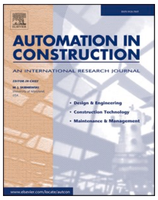
Contents lists available at ScienceDirect

Automation in Construction

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



Pavement crack instance segmentation using YOLOv7-WMF with connected feature fusion

Guanting Ye [a,c,1], Sai Li [b,1], Manxu Zhou [a,c], Yifei Mao [a,c], Jinsheng Qu [c], Tieyu Shi [a,c], Qiang Jin [a,c,*]

[a] College of Hydraulic and Civil Engineering, Xinjiang Agricultural University, Urumqi 830052, China
[b] College of Information Engineering, Xuzhou University of Technology, Xuzhou 221000, China
[c] Xinjiang Key Laboratory of Hydraulic Engineering Security and Water Disasters Prevention, Urumqi 830052, China

ARTICLE INFO

Keywords:
Concrete cracking
Instance segmentation
Deep learning
Convolutional neural network
YOLOv7

ABSTRACT

The detection and classification of concrete damage is essential for maintaining good infrastructure condition. Traditional semantic segmentation methods often can not provide accurate crack boundary information, which limits the further location and measurement analysis. In this study, the case segmentation method is used to solve the shortcomings of the previous detection methods and achieve more accurate crack identification results. This paper presents an improved YOLOv7 network design scheme. The network includes three different custom modules that can optimize the algorithm to solve missing feature problems, small recognition frames, and gradient problems, thereby improving accuracy. In addition, data sets with different sizes, exposures and noise are used to train the network, which expands the prediction range of the network and enhances the stability of the network. The experimental results show that compared with YOLOv7, YOLOv5, SOLOv2, Cascade Mask R-CNN, Condinst, Sparseinst, mAP is significantly improved. Thus, the proposed network algorithm has high practical engineering value.

1. Introduction

Cracking is one of the main defects exhibited by concrete structures, and it has become an important element when inspecting and repairing such structures [1]. Therefore, it is necessary to check the morphological changes and development trends of cracks by measuring their statuses and assessing the extents of their impacts on the target structure. Crack detection is important for performing daily building safety maintenance, rapidly assessing building damage after a disaster, and preventing the loss of life and property [2]. Manual inspection is a common crack detection strategy. However, manual methods are subjective, inefficient, laborious and dangerous. In addition, the accuracy and scope of manual detection are limited [3]. Therefore, the detection of concrete cracks using image processing or deep learning techniques has become a hot research topic [4]. With the rapid development of artificial intelligence, deep convolutional neural networks (CNNs) have been developed for automatic crack detection. This has opened a path to the development of an inexpensive, efficient and safe pavement inspection method [5]. Deep learning-based methods have been used in image classification [6], object detection [7] and pixel segmentation tasks [8], all of which are applicable to the crack detection problem. Classification-based methods have been widely used and have exhibited better performance than traditional image-based processing algorithms. Cha et al. [9] first combined a CNN model and the sliding window method to detect cracks in concrete surfaces. A training dataset containing 40,000 subimages with resolutions of 256 ± 3256 pixels was fed into the CNN model. The validation results showed that the trained neural network model yielded higher accuracy and robustness than an interactive multimodel. Chen et al. [10] and Cao et al. [11] used CNN-based methods on the same set of 40,000 images at a 227 × 227 resolution, and both studies achieved good recognition accuracies exceeding 99% and 90%, respectively. In addition, other studies [12] have implemented CNN-based automatic pavement crack identification models to distinguish between defective and noncracked concrete. These methods undoubtedly perform well in terms of automatic crack classification. However, they do not accurately locate cracks, and their usefulness for pavement maintenance and

[*] Corresponding author at: College of Hydraulic and Civil Engineering, Xinjiang Agricultural University, Urumqi 830052, China.
E-mail address: tm-jinq@xjau.edu.cn (Q. Jin).

[1] Guanting Ye and Sai Li contributed equally to this manuscript.

https://doi.org/10.1016/j.autcon.2024.105331
Received 6 July 2023; Received in revised form 30 December 2023; Accepted 7 February 2024
Available online 14 February 2024

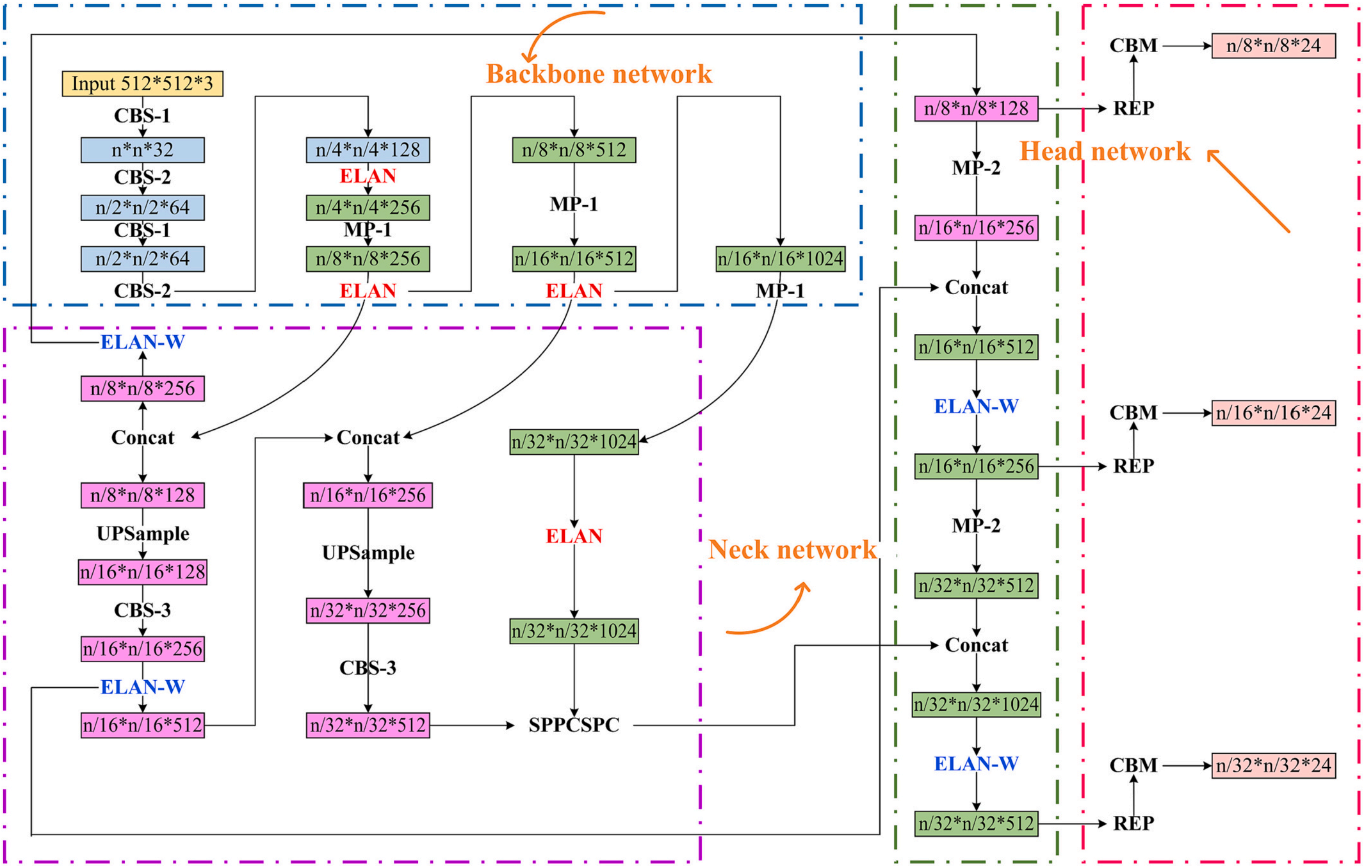


Fig. 1. YOLOv7 framework.

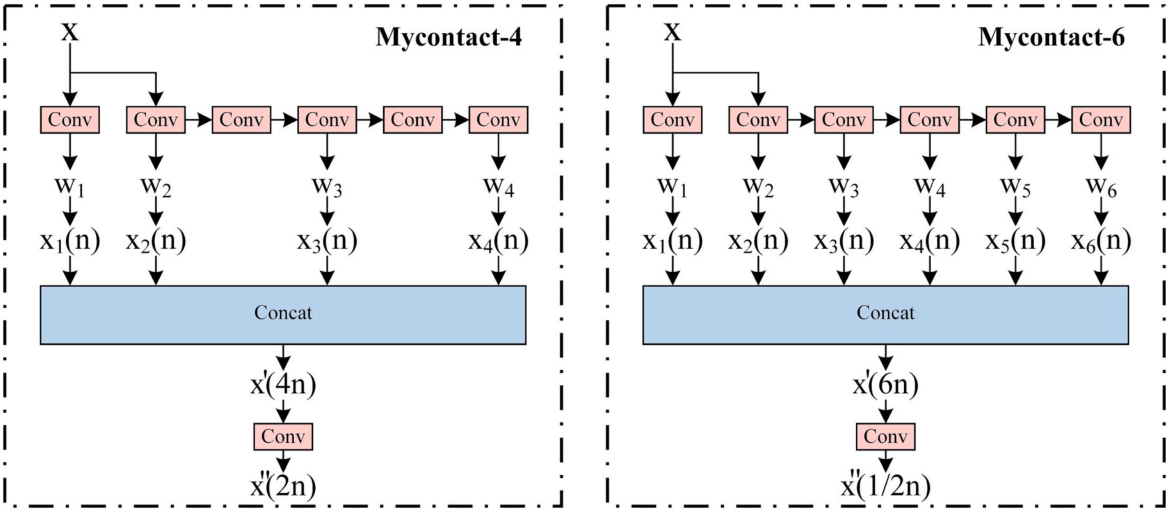


Fig. 2. Mycontact-4 module and Mycontact-6 module.

management appears to be limited.

Target detection solves the problem of how to localize and classify multiple targets in an image. Cha et al. [13] applied the Faster R-CNN approach to automatic crack detection and produced good results. Yu et al. [14] proposed a YOLOv4 model for bridge crack detection. In addition, Mohtasham Khani et al. [15] demonstrated that applying smoothing methods during preprocessing could significantly improve the performance of crack detection models. Notably, the sliding window approach is applied in all of the above algorithms. This detection method can only mark the type and position of each window. However, methods for detecting the distribution path, shape and density of the target crack do not provide highly accurate detection information concerning these crack aspects. To more accurately measure cracks, they must be detected at the pixel level.

The purpose of segmentation is to identify the target object at the pixel level. Pixel segmentation is divided into instance segmentation (i.e., pixel-level segmentation of each individual target object, which is mainly used for complex scenes with multiple target objects of the same type) and semantic segmentation (i.e., pixel-level segmentation of all target objects of the same type, which is mainly applied to recognize a

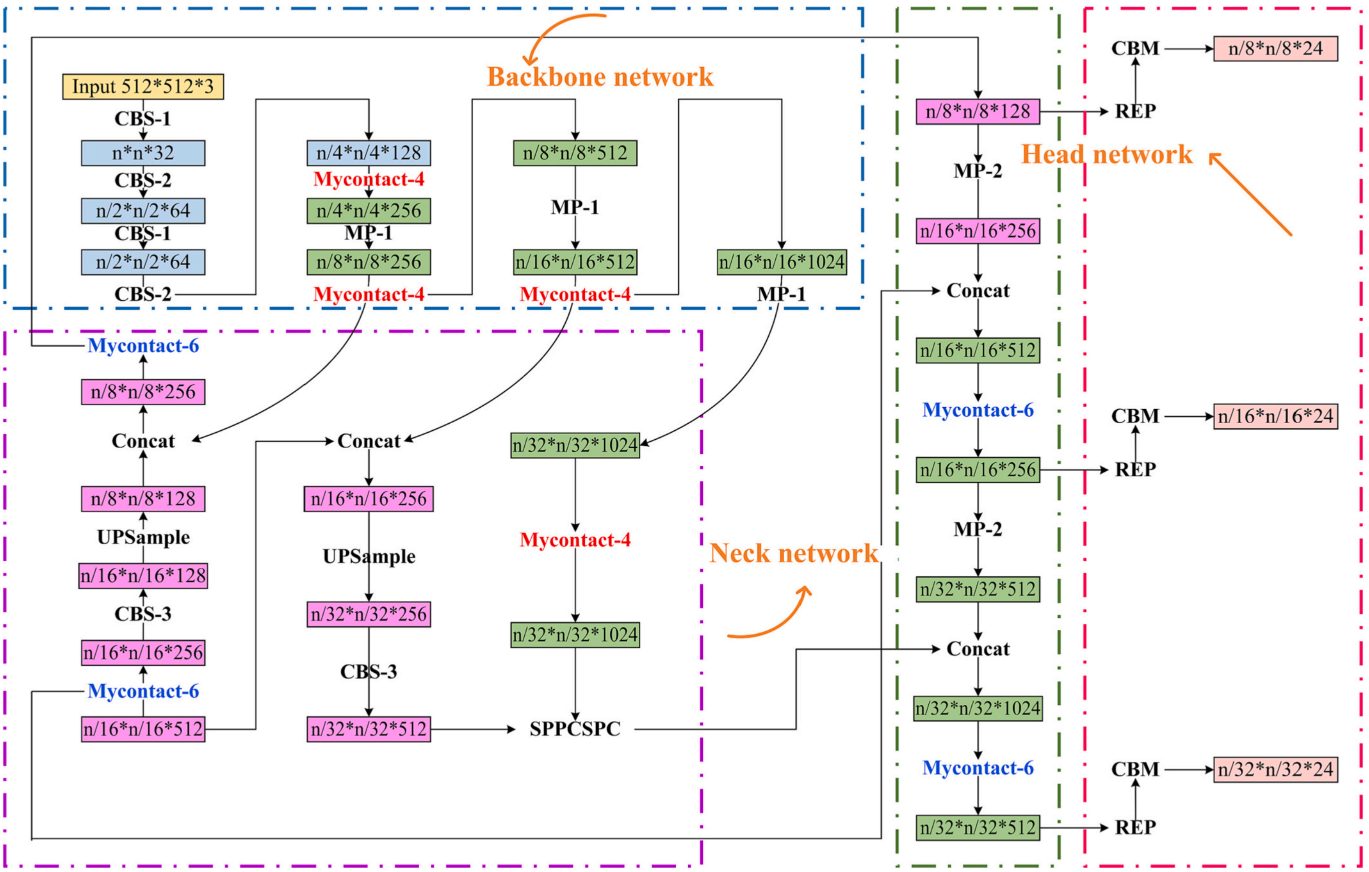


Fig. 3. A YOLOv7 framework with the Mycontact-4 and Mycontact-6 modules added.

few target objects). Chen et al. [16] proposed a novel neural network for pixel-level road crack detection and assessment that combines the advantages of an encoder-decoder network and an attention mechanism, and the network has excellent detection performance. Qu et al. [17] proposed a deeply supervised convolutional neural network for crack detection by means of a novel multiscale convolutional feature fusion module, which was validated on three public crack datasets, and the results showed that the model achieved better performance than that of the competing methods. Regarding concrete crack detection, much research has been conducted on semantic segmentation-based methods. For example, Zhang et al. [18] achieved improved crack detection performance by introducing dilation convolutions with different expansion rates and a multibranch fusion strategy to detect cracks. Bang et al. [19] proposed a method based on an encoder-decoder network for the automatic detection of road cracks at the pixel level. ResNet-152 with transfer learning was chosen as the encoder, but the experimental results did not achieve the expected accuracy. Fan et al. [20] proposed an integrated network for the automatic detection and measurement of road cracks. Zhang et al. [21] proposed a CNN-to-fully convolutional network (FCN) method to approximately localize cracks with a CNN and then segmented the cracks with an FCN. Xu et al. [22] proposed an improved fused CNN to identify cracks in complex images of the interiors of steel box bridge girders. By adding a bypass stage at the end of the regular stage, multilevel and multiscale image fusion can be performed. However, semantic segmentation-based crack detection methods have some limitations; e.g., they can only provide accurate information about the locations and extents of cracks but cannot distinguish between different cracks. Semantic segmentation-based approaches cannot fully detect some particularly fine cracks due to their insufficient segmentation performance and the small amount of available data. Instance segmentation can provide a good solution to this problem by assigning a unique identifier to each crack in the given dataset so that each crack can be treated individually for distinguishing the cracks and obtaining boundary information.

Therefore, this paper uses the YOLOv7 network from the You Only Look Once (YOLO) family of networks to segment cracks at the pixel level. In recent years, the YOLO architecture has achieved excellent results in computer vision and concrete crack detection applications in comparison with the state-of-the-art CNNs. Teng et al. [23] used 11 well-known CNN models as YOLOv2 feature extractors for crack detection purposes. The results confirmed that the YOLOv2 network uses different feature extraction models, leading to variability in its detection results. Nie and Wang et al. [24] used the YOLOv3 architecture to detect cracks in pavement images. The proposed architecture outperformed the accepted CNN in terms of the detection rate, but its accuracy was lower than that of the CNN approach. Peraka et al. [25] used YOLOv4 to detect and quantify the condition statuses of pavements collected by road agencies via a machine learning architecture and combined this method with a migration learning approach to identify multiple severity-based damage instances in the images. However, the training and prediction times of YOLOv4 are long, and its real-time performance still has room for improvement. Qu et al. [26] proposed an improved multiscale cross-layer feature fusion network based on the YOLOv5 method to mitigate the problem by which missed and false detections occur for large targets and to achieve better detection results on the PASCAL VOC and MS COCO datasets. However, YOLOv5 requires a large amount of training data to achieve improved accuracy. Ye et al. [27] proposed an improved YOLOv7 network that can better distinguish concrete cracks from numerous misleading targets to compensate for the shortcomings of the existing detection methods. Ma et al. [28] proposed a deep learning-based crack detection method with data collection and defect counting difficulties, as well as a system consisting of a pavement crack-based generative adversarial network (PCGAN) and a crack detection and tracking network named YOLO-MF, which has achieved excellent field

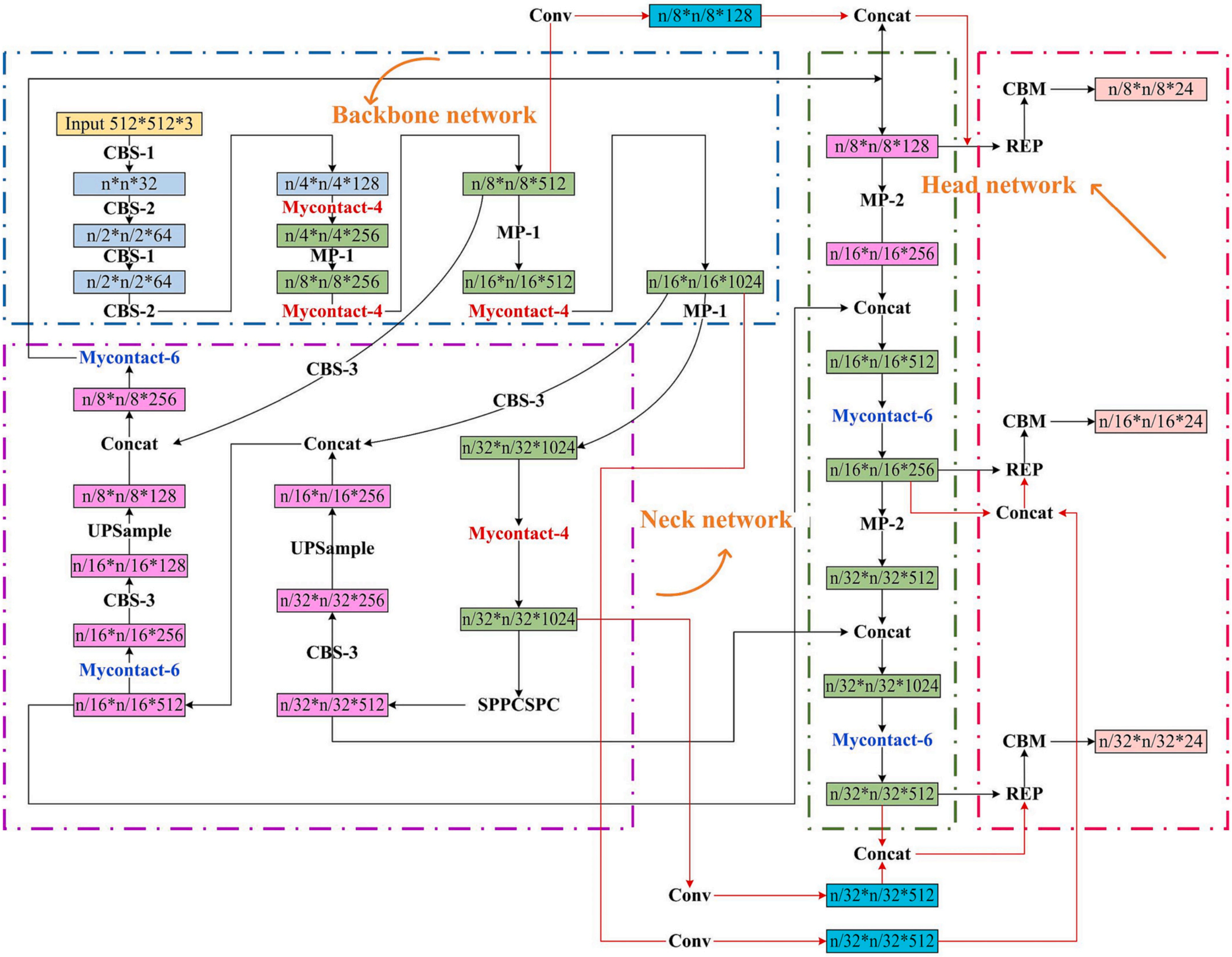


Fig. 4. The YOLOv7 framework in Fig. 3 with residual connections added.

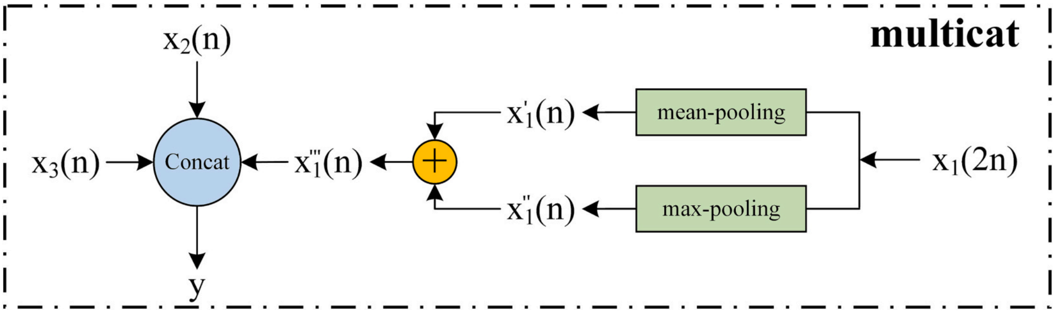


Fig. 5. The multicat module.

measurement results. While it is a fact that the YOLO series of networks have proven to be highly effective in object detection tasks, relying solely on a YOLO-based network for object detection fails to accurately capture detailed information about each crack. This limitation hinders our comprehensive understanding of crack features. Therefore, it is necessary to optimize the YOLO series of networks to provide detailed information about each detected crack while retaining the inherent fast detection capabilities of the YOLO framework.

In this paper, we propose a new pixel-level instance segmentation network named YOLOv7-Weights-Multicat-Fusion (WMF), which is specifically designed for accurate crack detection and comprehensive analysis purposes. This network incorporates new techniques to extract subtle crack details, thus enhancing its overall crack detection performance.

The main contributions of this study can be concluded as follows.

- Our innovation, as opposed to a semantic crack segmentation approach, is an instance-level segmentation approach. Unlike semantic segmentation, which focuses only on the overall segmentation of the crack region, our method is able to accurately segment

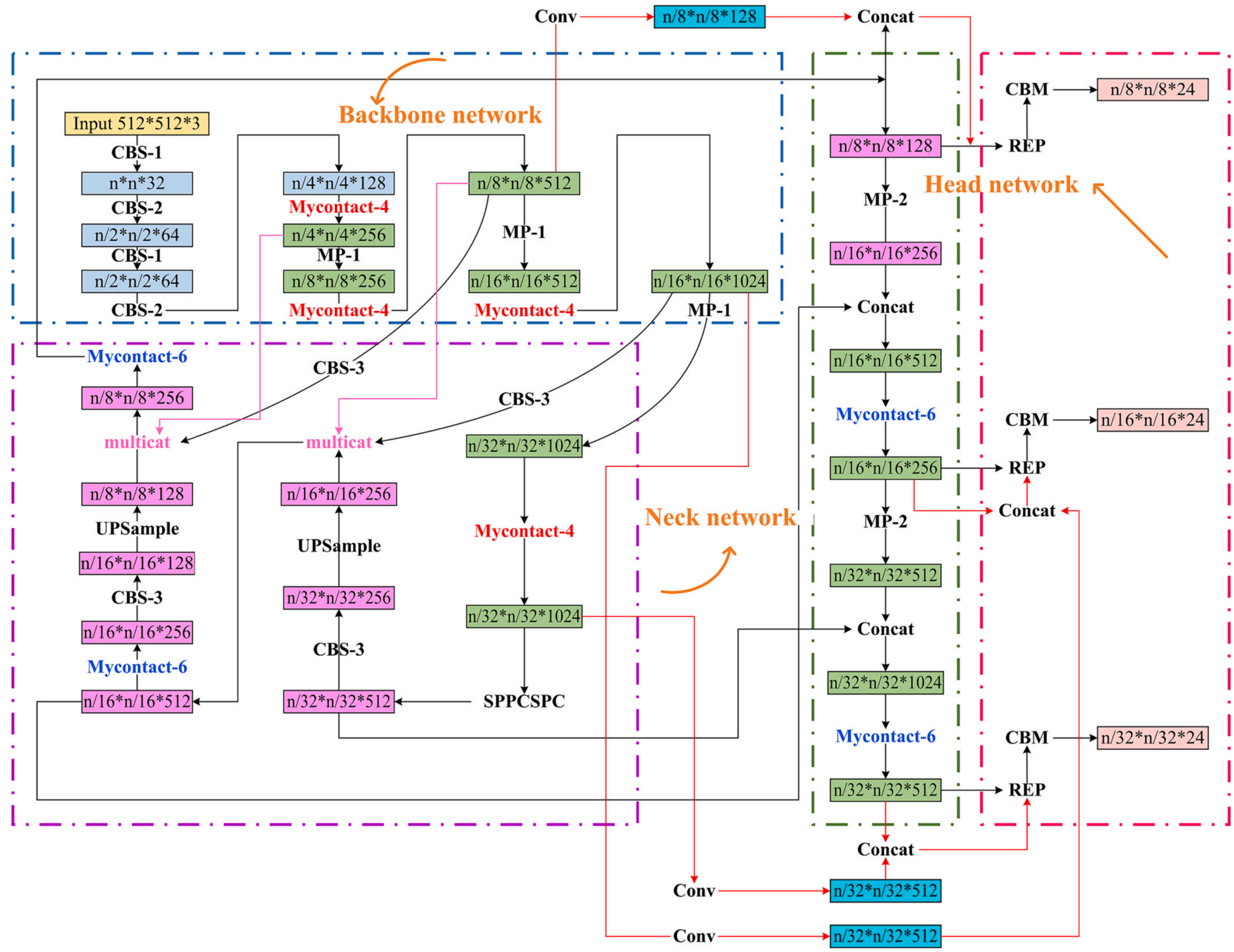


Fig. 6. The YOLOv7 framework in Fig. 4 with the Multicat module added on top.

Table 1
Number of cracks per type.

Dataset	Training set	Validation set	Total number
Crack dataset	960	411	1371
Longitudinal	357	137	494
Transverse	320	123	443
Fatigue	283	151	434

each individual crack instance. This provides important information for conducting more detailed crack localization and analysis processes, making our method more advantageous for practical applications.

- The original and the classic instance segmentation networks always produce incomplete recognition results and attain insufficient accuracy when detecting cracks in complex environments. For this reason, we develop three innovative modules and a unique connection and add them to the original network so that it can provide sufficient crack detail information and achieve higher accuracy and recognition even in complex environments.
- Our combination of object detection and segmentation takes full advantage of YOLOv7 as the primary object detection network. By fusing the instance segmentation task with YOLOv7, we are able to achieve both object detection and accurate crack segmentation, thus providing more comprehensive information about the morphologies and characteristics of cracks. This combination makes our approach superior to the competing methods in terms of crack identification and analysis.
- We also innovate in terms of the utilized training datasets and data enhancement process. We construct a high-quality segmented dataset consisting of fracture instances and employ effective data enhancement strategies. These strategies not only improve the generalization and robustness of the proposed model but also enhance its ability to learn from various crack instances, thus further improving the performance of our instance segmentation cracking method.

2. Theoretical background

2.1. YOLOv7-WMF architecture

The structure of YOLOv7 is based on a series of downsampling and upsampling layers. YOLOv7 enables the network to learn multiscale layered features by increasing the number of feature maps and decreasing the spatial resolution of the input image, thus accurately detecting objects with different sizes in the input image. This framework is shown in Fig. 1.

The architecture of YOLOv7 uses a convolutional layer as its first layer to process the input image and extract low-level features. This is followed by a series of downsampling layers that reduce the spatial

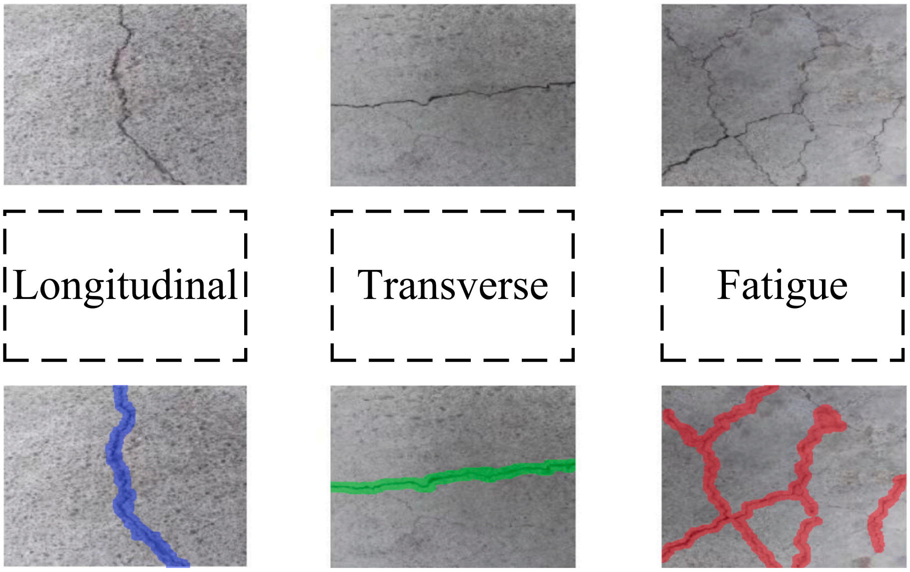


Fig. 7. Three typical fracture types.

Table 2
Number of labels for each type of crack.

Dataset	Validation dataset	Training dataset	Total number
Crack dataset	480	1984	2464
D00	116	512	628
D10	227	873	1100
D20	137	599	736

resolution of the feature map by applying 2 convolutional layers. The downsampling factor is 2 in each spatial dimension, thus increasing the number of feature maps while reducing their spatial dimensionality. In the YOLOv7 architecture, the downsampling layers are followed by a series of upsampling layers that are designed to reduce the number of feature maps while increasing their spatial resolution. The upsampling layers are usually implemented using transposed convolutional layers, which perform the upsampling operation by inserting zeros between the elements of the input feature map and convolving the generated tensor using a set of filters.

To achieve improved object detection accuracy, the YOLOv7 architecture also includes some jump connections. The role of these jump connections is to combine the features acquired from different layers. They do this by adding the output of one layer to the input of a later layer and are typically used to fuse features derived from a downsampled layer with the features obtained from an upsampled layer.

In addition to convolutional and upsampling layers, the YOLOv7 architecture introduces the concept of anchor frames. Anchor boxes are predefined bounding boxes that are used to detect objects in the input image. These anchor frames are learned during the network training process and are used to predict the positions and sizes of the objects in the input image. In addition, the YOLOv7 architecture includes classification and regression layers for predicting the category labels and bounding box coordinates of each anchor box.

2.2. Mycontact model structure

To make the network model more accurate in terms of concrete crack identification, the original network model is modified in this study. First, its ELAN module is improved, as shown in Fig. 2. The purpose of this module is to stitch multiple input tensors according to some dimension (the first dimension by default). In the newly constructed Mycontact-4 module, each tensor input from the CBS to the concatenation layer is first multiplied by a training-derived weight. The weights are defined via the NN parameters in PyTorch, which means that they are part of the model and can be obtained via training. The weights are set to 1 at initialization, and then in each forward propagation step, the weights are normalized to ensure that they sum to 1. This can be seen as a method for performing a weighted average calculation on the input tensor. Through this operation, multiple feature maps can be combined according to their specific weights.

The two modules in Fig. 2 are very similar, the difference being that Mycontact-4 is used to process 4 input tensors, while Mycontact-6 is

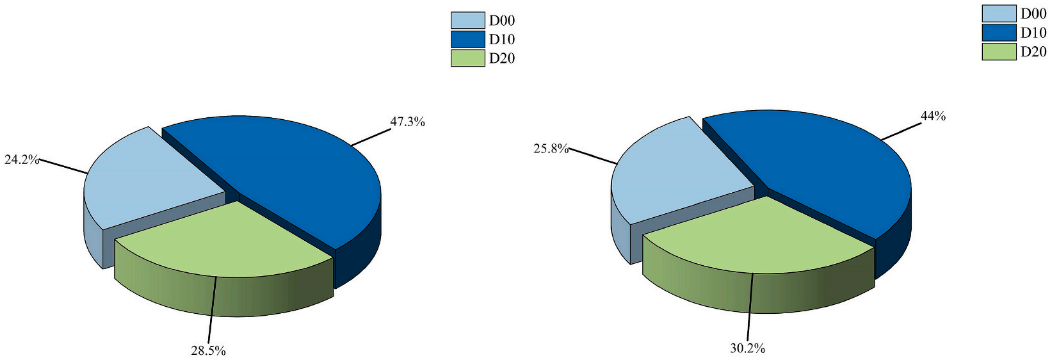


Fig. 8. The proportion of the number of labels for each type of crack.

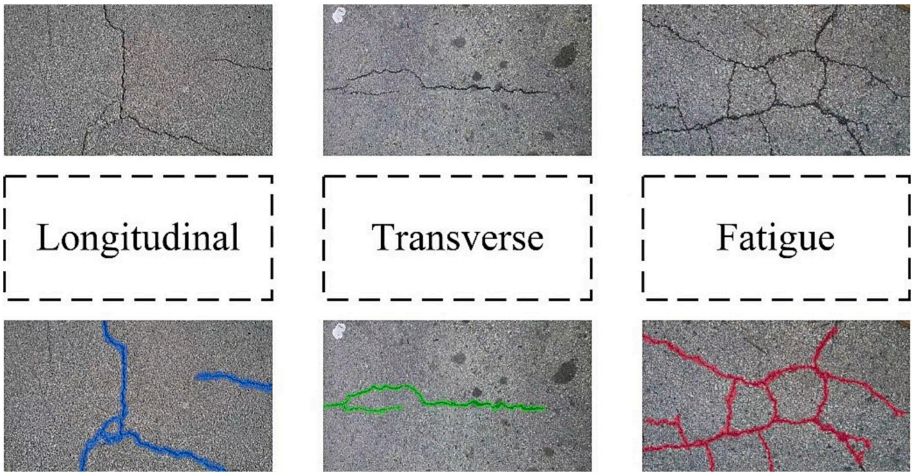


Fig. 9. The crack map of crack 500 is shown as an example.

Table 3
The results of the comparative studies involving different modules.

The utilized network	Box			Seg		
	Precision	Recall	mAP_{50}	Precision	Recall	mAP_{50}
W	89.30%	89.80%	93.47%	80.83%	80.97%	79.74%
WM	90.61%	86.70%	93.67%	84.00%	80.19%	80.89%
WF	89.93%	88.47%	93.41%	81.68%	82.81%	80.98%
M	89.56%	82.09%	91.69%	83.65%	74.52%	76.82%
MF	90.92%	85.12%	91.68%	81.63%	76.98%	78.58%
F	90.62%	87.57%	92.52%	83.55%	80.47%	80.39%
YOLOv7	82.88%	82.08%	93.78%	71.46%	70.30%	70.45%
YOLOv7-WMF	89.48%	88.25%	94.15%	84.59%	85.81%	83.09%

used to process 6 input tensors. That is, four feature maps are weighted in Mycontact-4, and six feature maps are weighted in Mycontact-6. By weighting the feature maps, more useful feature information can be extracted, as shown in Fig. 3.

In addition, residual connections are added to the approach developed in this study, as shown in Fig. 4. We resize each feature map output by the Mycontact-4 module via a convolutional layer and stitch it with the output of the corresponding Mycontact-6 module. The result of this splicing process is then fed into the REP module along with the output of the Mycontact-6 module. The same operation is performed again in the subsequent Mycontact-4 and Mycontact-6 modules.

This series of operations aims to fuse the shallow and deep features of the input image. The shallow network extracts features that are closer to the input with smaller perceptual fields and smaller overlapping areas, thus capturing more details and pixel-level information. In contrast, the features extracted by the deep network are closer to the output, their perceptual fields are increased, and their overlapping areas are increased, thus obtaining the holistic information of the input image. By fusing these two groups of features, the dependence of the model on individual features can be reduced, and the stability and accuracy of the model can be improved. Feature fusion can also reduce the sensitivity of the model to noise and outliers and improve the robustness of the model. In addition, since the two feature groups come from different levels, redundant or complementary relationship may be present between them. Through feature fusion, these features can be integrated into a richer and more comprehensive feature representation, thus enhancing the representational power of the model.

2.3. *Multicat model structure*

The concatenation module is also modified. A new Multicat module is constructed in this study, as shown in Fig. 5; in addition to the original structure, this module also fuses information from the earlier Mycontact-4 module.

After this part of the information enters the Multicat module, both average pooling and maximum pooling are performed, and then the two parts are summed. By conducting average pooling, the average value of the pixel values in the target region can be calculated, and the overall distribution features can be extracted. The maximum pooling process, on the other hand, selects the most significant features in the region of interest and provides better responses for local features, such as edges and textures. This step aims to reduce the spatial dimensionality by dividing the input feature map into nonoverlapping regions and converging (by averaging or taking the maximum value) for each region. This helps reduce the numbers of computations and parameters and makes the network more robust to translations and spatial variations.

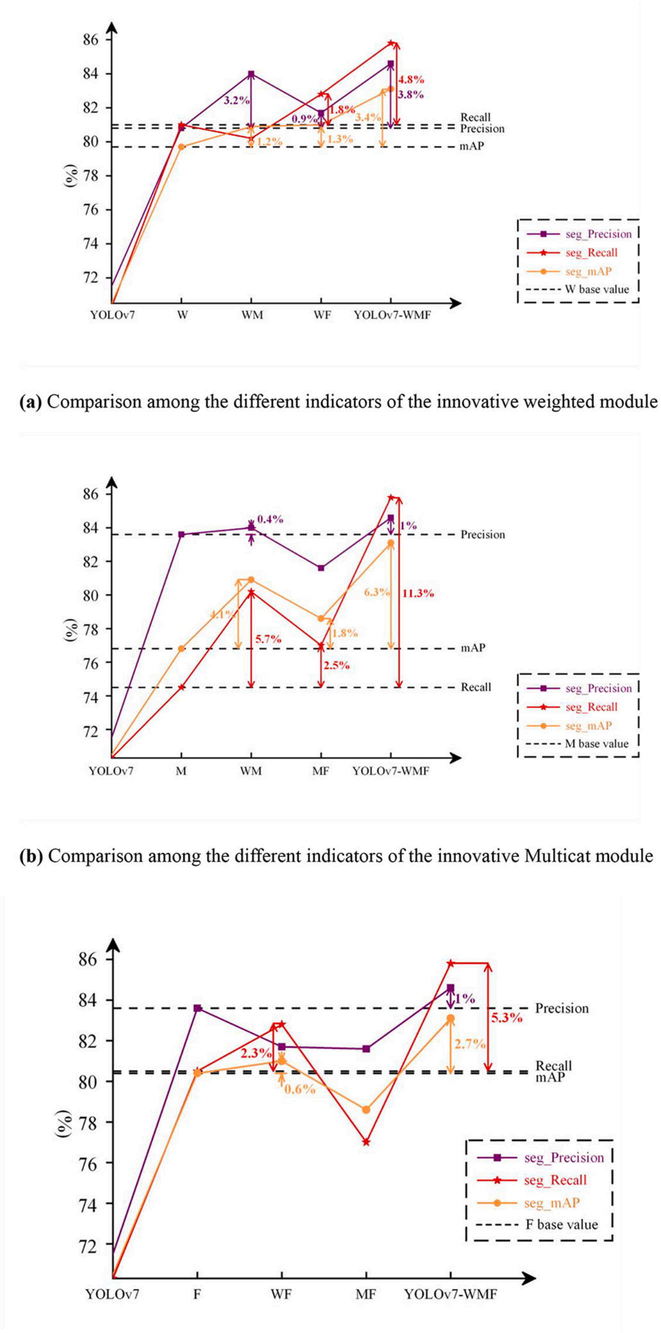
The multiscale fusion process executed through the Multicat module can obtain more comprehensive and richer feature representations and improve the ability of the model to represent the target object, as shown in Fig. 6. In addition, the perceptual field of the model can be expanded so that it can capture a wider range of scene information. By introducing features at different scales, the model can better understand the contextual and global information of the whole scene, thus improving its recognition and understanding of complex scenes and large-scale targets. Moreover, the objects contained in the input images may have scale variations, posing challenges for tasks such as target detection, tracking and segmentation. By performing multiscale fusion, the model can become robust to scale changes. Features at different scales can complementarily provide information about the target object, thus enhancing the adaptability of the model to scale changes.

3. YOLOv7 training process

3.1. *Datasets*

3.1.1. *Concrete crack dataset*

The image dataset used for segmentation in this paper consists of 457 images of concrete damage taken by high-resolution cameras. Since the complexity of concrete damage detection lies in the diversity of the observed weather conditions, light intensity diversity greatly affects the accuracy of the damage detection process. Therefore, to simulate crack pictures under different lighting and weather conditions, we expose, dim and overnoise the pictures with three modes: longitudinal, transverse and fatigue concrete cracking. The number of fracture types in the



(a) Comparison among the different indicators of the innovative weighted module

(b) Comparison among the different indicators of the innovative Multicat module

(c) Comparison among the different indicators of the innovative fusion module

Fig. 10. Recognition results obtained by adding different modules.

dataset is shown in Table 1. The dataset pictures are shown in Fig. 7.

Transverse road cracks are cracks that are perpendicular to the direction of road travel, and they are caused by factors such as pavement base settlement, pavement material fatigue, and vehicle loading and temperature changes. Although these cracks may have smaller impacts on vehicle movement, their long-term presence can lead to problems such as fractures and loosening in concrete pavements.

On the other hand, longitudinal cracks run parallel to the road travel direction and are mainly caused by roadbed settlement, reinforcement corrosion or improper pavement design. These types of cracks tend to cause bumps and vibrations during vehicle travel, thus accelerating pavement damage.

Fatigue cracks are formed due to the regular action of traffic loads on pavements, mainly on highways that are frequently used by heavy vehicles. Fatigue cracks can have a major impact on vehicle driving safety, so detecting them is an important aspect of road maintenance and safety. The crack images are manually annotated as binary images using Photoshop. A total of 1371 cropped images with resolutions of 512 × 512 are obtained after cropping. The labels of the dataset are shown in Table 2 and Fig. 8.

3.1.2. *Crack500 dataset*

Yang et al. obtained 500 images of pavement cracks with resolutions of approximately 2000 × 1500 pixels [33] using a mobile device at the main campus of Temple University. Each image is annotated at the pixel level. Due to the graphics processing unit (GPU) memory limitation of the utilized computer, each image is cut into 16 small images. With this preprocessing strategy, the training set, validation set, and test set contain 1896, 348 and 1124 images, respectively. The resolution of each cropped image is 640 × 360. In this dataset, the pavement material is asphalt, and the images contain shadows and inhomogeneous lighting conditions, which pose significant challenges for achieving accurate crack segmentation. A sample of the crack images can be seen in Fig. 9.

3.2. *Implementation details of the training process*

All experiments are performed on PyTorch with the CentOS7 operating system and implemented on a workstation with a Linux system using the GPU mode. Compared to central processing units (CPUs), GPUs can perform deep learning tasks faster and more efficiently due to their efficient parallel operations. The two employed GPUs are NVIDIA Ampere A100 units with 80 GB of memory.

In general, it is tedious to configure the optimal hyperparameters. For deep learning algorithms, the selected hyperparameters have an important impact on the training time, storage cost and quality of the trained model. Stochastic gradient descent (SGD) is combined with the momentum method and used as the optimizer to train the model via backpropagation. The new layers are randomly initialized with weights derived from a zero-mean Gaussian distribution possessing a standard deviation of 0.01. The basic learning rate of the training network is an important hyperparameter. In this study, the learning rate is set to 0.001 to maintain a balance between computational speed and accuracy. The batch size is set to 1 during training based on the GPU memory. The base size of the anchor is the basic parameter affecting the anchor size; it is set to 16. In addition, the weight decay and momentum parameters are set to 0.0005 and 0.937, respectively.

4. Experiments and results

4.1. *Evaluation indicators*

To accurately and fairly evaluate the performance of the proposed instance segmentation model, we use three evaluation metrics that are widely employed in other models, including precision, recall, and mean average precision (mAP). We treat the cracked pixels in each image as positive instances and the background pixels as negative instances. These three metrics are defined as follows:

$$Presion = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + TN}$$

$$mAP = \frac{1}{m}\sum_{n=1}^{m} AP_n$$

where TP indicates the number of true positives, FP indicates the number of false positives, and FN indicates the number of false negatives.

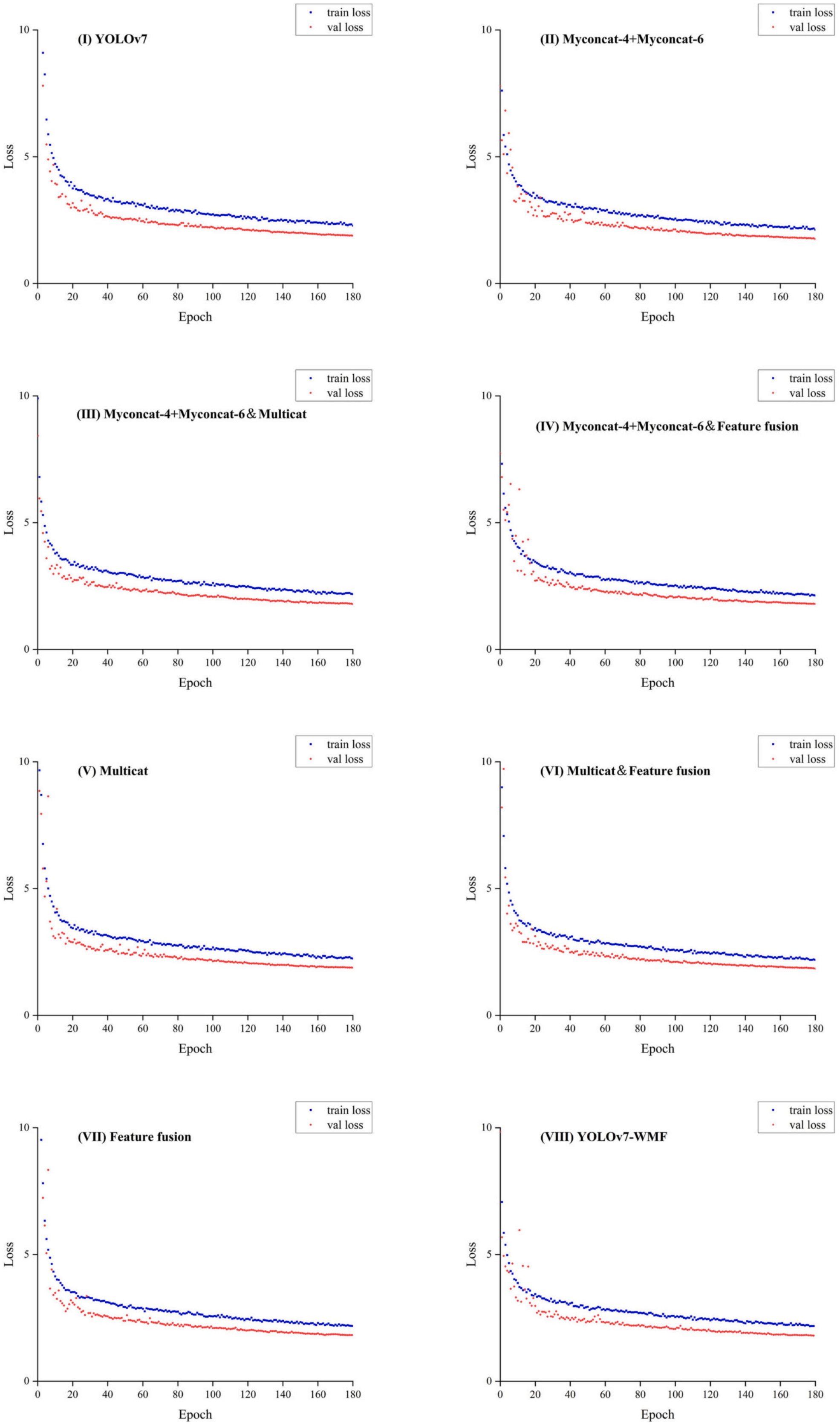


Fig. 11. The losses calculated for YOLOv7 under different modules.

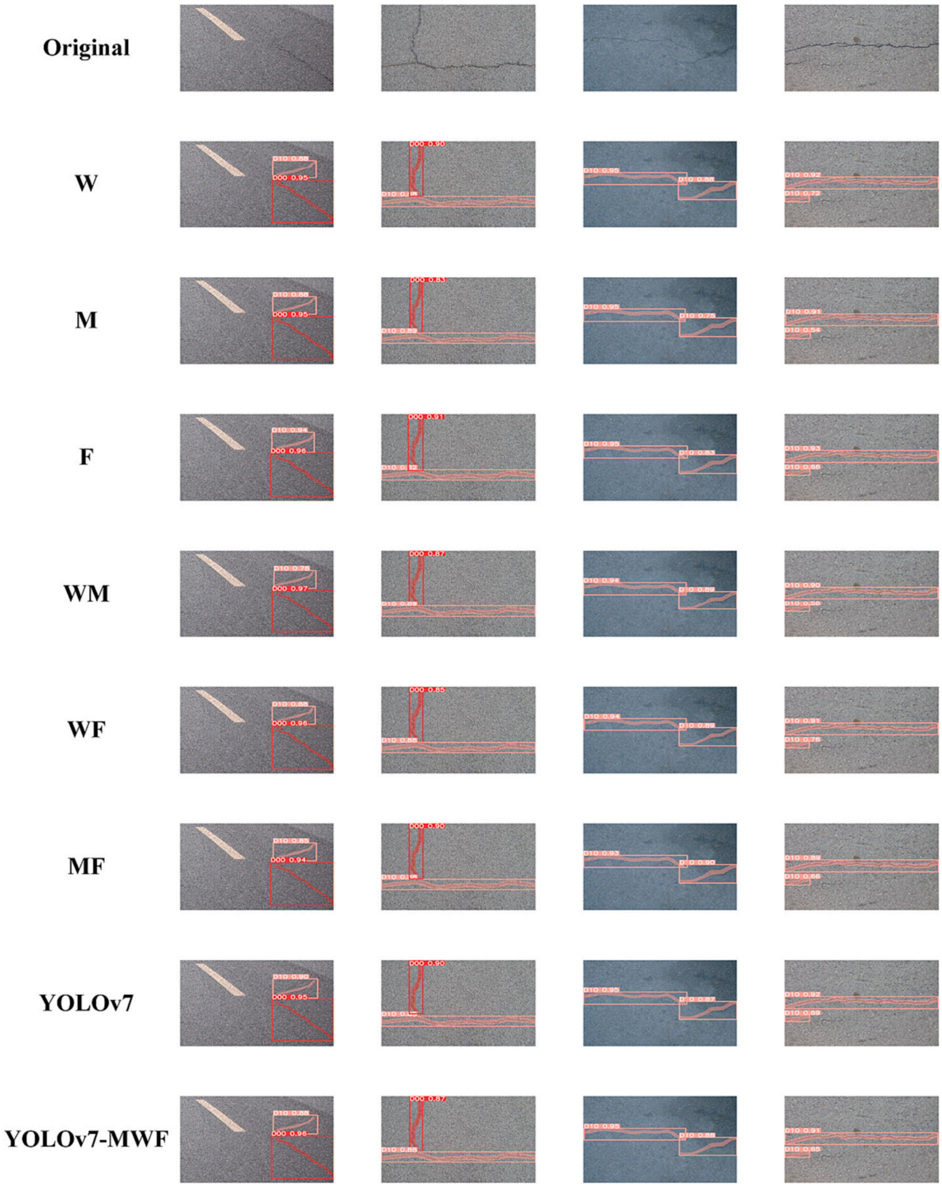


Fig. 12. Predicted crack visualization results produced by different modules.

Table 4
Comparison results produced by different network models.

The utilized network	Box			Seg		
	Precision	Recall	mAP$_{50}$	Precision	Recall	mAP$_{50}$
YOLOv7-WMF	0.908	0.902	0.955	0.872	0.864	0.879
YOLOv7	0.904	0.894	0.937	0.808	0.815	0.811
SOLOv2	/	/	/	/	/	0.697
Cascade Mask R-CNN	/	/	0.671	/	/	0.455
Condinst	/	/	0.776	/	/	0.661
Sparseinst	/	/	/	/	/	0.440
YOLOv5	0.857	0.772	0.824	0.809	0.652	0.687

4.2. Ablation experiments

Usually, the modification of a module results in an improvement in the performance of the associated artificial neural network. However, when only certain modules are added to work alone, good results are obtained. When different modules work together, this setting may degrade the resulting network performance. Therefore, to construct an improved network that is suitable for concrete crack identification based on YOLOv7, tests are performed on different optimization modules in this section.

The network structure of YOLOv7 returns to boxes enclosing the spine, neck and head, and the categories of the cracks in each enclosed box are identified. Therefore, three optimization scenarios are considered in our experimental analysis, and in each scenario, different experimental optimization cases are executed. We verify the performance of the Mycontact-4 and Mycontact-6 modules by adding residual connections to the overall network structure and replacing the original concatenation module with the custom Multicat module in the neck. In addition, this section evaluates and validates the different module combinations to further determine the specific impacts of the different modules on each other, as shown in Table 3.

The YOLOv7 recognition results obtained after adding different modules are shown in Fig. 10. First, the original YOLOv7 network is tested, and its mAP$_{50}$ values for target detection and segmentation reach 93.78% and 70.45%, respectively, when the number of epochs is set to 130 during the training process. Subsequently, YOLOv7 is added to the

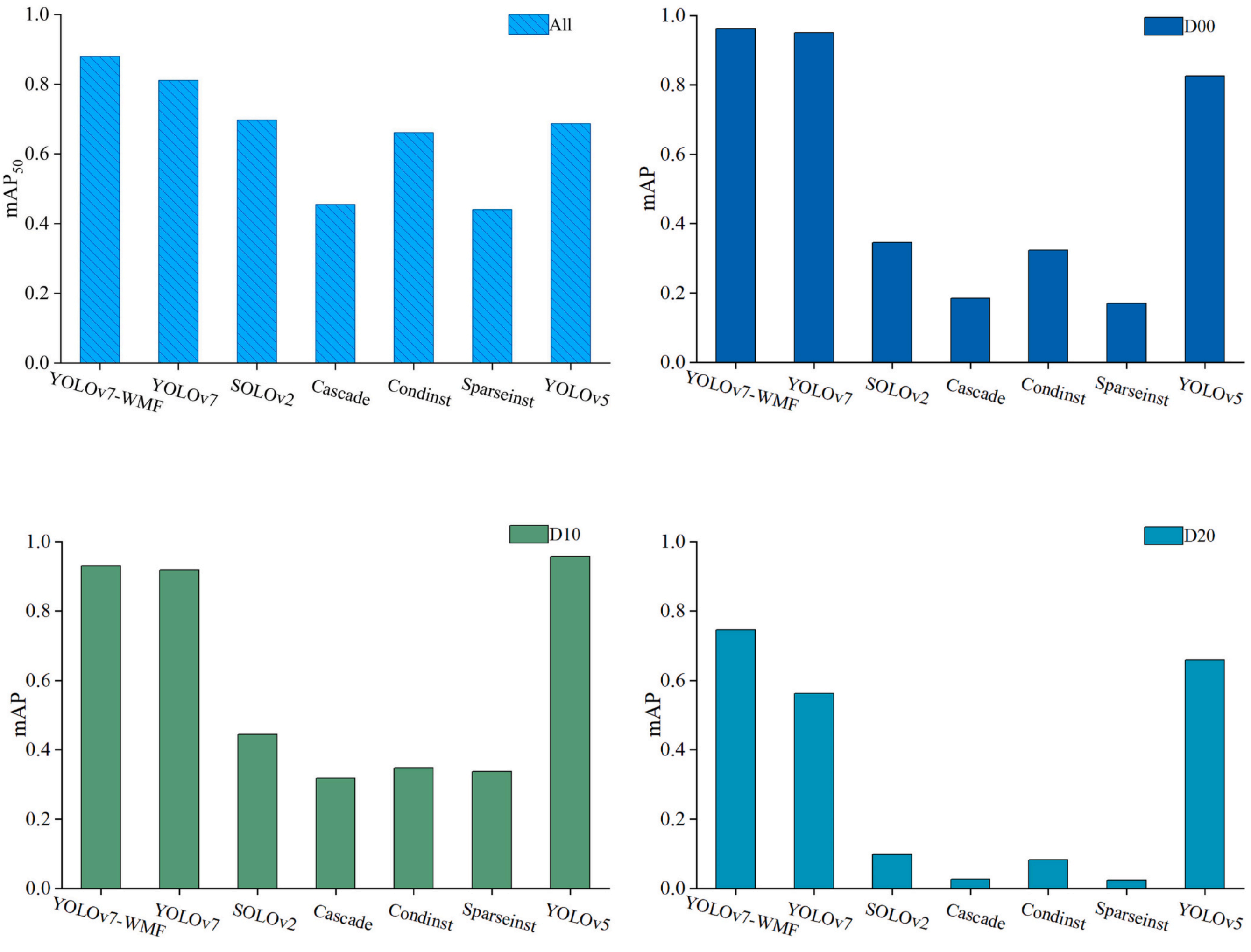


Fig. 13. mAP_{50} results obtained for comparison studies involving three different types of crack identification tasks.

Mycontact-4 module, Mycontact-6 module and residual connections for ablation experiments. The addition of the Mycontact-4 module and Mycontact-6 module provides some target detection performance improvements, with slight increases in both the mAP_{50} and recall metrics (reaching 93.47% and 89.80%, respectively). However, these modules have smaller impacts on the performance achieved in the segmentation task. The addition of the residual network provides significant target detection and segmentation performance improvements. The mAP_{50} and precision of the segmentation results improve to 76.82% and 83.65%, respectively. While the recall value decreases slightly to 82.09% for target detection, it increases slightly to 74.52% in the segmentation task. However, when the two modules are combined, although both metrics improve, with the mAP_{50} values reaching 93.67% and 80.89%, they are lower than those obtained when only the Mycontact-4 module or Mycontact-6 module is added alone.

The reason for this result is that when too much feature information is available, the model may have difficulty distinguishing which features are useful for the given task and which features consist of noisy or irrelevant information. This causes the model to be disturbed by useless features and reduces the model's attention to and recognition of key features. Therefore, in the neck, we construct a new Multicat module for multiscale fusion to improve the ability of the model to represent and provide information about the target object through the features observed at different scales in a complementary manner, thus enhancing the adaptability of the model to scale changes. After adding the Multicat module to YOLOv7, a certain target detection performance improvement is achieved, especially in terms of the mAP_{50} and precision metrics.

However, in the segmentation task, the boosts provided for the mAP_{50} and precision values are smaller, while the recall increases slightly. The mAP_{50} values reach 92.52% and 80.39% on the target detection and segmentation tasks, respectively. When the three improved parts are added together to the YOLOv7 network, the mAP_{50} peaks at 94.15% for the target detection task, while the highest mAP_{50} values for the other combined approaches are below 93.78%. Similarly, the highest mAP_{50} value of 83.09% is achieved in the segmentation task, while the highest mAP_{50} values for the other combined approaches are below 80.98%. Fig. 11 shows the performance achieved by the YOLOv7 network under the addition of different modules during the training process. The loss curves obtained on the training and validation sets during the training process of the network are given, and it can be seen that the training algorithm converges quickly and that a high mAP can be obtained. Therefore, it is experimentally verified that the information contained in feature maps can be effectively extracted by adding the improved Mycontact-4 module, Mycontact-6 module and residual connections, and the multiscale fusion process of the Multicat module is used to understand the before-and-after and global information contained in the feature maps. It is shown that simultaneously adding the three improved parts can enable the model to more accurately detect, locate, and segment the target object. The segmentation prediction for each module is shown in Fig. 12.

4.3. *Comparative experiments*

The improved detection method is compared with other detection

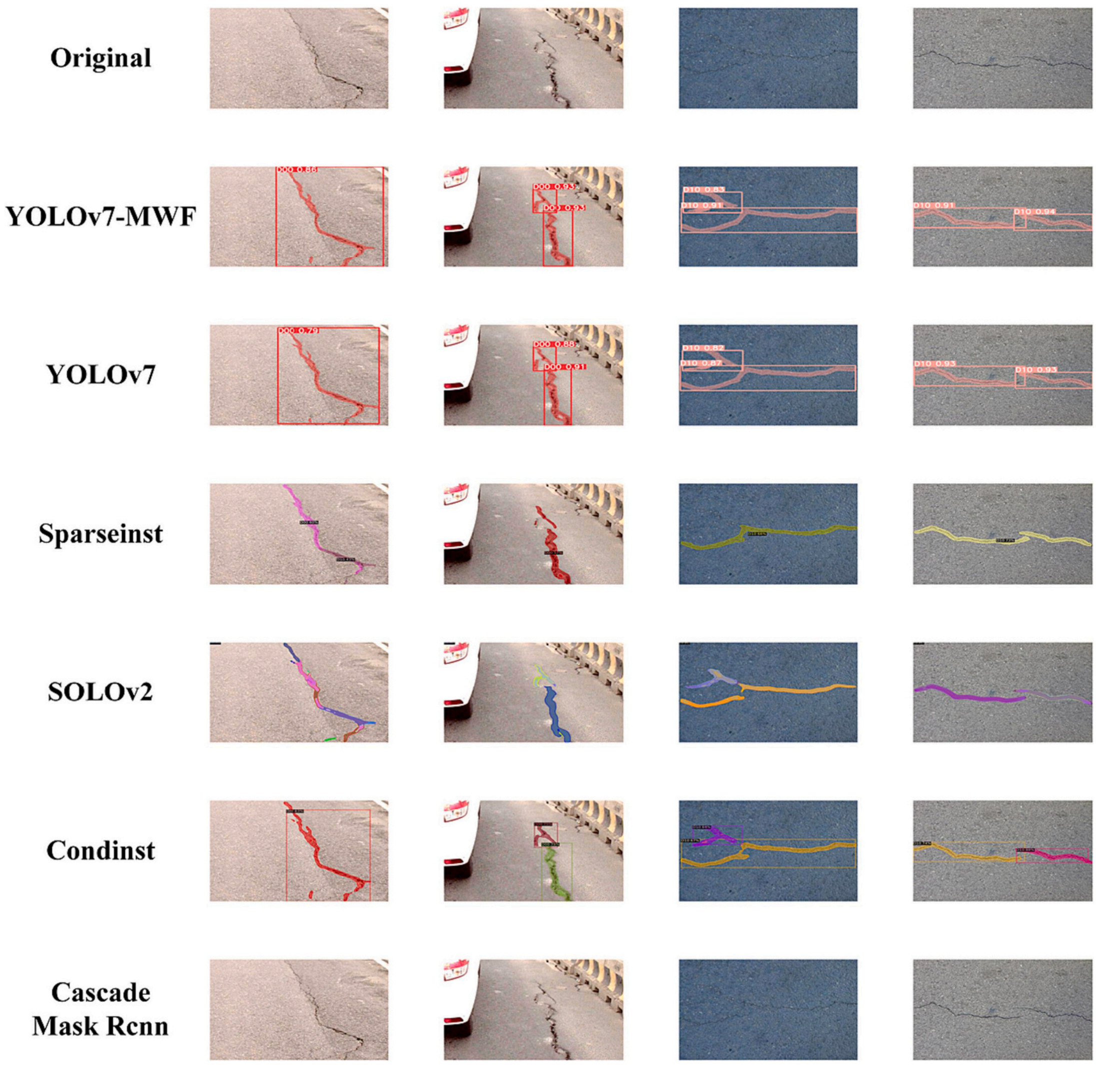


Fig. 14. Experimental results of different crack segmentation methods.

Table 5
Comparison among different network models on crack 500.

The utilized network	Box			Seg		
	Precision	Recall	mAP50	Precision	Recall	mAP_{50}
YOLOv7-WMF	0.861	0.812	0.809	0.812	0.735	0.722
YOLOv7	0.718	0.778	0.772	0.634	0.692	0.664
SOLOv2	/	/	/	/	/	0.541
Cascade Mask R-CNN	/	/	0.716	/	/	0.556
Condinst	/	/	0.637	/	/	0.538
Sparseinst	/	/	/	/	/	0.553
YOLOv5	0.807	0.775	0.784	0.724	0.685	0.657

methods, such as SOLOv2, Cascade Mask R-CNN, Condinst, Sparseinst, YOLOv5 and YOLOv7. To ensure a fair comparison, the operating environment and network parameters are kept consistent, and all methods are trained until they reach convergence to achieve optimal performance. Table 4 shows the detection results obtained by each method evaluated using the same test set. In terms of target detection, the model proposed in this study yields higher accuracy (mAP_{50}) than YOLOv7 and YOLOv5. The Mycontact-4 module and Mycontact-6 module in YOLOv7-WMF can help the model focus more on cracks, thus greatly improving its accuracy. The target detection mAP50 of YOLOv7-WMF is 13.1% higher than that of YOLOv5 and 1.8% higher than that of YOLOv7. For the surface segmentation task, the segmentation accuracy (mAP_{50}) of the proposed model is much higher than that of a series of classic example segmentation models, such as YOLOv7. A comparison among the instance segmentation metrics yielded by different networks is shown in Fig. 13.

Fig. 14 shows the four pavement crack images, their corresponding labels and the prediction results obtained using different models on the same dataset. The first row shows the original RGB images of the cracks, and from the second row to the sixth row, the predicted pavement crack images generated by our proposed YOLOv7-WMF approach and the other five networks are shown. Importantly, all test images are randomly selected to reflect the complex conditions of real coagulation cracks.

The results in Fig. 14 show that Cascade Mask R-CNN uses a predefined representation of the crack shape, such as a rectangle or a polygon. This representation may not be properly adapted to the irregular shape of the given crack and may be limited, especially when dealing with complex, curved or irregularly shaped cracks. Therefore, additional postprocessing steps or the use of other methods that are more suitable for irregular shapes may be needed when splitting cracks with this approach. YOLOv7-WMF uses weighted feature fusion to effectively solve this problem because this technique can synthesize multiple pieces of feature information, strengthen the key features, and complement each them with information. Two main reasons account for the segmentation accuracy differences between SOLOv2, Condinst, Sparseinst and the model proposed in this paper. (1) A few crack regions

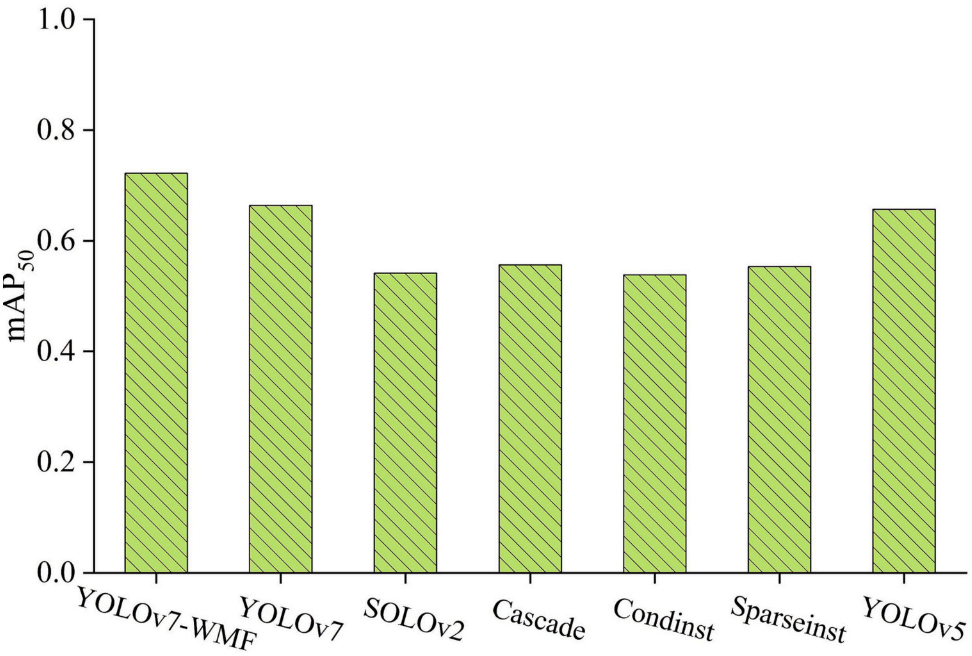


Fig. 15. mAP_{50} results obtained in crack identification comparison studies.

that cannot be wrapped by bounding boxes remain, resulting in an inability to identify these cracks. (2) Difficult-to-handle small and dense targets may be present. In the example crack splitting task, small target cracks and dense target cracks are often highly difficult to distinguish. The utilized methods may be affected by resolution and perceptual field size limitations when dealing with small targets, and they may suffer from performance degradation when addressing dense target cracks due to computational and storage limitations. This results in a segmentation mAP_{50} that is far inferior to that of the network proposed in this paper.

To demonstrate the generality of the proposed network for crack recognition, this paper also uses 500 images of pavement cracks with resolutions of approximately 2000 × 1500 pixels taken by Yang et al. for experiments [34]. The improved detection method is compared with the above networks, as shown in Table 5 and Fig. 15. The results show that the detection accuracy of YOLO-WMF is better than that of the other traditional instance segmentation methods (by 5.8%, 18.1%, 16.6%, 18.4%, and 16.9% higher). The improved model is more suitable for the detection of concrete cracks, and the provided improvement is effective. Based on the above comparison conducted on the same dataset, under the same test conditions, and with the use of both hardware devices, we can clearly see that the improved network presented in this study has better performance and is more suitable for concrete crack detection than the competing techniques.

Fig. 16 shows the prediction results obtained for four randomly selected images from the Crack500 dataset. The first row shows the original RGB images of the cracks, and from the second to the sixth row, the predicted pavement crack images generated by our proposed YOLOv7-WMF method and the other five networks are presented. Importantly, all test images are randomly selected to reflect the complex conditions of real coagulation cracks.

From the first and second columns, we can easily find that only our modified YOLOv7-WMF network and YOLOv7 can accurately predict pavement cross-cracks. The other networks, SOLOv2, Condinst, Sparseinst, and YOLOv5, can only roughly predict pavement cracks. The prediction results of Cascade Mask R-CNN only provide a small fraction of the fuzziness of the cross-slit.

The third column shows more challenging complex transverse cracks, and the fourth column shows wider transverse cracks. Compared with the other prediction results, those predicted by the YOLOv7-WMF network and YOLOv7 are able to most closely label the ground truth of the crack pattern. Regarding the finely cracked part in the middle of the image, only YOLOv7-WMF can represent the intermittent part of the cracks in comparison with the other models. The remaining models are unable to accurately predict the difficult and imperceptible fractures, indicating poor feature extraction capabilities at the global level. The fourth column contains wider transverse cracks. However, due to the significant differences between the pixel gradients of the pavement cracks and the background, all models can satisfactorily predict the crack skeletons, but YOLOv7-WMF can better predict the widths and boundaries of the cracks.

5. Discussion

In this paper, we propose using the instance segmentation method to identify cracks, which is very different from the previous approach of using semantic segmentation to identify cracks. For crack identification tasks, instance segmentation has more accurate localization and segmentation abilities than semantic segmentation. Thus, in engineering applications, instance segmentation allows each crack to be segmented as a separate instance, resulting in finer crack localization and segmentation processes. This is very important for crack detection and evaluation purposes.

Whereas semantic segmentation usually represents cracks with pixel-level markers, it does not provide fine localization and segmentation results for cracks. This can lead to the inability to accurately measure and evaluate important information such as the sizes, shapes, and locations of cracks in engineering applications. During the crack identification process, instance segmentation can also provide an individual identifier for each crack, thus enabling individual-level analysis and processing steps for each crack in subsequent analyses. In contrast, semantic segmentation cannot accurately distinguish overlapping crack instances; it can only label the entire overlapping region as a crack class and cannot separate each crack instance. This can result in the inability to analyze overlapping cracks at the individual level in subsequent analysis and processing tasks.

The following engineering applications are available. 1. Targeting and counting: Instance segmentation can accurately locate and split each crack instance, providing a bounding box and identifier for each

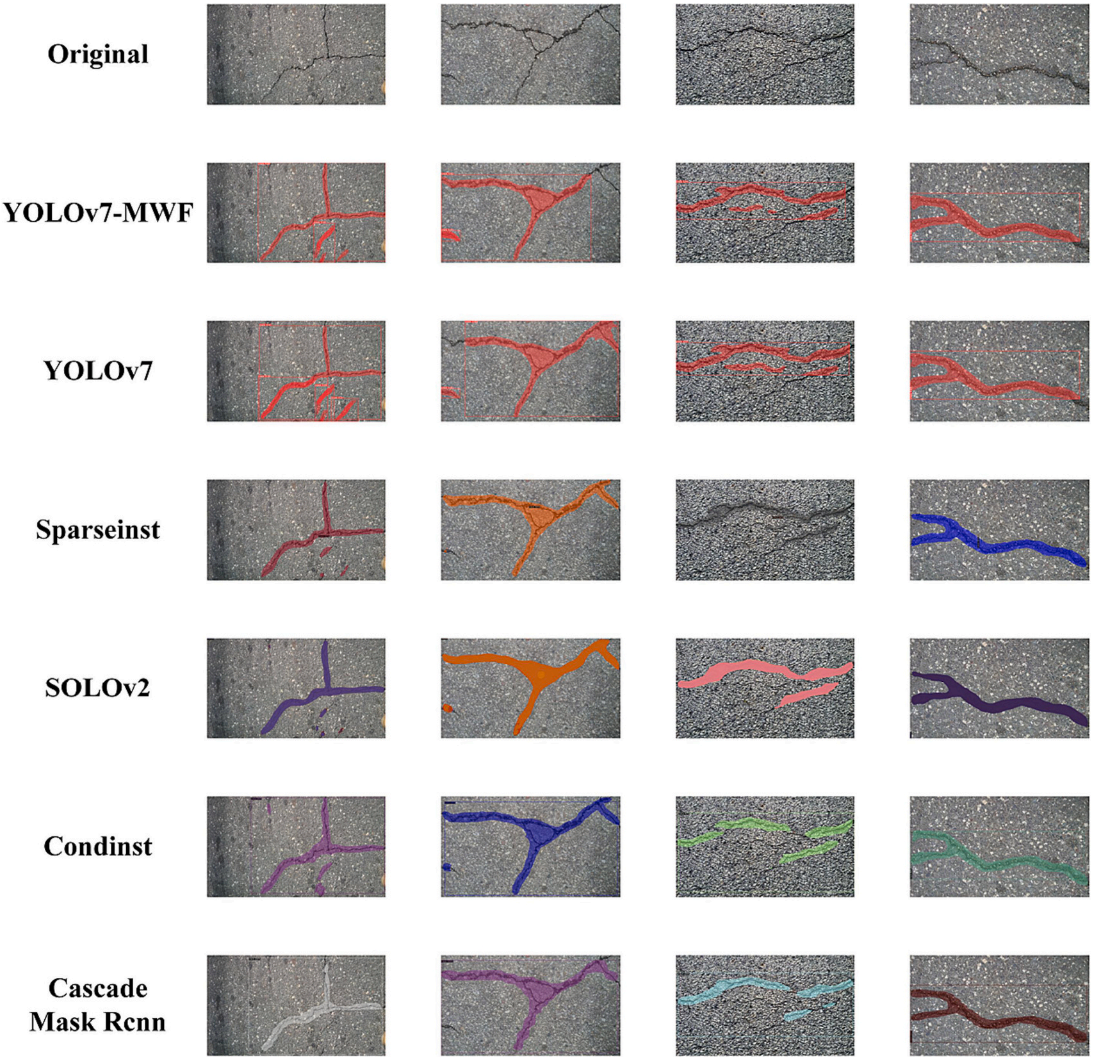


Fig. 16. Experimental results obtained by different segmentation methods on fracture 500.

crack. This enables the precise location and counting of cracks in engineering applications and facilitates quantitative crack analyses and assessments. 2. Fine feature extraction: Instance segmentation focuses on the individual features of each crack instance, so it can extract finer features such as crack shapes, textures and sizes. These characteristics are important for engineering tasks such as class type identification and classification, crack severity assessment and defect detection during construction. 3. Individual analysis and statistics: Instance segmentation allows for independent analyses and statistics to be produced for each crack instance. This helps us understand the distributions of cracks, the interrelationships between cracks and the trends of cracks. This information is important for making engineering decisions, developing maintenance plans and implementing quality control. 4. Defect detection and analysis: Split crack examples can be used to identify smaller, hidden cracks and analyze them in fine detail. This is very helpful for performing defect detection and quality control in engineering construction cases, as cracks can be detected and repaired in time to reduce safety hazards and engineering quality problems.

6. Conclusions

In this study, a high-precision pavement crack instance segmentation model based on YOLOv7 is proposed. The splitting model achieves the dual tasks of target crack detection and crack splitting. We propose two modules by splitting feature map data and assigning different weights to different data. The Mycontact-4 module is applied to the backbone mainly to help the network remove irrelevant crack recognition details. The Mycontact-6 module applied in the head is mainly employed to solve the limitations of the network recognition environment, and then we include special residual connections in the underlying model architecture. To make the feature map information of the backbone available to the head, segmentation can account for the feature map of the original image so that the subtle pre-existing differences between the input and the target can be captured. In addition, we introduce Multicat, which performs feature extraction and integration operations on three different subimages of the input. The message passing operations (residual connections) in the module, which connect the extracted features together, allow the network to extract crack information from the different subimages, especially the features of the cracks. Through operations such as interpolation, scale adaptation is ensured for the different subimages.

The proposed method trains on our dataset and obtains an instance segmentation model for multiscene concrete crack images. The results show that the precision, recall, and mAP_{50} reach 87.2%, 86.4%, and 87.9%, respectively, when the proposed model is verified on the testing set. The developed concrete crack detection approach using the modified YOLOv7-WMF network is an effective and practically meaningful method.

However, this study requires further research and improvement in practical applications to suit the needs of different regions and environments. (1) The improved YOLOv7 network developed in this study has room for improvement in terms of reducing the number of network parameters and its complexity level to ensure that the network can operate more efficiently in resource-constrained environments. (2) The improved network is slightly slower in terms of operating speed than the legacy YOLOv7 approach. This suggests that networks are becoming lighter. However, some performance may be sacrificed in some cases. This may require further optimization. (3) Water damage, tree branches, and other crack-like conditions can be added to future research to enrich the content of the formed dataset. This will help the network better respond to a variety of practical situations. (4) A finer and more specific delineation of crack types can be used to categorize the types of cracks that are more harmful to the target structure. This can improve the usefulness and effectiveness of the network, making it become more specific about the levels of harm produced by different cracks; this would increase its practical value in engineering applications.

CRediT authorship contribution statement

Guanting Ye: Writing – original draft, Visualization, Methodology, Conceptualization. **Sai Li:** Writing – original draft, Software, Data curation, Conceptualization. **Manxu Zhou:** Resources, Investigation. **Yifei Mao:** Formal analysis. **Jinsheng Qu:** Validation. **Tieyu Shi:** Funding acquisition. **Qiang Jin:** Writing – review & editing, Methodology, Funding acquisition.

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

Data will be made available on request.

Acknowledgments

The authors would like to acknowledge the financial support provided by the National Natural Fund (grant no. 52269028) and the 2022 Xinjiang Autonomous Region-level Grand Innovation Project for University Students (grant no. S2022107580107).

References

[1] L. Yu, S. He, X. Liu, S. Jiang, S. Xiang, Intelligent crack detection and quantification in the concrete bridge: a deep learning-assisted image processing approach, Adv. Civil Eng. 2022 (2022) 1–15, https://doi.org/10.1155/2022/1813821.

[2] S. Meng, Z. Gao, Y. Zhou, B. He, A. Djerrad, Real-time automatic crack detection method based on drone, Comput. Aided Civil Infrastruct. Eng. 38 (2023) 849–872, https://doi.org/10.1111/mice.12918.

[3] E. Zhang, L. Shao, Y. Wang, Unifying transformer and convolution for dam crack detection, Autom. Constr. 147 (104712) (2023) 1–14, https://doi.org/10.1016/j.autcon.2022.104712.

[4] I. Abdel-Qader, O. Abudayyeh, M.E. Kelly, Analysis of edge-detection techniques for crack identification in bridges, J. Comput. Civ. Eng. 17 (2003) 255–263, https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(255).

[5] A. Ji, X. Xue, Y. Wang, X. Luo, W. Xue, An integrated approach to automatic pixel-level crack detection and quantification of asphalt pavement, Autom. Constr. 114 (103176) (2020) 1–15, https://doi.org/10.1016/j.autcon.2020.103176.

[6] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM 60 (2017) 84–90, https://doi.org/10.1145/3065386.

[7] R. Girshick, Fast R-CNN, in: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE, Santiago, Chile, 2015, pp. 1440–1448, https://doi.org/10.1109/ICCV.2015.169.

[8] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Boston, MA, USA, 2015, pp. 3431–3440, https://doi.org/10.1109/CVPR.2015.7298965.

[9] Y. Cha, W. Choi, O. Büyüköztürk, Deep learning-based crack damage detection using convolutional neural networks, Comput. Aided Civil Infrastruct. Eng. 32 (2017) 361–378, https://doi.org/10.1111/mice.12263.

[10] K. Chen, A. Yadav, A. Khan, Y. Meng, K. Zhu, Improved crack detection and recognition based on convolutional neural network, Model. Simul. Eng. 2019 (2019) 1–8, https://doi.org/10.1155/2019/8796743.

[11] W. Cao, Q. Liu, Z. He, Review of pavement defect detection methods, IEEE Access 8 (2020) 14531–14544, https://doi.org/10.1109/ACCESS.2020.2966881.

[12] H. Nhat-Duc, Q.-L. Nguyen, V.-D. Tran, Automatic recognition of asphalt pavement cracks using metaheuristic optimized edge detection algorithms and convolution neural network, Autom. Constr. 94 (2018) 203–213, https://doi.org/10.1016/j.autcon.2018.07.008.

[13] Y. Cha, W. Choi, G. Suh, S. Mahmoudkhani, O. Büyüköztürk, Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types, Comput. Aided Civil Eng. 33 (2018) 731–747, https://doi.org/10.1111/mice.12334.

[14] Z. Yu, Y. Shen, C. Shen, A real-time detection approach for bridge cracks based on YOLOv4-FPM, Autom. Constr. 122 (103514) (2021) 1–14, https://doi.org/10.1016/j.autcon.2020.103514.

[15] M. Mohtasham Khani, S. Vahidnia, L. Ghasemzadeh, Y.E. Ozturk, M. Yuvalaklioglu, S. Akin, N.K. Ure, Deep-learning-based crack detection with applications for the structural health monitoring of gas turbines, Struct. Health Monit. 19 (2020) 1440–1452, https://doi.org/10.1177/1475921719883202.

[16] J. Chen, Y. He, A novel U-shaped encoder–decoder network with attention mechanism for detection and evaluation of road cracks at pixel level, Comput. Aided Civil Eng. 37 (2022) 1721–1736, https://doi.org/10.1111/mice.12826.

[17] Z. Qu, C. Cao, L. Liu, D.-Y. Zhou, A deeply supervised convolutional neural network for pavement crack detection with multiscale feature fusion, IEEE Trans. Neural Netw. Learn. Syst. 33 (2022) 4890–4899, https://doi.org/10.1109/TNNLS.2021.3062070.

[18] J. Zhang, C. Lu, J. Wang, L. Wang, X.-G. Yue, Concrete cracks detection based on FCN with dilated convolution, Appl. Sci. 9 (2019) 2686, https://doi.org/10.3390/app9132686.

[19] S. Bang, S. Park, H. Kim, H. Kim, Encoder–decoder network for pixel-level road crack detection in black-box images, Comput. Aided Civil Infrastruct. Eng. 34 (2019) 713–727, https://doi.org/10.1111/mice.12440.

[20] Z. Fan, C. Li, Y. Chen, P.D. Mascio, X. Chen, G. Zhu, G. Loprencipe, Ensemble of deep convolutional neural networks for automatic pavement crack detection and measurement, Coatings. 10 (152) (2020) 1–14, https://doi.org/10.3390/coatings10020152.

[21] J. Zhang, J. Zhang, An improved nondestructive semantic segmentation method for concrete dam surface crack images with high resolution, Math. Probl. Eng. 2020 (2020) 1–14, https://doi.org/10.1155/2020/5054740.

[22] Y. Xu, Y. Bao, J. Chen, W. Zuo, H. Li, Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images, Struct. Health Monit. 18 (2019) 653–674, https://doi.org/10.1177/1475921718764873.

[23] S. Teng, Z. Liu, G. Chen, L. Cheng, Concrete crack detection based on well-known feature extractor model and the YOLO_v2 network, Appl. Sci. 11 (813) (2021) 1–13, https://doi.org/10.3390/app11020813.

[24] M. Nie, C. Wang, Pavement crack detection based on yolo v3, in: 2019 2nd International Conference on Safety Produce Informatization (IICSPI), IEEE, Chongqing, China, 2019, pp. 327–330, https://doi.org/10.1109/IICSPI48186.2019.9095956.

[25] N.S.P. Peraka, K.P. Biligiri, S.N. Kalidindi, Development of a multi-distress detection system for asphalt pavements: transfer learning-based approach, Transp. Res. Rec. 2675 (2021) 538–553, https://doi.org/10.1177/03611981211012001.

[26] Z. Qu, L. Gao, S. Wang, H. Yin, T. Yi, An improved YOLOv5 method for large objects detection with multi-scale feature cross-layer fusion network, Image Vis. Comput. 125 (104518) (2022) 1–11, https://doi.org/10.1016/j.imavis.2022.104518.

[27] G. Ye, J. Qu, J. Tao, W. Dai, Y. Mao, Q. Jin, Autonomous surface crack identification of concrete structures based on the YOLOv7 algorithm, J. Build. Eng. 73 (106688) (2023) 1–15, https://doi.org/10.1016/j.jobe.2023.106688.

[28] D. Ma, H. Fang, N. Wang, C. Zhang, J. Dong, H. Hu, Automatic detection and counting system for pavement cracks based on PCGAN and YOLO-MF, IEEE Trans. Intell. Transp. Syst. 23 (2022) 22166–22178, https://doi.org/10.1109/TITS.2022.3161960.