Received December 8, 2020, accepted December 16, 2020, date of publication January 11, 2021, date of current version January 22, 2021.
Digital Object Identifier 10.1109/ACCESS.2021.3050401

Pavement Crack Detection Algorithm Based on Densely Connected and Deeply Supervised Network

HAIFENG LI[1], JIANPING ZONG[1], JINGJING NIE[2], ZHILONG WU[2], AND HONGYANG HAN[1]
[1]College of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China
[2]Chengdu Tianfu International Airport, Chengdu 641419, China
Corresponding author: Haifeng Li (lihf_cauc@126.com)

This work was supported by the National Key Research and Development Project of China under Grant 2019YFB1310601.

ABSTRACT In order to improve the accuracy and robustness of existing automated crack detection methods, a fully convolutional neural network for pixel-level detection based on densely connected and deeply supervised network is proposed. First, the densely connected layers are applied for enhancing the propagation and reuse of crack features. Then, the deeply supervised modules are designed to make network extract more significant features through multi-scale levels. Finally, the feature maps from different scales are fused to achieve complementarity at different levels. In addition, a class-balanced cross-entropy loss function is designed to balance backgrounds and cracks by increasing the weight of crack pixel loss. The proposed method is tested on three public datasets, and the experiments show that our method is superior to state-of-the-art methods in accuracy, speed and robustness.

INDEX TERMS Crack detection, deep learning, densely connected network, deeply supervised network.

I. INTRODUCTION

In recent years, highway and airport constructions are booming all over the world, especially in the developing countries. To keep good condition of infrastructure, prompt and efficient maintenance of pavement surface has become an important issue in the field of transportation industry. Cracks are the very early forms of most diseases on pavement surfaces. Prompt and accurate detection of cracks could minimize maintenance costs and improve efficiency. However, nowadays manual inspection shows the disadvantages of poor accuracy, high subjectivity and inefficiency, which cannot satisfy the needs of rapid highway construction. Thus, efficient and automated crack detection has become a research hotspot.

Numerous efforts have been applied on traditional digital image processing techniques to detect cracks, such as threshold segmentation, feature extraction, edge detection, filter and minimum path methods. Oliveira and Correia [1] extract crack feature with the combination of connected component and automatic threshold segmentation. Li *et al.* [2] use improved OTSU threshold and adaptive iterative threshold to detect cracks on airport runway surface. Wei *et al.* [3] adopt gray difference and Hough transform to realize automatic detection of small cracks. Kapela *et al.* [4] utilize Hough transform feature (HTF) and local binary pattern (LBP) to extract the edge direction and texture features of cracks respectively. Qu *et al.* [5]employ structural forest edge detector to extract crack edge, and seepage model to complete denoising. Amhaz *et al.* [6] propose an automatic detection algorithm of two-dimensional pavement cracks based on minimum path location. The crack detection algorithms based on traditional digital image processing transform or map the original image to a specific space, and obtain the final detection result by learning the structure of shallow crack features. However, due to the complexity of real pavement conditions and the various uncertainties of environmental impacts, such as texture diversity, strong noise interference, irregular crack direction and so on, these algorithms are easy to be interfered by environmental factors, and cannot meet the needs of accuracy and speed at the same time. Therefore, the efficient and robust crack detection algorithms still need to be studied.

Since the cracks and edges have similar characteristics in shape, structure and thickness, it is practicable to apply edge detection method to detect cracks. Based on structural

The associate editor coordinating the review of this manuscript and approving it for publication was Tomasz Trzcinski.

forest [7], Shi *et al.* [8] propose CrackForest algorithm to detect pavement cracks by the combination of complementary features of cracks, and the result is more accurate than Free-Form Antioxidant (FFA) [9] and Minimal Path Selection (MPS) [6]. However, the algorithm is still based on the human-selected features of crack, which have weak adaptability and poor robustness in complex background. Richer Convolutional Feature (RCF) [10], as one of the most advanced edge detection algorithms, can produce high-quality edges efficiently by combining multiscale and multilevel information of objects. But the backbone of RCF is only composed of multiple convolution layers, and the high-level convolution layer only uses the feature map which is transmitted from the previous layer, and it leads to that the high-level convolution neglects many crack features even if the final fusion combines the results of all scales. Thus, RCF is not fully applicable to crack detection.

Deep learning has been widely used in the field of computer vision. Some studies have been committed to apply deep learning to detection and recognition of pavement surface cracks. Eisenbach *et al.* [11] propose a road disease dataset for training deep learning networks, and evaluate the current situation of pavement disease detection technology for the first time. Zhang *et al.* [12] apply a convolution neural network to the classification of fracture panel and non-fracture panel, and prove the advantage of deep learning in fracture detection. Li *et al.* [13] propose a classification model based on convolutional neural network, Deep Bridge Crack Classify (DBCC), and conduct optimized sliding window algorithm to detect bridge cracks. The above methods regard crack detection as a task of image block classification based on deep neural network. Besides, those methods neglect the spatial relationship between crack pixels which causes the lack of global crack features. Inspired by Fully Convolutional Networks (FCN) [14], some studies have been devoted to apply semantic segmentation for crack detection. Schmugge *et al.* [15] propose a remote video crack detection method based on semantic segmentation network. Wei [16] applies semantic segmentation method to automatically learn the linear, direction and edge features of cracks for pixel classification. Li *et al.* [17] develop a lightweight semantic segmentation model based on crack characteristics, and obtained the average crack width using the axis skeleton algorithm. However, since the features generated by deep-level layers are abstract semantic features, the general CNN based semantic segmentation methods may miss the detail feature of cracks and lead to inaccuracy detection results. In addition, with growing depth of neural network structures and increasing number of layers, the extraction of crack feature could be more difficult, and the gradients are going to vanishing. In 2017, Gao, *et al.* [20] proposed a classification network, DenseNet, to strengthen feature propagation and alleviate the vanishing-gradient problem. In DenseNet, each layer has direct access to the gradients from the loss function and the original input signal, leading to an implicit deep supervision. By densely connecting the feature maps, DenseNet provides us an efficient way for feature extraction. However, although the DenseNet based algorithms have achieved superior performance for feature extraction, due to the semantic feature distribution of cracks, and the imbalance of foreground and background ratio in crack detection, it is necessary to supervise and fuse the features from different scales when adopting DenseNet, which induces to our work in this paper. Since deeply-supervised nets (DSN) method simultaneously minimizes classification error while making the learning process of hidden layers direct and transparent, it provides the potential to supervise the feature extraction with DenseNet in our crack detection applications.

To overcome the difficulties in crack detection due to its very thin shape and semantic feature distribution, we propose a fully convolutional neural networks for pixel-level detection based on densely connected and deeply supervised network. The main contributions are listed as follows.

1) The dense connection module is designed for extracting the feature map from the image at various scales. Densely connected convolution is used to extract the features of cracks more sufficiently.

2) The deep supervision module is used to constraint multiple hidden layers and extract multiscale detail features of crack.

3) The multiscale information of crack features generated from all the deep supervision modules are fused by the fusion module to obtain the final crack detection results.

4) To deal with the imbalance of crack and non-crack pixels, a class balanced cross entropy loss function is designed to obtain more stable training results by dynamic adjusting the weight of crack pixel loss.

The proposed method is tested on three public datasets: AEL [16], Crack500 [18] and Cracktree200 [19]. The experiment results validate our method.

II. OVERVIEW OF METHODS

The main structure of our proposed network is shown in Fig. 1. The network is composed of convolution modules, dense connection modules, conversion modules, deep supervision modules, deconvolution layers and fusion module. The input of the network is a road surface image, while the output is a crack prediction map with the same size as the input, and the crack pixels have higher probability than non-crack pixels.

Given an image into the network, firstly the multiscale feature maps are extracted by the convolution modules and dense connection modules, then the dense connection modules are connected by the conversion modules which mainly compresses the dense features from the previous modules to alleviate the feature redundancy. Following each convolution module and dense connection module, a deep supervision module is connected. Each convolution module and dense connection module extracts a feature map for deep supervision module, and each deep supervision module generates a prediction map with loss function. During training, the loss function of the feature maps generated by deep supervision

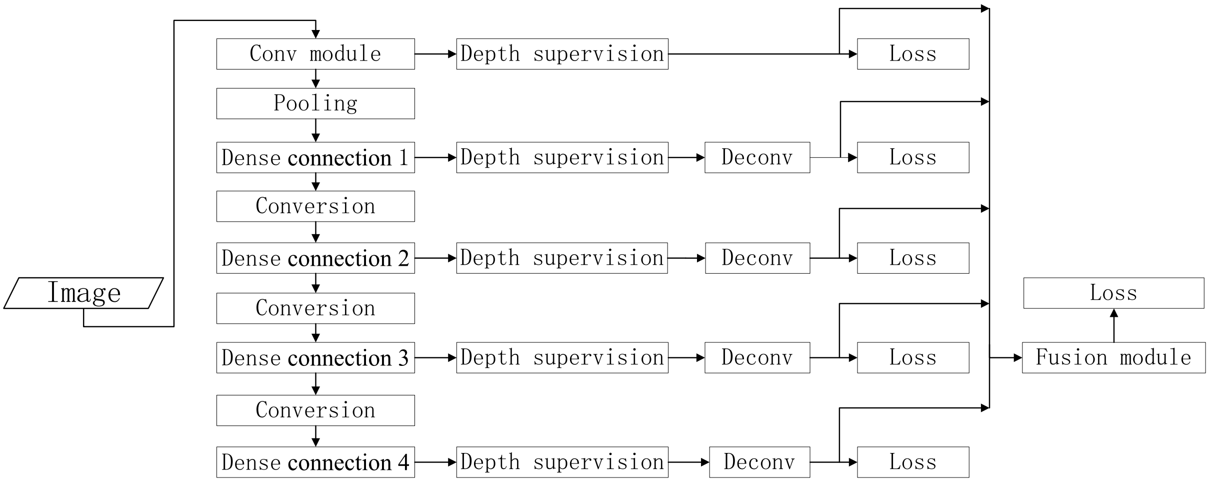

FIGURE 1. The overall module block diagram of proposed model.

modules or deconvolution layers are calculated. Since the sizes of the feature maps extracted by deep supervision modules are different, deconvolution is used to restore the feature map to the original image size after the deep supervision module. Finally, the deconvoluted feature map feeds into the fusion module to obtain the final crack prediction map.

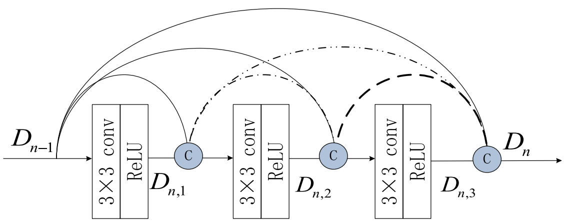

FIGURE 2. The connection mechanism of dense block.

A. DENSE CONNECTION MODULE

Inspired by the idea of DenseNet structure, the dense connection module is designed to extract the crack features and ensure the effective propagation of the gradient. Fig. 2 shows the dense connection mechanism of the module. In the dense connection module, each layer uses the concatenation of feature maps produced by all previous layers as the inputs, that means the feature map produced by the layer is one of the inputs for all following layers. Denote D_{n-l} and D_n as input and output of the n-th dense connection module, respectively, and the output of the layers in the module is defined as

$$D_{n,l} = H_l([D_{n-l}, D_{n,l}, \ldots D_{n,l-1}]), \tag{1}$$

where $D_{n,l}$ denotes the output of the i^{th} layer in the dense connection module n, while $[D_{n-l}, D_{n,l}, \ldots D_{n,l-1}]$ refers to the concatenation of feature maps from all layers $l, \ldots, l-1$. And the nonlinear transformation $H_i(\square)$ is a composite function of $3 * 3$ convolutions and the rectified linear unit (ReLU) is the activation function. By establishing the dense connection of features in different layers, the modules can extract the crack features more sufficiently and alleviate the gradient vanishing problem. Besides, it can reduce the number of network parameters and the calculation cost.

B. CONVERSION MODULE

As the dense features extracted from the dense connection module should be compressed, and the redundant features should be reduced, the conversion modules are used to connect to dense connection modules adjacently. A conversion module consists of a 1×1 convolution layer and a 2×2 max pooling layer in which 1×1 convolution can fuse features of different levels from the dense connection module and persist more favorable information, and max pooling layers facilitate calculation.

C. DEEP SUPERVISION MODULE

The structure of dense connection module can strengthen the extraction of crack features, but it is still a single-stream supervision network structure overall. As the network structure is deepened, the gradient of the backpropagation will gradually shrink and the learning speed of the model during training will decrease. In addition, with the increasing number of feature layers, the supervision for the output layer of the network cannot achieve effective training for the extraction of low and mid-level features, which leads to poor performance.

Inspired by the idea of Deeply-Supervised Nets (DSN) [21], deep supervision module is designed to speed up the model convergence and improve the feature extraction capability of both the low-level layers and the high-level layers. It is connected to each dense connection module. Besides, the deep supervision modules extract feature maps from different levels, which solves the problem of losing crack details when using high-level semantic features for crack segmentation alone.

The deep supervision modules of the proposed network are designed as follows: the dense connection module is considered as a unit, and the feature maps from each convolutional layer are concatenated in channel dimension. Then a convolution layer with kernel size 1×1 and channel depth 64 is connected to the concatenate operation, followed by a $1 \times 1 - 1$ convolution layer. As the output feature maps from deep supervision modules are smaller than the original image in size, a deconvolution layer is used to up-sample for changing the size of the feature map, where the deconvolution layers are excluded while learning, and their parameters are initialized with bilinear interpolation algorithm.

D. CLASS BALANCE CROSS ENTROPY LOSS FUNCTION

In real scenes, the number of non-crack pixels is far more than crack pixels in typical images with cracks, and the imbalance makes the network difficult to converge correctly. Therefore, a class balanced cross entropy loss function is designed to balance the contribution to the loss from crack pixels and non-crack pixels for stabilizing training. The loss function of a single pixel is defined as

$$l(X_i, U) = \begin{cases} \log(1 - \sigma(X_i, U)) & y_i = 0 \\ \alpha \log(1 - \sigma(X_i, U)) & y_i = 1 \end{cases} \tag{2}$$

where $\alpha = \frac{|X_+|}{|X_-|}$. X_+ and X_- denote the crack and non-crack pixels in the label respectively. U represents all the parameters that need to be learned in the proposed network, and X_i denotes the value at the pixel i, y_i is the corresponding label in ground truth image, and $\sigma(X_i, U)$ is the standard sigmoid activation function.

Therefore, the loss of deep supervision module j is defined as

$$L^j(U) = \sum_{i=1}^{N} l(X_i^j, U). \tag{3}$$

and the loss function of fusion module is

$$L^{fuse}(U) = \sum_{i=1}^{N} l(X_i^{fuse}, U). \tag{4}$$

Therefore, the overall loss function can be formulated as

$$E(U) = \sum_{j=1}^{M} \sum_{i=1}^{N} l(X_i^j, U) + \sum_{i=1}^{N} l(X_i^{fuse}, U). \tag{5}$$

where N is the number of pixels in the input image and M is the number of deep supervision module. Specifically, five deep supervision modules are used in the proposed network.

E. OVERALL STRUCTURE

The proposed network architecture is shown in Fig. 3. The network is composed of a convolution module, four dense connection modules, five deep supervision modules, three conversion modules and one fusion module. The detailed architecture of proposed network can be described as follows:

- The main parts of the network are convolution module, dense connection modules and conversion modules to extract the features from the input image.
- The convolution module is composed of two convolution layers with kernel size 3×3 and channel depth 64.
- The first dense connection module consists of two convolution layers, and the other three dense connection modules are composed of three convolution layers. In a dense connection module, the output is as the input for each layer, which is concatenated by the feature-maps produced by all preceding layers and the feature map served into dense connection modules. Dense connection modules are connected by convolution modules, which do convolution and pooling.
- To extract the multi-scale feature information of cracks more sufficiently and make the learning process of hidden layers transparent, deep supervision module is designed. Each deep supervision module is connected to the convolution module or dense connection modules for generating multi-level feature maps.
- A deconvolution layer follows to resize feature map to the original size, and the class balance cross entropy loss will be computed with it.
- Finally, all the resized feature maps are fed into the fusion module for fusing the detailed features and semantic features from different levels.

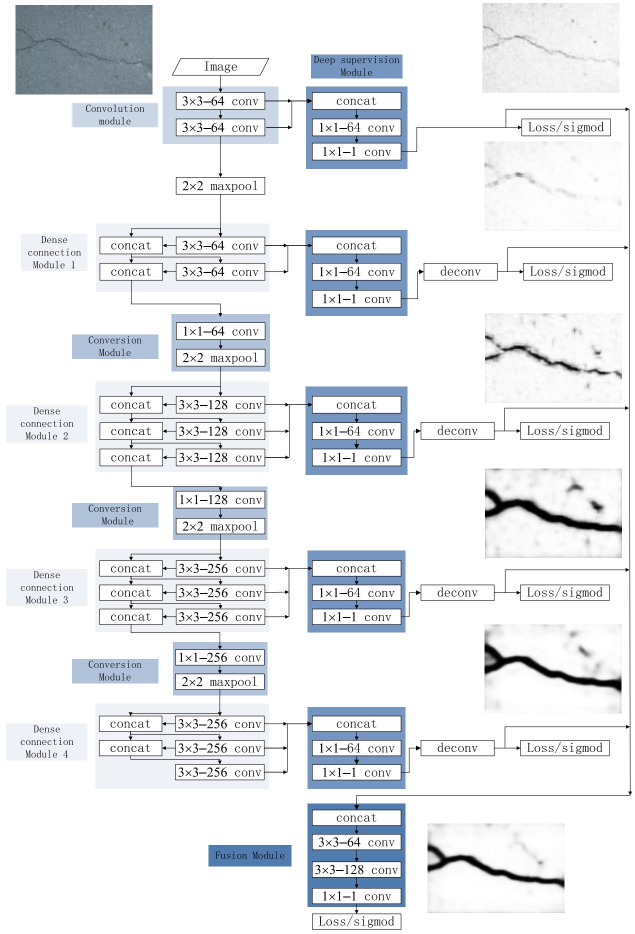

FIGURE 3. The detailed architecture of proposed network.

Besides, the parameters of each module are shown in Table 1, where padding denotes the filling size of feature map before input, and stripe denotes the moving step size of convolution layer filter or pooling layer window.

TABLE 1. The parameters of model backbone network.

Module	Network layer	padding	stride
Convolution module	conv	[1,1,1,1]	[1,1]
Dense Connection module	conv	[1,1,1,1]	[1,1]
Conversion module	conv	[0,0,0,0]	[1,1]
	maxpool	[0,0,0,0]	[2,2]
Deep supervision module	1×1-64 conv	[0,0,0,0]	[1,1]
	1×1-1 conv	[0,0,0,0]	[1,1]
	3×3-64 conv	[1,1,1,1]	[1,1]
Fusion module	3×3-128 conv	[1,1,1,1]	[1,1]
	1×1-1 conv	[0,0,0,0]	[1,1]

III. EXPERIMENTS AND RESULTS

The proposed method is implemented and trained with PyTorch framework. Our method is tested on a computer with 64GB RAM, 11GB GeForce GTX 1080 Ti, and i7-8700 CPU @ 3.2GHz.

A. DATASET

We have evaluated the proposed method on three public datasets: AEL, crack500 and cracktree200. The details of those datasets are shown in Table 2.

AEL is composed of three data named Aigle-RN, ESAR and LCMS including 58 crack images. Crack500 is a pavement crack dataset including 3368 images captured by a cell phone on main road of Temple University, which has size of 1440 × 2560 or 2560 × 1440. Cracktree200 is a visible light dataset containing various kinds of cracks in Complex interference environments like shadow, occlusion, low contrast, noise and other interferences, and it contains 206 crack images of size 800 × 600.

Crack pixels have been manually labeled in the three datasets. And we use the training data from Crack500 to train the proposed method, and the test data contains the test data of Crack500, AEL and Cracktree200. Since images from the datasets of AEL and Cracktree200 have several different sizes around 800 ∗ 800 pixels, to guarantee the same image size for training and validation with as little information loss as possible, we first crop the images in Crack500 into 800 ∗ 800 pixels, and then resize the images from AEL and Cracktree200 into the same size.

B. NETWORK TRAINING PARAMETERS SETTING

Training data of crack500 only contains 1896 crack images, and the lack of quantity may lead to poor training results. Therefore, image enhancement methods (rotation and clipping) are used to enhance the training data. The final training data contains 13272 crack images.

Stochastic gradient descent (SGD) with momentum is adopted for network parameters optimization. The mini-batch is set to 10, the momentum is set to 0.9, and the weight decay coefficient is set to 0.0002. While training, Gaussian kernel with zero-mean and standard deviation 0.01 is used to initialize each layer. The learning rate is set to 1e-6. The learning rate is divided by 10 for each iteration of 10000 times. The method is trained for a total of 50000 iterations.

C. COUNTERPARTS

The four existing methods which we compare our algorithm to are CrackForest [8], FCN [14], RCF [18] and FC-DenseNet [22]. CrackForest is a road crack detection framework based on random structured forests, by learning the inherent structured information of cracks. FCN is a general semantic segmentation neural network. RCF is an accurate edge detector using richer convolutional features. FC-DenseNet investigates the use of Densely Connected Convolutional Networks for semantic segmentation.

D. EVALUATION CRITERIA

Given a crack map, a crake prediction map is produced by our method, and the threshold is needed for yielding the final detection results. The proposed method uses two thresholds respectively, which are optimal dataset scale (ODS) and optimal image scale (OIS) because of the similarity between crack detection and edge detection. ODS employs a fixed threshold for the whole dataset, while OIS employs the best threshold for each image. Then, the best F-measure of both ODS and OIS are defined as follows

$$F_{ODS} = \max\{\frac{1}{N}\sum_{i}^{N} 2\frac{P_t^i \times R_t^i}{P_t^i + R_t^i} : t = 0.01, 0.02, \ldots, 0.99\} \tag{6}$$

$$F_{OIS} = \frac{1}{N}\sum_{i}^{N} \max\{2\frac{P_t^i \times R_t^i}{P_t^i + R_t^i} : t = 0.01, 0.02, \ldots, 0.99\} \tag{7}$$

where t denotes the threshold, N is the total number of images in the dataset, P_t^i is the precision of the i^{th} image at the threshold t, R_t^i is the recall of the i^{th} image at the threshold t. As the ground truth annotation of edge detection task and crack detection task are binary boundary images and binary segmentation images respectively, the detection result and the ground truth are processed by non-maximum suppression, and the foreground is refined to single pixel width before calculation.

E. EXPERIMENTAL RESULT

According to the above experimental settings, we have completed the compared experiments on the three datasets of Crack500, AEL and Cracktree200, and the test results are showed in table 3–5 according to the evaluation criteria.

And the results tested on Crack500, Cracktree200 and AEL with standard deviation are listed in Table 6. The visualization results of each model on the three datasets are shown in Fig. 4,

TABLE 2. Datasets for our experiments.

Dataset usage	train	validation	test		
Dataset	Crack500	Crack500	AEL	Crack500	Cracktree200
Image number	13272	348	58	1124	206

TABLE 3. Crack detection results on Crack500 test dataset.

Method	ODS	OIS	Time/image(s)
FCN	0.383	0.389	0.071(GPU)
FC-DenseNet	0.273	0.412	0.177(GPU)
CrackForest	0.065	0.065	1.552(CPU)
RCF	0.431	0.545	**0.027(GPU)**
Our method	**0.627**	**0.669**	**0.027(GPU)**

TABLE 4. Crack detection results on Cracktree200 test dataset.

Method	ODS	OIS	Time/image(s)
FCN	0.287	0.287	0.127(GPU)
FC-DenseNet	0.049	0.076	0.177(GPU)
CrackForest	0.369	0.369	3.063(CPU)
RCF	0.229	0.396	0.057(GPU)
Our method	**0.547**	**0.637**	**0.056(GPU)**

TABLE 5. Crack detection results on AEL test dataset.

Method	ODS	OIS	Time/image(s)
FCN	0.355	0.356	0.086(GPU)
FC-DenseNet	0.156	0.214	0.177(GPU)
CrackForest	0.298	0.298	2.011(CPU)
RCF	0.155	0.432	0.044(GPU)
Our method	**0.555**	**0.658**	**0.042(GPU)**

in which the optimal values of the results are highlighted in bold. Besides, as the detection results of both CrackForest and FCN are binary segmentation images, the F_{ODS} and F_{OIS} values are basically same respectively.

As shown in table 3–5, the proposed method achieves best performances on all the datasets. CrackForest detects road cracks by the combination of multi-levels complementary features of cracks. However, as CrackForest still relies on human selected features for crack detection, it may lead to the poor robustness and false detection in complex background. FCN cannot segment the relatively small cracks very well due to the significant imbalance between foreground and background. RCF can produce high-quality detection results very efficiently by fusing the multi-scale and multi-level information of crack. However, as the backbone of it is only composed by multiple convolutional layers and each layer only uses the information from its preceding layer, RCF may lose some crack features. Although FC-DenseNet improves the OIS on Crack500 dataset, it achieves unsatisfactory results on AEL and Cracktree200. The main reason is that it is difficult to learn features by single loss with lacking of fusion module.

TABLE 6. Crack detection results on AEL, Cracktree200, and Crack500.

Dataset	ODS (std)	OIS (std)
Crack500	0.627±0.185	0.669±0.147
Cracktree200	0.547±0.261	0.637±0.219
AEL	0.555±0.243	0.658±0.193

The proposed method also has a good performance in complex interference environments such as low contrast, shadow, occlusion, noise and other interferences. The crack detection results of each method are shown in Fig. 5 in the complex background. We can see that when the background is too complex to detect the crack area even if it is identified by the manual inspecting, the proposed method and RCF can detect the crack area. Compared with RCF, the detection results of the proposed method have lower false positive rate. Experiment results show the superior performance in accuracy and robustness of the proposed method.

Besides, the average detection speed of the compared methods is showed in table 3∼table 5. Especially, RCF, FCN and the proposed method only count the calculation time on GPU, and the time consumption of loading images on CPU and saving results is neglected. We note that the proposed method does not increase computation time, even though the number of network layers and the input feature maps for each layer is increased. The speed of the proposed method is as fast as RCF with a better performance in speed than FCN and CrackForest.

F. ABLATION STUDY

We have evaluated different functions of our method to conclude the impacts from those functions. We have fulfilled two group of compared experiments for the ablation study. One experiment is to compare the performance when using different loss function (class-balanced cross-entropy loss and traditional cross-entropy loss). The other experiment is to validate the effects of dense connection module and fusion module by removing these modules or not.

The experimental results are shown in table 7, where we can find that, the dense connection module and fusion module can improve the ability of extracting crack features, and the class balanced cross entropy loss can improve the accuracy of

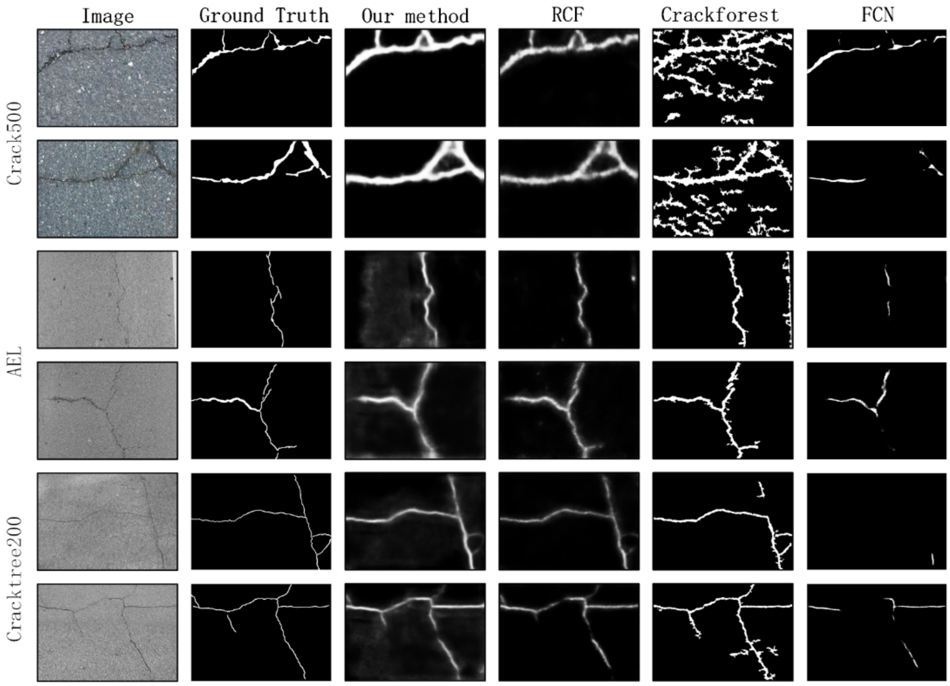

FIGURE 4. The visualization of detection results of compared methods on three datasets.

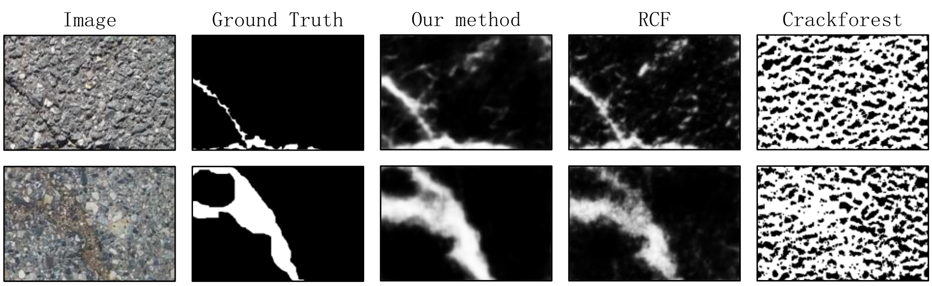

FIGURE 5. Comparison of crack detection results under complex environment.

TABLE 7. Ablation experimental results.

Method	Crack500		Cracktree200		AEL	
	ODS	OIS	ODS	OIS	ODS	OIS
proposed method	**0.627**	**0.669**	**0.547**	**0.637**	**0.555**	**0.658**
without class-balanced cross-entropy loss	0.572	0.661	0.489	0.602	0.490	0.628
without dense connection and fusion modules	0.604	0.635	0.517	0.579	0.492	0.507

crack detection, which benefits from more contribution from crack pixels.

IV. CONCLUSION

In this work, we propose a pavement crack detection algorithm based on densely connected and deeply supervised network, which improves the detection accuracy and efficiency of pavement cracks. Firstly, the dense connection module is designed for enhancing the features of cracks continuous propagation, reusing features, and ensuring the effective propagation of gradient; Then, the feature information in multi-scale space is extracted and the convergence speed of the model is accelerated by the deep supervision of multiple hidden layers; Finally, the feature maps of crack outputted from multi-level layers is fused to obtain more accurate detection results. Besides, using a class balanced cross entropy loss function helps increase the weight of crack pixel loss. The method is tested on several public crack datasets, showing the accuracy performance with much less false positive detection, stronger robustness and faster detection speed of the proposed method compared with RCF, FCN, FC-DenseNet and CrackForest. The method can provide a certain technical support for the rapid and accurate detection of pavement cracks in the practical engineering.

REFERENCES

[1] H. Oliveira and P. L. Correia, "CrackIT-an image processing toolbox for crack detection and characterization," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Piscataway, NJ, USA, May 2014, pp. 798–802.

[2] L. Peng, W. Chao, L. Shuangmiao, and F. Baocai, "Research on crack detection method of airport runway based on twice-threshold segmentation," in *Proc. 5th Int. Conf. Instrum. Meas., Comput., Commun. Control (IMCCC)*, Piscataway, NJ, USA, Sep. 2015, pp. 1716–1720.

[3] W. Chuntao, "Automatic crack detection method based on adaptive threshold for small cracks and micro-gray difference," *J. China Foreign Highway*, vol. 39, no. 1, pp. 58–63, 2019.

[4] R. Kapela, P. Sniatala, A. Turkot, A. Rybarczyk, A. Pozarycki, P. Rydzewski, M. Wyczalek, and A. Bloch, "Asphalt surfaced pavement cracks detection based on histograms of oriented gradients," in *Proc. 22nd Int. Conf. Mixed Design Integr. Circuits Syst. (MIXDES)*, Piscataway, NJ, USA, Jun. 2015, pp. 579–584.

[5] Q. Zhong and J. F. C. Siqi, "Concrete surface cracks detection combining structured forest edge detection and percolation model," *Comput. Sci.*, vol. 45, no. 11, pp. 288–291 and 311, 2018.

[6] R. Amhaz, S. Chambon, J. Idier, and V. Baltazart, "Automatic crack detection on 2D pavement images: An algorithm based on minimal path selection," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2718–2729, Sep. 2016.

[7] P. Dollar and C. L. Zitnick, "Structured forests for fast edge detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Piscataway, NJ, USA, Dec. 2013, pp. 1841–1848.

[8] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 12, pp. 3434–3445, Dec. 2016.

[9] T. S. Nguyen, S. Begot, F. Duculty, and M. Avila, "Free-form anisotropy: A new method for crack detection on pavement surface images," in *Proc. 18th IEEE Int. Conf. Image Process.*, Piscataway, NJ, USA, Sep. 2011, pp. 1069–1072.

[10] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Piscataway, NJ, USA, Jul. 2017, pp. 5872–5881.

[11] M. Eisenbach, R. Stricker, D. Seichter, K. Amende, K. Debes, M. Sesselmann, D. Ebersbach, U. Stoeckert, and H.-M. Gross, "How to get pavement distress detection ready for deep learning? A systematic approach," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Piscataway, NJ, USA, May 2017, pp. 2039–2047.

[12] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Piscataway, NJ, USA, Sep. 2016, pp. 3708–3712.

[13] L. Liangfu, W. Ma, L. Li, and C. Lu, "Research on detection algorithm for bridge cracks based on deep learning," *Acta Autom. Sinica*, vol. 45, no. 9, pp. 1727–1742, 2019.

[14] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[15] S. J. Schmugge, L. Rice, J. Lindberg, R. Grizziy, C. Joffey, and M. C. Shin, "Crack segmentation by leveraging multiple frames of varying illumination," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Piscataway, NJ, USA, Mar. 2017, pp. 1045–1053.

[16] W. Fang, "Research of vehicle-mounted automatic pavement crack identification technology based on semantic segmentation," Changan Univ., Xi'an, China, Tech. Rep., 2019, doi: 10710-2016122011.

[17] L. Gang, "Study on improved global convolutional network for pavement crack detection," *Laser Optoelectron. Prog.*, vol. 57, no. 8, 2020, Art. no. 081011.

[18] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, "Feature pyramid and hierarchical boosting network for pavement crack detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1525–1535, Apr. 2020.

[19] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "Crack tree: Automatic crack detection from pavement images," *Pattern Recognit. Lett.*, vol. 33, no. 3, pp. 227–238, Feb. 2012.

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Piscataway, NJ, USA, Jul. 2017, pp. 2261–2269.

[21] C. Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Proc. 18th Int. Conf. Artif. Intell. Statist.*, San Diego, CA, USA, 2015, pp. 562–570.

[22] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 11–19.

HAIFENG LI was born in Tongliao, Inner Mongolia, China, in 1984. He received the B.S. degree in computer science and technology and the Ph.D. degree in control theory and control engineering from Nankai University, Tianjin, China, in 2007 and 2012, respectively.

He is currently an Associate Professor with the College of Computer Science and Technology, Civil Aviation University of China, Tianjin. He has authored or coauthored more than 30 technical articles. His research interests include computer vision, image processing, robotic sensing, multisensor fusion, and robot localization and navigation.

JIANPING ZONG received the bachelor's degree from Central South University, Changsha, China. He is currently pursuing the master's degree in computer technology with the Civil Aviation University of China. His research interests include image processing and deep learning.

JINGJING NIE received the B.S. degree from the Chongqing University of Posts and Telecommunications, Chongqing, China, and the master's degree in computer technology from the Civil Aviation University of China, in 2020. Her research interests include image processing and deep learning.

ZHILONG WU received the B.S. degree from the Chongqing University of Posts and Telecommunications, Chongqing, China, and the master's degree in computer technology from the Civil Aviation University of China, in 2020. His research interests include image processing and deep learning.

HONGYANG HAN received the B.S. degree from the Zhengzhou University of Aeronautics, Zhengzhou, China. He is currently pursuing the master's degree in computer technology with the Civil Aviation University of China. His research interests include image processing and deep learning.