

Article

Fatigue Crack Detection Based on Semantic Segmentation Using DeepLabV3+ for Steel Girder Bridges

Xuejun Jia ^{1,2}, Yuxiang Wang ^{3,*} and Zhen Wang ^{4,*}¹ College of Transportation Engineering, Nanjing Technology University, Nanjing 211899, China; jxj1147208@sina.com² China Construction Second Engineering Bureau Co., Ltd., Central China Branch, Wuhan 430062, China³ School of Civil and Hydraulic Engineering, Huazhong University of Science and Technology, Wuhan 430074, China⁴ School of Civil Engineering and Architecture, Wuhan University of Technology, Wuhan 430070, China

* Correspondence: wy_xiang@hust.edu.cn (Y.W.); wang_zhen@whut.edu.cn (Z.W.)

Abstract: Artificial intelligence technology is receiving more and more attention in structural health monitoring. Fatigue crack detection in steel box girders in long-span bridges is an important and challenging task. This paper presents a semantic segmentation network model for this task based on DeepLabv3+, ResNet50, and active learning. Specifically, the classification network ResNet50 is re-tuned using the crack image dataset. Secondly, with the re-tuned ResNet50 as the backbone network, a crack semantic segmentation network was constructed based on DeepLabv3+, which was trained with the assistance of active learning. Finally, optimization for the probability threshold of the pixel category was performed to improve the pixel-level detection accuracy. Tests show that, compared with the crack detection network based on conventional ResNet50, this model can improve MIoU from 0.6181 to 0.7241.

Keywords: semantic segmentation; DeepLabv3+; crack detection; threshold; active learning

Citation: Jia, X.; Wang, Y.; Wang, Z. Fatigue Crack Detection Based on Semantic Segmentation Using DeepLabV3+ for Steel Girder Bridges. *Appl. Sci.* **2024**, *14*, 8132. <https://doi.org/10.3390/app14188132>

Academic Editor: José António Correia

Received: 23 July 2024

Revised: 31 August 2024

Accepted: 3 September 2024

Published: 10 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the continuous advancements in bridge design and construction technology, long-span steel bridges have developed rapidly. Among these, steel box-girder bridges have become popular due to their light weight, high torsional rigidity, and other advantages. However, due to the coupling effect between initial material defects and dynamic vehicle loads, fatigue cracks often occur in steel bridge joints, especially around welded joints.

Fatigue cracks in steel girder bridges pose significant safety risks and can lead to catastrophic failures if not detected and repaired in time. Traditional inspection methods, which rely heavily on manual visual inspections, are labor-intensive, time-consuming, and prone to human error. In recent years, advancements in artificial intelligence (AI) have revolutionized the field of structural health monitoring, offering more efficient, accurate, and automated solutions for fatigue crack detection. Techniques employed for fatigue crack detection in steel girder bridges include machine learning-based techniques, convolutional neural networks (CNNs), and semantic segmentation networks, among others.

Machine learning (ML) has been widely applied in structural health monitoring for feature extraction and pattern recognition. Various ML algorithms, such as support vector machines (SVMs), k-nearest neighbor (KNN), and decision trees, have been employed to classify and detect cracks in steel structures. For example, Zhang et al. [1] used SVM to

classify features extracted from acoustic emission signals, achieving a high detection accuracy for fatigue cracks in steel beams. Similarly, Li et al. [2] implemented a KNN-based approach to analyze strain data collected from sensors on steel girder bridges and successfully identified crack initiation and propagation phases.

CNNs have gained popularity due to their superior performance in image-based applications. CNNs can automatically extract features from raw image data, making them highly effective for crack detection in bridge structures. By processing images captured from inspections, CNNs can identify and classify cracks with high precision. Kim et al. [3] developed a CNN model for crack detection in steel bridges using images taken by drones. The model achieved an accuracy of over 95% in identifying fatigue cracks. Furthermore, Yang and Yao [4] designed a multi-scale CNN to detect cracks of varying sizes and orientations, demonstrating robust performances under different lighting and environmental conditions.

Semantic segmentation networks, such as DeepLabv3+ [5] and U-Net, are advanced deep learning models that provide pixel-level classification, allowing for the precise localization and segmentation of cracks in images. These networks are particularly useful for identifying small and complex crack patterns that might be missed by traditional methods. Chen et al. [6] introduced DeepLabv3+, which utilizes atrous spatial pyramid pooling to capture multi-scale contextual information. This model has been applied to crack detection, offering high accuracy in segmenting cracks from background noise. Similarly, Ronneberger et al. [7] proposed the U-Net architecture, which has been widely adopted for medical image segmentation and recently adapted for structural health monitoring tasks.

Transfer learning involves pre-training a model on a large dataset and fine-tuning it on a smaller, specific dataset. This approach is beneficial when the labeled data for crack detection are limited, as it leverages the knowledge gained from related tasks. Xu et al. [8] applied transfer learning by fine-tuning a pre-trained ResNet model on a dataset of bridge crack images. The transfer learning approach significantly improved the detection accuracy, especially in scenarios with limited training data.

The integration of AI techniques into fatigue crack detection for steel girder bridges has significantly advanced the field of structural health monitoring. However, developing more robust models that can handle diverse environmental disturbance and provide more accurate prediction results requires further investigation. Aiming to achieve this objective, this paper presents a semantic segmentation model for fatigue crack detection based on DeepLabv3+, ResNet50, and active learning.

This paper is organized as follows. Section 2 provides a brief introduction to DeepLabv3+. The proposed fatigue crack detection model is presented in Section 3. In Section 4, relevant models are compared in terms of validation accuracy and mean intersection over union (MIoU). To further improve the model performance and robustness, the pixel category threshold optimization is discussed in Section 5. Section 6 examines and presents the compound model by comparing the predicted fatigue cracks with the ground truth. Finally, the main contents are summarized in Section 7.

2. Overview of Semantic Segmentation Model: DeepLabv3+

In the field of computer vision, deep learning is mainly used for three aspects, namely: image classification, object detection, and semantic segmentation. Image classification predicts labels for an image. Object detection focuses on providing labels and locations for objects in the images. The semantic segmentation technique involves labeling each pixel in the image and hence segmenting the image into several regions with different attributes and categories. Lately, semantic segmentation has been receiving more attention, since it can provide more accurate locations and other information. This paper studies the fatigue crack detection method in steel bridges based on DeepLabv3+ semantic segmentation technology, which can determine the location, length, and shape of the crack.

DeepLabv3+ adopts an encoding–decoding architecture, as shown in Figure 1. The encoding part designs multiple dilated convolution modules with different dilating rates to obtain semantic information at multiple scales. The features outputted by the encoder are firstly up-sampled to obtain semantic information, which is connected to the feature layer with the same spatial resolution outputted by the low-level network. Bilinear interpolation up-sampling is used to restore the output feature map to the spatial resolution of the input image, leading to semantic segmentation outputs.

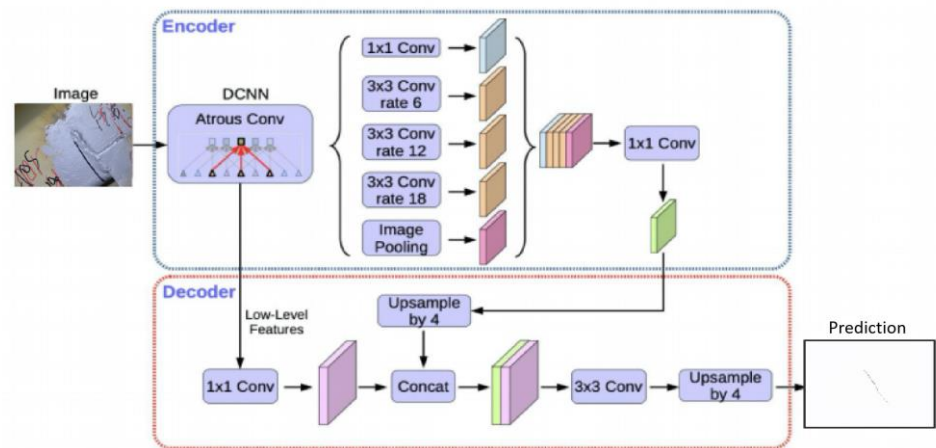


Figure 1. DeepLabv3+ model.

In order to accurately assess the quality of semantic segmentation results, evaluation indicators [9], such as category accuracy, intersection union ratio (IoU), and mean intersection union ratio (MIoU), are often employed. For readers' convenience, the latter two are described herein. The IoU of crack pixels can be written as follows:

$$IoU_i = \frac{TP_i}{TP_i + FP_i + FP_j} \quad (1)$$

where TP_i is the number of pixels for true crack prediction, and FP_j is the pixel number for false background (non-crack) prediction. MIoU is defined as the average of the IoU of two categories, namely:

$$MIoU = \frac{IoU_i + IoU_j}{2} \quad (2)$$

3. Proposed Fatigue Crack Detection Model Based on Sematic Segmentation

3.1. Overview of the Proposed Model

The semantic segmentation model for fatigue crack detection proposed in this paper is shown in Figure 2. First, based on the ResNet50 network [10], the image sample dataset is used for training the classification network ResNet50_crack. Compared with ResNet50, ResNet50_crack captures crack features and facilitates semantic segmentation network training in the next step. Therefore, the DeepLabv3+ semantic segmentation network, with ResNet50_crack as the backbone network, can treat crack semantic segmentation in complex backgrounds with smaller generalization errors and more accurate results. Active learning [11,12] is employed to screen more valuable training images. The new dataset is chosen to train the semantic segmentation network DeepLabv3+, and finally, the category threshold of the model is optimized, as shown in Figure 2. The following section will describe in detail the method of dividing the dataset, the screening method for effective samples, and the training of the classification network ResNet50_crack and the semantic segmentation network DeepLabv3+.

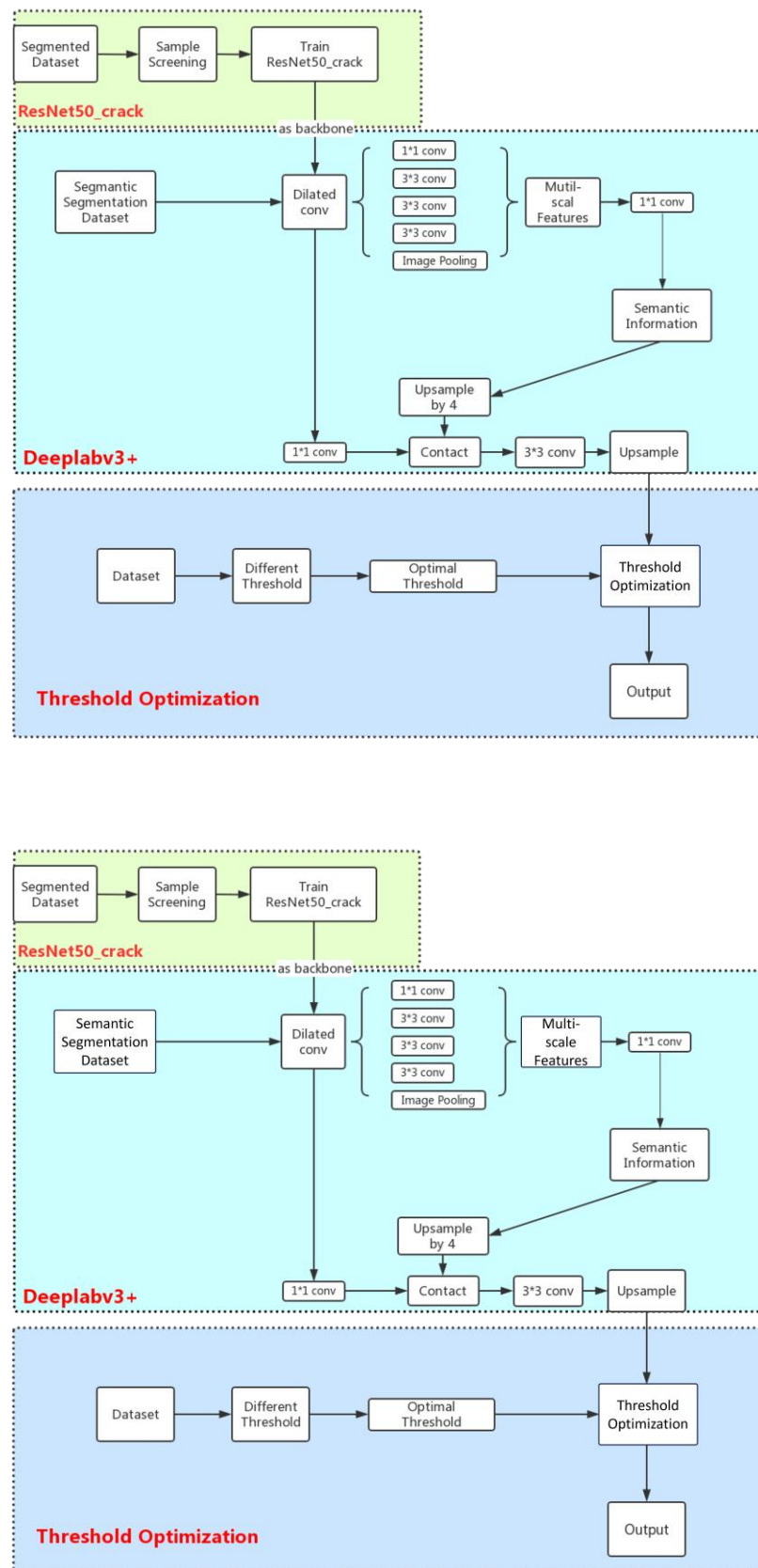


Figure 2. Fatigue crack detection model.

3.2. Classification Network ResNet50_crack

(1) Initial dataset for classification network

The original image dataset includes 120 pictures with a size of 4928×3264 pixels or 5152×3864 pixels. A total of 100 images are randomly selected for training, while the remaining 20 images are used for testing the network. Generally, the image size has a certain impact on the training results of the deep learning model [13,14]. Subsequently, these large images and ground truths are cropped into samples with a size of 224×224 pixels, resulting in 32,128 small image samples and corresponding ground truths. However, most of these samples are intact and less informative for crack detection. For improving the computation efficiency of the training process, these intact samples can be removed from the dataset. Therefore, pixel variances in samples are evaluated and sorted, and then the 60% of samples with the smallest variance are discarded. Subsequently, samples are classified into crack and background datasets according to their ground truth. This process is schematically shown in Figure 3a.

(2) Image screening based on active learning

The concept of active learning [11,12] originally referred to the case of semi-supervised learning (that is, a part of the data would be labeled, with the remaining parts unlabeled), where the algorithm could actively select the samples that were more informative and representative. These samples were artificially labeled and then added to the dataset for training. This paper adopts this idea of active learning to screen the most important samples (namely, those with larger cross-entropy) for re-tuning the classification network ResNet50_crack. The specific process is as follows, as shown in Figure 3b.

(3) Training classification network ResNet50_crack

Dataset 2 in Figure 3c is randomly divided into two sets, namely, the training and validation sets. This ensures that the training set and the validation set are consistent in distribution. The fully connected layers of ResNet50 for 1000 categories are modified for 2 to accommodate this crack classification. The random gradient descent method is adopted to train the network, with an initial learning rate of 0.001, a maximum epoch number of 20, and a mini-batch-size of 32.

The tuning process for the classification network to obtain ResNet50_crack is depicted in Figure 3.

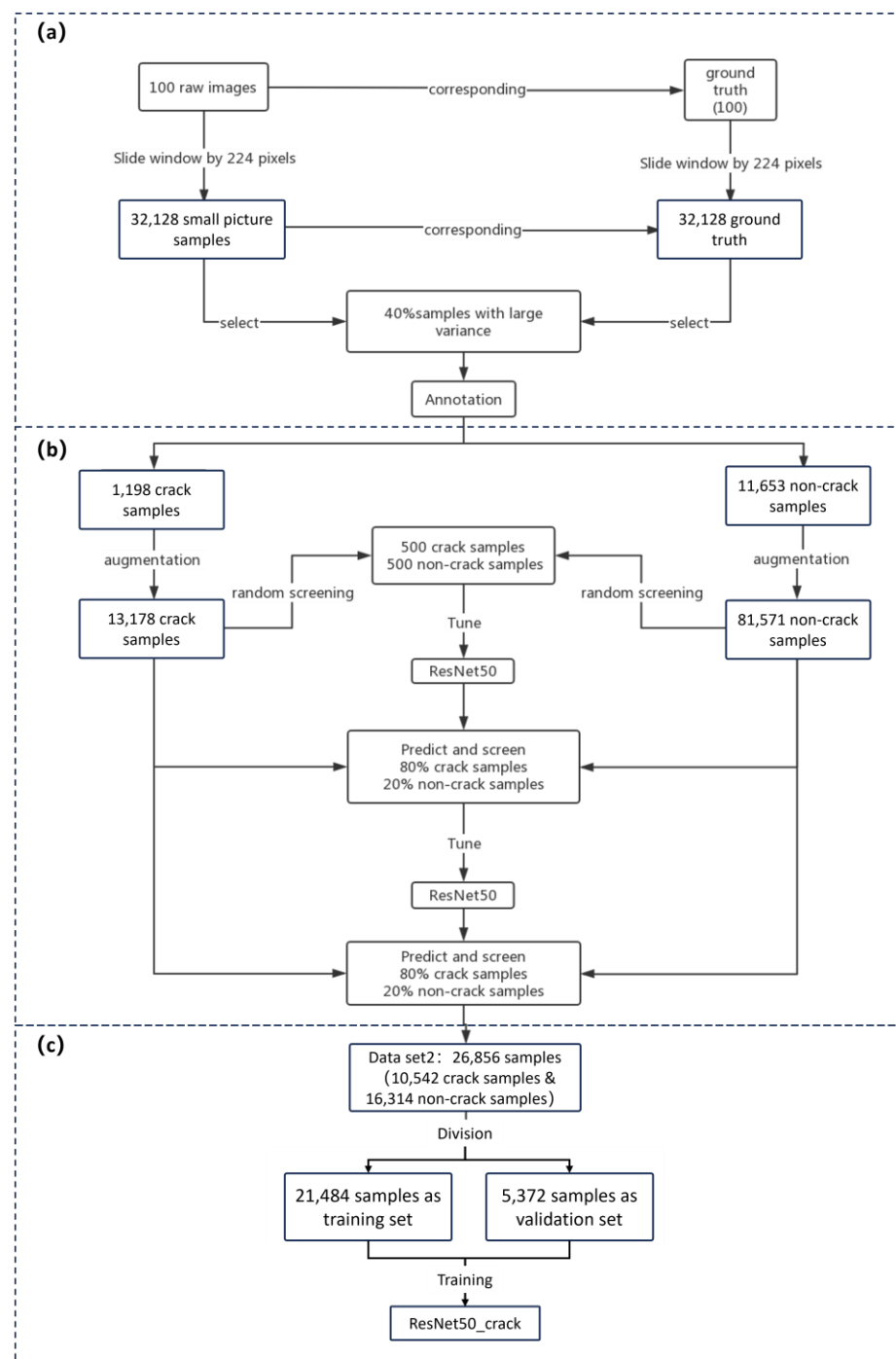


Figure 3. Tuning process of classification network ResNet50_crack. (a): dataset construction; (b): active learning; (c): tuning process for ResNet50_crack.

3.3. Semantic Segmentation Network DeepLabv3+ for Crack Detection

The training process for the semantic segmentation network DeepLabv3+ is shown in Figure 4. Since this figure is very clear, the description here is omitted.

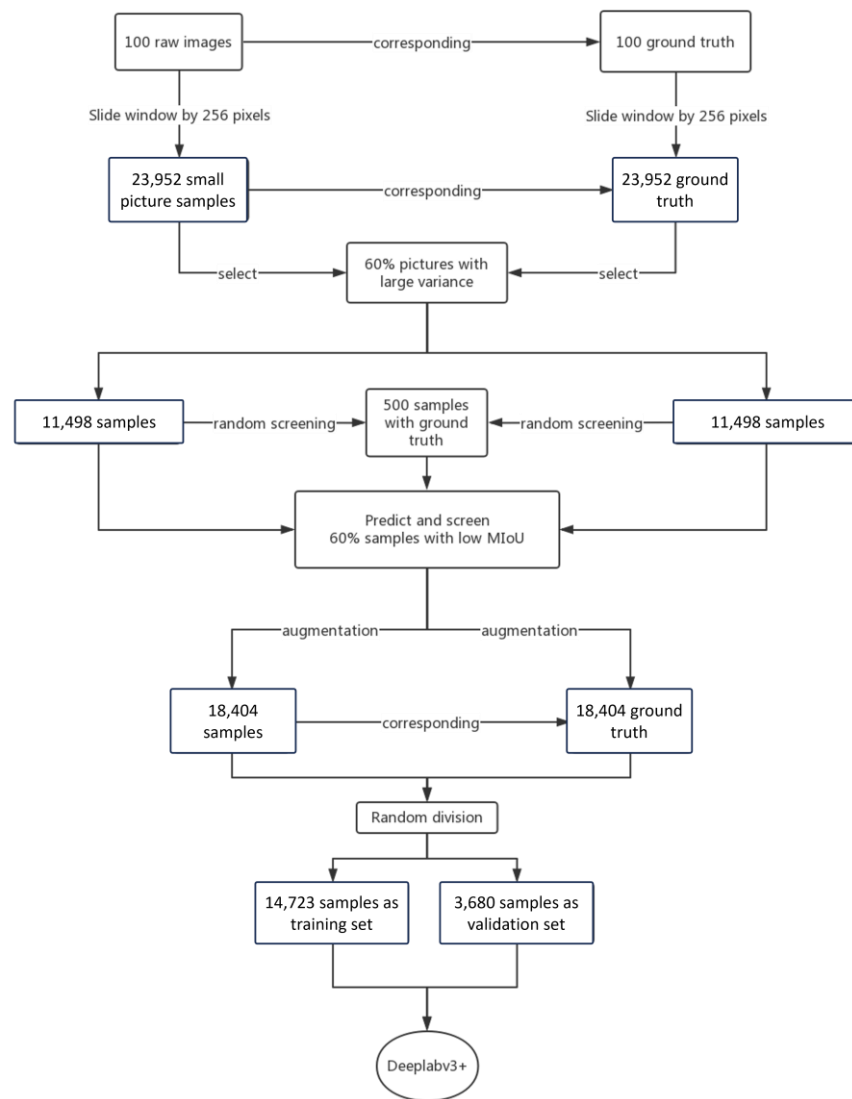


Figure 4. The training process of the semantic segmentation network Deeplabv3+.

4. Comparative Studies of Different Models

This section describes the comparison results of different models. In order to verify the effectiveness of the improved strategy proposed in this paper, we conducted comparative experiments based on the Deeplabv3+ algorithm. The first and second models are the DeeplabV3+ with ResNet50 and mobilenet-v2 [15] as backbone networks. They are compared with the proposed one. It should be noted that the difference between the second network and the proposed one is that the proposed one is based on ResNet50, which is re-tuned as a classification network using a crack detection dataset. Every model was trained for 20 epochs.

This comparison is carried out on the test dataset, which contains 20 raw pictures. All investigations are performed on a Lenovo workstation P910, which is installed with Nvidia GeForce, Santa Clara, CA, USA, GTX 1080 Ti, and MATLAB 2020a.

The training results are shown in Table 1. The lowest training accuracy is seen in mobilenet-v2 and is 99%, and the highest is ResNet50_crack, which is 99.26%. Compared with mobileNet-v2, ResNet50_crack and ResNet50 are deeper and then have stronger generalization ability for samples with complex backgrounds and finer cracks. Compared with ResNet50, ResNet50_crack has undergone crack classification pre-training, and thus the training of the semantic segmentation network is easier and has higher accuracy. Table 1 shows that the training process is slow but acceptable.

Table 1. Statistics for network training.

Network Type	Time (h)	Training Accuracy of Crack Pixels	Training Accuracy of Background Pixels	Global Accuracy	Validation Accuracy
Model 1: DeeplabV3+, ResNet50	18.9	78.52%	99.74%	99.71%	99.11%
Model 2: DeeplabV3+, MobileNet-v2	14.7	70.09%	99.67%	99.64%	99.00%
Proposed Model: DeeplabV3+, ResNet50_crack, Active Learning	12.25	75.59%	99.89%	99.86%	99.26%

(1) Comparison of semantic segmentation indicators

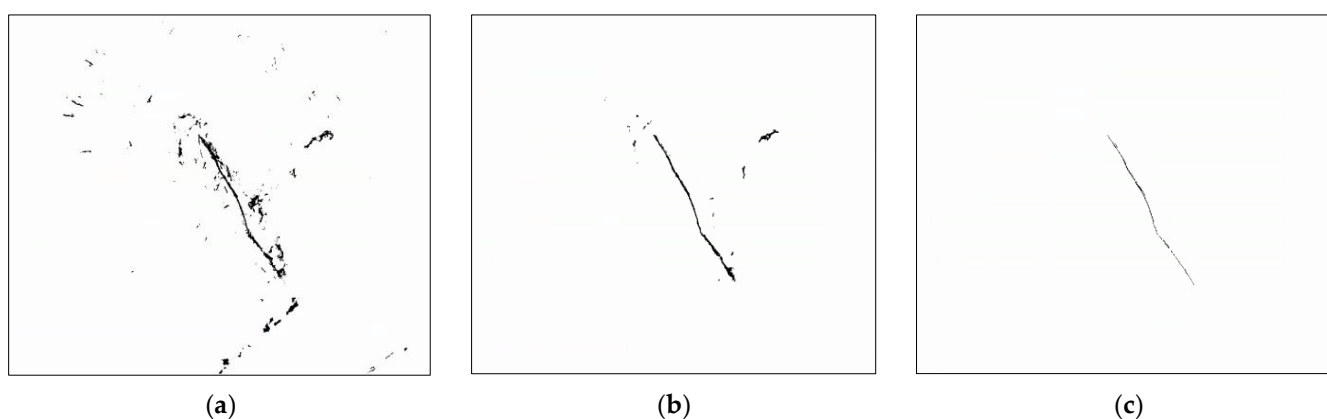
In order to measure the role and contribution of the segmentation system, its performance needs to be rigorously evaluated. Moreover, the evaluation must use standard and recognized methods to ensure fairness. Therefore, the commonly used indicators of semantic segmentation usually include: accuracy, intersection ratio (IoU), and average intersection ratio (MIoU).

The performance indicators of three models for the test dataset are summarized in Table 2. Clearly, the semantic segmentation network based on MobileNet-v2 performs the worst in terms of IoU for crack pixels and MIoU for the model. Conversely, the proposed model outperforms the others, which can be attributed to re-tuning the classification network ResNet50 and active learning.

Table 2. Semantic segmentation indicators for the test dataset.

Network Type	IoU		MIoU
	Crack	Background	
Model 1: DeeplabV3+, ResNet50	0.2391	0.9971	0.6181
Model 2: DeeplabV3+, MobileNet-v2	0.1910	0.9964	0.5937
Proposed Model: DeeplabV3+, ResNet50_crack, Active Learning	0.3897	0.9986	0.6942

The crack distribution predicted by Model 1 and the proposed model are compared in Figure 5 together with the ground truth. Clearly, the proposed model can provide results that are more consistent with the ground truth.



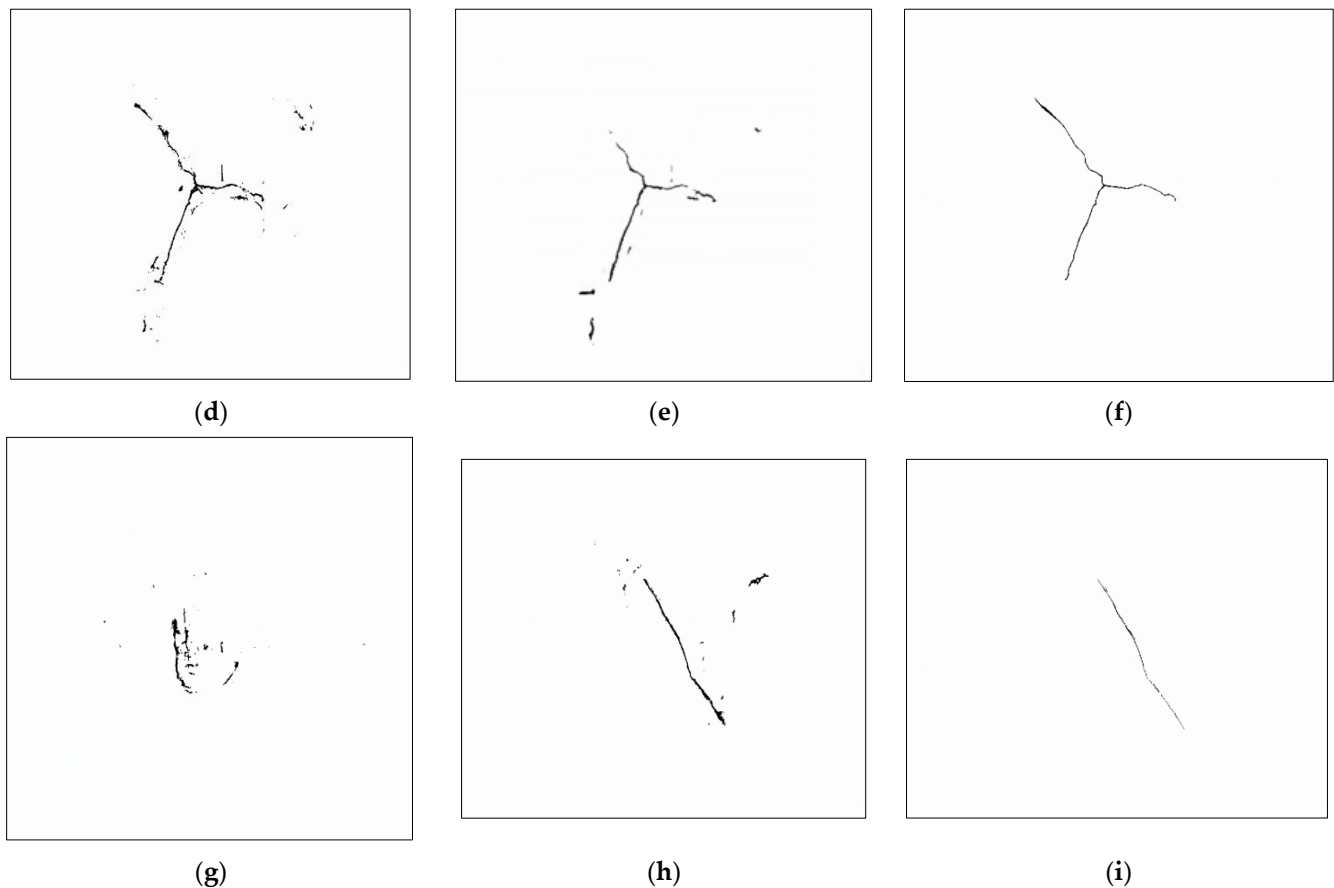


Figure 5. Cracks predicted by different models: (a,d,g) Model 1; (b,e,h) proposed model; (c,f,i) ground truth.

5. Pixel Category Threshold Optimization

The abovementioned analysis shows that the proposed model tends to classify more pixels as crack ones, indicating a relatively good classification accuracy and poor IoU for the crack pixels. Therefore, in order to further improve the effect of semantic segmentation, this paper sets the crack pixel category determination threshold to be greater than 0.5.

Prior to setting this appropriate threshold, one must examine whether the relationships between the model performance and category threshold exhibit similarity for the training and test datasets. This similarity implies its generalization; an optimal threshold chosen from the training dataset can be adopted for the test dataset. To reveal this, the classification accuracy and IoU for the crack pixels and the MIOU for the model are evaluated with different thresholds using the training and test datasets, which are depicted in Figure 6. It can be observed that (i) the performance indicators for the training dataset are always superior to those for the test dataset; (ii) the ideal similarity exists between two datasets, and thus good generalization is expected for this threshold; (iii) as the threshold increases, the accuracy of the crack pixel detection always decreases, whereas the IoU and the MIOU increase with the threshold < 0.9 and decline with the threshold > 0.9 . Actually, a larger threshold means that fewer pixels are classified as crack, indicating fewer true positive crack pixels and lower accuracy. As defined in Equation (1), the IoU increases in that its denominator is effectively reduced. Notably, the threshold step herein is 0.05 to save computation time.

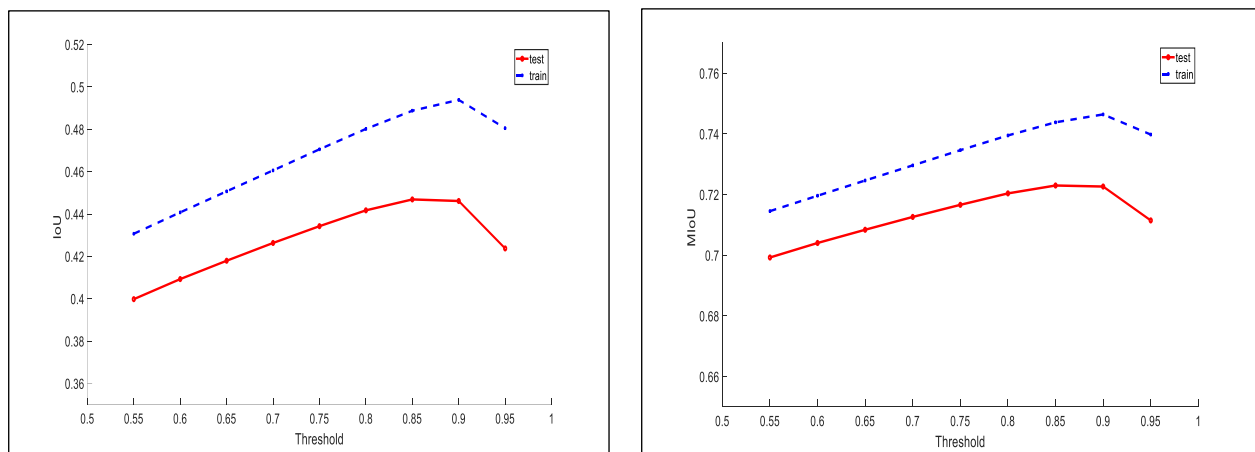


Figure 6. Semantic segmentation indicators with different category thresholds.

It can be seen from Figure 6 that the maximum values of the IoU and MIoU must correspond to a threshold around 0.9. In order to precisely identify the optimal threshold, a smaller sampling step, namely, 0.01, is chosen and adopted for analysis, with the results plotted in Figure 7. The results show that when the threshold is 0.89, the highest MIoU is 0.7466 for the training dataset and 0.7231 for the test dataset. This means that 0.89 is the optimal threshold value.

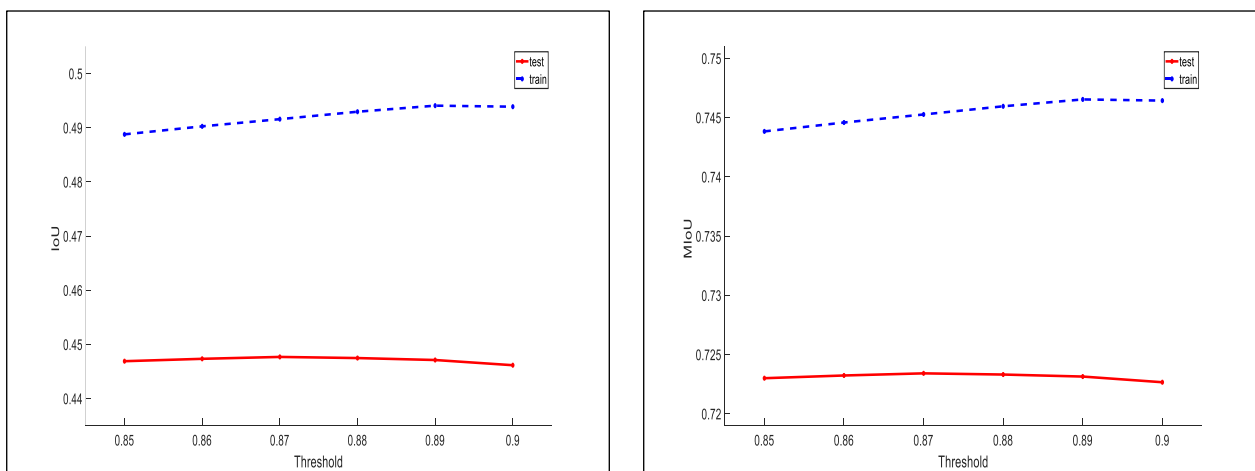


Figure 7. Semantic segmentation indicators around the indicator peak.

6. Fatigue Crack Detection Results Using the Proposed Model

Since fatigue cracks in steel box girders in long-span bridges are slender, we use the original map, ground truth map, and prediction map to visualize the results of crack detection, as shown in Figure 8. It can be seen that the predicted results are considerably consistent with the ground truth, although the raw pictures are very complex in terms of detection owing to various disturbances. This illustrates the favorable performances of the proposed model.

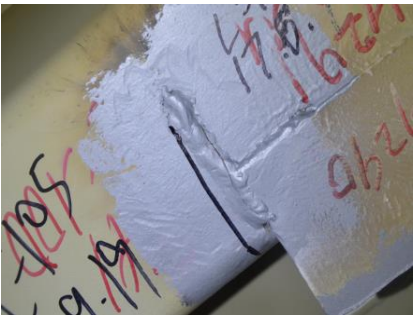
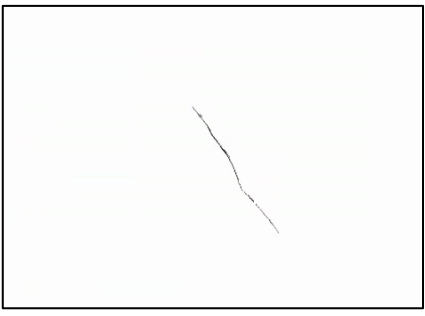
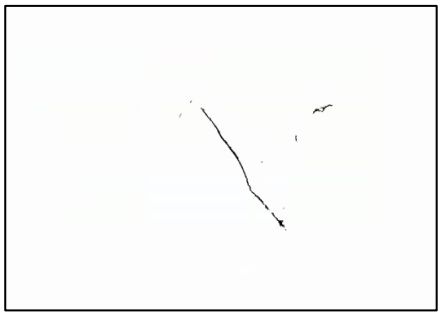
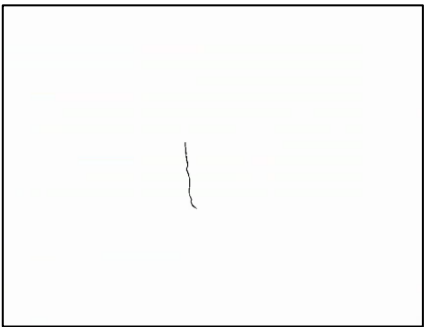
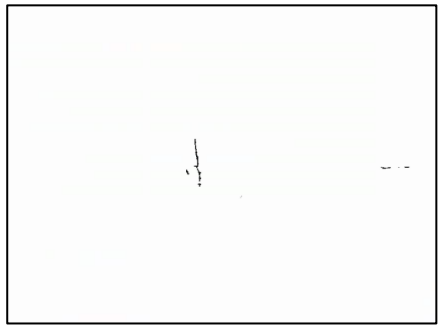
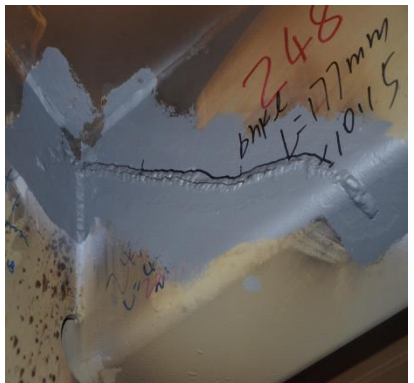
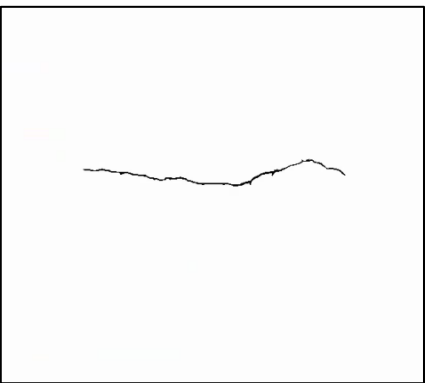
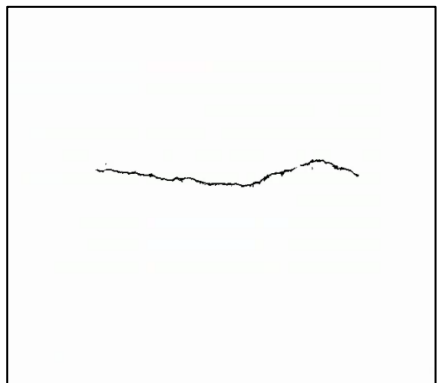
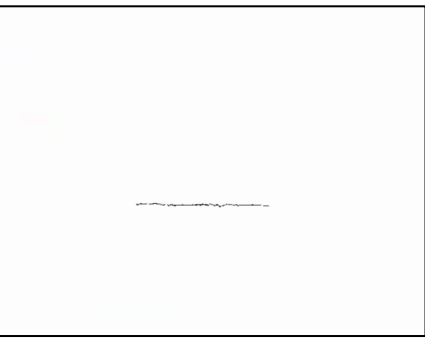
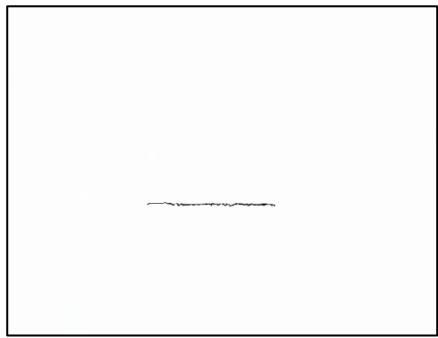
		
		
		
		
Raw pictures	Ground truth	Prediction

Figure 8. Selected fatigue crack detection results using the proposed model.

7. Conclusions

This paper presents a semantic segmentation network model for fatigue crack detection based on DeepLabv3+, ResNet50, and active learning. The classification network ResNet50 is re-tuned using the crack image dataset, leading to ResNet50_crack, which is adopted as the backbone network for DeepLabv3+ to construct a crack semantic segmentation network. This network was trained with the assistance of active learning, followed by the optimization of the probability threshold of the pixel category. Compared with the crack detection network based on conventional ResNet50, this model can improve MIOU from 0.6181 to 0.7241. In future research, we are to further improve the detection accuracy

of the model for small cracks and to achieve the automatic and fast calculation of crack width, length, and other information based on the Transformer network.

Author Contributions: Conceptualization, X.J.; Writing—original draft, Y.W.; Writing—review and editing, X.J. and Z.W. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: The datasets presented in this article are not readily available because the data are part of an ongoing study.

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

References

1. Zhang, Y.; Li, H.; Wang, Y. Fatigue Crack Detection in Steel Beams Using Support Vector Machines. *J. Struct. Health Monit.* **2020**, *15*, 234–246.
2. Li, Z.; Zhao, J.; Chen, Q. K-Nearest Neighbors-Based Crack Detection Using Strain Data from Steel Girder Bridges. *Struct. Control Health Monit.* **2019**, *26*, e2467.
3. Kim, S.; Park, J.; Choi, J. Drone-Based Crack Detection in Steel Bridges Using Convolutional Neural Networks. *Autom. Constr.* **2021**, *126*, 103675.
4. Yang, Y.; Yao, X. Multi-Scale Convolutional Neural Network for Crack Detection in Steel Bridges. *Eng. Struct.* **2023**, *259*, 114245.
5. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLabv3+: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848.
6. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Tel Aviv, Israel, 23–27 October 2022; Springer: Cham, Switzerland, 2019; pp. 801–818.
7. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*; Springer: Cham, Switzerland, 2015; Volume 9351, pp. 234–241.
8. Xu, K.; Zhang, J.; Li, W. Transfer Learning for Crack Detection in Steel Bridges. *Comput.-Aided Civ. Infrastruct. Eng.* **2023**, *38*, 265–278.
9. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 15 October 2015; pp. 3431–3440.
10. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 12 December 2016; pp. 770–778.
11. Settles, B. *Active Learning Literature Survey*; Computer Sciences Technical Report 1648; University of Wisconsin-Madison: Madison, WI, USA, 2009.
12. Wang, Z.; Xu, G.; Ding, Y.; Wu, B.; Lu, G. A vision-based active learning convolutional neural network model for concrete surface crack detection. *Adv. Struct. Eng.* **2020**, *23*, 2952–2964.
13. Rukundo, O. Effects of Image Size on Deep Learning. *Electronics* **2023**, *12*, 985.
14. Rukundo, O. Evaluation of extra pixel interpolation with mask processing for medical image segmentation with deep learning. *Signal Image Video Process.* **2024**, *18*, 1–8.
15. Mark, S.; Andrew, H.; Menglong, Z.; Andrey, Z. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.