Hindawi
Modelling and Simulation in Engineering
Volume 2021, Article ID 5298882, 14 pages
https://doi.org/10.1155/2021/5298882

Research Article

A Modified Fully Convolutional Network for Crack Damage Identification Compared with Conventional Methods

Meng Meng,[1] **Kun Zhu**,[2] **Keqin Chen**,[3] **and Hang Qu**[4]

[1]*School of Civil Engineering, Southeast University, Nanjing, China*
[2]*School of Computer Science, University of Science Malaysia, Penang, Malaysia*
[3]*Department of Big Data Management and Applications, School of Information and Business Management, Chengdu Neusoft University, Dujiangyan, Chengdu, China*
[4]*Medical Imaging Center, Affiliated Hospital of Yangzhou University, Yangzhou University, Yangzhou, China*

Correspondence should be addressed to Meng Meng; 220180956@seu.edu.cn

Received 25 June 2021; Revised 5 August 2021; Accepted 24 October 2021; Published 10 November 2021

Academic Editor: Ricardo Perera

Large-scale structural health monitoring and damage detection of concealed underwater structures are always the urgent and state-of-art problems to be solved in the field of civil engineering. With the development of artificial intelligence especially the combination of deep learning and computer vision, greater advantages have been brought to the concrete crack detection based on convolutional neural network (CNN) over the traditional methods. However, these machine learning (ML) methods still have some defects, such as it being inaccurate or not strong, having poor generalization ability, or the accuracy still needs to be improved, and the running speed is slow. In this article, a modified fully convolutional network (FCN) with more robustness and more effectiveness is proposed, which makes it convenient and low cost for long-term structural monitoring and inspection compared with other methods. Meanwhile, to improve the accuracy of recognition and prediction, innovations were conducted in this study as follows. Moreover, differed from the common simple deconvolution, it also includes a subpixel convolution layer, which can greatly reduce the sampling time. Then, the proposed method was verified its practicability with the overall recognition accuracy reaching up to 97.92% and 12% efficiency improvement.

1. Introduction

With the development of construction industry and building technology, more and more traditional concrete structures have experienced a long service life with the deterioration of material properties, so the structural performance will be influenced by the fatigue damage, which will often affect the normal use of these buildings or structures (including bridges, dams, and tunnels). What is worse, if not treated properly and quickly, these damage or cracks will even threaten the safety of people or citizens. To avoid aging, fatigue, damage, and other problems that usually occurred in infrastructures, it is necessary to carry out long-term and effective monitoring for them so that we can use the results to repair or maintain them.

The traditional monitoring method based on the sensor optimization method has made some progress in recent years [1], but it has a series of problems, such as the cumbersome sensor layout, high cost, and large disturbance by environmental uncertainties (such as the noise or the vibrations) [2–4], which is not suitable for large-scale promotion in most of the areas in the world. Thanks to the development of artificial intelligence, the integration of computer vision and machine learning provides a method which is worthy of reference to solve this problem, that is applying deep convolution neural network (DCNN) to classify and learn the structural damage, then timely predict the structures that may have shown or suffered from the external or internal damage.

The CNN model was selected for the task of interest despite other machine/deep learning methods including the support vector machine (SVM), deep belief network (DBN), and stacked denoising autoencoder (SDAE); the reasons are as follows: SVM is a typical shallow classifier, which

often faces the problem of dimension disaster. DBN training is usually difficult, the initial weight is directional, and it is easy to fall into local optimization. The sigmoid activation function used in SDAE is difficult to establish the accurate mapping relationship between concrete surface damage and image signal. As a special neural network, CNN has few parameters and is easy to stack depth structure. Using CNN, all Sigmoid activation functions of SDAE can be replaced with ReLU, which means making superdepth structure and more sparsity possible. Most importantly, it can effectively extract sparse features of the target according to label.

As a good example to illustrate the application of CNN in civil structure condition assessment, Yu et al. [5] in 2019 proposed a novel method based on deep convolutional neural networks to identify and localise damages of building structures equipped with smart control devices. Meanwhile, we followed the methods of Yin and Zhu [6] in 2018 and made some innovations in ANN (artificial neural network) design.

Compared with the classic CNN (sliding window convolution method) proposed by Cha et al. [7], which is similar to the improved convolution neural network proposed by Chen et al. [8] and U-net method proposed by Liu et al. [9], a new modified FCN method is proposed. Differed from the former, the latter has some advantages in recognition accuracy and efficiency, besides the data loss during training process is also lower than the former. The recognition results showed that the prediction results of the latter are missed, and the number of misjudgments is lower than that of the former. When the mIoU (intersection of union) index reaches up to 0.8, the former will still have duplicate judgments, missed judgments, even with content including misjudgments because of the disturbance of uncertain factors such as stains, shadows, and environmental noise which sometimes happened on the surface of the collected images. To improve the accuracy of recognition and prediction, the image is intentionally preprocessed by gray-scale binarization and threshold segmentation through digital image correlation technology, which can be referred to in the electronic digital image processing written by Gonzalez et al. [10]. The modified fully convolutional network method proposed in this paper weakens the output size of the spatial kernel through the downsample in the special hourglass structure and increases the output size by using upsample.

This method consists of two convolution layers: one is forward convolution and the other is reverse convolution. Upsampling is a distinct form of deconvolution that includes a subpixel convention layer and greatly reduces the sampling time. Compared with ordinary convolutional neural network, it can solve classification and localization.

Then, it is unavoidable to focus on the internal structure of some deep convolution neural networks. The convolution kernel is a convolution layer that contains various characteristic graphs, each of which contains neurons arranged in a triangular array. Convolution kernel refers to the weight shared by neurons of the same feature graph. Each input graph contains the convolution layer and some convolution layer neurons to reduce the lengthy calculation and improve the calculation efficiency. More importantly, it also reduces the space size, saves effective information, and prevents overfitting. The dropout layer is also used to delete some neurons to prevent overfitting. The function of the fully connected (FC) layer will be used to create a number from the three-dimensional matrix of input (height × width × dimension). The created number means the number of neurons. The softmax layer outputs the probability weight matrix of each category. It is significant for the 256×256 resolution image to conform to the recommended input size of the CNN model input. About weight and deviation, the initial value of weight is Gaussian distribution, and the initial value of deviation is set by a continuous initial value. To reduce the training bias, the weight needs to be updated constantly. The stochastic gradient descent method (SGD) is used to update the weight and finally converge during the process of several updates. CNN automatically identifies features in the process of updating the weight again and again. As for the loss function, it is mainly used to measure the deviation between the actual and predicted image categories. The total collected images were divided into a training set and a test set, at a ratio of 4 : 1. In the training set this paper used, train: Val = 2 : 1, the ratio of cracked images to uncracked images is 1 : 1; each batch contains 256 pieces.

The main innovation of this research can be generally concluded as follows: First, the image is intentionally preprocessed by gray-scale binarization and threshold segmentation through digital image processing technology. Second, upsampling is namely being as a form of deconvolution. Compared with ordinary convolutional neural network, it not only solved the problem of classification but also solved the problem of localization, which greatly increases the scope of recognition. Moreover, differed from the common simple deconvolution, it also includes a subpixel convolution layer, which can greatly reduce the sampling time. Finally, the performance of the proposed method was validated by a combined dataset composed of published one and the images collected by the authors using the drone with satisfactory results.

2. Related Work

The neural network is an important part of artificial intelligence research field in recent years. The most popular neural network among all the categories is the deep convolution neural network (DCNN), whose layers vary from tens to hundreds. The application of the convolutional neural network has achieved great success in many kinds of fields, but its main application value can be reflected in that CNN can automatically learn features from large-scale data and transfer the results to the same unknown data generalization which can be deemed as taking effect in playing an intelligent prediction role. Similarly, for image acquisition, with the rapid development of unmanned aerial vehicle (UAV) technology [11, 12] and its wide application in civil engineering industry during recent years, UAV can be utilized as an ideal medium to collect datasets and sample images. It can often overcome many disadvantages like climate, terrain, and temperature. Typically, it is easy for UAV to detect

ancient buildings and bridges which are difficult to be found by human naked eyes. Therefore, from the perspective of protecting old relics and historic landscapes, the method of data collection combined with UAV is nondestructive and environmentally friendly. It can be concluded from the topics mentioned above that applying the deep convolution neural network in structural health monitoring is feasible and promising. Many scholars have also carried out such research, for instance, Dung et al. [13] successfully applying the depth convolution neural network to the crack damage identification of steel bridge joints combined with image enhancement technology. Dorafshan et al. [14] have proved that the depth convolution neural network has superior advantages than the edge detection operator based on image in concrete surface damage detection. What is more, Nagata et al. [15] skillfully combined the support vector machine (SVM) and deep convolution neural network (DCNN) through connecting the trained Alex-Net with two SVM and further proposed a template matching technology to extract the features of important targets effectively, thus improving the reliability and accuracy of defect detection.

Notably, fully convolutional networks are a rich class of models, which have a big upgrade space. Long et al. [16] adapted contemporary classification networks into fully convolutional networks and transferred their learned representations by fine-tuning to the segmentation task. This fully convolutional network achieved state-of-the-art segmentation of PASCAL VOC, NYUDv2, and SIFT Flow. In addition, many researchers made some improvements based on the traditional FCN: Chen and Jahanshahi [17] proposed a new approach called NB-fully convolutional network (NB-FCN) that detects cracks from inspection videos in real time with high precision. Attard et al. [18] demonstrated that Mask R-CNN can be used to localize cracks on concrete surfaces and gain their corresponding masks to aid extract other properties that are useful for inspection. Rafeed et al. [19] analyzed how well a fast fully convolutional network (FastFCN) semantically segment satellite images and thus classifies land use/land cover (LULC) classes.

More recently, some state-of-the-art papers in this topic with a wider application range and more complex application background are worth mentioning: Cha et al. [20] Proposed an autonomous structural damage inspection method based on faster R-CNN to provide quasireal-time simultaneous detection of multiple types of damages and achieved satisfactory results, which include concrete cracks, steel corrosion (medium and high levels), bolt corrosion, and steel delamination. Choi and Cha [21] developed a pure deep learning method for segmenting concrete cracks in images, which is called the semantic damage detection network (SDD Net); the test results show that the SDD Net segments crack effectively unless the features are too faint. Meanwhile, it returns better evaluation metrics and processes more real-time images compared with some recent models. Kang et al. [22] developed a crack detection, localization, and quantification method by integrating three different methods (i.e., a faster R-CNN, a modified TuFF method, and a modified DTM) for application to realistic and practical problems that have various complex backgrounds in different environmental conditions.

3. Methodology

3.1. Convolutional Neural Network. A conventional convolutional neural network usually includes an input layer, a convolutional layer, a pooling layer (typically, it is shown as the form of max pooling layer or the average pooling layer), a fully connected layer, a softmax layer, a dropout layer, and an output layer [23–25].

For CNN structure, convolutional layer mostly changes the number of channels, and pooling layer will play a role in reducing the image size by downsampling the feature maps or reduce their dimensionality, without losing any significant information or any background information.

The softmax layer acts as a role of the classifier; the fully connected layer typically follows the dropout layer, activation function such as the rectified linear unit function (ReLU) or the logistic/Sigmoid function will follow convolution layer [26]. In most of the situations, ReLU ought to be given the priority because it has the capability to training the neural networks multiple times faster than the other activation functions. ReLU is also an element-wise operation, and all the negative values are set to zero. As a whole, the activation function makes sure that the feature map can be applied with nonlinear operations after the linear convolution because most of the data this paper used for learning is nonlinear. The LRM (logistic regression method) layer will follow the pooling layer [27]; generally, a convolutional layer will be followed by a pooling layer.

Next, it can be focused on the internal structure of some deep convolution neural networks. Convolution kernel refers to a convolution layer that contains a variety of characteristic graphs, each of which contains neurons arranged in a triangular array [28]. The convolution kernel refers to the weight shared by neurons of the same feature graph. Each input graph contains the convolution layer and some convolution layer neurons to reduce the lengthy calculation and improve the calculation efficiency. More importantly, it also reduces the space size, saves effective information, and prevents overfitting. The dropout layer is also used to delete some neurons to prevent overfitting. The function of the FC (fully connected) layer will be used to generate a number from the three-dimensional matrix of input (height × width × dimension). The generated number means the number of neurons. The softmax layer outputs the probability weight matrix of each category. It is significant that the raw image conforms to the recommended input size of CNN input. About the weight and deviation, the initial value of weight is Gaussian distribution, and the initial value of deviation is set by continuous initial value. To reduce the training bias, the weight needs to be updated constantly. The stochastic gradient descent method (SGD) is used to update the weight and finally converge during the process of several updates. CNN automatically identifies features in the process of updating the weight again and again. The loss function is mainly used to measure the deviation between the actual and predicted image categories.

3.2. Typical U-Net Network. Ronneberger et al. [29] proposed a U-net architecture to improve the classic convolutional neural network in its recognition efficiency and prediction accuracy. One of the U-net's unique features is the symmetrical contracting and expanding blocks, which is known as an encoder-decoder network, compared with the traditional convolution networks including the VGG-16 [30], LeNet-5 [31], Alex-Net [32], Res-Net [33], it is special enough to work as an upsampling layer and dense skip connections.

The skip connections are introduced to train a quite deep network that is difficult for us to train through other methods because of the gradient vanishment and gradient explosion. It can get the activation value from a certain layer of the network and transfer the information to the deeper network (combining the input steams and adding the points); the residual block can train the deeper neural network [15].

It is necessary to mention that high-resolution features from the encoder network are combined with the upsampled feature maps by skip connections, allowing the convolution layers in the decoder network to output more precise results. An encoder-decoder network and dense skip connections are also applied in our model to obtain better predictions [34].

3.3. Modified Fully Convolutional Network

3.3.1. Dilated Convolutions. The dilated convolutional layer (also called as atrous convolution) allows for exponential increase in field of view without decrease of spatial dimensions. Holes (zeros) are inserted into the convolutional kernels to increase image resolution.

If the kernel size = k, and the step size of hole convolution is r, which is equal to the $k \times k$ values used to calculate the convolution. The receptive field is obtained from the position separated by $r - 1$ in the feature map, so the receptive field changes from $k \times k$ to $k + (r - 1) \times (k - 1)$, and the latter part represents the number of 0 to be inserted. Actually, in the dilated convolution, different dilation rates at different ranges into convolution are used. In all, this method can replace the subsampling layers of CNNs to expand the receptive field without loss of image resolution and convergence.

3.3.2. Spatial Pyramid Pooling. To apply a depth learning network into the images with any size, the last pooling layer is replaced with a spatial pyramid pooling (SPP) layer (for example, pooling 4 after the last convolution layer). In each spatial interval, maximum pooling can be used to pool the response of each filter (the last convolution in the graph has 256 filters). The output of the spatial pyramid pool is a $k \times m$-dimensional vector, where the number of regions is expressed as m (k is the number of filters in the last convolution layer). Fixed dimensional vectors are input into the full connect layer.

Spatial pyramid pooling is one of the most successful methods for multiple scale image fusion in computer vision. This method has the function to segment the image from coarse to fine levels and aggregates multilevel features into the output for effective image classification and object detection.

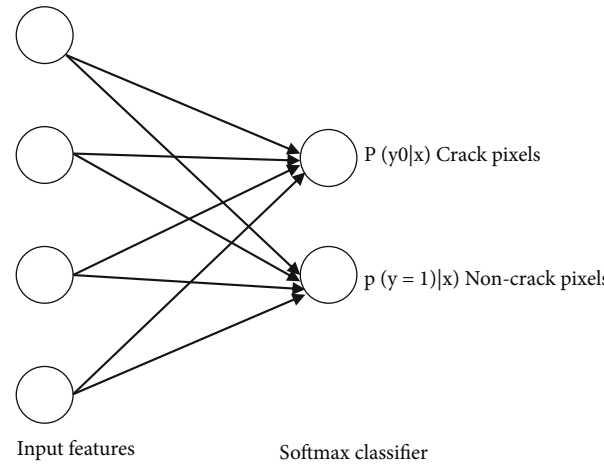


Figure 1: Structure of a softmax layer.

3.3.3. Deconvolutional Layer. In pixel-wise prediction such as the image segmentation and the image generation, it is necessary to predict the size space of the original image. While for convolution, since the size of the image is often reduced by striding, it is often needed to restore the original image size by upsampling, and the deconvolution layer often plays a role of upsampling. Deconvolution is the opposite of convolution in the forward and backward propagation of the neural network structure. Compared with deconvolution, transposed convolution is a more suitable term.

The deconvolutional layer is actually a transposed convolutional layer with a specific stride, length, and padding, which has the capability to convert a coarse input tensor into a dense output tensor. At first, each element of the input tensor (image) is multiplied by the deconvolutional kernel, and then, these middle matrixes are combined with strides in both the horizontal and vertical axes. When elements are overlapped, their values are added together to extract an extended matrix of input. Finally, the matrix with objective size is cropped, followed by adding biases. It is obvious that the size of output is larger than that of input, leading to an efficient approach to upsampling. Notably, implementation of deconvolutional layers is the same as normal convolutional layers, but the filter is the transpose form of normal filter. The deconvolutional layer is the core part which enlarges the size of tensor generated by down-sampling process to the original image size. Such structure enables CNN to detect cracks without sliding original images into patches, leading to a satisfactory accuracy with local and global dependencies.

3.3.4. Softmax Layer. The output of the final deconvolutional layer is a tensor that each pixel within the original image has been scored with each class. To generalize the output, a logistic function named softmax is utilized for multiple class classification [35]. That means the softmax function plays a role of classifier. Figure 1 shows that the softmax function

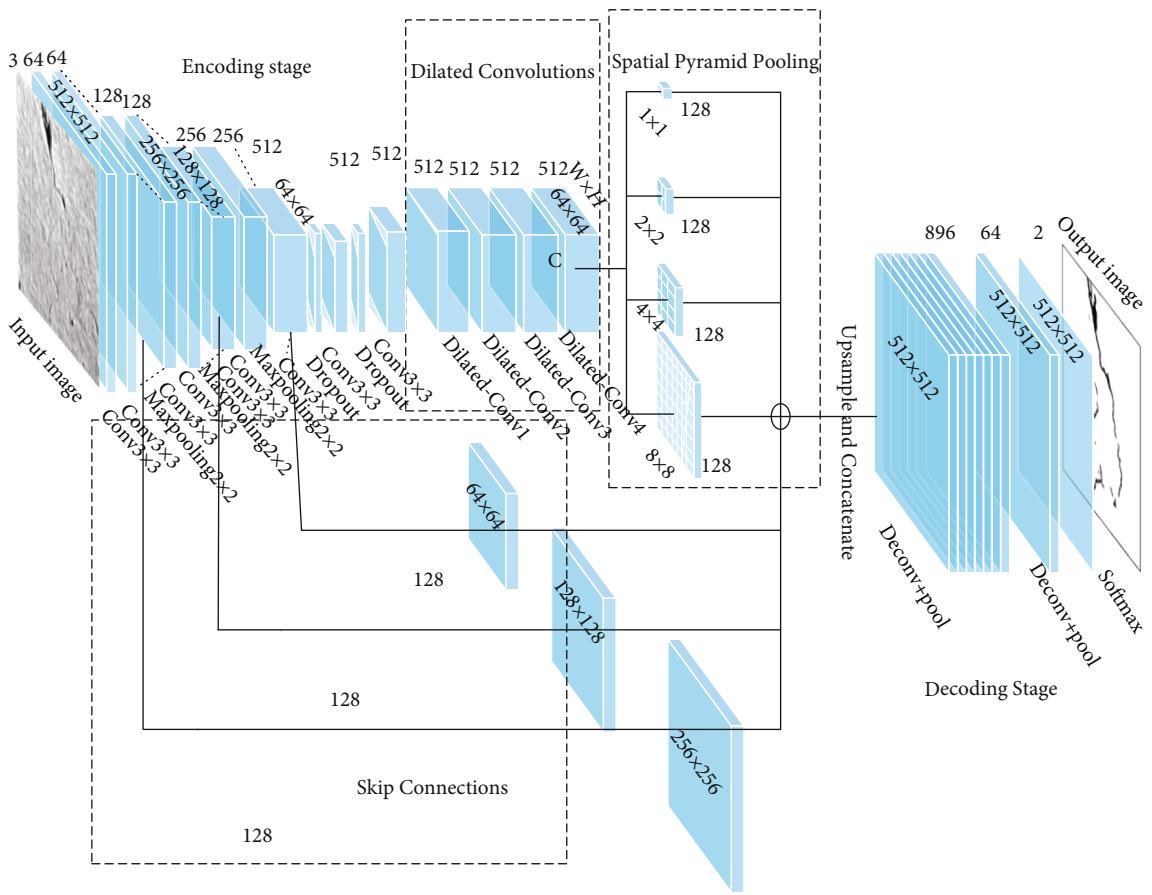


Figure 2: Architecture of the modified FCN network.

can change the input values into a multiple class categorical probability distribution:

3.3.5. Network Architecture. The architecture is listed in Figure 2.

The modified fully convolutional network was developed to carry out the effective crack damage monitoring for the civil infrastructures, which adopted modular design concepts of an encoder path, a decoder path, dilated convolutions, spatial pyramid max pooling, deconvolutional layer, softmax layer, and skip connections. All of these components were combined in a reasonable way. The backbone net in the encoder path is transformed from the classical convolutional network VGG-19, which contains a succession of two 3×3 convolutional layer and a 2×2 max pooling layer. ReLU acted as the activation function and followed each convolutional network. After 4 stages of convolutional and max pooling layers, as well as the dropout layer to avoid the problem of data overfitting, the feature vector is input to 4 dilated convolutional layers with dilation of 2, 4, 2, and 4 for additional feature extraction without downsampling. Afterwards, the decoder path uses spatial pyramid max pooling to generate diverse subregion representations, which were followed by upsampling and concatenation. The feature maps through each convolution stage in the encoder part were upsampled and concatenated with the features after spatial pyramid max pooling by skip connections. Finally, a feature map was inserted into two deconvolutional layers and the softmax layer to obtain the accurate pixel-level prediction.

4. Training Process

4.1. Dataset. To train the modified FCN model, the authors collected more than 20000 images, including some of the datasets downloaded from Kaggle. Part of the datasets were collected by Dajiang Mavic air2 unmanned aerial vehicle (UAV) to carry out the attached image acquisition for the hidden concrete bridge underwater structure. This concrete continuous bridge was collected from a town in Yangzhou, China. The environment of underwater structures, such as concrete piers, is more complex than that of the superstructure. It has the characteristics of rapid flow, turbid water quality, low visibility, strong corrosivity, and large sediment concentration on the surface of the structure. The quality of image acquisition is often disturbed by a series of uncertain environmental factors such as wind vibration, noise, and water impact. To solve these problems and conduct monitoring in the complex environment like these, we sometimes use ROV (remoted operated vehicle) where microcamera attached to dive into the water and conduct the image acquisition. We also apply related technology such as the image generation and raw data enhancement or restoration to make sure the data availability for the training in our model.

Table 1: Typical confusion matrix.

Data	Classified as positive	Classified as negative
Positive	True positive (TP)	False negative (FN)
Negative	False positive (FP)	True negative (TN)

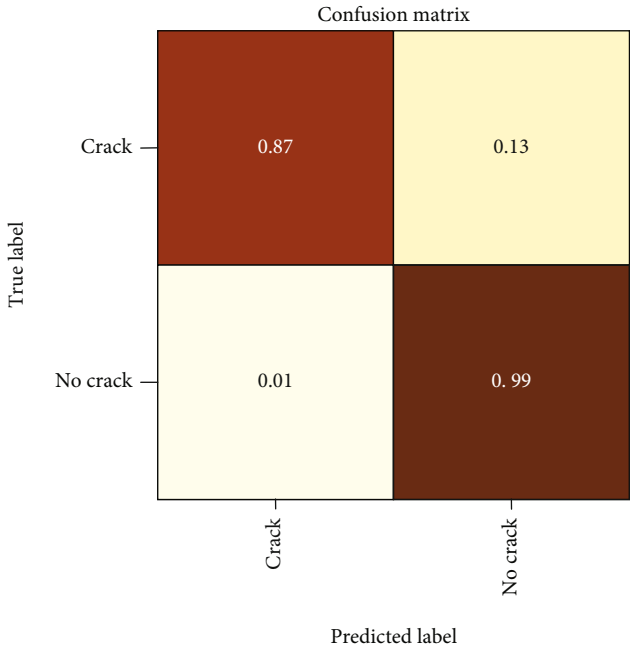


Figure 3: Confusion matrix of our model.

To increase the diversity of datasets, some extra crack images taken by UAV from the existing buildings in Chengdu, China, were also added. The width of cracks varies from one pixel to 100 pixels; the shape of which is hard to recognize at image level because of the noise robustness of the environment or some surface stains, rust, and corrosion. It is not only the asphalt or cement concrete pavement but also includes the concrete walls of buildings or structures. All of these data images are saved and combined into a JPG format. Cracks in these images were taken at different distances depending on their sizes; each image has RGB pixels of 256 pixels × 256 pixels. As for the ground truth of the image dataset for model training, the crack images were manually annotated with Photoshop2020, a common and useful software package. The crack images were labeled as the following method: background pixels were marked by 0 and crack pixels were labeled by 1, allowing storage of the image information in binary format.

Meanwhile, the model applied in this article is trained from data of the labeled crack images using backward propagation of errors, an algorithm used for supervised learning of artificial neural networks using stochastic gradient descent. This algorithm can be conducted into separated procedures.

(a) The initial weight value and bias are assigned by the Gauss distribution method

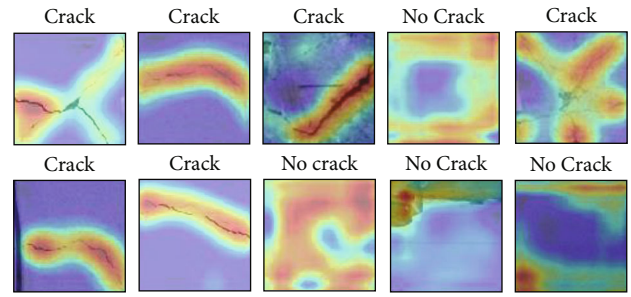


Figure 4: Samples of heat map in our model.

(b) Insert an example and obtain a prediction

(c) Calculate the error between the prediction value and the true value

(d) Constantly update the weights according to the decaying learning rate which is shown by the given formula (1) to reduce the training bias

$$\omega^{\tau+1} = \omega^{\tau} - \alpha \frac{\partial E}{\partial \omega}, \tag{1}$$

where E is a loss function that defines the error between the prediction value and the truth value and means the weights of the neural network for iteration t by gradient descent.

4.2. Loss Function. The loss function is used to evaluate the difference between the predicted value and the real value of the model. The better the loss function is, the better the performance of the model is. Different models usually use different loss functions. Dice loss function is a smooth (Dice coefficient) function, which is most commonly used in segmentation problems [36]. Boundary loss function is applied to tasks with highly unbalanced segments [37]. Lovasz softmax loss function is often used to solve the problem of submodular minimization [38]. To perfectly solve the problem of slow weight update of square loss function and assess the discrepancy between ground truth and predicted logits, cross-entropy loss function was selected as the loss function [39]. The generated loss can also be utilized to update the model parameters as well as to evaluate the performance of crack detection [40].

Cross-entropy loss function measures the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss increases as the predicted probability diverges from the actual label [24]. For example, a probability of 0.015 when the actual label is 1 would be bad and result in a high loss value. A perfect model would have a log loss of 0. When talking about the cross-entropy loss function, its formula immediately comes to mind, and its formula can be shown in

$$L_{ce} = -[y \log \hat{y} + (1 - y) \log (1 - \hat{y})], \tag{2}$$

Table 2: Comparisons of different methods.

Method	Author	Size of image	Number of dataset for training and validation	Precision	Recall	$F1$	mIoU
U-net	Liu et al. (2019)	512×512	10000	91%	91%	90%	61.28%
Sliding window Method	Cha et al. (2017)	512×512	10000	85%	82%	83%	46.54%
FCN	Yang et al. (2018)	512×512	10000	82%	79%	80%	57.43%
Modified FCN (no data processing)	Meng et al.	512×512	10000	96.79%	92%	90%	71.12%
FCN (with data processing)	Meng et al.	512×512	10000	85%	81%	82%	59.36%
Modified FCN	Meng et al.	512×512	10000	97.92%	92%	91%	80.24%

where $y \epsilon \{0, 1\}$ denotes the truth value class probability and $y \epsilon [0, 1]$ denotes the prediction class probability.

The sample imbalance usually exists in the dataset. Sample imbalance is a problem that many machine learning meets. If a sample of a certain class in a train set occupies most of the proportion, it is called a simple sample. Because of the large number of simple samples, the contribution to the loss of the entire train set will be very large, leading to the model not being well trained or fully trained. It may be that the loss function got stuck in a worse local optimum. There will be a relatively large "uphill" between the best local optimum and the better global optimum. Therefore, the loss function cannot converge to a best result during the training process. To solve the imbalance problems, this paper introduced a weighting factor and an adjustable focusing parameter to regulate the cross-entropy function. And this function formula can be shown in formula (3): in this equation, γ will be set as 0.25 and β will be set as 2.

$$L_{fl} = -[y\beta(1 - y\wedge)^{y} \log \widehat{y} + (1 - y)(1 - \beta)y\wedge^{y} \log (1 - \widehat{y})]. \quad (3)$$

4.3. Optimizer. This article uses the Adam algorithm to optimize the model. The Adam algorithm combines both the momentum algorithm and the RMS (rate-monotonic scheduling) prop algorithm [41]. It is also based on the gradient descent method, but the Adam algorithm has a certain range of parameter changes during each iteration. The parameter will not change sharply due to the large gradient value calculated at a certain time, and the value of the parameter is relatively stable. According to the method proposed by Smith in 2017 [42], first set a very small learning rate, such as 0.00001, update the network after each batch, increase the learning rate with 10 as the index, and count the loss calculated by each batch. Then, the optimal learning rate was determines according to it. The initial optimal learning rate here is set as 0.0001 after adjustments. Weight decaying is a form of regularization, which plays an important role in training, so it needs to be set appropriately. Weight decaying is defined as multiplying each weight in the gradient descent for each period by a factor λ $(0 < \lambda < 1)$. According to the experience, the weight decaying value which can be selected for testing is 0.001, 0.0001, 0.00001, and 0. Larger weight decaying values are set for smaller datasets and model structures, while smaller values are set for larger datasets and deeper model structures. Considering the size of the dataset used in this study and the test results, the weight decaying value was set as 0. Meanwhile, about the momentum value ranging from 0.9 to apprpoximately 0.99 is suitable to be selected according to the related research. So the initial learning rate of the Adam algorithm in this paper is $\eta = 0.0001$. The exponential decay rate of the first moment estimate is $\beta_1 = 0.9$.

The Adam updated formula is given as the following:

$$m_t = \beta_1 m_{t-1} + (1 - \beta) g_t, \quad (4)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \quad (5)$$

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t}, \quad (6)$$

$$\widehat{v}_t = \frac{v_t}{1 - \beta_2^t}, \quad (7)$$

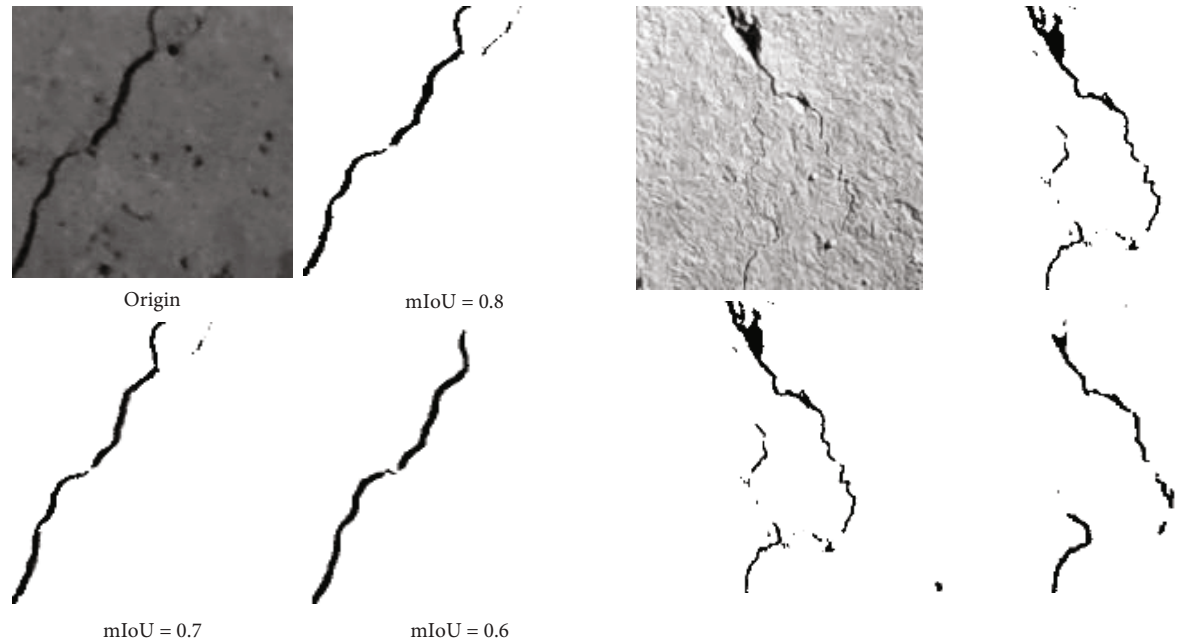
$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\widehat{v}_t} + \varepsilon} \widehat{m}_t, \quad (8)$$

where β_1 will be set as 0.9, β_2 will be set as 0.999 and ε will be set as 1×10^{-8}, ϵ is to prevent division by 0, and g_t denotes the gradient.

Formulas (4) and (5) represent a moving average of the gradient and the square of the gradient so that each update is related to the historical value. Formulas (6) and (7) denote a correction for the larger initial moving average deviation which is called the bias correction. Formula (8) is the parameter update formula.

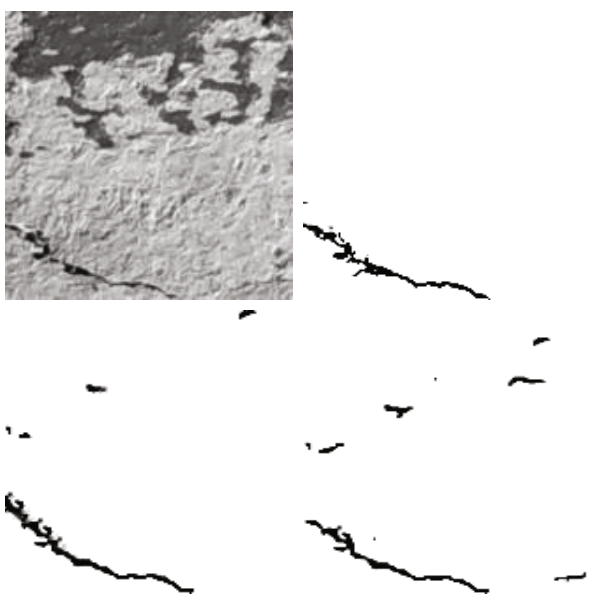
4.4. Training Process. To save training time and generalize the training process, in the downsampling, part of the parameters in convolutional layers are initialized from pretrained VGG19 weights; for the upsampling part, the filters are initialized by truncated normal distribution with mean of zero and standard deviation of 0.01, and the biases are initialized with constant zero vectors; the keep probability for dropout layers is determined to be 0.4.

As a whole, the deep learning framework of Keras with TensorFlow is applied as a backbone to carry out the training, validation, and prediction of the proposed modified fully convolutional network on a single GPU of NVIDIA GTX1080Ti. Several rounds of adjustment for

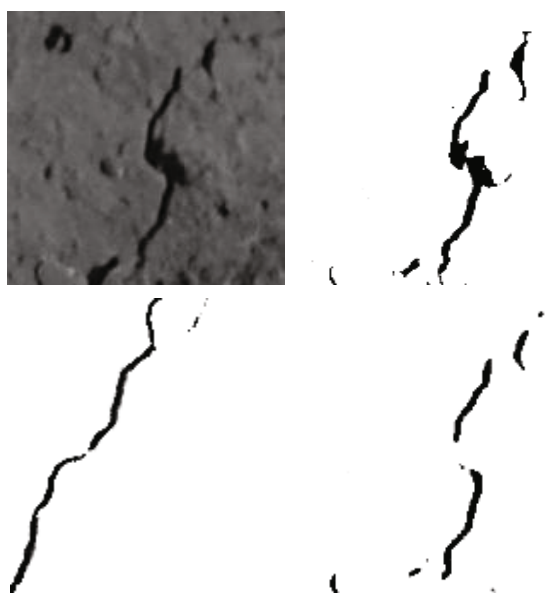


(a) Omission of judgment

(b) Omission of judgment

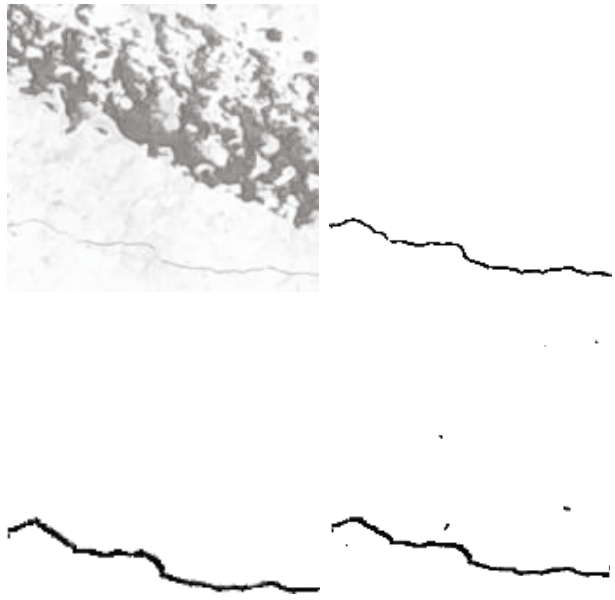


(c) Misjudgment



(d) Insufficient depth

Figure 5: Continued.



(e) Fair

Figure 5: (a–e) Comparison situation of crack damage identification.

hyperparameters and training are carried out for each model to maximize the convergence of loss function to the global optimum. An average of 30 training epochs was used, with 2000 training rounds per epoch. Different from the learning rate, the value of batch size does not affect the calculation of training time. The batch size is limited by the hardware storage, while the learning rate is not. The set of batch size needs to take into account the learning rate and GPU computing power. Generally speaking, the learning rate is directly proportional to the batch size. Considering the hardware storage and GPU computing power of this computer, the value was set as 2. Models during the training process are autosaved after every epoch with monitoring of the minimum loss value. The accuracy of the model was then verified on the test set, and the model with the highest accuracy was saved as the final model.

The output of the modified fully convolutional network is a probability map with pixel values that range from 0 to 1, in which a black color on the white background indicates that the pixel is more likely to be a crack pixel. The threshold of the probability map is set to 0.5 to obtain a binarized crack segmentation image.

It is common to use K-fold cross-validation to evaluate the machine learning models; this paper use $k = 3$ in this article. The total over 20000 sample images are randomly split into 3 groups numbering from 0 to 2. Each time, two groups are selected as the training set and the remaining one group as the validation set.

Each image in the test set is processed within 111 ms, while in the U-net model proposed by Liu et al., the processing time for each test image is 126 ms. As for the efficiency of the model proposed in this article, frame per second can be applied to evaluate it. Through the experiment, our algorithm is about 12 fps on the GPU side, which is higher than the U-net model proposed by Liu et al. The latter shows 11 fps on the GPU side, and our model achieves the improvement nearly 12% in efficiency.

4.5. Accuracy Evaluation. To evaluate and assess the accuracy in the semantic segmentation task, several metrics are commonly used. They are given in formulas (9)–(14), including pixel accuracy (PA), intersection over union (IoU), mean intersection of union (mIoU), precision, recall, and $F1$ score [43, 44].

For the crack detection task in this paper, the commonly used evaluation indexes are mIoU, precision (P), recall (R), and $F1$. The crack pixels are defined as positive instances. According to combinations of labeled case and $F1$, The crack pixels are defined as positive instances. According to combinations of labeled case and predicted case, pixels are divided into four types: true positive (TP), false positive (FP), true negative (TN), and false negative (FN) [40, 45, 46]. The confusion matrix is usually used to evaluate the model. The confusion matrix is a situation analysis table that summarizes the prediction results of classification model in machine learning. The records in the dataset are summarized in the form of matrix according to the real category and the category judgment predicted by the classification model. The row of the matrix represents the real value, and the column of the matrix represents the predicted value. Table 1 shows a sample of the confusion matrix. Figure 3 gives the evaluation result of our model.

$$\mathrm{PA} = \frac{\sum_{i=0}^{K} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}}, \tag{9}$$

$$\mathrm{IoU} = \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}, \tag{10}$$

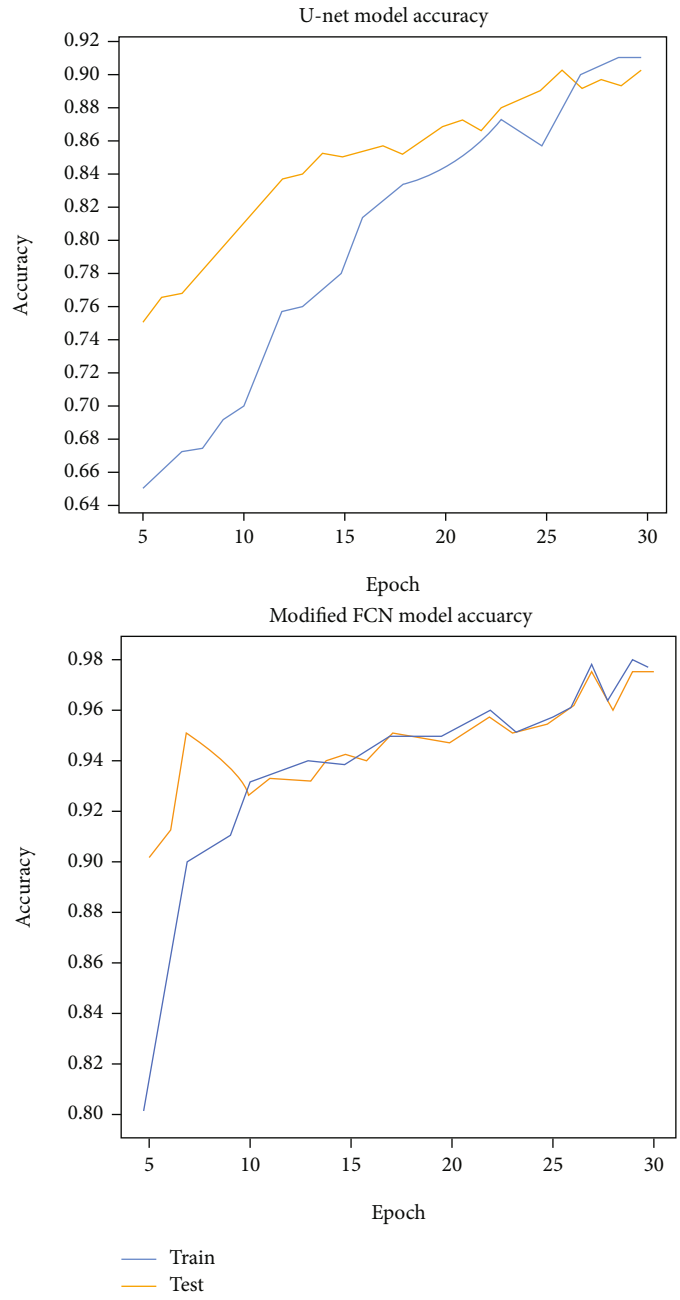


Figure 6: Curve of training and detection accuracy of two models with epoch (U-net and modified FCN).

$$\text{mIoU} = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k}p_{ij} + \sum_{j=0}^{k}p_{ji} - p_{ii}}, \tag{11}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{12}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{13}$$

$$F1 = \frac{2 \times P \times R}{P + R} = \frac{2 \times TP}{2 \times \text{TP} + \text{FP} + \text{FN}}. \tag{14}$$

The total class number including background is $k + 1$, and P_{ij} represents the number of pixels of class i inferred to belong to class j. So, P_{ii}, P_{jj}, and P_{ji} represent the pixel number of true positives (TP), false positives (FP), and false negatives (FN), respectively. P and R in $F1$ represent precision and recall, respectively. PA is the ratio of the number of pixels with correct prediction category to the total number of pixels [47]. $F1$ is a harmonic mean value. IoU is the result obtained by dividing the overlapped part of two regions by the set part of the two regions [48].

PA is the simplest evaluation metric, which calculates the ratio between the correctly classified pixels and the total number of pixels in an image. IoU is a standard metric which has the function to assess the performance in semantic segmentation tasks.

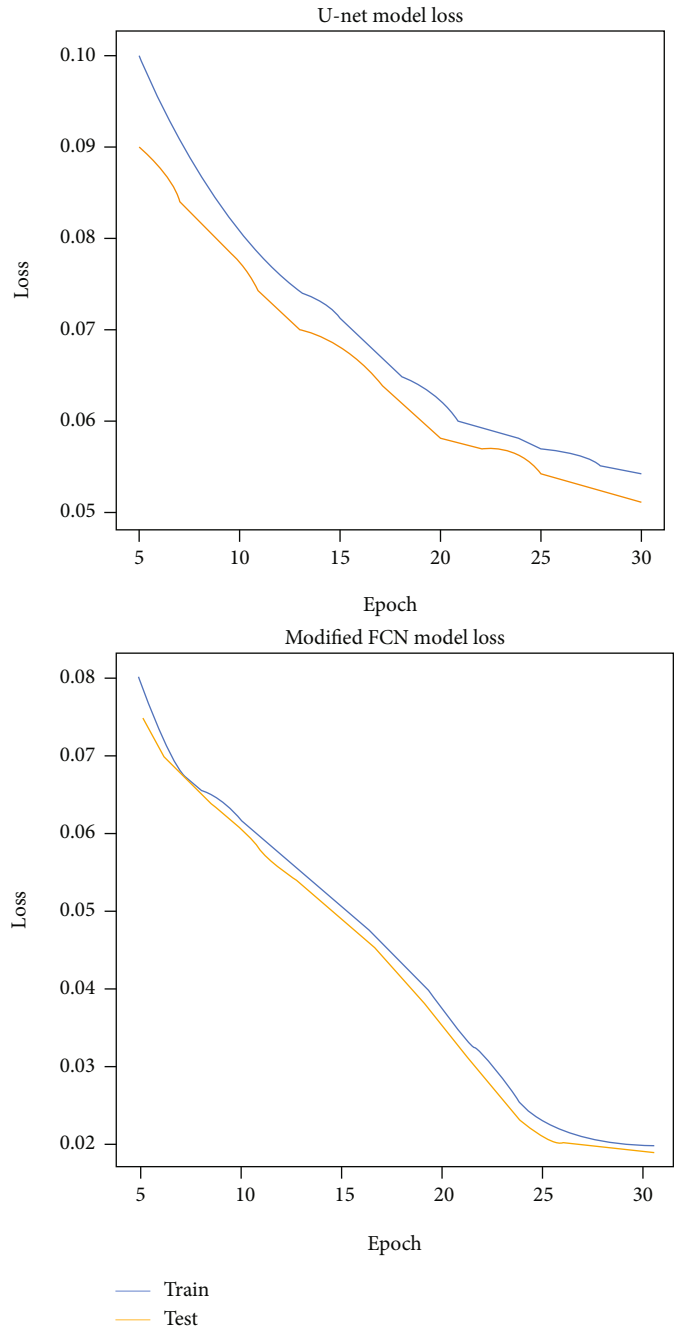


Figure 7: Curve of training and detection loss of two models with epoch (U-net and modified FCN).

5. Results and Comparison

5.1. Heat Map. To achieve better results, a most intuitive method to show the process of classification and learning of the crack damage in an artificial machine is used in this article. That is, it can easily visualize the recognition process with the help of thermal map. In the deep learning process, the heat map is to extract the weights of all classes, and then, the convolution layer will keep forward to find the corresponding feature map to carry out the weighted sum. Generally speaking, the heat map tells us which pixels the model uses to know which kind of image (cracked or uncracked) it belongs to. Different training models have requirements for the size of the output image. Figure 4 gives several samples of heat map in our model. It vividly shows the cracked and uncracked type.

5.2. Comparison of Crack Detection Accuracy between Different Models. This paper compared the accuracy of the modified fully convolutional network proposed in this article and the U-net proposed by Liu et al. in [9] for crack or damage identification using the same test set. U-net was used as a baseline, while the modified fully convolutional network was evaluated with multiple modules such as the backbone

network including the batch normalization layer and the convolutional layer, it also contains the spatial pyramid average pooling, spatial pyramid max pooling, deconvolutional layer, dilated convolutions, and a softmax layer. For the modified fully convolutional network, focal loss is applied during the training with the final model. The Adam optimizer is also used to update the weights. Each network includes one new module compared to the previous network and then is retrained to compare their accuracy and running time. Evaluation metrics of PA, IoU, precision, recall, and $F1$ score for different methods were calculated as listed in Table 2. Identification results of different methods are presented in Figures 5(a)–5(e). In order to make the recognition results more clearly, this paper refer to the case of mIoU = 0.8.

As a quite practical drawing library, Matplotlib is also used to draw the curve of training and detection accuracy of two models with epochs (Figure 6). Figure 7 depicts the change in loss function value during the training process between two different models.

According to the running results of FCN [9] and FCN with data preprocessing, the image preprocessing (including graying, binarization, and threshold segmentation) for the raw dataset has a certain improvement on the overall recognition accuracy of the model, but the promotion range is relatively small. Compared with the conventional FCN, the modified FCN with innovative structure has significantly improved the recognition accuracy and recall. The modified FCN model of using image preprocessing technology has the highest index among all the above methods, the precision, recall, and F1 reach 97.92%, 92%, and 91%, respectively, which also verifies the feasibility and superiority of our method for structural crack extraction and damage identification.

6. Conclusions

This research concentrates on the method applying convolutional network as well as the computer vision technology to identify or detect the concrete cracks in the architecture. According to the characteristics of concrete cracks, it is essentially classified as a semantic segmentation problem in computer vision, and the modified fully convolutional network structure is used to build a deep learning model for crack detection. The performance of the concrete crack detection method based on the modified fully convolutional network structure was tested and compared with U-net proposed by Liu et al. and Cha's CNN. From the data and figure, it can be concluded that modified fully convolutional network will be more elegant than U-net or other conventional DCNN methods with more robustness, more effectiveness, and more accuracy. This paper also examines the fundamental parameters during the performance of the method; the modified fully convolutional network proposed by us is found to obtain high accuracy and high efficiency with enough dataset than the previous.

Considering this method uses image processing technology, the camera ought to have the capability to obtain a clear field of view for cracks. This method is only applicable for concrete cracks. Thus, its applicability to the inspection of other cracked engineering materials may be restricted. Besides, due to the limitations of acquisition equipment and acquisition environment, it is difficult to capture the microcracks or damages on some structural surfaces, and these microinformation often has very important guiding significance for obtaining the service state and failure mechanism of structures. In the future, the application research of the deep learning algorithm model suitable for structural microcrack detection can be carried out. On the other hand, the accurate crack detection in the proposed model is able to actuate the multidimensional data fusion between the image and other high-precision data such as radar, topology instrument, and laser scanner, compensate the defects of data types collected by a single sensor, and promote the quantitative detection and intelligent management of structural damage information.

Data Availability

The data presented in this study are available in a public repository (link: https://1drv.ms/u/s!Araj_Ctb9II1iUDI097Wrqs0_Tt?e=o0EhV2).

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

Conceptualization was done by Meng Meng and Kun Zhu. Methodology was done by Meng Meng. Software was acquired by Kun Zhu. Validation was conducted by Hang Qu, Kun Zhu, and Meng Meng. Formal analysis was performed by Meng Meng. Investigation was conducted by Kun Zhu. Resources were acquired by Keqin Chen. Data curation was performed Hang Qu. Writing—original draft preparation was done by Meng Meng. Writing—review and editing was done by Kun Zhu and Meng Meng. Visualization was conducted by Meng Meng. Supervision was done by Keqin Chen. Project administration was done by Keqin Chen. Funding acquisition was done by Keqin Chen.

Acknowledgments

The authors would like to thank the dataset and the funding funded by Meng Meng from Southeast University and Hang Qu from Yangzhou University.

References

[1] M. Z. A. Bhuiyan, G. Wang, J. Wu, J. Cao, X. Liu, and T. Wang, "Dependable structural health monitoring using wireless sensor networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 4, pp. 363–376, 2017.

[2] K. Chintalapudi, T. Fu, J. Paek et al., "Monitoring civil structures with a wireless sensor network," *IEEE Internet Computing*, vol. 10, no. 2, pp. 26–34, 2006.

[3] R. N. Soman, T. Onoufrioua, M. A. Kyriakidesb, R. A. Votsisc, and C. Z. Chrysostomou, "Multi-type, multi-sensor placement

optimization for structural health monitoring of long span bridges," *Smart Structures and Systems*, vol. 14, no. 1, pp. 55–70, 2014.

[4] M. Abdulkarem, K. Samsudin, F. Z. Rokhani, and M. F. A Rasid, "Wireless sensor network for structural health monitoring: a contemporary review of technologies, challenges, and future direction," *Structural Health Monitoring*, vol. 19, no. 3, pp. 693–735, 2020.

[5] Y. Yu, C. Wang, X. Gu, and J. Li, "A novel deep learning-based method for damage identification of smart building structures," *Structural Health Monitoring*, vol. 18, no. 1, pp. 143–163, 2019.

[6] T. Yin and H. P. Zhu, "Probabilistic damage detection of a steel truss bridge model by optimally designed Bayesian neural network," *Sensors*, vol. 18, no. 10, p. 3371, 2018.

[7] Y. J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361–378, 2017.

[8] K. Chen, A. Yadav, A. Khan, Y. Meng, and K. Zhu, "Improved crack detection and recognition based on convolutional neural network," *Modelling and Simulation in Engineering*, vol. 2019, 8 pages, 2019.

[9] Z. Liu, Y. Cao, Y. Wang, and W. Wang, "Computer vision-based concrete crack detection using U-net fully convolutional networks," *Automation in Construction*, vol. 104, pp. 129–139, 2019.

[10] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *In Digital Image Processing Using MATLAB, 1nd ed*, Pearson Education Press, India, 2009.

[11] Y. Wu, Y. Qin, Z. Wang, and L. Jia, "A UAV-based visual inspection method for rail surface defects," *Applied Sciences*, vol. 8, no. 7, p. 1028, 2018.

[12] W. S. Na and J. Baek, "Impedance-based non-destructive testing method combined with unmanned aerial vehicle for structural health monitoring of civil infrastructures," *Applied Sciences*, vol. 7, no. 1, p. 15, 2017.

[13] C. V. Dung, H. Sekiya, S. Hirano, T. Okatani, and C. Miki, "A vision-based method for crack detection in gusset plate welded joints of steel bridges using deep convolutional neural networks," *Automation in Construction*, vol. 102, pp. 217–229, 2019.

[14] S. Dorafshan, R. J. Thomas, and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image- based crack detection in concrete," *Construction and Building Materials*, vol. 186, pp. 1031–1045, 2018.

[15] F. Nagata, K. Tokuno, K. Mitarai et al., "Defect detection method using deep convolutional neural network, support vector machine and template matching techniques," *Artificial Life and Robotics*, vol. 24, no. 4, pp. 512–519, 2019.

[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 3431–3440, Boston, USA, June 2019.

[17] F. C. Chen and M. R. Jahanshahi, "NB-FCN: real-time accurate crack detection in inspection videos using deep fully convolutional network and parametric data fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 8, pp. 5325–5334, 2020.

[18] L. Attard, C. J. Debono, G. Valentino, M. Di Castro, A. Masi, and L. Scibile, "Automatic crack detection using mask R-CNN," in *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 152–157, Dubrovnik, Croatia, July 2019.

[19] M. Saif Hassan Onim, A. Rafeed Ehtesham, A. Anbar, A. K. M. Nazrul Islam, and A. K. M. Mahbubur Rahman, "LULC classification by semantic segmentation of satellite images using FastFCN," http://arxiv.org/abs/2011.06825.

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

[21] W. Choi and Y. J. Cha, "SDDNet: real-time crack segmentation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 9, pp. 8016–8025, 2020.

[22] D. Kang, S. S. Benipal, D. L. Gopal, and Y. J. Cha, "Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning," *Automation in Construction*, vol. 118, article 103291, 2020.

[23] Z. Tong, J. Gao, and H. Zhang, "Recognition, location, measurement, and 3D reconstruction of concealed cracks using convolutional neural networks," *Construction and Building Materials*, vol. 146, pp. 775–787, 2017.

[24] Y. H. Kim and J. R. Lee, "Videoscope-based inspection of turbofan engine blades using convolutional neural networks and image processing," *Structural Health Monitoring*, vol. 18, no. 5-6, pp. 2020–2039, 2019.

[25] F. Ni, J. Zhang, and Z. Chen, "Pixel-level crack delineation in images with convolutional feature fusion," *Structural Control and Health Monitoring*, vol. 26, no. 1, article e2286, 2019.

[26] B. Kim and S. Cho, "Image-based concrete crack assessment using mask and region-based convolutional neural network," *Structural Control and Health Monitoring*, vol. 26, no. 8, article e2381, 2019.

[27] A. Ji, X. Xue, Y. Wang, X. Luo, and W. Xue, "An integrated approach to automatic pixel-level crack detection and quantification of asphalt pavement," *Automation in Construction*, vol. 114, article 103176, 2020.

[28] B. Kim and S. Cho, "Automated vision-based detection of cracks on concrete surfaces using a deep learning technique," *Sensors*, vol. 18, no. 10, p. 3452, 2018.

[29] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention (MICCAI)*, pp. 234–241, Munich, Germany, October 2015.

[30] S. Tammina, "Transfer learning using VGG-16 with deep convolutional neural network for classifying images," *International Journal of Scientific and Research Publications*, vol. 9, no. 10, pp. 143–150, 2019.

[31] C. W. Zhang, M. Y. Yang, H. J. Zeng, and J. P. Wen, "Pedestrian detection based on improved LeNet-5 convolutional neural network," *Journal of Algorithms & Computational Technology*, vol. 13, article 1748302619873601, 2019.

[32] J. E. Espinosa, S. A. Velastin, and J. W. Branch, "Vehicle detection using Alex Net and faster R-CNN deep learning models: a comparative study," *International visual informatics conference*, 2017, pp. 3–15, Bangi, Malaysia, November 2017.

[33] W. Ma, Z. Pan, J. Guo, and B. Lei, "Achieving super-resolution remote sensing images via the wavelet transform combined

with the recursive Res-Net," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 6, pp. 3512–3527, 2019.

[34] W. Xu, G. Xu, Y. Wang, X. Sun, D. Lin, and Y. Wu, "Deep memory connected neural network for optical remote sensing image restoration," *Remote Sensing*, vol. 10, no. 12, p. 1893, 2018.

[35] Y. Xu, Y. Bao, J. Chen, W. Zuo, and H. Li, "Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images," *Structural Health Monitoring*, vol. 18, no. 3, pp. 653–674, 2019.

[36] L. Wang, C. Wang, Z. Sun, and S. Chen, "An improved dice loss for pneumothorax segmentation by mining the information of negative areas," *IEEE Access*, vol. 8, pp. 167939–167949, 2020.

[37] H. Kervadec, J. Bouchtiba, C. Desrosiers, E. Granger, J. Dolz, and I. B. Ayed, "Boundary loss for highly unbalanced segmentation," in *International conference on medical imaging with deep learning*, pp. 285–296, London, UK, July 2019.

[38] M. Berman, A. Rannen Triki, and M. B. Blaschko, "The Lovász-softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4413–4421, Salt Lake City, USA, June 2018.

[39] Y. Xu, S. Wei, Y. Bao, and H. Li, "Automatic seismic damage identification of reinforced concrete columns from images by a region-based deep convolutional neural network," *Structural Control and Health Monitoring*, vol. 26, no. 3, article e2313, 2019.

[40] Y. Ren, J. Huang, Z. Hong et al., "Image-based concrete crack detection in tunnels using deep fully convolutional networks," *Construction and Building Materials*, vol. 234, article 117367, 2020.

[41] A. M. Taqi, A. Awad, F. Al-Azzo, and M. Milanova, "The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 140–145, Miami, USA, April 2018.

[42] L. N. Smith, "Cyclical learning rates for training neural networks," in *2017 IEEE winter conference on applications of computer vision (WACV)*, pp. 464–472, Santa Rosa, CA, USA, 2017.

[43] S. Cofre-Martel, P. Kobrich, E. Lopez Droguett, and V. Meruane, "Deep convolutional neural network-based structural damage localization and quantification using transmissibility data," *Shock and Vibration*, vol. 2019, Article ID 9859281, 2019.

[44] C. M. Yeum, J. Choi, and S. J. Dyke, "Automated region-of-interest localization and classification for vision-based visual assessment of civil infrastructure," *Structural Health Monitoring*, vol. 18, no. 3, pp. 675–689, 2019.

[45] Y. Dong, C. Su, P. Qiao, and L. Sun, "Microstructural crack segmentation of three-dimensional concrete images based on deep convolutional neural networks," *Construction and Building Materials*, vol. 253, article 119185, 2020.

[46] D. W. Abueidda, M. Almasri, R. Ammourah, U. Ravaioli, I. M. Jasiuk, and N. A. Sobh, "Prediction and optimization of mechanical properties of composites using convolutional neural networks," *Composite Structures*, vol. 227, article 111264, 2019.

[47] R. Fu, H. Xu, Z. Wang et al., "Enhanced intelligent identification of concrete cracks using multi-layered image preprocessing-aided convolutional neural networks," *Sensors*, vol. 20, no. 7, p. 2021, 2020.

[48] K. Jang, N. Kim, and Y. K. An, "Deep learning–based autonomous concrete crack evaluation through hybrid image scanning," *Structural Health Monitoring*, vol. 18, no. 5-6, pp. 1722–1737, 2019.