Smart Structures and Systems, *Vol. 35, No. 3 (2025) 163-175*
https://doi.org/10.12989/sss.2025.35.3.163

Lightweight deep learning-based automated concrete surface inspection: Crack classification and pixel-level segmentation

Dong Yang [1,2], Yuan-Yuan Cai [3], En-Dian Xu [3], Jing Zhang*[4], Ye Yuan [5] and Yan-Jia Wang [5]

[1] *Earthquake Engineering Research & Test Center (EERTC), Guangzhou University, Guangzhou, China*
[2] *Key Laboratory of Earthquake Resistance, Earthquake Mitigation and Structural Safety, Ministry of Education, Guangzhou University, Guangzhou, China*
[3] *Department of Civil Engineering, Hefei University of Technology, Hefei, Anhui Province, China*
[4] *Department of Mechanics and Construction Engineering, Jinan University, Guangzhou, China*
[5] *Department of Civil Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, China*

(*Received January 23, 2024, Revised April 2, 2025, Accepted April 10, 2025*)

Abstract. Crack detection is an important measure in the field of structural health monitoring. However, visual crack detection is labor-intensive, time-consuming, inefficient, and expensive. Although image-based detection and processing provides an efficient way for structural crack detection, its accuracy depends on image quality. For engineering structures, especially bridges, the change of light conditions and the difference of surface characteristics of structural components pose a major challenge to traditional crack detection methods. In this paper, a novel crack detection method based on convolutional neural networks is proposed. The development of this method is divided into the following stages. The initial automated crack classification is carried out by using MobileNetV3, and then the improved DeepLabv3+ network is used to segment the classified crack image semantically accurately. Finally, the real crack image is used for verification. To verify the proposed method, several conventional deep learning networks are trained and compared. The improved DeepLabV3+ integrates MobileNetV3 as its feature extraction backbone and incorporates the convolutional block attention module, which achieves 87.79% average intersection and 93.87% average pixel accuracy on public and real data sets. Compared with traditional models such as VGG16, the proposed method shortens the training time by more than 80% while maintaining high detection accuracy. In addition, the compact parameter configuration and moderate model size make it particularly suitable for deployment on mobile detection devices.

Keywords: attention mechanism; crack detection; improved DeepLabv3+; lightweight deep learning; semantic segmentation

1. Introduction

Traditional maintenance strategies for structural crack detection involve labor-intensive and costly on-site inspections (Gribniak *et al*. 2008, Tang *et al*. 2022). Assessing hard-to-reach areas, such as the bottom of the girder, adds further cost, danger, and complexity (Dorafshan and Maguire 2018, Gucunski *et al*. 2014, Qi *et al*. 2021). While some bridges incorporate structural health monitoring systems with sensors, manual inspections are still critical and require proper training on equipment operation (Dan and Dan 2021). Consequently, the quality of the inspection depends on environmental factors and the professional knowledge of the inspectors (Agdas *et al*. 2016, Ye *et al*. 2022). Therefore, the automatic crack detection system via mobile devices has become a promising solution, which provides economical, safe and consistent performance. In places where manual inspection is difficult to reach, this automatic crack detection system can be deployed to mobile devices to achieve real-time detection and improve detection efficiency.

In recent years, computer vision has gained widespread adoption in the detection of concrete surface defects (Wang and Xiang 2021). To address the limitations of artificial concrete crack detection, the combination of crack image capture and image processing algorithms has become the main method for safely and effectively identifying cracks in concrete structures. These algorithms include matched filtering (Yeum and Dyke 2015), threshold segmentation (Moreau *et al*. 2021) and edge detection (Tian and Wei 2022). Yan *et al*. (2007) adopted a specifically designed median filter for image enhancement and used an edge detection algorithm based on grayscale morphological operator for crack detection. This approach successfully extracted the edge of pavement crack, minimizes the image noise, and accurately captures the characteristics of pavement cracks. Akagic *et al*. (2018) introduced a pavement crack detection method using gray histogram and Ostu threshold algorithm, which can show satisfactory performance even under low signal-to-noise ratio conditions. Song *et al*. (2016) proposed an adaptive Canny algorithm and iterative threshold segmentation method for surface crack detection, overcoming the limitations of the

*Corresponding author, Ph.D., Professor,
E-mail: zhangjing@jnu.edu.cn

ISSN: 1738-1584 (Print), 1738-1991 (Online)

traditional Canny algorithm based on the adaptive parameters of the actual image. The method can effectively preserve the crack edge and eliminate the noise. Although these traditional image processing methods have achieved good segmentation results, they are very sensitive to image noise and difficult to distinguish cracks from backgrounds. Furthermore, their application in automatic crack detection is also facing challenges. Most traditional algorithms are easily affected by light conditions and oil stains, and require a lot of pre-processing and post-processing, and even manual intervention, to improve the detection accuracy. Usually, the involvement of adjustable parameters is required, which vary from image to image, making full automation difficult. Therefore, traditional algorithms face considerable obstacles in realizing automatic crack detection in practical engineering scenarios.

In order to realize automatic crack detection in practical engineering, the computer learning method based on supervised learning has successfully realized automatic crack detection. Kanaeva and Ivanova (2021) designed a shallow coding network to extract crack image features through statistical crack analysis, and integrated an attention module to enhance context relevance. The results show that the quantification error is less than 4%. Dung (2019) studied deep convolutional neural networks, and explored the image classification and boundary box techniques in vision-based automatic detection of concrete cracks. A complete encoder-decoder FCN network with a VGG16-based encoder is employed for semantic segmentation, with an accuracy of 90%. Lee *et al*. (2020) introduced a shape-sensitive kernel in the semantic segmentation framework and combined it with an improved deep layer model for crack detection. The model can predict crack width more accurately than traditional semantic segmentation methods, with only one or two pixels. To further improve the accuracy and efficiency of crack image segmentation, a set of segmentation networks based on FCN and encoder structures have been developed, including U-Net (Du *et al*. 2020), PSPNet (Zhu *et al*. 2021) and DeepLabv3+ (Chen *et al*. 2018). Liu *et al*. (2019) used U-Net fully convolutional network to achieve automated crack detection. Ji *et al*. (2020) proposed a comprehensive method that used DeepLabv3+ to detect cracks and quantify various crack indicators which include crack length, average width, maximum width, area, and ratio. Fu *et al*. (2021) introduced an enhanced DeepLabv3+ for surface crack segmentation of bridge concrete which integrates a densely connected atrous spatial pyramid pooling module, enhancing the network's feature extraction capabilities. Experimental results show that the average intersection rate of the improved DeepLabv3+ algorithm is 82.37%, which is better than the original DeepLabv3+ algorithm.

Although the pixel segmentation networks have achieved impressive crack identification results, traditional methods are challenged due to operational factors such as geographical location and weather, changes in the size of cracks in structures and bridges, and variations in the surrounding environmental conditions of cracks. This complexity often leads to identification results below the standard. In order to improve the robustness of crack detection, the proposed method integrates two key strategies: adaptive data augmentation and attention mechanism. Initially, during model training, dynamic data augmentation techniques such as random brightness adjustment (±30% of the original intensity) and contrast normalization are used to simulate different lighting scenarios, including shadows and strong sunlight. This enables the model to learn invariant features of lighting changes. Secondly, the convolutional block attention module (CBAM) is added into the segmentation network to minimize the interference of background noise factors such as water stains or uneven surfaces. CBAM sequentially applies channel-wise and spatial attention maps to dynamically emphasize critical crack features while suppressing interference from complex backgrounds.

In addition, traditional deep learning networks often have complex and large architectures, resulting in high parameter counts and large computational requirements. These complex models encounter limitations related to hardware capabilities and cannot meet the demands of high responsiveness and low latency. To this end, the lightweight network MobileNetV3 (Koonce 2021) was used for crack classification, addressing the need for efficiency. Furthermore, on the basis of the original DeepLabv3+, MobileNetV3 is used instead of Xception (Rahimzadeh and Attar 2020) to enhance the feature extraction backbone of the crack segmentation model. This strategic replacement significantly reduced training time and computational requirements. However, although the modified DeepLabv3+ model improves the training speed, its segmentation accuracy still lags behind some typical deep learning segmentation models. To solve this problem, a neural

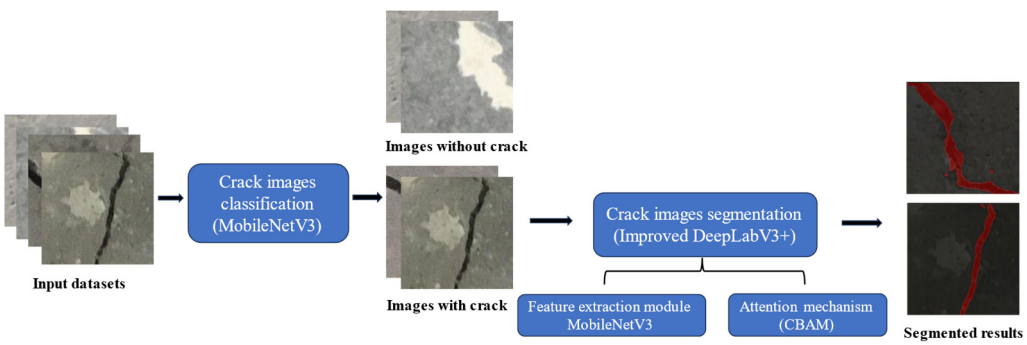


Fig. 1 Proposed crack detection and segmentation framework for the input data set

network attention mechanism named CBAM is proposed in this paper to improve the accuracy of the crack segmentation. The schematic overview of the proposed crack detection and segmentation framework is shown in Fig. 1.

The overall process of the study is as follows: Section 1 summarizes the development of crack detection and segmentation research. Section 2 introduces the principles of the lightweight network MobileNetV3 and the improved DeepLabv3+. In Section 3, two data sets are used for crack classification and pixel segmentation, and the implementation details of the training are described step by step. Section 4 illustrates the effectiveness of the proposed method through experiments and comparisons. In addition, some real crack images are tested to verify the accuracy of the training model in crack classification and pixel segmentation. Finally, Section 5 is the conclusion.

2. Methodology

In this section, the advantages of the lightweight network MobileNetV3 as a crack image classification network and the proposed improved Deeplabv3+ network for crack image pixel segmentation are introduced in detail. The combination of these two algorithms provides an effective crack detection approach and promotes fine automation. Firstly, the crack image classification algorithm can distinguish the images with cracks from those without cracks, which greatly reduces the complexities of crack pixel segmentation. Then the classified images containing the crack are input into the crack segmentation algorithm, which separates the crack from the background to calculate the size of the crack, such as length and width. In addition, in order to speed up the model training and optimize model size for mobile devices, the lightweight MobileNetV3 network serves as the backbone of the crack classification model. At the same time, an enhanced version of the original DeepLabv3+ network is introduced. The improved

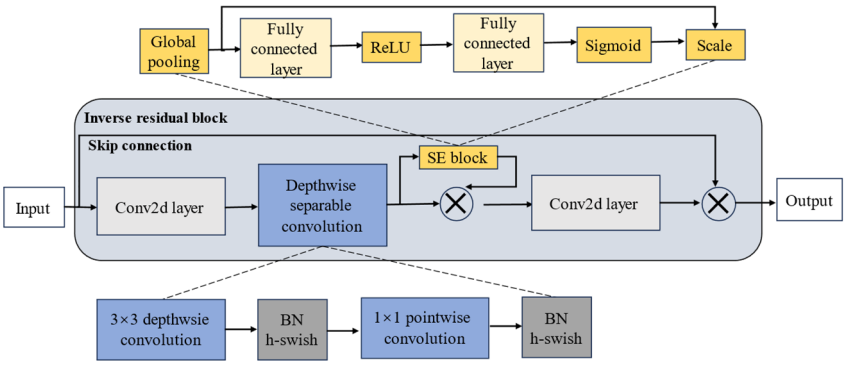


Fig. 2 Structure of MobileNetV3 blocks

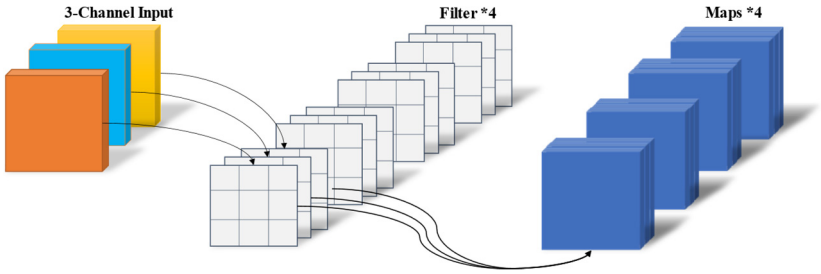


Fig. 3 Traditional convolution of 3-channel input

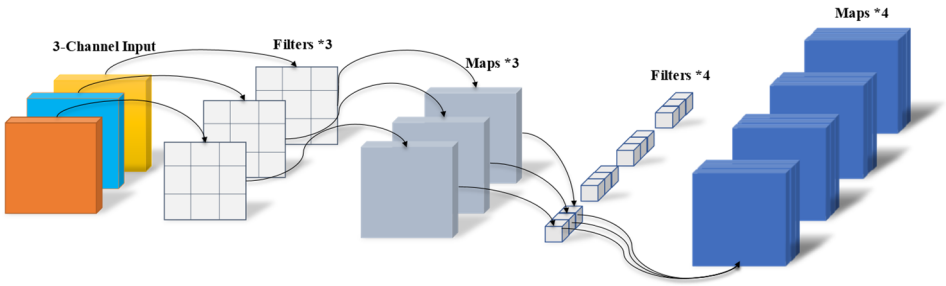


Fig. 4 Depthwise separable convolution of 3-channel input

model integrates the feature extraction module of MobileNetV3, replaces Xception, and significantly reduces the training time and computational amount of the model. Besides, the CBAM is adopted to improve the segmentation accuracy of the proposed framework.

2.1 MobileNetV3 lightweight network for crack image classification

MobileNetV3 is proposed to improve the efficiency and real-time performance of deep learning neural networks on hardware with limited computing power. The network reduces the parameter count while maintaining accuracy. The schematic diagram of the MobileNetV3 network is illustrated in Fig. 2, showing four unique structural attributes: depthwise separable convolution, inverted residual with a linear bottleneck, lightweight attention mechanisms achieved by squeezing and excitation, and integration of the h-swish function.

In each inverse residual structure of MobileNetV3, the traditional standard convolution is replaced by the introduction of depthwise separable convolution. This method consists of two steps: depthwise convolution followed by a point-by-point convolution, as shown in Fig. 3. Unlike the traditional convolution calculation, the depthwise separable convolution divides the calculation into two different steps, as depicted in Fig. 4. In the first step, the input data is convolved with a depthwise convolution kernel to generate a set of feature mapping channels equal to the number of input channels. Subsequently, the point-to-point convolution uses a single convolution kernel to merge feature maps from the initial step, producing new correlated feature maps. This decomposition greatly reduces both the computational requirements and model size. Furthermore, previous research has shown that MobileNetV3 achieves comparable classification accuracy to Vgg16 on ImageNet, while the computational cost and model size are only 1/30 of the former under the same dataset (Zhang *et al*. 2023).

In the pursuit of extracting features from high-dimensional space while minimizing information loss, linear bottlenecks are introduced to compress the dimension of the input. Recognizing that the traditional ReLU function transformation introduces nonlinearity and potential information loss, MobileNetV3 takes a different approach, instead of using the traditional ReLU function, it incorporates inverted residuals with linear bottlenecks into its convolutional blocks. As shown in Fig. 2, unlike traditional residual blocks (such as the 'compress-extract-expand' architecture used in ResNet), the reverse residual strategy uses the 'expand-extract-compress' method. This starts from an expansion layer, which uses two-dimensional convolution to increase the input channels into a high-dimensional space, thereby enhancing the feature representation ability. After that, the extraction layer independently applies a 3 × 3 depthwise separable convolution to each input channel, and then uses a 1 × 1 convolution to integrate features between channels. This process extracts spatial features while minimizing computational complexity. The features from the extraction layer are further refined by an SE (squeeze-and-excitation) module, which strengthens the network's attention on critical features. The output of the SE module is multiplied by the output of the depthwise separable convolution by elements. Then, the improved features are implemented as another two-dimensional convolution through the compression layer, and merged with the initial input feature map by skipping the connection. By extending the feature dimension before performing complex transformations in low-dimensional space, this method preserves finer details, which is particularly important for detecting microcracks. In addition, to improve feature extraction accuracy while optimizing computational resources, MobileNetV3 uses the ReLU6 function as an approximation of the sigmoid function in Swish. This adjustment produces an approximate Swish called the "hard version of Swish" (h-swish). Compared with ReLU's sharp truncation of the negative region so that the gradient is zero, the h-swish function maintains a smooth gradient feature. This method helps to maintain a small gradient in the negative interval, thereby reducing the risk of gradient disappearance in the deep networks. In addition, it limits the numerical range to [0, 6], making it very suitable for low-precision inference on mobile devices. The mathematical expression for h-swish is as follows

$$\mathrm{h-swish} = x \cdot \frac{\mathrm{ReLU6}(x+3)}{6} \tag{1}$$

where x is the output of the previous layer, ReLU6 is an activation function, which is a variant of the rectified linear unit (ReLU), limiting the output values to the range of [0, 6]. Mathematically, it can be defined as: ReLU6(z) = max(0, min(z, 6)).

2.2 Improved DeepLabv3+ for crack image segmentation

DeepLabv3+ represents an encoder-decoder architecture tailored for semantic segmentation. Evolving from DeepLabv3, it introduces a streamlined and powerful decoder module to refine segmentation results, especially along object boundaries. This refinement is enhanced by incorporating the atrous spatial pyramid pooling (ASSP) module and the fusion of encoding and decoding structures, resulting in better segmentation results. The feature extraction backbone of the original DeepLabv3+, including Xception and similar deep convolutional neural networks, is primarily used for image classification and detection. However, due to the expansion of receptive field and channel interactions, the performance of object detection and image semantic segmentation is not satisfactory. Notably, these networks are burdened with a large number of parameter counts and significant model scales, exacerbating hardware limitations and associated computational challenges. The escalating model depth exacerbates these concerns. To solve the problems, the lightweight network MobileNetV3 is adopted in this paper as the backbone of the feature extraction in the crack segmentation model, replacing Xception in the original DeepLabv3+. Compared with the DeepLabv3+ initial backbone network, MobileNetV3 has the characteristics of shallower layers, fewer calculation parameters, and lower model complexity. Therefore, the network training speed

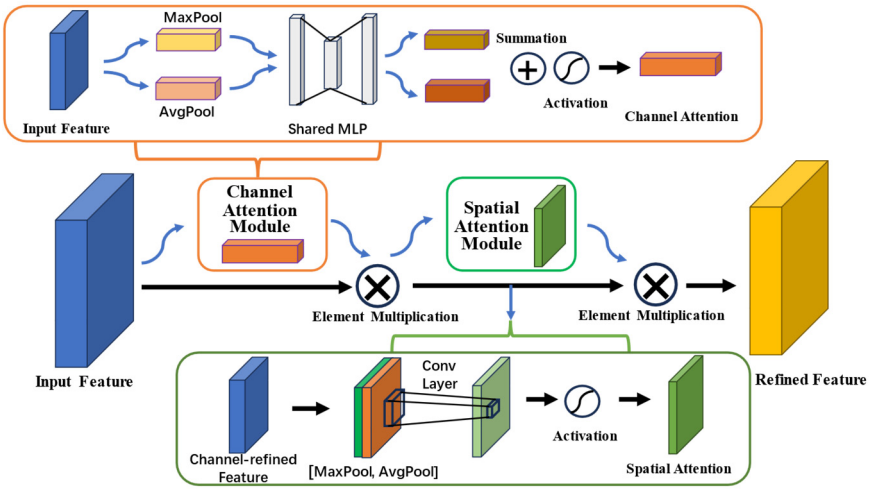


Fig. 5 Network architecture of convolutional block attention module

and the model convergence speed are accelerated. The advantages of the MobileNetV3 backbone network are described in detail in Section 2.1.

Furthermore, the improved DeepLabv3+ incorporates a neural network attention mechanism called CBAM (Liang *et al*. 2022) to enhance the ability to extract complex edge features from recognized objects. In neural networks, the attention mechanism enables the models to assign different weights to different input components, thus capturing key and critical information. This enhancement improves recognition accuracy without increasing model complexity and computational requirements. As shown in Fig. 5, CBAM embodies a simple and efficient feedforward convolutional attention module neural network. It infers an attention map along the channel and spatial dimensions in turn. The attention map is then multiplied by the input feature map to promote adaptive feature refinement. The structure of CBAM is multifaceted, including the channel attention module and spatial attention module. The channel attention module includes global maximum pooling and average pooling for feature mapping of different channels, thereby generating a maximum pool channel attention vector. Then, the vector is processed by a multi-layer perceptron (MLP) with a hidden layer, and finally the channel attention map is generated by vector addition and activation through the sigmoid function. For the spatial attention module, two 2D feature maps are generated from different pooling operations and channel information is transmitted from the feature map. The generated maps are combined and convolved by a conventional convolution layer to obtain a 2D spatial attention map.

In order to accelerate the training speed of crack segmentation model training and meet the requirement of fast response and minimum delay in automatic crack identification, the enhanced DeepLabv3+ is introduced in this paper. It uses MobileNetV3 as the feature extraction module and integrates the CBAM in the network structure to improve the accuracy of the model in crack segmentation. The architectural layout of the improved DeepLabv3+ is shown in Fig. 6. The process starts with the utilization of the MobileNetV3 backbone network, which effectively captures multi-scale information by manipulating feature resolution and filtering the field of view through different convolution kernel sizes. MobileNetV3 is specifically designed to improve efficiency, using depthwise separable convolutions and lightweight building blocks such as reverse residual structures with linear bottlenecks. This configuration helps to extract features quickly without affecting the basic representation ability. In order to further enhance the model's attention to crack-specific features, CBAM is strategically integrated into the MobileNetV3 backbone, usually after key convolutional layers. CBAM consists of two consecutive sub-modules, the channel attention module (CAM) and the spatial attention module (SAM). The former recalibrates the importance cross-channel features, and the latter emphasizes spatially related regions such as crack edges and textures. This interaction enables MobileNetV3 to prioritize and refine the most salient features for crack detection, effectively making up for the potential limitations of its lightweight design in capturing fine-grained details. After feature extraction, the ASPP mechanism is employed to reduce the dimension of feature map to capture the complex and multi-scale details of the target object features, such as the change of crack width and direction. This step provides comprehensive low-level and high-level functions as input of the decoding module. In the decoder module, a 4-fold bilinear upsampling is performed to deconvolution the input feature map. The subsequent output is then concatenated to the corresponding low-level features from the MobileNetV3 backbone, which are enriched by CBAM's attention-guided refinements. This concatenation facilitates the gradual recovery of the feature maps to their initial spatial dimensions, so that the crack area can be refined and accurately semantically segmented.

The rationale for combining MobileNetV3 and CBAM in this enhanced DeepLabV3+ architecture lies in their complementary advantages, which makes them particularly suitable for crack segmentation tasks. Crack segmentation requires both computational efficiency, which is necessary for real-time or near-real-time processing in practical applications such as infrastructure monitoring, and high

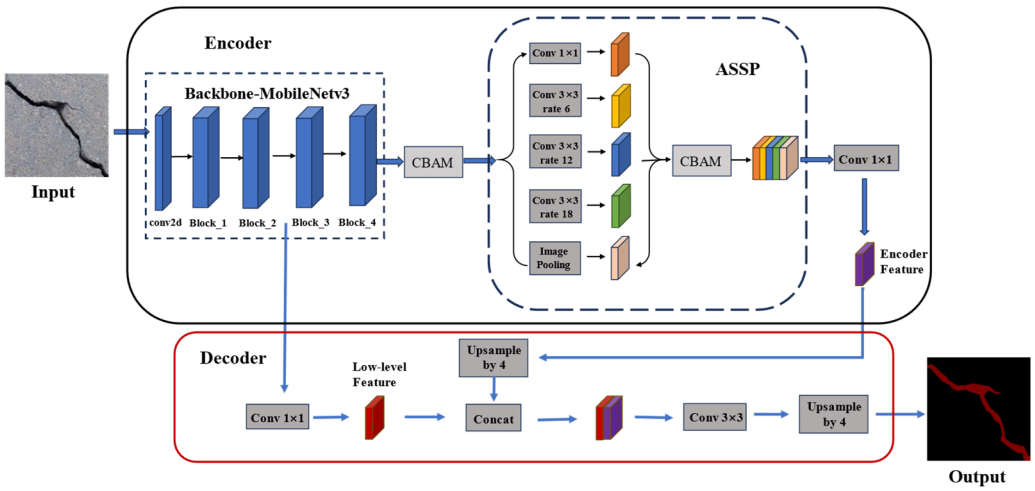


Fig. 6 Network architecture of improved DeepLabv3+

accuracy to detect subtle and irregular crack patterns in noisy backgrounds. MobileNetV3 addresses the efficiency requirements by providing a lightweight but powerful backbone. Compared with more substantial models like ResNet, it significantly reduces computational complexity and inference time. However, its focus on efficiency may impair its ability to capture complex details. CBAM solves this limitation by introducing the attention mechanism which can improve the sensitivity of the model to crack-specific features, thereby enhancing the segmentation accuracy without significantly increasing the computational requirements. Therefore, this combination achieves a balanced trade-off, MobileNetV3 ensures fast processing, while CBAM improves feature representation, enabling the model to accurately depict fine crack boundaries and distinguish them from similar textures.

3. Model training and validation

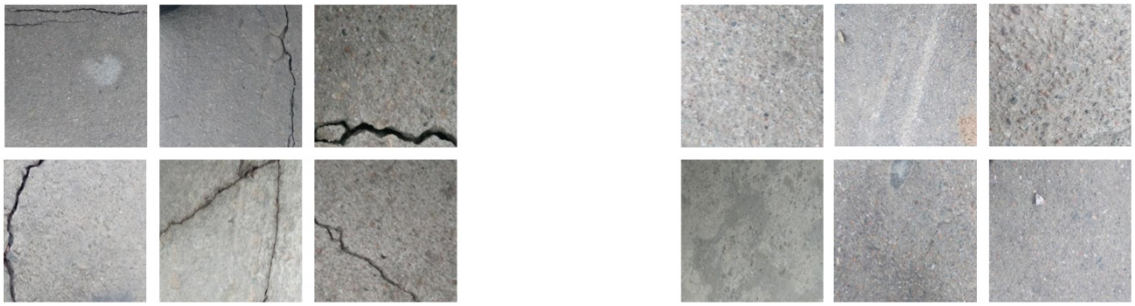
3.1 Crack image dataset

The collection and preparation of dataset is the key stage of crack classification and semantic segmentation in deep learning models. The selected crack images are crucial for both training and testing of the model, and they are extracted from the publicly available and real crack image datasets (Yang *et al*. 2019). The dataset contains 2069 bridge crack images with a resolution of 1024 × 1024 per image. In order to optimize the process, the 1024 × 1024 image is divided into four 512 × 512 images. This enhances the sample diversity and is consistent with the MobileNetV3 input size of 224 × 224, while avoiding the loss of details usually associated with direct downsampling. The result of this step is the creation of a comprehensive dataset consisting of 4055 images filled with cracks and 2014 images without cracks, which is achieved by filtering out fuzzy representations. In order to meet the input specification of the proposed network, the crack images are subjected to a random center clipping operation, which is restricted to a size of 224 × 224 pixels. Despite the preprocessing measures, as shown in Fig. 7, the crack image dataset retains attributes such as bridge shadows, water stains, and intense lighting.

The difference in labeling between the dat'sets of the crack image classification and pixel segmentation modules requires the creation of two different datasets. Initially, 6069 images from the public dataset are randomly divided into two parts: 4856 images for model training and 1213 images for model validation, maintaining a 4:1 ratio. This division facilitates the subsequent classification process. Then, the open-source LabelMe tool (Russell *et al*. 2008) is adopted for semantic/instance annotation. For crack semantic segmentation, 500 crack images are randomly selected from the dataset and fine pixel annotation is performed during the manual labeling process. This process involves a comprehensive classification and labeling of cracks in the image. The obtained annotated image set, the original image set, and the labeled image set together constitute the dataset. Then the merged dataset is divided into a training set and a test set for observation in a ratio of 4:1. The labeled sample images are shown in Fig. 8, where the red region represents the crack in the image and the black region represents the background. Table 1 details the distribution of the training set and validation set of all datasets.

Table 1 Number of training set and validation set

Function	Number of images				
	Total samples	Training samples	Validating samples	With cracks	Without cracks
Crack classification	6069	4856	1213	4055	2014
Crack segmentation	500	4856400	100	500	0



(a) Crack images (b) Non-crack images

Fig. 7 Examples of images contained in crack classification

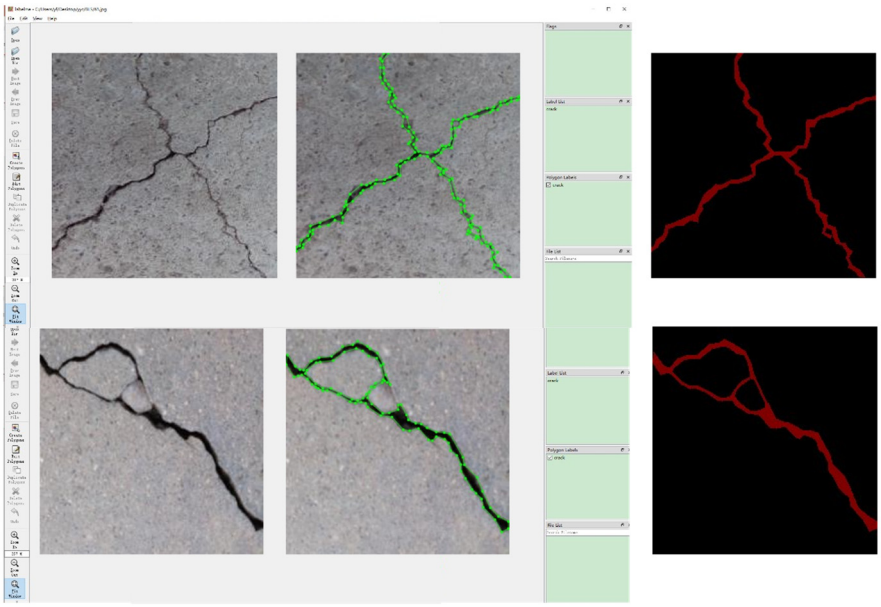


Fig. 8 Examples of collected images, image annotations, ground truths, and label visualization

3.2 Implementation details of training

The training is conducted on a workstation using an Intel (R) Core (TM) i5-8400HQ CPU @2.80 Hz, 2.81 GHz, and 16.0 G RAM, and an NVIDIA Quadro P4000, 8G GPU. The software configuration is as follows: Windows 10, CUDA 9.0, cuDNN 7.1, and Python 3.6, Keras 2.3.1, Tensorflow 2.1.0.

Firstly, the lightweight neural network MobileNetV3 is used for crack classification. In order to comprehensively measure the classification capability of MobileNetV3, its performance is compared with other respected and well-established deep CNNs, including VGG16 (Simonyan and Zisserman 2014), Inceptionv3 (Dong *et al*. 2020), Xception, and Resnet50 (Wen *et al*. 2020). These models are trained on the crack datasets, allowing for an in-depth evaluation of the accuracy, efficiency, and robustness of MobileNetV3. In order to train the optimal crack detection model based on these neural networks, a diligent parameter calibration is performed to fine-tune the model accuracy and loss. This effort includes manual adjusting parameters to achieve the best performance. Therefore, these deep CNN models are trained with a learning rate of 0.001, a decay rate of 0.1, momentum of 0.9, a batch size of 32, and a fixed number of iterations of 50. The initial weights of these models come from the ImageNet dataset. It is worth noting that for the deep learning models, only the last layer needs to be trained, while the rest of the layers remain the same.

In the framework of the traditional DeepLabv3+ model, MobileNetV3 takes the role of replacing Xception as the backbone of basic feature extraction. In order to determine the effectiveness and accuracy of crack segmentation in the proposed model, three well-known segmentation models which are DeeplabV3+, U-Net, and PSPNet are used for crack segmentation in the datasets of this study. For the purpose of fair comparison, all segmentation models use abstract features extracted from the pre-trained standard backbone network derived from the ImageNet dataset. These include VGG16, Xception, ResNet50, MobileNetV2 (Sandler *et al*. 2018), and MobileNetV3. All models are trained for 50 epochs, and a fixed batch size of 8 is maintained. Stochastic gradient descent optimization function with learning rate of 0.01 and momentum of 0.9 is used as initial parameters to train the model.

The error function used in the training process of different classification networks is Cross-Entropy (De Boer

et al. 2005), and the binary Cross-Entropy is expressed

$$\begin{aligned} \text{Loss} &= \frac{1}{N}\sum_{i} \text{Loss}_i \\ &= \frac{1}{N}\sum_{i} -[z_i \cdot \log(p_i) + (1-z_i)\cdot \log(1-p_i)] \end{aligned} \tag{2}$$

where z_i denotes the label of the sample i of which the positive class is 1 and the negative class is 0, p_i represents the probability that sample i is predicted to be positive; N is the total number of samples. The error function adopted by the semantic segmentation network in the training process is Dice Loss (Sudre *et al*. 2017), and the expression of Dice Loss is

$$Dice\ Loss = 1 - \frac{2|\mathbf{X} \cap \mathbf{Y}|}{|\mathbf{X}| + |\mathbf{Y}|} \tag{3}$$

where $|\mathbf{X} \cap \mathbf{Y}|$ is the intersection between $\mathbf{X}$ and $\mathbf{Y}$, $|\mathbf{X}|$ and $|\mathbf{Y}|$ represent the number of elements in $\mathbf{X}$ and $\mathbf{Y}$, respectively. The smaller the value of the Dice Loss function, the higher the degree of agreement between the predicted results and the actual results.

3.3 Evaluation indicators

In order to comprehensively evaluate the effectiveness of the training model in the training and testing stages, a series of widely used evaluation indicators are adopted. For all kinds of crack classification networks, their quantification is achieved through a single evaluation index, namely accuracy. Instead, for different crack semantic segmentation networks, the evaluation process includes four different indicators: accuracy, mean pixel accuracy (mPA), intersection over union (IoU), and mean intersection over union (mIoU). The definitions and equations of these evaluation indicators are outlined as follows.

Accuracy is defined as the percentage of cracks correctly identified in all images

$$Accuracy = \frac{Tp + Tn}{Tp + Fp + Tn + Fn} \tag{4}$$

where Tp is the number of true cracks, Tn is the number of true non-cracks, Fp is the number of false cracks and Fn is the number of false non-cracks.

The mPA represents the ratio of accurately segmented pixels to the total number of pixels in the image

$$mPA = \frac{1}{n+1}\sum_{i=0}^{n} \frac{q_{ii}}{\sum_{j=0}^{n} q_{ij}} \tag{5}$$

where n is the number of the class, q_{ii} corresponds to the total number of true positives of the class i, and q_{ij} is the total number of pixels labeled as class j.

IoU is a metric used to evaluate the overlap rate between the real crack pixels B_1 and the predicted crack pixels B_2 on the concrete surface, assuming that they belong to a single category. Fig. 9 shows the IoU between the real crack pixels (in green) and the predicted crack pixels (in red) on

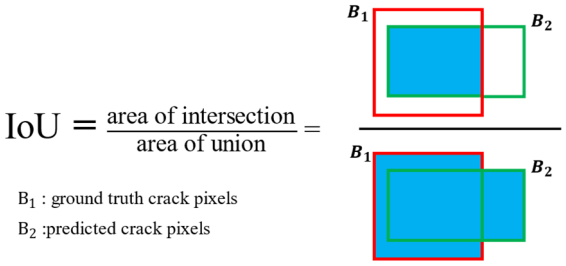


Fig. 9 IoU schematic representation of semantic segmentation

the concrete surface. IoU is defined as follows

$$\text{IoU} = \frac{area(B_1 \cap B_2)}{area(B_1 \cup B_2)} \tag{6}$$

The mIoU is the average of the IoU values for each category in the dataset which is the most commonly used evaluation index in semantic segmentation because of its representativeness and simplicity, defined as

$$\text{mIoU} = \frac{IoU}{n} \tag{7}$$

4. Results and discussion

Aiming at reducing the training time of the model and achieving high precision and efficient performance, the current study advocates integrating lightweight networks to drive the classification and segmentation of crack images. Firstly, the MobileNetV3 network takes the lead in crack classification, and finally identifies images containing cracks. Then the crack segmentation model is responsible for pixel segmentation of the classified crack images based on the enhanced DeepLabv3+. In this enhanced DeepLabv3+ architecture, the MobileNetV3 lightweight network replaces Xception as the feature extraction backbone which follows the traditional DeepLabv3+ framework while reducing computational requirements and improving model efficiency. In addition, the introduction of CBAM enriches the segmentation process by extracting key information from the crack images, thereby improving the segmentation accuracy. This strategy of merging seeks to coordinate fast model training, accuracy, and effective performance in the field of crack classification and segmentation.

4.1 Crack image classification results

The performance evaluation of the proposed crack detection model includes the use of crack classification datasets to train and verify various crack classification algorithms for informative comparison. Considering the large number of parameter counts of the deep CNN models, the potential overfitting problem when training on a limited dataset needs to be addressed. Therefore, the fine-tuned initial weights obtained from the ImageNet dataset are employed for these deep CNNs. The training accuracy, validation accuracy, and training time of each model are

Table 2 Comparison of results using different methods

Method	Training accuracy	Validating accuracy	Training time (s)
VGG16	0.9626	0.9644	34793.4
Xception	0.9714	0.9685	19125.1
Inceptionv3	0.9280	0.9283	10920.8
Resnet50	0.9628	0.9579	14875.5
MobileNetV2	0.9639	0.9626	7860.2
MobileNetV3	**0.9868**	**0.9800**	**6524.2**

shown in Table 2. Fig. 10 gives the training and verification loss curves of the crack image classification model anchored by deep CNNs. This graphical representation emphasizes that as the iteration progresses, the loss of the model is close to the minimum and stable state, affirming the successful training of the crack classification model. It can be seen from Table 2 that the verification accuracy of the crack image classification model based on MobileNetV3 is 0.9868, which is slightly better than other traditional deep learning models in accuracy. In addition, compared with other crack classification models, the MobileNetV3 model has a significantly lower training time of 6524.2 s. It is worth noting that VGG16 has a verification accuracy of 0.9644, but it takes 34793.4 s of training time. Therefore, the study infers that the MobileNetV3 network can quickly complete model training

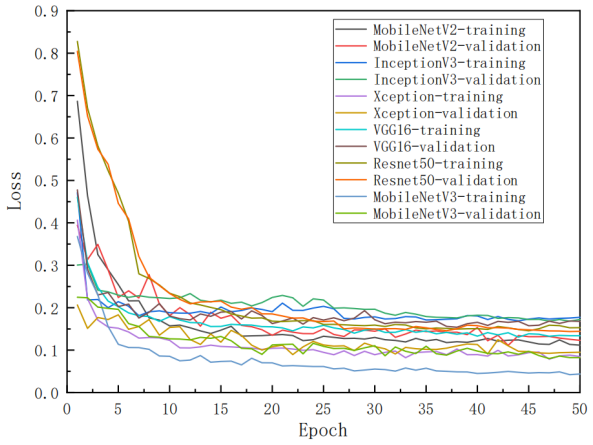


Fig. 10 Loss curves for classification of different algorithms using crack datasets

and provide commendable classification performance, which is obtained by comparing with various network models.

4.2 Results of crack image segmentation

In this section, the performance of the proposed improved DeepLabV3+ model in crack image segmentation is discussed. The segmentation results of all the network variations based on the standard evaluation index are shown in Fig. 11. In general, the improved DeepLabV3+ utilizes

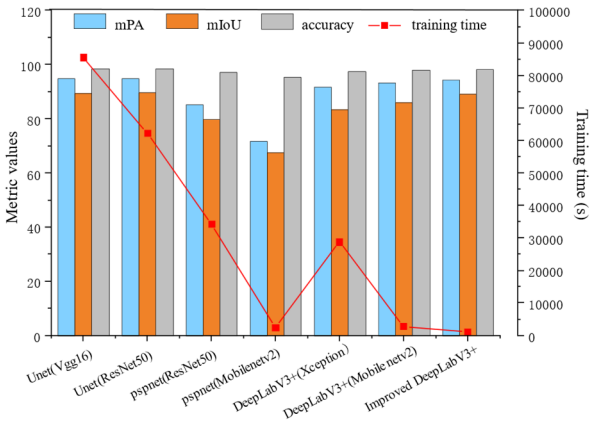
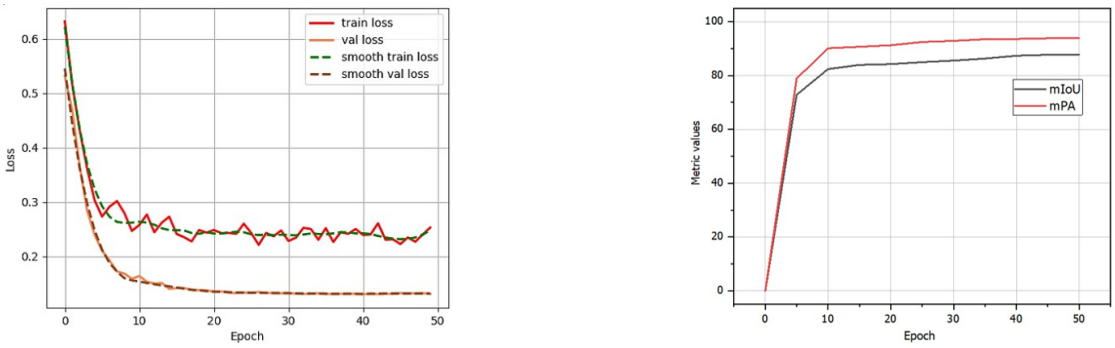


Fig. 11 The mPA, mIoU, accuracy and training time of the segmentation models

Table 3 Detailed parameters and training time of different crack segmentation models

Method	mPA	mIoU	Total parameters (float)	FLOPs (G)	Training time (s)
Unet (Vgg16)	94.95%	89.5%	24,892,437	451.250	85536
Unet (ResNet50)	**94.92%**	**89.82%**	44,013,781	181.560	62151
PSPNet (ResNet50)	85.26%	79.84%	46,774,997	116.519	34243
DeepLabV3+ (Xception)	91.85%	83.47%	41,253,330	103.155	28720
DeepLabV3+ (MobileNetV2)	92.18%	85.06%	2,753,714	11.281	2730
PSPNet (MobileNetV2)	71.79%	67.60%	2,409,765	5.872	2443
Improved DeepLabV3+	**93.87%**	**87.79%**	**1,496,696**	**2.094**	**1155**



(a) Training and validation loss (b) Validation mIoU and mPA

Fig. 12 Loss, mIoU and mPA of the improved DeepLabV3+

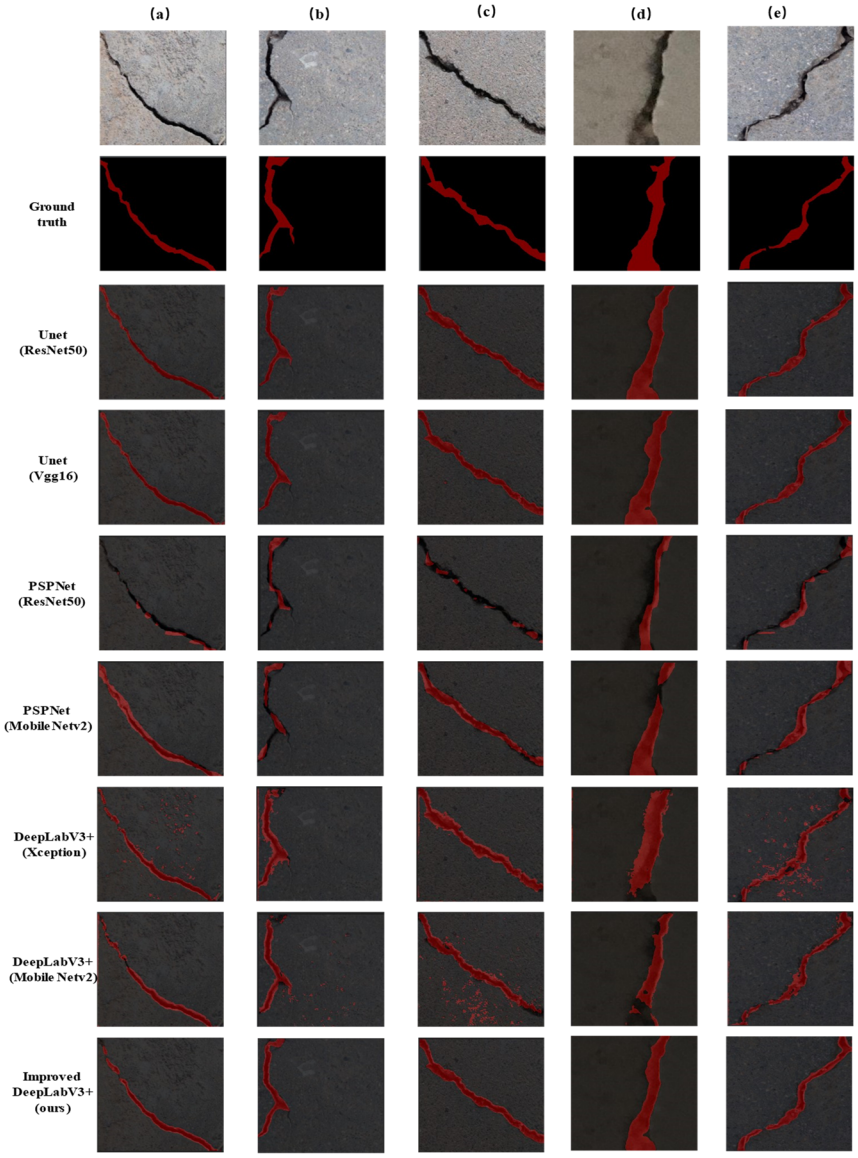


Fig. 13 Comparison of pixel segmentation results of different network models

the pre-trained MobileNetV3 as the encoder and combines the CBAM attention mechanism to produce a typical crack segmentation result. While the mPA and mIoU scores of the improved DeepLabV3+ model are 93.87% and 87.79%, respectively, they do show a marginal decrease of 1.05% and 2.03% compared with the U-Net model using the ResNet50 network as the encoder. Importantly, the proposed crack segmentation model significantly reduces

the training time to only 1155 s, making it significantly different from similar models.

The improved DeepLabV3+ outperforms the original DeepLabV3+ on both mIoU and mPA, highlighting the superior performance of the enhanced segmentation model. A comprehensive comparison of the total parameters and floating-point operations (FLOPs) of various models is shown in Table 3. FLOPs quantify the computational requirements of the model as a gauge of its complexity. It is worth noting that the proposed improved DeepLabV3+ has a minimum of 1496,696 float parameters and 2.094G FLOPs, making it the lightest of all the segmentation models. Therefore, the improved DeepLabV3+ is considered to be the best model for crack segmentation using the processed datasets. Fig. 12 shows the mIoU and loss curves on the training and validation datasets to prove the effectiveness of the training process and network convergence of the improved DeepLabV3+ model. In general, the model achieves stable convergence throughout the training process. In the initial stage, the verified mIoU and mPA fluctuate greatly, stabilizing around 83% and 89%, respectively. After that, the curves take a more gradual trajectory, stabilizing at about 87% mIoU and 93% mPA. In addition, the validation loss of the proposed model undergoes a significant change before reaching a steady state of 0.12, reflecting its good generalization capability.

In order to evaluate the performance of the improved DeepLabV3+ model in crack detection and crack density assessment tasks, a set of concrete surface crack images from actual scenes are used for testing. These test images are from the publicly available bridge crack dataset described in Section 3.1, including real bridge surface images taken under various environmental conditions. The test set is independently segmented from the same dataset as the validation set to ensure data consistency while avoiding overlap with training and validation samples. This setting ensures that the test set accurately reflects the challenges encountered in practical engineering applications, such as bridge inspection under natural conditions. The test results are shown in Fig. 13. In addition, other trained deep-learning segmentation models are also tested to comprehensively compare their segmentation capabilities. As shown in Fig. 13, the proposed improved DeepLabV3+ model achieves excellent results in testing image segmentation. For example, the predicted output of the test image Fig. 13(a) using the improved DeepLabV3+ model is very close to the ground truth. On the contrary, when observing the test results of DeepLabV3+ (MobileNetV2) in Figs. 13(c) and (d), some non-cracked areas are mistakenly identified and segmented into cracks, while some cracks themselves are not sufficiently segmented. Therefore, the testing results show that compared with the proposed model, the original DeepLabV3+ with MobileNetV2 and Xception as the feature extraction backbone exhibits poor segmentation performance. Notably, the Unet model also shows excellent segmentation results, slightly exceeding the improved DeepLabV3+. However, compared with the Unet model, the proposed model retains the advantages of fast network training and minimal model parameters.

5. Conclusions

Deep learning algorithms play a key role in the automatic classification and segmentation of concrete surface crack images. In this paper, the feasibility and superior performance of the improved DeepLabV3+ method in classification and segmentation tasks are proved. The proposed model is trained on open and real-world crack datasets, as well as meticulously annotated pixel-level crack datasets, and shows remarkable results. The lightweight MobileNetV3 network for crack image classification shows excellent validation accuracy, slightly better than other well-known deep learning networks. The model also significantly shortens the training time. For crack image segmentation, the improved DeepLabV3+ model exhibits excellent performance in multiple tests, with mIoU reaching 87.79% and mPA reaching 93.87%. Although the Unet model reported slightly higher mIoU and mPA values, the proposed model performs well in terms of training efficiency and number of network parameters, which are key factors for crack segmentation models. In the context of a variety of feature extraction modules, the model proposed in this paper is superior to Unet, PSPNet, and the original DeepLabV3+, and is the optimal choice. Finally, the model is applied to the actual concrete surface crack images to verify robustness and effectiveness. The results show that the model has practical application potential in the field of concrete surface crack detection and segmentation.

Acknowledgments

The work described here has been supported by Gansu Provincial Major Scientific and Technological Project (Project No. 24ZDGA001), the National Natural Science Foundation of China (Grant No. 52278303), the Department of Science and Technology of Guangdong Province, China (Grant No. 2023A1111120008) and Guangzhou Municipal Education Bureau Project (Grant No. 2024312301).

References

Agdas, D., Rice, J.A., Martinez, J.R. and Lasa, I.R. (2016), "Comparison of visual inspection and structural-health monitoring as bridge condition assessment methods", J. *Perform. Constr. Facil.*, **30**(3), p. 04015049. http://dx.doi.org/10.1061/(asce)cf.1943-5509.0000802

Akagic, A., Buza, E., Omanovic, S. and Karabegovic, A. (2018), "Pavement crack detection using Otsu thresholding for image segmentation", In: *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics* (*MIPRO*), pp. 1092-1097. http://dx.doi.org/10.1155/2022/1155814

Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F. and Adam, H. (2018), "Encoder-decoder with atrous separable convolution for semantic image segmentation", In: Computer Vision – ECCV 2018, Lecture Notes in Computer Science, Vol. 11211. https://doi.org/10.1007/978-3-030-01234-2_49

Dan, D. and Dan, Q. (2021), "Automatic recognition of surface cracks in bridges based on 2D-APES and mobile machine vision", *Measurement*, **168**, p. 108429.

http://dx.doi.org/10.1016/j.measurement.2020.108429
De Boer, P.T., Kroese, D.P., Mannor, S. and Rubinstein, R.Y. (2005), "A tutorial on the cross-entropy method", *Ann. Operat. Res.*, **134**(1), 19-67. http://dx.doi.org/10.1007/s10479-005-5724-z
Dong, N., Zhao, L., Wu, C.H. and Chang, J.F. (2020), "Inception v3 based cervical cell classification combined with artificially extracted features", *Appl. Soft. Comput.*, **93**, p. 106311. https://doi.org/10.1016/j.asoc.2020.106311
Dorafshan, S. and Maguire, M. (2018), "Bridge inspection: human performance, unmanned aerial systems and automation", *J. Civil Struct. Health Monitor.*, **8**(3), 443-476. http://dx.doi.org/10.1007/s13349-018-0285-4
Du, G., Cao, X., Liang, J., Chen, X. and Zhan, Y. (2020), "Medical image segmentation based on U-Net: A review", *J. Imag. Sci. Techn.*, **64**(2).
Dung, C.V. (2019), "Autonomous concrete crack detection using deep fully convolutional neural network", *Automat. Constr.*, **99**, 52-58. http://dx.doi.org/10.1016/j.autcon.2018.11.028
Fu, H., Meng, D., Li, W. and Wang, Y. (2021), "Bridge crack semantic segmentation based on improved Deeplabv3+", *J. Marine Sci. Eng.*, **9**(6), p. 671. http://dx.doi.org/10.3390/jmse9060671
Gribniak, V., Kaklauskas, G. and Bacinskas, D. (2008), "Shrinkage in reinforced concrete structures: A computational aspect", *J. Civil Eng. Manag.*, **14**(1), 49-60. http://dx.doi.org/10.1088/1361-6501/ab79c8
Gucunski, N., Boone, S.D., Zobel, R., Ghasemi, H., Parvardeh, H. and Kee, S.-H. (2014), "Nondestructive evaluation inspection of the Arlington Memorial Bridge using a Robotic Assisted Bridge Inspection Tool (RABIT)", In: *Nondestructive Characterization for Composite Materials*, *Aerospace Engineering*, *Civil Infrastructure*, *and Homeland Security 2014*, pp. 148-160. https://doi.org/10.1117/12.2063963
Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L.-C., and Tan, M. (2019), "Searching for MobileNetV3", In: *2019 IEEE/CVF International Conference on Computer Vision* (*ICCV*), pp. 1314-1324. http://dx.doi.org/10.1109/iccv.2019.00140
Indraswari, R., Rokhana, R. and Herulambang, W. (2022), "Melanoma image classification based on MobileNetV2 network", *Procedia Comput. Sci.*, **197**, 198-207. https://doi.org/10.1016/j.procs.2021.12.132
Ji, A.K., Xue, X.L., Wang, Y.N., Luo, X.W. and Xue, W.R. (2020), "An integrated approach to automatic pixel-level crack detection and quantification of asphalt pavement", *Automat. Constr.*, **114**., p. 103176\. http://dx.doi.org/10.1016/j.autcon.2020.103176
Kanaeva, I. and Ivanova, J.A. (2021), "Road pavement crack detection using deep learning with synthetic data", In: *IOP Conference Series*: *Materials Science and Engineering*, **1019**, p. 012036. http://dx.doi.org/10.1088/1757-899X/1019/1/012036.
Koonce, B. (2021), "MobileNetV3", In: *Convolutional Neural Networks with Swift for Tensorflow*: *Image Recognition and Dataset Categorization*, pp. 125-144.
Le, T., Gibb, S., Pham, N., La, H.M., Falk, L. and Berendsen, T. (2017), "Autonomous robotic system using non-destructive evaluation methods for bridge deck inspection", In: *2017 IEEE International Conference on Robotics and Automation* (*ICRA*), pp. 3672-3677.
Lee, J.S., Hwang, S.H., Choi, I.Y. and Choi, Y. (2020), "Estimation of crack width based on shape-sensitive kernels and semantic segmentation", *Struct. Control. Health Monitor.*, **27**(4). http://dx.doi.org/10.1002/stc.2504
Liu, Z.Q., Cao, Y.W., Wang, Y.Z. and Wang, W. (2019), "Computer vision-based concrete crack detection using U-net fully convolutional networks", *Automat. Constr.*, **104**, 129-139. http://dx.doi.org/10.1016/j.autcon.2019.04.005
Liang, Y., Lin, Y. and Lu, Q. (2022), "Forecasting gold price using a novel hybrid model with ICEEMDAN and LSTM-CNN-CBAM", *Expert Syst. Applicat.*, **206**, p. 117847. https://doi.org/10.1016/j.eswa.2022.117847
Moreau, N., Rousseau, C., Fourcade, C., Santini, G., Ferrer, L., Lacombe, M., Guillerminet, C., Jezequel, P., Campone, M., Normand, N. and Rubeaux, M. (2021), "Comparison between threshold-based and deep learning-based bone segmentation on whole-body CT images", In: *Medical Imaging 2021*: *Computer-Aided Diagnosis*, 11597, pp. 661-667.
Qi, Y.Z., Yuan, C., Kong, Q.Z., Xiong, B. and Li, P.Z. (2021), "A deep learning-based vision enhancement method for UAV assisted visual inspection of concrete cracks", *Smart Struct. Syst.*, *Int. J.*, **27**(6), 1031-1040. http://dx.doi.org/10.12989/sss.2021.27.6.1031
Rahimzadeh, M. and Attar, A. (2020), "A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2", *Informatics in medicine unlocked*, **19**, p. 100360. https://doi.org/10.1016/j.imu.2020.100360
Russell, B.C., Torralba, A., Murphy, K.P. and Freeman, W.T. (2008), "LabelMe: A database and web-based tool for image annotation", *Int. J. Comput. Vision*, **77**(1-3), 157-173. http://dx.doi.org/10.1007/s11263-007-0090-8
Simonyan, K. and Zisserman, A. (2014), "Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv:1409.1556.
Song, Q., Lin, G., Ma, J. and Zhang, H. (2016), "An edge-detection method based on adaptive canny algorithm and iterative segmentation threshold", In: *2016 2nd International Conference on Control Science and Systems Engineering* (*ICCSSE*), pp. 64-67. http://dx.doi.org/10.1109/ccsse.2016.7784354
Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S. and Jorge Cardoso, M. (2017), "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations", In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*: *Third International Workshop*, *DLMIA 2017*, *and 7th International Workshop*, Québec City, QC, Canada, September, pp. 240-248. http://dx.doi.org/10.1007/978-3-319-67558-9_28.
Tang, W., Wu, R.T. and Jahanshahi, M.R. (2022), "Crack segmentation in high-resolution images using cascaded deep convolutional neural networks and Bayesian data fusion", *Smart Struct. Syst.*, *Int. J.*, **29**(1), 221-235. http://dx.doi.org/10.12989/sss.2022.29.1.221
Tian, B. and Wei, W. (2022), "Research overview on edge detection algorithms based on deep learning and image fusion", *Secur. Commun. Network*, **2022**, p. 1155814. http://dx.doi.org/10.1155/2022/1155814
Wang, G. and Xiang, J.W. (2021), "Railway sleeper crack recognition based on edge detection and CNN", *Smart Struct. Syst.*, *Int. J.*, **28**(6), 779-789. https://doi.org/10.12989/sss.2021.28.6.779
Wang, Z., Wang, J., Yang, K., Wang, L., Su, F. and Chen, X. (2022), "Semantic segmentation of high-resolution remote sensing images based on a class feature attention mechanism fused with Deeplabv3+", *Comput. Geosci.-Uk*, **158**, p. 104969. https://doi.org/10.1016/j.cageo.2021.104969
Wen, L., Li, X.Y. and Gao, L. (2020), "A transfer convolutional neural network for fault diagnosis based on ResNet-50", *Neural Comput. Appl.*, **32**(10), 6111-6124. http://dx.doi.org/10.1007/s00521-019-04097-w
Yan, M., Bo, S., Xu, K. and He, Y. (2007), "Pavement crack

detection and analysis for high-grade highway", In: *2007 8th International Conference on Electronic Measurement and Instruments*, 4-548-4-552. https://doi.org/10.1109/ICEMI.2007.4351202

Yang, X.M., Yi, T.H., Qu, C.X., Li, H.N. and Liu, H. (2019), "Automated eigensystem realization algorithm for operational modal identification of bridge structures", *J. Aerospace. Eng.*, **32**(2), p. 04018148. http://dx.doi.org/10.1061/(asce)as.1943-5525.0000984

Ye, X.W., Li, Z.X. and Jin, T. (2022), "Smartphone-based structural crack detection using pruned fully convolutional networks and edge computing", *Smart Struct. Syst., Int. J.*, **29**(1), 141-151. https://doi.org/10.12989/sss.2022.29.1.141

Yeum, C.M. and Dyke, S.J. (2015), "Vision-based automated crack detection for bridge inspection", *Comput-Aided. Civil Infrastr. Eng.*, **30**(10), 759-770. http://dx.doi.org/10.1111/mice.12141

Zhang, J., Cai, Y.Y., Yang, D., Yuan, Y., He, W.Y. and Wang, Y.J. (2023), "MobileNetV3-BLS: A broad learning approach for automatic concrete surface crack detection", *Constr. Build. Mater.*, **392**, p. 131941. http://dx.doi.org/10.1016/j.conbuildmat.2023.131941

Zhu, X., Cheng, Z., Wang, S., Chen, X. and Lu, G. (2021), "Coronary angiography image segmentation based on PSPNet", *Comput. Meth. Prog. Biomed.*, **200**, p. 105897. https://doi.org/10.1016/j.cmpb.2020.105897