Signal, Image and Video Processing
https://doi.org/10.1007/s11760-021-02034-w

ORIGINAL PAPER

DcsNet: a real-time deep network for crack segmentation

Jie Pang[1,2] · Hua Zhang[1,2] · Hao Zhao[1,2,3] · Linjing Li[1,2]

Received: 15 May 2021 / Revised: 15 August 2021 / Accepted: 17 September 2021

Abstract
Detecting cracks are a great significance for the maintenance of the man-made buildings, and deep learning methods such as semantic segmentation have greatly boosted this process in recent years. However, the existing crack segmentation methods often sacrifice feature resolution to achieve real-time inference speed which leads to poor performance, or use complex network module to improve the accuracy which leads to lower inference speed. In this paper, we propose a novel Deep Crack Segmentation Network (DcsNet) that incorporates two feature extraction branches to achieve the balance of speed and accuracy. We first design a morphology branch (MB) to preserve the morphology information of scale invariance that consists of a lightweight convolution network, a pyramid pooling module (PPM), and an attention module (CSA). Meanwhile, a shallow detail branch (DB) with a small stride is constructed to supplement detailed information. Extensive experiments are conducted on five challenging datasets (Crack500, Deepcrack, Gaps384, Structure, and Damcrack), and the results demonstrated that the proposed network achieves a good trade-off between accuracy and inference speed and outperforms state-of-the-art methods.

Keywords Crack segmentation · Real-time · Morphology information · Detailed information

1 Introduction

Crack is a common defect in the man-made concrete buildings, such as road, bridge and dam. These defects have a serious impact on the safety and maintenance of the building. Crack detection is a major approach to measure the degree of defect, and its result is a significant indicator to evaluating the service cycle [1–3]. Currently, the main approach of crack detection is still manual way. The workers are required high experience and limited by the time-consuming and low efficiency [4]. It is necessary to study fast and accurate automatic crack detection methods to improve the efficiency and alleviate the workload of workers [5].

Recently, with the development of computer vision and deep learning, convolutional neural networks (CNNs) have been widely used in automatic crack detection, especially semantic segmentation [6–8]. These CNN-based methods can not only extract robust features related to the cracks, but also provide an end-to-end pixel-level classification result [9]. In order to boost the discriminative ability of model, some researchers apply the high-resolution images, multi-scale features, multiply feature channels or deeper networks [10–12], such as U-shape structure and Feature Pyramid module. However, the speed of these crack segmentation networks is inevitably affected by the large amount of parameters and redundant computation. On the other hand, reducing the resolution of feature maps and pruning the depth and channel of network are utilized to improve the inference speed [13, 14]. But these strategies will lead to a serious loss of detail information. For example, the detail information loss of edge will directly affect the segmentation accuracy. Moreover, the crack on the building surface image shows the characteristics of dramatic changes in size and structure, as well as extremely imbalance of foreground and background of area. Therefore, the existing segmentation methods do not obtain a satisfactory trade-off between accuracy and speed.

According to above observation, we propose the Deep Crack Segmentation Network (DcsNet) with two feature extraction branches, and Fig. 1 shows the structure of the network. DcsNet consists of morphology branch (MB) and detail branch (DB). Morphology branch devised to extract global morphology features with scale invariance and detail

✉ Hao Zhao
zhaohao@swust.edu.cn

1 School of Information Engineering, Southwest University of Science and Technology, Mianyang 621000, China

2 Special Environment Robot Technology Key Laboratory of Sichuan Province, Mianyang 621000, China

3 Department of Automation, University of Science and Technology of China, Hefei 230000, China

Published online: 17 January 2022

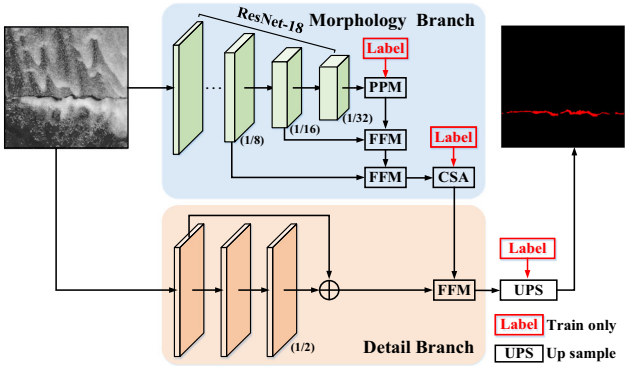

Fig. 1 An overview of DcsNet. Each box corresponds to a multi-channel feature map, size ratios to the full-resolution input. '**PPM**' stands for the pyramid pooling module, '**FFM**' stands for the fusion of two feature maps, '**CSA**' stands for the channel and spatial attention module, and '**UPS**' stands for the final × 2 up sampling. The red characters '**Label**' represents deep supervision we take only during training

branch devised to extract with local detailed features. Morphology Branch adopts a lightweight network as backbone to extract three low-resolution feature maps (ie.,1/8-, 1/16-, 1/32-resolution feature maps) from the crack image as high-dimensional global morphology information. The pyramid pooling module (PPM) is used as top feature mapping to enhance scale invariance. Then, adding the channel and spatial attention module (CSA) to optimize the fused feature map in space and channel dimensions, which is the result of two FFM modules fusing three feature layers. Finally, each feature extraction stages are supervised separately during the training process to further improve the feature extraction ability. Extensive experiments are conducted on five crack datasets Crack500 [15], Deepcrack [16], Gaps384 [17], Structure [18], and Damcrack. Those experiments prove that the proposed method achieves the best balance between speed and accuracy compared with other typical segmentation network. The comparison with other methods on the Structure dataset is shown in Fig. 2.

In summary, the main contributions of this paper are summarized as follows:

1. We propose a morphology branch network to extract the global morphological features of the input image, which is separated from the local detail feature extraction subnetwork.
2. We introduce three plug-and-play components (a pyramid pooling module, a lightweight attention module and a simple convolution block) to enhance the ability of feature extraction with less cost of the computation.
3. In the training process, we apply the same ground truth to supervise learning of each feature extraction stage. Besides, the deep supervision modules will be discarded in the inference process. As a result, DcsNet can maintain low inference overload.

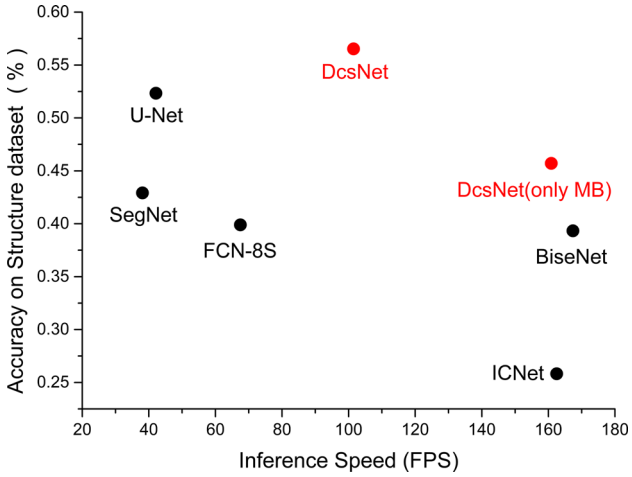

Fig. 2 Comparison between accuracy and frames per second on structure dataset with size of 256 × 256

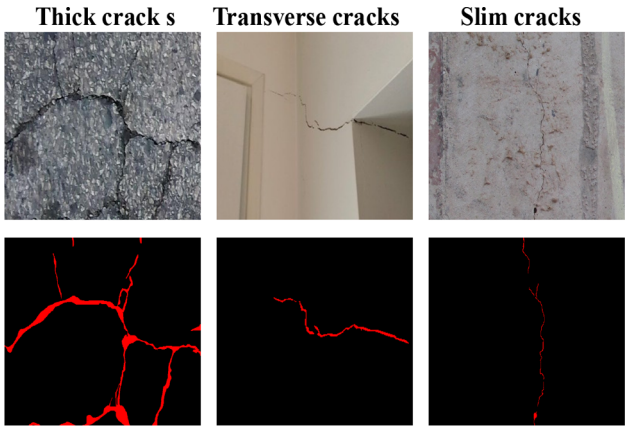

Fig. 3 Example images and labels of five datasets. The top row is some example images, the second row is the crack label of the image directly above

2 Related work

In this section, we analyze the characteristics of crack on building surface and review the progress of related research.

2.1 Crack image observations

The five different datasets adopted in this paper are Crack500, Deepcrack, Gaps384, Structure, and Damcrack. In experiments, all the images are labeled as crack or background pixel-by-pixel. Figure 3 shows some sample images and their corresponding labels.

According to the type of building, this paper summarizes the characteristics of crack as follows:

Morphology differences: The size, morphology and the angle of cracks are random. As shown in Fig. 3, the length and width of cracks may be narrow and long or short, and the morphology is irregular. In feature extraction, the informa-

tion of multi-scale receptive field and high-dimensional deep semantic is very important, which is conducive to enhance the scale invariance and morphological invariance of features.

Tiny boundary: The boundary of cracks is always not smooth and irregular, the width of crakcs is usually narrow even a few pixels. To further improve the segmentation accuracy of tiny cracks, it is necessary to use high-resolution input images and feature maps with small stride to obtain generous detailed information.

Category imbalance and difficult samples: In all the images, only a small part of the pixels belongs to the crack class. Therefore crack datasets have significant category imbalance problem. Besides, there are many crack pixels that are hard to distinguish due to the complex background. These problem reduce the recognition ability of segmentation network.

2.2 Feature extraction and crack segmentation

In recent years, multiple types of full convolution networks have been applied crack segmentation, achieved good performance [19]. Some representative crack segmentation methods in recent years are listed in this paper. Dung et al. [20] used FCN to segment cracks on the surface of concrete buildings; Wang et al. [21] deepened the depth of FCN to increase the multi-scale of features, and used multi-scale structured forests to optimize the output; Jacob K¨onig et al. [22] used multi-level deep supervised training on U-Net to improve the feature extraction ability; These methods achieve good segmentation accuracy on some datasets. A series of other methods have been proposed to improve the speed of segmentation. Wooram Young [24] designed the densely connected separable convolution modules as the block of SDDNet to increase the receptive field information of feature maps and reduce the number of parameters, and a modified spatial pyramid pooling module is used to fuse multi-scale feature maps. In addition, the segmentation network represented by ICNet [25] and BiseNet [26] has achieved the balance of speed and accuracy in urban road segmentation task, but its maximum resolution of feature maps is only 1/8 of the original image, which is not enough to obtain enough detailed information, resulting in insufficient segmentation accuracy.

Obviously, crack segmentation depends on global high-level feature maps with multi-scale receptive fields and morphological information, high-resolution local detailed information is also very important to improve crack segmentation. The existing crack segmentation methods often sacrifice feature resolution to achieve real-time inference speed which leads to poor performance, or use complex network module to improve the accuracy which leads to lower inference speed. In this study, we propose a novel architecture to treat the morphology information and detailed information based on the feature of cracks, which obtains a good trade-off between segmentation accuracy and inference speed.

3 Crack segmentation networks

The proposed DesNet is a pixel-level classification network for crack segmentation, DcsNet mainly consists of two components: (1) morphology branch: Composed by the lightweight ResNet-18, the pyramid pooling module, the feature fusion modules and the channel and spatial attention module. The three top feature maps (ie.,1/8-, 1/16-, 1/32-resolution feature maps) are fused to represent high-dimensional segmentation information. (2) Detailed Branch: 3 convolution layers are spliced to extract a 1/2-resolution feature map, expressing local detailed information. Low-layer convolution will not increase the network scale and have acceptable cost. The two feature maps extracted from MB and DB are fused by the FFM, and the fused feature map are up-sampled twice to 1-resolution input image. Finally, a 1×1 convolution layer and a sigmoid function are used to generate a probability map of crack. Each value on the map represents the probability that the pixel at the same position on the input image is crack. In addition, the Deep Supervision Module are utilized in three stages of MB and DB during the training.

3.1 Morphology branch

In order to obtain large receptive fields and high-dimensional semantic features [27] in large crack segmentation task, deeper convolution architecture is widely used. However, feature pyramid [28], large convolution kernel, many channels [29] and deeper network will increase calculation cost, resulting in lower speed. For the sake of higher speed, DcsNet extract high-dimensional features by these way: First, morphological features are presented by multi-scale receptive field. Then PPM [26, 30] is utilized to process the smallest top feature map for richer information of receptive fields. Finally, we fused the three feature maps through two FFM, and the output is optimized by the CSA. The detailed components and architecture of each different module of MB are as shown in Figs. 4, 5 and 6.

Pooling pyramid module: This module add more semantic information based on the proportion of crack in image. PPM pools the top feature map by 4 average pooling layers with different size of window. The receptive field of the pooled four feature maps (i.e., 1×1, 2×2, 3×3, and 6×6 size feature maps) is four scales bigger than original. Each pooling layers has a bilinear up-sampling [31] with the same window size, and the up-sampling is used to resize the pooled feature

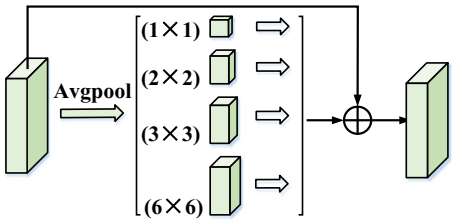

Fig. 4 Architecture of the Pooling Pyramid Module (PPM). '**Avgpool**' stands for four average pooling layers

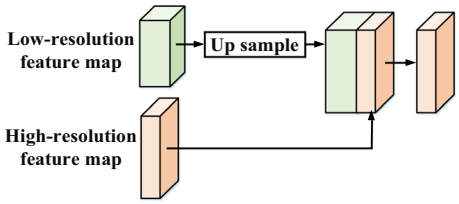

Fig. 5 Architecture of the Feature Fusion Module (FFM). '**Up sample**' stands for the up-sampling operations based on bilinear interpolation

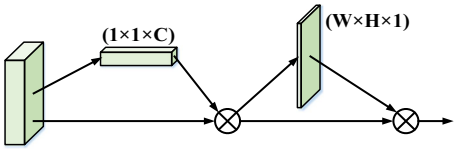

Fig. 6 Components of the Channel and Spatial Attention Module (CSA). '**1 × 1 × C**' stands for channel attention and '**W × H × 1**' stands for the spatial attention

map to the size of the top feature map. Finally, the five feature maps are added linearly as the output of the module.

Feature fusion module: The smaller feature map is up-sampled twice based on biliner to match the larger one, and two feature maps is contracted in the dimension of the channel. Finally, a convolution unit is used to eliminate the semantic gap of the contracted feature maps and reduce the number of channels. The convolution unit in this module consists of a 3×3 convolution layer, a batch normalization layer and a Relu function.

Channel and spatial attention module: In this paper, CSA is composed of channel attention module and spatial attention module in serial. The channel attention module process feature map continuously in the channel dimension and spatial attention module process feature map continuously in the spatial dimension. The detailed structure can be referred to in [32] and Fig. 6. Each convolution unit consists of a convolution layer and a batch normalization layer.

3.2 Detail branch

Local detail information is essential to achieve high-quality crack segmentation. In this paper, we proposed a detailed feature extraction branch to extract 1/2-resolution feature map, which compensate detailed semantic information.

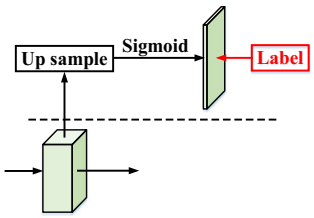

Fig. 7 Architecture of the deep supervision module

This branch consists of three 3×3 convolution units and a 1×1 convolution layer. Specifically, the first unit contains one convolution layer with stride of 2, batch normalization layer and Relu function. The second unit contains one convolution layer with stride of 1, batch normalization layer and Relu function. The third unit contains one convolution layer with stride of 1, the output of which is linearly added to the feature map convolved on the output of the first unit by a 1×1 convolution layer, and processed by batch normalization layer and Relu function as the output feature map of DB.

Finally, the two feature maps obtained from MB and DB are fused through FFM, and the fused one is up-sampled twice to 1-resolution based on bilinear interpolation. The 1×1 convolution layer is used to reduce the channel of feature map, and the sigmoid function is utilized to output the crack probability of each pixel.

3.3 Deep supervision

In order to improve the feature expression ability of different feature extraction branches and different convolution stages in DcsNet, this paper designed a deep supervision module to guide the different feature extraction stages which used in training process [33]. The architecture of the deep supervision module is shown in Fig. 7, and the used location in DcsNet was shown in Fig. 1.

In each deep supervision module, the supervised feature map is up-sampled to 1-resolution and processed by a 1×1 convolution layer, and calculating the loss based on the ground truth through sigmoid function. According to the structure of DcsNet, three deep supervision modules are used to supervise the output of PPM in MB, the global morphology feature map obtained from MB and the final crack probability. Specifically, the first deep supervision module used to auxiliary guide the minimum resolution feature map can improve the information with multi-scale receptive fields, the second module is used to auxiliary guide the structure feature extraction of MB to express global morphology information, and the third module is used to supervise and evaluate the final result of DcsNet. In inference process, the first two auxiliary supervision modules are directly discarded, and the final result is directly regarded as the predicted result, which without any additional model parameters and computation.

3.4 Loss functions

Semantic segmentation tasks usually use cross entropy loss [34] to calculate the difference between the prediction and ground truth. In the task of crack segmentation on building surface, there is a serious category imbalance in the number of pixels belonging to crack and background so that the background pixels comprise the majority of the loss and dominate update direction of gradient. In addition, there are a large number of difficult pixel samples similar to the crack in the background.

Dice coefficient is a measurement that gauges the similarity of two sets [35]. In our task, let P denote the set that contains crack pixels predicted by DcsNet and let Y denote crack pixels in ground truth, so the dice coefficient between them is given as follows in Eq. (1):

$$D(P, Y) = \frac{2P \times Y + \gamma}{P + Y + \gamma} \tag{1}$$

where γ is a minimal constant used to avoid the situation that the ground truth and prediction are both. It's obvious that none of the background samples contributed to objective and the dice coefficient can be used to design the loss function of DcsNet for crack segmentation. The ground truth and prediction are changed to the square form for smoothing purposes and faster convergence, so the dice loss is obtained as follows in Eq. (2):

$$L_d = 1 - \frac{\text{sum}(P^2 \times Y^2) + \gamma}{\text{sum}(P^2) + \text{sum}(Y^2) + \gamma} \tag{2}$$

According to the deep supervision modules, the loss function for optimizing DcsNet in the training process as follows in Eq. (3):

$$Loss = L + \alpha \times L_M^8 + \beta \times L_M^{32} \tag{3}$$

where L_M^8 and L_M^{32} represent the loss of two auxiliary deep supervision modules, and L represents the main loss that between final result of DcsNet and ground truth. α and β represent the weight of the two auxiliary supervision loss, which are set to 0.4 in this paper according to their importance.

4 Experiments

We conduct various experiments on five crack datasets to evaluate this proposed network. Section 4.1 introduces the five datasets and data augmentation tricks. Next two sections discuss experiment environment and the evaluation metrics. The ablation experiments and compare experiments are shown in the last two sections. Our code is now available at https://github.com/PANGJIE-PANDA/DcsNet.

4.1 Datasets

4.1.1 Datasets

The five datasets used in this paper are Crack500, DeepCrack, Structure, Gaps384 and Dam crack.

(1) Crack500: This dataset is the largest concrete pavement crack dataset consist of 500 RGB images of size around 2000×1500 pixels, and each image has a pixel-level binary label image . In our experiments, the dataset is divided into 250 images as the training dataset and the other 250 as the validation dataset. Due to the limited computation resource, each image was cropped into a series of non-overlapped images with size of 360×640 pixels. Through this way, the training dataset consists of 2244 images, validation dataset contains 1124 images.

(2) Deepcrack: This dataset that came from concrete surface has 537 RGB images with size of 544×384, and each image has a pixel-level binary label image . In our experiments, the dataset is divided into 300 images as the training dataset and the other 237 as the validation dataset.

(3) GAPs384: The GAPs includes 1969 images with various classes of distress such as cracks, potholes, inlaid patches. The image resolution is 1920×1080 pixels. 384 images from the GAPs which only includes crack class were selected and pixel-level annotated as the GAPs384 dataset. Due to the large size of image and limited memory of GPU, each image is cropped and padded with zeros to 6 non-overlapped images with size of 640×544. In our experiments, the dataset is divided into 401 images as the training dataset and the other 108 as the validation dataset.

(4) Structure: Structure is composed of 661 RGB images with size of 256×256 which is mainly from the structural components surface of concrete buildings and bridges, and each image has a pixel-level binary label image . In our experiments, the dataset is divided into 551 images as the training dataset and the other 110 as the validation dataset.

(5) Damcrack: We built a self-made crack dataset came from dam surface due to lack of public datasets. All the images were collected on the surface of one large hydropower station in Southwest China's Sichuan Province and provided by Energy Internet Research Institute of Tsinghua University. This dataset has 1000 RGB images with size of 608×608, and each image has a pixel-level binary label image that manually annotated. In our experiments, the dataset is divided into 800 images as the training dataset and the other 200 as the validation dataset.

4.1.2 Data augmentation

Based on our previous research and analysis [37], there is a significant difference in brightness, contrast, and noise distribution in the concrete surface crack images. In view of this kind of situation, horizontal flip, brightness and contrast adjustment and geometric distortion are used to enlarge the training datasets.

Brightness and contrast adjustment: In order to make the trained model can adapt to the different brightness of the concrete surface, it is necessary to evidently adjust the brightness and contrast of the training images. In this paper, the gray mean value u and square error σ are used to describe the brightness and contrast of the image x, as follows in Eq. (4).

$$u = \frac{1}{m}\sum_{i=1}^{m} x_i,\ \sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - u)^2 \tag{4}$$

where x_i is the gray value of i-th pixel of the image and m is the number of pixels. In order to ensure the significant difference between the adjusted image and the original image, the transformation coefficient η is generated according to Eq. (5):

$$\eta = \frac{1}{1 + \exp(5 - u/25)} \tag{5}$$

The mean value 255η and the standard deviation 2η are used to generate a new luminance value $\hat{u}$ and a new contrast value $2\eta\sigma$, so the adjusted image is represented as Eq. (6):

$$\hat{\boldsymbol{x}} = 2\eta\sigma\frac{\boldsymbol{x} - u}{\sqrt{\sigma^2 + \varepsilon}} + \hat{u} \tag{6}$$

where ε is a minimum constant. This method is used to ensure that the adjusted images has obvious difference with the original images.

Geometric distortion: According to irregular of cracks, the trigonometric function is applied to change the position of each pixel to achieve the geometric distortion. The method is as follows in Eq. (7):

$$\begin{aligned}\hat{h} &= h - 10\sin\left(\frac{2\pi w}{152} + 10\right)\\ \hat{w} &= w - 10\sin\left(\frac{2\pi h}{152} + 10\right)\end{aligned} \tag{7}$$

h, $\hat{h}$, w and $\hat{w}$ are the coordinate positions of each pixel before and after distortion.

4.2 Implementation details

Computation platform: All CNN based models are trained and validated on one 12G NVDIA GTX TITAN XP, and non-deep learning code is executed on a computer with 16G RAM and i7-7770 CPU@4.20 GHz.

Training parameters: The optimizer uses Adam with learning rate of 0.00001 and minibatch of 4 to optimize all segmentation networks in the training stage. The model with minimum average loss value on validation datasets is choosed as the final model during the training process of 100 epochs. In the inference stage, the pixels whose output probability value is not less than 0.5 are identified as crack, and others are identified as background.

4.3 Evaluation metrics and training

We completed all experiments on five datasets and used four common metrics of semantic segmentation to evaluate this network. Let TP denote the number of the crack pixels that were correctly classified, FN denote the number of crack pixels that were misclassified as background, and FP denote the number of background pixels that were misclassified as crack, the metrics are shown in Eqs. (8)-(11).

$$Pa = \frac{TP}{TP + FP},\ Rc = \frac{TP}{TP + FN} \tag{8}$$

$$F1 = \frac{2 \times PA \times Rc}{PA + Rc},\ IoU = \frac{TP}{TP + FN + FP} \tag{9}$$

The recall (Rc) is used to evaluate the performance of the crack pixels are correctly classified, and the precision (Pa) is used to evaluate the performance of that the pixels classified as crack is really crack. The intersection over union (IoU) and F1-score ($F1$) are selected as the import comprehensive metrics that can be regarded as the harmonic average of Rc and Pa. The values of the four metrics are distributed between 0 and 1, the model's ability to segment the crack regions is better when their values are closed to 1.

Since the images of Structure dataset has background with more complex noises, different networks have obvious segmentation gap on the dataset, so this paper mainly shows more detailed comparison on the dataset.

Figure 8 shows the curves of the IoU of DcsNet and other networks on the Structure validation dataset in the training stage. As shown in Fig. 8, the IoU increased with the increase in training epochs, reaching the basic convergence state about 20 epochs. It can be seen that all networks are well trained and DcsNet achieved the best performance.

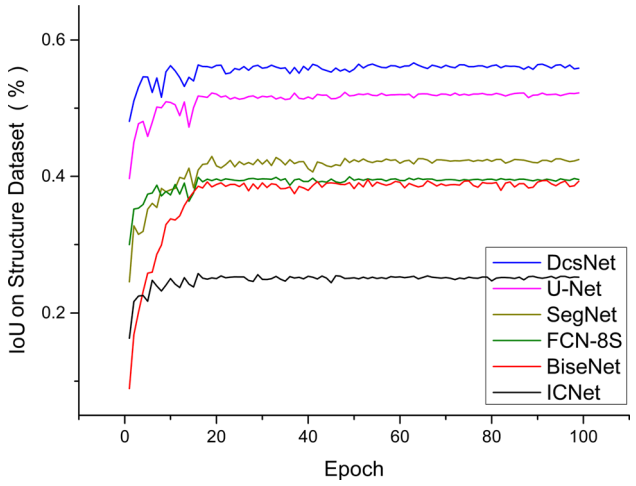

Fig. 8 The curves of *IoU* over training epochs of FCN-8S, SegNet, U-Net, ICNet, BiseNet on Structure validation dataset

Table 1 Comparison of crack segmentation networks with different improvement modules %

Network	*Pa*	*Rc*	*IoU*	*F*1
MB-32	52.8	57.9	38.1	55.2
MB-32 + PPM	57.7	55.7	39.6	56.7
MB-8	62.3	57.0	42.4	59.5
MB-8 + DS	63.4	60.7	44.9	62.0
DcsNet(only MB)	69.9	57.0	45.8	62.8
DcsNet	72.6	72.1	56.6	72.3

4.4 Ablative studies

In this section, the effectiveness of five modules of DcsNet proposed in this paper is analyzed on Structure dataset that PPM, FFM, deep supervision, CSA and DB are separately added on the basis of ResNet-18 one by one, so there were 6 ablation experiments named **MB-32**, **MB-32 + PPM**, **MB-8**, **MB-8 + DS**, **DcsNet (only MB)**, **and DcsNet**. Table 7 reports the contributions of each module and their combinations in terms of four metrics in this paper. It is observed that all of five module lead to the improvement of performance.

Compared with the basis of ResNet-18 that not add any module, PPM, FFM, deep supervision, CSA and DB brought a total of improvement of 18.5% *IoU* and 17.1% *F*1. Specifically, only using PPM leads to the improvement of 1.5% *IoU* due to richer information of multi-scale receptive field, and CSA further improves the of *IoU* of 0.9%. One interesting observation is that **MB-8** and **DB** separately achieved obvious improvement of 2.8% and 10.8% *IoU* compared with the previous experiments.

4.5 Comparative studies

The proposed DcsNet is compared to five classic segmentation networks using same loss function: SegNet [37], FCN-8S [38] and U-Net [39], ICNet [25] and Bisenet [26].

For the accuracy comparison, Table 2 shows two comprehensive metrics on five validation datasets. Besides, Fig. 8 shows the training process on the Structure validation dataset.

The proposed DcsNet achieved the best performance comparing with other CNN-based segmentation networks in *IoU* and *F*1 on all crack datasets, and proved the effectiveness and accuracy of DcsNet for crack feature extraction. These conclusions can also be seen in Fig. 9 in which some segmentation results of image examples are illustrated. Specifically, U-Net and DcsNet identified tiny cracks and more correctly than other networks depend on the high resolution feature maps with rich detail information, some details are clearer and the boundaries are smoother. In addition, the prediction results of DcsNet for the larger crack regions have better continuity and integrity than that of U-Net, which is mainly because the MB of the former has larger receptive field information. The segmentation performance has been improved obviously.

The segmentation speed of network is directly related to the possibility of applying the algorithm to engineering. For the speed comparison, we compare the speed and accuracy with the other networks on the Structure validation dataset with the input size of 256×256, we also explore the impact of DB on the performance of DcsNet at the same time. The detailed results are shown in Table 3 and Fig. 2.

The DcsNet achieved the best accuracy and real-time speed that has 101Fps frame on Structure dataset with the size of 256×256. The amount of parameters and computa-

Table 2 Comparison of different crack segmentation networks on five validation datasets %

Datasets	FCN-8S		SegNet		ICNet		BiseNet		U-Net		DcsNet	
	IoU	*F*1	*IoU*	*F*1	*IoU*	*F*1	*IoU*	*F*1	*IoU*	*F*1	*IoU*	*F*1
Crack500	57.5	73.0	56.6	72.3	56.1	71.9	57.8	73.3	58.1	73.5	58.5	73.8
Gpas384	35.6	52.5	31.6	48.0	30.4	46.7	35.3	52.1	38.7	55.8	39.5	56.7
Deepcrack	69.3	81.8	68.5	81.3	67.4	80.6	61.7	76.3	75.6	86.1	77.9	87.5
Structure	39.9	57.1	42.9	60.0	25.8	41.1	39.4	56.5	52.3	68.7	56.6	72.3
Damcrack	46.4	63.4	43.5	60.6	42.4	59.5	39.5	56.6	51.6	68.0	52.1	68.5

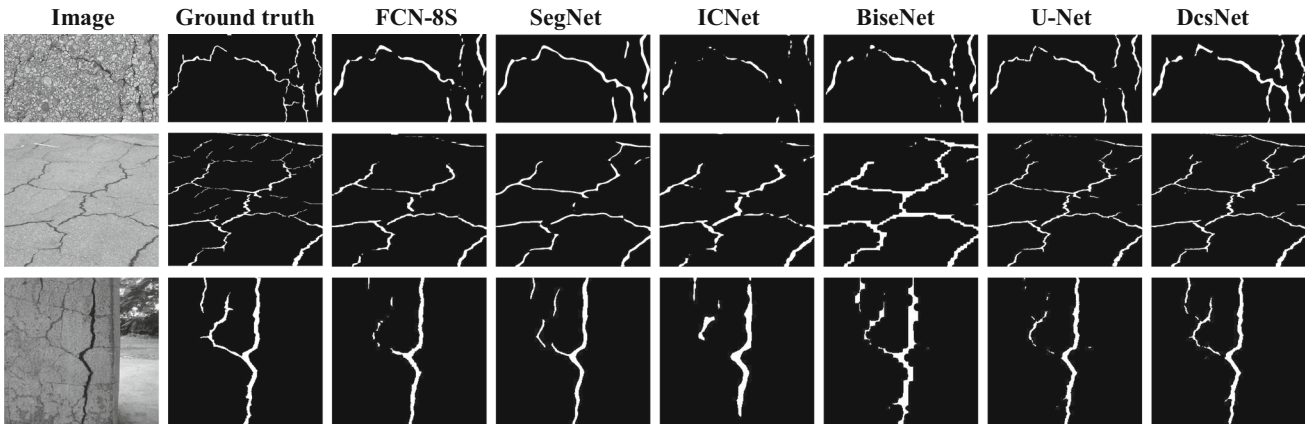

Fig. 9 Some segmentation results of image examples of different networks

Table 3 Comparison of different networks on Structure dataset

Network	P(M)	G(GMac)	F(Fps)	*IoU*(%)	*F*1(%)
FCN-8S	20.1	20.69	67.5	39.9	57.1
SegNet	43.6	58.26	37.9	42.9	60.1
U-Net	17.3	40.12	42.1	52.3	68.7
ICNet	**11.5**	6.64	162.5	25.8	41.1
BiseNet	12.4	**3.03**	**167.5**	39.4	56.6
MB	14.9	3.7	160.8	45.8	62.8
DcsNet	15.1	8.58	101.4	**56.6**	**72.3**

Table 4 Comparison with other SOTA methods

Dataset	Size	Methods	P(M)	F(Fps)	*F*1(%)
Crack 500	375 × 500	FPHBN	N/A	**81.6**	60.4
	384 × 640	DcsNet	N/A	67.5	**73.8**
Gpas 384	640 × 540	FPHBN	N/A	24.9	22.0
	640 × 544	DcsNet	N/A	**51.7**	**56.7**
Deep crack	544 × 384	DeepCrack	14.0	9.2	86.5
	544 × 384	SDDNet	**0.16**	**75.8**	87.1
	544 × 384	DcsNet	15.1	74.8	**87.5**

tion are acceptable that are close to the lightweight networks ICNet and BiseNet with poor performance.

4.6 Comparison with state-of-the-arts

In order to further support the effectiveness of the proposed DcsNet, which is compared with FPHBN [15], DeepCrack [16] and SDDNet [24] that are latest CNN-based networks on three public datasets. To ensure fair comparisons, the input size is considered, the comparison results are shown in Table 4.

The results show that DcsNet has achieved better segmentation accuracy than comparison methods on three public datasets and real-time inference speed lower than SDDNet only.

5 Conclusion

Aiming at the characteristics of the global morphology and size change, small local details, and irregular shapes of the concrete surface cracks, a deep network for crack segmentation (DcsNet) is proposed in this paper that composed of a morphology feature extraction branch and a detailed feature extraction branch. The trained DcsNet was tested and discussed on five validation datasets with other existing methods, the results showed that DcsNet achieved the best accuracy and real-time inference speed.

References

1. Zheng, M.J., Lei, Z.J., Zhang, K.: Intelligent detection of building cracks based on deep learning. Image Vis. Comput. **103**(11), 103987 (2020)
2. Wu, C.F., Sun, K.K., Xu, Y.M., Zhang, S., Huang, X., Zeng, S.Q.: Concrete crack detection method based on optical fiber sensing network and microbending principle. Saf. Sci. **117**(9), 299–304 (2019)
3. Kim, B., Yuvaraj, N., Preethaa, K., Pandian, R.: Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. Neural Comput. Appl. (2021). https://doi.org/10.1007/s00521-021-05690-8
4. Fang, F., Li, L.Y., Gu, Y., Zhu, H.Y., Lim, J.H.: A novel hybrid approach for crack detection. Pattern Recognit. **107**(11), 107474 (2021)
5. Guilherme, F.G., Yohan, A.D.M., Patrícia, D.S.L.A., Sebastião, S.D.C.J., Antonio, C.A.J.: The use of intelligent computational tools for damage detection and identification with an emphasis on composites–a review. Compos. Struct. **196**(7), 44–54 (2018)
6. Juan, J.R., Takahiro, K., Teera, L., Wenlong, D., Kohei, N., Sergio, E., Kotaro, N., Yutaka, M., Helmut, P.: Multi-class structural damage segmentation using fully convolutional networks. Comput. Ind. **112**(11), 103121 (2019)

7. Amir, R., Radhakrishna, A., Michele, G., Katrin, B.: Comparison of crack segmentation using digital image correlation measurements and deep learning. Constr. Build. Mater. **261**(11), 120474 (2020)
8. Uche, A.N.: Fully adaptive segmentation of cracks on concrete surfaces. Comput. Electr. Eng. **83**(5), 106561 (2020)
9. Zhou, S.L., Song, W.: Concrete roadway crack segmentation using encoder-decoder networks with range images. Automat. Constr. **120**(12), 103403 (2020)
10. Mohan, R., Abhinav, V.: EfficientPS: Efficient panoptic segmentation. (2020)
11. Lin, D.Y., Li, Y.Q., Tin, L.N., Dong, S., Zaw, M.O.: RefineU-Net: improved U-Net with progressive global feedbacks and residual attention guided local refinement for medical image segmentation. Pattern Recogn. Lett. **138**(10), 267–275 (2020)
12. Sang, H.W., Zhou, Q.H., Zhao, Y.: PCANet: Pyramid convolutional attention network for semantic segmentation. Image Vis. Comput. **103**(11), 103997 (2020)
13. Adam, P., Abhishek, C., Sangpil, K., Eugenio, C.: ENet: A deep neural network architecture for real-time semantic segmentation. (2016)
14. Si, H.Y., Zhang, Z.Q., Lv, F.F., Yu, G., Lu, F.: Real-time semantic segmentation via multiply spatial fusion network. (2019)
15. Yang, F., Zhang, L., Yu, S.J., Danil, P., Mei, X., Ling, H.B.: Feature pyramid and hierarchical boosting network for pavement crack detection. IEEE Trans. Intell. Transp. Syst. **4**(4), 1525–1535 (2020)
16. Liu, Y.H., Yao, J., Lu, X.H., Xie, R.P., Li, L.: DeepCrack: a deep hierarchical feature learning architecture for crack segmentation. Neurocomputing **338**, 139–153 (2019)
17. Chen, F.C., Mohammad, R.J.: ARF-Crack: rotation invariant deep fully convolutional network for pixel-level crack detection. Mach. Vis. Appl. (2020). https://doi.org/10.1007/s00138-020-01098-x
18. Bai, Y.S., Zha, B., Halil, S., Alper, Y.: Deep cascaded neural networks for automatic detection of structural damage and cracks from images. In: ISPRS2020, pp. 411–417 (2020)
19. Christian, K., Kristina, D., Varun, K., Burcu, A.: A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. Adv. Eng. Inform. **29**(2), 196–210 (2015)
20. Cao, V.D., Le, D.A.: Autonomous concrete crack detection using deep fully convolutional neural network. Autom. Constr. **99**(3), 52–58 (2018)
21. Wang, S., Wu, X., Zhang, Y.H., Liu, X.Q., Zhao, L.: A neural network ensemble method for effective crack segmentation using fully convolutional networks and multi-scale structured forests. Mach. Vis. Appl. (2020). https://doi.org/10.1007/s00138-020-01114-0
22. Jacob, K., Mark, D.J., Mike, M., Peter, B., Gordon, M.: Optimized deep encoder-decoder methods for crack segmentation. Digit. Signal Process. **108**, 102907 (2020)
23. Mei, Q.P., Mustafa, G., Md, R.A.: Densely connected deep neural network considering connectivity of pixels for automatic crack detection. Autom. Constr. **110**(2), 10301 (2020)
24. Wooram, C., Young, J.C.: SDDNet: real-time crack segmentation. IEEE Trans. Industr. Electron. **67**(9), 8016–8025 (2019)
25. Zhao, H.S., Qi, X.J., Shen, X.Y., Shi, J.P., Jia, J.Y.: ICNet for real-time semantic segmentation on high-resolution images. In: ECCV2018, pp. 418–434 (2018)
26. Yu, C.Q., Wang, J.B., Peng, C., Gao, C.X., Yu, G., Sang, N.: BiSeNet: bilateral segmentation network for real-time semantic segmentation. In: ECCV2018, pp. 334–349 (2018)
27. Peng, C., Zhang, X.Y., Yu, G., Luo, G.M., Sun, J.: Large kernel matters-improve semantic segmentation by global convolutional network. In: CVPR2017 (2017)
28. Alexander, K., Ross, G., He, K.M., Piotr, D.: Panoptic feature pyramid networks. In: CVPR2019 (2019)
29. Szegedy, C., Liu, W., Jia, Y.Q., Pierre, S., Scott, R., Dragomir, A., Dumitru, E., Vincent, V., Andrew, R.: Going deeper with convolutions. In: CVPR2015 (2015)
30. He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
31. Xu, H.J., Gao, Y., Li, J., Gao, X.B.: CBFNet: constraint balance factor for semantic segmentation. Neurocomputing **397**(15), 39–47 (2020)
32. Sanghyun, W., Jongchan, P., Lee, J.Y., In, S.K.: CBAM: Convolutional block attention module. In: ECCV2018, pp. 3–19 (2018)
33. Mo, J., Zhang, L.: Multi-level deep supervised networks for retinal vessel segmentation. Int. J. Comput. Assist. Radiol. Surg. **12**(12), 2181–2193 (2017)
34. Tsungyi, L., Priya, G., Ross, G., He, K.M., Piotr, D.: Focal loss for dense object detection. IEEE Trans. Pattern Anal. Mach. Intell. **42**(2), 318–327 (2020)
35. Li, X.Y., Sun, X.F., Meng, Y.X., Liang, J.J., Wu, F., Li, J.W.: Dice loss for data-imbalanced NLP tasks. ArXiv: 1911.02855 (2019)
36. Pang, J., Zhang, H., Feng, C.C., Li, L.J.: Research on crack segmentation method of hydro-junction project based on target detection network. KSCE J. Civ. Eng. **24**(7), 2731–2741 (2020)
37. Vijay, B., Alex, K., Roberto, C.: SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017)
38. Jonathan, L., Evan, S., Trevor, D.: Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 640–2495 (2017)
39. Olaf, R., Philipp, F., Thomas, B.: U-Net: convolutional networks for biomedical image segmentation. In: MICCAI2015, pp. 234–241 (2015)