{-6}
Optimizer	RMSprop（momentum=0.999，weight_decay=1e-8）
Input image resolution	256 × 256

4.3. Convergence performance analysis of SFAB-Net

4.3.1. Convergence performance analysis of SFAB-Net on the NEU-DET dataset

As shown in Figure 8, the training process of the SFAB-Net proposed in this paper converges well overall on the NEU-DET dataset. The loss curve drops rapidly during the early stage of training and then stabilizes at a relatively low level. In addition, the training and validation curves essentially coincide, indicating that the model optimization is stable and that no obvious overfitting occurred.

The F1 score, accuracy, and mIoU all rose continuously and eventually stabilized during training, indicating that SFAB-Net gradually learned effective defect features. Precision and recall increased in parallel, and validation results closely matched those on the training set, suggesting that SFAB-Net not only reduced false detections but also demonstrated strong detection performance.

Overall, these six groups of curves jointly indicate that SFAB-Net exhibits good trainability and generalization on NEU-DET.

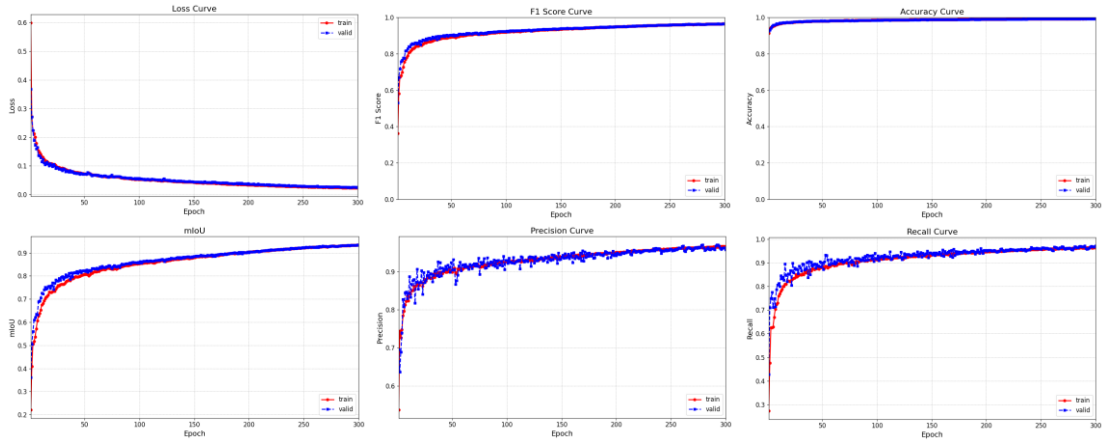

Fig. 8. Convergence curve of SFAB-Net on the NEU-DET dataset

4.3.2. Convergence performance analysis of SFAB-Net on the self-built rail surface image dataset

As shown in Figure 9, SFAB-Net exhibits stable training behavior on the self-built rail-surface dataset. The loss curve drops rapidly in the early stages. Although the validation curve fluctuated considerably, primarily due to large scene variations and complex defect morphologies, it showed an overall downward trend and eventually stabilized, suggesting that SFAB-Net adapts well to this dataset.

During training, the F1 score, accuracy, and mIoU steadily improve and then stabilize in later stages. The training and validation values remain similar, indicating that SFAB-Net effectively learned defect information from this dataset. Although the precision and recall curves fluctuated slightly compared with those for the NEU-DET dataset, their overall levels stayed relatively high, suggesting that SFAB-Net balanced detection rate and false-detection control in complex real-world scenarios.

In summary, the training process of SFAB-Net on the self-built rail surface image dataset is controllable, and all final indicators stabilize. This result demonstrates that the model is effective not only on standard public datasets but also on self-collected scenarios that more closely reflect practical applications, and it can be used for subsequent comparison and deployment analyses.

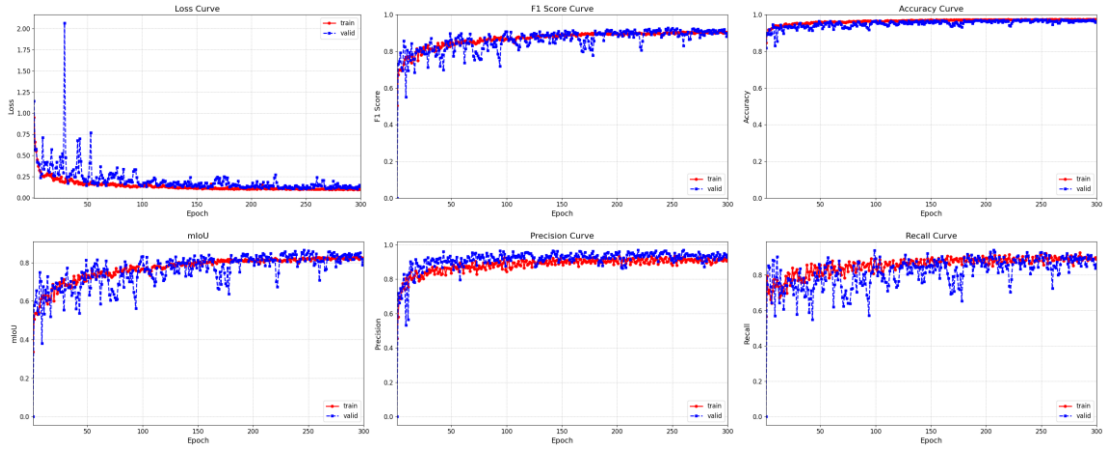

Fig. 9. Convergence curves of SFAB-Net on the self-built rail surface image dataset

4.4. Comparative experiments of SFAB-Net

In this section, we selected four relevant models for comparison: U-Net (Ronneberger et al., 2015), DeepLabv3+ (Chen et al., 2018), HRNet (Wang et al., 2020), and SegFormer (Xie et al., 2021). All models were trained and tested on the NEU-DET dataset and a self-built rail surface image dataset, using the training protocols and default parameters reported in their original papers.

4.4.1. Comparison of related methods

1. U-Net: Proposed by Ronneberger et al. (2015) U-Net is a classic encoder-decoder segmentation network. Through a symmetric contraction path, an expansion path, and multi-level skip connections, it captures contextual information while maintaining fine spatial resolution. It has been widely applied in medical and industrial visual segmentation tasks.
2. DeepLabv3+ (Chen et al., 2018): Based on DeepLabv3, DeepLabv3+ combines Atrous Spatial Pyramid Pooling (ASPP) with a lightweight encoder-decoder structure. It captures multi-scale contextual information at multiple sampling rates via dilated convolutions and refines object boundaries with a simple decoder branch. It has achieved excellent performance in semantic segmentation tasks of natural scenes.
3. HRNet (Wang et al., 2020): The High-Resolution Network (HRNet) connects multi-resolution sub-networks in parallel and repeatedly performs multi-scale feature exchange throughout the network. This enables it to integrate deep semantic information while maintaining high-resolution representations, demonstrating strong accuracy and robustness in tasks such as human pose estimation and semantic segmentation.
4. SegFormer (Xie et al., 2021): SegFormer is a Transformer–CNN hybrid segmentation framework proposed in recent years. It employs a hierarchical Transformer encoder to generate multi-scale features and uses a lightweight MLP decoder for feature fusion. It does not rely on positional encoding and achieves both accuracy and efficiency on multiple public datasets.

4.4.2. Comparison results on the NEU-DET dataset

Table 3 shows that SFAB-Net achieved the best performance on four metrics: mIoU was 93.92%, Precision was 96.52%, Recall was 97.21%, and AUROC-pixel was 98.40%. Compared with the classic U-Net (Ronneberger et al., 2015), SFAB-Net's mIoU increased by 4.45 percentage points, and its AUROC-pixel increased by 3.38 percentage points, indicating that while retaining the deployability of the encoder–decoder framework, it produced more stable pixel-level discrimination of defect regions. Compared with DeepLabv3+ (Chen et al., 2018) and HRNet (Wang et al., 2020), SFAB-Net attained higher mIoU and Recall, showing that it detected fine defects and boundary regions more comprehensively. Compared with the Transformer-based SegFormer (Xie et al., 2021), SFAB-Net maintained advantages in Precision, Recall, and AUROC-pixel. These results indicate that, for this task, SFAB-Net's lightweight combination of directional structure

modeling, multi-scale context aggregation, and decoding refinement achieved a better balance of accuracy and robustness.

Tab. 3. Comparison of segmentation performance of different methods on the NEU-DET (%)

Method	mIoU	Precision	Recall	AUROC-pixel
U-Net (Ronneberger et al., 2015)	89.47	92.18	92.96	95.02
DeepLabv3+ (Chen et al., 2018)	92.20	94.90	95.50	97.30
HRNet (Wang et al., 2020)	92.85	95.11	96.02	97.75
SegFormer (Xie et al., 2021)	93.40	94.93	95.98	97.91
SFAB-Net (ours)	**93.92**	**96.52**	**97.21**	**98.40**

4.4.3. Comparison results on the self-built rail surface image dataset

To evaluate the model's generalization in real acquisition scenarios, this section compares SFAB-Net with U-Net (Ronneberger et al., 2015), DeepLabv3+ (Chen et al., 2018), HRNet (Wang et al., 2020), and SegFormer (Xie et al., 2021) on the self-built rail surface image dataset. As shown in Table 4, SFAB-Net achieved the best performance across four metrics: mIoU 84.92%, Precision 92.62%, Recall 90.11%, and AUROC-pixel 94.05%. Relative to U-Net, mIoU rose by 4.07 percentage points, Precision by 5.22 percentage points, Recall by 3.81 percentage points, and AUROC-pixel by 3.95 percentage points, indicating that SFAB-Net provided more stable pixel-level discrimination under complex background and texture disturbances. Compared with DeepLabv3+, both mIoU and Precision increased by 3.52 percentage points and AUROC-pixel by 2.60 percentage points, showing that SFAB-Net better distinguished defect regions from the background. SFAB-Net also outperformed HRNet and SegFormer: versus HRNet, mIoU, Precision, Recall, and AUROC-pixel increased by 2.62, 1.47, 0.91, and 1.85 percentage points, respectively; versus SegFormer, they increased by 2.55, 1.57, 1.06, and 0.95 percentage points, respectively. Overall, SFAB-Net adapted better to scale variation, interference from noise, and unclear boundaries in real rail images.

Tab. 4. Comparison of segmentation performance of different methods on the self-built dataset (%)

Method	mIoU	Precision	Recall	AUROC-pixel
U-Net (Ronneberger et al., 2015)	80.85	87.40	86.30	90.10
DeepLabv3+ (Chen et al., 2018)	81.40	89.10	87.90	91.45
HRNet (Wang et al., 2020)	82.30	91.15	89.20	92.20
SegFormer (Xie et al., 2021)	82.37	91.05	89.05	93.10
SFAB-Net (ours)	**84.92**	**92.62**	**90.11**	**94.05**

4.4.4. Comparison of visualization results

As shown in Figure 10, each row presents five types of representative defect samples: the first column shows the original images, the second column shows the manually annotated ground-truth masks, and the third through seventh columns show the segmentation results of U-Net (Ronneberger et al., 2015), DeepLabv3+ (Chen et al., 2018), HRNet (Wang et al., 2020), SegFormer (Xie et al., 2021), and SFAB-Net, respectively. Overall, several comparison methods roughly outline the defect areas, but they differ in how they preserve detail and suppress background. For example, in the first row's block-spalling case, the main defect contours produced by U-Net and DeepLabv3+ are slightly shrunk, and small blocks above the main defect exhibit missing boundaries; HRNet and SegFormer provide more complete coverage but produce slightly rough boundaries. In the second and third rows, for defects composed of multiple small blocks or slender strips, U-Net and DeepLabv3+ tend to generate local fractures and fragmented regions, whereas HRNet and SegFormer recover most connected structures but still show adhesions or minor misclassifications in individual small blocks. In the fourth row, several methods reconstructed the central "waist" region either too thin or too thick, resulting in shape differences from the ground truth. In the fifth row, where multiple thin cracks appeared, the comparison methods generally detected the main cracks. However, U-Net and DeepLabv3+ produced slight interruptions at crack termini, and HRNet and SegFormer tended to expand boundaries. By contrast, the mask contours produced by SFAB-Net for each row most closely matched the manual annotations. The blocky defect regions remained continuous and complete, and multiple defects were properly separated. The length, orientation, and thickness of the slender cracks matched the ground truth, and the background showed almost no redundant noise response. These visualization results agreed with the quantitative indicators cited above

and further demonstrated that SFAB-Net preserved the integrity of defect areas while maintaining boundary accuracy and suppressing background noise.

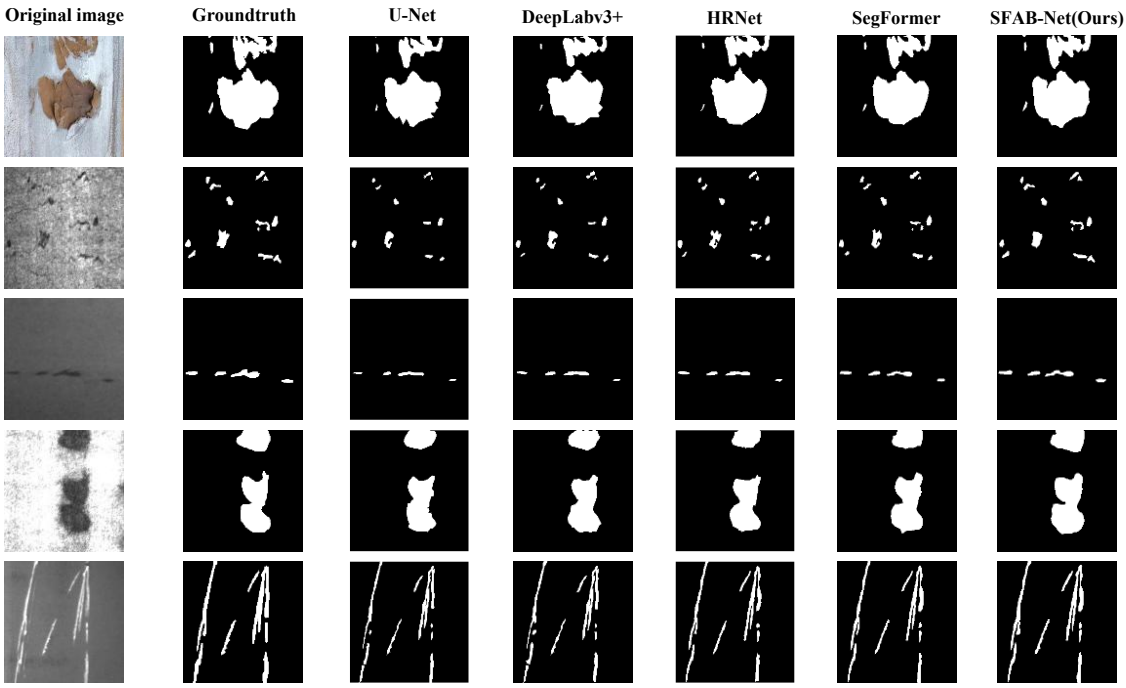

Fig. 10. Visualized segmentation results of different methods on typical defect samples

4.4.5. Comparison of model complexity

To evaluate the engineering deployability of the proposed model, we further compare its complexity with that of representative segmentation networks in terms of parameter count, FLOPs, and inference speed. As shown in Table 5, the traditional U-Net incurred relatively high parameter and computational costs (21.63 M parameters, 173.86 G FLOPs) and achieved an inference speed of approximately 18.12 FPS. In contrast, SFAB-Net maintained a parameter count of 5.96 M, which is substantially lower than common configurations of U-Net and HRNet. Under a 256 × 256 input, its FLOPs were 36.03 G. FLOPs, and FPS depend on input resolution, hardware platform, and implementation details; therefore, the baseline results in Table 5 provide an engineering-scale reference under a unified standard. Overall, while preserving a small parameter footprint, SFAB-Net allocated additional computational resources to direction-sensitive spatial modeling and multi-scale context enhancement, yielding a balanced trade-off between accuracy gains and computational overhead.

Tab. 5. Comparison of different model complexities and inference speeds

Method	Params (M)	FLOPs (G)	FPS
U-Net (Ronneberger et al., 2015)	21.63	173.86	18.12
DeepLabv3+ (Chen et al., 2018)	5.23	23.36	59.16
HRNet (Wang et al., 2020)	8.59	15.98	22.93
SegFormer (Xie et al., 2021)	4.13	7.08	64.25
SFAB-Net (ours)	5.96	36.03	46.70

4.5. Ablation experiments of SFAB-Net

To evaluate the individual contributions of the proposed modules, we conducted a progressive ablation study on our self-built rail surface dataset. As shown in Table 6, using the U-Net as the baseline (80.85% mIoU), the step-by-step integration of our components yields steady improvements.

The Spatial-Fusion module provides a significant initial boost (+1.85% mIoU) by effectively capturing the directional features of cracks. The subsequent addition of SimSPPF enhances multi-scale context aggregation, which notably increases Recall (from 87.80% to 89.00%) by preventing the missed detection of varying-sized defects. During the decoding phase, the CNNAdapter refines feature reconstruction, and the final scSE

attention mechanism successfully suppresses background noise. Ultimately, the complete SFAB-Net achieves the best overall performance, peaking at 84.92% mIoU and 92.62% Precision, proving the indispensability of each module.

Tab. 6. Ablation study of different modules on segmentation performance for the self-built dataset (%)

Spatial-Fusion	SimSPPF	CNNAdapter	scSE	mIoU	Precision	Recall	AUROC-pixel
				80.85	87.40	86.30	90.10
√				82.70	89.50	87.80	91.70
√	√			83.80	90.80	89.00	92.80
√	√	√		84.45	91.85	89.55	93.50
√	√	√	√	84.92	92.62	90.11	94.05

4.6. Statistical validation

To ensure the reliability of the performance gains, we conducted five independent runs for both the baseline and SFAB-Net using different random seeds. As summarized in Table 7, SFAB-Net consistently outperforms U-Net across all metrics with a lower standard deviation (e.g., 0.32% for mIoU). This statistical evidence confirms that the improvements are not due to stochastic variance but stem from the architectural robustness of the proposed modules.

Tab. 7. Statistical stability on the self-built dataset (%)

Method	mIoU	Precision	Recall	AUROC-pixel
U-Net	80.85±0.45	87.40 ±0.52	86.30±0.61	90.10±0.38
SFAB-Net	84.92±0.32	92.62 ± 0.28	90.11 ± 0.41	94.05±0.25

4.7. Performance under extreme conditions

To systematically analyze the limitations and boundary performance of SFAB-Net, we constructed challenging subsets by introducing specific extreme conditions into the test set. The quantitative degradation of model performance under these conditions is detailed in Table 8.

When evaluating images with high specular reflection or overexposure, the mIoU drops to 79.54%. Notably, the Recall decreases significantly (from 90.11% to 83.42%). This occurs because intense glare diminishes the local contrast, causing the model to miss hair-like or extremely slender cracks.

The introduction of heavy oil stains and complex rust patterns primarily affects the model's Precision, which falls sharply to 82.35%. These contaminants share textural and morphological features similar to those of actual spalling, occasionally leading the network to generate false positives.

While SFAB-Net maintains highly competitive performance on the standard dataset, these failure cases highlight the inherent challenges of purely vision-based rail inspection. In future work, we plan to integrate adaptive illumination preprocessing and explore multispectral imaging to better distinguish genuine structural defects from surface contaminants.

Tab. 8. Performance degradation of SFAB-Net under extreme conditions on the self-built dataset (%)

Method	mIoU	Precision	Recall	AUROC-pixel
Standard Test Set	84.92	92.62	90.11	94.05
+ Extreme Illumination	79.54	90.15	83.42	91.20
+ Heavy Oil / Rust	78.12	82.35	89.50	90.55

5. CONCLUSIONS

This paper addresses the coexisting challenges of strong directionality, large multi-scale variation, and lightweight constraints in the semantic segmentation of railway track surface defects. We propose and validate a segmentation network, SFAB-Net, that combines spatial fusion with adaptive bottleneck feature enhancement. Experimental results showed that incorporating direction-sensitive spatial fusion modeling effectively improved representation of slender cracks aligned with the track and of transverse structural

variations, reducing both crack fragmentation and structural adhesion. The multi-scale context aggregation and the adaptive feature reconstruction mechanism in the decoder preserved the integrity of defect regions and boundary accuracy in complex backgrounds.

Comparison results on public datasets and on real-collected railway track data indicate that the above-mentioned technical design achieves an effective balance between segmentation accuracy and computational efficiency without significantly increasing model complexity. SFAB-Net consistently outperforms the comparison methods in detecting minor defects, restoring complex shapes, and suppressing background noise, demonstrating the practical value of this direction-aware adaptive feature-enhancement strategy for rail-surface defect segmentation.

Acknowledgments

The authors declare that this research received no funding from any organization or institution.

Conflicts of Interest

The authors declare no conflict of interest.

REFERENCES

Chen, H., & Min, B. W. (2025). A high-precision segmentation network for industrial surface defect detection. *AIP Advances, 15*(5). https://doi.org/10.1063/5.0274903

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *ArXiv, abs/1802.02611*. https://doi.org/10.48550/arXiv.1802.02611

Demir, K., Ay, M., Cavas, M., & Demir, F. (2023). Automated steel surface defect detection and classification using a new deep learning-based approach. *Neural Computing and Applications, 35*(11), 8389–8406. https://doi.org/10.1007/s00521-022-08112-5

Frydrych, K., Tomczak, M., Jasiński, J., & Papanikolaou, S. (2025). Steel surface defects analysis with machine vision and deep learning. *The International Journal of Advanced Manufacturing Technology, 140*(7), 3691–3710. https://doi.org/10.1007/s00170-025-16539-y

Guclu, E., & Akin, E. (2025). Enhanced defect detection on steel surfaces using integrated residual refinement module with synthetic data augmentation. *Measurement, 250*, Article 117136. https://doi.org/10.1016/j.measurement.2025.117136

Guo, Q., Chen, Y., Zhu, Y., & Chen, D. (2025). CM-UNetv2: An enhanced semantic segmentation model for precise PCB defect detection and boundary restoration. *Sensors, 25*(16), Article 4919. https://doi.org/10.3390/s25164919

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(9), 1904–1916. https://doi.org/10.1109/TPAMI.2015.2389824

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). *MobileNets: Efficient convolutional neural networks for mobile vision applications*. *ArXiv, abs/1704.04861*. https://doi.org/10.48550/arXiv.1704.04861

Kumar, A., & Harsha, S. P. (2025). A systematic literature review of defect detection in railways using machine vision-based inspection methods. *International Journal of Transportation Science and Technology, 18*, 207–226. https://doi.org/10.1016/j.ijtst.2024.06.006

Li, H., Liu, M., Yin, Y., & Sun, W. (2025). Steel surface defect detection based on multi-layer fusion networks. *Scientific Reports, 15*(1), Article 10371. https://doi.org/10.1038/s41598-024-74601-3

Mao, Y., Zheng, S., Li, L., Shi, R., & An, X. (2024). Research on rail surface defect detection based on improved CenterNet. *Electronics, 13*(17), Article 3580. https://doi.org/10.3390/electronics13173580

Ming, G., Zhou, B., Luo, X., Ling, R., & Zhou, M. (2023). Rail surface defect detection method based on deep learning method with 3D range image. In *Advances in Frontier Research on Engineering Structures* (pp. 45–59). Springer Nature. https://doi.org/10.1007/978-981-19-8657-4_5

Pan, Y., Chen, J., Wu, P., Zhong, H., Deng, Z., & Sun, D. (2025). Enhanced rail surface defect segmentation using polarization imaging and dual-stream feature fusion. *Sensors, 25*(11), Article 3546. https://doi.org/10.3390/s25113546

Pappaterra, M. J., Pappaterra, M. L., & Flammini, F. (2024). A study on the application of convolutional neural networks for the maintenance of railway tracks. *Discover Artificial Intelligence, 4*(1), Article 30. https://doi.org/10.1007/s44163-024-00127-2

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. W N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (Lecture Notes in Computer Science, Vol. 9351, s. 234–241). Springer. https://doi.org/10.1007/978-3-319-24574-4_28

Roy, A. G., Navab, N., & Wachinger, C. (2018). Concurrent spatial and channel 'squeeze & excitation' in fully convolutional networks. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018* (pp. 421–429). Springer. https://doi.org/10.1007/978-3-030-00928-1_48

Si, C., Luo, H., Han, Y., & Ma, Z. (2024). Rail-STrans: A rail surface defect segmentation method based on improved Swin Transformer. *Applied Sciences, 14*(9), Article 3629. https://doi.org/10.3390/app14093629

Song, K., Hu, S., & Yan, Y. (2014). Automatic recognition of surface defects on hot-rolled steel strip using scattering convolution network. *Journal of Computational Information Systems, 10*(7), 3049–3055.

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., & Xiao, B. (2020). Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 43*(10), 3349–3364. https://doi.org/10.1109/TPAMI.2020.2983686

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. *ArXiv, abs/2105.15203*. https://doi.org/10.48550/arXiv.2105.15203

Yilmazer, M., & Karakose, M. (2024). Fastener and rail surface defects detection with deep learning techniques. *International Journal of Advances in Intelligent Informatics, 10*(2). https://doi.org/10.26555/ijain.v10i2.1237

Zhang, H., Zhao, Z., Liu, Y., Liu, J., Ma, T., Wu, K., & Wang, J. (2025). Steel surface defect segmentation with SME-DeeplabV3+. *PLoS One, 20*(8), Article e0329628. https://doi.org/10.1371/journal.pone.0329628

Zheng, D., Li, L., Zheng, S., Chai, X., Zhao, S., Tong, Q., & Guo, L. (2021). A defect detection method for rail surface and fasteners based on deep convolutional neural network. *Computational Intelligence and Neuroscience, 2021*, Article 2565500. https://doi.org/10.1155/2021/2565500