

Research Article

Pixel-Level Crack Detection and Quantification of Nuclear Containment with Deep Learning

Jian Yu ¹, Yaming Xu,¹ Cheng Xing ¹, Jianguo Zhou,² and Pai Pan¹

¹School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China

²School of Civil Engineering, Architecture and Environment, Hubei University of Technology, Wuhan 430068, China

Correspondence should be addressed to Cheng Xing; chxing@sgg.whu.edu.cn

Received 3 February 2023; Revised 16 June 2023; Accepted 26 June 2023; Published 10 August 2023

Academic Editor: Fabio Casciati

Copyright © 2023 Jian Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Crack detection based on deep learning is an advanced technology, and many scholars have proposed many methods for the segmentation of pavement cracks. However, due to the difference of image specifications and crack characteristics, some existing methods are not effective in detecting cracks of containment. To quickly detect cracks and accurately extract crack quantitative information, this paper proposes a crack detection model, called MA_CrackNet, based on deep learning and a crack quantitative analysis algorithm. MA_CrackNet is an end-to-end model based on multiscale fusions that achieve pixel-level segmentation of cracks. Experimental results show that the proposed MA_CrackNet has excellent performance in the crack detection task of nuclear containment, achieving a precision, recall, F1, and mean intersection-over-union (mIoU) of 86.07%, 89.96%, 87.97%, and 89.19%, respectively, outperforming other advanced semantic segmentation models. The quantification algorithm automatically measures the four characteristic indicators of the crack, namely, the length of the crack, the area, the maximum width, and the mean width and obtains reliable results.

1. Introduction

Nuclear containment is the last barrier to prevent the radioactive material from spilling out of the reactor in the nuclear power plant. Regularly checking and maintaining nuclear containment is critical in ensuring the safe operation of nuclear power plants. As a manifestation of joint ageing damage of nuclear containment, crack is an essential index in inspecting nuclear containment appearance.

Early crack detection of nuclear containment mainly adopted the artificial visual detection method, which was time-consuming, high-risk, and low-efficiency. To improve the detection efficiency, digital image acquisition technology has been applied to the defect detection of structures, among which the more successful applications are the remote defect detection system Scanite developed by the French company SITES [1] and the containment external defect inspecting system (CEDIS) developed by Wuhan University. Although Scanite realizes remote acquisition, it still needs to manually collect images one by one, while CEDIS realizes route

planning and automatic image acquisition. The CEDIS system is in use at several nuclear power plants. The remote acquisition of the images of the nuclear containment surface has been realized, but the automatic processing of the observation images has not been realized.

With the development of science and technology updates, digital image processing technology has been widely used in life and work. Compared with the manual labeling method, this method has the characteristics of good reproducibility, high precision, and high flexibility. Researchers put forward different algorithms according to different application scenarios, and the current plans are mainly divided into threshold segmentation algorithms [2, 3], edge detectors [4–6], and morphological algorithms [7]. Detection technology based on digital image processing has the advantages of fast detection speed and timely detection, gradually replacing the manual detection method. However, crack detection using traditional digital image processing methods is very dependent on the experience of algorithm designers, and the detection results are

significantly affected by noise caused by factors such as illumination, angle, and deformation, resulting in low detection accuracy and poor robustness [8].

In recent years, deep learning has developed rapidly in computer vision. Computer vision and deep learning technologies are increasingly being used to inspect, monitor, and evaluate infrastructure, including crack detection. Crack detection methods based on deep learning can be roughly divided into image classification, target detection, and semantic segmentation.

Image classification refers to the deep learning method that allows the computer to analyse and process the image and determine the image category from the given classification set. For example, Cha et al. [9] combined the CNN with sliding window technology to classify images with cracks and images without cracks, calibrate the cracks in the images, and detect road cracks. Leo Pauly and Luo [10] proposed a deeper network model and proved through experiments that a deeper network could improve the detection performance of pavement cracks. Yu et al. [11] proposed an improved Dempster–Shafer (D-S) algorithm to fuse the results of multiple CNN models, which overcome the noise interference and improve the crack detection accuracy and robustness of the model. Wu et al. [12] presented a crack detection technology based on GoogLeNet Inception V3 and used transfer learning to improve the performance of the networks for crack detection. The image classification method can more accurately determine whether a crack exists in the image but cannot precisely locate the crack position.

The difference between target detection and image classification is that the former detects and locates specific multiple targets from the image. Hacıfendioğlu and Başağa [13] proposed a detection method for pavement perfusion cracks based on improved faster R-CNNs. Cha et al. [14] applied a network based on the faster R-CNN which can provide quasi real-time, autonomous vision-based structural damage detection, realizing the detection of five defects with an 89.7% mAP. Li et al. [15, 16] also applied YOLOv3 to crack detection on the dam surface and steel structure surface, achieving a good detection effect.

Segmentation is similar to image classification, except that the content of classification is each pixel instead of the whole image. Yang et al. [17] proposed an FCN to identify and segment cracks of different sizes through multiple types of cracks, but the FCN needs a lot of data for training. Recently, Liu et al. [18] applied U-Net to crack detection tasks for the first time and found that it can achieve higher accuracy using a smaller training set than the FCN. To improve the operation efficiency and reduce the interference, Chen et al. [19] proposed a two-step deep learning method composed of CNN classification and U-Net segmentation modules, which were used to automatically detect the facade cracks from the images captured by drones. Convolutional tends to ignore the relation between pixels when processing local neighborhood information and it is difficult to capture the long-range and global dependencies. Detection results can be influenced by highly complex scenes. To address this issue,

researchers have integrated the attention mechanism into neural networks for defect detection, as seen in studies on generating training data and concrete crack segmentation [20–22].

Ji et al. [23] proposed an integrated approach to automatically detect and quantify cracks in asphalt pavement at the pixel level. Kang et al. [24] proposed an improved DTM algorithm to eliminate the centerline loss in the crack thinning process and calculate more accurate crack length. However, only the maximum distance from the edge to the center line in each row and column is compared to calculate the width of the crack, and the directional property of the crack is not fully considered. Zhao et al. proposed an intelligent solution combining YOLOv5 detection and Crack-FPN segmentation to improve detection efficiency and accuracy [25]. Compared with image classification and object detection, the semantic segmentation method classifies all the pixels in the image and achieves pixel-level segmentation of the target object. Cracks' distribution, shape, and size are described effectively and accurately. Further extracting quantifiable information at the pixel level, such as crack length, maximum width, area, and ratio, is possible.

In summary, although some scholars have studied crack detection and segmentation technology based on deep learning, it has mainly been applied to road surfaces. The crack detection and segmentation on the nuclear containment surface have not been thoroughly discussed. Unlike roads, nuclear containment images are more difficult to collect, and the backgrounds are even more complex. Therefore, according to the crack characteristics of containment, this paper proposes a deep learning-based method for containment crack detection.

This work will focus on nuclear containment crack segmentation using deep convolutional neural networks. Deep learning methods require data for training. We obtained the image of the nuclear containment through our self-developed remote appearance acquisition system. We manually annotated it to create a dataset for nuclear containment crack segmentation. To more efficiently and accurately segment nuclear containment cracks, this paper proposes a novel U-shaped encoder and decoder neural network. A novel multiscale feature fusion module is proposed to obtain information at different scales. ASPP module and attention mechanism are introduced to improve the detection effect of microcracks. The mixed loss function of the binary cross-entropy (BCE) and Dice coefficient loss function is used to effectively alleviate the effect of class imbalance of datasets' class imbalance.

The content of this paper is organised as follows: Section 2 introduces the construction of the nuclear containment crack dataset. Section 3 presents the architecture of the network model proposed in this paper, MA_CrackNet, and describes the postprocessing and crack quantification processes. Section 3 introduces the training process of MA_CrackNet including training implementation details, loss function, and performance evaluation index. Section 4 introduces the experimental results. Section 5 concludes the paper and puts forward the direction of the future work.

2. Nuclear Containment Crack Dataset

2.1. Image Acquisition. The external image of the nuclear containment was collected by the remote data acquisition system independently developed by the laboratory of Wuhan University. Figure 1 displays the acquisition equipment and software interface. A control network was laid on the ground near the nuclear containment, and the shooting scheme was planned according to the terrain. The acquisition device composed of a self-developed SLR camera and pin-top equipment can quickly and automatically collect high-resolution images of the appearance at a long distance, guaranteeing the coverage of most of the surface area of the nuclear containment.

2.2. Image Mosaic and Crack Labeling. After obtaining the original acquisition image, according to the coordinates of the collection point and the internal and external orientation elements, the tilted image is corrected, splinted, resampled, and clipped using the photogrammetry principle to obtain the forward photographic image with uniform size and resolution, forming the panoramic image of the outer appearance of the nuclear containment. Image correction is a differential correction of the acquired oblique image to obtain the orthographic image of the nuclear containment; the process is shown in Figure 2. The size of the image after resampling is 4593×3675 , and the actual size of the corresponding range of each image is $750 \text{ mm} \times 600 \text{ mm}$. Figure 3 shows a panoramic view of part of the containment appearance.

In this experiment, image data of multiple nuclear containments from different periods were used, and the images containing cracks were selected. Sliding window technology was then used to obtain the images of the required specifications for network model training from the resampling images with a window size of 512×512 , and 450 images with cracks were selected for annotation. The annotation tool was used to highlight all the cracks in the image in green. Then, the threshold method based on colour is applied to Figure 4(c) to generate binary images, as shown in Figure 4(d), where the pixel value of the crack area was 255 and the pixel value of the background area was assigned to 0.

2.3. Data Augmentation. Owing to the small number of cracks on the containment surface, only 450 original images and 450 corresponding mask images were obtained after these operations. A total of 50 images were randomly picked as the testing set to assess the model's robustness. The rest were used as training images. To overcome the problem of insufficient labeled data, data augmentation was applied, which can effectively avoid overfitting and meanwhile increase the generalization capability and robustness of the model [26]. The following augmentations were applied in our approach: (1) flipping the image horizontally and vertically, (2) rotation of 90° , 180° , and 270° and random rotation in the range (-60° and 60°), (3) scaling images by the ratios of 0.8 and 1.2, and (4) colour transformations: by changing the brightness, contrast, and saturation of the

image, the scene with different illumination is simulated. After augmentation, the number of training images increased from 400 to 7000. These images were further divided into a training set (6,300 images) and a validation set (700 images), used to fit the model parameters, adjust the hyperparameters of the model, and make a preliminary assessment of the model's crack identification ability.

2.4. Nuclear Containment Crack Characteristics. Some of the cracks in the image are very small, only one or two pixels wide. Owing to the high level of safety, the concrete grade of the nuclear containment is different from that of ordinary concrete structures. Some differences arise in the behaviour of crack forms. The statistical characteristics of nearly 4,000 images collected from a containment indicate that these submillimetre cracks appear to be much smaller than cracks in sidewalks or other concrete structures [27]. In addition, the cracks are mainly long and narrow vertical cracks. Even on the same crack, the crack width varies greatly. Table 1 shows information on containment crack dataset and three other crack datasets (CFD, DeepCrack, and Crack500). The crack pixel ratio is defined as the ratio of crack pixels to total pixels in the image. It can be seen that compared with other datasets, the issue of class imbalance is more serious in the nuclear containment crack dataset. In addition, we also produce statistics on the width of crack for each dataset. In Figure 5, 320×320 sized images were cropped from each dataset for comparison. As can be seen from the figure, the bitumen material of the Crack500 is significantly different from that of other datasets. In the nuclear containment data, the pixel width of the cracks in the image is finer than that of other datasets.

3. Methods

The primary process of nuclear containment crack segmentation is shown in Figure 6, including four main steps. (1) High-resolution images of the appearance are collected using the automatic acquisition equipment for the appearance of the nuclear containment. The collected images are then corrected and stitched. Then, the photos with cracks are cropped from the figure and manually annotated to obtain the nuclear containment crack segmentation dataset. (2) A deep learning model is proposed to segment the nuclear containment cracks. (3) The predicted image is postprocessed to connect the fractured cracks in the image. A noise removal method that removes noise independently from the prediction graph is built. (4) Finally, the skeletonisation method is used to analyse the length of the crack and the maximum width and area of the crack.

3.1. Deep Learning-Based Crack Segmentation. Although the U-Net model has good performance in the segmentation task of medical images, it will continuously reduce the image size during the downsampling calculation process. The cracks on the nuclear containment have complex shapes and different sizes, resulting in the crack segmentation task. U-Net's detection effect on small-scale cracks is poor. To solve

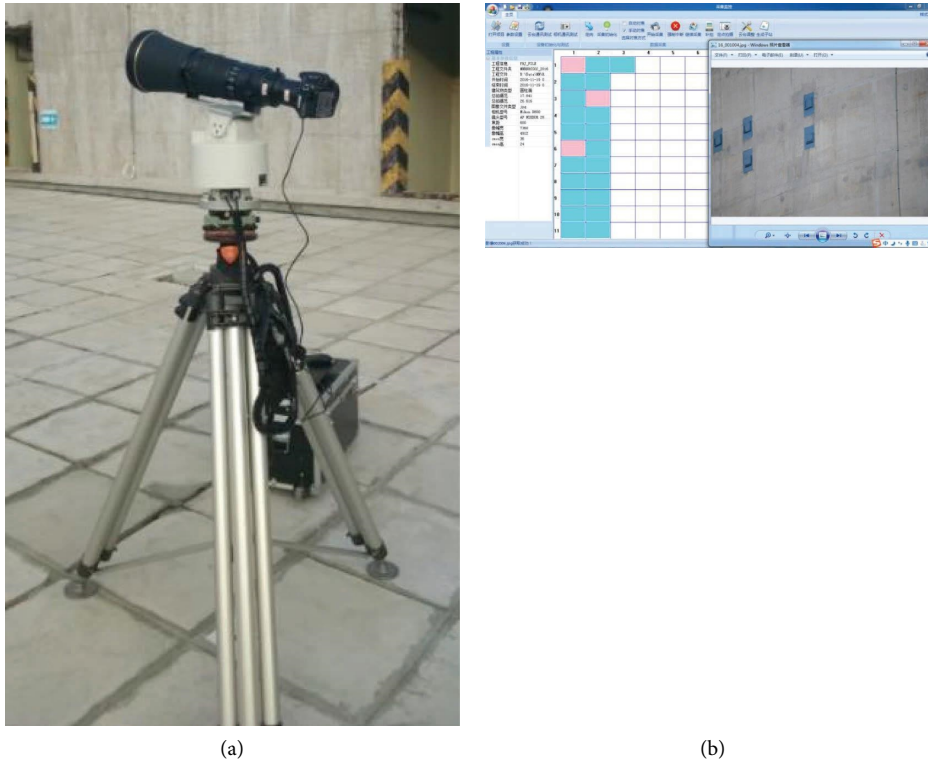


FIGURE 1: Nuclear containment appearance image acquisition. (a) Structure of the image acquisition equipment. (b) Software interface of the image processing host computer.

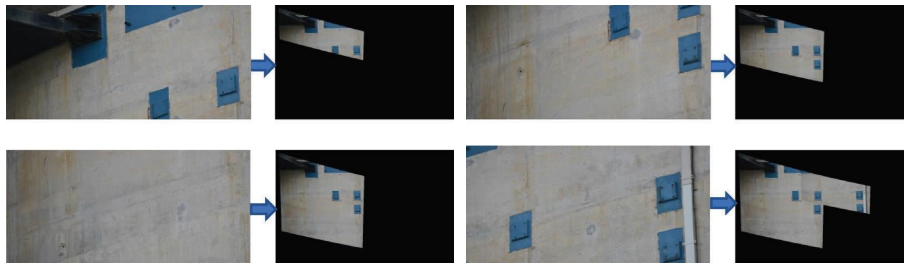


FIGURE 2: Image correction and mosaic.

this problem, the paper proposes a novel U-shaped encoder and decoder neural network that more efficiently and accurately distinguishes between cracked and noncracked pixels, providing a more accurate basis for subsequent quantitative assessment of the damage.

Compared with the classic U-Net, the proposed network is distinct in three aspects. First, to eliminate the influence of class imbalance, the attention module is introduced. Second, an ASPP module is added to the tail of the encoder. Finally, to make full use of different levels of semantic and geometric information, we construct a multiscale fusion structure.

The structure of the improved network is shown in Figure 7, which is divided into two parts: encoder and decoder. The encoder mainly comprises the convolution module, pooling layer, and ASPP module. The structure of the convolution module is shown in Figure 8. Specifically,

the convolution module mainly comprises the convolution layer, batch normalisation operation, activation function, and DropBlock. Given that cracks account for a small proportion of the image, the convolutions are all 3×3 convolutions (Conv) with stride 1. The activation layer uses the rectified linear unit (ReLU) function, which is used to improve the nonlinear expression ability of the model. A batch normalisation (BN) layer is added after the convolution, and a DropBlock is added at the end of the convolution module to prevent overfitting. A max-pooling layer is added between convolution modules to operate on the feature maps for dimensionality reduction. The ASPP module is added at the end of the coding part, and the hole rates used in the hole convolution are 6, 12, and 18. ASPP obtains three different receptive fields through parallel atrous convolution, which is beneficial to segment objects of different scales.

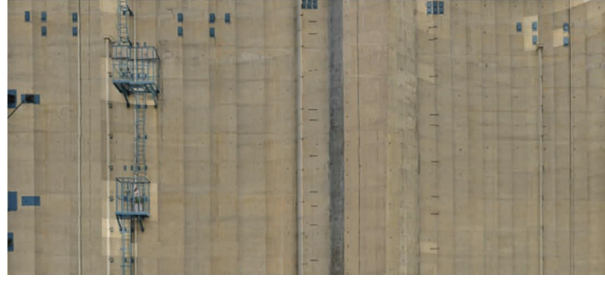


FIGURE 3: Partial panoramic view of nuclear containment appearance.

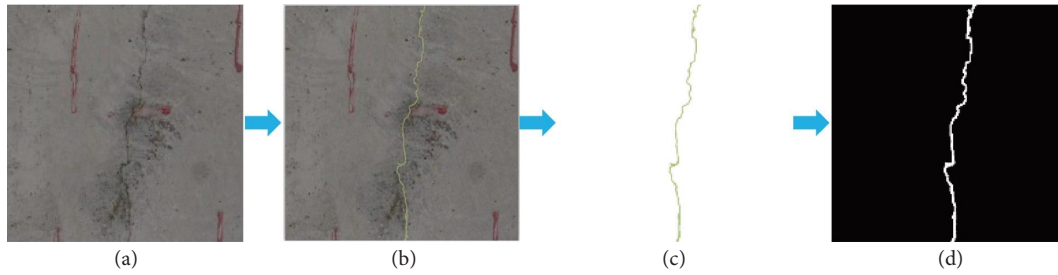


FIGURE 4: Dataset annotation.

TABLE 1: Information of different datasets.

Dataset	Scenes	Size	Crack pixel ratio (%)			Crack width (pixel)	
			Average	Max	Min	Range	Average
CFD	Pavements	480 * 320	1.66	6.33	0.35	1–15	4.75
DeepCrack	Pavements and buildings	544 * 384	2.91	19.94	0.26	1–180	8.10
Crack500	Road	2560 * 1440	3.03	9.46	0.22	1–128	25.08
Nuclear containment	Nuclear containment	512 * 512	0.65	1.67	0.21	1–13	3.38

The bold values indicate the extreme values.

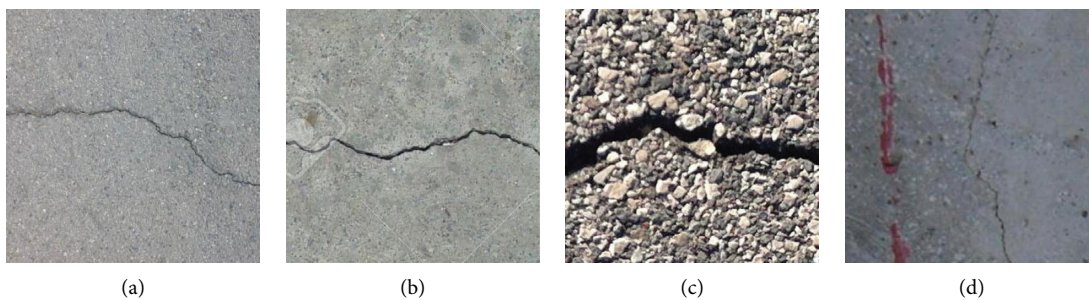


FIGURE 5: Example images of different crack datasets: (a) CFD, (b) DeepCrack, (c) Crack500, and (d) nuclear containment.

The decoder needs to reduce the high-dimensional feature map obtained by the encoder to the same size as the input image. It mainly consists of an upsampling module and an attention module. The upsampling module contains an upsampling operation and the same convolution module as the encoding process. The upsampling operation can effectively expand the size of the high-dimensional feature maps. The attention mechanism, incorporating contextual information, highlights regions that require significant attention while suppressing the activation of features on irrelevant regions to improve model sensitivity and accuracy.

At the end of the decoder part, the feature maps of each decoder layer are fused by making full use of the richer spatial features of the low-level feature maps. At the end of the network, 1×1 convolution and sigmoid function are utilised to map the 64-channel features to two classes (“crack” and “noncrack”) pixel-level segmentation maps.

3.1.1. Convolutional and Pooling Layers. The convolutional layer is the basic structure of the convolutional neural network, which processes the input information through

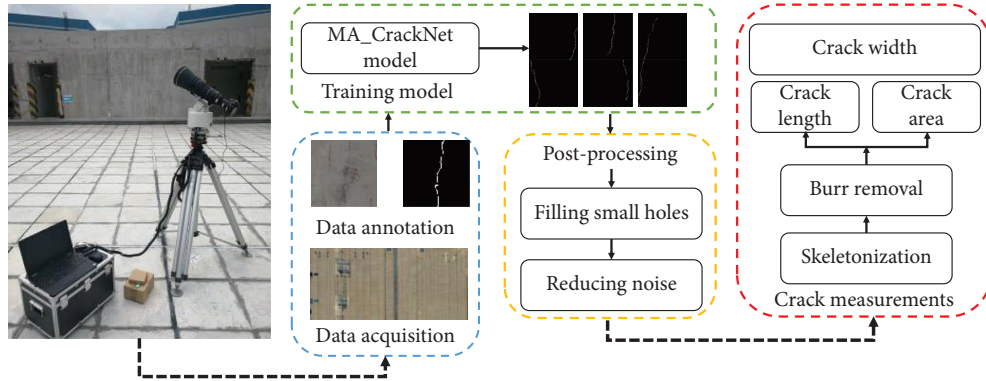


FIGURE 6: Main flow of nuclear containment crack segmentation.

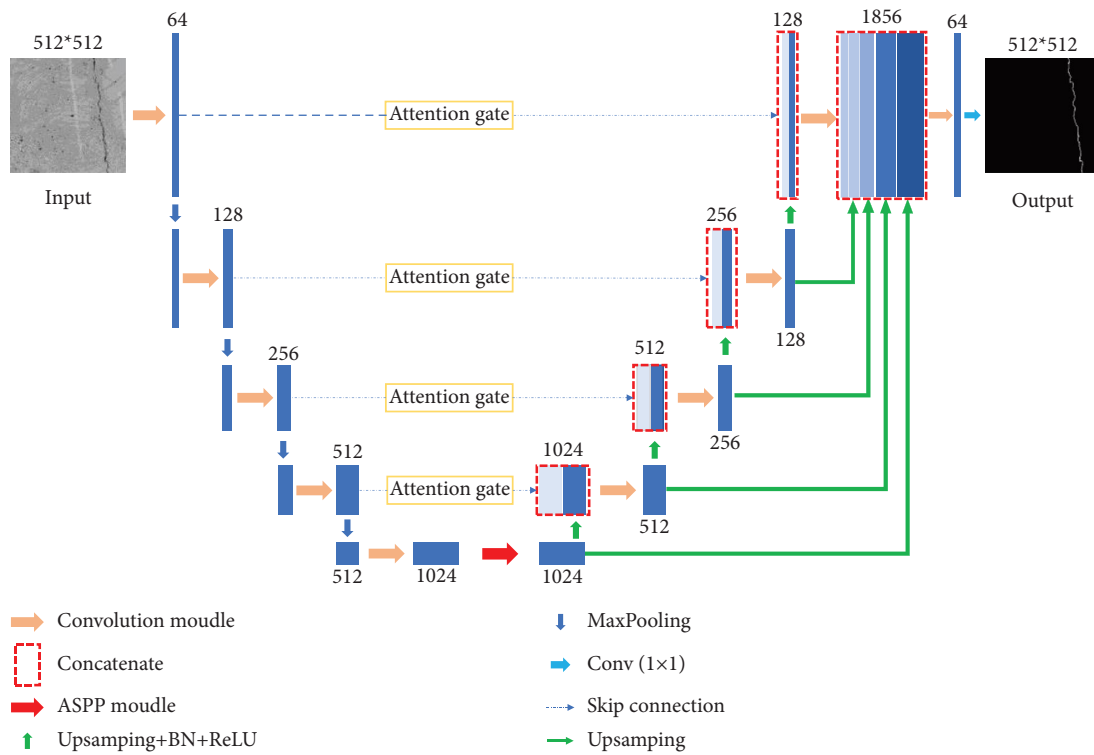


FIGURE 7: MA_CrackNet network structure.

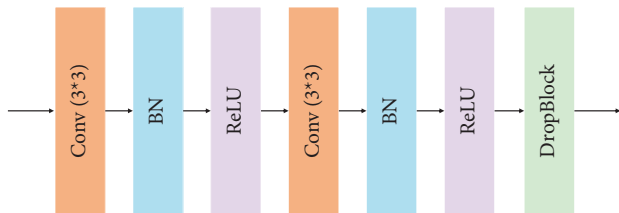


FIGURE 8: Convolution module.

convolution to complete the feature extraction. It has two feature attributes, namely, local connection and weight sharing, which can effectively reduce the amount of data, speed up the training, and reduce the overfitting phenomenon in the training process [28].

The features of the image can be extracted by convolution calculation, but a large number of parameters will be generated in this process, and the problem of overfitting is easily caused. Therefore, a pooling layer is generally added between the convolutional layers, and the features obtained by the convolution calculation are calculated. The data matrix of the graph is compressed to reduce the parameters passed, reduce the calculation amount of the model, and ensure the translation invariance of the model [29].

3.1.2. DropBlock. DropBlock is a regularisation method to alleviate overfitting. From Figure 9, we can intuitively see that unlike dropout, which randomly deletes the information of some points, DropBlock deletes the entire local

area. Given that the convolutional layer is generally connected to the pooling layer and the activation layer, it is not sensitive to the random discarding of dropout. DropBlock deletes the local area as a whole, which can effectively solve the problem of semantic information redundancy and, to a certain extent, make the model more accurate. Hyperparameter selection is more robust [30].

3.1.3. Attention Gate (AG) Module. Owing to the small proportion of cracks in the image, part of the crack features may be lost after multilayer coding. At the same time, the complex background is easy to cause interference. The attention mechanism is introduced to connect the encoder layer and decoder layer, combine the context information, suppress the influence of background, and strengthen the learning of crack features. The role of the AG module is to inhibit the learning of irrelevant features while strengthening the learning of task-related features during training. Its structure is shown in Figure 10, where σ_1 represents Relu, σ_2 represents Sigmoid, g represents the output of decoder, x^l represents the output of encoder, Conv represents convolutional operations, BN represents batch normalisation layer, and \hat{x}^l represents output [31].

3.1.4. ASPP Module. Containment cracks are small and difficult to identify. The ASPP module is introduced to improve the identification effect of the model on small targets [32]. Its structure is shown in Figure 11. ASPP consists of (a) one 1×1 convolution, three 3×3 dilated convolutions with rates $=(6, 12, \text{ and } 18)$, and one image pooling. Image pooling consists of one pooling layer, one 1×1 convolution, and one unsample layer. The results of the four convolutions are concatenated with the results of the image pooling and provided to another 1×1 convolution. The ASPP module is added at the end of the encoder, which can optimize the extraction of local and global information and enhance the ability of the image to obtain contextual information and multiscale information without increasing the number of parameters [33].

3.1.5. Multiscale Feature Fusion. Accurate extraction of features for multiscale cracks is difficult due to the variety of lengths and widths of containment cracks. In this paper, this challenge is effectively dealt with by multiscale feature fusion. Both U-Net and FPN use a U-shaped feature fusion method with the structure shown in Figure 12(a), but as the high-level features are fused with the shallow features, their rich semantic information is also diluted layer by layer. Therefore, this paper constructs a multiscale fusion structure as shown in Figure 12(b). Not only the features of the deep layer are fused with the features of the shallow layer but also the features of the bottom three layers are directly introduced into the final feature of the shallowest layer for fusion. The semantic and geometric information of the different layers are fully utilised.

3.2. Postprocessing. When using MA_CrackNet to obtain the crack segmentation image, some holes and noise are inevitable. Therefore, the predicted crack image needs to be processed before quantitative analysis.

3.2.1. Hole Filling and Broken Connections. In this paper, the closing operation is used to eliminate the holes in the graph. The operation of dilating and then corroding the image is called the closing operation. The dilation operation can fill small holes and fill up the depressions around the image, and the erosion operation can eliminate the noise in the image. The two seem to be opposite, but they are not mutually inverse operations, and changing the order of operations will have different effects. The result of performing the dilation operation first and then the erosion operation is to connect the nonconnected bright regions in the graph and keep their size basically unchanged.

3.2.2. Noise Cancellation. We use morphological features and the area of connected regions to remove some small areas of isolated noise. By calculating the area A_i of each isolated connected area and then comparing it with the set threshold L , if it is less than the set threshold, it will be judged as noise and removed from the graph, where the threshold L is defined as 20.

3.3. Quantitative Analysis

3.3.1. Skeleton Extraction. Skeleton extraction is a thinning operation on the image, reducing each connected region in the binary image to a single-pixel wide skeleton. After the skeleton extraction is completed, the redundant information in the image can be effectively reduced, the cracks can be described more concisely, and the extracted skeleton information can be used to analyse the cracks in the image quantitatively. Many algorithms have been introduced to obtain the skeleton image from a binary segmented image, such as Zhang and Suen [34], medial axis [35], and K3M [36]. Figure 13 shows the result of different skeleton extraction algorithms. The medial axis algorithm will have more short branches, and the results of the K3M and Zhang–Suen algorithms have fewer short branches. The K3M algorithm is faster and more suitable for the skeletonisation of nuclear containment cracks.

3.3.2. Burr Removal. When extracting the skeleton of the crack on the binary image, some branches will appear on the skeleton, and the length of the branch is much smaller than the overall length of the skeleton. These branches exist because of the uneven width of the cracks or the presence of noise, also known as burrs. In this paper, the method based on the direction chain code is used to eliminate the burr. First, the breakpoints and nodes in the search graph are traversed, and the marks are saved. Then, scanning the branch of the image from the node is continued and a step size threshold is set. If the step size exceeds the threshold value, it will stop. Otherwise, if the other end of the node is

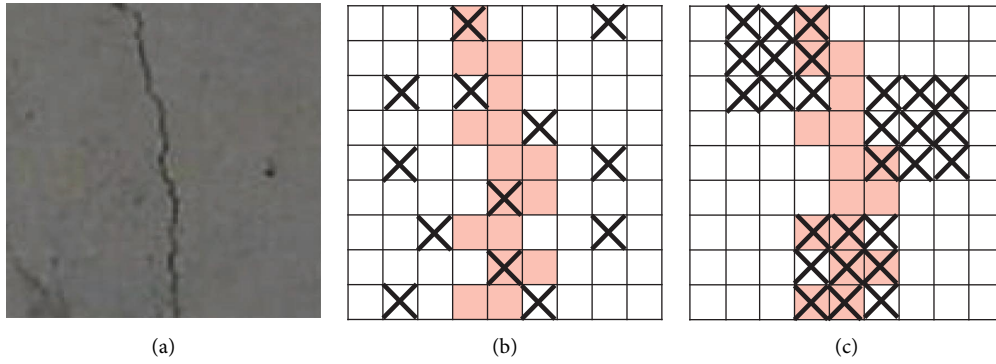


FIGURE 9: Feature extraction of dropout and DropBlock: (a) images, (b) dropout, and (c) DropBlock.

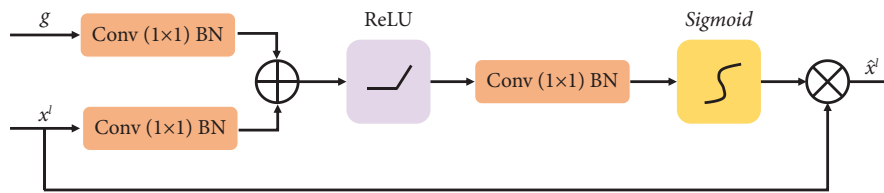


FIGURE 10: Attention gate.

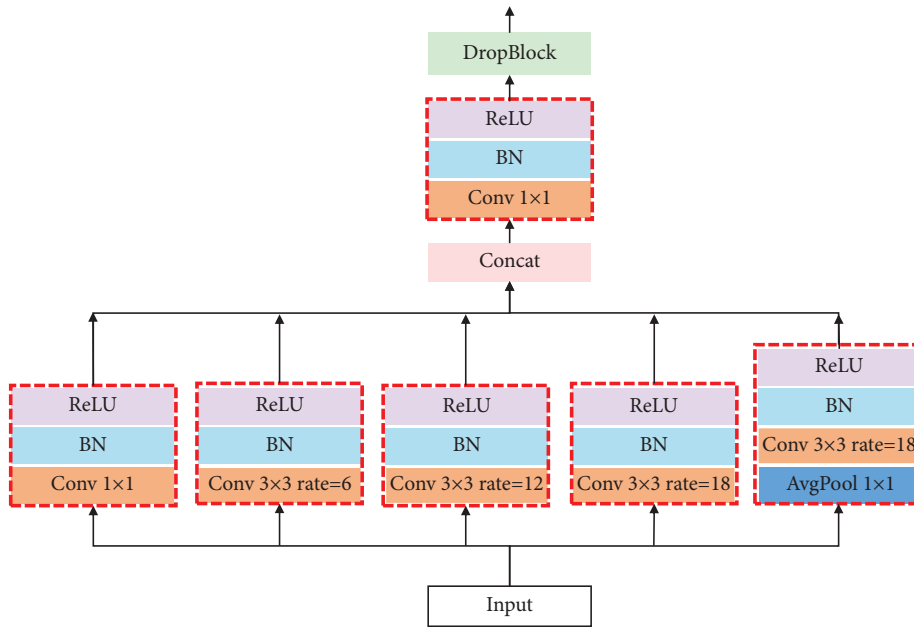


FIGURE 11: ASPP module.

obtained within the threshold value and the other segment is the endpoint, it is judged as a burr and vice versa. Scanning the remaining branches of the node is continued. Then, the burr is removed. If the deletion of the node does not affect the connectivity, they are deleted together. Finally, the branches existing on all endpoints are scanned to remove the image glitch.

3.3.3. *Crack Area Calculation.* For the calculation of the crack area of the nuclear containment vessel, the method is to obtain the number of pixels representing the crack in the crack segmentation image and then the actual area of a single pixel. Finally, the two numbers are multiplied to obtain the area linear crack S , and the solution formula is shown in the following equation:

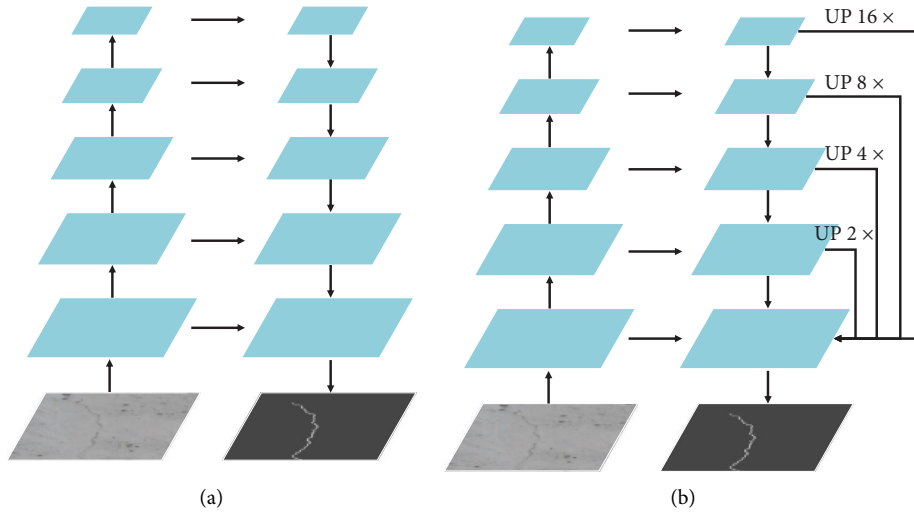


FIGURE 12: Multiscale fusion structure.

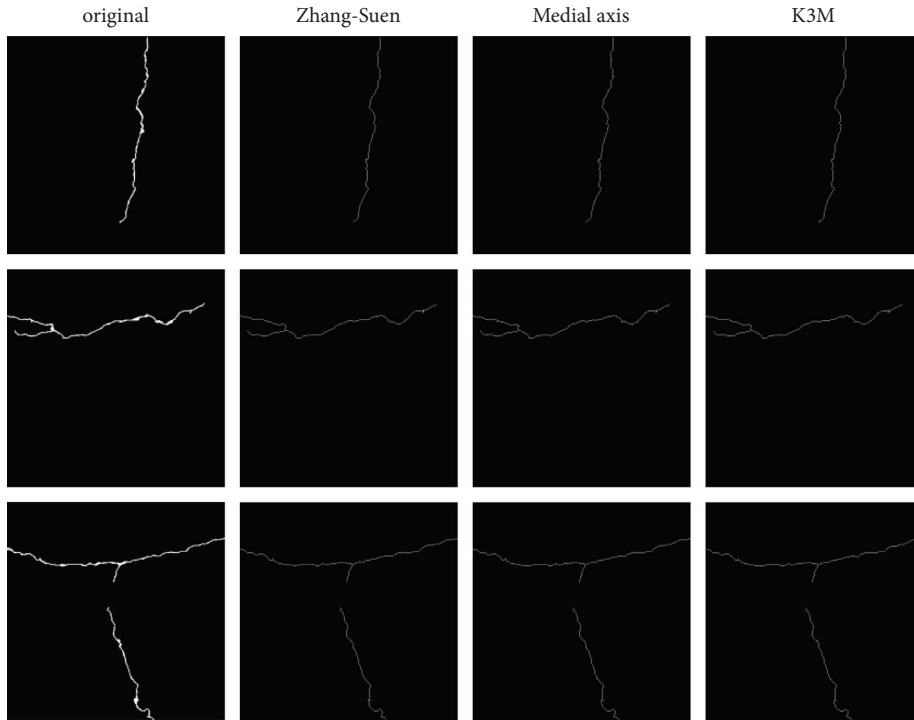


FIGURE 13: Performance comparison of three different skeletonisation algorithms for three types of cracks.

$$S = N * s, \tag{1}$$

where N is the number of white points in the figure and s is the actual area represented by a single pixel.

3.3.4. Crack Length Calculation. Before calculating the crack length, refining the cracker and changing its width to a single-pixel width are necessary. However, the shape of the crack remains unchanged, and the remaining pixels can represent the whole pixel of the crack length. The crack length calculation is based on the crack image obtained by

refinement, and the statistical graph represents the pixel points of the crack. Assuming that the total number is M , the actual length L of the crack is

$$L = M * \mu, \tag{2}$$

where μ refers to the actual length corresponding to a single pixel.

3.3.5. Calculation of Maximum Crack Width. As shown in Figure 14, first, the left and right boundaries of the crack are obtained, the skeleton points (x_{mi}, y_{mi}) obtained by the

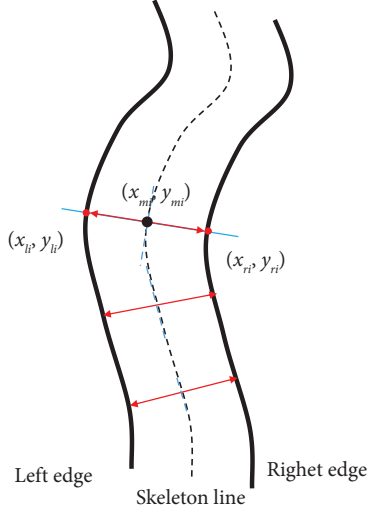


FIGURE 14: Schematic diagram of crack width measurement.

abovementioned method are traversed, and the eight-direction chain code is used to obtain the front and rear skeleton points for each skeleton pixel point that is not the endpoint and fits the quadratic function with three points. The function derivation calculation can obtain the normal direction corresponding to the skeleton point, that is, the crack width direction of the point. Then, the point (x_{li}, y_{li}) closest to the normal in the left boundary and the point (x_{ri}, y_{ri}) closest to the normal in the right boundary are found. The distance between the two points are identified, which is the crack width corresponding to the position of the skeleton point. The formula for calculating the crack width is shown in the following equation:

$$\text{Width} = \sqrt{(x_{li} - x_{ri})^2 + (y_{li} - y_{ri})^2} * \mu. \quad (3)$$

3.3.6. Calculation of Mean Crack Width. Because the edges of some cracks peel off, it is difficult to measure the true width of cracks from the appearance image, which also leads to the measured maximum crack width being larger than that in practice. Therefore, it is necessary to calculate the average width of cracks to further evaluate the cracks accurately. The formula for calculating the mean width is shown in the following equation:

$$\text{Mean_Width} = \frac{S}{L} * \mu, \quad (4)$$

where S refers to the area, L refers to the length, and μ refers to the actual length corresponding to a single pixel.

4. Experimental Results

Our experiment was carried out on three NVIDIA Tesla V100 GPUs in the Supercomputing Center of Wuhan University. All models are implemented in the open-source

deep learning platform PyTorch. We train the proposed model using the Adam optimizer with exponential decay rate parameters set to $\beta_1 = 0.9$ and $\beta_2 = 0.999$ [37]. This method has the advantages of high computational efficiency, low memory requirements, stable parameter update unaffected by gradient expansion and transformation, and faster convergence. The initial learning rate is set to 10⁻⁴, and the polynomial StepLR is used as the learning rate scheduler. After every 20 epochs, the learning rate is reduced by a factor of 0.5. A reduction in the learning rate will prevent overfitting during training. The batch is set to 6 to train the model, and the input image size for training is 512 × 512.

4.1. Loss Function. The loss function is one of the most essential functions in deep learning and is used to calculate the difference between the predicted value and the actual value. The network model adjusts the model parameters through a loss function to minimise the difference. The BCE loss function is commonly used as a cost function in binary classification tasks. The BCE loss function is defined in the following equation:

$$L_{\text{BCE}} = -\frac{1}{n} \sum_{i=1}^n y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i), \quad (5)$$

where y_i is the ground truth of pixel i and \hat{y}_i is the predicted probability for the corresponding pixel.

However, in the crack segmentation task, the proportion of cracks and noncracks in the detected image is highly unbalanced. The pixel ratio of cracks and noncracks in the dataset used in this paper is shown in Table 1. The number of noncrack pixels is much higher than that of cracks. Such imbalance in the number of pixels can cause poor performance of neural network models [38]. In medical image segmentation, the Dice coefficient (Dice) loss function used to solve the class imbalance problem also achieves good results in the crack segmentation task [39].

The calculation of Dice coefficient loss is as follows:

$$L_{\text{Dice}} = 1 - \frac{2|X \cap Y|}{|X| + |Y|}. \quad (6)$$

However, when using Dice coefficient loss, the gradient may change drastically, resulting in unstable training. Therefore, the loss function of the network in this paper adopts a combination of BCE and Dice coefficient loss. It can not only effectively alleviate the effect of the imbalance of the dataset class but also ensure the normal change of the gradient. The loss function is calculated as follows:

$$L_{\text{Mix}} = 0.5 L_{\text{BCE}} + 0.5 L_{\text{Diceloss}}. \quad (7)$$

4.2. Evaluation Metrics. To accurately evaluate the performance of the model, precision, recall, F1 score, and mean intersection-over-union (mIoU) are applied as evaluation

parameters in this article. The definitions are shown in equations (8)–(11). Precision describes how many of the predicted crack points are actual crack points; that is, how many predicted crack points are accurate in the binary case. Recall explains how many actual crack points are predicted. F1 is a comprehensive indicator that combines precision and recall. Our detection tasks include two categories: crack and background. mIoU is the ratio of the intersection and union of the two sets of ground truth and predicted. The evaluation metrics are defined as follows:

$$\text{precision} = \frac{TP}{TP + FP}, \quad (8)$$

$$\text{recall} = \frac{TP}{TP + FN}, \quad (9)$$

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}, \quad (10)$$

$$\text{mIoU} = \frac{1}{2} * \left(\frac{TP}{TP + FN + FP} + \frac{TN}{TN + FN + FP} \right), \quad (11)$$

where the value of the pixel point belonging to the crack is marked as 1 and the value of the pixel point belonging to the background is marked as 0. TP (true positive): crack pixels are correctly predicted as cracks; FN (false negative): background pixels are incorrectly predicted as cracks; FP (false positive): background pixels are correctly predicted as background; TN (true negative): crack pixels are mis-predicted as background.

4.3. Crack Segmentation

4.3.1. Ablation Experiment. First, we design an ablation experiment to evaluate the performance of different loss functions. In addition to BCE and Dice coefficient loss, focal loss and fusion loss functions are also used to train MA_CrackNet. Then, it was tested on the test set, and the results are shown in Table 2.

The table shows that using the BCE loss function can obtain the highest precession, but the recall is the lowest at 86.64%. This means that although the accuracy of the model is higher on the test set, a huge amount of background pixels may overwhelm the model, leading to overly conservative prediction results. It detects fewer real cracks. Compared with BCE, using Dice coefficient loss and focal loss to deal with the problem of class imbalance, recall is increased by 5.26% and 7.44%, respectively. However, the precession indicators have decreased, especially focal loss, whose precession decreased by 7.35%. Precision and recall are often in conflict with each other, and a trade-off between recall and precision is needed to select the best model by comprehensive evaluation metrics F1 and mIoU. Compared with BCE, the comprehensive index of Dice coefficient loss has a small improvement: F1 (+0.99%) and mIoU (+0.78%). The focal loss reduces the accuracy too much, resulting in a decrease in comprehensive indicators: F1 (−0.53%) and mIoU (−0.42%). The fusion loss function achieves the best

TABLE 2: Evaluation metrics of different loss functions on the test set of nuclear containment.

Loss function	Precision (%)	Recall (%)	F1 (%)	mIoU (%)
BCE	86.64	86.00	86.32	87.88
Dice coefficient	83.68	91.26	87.31	88.65
Focal loss	79.29	93.44	85.79	87.46
Mix	86.07	89.96	87.97	89.19

Bold values indicate the best values.

results on the test set, which effectively solve the effect of the imbalance problem of the datasets, and the comprehensive indicators are significantly improved: F1 (+1.65%) and mIoU (+1.31%).

Secondly, to verify the effectiveness of each module in the proposed MA_CrackNet, ablation experiments were carried out on the Nuclear containment dataset. Based on MA_CrackNet, different network configurations are constructed for the ablation experiments and the effectiveness of the proposed ASPP module, attention module, and multi_scale fusion algorithm are analysed, respectively. Four evaluation metrics, including precision, recall, F1, and mIoU, are used to assess the performance of each module.

As shown in Table 3, the performance of the model degrades regardless of which module is removed. The multi_scale fusion algorithm has the greatest impact on the performance of the model. This is because the fact that combining features from different layers gives more semantic and geometric information. In addition, the attention module improves the model’s learning of crack features, which helps detect cracks where there is a class imbalance problem and has the greatest impact on the precision. On the other hand, after removing the ASPP module, the metrics (F1 and mIoU) decreased by 1.39% and 1.1%, respectively.

In summary, the results of the proposed MA_CrackNet ablation experiment in Nuclear containment dataset prove the effectiveness of each module proposed in this paper, and the best experimental results can be obtained by merging the three modules together.

Figure 15 shows the prediction results of the model trained by the fusion loss function to obtain a more intuitive understanding of the performance of the model.

The first row shows a cropped image containing cracks. The second row represents the manually labeled corresponding ground truth masks. The segmentation results of our proposed model are listed in the third row. Overall, the proposed model accurately locates the fractures. Although some images have small noise and cracks, we can effectively solve these problems through postprocessing operations.

4.3.2. Comparative Experiments and Discussion. To further examine the performance of MA_CrackNet, we also conduct a comparative study with six other crack detection methods: CrackU-Net [40], DeepCrack [41], Attention U-Net [31], U-Net [42], Segnet [43], and DeeplabV3+ [44]. To ensure valid comparisons, these networks were trained and tested on the same training, validation and testing dataset. The performance of these methods is then compared based on the prediction results.

TABLE 3: Evaluation metrics of different modules on the test set of nuclear containment.

Method	Precision (%)	Recall (%)	F1 (%)	mIoU (%)
MA_CrackNet without_ASPP	85.84	87.34	86.59	88.09
MA_CrackNet without_attention	83.64	88.89	86.19	87.77
MA_CrackNet without_multi_scale	84.44	86.07	85.24	87.05
MA_CrackNet	86.07	89.96	87.97	89.19

Bold values indicate the best values.

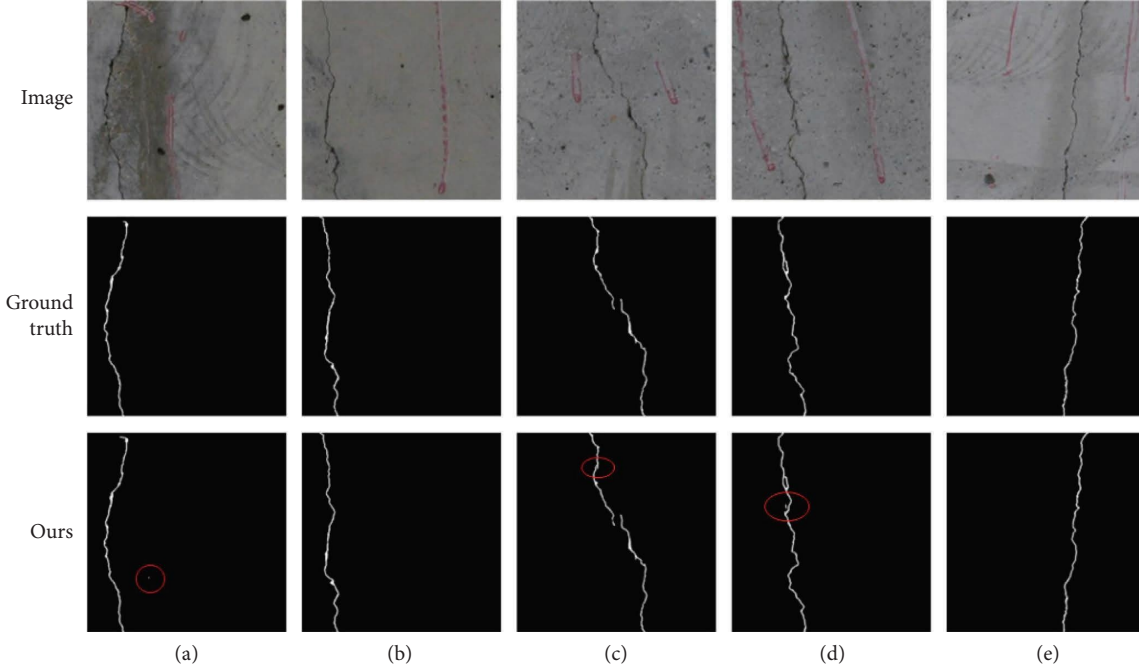


FIGURE 15: Prediction results of the model trained by fusion loss function.

TABLE 4: Evaluation metrics of different networks on the test set of nuclear containment.

Network	Precision (%)	Recall (%)	F1 (%)	mIoU (%)	Parameters (million)	Prediction time (s/image)
DeepLabv3+	79.35	85.51	82.31	72.43	59.30	0.042
Segnet	83.06	81.60	82.33	84.86	29.40	0.028
U-Net	83.89	82.68	83.28	84.87	34.53	0.040
Attention U-Net	85.49	82.77	84.11	85.57	34.90	0.046
DeepCrack	71.52	79.96	75.50	80.16	14.71	0.037
CrackU-net	78.29	89.48	83.51	85.73	62.44	0.078
MA_CrackNet	86.07	89.96	87.97	89.19	71.10	0.071

Bold values indicate the best values.

To quantitatively compare the detection performance of this algorithm with various other algorithms, Table 4 presents the quantitative results of various evaluation indicators such as precision, recall, F1 score, and mIoU. The results show that the proposed network structure is superior to the other six network models in precision, recall, F1, and mIoU. Compared with other networks, the comprehensive index F1 has increased by 25.55%, 5.66%, 5.65%, 4.69%, 3.87%, 14.55%, and 7.78%. mIoU has increased by 16.76%, 4.33%, 4.32%, 3.26%, 3.0%, 9.03%, and 3.45%. Compared with other networks, the proposed network structure shows obvious advantages in the evaluation metrics of the test set. In this paper, we use a model with more layers than the other models to better extract

cracks. Although this has led to an increase in parameters and detection time, the detection speed is still good, and there has been a significant improvement in detection performance. Besides, as an offline containment surface defect detection method, the MA_CrackNet with outstanding detection accuracy is promising and competent.

To show the comparison results of the models more intuitively, this paper presents the comparison of the prediction results between our method and other methods, as shown in Figure 16.

Figure 16 shows the performance of five different models on a single crack and multiple intersecting cracks and also includes noise such as paint and scratches. The rows, in

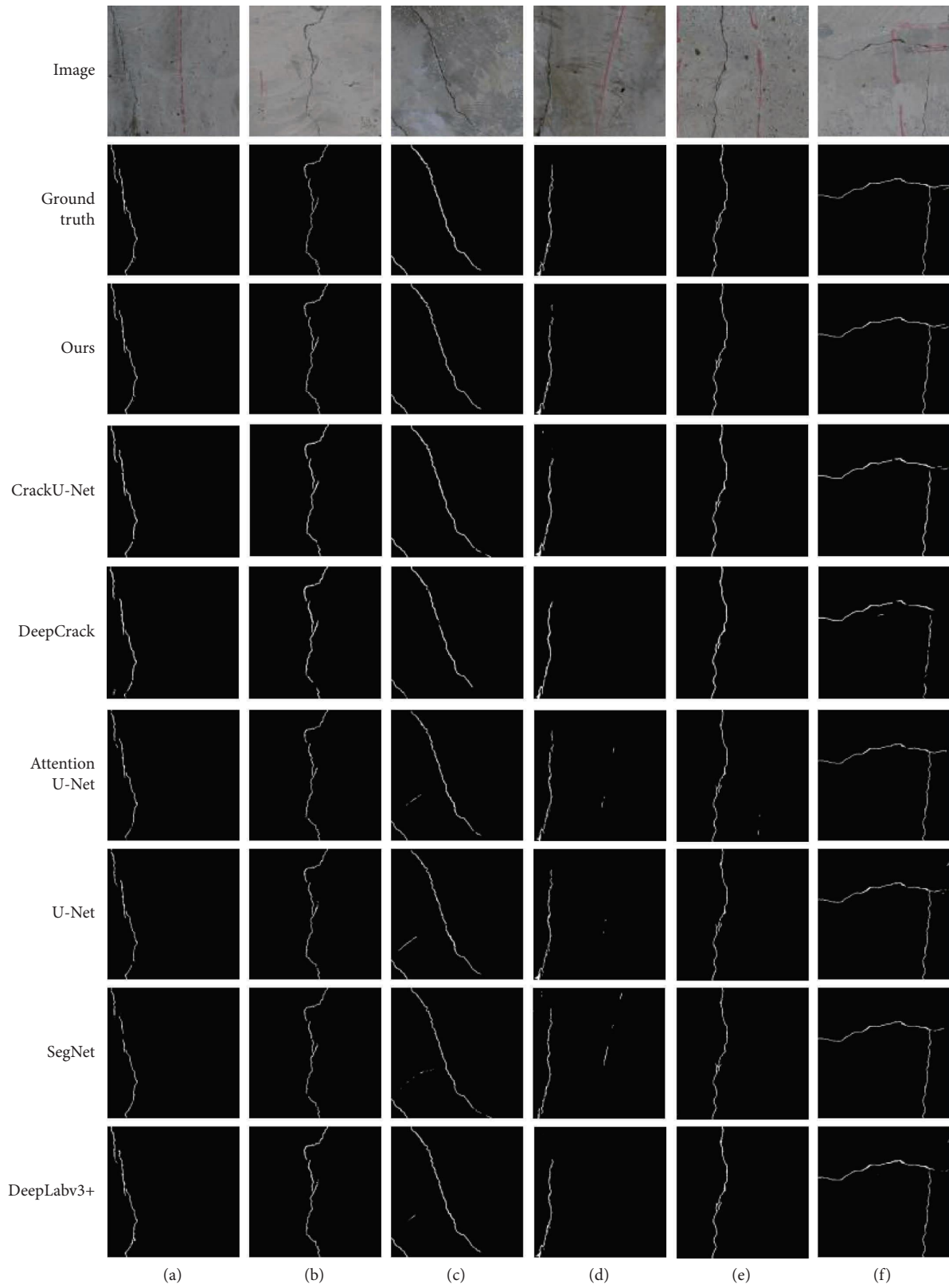


FIGURE 16: Test results of different prediction methods.

order, correspond to the original image, the corresponding ground truth, our method, CrackU-net, DeepCrack, Attention U-Net, U-Net, Segnet, and Deeplabv3+.

In Figures 16(a), 16(b), and 16(e), the cracks detected by the proposed network are more continuous and complete compared with other networks. In Figures 16(c) and 16(d),

our proposed model has a strong anti-interference ability, and the rest of the network models more or less wrongly segment paint traces and scratches into the crack. For more complex cracks, such as in images in Figure 16(f), our proposed network and Attention U-Net and U-Net can segment the cracks more completely, but U-Net is more sensitive to noise, resulting in poor segmentation accuracy. Our proposed network has better anti-interference and better performance in detecting small cracks. Therefore, the network proposed in this paper performs better than other models in the test set results in terms of anti-interference and segmentation of fine cracks. At the same time, it can be seen that the two models CrackU-Net and DeepCrack for pavement cracks achieve general results, especially as can be seen from the results of DeepCrack in Figure 16, which has poor continuity of the cracks obtained by segmenting the cracks, with many breaks in between.

4.3.3. Model Generalization Test. To further verify the performance and generalization of MA_CrackUnet, we conducted training and testing on DeepCrack dataset, which is also a concrete material. DeepCrack dataset is a new benchmark dataset widely used to evaluate the performance of crack detection algorithms. We resized each image and converted it to 384×384 pixels and then performed data enhancement. We divided the data with 3000 images as the training set and 374 images as the test set. We compare the results of MA_CrackUnet with those of existing models. As can be seen in Table 5, MA_CrackUnet can also maintain good performance for cracks of concrete pavement and buildings in DeepCrack dataset, and the recall and F1 indexes are better than those of other models. At the same time, compared with the dataset of the containment crack, the pixel width displayed on the image of the crack in DeepCrack dataset is wider, so MA_CrackUnet achieves better effect, and the four indexes of precision, recall, F1, and mIoU are all improved.

4.4. Crack Measurement. To check the crack measurement performance, the prediction results of the proposed network in the test set were selected for testing. A total of 50 images were used. The predicted image is then subjected to mid-axis skeletonisation. Finally, crack information such as length, width, and area are calculated based on the generated skeleton.

Before measuring the length, width, and area of the crack, we need to process the predicted image to remove the holes and noise of the crack in the image. The effect is shown in Figure 17.

The relationships between the predicted value and the actual value of the four characteristics of the crack (length, area, maximum width, and mean width) are shown in

TABLE 5: Evaluation metrics of different networks on the test set of DeepCrack.

Network	Precision (%)	Recall (%)	F1 (%)	mIoU (%)
U-Net	90.94	73.77	81.47	83.57
Attention U-Net	91.17	73.10	81.14	83.32
DeepCrack	86.10	86.90	86.5	—
CrackU-net	93.57	80.54	86.57	87.54
ECDFNet [45]	90.30	84.30	82.33	—
DMA-Net [46]	86.90	87.10	87.00	—
HACNet [47]	89.20	84.70	86.80	87.80
MA_CrackNet	90.12	89.33	89.73	90.17

Bold values indicate the best values.

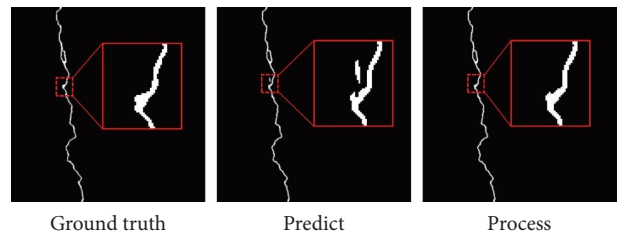


FIGURE 17: False positives and postprocessing in output image.

Figure 18. Figures 18(a) and 18(b) show that the point distribution of length and area is relatively concentrated, the slope of the fitting line of length is 0.91, and the R^2 is 0.8265. The slope of the fitting line of the area is 0.87, and R^2 is 0.8047. In the prediction results of the maximum crack width in Figure 18(c) and the average crack width in Figure 15(d), the point distribution is scattered. The slope of the fitting line of maximum width is 0.99, and R^2 is 0.72. The slope of the fitting line of mean width is 0.76, and R^2 is 0.62. As can be observed from Figure 19, after removing outliers in the figure, the relative error distribution range and relative error average value of crack length are -14.09% to 1.38% and -3.52% , respectively. For the maximum crack width, the relative error distribution range and average relative error are -19.04% to 21.62% and 0.60% , respectively. The relative error distribution range and relative error average of the crack area are -13.86% to 20.89% and 4.6% , respectively. The relative error distribution range and relative error average value of crack mean_width are -11.94% to 55.65% and 24.0% , respectively. Most of the crack lengths measured by our proposed method are smaller than the true value. This is because the fine cracks at the crack tip and the cracks with severe boundary interference are not sufficiently segmented, and cracks are easily mistakenly identified as the background. Part of the reason for the relatively large relative error of the prediction results of the maximum crack width is that the surface cracks of the nuclear containment are very thin. The maximum crack width is only about 9 pixels in the figure, and the average crack

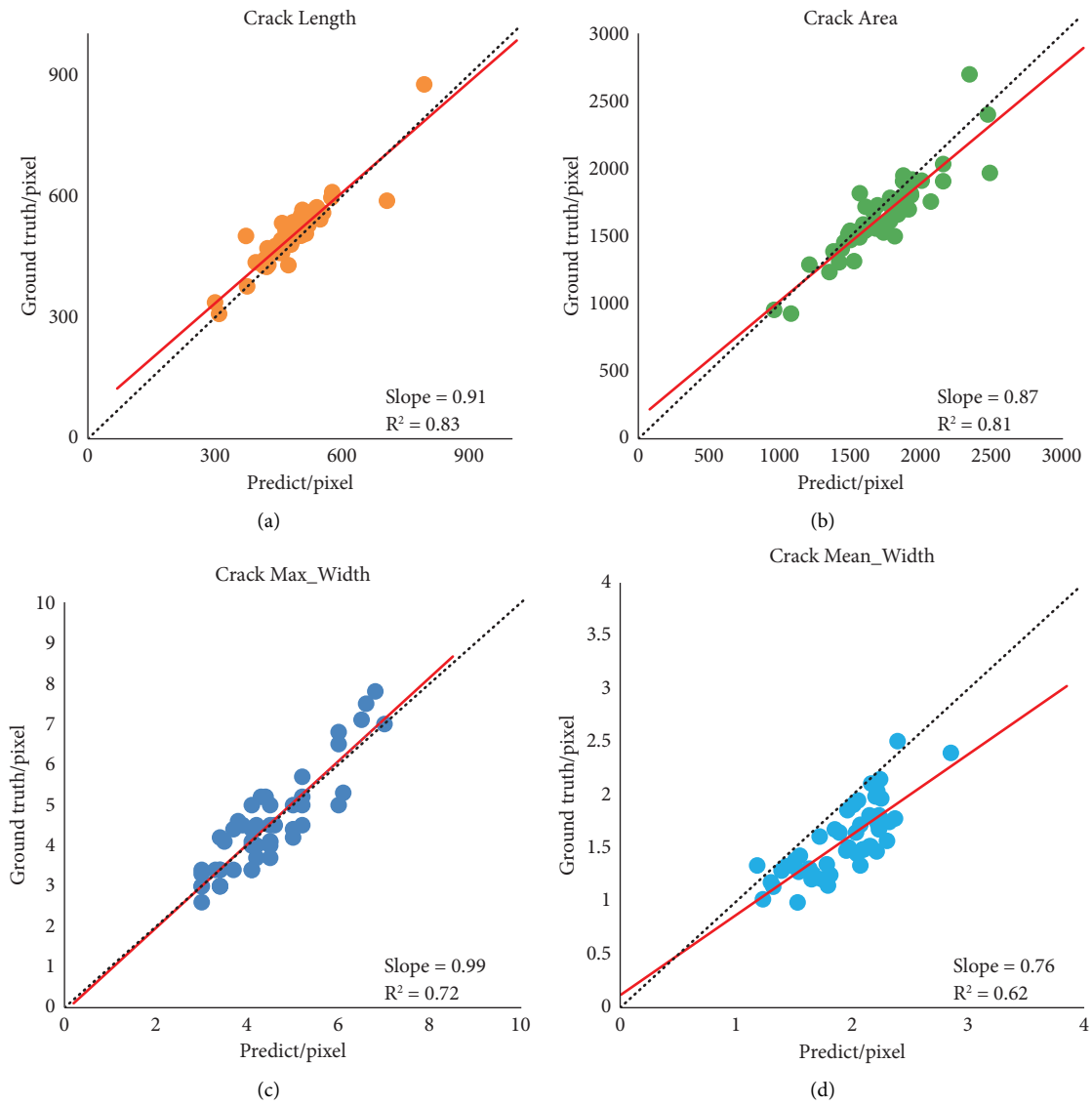


FIGURE 18: Comparison of crack measurement indicators: (a) crack length, (b) crack area, (c) crack max_width, and (d) crack mean_width.

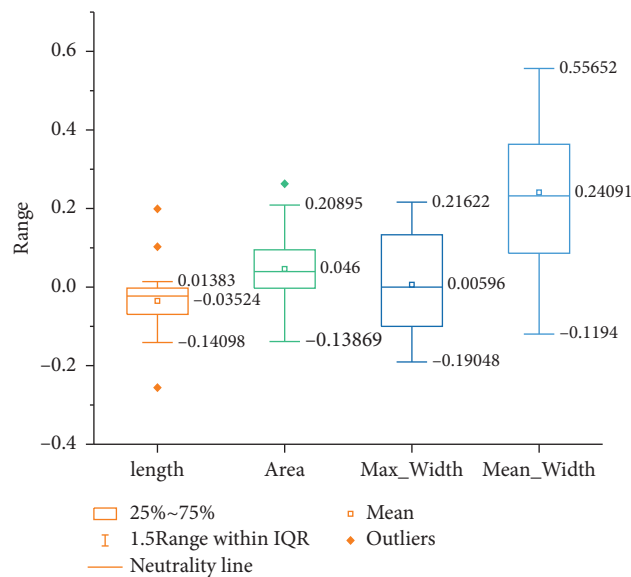


FIGURE 19: Box-plot of the relative error rates for crack measurement indicators.

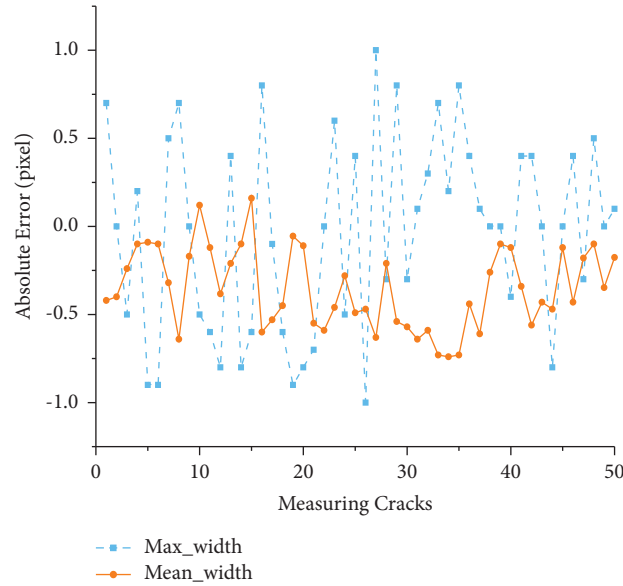


FIGURE 20: Absolute error for crack measurement indicators.

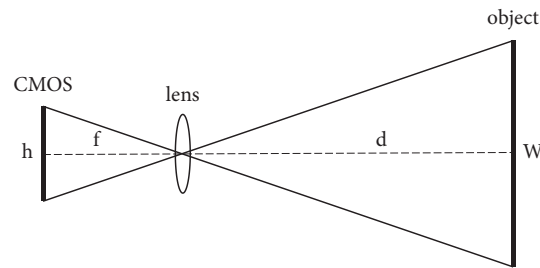


FIGURE 21: Camera imaging principle.

width is about 2 pixels. Misjudging one pixel will cause a large fluctuation in the relative error, so the relative error range of maximum crack width and average crack width is large. As shown in Figure 20, the maximum absolute values of the absolute errors of the measurement results of the maximum width and the average width are 1 and 0.73, respectively. Compared with the results of some existing crack quantification methods, the results of the quantitative calculation of the cracks of the containment in this paper are better. For example, the FPT algorithm used by Ji et al. to quantify the relative error range of asphalt pavement results are length: -25.93% to 14.11% , area: 7.13% to 152.59% , max_width: 30.66% to 152.59% , and mean_width: 19.13% to 156.03% [23], and the relative errors of the calculated results of Chen and He are length: -29.63% to 36.67% , area: -35.90% to 42.72% , and max_width: -31.75% to 28.57% [48].

After obtaining the pixel size of the crack feature, further converting the geometric size represented in pixels into real size is needed to accurately evaluate the crack. The imaging principle of the camera is shown in Figure 21.

The formula for calculating the relationship between the focal length and the photographic distance is as follows:

$$\frac{f}{d} = \frac{h}{W}, \quad (12)$$

where f is the focal length of the lens, W is the object width of the shooting range, d is the shooting distance, and h is the width of the photosensitive element. The object size represented by a single pixel is calculated as follows:

$$x = \frac{W}{w}, \quad (13)$$

where w is the image width. The image width of the camera used for photo shooting is 6,000 pixels and the width of the photosensitive element is 23.5 mm. The camera is equipped with a Nikkor 600 mm lens and a 2x teleconverter. The focal length f is 1200 mm, and the camera stand is located at a distance of $d=50$ m from the surface of the nuclear containment. The image resolution is calculated to be 0.163 mm/pixel.






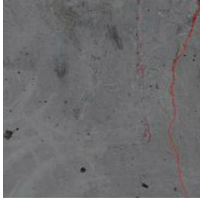
Ground truth		Prediction
	Crack length=84.04 mm Crack area=42.09 mm ² Crack max_width=0.49 mm Crack mean_width=0.20 mm	 Crack length=81.663 mm Crack area=42.19 mm ² Crack max_width=0.55 mm Crack mean_width=0.27 mm
	Crack length=69.11 mm Crack area=39.05 mm ² Crack max_width=0.82 mm Crack mean_width=0.22 mm	 Crack length=68.62 mm Crack area=39.85 mm ² Crack max_width=0.81 mm Crack mean_width=0.23 mm
	Crack length=81.99 mm Crack area=36.96 mm ² Crack max_width=0.65 mm Crack mean_width=0.30 mm	 Crack length=79.54 mm Crack area=40.94 mm ² Crack max_width=0.73 mm Crack mean_width=0.31 mm

FIGURE 22: Crack quantifications at real results in nuclear containment.

The quantification results of the segmented cracks in the nuclear containment appearance image are shown in Figure 22. Given that the resolution of the captured photo is sufficiently large, the actual size represented by the unit pixel is small, so after conversion to the actual size, the measurement results of the length and maximum width of the crack are very close to the actual value. Although the fine cracks at the edge of the image are prone to segmentation failure due to the influence of noise, these errors are all within an acceptable range. Overall, the proposed crack measurement method is valid and accurate.

5. Conclusion

In this study, a deep learning-based crack segmentation network for nuclear containment, called MA_CrackNet, is proposed. Using images obtained by a self-developed remote acquisition system, a dataset of nuclear containment cracks is constructed. Our dataset consists of 450 raw images of nuclear containment damage. The training set was expanded from 400 to 7000 by data augmentation. Based on the results in this study, the following conclusions can be drawn:

- (1) The best performance was obtained by using the fusion loss on the training dataset. It is demonstrated that the loss function of fusion loss outperforms other methods (e.g., BCE, Dice loss, and focal loss) in dealing with the dataset class imbalance problem.
- (2) The precision, recall, F1, and mIoU of the MA_CrackNet network on the containment crack test dataset were 86.07%, 89.96%, 87.97%, and 89.19%, respectively. All of them are better than some advanced models (CrackU-net, DeepCrack, Attention U-Net, U-Net, SegNet, and Deeplabv3+).

To further verify the generalization of the proposed network, the proposed network was compared with the advanced network on the DeepCrack dataset (U-Net, Attention U-Net, CrackU-Net, DeepCrack, ECDFNet, DMA-Net, and HACNet). Among all experimental methods, MA_CrackNet obtained the highest F1 and mIoU scores of 89.73% and 90.17%, respectively. The comparison experiments on two datasets show that the proposed network has better segmentation performance and stronger robustness for thin cracks in nuclear containment.

- (3) In the process of quantitative crack calculation, the relative error range of crack length, area, maximum width, and average width is large, but the deviation is obviously smaller than that of the previous work. Moreover, the average relative error of the results in this paper is small, and the error ranges of the maximum crack and average crack are -1 pixel to 1 pixel and -0.64 pixel to 0.16 pixel, respectively. Translated into the actual distance, the maximum difference of the maximum width is only 0.16 mm, and the maximum difference of the average width is only 0.1 mm, both of which are within the acceptable range and meet the needs of the project.

In summary, the experimental results show that MA_CrackNet can effectively segment the cracks from the appearance image of the nuclear containment and reliably detect the length, area, maximum width, and mean width of the cracks, showing that it has considerable practical potential. However, there are still some limitations that must be addressed. In future studies, special attention should be paid to collecting more high-quality containment images to show more types of containment cracks and improve the

robustness of the proposed method. In the future, various defects such as spalling and corrosion can be automatically segmented and quantitatively analysed to evaluate the appearance of nuclear containment more comprehensively.

Data Availability

The nuclear containment dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] H. Lançon and S. Piot, "A new approach for large structures monitoring: SCANSITES 3D (R), International Symposium on Dams and Reservoirs Under Changing Challenges During the 79th Annual Meeting of the International-Commission-on-Large-Dams (ICOLD)," in *Crc Press-Taylor & Francis Group, Lucerne*, pp. 11–18, Switzerland, 2011.
- [2] Y. Fujita, Y. Mitani, and Y. Hamamoto, "A method for crack detection on a concrete structure," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)*, vol. 3, pp. 901–904, IEEE, Hong Kong, China, August 2006.
- [3] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 1, pp. 62–66, 1979.
- [4] I. Abdel-Qader, O. Abudayyeh, and M. E. Kelly, "Analysis of edge-detection techniques for crack identification in bridges," *Journal of Computing in Civil Engineering*, vol. 17, no. 4, pp. 255–263, 2003.
- [5] G. S. Robinson, "Edge detection by compass gradient masks," *Computer Graphics and Image Processing*, vol. 6, no. 5, pp. 492–501, 1977.
- [6] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [7] B. Y. Lee, Y. Y. Kim, S. T. Yi, and J. K. Kim, "Automated image processing technique for detecting and analysing concrete surface cracks," *Structure and Infrastructure Engineering*, vol. 9, no. 6, pp. 567–577, 2013.
- [8] R. Ali, J. H. Chuah, M. S. A. Talip, N. Mokhtar, and M. A. Shoaib, "Structural crack detection using deep convolutional neural networks," *Automation in Construction*, vol. 133, Article ID 103989, 2022.
- [9] Y. J. Cha, W. Choi, and 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.
- [10] H. P. Leo Pauly and S. Luo, "David hogg and raul fuentes, deeper networks for pavement crack detection," in *Proceedings of the 34th International Symposium on Automation and Robotics in Construction (ISARC)*, pp. 479–485, Kathmandu, Nepal, July 2017.
- [11] Y. Yu, B. Samali, M. Rashidi, M. Mohammadi, T. N. Nguyen, and G. Zhang, "Vision-based concrete crack detection using a hybrid framework considering noise effect," *Journal of Building Engineering*, vol. 61, Article ID 105246, 2022.
- [12] L. Wu, X. Lin, Z. Chen, P. Lin, and S. Cheng, "Surface crack detection based on image stitching and transfer learning with pretrained convolutional neural network," *Structural Control and Health Monitoring*, vol. 28, no. 8, Article ID e2766, 2021.
- [13] K. Hacıfendioğlu and H. B. Başağa, "Concrete road crack detection using deep learning-based faster R-CNN method," *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, vol. 46, no. 2, pp. 1621–1633, 2022.
- [14] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, "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.
- [15] L. Li, H. Zhang, J. Pang, and J. Huang, *Dam Surface Crack Detection Based on Deep Learning*, Association for Computing Machinery, New York, NY, USA, 2019.
- [16] Q. Han, X. Liu, and J. Xu, "Detection and location of steel structure surface cracks based on unmanned aerial vehicle images," *Journal of Building Engineering*, vol. 50, Article ID 104098, 2022.
- [17] X. Yang, H. Li, Y. Yu, X. Luo, T. Huang, and X. Yang, "Automatic pixel-level crack detection and measurement using fully convolutional network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 12, pp. 1090–1109, 2018.
- [18] 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.
- [19] K. Chen, G. Reichard, X. Xu, and A. Akanmu, "Automated crack segmentation in close-range building façade inspection images using deep learning techniques," *Journal of Building Engineering*, vol. 43, Article ID 102913, 2021.
- [20] Y. Pan, G. Zhang, and L. Zhang, "A spatial-channel hierarchical deep learning network for pixel-level automated crack detection," *Automation in Construction*, vol. 119, Article ID 103357, 2020.
- [21] R. Ali and Y.-J. Cha, "Attention-based generative adversarial network with internal damage segmentation using thermography," *Automation in Construction*, vol. 141, Article ID 104412, 2022.
- [22] 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.
- [23] A. Ji, X. Xue, Y. Wang, X. Luo, and W. Xue, "An integrated approach to automatic pixel-level crack detection and quantification of asphalt pavement," *Automation in Construction*, vol. 114, Article ID 103176, 2020.
- [24] D. Kang, S. S. Benipal, D. L. Gopal, and Y.-J. Cha, "Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning," *Automation in Construction*, vol. 118, Article ID 103291, 2020.
- [25] W. Zhao, Y. Liu, J. Zhang, Y. Shao, and J. Shu, "Automatic pixel-level crack detection and evaluation of concrete structures using deep learning," *Structural Control and Health Monitoring*, vol. 29, no. 8, Article ID e2981, 2022.
- [26] M. Mousavi and A. Bakhshi, "Optimized u-shape convolutional neural network with a novel training strategy for segmentation of concrete cracks," *Structural Health Monitoring*, vol. 22, 2022.
- [27] J. Huang, P. Tian, Y. Xu, and C. Xing, "An improved adaptive width template method for crack detection of nuclear containments," *Journal of Surveying Engineering*, vol. 143, no. 4, 2017.

- [28] J. Gu, Z. Wang, J. Kuen et al., "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.
- [29] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," 2013, <https://arxiv.org/abs/1301.3557>.
- [30] G. Ghiasi, T.-Y. Lin, Q. V. Le, and DropBlock, "A regularization method for convolutional networks," *Advances in neural information processing systems*, vol. 31, 2018.
- [31] O. Oktay, J. Schlemper, L. L. Folgoc et al., "Attention u-net: learning where to look for the pancreas," 2018, <https://arxiv.org/abs/1804.03999>.
- [32] W. Choi and Y. J. Cha, "SDDNet: real-time crack segmentation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 9, pp. 8016–8025, 2020.
- [33] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: semantic image segmentation with deep convolutional nets," *Atrous Convolution, and Fully Connected CRFs*, vol. 40, 2016.
- [34] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Communications of the ACM*, vol. 27, no. 3, pp. 236–239, 1984.
- [35] G. K. Viswanathan, A. Murugesan, and K. Nallaperumal, "A parallel thinning algorithm for contour extraction and medial axis transform," in *Proceedings of the 2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN)*, pp. 606–610, Tirunelveli, India, March 2013.
- [36] K. Saeed, M. Tabędzki, M. Rybnik, and M. Adamski, "K3M: a universal algorithm for image skeletonization and a review of thinning techniques," *International Journal of Applied Mathematics and Computer Science*, vol. 20, no. 2, pp. 317–335, 2010.
- [37] D. P. Kingma and J. Ba, "A method for stochastic optimization," in *Proceedings of the International conference on learning representations (ICLR)*, San Diego, California, USA, May 2014.
- [38] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: the effects of imbalanced datasets on classification performance," *Neural Networks*, vol. 21, no. 2-3, pp. 427–436, 2008.
- [39] B. Chen, H. Zhang, Y. Li, S. Wang, H. Zhou, and H. Lin, "Quantify pixel-level detection of dam surface crack using deep learning," *Measurement Science and Technology*, vol. 33, no. 6, pp. 065402–065414, 2022.
- [40] J. Huyan, W. Li, S. Tighe, Z. Xu, J. Zhai, and CrackU-net, "CrackU-net: a novel deep convolutional neural network for pixelwise pavement crack detection," *Structural Control and Health Monitoring*, vol. 27, no. 8, Article ID e2551, 2020.
- [41] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "DeepCrack: a deep hierarchical feature learning architecture for crack segmentation," *Neurocomputing*, vol. 338, pp. 139–153, 2019.
- [42] O. Ronneberger, P. Fischer, and T. Brox, "Convolutional networks for biomedical image segmentation," in *Proceedings of the Medical Image Computing and Computer-Assisted Intervention--MICCAI 2015*, Munich, Germany, October 2015.
- [43] 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.
- [44] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Re-thinking atrous convolution for semantic image segmentation," 2017, <https://arxiv.org/abs/1706.05587>.
- [45] Q. Zhou, Z. Qu, S. Y. Wang, and K. H. Bao, "A method of potentially promising network for crack detection with enhanced convolution and dynamic feature fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18736–18745, 2022.
- [46] X. Sun, Y. Xie, L. Jiang, Y. Cao, B. Liu, and Dma-Net, "DMA-net: DeepLab with multi-scale attention for pavement crack segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18392–18403, 2022.
- [47] H. Chen and H. Lin, "An effective hybrid atrous convolutional network for pixel-level crack detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.
- [48] J. Chen and Y. He, "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*, vol. 37, no. 13, pp. 1721–1736, 2022.