Received: 10 September 2024 | Accepted: 7 February 2025

DOI: 10.1111/mice.13444

COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING WILEY

RESEARCH ARTICLE

An optimized and precise road crack segmentation network in complex scenarios

Gang Wang[1] | **MingFang He**[1] | **Genhua Liu**[1] | **Liujun Li**[2] | **Exian Liu**[1] | **Guoxiong Zhou**[1]

[1]School of Electronic Information and Physics, Central South University of Forestry and Technology, Changsha, China

[2]Department of Soil and Water Systems, University of Idaho, Moscow, Idaho, USA

Correspondence
Exian Liu and Guoxiong Zhou, School of Electronic Information and Physics, Central South University of Forestry and Technology, Changsha 410004, China.
Email: exianliu@csuft.edu.cn and GuoxiongZhou01@hotmail.com

Funding information
Changsha Municipal Natural science Foundation, Grant/Award Number: kq2014160; Hunan Provincial Natural Science Foundation Project, Grant/Award Number: 2025JJ50385

[Correction added on March 5, 2025, after first online publication: The author's first affiliation and Correspondence section have been updated.]

Abstract

Road cracks pose a serious threat to the stability of road structures and traffic safety. Therefore, this paper proposes an optimized accurate road crack segmentation network called MBGBNet, which can solve the problems of complex background, tiny cracks, and irregular edges in road segmentation. First, multiscale domain feature aggregation is proposed to address the interference of complex background. Second, bidirectional embedding fusion adaptive attention is proposed to capture the features of tiny cracks, and finally, Gaussian weighted edge segmentation algorithm is proposed to ensure the accuracy of crack edge segmentation. In addition, this paper uses the preheated bat optimization algorithm, which can quickly determine the optimal learning rate to converge the equilibrium. In the validation experiments on the self-built dataset, mean intersection over union reaches 80.54% and precision of 86.38%. MBGBNet outperforms the other seven state-of-the-art crack segmentation networks on the three classical crack datasets, highlighting its advanced segmentation capabilities. Therefore, MBGBNet is an effective auxiliary method for solving road safety problems.

1 | INTRODUCTION

The durability of roads is essential to ensure the stable operation of modern transportation networks; however, long-term traffic loads and environmental factors often lead to cracks in the pavement (Chen & He, 2022). This not only affects the service life of the road but also poses a potential threat to traveling safety (Bavelos et al., 2024). Over time, the strength and stability of the pavement structure will decrease and the road condition will deteriorate (Lau et al., 2020).

Since manual methods of detecting road cracks (Zheng et al., 2022) are inefficient and costly to meet the needs of modern transportation networks, researchers have turned to advanced techniques such as computer vision, image processing, and machine learning to improve the efficiency and accuracy of road crack detection (Çelik & König, 2022) In addition, researchers often use Unet++ (Zhou et al., 2019) for crack segmentation, which performs well on the common task of segmenting roadway cracks (Chen & He, 2022), but performs poorly in the following areas, as shown in Figure 1A: (a) Due to the influence of light, shooting distance, shooting angle, equipment difference, and other factors, it is not accurate to extract crack features from crack images with complex background, and the image segmentation accuracy

is low. (b) Conventional attention mechanisms may not be sufficient to retain all fine feature information when dealing with tiny cracks, especially after multiple downsampling, and these critical details may be overlooked. (c) Cracks have complex topologies, and their edge profiles are not only variable but also irregular in shape, and these features increase the difficulty of accurately recognizing and segmenting crack edges. (d) Learning rate (LR) is a key hyperparameter in deep learning, which optimizes the crack detection performance by adjusting the model weights, and is crucial for improving the model learning capability and performance. Improperly setting the learning rate, whether too large or too small, may prolong the time for the network to find the optimal value of function convergence and affect the segmentation performance.

In order to solve the problem of inaccurate segmentation in complex backgrounds, Siriborvornratanakul

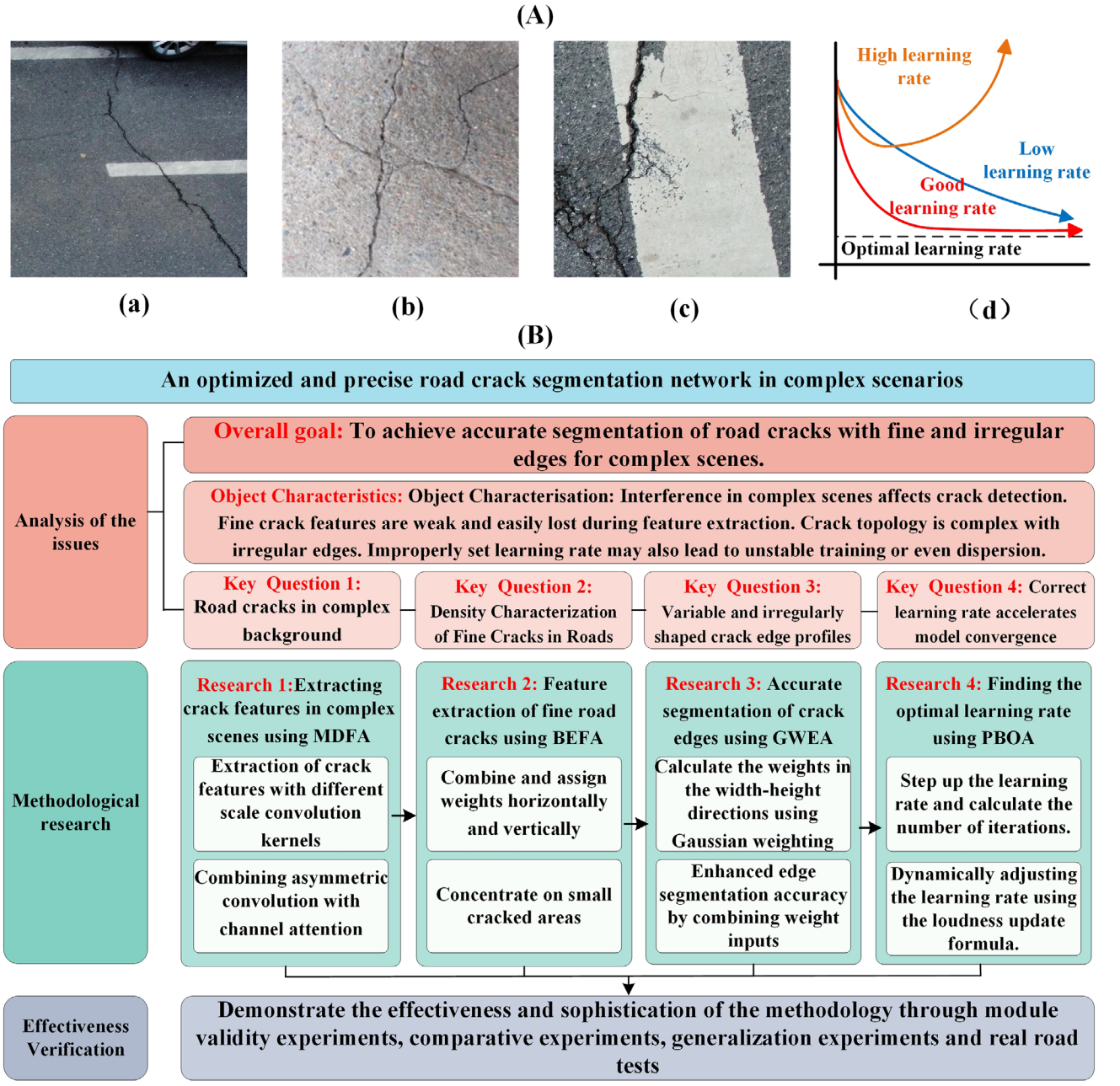

FIGURE 1 (A) Denotes problems of road crack detection where (a) denotes cracks disturbed by background noise, (b) denotes tiny cracks, (c) denotes irregular crack edges, and (d) denotes the effect of different learning rates on model convergence. (B) denotes the structure of this article. BEFA, bidirectional embedding fusion adaptive attention; GWEA, Gaussian weighted edge segmentation algorithm; MDFA, multi-scale domain feature aggregation; PBOA, preheated bat optimization algorithm.

(2022) proposed a downstream model to accelerate the pixel-level deep learning model for crack detection by using the focal loss function to train an off-the-shelf semantic segmentation model for DeepLabV3-ResNet101, which has a better performance on multiple datasets. Xiang et al. (2023) proposed a dual encoder network (DTrC-Net). This network combines transformer and neural network and adds feature fusion and residual paths to form a feature extraction module for capturing global contextual information of crack images. However, the above methods are not effective in enhancing the feature representation of cracks in the image when dealing with complex background interference. Therefore, the paper proposes a multi-scale domain feature aggregation (MDFA) to solve the interference problem of complex background, which enhances the channel dimension of single-layer feature mapping, fuses different channels with high- and low-level crack features, reduces the interference of complex background information, and improves the segmentation accuracy of cracks.

To solve the crack segmentation problem for tiny cracks, Cui et al. (2021) proposed an improved full convolutional neural network Att-Unet based on the attention mechanism, which realizes end-to-end pixel-level crack segmentation by introducing the Attention Gate (AG) module. Compared with the mainstream semantic segmentation models Fully Convolutional Networks (FCN) and Unet, the Att-Unet model shows better generalization ability and achieves better results in terms of accuracy and F1 score. Qiao et al. (2021) proposed a feature-aggregated crack segmentation algorithm: crackDFANet. This algorithm combines the attention mechanism module scSE with spatial channel squeezing and excitation to enhance the anti-jamming capability. crackDFANet exhibits strong robustness under challenging conditions (e.g., water damage and plant interference) and a generalization capability. However, the above attention mechanisms do not capture the features of tiny cracks well. Therefore, this paper proposes two-way embedding fusion adaptive attention to solve the above problem, which can combine horizontal and vertical weights to capture micro-cracks and enhance their features.

To address the challenge of crack edge segmentation, Chu et al. (2022) utilizes an improved residual network to capture the local features of cracks and integrates a dual-attention module in the architecture. It preserves the edge details of tiny cracks through multi-scale fusion operation. Han et al. (2021) developed an improved Otsu method that combines edge detection and a decision tree classifier for crack identification in asphalt pavements. They evaluated four state-of-the-art (SOTA) edge detection operators and concluded that Canny edge detection showed excellent performance in crack edge segmentation. However, the above method is difficult to accurately segment crack edges. Therefore, this paper combines a Gaussian filtering algorithm with a convolutional kernel applied to each crack pixel to smooth the image and remove the noise so as to preserve the edge features of the cracks.

Aiming at the problem of insufficient optimization of the learning rate, Tang et al. (2021) introduced different adaptive LR strategies to regulate the LR of the model parameters, which improved the classification performance of the model after fine-tuning and optimizing the model by setting the original LR to the global LR strategy. L. Liu et al. (2019) proposed a new adaptive optimization algorithm, Radam, after observing the early training of Adaptive Moment Estimation (Adam), it was found that there were problems with the use of warm-up as a method to reduce variance. Subsequently, by changing the initial values and choosing a larger variance to train the model, Radam was able to converge quickly in a variety of tasks with different network architectures, improving the performance of the network. Therefore, this paper combines the warm-up bat optimization algorithm to find the optimal learning rate for model training by searching for the optimal solution in advance through the warm-up phase. The contributions of this paper are as follows:

1. Acquiring and constructing the crack dataset (MCCD) through the network and intelligent camera devices devices. The MCCD contains a total of 2342 road images with different complex backgrounds and various shapes of cracks, which are labeled with high accuracy and help the model to obtain better training results.
2. A new road crack segmentation model MBGBNet is proposed in this paper:
   a. In order to better solve the problem of complex background, the paper proposes MDFA. The method utilizes convolutional kernels of different scales to extract features and uses adaptive asymmetric modules to process the complex background information and improve the overall segmentation accuracy.
   b. To better solve the problem of tiny cracks, this paper proposes the bidirectional embedding fusion adaptive attention (BEFA). It enhances crack characterization by combining horizontal and vertical weights to capture tiny cracks.
   c. In order to better solve the irregular crack segmentation problem with undulating edges, this paper proposes a Gaussian weighted edge segmentation algorithm (GWEA). It realizes the accurate segmentation of crack edges by combining the Gaussian weighting algorithm with image smoothing.
   d. In order to optimize the segmentation performance of the road crack detection model and effectively solve the problem of insufficient optimization of the learning rate, the paper proposes the preheated bat optimization algorithm (PBOA), which quickly

approaches the global optimal solution by setting a larger initial value of the learning rate in advance.
3. MBGBNet outperforms seven state-of-the-art segmentation networks in comparison and generalization experiments on self-built datasets and three public datasets. It is demonstrated that MBGBNet has better performance under the tasks of complex background, tiny cracks, and crack edge detection.

In addition, we better demonstrate each key step and methodology in the study in the flowchart image in Figure 1B.

2 | DATASETS AND METHODS

2.1 | Collection of datasets and processing

The collection of a crack dataset is crucial for the road crack segmentation task, which not only provides the base data for model training and optimization but is also used for model evaluation and performance analysis to ensure the accuracy and reliability of the model (Van Hauwermeiren et al., 2022). The traditional crack dataset has a simple background and a single crack shape; therefore, we conducted road dataset acquisition by cell phone and unmanned aerial vehicle (UAV) on Xiangzhang Road and Furong South Road in Changsha downtown, Hunan Province, and selected some of the crack images on the network, which constitutes the MCCD dataset, containing 2342 crack images with a resolution of 4096 × 3072, which were saved in JPG format. The MCCD dataset consists of the following parts: (a) complex background cracks (containing light variations, shadow interference, and obstacle occlusion effects); (b) tiny cracks (tiny cracks gradually produced by the pavement material after a long period of time); (c) cracks with irregular edges (due to the crushing by vehicles or natural weathering, their edges show irregular shapes), and (d) cracks with distinctive features (easily recognizable by a single and prominent shape feature).

Considering the differences in shooting angles and environments as well as the inconsistency in the acquisition of the crack dataset, and to avoid fluctuations in image quality that may be encountered when shooting with a cell phone, we removed some of the images with poor image quality and insignificant crack features in the data preprocessing stage. Subsequently, in order to standardize the data and facilitate the computational efficiency during model training and inference, we uniformly resized the remaining images to a size of 256 × 256 pixels.

We use the interactive annotation tool Labelme software for image annotation by drawing and editing the annotations directly on the image. In this study, we labeled the crack pixels in red and the background pixels in black to achieve pixel-level accurate labeling. Due to the irregular and non-uniform crack boundaries, bifurcation and fracture are likely to occur, which greatly increases the difficulty of crack marking. The average labeling time for each image is 20 to 30 min. After labeling, we use JSON format to save the labeled data to record the labeling information of each pixel, which is convenient for subsequent data processing and model training.

TABLE 1 Flow of the road crack segmentation network (MBGBNet) algorithm.

Algorithm: All experiments in this paper used the default values

Adam ($\partial = 0.00004, m = 0.9,$), batch size = 8, Input Size = 256 × 256

Require: ∂, the start learning rate. m, the momentum

1: while θ has not converged

2: **for** epoch = 0, …, n **do**

3: Image→MDFA→(BN+ReLU)

4: Down (MDFA+BN+ReLU, …, MDFA+BN+ReLU)

5: Skip connection (MDFA+BN+ReLU, …,)

6: MDFA→(BN+ReLU) →BEFA→PBOA→…

7: Up (MDFA+BN+ReLU, …, MDFA+BN+ReLU)

8: Mapping all features→Output Image

9: Loss(Output, Labe)→PBOA (Update Learning Rate)

10: **end for**

Abbreviations: Adam, Adaptive Moment Estimation; BEFA, bidirectional embedded fusion adaptive attention block; BN, batch normalized; MDFA, multi-scale domain feature aggregation; PBOA, preheated bat optimization algorithm.

2.2 | MBGBNet

In this paper, we propose a novel road crack segmentation network, MBGBNet, which is based on the Unet++ architecture and aims to solve the problems of complex backgrounds, tiny cracks, and irregular edges that occur in road crack segmentation. Unet++ efficiently extracts the image features through an encoder and performs upsampling and feature fusion through a decoder to achieve accurate recognition of cracks. The design of MBGBNet makes full use of the codec structure of Unet++ to enhance the network's ability to capture crack features through the MDFA, the BEFA, and GWEA.

The working principle of the network architecture, the way layers are connected to each other, and the cooperative working mechanism of each component are shown in Table 1. The MBGBNet network architecture is divided into five layers, and each layer is equipped with an MDFA

module with skip connections. The input images of the network are first subjected to initial feature extraction by MDFA, a module capable of isolating sensitive crack features from complex backgrounds. Subsequently, the features are batch-normalized (BA) to enhance the stability of the model and to mitigate the gradient vanishing problem by correcting the linear unit (ReLU).

The extracted features are then passed to other MDFA modules in the same layer. In the deeper layers of the network, the image features are further processed through down-sampling operations, while the information is passed back to the MDFA modules in the upper layers through skip connections. This inter-layer feature transfer and information-sharing mechanism ensures the effective propagation of crack features through the network. After extracting the basic features of the cracks, the MDFA of the fifth layer goes through BEFA to process the tiny cracks in the image features and extracts the segmentation of the edge features of the cracks through GWEA, and finally all the data will be mapped to the MDFA outputted from the first layer. In addition, the PBOA dynamically adjusts the learning rate in order to optimize the training process of the network in the course of training, which ensures that the model can converge quickly and achieve better performance. With this design, MBGBNet is able to accurately identify and segment cracks in a variable road environment, improving the efficiency and safety of road maintenance.

2.3 | MDFA

Convolution is a basic neural network operation (Ghosh et al., 2020) that filters out noise from an image and extracts effective features from it. Traditional single-scale convolution can only learn a limited number of features and is unable to handle the interference of complex backgrounds and effectively segment road cracks. The proposed MDFA is used to solve the above problem. Since the background of pavement cracks has the effects of roughness, unevenness, stains, and so forth, MDFA can effectively remove the complex background information by going to extract the relevant information from different scales while retaining the detailed features of the cracks. As shown in Figure 2b, the image is processed with 3×3, 5×5, and 7×7 convolution kernels in the input part of the MDFA, which effectively removes the interference of the complex background by capturing the crack features at different scales.

The results obtained from the three branches are subsequently combined, the channels are compressed and up-sampled using 1×1 convolution, and the features extracted from the three channels are merged. In addition, channels containing cracks are selectively attended to using the channel's adaptive attention block (ECA-Block). The Efficient Channel Attention (ECA) module is a channel-based adaptive attention mechanism that operates in three steps. First, the ECA module globally averages the input feature map of size $w \times h \times c$ to obtain a new feature map. The new feature map has a global receptive field with a specific size of $1 \times 1 \times C$. Equation (1) explains the computation process of the ith channel feature map:

$$z_i = \frac{1}{w \times h} \sum_{p=1}^{W} \sum_{q=1}^{H} u_i(p, q) \tag{1}$$

In Equation (1), $w \times h$ denotes the original image resolution, $u_i(p, q)$ represents the number of channels as i, the element with coordinates (p, q), and the total number of channels as C; z_i is the number of feature maps for that channel. Second, the compression step generates a $1 \times 1 \times C$ vector by compressing $z \in R^c$. Equation (2) describes the cross-channel interaction process:

$$\omega_i = \sigma\left(\sum_{j=1}^{k} \omega^j y_i^j\right), \; y_i^j = \Omega_i^k \tag{2}$$

Only considering the interaction between y_i and its k neighboring nodes, Ω_i^k denotes the set of k neighboring channels of y^j, $\omega^j y_i^j$ represents the product of the weights ω^j and the number of channels at different coordinates y_i^j. The information interaction between channels is realized by a one-dimensional convolution with kernel size k (where k is adaptively determined by the mapping of channel dimension C) so that all channel numbers share the same learning parameters to ensure efficiency and effectiveness.

Eventually, the sigmoid (σ) function is employed to produce the standardized channel-wise weights $s \epsilon R^c$ and scaled to $1 \times 1 \times C$ outputs, and the pertinent channels in the initial feature map are multiplied through normalization of the channel weights. This procedure results in obtaining channel attention features.

The output is subsequently passed through a normalization layer (BN) and a linear function activation layer (ReLU) to accelerate the training of MBGBNet and improve the computational efficiency, and an asymmetric convolutional block (ACBlock) is used to enhance the robustness of MDFA and avoid the interference of the complex background to further extract the crack features. The asymmetric convolution block (ACBlock) implementation steps are shown below:

The input of the feature map with channel number C passes through the convolution layer of the convolution kernel size $H \times W$, D layer filter, using $F \epsilon R^{H \times W \times C}$

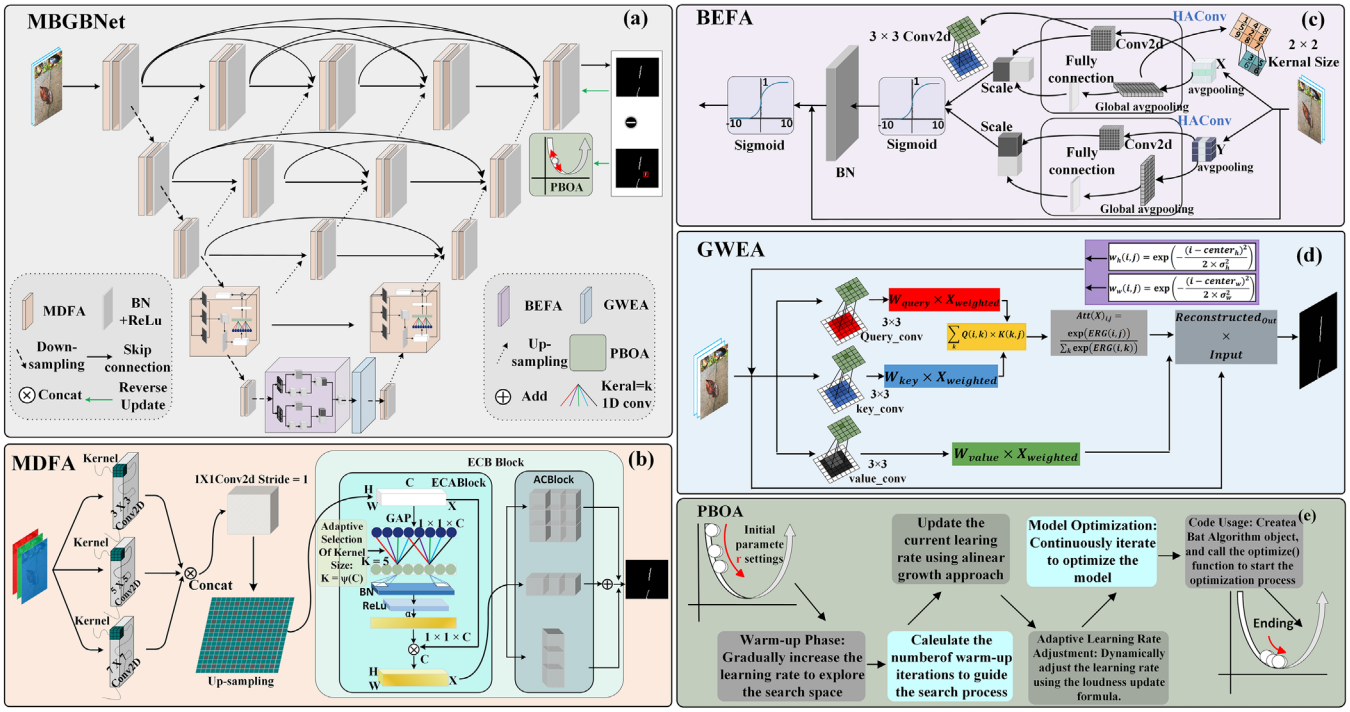

FIGURE 2 MBGBNet topology diagram: (a) denotes the overall structure of MBGBNet, (b) denotes multi-scale domain feature aggregation (MDFA), (c) denotes bidirectional embedding fusion adaptive attention (BEFA), (d) denotes the Gaussian weighted edge segmentation algorithm (GWEA), and (e) denotes preheated bat optimization algorithm (PBOA).

to denote the 3D convolution kernel of the filter and $M \epsilon R^{U \times V \times C}$ to denote the input.

The obtained feature mapping is shown in Equation (3)

$$O_{:,:,:j} = \sum_{k=1}^{C} M_{:,:,:k} * F^{j}_{:,:,:k} \tag{3}$$

where $M_{:,:,:k}$ is the kth channel of M represented as a $U \times V$ matrix, the representation is enhanced by the batch normalization layer after which a linear variation is achieved and the output equation is shown in Equation (4):

$$O_{:,:,:j} = \left(\sum_{k=1}^{C} M_{:,:,:k} * F^{j}_{:,:,:k} - \mu_j \right) \frac{\gamma_j}{\sigma_j} + \beta_j \tag{4}$$

where μ_j and σ_j represent the BN channel mean and standard deviation, and γ_j and β_j denote the learned scale factor and deviation term, respectively. Equivalent kernels with the same output are generated by applying multiple 2D kernels of compatible sizes on the same inputs with the same step size. These kernels are then combined through addition to produce the same output as shown in Equation (5)

$$I * K^{(1)} + I * K^{(2)} = I * \left(K^{(1)} \oplus K^{(2)} \right) \tag{5}$$

The three BN fusion branches are merged into a standard convolutional layer by incorporating asymmetric kernels into the corresponding positions of the square kernel. For each filter j, the fused 3D kernel is represented as $F'^{(j)}$, the obtained bias term is denoted as b_j, and $\bar{F}^{(j)}$ and $\hat{F}^{(j)}$ denote the kernels of the filters with dimensions 1×3 and 3×1, respectively. The obtained fusion kernel bias term is shown in Equation (6)

$$F'^{(j)} = \frac{\gamma_j}{\sigma_j} F^{(j)} \oplus \frac{\bar{\gamma}_j}{\bar{\sigma}_j} \bar{F}^{(j)} \oplus \frac{\hat{\gamma}_j}{\hat{\sigma}_j} \hat{F}^{(j)} \tag{6}$$

Details of the experiments on the effectiveness of MDFA can be found in Section 3.3.1, and the results of the related experiments are represented in Table 5.

2.4 | BEFA

Tiny cracks in road cracks are difficult to segment. Traditional models struggle to distinguish them effectively. The attention framework is inspired by the manner in which humans process visual information and achieves accurate extraction of specific features in neural networks by utilizing the weight allocation coefficient (Z. Liu et al., 2025).Traditional attention mechanism algorithms usually only assign weights in one direction, resulting in the loss

of feature information. In order to effectively utilize the limited visual information processing resources, this paper proposes BEFA, which focuses attention on the tiny crack region, as shown in Figure 2c.

By combining and assigning weights in the horizontal and vertical directions, feature map channels with tiny crack information are emphasized, while channels with interference information are suppressed. This module enhances the feature representation of the crack image, enabling the model to capture more precise and comprehensive details while minimizing the influence of uncorrelated regions. As a result, the image classification outcome incorporates multi-scale information, thereby enhancing the model's capability to generalize across different scales. The BEFA structure is divided into three parts:

1. Embedding coordinate information: As shown in Figure 2c, the input is a crack feature map with dimensions of $16 \times 16 \times 512$. First, global pooling is embedded in the feature map for perform decomposition, which transforms it into a one-dimensional feature encoding operation as described in Equation (7):

$$Z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_c(i, j) \tag{7}$$

Second, two spatial range pooling kernels $(H, 1)$ and $(1, H)$ are used to encode each channel. This results in weighted outputs for horizontal and vertical coordinates as shown in Equations (8) and (9):

$$Z_c^h(h) = \frac{1}{W} \sum_{0 \le i \le W} x_c(h, i) \tag{8}$$

$$Z_c^w(w) = \frac{1}{H} \sum_{0 \le j \le H} x_c(j, w) \tag{9}$$

2. Feature extraction: The interdependence between channels in the input feature mapping is learned by conducting feature aggregation of spatially sensed features in two different directions. This is achieved using the HAConv module. By incorporating a 3×3 convolutional layer, a global average pooling layer, a full connectivity layer, and linear computation, the HAConv module facilitates the exploration of the interconnections among input feature mapping channels. In the global feature learning branch, the input undergoes global average pooling and fully connected layers to capture the relationship between the input feature mapping channels, resulting in 1×1 vector $A_i (i = 1, 2, 3)$. In the spatial feature learning branch, the spatial relationship of the feature mapping is learned using 3×3 convolutional layers, producing feature map $B_i (\mathrm{I} = 1, 2, 3)$ with the same size of 16×16 as the input. Feature fusion is accomplished through linear operations to obtain a 16×16 feature map $C_i (i = 1, 2, 3)$.
3. Weight fusion: Once the shared network receives the input vectors from different directions, the channel dimensions are utilized to generate the distribution of attention weights, followed by merging the two output vectors by performing elementwise summation. The acquired feature vectors are then placed in an activation function (sigmoid) to generate A_c, $A_c \epsilon \mathrm{R}^{(1 \times 1 \times t)}$ channel attention weights (t denotes the number of channels).

Finally, after global feature learning in horizontal and vertical directions, the obtained output feature vectors are connected to a batch normalization layer and can significantly expedite the convergence of the structure, enhancing its training efficiency. Fusing the output feature map with the input Feather in different channels and spaces improves the perception of small cracks and preserves the global feature information. The channel attention extraction and fusion process, from Steps 1 to 3, is illustrated in Equations (10) and (11):

$$A_c = \sigma\{M[P_a(F)] \oplus M[P_a(F)]\} \tag{10}$$

$$F_c = F \odot A_c \tag{11}$$

In Equation (10), P_a denotes the average pooling function, M denotes the shared inclusion multilayer perceptron network, and $\oplus$ denotes the corresponding elementwise summation operation of the vector. σ denotes the sigmoid activation function, A_c is the channel attention weights generated in Step 3, $A_c \epsilon \mathrm{R}^{(1 \times 1 \times t)}$, and $\odot$ is the vector-wise corresponding elementwise multiplication operation, which multiplies the weights by the elements with the feature maps to obtain the fused attention feature map F_c, $F_c \epsilon \mathrm{R}^{(w \times h \times t)}$. In Section 3.3.2, we perform a comparison test of the BEFA module to verify its effectiveness.

2.5 | GWEA

The edges of cracks are usually irregular and have complex shapes, such as wavy, serrated, or curved. This irregular shape causes the edges of cracks to present variable contours in the image, with fractures, intersections, or forks, and even the transition area between the edge and the background is difficult to distinguish clearly. This makes the segmentation of crack edges need to consider different geometric features and deal with the diversity of edge shapes, which increases the complexity and difficulty of edge segmentation. Therefore, this paper proposes the

GWEA, which combines the Gaussian filtering algorithm to (Bilmes, 1998) smooth crack images and achieve accurate segmentation of crack edges, effectively identifying and extracting irregular shapes of crack edges.

The structure of GWEA is shown in Figure 2d. First, starting from the input tensor x with dimensions $b \times c \times w$. Next, the weight distributions of the height and width dimensions are computed using a Gaussian weighting function, that is, $w_{h(i,\ j)}, w_{w(i,\ j)}$.

$$w_h\ (i,j) = \exp\left(-\frac{(i - center_h)^2}{2 \times \sigma_h^2}\right) \tag{12}$$

$$w_w\ (i,j) = \exp\left(-\frac{(i - center_w)^2}{2 \times \sigma_w^2}\right) \tag{13}$$

where i denotes the coordinate of the height of the pixel, and j denotes the coordinate of the width of the pixel. $center_h$ and $center_w$ represent different center dimensions, σ_h and σ_w width controls the diffusion of Gaussian distribution.

The computed Gaussian weights are then applied element-by-element to the input tensor x, and a 1×1 convolutional layer is applied to the modified input tensor to project the query (Q), key (K), and value (V) spaces.

$$Q = W_{query}\ \times X_{weighted} \tag{14}$$

$$K = W_{key}\ \times X_{weighted} \tag{15}$$

$$V = W_{value}\ \times X_{weighted} \tag{16}$$

After the projection is complete, the energy matrix is obtained by computing the dot product of query and key.

$$Energy\ (i,j) = \sum_k Q(i,k)\ \times K(k,j) \tag{17}$$

In order to obtain the final attention weight (Att), the softmax is applied function to the energy matrix.

$$Att\ (i,j) = \frac{\exp(Energy(i,j))}{\sum_k \exp(Energy(i,k))} \tag{18}$$

By applying this attention weight to the value (V), a weighted value tensor is obtained. Next, the output tensor is reconstructed by multiplication between the weighted value tensor and the attention weight transpose matrix.

$$Weighte\ d_{Value(i,j)} = V(i,j) \times Att(i,j) \tag{19}$$

Finally, in order to preserve the key information of the original input, the reconstructed output tensor is multiplied by the input tensor x.

$$Output_{final} = Weighted_{Value} \times Att^T \times x \tag{20}$$

Through the above process, accurate edge segmentation of cracks is achieved while improving the performance of the model. The effectiveness experiment of GWEA is conducted in Section 3.3.3.

2.6 | PBOA

In order to achieve the optimal learning rate of the road crack segmentation model, this paper uses the PBOA. It is designed to accelerate the convergence rate of the model and improve the segmentation accuracy as shown in Figure 2e. The learning rate is a key hyperparameter that determines the rate at which parameters are updated in a model.

Nevertheless, conventional learning rate adjustment strategies may lack the necessary flexibility in certain scenarios, particularly when confronted with volatile or highly non-convex model parameter spaces. Consequently, this paper introduces the PBOA, an adaptive learning rate optimization approach. This method dynamically adjusts the learning rate according to different training stages to optimize performance, ensuring that the convergence speed and retrieval precision of the model are optimized. The core principle of our algorithm lies in dynamically adjusting the learning rate is adapted according to the progress of training iterations.

Initially, a preheating phase is employed, where a high learning rate is set to expedite convergence toward the global optimal solution. Subsequently, as training proceeds, the learning rate is systematically decreased to ensure the stability of model parameter convergence. By employing this adaptive approach, the training challenges associated with traditional fixed learning rate strategies can be effectively overcome.

Figure 2e illustrates the initial parameter setup of the PBOA, followed by its preheating phase, also known as the warm-up period. During this phase, the algorithm aims to gradually guide the quest toward optimal outcomes, starting from the initial state and progressively increasing the learning rate to broaden exploration of the search space. Notably, the warm-up duration constitutes 1/20th of the total iterations. The algorithm determines the iteration count for this warm-up period as detailed in Equation (21).

$$Warmu\ p_{iters} = \frac{Max_{iterations}}{20} \tag{21}$$

Within the range of iterations from 0 to $Warmup_{iters} - 1$, the algorithm employs a linear growth strategy to adjust the current learning rate as detailed in Equation (22).

$$LR_{current} = LR_{initial} * \frac{iteration + 1}{Warmup_{iters}} \quad (22)$$

The $LR_{current}$ represents the learning rate at a given iteration, while $LR_{initial}$ is the starting learning rate. The variable iteration indicates the current iteration count, and $Warmup_{iters}$ specifies the total number of iterations during the warmup phase.

During the initial 1/20 iterations, the learning rate rapidly increases in a linear fashion to expedite the convergence of the model. This approach aims to initiate the search with a smaller learning rate, gradually guiding it toward more promising regions, thus enhancing the efficiency of the overall training process.

Next, the bat optimization algorithm phase of learning rate updating is entered, and once the warm period is over, the algorithm will enter the soundness adjustment phase. In this phase, the algorithm employs the soundness value to regulate the adjustment of the learning rate, facilitating a more meticulous search. The algorithm determines the variation in the soundness value (loudness) by adhering to a prescribed soundness update Equation (23).

$$Loudness = \min_{loudness} + (\max_{loudness} - \min_{loudness}) * \exp(-pulse_{rate} * iteration_{current} / iterations_{\max}) \quad (23)$$

Herein, loudness serves as an indicator of the audio intensity, with $\min_{loudness}$ and $\max_{loudness}$ representing the respective lower and upper bounds of this intensity. The pulse rate parameter governs the frequency of pulses, while $iteration_{current}$ reflects the ongoing iteration count, and $iterations_{\max}$ establishes the overall limit for iterations.

Subsequently, the algorithm employs acoustic degree values to refine the learning rate, following the computation outlined in Equation (23).

$$LR_{current} = \max(LR_{current} * loudness \mid LR_{\min}) \quad (24)$$

Here, $LR_{current}$ represents the learning rate at the current stage, whereas min_learning_rate is the lower bound for this rate.

Equation (24) guarantees that the learning rate remains no lower than the minimum threshold and adapts flexibly based on soundness dynamics.

The optimization of the PBOA is conducted in its crucial search and refinement phase. Herein, the algorithm leverages a dynamic learning rate to steer its exploration, aiming to uncover the optimal solution. At each iteration, the algorithm, influenced by the prevailing learning rate, executes one of two strategies: First, it may devise a novel solution, capitalizing on the current learning rate to broaden the search domain and uncover potential superior candidates. Second, it can refine the present solution, further enhancing its quality in accordance with the learning rate. Throughout the optimization phase, the PBOA persists in exploring diverse solutions until it attains a preset iteration limit (max_iterations), at which point it terminates its search.

The PBOA efficiently overcomes the challenge of limited learning rate optimization, alleviating concerns over diminished model accuracy and protracted convergence during the terminal retrieval stage. By enhancing the optimization process, it accelerates model training convergence and elevates the precision of road crack retrieval.

TABLE 2 Experimental environmental parameters.

Hardware settings	**CPU**	**Intel Core i9-10980XE**
	GPU	NVIDIA GeForce RTX 2080 Ti
	RAM	32G
	Video Memory	32G
Software settings	**OS**	**Windows 11**
	CUDA Version	12.1.12 driver
	CUDNN	V7.0.5
	Python	3.9.7
	torch	1.12.1
	torchvision	0.13.1

3 | EXPERIMENTATION AND RESULTS

3.1 | Experimental setup and parameter setting

To ensure the reliability of the experimental results, the experiments were carried out in a unified hardware and software environment, and the specific configuration details are shown in Table 2. To accelerate the convergence speed and improve the stability of the model, we used the Adam optimizer. In addition, for the optimization of the learning rate, we introduced the PBOA with an initial learning rate of 0.00004. This algorithm adapts to the learning rate during training, and a reasonable attenuation is performed to ensure that the model retrieval performance can converge to the global optimum. To simplify the network output, we chose BCEWithLogitsLoss to cope with the problems posed by unbalanced data. The logits are the output of the inactivation function

TABLE 3 Parameters for model.

Training parameter	Setting
Loss function	BCEWithLogitsLoss
Activation function	ReLU, Sigmoid
Batch size	8
Number of epochs	500
Optimizer	Adam
Momentum	0.9
Image size	256×256
Data preprocessing	/

Abbreviation: Adam, Adaptive Moment Estimation.

(BCEWithLogitsLoss combines the sigmoid activation function and the binary cross-entropy loss). It avoids the problem of too large values of the loss function or vanishing gradients at probability extremes (0 or 1; Cai et al., 2022). Too much training can cause overfitting of the model and consume too many resources, so we have 500 training rounds. The training parameters are shown in Table 3.

3.2 | Image processing and selection of evaluation indicators

The dataset is divided in the ratio of 7:2:1, where the training set is used to train the model, while the validation set is used to randomly select images from the original dataset for evaluating the performance of the training model and 10% of the images are used for the test set, which aims to optimize the training and evaluation of the model and ensure that the model can be adequately learnt and validated at different stages, so as to improve the model's generalization ability and accuracy.

The mean intersection over union (MIOU) is an important evaluation in crack image segmentation, which calculates the intersection of the predicted segmentation result with the true value and the ratio of the union of the two sets. The Dice coefficient is a value that evaluates the similarity or overlap between two objects or sets. The equations are shown in (25) and (26).

$$MIOU = \left(\frac{TP}{TP + FP + FN} + \frac{TN}{TN + FN + FP}\right)/2 \quad (25)$$

$$Dice = \frac{TP \cap Lable}{TPLable} \quad (26)$$

In addition, precision and recall were used as indicators for model validation. The equations are shown in (27) and (28).

$$Precision\ (P) = \frac{TP}{TP + FP} \quad (27)$$

$$Recall\ (R) = \frac{TP}{TP + FN} \quad (28)$$

3.3 | Module validity analysis

This section offers a comprehensive overview of the impact and performance of MDFA, BEFA, GWEA, and PBOA techniques. The experimental results of Section 3.3.1 show that MDFA can solve the interference of complex background, and the feature extraction effect in the crack is better than the ordinary convolution. The experiments in Section 3.3.2 show that BEFA is better at extracting tiny crack features than traditional attention mechanisms. The experimental results of Section 3.3.3 indicate that the GWEA used can effectively segment crack edges and improve segmentation accuracy. The experimental results of Section 3.3.4 show that PBOA achieves convergence to equilibrium faster compared to other algorithms. The results of the ablation experiments in Section 3.3.5 show the efficiency of each individual module. The results of the experiment in Section 3.3.6 show that the segmentation ability of MBGBNet is shown to be superior to other network models through comparison tests with other models. The experimental results in Section 3.3.7 show that the MBGBNet model has strong generalization in the public dataset, CrackSegNet (Ren et al., 2020), MOACA_Crack (Qifan Wang et al., 2023), and UNFSI (He et al., 2023).

3.3.1 | Effectiveness of MDFA

In order to solve the difficult problem of crack segmentation under complex background, an MDFA is proposed in this paper, which enhances the feature expression ability of cracks, better reduces the interference of complex background, and improves the crack segmentation accuracy.

The MDFA module is mainly composed of two parts: convolution and ECABlock (Qilong Wang et al., 2020). The design of the convolution kernel is closely related to feature extraction. A single convolution kernel can only be used for feature extraction in special environments. Too many convolution kernels can greatly increase the burden on the network, resulting in larger network parameters and longer running time. To determine the best case, we compared the performance of various convolutional kernel sizes on MDFA. We used Unet++ as the backbone for our experiments using different-sized convolutional kernels combined with ECA, and the results are shown in Table 4.

The experimental results show that with the increase of convolutional kernel size, the performance indicators of the model such as MIOU, precision, recall, and Dice coefficients show a trend of increasing and then leveling off, and when three convolutional kernels are used, the MIOU of the network reaches 75.09% and the precision reaches 83.83%; with the increase of the number of convolutional kernels, the performance enhancement The amplitude

TABLE 4 Experiment results for different kernel combination.

Kernel combination	Mean intersection over union (MIOU; %)	Precision (%)	Recall (%)	Dice (%)
Backbone	69.12%	80.40%	73.10%	74.81%
3×3	72.25%	81.35%	73.85%	75.20%
5×5	73.45%	81.97%	73.91%	75.54%
7×7	71.96%	81.04%	73.37%	75.11%
$3 \times 3,\ 5 \times 5$	74.73%	82.94%	74.22%	75.93%
$3 \times 3,\ 7 \times 7$	74.04%	82.48%	74.55%	75.82%
$5 \times 5,\ 7 \times 7$	73.98%	82.21%	74.56%	75.79%
$3 \times 3,\ 5 \times 5,\ 7 \times 7$	75.09%	83.83%	**74.92%**	76.74%
$3 \times 3,\ 5 \times 5,\ 7 \times 7,\ 9 \times 9$	75.33%	84.07%	74.90%	76.88%
$3 \times 3,\ 5 \times 5,\ 7 \times 7,\ 9 \times 9,\ 11 \times 11$	75.25%	83.98%	74.88%	76.81%

Abbreviation: Dice, Dice similarity coefficient.
Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

TABLE 5 Effectiveness of the multi-scale domain feature aggregation (MDFA).

	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
Backbone	69.12%	80.40%	73.10%	74.81%
Backbone + MFC	69.38%	80.68%	73.16%	75.05%
Backbone + ECB	70.03%	81.77%	73.32%	75.52%
Backbone + MDFA	**75.09%**	**83.83%**	**74.92%**	**76.74%**

Note: MFC represents the part of Figure 2b where the dimensionality is adjusted using 1×1 convolution and up-sample after 3×3 convolution, 5×5 convolution, and 7×7 convolution concat.
Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

begins to decrease, the network has been basically fitted, and too many convolutional kernels cannot enhance the network.

In order to verify the effectiveness of MDFA, the feature extraction modules in Unet ++ are replaced with MFC, ECB, and MDFA. To observe the experimental results, the results of MIOU, precision, recall, and Dice coefficients evaluation indexes are shown in Table 5.

Incorporating MFC and ECA into the network resulted in a substantial improvement in the crack feature extraction capability of the network, exceeding the performance of the basic Unet++ model. However, after using MDFA, the performance indexes of all aspects are significantly improved, and MIOU and precision are improved by 5.97% and 3.43%, respectively, which indicates that it is more effective in the feature extraction of cracks and complex background processing. Therefore, we choose MDFA as the feature extraction block to solve the crack segmentation in complex backgrounds.

3.3.2 | Effectiveness of the BEFA

In order to accurately capture small cracks and avoid leakage detection, this paper proposes a BEFA. By capturing tiny cracks at different spatial scales, more weight is allocated to extract tiny cracks. In order to verify the effectiveness of BEFA, comparisons were made with commonly used attention mechanisms. Dice and Params were used to evaluate the effect of these modules on the model performance, and the precision of crack segmentation was compared.

Table 6 presents the experimental results showing that MIOU is enhanced by 0.47% and 0.27% after adding Squeeze-and-Excitation Networks (SE) (Hu et al., 2018) and Cbam (Woo et al., 2018) to the basic backbone network, Unet++. After adding Coordinate attention (Hou et al., 2021) to Unet++, MIOU and Dice scores were enhanced by 1.63% and 0.18%, respectively, but the enhancement was not significant, and after adding BEFA, which focuses on tiny crack features, all the metrics were effectively increased, and MIOU and precision scores were increased by 5.67% and 3.13%, respectively. Therefore, in this paper, BEFA is used to extract the tiny crack blocks. Figure 3a shows the PR curve of the model comparison experiment. The PR curve was chosen to show the model performance more intuitively. The larger the area enclosed by the closed curve in the PR curve, the better the effect of the curve. As shown in Figure 3a, all curves except (Backbone + BEFA) did not exceed their area, and the best results were

TABLE 6 Effectiveness of bidirectional embedded fusion adaptive attention block (BEFA).

	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
Backbone	69.12%	80.40%	73.10%	74.81%
Backbone + SE Attention	69.59%	81.02%	73.74%	74.85%
Backbone + Cbam Attention	69.39%	80.59%	73.67%	74.96%
Backbone + Coordinate Attention	70.75%	81.54%	74.10%	74.99%
Backbone + BEFA	**74.75%**	**83.53%**	**75.16%**	**77.47%**

Note: SE Attention (Hu et al., 2018), Cbam Attention (Woo et al., 2018), Coordinate Attention (Hou et al., 2021).
Abbreviation: Dice, Dice similarity coefficient.
Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

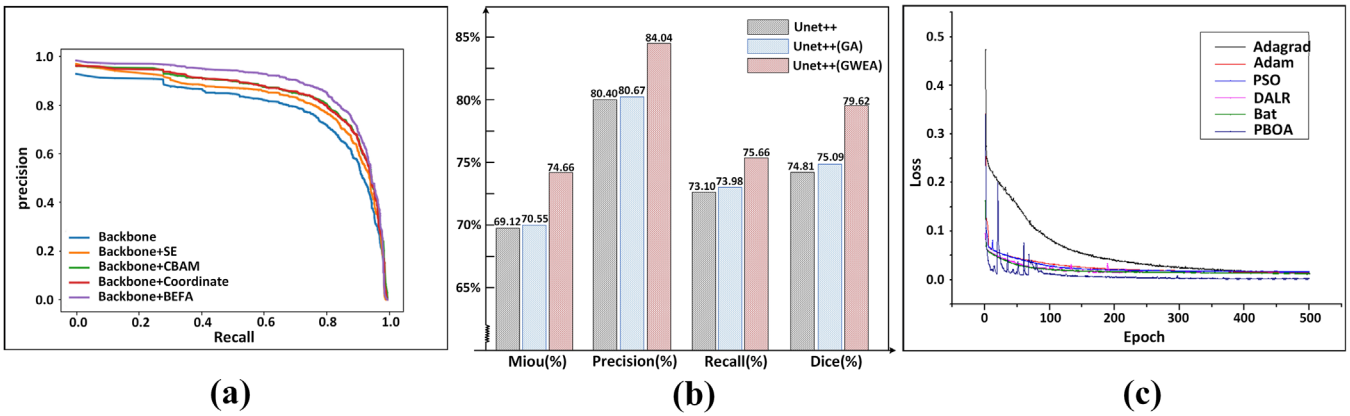

FIGURE 3 Performance comparison experiments, (a) represents the results of PR curves under different attentions, (b) represents the comparison results of GWEA, and (c) represents the loss values of different optimization algorithms under 500 rounds of training per round.

achieved on the MBGBNet dataset, which, in combination with Table 6, proved the validity of BEFA.

3.3.3 | Effectiveness of GWEA

The edges of cracks are difficult to accurately segment. In response to complex and irregular shapes, this paper proposes the GWEA. By combining Gaussian weighted weight and self-attention weight, the crack image is smoothed to effectively identify and extract the crack edges.

Using Unet++ as a benchmark, after the feature extraction is completed, a comparison experiment is conducted by adding only the Gaussian algorithm (GA) and GWEA to validate the effectiveness of the algorithm by evaluating the evaluation metrics. The experimental results are shown in Figure 3b, which is compared with the basic Unet++. By adding GA, noise can be removed to a certain extent, and MIOU and precision are improved by 1.43% and 0.27%, respectively.

A higher Dice indicates a significant overlap between the segmentation results and the actual crack region, and a higher degree of overlap at the crack edges, which also indicates that it possesses a more accurate segmentation effect on the crack edges. After the addition of GWEA, the crack edge is accurately divided by smoothing the image, and all performance indexes are improved, the Dice value is increased by 4.81%, and the recall rate score is increased by 2.56%. Therefore, GWEA edge extraction algorithm is used in this paper to improve the edge segmentation of road cracks.

3.3.4 | Effectiveness of PBOA

In order to verify that the algorithm used in this paper achieves faster convergence, balances the model, and finds the optimal learning rate while improving the prediction accuracy, the following several optimizer algorithms are compared: AdaGrad (Adaptive Gradient), Adam (Diederik, 2014), Particle Swarm Optimization (PSO) (Marini & Walczak, 2015), DALR (Deep Q-Network + Adam), Bat (Ehteram et al., 2018), and PBOA. The model was trained using each optimizer algorithm, and the changes in the magnitude of the loss function were recorded in each round, as depicted in Figure 3c.

It can be seen that the reduction in the value of losses and the number of rounds required to reach equilibrium is greater when the PSO optimization algorithm is used compared to the Stochastic Gradient Descent (SGD) and AdaGrad optimization algorithms. The minimum value is reached in the 170th round. By adding the reinforcement

learning DQN algorithm to assist in selecting the optimal descent gradient, providing a policy value function, and updating the learning rate based on the value function, DALR improves over the previous algorithms in terms of convergence speed and number of rounds to stabilize. The Bat algorithm is trained by locating the global optimal solution and updating it using echoes. Both Bat and DALR converge in only the 140th round while maintaining a strong stability.

Subsequent experiments were conducted to improve Bat's PBOA by quickly finding a larger learning rate to achieve convergence after the first search iteration after the first 50 rounds of training and balancing the loss values through continuous optimization. Compared with traditional learning rate optimization algorithms, PBOA outperforms other optimization algorithms both in terms of convergence speed and the number of rounds required to reach equilibrium. Therefore, the suitability of the proposed PBOA algorithm for road crack segmentation is confirmed by optimizing the learning rate adaptation, including improving the initial state settings.

The algorithm's performance is further validated through effectiveness experiments conducted in the module of Section 3.3.5.

3.3.5 | Ablation experiments

To evaluate the effectiveness of our proposed method, we conducted a series of ablation experiments to compare the segmentation performance of the models while keeping the experimental conditions consistent. First, this paper uses Unet++ as the backbone network for testing, followed by replacing the convolution in Unet++ with MDFA to verify the effectiveness of MDFA. After that, we add BEFA at the codec connection and GWEA to record the segmentation results, respectively. To verify the effectiveness of PBOA, at the beginning of training, we use PBOA for learning rate optimization. Subsequently, we add multiple models or algorithms by combination to compare their effectiveness. The segmentation results are shown in Figure 4.

In Column I, the background interference is small, and there is no significant difference between the cracks and the environment. Unet++ can segment the cracks effectively. However, when the cracks in the road are thin, such as in Columns II and III, Unet++ may not detect them. By adding BEFA between the down-sampling and up-sampling of Unet++, combined with horizontal and vertical weights, the information of the tiny cracks is enhanced, more tiny crack features are captured, and the ability of the network model to control the cracks is improved.

In Columns IV and V, the background around the cracks is more complicated, and there are also influences around the cracks such as manhole covers, orange lane lines, and so forth. Unet++ is unable to segment the cracks effectively. In this paper, MDFA is added to Unet++, which enhances the information extraction ability of the network by fusing multi-scale and channel concerns, effectively suppresses the interference of irrelevant background features, and realizes accurate crack segmentation. In Columns VI and VII, there are interferences of traveling cars and traffic lines in the image, and the shooting distance is farther and the field of view is wider, which makes it difficult to locate and capture the features of the cracks. By combining both MDFA and BEFA in Unet++, the proposed method improves the segmentation performance of the road cracks under the above-mentioned complex scenarios. In addition, by adding GWEA, the integrity of the cracks in Columns VIII and IX can be ensured when dealing with the irregularity of crack edges that are difficult to separate. By combining PBOA to achieve fast convergence, it helps to improve the segmentation ability of the model.

As shown in Table 7. After using MDFA instead of normal convolution in Unet++, the MIOU is improved by 5.97%, and the recall and Dice scores are improved by 1.82% and 1.93%, respectively, which, combined with the demonstration results in Figure 4, shows that MDFA significantly improves the network performance when solving complex background interference. After adding BEFA to Unet++, all aspects of the model's performance are improved to a certain extent, among which MIOU is improved by 5.63%, which further confirms the effectiveness of BEFA in tiny crack segmentation. After adding GWEA alone, Dice is improved by 5.54%, which shows that the image with irregular edge cracks can also be smoothed well. In the comprehensive experiments with one or more modules added, all methods outperform the benchmark network in improving segmentation accuracy, and when MDFA, BEFA, GWEA, and PBOA are added, MIOU reaches 80.54% (11.42%), precision reaches 86.38% (+1.98), recall reaches 81.57% (+7.14%), and Dice reaches 81.34% (+6.53%), further confirming the effectiveness of MBGBNet in dealing with road crack segmentation.

3.3.6 | Compared with advanced methods

In order to better validate the performance and effectiveness of MBGBNet, we compare MBGBNet with seven SOTA-specialized crack segmentation networks, including the lightweight tunnel crack segmentation network CrackSegNet (Ren et al., 2020), the bridge crack segmentation network BC-Unet (T. Liu et al., 2022), the morphology-transformation-enhanced evaluation crack

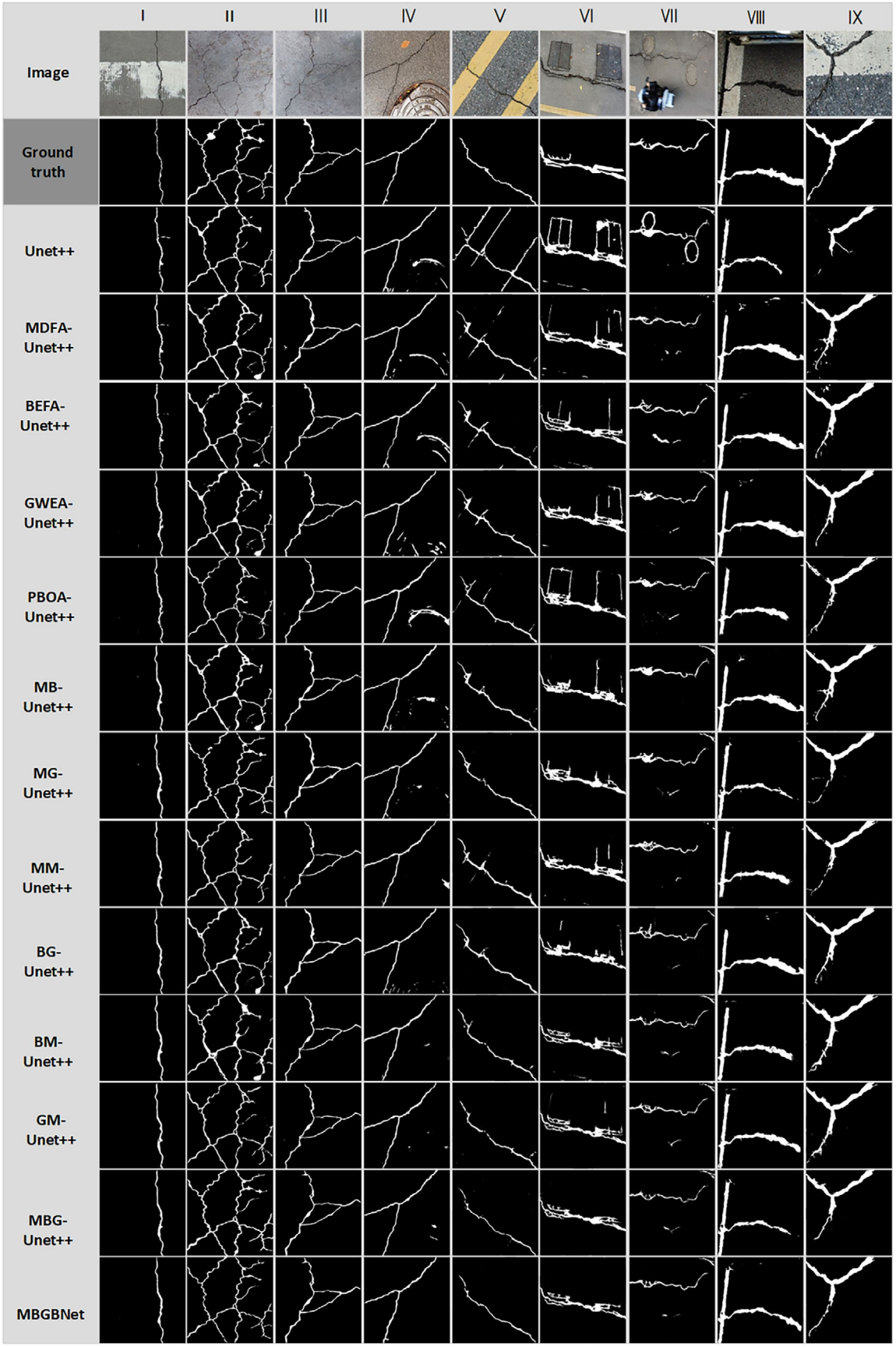

FIGURE 4 Experimental results of Ablation experiment, I for simple cracks, II and III for tiny cracks, IV–VII for cracks in complex contexts, and VIII and IX for irregular edge cracks.

TABLE 7 Experimental results of Ablation experiment.

Unet++	MDFA	BEFA	GWEA	PBOA	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
√					69.12%	80.40%	73.10%	74.81%
√	√				75.09%	83.83%	74.92%	76.74%
√		√			74.75%	83.53%	75.16%	77.47%
√			√		74.66%	84.04%	75.66%	79.62%
√				√	70.32%	81.16%	74.22%	75.20%
√	√	√			77.37%	84.13%	77.31%	78.21%
√	√		√		78.94%	85.74%	79.03%	80.13%
√	√			√	76.11%	84.27%	76.07%	77.56%
√		√	√		77.28%	84.70%	78.95%	80.35%
√		√		√	75.83%	84.01%	75.87%	77.51%
√			√	√	75.98%	84.25%	76.32%	79.99%
√	√	√	√		78.39%	85.62%	79.29%	80.95%
√	√	√	√	√	**80.54**%	**86.38**%	**81.57**%	**81.34**%

Abbreviation: GWEA, Gaussian weighted edge segmentation algorithm; PBOA, preheated bat optimization algorithm.
Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

network Unet+Aug+Morph (Yamaguchi & Mizutani, 2024a), non-crack category learning network VGG16-SOM model (Yamaguchi & Mizutani, 2024b), road crack segmentation network HC-Unet++ (Cao et al., 2023), ARD-Unet (Gao et al., 2023), and bridge crack segmentation network CCSNet (Sun et al., 2024). All models were trained under the same environmental conditions to ensure the reliability and consistency of the experiments. The crack segmentation results of these segmentation networks under different complex conditions are shown in Figure 5a.

In Column 1, the shape of the crack is obvious and single, and all models can segment it well. For Columns 2 and 3, the environment where the crack is located is not complicated, but due to the narrow width of the crack and the complex grid-like shape, Unet++ and CrackSegNet have difficulty in segmenting it accurately. At the tail end of the crack, most networks like Unet+Aug+Morph, VGG16+SOM, and so forth are difficult to segment except MBGBNet. In addition, due to the effect of shooting height and field of view range, for the complex environmental effects of manhole covers, orange lane lines, and traveling cars around the crack, as in Columns 3–7, BC-Unet, HC-Unet++, ARD-Unet, and CCSNet introduce a feature fusion and attention mechanism between down-sampling and up-sampling, which show better segmentation ability, compared to the previous networks. This mechanism increases the weight of cracks in the feature extraction process, allowing the networks to obtain more information about the cracks. However, these models still have shortcomings. For the boundary between the manhole cover and the lane line, the above networks incorrectly classify it as a crack, which affects the segmentation accuracy. For Columns 6 and 7, the cracks have irregular edges as well as interference from the background environment, and all networks except MBGBNet suffer from misdetection, omission, and obvious segmentation errors. Whether it is a single simple crack or a small crack with complex and irregular background, the MBGBNet proposed in this paper can segment road crack images more effectively and accurately under different conditions.

As shown in Table 8, MBGBNet has the best performance, compared with the seven SOTA segmentation models, and the MIOU, precision, recall, and Dice values of MBGBNet are 80.54%, 86.38%, 81.57%, and 81.34%, respectively, which are higher than the other models. Combined with the comparison experiments in Figure 5a, MBGBNet is more effective in dealing with complex backgrounds, tiny cracks, and irregular edge effect problems, which further proves the effectiveness of MBGBNet.

3.3.7 | Generalization experiments

In order to verify whether the proposed model can accurately segment cracks under different scenarios and environmental conditions, three authoritative public datasets are selected for experiments in this part. In order to evaluate the performance of the model on the tunnel crack dataset, this section tests the validity experiments of the model on the tunnel crack dataset provided by CrackSegNet (Ren et al., 2020), which contains a lot of interferences, such as paint and stains, and photos with darker colors, which further increases the difficulty of segmentation. In order to evaluate the effectiveness of the model on concrete bridge crack image data, this section uses the dataset provided by MOACA-CrackNet (Qifan Wang et al., 2023).These cracks are present in sections such as the

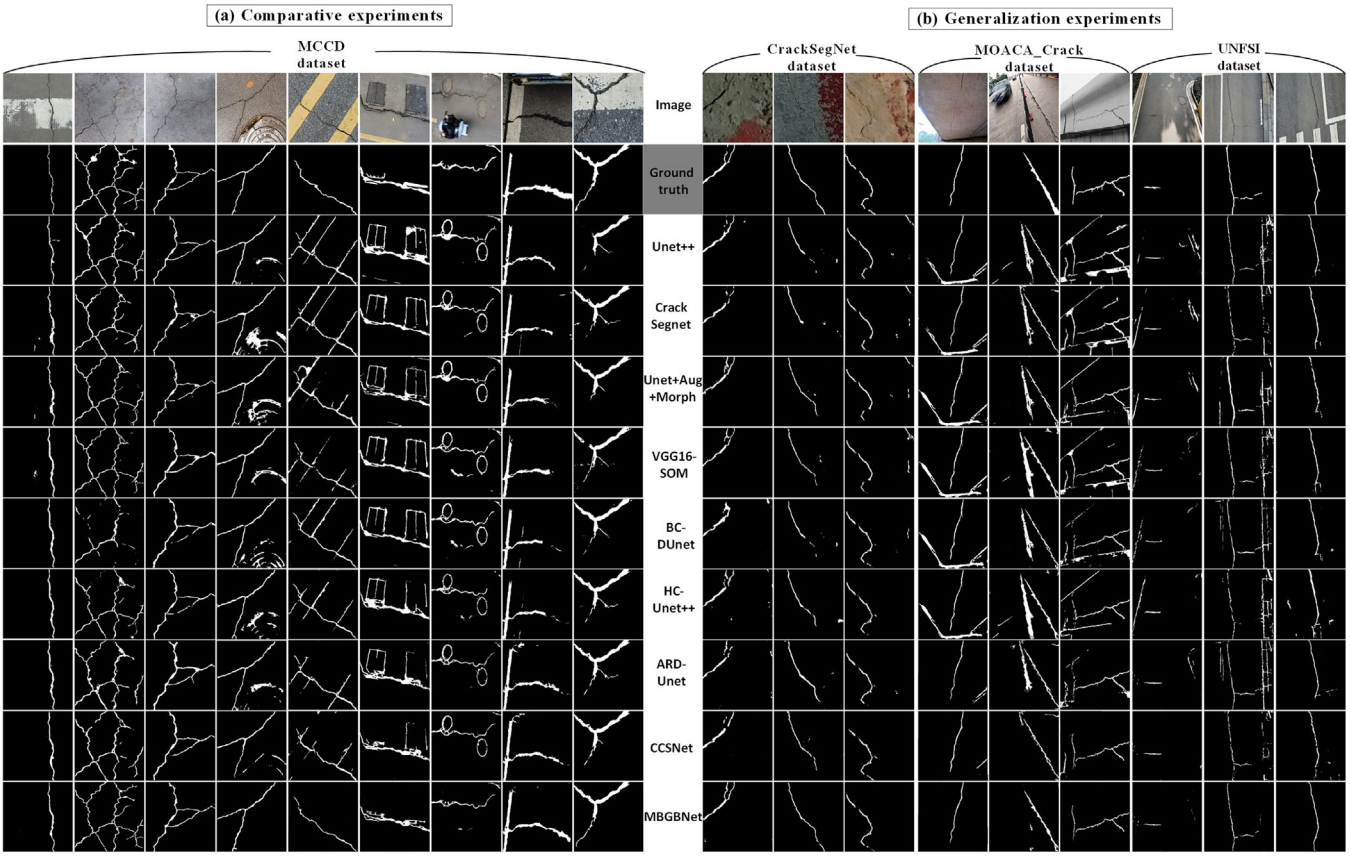

FIGURE 5 Segmentation results for each experiment: (a) denotes the segmentation results for the MCCD comparison experiment and (b) denotes the segmentation results for the generalization experiment and shows the segmentation results of each network on public data (CrackSegNet dataset—Ren et al., 2020; MOACA_Crack dataset—Qifan Wang et al., 2023; and UNFSI dataset—He et al., 2023).

TABLE 8 Comparison of the MCCD dataset.

	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
Unet++	69.12%	80.40%	73.10%	74.81%
CrackSegNet	69.17%	79.97%	72.71%	74.21%
Unet+Aug+Morph	69.87%	80.95%	73.28%	75.12%
VGG16-SOM	70.10%	81.23%	73.74%	75.09%
BC-DUnet	70.14%	81.36%	74.11%	75.08%
HC-Unet++	71.03%	82.15%	74.96%	74.10%
ARD-Unet	71.83%	82.99%	75.75%	74.72%
CCSNet	72.38%	83.11%	76.19%	74.33%
MBGBNet	**80.54**%	**86.38**%	**81.57**%	**81.34**%

Abbreviations: Dice, Dice similarity coefficient.
Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

bridge deck, bridge sides with pillars, and so forth, and due to the large span of the bridge and the direct loads from traffic, the cracks result in longer ductility and larger dimensions. In order to evaluate the performance of the model on the concrete road crack dataset, this section uses the UNSFI dataset provided by MUENet (He et al., 2023), these cracks were photographed by UAVs at higher heights, which resulted in lower segmentation accuracy due to the large field of view coverage, small crack targets, and complex image background. The details of the dataset are shown in Table 9, and the experimental results are shown in Tables 10–12.

The results from Table 10 and Figure 5b show that MBGBNet achieves good results in processing the CrackSegNet dataset containing paint stains, and the segmentation results are all better than the other SOTA networks;

TABLE 9 Details of the dataset.

Datasets	Number	Size	Type	Link
CrackSegNet	409	512 × 512	Tunnel	https://www.kaggle.com/datasets/tarkanatakkan/dataforcrack
MOACA-CrackNet	2062	1024 × 1024	Bridge	https://github.com/VesperCi/MOACA-CrackNet
MUENet	5705	640 × 640	Road	t20060599@csuft.edu.cn

TABLE 10 Comparison of the CrackSegNet dataset.

	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
Unet++	71.93%	81.38%	77.18%	76.92%
CrackSegNet	71.88%	81.32%	77.43%	77.09%
Unet+Aug+Morph	71.76%	81.40%	77.35%	77.03%
VGG16-SOM	71.80%	81.31%	78.22%	77.81%
BC-DUnet	72.25%	82.17%	78.88%	77.55%
HC-Unet++	73.95%	83.71%	79.22%	78.17%
ARD-Unet	73.53%	83.45%	79.10%	78.14%
CCSNet	74.29%	83.94%	80.10%	78.97%
MBGBNet	**78.27%**	**86.83%**	**83.25%**	**82.27%**

Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

TABLE 11 Experimental results of different methods on MOACA_Crack.

	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
Unet++	75.35%	84.54%	81.13%	78.29%
CrackSegNet	75.98%	84.76%	82.59%	78.95%
Unet+Aug+Morph	76.52%	85.79%	82.86%	79.66%
VGG16-SOM	77.37%	86.05%	83.14%	80.17%
BC-DUnet	78.03%	86.54%	83.60%	80.50%
HC-Unet++	79.46%	86.84%	84.41%	81.78%
ARD-Unet	80.24%	87.33%	85.25%	82.42%
CCSNet	80.67%	88.06%	85.57%	82.91%
MBGBNet	**83.71%**	**89.72%**	**87.35%**	**85.34%**

Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

however, due to the low light in the tunnel, the cracks are less different from the background, which makes the segmentation accuracy decrease.

For MOACA_Crack, the results from Figure 5b with Table 11 show that the large bridge span makes the crack length longer, resulting in the model being less effective in dealing with the crack ends, and for non-cracked bridge intervals, the other networks also have difficulty in segmentation, and mis-segmentation occurs, and the segmentation is made effective due to the fact that MBGBNet can detect complex backgrounds as well as small cracks very well are due to other networks. For the UNFSI dataset captured by the UAV, due to the field of view and crack size, most of the networks will misclassify the non-cracked regions (steps, crosswalk) as cracked regions, which affects the segmentation effect. Since the image size of the MCCD data used for training is 256 × 256, which is smaller than the image size used for generalization experiments, the increase in pixels leads to an increase in the amount of model segmentation task. It makes the model segmentation accuracy of the above dataset affected to some extent.

The segmentation results and the data from Table 12 prove that MBGBNet has some advancement and robustness, which indicates that it can be better applied to crack segmentation than CCSNet.

4 | REAL-WORLD ROAD TEST

In order to evaluate the actual performance of MBGBNet, we collected a large number of challenging road crack images (2400 × 1344 pixels) around the Tianxin District of Changsha City, Hunan Province, and conducted actual tests. The test results of some crack characteristics

TABLE 12 Experimental results of different methods on UNFSI.

	MIOU (%)	Precision (%)	Recall (%)	Dice (%)
Unet++	73.62%	82.12%	79.50%	80.21%
CrackSegNet	74.05%	83.16%	80.91%	80.86%
Unet+Aug+Morph	74.38%	83.21%	81.06%	81.13%
VGG16-SOM	74.73%	83.81%	81.51%	81.55%
BC-DUnet	74.82%	83.70%	81.37%	81.57%
HC-Unet++	75.38%	84.56%	82.87%	82.94%
ARD-Unet	75.96%	85.05%	83.69%	83.15%
CCSNet	76.54%	85.99%	84.72%	83.96%
MBGBNet	**80.87%**	**89.13%**	**87.61%**	**85.87%**

Bold font in the table indicates that the experimental indicator worked best in the comparison experiments with other methods.

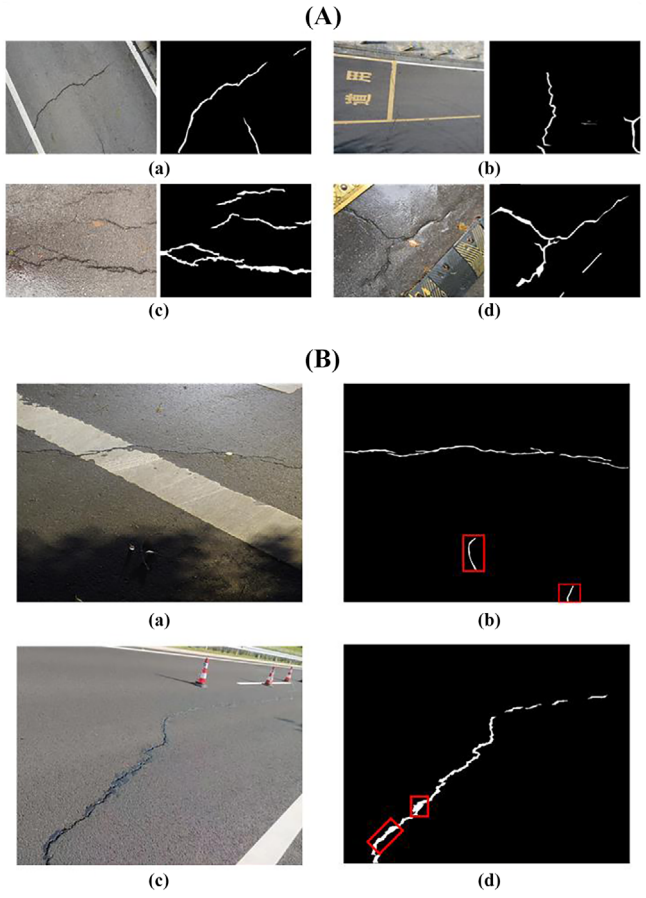

FIGURE 6 Real-world testing and modeling flaws. (a) and (b) in (A) indicate road cracks with lane line interference photographed from a high location. (c) and (d) represent road cracks with reflections caused by water in rainy conditions. (a) in (B) indicates road cracks under nighttime conditions, and (c) indicates road cracks under asphalt conditions.

obtained are shown in Figure 6A, where (a) and (b) show the opposite and same direction lane lines with cracks taken from a height, the lane lines in the background are shown in white and yellow with font interference; (c) shows an image of a cracked roadway taken in a rainy-day environment with complex crack shapes and reflections of the roadway caused by water accumulation in the rainy day; (d) shows a crack image taken under rainy conditions with complex interference from foliage and speed bumps. From the segmentation results, although there are few false-positive detection results, MBGBNet accurately segments cracks in complex scenes with good adaptability and generalization ability, which can be effectively used for segmentation tasks in different scenes.

However, MBGBNet still has some drawbacks in some environments. The segmentation results of MBGBNet on nighttime roads as well as real tests on asphalt roads are shown in Figure 6B. Under nighttime lighting conditions, the difference between the cracks and their surroundings in terms of color and brightness is not obvious due to the light conditions, which poses a challenge for accurate segmentation of the cracks; (b) from Figure 6B shows that MBGBNet misidentifies non-crack objects such as tree branches as cracks, leading to biased detection results. The asphalt repair construction is simple and can be completed quickly, but there will be asphalt residue around the repaired cracks, as shown in (d), which leads to errors in MBGBNet when segmenting the subsequently generated cracks, and the segmentation range is larger than the cracks. In our future work, we will explore the performance of the model on a variety of pavement materials, such as asphalt and masonry, and adjust the crack detection algorithm for darker environments such as nighttime, optimize the crack detection strategy, and further improve the robustness and accuracy of the model.

5 | CONCLUSION

As a special edge detection task, the core point of crack segmentation is to accurately identify and localize the crack features in the image, which is of key significance for the accuracy and efficiency of road crack detection. Therefore, a road crack segmentation algorithm based

on MBGBNet is proposed in this paper. The algorithm solves the traditional feature loss problem by introducing a Multi-scale Domain Feature Aggregation (MDFA) module, which uses convolutional kernels of different sizes to extract the information and combines with an adaptive asymmetric module to effectively suppress the interference of the background information, so as to present the basic morphological features of the cracks in an excellent way.

Subsequently, the BEFA is applied to further deepen the extraction effect of the crack features by fusing the horizontal and vertical weights and focusing on highlighting the information of the tiny cracks. In addition, this paper also adopts the GWEA module to smooth the crack image, and this operation can effectively identify cracks with irregular edges, thus realizing the accurate segmentation of crack edges. Finally, combined with the PBOA, the optimal solution is searched in advance during the preheating phase to accurately determine the optimal learning rate for model training, which significantly improves the training effect of the model.

In the comparison experiments with seven advanced crack segmentation models, the test results based on the road crack dataset MCCD show that MBGBNet exhibits better performance. Its MIOU, precision, recall, and Dice coefficients reach 80.54% (11.42 percentage points), 86.38% (5.98 percentage points), 81.57% (8.14 percentage points), and 81.34 (6.53 percentage points), respectively, and it successfully realizes more accurate and detailed crack segmentation. It successfully realizes a more accurate and tiny crack segmentation effect. In addition, MBGBNet also outperforms other methods in the public crack datasets of three different scenarios. Finally, MBGBNet demonstrates higher detection accuracy and better performance through complex road tests, which further confirms its excellent performance in road crack detection and strong anti-interference ability in complex environments, and provides strong support and a new direction for the development of road crack detection technology.

In the comparison and generalization experiments, it can be found that the model shows high accuracy in the task of segmenting crack images with high brightness but not in segmenting crack images in dark areas. In particular, the model incorrectly recognizes tree branches and leaves as cracks during nighttime road photography. In addition, for roads with cracks regenerated after asphalt patching, the model misidentified the patched part as a crack, and the segmentation width was larger than the crack. To solve the above problems, future research will focus on improving the model's ability to recognize crack features in darkness, roads made of other materials, such as asphalt, and enhancing the model's performance in distinguishing the difference between cracks and the background.

MBGBNet can effectively handle the road crack segmentation problem in complex environments with tiny cracks and irregular edges, and the optimization algorithm can make the model converge quickly. In the future, the model can be integrated into devices such as vehicles or drones to realize a wider range of intelligent road maintenance, which is of great significance for improving road safety and maintenance efficiency.

ACKNOWLEDGMENTS

We are grateful to Liujun Li from the University of Idaho for graciously providing assistance in refining this manuscript for enhanced precision in expression.

REFERENCES

Bavelos, A. C., Anastasiou, E., Dimitropoulos, N., Oikonomou, G., & Makris, S. (2024). Augmented reality-based method for road maintenance operators in human–robot collaborative interventions. *Computer-Aided Civil and Infrastructure Engineering*, *39*(7), 1077–1095.

Bilmes, J. A. (1998). A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models (Technical Report No. ICSI-TR-97-021). International Computer Science Institute.

Cai, W., Ning, X., Zhou, G., Bai, X., Jiang, Y., Li, W., & Qian, P. (2022). A novel hyperspectral image classification model using bole convolution with three-direction attention mechanism: Small sample and unbalanced learning. *IEEE Transactions on Geoscience and Remote Sensing*, *61*, 1–17.

Cao, H., Gao, Y., Cai, W., Xu, Z., & Li, L. (2023). Segmentation detection method for complex road cracks collected by UAV Based on HC-Unet++. *Drones*, *7*(3), 189.

Çelik, F., & König, M. (2022). A sigmoid-optimized encoder–decoder network for crack segmentation with copy-edit-paste transfer learning. *Computer-Aided Civil and Infrastructure Engineering*, *37*(14), 1875–1890.

Chen, J., & He, Y. (2022). A novel U-shaped encoder–decoder network with attention mechanism for detection and evaluation of road cracks at pixel level. *Computer-Aided Civil and Infrastructure Engineering*, *37*(13), 1721–1736.

Chu, H., Wang, W., & Deng, L. (2022). Tiny-Crack-Net: A multiscale feature fusion network with attention mechanisms for segmentation of tiny cracks. *Computer-Aided Civil and Infrastructure Engineering*, *37*(14), 1914–1931.

Cui, X., Wang, Q., Dai, J., Xue, Y., & Duan, Y. (2021). Intelligent crack detection based on attention mechanism in convolution neural network. *Advances in Structural Engineering*, *24*(9), 1859–1868.

Diederik, P. K. (2014). Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR)*, Banff, AB, Canada (pp. 1–5).

Ehteram, M., Binti Othman, F., Mundher Yaseen, Z., Abdulmohsin Afan, H., Falah Allawi, M., Bt Abdul Malek, M., Najah Ahmed, A., Shahid, S. P., Singh, V., & El-Shafie, A. (2018). Improving the Muskingum flood routing method using a hybrid of particle swarm optimization and bat algorithm. *Water*, *10*(6), 807.

Gao, Y., Cao, H., Cai, W., & Zhou, G. (2023). Pixel-level road crack detection in UAV remote sensing images based on ARD-Unet. *Measurement*, *219*, 113252.

Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2020). Fundamental concepts of convolutional neural network. In V

Balas, R. Kumar, & R. Srivastava (Eds.), *Recent Trends and Advances in Artificial Intelligence and Internet of Things* (pp. 519–567). Springer.

Han, H., Deng, H., Dong, Q., Gu, X., Zhang, T., & Wang, Y. (2021). An advanced Otsu method integrated with edge detection and decision tree for crack detection in highway transportation infrastructure. *Advances in Materials Science and Engineering*, *2021*(1), 9205509.

He, X., Tang, Z., Deng, Y., Zhou, G., Wang, Y., & Li, L. (2023). UAV-based road crack object-detection algorithm. *Automation in Construction*, *154*, 105014.

Hou, Q., Zhou, D., & Feng, J. (2021). Coordinate attention for efficient mobile network design. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN (pp. 13713–13722).

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT (pp. 7132–7141).

Lau, S. L., Chong, E. K., Yang, X., & Wang, X. (2020). Automated pavement crack segmentation using U-Net-based convolutional neural network. *IEEE Access*, *8*, 114892–114899.

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., & Han, J. (2019). *On the variance of the adaptive learning rate and beyond*. arXiv preprint arXiv:1908.03265. https://doi.org/10.48550/arXiv.1908.03265

Liu, T., Zhang, L., Zhou, G., Cai, W., Cai, C., & Li, L. (2022). BC-DUnet-based segmentation of fine cracks in bridges under a complex background. *PLoS ONE*, *17*(3), e0265258.

Liu, Z., Yang, Z., Ren, T., Wang, Z., Deng, J., Deng, C., Zhao, H., Zhou, G., Chen, A., & Li, L. (2025). A hierarchical progressive recognition network for building change detection in high-resolution remote sensing images. *Computer-Aided Civil and Infrastructure Engineering*, *40*(2), 243–262.

Marini, F., & Walczak, B. (2015). Particle swarm optimization (PSO). A tutorial, *Chemometrics and Intelligent Laboratory Systems*, *149*, 153–165.

Qiao, W., Liu, Q., Wu, X., Ma, B., & Li, G. (2021). Automatic pixel-level pavement crack recognition using a deep feature aggregation segmentation network with a scSE attention mechanism module. *Sensors*, *21*(9), 2902.

Ren, Y., Huang, J., Hong, Z., Lu, W., Yin, J., Zou, L., & Shen, X. (2020). Image-based concrete crack detection in tunnels using deep fully convolutional networks. *Construction and Building Materials*, *234*, 117367.

Siriborvornratanakul, T. (2022). Downstream semantic segmentation model for low-level surface crack detection. *Advances in Multimedia*, *2022*(1), 3712289.

Sun, L., Yang, Y., Zhou, G., Chen, A., Zhang, Y., Cai, W., & Li, L. (2024). An integration–competition network for bridge crack segmentation under complex scenes. *Computer-Aided Civil and Infrastructure Engineering*, *39*(4), 617–634.

Tang, S., Zhu, Y., & Yuan, S. (2021). An improved convolutional neural network with an adaptable learning rate towards multi-signal fault diagnosis of hydraulic piston pump. *Advanced Engineering Informatics*, *50*, 101406.

Van Hauwermeiren, W., Filipan, K., Botteldooren, D., & De Coensel, B. (2022). A scalable, self-supervised calibration and confounder removal model for opportunistic monitoring of road degradation. *Computer-Aided Civil and Infrastructure Engineering*, *37*(13), 1703–1720.

Wang, Q., Chen, A., Cai, W., Cai, C., Fang, S., Li, L., Wang, Y., & Zhou, G. (2023). Segmentation network of concrete cracks with multi-frequency octaveres dual encoder and cross-attention mechanism optimized by average weight. *Automation in Construction*, *155*, 105050.

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., & Hu, Q. (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA (pp. 11534–11542).

Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Lecture notes in computer science: Vol. 11211. Proceedings of the European conference on computer vision (ECCV)* (pp. 3–19). Springer.

Xiang, C., Guo, J., Cao, R., & Deng, L. (2023). A crack-segmentation algorithm fusing transformers and convolutional neural networks for complex detection scenarios. *Automation in Construction*, *152*, 104894.

Yamaguchi, T., & Mizutani, T. (2024a). Road crack detection interpreting background images by convolutional neural networks and a self-organizing map. *Computer-Aided Civil and Infrastructure Engineering*, *39*(11), 1616–1640.

Yamaguchi, T., & Mizutani, T. (2024b). Quantitative road crack evaluation by a U-Net architecture using smartphone images and Lidar data. *Computer-Aided Civil and Infrastructure Engineering*, *39*(7), 963–982.

Zheng, Y., Gao, Y., Lu, S., & Mosalam, K. M. (2022). Multistage semisupervised active learning framework for crack identification, segmentation, and measurement of bridges. *Computer-Aided Civil and Infrastructure Engineering*, *37*(9), 1089–1108.

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2019). UNet++: Redesigning skip connections to exploit multiscale features in image segmentation. *IEEE Transactions on Medical Imaging*, *39*(6), 1856–1867.

How to cite this article: Wang, G., He, M. F., Liu, G., Li, L., Liu, E., & Zhou, G. (2025). An optimized and precise road crack segmentation network in complex scenarios. *Computer-Aided Civil and Infrastructure Engineering*, 1–20. https://doi.org/10.1111/mice.13444