Contents lists available at ScienceDirect

Machine Learning with Applications

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

An improved U-Net model for concrete crack detection

Chenglong Yu [a,b], Jianchao Du [a,*], Meng Li [a], Yunsong Li [a], Weibin Li [b]

[a] *School of Telecommunications Engineering, Xidian University, Xi'an 710071, Shaanxi Province, China*
[b] *School of Artificial Intelligence, Xidian University, Xi'an 710071, Shaanxi Province, China*

ARTICLE INFO

Keywords:
Crack detection
U-net
Multi-scale features
Strip pooling
Attention mechanism

ABSTRACT

Crack detection plays an important role in disease assessment of concrete buildings. However, factors such as complex background, irregular edge, and the real-time and accuracy requirement also make crack detection a challenging task. Aiming at the above challenges, an improved U-Net model for concrete crack detection is proposed, which has strong capability to extract the linear object, improving the performance in crack detection. The model is named Residual Linear Attention U-Net (RLAU-Net). There are three key measures in this paper. First, mirror padding the source image before convolution. Second, the multi-level features are obtained by aggregating the multi-scale features level by level. Third, strip pooling kernels are used to extract global contextual information, reducing information interference from the background. We tested the performance of RLAU-Net on our crack dataset, and the experimental results exhibited that it can improve the quantitative results of mean Intersection Over Union to 81.69%. In addition, F1 score has increased to, 78.21%, the Intersection Over Union of crack increased to 64.47%. We also compared the detect time-consuming of RLAU-Net and that of the original U-Net. Results demonstrate that the proposed model has a short processing time while maintaining a high detection accuracy for crack detection.

1. Introduction

Nowadays, the surface of many buildings are all made of concrete, and crack is a kind of main diseases for it. The traditional method of crack detection is by manual. Its disadvantage is that the traffic lane needs to be closed and the work condition is not safe. So it is necessary to research the automatic crack detection. Many effective detection methods have been proposed to meet the requirements in various applications over the last few decades and can be classified into two categories: image-processing-based algorithms and deep-learning-based algorithms. The former includes grayscale-based methods, texture-based methods, and geometric feature-based methods, such as threshold segmentation (Talab, Huang, Xi, et al., 2019), wavelet transform (Zhou, Xu, & Wang, 2019), tensor voting (Liu, Yan, Meng, et al., 2021; Xing, Huang, Xu, et al., 2018), edge detection (Berwo, Fang, Mahmood, et al., 2021), percolation (Yamaguchi & Hashimoto, 2010) and so on. But the above methods only extract the handcrafted feature representations, limiting their performance in representing detail and complex semantic information. Recently, deep learning has shown impressive performances in image segmentation. Compared to traditional methods, they are capable of learning different features from different layers and different scales. Hence, a number of crack detection approaches based on deep learning are developed, such as Fully Convolution Network(Long, Shelhamer, & Darrell, 2015), U-Net (Ronneberger, Fischer, & Brox, 2015), Deeplab series (Chen, Papandreou, Kokkinos, et al., 2014, 2018; Chen, Papandreou, Schroff, et al., 2017) and so on. Among them, U-Net is widely used due to its fast detection speed, simple architecture, and favorable results, but there are still many inevitable shortcomings to be solved when it is used to detect cracks in images. One is that the resolution of the output image is smaller than that of the input image, and the other is that many errors are still existed in the detection results of U-Net.

The main reason for the first disadvantage is that there are convolution layers and pooling layers in U-Net. These two cause the image to lose some pixels during the convolution and pooling operations. Therefore, the resolution of the output image is smaller than the input image, and the more the number of convolution and pooling layers, the greater the above difference. The second defect can be summarized as how to optimize the architecture to make it more suitable for crack detection. With the above aim, several effective approaches have been proposed to deal with crack detection tasks. Most of these methods optimize the backbone or skip connection. Generally speaking, the main effect of optimizing the backbone is to get more feature information, such as VGG (Simonyan & Zisserman, 2015), GoogleNet (Szegedy, Liu, Jia, et al., 2015), Xception (Chollet, 2017), Residual Network (He, Zhang, Ren, et al., 2016), Densely Connected Network (Huang, Liu, Van Der Maaten, et al., 2017), ASPP (Chen et al., 2018),

* Corresponding author.
E-mail addresses: yclxdu1998@163.com (C. Yu), jcdu@xidian.edu.cn (J. Du), lm15534941474@163.com (M. Li), ysli@mail.xidian.edu.cn (Y. Li), weibinli@xidian.edu.cn (W. Li).

https://doi.org/10.1016/j.mlwa.2022.100436
Received 30 August 2022; Received in revised form 9 November 2022; Accepted 13 November 2022
Available online 17 November 2022

PSP (Zhao, Shi, Qi, et al., 2017). The main effect of optimizing skip connection is to strengthen the flow between feature maps, so that feature maps from different layers complement each other, such as U-Net++ (Zhou, Rahman Siddiquee, Tajbakhsh, et al., 2018), U-Net3+ (Huang, Lin, Tong, et al., 2020). The above methods have achieved certain performance improvements, but when they are used to extract cracks, multi-scale feature maps are not sufficiently reused. What is more, large-sized square pooling kernel is an effective method to extract long-distance context information, but this is not suitable for crack detection because it will inevitably incorporate irrelevant information from the background.

To address the shortcomings of previous works, we propose an improved U-Net model to aggregate multi-scale feature maps level by level and reduce the interference of the background to the target. The main contributions are listed as follows.

(1) In order to compensate for lost pixels during detection, it is effective to mirror padding the source image before convolution and filling feature maps from encoder before merging with feature maps from decoder.

(2) In different scale levels, the multi-scale feature maps are reused by aggregating the multi-scale features level by level.

(3) The residual linear attention module is designed to extract global contextual information, reducing information interference from the background. Its core is the strip pooling.

This paper is organized as follows: In Section 2, we give an overview of the state of the art regarding crack detection. Section 3 introduces an overall architecture of our proposed system. The experimental results and comparisons are presented and analyzed in Section 4. Finally, the conclusion is drawn in Section 5.

2. Related works

This section reviews a number of effective solutions proposed in recent years to solve the two shortcomings mentioned above.

For the first disadvantage, an overlap-tile strategy was proposed in Ronneberger et al. (2015). This method compensates for missing pixels by dividing the source image into some patches with overlapping parts. But, with the increase of depth, the calculation amount will raise rapidly, which will have a bad impact on the detection speed.

Some representative methods are listed below so as to overcome the second shortcoming. In order to detect cracks more accurately, Gao (2019) placed an atrous spatial pyramid pooling module and residual convolution blocks at the encoder of U-Net. Li, Wu, and Xu (2021) placed two channel spatial modules at the encoder of U-Net, aiming to make the model pay more attention to crack regions. Guo and Markoni (2021) designed a novel transformer-based refinement network to detect cracks, which has better performance than CNN architecture. A new model named APLCNet is proposed in Zhang, Chen, Wang, et al. (2020). It added a semantic segmentation branch in Mask R-CNN, it can extract the detail information of crack and improves the accuracy of crack mask prediction. In Song, Jia, and Jia (2019), the author establishing a multiscale dilated convolution module to extract the feature information in different scales. These modules help U-Net obtain contextual information in a wider range and improve its performance. The above approaches only optimize the backbone, but skip connection is also an important part.

Zhou, Qu, Li, et al. (2022) propose a mixed attention module and an effective decoder, they are able to extract more long-range dependency information of cracks and map the feature information into the pixel space. Pang, Zhang, Feng, et al. (2021) propose the residual separable convolution and place a semantic compensation module in skip connection to improve the detect ability of U-Net. Qu and Xie (2021) proposed an improved skip connection that fuses the feature maps before and after pooling. Its purpose is to enhance the description ability of feature information. Li, Zong, and Nie (2021) proposed a fully convolutional neural network for crack detection. It contains the densely connected layers are applied for enhancing the propagation and reuse of crack features, and the deeply supervised modules are designed to make network extract more significant features through multi-scale levels. Lu, He, Wang, et al. (2022) proposed a novel multi-scale crack detection network, called MSCNet, comprising a texture enhancement mechanism and feature aggregation to enhance the visual saliency of the objects in the background for bridge crack detection. Yang, Zhang, Yu, et al. (2020) proposed a new network architecture, named feature pyramid and hierarchical boosting network (FPHBN), for pavement crack detection. It integrates context information to low-level features for crack detection in a feature pyramid way and balances the contributions of both easy and hard samples to loss by nested sample reweighting in a hierarchical way during training. Zheng, Hu, Yang, et al. (2022) designed an AFFU-Net which an attention feature fusion network with a hybrid loss and residual refinement module. It is able to achieve automated winter jujube crack detection.

In this paper, we are inspired by the ideas of Zhou et al. (2018), Hou, Zhang, Cheng, et al. (2020) and Zhou et al. (2022) to improve U-Net with multi-scale feature map fusion module and residual linear attention module. This method does improve the performance of U-Net while keeping the detection time short.

3. Proposed model

This section introduce the detail of the proposed model. In Section 3.1, U-Net is implemented to detect cracks and the best number of pooling layers is determined. In addition, an effective solution to compensate for the pixel loss from convolution and pooling layers is proposed. But U-Net merging the low-level and high-level feature maps directly, does not fully reuse feature maps of different scales. Thus, in Section 3.2, the multi-scale module is designed to connect feature maps of different scales. However, the above methods are all use square kernels to extract the contextual information. So in Section 3.3, the residual linear attention module is designed to extract long-distance context information and avoid the interference of irrelevant information.

3.1. U-net with mirror padding

The purpose of this section is to implement the crack detection by U-Net. Furthermore, the first shortcoming will be restored. U-Net adopts an encoder–decoder architecture, and fuses the low-level feature map from the encoder with the high-level feature map from the decoder. First, the abstract feature will be extracted from the input image through convolution layers. Then, pooling layers are used to retain the important feature while reduce the amount of computation. Moreover, the purpose of up-sampling layers is to restore the same resolution as the input image. Finally, the prediction of each pixel is our desired results. As mentioned above, pooling layers are essential components of U-Net. However, too many pooling layers will result in the loss of target information. Too few pooling layers will increase the complexity of U-Net and reduce the description ability of features. Therefore,it can be seen that four is the best number of pooling layers by comparing the detection results of different U-Nets. Its architecture is shown in Fig. 1.

The network shown in Fig. 1 can detect cracks in the image, but the resolution of the output image is smaller than that of the input image. For the convenience of expression, the difference between the resolution of the input image and that of the output image is denoted as Resolution Difference in the following text. An over-lap-tile strategy is proposed in Ronneberger et al. (2015) to compensate the Resolution Difference. However, according to the data listed in Table 1, it can be inferred that when the value of the Resolution Difference is larger, the above strategy will lead to an increase in the resolution of image patches and a decrease in the speed.

By analyzing the architecture of U-Net, the main cause of the above problems are convolution layers and pooling layers existing in

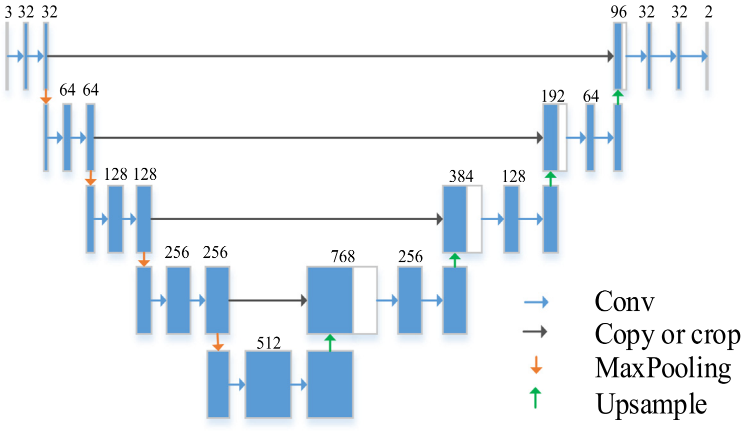
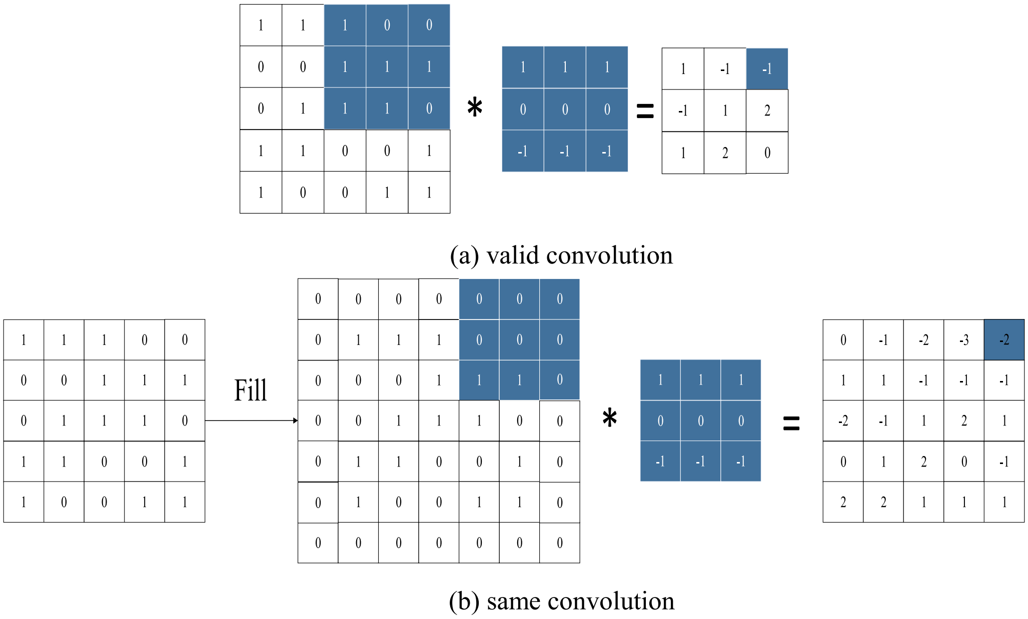


Fig. 1. U-Net contains four pooling layers.



(a) valid convolution

(b) same convolution

Fig. 2. Valid and same convolution.

Table 1
The different values of Resolution Difference.

The number of pooling layers	The value of Resolution Difference
2	16~17
3	40~43
4	88~95
5	184~199
6	376~407

U-Net. The proposer of U-Net mentioned in his paper (Ronneberger et al., 2015) that the convolution layers used in U-Net are all valid convolution layers. In other words, the input image does not undergo any previous processing before convolution. However, this method will lose some pixels at the edge of the source image, thus making the resolution of the output image smaller. The valid convolution is shown in Fig. 2(a), and the resolution of the output image can be universally formulated as

$$o = \frac{i - k + 2p}{s} + 1 \tag{1}$$

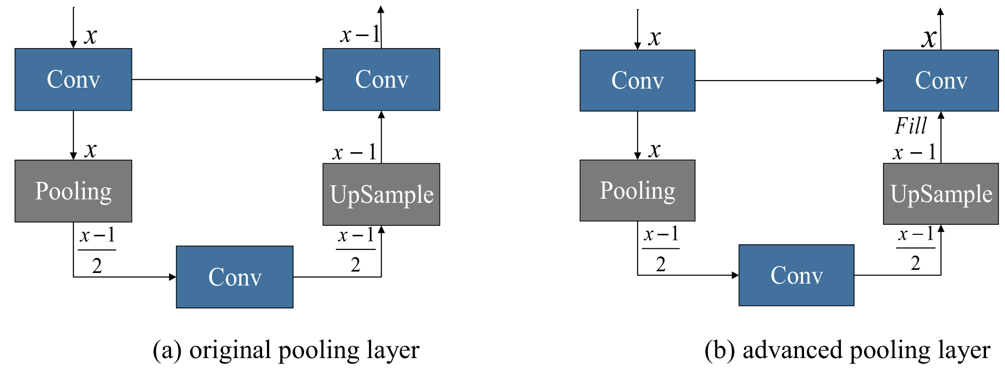
where o and i are the resolution of the output image and that of the input image, respectively. k is the size of convolution kernels, p is the number of rows or cols for mirror padding, and s is the distance that the convolution kernel slides each time. The proposer suggested setting k to 3, p to 0, and s to 1. According to Eq. (1), the difference between the resolution of the input image and that of the output image is 2. So it is necessary to mirror pad the input image before convolution. Its purpose is to keep the resolution changeless. The details are shown in Fig. 2(b).

Another important reason of the difference is pooling layers in U-Net. All pooling kernels are of size 2. In other words, the resolution of the output image is half of the resolution of the input image. But when the resolution of the input image is odd, the resolution of the output image is defined by (2).

$$o = \frac{i - 1}{2} \tag{2}$$

where o and i are the resolution of the output image and that of the input image. In other words, a row or column of pixels will be discarded. This is shown in Fig. 3(a). Aiming at this problem, a possible solution is to mirror pad the output image of the up sampling layer, as shown in Fig. 3(b).

After the above improvements, the resolution of the output image is equal to that of the input image. This method does not mirror-pad



(a) original pooling layer

(b) advanced pooling layer

Fig. 3. Pooling layers.

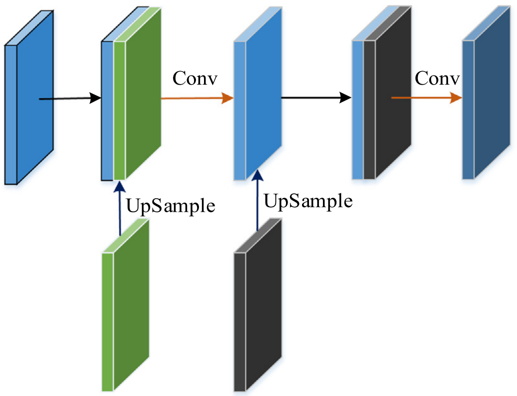


Fig. 4. Merging multi-scale feature maps.

the input image, but mirror-pad feature maps, which avoids the shortcomings of reducing the detection speed and increasing the memory footprint. This network is named Same U-Net, abbreviated as SU-Net.

3.2. Merging multi-scale feature maps

Since the appearance of U-Net, the encoder–decoder architecture has gradually been widely used. Its most significant advantage is the combination of low-level feature maps from the encoder and high-level feature maps from the decoder. But Zhou et al. (2018) mentioned, directly merging the two is not the best choice. Because there is an obvious gap between the two in semantic level and spatial resolution. Thus, the writer proposed corresponding strategies to solve these problems. Firstly, the model can more effectively capture fine-grained details of the foreground objects when low-level feature maps from the encoder are gradually enriched prior to fusion with high-level feature maps from the decoder. What is more, the network is able to obtain better prediction results when feature maps from the encoder and decoder are semantically similar.

Based on the above ideas, this paper designs a multi-scale feature map fusion module, the structure of which is shown in Fig. 4. This module is embedded in SU-Net that proposed in the previous section, resulting in a new network. Its architecture is shown in Fig. 5.

Compared with the original U-Net, it does not simply merge feature maps from encoder and decoder, but fuses multi-scale feature maps from encoder before fusing feature maps from decoder. The new model is able to capture more information about the target without a significant increase in weight. It is named Multi-Scale U-Net, abbreviated as MSU-Net.

3.3. Residual linear attention module

Related research on semantic segmentation shows that it is necessary to obtain local context information, but global context information is also indispensable. Pyramid SP, Atrous S Pyramid Pooling, Dilated convolution are classic and effective methods. However, when these methods are used to extract global context information, the shape of pooling kernel is usually N×N. But in some cases, the target may be long and narrow. They cannot be accurately extracted with a square pooling kernel, as it is inevitable to merge irrelevant information from

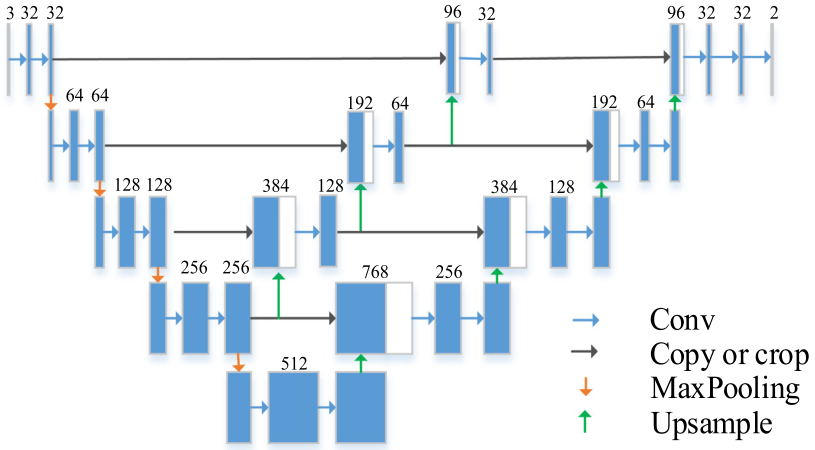
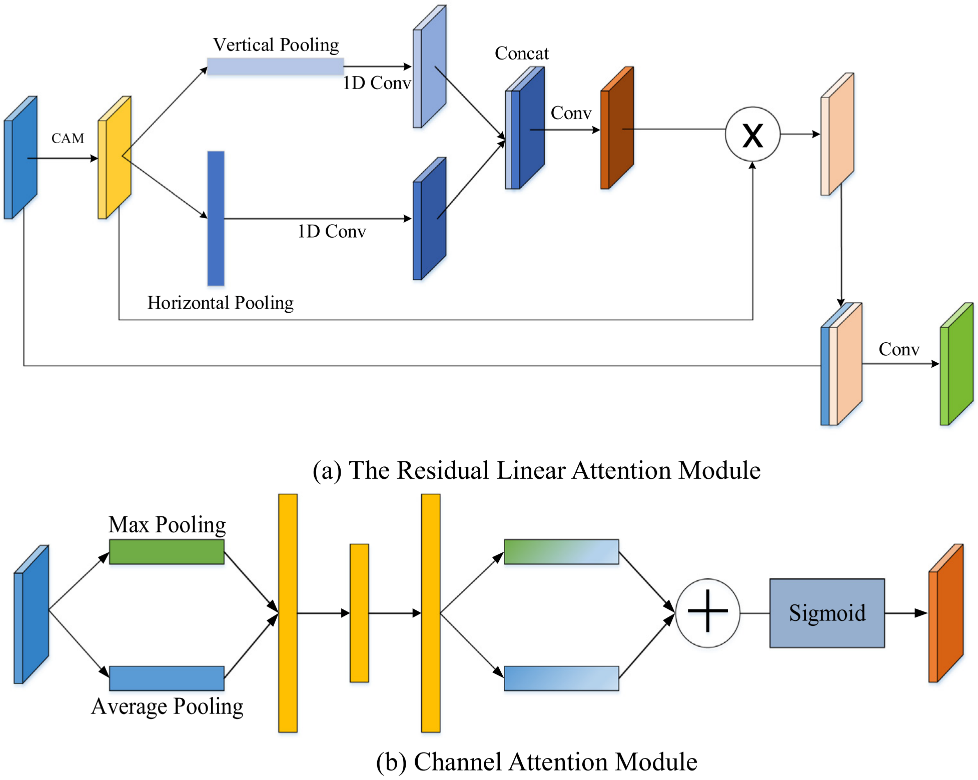


Fig. 5. MSU-Net.



(a) The Residual Linear Attention Module

(b) Channel Attention Module

Fig. 6. The Residual Linear Attention Module.

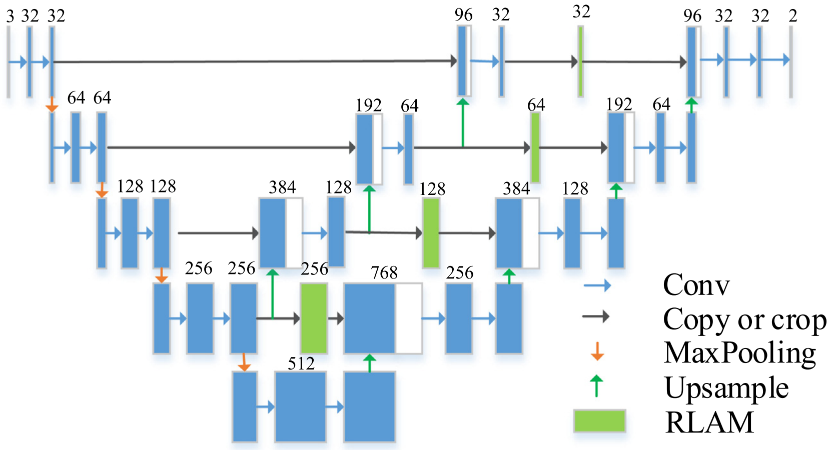


Fig. 7. RLAU-Net.

irrelevant regions with useful information from the target. Thus, the paper (Hou et al., 2020) proposed a residual linear attention module to solve the above problems. It has the following advantages. First, its kernel is narrow, which is beneficial to capture the linear target and reduce the impact of information from irrelevant areas. Second, the channel attention module and the improved spatial attention module can further highlight important features and suppress useless features. Finally, residual skip connections make it easier for the model to converge. The structure of this module is shown in Fig. 7. This module is embedded in MSU-Net, resulting in the newest model. By comparing results from different models, it is found that skip connection is the best location to place this module. So, the architecture of the newest model is shown in Fig. 8. Its name is Residual Linear Attention U-Net, or RLAU-Net for short (see Fig. 6).

4. Experiment preparation

4.1. Dataset

The dataset used in this experiment comes from many real concrete crack images collected by camera and assigns a class label to each pixel in every image. The original resolution of them is 5760*3840. They are divided into two parts, one is used for training and the other is used for testing. Considering the limitation of the hardware and the number of images for training is small, it is necessary to expand the number of images. First, the images used for training are divided into lots of patches by the sliding window algorithm, and then the patches containing cracks are screened out, about 500 in total. Finally, data augmentation was used to further expand the number of images, and

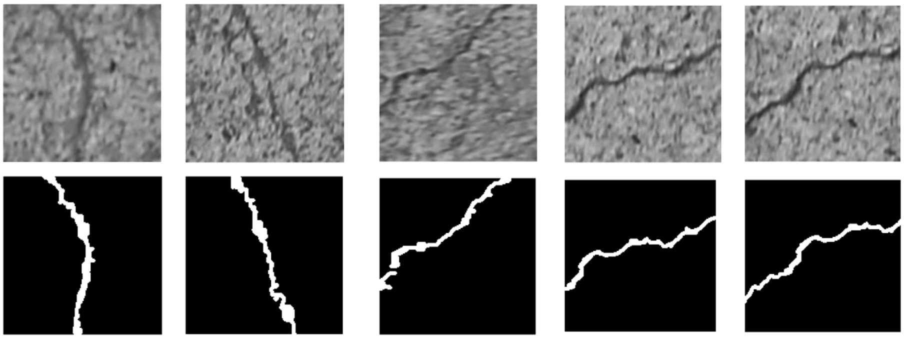


Fig. 8. Crack images and their labels.

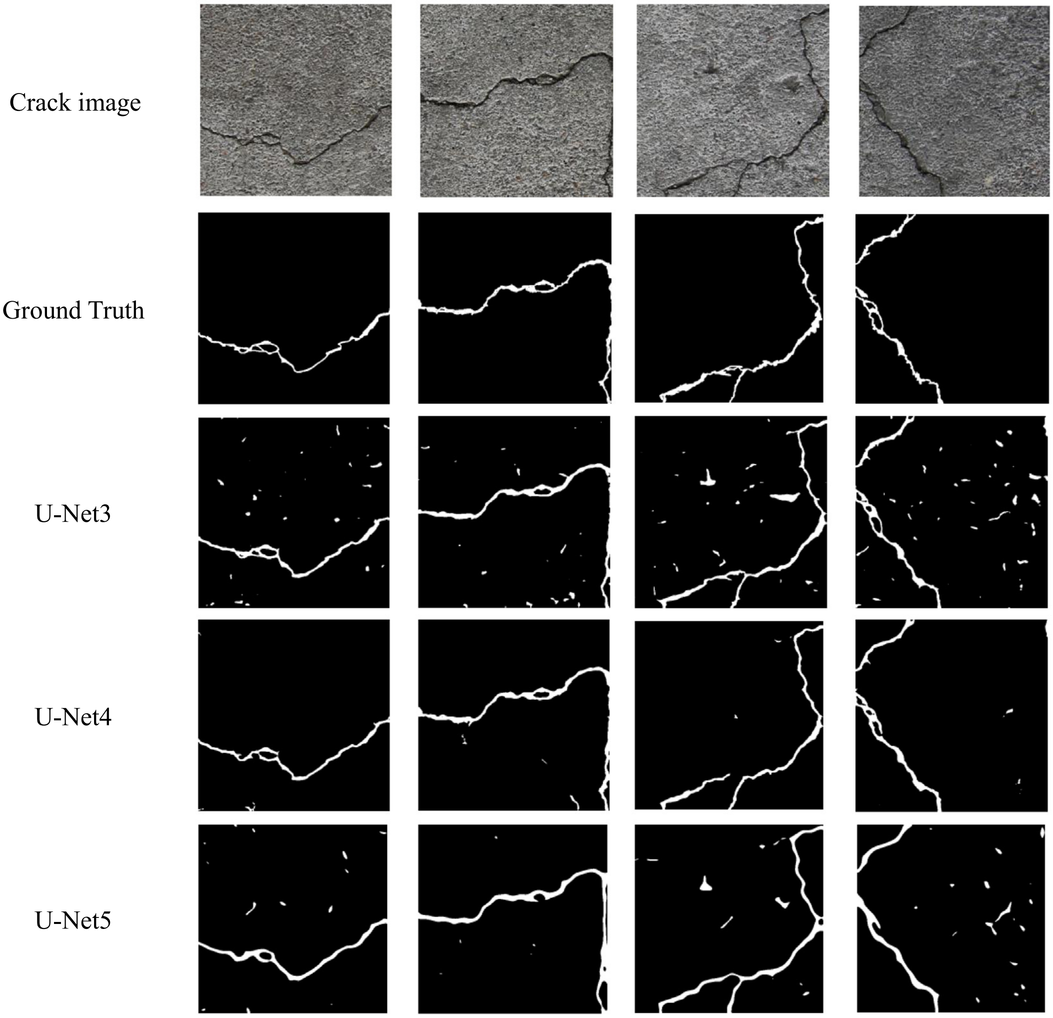


Fig. 9. The output images of multiple U-Nets with different number of pooling layers.

about 6000 crack images were obtained. What is more, the images used for testing were also subjected to the same processing as those used for training except for data augmentation. About 50 images were obtained. The experiments in this paper are all completed under the above dataset. Some crack images and their corresponding labels are shown in Fig. 8.

4.2. Experimental environment

This section mainly introduces the experimental environment, as shown in Table 2. It shows the software and hardware environment required for the experiment and some tricks used to train RLAUNet. Among them, the purpose of dynamically updating the learning rate is to make the network converge better. The learning rate will gradually decrease according to the exponential function model shown in (3).

$$ulr = \mathrm{c}lr * rate^{\frac{global}{step}} \tag{3}$$

where ulr represents the updated learning rate, clr represents the learning rate before dynamic update, $rate$ represents the weighting coefficient, which determines how fast the learning rate drops, $global$ represents the total number of iterations in the training process, and $step$ represents how often the learning rate is updated.

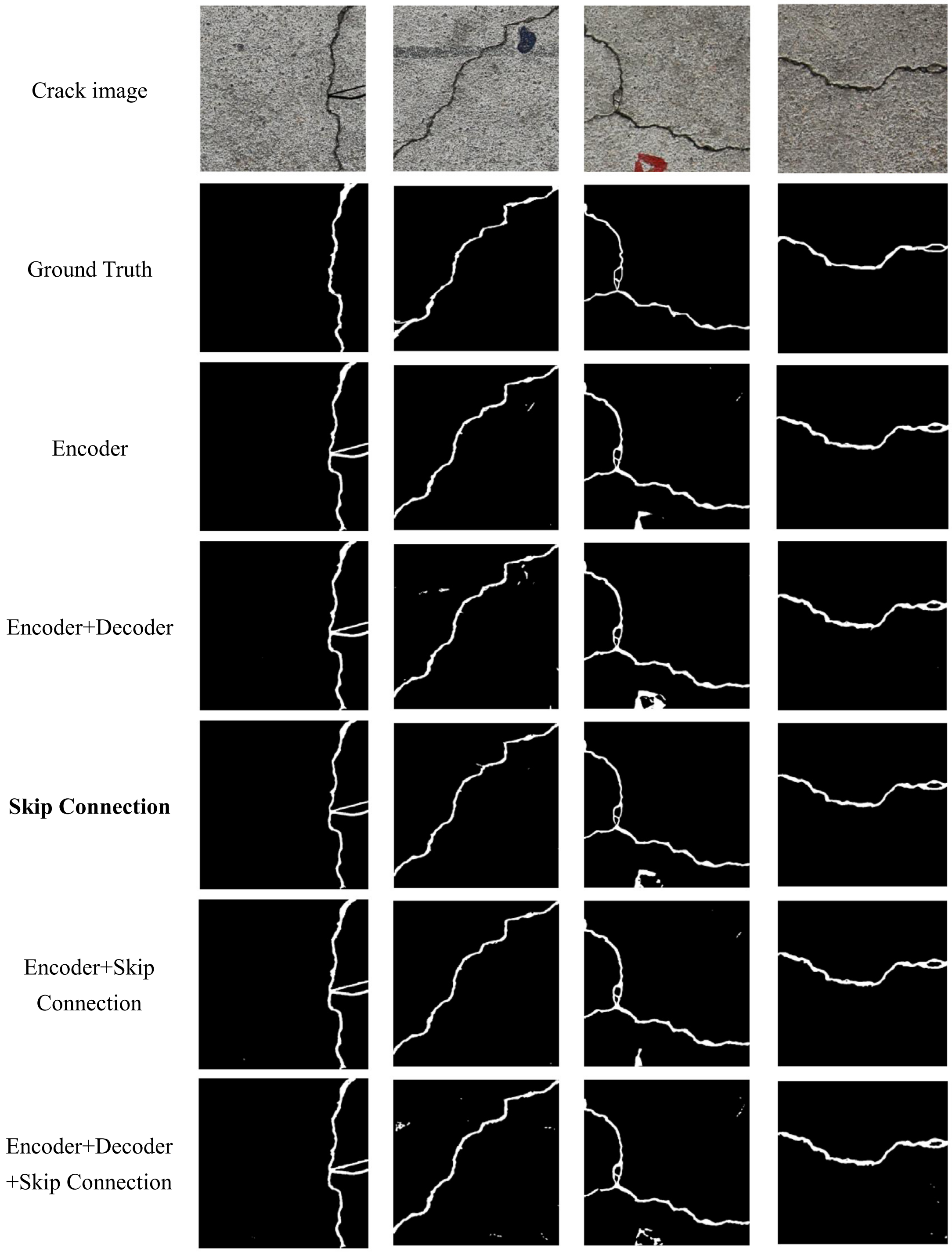


Fig. 10. The output images of multiple U-Nets with different locations.

Table 2
Experimental environment.

Experimental environment	Details	Parameters for network training	Details
CPU	Intel(R) Core(TM) i7-10870H CPU @2.20 GHz	Loss function	Softmax cross entropy
GPU	NVIDIA GeForce 2080 Ti	Penalty	L2 regularization
Running memory	16G	Penalty factor	0.0001
Programming language	Python	Optimizer	Adam
Software for compilation	Pycharm2019.1	Batch Size	30
Deep learning framework	Tensorflow1.13.1	Learning rate	Dynamic update the learning rate
CUDA	10.0	Initial learning rate	0.0003

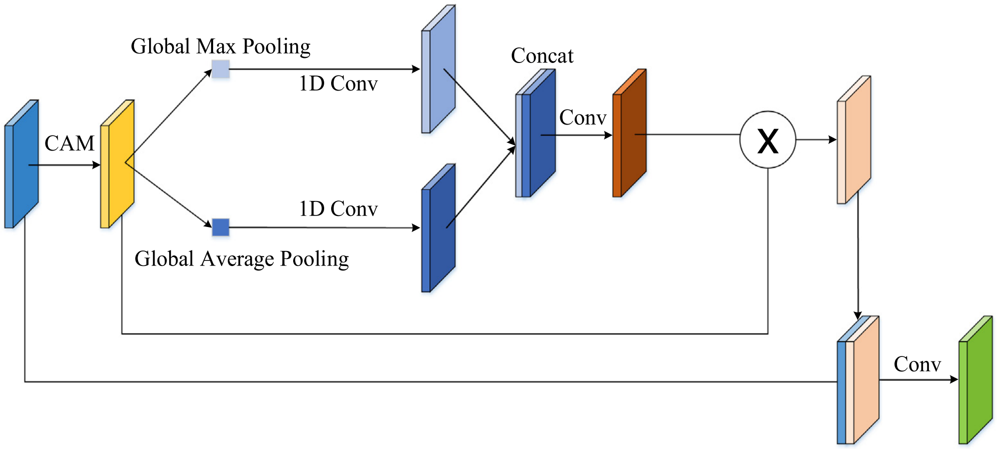


Fig. 11. The Residual Global Attention Module.

Table 3
The performance of multiple U-Nets with different number of pooling layers.

Models	P	R	F	cIou	mIou
U-Net3	60.05%	**81.75%**	68.50%	52.50%	75.32%
U-Net4	**73.68%**	80.43%	**76.09%**	**61.66%**	**80.22%**
U-Net5	36.63%	45.89%	40.38%	26.15%	61.44%

4.3. Evaluation metrics

Crack detection is a pixel-level classification task. Therefore, it is necessary to use pixel-level evaluation metrics to compare the performance of different models, mainly including precision (P), recall (R), f1 score (F), the intersection over union of crack (Iou(c)) and the mean intersection over union (mIou). Specifically, the metrics are defined as follows

$$P = \frac{TP}{TP + FP} \tag{4}$$

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

$$F = \frac{2 \times P \times R}{P + R} \tag{6}$$

$$Iou(\bullet) = \frac{TP}{TP + FP + FN} \tag{7}$$

$$mIou = \frac{Iou(c) + Iou(nc)}{2} \tag{8}$$

where TP, FP and FN are separately the true positive, false positive, and false negative based on the prediction and the ground truth.

5. Experiments and results

5.1. The best number of pooling layers

As mentioned above, the number of pooling layers will have a certain impact on the performance of the model. Therefore, multiple U-Nets with different number of pooling layers are designed in this experiment, and the best one is selected by comparing their performance. They are U-Nets contain three, four, five pooling layers, named U-Net3, U-Net4, U-Net5, respectively. The performance of them and the corresponding output images are shown in Table 3 and Fig. 9, respectively.

It can be inferred from Table 3 that the best performance is achieved when the number of pooling layers is 4. The accuracy rate reaches more than 70%, the recall rate and the mean intersection over union reaches more than 80%. The same conclusion is also shown in Fig. 10. When U-Net contains four pooling layers, the number of errors is minimized, both cracks can be detected completely and the edge of crack can be accurately delineated.

Table 4
The different location of residual linear attention module.

The position of residual linear attention module	P	R	F	Iou(c)	mIou
Encoder	75.66%	80.28%	77.24%	63.20%	81.03%
Encoder+Decoder	71.24%	82.21%	75.69%	61.22%	79.96%
Skip Connection	**76.16%**	81.27%	**78.21%**	**64.47%**	**81.69%**
Encoder+Skip Connection	75.43%	80.87%	77.38%	63.37%	81.11%
Encoder+Decoder+Skip Connection	70.75%	**83.50%**	76.00%	61.56%	80.14%

5.2. The best location of residual linear attention module

We know that it is beneficial to combine MSU-Net with the residual linear attention module. But it is difficult to determine the best location to place this module in MSU-Net. So, the residual linear attention module is placed in five different positions of MSU-Net, and the best one is selected for building RLAU-Net.

The main function of residual linear attention module is to extract the global context information. So it can be placed in the encoder, decoder and skip connection in MSU-Net. Putting it in the encoder helps the model to obtain more descriptive features, putting it in the decoder is beneficial to map the features into every pixel, and putting it in the skip connection can fuse the global context information with feature maps from the decoder. In summary, this paper places the module in the following positions. First, place it behind each convolution block in the encoder. Second, place it behind each convolution block in the encoder and the decoder. Third, place it in the skip connection. Fourth, place it in the skip connection and behind each convolution block in the encoder. Finally, place it in the skip connection and behind each convolution block in the encoder and decoder. When this module is placed in the above positions, the performance of different models are shown in Table 4 and Fig. 10.

According to the data shown in Table 4 and Fig. 10, it can be inferred that the best location is skip connection. Because the performance of this model is better than the others. The accuracy rate, F1 score, the intersection over union of crack and the mean intersection over union are all reach the max value. By comparing the first and fourth rows of Table 5, it can be concluded that the value of each evaluation metric is basically unchanged. When the module is placed in the two positions described in the second and fifth rows, the recall rate is slightly higher than that in other positions, which can reach 83.5%. However, the accuracy rate has dropped significantly, and the number of errors has increased. Based on the above analysis, the best location is the skip connection.

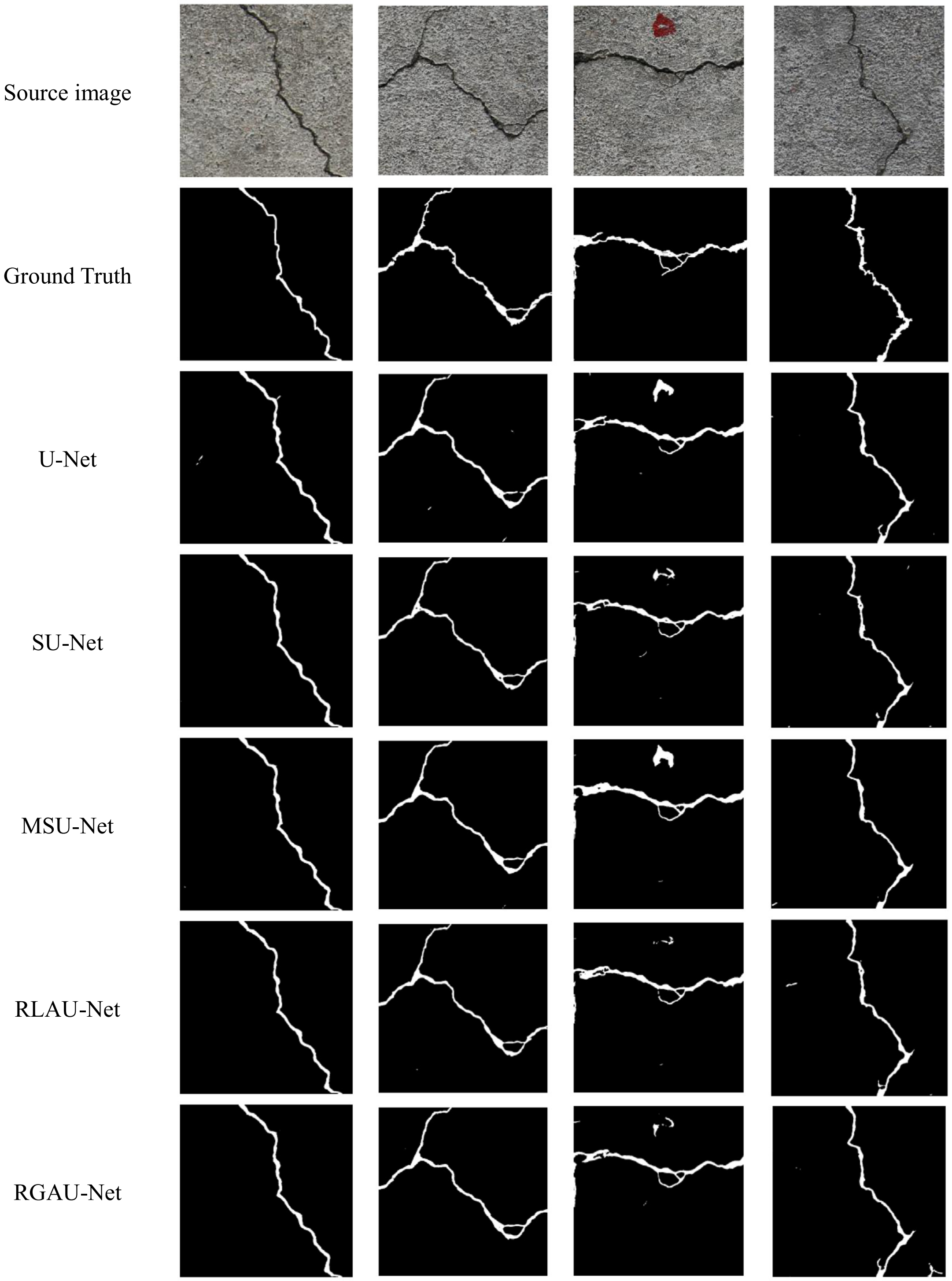


Fig. 12. The output images of five different models.

5.3. *Ablation experiment*

The purpose of this section is to demonstrate that the residual linear attention module and merging multi-scale feature maps are all useful to improve the performance of the model. Thus, five different models are designed in this experiment, namely U-Net, SU-Net, MSU-Net, RLAU-Net, and RGAU-Net, the effectiveness of the improvement is verified by comparing their detection capabilities. Among them, RGAU-Net is a new network obtained by replacing the strip pooling with the global pooling, which is used to verify the superiority of the strip pooling. Its core is the residual global attention module shown in Fig. 11.

The performance of five different models are shown in Table 5, and the output images are shown in Fig. 12. By comparing the data in Table 5, it can be seen that after a series of improvements, the value of each evaluation metric has gradually increased and the recall, F1 score, the intersection over union of crack and the mean intersection over union of RLAU-Net are better than others. In addition, by comparing the fifth and sixth rows in Table 6, it can be inferred that every evaluation metric of RGAU-Net is slightly lower than that of RLAU-Net, which verifies that the strip pooling is more beneficial to extract the linear object than the global pooling. More importantly, in order to avoid the influence of accidental errors, this paper makes a statistical test on

Table 5
Ablation experiment.

Models	P	R	F	cIou	mIou
U-Net	73.68%	80.43%	76.09%	61.66%	80.22%
SU-Net	**77.82%**	76.94%	76.80%	62.63%	80.75%
MSU-Net	75.18%	80.89%	77.20%	63.18%	81.01%
RLAU-Net	76.16%	**81.27%**	**78.21%**	**64.47%**	**81.69%**
RGAU-Net	75.90%	79.21%	76.94%	62.78%	80.81%

Table 6
The time spent by different models.

Models	580×580	1024×768	1440×960	1920×1080	5760×3840
U-Net3	**0.927 s**	2.2853 s	**3.646 s**	7.010 s	**59.3009 s**
U-Net4	0.945 s	2.2971 s	3.718 s	7.1553 s	60.4672 s
U-Net5	1.021 s	2.7736 s	4.2372 s	11.5642 s	68.8184 s
SU-Net	0.9105 s	**2.1731 s**	3.7033 s	**6.3462 s**	59.6930 s
MSU-Net	0.9224 s	2.2087 s	3.7692 s	6.4131 s	60.5858 s
RLAU-Net	0.9305 s	2.2519 s	3.8297 s	6.6161 s	61.77 s

the experimental data, and the results show that the performance of RLAU-Net is obviously better than that of the original U-Net.

5.4. The speed of different models

This section investigates the influence of the above improvements on the speed. In order to realize this purpose, four different models are used to detect different images. They are U-Net, SU-Net, MSU-Net, and RLAU-Net. Then, their average time-consuming is compared to verify that the optimizations in this paper are beneficial to improve the performance of U-Net without affecting the speed too much.

Table 6 shows the time spent by different models. The data in the first, second and the third row in Table 6 show that as the number of pooling layers increases, the speed becomes slower. The data in the second, fifth and sixth row in Table 6 show that the above improvements have no significant negative impact on the speed. In addition, the speed of U-Net4 is basically same as that of RLAU-Net. But, the performance of RLAU-Net is significantly better than that of U-Net4. Thus, RLAU-Net is an excellent variant of U-Net.

6. Conclusion

In this research, we executed a new model named RLAU-Net, which is a modified version of the original U-Net, to detect cracks from the surface of buildings. The proposed model has two advantages, one is the fusion of multi-scale feature maps, and the other is the use of linear pooling kernels to extract global context information. We trained our model based on the crack dataset containing over 6000 images. After experiments, the quantitative results demonstrated that the proposed model can enhance the results of mIou to 81.69%. In addition, F1 score has increased to 78.21%, the Intersection Over Union of crack has increased to 64.47%. The above improvements confirmed the strip pooling kernel is beneficial to extract the linear object. And the average time-consuming of the proposed model is basically equal to that of the original U-Net, which proved the proposed improvement does not adversely affect the speed.

Funding

The Key Research and Development Plan of Shaanxi Province, China (No. 2020GY-058), funded this research.

CRediT authorship contribution statement

Chenglong Yu: Algorithm design, Verification. **Jianchao Du:** Guide experiments. **Meng Li:** Data analysis. **Yunsong Li:** Program guidance. **Weibin Li:** Writing guidance.

Declaration of competing interest

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

Data availability

The authors do not have permission to share data.

References

Berwo, M., Fang, Y., Mahmood, J., et al. (2021). Automotive engine cylinder head crack detection: Canny edge detection with morphological dilation. In *Asia-pacific signal and information processing association annual summit and conference* (pp. 1519–1527).

Chen, L., Papandreou, G., Kokkinos, I., et al. (2014). Semantic image segmentation with deep convolutional nets and fully connected CRFs. https://arxiv.org/abs/1412.7062.

Chen, L., Papandreou, G., Kokkinos, I., et al. (2018). DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*, 834–848.

Chen, L., Papandreou, G., Schroff, F., et al. (2017). Rethinking atrous convolution for semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. https://arxiv.org/abs/1706.05587.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1800–1807).

Gao, W. (2019). *Bridge crack detection method based on improved u-net network*. Xidian University.

Guo, J., & Markoni, H. (2021). Transformer based refinement networkfor accurate crack detection. In *The international conference on system science and engineering* (pp. 442–446).

He, K., Zhang, X., Ren, S., et al. (2016). Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hou, Q., Zhang, L., Cheng, M., et al. (2020). Strip pooling: Rethinking spatial pooling for scene parsing. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4002–4011).

Huang, H., Lin, L., Tong, R., et al. (2020). Unet 3+: A full-scale connected unet for medical image segmentation. In *IEEE international conference on acoustics, speech and signal processing* (pp. 1055–1059).

Huang, G., Liu, Z., Van Der Maaten, L., et al. (2017). Densely connected convolutional networks. In *IEEE conference on computer vision and pattern recognition* (pp. 2261–2269).

Li, R., Wu, D., Xu, K., et al. (2021). Pixel-level crack detection using an attention mechanism. In *The international conference on intelligent computing and signal processing, Vol. 6*.

Li, H., Zong, J., Nie, J., et al. (2021). Pavement crack detection algorithm based on densely connected and deeply supervised network. *IEEE Access*, *9*, 11835–11842.

Liu, K., Yan, H., Meng, K., et al. (2021). Iterating tensor voting: A perceptual grouping approach for crack detection on EL images. *IEEE Transactions on Automation Science and Engineering*, *18*, 831–839.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Lu, G., He, X., Wang, Q., et al. (2022). MSCNet: A framework with a texture enhancement mechanism and feature aggregation for crack detection. *IEEE Access*, *10*, 26127–26139.

Pang, J., Zhang, H., Feng, C., et al. (2021). The crack segmentation of dam based on U-net with separable residual convolution and semantic compensation. *Computer Engineering*, *47*, 306–312.

Qu, Z., & Xie, Y. (2021). Concrete pavement crack detection algorithm based on full U network. *Computer Science*, *48*, 187–191.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 234–241).

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. https://arxiv.org/abs/1409.1556.

Song, W., Jia, G., & Jia, D. (2019). Automatic pavement crack detection and classification using multiscale feature attention network. *IEEE Access*, *7*, Article 171001-171012.

Szegedy, C., Liu, W., Jia, Y., et al. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1–9).

Talab, A., Huang, Z., Xi, F., et al. (2019). Detection crack in image using otsu method and multiple filtering in image processing techniques. *Optik*, *127*, 1030–1033.

Xing, C., Huang, J., Xu, Y., et al. (2018). Research on crack extraction based on the improved tensor voting algorithm. *Arabian Journal of Geosciences*, *11*, 1–16.

Yamaguchi, T., & Hashimoto, S. (2010). Fast crack detection method for large-size concrete surface images using percolation-based image processing. *Machine Vision and Applications*, *21*, 797–809.

Yang, F., Zhang, L., Yu, S., et al. (2020). Feature pyramid and hierarchical boosting network for pavement crack detection. *IEEE Transactions on Intelligent Transportation Systems*, *21*, 1525–1535.

Zhang, Y., Chen, B., Wang, J., et al. (2020). APLCNet: Automatic pixel-level crack detection network based on instance segmentation. *IEEE Access*, *8*, Article 199159-199170.

Zhao, H., Shi, J., Qi, X., et al. (2017). Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. https://arxiv.org/abs/1812.04103.

Zheng, Z., Hu, Y., Yang, H., et al. (2022). AFFU-net: Attention feature fusion U-net with hybrid loss for winter jujube crack detection. *Computers and Electronics in Agriculture*, (198).

Zhou, Q., Qu, Z., Li, Y., et al. (2022). Tunnel crack detection with linear seam based on mixed attention and multiscale feature fusion. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1–11.

Zhou, Z., Rahman Siddiquee, M., Tajbakhsh, N., et al. (2018). Unet++: A nested U-net architecture for medical image segmentation. In *The fourth deep learning in medical image analysis workshop*. https://arxiv.org/abs/1807.10165.

Zhou, X., Xu, L., & Wang, J. (2019). Road crack edge detection based on wavelet transform. In *IOP conference series earth and environmental science, Vol. 237*.