Computers, Materials & Continua
DOI: 10.32604/cmc.2023.035165
Article

Bridge Crack Segmentation Method Based on Parallel Attention Mechanism and Multi-Scale Features Fusion

Jianwei Yuan[1], Xinli Song[1,*], Huaijian Pu[2], Zhixiong Zheng[3] and Ziyang Niu[3]

[1]College of Civil Engineering, Changsha University of Science and Technology, Changsha, 410114, China
[2]College of Computer & Communication Engineering, Changsha University of Science and Technology, Changsha, 410114, China
[3]China Construction Fifth Engineering Bureau Co., Ltd., Changsha, 410004, China
*Corresponding Author: Xinli Song. Email: sxl@stu.csust.edu.cn
Received: 09 August 2022; Accepted: 24 October 2022

Abstract: Regular inspection of bridge cracks is crucial to bridge maintenance and repair. The traditional manual crack detection methods are time-consuming, dangerous and subjective. At the same time, for the existing mainstream vision-based automatic crack detection algorithms, it is challenging to detect fine cracks and balance the detection accuracy and speed. Therefore, this paper proposes a new bridge crack segmentation method based on parallel attention mechanism and multi-scale features fusion on top of the DeeplabV3+ network framework. First, the improved lightweight MobileNet-v2 network and dilated separable convolution are integrated into the original DeeplabV3+ network to improve the original backbone network Xception and atrous spatial pyramid pooling (ASPP) module, respectively, dramatically reducing the number of parameters in the network and accelerates the training and prediction speed of the model. Moreover, we introduce the parallel attention mechanism into the encoding and decoding stages. The attention to the crack regions can be enhanced from the aspects of both channel and spatial parts and significantly suppress the interference of various noises. Finally, we further improve the detection performance of the model for fine cracks by introducing a multi-scale features fusion module. Our research results are validated on the self-made dataset. The experiments show that our method is more accurate than other methods. Its intersection of union (IoU) and F1-score (F1) are increased to 77.96% and 87.57%, respectively. In addition, the number of parameters is only 4.10 M, which is much smaller than the original network; also, the frames per second (FPS) is increased to 15 frames/s. The results prove that the proposed method fits well the requirements of rapid and accurate detection of bridge cracks and is superior to other methods.

Keywords: Crack detection; DeeplabV3+; parallel attention mechanism; feature fusion

1 Introduction

As a crucial part of transportation lines, bridges play an essential role in the development of the social economy [1]. However, due to factors such as unstable foundations, excessive loads, and temperature differences, bridges can develop undesirable cracks, thus reducing the service life of the bridge and even causing bridge collapse accidents. The related studies have revealed that 90% of bridge damage collapses are caused by cracks [2]. Therefore, it is essential to inspect and repair bridge cracks regularly. Conventional bridge crack detection mainly adopts manual detection, which has the problems of low detection accuracy, time-consuming and high risk. An accurate and efficient method for automatically detecting bridge cracks is urgently needed to overcome the shortcomings of manual inspection methods.

Initially, image processing methods were the mainstream methods for crack detection. However, traditional image processing methods can only complete accurate detection on crack images with high contrast [3]. Subsequently, the proposed crack detection methods based on machine learning can improve the detection accuracy to a certain extent, but it is challenging to extract features [4].

In recent years, with deep learning becoming a research hotspot in artificial intelligence, a variety of deep learning networks have been proposed [5–8]. Since deep learning networks can solve complex problems by automatically learning features at different levels, researchers have applied them to the detection of cracks. The most noteworthy is the semantic segmentation algorithms, such as fully convolutional neural network (FCN) [7], unity networking (U-net) [8] with typical encoder-decoder structure and etc. Studies show that semantic segmentation algorithms have high accuracy in crack detection, but there are still some non-negligible problems. Schmugge et al. applied SegNet to pavement crack detection [9,10], where the pooling layer in the network leads to the loss of crack details in different regions, making the network unable to use context information fully. Thus, this method will produce discontinuous or missed detection for cracks with complex topology structures. Using U-net with the encoder-decoder structure to detect cracks will also have the same problem mentioned above [11]. On the other hand, the background of bridge crack images can be complex and noisy, which makes the detection work difficult. The introduction of the attention mechanism can establish the correlation between features, allowing us to focus more on the crack regions while suppressing information from irrelevant areas. In addition to the above problems, there are few studies on the imbalance between detection accuracy and speed of existing crack detection algorithms. If a complex backbone feature extraction network is used, it will increase training and prediction time. Conversely, if a single lightweight network is used, it will lead to a decrease in feature extraction capability and detection accuracy.

Based on the above analysis, we propose a bridge crack detection method based on parallel attention mechanism and multi-scale features fusion. The method is based on the framework of the deep learning algorithm DeeplabV3+. Firstly, we replace the backbone network Xception in the original network with a modified lightweight network based on MobileNet-v2. Then, the standard dilated convolution is replaced with the dilated separable convolution to improve the atrous spatial pyramid pooling (ASPP) module. The above two operations can reduce the complexity of the model and speed up the model training and prediction. Additionally, the parallel attention mechanism is introduced to improve the representation of important features in different stages. Finally, we fuse the multi-scale low-level features with the high-level features in the decoding stage to better utilize the features at different levels and improve the detection ability of fine cracks.

In summary, the main contributions of this paper are as follows:

(1) We improve the lightweight network MobileNet-v2 and apply it as the backbone network in the DeeplabV3+ network, which can significantly reduce the parameters. We introduce the dilated convolution into MobileNet-v2 to obtain a larger receptive field, which can improve the accuracy of crack segmentation.
(2) We propose the dilated separable convolution to replace the standard dilated convolution, which can dramatically reduce the number of parameters in the network and accelerate the training and forecasting speed.
(3) We introduce the parallel attention mechanism into the encoding and decoding stages to effectively improve the representation of the vital crack features.
(4) We adopt a multi-scale features fusion module to further improve the detection performance of the model for fine cracks.
(5) We use sliding window technology to create a dataset of bridge crack images and conduct evaluation experiments to prove the superiority of our method. The experimental results show the intersection of union (IoU) and F1-score (F1) of our method are increased to 77.96% and 87.57%, respectively, the number of parameters is only 4.10 M, and the frames per second (FPS) is increased to 15 frames/s.

The main contents of the remaining chapters of this paper are as follows: Section 2 introduces the development of the crack detection field. Section 3 details the overall structure of the proposed method and each module. Section 4 presents the preparation of datasets and the details of the experimental setup. Section 5 illustrates and analyses the experimental results. Section 6 concludes this paper.

2 Related Work

In this section, we introduce some typical algorithms of crack detection, divided into three categories: image processing-based methods, machine learning-based methods, and deep learning-based methods.

2.1 Image Processing

In order to realize automatically detect cracks, more and more methods based on image processing are proposed. The most widely used algorithms are the edge detection algorithm [3,12] threshold segmentation method [13,14], and filter-based algorithm [15,16].

In the early edge detection stage, local information such as brightness, color, texture, and gradient are mainly used. The commonly used edge detection algorithms include Canny and Sobel arithmetic. Ayenu-Prah et al. [3] applied bidimensional empirical mode decomposition (BEMD) to remove noise and analyzed the residual image with the Sobel edge detector for crack detection. However, the Sobel method still suffered from the effects of noise when the images had many irregularities. Zhao et al. [12] proposed a new Canny edge detection method based on the OpenCV library to process the video data so as to protect and improve the real-time accuracy of pavement crack detection. The principle of the threshold segmentation method comes from the difference between the gray scale range of target and background. The image pixels are divided into target and background regions by setting different feature thresholds. Akagic et al. [13] proposed a new unsupervised method for detecting cracks with gray color-based histograms and Otsu's thresholding method on two-dimensional pavement images. Chen et al. [14] combined the global and local thresholds to segment the images to distinguish cracked and non-cracked regions, but this method did not perform well in detecting cracks with high

clutter conditions. In addition, filter-based algorithms are also applied to crack detection, which can effectively remove the complex noise information around the bridge cracks. Li et al. [16] proposed a new detection algorithm based on the Bilateral-Frangi filter, which enhanced the crack structure while realizing edge protection and denoising simultaneously to improve the accuracy of crack detection.

2.2 Machine Learning

With the development of computer vision technology, machine learning is also gradually applied to the field of crack detection. Machine learning algorithms are mainly divided into supervised learning and unsupervised learning, and researchers mostly use supervised learning algorithms to detect cracks. Firstly, an optimal model is obtained by supervised learning training using existing crack samples, and then use the model to predict cracks.

Sheng et al. [4] trained the model with a gradient boost decision tree and then used the trained model to test each pixel in the crack image to determine whether it is a part of the crack. However, the method has only been performed on static images so far, limiting the model's scope of application. Li et al. [17] combined the linear support vector machine (SVM) using a greedy search strategy and the modified region-based active contour model for crack detection. This method can quickly detect cracks and obtain better detection accuracy. Chen et al. [18] proposed a novel crack detection method based on local binary patterns (LBP) [19], support vector machine [20], and Bayesian decision theory, which can enhance the accuracy and robustness of detection, but there were still some false detections. Peng et al. [21] proposed a triple-thresholds pavement crack detection method leveraging random structured forest, which can obtain preliminary crack detection results while suppressing noise. Nevertheless, the feature extraction engineering of traditional machine learning algorithms is often very time-consuming and labor-intensive. In addition, unique feature extraction methods need to be designed for bridge crack images with different textures, which makes the generalization performance of the model poor.

2.3 Deep Learning

Automatically learning features is the most significant advantage of deep learning algorithms, so some scholars have proposed crack detection methods based on image classification and object detection, which opened up a new way for the field of crack detection. Xu et al. [22] proposed an efficient and high-precision end-to-end crack detection model based on the convolutional neural network (CNN), which had higher recognition accuracy and faster recognition speed than previous classification models. Subsequently, based on this paper, Li et al. [23] proposed the Skip-Squeeze-Excitation Networks and added the ASPP module into the model. Compared with the method in Ref. [22], the detection accuracy of this method was further increased by 1.4% with the same model complexity. Besides, Yang et al. [24] used the trained you only look once (YOLO) V3 model can accurately detect cracks in images and videos. However, these methods can only identify the presence of cracks or roughly frame out cracks and cannot accurately extract cracks at the pixel level. Semantic segmentation-based crack detection methods can precisely meet this requirement. Therefore, more and more scholars tend to research crack detection based on semantic segmentation algorithms [25–30].

Dung et al. [28] used the fully convolutional network to achieve proper crack segmentation of the concrete crack images. However, due to the use of a more complex visual geometry group (VGG)-based encoding network, the detection speed of the model needed to be improved. Ren et al. [29] proposed an improved deep fully convolutional neural network Cracksegnet for tunnel crack segmentation, which showed higher detection accuracy and better generalization performance. Liu et al. [30] proposed using

U-net for pixel-level crack detection. Although this method repaired the detailed features and obtained higher accuracy through multi-scale features fusion and up-sampling, the crack information was lost to a certain extent. To address such a problem, Li et al. [31] applied dense atrous convolution (DAC) block and residual multi-kernel pooling (RMP) block to retain more crack information and features from the crack images. Xiang et al. [32] adopted the pyramid module to exploit context information fully. In addition, the attention mechanism has also been a research hotspot in crack detection in recent years. Song et al. [33] proposed a model that introduced a multi-scale expansion attention module after the encoder to generate more detailed crack detection results. Wan et al. [34] proposed a crack detection method based on an encoder-decoder deconstruction, which realized a high-precision detection of pavement cracks by adding a channel-spatial combined attention module after each encoder.

In summary, the existing crack detection algorithms based on deep learning do not achieve the balance between detection accuracy and speed. As a result, we propose to replace the original backbone network with the modified MobileNet-v2 network and add depthwise separable convolution to the ASPP module. This operation not only resolves this problem but also maintains the segmentation accuracy. Furthermore, combined with the above improvement ideas, a multi-scale features fusion module is introduced to further strengthen the attention to the underlying details and ensure that small targets are not lost, so as to obtain favorable detection results.

3 Our Method

As shown in Fig. 1, our model framework consists of two parts: the encoder composed of a modified backbone network and improved ASPP module, and the decoder based on multi-scale features fusion module. First of all, the collected bridge crack images are inputted into the improved backbone network to extract features and generate multi-scale feature maps for feature fusion in the following steps. Then, the ASPP module with depthwise separable convolution and parallel attention mechanisms is used to refine the high-level feature maps. Finally, we adopt the multi-scale features fusion module to generate more accurate crack segmentation results. The model based on the parallel attention mechanism and multi-scale features fusion proposed in this paper performs well in bridge crack detection tasks. The entire process for detecting bridge cracks using our novel method is shown in Algorithm 1.

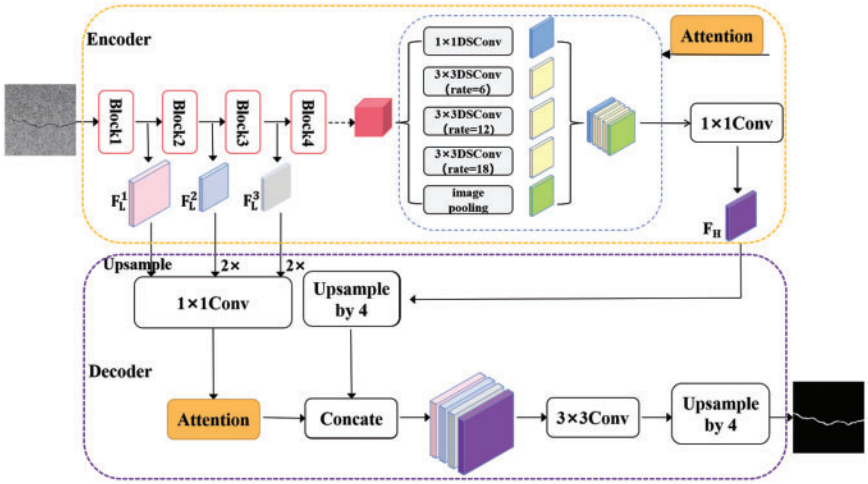

Figure 1: Structure of the crack detection model

Algorithm 1: The entire process for detecting bridge cracks

Input: Bridge crack image data D
Image clipping based on sliding window, and filter and construct the crack label images: $D \rightarrow D'$
Data augmentation: $D' \rightarrow D''$
Divide D'' into training set T_0, validation set V, test set T_1,
Build our crack detection model framework M
Initialize parameters of model M
Initialize iterations N
Initialize the index of iteration $i \leftarrow 0$
Repeat
  if $i < N$ **then**
    Calculate the loss function in each batch
    Training model M with back propagation
    $i \leftarrow i + 1$
  end
  else
    Complete the training of crack detection model M, and get the optimal parameters for model M
    Input bridge crack test image data T_1 to M
    Extract bridge cracks
    Output bridge crack detection results
  end
until Output;

3.1 Improvement of Backbone Network

The backbone network of the original DeeplabV3+ is Xception. However, it has a large number of parameters, making the model training and prediction slower, which does not apply to the task of fast crack detection. To meet the requirement of balancing the speed and accuracy of crack detection, we use the modified MobileNet-v2 as the backbone network. The network structure of the modified MobileNet-v2 is shown in Table 1. Since the continuous down-sampling of the network will lead to the reduction of the size of the feature maps and the loss of details, the dilated convolution is used instead of the standard convolution in the last four layers of the original network, and the step size of the 5th and 7th layers is changed to 1. Dilated convolution can obtain a larger receptive field without increasing the number of parameters while keeping the resolution size of the output feature map unchanged. If the original MobileNet-v2 is used as the backbone network, the size of the output feature map is 16×16 when the input image size is 512×512. In comparison, the improved MobileNet-v2 has an output feature map size of 64×64 due to the dilated convolution, which can retain more crack image details and improve the accuracy of crack segmentation.

3.2 Dilated Separable Convolution

In this paper, we combine dilated convolution and depthwise separable convolution to form dilated separable convolution, which can significantly reduce model parameters and improve the training and forecasting speed. Dilated separable convolution is mainly applied in the ASPP module in Fig. 1.

Table 1: Improved MobileNet-v2 network structure

Input	Operation	t	c	n	s	Rate
$512^2 \times 3$	Conv2d	/	32	1	2	1
$256^2 \times 32$	Block	1	16	1	1	1
$256^2 \times 16$	Block	6	24	2	2	1
$128^2 \times 24$	Block	6	32	3	2	1
$64^2 \times 64$	Dilated block	6	64	4	1	2
$64^2 \times 96$	Dilated block	6	96	3	1	2
$64^2 \times 160$	Dilated block	6	160	3	1	4
$64^2 \times 320$	Dilated block	6	320	1	1	4

Notes: Conv2d is standard two-dimensional convolution; Block is the inverted residual structural block; Dilated block is the inverted residual structural block with dilated convolution; t is the expansion multiple; *c* is the number of output channels; *n* is the number of corresponding modules; *s* is the convolution step size when the layer appears for the first time; rate is the dilated rate in dilated convolution.

The receptive field is significant in image segmentation, which determines the upper limit of the size that the network can detect. Dilated convolution aggregates a more extensive range of feature information by adding spacing to the kernel of standard convolution, which gives the network a larger receptive field without increasing the parameters and ensures that the size of the output feature map remains constant. Fig. 2 shows dilated convolution with different rates. Fig. 2a is standard convolution, and Fig. 2b shows the rate of dilated convolution is 2, which means that the receptive field of the convolution kernel is $5 \times 5 = 25$.

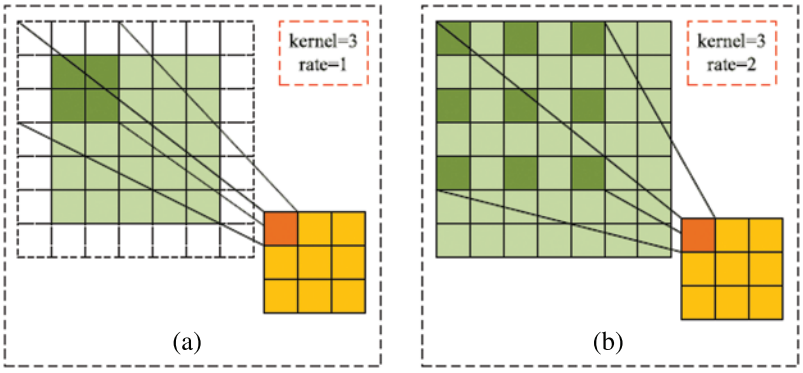

Figure 2: Dilated convolution. (a) Standard convolution, (b) Dilated convolution

The depthwise separable convolution [35] is composed of depthwise and pointwise convolution. Depthwise convolution exploits spatial location information, and pointwise convolution fuses inter-channel information. As shown in Fig. 3, convolutions with a kernel size of 3×3 are carried out for

each channel of the input feature map to output the channel-separated characteristic attributes. Then the output feature map is obtained using a 1×1 convolution at each location. The depthwise separable convolution can effectively suppress the increase of the model's parameters under the premise that the performance is comparable to the standard convolution, thereby reducing the complexity of the model.

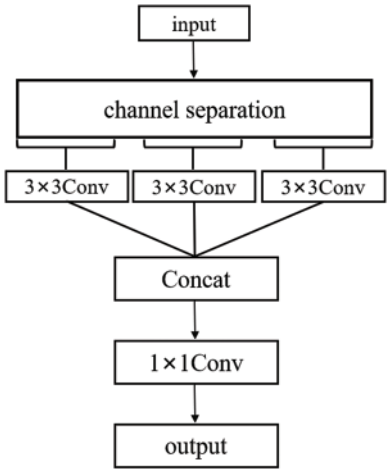

Figure 3: Depthwise separable convolution

3.3 Parallel Attention Mechanism

Segmentation of bridge cracks is tremendously disturbed by noise in bridge crack images. The attention mechanism can refine the characteristics of cracks so that the network can rapidly scan the entire crack image and identify the areas that require attention. In this paper, the parallel attention module is added to the encoder and decoder to efficiently select and filter features and suppress interference brought on by irrelevant information. Fig. 4 demonstrates the primary structure of the parallel attention module.

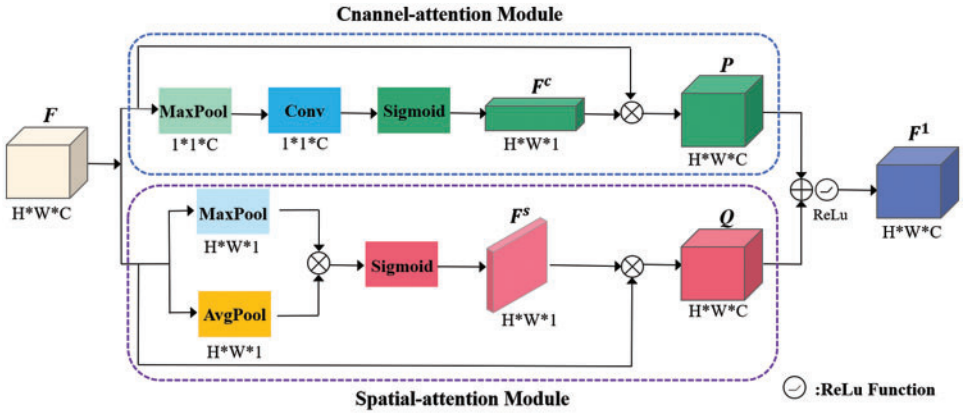

Figure 4: Parallel attention mechanism

Assuming that the input feature map is F, we can obtain the feature map P and Q through the parallel attention module. The P refers to the output of the channel attention branch, while Q refers

to the output of the spatial attention branch. The refined feature map F^1 is defined as:

$$F^1 = \sigma\left(P \oplus Q\right) = \sigma\left(\left(F^c \otimes F\right) \oplus \left(F^s \otimes F\right)\right) \tag{1}$$

where σ is ReLu activation function, F^c and F^s represent channel attention map and spatial attention map respectively, $\oplus$ denotes element-wise addition, and $\otimes$ represents multiply operation.

The upper part of Fig. 4 illustrates that the channel attention module is a combination of the global maximum pooling and one-dimensional convolution. First, we adopt the global maximum pooling on the input feature map F to generate a $1 \times 1 \times$ C feature vector. Then the one-dimensional convolution is performed on the feature vector. Finally, the sigmoid function is performed to gain the channel attention weighting map F^c, defined as:

$$F^c = \sigma_1\left(Conv\left(MaxPool\left(F\right)\right)\right) \tag{2}$$

where Conv represents one-dimensional convolution with convolution kernel size k, and the expression is $k = \left|\frac{log2C + b}{r}\right|_{odd}$, C reflects the number of channels, $|X|_{odd}$ conveys the nearest odd number to X, r and b are 2 and 1, respectively.

The spatial attention module comprises the average pooling and maximum pooling, as observed in the lower portion of Fig. 4. First, the average pooling and maximum pooling are performed on the input feature map F in the channel dimension to generate two feature maps. Then we multiply them element by element. Finally, the spatial attention weighting map F^s is procured using the sigmoid function and is defined as:

$$F^s = \sigma_1\left(AvgPool\left(F\right) \otimes MaxPool\left(F\right)\right) \tag{3}$$

where σ_1 represents Sigmoid activation function.

As displayed in Fig. 1, the attention module is connected to each branch of the ASPP module during the encoding stage, strengthening the quality of the generated high-level feature maps and gaining more beneficial high-level semantic feature information. Regarding the decoding stage, the attention module is connected after the low-level features generated by the network to refine the detailed features of the cracks on the low-level feature maps. Then, the multi-scale feature maps are fused by the feature fusion module to yield a more discriminative feature map.

3.4 Multi-Scale Features Fusion Module

The features at different levels of the neural network also vary tremendously. Specifically, the high-level features focus more on semantic information but are not enough to describe the detailed information, while the low-level features contain rich spatial detail information. The original DeeplabV3+ only fuses a single low-level feature map with the high-level feature map obtained by the encoder, then recovers to the resolution of the original image through four times bilinear interpolation up-sampling. Nevertheless, the bridge crack images contain a host of fine cracks. Suppose only a single low-level feature map and high-level feature map are fused to extract the target features. In that case, it will possibly lead to the loss of small targets, which is unquestionably fatal to the detection of fine bridge cracks.

Since each feature map generated in the bottom stages of the backbone network is of paramount importance to the final crack segmentation results, this paper makes improvements to the feature fusion module based on the idea of multi-scale spatial fusion [36]. Both multi-scale low-level feature maps and high-level feature maps are fused to enhance the detection ability of the model for fine

cracks. As depicted in Fig. 5, the improved network has four feature fusion branches, where F_L^1, F_L^2, and F_L^3 represent the low-level feature maps, and F_H represents the high-level feature map. Firstly, we up-sample low-level feature maps F_{L2} and F_{L3} by two times to acquire the same size as F_{L1}. Then, 1×1 convolution is utilized to compress the feature channels for these three feature maps to get feature indications $F_{L1}^{'}$, $F_{L2}^{'}$, and $F_{L3}^{'}$, which are considered as guides for the low-level features. In addition, we use up-sampling to change the size of the high-level feature map F_H and achieve the high-level feature indicator $F_H^{'}$. Finally, they are fused to fully integrate various detailed features and elevate the accuracy of image segmentation.

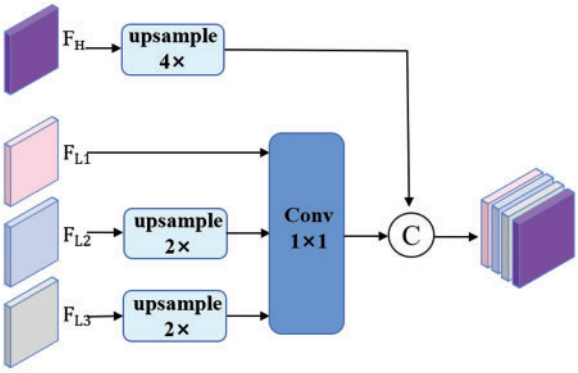

Figure 5: The structure of multi-scale features fusion module

3.5 Combined Loss Function

We use the improved binary cross-entropy loss (Focal loss) + dice coefficient loss (Dice loss) as the loss function of the algorithm. Due to the small proportion of the foreground part of the cracks in the bridge crack images, the ordinary binary cross-entropy loss will be affected by the category imbalance. As a result, the learning of crack characteristics by the network is inhibited. The Focal loss function [37] solves the category imbalance problem by introducing a weighting factor to reduce the weight on easily classified samples, defined as follows:

$$L_{Focal} = -\frac{1}{N}\sum_{i=1}^{N}\left[-y_i\beta\left(1-p_i\right)^{\gamma} log\left(p_i\right) + \left(1-y_i\right)\left(1-\beta\right)p_i^{\gamma} log\left(1-p_i\right)\right] \tag{4}$$

where N is the total number of picture pixels, y_i is the label value of the ith pixel, p_i is the predicted value of the ith pixel, γ and β are 2 and 0.25, respectively.

The Dice loss function [38] is a region-dependent loss function. Its essence is to measure the overlapping part between two samples, which makes the network prefer to tap the crack foreground part during the training process so that it can solve the imbalance between positive and negative samples to some extent. However, when y_i and p_i are too small, it will lead to a significant change in the loss values, thus making the gradient change drastically and causing training instability. The Dice loss function is defined as follows:

$$L_{Dice} = 1 - \frac{\sum_{i=1}^{N} y_i p_i + \varepsilon}{\sum_{i=1}^{N} y_i + p_i + \varepsilon} - \frac{\sum_{i=1}^{N}\left(1-y_i\right)\left(1-p_i\right) + \varepsilon}{\sum_{i=1}^{N} 2 - y_i - p_i + \varepsilon} \tag{5}$$

where N is the total number of picture pixels, y_i is the label value of the ith pixel, p_i is the predicted value of the ith pixel, ε is the smoothing coefficient, and the value is 1.

Using the combined loss function can make the training of the model more stable and effectively solve the imbalance between positive and negative samples of pixels in the crack images simultaneously so that the model has more accurate crack segmentation results. The combined loss function is defined as follows:

$$L = L_{Focal} + L_{Dice} \tag{6}$$

where L_{Focal} is the Focal loss function, L_{Dice} is the Dice loss function.

4 Datasets and Experimental Design

4.1 Datasets

The training of deep convolutional neural networks requires a large number of labeled datasets as a basis, but there is currently no suitable open-source dataset of bridge cracks. The dataset of this paper comes from two parts: First, 1000 crack images with a resolution of 1024×1024 were extracted from the Ref. [39], then a fixed-sized sliding window was used to crop the crack images without overlapping, and the cropped images are filtered to remove the crack-free and blurred images, finally got the crack images with a resolution of 512×512 for training. Second, 148 images with a resolution of 5472×3648 were obtained by unmanned aerial vehicle (UAV), then used the same image cropping and filtering method as above to get sub-images with 512×512 pixels. We combined the above two datasets and manually labeled the images using the image labeling software Labelme to obtain their masks. However, if we want the deep convolutional neural network to get better training, merely using these data is insufficient and can lead to overfitting of the network model. Therefore, we performed data augmentation on the dataset. As shown in Fig. 6, the data augmentation methods include horizontal flip, vertical flip, random brightness, and hue-saturation-value (HSV) sharpening. Finally, 4766 crack images with 512×512 pixels were obtained as the dataset and randomly divided images in the dataset, including 80% as the training set, 10% as the validation set, and 10% as the test set.

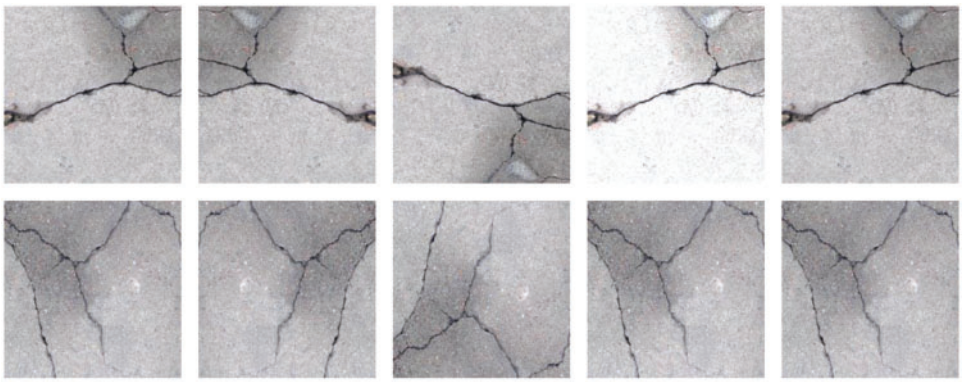

Figure 6: Crack dataset augmentation. (a) Raw image, (b) Horizontal flip, (c) Vertical flip, (d) Random brightness, (e) Hue-saturation-value (HSV) sharpening

4.2 Experimental Environment

The configuration of the experimental environment in this paper is shown in Table 2. Using Adam optimizer during the training process can accelerate the convergence of the network training. The

network will fail to converge or oscillate around the optimal value if there is an excessive learning rate. In contrast, an undersized learning rate can cause the network to converge very slowly or to a local optimum. Therefore, the initial learning rate is very significant. For the training, the initial learning rate is 0.0001, the step size is 1, and the decay coefficient is 0.94. In order to control the batch size of the experimental model is consistent, the batch size is 8, and the number of iterations is 100 epochs.

Table 2: Experimental environment configuration

Configuration	Parameter
Operating system	Ubuntu 18.04
GPU	NVIDIA RTX A4000
CPU	Xeon E5-2686 v4
Development environment	Pytorch 1.7.1 CUDA 11.0, cuDNN 8.0
Software	Python3.8

4.3 Evaluation Indicators

Semantic segmentation is a pixel-level classification, and crack segmentation is essentially a dichotomous classification of each image pixel. To evaluate the performance of bridge crack segmentation algorithms intuitively and quantitatively, we use the pixel IoU and F1 as evaluation indicators. The definition of these indicators is related to the pixel evaluation categories, shown in Table 3. The expressions of the evaluation indices are as follows:

$$IoU = \frac{TP}{TP + FP + FN} \tag{7}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{8}$$

where P is the Precision, R is the Recall, and the expressions are:

$$P = \frac{TP}{TP + FP} \tag{9}$$

$$R = \frac{TP}{TP + FN} \tag{10}$$

Table 3: Pixel evaluation categories

	Labels are crack pixels	Labels are background pixels
Predictions are crack pixels	TP	FP
Predictions are background pixels	FN	TN

5 Experimental Results and Analysis

This section is divided into three sub-sections, namely comparative experiment of different loss functions, ablation experiment of the improved module, and comparative experiment of different

models. We describe this section from loss function, detection accuracy, parameter, and FPS. Based on the above description to verify the effectiveness of our method.

5.1 Comparative Experiment of Different Loss Functions

To compare the effects of the three different loss functions on our method, the Focal loss function, the Dice loss function, and the combination of them are used as the loss functions of our algorithm, respectively. Then, as shown in Fig. 7, three sets of experiments are carried out on the same dataset to obtain the corresponding evaluation metrics.

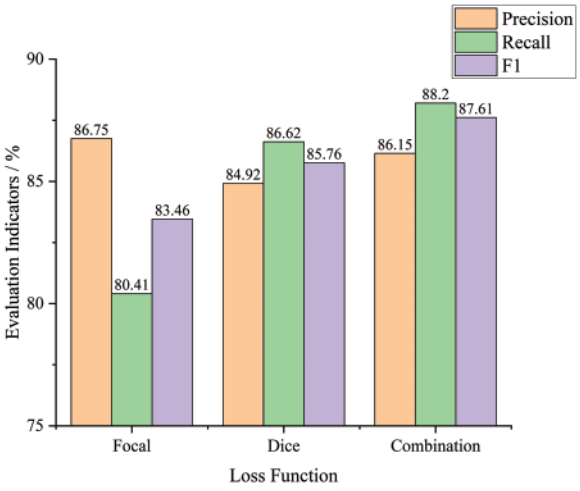

Figure 7: Evaluation in crack detection with different loss functions

When the Focal loss function is selected, the model has a low recall and a relatively high accuracy. In contrast, when using the Dice loss function, the model's accuracy is low, and the recall is relatively high. The combined loss function can combine these two advantages to make the model training more stable and solve the problem of unbalanced pixel categories. The recall of the model is significantly improved under the condition that a small amount of precision loss is guaranteed. The recall is 7.8% and 1.59% higher than the Dice loss function and Focal loss function, respectively, and the precision is only 0.6% lower than the Focal loss function alone. Through the above analysis, the combination loss function can get better all-around performance from the perspective of the actual demand for crack detection.

5.2 Ablation Experiment of the Improved Module

To prove the effectiveness of each improvement module in this paper, we train the network on the train set and evaluate it on the test set. The training parameters and loss function are the same for each model, and the loss function is the combined loss function. The results are compared as shown in Table 4, where IB represents the improved backbone network, DSC represents the dilated separable convolution, Attention represents the parallel attention mechanism, and MFF represents the multi-scale features fusion module.

Improvement of backbone: To prove the effectiveness of the improved backbone network, we conduct experiments to analyze and compare. The backbone network in this paper is the improved MobileNet-v2, which introduces dilated convolution to enlarge the receptive field without losing

information so that each convolution output contains an extensive range of information. Table 4 shows that the improved backbone network can reduce the model parameters by one order of magnitude compared with the original backbone network and increase the IoU and F1 by 1.2% and 0.8%, respectively.

Table 4: Comparison of ablation experiments of improved modules

Serial number	IB	DSC	Attention	MFF	IoU (%)	P (%)	R (%)	F1 (%)	Param (M)
1					73.24	82.38	86.93	84.55	54.71
2	√				74.44	86.59	84.14	85.35	5.81
3	√			√	75.43	86.70	85.31	86.00	6.05
4	√	√			74.28	84.46	86.04	85.24	3.86
5	√	√	√		77.35	85.97	88.52	87.23	3.86
6	√	√	√	√	**77.96**	**86.75**	88.21	**87.57**	4.10

Dilated separable convolution: As mentioned above, to reduce the redundancy of the model and accelerate the model training and prediction, we replace the dilated convolution in the ASPP module with the dilated separable convolution. As shown in Table 4, under the premise of the equivalent accuracy, the parameters of the model are reduced by 34%.

Parallel Attention Mechanism: We introduce the parallel attention module in the encoding and decoding stages to select and filter features effectively and suppress the interference caused by irrelevant information. It can be seen from Table 4 that the addition of the parallel attention module increases IoU and F1 by 2.91% and 1.88%, respectively. Our attention module can obtain more complementary features, which is more conducive to accurately segmenting cracks.

Multi-scale features fusion module: To enhance the detection performance of the model for fine cracks and improve the segmentation accuracy of the model, we introduce a multi-scale features fusion module to fully use the relatively complete spatial position information in the low-level feature maps. The indicators in Table 4 show that the multi-scale features fusion module can increase IoU by 0.61% and F1 by 0.34%. Therefore, we can conclude that the multi-scale features fusion module is adequate for bridge crack segmentation tasks.

5.3 Comparative Experiment of Different Models

In this section, we select several state-of-the-art methods for comparison, including FCN, U-net, pyramid scene parsing network (PSPNet), and Deeplabv3+. Fig. 8 shows the training losses of these five different models, and their training parameters are consistent. We can see that our method can minimize the loss value faster, and its convergence speed is significantly higher than the other four models, indicating that the performance of our method is higher than that of others.

As we can see from Table 5, on the self-made bridge crack dataset, the number of parameters of our proposed method is only 12.43%, 8.77%, 16.45%, and 7.49% of that of FCN, PSPNet, U-net, and DeeplabV3+, respectively. The FPS of our method is 15 frames per second, which is 3.4 frames per second slower than the FCN algorithm ranking first in FPS, but the IoU and F1 on the test set are increased to 77.96% and 87.57%, respectively, which are much higher than FCN. The experimental results show that our method performs better than the other four mainstream semantic segmentation algorithms and can segment bridge crack images more accurately.

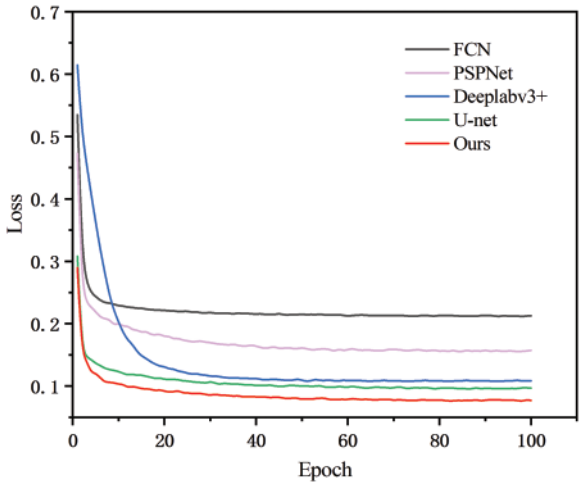

Figure 8: Loss function curve

Table 5: Comparison of results of different models on self-made bridge crack dataset

Method	IoU (%)	P (%)	R (%)	F1 (%)	Param (M)	FPS
FCN	69.02	78.4	85.1	81.61	32.95	18.4
PSPNet	73.12	82.99	86.01	84.47	46.71	10.7
U-net	73.59	82.47	87.24	84.79	24.89	9.4
DeeplabV3+	73.24	82.3	86.93	84.55	54.71	9.2
Ours	**77.96**	**86.75**	**88.21**	**87.57**	**4.10**	15

To illustrate the superiority of our method more intuitively, we randomly select five images from the test set and visualize the crack image segmentation results of the above methods for comparison. As shown in Fig. 9, where (a) is the original bridge crack image, (b) is the labeled image, and (c) to (g) are the detection results of FCN, PSPNet, U-net, DeeplabV3+, and our method, respectively. It can be found that the model proposed in this paper still maintains a better segmentation effect than other methods when detecting crack images containing noises such as shadows, occlusions, or the detected cracks are very fine. In Section 1, we mentioned that Liu et al. [11] used the U-net with a typical code-decode structure to detect cracks. Due to the presence of pooling layers, discontinuous detection and missed detection of fine cracks can occur. From Fig. 9, we can see that this situation is significantly improved after we introduce the parallel attention mechanism and multi-scale features fusion module, which further proves the effectiveness of the research in this paper.

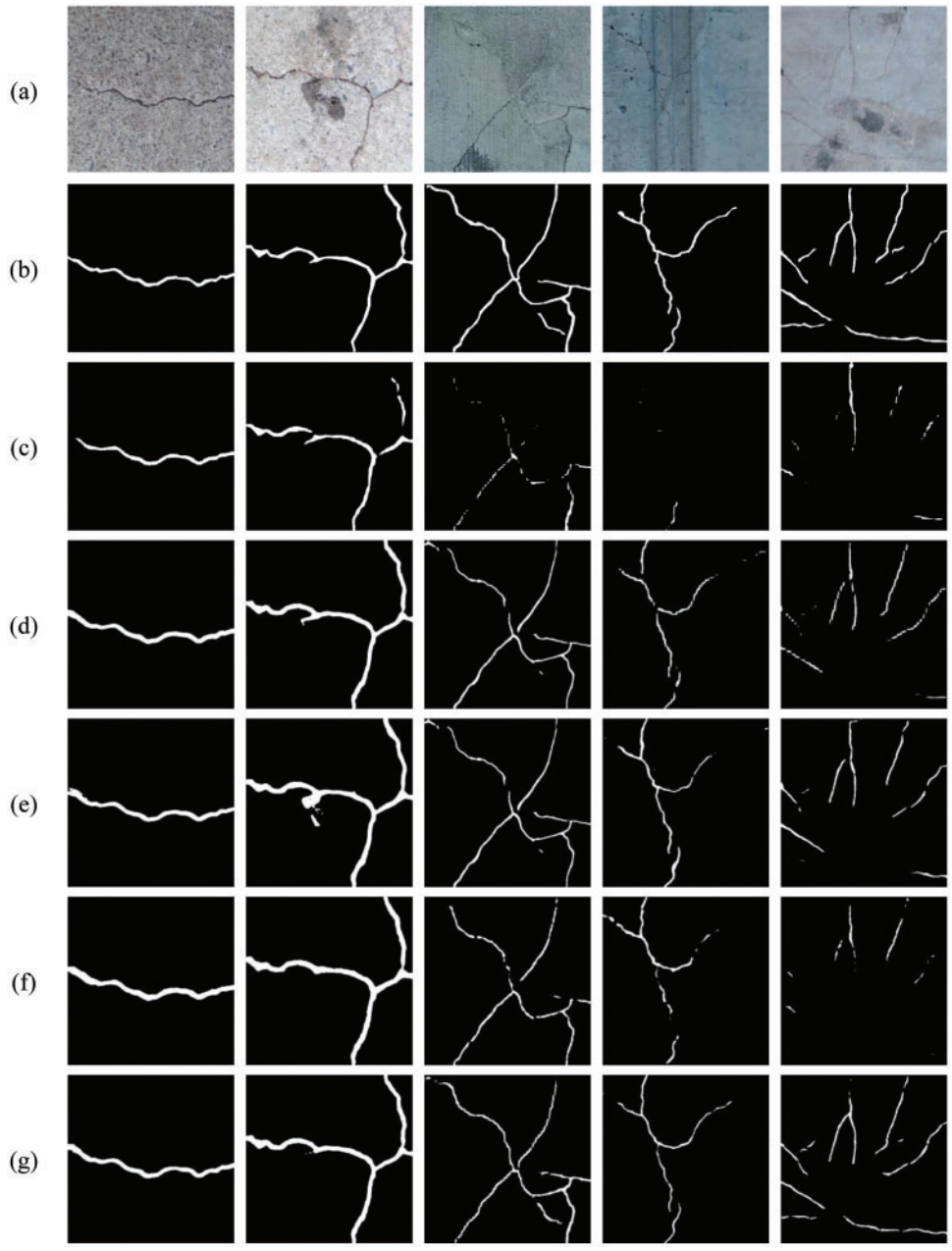

Figure 9: Visualization of segmentation results of each model. (a) Raw image, (b) Label image, (c) FCN, (d) PSPNet, (e) U-net, (f) DeeplabV3+, (g) Ours

6 Conclusion

In order to balance the speed and accuracy of crack detection while improving the detection capability for fine cracks, we proposed a bridge crack segmentation method based on parallel attention mechanism and multi-scale features fusion. To ensure the simplicity of the model and speed up model prediction, we adopted a lightweight operation for the backbone network and replaced the

dilated convolution of the four branches in the ASPP module with dilated separable convolution. Furthermore, we introduced the parallel attention mechanism and multi-scale features fusion module, which can effectively improve the performance of crack detection. The proposed algorithm was validated and analyzed by experiments on the self-made dataset. The results revealed that the IoU and F1 were increased to 77.96% and 87.57%, respectively, the number of parameters was only 4.10 M, and the FPS reached 15 frames/s. Compared with other algorithms, the proposed method in this paper had higher crack detection accuracy and faster detection speed, which can effectively solve the problem of unbalanced detection accuracy and speed in existing algorithms.

To sum up, our method is applicable for the efficient and accurate detection of bridge cracks. However, the similarity between specific noise and cracks is also high, and can be easily detected as cracks. Although our method has a higher detection ability than other mainstream methods, it cannot completely eliminate the influence of noise. In addition, the bridge crack data set utilized in this study is a small sample data set, and the diversity of cracks does not satisfy the demand for crack samples in practical engineering. Therefore, in future research, we will consider constructing a profound feature extraction network and a sizable bridge crack data set to lay the groundwork for subsequent automatic bridge crack detection.

Acknowledgement: The authors acknowledge the support of Changsha University of Science and Technology and the support of High-Tech Industry Science and Technology Innovation Leading Plan Project of Hunan Provincial.

Funding Statement: This work was supported by the High-Tech Industry Science and Technology Innovation Leading Plan Project of Hunan Provincial under Grant 2020GK2026, author B.Y, http://kjt.hunan.gov.cn/.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] X. G. Zhang, G. Liu, J. H. Ma, H. B. Wu, B. Y. Fu *et al.,* "Status and prospect of technical development for bridges in China," *Chinese Science Bulletin*, vol. 61, no. 4–5, pp. 415–425, 2016.

[2] H. Y. Xu, "Research on bridge crack detection method based on convolutional neural network," M.S. Theses, Tianjin University, China, 2019.

[3] A. Ayenu-Prah and N. Attoh-Okine, "Evaluating pavement cracks with bidimensional empirical mode decomposition," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, pp. 861701, 2008.

[4] P. Sheng, L. Chen and J. Tian, "Learning-based road crack detection using gradient boost decision tree," in *2018 13th IEEE Conf. on Industrial Electronics and Applications (ICIEA)*, Wuhan, China, pp. 1228–1232, 2018.

[5] K. M. He, X. Y. Zhang, S. Q. Ren and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, pp. 770–778, 2016.

[6] S. Q. Ren, K. M. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[7] J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, Boston, MA, USA, pp. 3431–3440, 2015.

[8] O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention*, Munich, Germany, pp. 234–241, 2015.

[9] S. J. Schmugge, L. Rice, J. Lindberg, R. Grizziy, C. Joffey *et al.,* "Crack segmentation by leveraging multiple frames of varying illumination," in *2017 IEEE Winter Conf. on Applications of Computer Vision (WACV)*, Santa Rosa, CA, USA, pp. 1045–1053, 2017.
[10] V. Badrinarayanan, A. Kendall and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
[11] F. Y. Liu and L. B. Wang, "UNet-based model for crack detection integrating visual explanations," *Construction and Building Materials*, vol. 322, pp. 126265, 2022.
[12] F. Zhao, W. H. Zhou, Y. T. Chen and H. C. Peng, "Application of improved Canny operator in crack detection," *Electronic Measurement Technology*, vol. 41, no. 20, pp. 107–111, 2018.
[13] A. Akagic, E. Buza, S. Omanovic and A. Karabegovic, "Pavement crack detection using Otsu thresholding for image segmentation," in *2018 41st Int. Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, pp. 1092–1097, 2018.
[14] C. Chen, H. Seo, C. H. Jun and Y. Zhao, "A potential crack region method to detect crack using image processing of multiple thresholding," *Signal, Image and Video Processing*, vol. 16, pp. 1673–1681, 2022.
[15] J. Yang, "Bridge crack detection algorithm based on Bilateral-Frangi filter," in *Journal of Physics: Conf. Series*, Dalian, China, pp. 012044, 2021.
[16] H. T. Li, X. D. Chen, H. Y. Xu, H. Y. Xu, Y. Wang *et al.,* "Bridge crack detection algorithm based on Bilateral-Frangi filter," *Laser & Optoelectronics Progress*, vol. 56, no. 18, pp. 170–176, 2019.
[17] G. Li, X. X. Zhao, K. Du, F. Ru and Y. B. Zhang, "Recognition and evaluation of bridge cracks with modified active contour model and greedy search-based support vector machine," *Automation in Constructions*, vol. 78, pp. 51–61, 2017.
[18] F. C. Chen, M. R. Jahanshahi, R. T. Wu and C. Joffe, "A texture-based video processing methodology using Bayesian data fusion for autonomous crack detection on metallic surfaces," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 4, pp. 271–287, 2017.
[19] D. Huang, C. F. Shan, M. Ardabilian, Y. H. Wang and L. M. Chen, "Local binary patterns and its application to facial image analysis: A survey," *IEEE Transactions on Systems Man & Cybernetics Part C Applications & Reviews*, vol. 41, no. 6, pp. 765–781, 2011.
[20] J. Cervantes, F. Garcia-Lamont and L. Rodríguez-Mazahua, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, 2020.
[21] C. Peng, M. Q. Yang, Q. H. Zheng, J. Zhang, D. Q. Wang *et al.,* "A triple-thresholds pavement crack detection method leveraging random structured forest," *Construction and Building Materials*, vol. 263, pp. 120080, 2020.
[22] H. Y. Xu, X. Su, Y. Wang, H. Y. Cai, K. R. Cui *et al.,* "Automatic bridge crack detection using a convolutional neural network," *Applied Sciences*, vol. 9, no. 14, pp. 2867, 2017.
[23] H. T. Li, H. Y. Xu, X. D. Tian, Y. Wang, H. Y. Cai *et al.,* "Bridge crack detection based on SSENets," *Applied Sciences*, vol. 10, no. 12, pp. 4230, 2020.
[24] C. Yang, J. J. Chen, Z. Y. Li and Y. Huang, "Structural crack detection and recognition based on deep learning," *Applied Sciences*, vol. 11, no. 6, pp. 2868, 2021.
[25] W. D. Song, G. H. Jia, H. Zhu, D. Jia and L. Gao, "Automated pavement crack damage detection using deep multiscale convolutional features," *Journal of Advanced Transportation*, vol. 2020, pp. 1–11, 2020.
[26] L. K. Wang, X. H. He, S. Faming, G. L. LU, H. Cong *et al.,* "A real-time bridge crack detection method based on an improved inception-resnet-v2 structure," *IEEE Access*, vol. 9, pp. 93209–93223, 2021.
[27] Q. Song, Y. Q. Wu, X. S. Xin, L. Yang, M. Yang *et al.,* "Real-time tunnel crack analysis system via deep learning," *IEEE Access*, vol. 7, pp. 64186–64197, 2019.
[28] C. V. Dung and L. D. Anh, "Autonomous concrete crack detection using deep fully convolutional neural network," *Automation in Construction*, vol. 99, pp. 52–58, 2019.
[29] Y. P. Ren, J. H. Huang, Z. Y. Hong, W. Liu, J. Yin *et al.,* "Image-based concrete crack detection in tunnels using deep fully convolutional networks," *Construction and Building Materials*, vol. 234, pp. 117367, 2020.

[30] Z. K. Liu, Y. W. Cao, Y. Z. Wang and W. Wang, "Computer vision-based concrete crack detection using U-net fully convolutional networks," *Automation in Construction*, vol. 104, pp. 129–139, 2019.
[31] G. Li, X. Y. Li, J. Zhou, D. Z. Liu and W. Ren, "Pixel-level bridge crack detection using a deep fusion about recurrent residual convolution and context encoder network," *Measurement*, vol. 176, pp. 109171, 2021.
[32] X. Z. Xiang, Y. Q. Zhang and A. El. Saddik, "Pavement crack detection network based on pyramid structure and attention mechanism," *IET Image Processing*, vol. 14, no. 8, pp. 1580–1586, 2020.
[33] W. D. Song, G. H. Jia, D. Jia and H. Zhu, "Automatic pavement crack detection and classification using multiscale feature attention network," *IEEE Access*, vol. 7, pp. 171001–171012, 2019.
[34] H. F. Wan, L. Gao, M. N. Su, Q. R. Sun and L. Huang, "Attention-based convolutional neural network for pavement crack detection," *Advances in Materials Science and Engineering*, vol. 2021, pp. 5520515, 2021.
[35] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. CVPR*, Hawaii, HI, USA, pp. 1251–1258, 2017.
[36] G. M. Hou, J. H. Qin, X. Y. Xiang, Y. Tan and N. N. Xiong, "AF-Net: A medical image segmentation network based on attention mechanism and feature fusion," *Computers, Materials & Continua*, vol. 69, no. 2, pp. 1877–1891, 2021.
[37] T. Y. Lin, P. Goyal, R. Girshick, K. M. He and P. Dollar, "Focal loss for dense object detection," in *Proc. CVPR*, Hawaii, HI, USA, pp. 2980–2988, 2017.
[38] X. Y. Li, X. F. Sun, Y. X. Meng, J. J. Liang, F. Wu *et al.,* "Dice loss for data-imbalanced NLP tasks," *arXiv Preprint arXiv*, vol. 1911, pp. 02855, 2019.
[39] L. F. Li, W. F. Ma, L. Li and C. Lu, "Research on detection algorithm for bridge cracks based on deep learning," *Acta Automatica Sinica*, vol. 45, no. 9, pp. 1727–1742, 2018.