Crack Image Augmentation and Segmentation Based on Convolutional Block Attention Implicit Diffusion Model

Zhang Pengwei[1], Zhao Chen[1,a,*], Chen Jingxia[1], Wang Zikai[1]

[1]School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an, China
[a]jackerhack@163.com
[*]Corresponding author

Abstract: *To solve the problem of limited and difficult data collection in traditional road crack image segmentation, a Convolutional Block Implicit Diffusion Model (CBIDM) based on convolutional block attention mechanism is proposed to generate and enhance crack images, highlighting features with high correlation with cracks in the image, making the model more sensitive to the connection between small cracks and coarse and fine cracks. Based on the public dataset CRACK500, experiments were conducted on the model proposed in this paper. The results showed that after expanding the original data with generated images in a 1:1 ratio, the U-Net segmentation model was trained and tested. The mIoU and mAP indicators for crack segmentation were improved by 2.63% and 4.84% respectively compared to the original dataset, with an average accuracy of 97.05%. This verified that using the proposed model for data generation and enhancement can effectively improve the performance of crack image segmentation.*

Keywords: *diffusion model; crack segmentation; image generation; data augmentation; convolutional block attention*

1. Introduction

At present, the total mileage of highways in China has reached 5.2 million kilometers[1].Having a good road network and transportation system has become a reliable guarantee for promoting rapid national economic growth and accelerating the process of urban intelligent construction. Crack image segmentation refers to the technique of identifying and labeling cracks in an image and distinguishing them from the background area. It has important application value in fields such as geological exploration, road maintenance, and soil engineering.

The traditional crack image segmentation method is achieved by manually extracting the texture features, edge features, shape features, and color features of cracks. This approach has strong interpretability and specificity, but due to its excessive reliance on subjective judgment, it results in low robustness and generalization, and has the drawbacks of being time-consuming and labor-intensive. In recent years, crack image segmentation methods based on deep learning have achieved significant results in solving this problem[2].Due to the diverse shapes and sizes of cracks, as well as the interference factors of image noise and complex backgrounds, obtaining sufficient and accurately labeled crack image samples has become a challenging problem in the field of crack segmentation[3].Currently, most studies use volume stacking and image processing methods to expand and enhance data. Volume stacking refers to the merging of crack images from different scenes, which can increase the number of images. However, the lighting conditions and road materials of different cracks can cause the model to be unable to learn accurate crack features. Image processing refers to the processing of crack images through cropping, angle rotation, brightness adjustment, horizontal or vertical flipping, and using the processed image as an expansion part of the data. This approach introduces redundant information and bias, and simple image processing cannot guarantee sample diversity.

In response to the above issues, due to the high quality of generated images and fast sampling speed of DDIM (Denoising DIFFUSION EXPLICIT MODELS), this paper proposes a CBIDM model for image generation and enhancement based on the DDIM model and the convolutional block attention mechanism, to solve the problem of low performance in crack segmentation using deep learning models due to insufficient crack images.

2. Related Work

At present, the road crack image segmentation technology mainly relies on two methods: digital image processing and deep learning.

The commonly used road crack segmentation methods based on digital image processing technology mainly include controllable filters, non local means, projection integration, and image thresholding methods. Lyer et al.[4] Lyer et al. used mathematical morphology and curvature evaluation techniques to separate cracks from background patterns in noisy environments through filtering, thus completing crack segmentation tasks. Yiyang et al.[5] used threshold segmentation method, first calculate the roundness index of the target area and perimeter, and then compare it with other areas to achieve the detection and segmentation of glass surface cracks. On the basis of analyzing the characteristics of asphalt pavement cracks, Hoang et al.[6] discovered a new set of features guided by image projection integration, which significantly improved the predictive performance of cracks. Akagic et al.[7] divided the concrete crack image into four equally sized independent sub images, searched for the target crack area using the ratio between Ostu threshold and maximum histogram value based on each sub image, and finally combined it into the result image. This method is highly efficient in low signal-to-noise ratio situations. Although the above methods have improved the depth and speed of feature extraction to a certain extent, they still require manual adjustment of parameters according to specific situations, with weak generalization ability and high requirements for data quality and preprocessing, which cannot meet the high-quality crack segmentation requirements in data scarcity situations.

In recent years, deep learning methods have made significant progress in crack image segmentation due to their strong adaptability, high flexibility, end-to-end learning, and data-driven advantages [8]. Cheng et al.[9] developed a real-time neural network threshold method for road crack segmentation. Xu et al.[10] proposed a crack detection method based on BP neural network, which automatically identifies crack areas in images through histograms and spatial filtering. Jenkins et al.[11] proposed a Transformer based convolutional neural network that simplifies the input crack image into a set of low dimensional encoded features, and then maps the encoded features to the crack image through indexing, completing the crack segmentation task.

The diffusion model is a two-stage deep generation model based on forward diffusion and backward diffusion. Gu et al.[12] formalized the instance segmentation task as a denoising process from noise to filter, demonstrating the strong competitiveness of diffusion models in segmentation tasks. Nguyen et al.[13] proposed a text to image generation model that utilizes attention mechanism combined with class hint attachment to generate pixel level semantic segmentation labels through diffusion, demonstrating the potential of attention mechanism in the field of diffusion generation. Fang et al.[14] used a diffusion model to generate visual priors to control the generation of synthesized data, filtered noise using class calibrated CLIP scores, and enhanced data for image object detection tasks. Trabucco et al.[15] pre trained a text to image diffusion model, which solved the problem of lack of diversity in traditional data augmentation through image conversion, and improved the accuracy of the test domain in small sample image recognition tasks. Yu et al.[16] trained a diffusion model to synthesize tissue pathology images based on nuclear structure. The generated images were used to train the segmentation model. By synthesizing 10% of the real dataset as an extension, segmentation results comparable to fully supervised baseline methods can be achieved. Taking inspiration from this, this article applies diffusion models to the field of road crack segmentation, combines convolutional block attention mechanism to learn deep crack features, generates high-quality and diverse crack image samples, and improves the performance of crack image segmentation.

3. Construction of Convolutional Attention Implicit Diffusion Model

3.1 Implicit denoising diffusion model for crack image enhancement

In response to the scarcity of crack images and the inability to meet the needs of deep learning for a large amount of training data, resulting in low accuracy in crack segmentation, this paper adopts an implicit denoising diffusion model[17] as the backbone network to generate and enhance existing crack samples. Based on the original crack features, crack image data with sample diversity is generated, while improving the sample sampling speed.

The use of implicit denoising diffusion models to generate crack images is mainly divided into two processes: forward and backward. In the forward process, the specified crack image x_0 conforms to the probability distribution of $q(x)$, and gradually adds Gaussian noise to x_0 through k steps to obtain a crack sample x_t containing t times of noise, The forward process adds T times of noise to x_0. The step size of each extended walk depends on the hyperparameter group { $\beta_t \in (0,1)$ } $(1 \le t \le T)$ that conforms to the Gaussian distribution variance, as shown in formula (1). The crack sample x_t at time t is only related to the sample state at time $t-1$, and this process is considered as a Markov chain, as shown in formula (2). The larger the value of t, the closer x_t is to Gaussian noise. When T approaches ∞, x_t can be approximated as standard Gaussian noise related to the mean coefficient.

$$q(x_t \mid x_{t-1}) = N(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t I) \tag{1}$$

$$q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}) \tag{2}$$

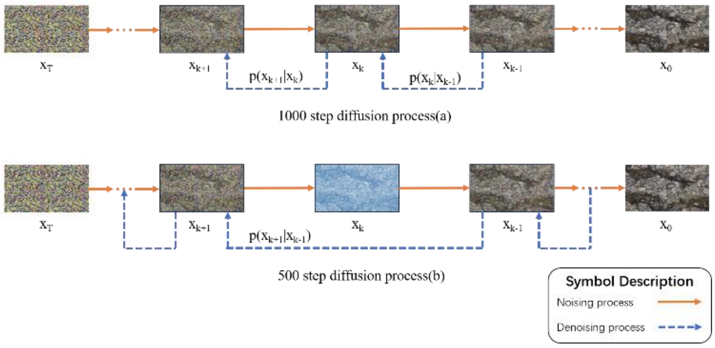
In the reverse process, DDPM obtains the noise of the current state x_k, which is predicted by the denoising state from the high noise sample x_T to the previous state x_{k+1} of x_k. The image at time T cannot be generated until the entire chain is generated. This prediction method can result in heavy computational burden due to too many sampling steps. The Denoising Diffusion Implicit Model (DDIM) used in this article uses a non-Markov method that does not require referencing all denoising states from x_T to x_{k+1} before the current crack image x_k state containing noise. It allows certain sampling steps in the reverse process to be skipped, and can also predict the noise of the current state for the sample. DDIM redefines the one-step denoising process, as shown in formula (3), where $\alpha_t = 1-\beta_t$, ε represents the predicted noise, and σ add θ represent the real vector $\mathrm{R}_{\ge 0}^{\mathrm{T}}$, both of which are learnable parameters. When σ approaches 0, formula (3) can be converted to denoising formula (4).

The training purpose of the network is to make the predicted noise distribution similar to the actual noise distribution of t at the current time step. By using the variational method, the following objective functions can be optimized at each time step:

$$x_{t-1} = \sqrt{\alpha_{t-1}} \left(\frac{x_t - \sqrt{1-\alpha_t}\, \varepsilon_\theta(x_t)}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \varepsilon_\theta(x_t) + \sigma_t \varepsilon_t \tag{3}$$

$$x_{t-1} = \sqrt{\alpha_{t-1}} \left(\frac{x_t - \sqrt{1-\alpha_t}\, \varepsilon_\theta(x_t)}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t-1}} \cdot \varepsilon_\theta \tag{4}$$

In formula (4), no noise is added to the x_{t-1} value. Under the condition of $\sigma = 0$, the reverse process is deterministic, and the only noise with randomness is the initial noise at x_0 in the crack image. Due to the fact that the Markov chain in the reverse process is only used in the probability part, the noise is deterministic in the above process. Therefore, it is possible to break the Markov chain to reduce the actual diffusion steps and improve sampling efficiency. The use of a 1000 step Markov chain diffusion process and a 500 step non-Markov chain diffusion process are shown in Figure 1. The latter is used in this experiment.



1000 step diffusion process(a)

500 step diffusion process(b)

Figure 1: Comparison of 1000 step and 500 step reverse processes

From Figure 1 (a) and (b), it can be observed that the forward process of the two is the same. (b) Set the step size to 2 in the reverse process, sample every other diffusion step during the diffusion process, and obtain the noise sequence $X^{'}$ ={x0,x2,…,xT-2,xT}. Remove the noise corresponding to each image in 500 steps in sequence, and finally obtain the generated crack image.

3.2 Crack noise prediction network based on U-net and convolutional block attention

Due to the relatively small proportion of crack pixels in an image sample, small crack pixels in some samples account for about 5% of the total pixels. Treating all channels of the feature map equally will make it difficult for the model to distinguish the role of each channel in the crack image segmentation task. Therefore, this article uses channel attention mechanism to focus the model's attention on feature channels that play an important role in crack segmentation tasks, thereby highlighting key features related to cracks in the image and improving the learning ability of crack features. In addition, due to the influence of different lighting conditions, crack material and shape size during image acquisition, there is a certain spatial correspondence between crack features and background features in the sample. Therefore, this article introduces CBAM Net[18] into the U-net network used for noise prediction in the DDIM model, and proposes the Convolutional Block Implicit Noise Diffusion Model (CBIDM), which utilizes channel attention to suppress useless feature channels for crack detection and reduce the interference of noise on the model; Combining spatial attention mechanism to improve the model's perception ability of crack location and morphology, paying more attention to local areas related to cracks in crack images, and helping crack segmentation models better understand the spatial structure and feature distribution of cracks.

The U-net network utilizes skip connections in the encoder decoder structure, which can improve the efficiency of noise prediction, reduce information loss in the network, and improve the accuracy of noise estimation. Therefore, the DDIM model uses U-net for reverse noise prediction. This article first utilizes a CBAM module that combines channel attention and spatial attention mechanisms to improve the U-Net[19] in the implicit denoising diffusion model, and applies it to road crack segmentation tasks. The improved U-Net crack noise prediction network architecture is shown in Figure 2.

The crack noise prediction network consists of four layers of Down sampling modules, intermediate modules, and four layers of Up sampling block modules. The crack features are first Down sampled using two residual blocks and one convolutional block attention module, gradually reducing the spatial size of the crack features. As a feature of the encoded diffusion step, time vector can capture the periodicity and relative position of time, helping the network distinguish different noise levels and crack image states. The network uses sine position encoding as a time vector embedding in the Up sampling blocks at each layer. After passing through the convolutional block attention module of the intermediate module, the network learns key features highly related to prediction noise. Then, through skip connections in the network, the corresponding Up sampling blocks of each layer are linked to the residual blocks in the Down sampling blocks, better preserving and utilizing the noise feature information of different levels, while reducing the possibility of gradient vanishing or exploding. Finally, the network obtains high-dimensional crack noise feature information through 4-layer Up sampling. The training of the crack noise prediction network aims to match the predicted noise distribution with the actual noise distribution at the current time. The loss function of the network can be derived using the variational method, as shown in equation (5).

$$Loss = \left\| \varepsilon_t - \varepsilon_\theta \left(\sqrt{\alpha_t} x_0 + \sqrt{1-\alpha_t} \varepsilon_t , t \right) \right\|^2 \tag{5}$$

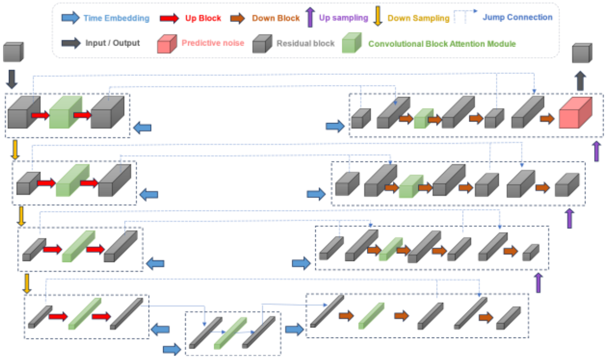


Figure 2: Framework diagram of crack noise prediction network based on U-net

The convolutional block attention module in this network combines the characteristics of channel attention and spatial attention, and its structure is shown in Figure 3. The input features are weighted by two attention mechanisms to obtain more relevant output features, as shown in equation (6). In the formula, represents input features, represents output features, and represents attention mechanism modules.

$$F' = F \cdot AM_{Channel}(F) \cdot AM_{Spatial}(F) \tag{6}$$

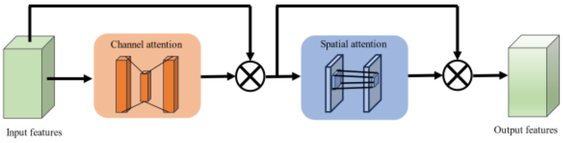


Figure 3: Schematic diagram of convolutional block attention module structure

The structure of channel attention is shown in Figure 4. Firstly, the input features are averaged and maximally pooled in the spatial dimension through a convolutional kernel with a height and width of 1×1, respectively, to compress the learning of channel features in spatial dimensions. Input the two results into a multi-layer perceptron to learn the features of the channel dimension and the weights of each channel, then add up the output results and activate them through the sigmoid function to obtain the channel attention value. The final output feature map is obtained by weighted product of the original input features using attention values, and the calculation formula is shown in (7). In the formula, represents the output feature, represents the output feature, represents the multi-layer perceptron, and represents the average pooling operation and the maximum pooling operation, respectively.

$$F_{out} = F_{in} \times \mathrm{sigmoid}(MLP(AVG\_POOL(F_{in})) + MLP(MAX\_POOL(F_{in}))) \tag{7}$$

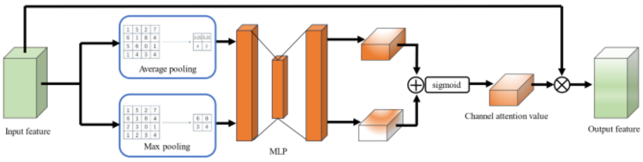


Figure 4: Schematic diagram of channel attention module structure

The structure of the spatial attention module is shown in Figure 5. Similar to the channel attention mentioned above, the input features are first averaged and maximally pooled, and the average and maximum values are taken on the channel of a feature point. Then concatenate the two results according to the channels to obtain a feature tensor with dimensions of length x width x 2. The final concatenated result is reduced to length×width×1 through convolution, and the spatial attention weight is obtained through activation function. Multiply the attention with the original input feature to obtain the output feature map, and the calculation formula is shown in (8). In the formula, represents the input features, represents the output features, represents the use of a convolution kernel with a size of 7×7 for convolution operations, and represents the average pooling operation and the maximum pooling operation, respectively.

$$F_{out} = F_{in} \times \mathrm{sigmoid}(Cov_{7\times 7}(AVG\_POOL(F_{in}); MAX\_POOL(F_{in}))) \tag{8}$$

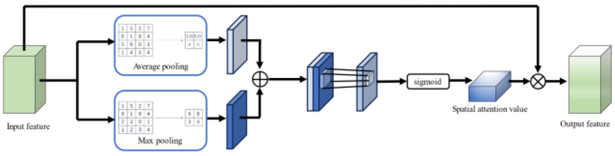


Figure 5: Schematic diagram of spatial attention module structure

3.3 Data synthesis and enhancement based on CBIDM model

Due to the probabilistic randomness of the generation of diffusion models, the segmentation labels generated by the model based on crack segmentation results also have randomness and cannot correspond to the generated image samples. Therefore, this article proposes a data augmentation method based on the CBIDM model, as shown in Figure 6.

This method first uses the CBIDM model to generate an expanded image sample (C_{aug}-Img) based on the crack image C_{src}*-Img* from the original training set. Then, the C_{aug}*-Img* in Figure 6 is input into the U-Net network trained from the original training set [19] to obtain the corresponding label (C_{aug}*-GT*) for C_{aug}*-Img*. The new dataset (C_{new}) obtained at this time is shown in equations (9) and (10).

$$C_{new}\text{-}Img = C_{src}\text{-}Img + C_{aug}\text{-}Img \tag{9}$$

$$C_{new}\text{-}Gt = C_{src}\text{-}Gt + C_{aug}\text{-}Gt \tag{10}$$

If the U-Net model in Figure 6 is trained on another crack dataset Q with more accurate annotations, the quality of the generated labels will not only depend on the quality of image generation, but also be affected by Q. The acquisition of C_{aug}*-GT* in this method does not rely on other crack data Q, thus eliminating this interference. This method makes the quality of image generation the only variable that affects the performance of the segmentation model, which can better test the impact of data augmentation on the performance of the crack segmentation model.

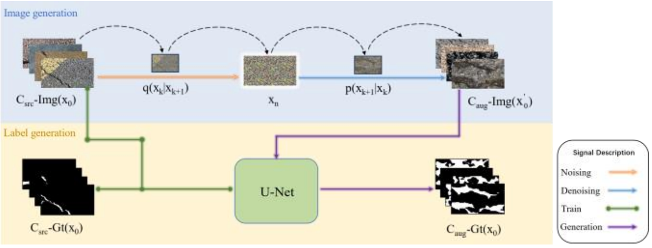


Figure 6: Framework diagram of data augmentation method based on diffusion model

This article proposes a data enhanced crack segmentation comparison method, as shown in Figure 7, to eliminate possible interference factors. Firstly, the U-Net network was trained using two datasets, C_{new} and C_{src}, to obtain U-Net1 and U-Net0 models. Use U-Net1 and U-Net0 to segment the test set C_{src}*-Img* (test) separately, and obtain segmentation labels GT1 and GT2. This method uses the method of controlling variables, so that the difference in segmentation results depends entirely on the training set of the model, and the evaluation index results can directly reflect the quality of the training set.

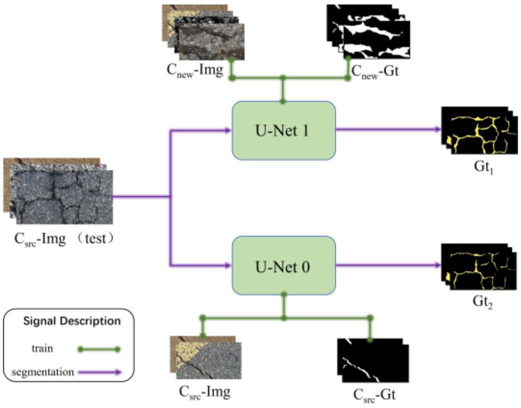


Figure 7: Framework diagram of crack segmentation method based on data augmentation

4. Experiments

4.1 Dataset

To verify the effectiveness of the proposed method, this paper conducted experiments using the publicly available crack dataset CRACK500. This dataset was taken by Yang et al.[20]at Temple University using a mobile phone to capture 500 road crack images with a pixel size of 1440 x 2560, and each image was annotated pixel by pixel. In order to meet the requirements of the network model, each original image and its corresponding label image are cropped into 16 non overlapping image regions of the same size. After filtering and removing sub images without cracks, the remaining 3368 crack images with a pixel size of 640 x 360 and their corresponding labels constitute the dataset used in this experiment.

1896 images in the CRACK500 dataset were used as the training set, 348 images were used as the validation set, and 1124 images were used as the testing set. Crack images and corresponding labels of four types of road materials and light conditions in the dataset as shown in Figure 8.

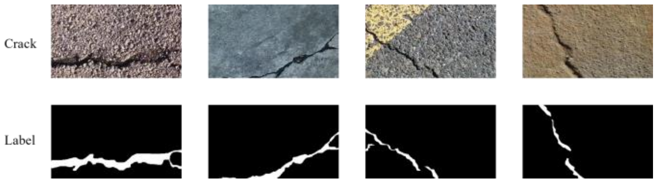


Figure 8: Partial presentation of the CRACK500 dataset

4.2 Experimental Environment and Settings

The hardware environment used in this experiment is an NVIDIA GeForce RTX3090 graphics card with an Intel (R) Xeon (R) Gold 6226R CPU and 24GB of memory. The software environment operating system is CentOS7, CUDA version is 11.6, and Python is used as the framework. The CBIDM experimental environment is Python 3.7 and its related Vision library, with a forward diffusion step of 1000 and a reverse sampling step of 500. The number of training rounds is 500, and the training batch is set to 1. The segmentation model U-net has 4 layers for Up sampling and Down sampling, 50 training rounds, 8 training batches, and an initial learning rate of 0.00001. The channel attention ratio parameter is set to 16, and the convolution kernel size for spatial attention is set to 7.

4.3 Evaluating indicator

In order to accurately evaluate the enhanced segmentation performance, this paper uses classic evaluation metrics in the field of semantic segmentation, including mean Intersection over Union (mIoU), mean Average Precision (mAP), and Accuracy (Acc). Among them, mIoU calculates the average intersection over Union ratio between all predicted results and the true labels. The higher the mIoU value, the better the segmentation performance of the model for all categories; mAP calculates the average accuracy between Precision and Recall for different categories, and after plotting the P-R curve for each image, averages the area under the curve; Acc calculates the ratio between the number of correctly classified pixels at the pixel level and the total number of pixels. The calculation of each indicator is as follows:

$$\mathrm{mIoU} = \frac{1}{\mathrm{n}} \sum_{i=0}^{\mathrm{n}} \frac{\mathrm{TP}}{\mathrm{FP+FN+TP}} \tag{11}$$

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{FP} + \mathrm{TP}} \tag{12}$$

$$\mathrm{mAP} = \frac{1}{\mathrm{n}} \sum_{i=0}^{\mathrm{n}} \mathrm{AP_i} \tag{13}$$

$$\mathrm{Acc} = \frac{1}{\mathrm{n}} \sum_{i=0}^{\mathrm{n}} \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{FP} + \mathrm{FN} + \mathrm{TP} + \mathrm{TN}} \tag{14}$$

In the above formula, it indicates that the model correctly predicts the number of pixels in the crack area as the crack area; Indicates that the model incorrectly predicted the number of pixels in non-crack areas as crack areas; Indicates that the model incorrectly predicted the number of pixels in the crack area as a non-crack area; Indicates that the model correctly predicts the number of pixels in non-crack areas as non-crack areas; Represents the area under the P-R curve of the image.

5. Experimental Results and Analysis

5.1 Analysis of ablation experiment results

In order to explore the impact of different numbers of enhanced samples on the experimental results, the expansion ratio (original training set data size: generated data size) was set to 0.5, 0.75, 1, 1.25, and 1.5, respectively. The CBIDM network using convolutional block attention mechanism was used for data augmentation to obtain five corresponding expanded training sets. The three performance indicators of the model obtained after training on the original CRACK500 training set and five expanded training sets are shown in Table 1.

Table 1: Comparison of model performance after setting different expansion ratios (%)

Enhancement ratio	mIoU/%	mAP/%	Acc/%
1:0.5	75.50	80.81	97.02
1:0.75	75.91	81.59	97.04
1:1	**76.32**	**82.51**	**97.05**
1:1.25	73.72	77.79	96.93
1:1.5	74.68	79.62	96.95

According to the data in Table 1, when the expansion ratio is 1:1, after enhancing the CRACK500 crack dataset, the mIoU reaches 76.32%, mAP reaches 82.51%, and Acc reaches 97.05%, all of which are higher than the experimental indicators of the other four expansion ratios. At this time, the enhancement effect reaches the best. When the expansion ratio exceeds 1:1, the noise in the generated crack image samples increases, making them far from the true crack features. The use of crack samples containing a significant amount of noise in model training leads to a decrease in the performance of the segmentation model. The experiment shows that using CBIDM to enhance crack data can effectively improve the segmentation effect of road cracks. When the number of generated images is the same as the original data, the enhancement effect reaches the best.

In order to further verify the enhancement effect of convolutional block attention on crack images, the original training set was used as a benchmark and the expansion ratio was set to 1:1. The experiments used enhancement methods including self attention mechanism, channel attention mechanism, spatial attention mechanism, and convolutional block attention mechanism, respectively. Add the generated crack samples to the crack dataset for data augmentation, train the model, and perform crack segmentation based on the CRACK500 test set. The visualization of the generated images in the experiment is shown in Figure 9, and the comparison of the values of the three indicators is shown in Table 2.

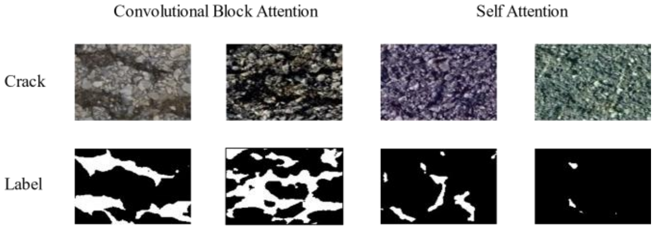


Figure 9: Comparison of image generation using two different attention mechanism for enhancement

Table 2: Comparison of Model Performance Obtained by Different Attention Mechanisms

Attention	mIoU/%	mAP/%	Acc/%
self attention	73.45	77.42	96.90
channel attention	74.51	78.90	96.99
spatial attention	75.91	81.51	97.02
convolutional clock-attention	76.32	82.51	97.05

According to Table 2, the method of using convolutional block attention has improved mIoU by 2.87%, mAP by 5.09%, and Acc by 0.15% compared to the method of using self attention. This indicates that the crack images generated by self attention have a significant difference from the real crack features, which leads to a decrease in the segmentation performance of the model after participating in model training. The method of using spatial attention increased mIoU by 1.4%, mAP by 2.61%, and Acc by 0.03% compared to the method of using channel attention, indicating that spatial attention can better capture crack features compared to channel attention. The experimental results show that CBIDM provides more clear crack features for the segmentation network while satisfying the authenticity of cracks, and improves the learning ability of the segmentation network for crack feature details.

5.2 Comparative experimental results analysis

The above experimental results analysis can fully demonstrate the improvement of the segmentation performance of the CBIDM model before and after crack data enhancement. In order to further verify the progressiveness nature of the proposed CBIDM model, a comparative experiment is carried out with the existing four crack segmentation methods based on depth learning. The comparison of the results of five segmentation experiments based on the CRACK500 dataset is shown in Table 3. In order to more intuitively demonstrate the segmentation effects of the five models, Figure 10 shows the visualization results of the segmentation of six crack samples using the five methods.

Table 3: Comparison of segmentation performance of different models on the CRACK500 dataset

Model	mIoU/%	mAP/%	Acc/%
U-Net	73.69	77.67	96.93
PSPNet	60.97	64.22	93.43
DeepLabV3+	71.19	74.04	93.18
HRNet	71.32	73.92	93.96
CBIDM	**76.32**	**82.51**	**97.05**

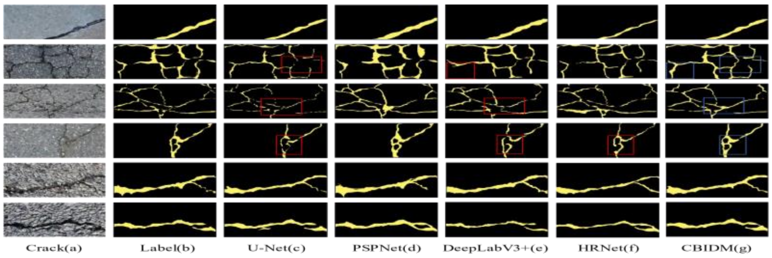


Figure 10: Visual comparison of segmentation results between different models

According to the data in Table 3, compared to the four methods of U-Net, PSPNet, DeepLabV3+, and HRNet, CBIDM achieves the best results in the three types of crack segmentation. From the visualization results in Figure 10, it can be observed that in the crack samples in the second and third images, U-Net and DeepLabV3+both show obvious neglect in defining the edges of small cracks. PSPNet and HRNet divide the background into crack areas on a large scale, while CBIDM has a keen perception ability for small cracks. In the fourth crack sample, the low contrast between the road background and the cracks increases the difficulty of crack segmentation. U-Net, DeepLabV3+, and HRNet all exhibit significant neglect of cracks. PSPNet is unable to accurately segment at the junction of coarse and fine cracks, while the segmentation results of CBIDM are more coherent. Based on the above experimental results analysis, it can be concluded that the CBIDM enhanced segmentation model has higher sensitivity for defining the edges of small cracks and the parts with larger changes in crack thickness. It can effectively reduce the phenomenon of missed cuts in the segmentation model and reduce the probability of fracture in the segmentation results.

6. Conclusion

In response to the problems of difficulty in collecting crack data samples, time-consuming and laborious manual labeling, resulting in insufficient training datasets and low model segmentation accuracy, this paper proposes a CBIDM model that utilizes convolutional block attention mechanism to generate and enhance crack images. By controlling variables to enhance crack image labels, the performance of the crack segmentation model is improved. After data augmentation on the publicly available dataset CRACK500 in a 1:1 ratio, the mIoU segmented using the U-Net segmentation model reached 76.32%, with an accuracy of 97.05%, both higher than the segmentation metrics of PSPNet, DeepLabV3+, and HRNet. This indicates that training the CBIDM enhanced crack segmentation dataset effectively improves the sensitivity of the segmentation model to the intersection of small cracks and coarse and fine cracks.

In the future, research on enhancing crack image segmentation data will mainly focus on the following aspects: improving the efficiency of diffusion reverse sampling; The crack images generated by improving the diffusion algorithm are closer to real cracks, and the generation effect is more stable; Improve the feature extraction ability of the algorithm and enhance the model's attention to small cracks at the edges of the image; Using a diffusion model to synchronously generate crack images and corresponding segmentation labels.

References

[1] Ministry of Transport of the People's Republic of China The 14th Five Year Plan for the Development

of Modern Comprehensive Transportation System [R] Railway Technical Supervision, 2022, 50 (2): 9-23, 27.

[2] Kheradmandi N, Mehranfar V. A critical review and comparative study on image segmentation-based techniques for pavement crack detection [J]. Construction and Building Materials, 2022, 321: 126162.

[3] Song Zegang, Liu Yanli, Zhang Changxing. Application and Development Review of Bridge Crack Detection Based on Machine Vision [J]. Science and Technology and Engineering, 2023, 23 (30): 12796-12805.

[4] Iyer S, Sinha S K. A robust approach for automatic detection and segmentation of cracks in underground pipeline images [J]. Image and Vision Computing, 2005, 23(10): 921-933.

[5] Yiyang Z. The design of glass crack detection system based on image preprocessing technology[C]// Proc of the 7th joint international information technology and artificial intelligence conference. Piscataway, NJ: IEEE Press, 2014: 39-42.

[6] Hoang N D, Huynh T C, Tran X L, et al. A novel approach for detection of pavement crack and sealed crack using image processing and salp swarm algorithm optimized machine learning[J]. Advances in Civil Engineering, 2022, 2022.

[7] Akagic A, Buza E, Omanovic S, et al. Pavement crack detection using Otsu thresholding for image segmentation[C]//Proc of the 41st international convention on information and communication technology, electronics and microelectronics (MIPRO). Piscataway, NJ: IEEE, 2018: 1092-1097.

[8] Hertz J A. Introduction to the theory of neural computation [M]. Crc Press, 2018.

[9] Cheng H D, Shi X J, Glazier C. Real-time image thresholding based on sample space reduction and interpolation approach [J]. Journal of computing in civil engineering, 2003, 17(4): 264-272.

[10] Xu G, Ma J, Liu F, et al. Automatic recognition of pavement surface crack based on BP neural network[C]//Proc of International conference on computer and electrical engineering. Piscataway, NJ: IEEE Press, 2008: 19-22.

[11] Jenkins M D, Carr T A, Iglesias M I, et al. A deep convolutional neural network for semantic pixel-wise segmentation of road and pavement surface cracks[C]//Proc of the 26th European signal processing conference (EUSIPCO). Piscataway, NJ: IEEE Press, 2018: 2120-2124.

[12] Gu Z, Chen H, Xu Z. Diffusioninst: Diffusion model for instance segmentation[C]// International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway, NJ: IEEE Press, 2024: 2730-2734.

[13] Nguyen Q, Vu T, Tran A, et al. Dataset diffusion: Diffusion-based synthetic data generation for pixel-level semantic segmentation [J]. Advances in Neural Information Systems, 2024, 36.

[14] Fang H, Han B, Zhang S, et al. Data augmentation for object detection via controllable diffusion models[C]//Proc of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2024: 1257-1266.

[15] Trabucco B, Doherty K, Gurinas M, et al. Effective data augmentation with diffusion models[J]. arxiv preprint arxiv:2302.07944, 2023.

[16] Yu X, Li G, Lou W, et al. Diffusion-based data augmentation for nuclei image segmentation[C]// Proc of International Conference on Medical Image Computing and Computer-Assisted Intervention. Cham: Springer Nature Switzerland, 2023: 592-602.

[17] Song J, Meng C, Ermon S. Denoising diffusion implicit models[J]. arxiv preprint arxiv:2010. 02502, 2020.

[18] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proc of the European conference on computer vision (ECCV). 2018: 3-19.

[19] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//Proc of the 18th Medical image computing and computer-assisted intervention–MICCAI international conference. Munich, Germany: Springer International, 2015: 234-241.

[20] Yang F, Zhang L, Yu S, et al. Feature pyramid and hierarchical boosting network for pavement crack detection [J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 21(4): 1525-1535.