remote sensing

MDPI

Article

ERNet: A Rapid Road Crack Detection Method Using Low-Altitude UAV Remote Sensing Images

Zexian Duan, Jiahang Liu *, Xinpeng Ling, Jinlong Zhang and Zhiheng Liu

College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
* Correspondence: jhliu@nuaa.edu.cn

Abstract: The rapid and accurate detection of road cracks is of great significance for road health monitoring, but currently, this work is mainly completed through manual site surveys. Low-altitude UAV remote sensing can provide images with a centimeter-level or even subcentimeter-level ground resolution, which provides a new, efficient, and economical approach for rapid crack detection. Nevertheless, crack detection networks face challenges such as edge blurring and misidentification due to the heterogeneity of road cracks and the complexity of the background. To address these issues, we proposed a real-time edge reconstruction crack detection network (ERNet) that adopted multi-level information aggregation to reconstruct crack edges and improve the accuracy of segmentation between the target and the background. To capture global dependencies across spatial and channel levels, we proposed an efficient bilateral decomposed convolutional attention module (BDAM) that combined depth-separable convolution and dilated convolution to capture global dependencies across the spatial and channel levels. To enhance the accuracy of crack detection, we used a coordinate-based fusion module that integrated spatial, semantic, and edge reconstruction information. In addition, we proposed an automatic measurement of crack information for extracting the crack trunk and its corresponding length and width. The experimental results demonstrated that our network achieved the best balance between accuracy and inference speed compared to six established models.

Keywords: UAV remote sensing; road cracks; semantic segmentation; crack quantification; edge detection

Citation: Duan, Z.; Liu, J.; Ling, X.; Zhang, J.; Liu, Z. ERNet: A Rapid Road Crack Detection Method Using Low-Altitude UAV Remote Sensing Images. *Remote Sens.* **2024**, *16*, 1741. https://doi.org/10.3390/rs16101741

Academic Editors: Riccardo Roncella and Belén Riveiro

Received: 1 April 2024
Revised: 3 May 2024
Accepted: 10 May 2024
Published: 14 May 2024

1. Introduction

Pavement cracking is a common type of damage that significantly reduces the service life of roads and poses a safety risk to road users [1–3]. Visual interpretation is the main approach to crack detection, but it is inefficient and prone to subjective errors. Over the past decade, various automatic and semi-automatic methods have been proposed, including the use of sensors such as line scan cameras, RGB-D sensors, and laser scanners [4–6]. However, these sensor-equipped vehicles are costly and frequently cause traffic disruption and road-type restrictions.

Nowadays, unmanned aerial vehicles (UAVs) have emerged as efficient and versatile tools for structural inspections [7,8]. UAV-based road crack detection offers significant advantages, including efficient, cost-effective, safe, and flexible image data acquisition [9]. In recent decades, digital image processing has been utilized for crack segmentation [10]. These approaches often require manual feature extraction, which can overlook the interdependence between cracks and lead to unsatisfactory results in practice [11]. UAV remote sensing has successfully solved the problem of a data source for crack detection; therefore, how to quickly and accurately detect and measure cracks has become the main problem at present.

Machine-learning-based detection methods have been rapidly developed in recent years and are becoming the mainstream approach for crack detection [12–14]. These algorithms require different preprocessing techniques for the image to be detected, which can be

time-consuming for large images. With the rapid development of deep learning, new ideas have been introduced into various computer vision tasks [15–17]. Semantic segmentation is often preferred for crack detection, as it provides more accurate and effective road health information, such as the crack distribution, width, length, and shape. Liu et al. combined an FCN and a deep supervision network (DSN) to propose DeepCrack [18], a multi-scene crack detection algorithm based on the idea of deep supervision. Ren et al. [19] proposed an improved CrackSegNet for the pixel-level crack segmentation of tunnel surfaces, which improves the accuracy and generalization by spatial pyramid pooling and skip-connection modules. Liu et al. [20] proposed the use of U-Net for automatic crack detection, but it may generate redundant recognition due to background interference. Wang et al. [21] found that, by combining a CNN with a transformer, an efficient feedforward network can be constructed for global feature extraction. These models often suffer from computational delays and inefficiency due to their large number of parameters and computational redundancy. This becomes particularly acute when dealing with large amounts of data, resulting in a high overhead of computational power.

To reduce computational costs, researchers have proposed some lightweight networks. Many initial lightweight models prioritized speed over spatial detail. These methods may lead to the loss of spatial detail and precision, particularly in the boundary regions. Yu et al. introduced the bilateral segmentation network (BiSeNet) in their groundbreaking research [22], processing semantic and detailed information separately. Lightweight models often lack the ability to effectively extract edge information due to the characteristics of narrow cracks, irregular edges, and the potential for confusion with the road background. This can significantly impact the detection accuracy. The short-term dense cascade (STDC) module was proposed as a solution to this issue [23]. The STDC segmentation network (STDC-Seg) uses the STDC module to extract multi-scale information, which solves the existing backbone network problems of BiSeNet. Overall, STDC-Seg is a suitable segmentation structure for road crack detection.

The quantitative extraction of physical information from cracks is a downstream task in crack detection. To acquire this information, researchers have combined crack detection algorithms with crack quantization algorithms to provide a safe and effective solution for road crack detection. Liao et al. [24] used a spatially constrained strategy for a lightweight CNNS to ensure fracture continuity. Yang et al. [25] attempted to conduct a quantitative analysis of detected cracks at the pixel level. However, the quantitative results did not meet expectations. Li et al. [26] proposed a pixel-level crack segmentation method that fused the SegNet and the dense condition random field and calculated the width and area of one-way and grid cracks. In general, the density, width, and length of cracks can provide important reference information for road health evaluations, and accurate crack detection results provide an important foundation for the extraction of these elements.

Due to the narrow shape of most cracks, crack edge information is very important for the accurate location and segmentation of cracks. In addition, in most crack detection tasks, irregular cracks, rough roads, light, shadows, and other factors will affect the location and segmentation of cracks. The accurate detection of crack edges is crucial for semantic segmentation networks to address these challenges and extract quantitative information, such as the crack length and width. Tao et al. [27] designed a boundary awareness module in their proposed approach, but their label-based learning was prone to misjudging the background noise. Pang et al. [28] introduced a two-branch lightweight network into crack detection, but the lightweight design limited the network's ability to extract global information, so the network easily missed small cracks. Holistic nested edge detection (HED) [29] and a side-output residual network (SRN) [30] are two edge detection networks that build on the idea of deep supervision. Tsai et al. [31] fused the edge detection results of different sizes extracted by the Sobel edge detector on the semantic branch. However, it is often difficult for existing methods to address the problems of a weak perception of crack edge details and an uneven crack distribution, which makes quantitative information extraction a challenge.

To overcome these limitations, we proposed a rapid road crack detection method for UAV remote sensing images. Specifically, we proposed a real-time edge reconstruction crack detection network (ERNet) that integrates edge aggregation and enhancement into semantic segmentation. Inspired by infrared small-target detection [32], we developed an edge input module by utilizing a soft gating mechanism for edge reconstruction. The proposed method achieved the best trade-off between inference speed and accuracy compared to the other models participating in the comparison experiment. The *mIoU* score for the Crack500 dataset was 82.48%, and the *F*1 score was 79.67%. The *mIoU* score for the DeepCrack dataset was 86.6%, and the *F*1 score was 84.86%. The *mIoU* score of the generalization experiment for the self-made UAV dataset was 80.25% and the *F*1 score was 76.21%. A comparative analysis demonstrated the feasibility and superiority of this method. Our main contributions are as follows:

(1) We proposed a novel ERNet to achieve high-precision and fast crack edge segmentation through edge reconstruction and realized the quantification of crack length and width information on this basis. The ERNet provides a whole solution from detection to extraction.

(2) We designed a key model called BDAM that effectively improves the attention at both the spatial and channel levels, selectively represents features in the channel and spatial domains, and captures global contextual information.

The rest of this article is organized as follows. Section 2 describes the architecture of the ERNet and its components in detail. Section 3 verifies the effectiveness of our method in improving the comprehensive performance of crack detection with experimental results. Section 4 is the conclusion.

2. Methodology

The difficulty of the crack detection task comes from a fuzzy boundary transition of the crack, a chaotic background, foreground interference, etc. The accurate location of a crack edge is the key to dealing with these challenges. By reconstructing the edge details, our proposed network improves the location accuracy of the crack edge, improves the coherence of the detection results, and provides accurate detection results for the quantitative extraction of cracks. The overall structure of the network is shown in Figure 1.

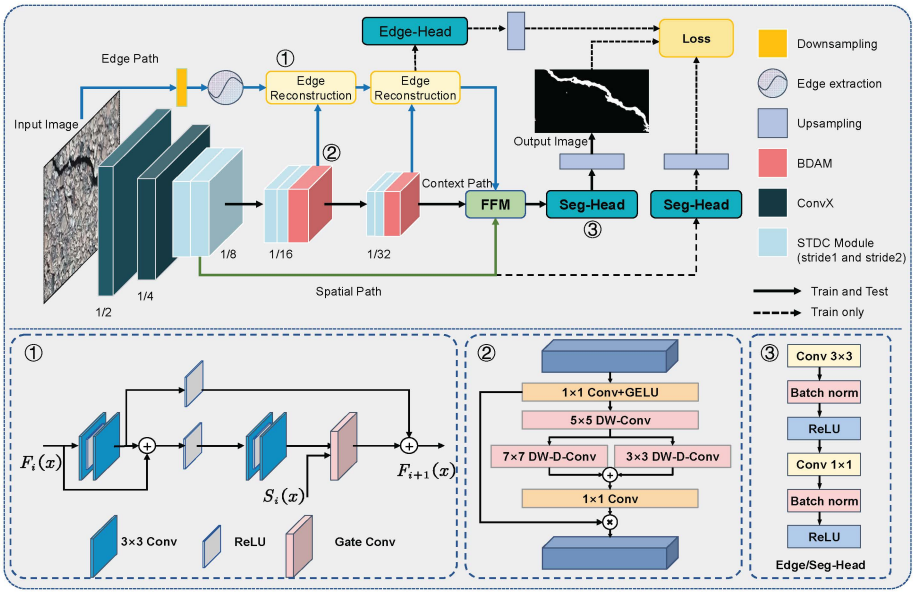


Figure 1. A structural overview of our proposed network for crack detection.

The network uses a three-branch structure to encode features at different levels, including an edge path for extracting and preserving high-frequency features, a spatial path for

preserving detailed information, and a semantic path for extracting deep semantic features. In the semantic path, we used an STDC module for local feature extraction and the BDAM for global feature extraction. In the edge path, we inputted high-frequency information and semantic information into the edge reconstruction module to encode the edge features, and used the key damage boundary information. In the spatial path, we implemented shallow and wide convolution layers to achieve fast downsampling and preserve spatial details.

In this section, we first introduce the backbone network we used, then introduce the bilateral decomposed convolutional feature attention module, and finally describe in detail the side input branch for edge detection and the feature fusion module of the model.

2.1. Backbone

Our proposed model used the STDC module as a feature extractor and retained the spatial branch. We used the STDC-Seg network backbone as the ERNet backbone. The operation of ConvX included a convolution layer, a batch normalization layer, and an ReLU activation layer. We used feature maps of 1/8 size instead of 1/4 size as the spatial branching input, because it reduced the amount of computation and preserved enough spatial detail. The STDC module was the core component of the backbone network, as shown in Figure 2.

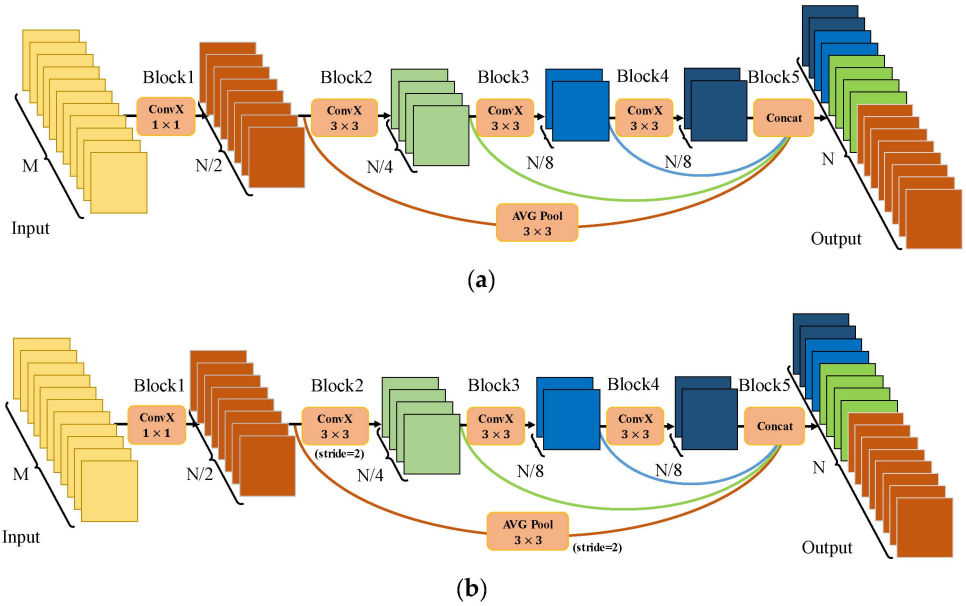


Figure 2. Illustration of the STDC module. (**a**) STDC module with a stride of 1. (**b**) STDC module with a stride of 2.

Two types of STDC modules, stride = 1 and stride = 2, were used for different tasks. The STDC module with a stride of 2 was used for downsampling the feature maps, and then the STDC module with a stride of 1 was used for further feature extraction. The number of filters in the i-th convolution layer of the block was N/2i, where N is the output of the STDC module. The number of filters in the last two convolution layers was set to be the same. The STDC module was divided into several blocks. The feature mapping of the i-th block was calculated using Equation (1):

$$\boldsymbol{X_{out}} = F(x_1, x_2, \ldots x_n) \tag{1}$$

where $\boldsymbol{X_{out}}$ represents the module output, F represents the concat fusion operation, and $x_1, x_2, \ldots x_n$ are the feature maps of all the blocks.

The output of the STDC module integrated the multi-scale information of all the blocks. As the number of blocks increased, the receptive field also increased, and the scalable receptive field and information were retained throughout the fusion operations.

2.2. BDAM

The significance of the global context in segmentation tasks has been confirmed by numerous studies [33–37]. Convolution-based methods accomplish this by enlarging the receptive field through an increased kernel size or stride, whereas transformer-based methods [38,39] usually consider the spatial dimension adaptability and ignore the channel dimension adaptability, which is important for visual tasks.

To capture distant relationships, we introduced decomposed convolution blocks and designed the efficient bilateral decomposed convolution attention module (BDAM). As illustrated in Figure 3, the large kernel convolution was divided into three parts by the BDAM: depth convolution for capturing the multi-scale context, multi-branch depth convolution, and 1 × 1 convolution for establishing relationships between distinct channels. During the decomposition process, we broke down the K × K convolution into the depth convolution, multi-scale depth expansion convolution, and 1 × 1 convolution. The BDAM can be described as follows:

$$\boldsymbol{Att} = Conv_{1\times1}(DW - DConv_{branch1}(\boldsymbol{X_{in}}) + DW - DConv_{branch2}(\boldsymbol{X_{in}})) \quad (2)$$

$$\boldsymbol{Out} = \boldsymbol{Att} \otimes \boldsymbol{X_{in}} \quad (3)$$

where $\boldsymbol{X_{in}}$ denotes the input features, corresponding to the multiplication operation of element matrices, and ***Att*** and ***Out*** represent the attention maps and outputs, respectively.

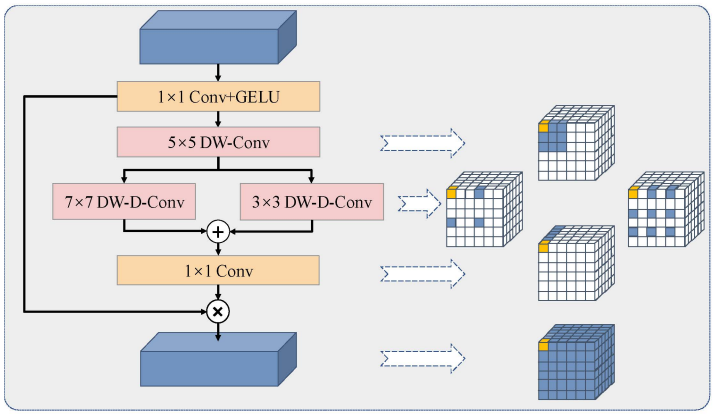


Figure 3. Illustration of the bilateral decomposed convolutional attention module. The BDAM was used to refine the corresponding combined features of the decoding stage. Among them, depth-wise separable convolution and dilated convolution were used to capture the global content, and an attention vector was used for guidance.

In this network, the depth-dilated convolutions in each branch had kernel sizes of 3 and 7. This configuration aligned with standard convolutions, which have kernel sizes of 7 and 19, and it enabled the capturing of remote relationships across different scales using depth-dilated convolutions with varying kernel sizes in a dual-branch structure. The output of the 1 × 1 convolution served as the attention weight for the input features, providing both spatial and channel adaptability.

2.3. Edge Reconstruction Module

In detection tasks, small and narrow cracks are often lost in multiple downsampling processes. The feature information of cracks is closely related to their edge information, which includes the fine details of the target. To address this issue, the Laplacian operator was adopted as an edge extraction operator to filter the image and further refine the coarse edge information extracted from it. However, the Laplacian operator's use at each stage increased the computational complexity, and setting the threshold for judging the boundary too high or too low can result in ineffective edge detection. In addition, it is difficult for the

Laplacian operator to extract edge information from convolutional encoded features. After several experiments, we used 1/8 size images as the input and chose a threshold of 40 for the Laplacian operator.

Inspired by small-target detection, we used the edge reconstruction module (ERM) based on the second-order Taylor finite difference equation to process rough edge features [40]. The structure of the ERM enabled a nonlinear transformation of the shallow edge feature map through two residual blocks to obtain features with less noise and clutter. Then, the soft gate mechanism was employed to perform directed learning on the rough edge results obtained by the Laplacian operator, which better suppressed background noise and focused on the edge information of the target using the semantic features extracted by the backbone, which is shown in Figure 4. In this figure, $\boldsymbol{F_i}(\boldsymbol{x})$ denotes rough edge features, $\boldsymbol{F_{i+1}}(\boldsymbol{x})$ denotes refined edge features, and $\boldsymbol{S_i}(\boldsymbol{x})$ denotes high-level semantic features.

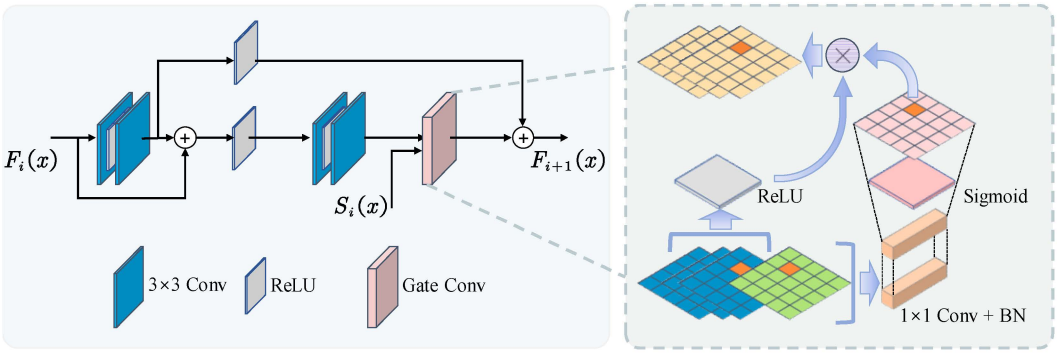


Figure 4. Illustration of the edge reconstruction module. (**a**) Edge reconstruction module. (**b**) Gate convolution.

The gate convolution learned a soft mask automatically from the data. Guided by the soft mask, the edge reconstruction module extracted accurate crack boundary information from the chaotic rough edge features. It can be formulated as shown in Equation (4):

$$\boldsymbol{Gate_{out}} = \phi(\boldsymbol{Feature_i}(\boldsymbol{x})) \odot \sigma\Big(W_f(\boldsymbol{S_i}(\boldsymbol{x}), \boldsymbol{Feature_i}(\boldsymbol{x}))\Big) \tag{4}$$

where σ is sigmoid; thus, the output gating values were between zero and one. ϕ is the ReLU. W_f is a sequence of convolutional filters.

In road surface crack detection, the number of crack pixels is significantly lower than that of non-crack pixels, resulting in a class imbalance problem. Weighted cross-entropy, as mentioned in Ref. [41], often leads to rough results. To address this issue, we jointly optimized edge learning using binary cross-entropy and the edge loss [42]. The edge loss is a general dice-based edge-aware loss module that includes a dice edge loss function for overall contour fitting. The required edge prediction results are defined as follows:

$$\hat{e}_{ij} = squash(g_{ij}) = \frac{|g_{ij}|}{|g_{ij}|+\alpha} \tag{5}$$

$$\Theta = argmaxDice(\boldsymbol{p_d}, \boldsymbol{g_d}) = argmax \frac{2\sum_{i=j}^{H}\sum_{j}^{W} e_j \hat{e}_{ij} \hat{e}_{ij}}{\sum_{i}^{H}\sum_{j}^{W} {e_{ij}}^2 + \sum_{i}^{H}\sum_{j}^{W} {e_{ij}}^2} \tag{6}$$

where $\hat{e}_{ij}$ and g_{ij} represent the edge prediction results and gradient information vectors at (i, j), respectively; e_{ij} is the true edge value directly obtained from the detail ground truth; and α is a hyperparameter that controls the model's sensitivity to object contours. In our experiments, we found that setting α to 1 achieved an optimal balance between intra-class unification and inter-class discrimination. The boundary refinement was represented by the

dice coefficient maximization problem, as defined in Equation (6) above, where $g_d \epsilon R^{H \times W}$ is the true segmentation map, $p_d \epsilon R^{H \times W}$ is the predicted segmentation map, and Θ represents the parameter of the segmentation network. To implement the SGD in the training process, the final edge loss was calculated using Equation (7). In addition, we used the dice loss and the BCE loss to train the spatial branch of the ERNet, and the main loss function of the network was the BCE loss.

$$L_{edge}(p_d, g_d, \Theta) = 1 - Dice(p_d, g_d, \Theta) \tag{7}$$

2.4. Feature Fusion Module

The proposed network's feature fusion module (FFM) extracted multiple feature responses and fused information from different levels of feature maps to achieve multi-element and multi-scale information encoding. As shown in Figure 5, the edge features were first concatenated with spatial and semantic features, and then the feature map size was divided into $C \times H \times 1$ and $C \times 1 \times W$ along the X and Y coordinates, respectively, using average pooling. The resulting feature maps were then divided into two separate tensors along the spatial dimension, and an attention vector was generated by a sigmoid to guide the feature response of the spatial branch. This encoding of multi-element and multi-scale information integrated low-level feature maps with spatial information, edge reconstruction feature maps with edge information, and high-level feature maps with large receptive fields.

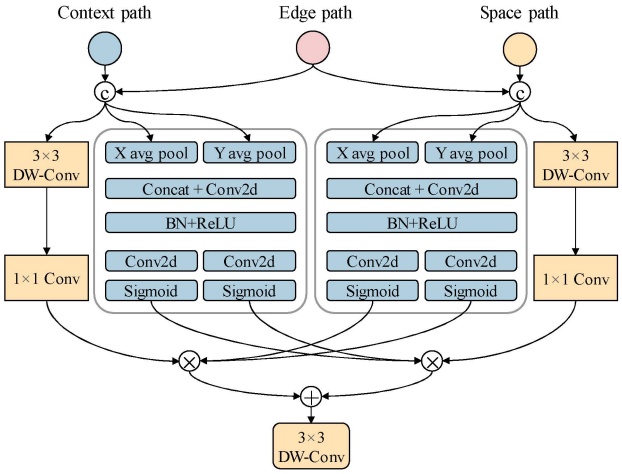
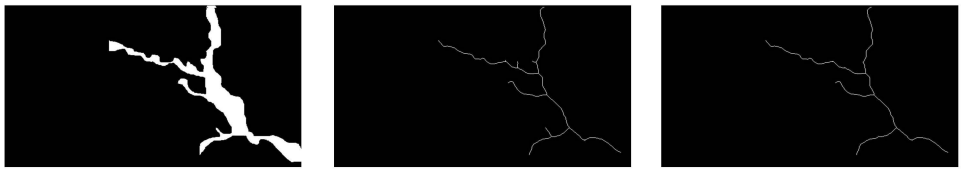


Figure 5. Illustration of the feature fusion module based on coordinates.

2.5. Crack Information Quantification

The crack length plays a crucial role in road safety predictions, as longer cracks indicate more severe road damage. The segmentation network generated a crack prediction map by predicting cracks at the pixel level. We extracted correct crack skeletons by eliminating a large number of erroneous branches, which were identified using an algorithm developed by Zhang and Suen et al. [43] (shown in Figure 6b) based on a connected domain analysis. The results of the crack skeleton extraction are shown in Figure 6c. The crack trunk was extracted effectively by a debranching algorithm based on a connected domain. Finally, the number of adjacent pixels in the crack skeleton and the distance between adjacent cracks were calculated pixel by pixel, and the maximum length value represented the crack length.



(**a**) (**b**) (**c**)

Figure 6. Diagram of crack skeleton backbone extraction. (**a**) Crack prediction map. (**b**) Zhang–Suen thinning algorithm. (**c**) Crack backbone extraction.

The width of the crack is equally important for road damage detection. Based on the distance transform method (DTM), the distance between the crack skeleton and the crack edge was calculated, so as to obtain the maximum width. As shown in Figure 7a, the wider the crack area, the greater the gray value. Figure 7b shows the results of the crack skeleton weighted with DTM values, so as to obtain the maximum width.

(**a**) (**b**)

Figure 7. Result of the crack based on the distance transformation method. (**a**) DTM values of the crack prediction map. (**b**) Result of weighting the crack skeleton with DTM values.

3. Experiment and Results

This section details the dataset used for the experiments, the training details of the proposed algorithm, the evaluation criteria, and the experimental results.

3.1. Dataset

It is difficult to obtain road crack data with UAVs. Considering that the camera resolution of a UAV is high enough and the angle of view of a UAV is similar to that of a mobile phone, it can be assumed that the images collected by both have a similar definition and imaging angle. We used the public road crack datasets Crack500 and DeepCrack, collected by mobile phones, as the training sets. In addition, we used the DJI UAV to take some road crack images, and made a small dataset to test the network generalization ability. The specific dataset can be described as follows.

The UAV dataset was collected for a generalization ability test. The images were captured using a DJI M300RTK drone equipped with a ZENMUSE H20 camera. The pixel resolution of the images was 5184 $\times$ 3888. Since the size of each single image was too large, we used LabelMe [44] for semantic annotation, and then cropped the image to a size of 512 $\times$ 512. By using data enhancement operations such as image flipping, we made a generalization dataset containing 4692 UAV images of aerial road cracks. The data were collected on the campus of the Nanjing University of Aeronautics and Astronautics in Nanjing, Jiangsu Province, China. The annotated dataset included both cement and asphalt road surfaces, with var*IoU*s types of cracks such as net-shaped cracks, longitudinal cracks, and transverse cracks.

The Crack500 dataset [45] consisted of 500 road crack images. In this experiment, each original image was divided into 16 non-overlapping images, each with a scale of 640 $\times$ 352. Images containing more than 1000 pixels of crack area were kept and further divided. The training set comprised 1896 images, the validation set comprised 348 images, and the test set comprised 1124 images.

The DeepCrack dataset [18] consisted of 537 road crack images with a size of 544 $\times$ 384, each with a pixel-level binary label image. In our experiments, the dataset was divided into 300 images for the training dataset and 237 images for the validation dataset.

3.2. Implementation Details

All the models in the experiments were implemented with the PyTorch framework on a single NVIDIA GTX 3090 GPU. We used the SGD [46] to train our ERNet with a batch size of 8, and the training epoch was set to 100; we applied the "poly" learning rate strategy in which the initial rate was multiplied by Equation (8):

$$lr = initial\_lr \times \left(1 - \frac{iter}{max\_iter}\right)^{\text{power}} \tag{8}$$

where *iter* is the number of iterations, *max_iter* is the maximum number of iterations, and power controls the shape of the curve. The initial learning rate was set to 0.01, and the power was set to 0.9.

3.3. Comparative Experiment

We compared our ERNet with three lightweight semantic segmentation networks (BiSeNet [22], STDC2-seg [23], and PIDNet [47]) and three crack detection networks (DeepCrackNet [18], CT-CrackSeg [27], and LinkCrack [24]) based on the same implementation details and platform.

The accuracy evaluation standards used in this experiment were the intersection over union (*IoU*), precision (*Pr*), recall (*Re*), *F*1 score (*F*1), and accuracy (*Acc*). We also calculated the average frames per second (*FPS*) of the network reasoning in the validation set while calculating the *IoU*. The measurements are shown in Equations (9)–(13):

$$IOU = \frac{N_{TP}}{N_{FP}+N_{TP}+N_{FN}} \tag{9}$$

$$Acc = \frac{N_{TP} + N_{TN}}{N_{TP} + N_{TN} + N_{FP} + N_{FN}} \tag{10}$$

$$Pr = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{11}$$

$$Re = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{12}$$

$$F1 = \frac{2 \cdot Pr \cdot Re}{Pr + Re} \tag{13}$$

where N_{TP} is the number of positive samples classified as positive, N_{TN} is the number of negative samples classified as negative, N_{FP} is the number of negative samples classified as positive, and N_{FN} is the number of positive samples classified as negative.

The precision and recall evaluate the detection ability of the method from different perspectives. The *F*1 score combines the above two metrics. The *IoU* can give a better response to the local details of the detection results, and the mean *IoU* (*mIoU*) is the average *IoU* of the road and the crack. *Acc* represents the proportion of correctly classified data (N_{TP} + N_{TN}) relative to the total data. These indicators can be used to evaluate the detection performance of the network more objectively. The values of these indicators range between 0 and 1. Higher values, closer to 1, indicate a better segmentation ability for crack areas. The validation data were used to select the optimal training iteration.

Table 1 presents the comprehensive indicators for the Crack500 validation dataset, with the best-performing values highlighted in bold. Our model achieved the highest results for the *IoU*, *Re*, *Acc*, *mIoU*, and *F*1 score. Its speed was also the fastest of the four crack detection networks; although our model had a 0.5 *FPS* lower frame rate than STDC2-seg, it achieved a 2.86% higher *IoU* for cracks compared to STDC2-seg and a 3.98% higher *F*1, and our model was 5.25 *FPS* faster than the third-place model, making this trade-off acceptable. Despite having a lower precision compared to PIDNet, our model's higher *F*1 score demonstrated its superior ability to distinguish between the background and cracks.

Table 1. Comparison of the experimental results of different semantic segmentation networks on the Crack500 dataset.

	***IoU* (%)**	***mIoU* (%)**	***Pr* (%)**	***Re* (%)**	***F*1 (%)**	***Acc* (%)**	***FPS***
BiSeNet (2018) [22]	62.83	79.99	73.31	78.68	75.90	97.87	11.86
STDC2-seg (2021) [23]	63.35	80.39	74.71	76.69	75.69	98.37	**22.1** *
PIDNet (2023) [47]	64.28	80.88	**79.87**	76.71	78.26	97.59	16.35
DeepCrackNet (2019) [18]	55.67	76.69	66.75	77.02	71.52	96.54	3.35
CT-CrackSeg (2023) [27]	62.54	80.04	60.50	78.00	73.30	97.02	3.42
LinkCrack (2022) [24]	57.45	77.92	72.97	72.98	72.98	96.95	11.54
ERNet (ours)	**66.21**	**82.48**	79.21	**80.14**	**79.67**	**98.51**	21.6

* The bolded value is the optimal value achieved by experiments, the same below.

Figure 8 presents the segmentation results for several image examples. The first row demonstrates that our model's recognition results had fewer breakpoints and were closer to the original images than the results of the other networks, which is not uncommon in experiments with two datasets. The inference results for the second row of background-mottled crack images reveal that our model exhibited fewer missed detections, clearer edge details, and smoother boundaries compared to the other networks. In the third and fourth rows of the image, all the other networks have false checks in the shaded area. The fifth row demonstrates our model's ability to recognize detailed information within the cracks, which was not achieved by the other networks.

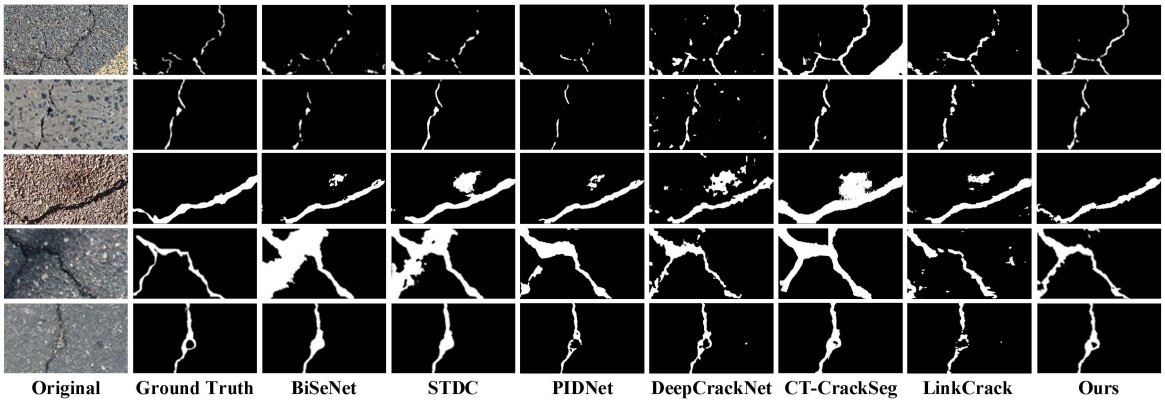


Figure 8. A visualization of the different semantic segmentation detection results of the compared methods on Crack500.

Upon testing our model on the DeepCrack dataset, it achieved superior *IoU*, recall, and *F*1 scores. Although ERNet had a 0.73% lower precision than PIDNet, it had a 3.28% higher recall and, thus, a 1.46% higher *F*1 score than PIDNet. On the other hand, although our model had a 1.15% lower recall than LinkCrack, it was 7.05% ahead in precision and, thus, had a 2.87% higher *F*1 score than LinkCrack. In terms of speed, our model was only lower than STDC2-seg, which may have been due to the fact that the two-branch STDC2-seg had a superior inference speed on images with a small resolution. Although our model had a slightly lower *Acc* than STDC2-seg (by 0.16%), it led by 7.31% for the *IoU* and 5.07% for the *F*1 score, demonstrating its robust stability across various datasets. The detailed comparison results are presented in Table 2. Figure 9 displays the segmentation results for several image examples with increased interference.

Table 2. Comparison of the experimental results of different semantic segmentation networks on the DeepCrack dataset.

	IoU (%)	*mIoU* (%)	*Pr* (%)	*Re* (%)	*F1* (%)	*Acc* (%)	*FPS*
BiSeNet	69.10	83.76	81.29	79.19	80.23	98.97	13.1
STDC2-seg	66.40	82.33	84.52	75.56	79.79	**99.40** *	**30.2**
PIDNet	71.54	85.06	**88.52**	78.85	83.40	98.64	25.47
DeepCrackNet	67.90	83.53	81.21	80.56	80.88	98.35	3.40
CT-CrackSeg	64.69	81.79	76.55	80.68	78.56	98.09	3.49
LinkCrack	69.48	84.29	80.74	**83.28**	81.99	98.42	14.07
ERNet (ours)	**73.71**	**86.60**	87.79	82.13	**84.86**	99.24	26.2

* The bolded value is the optimal value achieved by experiments, the same below.

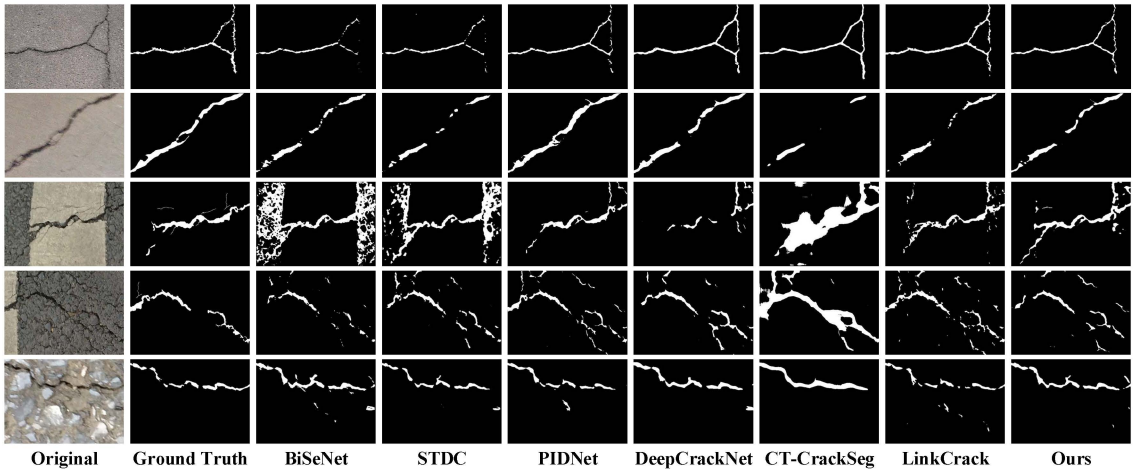


Figure 9. A visualization of the different semantic segmentation detection results of the compared methods on DeepCrack.

For the blurry images in the first and second rows, the small cracks extracted by our model were more coherent and retained more details. The images in the third and fourth rows demonstrate complex interference conditions due to shadows, light, and widespread net-shaped cracks, leading to decreased segmentation results for all the network models, with BiSeNet, STDC2-seg, and CT-CrackSeg all showing extensive levels of false detection; however, our model had a smaller false detection range compared to the other networks. The identification results of the fifth row showed that the detection results of our model preserved the edge details of the crack well, while reducing the number of breakpoints in the zigzag crack.

3.4. Generalization Ability Experiment

In this section, we used the best weight of each network obtained on the Crack500 training set to make predictions for the UAV crack dataset, so as to detect the generalization ability of each network. This generalization experiment can truly reflect the ability of each network in the actual detection task. The experimental results are presented in Table 3. Our inference results outperformed the results of other models in the *IoU*, *mIoU*, *Pr*, *F*1, and *Acc* metrics. Meanwhile, all the other networks showed a substantial decline in their performance indicators. The *mIoU* of our inference results was only 2.23% lower than that of the Crack500 dataset. These results demonstrate that our proposed model possesses a good generalization ability.

Table 3. Comparison of the experimental results of different semantic segmentation networks on the UAV remote sensing dataset.

	IoU (%)	*mIoU* (%)	*Pr* (%)	*Re* (%)	*F1* (%)	*Acc* (%)
BiSeNet	41.87	69.73	58.71	59.36	59.03	97.62
STDC2-seg	41.71	69.66	59.40	58.34	58.87	97.65
PIDNet	35.15	66.14	51.12	52.94	52.01	97.18
DeepCrackNet	21.25	57.55	23.95	65.32	35.05	93.02
CT-CrackSeg	48.13	73.45	62.09	68.16	64.98	97.88
LinkCrack	26.91	63.21	65.26	31.42	42.41	97.54
ERNet(ours)	**61.56** *	**80.25**	**69.91**	**83.75**	**76.21**	**98.36**

* The bolded value is the optimal value achieved by experiments, the same below.

Figure 10 illustrates the segmentation results of different networks on UAV remote sensing images. The detection results of DeepCrackNet and LinkCrack were very scattered and lacked clear boundaries. CT-CrackSeg performed the best after ERNet, but it still mistakenly detected pavement markings as cracks, and the missing detection of small cracks was also quite significant. Specifically, the second row of the figure compares the segmentation of shallow cracks; our model identified the main body of shallow cracks and correctly handled pavement markings. The third row contains both longitudinal cracks and transverse cracks. Other networks missed transverse cracks, while our model successfully segmented both types of cracks relatively completely. The fourth and fifth rows depict the same area from different angles, demonstrating that our model not only approximated the ground truth, but also exhibited a high consistency in the same area in the two images. DeepCrackNet made many false detections in the background area far from the crack, which indicates that its ability to extract global semantic information still has room for improvement. In conclusion, these experiments demonstrated that our proposed model possesses a good generalization ability and delivers an excellent performance in detecting remote sensing road images.

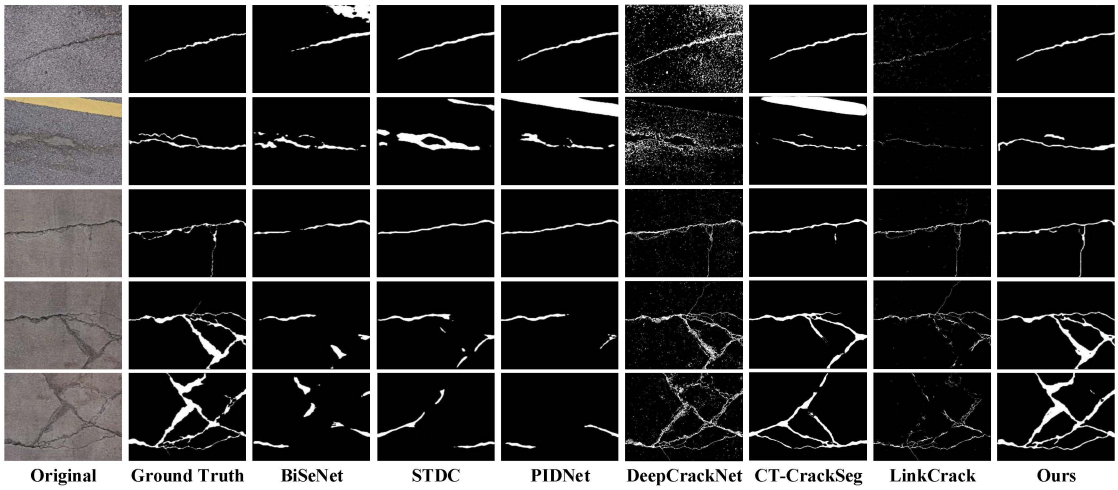


Figure 10. A visualization of the different semantic segmentation detection results of the compared methods on the UAV remote sensing dataset.

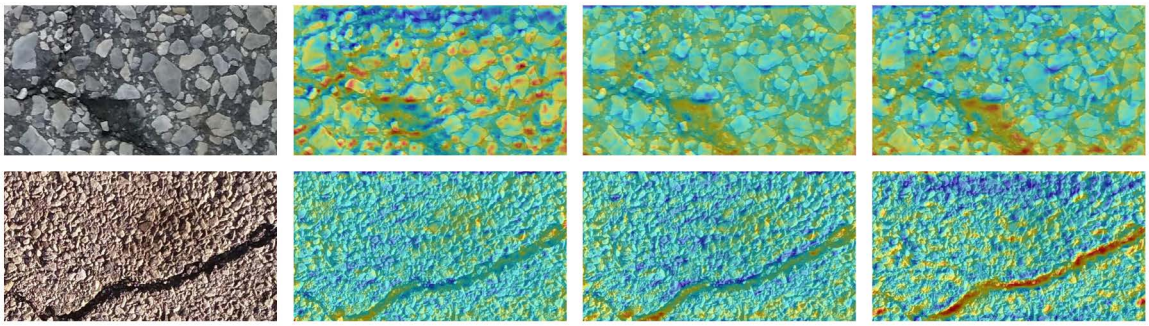
3.5. Ablation Experiment Results

In this section, we performed six experiments on the Crack500 dataset using the STDC backbone, incorporating the BDAM, ERM, and FFM sequentially. Table 4 presents the contributions of each module and their combinations. Figure 11 illustrates the visualization of different mechanisms. The introduced BDAM improved the ERNet's understanding of the overall scene and reduced the background interference; the introduced ERM focused on the crack region and improved the ERNet's accuracy in locating cracks.

Table 4. Impact of BDAM, ERM, and FFM on network performance.

	IoU (%)	*mIoU* (%)	*Re* (%)	*Acc* (%)
Original	62.71	80.02	76.03	97.48
Original+BDAM	63.42	80.41	76.36	97.51
Original+ERM	63.72	80.54	78.56	97.47
Original+BDAM+ERM	65.08	81.28	78.98	97.53
Original+ERM+FFM	63.79	80.57	79.06	97.47
Original+BDAM+ERM+FFM	**66.21** *	**82.48**	**80.14**	**98.51**

* The bolded value is the optimal value achieved by experiments, the same below.



(**a**) (**b**) (**c**) (**d**)

Figure 11. A visualization of different mechanisms. (**a**) Image; (**b**) without the BDAM; (**c**) without the ERM; and (**d**) with the BDAM and ERM.

BDAM: In the second experiment, the BDAM enhanced the multiscale receptive field information, resulting in a 0.71% improvement in the *IoU* compared to the original. This demonstrates the ability of the BDAM to capture global context information.

ERM: In the third experiment, the side input module's enhanced edge positioning led to a 1.66% improvement in the crack *IoU* and a 2.62% increase in the recall. This demonstrates that the side input module not only increased the accuracy of edge pixel segmentation, but also enhanced the overall segmentation accuracy for the category.

FFM: In the fourth experiment, the FFM was employed to replace the default feature concatenation operation, resulting in a 1.13% improvement in the *IoU*. This suggests that the FFM-based coordinate feature guidance efficiently encodes multi-element and multiscale information.

3.6. Crack Information Quantification Experiment Results

We simulated the actual detection process, inputted the images of the Crack500 dataset into the ERNet to create a prediction map, and obtained the calculated value of the crack length and width of the prediction map through the proposed crack information quantization algorithm. Two experiments were designed to verify the accuracy of the quantization results and the crack prediction results.

We selected 10 sets of calculated prediction values to compare with the calculated label values, in order to verify the effectiveness of the crack information quantization algorithm we proposed above. And we selected another 10 sets of calculated prediction values to compare with the measured values of the original RGB images, in order to verify the accuracy of the crack prediction results. To assess the universality of the results, we introduced the average absolute error and average relative error. The detailed results are presented in Tables 5 and 6.

Table 5. Comparison table of ground truth and prediction crack calculated parameters.

	Crack Length and Error (Pixel)				Crack Width and Error (Pixel)			
Number	**Calculated Label Value**	**Calculated Prediction Value**	**Absolute Error**	**Relative Error/%**	**Calculated Label Value**	**Calculated Prediction Value**	**Absolute Error**	**Relative Error/%**
1	640	691	51	7.97	38	42	4	10.53
2	409	441	32	7.82	34	37	3	8.82
3	638	651	13	2.04	64	66	2	3.13
4	339	368	29	8.55	24	26	2	8.33
5	238	272	34	14.29	80	80	0	0.00
6	150	151	1	0.67	37	31	6	16.22
7	286	324	38	13.29	46	49	3	6.52
8	158	175	17	10.76	26	25	1	3.85
9	247	264	17	6.88	44	42	2	4.55
10	355	365	10	2.82	40	42	2	5.00
Average			24.2	7.51			2.5	6.69

Table 6. Comparison table of measured and calculated crack parameters.

	Crack Length and Error (Pixel)				Crack Width and Error (Pixel)			
Number	**Measured Value**	**Calculated Prediction Value**	**Absolute Error**	**Relative Error/%**	**Measured Value**	**Calculated Prediction Value**	**Absolute Error**	**Relative Error/%**
1	341	323	18	5.28	27	26	1	3.70
2	697	691	6	0.86	44	39	5	11.36
3	405	441	36	8.89	24	24	0	0.00
4	638	651	13	2.04	27	26	1	3.70
5	623	721	98	15.73	30	31	1	3.33
6	166	151	15	9.04	80	80	0	0.00
7	522	500	22	4.21	60	56	4	6.67
8	215	208	7	3.26	50	51	1	2.00
9	375	351	24	6.40	26	25	1	3.85
10	355	365	10	2.82	44	42	2	4.55
Average			24.9	5.85			1.6	3.92

The crack length in the comparison ranged from 150 to 697 pixels, and the width ranged from 24 to 80 pixels. The experimental results showed that the average relative errors for the calculated values of the crack length and width relative to the labeled cracks were 7.51% and 6.69%, respectively, and the average relative errors for the calculated values relative to the original image were 5.85% and 3.92%, respectively. Since the crack information based on the original image was manually measured, there will inevitably be errors due to human factors and instrument influences.

4. Discussion

Our research aimed to automatically detect cracks from road images; thus, a novel method for the efficient detection and extraction of road cracks was proposed. A segmentation network based on edge reconstruction was utilized to achieve the real-time detection of road cracks. To improve the accuracy of edge reconstructions and to guide global detection, we introduced a soft-gate control mechanism to fuse high-level gradient semantic information. In addition, we proposed a depth-decomposed convolutional attention module that utilizes deep and dilated convolution techniques to process the global contextual information of images. The crack detection results were automatically quantized to extract the length and width of the crack backbone. The experimental results showed that our method outperformed other comparative methods. From the experimental results in Section 3.3, it can be seen that CT-CrackSeg was better at detecting fine cracks because it retained shallow information, but it was less effective at detecting road cracks in the presence of complex background disturbances. LinkCrack and PIDNet obtained better results in cases with complex backgrounds, but they missed the detection of fine cracks and had poorer coherence in the presence of fine cracks. Noting the phenomenon that edge information favors detail information and deep information favors semantic information, our method employed the ERM for the selective enhancement of edge information, and the

quantitative and visualization results showed that this method had a good performance for the edge localization of cracks. Figure 12 shows the segmentation results before and after the ERM was added, which demonstrates that the ERM can effectively improve the segmentation accuracy of crack boundaries.

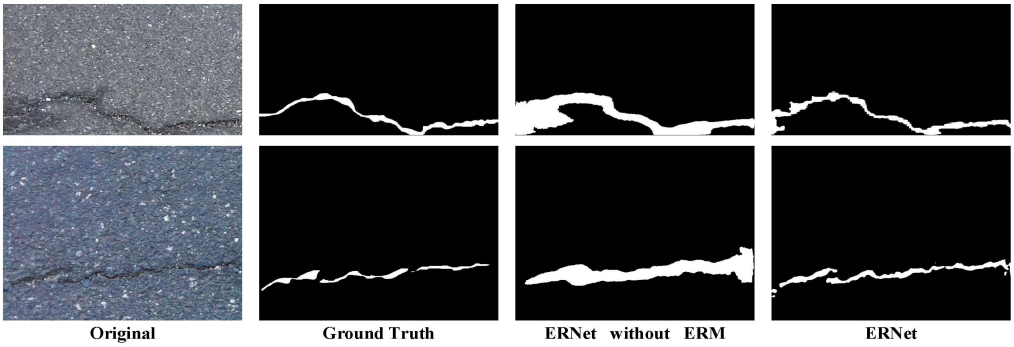


Figure 12. A visualization of the edge reconstruction results.

In addition, bridges, tunnels, and dams have civil engineering structural problems along with roads, and cracks usually appear as line-like anomalous areas on the images. This visual characterization of cracks is similar on both roads and other structures. Therefore, crack images of bridges, tunnels, and dams can be captured as the target domain, and the ERNet that performs well in the source domain (road crack images) can be optimized in the target domain by domain adaptation, so that the ERNet can be applied to the inspection of other structures such as bridges, tunnels, and dams. The high level of light interference in tunnels needs to be considered to improve the visibility of cracks under low-light conditions using image-processing techniques.

As can be seen from Tables 1 and 2, our method is faster than some networks, but slower than the STDC-Seg network, which may be due to the modeled three-branch structure that increases the amount of computation. In the next step, we will apply model compression techniques such as model pruning and quantization to speed up the inference process. In addition, how to further optimize the performance of the model on larger datasets is the focus of our next study.

Author Contributions: J.L. supervised the study, designed the architecture, and revised the manuscript; Z.D. wrote the manuscript and designed the comparative experiments. X.L. made suggestions for the manuscript and assisted Z.D. in conducting the experiments. J.Z. and Z.L. made suggestions for the experiments and assisted in revising the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Innovative Talent Program of Jiangsu under grant JSSCR2021501.

Data Availability Statement: The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to thank the editors and the reviewers for their valuable suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zheng, M.; Lei, Z.; Zhang, K. Intelligent Detection of Building Cracks Based on Deep Learning. *Image Vis. Comput.* **2020**, *103*, 103987. [CrossRef]
2. Wu, C.; Sun, K.; Xu, Y.; Zhang, S.; Huang, X.; Zeng, S. Concrete Crack Detection Method Based on Optical Fiber Sensing Network and Microbending Principle. *Saf. Sci.* **2019**, *117*, 299–304. [CrossRef]
3. Kim, B.; Yuvaraj, N.; Sri Preethaa, K.R.; Arun Pandian, R. Surface Crack Detection Using Deep Learning with Shallow CNN Architecture for Enhanced Computation. *Neural Comput. Appl.* **2021**, *33*, 9289–9305. [CrossRef]

4. Gavilán, M.; Balcones, D.; Marcos, O.; Llorca, D.F.; Sotelo, M.A.; Parra, I.; Ocaña, M.; Aliseda, P.; Yarza, P.; Amírola, A. Adaptive Road Crack Detection System by Pavement Classification. *Sensors* **2011**, *11*, 9628–9657. [CrossRef] [PubMed]
5. Jahanshahi, M.R.; Jazizadeh, F.; Masri, S.F.; Becerik-Gerber, B. Unsupervised Approach for Autonomous Pavement-Defect Detection and Quantification Using an Inexpensive Depth Sensor. *J. Comput. Civ. Eng.* **2013**, *27*, 743–754. [CrossRef]
6. Zhang, D.; Zou, Q.; Lin, H.; Xu, X.; He, L.; Gui, R.; Li, Q. Automatic Pavement Defect Detection Using 3D Laser Profiling Technology. *Autom. Constr.* **2018**, *96*, 350–365. [CrossRef]
7. Zhong, X.; Peng, X.; Yan, S.; Shen, M.; Zhai, Y. Assessment of the Feasibility of Detecting Concrete Cracks in Images Acquired by Unmanned Aerial Vehicles. *Autom. Constr.* **2018**, *89*, 49–57. [CrossRef]
8. Peng, X.; Zhong, X.; Zhao, C.; Chen, A.; Zhang, T. A UAV-Based Machine Vision Method for Bridge Crack Recognition and Width Quantification through Hybrid Feature Learning. *Constr. Build. Mater.* **2021**, *299*, 123896. [CrossRef]
9. Peng, X.; Zhong, X.; Zhao, C.; Chen, Y.F.; Zhang, T. The Feasibility Assessment Study of Bridge Crack Width Recognition in Images Based on Special Inspection UAV. *Adv. Civ. Eng.* **2020**, *2020*, 8811649. [CrossRef]
10. Mazzini, D.; Napoletano, P.; Piccoli, F.; Schettini, R. A Novel Approach to Data Augmentation for Pavement Distress Segmentation. *Comput. Ind.* **2020**, *121*, 103225. [CrossRef]
11. Yang, J.; Fu, Q.; Nie, M. Road crack detection using deep neural network with receptive field block. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *782*, 042033. [CrossRef]
12. Nguyen, T.S.; Begot, S.; Duculty, F.; Avila, M. Free-Form Anisotropy: A New Method for Crack Detection on Pavement Surface Images. In Proceedings of the 2011 18th IEEE International Conference on Image Processing, Brussels, Belgium, 11–14 September 2011; IEEE: Brussels, Belgium, 2011; pp. 1069–1072.
13. Liang, X. Image-based Post-disaster Inspection of Reinforced Concrete Bridge Systems Using Deep Learning with Bayesian Optimization. *Comput. Aided Civ. Infrastruct. Eng.* **2019**, *34*, 415–430. [CrossRef]
14. Du, P.; Bai, X.; Tan, K.; Xue, Z.; Samat, A.; Xia, J.; Li, E.; Su, H.; Liu, W. Advances of Four Machine Learning Methods for Spatial Data Handling: A Review. *J. Geovis. Spat. Anal.* **2020**, *4*, 13. [CrossRef]
15. Abou-Chacra, D.; Zelek, J. Effects of Spatial Transformer Location on Segmentation Performance of a Dense Transformer Network. *J. Comput. Vis. Imaging Syst.* **2017**, *3*. [CrossRef]
16. Islam, M.M.M.; Kim, J.-M. Vision-Based Autonomous Crack Detection of Concrete Structures Using a Fully Convolutional Encoder–Decoder Network. *Sensors* **2019**, *19*, 4251. [CrossRef]
17. König, J.; Jenkins, M.D.; Mannion, M.; Barrie, P.; Morison, G. Optimized Deep Encoder-Decoder Methods for Crack Segmentation. *Digit. Signal Process.* **2021**, *108*, 102907. [CrossRef]
18. Liu, Y.; Yao, J.; Lu, X.; Xie, R.; Li, L. DeepCrack: A Deep Hierarchical Feature Learning Architecture for Crack Segmentation. *Neurocomputing* **2019**, *338*, 139–153. [CrossRef]
19. Ren, Y.; Huang, J.; Hong, Z.; Lu, W.; Yin, J.; Zou, L.; Shen, X. Image-Based Concrete Crack Detection in Tunnels Using Deep Fully Convolutional Networks. *Constr. Build. Mater.* **2020**, *234*, 117367. [CrossRef]
20. Liu, Z.; Cao, Y.; Wang, Y.; Wang, W. Computer Vision-Based Concrete Crack Detection Using U-Net Fully Convolutional Networks. *Autom. Constr.* **2019**, *104*, 129–139. [CrossRef]
21. Wang, J.; Zeng, Z.; Sharma, P.K.; Alfarraj, O.; Tolba, A.; Zhang, J.; Wang, L. Dual-Path Network Combining CNN and Transformer for Pavement Crack Segmentation. *Autom. Constr.* **2024**, *158*, 105217. [CrossRef]
22. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. BiSeNet: Bilateral Segmentation Network for Real-Time Semantic Segmentation. In *Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2018; Volume 11217, pp. 334–349. ISBN 978-3-030-01260-1.
23. Fan, M.; Lai, S.; Huang, J.; Wei, X.; Chai, Z.; Luo, J.; Wei, X. Rethinking BiSeNet for Real-Time Semantic Segmentation. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; IEEE: Nashville, TN, USA, 2021; pp. 9711–9720.
24. Liao, J.; Yue, Y.; Zhang, D.; Tu, W.; Cao, R.; Zou, Q.; Li, Q. Automatic Tunnel Crack Inspection Using an Efficient Mobile Imaging Module and a Lightweight CNN. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 15190–15203. [CrossRef]
25. Yang, X.; Li, H.; Yu, Y.; Luo, X.; Huang, T.; Yang, X. Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network: Pixel-Level Crack Detection and Measurement Using FCN. *Comput. Aided Civ. Infrastruct. Eng.* **2018**, *33*, 1090–1109. [CrossRef]
26. Li, G.; Liu, Q.; Ren, W.; Qiao, W.; Ma, B.; Wan, J. Automatic Recognition and Analysis System of Asphalt Pavement Cracks Using Interleaved Low-Rank Group Convolution Hybrid Deep Network and SegNet Fusing Dense Condition Random Field. *Measurement* **2021**, *170*, 108693. [CrossRef]
27. Tao, H.; Liu, B.; Cui, J.; Zhang, H. A Convolutional-Transformer Network for Crack Segmentation with Boundary Awareness. In Proceedings of the 2023 IEEE International Conference on Image Processing (ICIP) 2023, Kuala Lumpur, Malaysia, 8–11 October 2023; pp. 86–90. [CrossRef]
28. Pang, J.; Zhang, H.; Zhao, H.; Li, L. DcsNet: A Real-Time Deep Network for Crack Segmentation. *Signal Image Video Process.* **2022**, *16*, 911–919. [CrossRef]
29. Xie, S.; Tu, Z. Holistically-Nested Edge Detection. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; IEEE: Santiago, Chile, 2015; pp. 1395–1403.

30. Ke, W.; Chen, J.; Jiao, J.; Zhao, G.; Ye, Q. SRN: Side-Output Residual Network for Object Reflection Symmetry Detection and Beyond. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 1881–1895. [CrossRef] [PubMed]
31. Tsai, T.-H.; Tseng, Y.-W. BiSeNet V3: Bilateral Segmentation Network with Coordinate Attention for Real-Time Semantic Segmentation. *Neurocomputing* **2023**, *532*, 33–42. [CrossRef]
32. Zhang, M.; Zhang, R.; Yang, Y.; Bai, H.; Zhang, J.; Guo, J. ISNet: Shape Matters for Infrared Small Target Detection. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; IEEE: New Orleans, LA, USA, 2022; pp. 867–876.
33. Hung, W.-C.; Tsai, Y.-H.; Shen, X.; Lin, Z.; Sunkavalli, K.; Lu, X.; Yang, M.-H. Scene Parsing with Global Context Embedding. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2650–2658. [CrossRef]
34. Liu, H.; Peng, C.; Yu, C.; Wang, J.; Liu, X.; Yu, G.; Jiang, W. An End-To-End Network for Panoptic Segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 6165–6174. [CrossRef]
35. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826. [CrossRef]
36. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.
37. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2018; Volume 11211, pp. 833–851. ISBN 978-3-030-01233-5.
38. Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; IEEE: Montreal, QC, Canada, 2021; pp. 548–558.
39. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; IEEE: Montreal, QC, Canada, 2021; pp. 9992–10002.
40. He, X.; Mo, Z.; Wang, P.; Liu, Y.; Yang, M.; Cheng, J. ODE-Inspired Network Design for Single Image Super-Resolution. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; IEEE: Long Beach, CA, USA, 2019; pp. 1732–1741.
41. Hu, P.; Caba, F.; Wang, O.; Lin, Z.; Sclaroff, S.; Perazzi, F. Temporally Distributed Networks for Fast Video Semantic Segmentation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Seattle, WA, USA, 2020; pp. 8815–8824.
42. Zheng, X.; Huan, L.; Xia, G.-S.; Gong, J. Parsing Very High Resolution Urban Scene Images by Learning Deep ConvNets with Edge-Aware Loss. *ISPRS J. Photogramm. Remote Sens.* **2020**, *170*, 15–28. [CrossRef]
43. Zhang, T.Y.; Suen, C.Y. A Fast Parallel Algorithm for Thinning Digital Patterns. *Commun. ACM* **1984**, *27*, 236–239. [CrossRef]
44. Russell, B.C.; Torralba, A.; Murphy, K.P.; Freeman, W.T. LabelMe: A Database and Web-Based Tool for Image Annotation. *Int. J. Comput. Vis.* **2008**, *77*, 157–173. [CrossRef]
45. Yang, F.; Zhang, L.; Yu, S.; Prokhorov, D.; Mei, X.; Ling, H. Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1525–1535. [CrossRef]
46. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
47. Xu, J.; Xiong, Z.; Bhattacharyya, S.P. PIDNet: A Real-Time Semantic Segmentation Network Inspired by PID Controllers. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; IEEE: Vancouver, BC, Canada, 2023; pp. 19529–19539.