Automation in Construction 140 (2022) 104346

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Crack detection algorithm for concrete structures based on super-resolution reconstruction and segmentation network

Chao Xiang [a], Wei Wang [a,b], Lu Deng [a,b,*], Peng Shi [c], Xuan Kong [a,b]

[a] College of Civil Engineering, Hunan Univ, Changsha, China
[b] Key Laboratory for Damage Diagnosis of Engineering Structures of Hunan Province, Hunan Univ, Changsha, China
[c] Hunan ZhongDeng Technology Co., Ltd, Changsha, China

ARTICLE INFO

Keywords:
Deep learning
Unmanned aerial vehicles
Super-resolution reconstruction
Semantic segmentation
Automatic crack detection
Magnification factor

ABSTRACT

Crack images collected from civil infrastructures through unmanned aerial vehicles suffer from motion blur and insufficient resolution, which reduces the accuracy of microcrack detection. Therefore, an automatic microcrack detection method based on super-resolution reconstruction (SRR) and semantic segmentation is proposed. Super-resolution (SR) images reconstructed by the proposed deep learning-based SRR model were input into the proposed semantic segmentation network for crack segmentation, and the length and width of cracks were measured through an improved medial axis transform approach. The accuracy of crack segmentation and feature quantification for SR images obtained using the deep learning-based SRR is significantly improved compared with low-resolution fuzzy images. The effects of three parameters on the results were analyzed. Compared with the Bicubic testset, the Intersection-over-Union of the SR testset is improved by 17% when a magnification factor of 4 is adopted. The results show that the proposed method achieves good performance in detecting concrete cracks.

1. Introduction

Computer vision (CV)-based techniques for structural damage detection and diagnosis have received much attention in the monitoring of civil infrastructures including highways, bridges, railways and tunnels [1]. Structural crack information provides an important basis for assessing the safety and durability of concrete structures, and it is of great significance for detecting cracks accurately [2]. Crack detection techniques based on CV and unmanned aerial vehicles (UAVs) are widely used in practical engineering because of the low cost, easy operation, non-contact and intuitive interpretation of the observed data [3,4]. With the increasing volume and complexity of the data collected from structures, traditional digital image processing (DIP) algorithms, including the threshold segmentation [5], edge detection [6], wavelet transforms [7], etc., may extract features excessively or incorrectly, which makes the subsequent data processing and analysis time-consuming, cumbersome, and even inaccurate [8]. Therefore, processing and analyzing the collected image data accurately and effectively has become the frontier and the focus of the research on structural health monitoring and detection.

With the vigorous development of deep learning, convolutional neural networks (CNNs) can be applied to automatically acquire the characteristics of images in the supervised learning process without prior knowledge. Compared with the characteristics extracted by the traditional DIP techniques, the characteristics learned by the CNN process can represent the texture features of images more accurately and robustly [9]. There have been many experimental studies on automatic crack detection and assessment based on the CNNs [10], which mainly included two aspects: (1) drawing the bounding boxes of crack regions [11,12], and (2) segmenting the cracks at the pixel level through semantic segmentation [13,14]. From the perspective of region-based deep CNNs, Cha et al. first introduced the deep CNN architecture for crack detection based on sliding windows [15], and subsequently introduced the faster regional convolution neural network (Faster RCNN) to improve the performance of the crack detection model [16], in which the high-quality crack images used for model training were collected under controlled conditions. Therefore, it is difficult for the trained model to test new images captured from other complex scenes

* Corresponding author at: Key Laboratory of Damage Diagnosis for Engineering Structures of Hunan Province, Hunan University, Changsha, Hunan 410082, China.

E-mail addresses: xiangchao@hnu.edu.cn (C. Xiang), wang_wei@hnu.edu.cn (W. Wang), denglu@hnu.edu.cn (L. Deng), shipeng@hnu.edu.cn (P. Shi), kongxuan@hnu.edu.cn (X. Kong).

https://doi.org/10.1016/j.autcon.2022.104346
Received 27 September 2021; Received in revised form 3 May 2022; Accepted 10 May 2022
Available online 26 May 2022

[17]. To overcome this shortcoming, the CNN models were then trained with various crack images containing complex background information [11,18], and the effects of noise [19] and light condition [20] on the segmentation accuracy were investigated. Actually, cracks usually appear as thin black bands with different angles and directions. Region proposal-based target detection algorithms can detect cracks well, but cannot provide accurate information about the direction and size of cracks. Therefore, some semantic segmentation frameworks were proposed for pixel-level identification of cracks on the surface of concrete structures, including the fully convolutional network (FCN) [21,22], the U-shaped network (U-Net) [23], the segmentation network (SegNet) [24] and their various variants [25].

Even though many of the deep learning-based techniques described above can be used to detect cracks successfully, there are still some critical technical issues that need to be addressed urgently [26]. In fact, the performance of CV-based crack detection models highly depends on the quality of crack images collected under different conditions [27]. UAVs are a common and efficient way to collect surface images of concrete structures [28]. However, UAVs will vibrate and cannot get too close to the target structure for the sake of safety during the image collection process [29], which may result in motion blur and insufficient resolution of the collected images. These problems may lead to the loss of image information and make it more difficult to detect cracks, resulting in a large number of thin cracks not being detected [27,30]. There exist many conventional image processing techniques for improving image quality, including the unsharp mask filtering, median filtering and histogram equalization [26]. However, these techniques improve the image quality based on the existing pixel points of the image rather than by enhancing the image resolution, which contributes little to the performance of crack detection models [31]. Super-resolution reconstruction is a new technology that can solve the problems of motion blur and insufficient resolution of images [32], and it can be based on three types of algorithms, including the interpolation algorithm, reconstruction algorithm, and machine learning algorithm [33]. These SRR algorithms all have their limitations. The interpolation-based algorithm simply performs operations on pixel points, and is prone to blurring images due to the loss of too many details. The reconstruction-based algorithm overcomes the difficulty with the interpolation-based algorithm in introducing prior knowledge, but it is not effective for the reconstruction of images with rich textures. The traditional machine learning-based algorithm can obtain more accurate results compared with the other two algorithms, but it is time-consuming to reconstruct and difficult to optimize the model [34].

Numerous research results have shown that the deep learning-based SRR can be used to address the above-mentioned technical obstacles in the automatic detection of structural cracks based on UAVs and CV [35]. The super-resolution convolutional neural network (SRCNN) was the first deep learning network proposed for SRR [36]. Subsequently, a series of SRR networks based on deep learning have been developed for improving the performance of SRR, including super-resolution generative adversarial network (SRGAN) [37], enhanced deep super-resolution (EDSR) network [38], residual channel attention network (RCAN) [39] and super-resolution feedback network (SRFBN) [40], etc. These deep learning-based SRR techniques have been successfully applied in various fields, including medical imaging [41,42], object detection [43] and face recognition [44]. However, the aspect ratio of the target object to be reconstructed in these studies is relatively small, while the aspect ratio of the crack structure is large, which presents a different challenge for processing crack images through the SRR networks. Few studies have used these techniques to improve the performance of crack detection [26,32,45,46]. Bae et al. compared the SR images reconstructed by the proposed SrcNet model with the low-resolution (LR) images, and the results showed that image SR can effectively improve the recall of detection, but there is a significant decrease in detection accuracy [26]. Sathya et al. concluded that SRR can significantly improve crack classification accuracy, but the extent of the effect on crack segmentation accuracy was not explored [32]. Kondo et al. [45] and Kim et al. [46] concluded that the crack segmentation accuracy was significantly improved with SRR for LR crack images, but the effect of SRR on the quantification of crack features was not considered. Although a general framework for SRR was demonstrated in the above work, the impact of various SRR networks and magnification factors on crack reconstruction had not been fully studied before.

The purpose of the study is to propose a method to improve the detection accuracy for thin cracks in fuzzy images based on the deep learning and semantic segmentation network. Firstly, a training dataset for the deep learning-based SRR was constructed, and the networks for SRR based on different deep learning algorithms were trained with the prepared dataset. The quality of reconstructed crack images obtained from the trained networks was initially evaluated with metrics of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM). Secondly, the proposed network for semantic segmentation, i.e. the CDU-Net, was trained with the original high-resolution (HR) crack dataset, and the trained model was used to identify the cracks of different testsets (namely, the original HR testset, the testset reconstructed by bicubic interpolation, and the testset based on deep learning SRR model), and the identification results were compared and analyzed. Thirdly, the skeletons of the crack segmentation maps were extracted and the crack features including length and width were quantified according to the improved medial axis transform (MAT) algorithm. Finally, the influence of different training datasets for SRR on the reconstruction effect was discussed, the segmentation results of the proposed CDU-Net were compared with the FCN and U-Net, and the segmentation accuracy of the crack images reconstructed with different magnification factors was analyzed.

The contents of this article are organized as follows. The flowchart of the proposed method is described in Section 2, including the process of SRR, the structure of the CDU-Net, and the quantification of crack features based on the improved MAT method. In Section 3, the implementation details and dataset preparation are presented, followed by the results of the experiments using different SRR algorithms and the comparative evaluation of the results through multiple metrics. The effects of three key parameters on the results are discussed in Section 4, followed by the conclusions in Section 5.

2. Proposed methodology

The flowchart of the proposed methodology is presented in Fig. 1. In the first step, the crack images, consisting of blurred or LR images, are reconstructed into HR images using the deep learning-based SRR model. The model learns the nonlinear mapping function between the LR and HR images from the training dataset, and then the new HR image corresponding to an LR image is reconstructed based on the learned mapping function. The second step is to perform pixel-level segmentation on the reconstructed crack images through the trained crack segmentation model, which is used to mark each pixel in the area of cracks. The third step is to quantify the crack features from the pixel-level segmentation results according to the improved MAT method. The details are illustrated in Sections 2.1 to 2.3.

2.1. Super-resolution reconstruction

In recent years, various deep learning networks for SRR of images have been proposed [33]. As shown in the left of Fig. 1, to train the SRR model, a training set needs to be constructed, in which the LR images are downsampled from a series of corresponding HR images by the image degradation model. Then, the SRR model is constructed by selecting an appropriate deep network, and optimal hyper-parameters (e.g., loss function, learning rate, etc.) and network parameters are continuously optimized based on the mentioned dataset to obtain the feature. Finally, new LR images are input into the trained SSR model, and the quality of the output SR images is evaluated. In the following sections, the deep

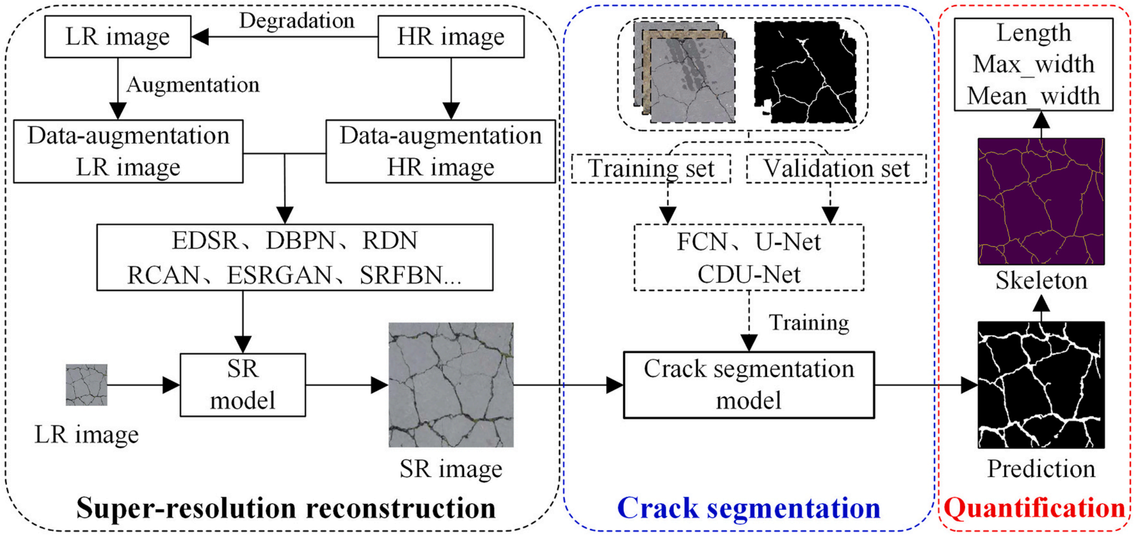


Fig. 1. Flowchart of the proposed methodology.

learning-based SRR networks used in the study are described, the training details are introduced, and the commonly used evaluation metrics for the quality of SRR images are presented.

2.1.1. Network architectures

Many improved SRR models have been proposed by scholars based on different network design strategies, including linear networks [36], residual learning networks [47], dense connection networks [48], recursive learning networks [33], and generative adversarial networks [49]. Linear networks, including SRCNN [36], fast super-resolution convolutional neural network (FSRCNN) and efficient sub-pixel convolutional network (ESPCN), have simpler structures, but cannot use all the information of image features to reconstruct new images. The residual learning networks, including super-resolution residual network (SRResNet), EDSR [38], residual dense network (RDN) [50] and enhanced residual network (ERN), can avoid the gradient degradation problem of deep neural networks and converge quickly. The dense connection networks (i.e., super-resolution dense network (SRDense-Net), RDN, and deep back-projection network (DBPN) [51]) are designed based on the dense connection strategy, which can effectively resolve the problem of vanishing gradient and reduce the model size without degrading the model performance due to the property of reusing features. Recursive learning strategies have also been introduced into some algorithms (e.g., SRFBN, deeply-recursive convolutional network

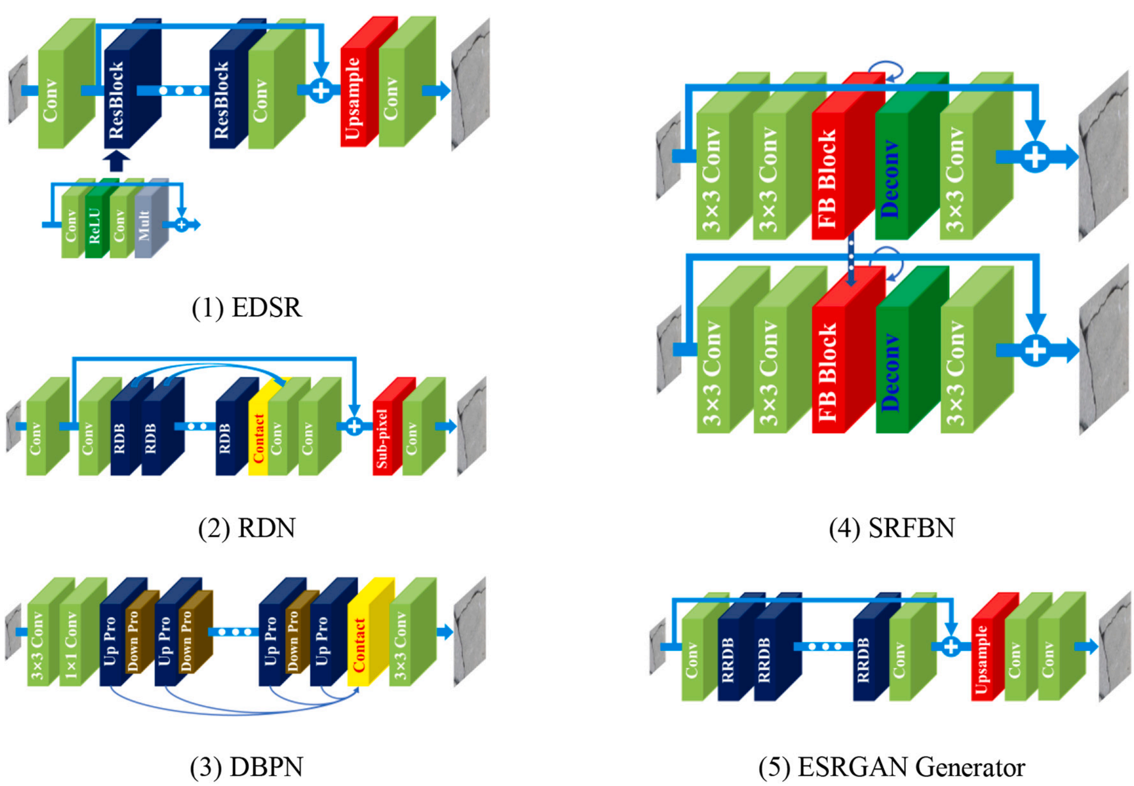


Fig. 2. Schematic depiction of SRR network architectures.

(DRCN), deep recursive residual network (DRRN), etc.) for model improvement [35]. The generative adversarial networks (GANs) [49], including SRGAN [37], enhanced super-resolution generative adversarial network (ESRGAN) [52] and super-resolution with feature discrimination (SRFeat), can also be used to train the SRR models to obtain high-quality images. In the present study, five representative networks designed based on each network strategy were selected to obtain SRR models and the architectures of these networks are shown in Fig. 2, in which Conv denotes the convolutional layer and Deconv denotes the deconvolutional layer.

EDSR was developed from the SRResNet by removing the batch normalization (BN) layer from the residual block (ResBlock) and the L1 loss function was adopted to optimize the network. Without the BN layer, EDSR can save about 40% of memory usage during the training process and can therefore stack more network layers and extract more features to achieve better performance with the same computational resources. The residual dense block (RDB) in RDN possesses the advantages of both the residual learning and dense connection, in which the input and output features of each layer are fused and reused and, therefore, can provide more clues for image reconstruction. The innovation of DBPN lies in the proposed up-down projection module (i.e., Up Pro and Down Pro), which can learn the feedback errors between the LR and HR images through a series of tightly connected up-down sampling layers. DBPN incorporated features of different resolutions with different depths for image reconstruction, which enables the network to obtain more information and helps promote the reconstruction performance. SRFBN was designed through a feedback block that uses recursive learning, in which the high-level information flows through the feedback blocks (FB Block) in a top-down manner to correct low-level features with more contextual information. ESRGAN was modified from SRGAN with two steps. The first step was to introduce the residual in residual dense block (RRDB) to improve the generator structure, by removing the BN layer and using residual scaling. The second step was to use the relativistic average GAN to determine whether one image is more realistic than the other, thus enhancing the performance of the discriminator.

2.1.2. Network training

During the training stage, the model parameters need to be updated based on the loss calculation between each input LR image and the corresponding HR image. The network parameters of the CNN-based model are optimized by the L1 loss function and the loss function is minimized by the Adam optimization algorithm. The network parameters of the GAN-based model are optimized by perceptual loss and adversarial loss, which can effectively improve the realism of the reconstructed images [52]. In the experiment, the maximum number of epochs, the batch size and the initial learning rate were set to be 100, 16, and 0.0001, respectively. The learning rate decay strategy was used for training networks to reduce the learning rate, which can be decreased by a factor of 2 after every 20 epochs.

2.1.3. Evaluation metrics

In the present study, both PSNR and SSIM were used to evaluate the reconstruction effect [33]. The PSNR shown in Eq. (1) is an index for measuring the similarity between two images, and a larger PSNR indicates a higher similarity between two images. The SSIM is an evaluation criterion for image quality, in which the brightness, contrast and structure of the image are considered to evaluate the similarity of two images. The SSIM is equal to 1 if the generated image is the same as the original one. The formulas for PSNR and SSIM are shown as follows:

$$PSNR = 10 \times lg \frac{255^2 \times W \times H \times C}{\sum_{i=1}^{W} \sum_{j=1}^{H} \sum_{z=1}^{C} [X(i,j) - Y(i,j)]^2} \tag{1}$$

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + K_1)(2\sigma_{xy} + K_2)}{\left(\mu_x^2 + \mu_y^2 + K_1\right)\left(\sigma_x^2 + \sigma_y^2 + K_2\right)} \tag{2}$$

In Eq. (1), W, H, and C denote the width, length, and channel number of the image, respectively; X denotes the SR image; and Y denotes the original image. In Eq. (2), μ_x and μ_y denote the mean values of image X and image Y, respectively; σ_x and σ_y denote the variance of image X and image Y, respectively; and σ_{xy} denotes the covariance of image X and image Y. Both K_1 and K_2 are constants adopted to avoid the denominator being zero.

2.2. Crack segmentation

2.2.1. Network architecture

Although the classical U-Net improves the accuracy of crack segmentation, the frequent pooling operations in the network result in the low-resolution feature map, leading to the loss of some image features and causing the missing detection of micro-cracks [53]. To overcome this shortcoming, the U-Net architecture is improved in three aspects. The first aspect is to replace the convolutional block in the encoding and decoding module of the classical U-Net with a recurrent residual convolutional (RRC) block that can capture the multi-scale features of crack images. The second aspect is to add a new dense atrous convolution (DAC) module to capture the deeper semantic features, which can retain more spatial information and improve the performance of crack segmentation. The third aspect is to adopt a loss function that combines the cross-entropy loss and dice coefficient loss to solve the problem of pixel sample imbalance during the training stage of the crack segmentation network. The new architecture of the context-encoding network for crack segmentation (CDU-Net), modified from the classical U-Net, is shown in Fig. 3. The network is composed of a feature encoder module, a DAC module, and a feature decoder module [54].

The role of the encoder module is to extract context information and semantic features from images. In the classical U-Net architecture, each block has a max-pooling layer and two convolution (Conv) layers. To extract more detailed information from crack images and to improve the segmentation of low-contrast regions (e.g., thin cracks) in various backgrounds, the pre-trained Resnet34 is introduced to replace the traditional block, and the shortcut mechanism is added to avoid gradient vanishing as the neural network deepens and to accelerate the convergence of the network. Besides, to improve the generalization ability of the network, the residual block in the original U-Net is optimized by adding a BN layer and rectified linear unit (ReLU) activation function before the convolutional layer. The modified residual blocks can deepen the network with fewer model parameters, obtain more abstract features of crack images and accelerate the training process.

To utilize the multi-scale feature maps of crack images, the DAC module is added between the feature encoder module and the feature decoder module to obtain the high-level semantic characteristics [54]. As shown in the middle of Fig. 3, the DAC module performs four atrous convolution operations on the same feature map, which can increase the receptive field of the feature map without sacrificing its resolution. The receptive field of each branch from top to bottom will be 3, 5, 7, and 9, respectively. The DAC module can learn the crack information of different scales by combining atrous convolutions with different atrous rates.

The function of the feature decoder module is to recover the high-level semantic features extracted from the previous modules into the HR image features. Commonly-used feature decoder operations include deconvolution and upsampling [54]. The feature decoder module proposed in the present study consists of five convolution blocks, each containing one upsampling layer, one 2 × 2 transposed convolution (Trans Conv) layer, and a recursive residual convolution block with a 1 × 1 convolution layer. The transposed convolutional layers can use adaptive mapping to recover features with more detailed information.

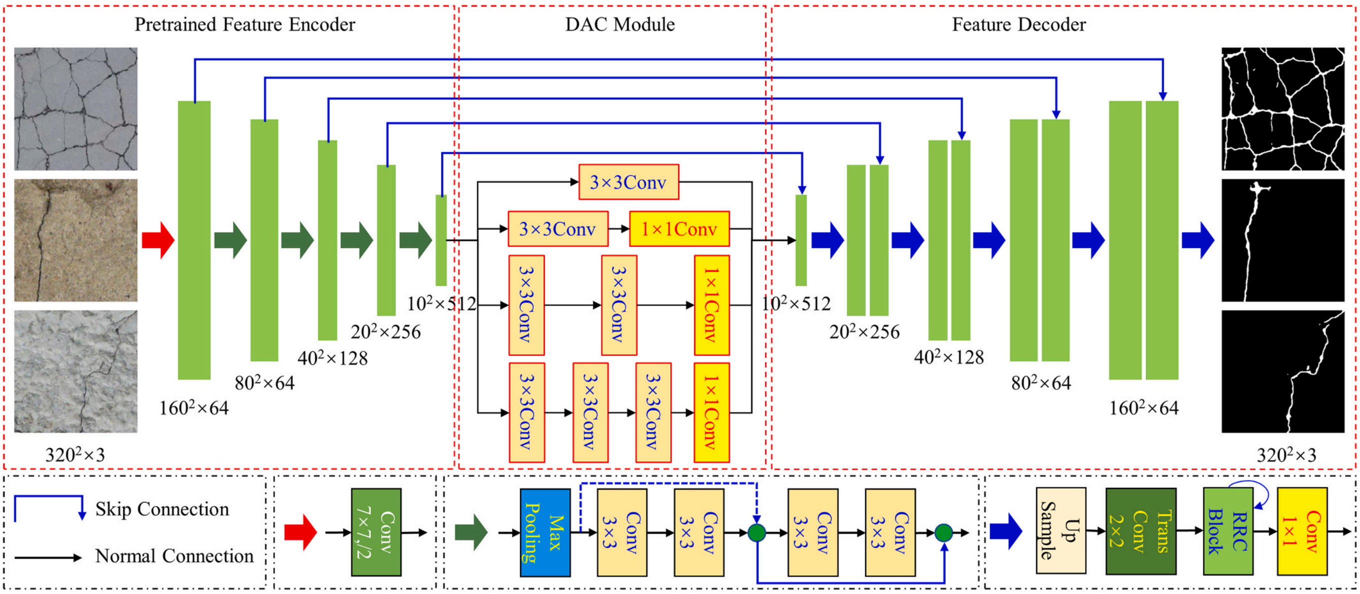


Fig. 3. Schematic diagram of the CDU-Net architecture.

To solve the problems of local information loss and the decrease of feature map resolution caused by the max-pooling and convolution operation, the crack features obtained in the encoding and decoding processes are fused through skip connection.

2.2.2. Training parameters

The semantic segmentation of crack images is a binary classification problem [55] and the binary cross-entropy (BCE) loss function is a commonly-used loss function:

$$L_{\mathrm{BCE}} = -\frac{1}{n}\sum[y_i lgp_i + (1-y_i)lg(1-p_i)] \tag{3}$$

where n is the total number of image pixels, y_i is the label value of the i-th pixel, and p_i is the predicted probability of the i-th pixel. Since the proportion of crack pixels in the detected images is very low, the BCE loss function cannot perform well in learning and recognizing cracks and tends to treat cracks as background information. Compared with the BCE loss function, the dice coefficient loss function can solve the problem of imbalance between positive and negative samples as shown in Eq. (4) [23].

$$L_{\mathrm{Dice}} = 1 - \frac{\sum p_i y_i + \varepsilon}{\sum(p_i + y_i) + \varepsilon} - \frac{\sum(1-p_i)(1-y_i) + \varepsilon}{\sum(2 - p_i - y_i) + \varepsilon} \tag{4}$$

where the meanings of the y_i and p_i are the same as those in Eq. (3), and ε is a constant. The reason for setting ε is that the gradient will vary greatly when y_i and p_i are too small, and will make the training more volatile and difficult. The combination of the BCE loss and dice coefficient loss can effectively deal with the problem of imbalanced positive and negative samples and make the model training process more stable. The combined loss function is calculated as follows [54]:

$$L_{\mathrm{Total}} = \lambda_1 L_{\mathrm{BCE}} + \lambda_2 L_{\mathrm{Dice}} \tag{5}$$

where L_{Total} is the total loss; L_{BCE} is the BCE loss; L_{Dice} is the dice coefficient loss; λ_1 and λ_2 are the weighting factors to balance the BCE loss and dice coefficient loss, respectively. The Adam optimization algorithm was used to accelerate the convergence of model training. The values of the mini-batch size, initial learning rate, decay coefficient and training epoch were set to 4, 0.001, 0.9 to 0.999, and 200, respectively.

2.2.3. Evaluation metrics

The crack semantic segmentation can provide pixel-level information such as coordinates and intensity of the crack region. Four commonly-used evaluation metrics, including precision, recall, F1-score and Intersection-over-Union (IoU), were adopted to evaluate the accuracy of obtained results [55]. The evaluation metrics are defined as:

$$Precision = TP/(TP + FP) \tag{6}$$

$$Recall = TP/(TP + FN) \tag{7}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{8}$$

$$IoU = TP/(TP + FN + FP) \tag{9}$$

where TP is the number of pixels of true cracks that have been detected correctly as cracks, FP is the number of pixels of non-cracks that are mistaken as cracks, and FN is the number of pixels of true cracks that are mistaken as non-cracks. It should be noted that the IoU of the foreground (cracks) was computed in this work. Calculating the IoU of the background cannot accurately reflect the effect of crack detection because more than 90% of the pixels in crack images are background and the proportion of the foreground is very small.

2.3. Quantitative evaluation of crack features at the pixel level

Obtaining the size of structural cracks is of great importance to accurately assess the state of structures and determine the maintenance schedule [21]. Morphological operations are commonly used to extract morphological characteristics of cracks and to reduce noises. The information about the crack skeleton can be extracted through the modified medial axis transform algorithm. Based on the obtained skeleton information, morphological characteristics of cracks (namely, length, width and area) can be obtained [56].

Due to the irregular shape of the cracks, the extracted crack skeleton is not a simple straight line [21]. Nevertheless, each crack can be divided into a series of curved segments based on the adaptive segmentation and the length of each curved segment can be calculated as the Euclidean distance between two endpoints [57]. Hence, the total length of the whole crack curve can be computed by accumulating the lengths of all segments, as defined in Eq. (10).

$$length = \sum_{i}^{n} \sqrt{(x_{i2} - x_{i1})^2 + (y_{i2} - y_{i1})^2} \tag{10}$$

where n denotes the number of segments of the crack, (x_{i1}, y_{i1}) and (x_{i2}, y_{i2}) denote the starting and ending points of the i-th segment of the crack, respectively.

The minimum distance l_d of each point on the skeleton to the boundary point can be obtained from the extracted skeleton information, and the maximum width of the crack can be calculated from Eq. (11). Based on the segmentation results, the area of the crack can be derived from the number of crack pixel points, and the mean width can be calculated using Eq. (12).

$$max\text{-}width = 2 \times max(l_d) \tag{11}$$

$$mean\text{-}width = \mathrm{sum}(pixel = crack)/length \tag{12}$$

To better evaluate the accuracy of the prediction and the effect of SRR on crack images, the absolute error and the relative error rate between the prediction and the true value are used to evaluate the performance of the algorithm. The absolute error (AE) is calculated as:

$$AE = S_p - S_g \tag{13}$$

The formula for the relative error rate (RER) is:

$$RER = \left(S_p - S_g\right)/S_g \times 100\% \tag{14}$$

where S_p is the predicted value and S_g is the true value.

3. Experimental results

3.1. Datasets and experiment setup

3.1.1. Datasets

In this study, the dataset used for training the SRR model consists of two categories of images, the DIV2K dataset (natural scene) [58] and the crack dataset (mainly concrete structures). The DIV2K dataset consists of 1000 natural scene images with a resolution of 1920 × 1080 pixels. One thousand sub-images with a resolution of 480 × 480 pixels with clean texture and clear edge were taken from the DIV2K dataset. The crack dataset consists of 1000 crack images with a resolution of 480 × 480 pixels collected from the surface of concrete structures. To enhance the representation and generation ability of the SRR model, the following operations were performed to augment the training set: (1) flip images horizontally and vertically, (2) rotate images 180 degrees, and (3) scale images by the ratios of 0.6, 0.7, 0.8 and 0.9. After augmentation, the number of images in the training set was increased by 30 times, resulting in an HR image set. The LR image dataset was obtained by image degradation on the HR image dataset [59,60], and the image degradation model can be expressed as:

$$g = (f \otimes h)\downarrow_s^{bicubic} + \eta \tag{15}$$

where g denotes an LR image, f denotes an HR image, and h denotes the point spread function under the uniform linear motion module, $\otimes$ indicates the convolution operation, ↓ is the downsampling operation, s is the magnification factor, *bicubic* is the interpolation algorithm, and η is the additive gaussian white noise.

To increase the diversity of the training set as well as to improve the training efficiency, a set of sub-images of size $l_{sub} \times l_{sub}$ pixels were randomly cropped from LR images, and the HR images at the corresponding positions were also cropped into sub-images of size $sl_{sub} \times sl_{sub}$. These LR and HR sub-images were paired and used as training samples.

The dataset, named Crack776, used for crack semantic segmentation was selected from the literature [21]. In the present study, the Crack776 dataset was defined as the HR crack dataset, in which the image resolution was uniformly adjusted to 320 × 320 pixels. The HR crack dataset was divided into the training set, validation set and test set, with a percentage of 70%, 10%, and 20%, respectively. The mean value of the crack widths in the HR images ranges from 4 to 8 pixels, and the downsampling factor was set to 4 to ensure sufficient information about the overall structure of the cracks in the LR images, while simulating most unfavorable conditions in practice. The LR testset with the resolution of 80 × 80 can be obtained through downsampling the test set of the HR crack dataset (HR testset) by a factor of 4 according to the degradation model described in Eq. (15). All SR testsets were generated from the LR testset using different SRR models. To highlight the reconstruction effect of crack images based on the SRR deep learning model, the results were compared at the same image size, and a Bicubic testset of the same size as the SR testset can be obtained by interpolating the LR testset using the bicubic method.

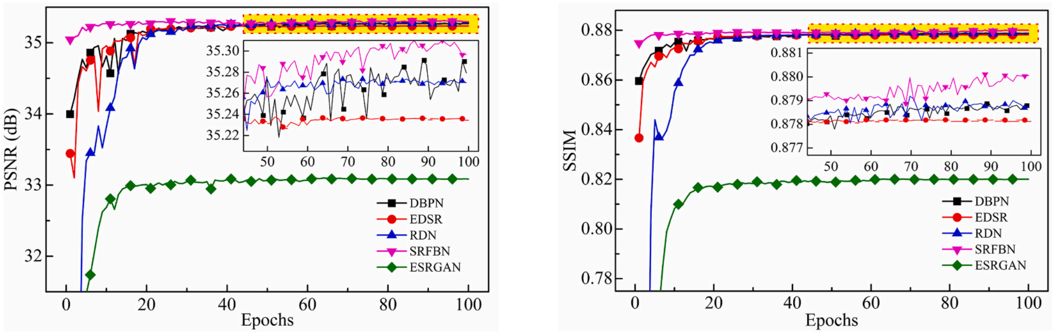
3.1.2. Experimental setup

The main purpose of using SRR to reconstruct crack images is to obtain better SR images. The effects of five SRR models for crack reconstruction, including the EDSR, RDN, DBPN, ESRGAN, and SRFBN, were compared using the same procedure. First, the five different SRR networks were well trained using the same training dataset, and the trained models were then used to perform SR reconstruction of new crack images in the LR testset, and the quality of the reconstructed images was evaluated with PSNR and SSIM. Second, the proposed semantic segmentation network was trained with the training and validation sets from the original HR crack dataset, and the trained segmentation model was used to semantically segment the testsets including the HR, Bicubic and SR testsets. The precision, recall, F1-score and IoU were used to evaluate the results of semantic segmentation. Third, the crack features of the segmentation results of all testsets under consideration were quantified, and the reconstruction effects and the segmentation accuracy of different crack testsets were evaluated in terms of the length, maximum width and mean width of cracks.

3.2. SRR for crack images

The SRR training dataset was fed into the EDSR, RDN, DBPN, ESRGAN, and SRFBN for training, and the performance of the corresponding trained model was evaluated using the LR testset. Convergence of the model is achieved when the loss reaches a minimum and converges to a constant. Fig. 4 shows the convergence curves of PSNR and SSIM during the training process of the five networks. It can be seen from Fig. 4 that the tendencies of PSNR and SSIM are similar as the epoch increases, and each model tends to converge when the epoch reaches 40. Table 1 shows the comparisons of PSNR and SSIM of the reconstructed images obtained using the six methods and the parameters required. It can be seen from Table 1 and Fig. 4 that the PSNR and SSIM of all SRR methods based on deep learning are larger than those of the Bicubic method. Larger values of PSNR and SSIM suggest a better image reconstruction effect. It can also be seen from Table 1 that the PSNR and SSIM of the CNN-based SRR methods are significantly larger than those of the GAN-based SRR method. The main difference between the CNN-based SRR method and the GAN-based SRR method is the loss function, where the GAN-based SRR method used the perceptual loss to better reconstruct details of high frequency, while the CNN-based SRR method used the L1 loss. Compared with these algorithms, SRFBN has the best reconstruction results and requires the least number of parameters.

Fig. 5 shows the original HR images and the corresponding reconstructed images with different methods. It can be seen from Fig. 5 that the line texture reconstructed by the Bicubic method completely deviates from that of the real HR images, while the edge and texture details of the images reconstructed by deep learning-based methods are clearer and closer to the original images. Besides, the CNN-based and GAN-based reconstructed images are sharper than those of the Bicubic method. It should be noted that the high-frequency details of the CNN-based images are still insufficient and dense textures appear to be very



(a) PSNR (dB) (b)SSIM

Fig. 4. Variation of the PSNR and SSIM with the increase of the number of epochs.

Table 1
Metrics of different methods on the LR testset.

Metrics	Bicubic	EDSR	RDN	DBPN	ESRGAN	SRFBN
PSNR (dB)	30.24	35.23	35.27	35.29	33.10	35.30
SSIM (%)	77.51	87.81	87.84	87.89	82.01	87.93
Number of Parameters	/	43,089,947	22,271,107	10,426,358	14,499,401	3,631,478

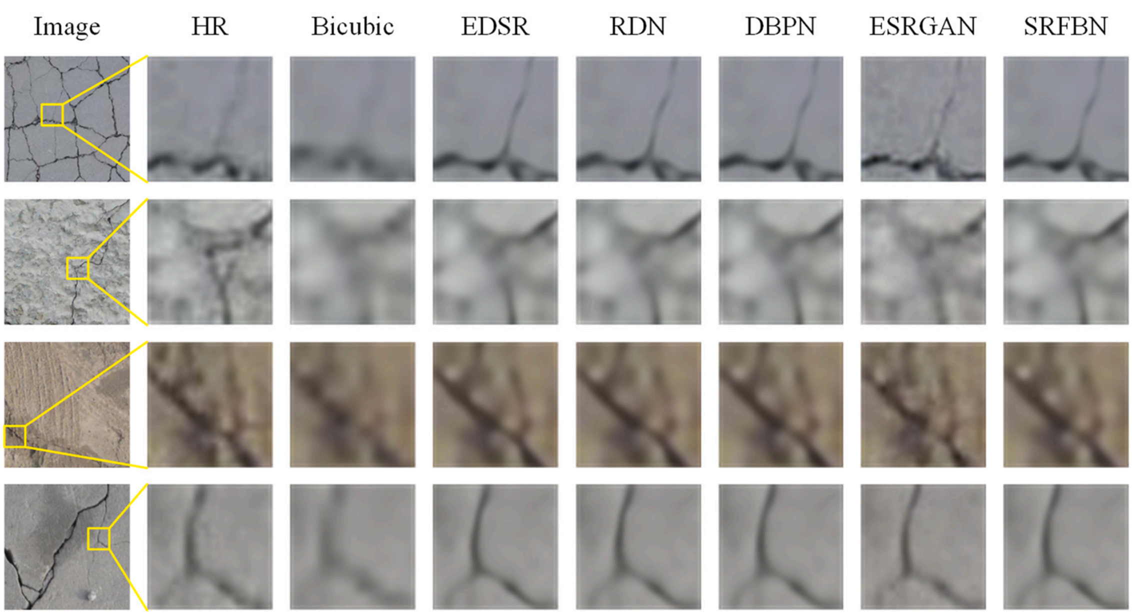


Fig. 5. Comparison of the visual effects of the reconstructed images based on the six methods.

smooth, while the generated textures of the GAN-based images are still quite different from the reference images although they look more realistic.

3.3. Semantic segmentation of reconstructed crack images

The proposed CDU-Net was used to evaluate the quality of the reconstructed crack images. The training set and validation set in the HR crack dataset were adopted to train the CDU-Net. The variation of training loss and IoU of the validation set with the increasing number of epochs during the training process are shown in Fig. 6. It can be seen from Fig. 6 that the training loss decreases drastically at first and then gradually stabilizes to about 1 and that the IoU of the validation set increases drastically at the beginning and finally converges to about 0.74.

To verify the effectiveness of atrous convolution on improving the

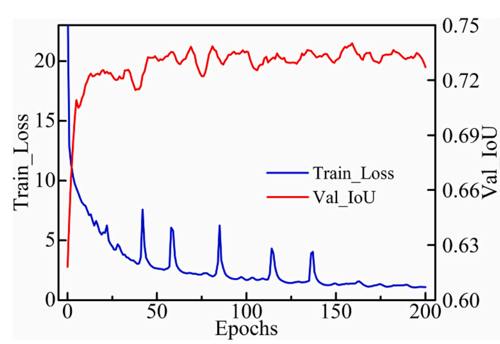


Fig. 6. Variation of training loss (Train_Loss) and IoU of the validation set (Val_IoU) with the increasing number of epochs.

receptive field and learning multi-scale features, experiments were conducted on the segmentation performance of three different networks. N1 is the network with the DAC module used in the study, N2 is the network with all dilation rates in the DAC module set to 1, and N3 is the network without the DAC module. All three networks have the same parameters except for different settings at the DAC module. The three networks were trained with the same training parameters, and the training models are tested on the HR test set. The segmentation results are shown in Table 2, from which it can be seen that the N1 model has the best segmentation results and the N3 model has the worst segmentation results. The results show that although the size of the input feature map is small, the DAC module enables the network to extract features from different receptive fields, thus increasing the width of the network, enriching its information, and improving its performance.

Crack segmentation evaluation metrics were used to evaluate the segmentation results of all testsets reconstructed with the Bicubic method and five SRR methods based on deep learning. Table 3 shows the segmentation results of the HR testset, Bicubic testset, and SR testsets obtained by different algorithms. As can be seen from Table 3, the HR testset achieves the highest precision (84.51%), recall (85.22%), F1-score (84.86%), and IoU (73.57%). It should be noted that the FCN proposed by Yang et al. [21] used the same dataset, and their achieved precision, recall and F1-score were 82%, 79% and 80%, respectively, which are all lower than those obtained by the proposed CDU-Net. The Bicubic testset achieves the worst results and all metrics are at least 15% less than that achieved by the HR testset. The values of the four metrics of the SR testsets are very close and also much larger than those of the Bicubic testset. Compared with the results of the Bicubic testset, the F1-score and IoU of the SR testsets are improved on average by 13% and 17%, respectively.

To visualize the difference between the crack images reconstructed by different algorithms, the segmentation results of some cracks reconstructed by different algorithms are shown in Fig. 7, in which the original labels are also included for comparison. It can be seen from Fig. 7 that the proposed segmentation network can achieve more accurate segmentation in the HR testset, and the detected cracks match the original labels best in terms of the overall structure and details of cracks. The thin cracks reconstructed by the Bicubic method are difficult to be detected, and the identified cracks are significantly wider than the ground truth labels. By contrast, most of thin cracks in the SR testsets can be well detected, and the accuracy of segmenting some thin cracks is much better than that of the HR testset. This may be because the edges of some thin cracks in the HR testset are blurred, while edge sharpening is implemented in the SRR algorithm. The detection results of different SR testsets are almost the same in the segmentation accuracy and are much better than those from the Bicubic testset.

3.4. Quantitative analysis results of crack features

In the practice of crack inspection, the length and width information of cracks is usually desired. Hence, the pixel-level segmentation results of different reconstructed datasets were calculated based on the method mentioned in Section 2.3.

The skeleton extraction operation was performed on the segmentation results of each testset and the ground truth label. The semantic segmentation results of the testsets reconstructed by different deep learning-based SRR algorithms do not vary much. Therefore, two representative semantic segmentation results (i.e., SRFBN, ESRGAN) were selected for comparison. Fig. 8 shows the skeleton extraction results of the HR testset, the testsets reconstructed by the selected two methods, the testset obtained by bicubic interpolation, and the real labels, in which the L, MW and AW represent the length, maximum width and average width of the cracks, respectively. It can be seen from Fig. 8 that the crack skeleton of the HR testset is consistent with the real skeleton, which proves the effectiveness of the proposed segmentation network and skeleton extraction algorithm. Due to the poor segmentation results of the Bicubic testset, the extracted skeleton obtained from the Bicubic testset differs greatly from the real skeleton, and the predicted crack widths are all significantly larger than the true values. The crack skeletons obtained from the SR testset match well with that obtained from the HR testset, indicating that the deep learning-based SRR algorithm performs well in terms of reconstruction.

Fig. 9 shows the histogram of the quantification error of the crack features, which reflects the difference between the predicted results and the true values of the label. It can be seen from Fig. 9 that the AEs of crack length, maximum width, and mean width in the segmentation results of the HR testset are 33.3 pixels, 2.2 pixels, and 1.0 pixels, respectively, and that the RERs of crack length, maximum width and mean width are about 6%, 10%, and 7%, respectively. It can be seen from Fig. 9 that the segmentation results of the Bicubic testset are very poor. The AEs of crack length, maximum width, and mean width of the Bicubic testset are 96.1 pixels, 3.8 pixels, and 2.0 pixels, respectively, and the RERs of the three features are around 13%, 19%, and 24%, respectively. The errors of segmentation results achieved from each SR testset are similar, and the average AEs of crack length, maximum width and mean width of all SR testsets are reduced by 59.1 pixels, 1.7 pixels and 0.9 pixels, respectively, compared to that obtained from the Bicubic testset.

Table 2
Segmentation results for the different nets.

Metrics (%)	N1	N2	N3
F1-score	84.86	83.83	82.38
IoU	73.57	72.16	70.44

4. Discussion

Since the effects of different SRR algorithms on the results are almost the same, only SRFBN was chosen to demonstrate the robustness of the proposed method. Three aspects were discussed as follows: (1) The effects of training sets on the reconstruction effect and semantic segmentation results were studied; (2) The performance of the proposed network and the commonly used semantic segmentation networks were compared; (3) The effects of magnification factors used in reconstructing images on the accuracy of crack semantic segmentation were investigated.

4.1. Effect of the training set on SRR

Adopting a dataset consisting of images captured from different scenes for model training is the key to obtaining high-quality images of super-resolution reconstructed cracks [32]. Especially, datasets including more realistic images with different textures and geometric features will contribute to improving the reconstruction accuracy of local cracks [61]. To investigate the effects of different datasets on the SRR of crack images, SRFBN was trained using three training sets separately, including Dataset A with images of natural scenes (DIV2K) only, Dataset B with crack images of concrete structures only and Dataset C consisting of both natural scene images and concrete crack images. The three datasets have the same number of images, where the ratio of natural scene images to concrete crack images in Dataset C is 1:1. After training SRFBN with the three datasets separately, the LR testset was fed into the corresponding SRFBN model trained with the three datasets to obtain three SR testsets. The CDU-Net was selected to evaluate the three SR testsets at the pixel level, and the reconstruction and segmentation results evaluated with different metrics are shown in Table 4. It can be seen from Table 4 that the quality of the reconstructed crack images is affected by the image types in the training sets, and Dataset C achieves the best reconstruction results. It can also be seen from Table 4 that Dataset A achieves better results than Dataset B as the former is collected from a variety of real scenes containing rich texture

Table 3
Evaluation metrics of segmentation results of the testset reconstructed by different algorithms.

Metrics (%)	HR	Bicubic	EDSR	RDN	DBPN	ESRGAN	SRFBN
Precision	84.51	68.52	83.69	83.53	83.45	**83.99**	83.59
Recall	85.22	70.58	81.68	81.90	81.92	80.74	**81.95**
F1-score	84.86	69.54	82.67	82.71	82.68	82.33	**82.76**
IoU	73.57	53.30	70.46	70.51	70.47	69.97	**70.59**

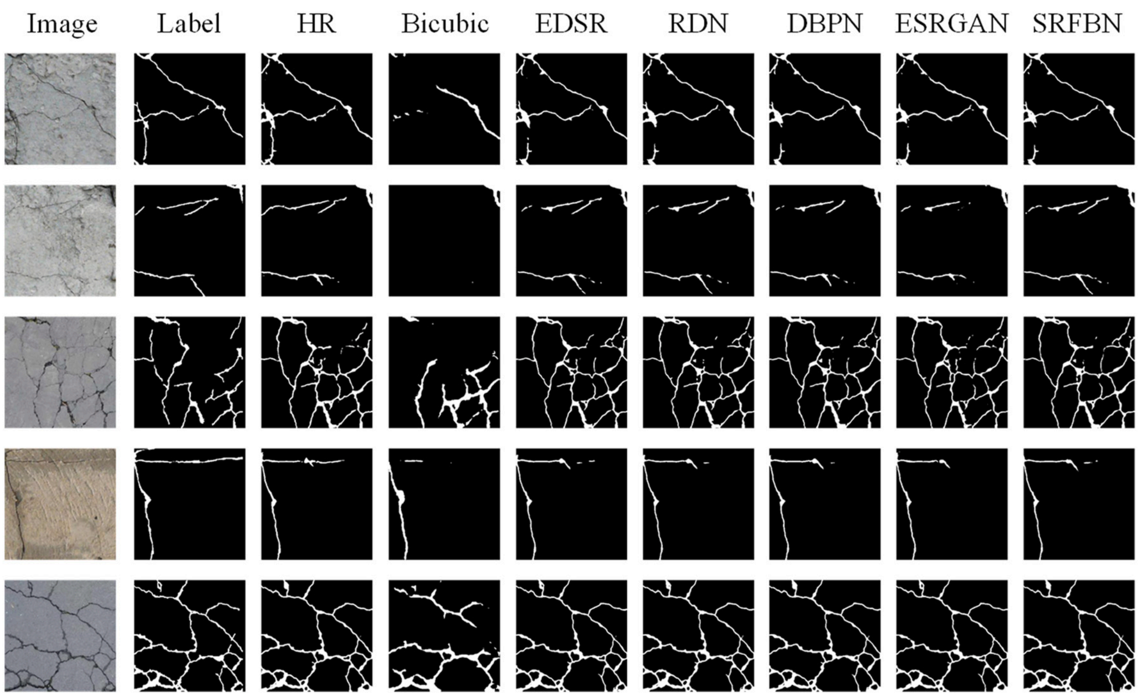


Fig. 7. Qualitative comparison of segmentation results of various testsets.

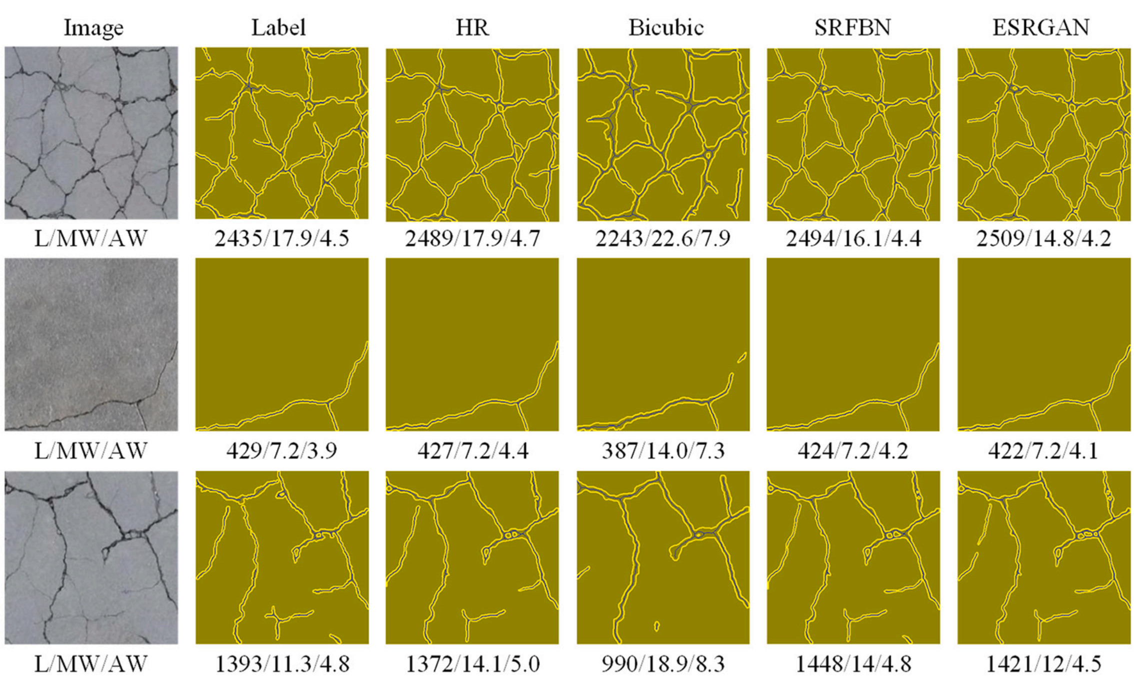


Fig. 8. The skeleton extraction results of some testsets.

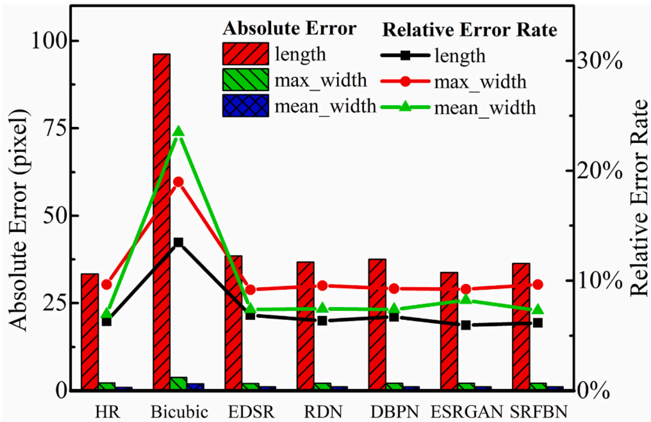


Fig. 9. Histogram of quantified error of crack features for all testsets.

Table 4
Reconstruction and segmentation results evaluated with different metrics.

Metrics	Dataset A	Dataset B	Dataset C
PSNR	35.12	34.08	**35.30**
SSIM (%)	87.59	85.11	**87.93**
Precision (%)	82.95	81.52	**83.59**
Recall (%)	81.43	77.90	**81.59**
F1-score (%)	82.18	79.67	**82.76**
IoU (%)	69.76	66.21	**70.59**

features. Although Dataset B is similar to the test set, only the crack region in Dataset B contains fewer distinct texture features and edge information, so the reconstruction is less effective. Dataset C contains both crack images of the same type as the test set and natural scene images with richer texture information, which makes the learning of the network more relevant to crack and leads to the best reconstruction results of the trained model.

4.2. Effect of the adopted semantic segmentation network on SSR

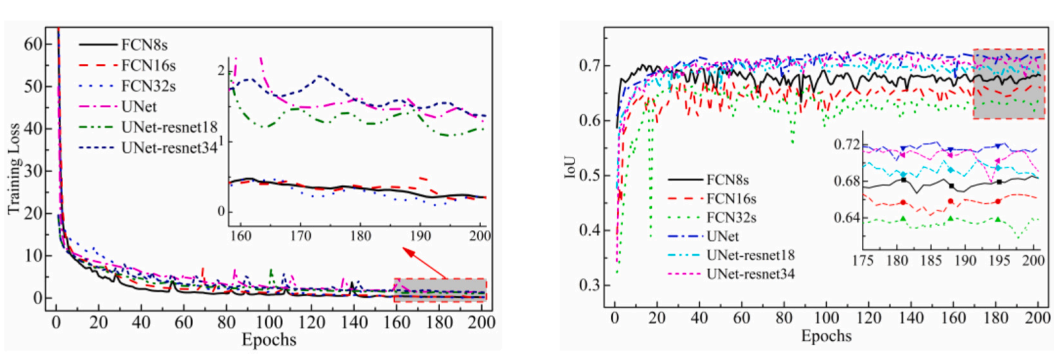
To investigate the effect of the semantic segmentation networks on the results, six other networks, i.e., FCN-8s, FCN-16s, FCN-32s, U-Net, UNet-resnet18, and UNet-resnet34, were trained with the training set and validation set in the HR crack dataset, and then they were tested on the HR testset, Bicubic testset, and SR testset, respectively. Fig. 10 shows the variation of the training loss and the IoU on the validation set with the increasing number of epochs when the six networks were trained on the HR crack dataset.

The segmentation results of seven networks on the three testsets are shown in Table 5. It can be observed from Table 5 that the network proposed in the present study performs the best among all networks on the three testsets and that all networks achieve the largest IoU on the HR testset and achieve the smallest IoU on the Bicubic testset. It can also be observed from Table 5 that the three U-Net networks achieve larger IoU on the three testsets than that obtained by the three FCN networks and that the three U-Net networks obtain larger IoU on the three testsets as the network depth increases. The 8s, 16s and 32s in the three FCN networks represent the downsampling factors, and the FCN networks achieve smaller IoU on the HR testset and SR testset, and larger IoU on the Bicubic testset as the multiplier increases. Specifically, the IoU of the proposed network is the highest on each testset, and the IoU on the HR testset is 73.57%, with a percentage of 3.9% larger than that of the classical U-Net. Fig. 11 shows the segmentation results of the six networks tested on the HR, Bicubic, and SR testset, from which the proposed network in the study achieves the best results for crack segmentation.

4.3. The influence of SRR magnification factor

As seen from previous studies, the reconstruction effect of SR images is significantly affected by the magnification factor [40]. The SRR magnification factor is the upsampling factor that increases the LR image size to the HR image size. The upsampling operations were also illustrated in the network structures shown in Fig. 2. It is necessary to find an optimal magnification factor to apply the SRR technology in the engineering field. Hence, SRR was performed on the same LR images, and the crack segmentation performance of the reconstructed images with different magnification factors was investigated.

The procedures of the experiment were introduced as follows. The images with a resolution of 320 × 320 in the Crack776 dataset were defined as the HR crack dataset, and the HR crack dataset was divided into the training set, validation set and test set, with a percentage of 70%, 10%, and 20%, respectively. The images with a resolution of 80 × 80 were obtained by downsampling the images of the HR crack test set with a factor of 4 according to the degradation model described in Eq. (15), which were denoted as the x1-LR testset. The x1-LR testset was used to represent the LR fuzzy dataset collected in realistic scenarios, on which the deep learning-based SRR was performed. SRFBN was chosen as the SRR network and the magnification factors were set to 2,3,4, and 5. The HR dataset for training the SRR-x2 model was the HR dataset mentioned in Section 3.1.1, and the LR dataset was downsampled by the factor of 2 from the HR dataset according to the degradation model described in Eq. (15). The HR dataset and the LR dataset with a factor of 2 were fed into SRFBN to train the SRR-2 model. The x1-LR testset (LR crack dataset) was fed into the SRR-2 model to obtain the x2-SR testset (160 × 160). The x3-SR testset (240 × 240), x4-SR testset (320 × 320), and x5-SR testset (400 × 400) were generated by a similar process to the



(a) Training loss (b) IoU on the validation set

Fig. 10. Variation of training loss and IoU on the validation set with the increasing number of epochs.

Table 5
The segmentation results of different networks on various testsets.

Metrics	Datasets	FCN-8 s	FCN-16 s	FCN-32 s	U-Net	UNet-resnet18	UNet-resnet34	Ours
	HR	69.98	67.46	66.34	69.65	71.22	72.84	**73.57**
IoU (%)	Bicubic	36.33	41.25	42.03	48.76	49.07	51.60	**53.30**
	SR	64.99	64.06	63.42	67.05	68.49	70.35	**70.59**

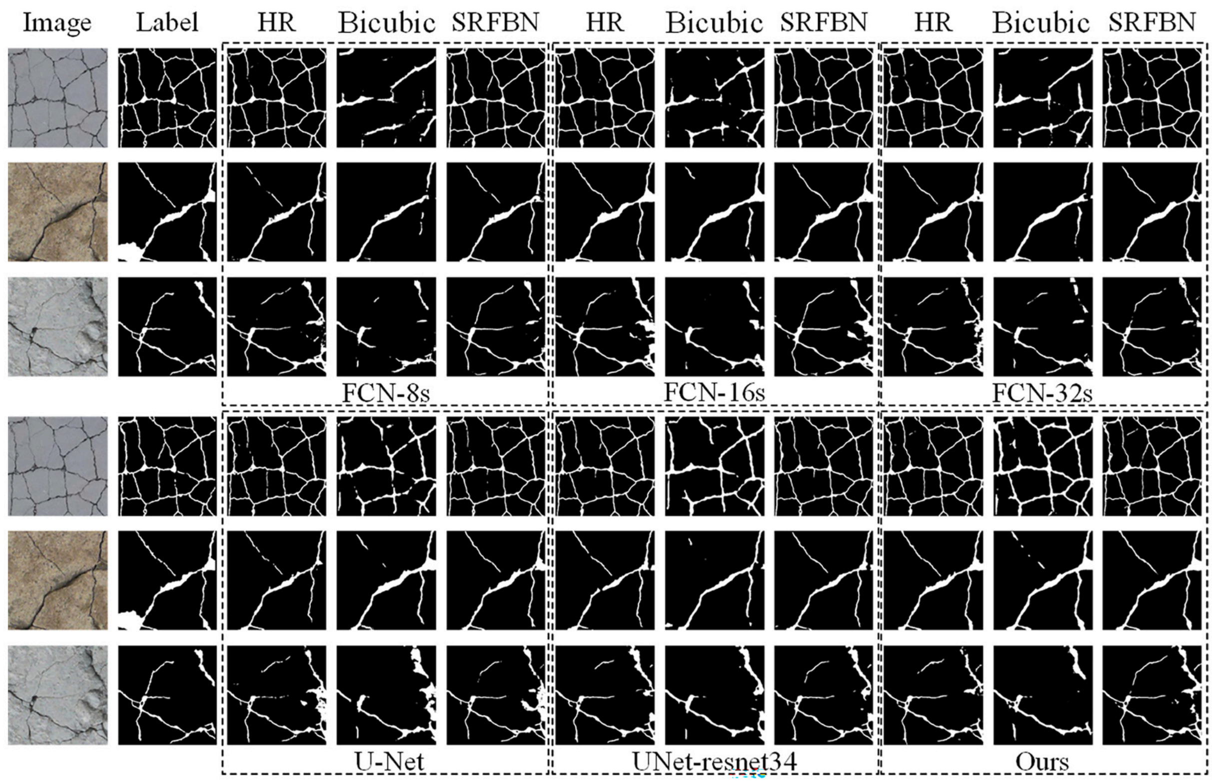


Fig. 11. Segmentation results of different networks on HR, Bicubic, and SR testset.

x2-SR testset that were reconstructed with the corresponding magnification factor. The training set and validation set in the HR crack dataset were fed into the CDU-Net to train the optimal segmentation model, which was then used to segment the testset consisting of reconstructed crack images with different factors.

The segmentation performance of the crack images reconstructed by the SRR algorithm with different magnification factors can be represented by the Precision-Recall (P-R) curves. Fig. 12 shows the P-R curves of the segmentation results for different crack testsets, in which x(m)-SR represents the segmentation results on the SR testset magnified by the factor m, and x1-LR represents the segmentation results on the x1-LR testset, in which F1 represents the evaluation metric F1-score. Fig. 13 illustrates the segmentation results of some reconstructed crack images with different magnification factors, and Fig. 14 demonstrates the variation of segmentation results (i.e., F1-score, IoU) with the SRR magnification factor. It can be seen from Fig. 14 that the F1-score and IoU of the reconstructed images increase dramatically at first with the increase of the magnification factor and then tend to stabilize. As expected, as the magnification factor increases the difference between the segmentation map of the reconstructed images and the ground truth label becomes smaller but the time required to reconstruct an image becomes longer. Therefore, a good balance between the segmentation accuracy and the computational efficiency can be achieved by adopting a magnification factor of 4.

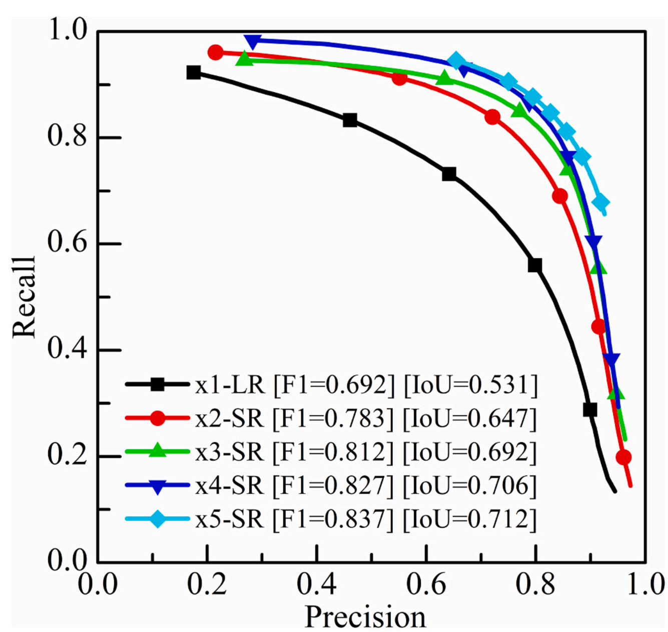


Fig. 12. P-R curves for the segmentation results of reconstructed crack images with different magnification factors.

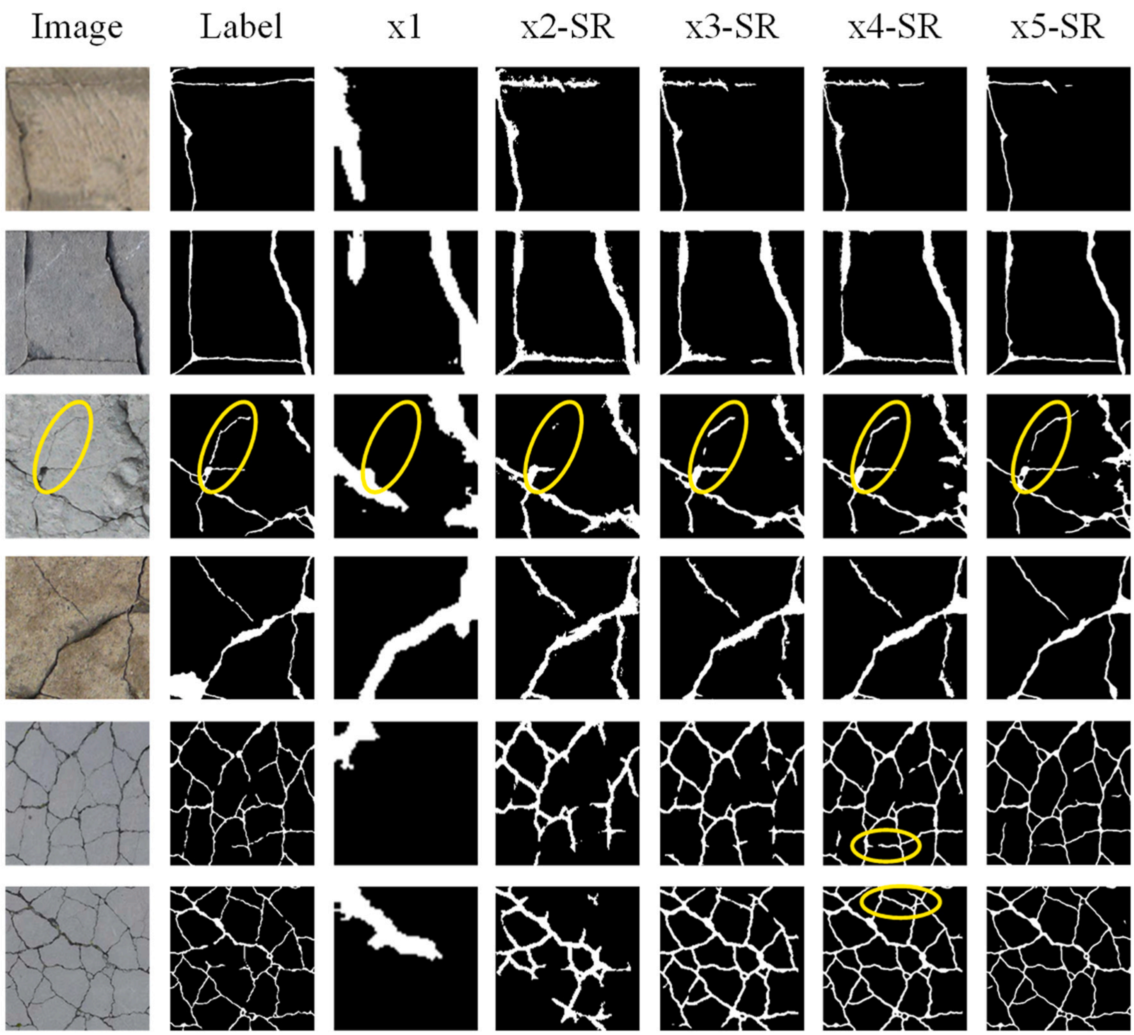


Fig. 13. Segmentation results of images from SR testsets obtained with different magnification factors.

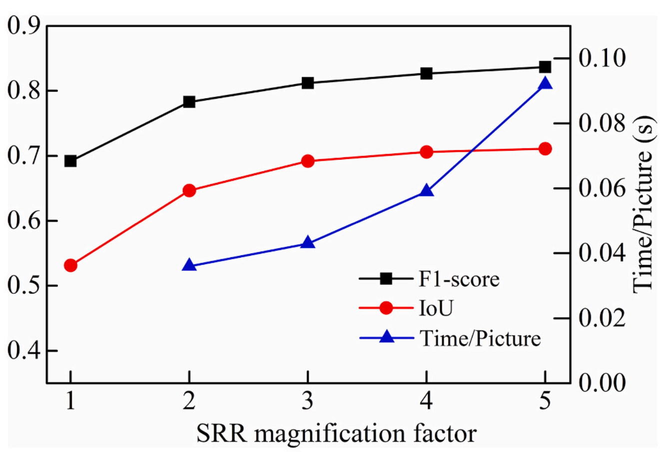


Fig. 14. The variation of segmentation results with SRR magnification factor.

5. Conclusion

To solve the problems of motion blur and insufficient resolution encountered in the process of image acquisition for infrastructure crack detection using UAVs, an automatic detection method for microcracks is proposed in this study based on deep learning SRR and semantic segmentation. Firstly, a deep learning-based SRR technique was used to reconstruct the LR images. Then, a new pixel-level crack segmentation network (i.e., CDU-Net) was proposed for segmenting reconstructed crack images. Finally, the length and width of the cracks were quantified by an improved medial axis transform algorithm.

The segmentation results of different SR datasets and the quantification of the crack features in the segmentation maps were investigated in detail. The results show that the current mainstream deep learning algorithms for SRR give similar results of crack image reconstruction from the perspective of visual observation and various evaluation metrics. SRFBN has the best reconstruction effect and requires the least number of parameters. The CNN-based and GAN-based reconstruction methods achieve better results than the Bicubic method in terms of sharpness of the reconstructed images. The accuracy of crack segmentation and feature quantification of SR images obtained using the deep learning-based SRR is much better than that of the low-resolution blurred images (i.e., the images obtained by the Bicubic method) and is almost the same as that of HR images. The CDU-Net presented in this study achieves the highest precision (84.51%), recall (85.22%), F1-score (84.86%), and IoU (73.57%) on the original HR testset, which are significantly better than other networks. The F1-score and IoU obtained from the SRFBN testset reached 82.76% and 70.59%, which are 13% and 17% higher than that from the Bicubic testset, respectively. Compared with the Bicubic testset, the AEs of crack length, maximum width and mean width obtained from the SRFBN testset are reduced by 59.1 pixels, 1.7 pixels and 0.9 pixels, respectively. The comparative study shows that the method proposed in the present study has better performance in detecting concrete cracks, especially for thin cracks.

The influence of different training datasets used for SRR on the reconstruction effect was discussed. It can be concluded that the training set incorporating natural scene images with crack images performs the best. The proposed crack segmentation network CDU-Net was also compared with FCN and U-Net, and the proposed network can achieve better crack detection results than the classical FCN and U-Net. Finally, the segmentation accuracy of crack images with different magnification factors was analyzed. The segmentation accuracy of the SR testset increases with the increase of magnification factor and then tends to stabilize. Therefore, it is very meaningful to use SRR with appropriate magnification factors in practice to improve the accuracy of pixel-level segmentation for microcrack detection. It is also found that a good balance between the segmentation accuracy and the computational efficiency can be achieved by adopting a magnification factor of 4.

However, the crack detection method proposed in the study still needs two separate stages, which may not fulfill the demand for real-time crack detection. In future studies, a new optimized network that can incorporate super-resolution reconstruction and semantic segmentation will be investigated. Also, the degradation model used in this study contains only a fixed number of basic degradation methods, while the degradation process in real life is diverse and usually contains multiple degradation factors, such as imaging systems, blur types, and compression methods, which lead to complex degradation simulation. To deal with more practical degradations, the classical degradation model will be further extended to a higher-order degradation model or modified to include more diverse fuzzy kernels.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to acknowledge the financial support provided by the National Natural Science Foundation of China (Grant No. 51808209 and 51778222) and the Fundamental Research Funds for the Central Universities through the Project of Young Teacher Growth of Hunan University (Grant No. 531118010081).

References

[1] D. Kang, S.S. Benipal, D.L. Gopal, Y.J. Cha, Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning, Autom. Constr. 118 (2020), 103291, https://doi.org/10.1016/j.autcon.2020.103291.

[2] F. Ni, J. Zhang, Z. Chen, Zernike-moment measurement of thin-crack width in images enabled by dual-scale deep learning, Computer-Aided Civil Infrast. Eng. 34 (5) (2018) 367–384, https://doi.org/10.1111/mice.12421.

[3] K. Gopalakrishnan, S.K. Khaitan, A. Choudhary, A. Agrawal, Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection, Constr. Build. Mater. 157 (2017) 322–330, https://doi.org/10.1016/j.conbuildmat.2017.09.110.

[4] H. Wang, L. Zhai, H. Huang, L.M. Guan, K.N. Mu, G.P. Wang, Measurement for cracks at the bottom of bridges based on tethered creeping unmanned aerial vehicle, Autom. Constr. 119 (2020), 103330, https://doi.org/10.1016/j.autcon.2020.103330.

[5] R.S. Adhikari, O. Moselhi, A. Bagchi, Image-based retrieval of concrete crack properties for bridge inspection, Autom. Constr. 39 (2014) 180–194, https://doi.org/10.1016/j.autcon.2013.06.011.

[6] H.N. Nguyen, T.Y. Kam, P.Y. Cheng, An automatic approach for accurate edge detection of concrete crack utilizing 2D geometric features of crack, J. Signal Process. Syst. 77 (3) (2013) 221–240, https://doi.org/10.1007/s11265-013-0813-8.

[7] T. Yamaguchi, S. Nakamura, R. Saegusa, S. Hashimoto, Image-based crack detection for real concrete surfaces, Transact. Electr. Electr. Eng. 3 (1) (2008) 128–135, https://doi.org/10.1002/tee.20244.

[8] S. Sankarasrinivasan, E. Balasubramanian, K. Karthik, U. Chandrasekar, R. Gupta, Health monitoring of civil structures with integrated UAV and image processing system, in, Procedia Comp. Sci. (2015) 508–515, https://doi.org/10.1016/j.procs.2015.06.058.

[9] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556, arXiv (2015) 1–14. http://arxiv.org/abs/1409.1556v6 (accessed April 10, 2015).

[10] Y.A. Hsieh, Y.J. Tsai, Machine learning for crack detection: review and model performance comparison, J. Comput. Civ. Eng. 34 (5) (2020) 04020038, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000918.

[11] M.T. Cao, Q.V. Tran, N.M. Nguyen, K.T. Chang, Survey on performance of deep learning models for detecting road damages using multiple dashcam image resources, Adv. Eng. Inform. 46 (2020), 101182, https://doi.org/10.1016/j.aei.2020.101182.

[12] M. Alipour, D.K. Harris, Increasing the robustness of material-specific deep learning models for crack detection across different materials, Eng. Struct. 206 (2020), 110157, https://doi.org/10.1016/j.engstruct.2019.110157.

[13] Y. Li, H. Li, H. Wang, Pixel-wise crack detection using deep local pattern predictor for robot application, Sensors (Basel). 18 (9) (2018) 3042, https://doi.org/10.3390/s18093042.

[14] W. Deng, Y. Mou, T. Kashiwa, S. Escalera, K. Nagai, K. Nakayama, Y. Matsuo, H. Prendinger, Vision based pixel-level bridge structural damage detection using a link ASPP network, Autom. Constr. 110 (2020), 102973, https://doi.org/10.1016/j.autcon.2019.102973.

[15] Y.J. Cha, W. Choi, O. Büyüköztürk, Deep learning-based crack damage detection using convolutional neural networks, Computer-Aided Civil Infrast. Eng. 32 (5) (2017) 361–378, https://doi.org/10.1111/mice.12263.

[16] Y.J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, O. Büyüköztürk, Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types, Computer-Aided Civil Infrast. Eng. 33 (9) (2018) 731–747, https://doi.org/10.1111/mice.12334.

[17] B.F. Spencer, V. Hoskere, Y. Narazaki, Advances in computer vision-based civil infrastructure inspection and monitoring, Engineering. 5 (2) (2019) 199–222, https://doi.org/10.1016/j.eng.2018.11.030.

[18] Z. Yu, Y. Shen, C. Shen, A real-time detection approach for bridge cracks based on yolov4-fpm, Autom. Constr. 122 (2021), 103514, https://doi.org/10.1016/j.autcon.2020.103514.

[19] J. Deng, Y. Lu, V.C.S. Lee, Concrete crack detection with handwriting script interferences using faster region-based convolutional neural network, Computer-Aided Civil Infrast. Eng. 35 (4) (2019) 373–388, https://doi.org/10.1111/mice.12497.

[20] J. Huyan, W. Li, S. Tighe, J. Zhai, Z. Xu, Y. Chen, Detection of sealed and unsealed cracks with complex backgrounds using deep convolutional neural network, Autom. Constr. 107 (2019), 102946, https://doi.org/10.1016/j.autcon.2019.102946.

[21] X. Yang, H. Li, Y. Yu, X. Luo, T. Huang, X. Yang, Automatic pixel-level crack detection and measurement using fully convolutional network, Computer-Aided Civil Infrast. Eng. 33 (12) (2018) 1090–1109, https://doi.org/10.1111/mice.12412.

[22] H.W. Huang, Q.T. Li, D.M. Zhang, Deep learning based image recognition for crack and leakage defects of metro shield tunnel, Tunn. Undergr. Space Technol. 77 (2018) 166–176, https://doi.org/10.1016/j.tust.2018.04.002.

[23] S.L.H. Lau, E.K.P. Chong, X. Yang, X. Wang, Automated pavement crack segmentation using u-net-based convolutional neural network, IEEE Access. 8 (2020) 114892–114899, https://doi.org/10.1109/access.2020.3003638.

[24] Y. Ren, J. Huang, Z. Hong, W. Lu, J. Yin, L. Zou, X. Shen, Image-based concrete crack detection in tunnels using deep fully convolutional networks, Constr. Build. Mater. 234 (2020), 117367, https://doi.org/10.1016/j.conbuildmat.2019.117367.

[25] T. Chen, Z. Cai, X. Zhao, C. Chen, X. Liang, T. Zou, P. Wang, Pavement crack detection and recognition using the architecture of SegNet, J. Ind. Inf. Integr. 18 (2020), 100144, https://doi.org/10.1016/j.jii.2020.100144.

[26] H. Bae, K. Jang, Y.K. An, Deep super resolution crack network (SrcNet) for improving computer vision-based automated crack detectability in in situ bridges, Struct. Health Monit. 20 (4) (2020), https://doi.org/10.1177/1475921720917227, 147592172091722.

[27] Y. Liu, J.K.W. Yeoh, D.K.H. Chua, Deep learning-based enhancement of motion blurred UAV concrete crack images, J. Comput. Civ. Eng. 34 (5) (2020) 04020028, https://doi.org/10.1061/(ASCE)cp.1943-5487.0000907.

[28] H. Kim, J. Lee, E. Ahn, S. Cho, M. Shin, S.H. Sim, Concrete crack identification using a UAV incorporating hybrid image processing, Sensors (Basel). 17 (9) (2017) 2052, https://doi.org/10.3390/s17092052.

[29] A. Ellenberg, A. Kontsos, F. Moon, I. Bartoli, Bridge related damage quantification using unmanned aerial vehicle imagery, Struct. Control. Health Monit. 23 (9) (2016) 1168–1179, https://doi.org/10.1002/stc.1831.

[30] Y. Huang, H. Zhang, H. Li, S. Wu, Recovering compressed images for automatic crack segmentation using generative models, Mech. Syst. Signal Process. 146 (2021), 107061, https://doi.org/10.1016/j.ymssp.2020.107061.

[31] J.S. Isaac, R. Kulkarni, Super resolution techniques for medical image processing, in: International Conference on Technology for Sustainable Development, Mumbai, India, 2015, pp. 1–6, https://doi.org/10.1109/ICTSD.2015.7095900.

[32] K. Sathya, D. Sangavi, P. Sridharshini, M. Manobharathi, G. Jayapriya, Improved image based super resolution and concrete crack prediction using pre-trained deep learning models, J. Soft Comput. Civil Eng. 4 (3) (2020) 40–51, https://doi.org/10.22115/scce.2020.229355.1219.

[33] Y.K. Ooi, H. Ibrahim, Deep learning algorithms for single image super-resolution: a systematic review, Electronics. 10 (7) (2021) 867, https://doi.org/10.3390/electronics10070867.

[34] M. Ding, Y. Zhang, Z. Sun, W. Li, H. Ju, Super-resolution reconstruction of road crack image based on enhanced generative adversarial networks, in: Proceedings of the 2018 World Transport Convention Beijing, China, 2018, pp. 66–75. http://road.cnki.net/download/jtdh/GL0601-016.pdf.

[35] W. Yang, X. Zhang, Y. Tian, W. Wang, J.H. Xue, Q. Liao, Deep learning for single image super-resolution: a brief review, IEEE Transact. Multimedia. 21 (12) (2019) 3106–3121, https://doi.org/10.1109/tmm.2019.2919431.

[36] D. Chao, C.L. Chen, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, in: European Conference on Computer Vision, Zurich, Switzerland, 2014, pp. 184–199, https://doi.org/10.1007/978-3-319-10593-2_13.

[37] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, Photo-realistic single image super-resolution using a generative adversarial network, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 4681–4690, https://doi.org/10.1109/CVPR.2017.19.

[38] B. Lim, S. Son, H. Kim, S. Nah, K.M. Lee, Enhanced deep residual networks for single image super-resolution, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 1132–1140, https://doi.org/10.1109/cvprw.2017.151.

[39] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. Fu, Image super-resolution using very deep residual channel attention networks, arXiv:1807.02758, arXiv (2018), https://doi.org/10.1007/978-3-030-01234-2_18 (accessed July 12, 2018).

[40] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, W. Wu, Feedback network for image super-resolution, in: Proceeding IEEE Computing Society Conference Computer Vision and Pattern Recognition, Long Beach, USA, 2019, pp. 3867–3876, https://doi.org/10.1109/CVPR.2019.00399.

[41] D. Mahapatra, B. Bozorgtabar, R. Garnavi, Image super-resolution using progressive generative adversarial networks for medical image analysis, Comput. Med. Imaging Graph. 71 (2019) 30–39, https://doi.org/10.1016/j.compmedimag.2018.10.005.

[42] Q. Delannoy, C.H. Pham, C. Cazorla, C. Tor-Diez, G. Dolle, H. Meunier, N. Bednarek, R. Fablet, N. Passat, F. Rousseau, SegSRGAN: super-resolution and segmentation using generative adversarial networks - application to neonatal brain MRI, Comput. Biol. Med. 120 (2020), 103755, https://doi.org/10.1016/j.compbiomed.2020.103755.

[43] Y. Jin, Y. Zhang, Y. Cen, Y. Li, V. Mladenovic, V. Voronin, Pedestrian detection with super-resolution reconstruction for low-quality image, Pattern Recogn. 115 (2021), 107846, https://doi.org/10.1016/j.patcog.2021.107846.

[44] S.S. Rajput, K.V. Arya, A robust face super-resolution algorithm and its application in low-resolution face recognition system, Multimed. Tools Appl. 79 (4) (2020) 23909–23934, https://doi.org/10.1007/s11042-020-09072-5.

[45] Y. Kondo, N. Ukita, Crack segmentation for low-resolution images using joint learning with super-resolution, in: 17th International Conference on Machine Vision Applications, Aichi, Japan, 2021, p. 9511400, https://doi.org/10.23919/MVA51890.2021.9511400.

[46] J. Kim, S. Shim, G.C. Cho, A study on the crack detection performance for learning structure using super-resolution, in: Advances in Structural Engineering and Mechanics, Seoul, Korea, 2021, pp. 1–5. http://www.i-asem.org/publication_conf/asem21/6.TS/3.%20W5A/4.%20TS1406_6949.pdf.

[47] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[48] G. Huang, Z. Liu, V. Laurens, K.Q. Weinberger, Densely connected convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 2017, pp. 2261–2269, https://doi.org/10.1109/CVPR.2017.243.

[49] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Adv. Neural Inf. Proces. Syst. 3 (2014) 2672–2680, https://doi.org/10.1145/3422622.

[50] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 2472–2481, https://doi.org/10.1109/CVPR.2018.00262.

[51] M. Haris, G. Shakhnarovich, N. Ukita, Deep back-projection networks for super-resolution, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 1664–1673, https://doi.org/10.1109/CVPR.2018.00179.

[52] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C.C. Loy, Y. Qiao, X. Tang, ESRGAN: Enhanced super-resolution generative adversarial networks, in: European Conference on Computer Vision, Germany, Munich, 2019, pp. 63–79, https://doi.org/10.1007/978-3-030-11021-5_5.

[53] L. Zhang, J. Shen, B. Zhu, A research on an improved UNet-based concrete crack detection algorithm, Struct. Health Monit. 20 (4) (2021) 1864–1879, https://doi.org/10.1177/1475921720940068.

[54] Z.W. Gu, J. Cheng, H.Z. Fu, K. Zhou, H.Y. Hao, Y.T. Zhao, T.Y. Zhang, S.H. Gao, J. Liu, CE-net: context encoder network for 2D medical image segmentation, IEEE Trans. Med. Imaging 38 (10) (2019) 2281–2292, https://doi.org/10.1109/Tmi.2019.2903562.

[55] J. Liu, X. Yang, S. Lau, X. Wang, S. Luo, V.C.S. Lee, L. Ding, Automated pavement crack detection and segmentation based on two-step convolutional neural network, Computer-Aided Civil Infrast. Eng. 35 (11) (2020) 1291–1305, https://doi.org/10.1111/mice.12622.

[56] Z. Fan, C. Li, Y. Chen, P.D. Mascio, X. Chen, G. Zhu, G. Loprencipe, Ensemble of deep convolutional neural networks for automatic pavement crack detection and measurement, Coatings. 10 (2) (2020) 152–166, https://doi.org/10.3390/coatings10020152.

[57] J. Keunyoung, A.Y. Kyu, K. Byunghyun, C. Soojin, Automated crack evaluation of a high-rise bridge pier using a ring-type climbing robot, Computer-Aided Civil Infrast. Eng. 36 (1) (2020) 14–29, https://doi.org/10.1111/mice.12550.

[58] E. Agustsson, R. Timofte, Ntire 2017 challenge on single image super-resolution: dataset and study, in: IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 1122–1131, https://doi.org/10.1109/Cvprw.2017.150.

[59] E. Sert, F. Ozyurt, A. Dogantekin, A new approach for brain tumor diagnosis system: single image super resolution based maximum fuzzy entropy segmentation and convolutional neural network, Med. Hypotheses 133 (2019), 109413, https://doi.org/10.1016/j.mehy.2019.109413.

[60] T. Hayashi, T. Tsubouchi, Estimation and sharpening of blur in degraded images captured by a camera on a moving object, Sensors. 22 (4) (2022) 1635, https://doi.org/10.3390/s22041635.

[61] M. Ciolino, D. Noever, J. Kahn, Training set effect on super resolution for automated target recognition, arXiv:1911.07934, arXiv (2020), https://doi.org/10.1117/12.2557845 (accessed February 7, 2020).