A Pavement Crack Segmentation Algorithm Based on I-U-Net Network

Yun Bai, En Lu, Hao-bo Wang

***Abstract*—Due to the influence of background noise, traditional pavement crack segmentation methods are unable to fully extract crack features and effectively fuse them, resulting in low segmentation accuracy, large segmentation errors, and numerous missed detections. To solve these issues, this paper proposes an I-U-Net (Improved U-Net) pavement crack segmentation algorithm based on the U-Net network. Firstly, the introduction of Bot-Res (Residual Module with Bottleneck Structure) facilitates the network in obtaining complete crack information, while the bottleneck structure reduces the high computational load of the residual module. Secondly, to eliminate the interference of background noise, we innovate the O-CBAM (Optimized Convolutional Block Attention Module), which enhances the shallow crack contour information and effectively acquires the spatial position information of deep crack pixels. Finally, an EDMSF (Encoder-Decoder Multi-Scale Fusion) module is constructed based on the idea of multi-scale fusion. Different convolutional kernels of different sizes are selected according to different levels to extract crack information, thereby enriching the extracted image feature information and improving segmentation performance. Additionally, to tackle the issue of uneven distribution between positive and negative samples, an enhanced cross-entropy loss function is introduced. We evaluate the performance of the I-U-Net network on CRACK500, CFD and our Self-built Dataset. Experimental results demonstrate that the proposed I-U-Net achieves superior performance across all evaluation metrics, with 95.75% accuracy, 96.06% precision, 86.83% recall, 91.79% F1-score, and 92.27% mIoU. Compared to the baseline U-Net, the I-U-Net exhibits significant improvements in segmentation performance, thereby validating the effectiveness of the proposed methodology.**

***Index Terms*—pavement crack segmentation, I-U-Net, Bot-Res, O-CBAM, EDMSF, cross-entropy loss function**

I. Introduction

PAVEMENT cracks, as a manifestation of structural degradation during road service, exhibit distinct textural characteristics. However, fine cracks occupying minimal pixel areas demonstrate low contrast against background textures and high susceptibility to noise interference. These inherent challenges cause conventional segmentation algorithms to struggle with accurate crack extraction. Recent breakthroughs in deep learning, particularly convolutional neural networks, have driven rapid advancements in image-based pavement crack segmentation technologies, offering solutions to these longstanding limitations.

In recent years, the digital image processing methods [1], [2] based on machine learning [3], [4] and neural networks [5] have demonstrated transformative potential across computer vision domains, including image classification [6], target detection [7], and semantic segmentation [8]. These methods have been proven to have faster detection speeds, greater accuracy, and more convenience than conventional manual inspection techniques.

Dang et al. [9] designed an enhanced YOLOFNC network based on the YOLOv7 model, which achieved better results by comparing the detection results with other models on four different datasets through the newly designed C3-faster module, the introduction of the CA attention mechanism, and the incorporation of normalized Wasserstein distances into GIoU (Generalized IoU). Yang et al. [10] proposed a new network structure called feature pyramid and hierarchical advance network, which combined contextual information with low-layer features in the form of the feature pyramid and nested the samples weighted hierarchically during training so that complete crack contour information were preserved in the detection results, however the algorithm missed detection for small cracks. Cao et al. [11] proposed a deeply parallel feature fusion module that utilized the SE-Net attention mechanism and Blur-pool pooling operation to remove complex backgrounds, and the problem of crack discontinuities in segmentation results was resolved. Inspired by Seg-Net, Chen et al. [12] proposed a segmentation model PCSN (Pavement Crack Seg-Net) for crack detection, which accepted images of arbitrary size as input data and outperformed other algorithms in crack detection on the same dataset. Cao et al. [13] proposed a crack detection method based on deep fully convolutional networks and evaluated the performance of three different pre-trained network frameworks as the backbone of convolutional neural network encoders, and all three pre-training frameworks achieved good crack detection results, but the network had too many parameters and the network training took a long time. Wang Y et al. [14] proposed an encoder-decoder semantic segmentation network named RUC-Net (Residual U-Net based Convolutional Neural Network) for pixel-level pavement crack image segmentation, the spatial channel squeezing and excitation attention modules were introduced to improve detection effect, and the focal loss function was used to deal

Manuscript received June 29, 2024; revised April 12, 2025.

This work was supported by Doctoral start-up capital project of Xi'an University of Science and Technology (6310120503) and the National College Student Innovation and Entrepreneurship Project (202110704022).

Yun Bai is a senior engineer of the Engineering training center, Xi'an University of Science & Technology, Xi'an, 710054, China(corresponding author; e-mail: 494361962@qq.com).

En Lu is a graduate student of the School of electrical and control engineering, Xi'an University of Science & Technology, Xi'an, 710054, China (email: 1149680954@qq.com).

Hao-bo Wang is a graduate student of the School of electrical and control engineering, Xi'an University of Science & Technology, Xi'an, 710054, China (email: 3173911616@qq.com).

with class imbalance in pavement crack segmentation tasks, finally good detection accuracy was obtained. Kang et al. [15] proposed a new semantic transformer representation network for real-time crack segmentation at the pixel level in complex environments. The network consisted of a squeezed and excitation attention-based encoder, a multi-headed attention-based decoder, coarse up-sampling, a focus-shifting loss function, and a learnable wiggle activation function, so the fast-processing speed of the network was maintained, concise network design was realized, and the network showed good performance in the evaluation. Liu et al. [16] proposed a two-step convolutional neural network-based pavement crack detection method. The modified U-Net was proposed in the second step, and a spatial channel squeezing and excitation module were added to the up-sampling section. The method of first detection and then segmentation was adopted, and the experiments have proven that this method could improve segmentation accuracy. Ale et al. [17] proposed a crack recognition method based on a deep convolutional neural network fusion model. Crack classification and segmentation accuracy were improved by improving the network feature extraction structure and optimizing the model parameters. Cha Y J et al. [18] proposed an improved convolutional neural network. The model consisted of a standard convolution, a separable convolution module, a modified Atlas spatial pyramidal ensemble module, and a decoder module. The results showed that the network could detect the crack well unless the crack features were too ambiguous. The proposed model was compared with the latest models, and the results showed that the network had the advantages of fewer network parameters and faster computational speed. Zhang et al. [19] proposed a multi-size feature fusion network with an attention mechanism. In order to solve the problem of tiny crack loss during segmentation, a double attention module was added to the network structure to better separate the tiny crack from the background, and the edge details of tiny crack were preserved by multi-size fusion. For complex scenes in practical applications, Kang et al. [20] proposed a new STRNet (Semantic Transformer Representation Network) for pixel-level real-time crack segmentation in complex scenes. The network contained a new encoder STR module, decoder with attention module, and coarse up-sampling. The new loss function was trained in the designed network in order to improve crack segmentation performance, the lightweight and the good segmentation effect of the network were obtained. Cha et al. [21] proposed a concrete crack damage detection method based on CNNs (convolutional neural networks), aiming at the crack structure health monitoring research. By utilizing the deep structure of CNNs based on visual method, concrete cracks detection could be achieved without calculating the defect features. Crack images were tested under various shooting conditions in the experiment, and sound detection results were obtained.

The above crack segmentation methods have achieved certain results, but the processing of crack background noise is still insufficient. Due to the influence of image background noise, the crack contour information cannot be completely retained, resulting in false detection and missed detection in the segmentation of fine cracks. To solve the above problems, this paper proposes an improved pavement crack segmentation algorithm based on U-Net network. The main contributions of this paper are as follows.

Firstly, the Bot-Res module is introduced into the U-Net network to obtain complete crack information and reduce the number of parameters and calculation of the model.

Secondly, the dimensionality reduction convolution layer is added in the O-CBAM module to reduce the network parameters, and the dilated convolution layer is introduced to enhance the spatial position information of deep pixels of cracks, especially the long-distance information.

Thirdly, the EDMSF module is constructed to fuse the crack comprehensive features of different layers in the network. The module can generate the corresponding feature map using convolution kernels of different sizes according to different layers, and finally realize the multi-scale fusion of crack features.

Finally, an improved cross-entropy loss function is proposed to solve the problem of unbalanced pixel samples in the training process of crack segmentation network, so that the network can obtain more refined segmentation results.

The rest of this paper is organized as follows. In Part II, the I-U-Net network model is proposed. In Sections 2.1, 2.2, and 2.3, the Bot-Res, O-CBAM, and EDMSF modules are described, respectively, and the role of each module is discussed in the network. In Part III, the relevant experiments are given, including the establishment of datasets, evaluation indicators, comparative experiments of each module and related networks, and the experimental results, which verify the advantage of the I-U-Net network. In Part IV, the conclusion is drawn.

II. The I-U-Net Network Model

The presence of fine cracks is particularly susceptible to background noise interference, leading to the loss of critical edge feature information during feature transmission. This phenomenon results in inadequate utilization of discriminative crack features within the U-Net [22] encoder-decoder architecture, consequently compromising segmentation accuracy and failing to meet practical engineering requirements. To address these limitations, this study proposes an I-U-Net network model specifically designed to enhance performance in pavement crack segmentation tasks. The I-U-Net network structure is shown in Fig. 1.

To address the issue of crack feature information loss during transmission, we introduce a Bot-Res module to replace conventional convolutional layers in the encoder-decoder architecture. Secondly, to mitigate interference from crack background noise, an O-CBAM module is incorporated into skip connections, enhancing the fusion of shallow crack contour information and deep pixel-level spatial location features. Finally, an EDMSF module is constructed to comprehensively utilize multi-level bridge crack features across the network, thereby addressing missed detection of crack contours and fine cracks. The following sections elaborate on the I-U-Net network by introducing the Bot-Res module, O-CBAM module, and EDMSF module.

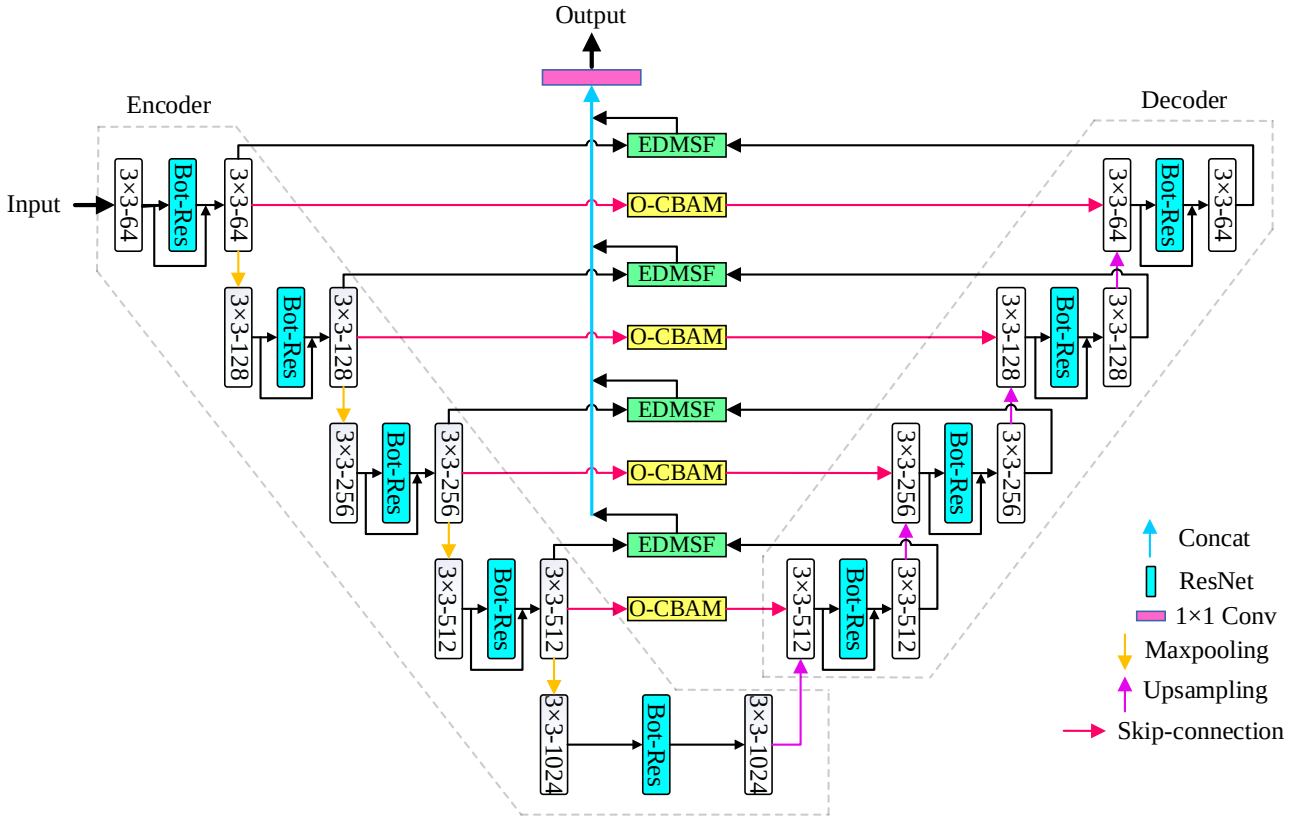

Fig. 1. The I-U-Net network

A. *The Bot-Res Module*

In this paper, we introduce the residual module to replace the traditional convolutional layer. This is done to minimize the loss of crack features during image transmission and to extract more semantic information from them. The residual module [23] employs a bottleneck structure to optimize the network. Its advantage lies in the use of multiple small-sized convolutional kernels instead of a single large-sized one. This approach reduces the number of module parameters and enhances the network's computational efficiency.

In terms of parameters, taking the third layer of the encoder as an example, the original residual module comprises two convolution layers with a convolution kernel size of 3×3. During the convolution process, each convolution layer has a parameter quantity of 3×3×128×128, resulting in a total parameter quantity of 294912 for the two layers. The improved residual module features three stages in the convolution process, as illustrated in Fig. 2 Input x passes through the first 1×1 convolution layer, with a parameter quantity of 1×1×128×64, then proceeds to the 3×3 convolution layer, where the parameter quantity is 3×3×64×64. Finally, it passes through the last convolution layer, where the parameter quantity in the 1×1 convolution process is 1×1×64×128. After undergoing these three convolution layers, the total parameter quantity in the process reaches 53248, 81.9% reduction. By comparing the internal parameters of the module before and after the improvement, it becomes evident that the calculation number of the improved residual module has significantly decreased. Additionally, the BN [24] (Batch Normalization) layer is inserted before each convolution layer, simplifying the network parameter adjustment process. Furthermore, the activation function ReLU is incorporated, reducing the model's sensitivity to network parameters and enhancing network learning stability.

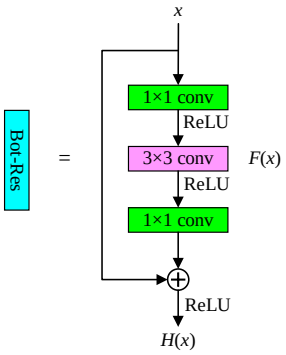

Fig. 2. The Bot-Res module

B. *The O-CBAM Module*

The CBAM (Convolutional Block Attention Module) [25], [26] is positioned at the skip connection of the network, aiming to enhance the extraction of target features while mitigating the impact of background noise. The CBAM comprises two distinct sub-modules: channel attention [27] and spatial attention [28], [29]. The channel attention module serves to bolster the acquisition of shallow crack contour information, whereas the spatial attention module focuses on amplifying the spatial position details of deep crack pixels. The integration of CBAM elevates the count of network parameters, thereby augmenting computational demands. Simultaneously, the amalgamation of crack weight information remains inadequate. The O-CBAM introduced in this paper not only diminishes the quantity of network parameters but also guarantees comprehensive integration of crack weight information. Based on the

significance of feature information, corresponding weights are formulated to weight the shallow contour details and spatial position information of deep crack pixels, thereby eliminating background noise and elevating the model's segmentation precision. The O-CBAM structure is shown in Fig. 3.

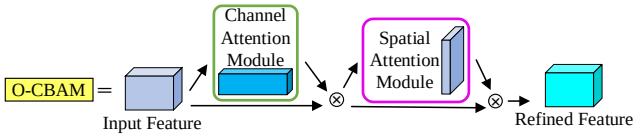

Fig. 3. The O-CBAM module

Our method reduces the number of network parameters through the channel attention module. First, the feature map x enters the channel attention module. It is then passed through a 1×1 convolutional layer to reduce the number of channels by half [30]. Next, both global average pooling and global max pooling are applied to extract shallow contour information of cracks in the lower-dimensional space, generating corresponding channel weights. Then, after a 1×1 convolutional layer to increase dimensionality, the feature map is restored to the original channel dimension while preserving spatial information of pixels. Then, the weights are allocated to each stage of the shrinkage path, and A shared MLP (Multi-layer Perceptron) [31], [32] is utilized to learn task-specific weights, filtering irrelevant features. Finally, the corresponding weight feature map is generated through the activation function. This process only deals with the channel dimension of the feature map, and preserves the spatial information of the feature map. The optimized channel attention module is shown in Fig. 4.

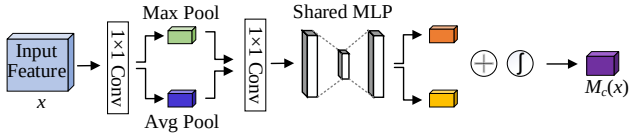

Fig. 4. The optimized channel attention module

The expression of channel attention module is as follows.

$$M_c(x) = \sigma(MLP(AvgPool(x)) + MLP(MaxPool(x))) \quad (1)$$

$$M_c(x) = \sigma(W_1(W_0(x^c_{avg})) + W_1(W_0(x^c_{max}))) \quad (2)$$

σ Represents the sigmoid activation function. W_0 and W_1 are the weights of the shared full connection layer, satisfying $W_0 \in x^{\frac{c}{r*c}}$ and $W_1 \in x^{\frac{c*c}{r}}$, where c denotes the number of feature channels, r is the reduction rate, x represents the input feature map, $\int$ expresses the sigmoid activation function. *AvgPool* and *MaxPool* indicate average pooling and max pooling, respectively, and *MLP* signifies the full connection layer.

The enhancement of shallow crack contour information is completed in the channel attention module. To effectively capture spatial information – particularly long-range contextual dependencies in deep crack regions – our spatial attention module replaces standard convolution with dilated convolution. This operation aggregates multi-scale contextual features from different receptive fields, thereby improving crack segmentation accuracy.

In the dilated convolution layer, the dilated convolution kernel with a dilation rate of 2 is used, and the convolution process is shown in Fig. 5.

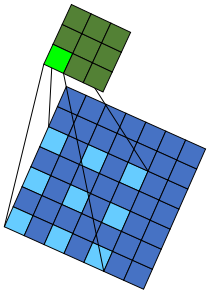

Fig. 5. The dilated convolution is 5 × 5 and the dilation rate is 2

In terms of computational complexity, the size of the convolution kernel in the feature extraction process is n×n, and the convolution operation is mathematically defined as:

$$Q(x) = \sum_{i=0,j=0}^{i=n,j=n} f(i,j) * g(i,j) \quad (3)$$

$Q(x)$ represents the feature value produced after convolution, $f(i,j)$ denotes the value of the pixel of the input image, and $g(i,j)$ means the corresponding weight in the convolution kernel. That is, when the convolution kernel size is 5×5, and 25 points multiplication operations are required. However, for the dilated convolution, the convolution kernel size is n×n during feature extraction, and its expression is as follows.

$$W(x) = \sum_{i=0,j=0}^{j=n,j=n} f(ri-1, rj-1) * g(ri-1, rj-1) \quad (4)$$

$W(x)$ represents the feature value produced after dilated convolution, $f(ri-1, rj-1)$ expresses the value of the pixel of the input image, $g(ri-1, rj-1)$ signifies the corresponding weight in the convolution kernel, and r indicate the dilation rate of the dilated convolution. In this paper, the dilated convolution with convolution kernel size of 5×5 and dilation rate of 2 is used. It can be found that Despite maintaining equivalent computational complexity to standard convolution, dilated convolution achieves a larger receptive field without increasing parameters. Generally, The dilated convolution kernel with size is n×n, which requires n×n point multiplication. However, due to the existence of dilution ratio, a larger receptive field can be obtained by n×n point multiplication. In this paper, For a dilated convolution kernel of size n×n with dilation rate r, the effective receptive field is $[(n-1)\times r+1]\times[(n-1)\times r+1]$. For example, a 5×5 kernel with r=2 expands the receptive field to 9×9. The receptive field is wider, which can more effectively fuse the weight information of long-distance cracks.

The optimized spatial attention module is shown in Fig. 6.

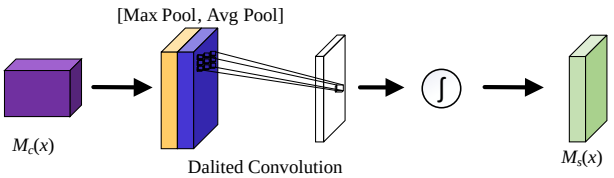

Fig. 6. The optimized spatial attention module

The expression of spatial attention module is as follows.

$$M_s(x) = \sigma(f^{5\times5}([AvgPool(x); MaxPool(x)])) \quad (5)$$

$$M_s(x) = \sigma \quad f^{5\times5}([x^s_{avg}; x^c_{max}]) \quad (6)$$

σ represents the Sigmoid activation function, x expresses the input feature map, $\int$ denotes the activation function, and $f^{5\times5}$ represents the dilated convolution with the convolution kernel size of 5×5, *Maxpool* and *Avgpool* indicate average pooling and max pooling, respectively.

C. The EDMSF Module

To address the misalignment between encoder-derived high-resolution shallow features and decoder-generated low-resolution deep features, we propose the EDMSF module. This module hierarchically fuses features through parallel convolution branches, where all branches maintain identical output dimensions via unit stride and symmetric padding. The fused features thereby retain crack topology integrity across scales, effectively connecting localized texture details with global structural context.

Combined with Fig. 7, the principle of EDMSF is described as follows.

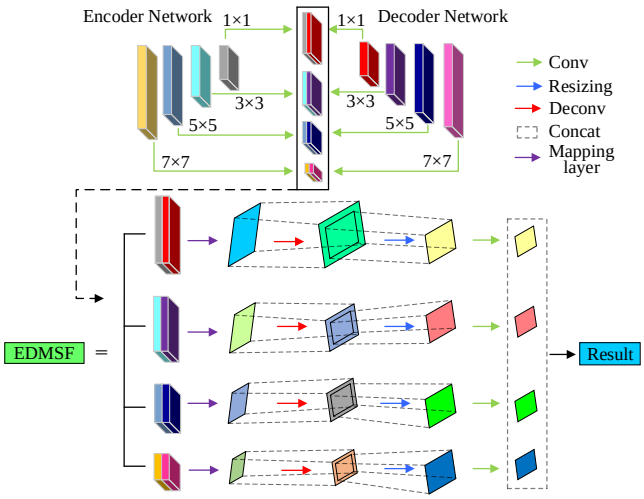

Fig. 7. The EDMSF module

The specific operational process of the EDMSF module involves fusing the feature maps of corresponding layers in the encoder and decoder through convolutional operations. Different layers employ convolutional kernels of varying sizes, and the fused results are mapped into two-dimensional feature maps. For instance, in the first layer, the 1×1 convolutional kernel is employed to extract features from the feature maps output by the encoder and decoder. followed by 3×3, 5×5, and 7×7 kernels in subsequent layers respectively. The advantage of this approach is that, compared to a single-sized convolutional kernel, it can capture multi-scale features, thereby enhancing crack segmentation performance. Due to the varying sizes of the generated two-dimensional feature maps, the feature maps of each layer must undergo up-sampling through a deconvolution layer. The resized feature maps of different sizes are standardized to identical dimensions through a resizing layer, and then concatenated. Finally, a convolutional layer generates the final prediction map and calculates the prediction loss.

III. Experiments

To demonstrate the effectiveness of the proposed I-U-Net network in this paper, we conducted experiments on the Bot-Res module, O-CBAM module, and EDMSF module, followed by an overall ablation study. The superior performance of the proposed method is further validated through comparison with other advanced models.

A. Experimental Environment

The hardware and software environment of the experiment are shown in Table I.

TABLE I
Experimental Environment

Configuration	Configuration parameter
Operating system	Windows 10
CPU	14th Gen Intel(R) Core (TM) i9-14900KF
GPU	NVIDIA GeForce RTX 4080 Super
RAM	16G
Deep learning framework	Pytorch 2.2.0
CUDA version	11.6
Python version	3.8

B. Datasets

To validate the proposed algorithm, we evaluate it on three crack datasets: the public benchmarks *CRACK500* [35] and *CFD* [36], along with a *Self-built Dataset*.

CRACK500: Contains 500 crack images with a size of 2000×1500. We rotate the image at 9 different angles (from 0 degrees to 90 degrees, spaced 10 degrees apart) and flip the image vertically and horizontally at each angle. In the end, we obtain 15200 images.

CFD: Contains 118 images marked with cracks, with a resolution of 480 × 320. On the basis of the original crack images, 3776 crack images and their corresponding annotated images are generated through methods such as blurring, brightness enhancement, brightness reduction, rotation, and horizontal mirroring.

Self-built Dataset: Contains 236 images of road cracks captured by a camera under visible light. Each image is rotated in 9 different angles (from 0 degrees to 90 degrees, spaced 10 degrees apart). Gaussian noise added to the image. Adjust the brightness, contrast, saturation, and hue of the image to obtain an enhanced dataset. The final dataset after expansion consists of 4720 samples.

As shown in Fig. 8, the above three datasets contain complex scenarios.

Fig. 8. The examples of various complex scenes

Due to the different image sizes of the above datasets, we screened 11298 images for our experiments, where the ratio of training, validation set and test sets is 8:1:1. Since the network model trained on small-sized images can scan any image larger than the cropped size, we chose a cropped size of 384×544 for our experiments.

The detailed information of the training set, validation set, and testing set are presented in Table II.

TABLE II
THE SETS FOR TRAINING, VALIDATION, AND TESTING EXPERIMENTAL ENVIRONMENT

	Training	Validation	Testing	Total
Size	384×544	384×544	384×544	384×544
Numbers	9038	1130	1130	11298

C. Experimental Parameter Design of The Model

Since the main task of the proposed I-U-Net network is segmentation, and there are two categories in the segmentation task, background, and crack, the cross-entropy function of the binary classification is selected as the loss function. During training, the epoch is set to 100, the batch size is set to 8, and the shuffle is set to True. When the learning rate is set too high, the parameter update step of the model increases, which may cause the model to oscillate back and forth near the optimal solution and unable to stably converge to the optimal solution, preventing the model from achieving optimal performance, the initial global learning rate is set to 1e-3, to achieve an accelerated model convergence process, and to regulate the effect of model complexity on the loss function, the momentum and weight decay are set to 0.9 and 0.0005, respectively. The stochastic gradient descent method (Adam) is used to update the network parameters. The training set is used to train the network model parameters, the validation set is used to evaluate the performance of the trained model, and the testing set is used to evaluate the generalization ability of the model and output the final segmentation results.

D. Evaluation Metrics

In this experiment, the pavement crack segmentation results are quantitatively analysed using Precision, Recall, F1-Score, and mIoU. The expressions are as follows.

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

$$Precision = \frac{TP}{TP+FP} \tag{8}$$

$$Recall = \frac{TP}{TP+FN} \tag{9}$$

$$F1-Score = \frac{2\times Precision\times Recall}{Precision\times Recall} \tag{10}$$

$$mIoU = \frac{1}{k}\sum_{i=1}^{k}\frac{TP}{FP+TP+FN} \tag{11}$$

Where, TP represents that the samples are divided into positive samples and allocated correctly, FP expresses that the samples are divided into positive samples but misallocated, TN denotes that the samples are divided into negative samples and allocated correctly, FN represents that the samples are divided into negative samples but misallocated. K signifies the total number of categories. Precision quantifies the percentage of correct predictions. Recall measures the proportion of positive samples captured by the model to actual positive samples. F1-Score is an important indicator that reflects Precision and Recall. mIoU indicates the mean Intersection over Union.

E. Loss

The semantic segmentation of pavement crack images is a binary classification problem, and the cross-entropy of the binary classification can be used as the loss function. Its expression is:

$$L_C = \frac{1}{N}\sum_{i=1}^{N} y_i \log yp_i + 1-y_i \quad \log\ 1-p_i \tag{12}$$

Where N is the number of image pixels, y_i represents the label value of the ith pixel, and p_i expresses the prediction probability of the ith pixel. However, in the practical situation, the area of the cracks tends to be small, which leads to uneven distribution of positive and negative samples when segmenting the background and cracks, resulting in imbalanced segmentation. To better reflect the practical situation of the segmentation results, the cross-entropy loss function is improved in this paper, and the uneven situation of the samples is balanced by the formula (12). The expression is as follows.

$$L = 1-\frac{\sum_{i=1}^{N} p_i y_i + \varepsilon}{\sum_{i=1}^{N} p_i + y_i + \varepsilon} - \frac{\sum_{i=1}^{N} 1-y_i \quad 1-p_i \quad + \varepsilon}{\sum_{i=1}^{N} 2-p_i-y_i + \varepsilon} \tag{13}$$

Where N is the number of image pixels, y_i denotes the label value of the ith pixel, p_i represents the predicted probability of the ith pixel, and ε is the regulatory factor, and its affection is to speed up convergence and prevent overfitting. The expression is as follows.

$$Loss = \alpha L_C + 1-\alpha\ L \tag{14}$$

Loss denotes total loss, L signifies binary cross-entropy loss, and L_C expresses improved loss function, α is the balance coefficient.

To determine the value of α, the experiments were carried out separately, and the experimental results obtained with all other conditions being equal are as follows Fig. 9:

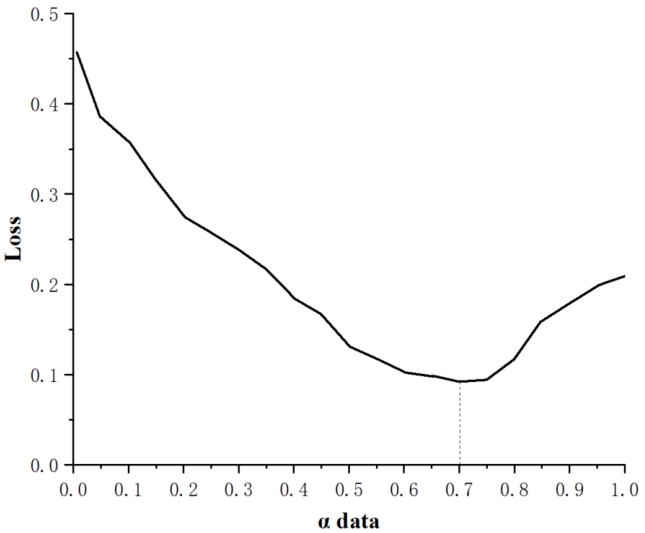

Fig. 9. The impact of different α values on model accuracy

According to the analysis of experimental results, different α has different degrees of impact on the results. Combined equation (14), the α value is 0.7.

F. The Training Process of the I-U-Net Model

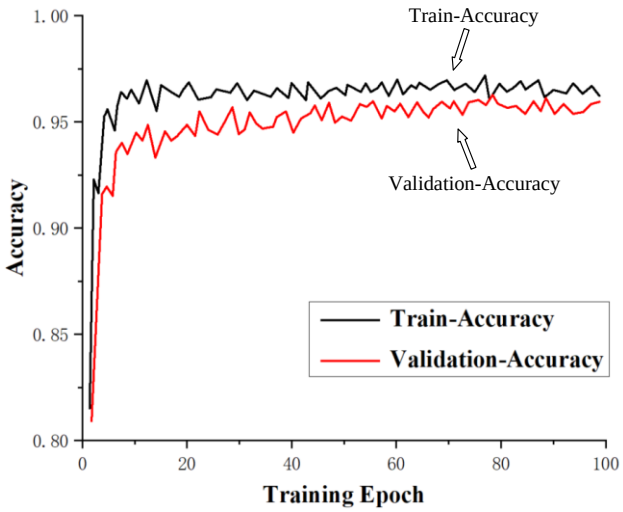

Fig. 10. Iterative Precision process of the I-U-Net model

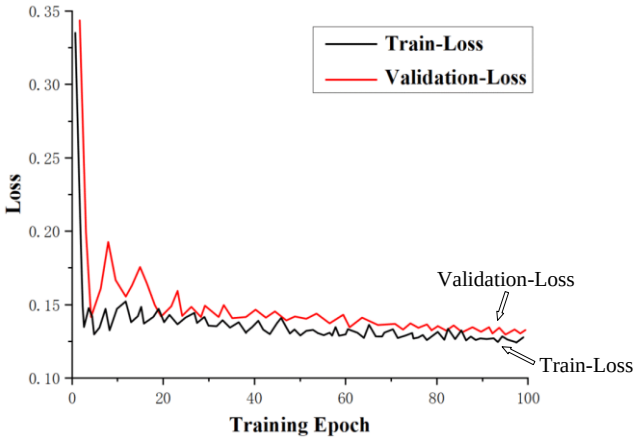

Fig. 11. Iterative Loss process of the I-U-Net model

Fig. 10 and Fig. 11 illustrates the training process of the I-U-Net model. Evidently, the accuracy of the training set ultimately converges to 95.75%, while that of the validation set converges to 95.16%. The high and closely matching final convergence values for both indicate that the model exhibits neither overfitting nor underfitting, confirming its accurate and reliable target prediction outcomes. Additionally, the loss of the training set converges more rapidly, reaching 0.13 beforehand, whereas the loss of the validation set converges to approximately 0.14. The substantial consistency in their final convergence values further underscores the advanced nature of the model presented in this paper.

G. The Bot-Res Module Experiment

Under the same experimental conditions, we compare the pavement crack segmentation performance between the original U-Net network and a U-Net network that replaces the convolution layers with Bot-Res modules.

Considering the numerous parameters of the U-Net network, the channel numbers for the encoder and decoder sections are 64, 128, 256, 512, and 1024, respectively. Therefore, in our experiments, we limit the channel count in each layer's feature map and employed fewer channels per layer for feature extraction. Specifically, we set the channel numbers for the encoder and decoder sections of the U-Net network to 64, 64, 128, 128, and 512 in the experiments, aiming to reduce the network's parameter count and achieve faster processing speeds. The experimental results are presented in Table III.

As evident from Table III, when compared to U-Net, U-Net + Bot-Res has shown improvements in Accuracy, Precision, Recall, F1-score, and mIoU by 0.42%, 2.25%, 0.6%, 1.31%, and 3.22%, respectively.

Despite being constrained by the number of channels in the feature maps, the U-Net network still boasts 7.86 M parameters. By substituting the original convolutional structure of U-Net with Bot-Res, the network parameters reduced by 1.44 M, reaching 6.42 M.

It is evident that incorporating the Bot-Res module into the U-Net network enhances the extraction of useful semantic information, elevates the network's ultimate segmentation performance, diminishes the parameter count.

H. The O-CBAM Module Experiment

To address noise interference in complex backgrounds, this paper introduces the attention mechanism to mitigate the interference. Under identical experimental conditions, we compare the network performance enhancements achieved by the CBAM and O-CBAM modules. The experimental results are presented in Table IV.

TABLE III
COMPARISON RESULTS OF ATTENTION MECHANISMS

Methods	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	mIoU(%)	Params(M)
U-Net	72.64	76.10	66.20	70.80	63.87	7.86
U-Net + Bot-Res	73.06	78.35	66.80	72.11	67.09	6.42

U-Net + Bot-Res: Use Bot-Res to replace convolutional layers in U-Net

TABLE IV
COMPARISON RESULTS OF ATTENTION MECHANISMS

Network structure	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	mIoU(%)
U-Net + Bot-Res + CBAM	77.83	81.26	78.94	80.08	77.28
U-Net + Bot-Res + O-CBAM	82.16	85.94	79.96	81.82	81.66

TABLE V
COMPARISON RESULTS OF CONVOLUTION KERNELS WITH DIFFERENT SIZES

Network structure	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	mIoU(%)
U-Net + Bot-Res + O-CBAM + 1×1	82.75	84.46	79.52	81.92	81.66
U-Net + Bot-Res + O-CBAM + 3×3	84.34	89.35	79.86	84.34	83.45
U-Net + Res-Bot + O-CBAM + 5×5	86.45	88.37	80.45	84.22	84.63
U-Net + Res-Bot + O-CBAM + 7×7	81.03	82.36	75.28	78.66	80.84
I-U-Net	95.75	96.06	86.83	91.79	92.27

TABLE VI
ABLATION EXPERIMENTAL RESULTS

No.	Module	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	mIoU(%)
#1	U-Net	72.64	76.10	66.20	65.74	63.87
#2	U-Net + Bot-Res	73.06	78.35	66.80	67.23	67.09
#3	U-Net + O-CBAM	80.27	84.78	78.94	81.76	80.56
#4	U-Net + EDMSF	81.75	83.60	81.26	82.41	78.86
#5	U-Net + Res-Bot + O-CBAM	82.16	85.94	79.96	81.59	81.66
#6	U-Net + Res-Bot + EDMSF	86.73	87.18	85.63	86.40	80.75
#7	U-Net + O-CBAM + EDMSF	92.54	93.45	87.35	90.30	89.46
#8	I-U-Net	95.75	96.06	86.83	91.79	92.27

Data from Table IV reveals that the O-CBAM module outperforms CBAM. Specifically, O-CBAM achieves higher Accuracy, Precision, Recall, F1-Score, and mIoU by 4.33%, 4.68%, 1.02%, 1.74%, and 4.38% respectively, compared to CBAM.

When the O-CBAM module integrates the weights derived from shallow crack contour information and deep pixel spatial position information, it utilizes the dilated convolution layer to enhance the receptive field characteristics, ensuring thorough fusion of weights. This aids the network in acquiring additional crack details while suppressing background noise.

I. The EDMSF Module Experiment

To verify whether the EDMSF module can acquire more location and detail information, thereby enabling more precise pavement crack segmentation, comparative experiments will be conducted under identical experimental conditions. We compare the results obtained using a single-size convolution kernel with those obtained using the multi-size convolution kernel proposed in this paper. The experimental results are presented in Table V. The single-size convolution kernels used are 1×1, 3×3, 5×5, and 7×7 in succession. I-U-Net (U-Net + Bot-Res + O-CBAM + EDMSF) is the model proposed in this paper, where EDMSF represents convolution kernels of varying sizes, specifically 1×1, 3×3, 5×5, and 7×7, depending on the layer.

By comparing the aforementioned data, it is the use of the EDMSF module results in a maximum improvement of 13.00% in Accuracy compared to the use of a single-size convolution kernel. Similarly, Precision is enhanced by up to 12.7%, Recall rate by up to 13.55%, F1-Score by up to 8.4%, and mIoU by up to 11.43%. It is observable that the single-size convolution kernel, due to its identical receptive field for the weights of crack feature information and shallow crack features across in the encoder, as well as the spatial position decoder, fails to fully integrate information, leading to the loss of semantic information and subsequently impacting target segmentation to some degree. In contrast, multi-scale fusion leverages convolution kernels of varying sizes with multiple receptive fields to enrich the extracted crack feature information, thereby enhancing the network's segmentation performance and yielding superior segmentation results. Additionally, it is noteworthy that the size of the convolution kernel is not necessarily better when larger during the information fusion process, as excessively large convolution kernels can also result in the loss of feature information.

J. Ablation Experiments

In order to validate the impact of various modules on enhancing network performance, this paper conducted ablation experiments, primarily focusing on assessing the effectiveness of Bot-Res, O-CBAM, and EDMSF in improving network segmentation performance. The experimental results are presented in Table VI.

Comparing the experimental results, adding different modules to the U-Net network model can bring different degrees of improvement to the network's performance. Comparing experiments #3 and #4, it can be found that O-CBAM improves 1.18% and 1.70% in Precision and mIoU, respectively, compared to EDMSF with the same conditions. Comparing experiments #5 and #6, EDMSF improves 4.57%, 1.24%, 5.67%, and 4.81% in Accuracy, Precision, Recall, and F1-Score, respectively, compared with O-CBAM based on adding Bot-Res module.

The I-U-Net achieved 95.75%, 96.06%, 86.83%, 91.79%, and 92.27% in Accuracy, Precision, Recall, F1-scoer, and

TABLE VII
COMPARISON RESULTS OF ATTENTION MECHANISMS

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	average mIoU(%)
Train	92.85	93.22	93.17	93.20	94.12	93.04	92.84	93.31	93.11	94.17	93.30
Test	92.43	92.26	92.27	92.24	92.31	92.29	92.10	92.17	92.40	92.33	92.25
Total	92.64	92.82	93.05	92.83	93.30	92.66	92.52	93.06	92.69	92.43	92.80

mIoU, respectively. This is an improvement of 31.81%, 26.23%, 31.16%, 39.63%, and 44.47% over U-Net, respectively.

To assess whether the model exhibits overfitting or under we perform K-fold random verification on the trained network model. In each validation experiment, we randomly select 10% (1100) datasets from the training, test and total sets. Experiment with the trained network model and calculate mIoU. In Table VII, use "Train", "Test" and "Total" respectively to represent all mIoU obtained from a total of 30 validation sets in the training, test and total sets. The average mIoU of these three sets are 93.3%, 92.25% and 92.80%, respectively, which are close to our experimental results of 92.26%. Therefore, the verification results obtained are very close to the final performance. Through the 30-fold verification process, a total of 33000 image were processed, which proved that the I-U-Net we trained had no under fitting and over fitting. This means that the results are not from a specific training set and test set, and the experimental results are valid.

In order to verify the real-time performance of the I-U-Net model, we test three different sizes of images (1400×1080, 980×720, and 384× 544), under the same other conditions, the experimental results are shown in Table VIII.

TABLE VIII
PROCESSING TIME OF I-U-NET

Image size	Milliseconds/image (FPS)
1400×1080	112.40/9
980×720	47.10/21
384×544	16.40/61

In video verification, the I-U-Net model is capable of processing 9 crack images of 1400×1080 resolution, 21 crack images of 980×720 resolution, and 61 crack images of 384×544 resolution per second, respectively. The image size (384×544) used by I-U-Net model in this paper can fully meet the real-time requirements. The network model trained with small-size images can scan any image larger than the training size, that is, the input image with large enough size is used to monitor the wide enough pavement area, which can meet the requirements of practical engineering.

K. Comparison Experiments

To validate the advancements of the proposed model, comparative experiments are conducted against representative networks including DeepCrack [37], SDDNet, CF, TCN [38], and I-U-Net. The iterative training process of the network is visualized in Fig. 12 and Fig. 13, illustrating convergence behavior and performance trends.

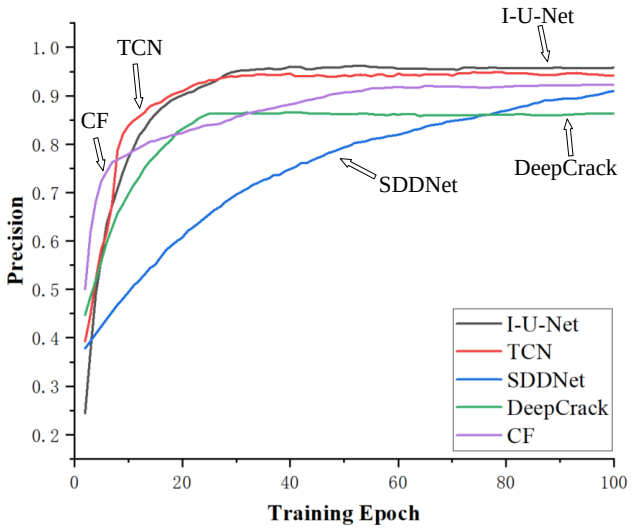

Fig. 12. Iterative Precision processes of different network models

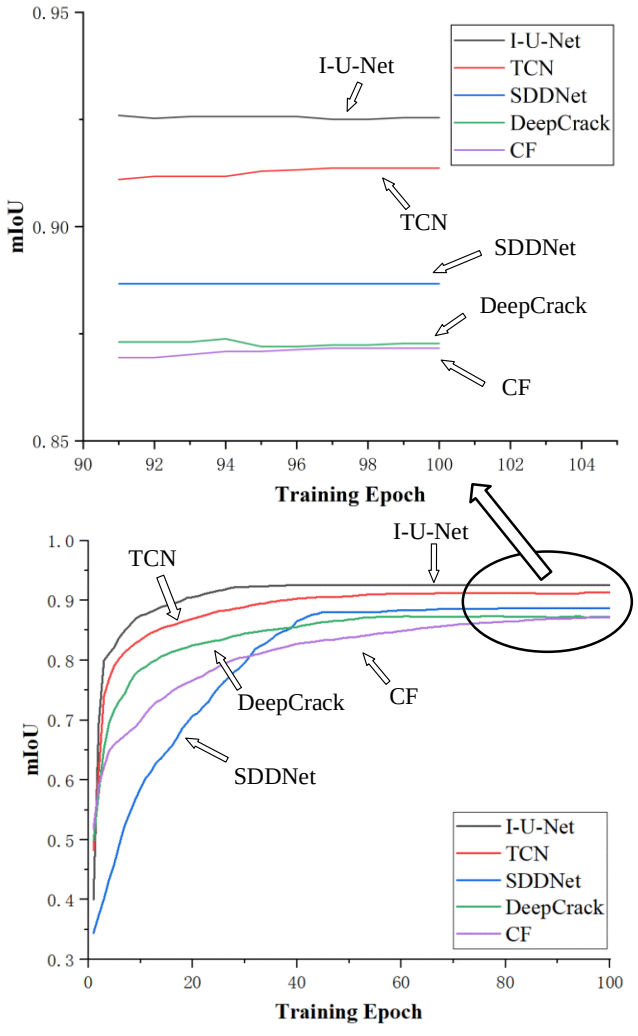

Fig. 13. Iterative mIoU processes of different network models

Experimental results demonstrate that the I-U-Net outperforms these comparative networks in both Precision and mIoU metrics.

The results indicate that SDDNet outperforms DeepCrack in terms of Precision, Recall, F1-score, and mIoU. This superiority stems from the fact that DeepCrack employs a single-sized convolution kernel for information fusion, leading to the loss of feature information. In contrast, SDDNet utilizes high-density connected separable convolution modules and an enhanced spatial pyramid pooling module, enabling it to capture crack feature information more comprehensively. Consequently, SDDNet surpasses DeepCrack in performance.

The primary distinction between CF and SDDNet lies in the latter's incorporation of an attention mechanism, which bolsters the network's capacity to extract contour features of pavement cracks, thereby enhancing the efficacy of pavement crack segmentation. In contrast to CF, the key difference of TCN is the integration of a dual attention mechanism, empowering the network to more effectively isolate micro-cracks from the background.Compared to DeepCrack, SDDNet, CF, and TCN, the I-U-Net proposed in this paper demonstrates notable enhancements in of evaluation metrics. Specifically, the Accuracy increased by 12.72%, 8.23%, 9.02%, and 2.91%, respectively; the Precision improved by 9.61%, 4.22%, 3.81%, and 1.51%, respectively; and the Recall rate risen by 7.73%, 2.41%, 1.22%, and 0.19%, respectively. Additionally, the F1-score

TABLE IX
COMPARISON RESULTS OF DIFFERENT NETWORK MODELS

Network	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	mIoU(%)	Params(M)
DeepCrack	83.03	86.45	79.10	77.33	87.13	10.8
SDDNet	87.52	91.84	84.42	78.13	88.65	12.4
CF	86.73	92.25	85.61	82.01	86.27	11.7
TCN	92.84	94.55	86.64	85.04	91.44	16.9
I-U-Net	95.75	96.06	86.83	87.79	92.27	11.4

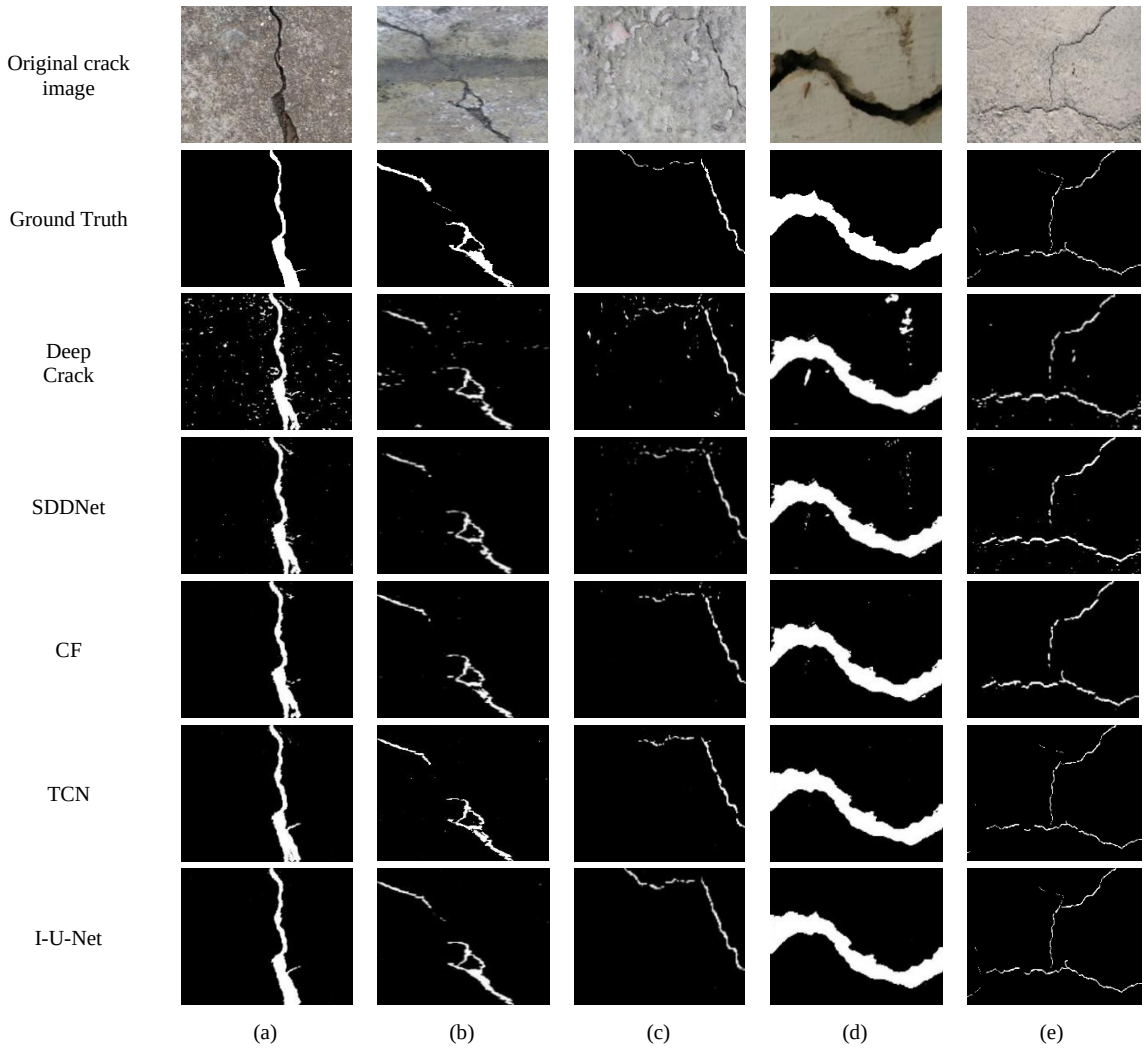

Fig. 14. Comparison of crack detection results of different networks

increased by 10.46%, 9.66%, 5.78%, and 2.75%, respectively, while the mIoU improved by 5.14%, 3.62%, 4.00%, and 0.77%, respectively.

It can be seen from Table IX, the I-U-Net proposed in this paper has more parameters than the DeepCrack network. This is attributed to the incorporation of the Bot-Res module and O-CBAM module in the I-U-Net, resulting in an increase in parameters compared to DeepCrack. In the Bot-Res module, we utilize multiple small-sized convolution kernels instead of a single large-sized one, aiming to minimize the parameter count while capturing more semantic information. Additionally, the use of dimensionality reduction convolution layers and dilated convolutions in O-CBAM helps reduce the network's parameter count, ensuring thorough integration of crack weight information.

As illustrated in Fig. 14, a comparison is made between the detection results of the I-U-Net crack segmentation model proposed in this paper and those of models such as DeepCrack, SDDNet, CF, TCN. It is evident that:

The DeepCrack network experiences missed and false detections when detecting pavement cracks, resulting in significant discrepancies between the final segmentation results and the actual cracks, as illustrated in (a) and (d).

In the detection results of the SDDNet network, pavement cracks exhibit fragmentation and missed detections, which are susceptible to background noise. Specifically, fine cracks are prone to being missed, as illustrated in (b), (d), and (e).

The segmentation results of the CF network exhibit significant deviations from the actual cracks, as illustrated in (b), (d), and (e).

The segmentation results produced by the TCN network exhibit significant deviations from the actual cracks, as illustrated in (c), (d), and (e).

The segmentation performance of the CF network surpasses that of the SDDNet network, yet it still experiences missed detections of fine cracks and is susceptible to background noise, leading to false detections as illustrated in (a), (c), and (e).

TCN incorporates a dual attention mechanism to bolster the extraction of features such as crack edges and shapes, resulting in segmentation outcomes that are more akin to the original image. However, there remains the issue of missing fine cracks, as illustrated in (a), (b), and (e).This paper introduces the I-U-Net model for pavement crack segmentation, which significantly mitigates the interference from background noise, enabling more precise segmentation of fine cracks and effectively enhancing the accuracy of pavement crack segmentation.

IV. Conclusion

To solve issues such as false positives, missed detections, and low segmentation accuracy in pavement crack segmentation processes due to background noise, this paper introduces a pavement crack segmentation model based on the I-U-Net network. Firstly, the Bot-Res module is incorporated into both the encoder and decoder of the original U-Net network. This enhancement not only reduces the computational complexity of network parameters but also ensures the extraction of crack features, thereby obtaining richer semantic information. Secondly, O-CBAM is introduced into skip connections. By utilizing dimensionality-reducing convolutional layers and dilated convolutional layers, the number of network parameters is decreased, the extraction capability for both shallow and deep crack features is enhanced, and the interference from background noise is eliminated. Lastly, within the EDMSF module, to better integrate crack features, convolution kernels of varying sizes are employed to generate feature maps of different receptive fields according to different layers. These feature maps from various layers concatenated to produce the final prediction map. Additionally, to address the class imbalance issue caused by uneven distribution of positive and negative samples, the cross-entropy loss function is modified to better reflect the actual segmentation results.

In this paper, we conduct extensive experiments on three pavement crack datasets (*CRACK500*, *CFD*, and *Self-built Datasets*) demonstrate that the proposed I-U-Net achieves superior segmentation, and compared the I-U-Net model with DeepCrack, SDDNet, CF, and TCN models. The experimental results demonstrate that the proposed I-U-Net exhibits superior performance, validating the correctness and effectiveness of the improvements made. In the ablation experiments, the improved I-U-Net achieved an Accuracy of 95.75%, Precision of 96.06%, Recall of 86.83%, F1-score of 91.79%, and mIoU of 92.27%, marking a significant improvement compared to the U-Net network. In the comparative experiments, the I-U-Net obtained the best experimental data among all compared models. Furthermore, in the experimental result figures, compared with other crack segmentation models, the proposed I-U-Net significantly suppresses background noise interference, while achieving superior crack contour accuracy. In future work, we will focus on architectural refinements of I-U-Net, particularly exploring lightweight designs and computation-efficient modules to enhance its practicality in processing high-resolution pavement crack images while maintaining real-time inference speeds.

References

[1] X. L. Bi , Y. M. Qiu , and B. Xiao, "Image histogram equalization detection method based on statistical features," Chinese Journal of Computers, vol. 44, no. 2, pp. 292-303, 2021.

[2] A. Anju and J. Anitha, "Machine learning based image processing techniques for satellite image analysis-a survey," 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon), IEEE, pp. 119-124, 2019.

[3] T. Milan, "Analysis of convolutional neural network based image classification techniques," Journal of Innovative Image Processing, vol. 3, no. 02, pp. 100-117, 2021.

[4] Y. J. Cha, W. Choi, and G. Suh, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," Computer-Aided Civil and Infrastructure Engineering, vol. 33, no. 9, pp. 731-747, 2018.

[5] Y. Liu, Y. B. Lei, and J. L. Fan, "Review of Image Classification Technology Based on Small Sample Learning," Acta Automatica Sinica, vol. 47, no. 2, pp. 297-315, 2021.

[6] Z. Q. Zhao, P. Zheng, and S. Xu, "Object detection with deep learning: A review," IEEE transactions on neural networks and learning systems, vol. 30, no. 11, pp. 3212-3232, 2019.

[7] W. Weng and X. Zhu, "INet: convolutional networks for biomedical image segmentation," IEEE Access, vol. 9, pp. 16591-16603, 2021.

[8] Z. Wang, E. Wang, and Y. Zhu, "Image segmentation evaluation: a survey of methods," Artificial Intelligence Review, vol. 53, no. 8, pp. 5637-5674, 2020.

[9] L. X. Dang, G. Liu, Y. E. Hou, and H. Y. Han, "YOLO-FNC: An Improved Method for Small Object Detection in Remote Sensing Images Based on YOLOv7," IAENG International Journal of Computer Science, vol. 51, no. 9, pp. 1281-1290, 2024.
[10] F. Yang, L. Zhang, and S. Yu, "Feature pyramid and hierarchical boosting network for pavement crack detection" IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 4, pp. 1525-1535, 2019.
[11] H. Cao, Y. Gao, and W. Cai, "Segmentation Detection Method for Complex Road Cracks Collected by UAV Based on HC-Unet++," Drones, vol. 7, no. 3, pp. 189, 2023.
[12] T. Chen, Z. Cai, and X. Zhao, "Pavement crack detection and recognition using the architecture of segNet," Journal of Industrial Information Integration, vol. 18, pp. 100144, 2020.
[13] C. V. Dung, "Autonomous concrete crack detection using deep fully convolutional neural network," Automation in Construction, vol. 99, pp. 52-58, 2019.
[14] G. Yu, J. Dong, and Y. Wang, "RUC-Net: A Residual-Unet-Based Convolutional Neural Network for Pixel-Level Pavement Crack Segmentation," Sensors, vol. 23, no. 1, pp. 53, 2022.
[15] D. H. Kang and Y. J. Cha, "Efficient attention-based deep encoder and decoder for automatic crack segmentation," Structural Health Monitoring, vol. 21, no. 5, pp. 2190-2205, 2022.
[16] J. Liu, X. Yang, S. Lau, X. Wang, S. Luo, V. C. S. Lee, and L. Ding, "Automated pavement crack detection and segmentation based on two-step convolutional neural network," Computer-Aided Civil and Infrastructure Engineering, vol. 35, no. 11, pp. 1291-1305, 2020.
[17] J. Huyan, W. Li, S. Tighe, Z. Xu, and J. Zhai, "CrackU-net: A novel deep convolutional neural network for pixelwise pavement crack detection," Structural Control and Health Monitoring, vol. 27, no. 8, pp. e2551, 2020.
[18] W. Choi and Y. J. Cha, "SDDNet: Real-time crack segmentation," IEEE Transactions on Industrial Electronics, vol. 67, no. 9, pp. 8016-8025, 2019.
[19] R. Y. Zhang, "Combining multi-scale features and attention mechanism for highway crack detection," Modern Electronics Technology, vol. 46, no. 3, pp. 100-104, 2023.
[20] D. H. Kang and Y. J. Cha, "Efficient attention-based deep encoder and decoder for automatic crack segmentation," Structural Health Monitoring, vol. 21, no. 5, pp. 2190-2205, 2022.
[21] Y. J. Cha, W. Choi, O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," Computer-Aided Civil and Infrastructure Engineering, vol. 32, no. 5, pp. 361-378, 2017.
[22] J. Cheng, W. Xiong, W. Chen, Y. Gu, and Y. Li, "Pixel-level crack detection using U-net," TENCON 2018-2018 IEEE Region 10 conference, IEEE, pp. 0462-0466, 2018.
[23] H, Liu, J. Yang, X. Miao, C. Mertz, and H. Kong, "CrackFormer Network for Pavement Crack Segmentation," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 9, pp. 9240-9252, 2023.
[24] J. Wang, S. Li, Z. An, X. Jiang, W. Qian, and S. Ji, "Batch-normalized deep neural networks for achieving fast intelligent fault diagnosis of machines," Neurocomputing, vol. 329, pp. 53-65, 2019.
[25] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 12, pp. 2481-2495, 2017.
[26] Y. Liu, Y. Yang, W. Jiang, T. Wang, and B. Lei, "3d deep attentive u-net with transformer for breast tumor segmentation from automated breast volume scanner," 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBS), IEEE, pp. 4011-4014, 2021.
[27] X. H. Yang, Y. Yi, Y. Juan, Y. Han, and W. Zheng, "Image Multi-Label Classification Based on Pyramid Convolution and Split-Attention Mechanism," 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), IEEE, pp. 534-538, 2021.
[28] Z. Chen, S. Tian, L. Yu, L. Zhang, and X. Zhang, "An object detection network based on YOLOv4 and improved spatial attention mechanism," Journal of Intelligent & Fuzzy Systems, vol. 42, no. 3, pp. 2359-2368, 2022.
[29] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, and D. Xu, "U-netr: Transformers for 3d medical image segmentation," Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 574-584, 2022.
[30] D. P. Bavirisetti, G. Xiao, J. Zhao, R. Dhuli, and G. Liu, "Multi-scale guided image and video fusion: A fast and efficient approach," Circuits, Systems, and Signal Processing, vol. 38, no. 12, pp. 5576-5605, 2019.
[31] X. Xie, Y. Tang, B. Tu, and Y. Yu, "HyperspectraI image classification with diverse region multi-scale feature extraction and spectral imaging," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 15, pp. 5427-5439, 2022.
[32] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, "Feature pyramid and hierarchical boosting network for pavement crack detection," IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 4, pp. 1525-1535, 2019.
[33] Y. F. Zhu, H. T. Wang, and K. Li, "A high-precision pavement crack detection network structure: Crack U-Net," Computer Science, vol. 49 no. 1, pp. 204-211, 2022.
[34] W. Li, K. Liu, L. Zhang, and F. Cheng, "Object detection based on an adaptive attention mechanism," Scientific Reports, vol. 10, no. 1, pp. 1-13, 2020.
[35] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 12, pp. 2481-2495, 2017.
[36] Z. Liu, Y. Cao, Y. 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.
[37] Y. Liu, J. Yao, X. Lu, and L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation," Neurocomputing, vol. 338, pp. 139-153, 2019.
[38] H. Chu, W. Wang, and L Deng, "Tiny-Crack-Net: A multiscale feature fusion network with attention mechanisms for segmentation of tiny cracks," Computer-Aided Civil and Infrastructure Engineering, vol. 37, no. 14, pp. 1914-1931, 2022.