RBNet: An Ultra Fast Rendering-based Architecture for Railway Defects Segmentation

Mingxu Li, Bo Peng, *Member, IEEE*, Jian Liu, Donghai Zhai, *Member, IEEE*

***Abstract*—Inspection of railway defects is crucial for the safe and efficient operation of trains. Recent advancements in convolutional neural networks have led to the development of many effective detection and segmentation algorithms, however, these algorithms often struggle to balance efficiency and precision. In this paper, we present a rendering-based fully convolutional network that generates segmentation results through a coarse-to-fine approach. This allows our framework to make full use of low-level features while minimizing the number of parameters. Additionally, our network generates segmentation results from multiple scales of the feature map and uses residual connections to improve low-level feature detection. To improve training, we propose a novel method that augments the dataset by cutting and pasting images and corresponding ground truth labels horizontally. To better understand the patterns learned by our model, we also generate importance and uncertainty maps to make our model explainable. Our results show that the proposed method outperforms other state-of-the-art image segmentation methods with a higher frame rate and better performance.**

***Index Terms*—railway surface defects, image segmentation, rendering mechanism.**

I. Introduction

In the past hundred years, rail transit has always been a core transportation component. The demand for passenger and goods transportation has increased in tandem with society's and the economy's ongoing development, increasing the burden on rail surface maintenance. The repetitive motion of trains on rails causes various surface defects, such as cracks, spalling, corrugation, and rolling contact fatigue [1]. These flaws make the rail more likely to collapse within a few meters, which can result in serious accidents [1]. However, the state of railroads will continuously deteriorate. It poses a significant threat to railway safety, making it essential to efficiently and accurately monitor rail surface conditions for stable train movement.

Railway surface defects were primarily identified in the early years through manual inspection [2], [3]. This method had several drawbacks, including inefficiency, high costs, and subjectivity based on the operator's skill level. The limitations of manual inspection have become increasingly unacceptable as rail traffic continues to rise, necessitating the development of objective solutions like ultrasonic inspection, eddy current systems, and laser testing. The efficiency and objectivity of the results are enhanced by these inspection techniques, which are less expensive than manual inspection. These methods are very effective at detecting internal defects. However, due to the weak physical signals generated by the railway surface, they are limited in detecting surface defects. Besides, these methods are sensor-based, affecting the results by environmental changes and sensor quality.

Considering robustness and high precision, object segmentation technology based on Convolutional neural networks (CNN) has recently been utilized in numerous approaches and applications for railway maintenance and monitoring. Proper segmentation can help identify and locate railway surface defects and failures. These defects and failures can lead to accidents or other safety hazards if not detected and repaired, so proper segmentation allows for early detection and preventative maintenance, improving railway safety.

This paper aims to investigate how to quickly and effectively segment the defect regions on rails. Rail defect segmentation faces the following significant difficulties: 1) As depicted in Fig. 6, it is simple to mis-segment because the boundary between the defect region and the other region is unclear and contains a mix of different information; 2) Common segmentation models often have a large number of parameters, which leads to high costs to segment defects on thousands of kilometers of railway surfaces; 3) The current data augmentation techniques can alter the track's shape and frequently introduce artificial pixels. Given this, we design RBNet. Our contributions are as follows:

- A novel, lightweight network named RBNet is proposed to automate rail defect segmentation. During the rendering process, the residual connections in this network are utilized to retrieve the specifics of the low-level features.
- The segmentation model can achieve high accuracy with minimal annotated samples thanks to a data enhancement technique that avoids the issue of introducing artificial pixels.
- A loss function is designed to improve segmentation accuracy, allowing the network to focus more on error-prone regions.

The rest of the paper is organized as follows: Section II provides an overview of the related previous approaches. Section III details the implementation of the proposed model. Both the dataset enhancement method and the experimental results are presented in Section IV. Finally, Section V provides the conclusions.

II. Related Works

With the development of computer vision technology, researchers have developed various applications and approaches

to inspection and segmentation problems. Ge et al. [4] proposed a novel active contour method driven by an adaptive local pre-fitting energy function based on Jeffreys divergence (APFJD) for image segmentation. The APFJD model, on the other hand, can only effectively segment double-phase images and produces fair results when it comes to segmentation. Li et al. [5] used the local normalization (LN) method to raise the contrast of the rail images. Second, they use defect localization based on projection profile (DLBP) to find defects. While maintaining a time of operation of less than 20 ms per image, this strategy produces excellent detection results. Similar studies have been conducted like [6], [7].

Since the boom of deep learning, more and more CNN-based defect inspection models are typically used. Cao et al. [8] propose a deep feature fusion-based network. Features from different levels are extracted, outputs are generated, and fusing all branch outputs produces the segmentation result. However, due to the high computational cost, the algorithm cannot be used for a broader range of inspections. Aydin et al. [9] designed a lightweight and fusion model that combines the two models' features. This work used support vector machines (SVM) [10] to identify the faults from the output of the fusion model. Even though the fusion model provides richer features for each sample, it may lead to confusion for SVM if the features from the two encoders are vastly dissimilar, which can result in wrong results. Zhang et al. [11] used the YOLOv3 and the improved SSD algorithms to generate two groups of bounding boxes for the railway images. Then a fusion method combines the outputs and generates the final results. However, the defect detection model will not save the shape of the defect area, making it difficult for maintenance staff to conduct additional statistical analysis. Zheng et al. [12] suggested that the railway be extracted from the image using the improved YOLOv5 algorithm. The results of the segmentation are then created using the Mask RCNN algorithm. Unfortunately, simultaneously running multiple models can significantly burden the computing system. Jin et al. [13] obtained segmentation results using the Expectation-Maximization algorithm. Moreover, the Faster RCNN algorithm is adopted to get detection results. Finally, generate the segmentation map by taking an intersection for the detection and segmentation results. However, the algorithm is challenged by a variety of interferences in the complex railway environment. Zhang et al. [14] proposed a new model to deal with the challenge of complex backgrounds. The network fully utilizes context information based on dense block, pyramid pooling module, and multi-information integration. The article suggests cutting, grinding, turning, and welding natural samples to create artificial defects to expand the dataset to address the issue of difficult sample collection. Artificial defects may not wholly replicate the features of natural defects, even though they address the lack of real samples. This can result in abnormal features that may not accurately reflect the material's or product's performance under actual conditions.

Despite the success of deep learning-based methods in inspecting railway surface defects, they are not free of limitations. While CNN-based models have proven effective, they often include a large number of parameters, making them difficult to use on devices with limited computing power. Additionally, there is a risk of overfitting due to the scarcity of open data sets available for training these models.

III. Proposed method

This paper introduces a novel network architecture based on the rendering mechanism to address the challenges of high parameter count and low running efficiency in segmentation models. Furthermore, to mitigate the issue of mis-segmented edges, a new loss function is presented that emphasizes the edges of defect areas, resulting in improved segmentation accuracy. Additionally, a novel data augmentation method is developed, which avoids the introduction of artificial pixels and enables a higher accuracy of the model.

We use a probability representation, ranging from 0 to 1, for each pixel's likelihood of being a defect. Specific details of our proposed method are presented in Fig. 2. The model consists of two components: the baseline and the rendering mechanism. Our approach is similar to conventional U-shape-based models, but with two notable differences. First, instead of combining the feature maps together, the proposed model outputs the corresponding rail surface defects segmentation results based on the feature maps of each individual scale. Second, the RBNet involves only three max-pooling operations, which mitigates the loss of detailed information that can occur due to due to the feature map being of small size. A comparison between our proposed model and conventional U-shape-based structures is shown in Fig. 1.

A. Baseline

1) Boneback: The backbone network plays a crucial role in extracting features from the input images and passing them to the subsequent component. In order to achieve simplicity, we have adopted an encoder that consists of a three-stage down-sampling structure to extract features from the input image. Each stage consists of a convolution block with filter numbers of 64, 128, and 256. As illustrated in Fig. 2(a), the convolution block consists of a convolution operation, an activation function, and normalization. At the end of each stage, we use a max-pooling layer to preserve the essential features and reduce the size of the feature map.

2) Render Network: To improve the performance of the primary task, we proposed an auxiliary task that uses multi-scale features. As shown in Fig. 2, the proposed network employs an auxiliary task to approach the decoding process as a rendering problem and generate segmentation results layer by layer. Notably, our approach employs the auxiliary segmentation task only during the training phase and eliminates it for the testing phase, thus ensuring that the running speed of our method remains unaffected. In particular, the number of filters in the decoder block of each stage is the same as the corresponding encoder stage, and the feature maps are resized using the bilinear interpolation method. The pooling block consists of a convolutional layer that has 256 filters with a kernel size of 1×1, a batch normalization layer, and a ReLU layer. The output unit of the segmentation model generates the final segmentation map. It consists of a sigmoid layer

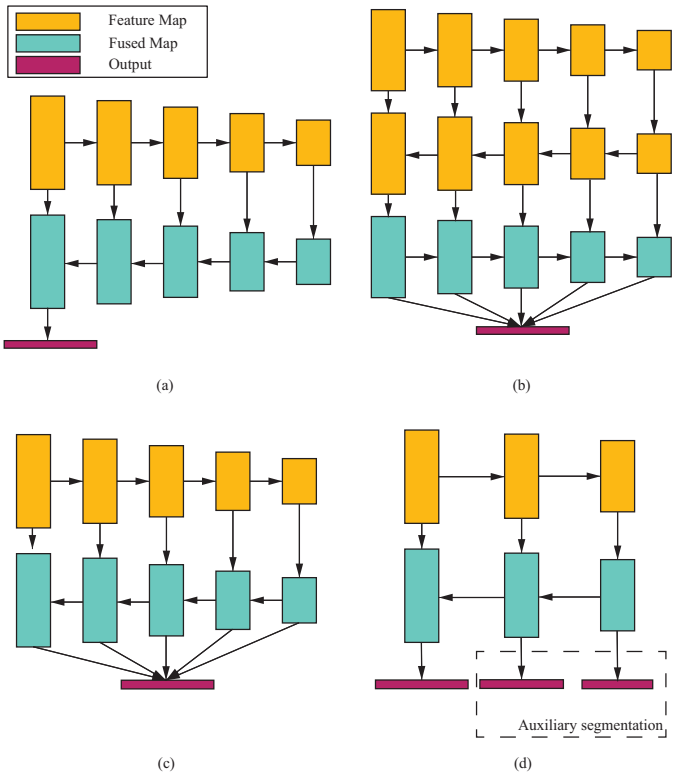


Fig. 1. The comparison between different designs. (a) U-Net combines the feature maps through lateral connections and produces the final result at the deepest layer. (b) PANet employs the U-shape structure with an additional bottom-up pathway. (c) FPN includes an extra unit for integrating features and creates the final output. (d) RBNet integrates high-level features to recover lost details and generates outputs of different resolutions.

followed by a convolutional layer that employs a filter with a 3 × 3 kernel size. This design enables the network to effectively exploit multi-scale features, thereby enhancing its performance on the primary segmentation task.

B. Loss Function

The proposed architecture generates outputs at multiple scales, and consequently, the loss function is a combination of multiple layers. Let M denote the number of up-sampling stages in our network. The parametric set of the backbone is denoted as W, and the parametric set for each output unit is denoted as $\omega \in \omega^{(1)}, \ldots, \omega^{(M)}$. Therefore, the total loss function can be expressed as the sum of the losses of all output layers:

$$\mathcal{L}(W, \omega) = \sum_{m=1}^{M} \alpha_m l(W, \omega^{(m)}) \tag{1}$$

The loss function used in our architecture generates outputs at different scales separately and is a combination of multiple layers. Let α_m be the weight of the loss in the m-th layer and l be the loss function.

For the railway surface defect segmentation problem, we observe that incorrect segmentation results are often distributed at the boundary of defects, as shown in Fig. 6. This is due to the fact that these pixels are located at the junction of two regions that mix different information, making them difficult to distinguish. In addition, rail defects are typically small, and there is a significant imbalance between positive and negative samples. As a result, whether or not the defect regions are correctly classified may have little impact on the value of the commonly used loss function. To address these issues, we propose a loss function defined as:

$$l^{(m)}(W, \omega^{(m)}) = \beta \times l_a^{(m)} + (1 - \beta) \times l_b^{(m)} \tag{2}$$

where l_a represents the loss of the boundary and l_b represents the loss of the defects area. To compute the loss of each output layer, we employ the Tversky Loss [15] and denote it as $l^{(m)}$. Formally, the Tversky Loss is defined as follows:

$$\begin{aligned} l = & \frac{\sum_{i \in Y^+} \Pr(\hat{y}_i = 1 | W, \omega)}{\sum_{i \in Y^+} \Pr(\hat{y}_i = 1 | W, \omega)} \\ & + \theta \times \sum_{i \in Y^+} \Pr(\hat{y}_i = 0 | W, \omega) \\ & + (1 - \theta) \times \sum_{i \in Y^-} \Pr(\hat{y}_i = 1 | W, \omega) \end{aligned} \tag{3}$$

where $\hat{y}_i \in (0, 1)$ denotes the output of the sigmoid function at pixel i. The term $\sum_{i \in Y^+} \Pr(\hat{y}_i = 1 | W, \omega)$ is the True Positives and θ is the weight of the False Negatives.

To improve the segmentation performance, especially on the edges of the defect regions, we use convolution operations to extract the boundaries of both the outputs and the ground truth. In binary images, similar to the Laplacian of the Gaussian edge detector [16], we use convolutional operations to extract the boundaries. The kernel is presented in Eq. (4):

$$k = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{4}$$

We use Y_a^+ and Y_a^- respectively to denote the set of defective pixels and the set of non-defective pixels in the edges of ground truth. Thus for l_a, Y^+ represents Y_a^+ in Eq.(3) and for l_b, Y^+ is the set of positive pixels in ground truth.

IV. Experimental Results And Analysis

In this section, we present the implementation of RBNet, which includes a proposed data enhancement method for the railway surface defects dataset, as well as the evaluation metrics and experimental results.

A. Implementation Details

We scaled the input image to 64×1280 to better preserve the railway features and minimize the number of parameters. The segmentation model, as shown in Fig. 2, consists of three outputs with decreasing resolutions from high to low, which we refer to as the 3rd, 2nd, and 1st outputs. In Eq. (1), M is equal to 3. To make RBNet focus on the last output, we set $\alpha_3 = \alpha_2 = 0.5$ and $\alpha_1 = 1$. As shown in Fig. 3, we chose the parameter value that gave the highest IoU and F1 scores and set β to 0.3. Additionally, we set θ to 0.3 in Eq. (3) because false negatives lead to more serious problems than false positives in defect inspection.

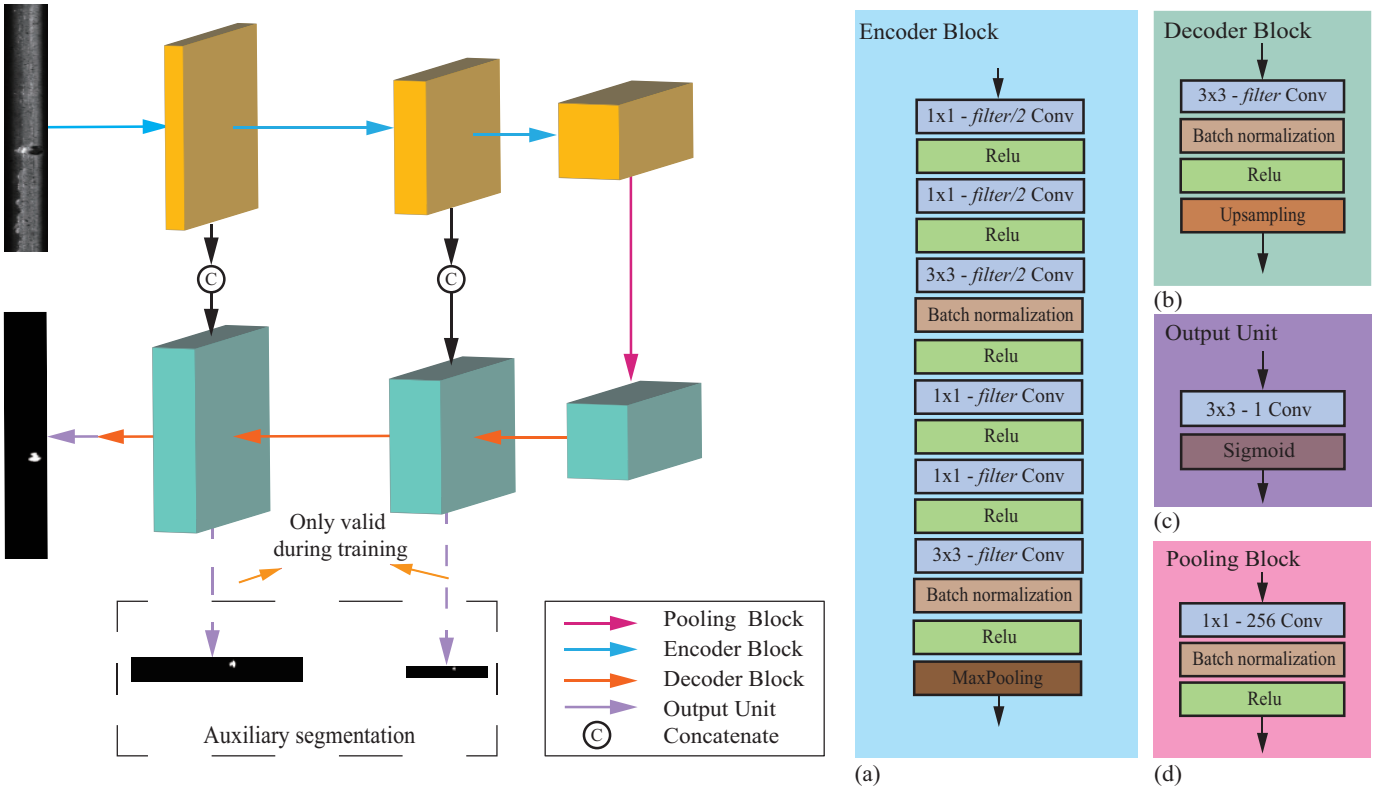


Fig. 2. Overview of the proposed neural network architecture for defect inspection. (a), (b), (c) and (d) are the details of the Encoder Block, the Decoder Block, the Output Unit and the Pooling Block.

During training, we used the Adam optimizer with a learning rate of 1e-4 to update the network parameters, and set the batch size to 8. All experiments were conducted on a machine equipped with an Nvidia RTX 3060 12G.

B. Evaluation Metrics

To provide a comprehensive evaluation, we use five evaluation metrics. Besides, True positives, false positives, true negatives and false negatives are referred to as TP, FP, TN, and FN respectively. The metrics are listed below:

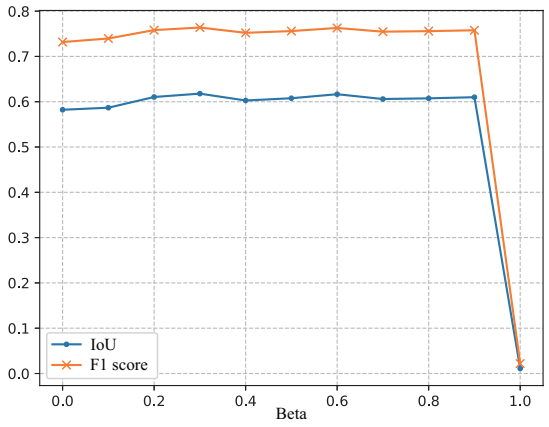


Fig. 3. The IoU and F1 scores vary along with *beta* value.

- Precision, indicates the proportion of true defects among all predicted defects, is defined as follows:

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

- Recall, which indicates the proportion of correctly predicted defects to all defect regions, is defined as follows:

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

- F-measure, which is the harmonic mean of precision and recall, is defined as follows:

$$F = \frac{(1 + \beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{7}$$

where β is the parameter and is used to control the influence between precision and recall. When β is set to 1, precision and recall have the same impact on the F-measure and the F-measure is named F1.

- IoU (Intersection over Union), which measures the overlap between the predicted and ground truth masks, giving the similarity between them, is defined as follows:

$$IoU = \frac{TP}{FP + TP + FN} \tag{8}$$

- Pixel Accuracy (PA), indicates the accuracy ratio of all pixels, is defined as follows:

$$PA = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

C. Dataset

To achieve effective and robust segmentation results, a large and diverse training dataset is critical. However, obtaining and annotating a comprehensive railway surface defect dataset can be challenging and expensive, resulting in a limited number of publicly available datasets. The RSDD dataset [17] collected by Gan et al. contains 67 and 128 samples for regular and high-speed railways. Although this dataset is well annotated and covers a wide range of scenarios, its size may not be sufficient for training CNNs due to the limited number of samples.

To address a problem in railway surface defect inspection, a data enhancement method based on the CutMix method [18] is proposed. The method comprises three steps. Firstly, two samples of the same type are randomly chosen and designated as x_A and x_B, with corresponding labels denoted as y_A and y_B, respectively. Subsequently, a cutting ratio R_c is randomly selected from the range [0.25, 0.75]. A binary mask $N_{64\times1280} \in \{0,1\}$, with the same size as the input images, is generated, indicating where to cut and fill in from the two samples. Finally, as illustrated in Fig. 4, the enhanced image is produced, and the enhanced image and labels are denoted as x_{mix} and y_{mix}, respectively:

$$\begin{aligned} x_{mix} &= N \odot x_A + (1-N) \odot x_B \\ y_{mix} &= N \odot y_A + (1-N) \odot y_B \end{aligned} \tag{10}$$

This data enhancement method addresses the limitation of the RSDD dataset, which includes scenarios of rail connections, without introducing artificial pixels. In addition, it merges two samples to create a fusion of different contextual information, resulting in an augmented dataset. After augmentation, the augmented dataset has a total of 5500 images, 5000 of which are used as the training set. To evaluate the effectiveness of our proposed data augmentation method, we conducted experiments comparing it with commonly used data augmentation techniques (random rotation, scaling, cropping, and mirroring) on the rail surface defect segmentation problem. As shown in Table. I, it is worth noting that our proposed method produced significantly better results.

D. Experimental Results

To evaluate the segmentation performance of RBNet, we compared the model with several state-of-the-art algorithms.

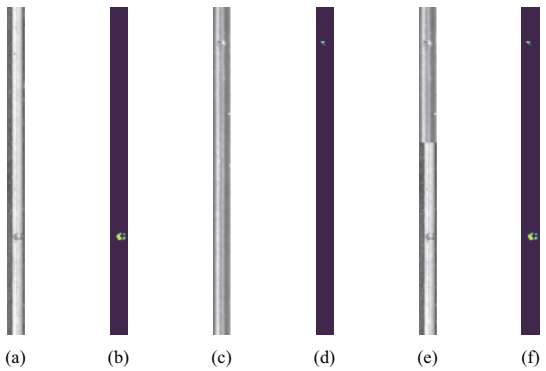


Fig. 4. Visualization of the data augmentation. (a) x_A. (b) y_A. (c) x_B. (d) y_B. (e) x_{mix}. (f) y_{mix}.

TABLE I
QUANTITATIVE COMPARISON BETWEEN DIFFERENT AUGMENTATION METHODS

Method	Precision	Recall	F1	IoU	PA
the proposed method	**0.8187**	**0.7342**	**0.7741**	**0.6315**	**0.9953**
random rotation et al.	0.7753	0.4169	0.5423	0.3720	0.9919

All algorithms were implemented using the same experimental setup as our model to ensure a fair comparison.

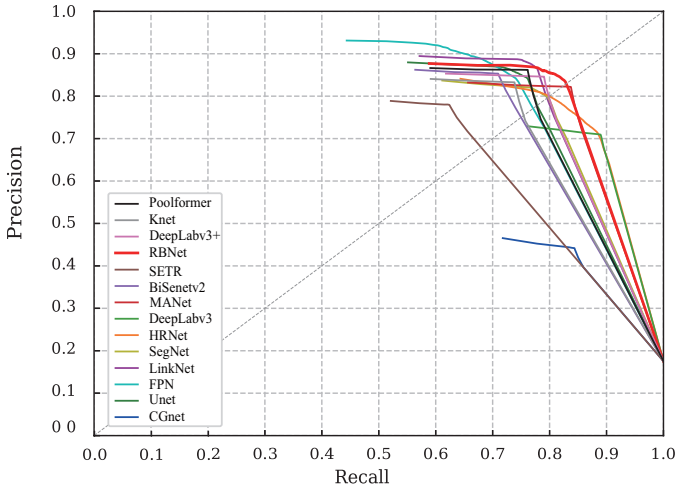


Fig. 5. Precision–Recall curves on the enhanced RSDD dataset.

TABLE II
QUANTITATIVE COMPARISON ON OUR DATASET

Method	Precision	Recall	F1	IoU	PA
RBNet	0.8187	0.7342	**0.7741**	**0.6315**	**0.9953**
SETR [19]	0.7895	0.5413	0.6422	0.4730	0.9952
BiSenetv2 [20]	0.8410	0.5789	0.6858	0.5218	0.9940
MANet [21]	0.7929	0.7429	0.7671	0.6222	**0.9953**
DeepLabv3 [22]	0.6756	**0.8401**	0.7489	0.5986	0.9936
HRNet [23]	0.7178	0.7986	0.7560	0.6078	0.9942
SegNet [24]	0.7796	0.6683	0.7197	0.5621	0.9941
LinkNet [25]	0.8558	0.6460	0.7363	0.5826	0.9948
FPN [26]	**0.8627**	0.5632	0.6815	0.5170	0.9940
Unet [27]	0.8200	0.6381	0.7176	0.5997	0.9943
CGnet [28]	0.2284	0.7657	0.3519	0.2135	0.9681
DeepLabv3+ [29]	0.8228	0.6426	0.7216	0.5645	0.9681
Knet [30]	0.8510	0.6134	0.7129	0.5538	0.9944
Poolformer [31]	0.8572	0.6496	0.7391	0.5862	0.9948

The experimental results, as shown in Table II, indicate that RBNet achieves a precision of 0.8186, a recall of 0.7341, an F1 score of 0.7741, and an IoU of 0.6315. The comparison with state-of-the-art algorithms, including HRNet, CGNet, and DeepLabv3, shows that these methods have higher recall values than RBNet. However, their precision and F1 scores are significantly lower, which does not necessarily indicate better segmentation performance.

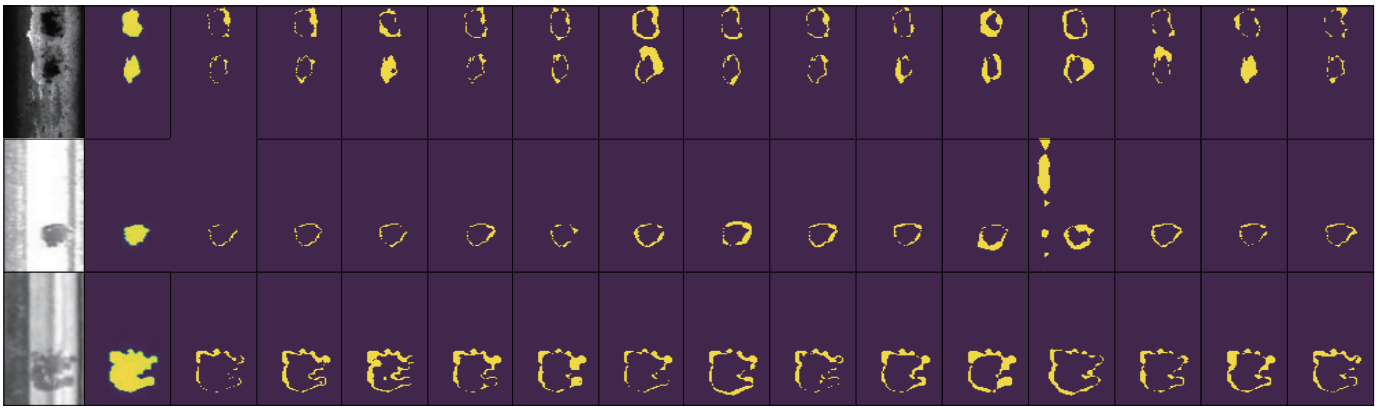


Fig. 6. Visualization of the difference between ground truth and outputs obtained by various models. The fewer bright yellow points from (c) to (j), the better the algorithm is. (a) Original images. (b) Ground truth.. (c) RBNet. (d) Unet. (e) FPN. (f) LinkNet. (g) SegNet. (h) HRNet. (i) DeepLabv3. (j) MANet. (k) BiSenetv2. (l) SETR. (m) CGNet. (n) DeepLabv3+. (o) Knet. (p) Poolformer.

Furthermore, we plotted a precision-recall (PR) curve to further evaluate the performance of RBNet. As shown in Fig. 5, the precision of the FPN model is slightly higher than that of our model in the region with lower recall. However, RBNet outperforms the FPN model in the higher recall region. This analysis shows that RBNet is an effective algorithm for accurate detection and segmentation of rail surface defects.

Furthermore, to further evaluate the segmentation performance of the proposed RBNet algorithm, we selected several samples and calculated their segmentation results. Since the differences between the outputs of different segmentation approaches are often tiny, we used visualization techniques to highlight these differences. Specifically, we compared the ground truth segmentation with the results obtained by different methods and calculated the differences, which we denoted as D:

$$D = GT \cup Pre - GT \cap Pre \tag{11}$$

The ground truth segmentation is denoted as GT, while the output from different models is denoted as Pre. The visualization results are shown in Fig. 6. In the error map D, pixels in bright green represent mis-segmented pixels. As can be seen in the figure, RBNet produces fewer errors and is able to segment rail defects more accurately. In particular, RBNet shows fewer errors at the edges, which can be attributed to the layer-by-layer rendering mechanism and the hybrid loss function.

To further validate our approach, we also used real-world images provided by ProRail [32]. The results are shown in Fig. 7, which indicates that our model performs well in real-world scenarios and has a strong generalization ability.

E. Robustness and Generalization Ability

To assess the robustness and generalization of our model to different sets of training and test images, we performed a 5-fold cross-validation. This ensures that both sets are from a similar domain. The experimental results of the 5-fold cross-validation are shown as box plots in Fig. 8. Note that the differences between the experiments are negligible, as all

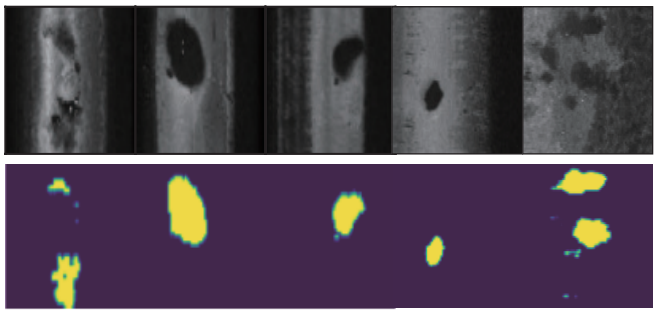


Fig. 7. Qualitative results of RBNet on the dataset provided by ProRail.

median lines fall inside the boxes. The interquartile ranges for each metric are also not significantly different, indicating that the results are not widely distributed. Therefore, the cross-validation results demonstrate the generalizability and robustness of our proposed model on unseen data.

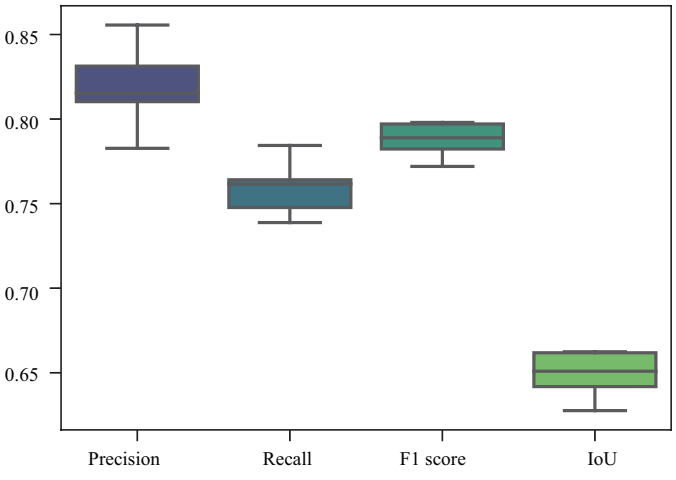


Fig. 8. Box plot for cross-validation results.

In addition, we conducted experiments on several challenging samples from the dataset to evaluate the robustness of the proposed model in dealing with noise, such as rust marks,

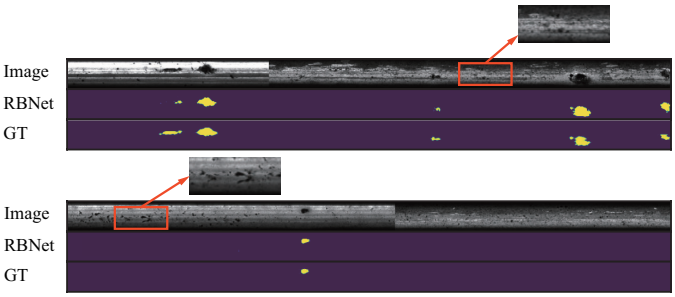


Fig. 9. Performance of the algorithm under different types of noise.

stains, and spalling, which are often present on the railway surface, as shown in Fig. 9. The experimental results show that RBNet works well and there is almost no mis-segmentation when noise is present on the track. This can be partly attributed to the model have learned reasonable patterns during the training process and the extracted features for defects and noise are significantly different.

F. Ablation Studies

To analysis the effect of each module in RBNet, ablation studies are conducted on the enhanced dataset. All other configurations are kept the same, except for the variations specified in the experiments. The results of the experiments are presented in Table III. The model proposed in Fig. 2 is taken as the baseline. The results show that the proposed combination of rendering mechanism and loss function has significant advantages over the other configurations in the main indicators. Experiment (a) achieves the best performance compared to experiments (b), (c), (d) and (e), which shows that the improvement of RBNet is cumulative and progressive.

TABLE III
THE ABLATION STUDIES FOR THE PROPOSED ARCHITECTURE ON OUR DATA SETS

Settings	Precision	Recall	F1	IoU
(a) Ours	**0.8187**	**0.7342**	**0.7741**	**0.6315**
(b) without rendering mechanism				
+ with the proposed loss function	0.8043	0.6995	0.7482	0.5978
(c) without rendering mechanism				
+ with the loss function in 1, 2 th layer	0.7987	0.6932	0.7422	0.5951
(d) without rendering mechanism				
+ with the loss function in 1 th layer	0.7835	0.6995	0.7391	0.5949
(e) without rendering mechanism				
+ without the proposed loss function	0.7811	0.6937	0.7348	0.5947

G. Model Complexity

Efficiency is crucial for defect inspection, and computational complexity plays a significant role in determining it. In this section, we conducted an analysis based on the number of parameters (Params), frames per second (FPS), and floating-point operations (FLOPs). Table IV presents the results of all methods. Compared to the competing models, RBNet strikes a balance between speed and accuracy while maintaining a reasonable number of parameters and FLOPs.

TABLE IV
THE COMPLEXITY ANALYZE FOR DIFFERENT ARCHITECTURE

Models	Params	FPS	FLOPs
RBNet	2.36 M	**95.60**	**4.71 M**
SETR	310.68 M	14.98	66.42 G
BiSenetv2	14.8 M	53.06	3.87 G
MANet	140.61 M	57.96	23.28 G
DeepLabv3	39.63 M	51.96	51.23 G
HRNet	28.55 M	4.60	57.11 M
SegNet	2.94 M	87.34	5.871 M
LinkNet	28.73 M	23.36	57.46 M
FPN	26.86 M	24.41	53.73 M
Unet	32.51 M	24.30	65.03 M
CGNet	**0.50 M**	50.19	1.07 G
DeepLabv3+	43.58 M	28.17	55.07 G
Knet	81.31 M	24.72	85.29 G
Poolformer	59.74 M	27.20	21.05 G

V. CONCLUSION

This study presents a lightweight fully convolutional network for railway defect segmentation which introduces a novel rendering mechanism to the segmentation task. The rendering mechanism is designed to compensate for the low-level information lost during the encoding process. In addition, a novel loss function is employed to handle the imbalance between positive and negative samples and to address edge-prone error prediction. The performance of the proposed method is evaluated through extensive experiments, and the results show that it outperforms existing state-of-the-art models. With a small number of parameters and fast computation speed, RBNet is suitable for low computational power devices and has significant application potential.

Despite achieving a lightweight design, the proposed network lacks the ability to extract high-level semantics, rendering it incapable of directly segmenting railway defects from images with complex backgrounds such as sleepers or insulated joints, as shown in Fig. 10. In addition, accurately

Fig. 10. A sample that our model is unable to provide optimal results.

determining the severity of the broken area from digital images is of significant value in railway defect detection and is a research direction we intend to explore in the future.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (No. 61961038), the National Key Research and Development Program of China (No. 2019YFB1705602), and the Fundamental Research Funds for the Central Universities (No. 2682021ZTPY069).

REFERENCES

[1] D. Cannon, K.-O. Edel, S. Grassie, and K. Sawley, "Rail defects: an overview," *Fatigue & Fracture of Engineering Materials & Structures*, vol. 26, no. 10, pp. 865–886, 2003.

[2] F. Marino, A. Distante, P. L. Mazzeo, and E. Stella, "A real-time visual inspection system for railway maintenance: automatic hexagonal-headed bolts detection," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 3, pp. 418–428, 2007.

[3] M. Molodova, Z. Li, A. Núñez, and R. Dollevoet, "Automatic detection of squats in railway infrastructure," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 1980–1990, 2014.

[4] P. Ge, Y. Chen, G. Wang, and G. Weng, "An active contour model driven by adaptive local pre-fitting energy function based on jeffreys divergence for image segmentation," *Expert Systems with Applications*, vol. 210, p. 118493, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417422015779

[5] Q. Li and S. Ren, "A real-time visual inspection system for discrete surface defects of rail heads," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 8, pp. 2189–2199, 2012.

[6] M. Yongzhi, Y. Biao, M. Hongfeng *et al.*, "Rail surface defects, detection based on gray scale gradient characteristics of image,[j]," *Chinese Journal of Scientific Instrument*, vol. 39, no. 4, pp. 220–229, 2018.

[7] F. Wang, T. Xu, T. Tang, M. Zhou, and H. Wang, "Bilevel feature extraction-based text mining for fault diagnosis of railway systems," *IEEE transactions on intelligent transportation systems*, vol. 18, no. 1, pp. 49–58, 2016.

[8] J. Cao, G. Yang, and X. Yang, "A pixel-level segmentation convolutional neural network based on deep feature fusion for surface defect detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[9] I. Aydin, E. Akin, and M. Karakose, "Defect classification based on deep features for railway tracks in sustainable transportation," *Applied Soft Computing*, vol. 111, p. 107706, 2021.

[10] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," 1998.

[11] H. Zhang, Y. Song, Y. Chen, H. Zhong, L. Liu, Y. Wang, T. Akilan, and Q. J. Wu, "Mrsdi-cnn: Multi-model rail surface defect inspection system based on convolutional neural networks," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[12] D. Zheng, L. Li, S. Zheng, X. Chai, S. Zhao, Q. Tong, J. Wang, and L. Guo, "A defect detection method for rail surface and fasteners based on deep convolutional neural network," *Computational Intelligence and Neuroscience*, vol. 2021, 2021.

[13] X. Jin, Y. Wang, H. Zhang, H. Zhong, L. Liu, Q. J. Wu, and Y. Yang, "Dm-ris: Deep multimodel rail inspection system with improved mrf-gmm and cnn," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 4, pp. 1051–1065, 2019.

[14] D. Zhang, K. Song, J. Xu, Y. He, M. Niu, and Y. Yan, "Mcnet: Multiple context information segmentation network of no-service rail surface defects," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[15] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, "Tversky loss function for image segmentation using 3d fully convolutional deep networks," in *International workshop on machine learning in medical imaging*. Springer, 2017, pp. 379–387.

[16] V. Torre and T. A. Poggio, "On edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 2, pp. 147–163, 1986.

[17] J. Gan, Q. Li, J. Wang, and H. Yu, "A hierarchical extractor-based visual rail surface inspection system," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7935–7944, 2017.

[18] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6023–6032.

[19] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr *et al.*, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 6881–6890.

[20] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation," *International Journal of Computer Vision*, vol. 129, no. 11, pp. 3051–3068, 2021.

[21] T. Fan, G. Wang, Y. Li, and H. Wang, "Ma-net: A multi-scale attention network for liver and tumor segmentation," *IEEE Access*, vol. 8, pp. 179 656–179 665, 2020.

[22] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[23] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5693–5703.

[24] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[25] A. Chaurasia and E. Culurciello, "Linknet: Exploiting encoder representations for efficient semantic segmentation," in *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2017, pp. 1–4.

[26] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6399–6408.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

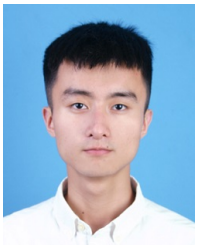
[28] T. Wu, S. Tang, R. Zhang, J. Cao, and Y. Zhang, "Cgnet: A light-weight context guided network for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 1169–1179, 2021.

[29] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018.

[30] W. Zhang, J. Pang, K. Chen, and C. C. Loy, "K-Net: Towards unified image segmentation," in *NeurIPS*, 2021.

[31] W. Yu, M. Luo, P. Zhou, C. Si, Y. Zhou, X. Wang, J. Feng, and S. Yan, "Metaformer is actually what you need for vision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 819–10 829.

[32] ProRail, "Image data of insulation joints - prorail," https://www.kaggle.com/datasets/oscarvanhees/insulation-joint-training-set-prorail, 2019.



Mingxu Li received the M.S. degree in software engineering from Southwest Jiaotong University, Chengdu, China, in 2021. He is currently pursuing his Ph.D. degree in computer science technology from Southwest Jiaotong University. His research interests include automatic driving and computer vision.

Bo Peng (IEEE Member) is an Associate Professor in the School of Computing and Artificial Intelligent, Southwest Jiaotong University, Chengdu, China. She received the M.S. degree from the Department of Computer Science, University of Western Ontario (UWO) in 2008 and the PhD degree from the Department of Computing, The Hong Kong Polytechnic University in 2012. From Aug. 2011 to Jan. 2012, she worked as a Research Assistant in the Department of Computing, The Hong Kong Polytechnic University. Her research interests include image segmentation, segmentation quality evaluation, and pattern recognition. She is a member of IEEE.

Jian Liu received the Ph.D. degree in mathematics from the University of California at Los Angeles, Los Angeles, CA, USA, in 2009. He is currently a Lecturer with the School of Computing, University of Leeds, Leeds, U.K. His current research interests include computational neuroscience and brain-like computation.

Donghai Zhai (IEEE Member) received his Ph.D. degree in traffic information engineering and control from Southwest Jiaotong University, China, in 2003.

From 2003 to 2005 he was employed at IBM China Research Laboratory. Since 2006 he has been associated with Southwest Jiaotong University in School of Computing and Artificial Intelligence. He has been a visiting scholar at Louisiana State University, Baton Rouge in 2016. His research interests include autonomous driving, digital image processing, computer vision, and pattern recognition.