DOI: 10.1111/mice.12710

RESEARCH ARTICLE

COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING WILEY

Hybrid deep learning architecture for rail surface segmentation and surface defect detection

Yunpeng Wu[1,2] | **Yong Qin**[1] | **Yu Qian**[2] | **Feng Guo**[2] | **Zhipeng Wang**[1] | **Limin Jia**[1]

[1] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

[2] Department of Civil and Environmental Engineering, University of South Carolina, Columbia, South Carolina, USA

Correspondence
Yong Qin, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China.
Email: yqin@bjtu.edu.cn
Yu Qian, Department of Civil and Environmental Engineering, University of South Carolina, Columbia, SC 29208, USA.
Email: yuqian@sc.edu

Funding information

National Natural Science Foundation of China, Grant/Award Number: 91738301; Research project of Beijing Shanghai High Speed Railway Company, Grant/Award Number: I20D00010

Abstract

Rail surface defects (RSDs) are a major problem that reduces operation safety. Unfortunately, the existing RSD detection systems have very limited accuracy. Current image processing methods are not tailored for the railway track and many fully convolutional networks (FCN)-based methods suffer from the blurry rail edges (RE). This paper proposes a new rail boundary guidance network (RBGNet) for salient RS detection. First, a novel architecture is proposed to fully utilize the complementarity between the RS and the RE to accurately identify the RS with well-defined boundaries. The newly developed RBGNet injects high-level RS object information into shallow RS edge features by a progressive fused way for obtaining fine edge features. Then, the system integrates the refined edge features with RS features at different high-level layers to predict the RS precisely. Second, an innovative hybrid loss consisting of binary cross entropy (BCE), structural similarity index measure (SSIM), and intersection-over-union (IoU) is proposed and equipped into the RBGNet to supervise the network and learn the transformation between the input and ground truth. The input and ground truth then further refine the RS location and edges. Conveniently, an image-based model for RSD detection and quantification is also developed and integrated for an automatic inspection purpose. Finally, experiments conducted on the complex unmanned aerial vehicle (UAV) rail dataset indicate the system can achieve a high detection rate with good adaptation capability in complicated environments.

1 | INTRODUCTION

The rapid growth of the railroad network has put tremendous pressure on track inspection and maintenance. As of 2020, United States has over 250,000 km of railroad track, which is the biggest network in the world (Railway Technology, 2020). China operates about 141,400 km of track, ranking the second in the world, while its 36,000 km of high-speed track is the most comprehensive high-speed passenger service network in the world (Xinhuanet, 2020). Russia and India rank third and fourth in terms of the track mileage with over 85,500 km and 65,000 km of track, respectively. Rail breakage, rail defects, and derailment are the leading factors of train accidents (Guo et al., 2021; Sharma et al., 2018). Specifically, it is reported that around 90% of railway derailment accidents can be related to rail defects (AlNaimi, 2020). In general, rail surface defects (RSDs) reference to the loss of materials on the rail head

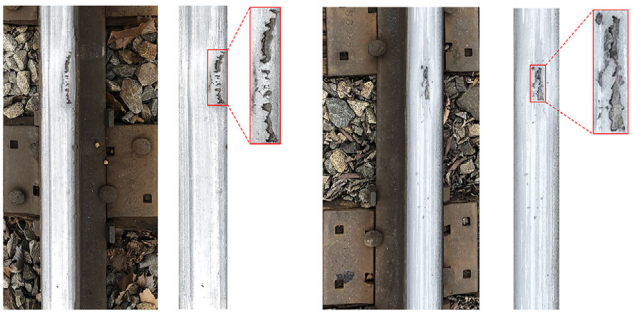

FIGURE 1 Examples of rail surface defect

surface, usually within a very shallow depth and generated on the rail surface in an apparently random manner (overload, severe corrosion and so on), as shown in Figure 1. The factors leading to rail surface defects are complex but can be primarily related to the rail and wheel interactions. The presence of rail surface defect does not mean the track is unsafe. However, if not properly addressed, rail surface defect could lead to more severe issues, such as loss of running surface, rail breakage, and eventually, derailment. Thus, rail surface defect is inspected regularly and is one of the key factors in rail maintenance planning in practice. Historically, railway tracks are inspected by trained personnel. However, manual inspection has low efficiency and low accuracy because it is heavily depending on the experience of the inspectors (X. Kong & Li, 2018; Marino et al., 2007). Many automatic track inspection systems have been developed over time, usually mounted on an inspection car or a hi-rail vehicle with various types of sensors. Those systems, mainly based on laser, acoustic emission, and ultrasonic wave technologies, did not resolve the challenges to accurately detect surface defects (Q. Li & Ren, 2012b; Wu, Qin, Wang, et al., 2018). Systems based on LiDAR (Ariyachandra & Brilakis, 2020) and ground penetration radar (Ciampoli et al., 2020) are effective in identifying or monitoring relatively large targets, such as railway mast and railway ballast, but not small targets. Unlike vision methods, those aforementioned systems have limited effectiveness in detecting the RS defects due to lack of ample heuristic structure information or texture features for rail surface defect detection (see Figure 1).

Visual inspection systems using the image processing (IP) algorithms and deep learning-based object detection methods have been introduced for the RS inspection over the past decade. Typically, these systems use images taken by cameras mounted on rail inspection vehicles, so the models are usually developed based on images taken with a consistent angle and good contrast between the rail surface and the track. For instance, Q. Li and Ren (2012b) proposed an IP-based inspection system, which uses the track extract based on projection profile (TEBP) algorithm to segment track regions, and then uses local normalized (LN) and defect location based on projection profile (DLBP) for RSD detection. Meanwhile, Q. Li and Ren (2012a) also presented a histogram-based tack extraction (HBTE), combined with Michelson-like contrast (MLC) and proportion emphasized maximum entropy (ME) algorithms for RS segmentation and RSD detection. However, TEBP and HBTE cannot handle arbitrary-oriented RS in images, especially images taken by a portable platform, such as hand-held and unmanned aerial vehicle (UAV) cameras. Furthermore, the LN performs poorly with images having disorder pixel level due to corrosion or irregular points. He et al. (2016) presented an inverse Perona-Malik diffusion model, which aims to enhance the LN method by taking the reciprocal of the image gradient as a feature to adjust the diffusion coefficient. Nevertheless, the method still yields high false detection rate for RSD detection, especially for highly irregular interference. Recently, new deep learning-based object detection methods (Wei et al., 2020; Yanan et al., 2018) used improved YOLOv3 (You only look once version 3) model for RSD detection and achieved results. However, it should be noted these studies only focused on RSD detection from a clearly outlined RS. Without RS recognition capability from some general images, RSD detection models by themselves cannot be applied to practices.

Specifically, there are two main difficulties preventing these existing models from being applied in the field. First, the detection accuracy is highly sensitive to the image quality, especially the shooting angle and the image contrast. This requires the cameras to be mounted highly precisely and with consistent illumination conditions. Unfortunately, a consistent image shooting angle and illumination condition is hard to achieve in practice due to a multitude of factors, such as the hunting motion of the cars, varied levels of sunlight, and track appearances. The oscillation of the car body is especially inevitable. Second, the inspection intervals are dependent on the availability of the inspection vehicles, which has become more difficult to schedule for the saturated timetable. Thus, alternative methods and platforms have become popular, including handheld cameras, on-track robotics, and UAVs.

Lately, using UAV based cameras has drawn great attention due to its user-friendly convenience factor. More importantly, using UAVs to acquire RS images does not require track time and would not disturb the normal train operations. The hardware cost is also much lower compared with the previous vehicle mounted systems. Wu, Qin, and Jia (2018) proposed a Hough-based pixel column cumulation gray (HPCG) algorithm which extracts edge features to detect salient RS regions from UAV images. RSD was then detected with the maximum entropy algorithm. Later, Wu, Qin, Wang, et al. (2018) proposed another RS

enhancement algorithm called local Weber-like contrast (LWLC) in order to supplement the previous approach for RSD detection and achieved reasonable results. Unfortunately, the two systems using HPCG often fail on RS segmentation from UAV images when the variations of track width, location, and sunlight intensity are large. Note that the image consistency issue is more pronounced when using UAV-based cameras, which prevent UAV-based rail inspection methods from more popular applications.

In sum, for these RSD inspection systems based on an inspected vehicle, the RS segmentation methods are not able to handle arbitrary-oriented RS in images, especially images taken by a portable platform, such as hand-held and UAV cameras. For the existing RSD detection systems aimed at UAV images, the segmentation methods using HPCG often fail during line detection or track width setting from UAV images when the variations of track width, location, and sunlight intensity are large, thereby leading to the RS segmentation failure.

Fortunately, following the development of neural network and computer vision in recent decades, attractive attempts have devoted into applications for infrastructure in civil engineering (Adeli, 2001; Amezquita-Sanchez et al., 2016). Examples include civil engineering application of neural networks (Adeli & Yeh, 1989), building damage detection (Rafiei & Adeli, 2017b; Wang et al., 2020), earthquake early warning (Rafiei & Adeli, 2017a), construction vehicle detection (Arabi et al., 2020), structure health assessment (Rafiei & Adeli, 2018), bridge inspection (Ni et al., 2020; Sajedi & Liang, 2021) (X. Liang, 2019), condition classification of jointed plain concrete pavement (Hsieh et al., 2020), concrete and steel damage detection (Cha et al., 2017, 2018), concrete crack detection (S. Y. Kong et al., 2021), and road damage detection (Maeda et al., 2018). Distinctively, with the breakthrough of the pixel-wise fully convolutional networks (FCNs) for salient object detection, new efforts have been made for road surface inspection (Bang et al., 2019) and rail surface inspection (T. Wang, Zhang, et al., 2018), aiming to address the image inconsistency issue. Because the pixel-wise FCNs perform well with the pixel labeling issue, recent end-to-end salient object detection (SOD) networks (Hou et al., 2019; Jia & Bruce, 2019; G. Li & Yu, 2016b; Zhao et al., 2019) have shown improved performances, compared with the visual methods based on hand-crafted features (J. Liang et al., 2018; Lu & Lim, 2012; J. Zhang, Ehinger, et al., 2017; Q. Zhang et al., 2019) and the patch-wise deep methods (G. Li & Yu, 2016a; N. Liu et al., 2015; R. Zhao et al., 2015). These FCNs could highlight the object by the pixel-wise segmentation. However, most of the FCNs are still hard to generate clear salient object boundaries because of ignoring the structural information, such as texture or edge features (Hou et al., 2017, 2019; Jia & Bruce, 2019), which is essential to SOD (J.-X. Zhao, Liu, et al., 2019). To capture more fine structural information and accurate edge features, many coarse-to-fine FCN methods have been proposed to refine the networks such as, context-aware refinement network (Islam et al., 2017), R3Net (Deng et al., 2018), and DGTL (T. Wang, Zhang, et al., 2018). Although these methods did improve the detection results for salient object significantly, there is still room for critical improvements in terms of structural segmentation quality, object details, and particularly, boundary prediction.

Recently, many enhanced UNet (Ronneberger et al., 2015) models with encoder-decoder, such as UUNet (Qin et al., 2020) based on a two-level nested structure and UNet++ (Zhou et al., 2018) using a series of nested, dense skip pathways, are proposed to mine structural information more effectively and obtain fine-grained object details. Several UNet architecture-based SOD methods (N. Liu & Han, 2016; L. Zhang, Dai, et al., 2018; Zhang et al., 2017) employ a bidirectional or recursive way for refining the high-level information with the local features, in order to get the fine boundary predictions. The new PoolNet (J.-J. Liu et al., 2019) has a U-shape feature pyramid network (FPN) structure for SOD by introducing a global guidance module and a multi-scale feature aggregation module, but the complementary information between salient edge features and salient object features is still not explicitly utilized and modeled. To solve this problem, the EGNet (J.-X. Zhao, Liu, et al., 2019) was introduced to make full use of the shallow boundary information from the object to fuse high-level object location information. From a different study, BASNet (Qin et al., 2019), which consisted of an encoder-decoder structure with hybrid loss, was proposed to detect the salient object locations and generate the fine object edges. Despite other improvements, EGNet and BASNet may still be inaccurate for RS saliency detection, especially for images having shallow defects or various reflection levels of the rail surface. The sophisticated network parameters are just too difficult for practical railway field applications (see Section 3.3.2).

Clearly, the RS segmentation is the first and most critical step in terms of RSD detection, which will determine whether the following RSD detection algorithms are meaningful. Although a good method for RS segmentation is key, there are ineluctable challenges for RS segmentation using UAV images as follows:

- **Unstable camera angle and positions**. Since UAVs are easily affected by the airflow in open areas, the angle and height of the camera on UAVs facing the track would not be consistent. Consequently, the size, position, and angle of the same rail would not be the same for all the images. Thus, it is very challenging for the existing IP-based algorithms (Wu, et al., 2018; Wu, Qin, & Wang,

et al., 2018) to have consistent performance in terms of rail surface segmentation.

- **Large aspect ratio and complex edges**. By nature, the rail surface has a large aspect ratio, and the edge of the rail surface is very similar to the web and the edge of the rail base. Popular nonedge-aware saliency detection frameworks such as UNet, U2Net, and UNet++ may detect the rail surface with blur, incomplete, or even incorrect rail boundaries due to the lack of auxiliary rail edge guidance information (see Section 3). Hence, a good network should be able to model complementarity between the rail surface information and the rail edge features and sufficiently leverage the complementary information to accurately produce the rail surface with fine boundaries. This is also the stepping stone for effective surface defect detection.
- **Various sunlight intensity, shadows, and reflection of the rail surface**. The brightness of the track images could fluctuate a lot because of various sunlight intensities. From a different perspective, rail surface generally has high pixel level in images; however, the order of these pixel level is often disrupted by shadows created by both the surroundings along the track and the various reflectance properties of different rail segments. Therefore, it probably leads to inconsistency in saliency detection for different regions of the rail surface with the existing pixel-wise FCN-based methods.

Besides rail surface segmentation, other factors such as corrosion and rust could also challenge the IP-based image segmentation algorithms for defect detection, such as ME (Kapur et al., 1985), k-means clustering (Tatiraju & Mehta, 2008) and Otsu (Otsu, 1979). Also, the rail surface defects do not have consistent and easily distinguishable structural or shape characteristics, causing many deep learning-based object detection methods, which depend on mining sophisticated object textures, to be infeasible.

Compared with classic IP-based methods used to inspect vehicle images for RSD detection (He et al., 2016; Q. Li & Ren, 2012a, 2012b), this work proposes a framework for rail surface segmentation which addresses the inaccurate RS extraction issue caused by inconsistent angle, uneven illumination, and poor contrast between the rail surface and the track. Compared with the existing IP-based methods used UAV images for RSD detection (Wu, Qin & Jia, 2018; Wu, Qin, Wang, et al., 2018), this paper addresses the inaccurate RS segmentation issue caused by shadows, inconsistent angle, and sunlight. Similarly, compared with recent deep learning-based object detection methods (Wei et al., 2020; Yanan et al., 2018), this paper builds an FCN for precise RS segmentation combined with an IP-based model for efficient RSD detection by mining pixel relationship between RSDs and the background of the rail surface. Therefore, this method is able to outperform deep learning-based methods because the RSDs in images, especially in UAV images, do not have consistent and easily distinguishable structural or shape characteristics, thereby causing many deep learning-based object detection methods very unpopular in field applications.

In summary, IP-based methods often have difficulties to accurately segment rail surface from images taken by a UAV. The precise pixel-wise SOD network should obtain enough heuristic structure information and boundary information to better recognize the track areas and preserve rail boundaries with a reasonable processing speed. Recent studies have improved the performance of the existing FCNs for different detection capabilities. However, the accuracy on both rail boundary extraction and rail surface segmentation still needs improvements for field applications. A new approach that takes a breakthrough step in RS segmentation is critical and is precisely the motivation of this study. This study proposes an integrating system based on newly developed FCN and IP technologies that are tailored for accurate RS segmentation and RSD inspection. A novel FCN named RBGNet, which can fully utilize the complementarity between rail edge information and rail object information for accurate RS segmentation, is proposed. An IP-based model fusing both local Weber-like contrast and maximum entropy algorithm is developed and integrated into RBGNet for RSD detection. Figure 2 and Table 1 give the main procedures of the proposed innovative solution for RSD detection, and the main contributions are as follows:

- An innovative automatic rail surface inspection system is developed which can accurately segment rail surface and detect rail surface defects simultaneously. The developed system can handle images taken by difference platforms, such as railway cars, hand-held devices, and UAVs.
- The novel FCN, called RBGNet, is developed to model the complementary salient rail surface information and the rail edge features within the network for preserving rail boundaries in the images. The network architecture, which is based on a backbone introducing enhanced residual block (ERB), can optimize the complementary tasks between the rail surface and the rail edge by impelling them to mutually complement each other. This new architecture remarkably improves the accuracy of rail surface prediction. Note the developed RBGNet is trained from scratch, without loading any pretrained models trained in ImageNet data, yet maintains a high detection rate.
- A new hybrid loss function integrating binary cross entropy (BCE), SSIM, and IoU is developed and leveraged in RBGNet for supervising the training process

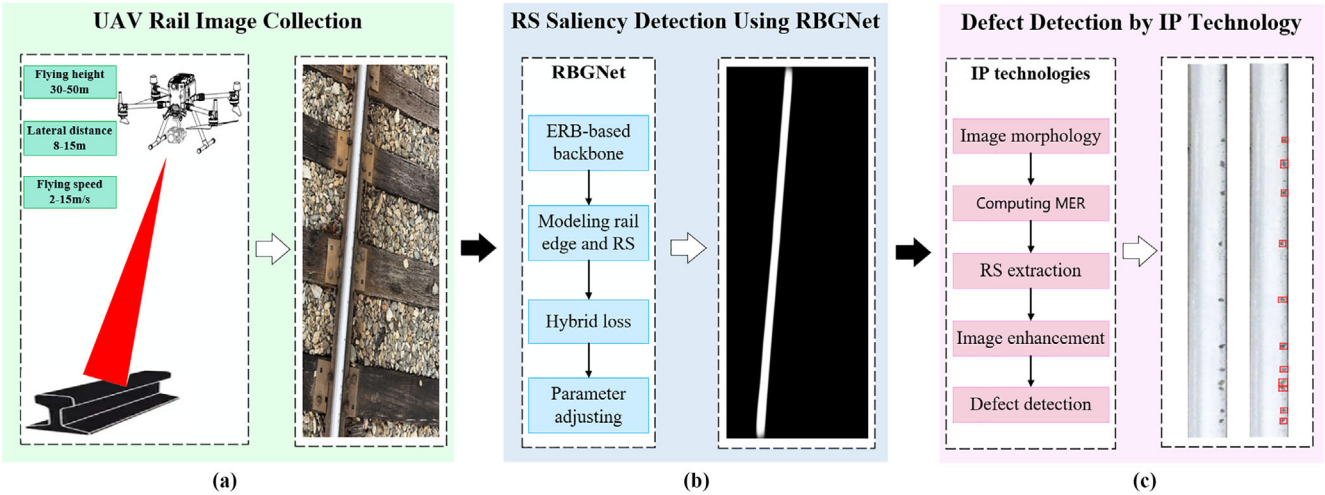

FIGURE 2 Overview of the proposed rail surface defect (RSD) inspection system. MER denotes minimum enclosing rectangle

TABLE 1 Main flow of the RS inspection system

Procedure
for
1: Obtaining the saliency map S of images by RBGNet.
2: Acquired optimized S using image morphology such as image closing operation, small aeras removal.
3: Extracting RS by computing the minimum enclosing rectangle of S.
4: Enhancing RS using LWLC.
5: Detecting defects on RS based on ME.
end for
return The RS inspection images
end Procedure

of salient RS and rail edge prediction under three levels: pixel-level, patch-level, and map-level, which further improve RS prediction precision.

- The quantitative and qualitative comparison experiments are conducted with a track dataset established with UAV images. The results show the proposed system achieves promising performance with good robustness in complicated environments.

The remaining sections are organized as follows: The development of the proposed system is elaborated in Section 2. Experiments are described in Section 3, and the conclusion is presented in the end.

2 | METHODOLOGY

This section introduces the proposed system that integrates the FCN-based RBGNet for RS segmentation and the LWLC-ME-based IP model for RSD detection, aiming to overcome the difficulties discussed earlier. Table 1 and Figure 2 give the procedures and the flow diagram that illustrates the organization of the entire inspection system, respectively.

2.1 | Rail boundary guidance network: RBGNet

Based on the literature, it is found that the object segmentation and locating tasks could benefit from the salient boundary detection results (J.-X. Zhao, Liu, et al., 2019), and SSIM loss succeeded in SOD detection (Qin et al., 2019) and image dehazing (C. Li et al., 2018; Wu et al., 2020) for considering the structure and spatial coherence of images. Therefore, inspired by those pioneer efforts, an end-to-end RBGNet with an architecture fusing the complementary salient rail edge information and salient RS information, together with a newly developed hybrid loss for RS saliency detection is developed. The RBGNet mainly consists of four modules and saliency supervision. The four key modules are backbone, including ERB, RE_SFE (rail edge saliency feature extraction), and RS_SFE (rail surface saliency feature extraction) and guidance, respectively, as shown in Figure 3.

2.2 | Backbone

Most existing visual detection approaches, such as image super-resolution (Zhang et al., 2018), object detection (Afif et al., 2020; Lin et al., 2017), and salient object detection (J.-X. Zhao, Liu, et al., 2019) (J.-J. Liu et al., 2019), simply utilize ResNet structure as backbone to solve time and memory optimization issues. However, directly

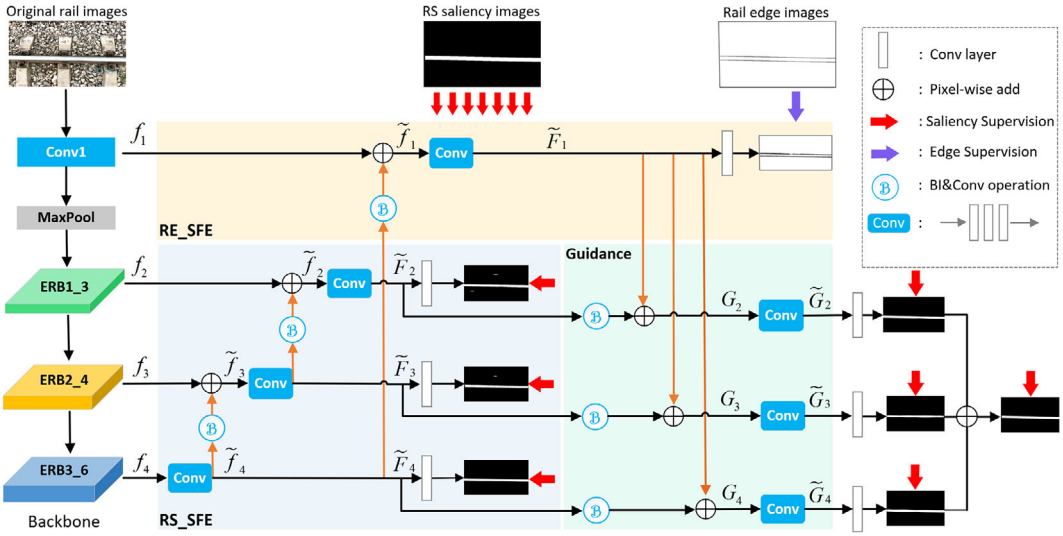

FIGURE 3 The overview of RBGNet. BI represents a bilinear interpolation operation. *Note*: The purple arrow denotes the nth ($n = 1$) side output loss in the nth ($n = 1$) side output path. Similarly, the red arrows represent the nth ($n = 2,3,4,5,6,7,8$) side output loss

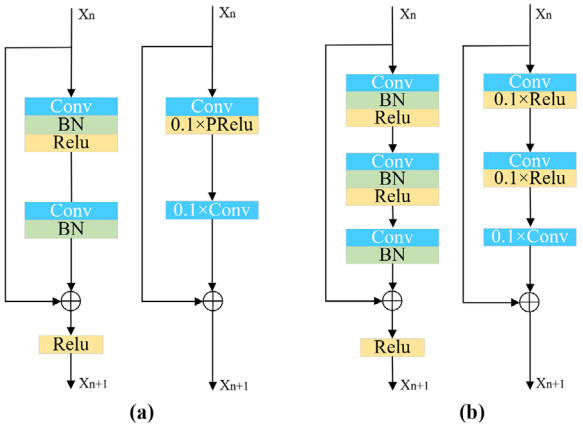

FIGURE 4 Residual block examples. (a) Left: original ResNet block. Right: improved ResNet block used in DPRNet. (b) Left: residual block used in EGNet. Right: Enhanced residual block

implementing the original ResNet into low-level vision jobs could cause suboptimal models due to ResNet being proposed to solve high-level computer vision jobs such as multi-object classification tasks (Wu et al., 2020). Some studies (S.-H. Gao et al., 2020) have put Res2net (S. Gao et al., 2019) as the backbone for salient object detection, however, it brings extra computation cost for adding the parallel branches. As shown in Figure 4a (right), Wu et al. (2020) successfully used improved ResNet without BN (batch normalization) layers to build a network named DPRNet for image haze removal. Besides, Lim et al. (2017) employed ResNet without BN layers in an image super-resolution task. Nah et al. (2017) removed the BN layers in their network for image deblurring of dynamic scenes. Therefore, motivated by these successes of BN removal, we propose using ERB, see Figure 4b (right), as the backbone of RBGNet with the following reasons:

- Network could be redundant and hard to converge due to overdose of complex operation functions such as BN, Conv and Relu (Wu et al., 2020). RS detection may not need a complex network.
- Original flexible structural information is likely be lost by using BN layers because BN would normalize these features (Lim et al., 2017).
- BN layers consume as much memory as the Conv layers do. Enlightened by Wu et al. (2020), a parameter 0.1 is employed in ERB instead of BN layers, see Figure 4b, which saves approximately 40% of storage requirement compared with ResNet in EGNet (J.-X. Zhao, Liu, et al., 2019).
- As the network structure is built based on ERB without loading any pre-trained backbones, such as visual geometry group (VGG) or ResNet50 adapted from other image classification works, the proposed RBGNet could have consistent performance regardless of the track environments without significant accuracy loss, which would be an important improvement from the previous models (J.-J. Liu et al., 2019; J.-X. Zhao, Liu, et al., 2019).

As shown in Figure 3, three ERBs are employed as the basic units of network backbone, simultaneously generating three side paths. Following a similar idea as EGNet (J.-X. Zhao, Liu, et al., 2019), the backbone is built without fully connected layers, which include a Conv layer for generating an additional side path and a Maxpool layer for parameters reduction. Therefore, four side features, Conv1, ERB1_3, ERB2_4, and ERB3_6 can be collected from

TABLE 2 The configuration of backbone network

Layer	Type	Filter size	Stride	Padding	Output channels
Conv1		7×7	3	3	64
Max pool		3×3	2	1	64
ERB1	Bottleneck	$\begin{Bmatrix} 1\times1\times64 \\ 3\times3\times64 \\ 1\times1\times256 \end{Bmatrix} \times 3$	1	1	64
			1		64
			1		256
ERB2	Bottleneck	$\begin{Bmatrix} 1\times1\times128 \\ 3\times3\times128 \\ 1\times1\times512 \end{Bmatrix} \times 4$	1	2	128
			1		128
			1		512
ERB3	Bottleneck	$\begin{Bmatrix} 1\times1\times256 \\ 3\times3\times256 \\ 1\times1\times1024 \end{Bmatrix} \times 6$	1	1	256
			1		256
			1		1024

backbone network, respectively. It is worth noting that because low-level features in the Conv1 layer could better preserve edge properties (Mahendran & Vedaldi, 2015; Zeiler & Fergus, 2014), Conv1 extracts the edge features, while other side paths obtain the salient RS features. The four original side features can be represented by a backbone feature set f, as shown in Figure 3. The configuration of backbone is summarized in Table 2.

$$f = \{f_1,\ f_2,\ f_3,\ f_4\} \tag{1}$$

2.3 | RS_SFE, RE_SFE and guidance modulus

RS_SFE in Figure 3 shows a specific network module used to generate multi-resolution features which is developed and implemented in RBGNet. A similar idea can be found in UNet (Ronneberger et al., 2015) and EGNet (J.-X Zhao, Liu, et al., 2019). Unlike the original UNet, Conv operators on side paths is added for acquiring more salient rail surface information, and a Relu layer is added after each Conv layer to ensure the nonlinearity of the model. In addition, in the U-shape architecture, when high-level context information is progressively returned to the shallow layer, the high-level location information is gradually diluted at the same time (J.-J. Liu et al., 2019). Hou et al. (2017) also reported the receptive field size of the high-level feature map is larger and the location is more accurate. Hence, a high-low location information propagation mechanism is leveraged to fuse the high-level information to each side path. Starting by setting the fused feature set as

$$\widetilde{f} = \{\widetilde{f}_1,\ \widetilde{f}_2,\ \widetilde{f}_3,\ \widetilde{f}_4\} \tag{2}$$

then each fused feature can be computed by

$$\begin{aligned} \widetilde{f}_k &= f_k + \Phi(\Gamma(Trans(\widetilde{f}_{k+1},\ \varepsilon(f_k))),\ \mu(f_k)), \\ k &= 2,\ 3\ \widetilde{f}_4 = \psi(f_4), \quad k = 4 \end{aligned} \tag{3}$$

where $Trans(input,\ \varepsilon(*))$ represents a Conv operation, which aims to change the number of input feature channels to $\varepsilon(*)$. $\varepsilon(*)$ denotes the number of $*$ feature channels. Γ represents a Relu function. $\Phi(input,\ \mu(*))$ is a bilinear interpolation operation, which is used to reshape the input to $\mu(*)$. $\mu(*)$ denotes the size of $*$. $\psi(\cdot)$ represents a series of Conv and nonlinear operations. Note that f is the original feature set from the backbone, and $\widetilde{f}$ denotes the improved feature set.

Finally, Conv operations can be employed to enhance salient RS features, thus the enhanced RS features from each side path can be defined as

$$\begin{aligned} \widetilde{F}_k &= \psi(\widetilde{f}_k), \quad k = 2,\ 3 \\ \widetilde{F}_4 &= \psi(f_4), \quad k = 4 \end{aligned} \tag{4}$$

RE_SFE module in Figure 3 is used to model salient edge information, but it is not enough with the local information only. As discussed earlier, the receptive field of top-level feature maps is larger, and the location is more accurate. Thus, it is meaningful to add the high-level RS context information to edge features f_1 for suppressing the nonsalient edge. Similarly, the fine enhanced salient rail edge features are represented as

$$\begin{aligned} \widetilde{F_1} &= \psi(f_1 + \Phi(\Gamma(Trans(\widetilde{f}_4,\ \varepsilon(f_1))),\ \mu(f_1))), \\ k &= 1 \end{aligned} \tag{5}$$

thereby the enhanced feather set consisting of salient rail edge features and salient RS features are collected as

$$\widetilde{F} = \{\widetilde{F}_1,\ \widetilde{F}_2,\ \widetilde{F}_3,\ \widetilde{F}_4\} \tag{6}$$

Guidance module sketched in Figure 3 shows the rail edge features are utilized to guide the salient RS in terms of both location and segmentation after collecting the complementary edge and salient RS features. For the original

TABLE 3 Details of each side output path

SO	U1			U2			U3		
1	128	3	1	128	3	1	128	3	1
2	256	3	1	256	3	1	256	3	1
3	512	5	2	512	5	2	512	5	2
4	512	5	2	512	5	2	512	5	2
5	128	3	1	128	3	1	128	3	1
6	128	5	2	128	5	2	128	5	2
7	128	5	2	128	5	2	128	5	2

U-shape architecture, the simple way is that high-level rail object features are progressively transmitted to the low-level (rail edge extraction), while the high-level location information is gradually diluted at the same time. The proposed structure of RBGNet aims at fusing salient rail edge features and salient RS features of each side path with the guidance of the rail edge information and getting more accurate predictions. Thus, the guidance module is built and implemented in RBGNet to integrate the refined rail edge features into salient RS features of each side path. Note rail surface location predictions of each side path and the segmented rail surface details can be abundant and high-level predictions. It could be more accurate by adding the salient rail edge information. The fused salient RS guidance feature set can be defined by

$$G = \{G_2, \ G_3, \ G_4\} \tag{7}$$

then each fused guidance feature can be represented as the following:

$$\begin{aligned} G_k &= \Phi(\Gamma(Trans(\widetilde{F}_k, \ \varepsilon(\widetilde{F}_1))), \ \mu(\widetilde{F}_1)) + \widetilde{F}_1, \\ k \in [2,4] \ \widetilde{G}_k &= \psi(G_k) \end{aligned} \tag{8}$$

where $Trans(\cdot)$, $\Gamma(\cdot)$, $\Phi(\cdot)$ and $\psi(\cdot)$ represent a Conv operation, a Relu function, a bilinear interpolation operation, and a series of Conv and nonlinear operations, respectively. $\psi(\cdot)$ represents details of each side output path, as shown in Table 3. Note that SO denotes the side output path (including Conv) shown in Figure 3. While, U denotes the Conv shown in Figure 3, consisting of three convolutional layers: U1, U2, and U3, followed by three Relu layers. The channel number of each convolutional layer, kernel size, and padding are given. For example, "128, 3, 1" denotes a convolutional layer of which channel number is 128, kernel size is 3, and padding is 1. Hence, an enhanced rail surface object set of edge feature guidance can be collected by

$$\widetilde{G} = \{\widetilde{G}_2, \ \widetilde{G}_3, \ \widetilde{G}_4\} \tag{9}$$

2.4 | Supervision: Hybrid loss

BCE (De Boer et al., 2005) is popular as the training loss function in most of the salient object detection methods. However, models only trained with cross entropy (CE) loss usually have poor performance in distinguishing boundary pixels (Qin et al., 2019). Other popular losses, such as IoU (Nagendar et al., 2018), can supervise the predicted results in the global scope yet lack the capability of capturing a fine structure. Contrarily, SSIM loss succeeded in recent visual inspection tasks (C. Li et al., 2018; Wu et al., 2020) by considering the structure and spatial coherence of images. Therefore, to make the best use of different popular losses, a tailored hybrid loss consisting of BCE, SSIM, and IOU is proposed.

Salient rail edge and all the salient RS supervisions are based on the proposed hybrid training loss, and the total loss can be denoted as the summation over all the outputs:

$$\begin{aligned} Loss &= \sum_{n=1}^{N} l^{(n)}, \quad N = 8 \\ l^{(n)} &= l^n_{BCE} + l^n_{SSIM} + l^n_{IoU} \end{aligned} \tag{10}$$

where $l^{(n)}$ is the nth side output loss and N represents the total number of the output (if $n = 1$, $l^{(n)}$ is salient rail edge supervision, as shown by the purple arrow in Figure 3; else if n = 2,3,4,5,6,7,8, $l^{(n)}$ is salient rail surface supervision, as shown by the red arrows in Figure 3). l^n_{BCE}, l^n_{SSIM} and l^n_{IoU} are BCE, SSIM, and IoU loss function, respectively. As shown in Figure 3, the salient rail edge and RS model is supervised deeply by eight outputs, thus $N = 8$, composed of one output from RE_SFE module, three outputs from RS_SFE module, and four outputs from the final supervision.

BCE is one of the most common loss functions in binary classification and is defined as

$$\begin{aligned} l_{BCE} &= -\sum_{(i,j)} [GT(i,j)\log(S(i,j)) \\ &\quad + (1 - GT(i,j))\log(1 - S(i,j))] \\ S &\in \{\varphi(\widetilde{F}), \ \varphi(\widetilde{G}), \ \varphi(Sum(\widetilde{G}))\}, \\ GT &\in \{Edge, \ Object\} \end{aligned} \tag{11}$$

where $GT(i,j) \in \{0,1\}$ denotes the ground truth label of the pixel (i,j), and $S(i,j)$ represents the predicted probability of being salient RS object or salient rail edge. $\varphi(\cdot)$ is a transition Conv layer with kernel size of 3, padding of 1, and channel number of 1 and is used to convert multi-channel input features to a one-channel activation map. As introduced earlier, $\widetilde{F}$ is the enhanced rail edge features and RS features set. $\widetilde{G}$ is the enhanced RS feature set under edge

feature guidance. *Edge* and *Object* are salient rail edge ground truth and salient RS ground truth, respectively.

SSIM index can capture the structural information from an image and was originally proposed to assess image quality (Z. Wang et al., 2003). Several computer visual tasks, such as (C. Li et al., 2018; Wu et al., 2020), successfully applied SSIM to training loss and got promising results. Thus, SSIM was supplemented in the proposed hybrid loss function to refine the structural information from the RS images. For two-pixel value sets of two corresponding patches x and y (size: $N \times N$), which are cropped from the predicted probability map $S(i, j)$ and the binary ground truth $GT(i, j)$, respectively, the two sets can be defined as

$$\begin{aligned} &x = \{x_{i,j} : \ i, j = 1, \dots, N\}, \ y = \{y_{i,j} : \ i, j = 1, \dots, N\} \\ &x \subseteq S, \ y \subseteq GT \\ &S \in \{\varphi(\widetilde{F}), \varphi(\widetilde{G}), \varphi(Sum(\widetilde{G}))\}, \ GT \in \{Edge, Object\} \end{aligned} \tag{12}$$

and then the SSIM of x and y is given by

$$l_{SSIM} = 1 - \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{13}$$

where μ_x, μ_y and σ_x, σ_y are the mean and standard deviations of x and y, respectively. σ_{xy} is covariance. C_1 and C_2 are set to 0.01^2 and 0.03^2, respectively, to prevent singularity caused by dividing by zero.

IoU was originally proposed to measure the similarity of any two sets; however, it was quickly adopted as a standard parameter for object segmentation and detection evaluation. Recently, many state-of-the-art methods of image segmentation have begun to employ IoU as training loss (Qin et al., 2019). Thereby, an IoU loss function is also provided into RBGNet training. The IoU loss is expressed by

$$\begin{aligned} &l_{IoU} = 1 - \frac{\sum_{i=1}^{H}\sum_{j=1}^{W} S(i,j)\, GT(i,j)}{\sum_{i=1}^{H}\sum_{j=1}^{W} [S(i,j) + GT(i,j) - S(i,j)GT(i,j)]} \\ &S \in \{\varphi(\widetilde{F}), \varphi(\widetilde{G}), \varphi(Sum(\widetilde{G}))\}, \ GT \in \{Edge, Object\} \end{aligned} \tag{14}$$

where $GT(i, j) \in \{0, 1\}$ is the ground truth label of the pixel (i, j), and $S(i, j)$ represents the predicted probability of being salient RS object or salient rail edge.

2.5 | RSD detection model: LWLC and ME

After extracting the rail surface, rail surface defect detection is performed by an IP-based model consisting of the LWLC for image enhancement and the ME.

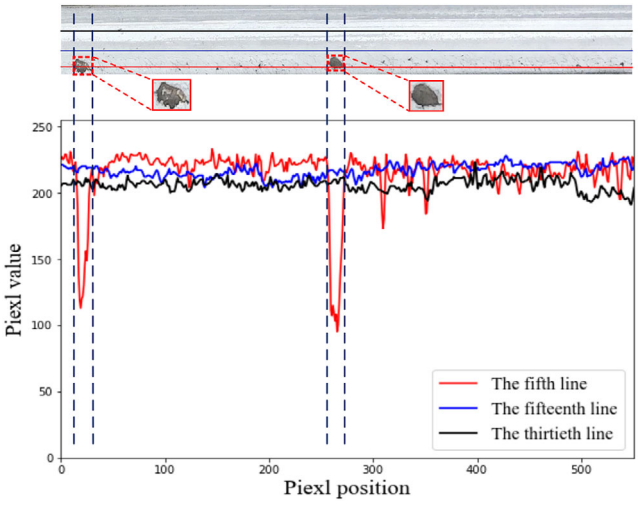

FIGURE 5 Example of rail surface defect (RSD) induced gray level pixel value drops along the rail longitudinal direction

2.6 | LWLC algorithm for RS image enhancement

RS images have following characteristics: (1) The pixel values of RS do not vary much, especially when the illumination is very consistent; (2) Generally, the pixel values of RS are higher compared to other objects or to the background in an image because of the high reflectivity of the smooth rail surface; and (3) The appearances of the track are almost identical along the track's longitudinal direction (Wu, et al., 2018). Figure 5 gives an example of the pixel value variation along the RS longitudinal direction. Note the RS is divided into pixel-level lines to check the value variation. Figure 5 shows the pixel gray mean values are generally high, but the defects would cause distinguishable drops in brightness. In general, the number of defects in an RS image is relatively small, so it is reasonable to use the gray mean value along a longitudinal line as the background. The nature of RS and RS defects well satisfies the prerequisite of Weber contrast (Whittle, 1994). Thus, the LWLC algorithm is employed to enhance the UAV RS images, which could handle the dramatic changes of gray scale values for RSD detection.

Let a gray set $T = \{I(i, j) : \ i = 1, \dots, N; \ j = 1, \dots, M\}$ which is a sliding window cropped from an RS image, and the *LWLC* of each pixel in T is given as

$$LWLC(i, j) = \frac{I(i, j) - E(T)}{E(T)} \tag{15}$$

where $I(i, j)$ is pixel gray value and E denotes the mean gray value of the pixels within window T.

A transformed *LWLC* image matrix with contrast enhancement can be obtained by Equation (15). The size

selection of the window T is critical. Based on experience, a size of $h \times 1$ (h is height of the RS image) is proposed. Within the window T, the pixel value of the RSD is lower than the surrounding region because the light reflected by the defect is less, assuming the ambient light intensity within the window T is constant. Thus, a pixel can be classified as a defect point if its gray scale value is lower than the mean gray value of T; otherwise, it will be treated as a background pixel. These nondefect pixel points within T are blended into the background by a dynamic threshold $E(T)$ defined as

$$LWLC(i,j) = \begin{cases} \frac{I(i,j)-E(T)}{E(T)}, & \text{if} \quad I(i,j) < E(T) \\ 0, & \text{otherwise} \end{cases}. \quad (16)$$

The function of *LWLC* can be summarized as 1) calculating *LWLC* (*i, j*) for each pixel in an image by Equation (16) for obtaining *LWLC* matrix of the image, and 2) mapping gray level of the *LWLC* matrix to [0, 255]. Other details regarding the *LWLC* are provided elsewhere (Wu, et al., 2018).

2.7 | ME for defect segmentation

The entropy threshold (ME) method is proposed to generate a segmentation threshold by computing a gray histogram entropy of an image. The ME (Kapur et al., 1985) can confirm the threshold by maximizing the total informational content between object probability distribution γ_O and background probability distribution γ_B, defined as

$$\gamma_O = \sum_{n=0}^{T-1} p_n,\ \gamma_B = 1 - \gamma_O;\ P_n = \frac{f_n}{K},$$
$$n \in [0,\ 255] \quad (17)$$

where P_n, f_n, and k denotes the probability of pixel value n, the frequency of gray value n, and the total pixel number in an RS image, respectively. Let an RS image I, and then the entropy H of γ_O and γ_B can be defined by

$$\begin{aligned} H_O(T) &= -\sum_{n=0}^{T-1} \frac{P_n}{\gamma_O(T)} \ln \frac{P_n}{\gamma_O(T)} \\ H_B(T) &= -\sum_{n=T}^{255} \frac{P_n}{\gamma_B(T)} \ln \frac{P_n}{\gamma_B(T)} \end{aligned} \quad (18)$$

then, an optimal threshold can be obtained by

$$T^* = \arg\max\left(H_O(T) + H_B(T)\right), \qquad T \in [0,\ 255] \quad (19)$$

The ME method can acquire an optimum threshold value for the sake of RSD segmentation after the previous two steps (RS extraction and RS enhancement). The ME method considers both the pixel level distribution and the spatial information of the pixels in an RS image.

3 | EXPERMENTS

In this section, the ablation study on RBGNet and extensive comparative experiments (including both quantitative and qualitative evaluations) for RS and RSD detection are conducted to check the performance of the proposed RBGNet and LWLC-ME approach with a track dataset of images taken by UAVs.

3.1 | Image acquisition and experimental environment

The track images are taken by the Zenmuse H20 aerial camera, which is mounted on a DJI Matrice RTK 300 Drone. The Zenmuse H20 camera can take images having a focal length between 6.83 mm and 119.94 mm and a video resolution of 3840 × 2160 or 1920 × 1080 @ 30fps. The UAV is controlled to fly about 30–50 m above the track with a cruise speed within 2–3 m/s. Track images are acquired in a section of Beijing-Shanghai high-speed passenger line near Langfang, China and a section of a Class I freight line in Columbia, South Carolina, USA. Therefore, the rail image dataset in this study covers tracks with different structures, rail sections, and a variety of track components. Figure 6c gives some examples out of the total 600 images. The training dataset has 540 randomly selected images, the remaining 60 images are used for testing. Note the training set and the test set for RBGNet are completely different. The training images are resized to 1280 × 720. The cross-validation method is used for the training strategy, and an optimal model can be chosen from several models trained based on different hyper-parameters and randomly selected training data and testing data with a ratio of 9:1. The learning rate, weight decay, and momentum in RBGNet are set to 5×10^{-5}, 0.0005, and 0.9, respectively. The network is implemented in PyTorch. The model is trained by 20 epochs, and the initial learning rate is divided by 10 after 15 epochs. Similar to the hybrid loss supervision of DPRnet for image dehazing, it is assumed SSIM and other loss items in the proposed hybrid loss have equal contributions for supervision between the produced saliency images and ground truth. In addition, following the pioneer works using the hybrid loss in BASnet and UUnet for saliency detection, the weight of each term in the proposed hybrid loss and of each side loss are set to 1. All the prediction maps are fused in the last output to obtain the fine salient predicted RS map. The Sobel operator is utilized to detect rail edge for each track saliency image.

FIGURE 6 Experimental environment and extensive examples of various samples from UAV railway track dataset

3.2 | Evaluation matrix

3.2.1 | RS evaluation

Three metrics are leveraged to quantitatively compare the proposed method with the classic methods: precision-recall (PR) curves, F-measure (F_β), and mean absolute error (*MAE*).

PR curve is a common evaluation reference for computer vision models. For a given predicted saliency probability map S and the corresponding ground truth G, S can be converted to a binary mask B using a segmentation threshold. The corresponding precision and recall can be computed by $|B \cap G|/|B|$ and $|B \cap G|/|G|$, respectively, where $\cap$ and $|\cdot|$ denotes a logic and operation and accumulates the non-zero entries in a mask. Each threshold can generate a pair of averaged PR over all the predicted saliency probability maps in a dataset, and then the PR curve can be plotted by varying the threshold from 0 to 1 to yield a sequence of PR pairs.

F-measure is defined by

$$F_\beta = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recal} \quad (20)$$

where β^2 is set to 0.3 so the mean F_β is similar as the literature (Deng et al., 2018; Hou et al., 2017).

MAE represents the average difference between each pixel by comparing the salient prediction map and the ground truth and is expressed by

$$MAE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} |P(i, j) - G(i, j)| \quad (21)$$

where P and G denote the probability map of the saliency detection and the corresponding ground truth, respectively. (H, W) represents the (*height*, *width*) and (i, j) is the pixel coordinates.

A larger *F-measure* and a smaller *MAE* shall be obtained for a better saliency detector (Hou et al., 2017). Thus, *F-measure* and *MAE* are also computed for all the state-of-the-art models for comparison.

3.3 | RSD detection and the entire system evaluation criteria

The entire RBGNet_LWLC+ME system is verified on an RSD dataset which includes 50 images out of the same 600 images in this study. However, the 50 images here are selected to ensure each image has at least two rail surface defects. The 50 images contain 188 RSDs in total. Note the 50 images used for RSD detection and the whole system evaluation are not the 540 images used for training the RBGnet. Based on our earlier efforts with UAV-based RSD detection, a defect will be labeled only if the defect area is more than 25 mm^2. The proposed system is implemented with Python, and a detected defect is labeled automatically by a red dotted rectangle. A predicted defect is considered correct if its red rectangle overlaps the corresponding ground truth over 80%; otherwise, it is considered incorrect. The ratio between the correctly identified defects and the total number of defects can be calculated to compare the performance of different models. Like the PR curve, the precision (P) and recall (R) are defined as

$$\begin{aligned} P &= TP/(TP + FP) \\ R &= TP/NP \end{aligned} \quad (22)$$

where *TP*, *FP*, and *NP* denote the number of correctly detected defects, the number of wrongly detected defects, and the number of labeled defects, respectively.

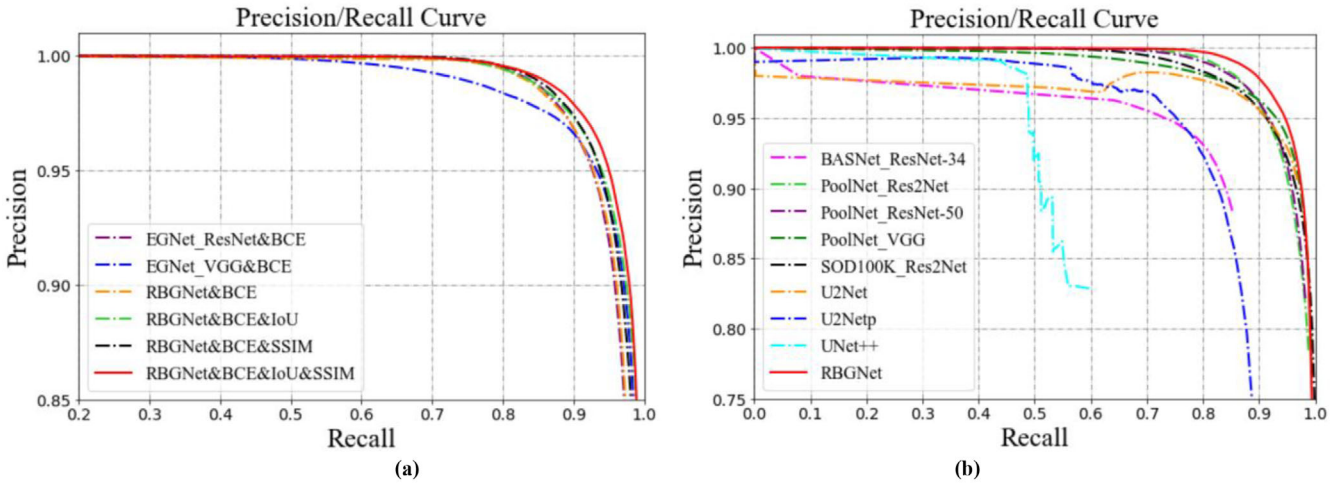

FIGURE 7 Precision–recall curves. (a) PR curves of EGNet (J.-X, Zhao, Liu, et al., 2019) and RBGNet with different loss. (b) PR curves of RBGNet and other typical state-of-the-art methods on rail surface (RS) dataset

3.4 | Performance evaluation

To evaluate the new architecture and the hybrid loss function in RBGNet, ablation study is conducted. The PR curve, *F-measure*, and *MAE* are calculated for quantitatively compare RBGNet and the state-of-the-art models, including BASNet, EGNet, UNet, PoolNet, SOD100K, U2Net, U2Netp, Unet++, R3Net, DSS, DHS, and DGRL. Visual evidence is also provided for a quick qualitative evaluation of RSD detection, followed by whole system quantitative comparison by precision (P) and recall (R).

3.5 | Ablation study on RBGNet

Extensive experiments are conducted to compare RBGNet and other state-of-the-art models, especially EGNet (J.-X. Zhao, Liu, et al., 2019). RBGNet may appear to be like EGNet at a glance; however, unlike EGNet, RBGNet has a notably distinct architecture using both high-low level and low-high level structures, a novel hybrid loss function, and a backbone including ERB without loading pre-trained models. A series of experiments with various losses based on the same RBGNet structure are also executed to validate the superiority of the new hybrid loss function.

Architecture ablation: Figure 7a compares the performance between different architectures and the backbones of EGNet and RBGNet. EGNet with ResNet or VGG are compared with RBGNet with different backbones, respectively. From the PR values shown in Figure 7a, it can be found that the proposed RBGNet equipped with a backbone including ERB and free of pre-trained models, outperforms EGNet, indicating that the effectiveness of the proposed RBGNet structure. Note that all the models use the same BCE loss function for fair comparison.

Loss ablation: Figure 7a also shows RBGNet with the developed hybrid loss, BCE + IoU + SSIM, achieves better PR than RBGNet with other loss functions, BCE + IoU, BCE + SSIM, or BCE only. Therefore, the proposed hybrid loss function does help RBGNet to achieve better performance.

Entire RBGNet: Figure 7b shows the comparison between the proposed RBGNet and other models. For existing methods, PoolNet and SOD100K perform better than those networks loaded with pretrained models including U2Netp, U2Net, UNet++, and BASNet equipped with ResNet-34 backbone. One possible reason as to why PoolNet and SOD100 outperform other existing models is that both of them use advanced network architecture, flexible convolution model, and pre-trained models (ResNet, VGG, or Res2Net) trained on ImageNet dataset. However, as indicated by the red solid line in Figure 7b, the proposed RBGNet with a novel network architecture without loading any pre-trained model achieves the best performance in term of precision.

3.6 | Comparison between RBGNet and state-of-the-art models

Quantitative comparison: To further verify the performance of RBGNet, *F-measure*, *MAE*, and the model parameter are calculated for different models. As shown in

TABLE 4 Quantitative comparison between RBGNet and the state-of-the-art models

Model	Backbone	F-measure	MAE	Parameter (M)
EGNet & BCE	VGGNet	0.929	0.031	108.07
EGNet & BCE	RestNet-50	0.919	0.030	111.69
BASNet (Qin et al., 2019)	ResNet-34	0.903	0.033	87.06
PoolNet (J.-J. Liu et al., 2019)	Res2Net	0.928	0.022	70.45
PoolNet	ResNet-50	0.937	0.020	68.26
PoolNet	VGGNet	0.942	0.020	52.51
SOD100K (Gao et al., 2020)	Res2Net	0.945	0.019	36.53
U2Net (Qin et al., 2020)	–	0.937	0.020	44.01
R3Net (Deng et al., 2018)	ResNeXt	0.939	0.027	56.16
DSS (Hou et al., 2019)	VGGNet	0.909	0.029	62.23
DHS (N. Liu & Han, 2016)	VGGNet	0.902	0.033	93.76
DGRL (T. Wang, Zhang, et al., 2018)	ResNet-50	0.922	0.030	161.74
RBGNet & BCE	ERB	0.925	0.031	54.64
RBGNet & BCE & IOU	ERB	0.947	0.019	54.64
RBGNet & BCE & SSIM	ERB	0.948	0.019	54.64
RBGNet & BCE & IoU & SSIM	ERB	0.967	0.013	54.64

Table 4, RBGNet performs the best with the track dataset in this study in terms of both *F-measure* and *MAE*. SOD100K gets the second place after RBGNet. Similar to the results shown in Figure 7b, PoolNet and SOD100K perform better than many other models, such as BASNet, DSS, DHS, and DGRL. The original PoolNet with the VGG backbone obtained better results than PoolNet with other backbones. In terms of model parameter, RBGNet is behind SOD100K, U2Net, and PoolNet with VGG backbone, but it is better than all the rest models. It is worth noting that the proposed RBGNet not only achieves better performances than EGNet, but also comes with an approximately 50% model parameter reduction. This further confirms the RBGNet is quite different from EGNet and vastly superior.

Qualitative comparisons: Figure 8 provides the visual evidence for qualitative comparison between RBGNet and other models. As shown in Figure 8, most FCN-methods cannot generate accurate RS which makes RSD detection impractical. RBGNet is able to accurately segment RS having complex edges, while other models either miss a considerable amount of the rail surface or yield blurry edges.

This is because RBGNet can handle the challenge of various reflectance properties in different regions and the inconsistent illumination conditions. As suggested by column (d), (g), (h), and (i) in Figure 8, UNet and these networks improved from UNet, such as U2Net, U2Net, and UNet++, and predict wrong RS or incomplete boundaries. Many edge-aware networks, such EGNet and BASNet struggle with RS images having complex boundaries or uneven illumination conditions, as indicated by column (b) and (c). Although the BASNet is equipped with an advanced loss function, its performance is not even close to EGNet. When comparing column (e) and (f), PoolNet and SOD100K with the advanced backbone, Res2Net, can achieve competitive performance. However, the U-shape network, PoolNet, struggles with dark parts along the rail surface, and SOD100K fails to complete the entire rail surface and boundaries. On the contrary, considering the limitations of the existing networks and benefit from modeling complementary salient RS information and salient rail edge features, the proposed RBGNet equipped with a novel architecture and a hybrid loss is able to accurately predict rail surface with well-defined boundaries.

3.7 | Comparison of LWLC+ME and other classic methods

Figure 9 provides a visual comparison between LWLC + ME and other classic methods for RSD detection. Note (a) shows RS extracted by RBGNet and the ground truth of RSD. First, as shown in row (b) and (d), k-means (KM) clustering and Otsu methods produce a lot of noise and irregular points. One possible reason is that they do not consider both the spatial information of pixels and the pixel-level distribution information in an image. Second, from row (c), although LN + ME filters out several irregular points, it also eliminates many defect points, leading to the inaccurate defect segmentation results. Third,

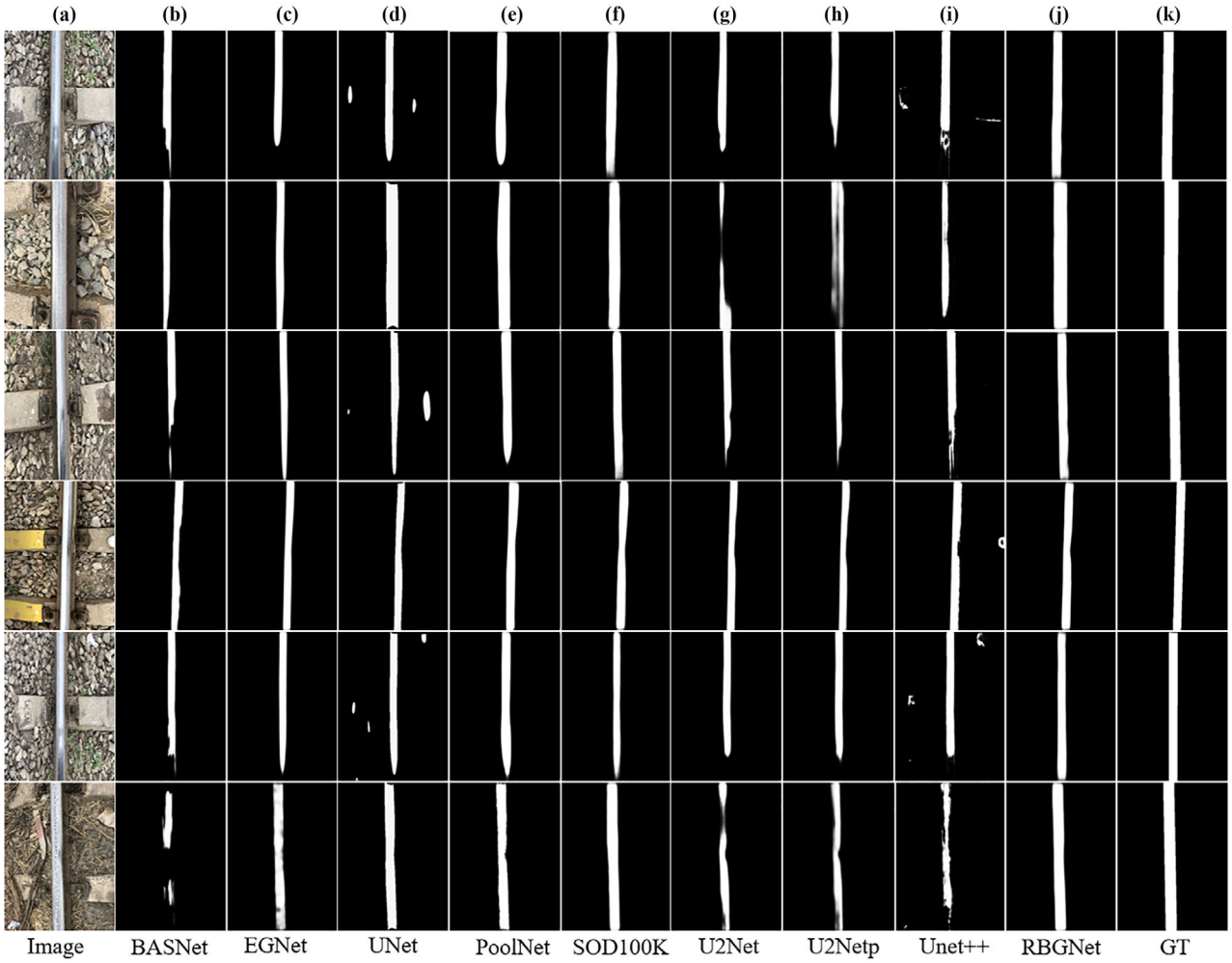

FIGURE 8 The qualitative comparisons with the state-of-the-arts. (a) Original image, (b) BASNet (Qin et al., 2019), (c) EGNet, (d) UNet (Ronneberger et al., 2015), (e) PoolNet, (f) SOD100K, (g) U2Net, (h) U2Netp, (i) Unet++ (Zhou et al., 2018), (j) RBGNet, and (k) Ground truth

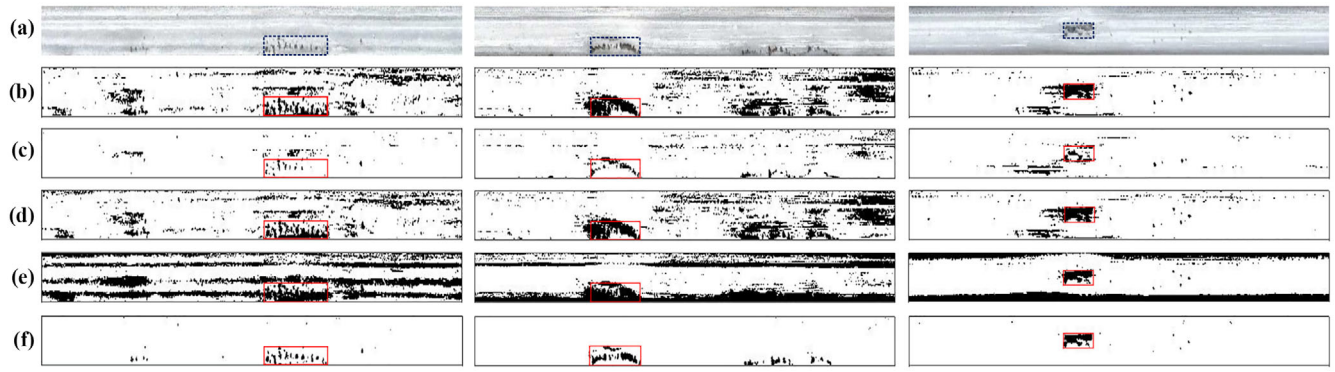

FIGURE 9 Examples of defect segmentation results by different classic methods. (a) Detected rail surface (RS) images by RBGNet and the blue rectangles denote ground truth of defects, (b) K-means (KM) clustering (Tatiraju & Mehta, 2008), (c) LN (Q. Li & Ren, 2012b) + ME, (d) Otsu (Otsu, 1979), (e) ME (Wu, Qin, & Jia, 2018), and (f) LWLC + ME. Note that these red rectangles are manually labeled

as shown in row (e), the ME method produces a plethora of false detections. Finally, as presented in row (f), the LWLC + ME method performs the best, which not only preserves the original defects features but also eliminates most of the noise.

Figure 10 presents the results from different fully automated RSD detection systems by integrating RBGNet with the earlier mentioned IP algorithms. Similarly, the system integrated by RBGNet and LWLC + ME model has the best performance.

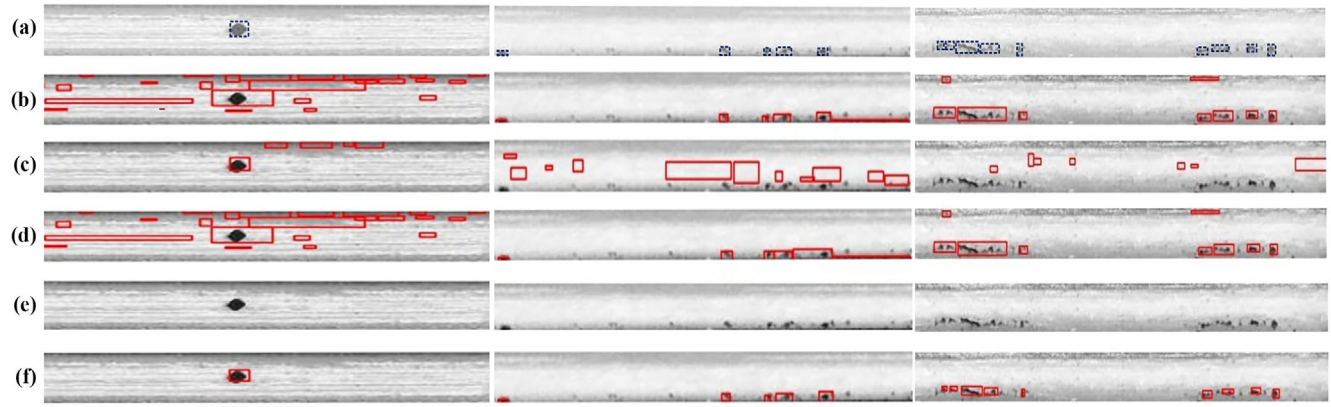

FIGURE 10 Examples of RS defect detection using different classic methods by extracting RS by RBGNet. (a) Detected RS images by RBGNet and the blue rectangles denote ground truth, (b) K-means (KM) clustering (Tatiraju & Mehta, 2008), (c) LN (Q. Li & Ren, 2012b) + ME, (d) Otsu (Otsu, 1979), (e) Maximum entropy (ME) (Wu, Qin & Jia, 2018), and (f) LWLC + ME. Note that these red rectangles are automatically labeled by our Python program

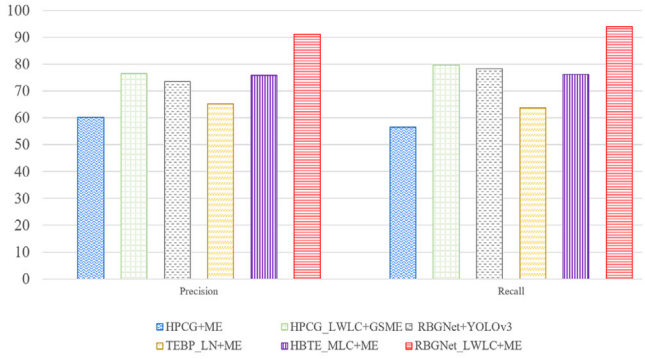

FIGURE 11 Precision (*P*) and recall (*R*) comparisons of ours and the classic methods

3.8 | Whole system evaluation

Finally, performance comparisons between different systems are conducted with the same track image dataset. Having accurate RS is the prerequisite for RSD detection, and RBGNet facilitates the following RSD detection work. As shown in Figure 11 and Table 5, HPCG + ME (Wu, Qin, & Jia, 2018) achieves the lowest *P and R* (precision, recall) and has the slowest processing speed because there are no image enhancement operations. HPCG_LWLC + GSME with image enhancement (Wu, Qin, Wang, et al., 2018) is able to reach a higher *PR* but suffers from slow speed, primarily because GSME reduces image noise in the frequency domain and the gray level is stretched based on wavelet transform. Although TEBG_LN + ME (Q. Li & Ren, 2012b) and HBTE_MLC + ME (Q. Li & Ren, 2012a) have better processing speeds, both have low *PR* values. The dominant reason is that TEBG and HBTE are not able to detect RS because both methods cannot handle the variations of track width and images angles. The original YOLOv3 method was applied to detect RS defect based on images taken from the inspection vehicles (Yanan et al., 2018). Thus, RBGNet is also integrated with YOLOv3 detector for comparison with the rail track images taken by UAVs. This system has a *PR* less than 80%, due to the limited capability to acquire sufficient structural and shape information of RSD. Thus, most FCN-based inspection methods depending on mining sophisticated object textures are impractical for RSD detection. Furthermore, different from the track component detection (Guo et al., 2021), the limited available RSD data cannot provide sufficient training data for these FCN-based methods. Figure 11 clearly shows the proposed system, RBGNet_LWLC + ME outperforms all the state-of-the-art systems yet achieves a reasonable processing speed. Note the successful RS

TABLE 5 Running time of systems

System	Stage1	Time	Stage2	Time	Stage 3	Time	Total (s)
HPCG + ME	HPCG	0.17	–	–	ME	14.01	14.18
HPCG_LWLC + GSME	HPCG	0.17	LWLC	0.46	GSME	4.62	5.25
RBGNet + YOLOv3	RBGNet	0.16	–	–	YOLOv3	0.18	0.34
TEBG_LN + ME	TEBG	0.06	LN	0.65	ME	3.06	3.77
HBTE_MLC + ME	HBTE	0.06	MLC	0.47	ME	2.93	3.46
Proposed system	RBGNet	0.16	LWLC	0.37	ME	2.31	2.84

recognition using RBGNet would lower the workload for the RSD detection. For example, as shown in Table 5, compared to HPCG_LWLC + GSME, its LWLC in stage 2 is 0.1 s slower than ours, this largely due to HPCG generating inaccurate RS with considerable falsely predicted regions, thereby causing the failure of the later enhancement and RSD detection tasks. Again, successful RS segmentation is the stepping stone for the success of the whole system.

4 | CONCLUSION

To well segment RS and perform RSD detection from images taken from rail vehicles, hand-held devices, and especially UAVs, is very challenging. This paper proposes an improved RSD inspection system. For RS recognition, a novel FCN type RBGNet is intended to model complementary salient RS information and rail edge features within this network for preserving RS boundaries in images. In addition, this architecture is developed without loading any pretrained models trained by ImageNet data and can optimize the complementary tasks between rail tack object and rail edge by impelling them to mutually help each other. Thus, this architecture remarkably improves the accuracy of RS prediction. Also, a new hybrid loss fusing BCE, SSIM, and IoU is leveraged in RBGNet to supervise the training of salient RS and rail edge prediction in the whole network from three levels: pixel-level, patch-level, and map-level. Finally, the quantitative and qualitative experiments conducted on a UAV rail track dataset indicate RBGNet has promising performance, and our proposed system, RBGNet_LWLC + ME, is able to detect RSD efficiently yet remaining at a reasonable processing speed, considering the ground speed of the UAV can reach 23 m/s, corresponding to a sample collection speed of 82.8 km/h. The developed RBGNet_LWLC + ME system would help to explore the great potential of UAV applications in RSD inspection.

The detection performance depends on the image quality. To have better detection results on small or tiny surface defects, cameras with high resolutions should be used. Note that UAVs for rail defect detection or other applications need to follow corresponding regulations for UAV operation. There are certain zones along the railroad tracks that UAVs are not allowed or need to be operated under restrictions.

For future research, we will devote our effort to develop edge computing capability. Mobile computing units with a customized software package having low cost, low computational complexity, low RAM consumption, and high precision are under development. The mobile computing units are potentially to be installed on a hi-rail vehicle, autonomous inspection car, hand-held platform, and UAV platforms for real-time rail track inspections. Meanwhile, more powerful supervised machine learning and classification algorithms, such as neural dynamic classification algorithm, dynamic ensemble learning algorithm, and finite element machine for fast learning, could be evaluated for other potential railroad inspection solutions.

REFERENCES

Adeli, H. (2001). Neural networks in civil engineering: 1989–2000. *Computer-Aided Civil Infrastructure Engineering*, *16*(2), 126–142.

Adeli, H., & Yeh, C. (1989). Perceptron learning in engineering design. *Computer-Aided Civil and Infrastructure Engineering*, *4*(4), 247–256.

Afif, M., Ayachi, R., Said, Y., Pissaloux, E., & Atri, M. (2020). An evaluation of RetinaNet on indoor object detection for blind and visually impaired persons assistance navigation. *Neural Processing Letters*, *51*, 2265–2279.

AlNaimi, N. R. (2020). Rail robot for rail track inspection *(Master dissertation)*. Qatar University, Qatar.

Amezquita-Sanchez, J. P., Valtierra-Rodriguez, M., Aldwaik, M., & Adeli, H. (2016). Neurocomputing in civil infrastructure. *Scientia Iranica-A*, *23*(6), 2417–2428.

Arabi, S., Haghighat, A., & Sharma, A. (2020). A deep-learning-based computer vision solution for construction vehicle detection. *Computer-Aided Civil Infrastructure Engineering*, *35*(7), 753–767.

Ariyachandra, M., & Brilakis, I. (2020). Detection of railway masts in airborne lidar data. *Journal of Construction Engineering Management*, *146*(9), 04020105.

Bang, S., Park, S., Kim, H., & Kim, H. (2019). Encoder-decoder network for pixel-level road crack detection in black-box images. *Computer-Aided Civil Infrastructure Engineering*, *34*(8), 713–727.

Cha, Y. J., Choi, W., & Büyüköztürk, O. J. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil Infrastructure Engineering*, *32*(5), 361–378.

Cha, Y. J., Choi, W., Suh, G., Mahmoudkhani, S., & Büyüköztürk, O. (2018). Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civil Infrastructure Engineering*, *33*(9), 731–747.

Ciampoli, L. B., Calvi, A., & Oliva, E. (2020). Test-site operations for the health monitoring of railway ballast using ground-penetrating radar. *Transportation Research Procedia*, *45*, 763–770.

De Boer, P.-T., Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A tutorial on the cross-entropy method. *Annals of Operations Research*, *134*(1), 19–67.

Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., & Heng, P.-A. (2018). R3Net: Recurrent residual refinement network for saliency detection. *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, 684–690.

Gao, S., Cheng, M.-M., Zhao, K., Zhang, X.-Y., Yang, M.-H., & Torr, P. H. (2019). Res2Net: A new multi-scale backbone architecture. *IEEE Transactions on Pattern Analysis Machine Intelligence*, *43*, 652–662.

Gao, S.-H., Tan, Y.-Q., Cheng, M.-M., Lu, C., Chen, Y., & Yan, S. (2020). Highly efficient salient object detection with 100k

parameters. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 1–19.

Guo, F., Qian, Y., Wu, Y., Leng, Z., & Yu, H. (2021). Automatic railroad track components inspection using real-time instance segmentation. *Computer-Aided Civil Infrastructure Engineering*, *36*(3), 362–377.

He, Z., Wang, Y., Yin, F., & , J. (2016). Surface defect detection for high-speed rails using an inverse P-M diffusion model. *Sensor Review*, *36*, 86–97.

Hou, Q., Cheng, M., Hu, X., Borji, A., Tu, Z., & Torr, P. (2019). Deeply supervised salient object detection with short connections. *IEEE Transactions on Pattern Analysis Machine Intelligence*, *41*(4), 815.

Hou, Q., Cheng, M.-M., Hu, X., Borji, A., Tu, Z., & Torr, P. H. (2017). Deeply supervised salient object detection with short connections. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 3203–3212.

Hsieh, Y. A., Yang, Z., & James Tsai, Y. C. (2021). Convolutional neural network for automated classification of jointed plain concrete pavement conditions. *Computer-Aided Civil and Infrastructure Engineering*, 1–16. https://doi.org/10.1111/mice.12640

Islam, M. A., Kalash, M., Rochan, M., Bruce, N. D., & Wang, Y. (2017). Salient object detection using a context-aware refinement network. *British Machine Vision Conference*, London, 1–12.

Jia, S., & Bruce, N. D. (2019). Richer and deeper supervision network for salient object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, United States, 4321–4331.

Kapur, J. N., Sahoo, P. K., & Wong, A. K. (1985). A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, image Processing*, *29*(3), 273–285.

Kong, S. Y., Fan, J. S., Liu, Y. F., Wei, X. C., & Ma, X. W. J. (2021). Automated crack assessment and quantitative growth monitoring. *Computer-Aided Civil Infrastructure Engineering*, 1–19. https://doi.org/10.1111/mice.12626

Kong, X., & Li, J. (2018). Vision-based fatigue crack detection of steel structures using video feature tracking. *Computer-Aided Civil Infrastructure Engineering*, *33*(9), 783–799.

Li, C., Guo, J., Porikli, F., Fu, H., & Pang, Y. (2018). A cascaded convolutional neural network for single image dehazing. *IEEE Access*, *6*, 24877–24887.

Li, G., & Yu, Y. (2016a). Visual saliency detection based on multiscale deep CNN features. *IEEE Transactions on Image Processing*, *25*(11), 5012–5024.

Li, G., & Yu, Y. (2016b). Deep contrast learning for salient object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 478–487.

Li, Q., & Ren, S. (2012a). A visual detection system for rail surface defects. *IEEE Transactions on Systems, Man, Cybernetics, Part C*, *42*(6), 1531–1542.

Li, Q., & Ren, S. (2012b). A real-time visual inspection system for discrete surface defects of rail heads. *IEEE Transactions on Instrumentation Measurement*, *61*(8), 2189–2199.

Liang, J., Zhou, J., Tong, L., Bai, X., & Wang, B. (2018). Material based salient object detection from hyperspectral images. *Pattern Recognition*, *76*, 476–490.

Liang, X. (2019). Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with bayesian optimization. *Computer-Aided Civil Infrastructure Engineering*, *34*(5), 415–430.

Lim, B., Son, S., Kim, H., Nah, S., & Mu Lee, K. (2017). Enhanced deep residual networks for single image super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 136–144.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, 2980–2988.

Liu, J.-J., Hou, Q., Cheng, M.-M., Feng, J., & Jiang, J. (2019). A simple pooling-based design for real-time salient object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, USA, 3917–3926.

Liu, N., & Han, J. (2016). Dhsnet: Deep hierarchical saliency network for salient object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 678–686.

Liu, N., Han, J., Zhang, D., Wen, S., & Liu, T. (2015). Predicting eye fixations using convolutional neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 362–370.

Lu, S., & Lim, J.-H. (2012). Saliency modeling from image histograms. *European Conference on Computer Vision*, (pp. 321–332). Springer.

Maeda, H., Sekimoto, Y., Seto, T., Kashiyama, T., & Omata, H. (2018). Road damage detection and classification using deep neural networks with smartphone images. *Computer-Aided Civil Infrastructure Engineering*, *33*(12), 1127–1141.

Mahendran, A., & Vedaldi, A. (2015). Understanding deep image representations by inverting them. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 5188–5196.

Marino, F., Distante, A., Mazzeo, P. L., & Stella, E. (2007). A real-time visual inspection system for railway maintenance: Automatic hexagonal-headed bolts detection. *IEEE Transactions on Systems, Man, Cybernetics, Part C*, *37*(3), 418–428.

Nagendar, G., Singh, D., Balasubramanian, V. N., & Jawahar, C. (2018). Neuro-Iou: Learning a surrogate loss for semantic segmentation. *British Machine Vision Conference*, Newcastle, UK.

Nah, S., Hyun Kim, T., & Mu Lee, K. (2017). Deep multi-scale convolutional neural network for dynamic scene deblurring. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 3883–3891.

Ni, F., Zhang, J., & Noori, M. N. (2020). Deep learning for data anomaly detection and data compression of a long-span suspension bridge. *Computer-Aided Civil Infrastructure Engineering*, *35*(7), 685–700.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, Cybernetics*, *9*(1), 62–66.

Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O. R., & Jagersand, M. (2020). U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern Recognition*, *106*, 107404.

Qin, X., Zhang, Z., Huang, C., Gao, C., Dehghan, M., & Jagersand, M. (2019). BASNet: Boundary-aware salient object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 7479–7489.

Rafiei, M. H., & Adeli, H. (2017a). NEEWS: A novel earthquake early warning model using neural dynamic classification and neural dynamic optimization. *Soil Dynamics and Earthquake Engineering*, *100*, 417–427.

Rafiei, M. H., & Adeli, H. (2017b). A novel machine learning-based algorithm to detect damage in high-rise building structures. *The Structural Design of Tall Special Buildings*, *26*(18), 1–11.

Rafiei, M. H., & Adeli, H. (2018). A novel unsupervised deep learning model for global and local health condition assessment of structures. *Engineering Structures*, *156*, 598–607.

Railway Technology. (2020). https://www.railway-technology.com/ .

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, (pp. 234–241). Springer.

Sajedi, S. O., & Liang, X. (2021). Uncertainty-assisted deep vision structural health monitoring. *Computer-Aided Civil Infrastructure Engineering*, *36*(2), 126–142.

Sharma, S., Cui, Y., He, Q., Mohammadi, R., & Li, Z. (2018). Data-driven optimization of railway maintenance for track geometry. *Transportation Research Part C: Emerging Technologies*, *90*, 34–58.

Tatiraju, S., & Mehta, A. (2008). Image segmentation using K-means clustering, EM and normalized cuts. *Department of EECS*, *1*, 1–7.

Wang, K., Cao, W., Xu, L., Yang, X., Su, Z., Zhang, X., & Chen, L. (2020). Diffuse ultrasonic wave-based structural health monitoring for railway turnouts. *Ultrasonics*, *101*, 106031.

Wang, N., Zhao, X., Zou, Z., Zhao, P., & Qi, F. (2020). Autonomous damage segmentation and measurement of glazed tiles in historic buildings via deep learning. *Computer-Aided Civil Infrastructure Engineering*, *35*(3), 277–291.

Wang, T., Zhang, L., Wang, S., Lu, H., Yang, G., Ruan, X., & Borji, A. (2018). Detect globally, refine locally: A novel approach to saliency detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 3127–3135.

Wang, Z., Simoncelli, E. P., & Bovik, A. C. (2003). Multiscale structural similarity for image quality assessment. *Proceedings of the Thirty-Seventh Asilomar Conference on Signals, Systems & Computers*, Pacific Grove, CA, USA, 1398–1402.

Wang, Z., Wu, X., Yu, G., & Li, M. (2018). Efficient rail area detection using convolutional neural network. *IEEE Access*, *6*, 77656–77664.

Wei, X., Wei, D., Suo, D., Jia, L., & Li, Y. (2020). Multi-target defect identification for railway track line based on image processing and improved YOLOv3 model. *IEEE Access*, *8*, 61973–61988.

Whittle, P. (1994). The psychophysics of contrast brightness. In A. L. Gilchrist (Ed.), *Lightness, brightness, and transparency* (pp. 35–110). Lawrence Erlbaum Associates, Inc.

Wu, Y., Qin, Y., & Jia, L. (2018). Research on rail surface defect detection method based on UAV images. *Proceedings of 2018 Prognostics and System Health Management Conference*, Chongqing, China, 553–558.

Wu, Y., Qin, Y., Wang, Z., & Jia, L. (2018). A UAV-based visual inspection method for rail surface defects. *Applied Sciences*, *8*(7), 1028.

Wu, Y., Qin, Y., Wang, Z., Ma, X., & Cao, Z. (2020). Densely pyramidal residual network for UAV-based railway images dehazing. *Neurocomputing*, *371*, 124–136.

Xinhuanet. (2020). http://www.xinhuanet.com/home.htm.

Yanan, S., Hui, Z., Li, L., & Hang, Z. (2018). Rail surface defect detection method based on YOLOv3 deep learning networks. *Proceedings of 2018 Chinese Automation Congress*, Xi'an, China, 1563–1568.

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. *European Conference on Computer Vision*, (pp. 818–833). Springer.

Zhang, J., Ehinger, K. A., Wei, H., Zhang, K., & Yang, J. (2017). A novel graph-based optimization framework for salient object detection. *Pattern Recognition*, *64*, 39–50.

Zhang, L., Dai, J., Lu, H., He, Y., & Wang, G. (2018). A bi-directional message passing model for salient object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 1741–1750.

Zhang, P., Wang, D., Lu, H., Wang, H., & Ruan, X. (2017). Amulet: Aggregating multi-level convolutional features for salient object detection. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, 202–211.

Zhang, Q., Huo, Z., Liu, Y., Pan, Y., Shan, C., & Han, J. (2019). Salient object detection employing a local tree-structured low-rank representation and foreground consistency. *Pattern Recognition*, *92*, 119–134.

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., & Fu, Y. (2018). Residual dense network for image super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2472–2481.

Zhao, J.-X., Liu, J.-J., Fan, D.-P., Cao, Y., Yang, J., & Cheng, M.-M. (2019). EGNet: Edge guidance network for salient object detection. *Proceedings of the IEEE International Conference on Computer Vision*, Seoul, Korea (South). 8779–8788.

Zhao, K., Gao, S., Wang, W., & Cheng, M.-M. (2019). Optimizing the F-measure for threshold-free salient object detection. *Proceedings of the IEEE International Conference on Computer Vision*, Seoul, Korea (South), 8849–8857.

Zhao, R., Ouyang, W., Li, H., & Wang, X. (2015). Saliency detection by multi-context deep learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 1265–1274.

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). Unet++: A nested U-Net architecture for medical image segmentation. *Proceedings of Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Granada, Spain, 3–11.

How to cite this article: Wu Y, Qin Y, Qian Y, Guo F, Wang Z, Jia L. Hybrid deep learning architecture for rail surface segmentation and surface defect detection. *Comput Aided Civ Inf*. 2022;*37*:227–244. https://doi.org/10.1111/mice.12710