Autonomous Bridge Crack Detection Using Deep Convolutional Neural Networks

Hongyan Xu [a], Xiu Su [b], Huaiyuan Xu [c] and Haotian Li [d]

School of Jingyi, Tianjin University, Tianjin 300072, China.

[a]tjdxxhy@tju.edu.cn, [b] suxiu@tju.edu.cn, [c] hyxu@tju.edu.cn, [d]lihaotian@tju.edu.cn

Abstract. Traditional image processing algorithms have a lot of limitations when dealing with crack detection problems. And the effect is not ideal if the classical deep learning model were used to detect bridge cracks directly. In order to solve these problems, a CNN-based bridge crack detection method is proposed in this paper, in which a feature extraction module based on arous space pyramid pool (ASPP) and depthwise separable convolution is designed. The former can obtain multi-scale image feature information, and the atrous convolution can provide a larger receptive field, so large-scale contextual information can be fused more effectively on feature maps. The latter can significantly reduce the computational complexity of the model and improve computational efficiency. The experimental results show that the method proposed in this paper achieved a crack detection accuracy of 96.69%, which is approximately 10% higher than other similar methods.

Keywords: Deep learning; convolution neural networks; bridge crack detection.

1. Introduction

Bridges play an important role in daily life as a kind of transportation infrastructure. According to statistics, 90% of the bridge damage that occurs each year is caused by cracks. Therefore, detecting the condition of concrete cracks is critical to the safe use of bridges. The traditional detection method is mainly based on manual visual inspection, which not only has low detection efficiency, but also is difficult to guarantee the quality of inspection. In recent years, machine learning and computer vision technology have been successfully applied in the field of automatic crack detection [1–5], and have attracted more and more people's attention.

Traditional crack detection methods are based on digital image processing techniques. Abdel-Qader et al. [6] proposed in 2003 to use four edge detection method to find concrete cracks. Subirats P et al. [7] proposed a method for extracting image cracks by establishing a multi-scale two-dimensional wavelet transform in 2006. Li et al. [8] proposed a FoSA algorithm in 2011 that used F* seed growth to connect crack target points as seed points to obtain cracks. Q Zou et al. [4] proposed a method combining seed point detection and tensor voting to obtain higher detection accuracy, but cracks may break. In 2015, Guan et al. [9] proposed the ITV-crack algorithm for detecting cracks using iterative tensor voting based on Zou's method. As actual images often have large noise interference, these image processing methods cannot accurately identify cracks.

The concept of deep learning was proposed by G Hinton et al. [10] in 2006. Two viewpoints are put forward in the thesis: (1) The multi-layer artificial neural network model has strong feature learning ability, and the feature data obtained by the deep learning model has more essential representation of the original data, which will greatly facilitate classification and visualization. (2) For deep neural networks, it is difficult to achieve optimal while training, and this problem can be solved by layer-by-layer training. Later, SERMANET et al. [11] proposed Convolutional Neural Network (CNN), which uses image space information to reduce the number of training parameters, thus greatly improving the model training performance.

Deep learning can learn a deep nonlinear network structure, characterize input data, implement complex function approximation, and have a powerful ability to learn the essential features of data sets from a small sample set. Many computer vision tasks fully demonstrate the effectiveness of deep features extracted by deep neural networks. Deep learning, especially the Convolutional Neural Network (CNN) in deep learning, has achieved great success in image segmentation, video recognition, speech recognition, target detection, etc. [12-16] computer vision tasks. But deep learning has not been applied to the field of crack detection until 2016, L Zhang et al. [17] used

Convolutional Neural Network (CNN) for crack detection for the first time. However, since the method did not use strict criteria for positive and negative training samples, the detection accuracy was not high.

In this paper, we construct a concrete crack detection classifier based on CNN. The proposed method has the following contributions:

- The classifier we built is less affected by noise such as light, water stains and background shading, so it has good robustness.
- A defect detection model has been established in this paper, which can be widely used to detect other types of structural damage, such as delamination, voiding, spalling and corrosion, and has wide adaptability.
- We propose a new network feature extraction method based on ASPP module and depth separable convolution, which can be combined with the network to improve network performance and extract better image features.

2. Proposed Method

Bridge crack detection is essentially a classification task, which aims to judge whether a given picture contains cracks. Characterizing the image is the main research content of the task. To solve this problem, our proposed method is based on a convolutional neural network to train the pictures with given ground truth tag. This section describes the implementation of the algorithm and the overall structure of the network.

2.1 Data Preparaten

According to [18], there is no unified bridge crack database in academia. Therefore, in this work, the bridge crack data set in [18] is combined and preprocessed to meet the needs of convolutional neural network training. The total number of original images is 2068, which was acquired by the CMOS area array camera that comes with the DJI Phantom 4 pro. The size of each original image is 1024 × 1024, and the final data set is generated from the original image by the following steps:

1. Since the original images all contain cracks, which is not conducive to our classification task. Therefore, each original image is cropped to a size of 512×512, and after the blurred image is removed, 6069 images are finally obtained as our data set. It contains 4058 crack pictures, 2011 background pictures, and the rate of crack pictures to background pictures is about 2:1. In the end, we used 4,856 images as our training set and 1213 images as our test set.
2. These images are further random cropped to 224 x 224 to meet the input needs of the Resnet50 network. At the same time, a network trained on a relatively small picture can scan any image larger than the design size [19].

The resulting data set includes a wide range of image changes for generating a powerful crack classifier, as shown in Figure 1.

(a) (b) (c)

(a) fine images, (b) background shading images, and (c) strong light spotted images.

Fig. 1 Examples of images used in training. The presented images have 224 × 224-pixel resolutions:

As shown in Figure 2, our dataset also contains cropped images of crack locations at the four edges of the image space. In these pictures, the crack area only accounts for a small part of the picture. After passing through the CNN, the picture size is further reduced, so the edge crack is more difficult to be recognized by the network than the center crack. The use of such images can increase the

generalization of the network we train, because the proportion of cracked areas in the actual detection is usually small.

Fig. 2 Pictures of crack at four edges

2.2 Hyperparameters

The neural network we use is trained using the Momentum optimization algorithm. As small and decreasing learning rates are recommended [20], the network applies a mathematically reduced learning rate. There are 32 samples per batch during training, the momentum is set to 0.9, the initial learning rate is set to 0.005, and the weight decay is 0.2.

2.3 Overall Network Structure

The network structure we designed is shown in Figure 3, and the structure of the ASPP module is shown in Figure 5. The network uses Resnet50 [21] as the backbone. In this work, we developed a convolutional network module in which Atrous Spatial Pyramid Pooling (or ASPP) was applied to capture multi-scale context information. At the same time, a depthwise separable convolution is applied after each 3×3 convolutional layer of the ASPP module to reduce the computational loss and the number of parameters while maintaining the accuracy of the results. Our module can be inserted anywhere in the convolutional neural network model, and it can reuse the underlying image feature information.

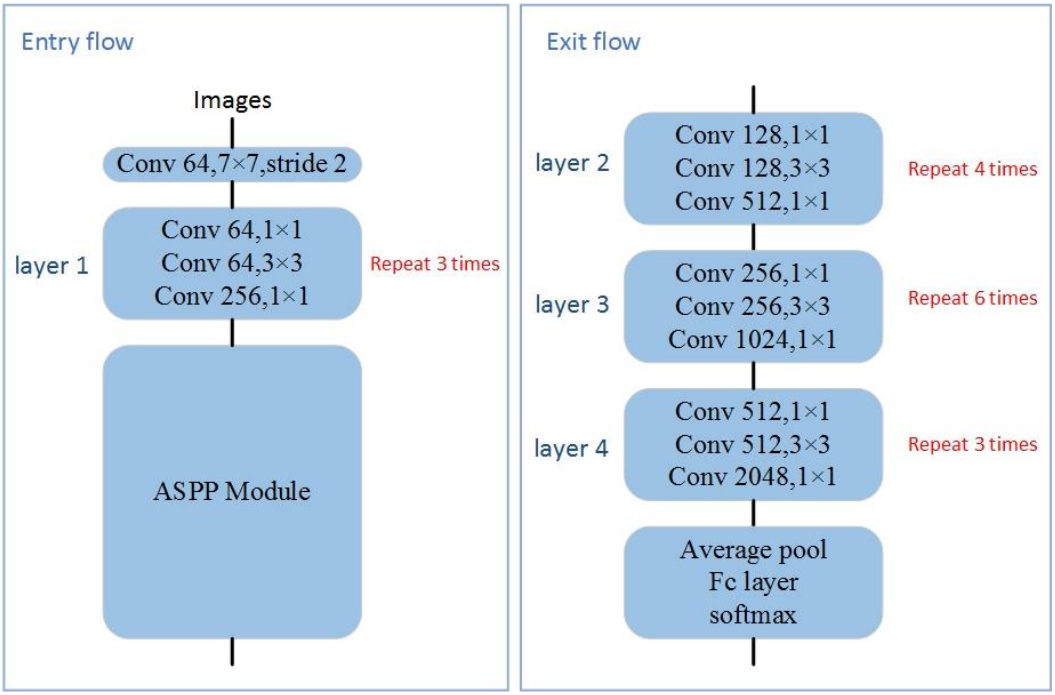


Fig. 3 Illustration of the architecture of the proposed ConvNet.

2.3.1 Atrous Convolution

Modern image classification networks integrate multi-scale context information through continuous pooling and sub-sampling layers, resulting in loss of detail information about object edges and degradation of image resolution ([22],[23]). Since the crack detection task focuses on the crack edge information in the image, we introduce an atrous convolution in the network structure. Compared to traditional convolution, atrous convolution (or dilated convolution) can provide a larger receptive field with a comparable amount of computation, thus enabling the extraction of more dense feature maps [24].

Let $F: Z^2 \to R$ denote a discrete function. Let $\Omega_r = \left[-r, r^2\right] \cap Z^2$ and let $k: \Omega_r \to R$ be a discrete filter of size. Then the discrete convolution operator $*$ can be defined as

$$(F * k)(p) = \sum_{s+t-p} F(s)k(t). \tag{1}$$

We then generalize this operator. Let l be a dilation factor and let $*_l$ be defined as

$$(F *_l k)(p) = \sum_{s+lt-p} F(s)k(t). \tag{2}$$

Here $*_l$ is an atrous convolution or l-atrous convolution. The familiar discrete convolution is simply the 1-atrous convolution.

Atrous convolution enables exponential expansion of the receptive field without loss of resolution and coverage [24]. Let $F_0, F_1, ..., F_{n-1}: Z^2 \to R$ be a discrete function and let $k_0, k_1, ..., k_{n-2}: \Omega_1 \to R$ be a discrete 3×3 filter. Consider applying a cavity filter with an exponential increase:

$$F_{I+1} = F_i *_{2^i} k_i \quad i = 0,1, \ldots, n-2. \tag{3}$$

The receptive field of the element pin F_{i+1} is defined as the set of elements in F_0 that modify the value of F_{i+1}. Assume that the size of the receptive field of p in F_{i+1} is the number of these elements. Then the receptive field size of each element in F_{i+1} is $\left(2^{i+2}-1\right) \times \left(2^{i+2}-1\right)$. The receptive field is an exponentially increasing square, as shown in Figure 4.

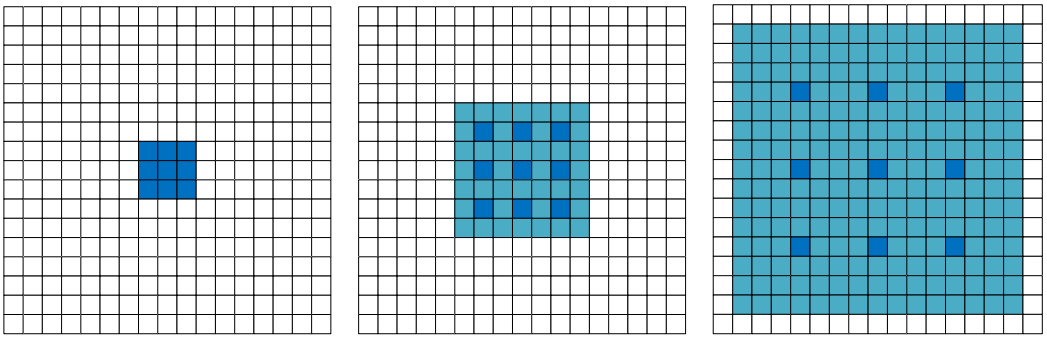


Fig. 4 Schematic diagram of atrous convolution.

2.3.2 ASPP Module

For bridge crack images, cracks usually only occupy a small portion of the picture. Therefore, in order to accurately identify cracks in the image, it is necessary to accurately extract the feature information of the crack, so in this work, the atrous spatial pyramid pooling (ASPP) module is used to obtain multi-scale crack characteristic information.

ASPP is part of the DeepLabv2[25] network proposed by the google team in 2017. Inspired by Spatial Pyramid Pooling (SPP) [26], it is parallel sampling with different sampling rates of atrous convolution on a given input, which is equivalent to capturing the context of the image in multiple scales, thereby obtaining multi-scale image feature information [27]. It has several notable features for deep convolutional neural networks: (1) ASPP uses multi-level spatial sampling, and [28] has

proven multi-layer pooling can effectively cope with object deformation; (2) Due to the flexibility of the input rate, ASPP can collect features extracted in variable scale. These factors can improve the recognition accuracy of deep neural networks.

A schematic diagram of the ASPP module used is shown in Figure 5. Three convolutional layers with different atrous rates were used, with atrous rates of 2, 4, and 8, respectively. In order to generate the result, the features extracted at each sampling rate are processed by separate branches, and the feature maps are bilinearly interpolated from the parallel network branches to the original images, and they are fused to obtain the maximum response of the location at different scale. Multi-scale processing significantly enhances performance, but requires computation of the feature response on all depth convolutional neural network layers for multiple dimensions of the input image.

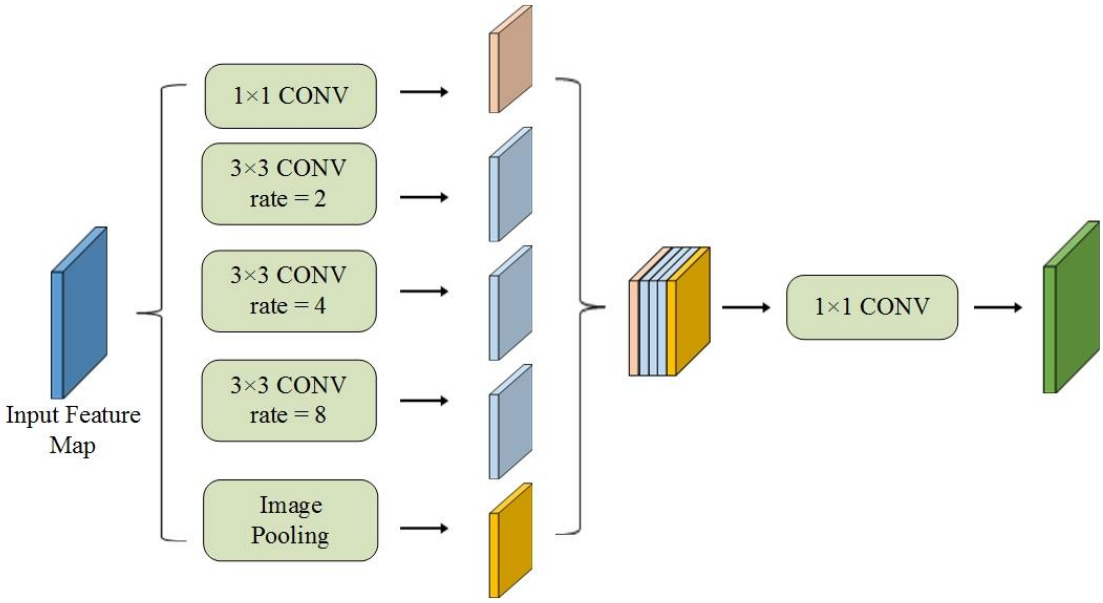
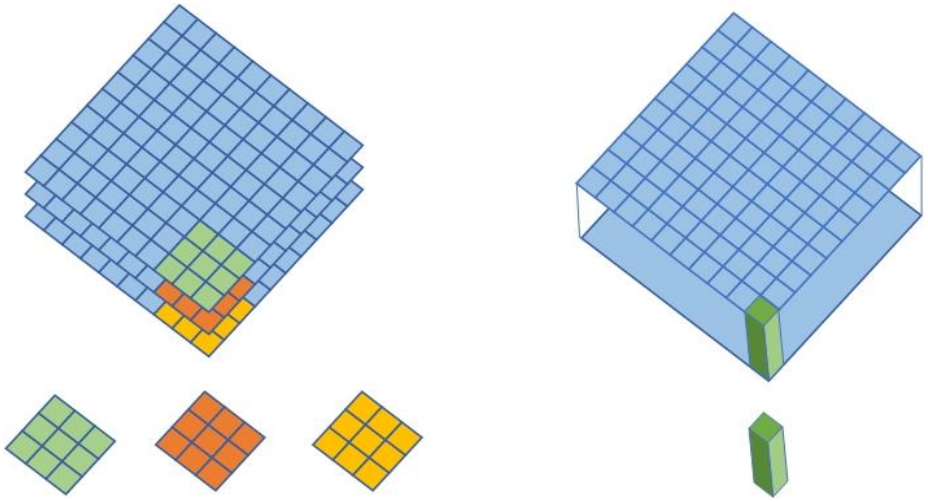


Fig. 5 Structure of the ASPP module we use.

2.3.3 Depthwise Separable Convolution Structure

The depthwise separable convolution was first proposed by [29] and then used in Inception Models [30] to reduce the computational complexity of the first few layers. In [31], the MobileNet structure was further designed. By adopting the depthwise separable convolution method, the effect of reducing the number of parameters and increasing the operaten speed is achieved. The Google team's Deeplabv3+ [27] used a depthwise separable convolution structure with atrous convolution and demonstrated that the atrous separable convolution significantly reduced the computational complexity while maintaining good performance.

The depthwise separable convolution resolves the standard convolution into a deep convolution and a 1x1 convolution called point-by-point convolution, its calculation process is shown in Figure 6. Deep convolution applies a single filter to each input channel. The point-by-point convolution then applies a 1x1 convolution to combine the outputs of the deep convolution. The standard convolution filters the input in one step and combines the inputs into a new set of outputs. The depthwise separable convolution divides it into two layers, one for filtering and one for combination. This decomposition greatly reduces the amount of calculation and the complexity of the model.



(a)Depthwise Convolution (b) Pointwise Convolution

Fig. 6 Depthwise separable convolution

For the convolutional neural network in this work, the addition of deep separable convolution can improve the computational efficiency of the network on a limited GPU, and on the other hand, it also saves computation time, thereby improving the effectiveness of the network in practical applications.

3. Experimental Results and Evaluation

All experiments were performed on an Intel(R) Core (TM) i9-7980XE CPU @ 2.60GHz CPU with 32GB RAM and NVIDIA 2080Ti * 2 GPU. The convolutional neural network is constructed by Pytorch.

3.1 Accuracy of the Pre-trained Resnet50 and Our Structure

To fully test the performance of our proposed model, we compared the performance of the un-pretrained Resnet50 with our model on the dataset we produced in 2.1. The experimental results are shown in Table 1. Experiments show that the accuracy of our model is 8% higher than that of Resnet50 without pre-training.

Table 1. Comparison of un-pretrained Resnet50 and our model

Models	Training Accuracy	Testing Accuracy
Resnet50	84.93%	73.78%
Proposed model	85.83%	81.86%

3.2 Pre-trained Resnet50 and the Accuracy of Our Model

3.2.1 Comparison of Effects Placed on Different Layers

[32] proposed that pre-training becomes an effective method for initializing deep convolution networks when the training data set is insufficient. Since our network takes RGB images as input, we import the pre-trained model on ImageNet to initialize our model. We found that when the crack image was processed directly with the original Resnet50, severe overfitting occurred, which may be caused by the small training data set. Therefore, we introduce ASPP module and depthwise separable structures, which enable the network to obtain multi-scale crack information, while improving computational efficiency and alleviating over-fitting problems. At the same time, the ASPP module and the deep separable structure we built can be easily inserted into any convolutional neural network structure, which effectively improves the generalization performance and detection accuracy of the model.

In order to test which part of the Resnet50 structure the ASPP module and the depth separable structure are best placed on, we have conducted several experiments, the final effect of which is shown in table 2. The input_channels here refers to the number of input channels for our ASPP

module, and out_channels determines the number of output channels for the depthwise convolution, as shown in equation 4:

$$\text{depthwise}\_channels = input\_channels \times output\_channels \tag{4}$$

Table 2. Summary of model performance

Models	location	Training accuracy	Testing accuracy	Input and output channels
Resnet50	/	99.79%	93.65%	/
Proposed model	after conv1	98.41%	95.80%	in=64,out=4
		98.93%	93.16%	in=64,out=8
		98.50%	92.66%	in=64,out=32
	after layer1	99.03%	96.70%	in=256,out=4
		98.72%	92.75%	in=256,out=8
		98.19%	91.76%	in=256,out=16
	after layer2	98.99%	91.76%	in=512,out=8
		98.95%	93.65%	in=512,out=16
	after layer3	99.81%	91.92%	in=1024,out=4
		99.69%	89.12%	in=1024,out=8
	after layer4	99.81%	87.96%	in=2048,out=8
		99.81%	88.29%	in=2048,out=16

From Table 2, we can see that our model obtained the best crack detection accuracy of 96.70% when inserting after Resnet50's Layer1, which is 3.05% higher than the original Resnet50. The number of input channels is 256 and the number of output channels is 4. At the same time, we found that when our model is placed at a higher level in the network, such as Layer3 and Layer4, the performance of the model begins to decline. We speculate that this is because the crack belongs to the underlying features of the image, so we can get better results when the information from bottom layer is reused as much as possible. And feature multiplexing in high dimensions is not useful for crack detection.

3.2.2 Comparison of Different Void Rates

From Table 2, we can find that when our model is inserted after Layer 1 of Resnet50, and when the number of input channels is 256 and the number of output channels is 4, then the best accuracy is obtained. The atrous rates of ASPP module are 2,4,8 respectively. In order to test the influence of different atrous rates on the accuracy of the model, we put the model after the Layer1 of Resnet50, and the number of input channels and the number of output channels remain unchanged at 256 and 4. The following experiments were carried out. And the experimental results are shown in Table 3.

Table 3. Comparison of model performances of different atrous rates

Atrous Rates	Training Accuracy	Testing Accuracy
[2,4,8]	99.03%	96.70%
[2,4,6]	99.61%	94.31%
[3,6,9]	99.55%	94.64%

It can be seen from Table 3 that the best results are obtained when the atrous rate is [2, 4, 8], and the accuracy of models with the atrous rate of [2, 4, 6] and [3, 6, 9] is slightly lower than it.

3.3 Model Evaluation

3.3.1 Evaluation Indicators

We use four evaluation indicators to evaluate the model proposed in this paper, namely the recall rate [33], the missed alarm rate, the accuracy rate and the false alarm rate [34]. We test 1213 images

in the test set, and calculate the four indicators based on the corresponding ground truth, and give the value of these indicators.

When calculating these indicators, we count the number of images instead of the pixel count. This is because when we do ground truth, we only mark which images contain cracks, which images are backgrounds, and are not accurate to pixels. Therefore, when evaluating, it is not accurate to the pixel level, and can only be counted and evaluated by whether the current picture contains cracks.

Several concepts commonly used in detecting problems are as follows:

(1) TP (true positive): refers to the total number of crack pictures classified as cracks.

(2) TN (true negative): refers to the total number of background pictures classified as background.

(3) FP (false positive): refers to the total number of background pictures classified as cracks.

(4) FN (false negative): refers to the total number of crack pictures classified as background.

The recall rate R indicates how many true crack pictures (reference ground truth) in the sample are correctly classified as cracks, and its calculation method is as shown in equation (5):

$$\mathrm{R} = \frac{TP}{TP+FN}. \tag{5}$$

The missed alarm rate MA indicates how many true crack pictures (reference ground truth) in the sample are misclassified as background, and its calculation method is as shown in equation (6):

$$\mathrm{MA} = \frac{FN}{TP+FN}. \tag{6}$$

The accuracy rate P indicates how many pictures that are predicted to be cracks (reference output) are true cracks, and its calculation method is as shown in equation (7):

$$\mathrm{P} = \frac{TP}{TP+FP}. \tag{7}$$

The false alarm rate FA indicates how many true background pictures (reference ground truth) in the sample are misclassified as cracks, and its calculation method is as shown in equation (8):

$$\mathrm{FA} = \frac{FP}{TP+FP}. \tag{8}$$

3.3.2 Other Detection Methods

In order to evaluate the CNN-based crack detection method proposed in this paper, we also introduce a method for comparison, which is crack detection method based on the convolutional neural network proposed by L Zhang [17] in the first section. In the article, the author used the CNN network structure shown in Figure 7 to train and detect the unprocessed original image. The network consists of four convolutional layers and two fully connected layers.

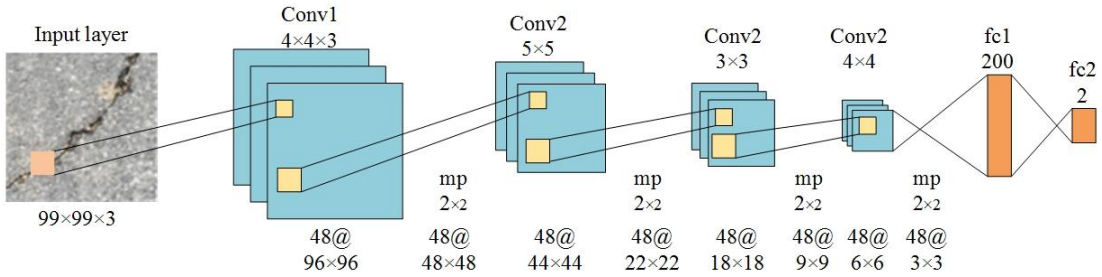


Fig. 7 CNN structure proposed by L Zhang

3.3.3 Comparison of Test Results

We evaluate the model proposed in this paper according to the evaluation indicators mentioned in 3.3.1. The comparison of test results of proposed method and the evaluation results of the original Resnet50 and L Zhang proposed structure are shown in Table 4.

Table 4. Model comparison

Methods	R	MA	P	FA
Resnet50	99.03%	96.70%	93.66%	6.34%
CNN(Zhang,2016 ICIP)	99.61%	94.31%	86.96%	—
Proposed Method	99.55%	94.64%	96.69%	3.31%

It can be seen from Table 4 that the crack detection model proposed in this paper is superior to the original Resnet50 and the structure proposed by L Zhang in all aspects.

Compared with L Zhang's CNN-based crack detection model, the advantages of the model structure of this paper are mainly reflected in:

(1) When designing the CNN network, considering that the first layer of CNN is usually used to extract image edge information, we use a 3×3 convolution kernel because the odd-numbered convolution kernel can extract edge features. L Zhang designed the first layer of CNN as a 4×4 convolution kernel, which violated the principle of extracting edge features of the first layer of CNN.

(2) The number of convolution kernels of each layer of the Resnet50 used in this paper is increasing, which are 64, 128, 256 and 512 respectively. This incremental relationship ensures that each layer can obtain complete and effective feature expression. Compared with the model structure of 48 convolution kernels in each layer of L Zhang, this paper has more advantages in extracting features.

(3) We add ASPP module to Resnet50, which samples parallel samples with different sampling rates at a given input, which is equivalent to capturing the context of images in multiple scales, thus obtaining multi-scale image feature information, improving the recognition accuracy of the network. At the same time, the use of atrous convolution can provide a larger receptive field with a comparable amount of computation, thereby enabling the extraction of more dense feature maps.

(4) We add a depthwise separable convolution to the model, which greatly reduces the computational complexity and model complexity.

According to Table 4, the crack detection model of this paper has a great advantage in accuracy rate, which is 9.51% higher than that of L Zhang. This is mainly because the proposed ConvNets can extract crack features more effectively and distinguish cracks from non-cracks. At the same time, compared to the original Resnet50, proposed method has a lower false alarm rate. This shows that the situation of mistakenly identifying the background as a crack occurs even less in our method.

4. Conclusion

This paper proposes a bridge detection algorithm based on deep learning, which aims to accurately detect cracks in concrete bridges. Based on the Resnet50 network, this paper designs a feature extraction module based on Atrous Spatial Pyramid Pooling (or ASPP) and depthwise separable convolution (depthwise separable convolution). Our module can be inserted anywhere in the convolutional neural network model, which can better extract image feature information and improve model identification accuracy.

References

[1]. A. Jahangiri, H.A. Rakha, and T.A. Dingus, adopting machine learning methods to predict red-light running violations, in Proceedings of IEEE International Conference on Intelligent Transportation Systems, Sept.2015, p. 650–655.

[2]. A. Jahangiri and H.A. Rakha, applying machine learning techniques to transportation mode recognition using mobile phone sensor data, IEEE Transactions on Intelligent Transportation Systems. Vol. 16(2015) No. 5, p. 2406–2417.

[3]. M. Salman, S. Mathavan, K. Kamal, et al. Pavement crack detection using the gabor filter, in Proceedings of IEEE International Conference on Intelligent Transportation Systems, Oct. 2013, p. 2039–2044.

[4]. Q. Zou, Y. Cao, Q. Li, et al. Cracktree: Automatic crack detection from pavement images, Pattern Recognition Letters. Vol. 33(2012) No. 3, p.227–238.

[5]. H. Oliveira and P.L. Correia, Crackit-an image processing toolbox for crack detection and characterization, in Proceedings of IEEE International Conference on Image Processing (ICIP), Oct. 2014, p. 798–802.

[6]. Abdel-Qader I, Abudayyeh O, Kelly M E, Analysis of Edge-Detection Techniques for Crack Identification in Bridges[J], Journal of Computing in Civil Engineering. Vol. 17(2003) No. 4, p.255-263.

[7]. Subirats, Dumoulin, Legeay, et al. Automation of Pavement Surface Crack Detection using the Continuous Wavelet Transform[C]// IEEE International Conference on Image Processing (ICIP). IEEE, 2007.

[8]. Li Q, Zou Q, Zhang D, et al. FoSA: F* Seed-growing Approach for crack-line detection from pavement images[J], Image and Vision Computing. Vol. 29(2011) No. 12, p.861-872.

[9]. Guan H, Li J, Yu Y, et al. Iterative Tensor Voting for Pavement Crack Extraction Using Mobile Laser Scanning Data[J]. IEEE Transactions on Geoscience & Remote Sensing. Vol. 53(2015) No. 3, p.1527-1537.

[10]. HINTON G E, SALAKHUTDINOV R R. Reducing the Dimen-sionality of Data with Neural Networks[J]. Science. Vol. 313(2006) No. 7, p.504-507.

[11]. Sermanet P, Chintala S, Lecun Y. Convolutional neural networks applied to house numbers digit classification[C]// 2012 21st International Conference on Pattern Recognition (ICPR 2012). IEEE Computer Society, 2012.

[12]. Zeiler M D, Fergus R. Visualizing and Understanding Convolutional Networks[J]. European Conference on Computer Vision (ECCV), Zurich, Switzerland, 2014.9.6-2014.9.12, p.818-833.

[13]. Tran D, Bourdev L, Fergus R, et al. Learning Spatiotemporal Features with 3D Convolutional Networks[J]. In Proceedings of the International Conference on Computer Vision (ICCV), Santiago, Chile,2015.12.13-2015.12.16, p.4489–4497.

[14]. SZEGEDY C, LIU W, JIA Y, et al. Going Deeper with Convolutions[J]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, Massachusetts , USA, 2015.6.8-2015.6.10,p.1-9.

[15]. Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the Inception Architecture for Computer Vision[J]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA ,2016.6.27-2016.6.30, p.2818-2826.

[16]. Girshick R. Fast R-CNN. IEEE International Conference on Computer Vision (ICCV). Santiago, Chile, 2015.12.13-2015.12.16, p. 1440-1448.

[17]. Zhang L, Yang F, Zhang Y D, et al. Road crack detection using deep convolutional neural network[C]// IEEE International Conference on Image Processing (ICIP). Phoenix, AZ, USA ,2016.9.25-2016.9.28.

[18]. Li Liang-Fu, Ma Wei-Fei, Li Li, Lu Cheng. Research on detection algorithm for bridge cracks based on deep learning. Acta Automatica Sinica, 2018.

[19]. Cha Y J, Choi W, B¨uy¨uk¨ozt¨urk, Oral. Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks[J]. Computer-Aided Civil and Infrastructure Engineering, Vol.32(2017), No.5, p.361-378.

[20]. Wilson, D. R. & Martinez, T. R., The need for small learning rates on large problems, in Proceedings of Inter-national Joint Conference on Neural Networks, Washington DC,2001.7. 15–2001.7.19, p.115–19.

[21]. He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016. 6.27-2016.6.30, p. 770-778.

[22]. Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. ImageNet classification with deep convolutional neural networks. In NIPS, 2012.

[23]. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[24]. Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions[J]. arXiv preprint arXiv:1511.07122, 2015.

[25]. Chen L C, Papandreou G, Kokkinos I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. IEEE transactions on pattern analysis and machine intelligence, 2018, 40(4): 834-848.

[26]. Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 2017: 2881-2890.

[27]. Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 801-818.

[28]. Lazebnik S, Schmid C, Ponce J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories[C]//2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). IEEE, 2006, 2: 2169-2178.

[29]. Sifre L, Mallat S. Rigid-motion scattering for image classification[J]. PhD thesis, Ph. D. thesis, 2014, 1: 3.

[30]. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. arXiv preprint arXiv:1502.03167, 2015.

[31]. Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[32]. Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos[C]//Advances in neural information processing systems. 2014: 568-576.

[33]. Davis J, Goadrich M. The relationship between Precision-Recall and ROC curves[C] // Proceedings of the 23rd international conference on Machine learning. ACM, 2006: 233-240.

[34]. Kou X. Research on Bridge Crack Detection Algorithm Based on Deep Learning [D]. Master thesis, Xidian University, 2017.