Mathematical Biosciences and Engineering

MBE, 21(3): 4269–4285.
DOI: 10.3934/mbe.2024188
Received: 23 November 2023
Revised: 17 January 2024
Accepted: 29 January 2024
Published: 26 February 2024

http://www.aimspress.com/journal/MBE

Research article

Road surface crack detection based on improved YOLOv5s

Jiaming Ding, Peigang Jiao*, Kangning Li and Weibo Du

Shandong Jiaotong University, Jinan 250357, China

*** Correspondence:** Email: jiaopeigang@163.com; Tel: +8613969015926.

Abstract: In response to the issues of low efficiency and high cost in traditional manual methods for road surface crack detection, an improved YOLOv5s (you only look once version 5 small) algorithm was proposed. Based on this improvement, a road surface crack object recognition model was established using YOLOv5s. First, based on the Res2Net (a new multi-scale backbone architecture) network, an improved multi-scale Res2-C3 (a new multi-scale backbone architecture of C3) module was suggested to enhance feature extraction performance. Second, the feature fusion network and backbone of YOLOv5 were merged with the GAM (global attention mechanism) attention mechanism, reducing information dispersion and enhancing the interaction of global dimensions features. We incorporated dynamic snake convolution into the feature fusion network section to enhance the model's ability to handle irregular shapes and deformation problems. Experimental results showed that the final revision of the model dramatically increased both the detection speed and the accuracy of road surface identification. The mean average precision (mAP) reached 93.9%, with an average precision improvement of 12.6% compared to the YOLOv5s model. The frames per second (FPS) value was 49.97. The difficulties of low accuracy and slow speed in road surface fracture identification were effectively addressed by the modified model, demonstrating that the enhanced model achieved relatively high accuracy while maintaining inference speed.

Keywords: road surface crack detection; deep learning; YOLOv5s; Res2-C3 module; attention mechanism

1. Introduction

In recent years, China has maintained a relatively stable development trend in its highway transportation, with significant improvements in its level of development. As of the end of 2022, the total length of highways in China has approached 5.2 million kilometers, and the mileage of the expressway network has reached 161,000 kilometers [1]. Given the swift advancement of highway construction in China, the country has now entered the phase of highway maintenance [2]. Therefore, it is crucial to timely detect and repair issues on the road surface. Road surface cracks are the most common type of pavement distress. If they can be detected promptly in their early stages and repaired, it not only effectively prevents them from evolving into more severe pavement issues but also extends the lifespan of expressways. Hence, a fast, convenient, and safe road surface crack detection method holds significant importance for road maintenance.

There are many problems in the traditional road crack detection algorithm. For instance, the road mileage is often too extensive, leading to high human resource costs. Human detection involves complex human factors and is not conducive to objectively evaluating road defect detection accuracy, which cannot be guaranteed. Additionally, the traditional manual road detection method is adverse to inspector safety. In recent years, with the rapid development of various new technologies, including computers, target detection, GPS (global position system), digital CCD (charge-coupled device), etc. [3–5], computer vision based on deep learning has gained wide acceptance and application in our daily lives. These issues can be addressed by adopting deep learning object detection algorithms. The YOLO (you only look once) series of algorithms is a neural network algorithm used for real-time object detection. Unlike traditional two-stage object detection methods, the YOLO series is a one-stage detector [6]. It directly predicts the position and category of targets through a single feed forward neural network without the need for candidate box generation and filtering steps, resulting in higher detection accuracy and faster inference speed. Currently, there is a substantial amount of research conducted both domestically and internationally on road surface crack detection and recognition using YOLO algorithms.

Researchers such as Park [7] have established a network model that combines segmentation and detection. During the segmentation process, only a part of the road surface is extracted, and road surface damage is detected based on that portion, which improves accuracy but reduces detection efficiency. X. Su [8] used MobileNetv2 as the backbone network for YOLOv4 and replaced conventional convolutions with depth-wise separable convolutions. Furthermore, the backbone and neck components of the original model also embedded coordinate attention processes and spatial attention mechanisms. These attention mechanisms significantly enhance the detection accuracy and speed for road surface cracks, but the model's final mean average precision (mAP) value is relatively low. M. Wang [9] proposed a method that replaces the GIoU (generalized intersection over union) loss function with EIoU (exponential intersection over union), resolving the issue of large GIoU errors while improving convergence speed and regression accuracy. However, the improved model's inference speed has decreased compared to the original model. A comprehensive analysis of the development from YOLOv1 to YOLOv8 is presented by J. R. Terven et al. [10]. The authors conclude that starting from YOLOv5 and moving onward, all official YOLO models have been fine-tuned to strike a balance between speed and accuracy, aiming to better adapt to specific applications and hardware requirements [11]. Classic YOLOv5 employs a simple convolutional neural network (CNN) architecture, while the latest YOLOv8 employs a more complex network structure comprising multiple residual units and branches [12]. Consequently, YOLOv5’s detection accuracy is not on par with that of YOLOv8 when processing road crack images. Although YOLOv8 enhances the model’s structure and training effectiveness, it sacrifices detection speed and has a larger parameter count. Therefore, YOLOv5 has been selected for improvement.

These aforementioned detection models each have their advantages and disadvantages, and cannot achieve a balance between detection accuracy and speed, limiting their practical application in engineering [8]. To achieve efficient and accurate road surface crack recognition, this paper introduces a road surface crack detection model that combines attention mechanisms. The main contributions of this paper can be summarized as follows:

1) Based on the Res2Net (a new multi-scale backbone architecture) network, an improved multi-scale Res2-C3 (a new multi-scale backbone architecture of C3) module is suggested to enhance the feature extraction performance.
2) The feature fusion network and backbone of YOLOv5 are combined with the GAM (global attention mechanism) attention mechanism to enhance the model’s ability to perceive fracture information.
3) Integrating dynamic serpentine convolution into the feature fusion network, the improved network enhances the model’s ability to address irregular shape and deformation problems, which is beneficial for improving the accuracy of road crack identification.

2. Improvement of the YOLOv5s model

2.1. Model overall structure

YOLOv5s is a variant of the YOLOv5 series, which is an object detection deep learning algorithm. Compared to other versions of YOLOv5, YOLOv5s is a lightweight model that preserves strong detection performance while decreasing the model's size and computational complexity. The YOLOv5s algorithm network structure consists of the head, neck, and backbone. CSPDarknet53 (cross stage paritial network) serves as the backbone network for YOLOv5s and it effectively extracts image features. Using feature maps with multi-scale information created by combining feature maps from various levels, the neck network integrates these feature maps with the features produced by the backbone network to increase object detection accuracy [13]. The head network is responsible for the final detection steps, constructing a neural network that determines the bounding box positions and recognition types, forming the ultimate output vector 2. Figure 1 represents the improved YOLOv5s network structure.

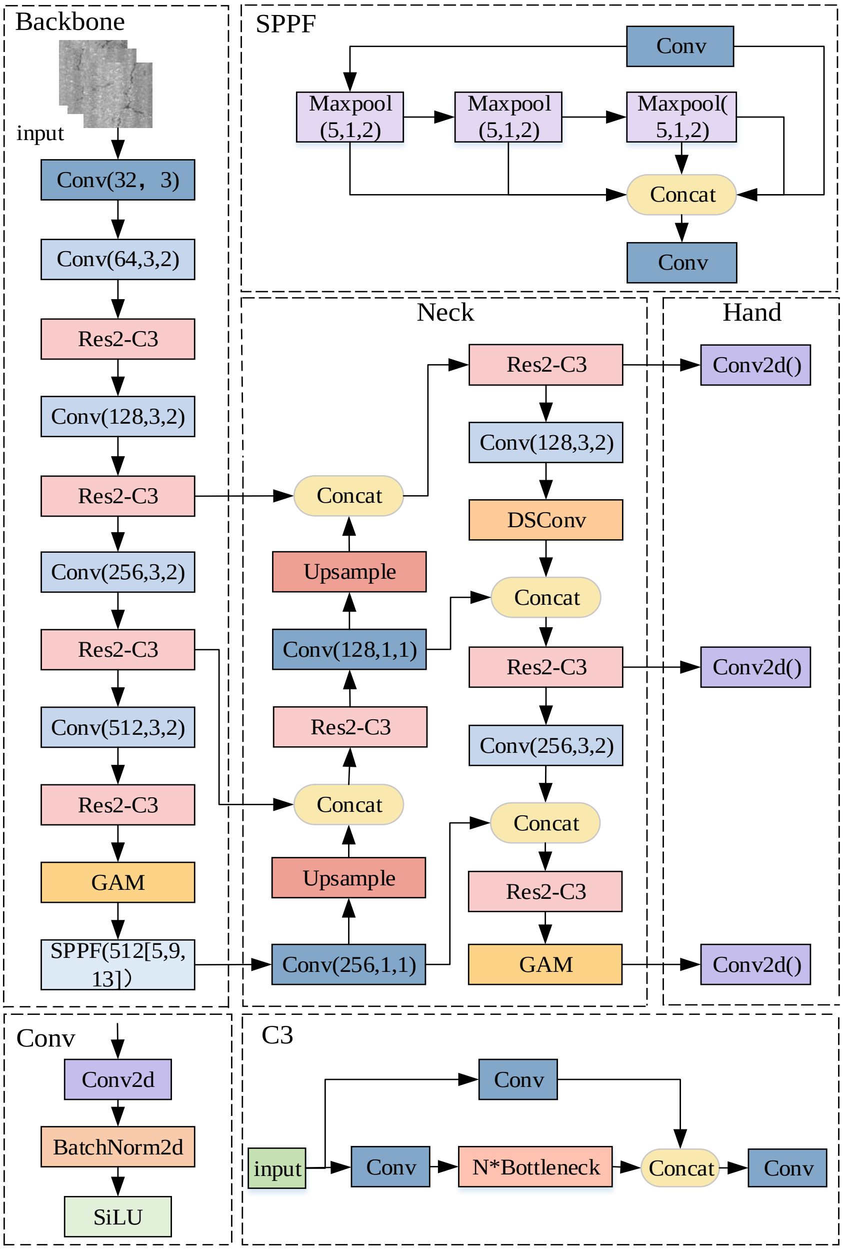

Figure 1. Improved YOLOv5s network structure.

2.2. Enhancing the multi-scale C3 structure

2.2.1. C3 module

As seen in Figure 2, the C3 module, which consists of two parallel branches, is the central component of the backbone network. After going through one of the branches' Conv modules, the input feature map is stacked with n bottleneck modules to extract high-level semantic information [14]. The output of the other branch, after passing through a Conv layer, is concatenated with the output of the first branch. Subsequently, feature fusion is performed through another Conv layer before the final output.

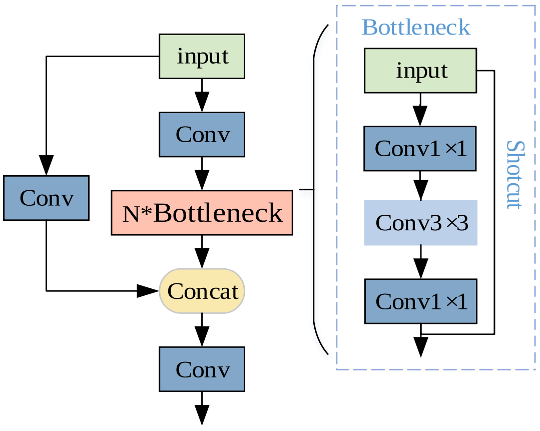

Figure 2. C3 module.

The C3 module of YOLOv5s mainly leverages the idea of extracting diversion using the CSPNet [15] and combines the concept of residual structures. It is designed with C3Block, and the bottleNeck module is the CSPNet main branch gradient module. The number of stacked modules is influenced by the parameter ‘n’, and the value of ‘n’ changes with the model’s size. The bottleneck module is integrated multiple times within the C3 module, to capture higher-level semantic information from the image.

2.2.2. Res2-C3 module

Drawing inspiration from the design concepts of VGG (visual geometry group) [16], GoogleNet [17], and CSPNet [15], the C3 module processes input information through two branches. One branch stacks n bottleneck structures to extract high-level semantic features, while the other branch maintains the original image features through a shortcut connection of the Conv module. Finally, two parallel convolution branches are used to merge the image features, enhancing the feature information within the image. Therefore, the core of feature extraction lies in the design of the bottleneck structure.

In order to improve the model’s capacity to extract feature information [18] and acquire richer feature information, deeper network architectures or a higher number of convolutional kernels are frequently used. A Res2-Bottleneck module is proposed by merging the bottleneck structure with the Res2Net module in order to improve the model's feature information extraction capability [19]. The primary idea is to decompose the feature maps from the 3 × 3 convolution layer in the original residual convolution, which receives feature maps from the 1 × 1 convolution layer of the input, into four parts [20]. The first part remains unchanged, the second part passes through a 3 × 3 convolution layer, the third part adds its features to those of the second part before passing through another 3 × 3 convolution layer, and the fourth part adds its features to those of the third part before passing through another 3 × 3 convolution layer. The parallel technique improves the model's capacity to extract characteristics across several scales. In the end, the feature maps from these four parts are concatenated to form feature maps with the same number of layers as the input layer, and are then sent to the output layer for 1 × 1 convolution to perform feature fusion. This structure is referred to as the enhanced multi-scale bottleneck (Res2-Bottleneck), as shown in Figure 3. The

Res2-Bottleneck not only increases the multi-scale feature information, but also maximally preserves the original feature information through the residual structure, reducing the loss of shallow features. As a result, more information about pavement cracks may be stored, which is highly advantageous for raising the model's detection accuracy.

In the original C3 module, one branch performs feature extraction by stacking multiple bottleneck modules. To improve the feature extraction performance, the original bottleneck structure is replaced with the improved Res2-Bottleneck structure, resulting in the improved Res2-C3 module, as shown in Figure 4.

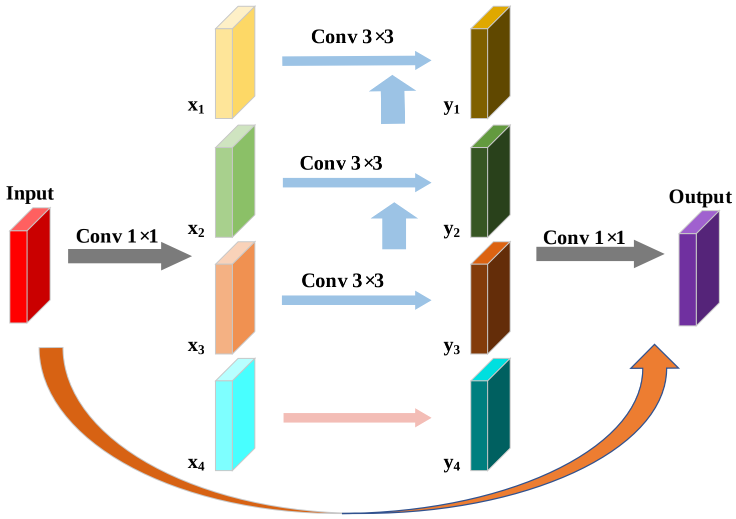

Figure 3. Enhance the multi-scale bottleneck structure.

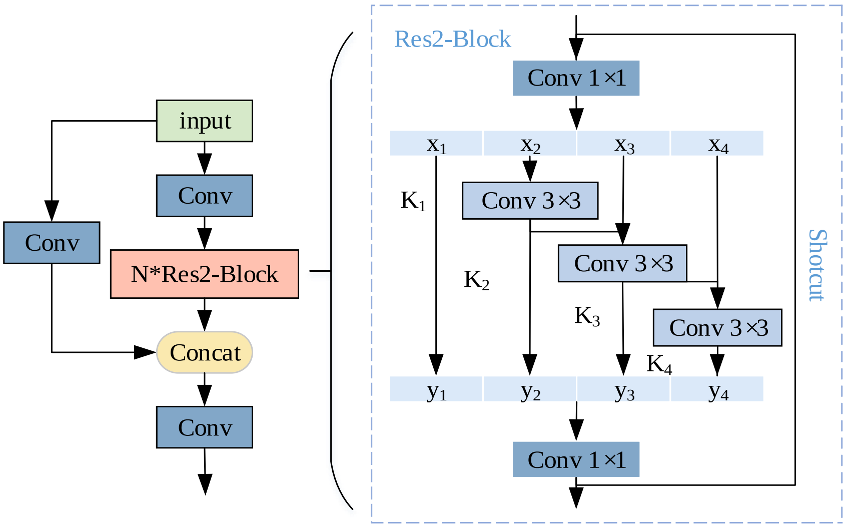

Figure 4. Res2-C3 module.

2.2.3. Multi-scale extraction module C3-RFEM

The C3 module is the heart of the YOLOv5 backbone network, and the bottleneck structure's design is essential to the C3 module. Building upon the bottleneck structure, the multi-scale extraction module C3-RFEM (receptive field enhancement module) is proposed based on the RFE (receptive field enhancement) module [21]. The main principle of the RFE module is to use four different scale expansion convolution branches to capture multi-scale information and different receptive ranges. These branches share weights, with the only difference being their receptive fields. This approach reduces model parameters and potential overfitting risks. Additionally, it allows for operations of different sizes, making full use of each feature's information. The RFE module can be divided into two parts: one is the multi-branch based on expansion convolution, and the other is the weighted layer, as shown in Figure 5. The multi-branch part uses different expansion convolutions with rates 1, 2, and 3; all these convolutions, however, employ a fixed 3 × 3 convolution kernel size. Residual connections are employed to prevent gradient explosion and vanishing during training. This structure can improve the model's detection accuracy and lessen feature loss during feature extraction.

Replacing the original bottleneck module with the RFE module results in C3RFEM, as illustrated in Figure 6. To ensure that the improved model exhibits better performance, comparative experiments are conducted between the C3RFEM module, which extracts multi-scale feature information based on the RFE module and the Res2-C3 module. This comparison aims to select a model with higher accuracy and faster speed.

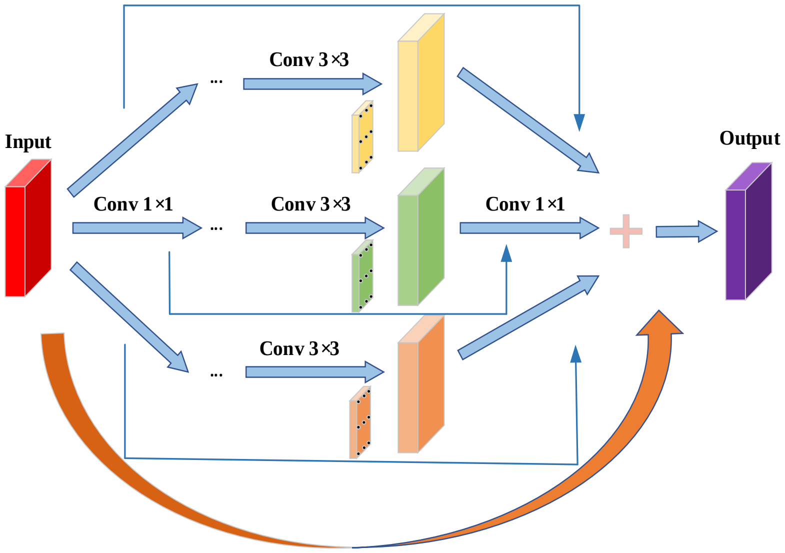

Figure 5. RFE module.

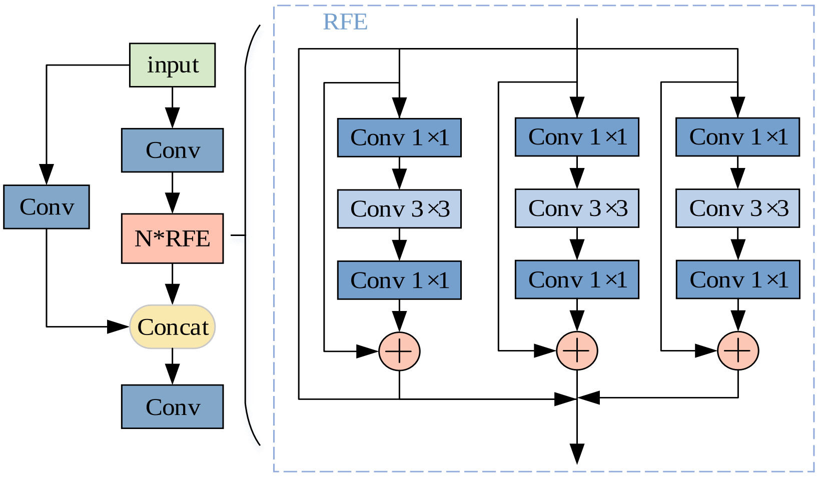

Figure 6. C3RFE module

2.3. Adaptive mechanism

The attention mechanism enables the model to selectively concentrate more on target information [22]. Separating the relevance of several channels and using it as a focus is the fundamental component of channel attention, thus weakening the role of uninterested channels. Hybrid attention combines channel attention with spatial attention, with these two parts being consecutive or parallel, forming an attention model for channel features and spatial features.

The SE (squeeze and excitation) [23] attention module is a channel attention module that enhances channel features in input feature maps. However, the SE attention mechanism neglects spatial information, failing to comprehensively extract the feature map information. CBAM (convolutional block attention module) [24] is a spatial attention mechanism that effectively overcomes the shortcomings of SE by utilizing channel information while considering spatial information. However, the CBAM attention mechanism loses cross-dimensional information by ignoring the relationship between channels and space. Recognizing the significance of interactions across dimensions, the GAM [25] attention mechanism is employed, which can lessen information dispersion and enhance global dimensions interaction features.

Though their approaches to channel attention and spatial attention are different, overall, the GAM and CBAM attention mechanisms are comparable. Figure 7 shows the full process, with Mc and Ms standing for channel attention maps and spatial attention maps, respectively.

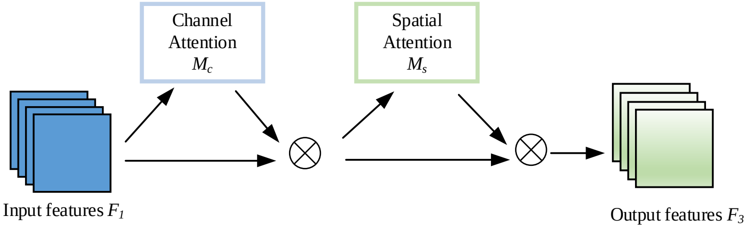

Figure 7. The overview of GAM.

Three-dimensional arrangements are used by the channel attention sub-module (CAM) to store information in three dimensions. Subsequently, a two-layer multilayer perceptron (MLP) is utilized to enhance the cross-dimensional interdependence across spatial channels. The channel attention sub-module is shown in Figure 8, where the input feature map undergoes dimension transformation, and the transformed feature map is processed through the MLP to restore its original dimensions, resulting in a Sigmoid output.

Two convolutional layers are employed for spatial information fusion in the spatial attention sub-module (SAM) in order to pay attention to spatial input [26]. For SAM, similar to the SE attention mechanism, it first reduces the quantity of channels before increasing them. A SAM is depicted in Figure 9, where channel reduction is achieved by a convolutional kernel with a size of 7, reducing computational load, followed by a convolution operation with a 7-sized kernel to boost the number of channels while preserving channel count consistency. Finally, a Sigmoid output is obtained.

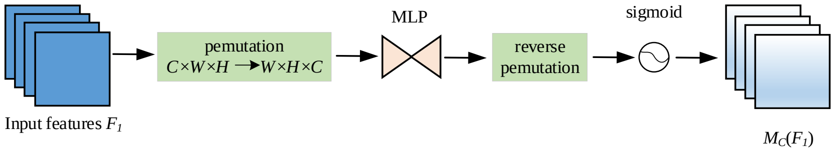

Figure 8. CAM.

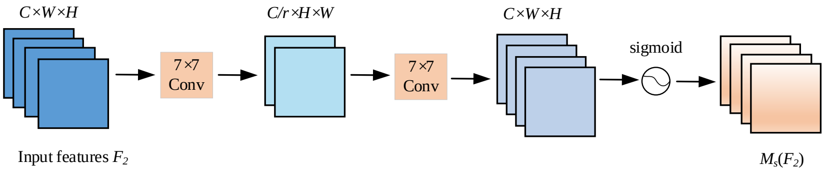

Figure 9. SAM.

2.4. Incorporate dynamic snake convolution

The receptive field of the traditional convolutional kernel is regular, but the shape of road cracks is irregular, causing some receptive fields not to be on the target. Additionally, the receptive field size of the convolutional kernel is fixed, but the size and extent of road cracks vary. If the target is too large, only local features can be extracted, and if the target is too small, interference from irrelevant information occurs. To address the limitation of traditional convolution in effectively adapting to geometric changes in objects, which makes it difficult to recognize objects undergoing rotation, symmetry, and scaling, dynamic snake convolution can be employed [27].

Inspired by deformable convolution [28], the model, in the process of learning features, obtains dynamic snake convolution by changing the shape of the convolutional kernel. Deformable convolution predicts offsets for sampling points, adaptively changes the sampling positions, and focuses on semantic feature points and geometric keypoints of the target. The sampling process of the convolutional kernel is illustrated in Figure 10. The input image first passes through the convolutional branch to calculate the offset. The output feature map has the same size as the input image, with dimensions of 2N. Then, based on the offset, the adaptive sampling points of the backbone network are obtained. Next, using bilinear interpolation, the feature values of the sampling points are obtained. However, the learning of the offset for sampling points is highly free, which may lead to positions far from the target. Moreover, each sampling point has the same weight for the output, and poor-quality sampling points can interfere with feature extraction. Dynamic snake convolution introduces continuity constraints into the design of the convolutional kernel. Each convolutional position is based on its previous position as a reference, freely choosing the oscillation direction and ensuring continuity of perception while allowing for free selection. This enables the convolutional kernel to fit structures and learn features freely on one hand, and on the other hand, it ensures that the convolutional kernel does not deviate too far from the target structure under constraint conditions. By adding dynamic snake convolution to the feature fusion network section of YOLOv5s, the improved network enhances the model's ability to handle irregular shapes and deformation problems. The adaptive changes in the receptive field size based on the target size are beneficial for improving the accuracy of road crack recognition.

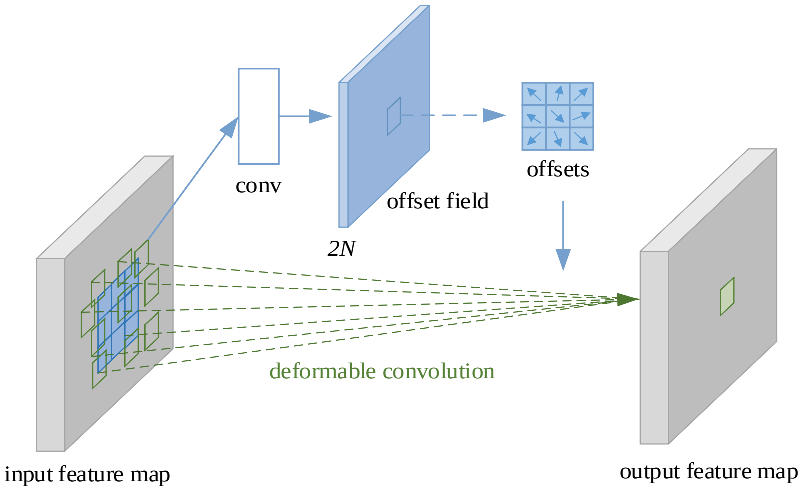

Figure 10. Deformable convolution sampling process.

3. Experimental data

3.1. Dataset introduction

The data collection device for the experiment is a multipurpose road inspection vehicle. On the inspection vehicle, there are two cameras arranged in parallel for capturing road information. The images are processed in grayscale to reduce the original data volume and enhance image information. There are 3000 pictures in the dataset that were utilized for this experiment. Training, validation, and test sets of these photos are split into 8:1:1 ratios at random. A portion of the photos is used as the test set, a portion as the validation set, and a portion as the training set. The dataset’s pictures used in the experiment are in JPG format with a size of 500 × 500 pixels.

3.2. Experimental environment

The experiments in this paper's software environment are built using PyTorch 1.8.0, a deep learning framework. It utilizes the GPU (graphics processing unit) of the NVIDIA GeForce RTX 3060 model for accelerated processing. The code is written in Python version 3.8, and it runs on the Windows 10 operating system. The hardware includes an Intel Xeon W-3225 processor, and the acceleration library is CUDA 10.0. There shall be 200 training batches in all [29], and the batch size is configured as 8. The weight file used is YOLOv5s.pt, with an initial learning rate of 0.001, a momentum of 0.9, a weight decay rate of 0.0005, and label smoothing set to 0.1.

4. Experimental results and analysis

4.1. Evaluation metrics

The mAP, recall (R), frames per second (FPS), and computational complexity are frequently employed in deep learning to assess the efficacy of models. These are their formulas:

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

where: *TP* represents the number of true positive detections of positive samples; *FP* represents the number of false detections of negative samples; *TN* represents the number of false detections of positive samples; and *FN* represents the number of true negative detections of negative samples.

mAP, which is the average of average precision (AP), is a key metric for object detection algorithms [30]. In object detection models, a higher mAP indicates better detection results on a specific dataset. FPS, which gauges the model detection speed, is employed to evaluate the fracture detection speed; a higher FPS value indicates faster detection and better model performance. The computational complexity of a convolutional neural network model is represented by the number of floating-point operations, known as FLOPs. FLOPs are used as an indirect measure of the speed of neural network models. A smaller FLOPs value indicates lower model complexity and faster target recognition and is calculated as follows:

$$FLOPs\_C = 2C_{in}K^2HWC_{out} \tag{3}$$

where: C_{in} represents the number of input channels; *K* represents the convolutional kernel size; *HW* represents the size of the output feature map; and C_{out} represents the number of output feature map channels [31].

4.2. Comparative experiments

To illustrate the appropriateness and efficacy of selecting the Res2-C3 module as a multi-scale extraction module, a horizontal comparison of the performance of Res2-C3 and C3RFEM as multi-scale feature extraction modules is presented. The outcomes of the comparative experiments are displayed in Table 1.

Table 1. Comparative experimental results of multi-scale extraction modules.

module	mAP/50%	AP/%(groove)	Recall/%(groove)	FPS/(frame/s)	FLOPs
YOLOv5s	0.838	0.816	0.766	76.324	15.8
Res2-C3	0.897	0.872	0.83	53.16	14.4
C3RFEM	0.873	0.85	0.798	41.308	10.1

Table 1 illustrates how the Res2-C3 module model strikes a balance between speed and accuracy of detection. Regarding detection accuracy, compared to the original model and the model using the C3RFEM module, the model using the Res2-C3 module improved mAP by 5.9 and 2.4%, respectively. The model with the Res2-C3 module reduced the risk of overfitting because of a decrease in parameters, but it also exhibited a modest decrease in FPS when compared to the original model in terms of detecting speed. In comparison to the model using the C3RFEM module, the FPS increased by 11.852 frames/s, significantly improving the detection speed. Therefore, the Res2-C3 module was chosen as the multi-scale module for the final improved model.

4.3. Ablation experiments

To fully validate the effectiveness of the proposed improvements in this paper, ablation experiments were conducted on the road crack dataset. The label smoothing for all models was set to 0.1 to prevent model overfitting. Each improvement module was embedded into the YOLOv5s model one by one, and the same training parameters and environmental conditions were used in each experiment. Table 2 displays the outcomes of the experiment. In terms of detection accuracy, the model with the highest mAP is the YOLOv5s+Res2-C3+GAM+DSConv model, which improved mAP by 10.1% compared to the YOLOv5s model. When Res2-C3, GAM, and DSConv (dynamic snake convolution) act individually, the highest mAP value is achieved by the YOLOv5s+Res2-C3 model, with an mAP value of 89.7%, which is 5.9% higher than the YOLOv5s model. This indicates that by stacking numerous bottleneck modules, the Res2-C3 module improves the model’s feature extraction capabilities and enhances the road fracture detection accuracy. When any two modules are combined, the model’s detection accuracy is improved to varying degrees. This suggests the possibility of a synergistic effect between the modules where they complement and enhance each other’s capabilities, contributing to improved accuracy and robustness.

In terms of the detection speed, FPS represents the model’s speed in terms of detection speed, with higher FPS indicating faster detection. According to the experimental results, the fastest detection model is YOLOv5s, with the FPS value as 72.324. Compared to the original model, all improved models experienced a decrease in detection speed. Among these, the largest decrease in detection speed was observed in the YOLOv5s+Res2-C3+GAM+DSConv model with an FPS value of 49.97 FPS. This is mainly because obtaining more feature information leads to an increase in the model's parameter count. The model requires more computations and weight updates, which increases the time cost of training and reduces detection speed.

Table 2. Improved module ablation experimental results.

module	mAP/50%	AP/%(groove)	Recall/%(groove)	FPS/(frame/s)
YOLOv5s	0.838	0.816	0.766	72.324
YOLOv5s+Res2-C3	0.897	0.872	0.83	53.16
YOLOv5s+GAM	0.885	0.876	0.815	62.449
YOLOv5s+DSConv	0.859	0.828	0.782	51.944
YOLOv5s+Res2-C3+GAM	0.934	0.927	0.863	53.985
YOLOv5s+Res2-C3+DSConv	0.917	0.905	0.842	50.917
YOLOv5s+GAM+DSConv	0.922	0.913	0.851	51.869
YOLOv5s+Res2-C3+GAM+DSConv	0.939	0.942	0.871	49.97

In terms of recall rate, the best-performing model is the YOLOv5s+Res2-C3+GAM+DSConv model, achieving a recall rate of 87.1%, which is a 10.5% improvement over the original model.

In summary, through the comparison of multiple indicators, the improved YOLOv5s+Res2-C3+GAM+DSConv model performs the best, with a 12.6% increase in AP for road crack detection and a 10.1% increase in mAP. Since the final improvement model adds three modules relative to the YOLOv5s network, the complexity of the model increases, and it is reasonable that the speed will decrease. The final improved model strikes a balance between mAP and FPS, maintaining high accuracy while achieving relatively fast speed.

4.4. Model Comparison

Tests were conducted on the YOLOv5s and the final improved model on the above-mentioned dataset, and their various parameters were compared with the newer YOLOv7 and YOLOv8 networks. Table 3 shows that, compared to other networks, the final improved model YOLOv5s+Res2-C3+GAM+DSConv has better parameters in terms of mAP and AP.

Table 3. Comparison of different models.

module	mAP/50%	AP/%(groove)	Recall/%(groove)	FPS/(frame/s)	FLOPs
YOLOv5s	0.838	0.816	0.766	76.324	15.8
YOLOv5s+Res2-C3+GAM+DSConv	0.939	0.942	0.871	49.97	18.4
YOLOv7	0.891	0.903	0.862	54.62	16.3
YOLOv8	0.924	0.915	0.892	50.479	17.9

Compared to YOLOv7 and YOLOv8, YOLOv5 has the lowest accuracy, with mAP values for YOLOv7 and YOLOv8 models being 5.3 and 8.6% higher than YOLOv5, respectively. However, YOLOv8's FPS significantly decreases, leading to slower detection speed. Due to the larger model parameters and more complex network structures of YOLOv7 and YOLOv8, the detection speed becomes slower.

Compared to YOLOv7 and YOLOv8 models, the final improved model YOLOv5s+Res2-C3+GAM+DSConv has the highest detection accuracy, with mAP values 4.8 and 1.5% higher than the YOLOv7 and YOLOv8 models, respectively. In terms of detection speed, the YOLOv5s+Res2-C3+GAM+DSConv model has an FPS value only 0.509 FPS lower than YOLOv8. Although the YOLOv5s+Res2-C3+GAM+DSConv model exhibits a slight decrease in detection speed compared to YOLOv8, it achieves a 2.7% increase in detection accuracy over YOLOv8. Overall, the YOLOv5s+Res2-C3+GAM+DSConv model strikes a balance between detection accuracy and speed.

4.5. Analysis of detection results

The enhanced model's detection results are shown in Figure 11. The numbers represent confidence scores, indicating the model's confidence level in its predictions. High confidence scores indicate that the model is very confident in its predictions, while low confidence scores indicate that the model is less certain about the results [32]. From the detection results of road cracks in the images, it is evident that not only were small cracks successfully detected, but these cracks also exhibited relatively high confidence scores. This observation strongly validates the feasibility and effectiveness of the improved algorithm, demonstrating its outstanding performance in crack detection tasks and providing robust support for road maintenance and safety.

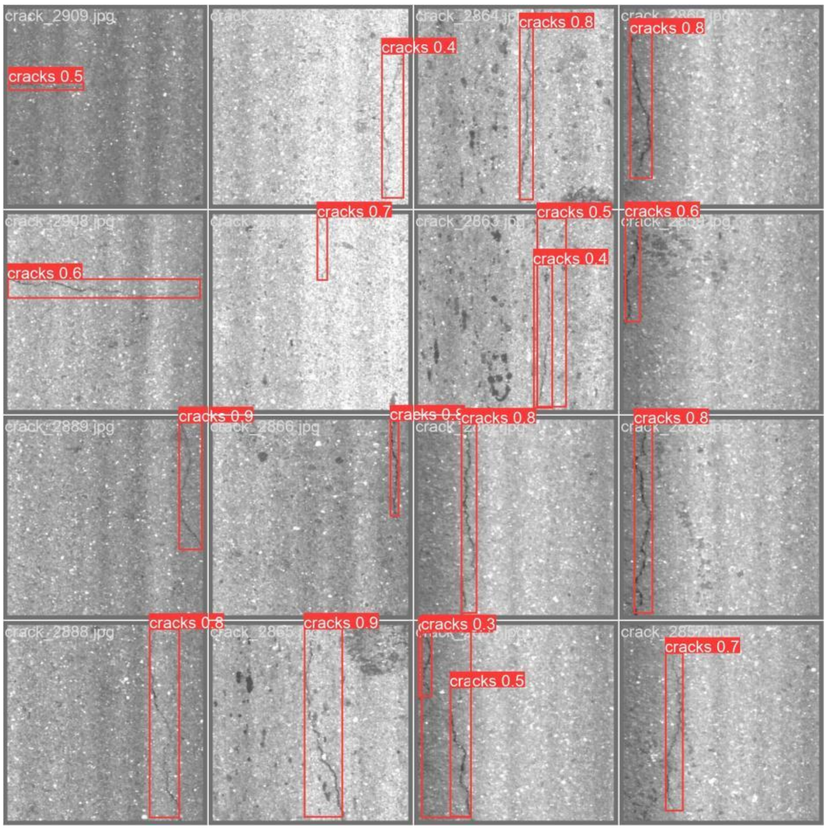

Figure 11. Detection result.

5. Conclusions

In reaction to the shortcomings of conventional techniques for detecting road cracks, such as their poor speed and low accuracy, this paper proposes a method to improve the YOLOv5s model. The method involves using the proposed Res2-C3 module as the core of the backbone network to replace the original C3 module, enabling the extraction of more feature information from input images to reduce the omission of valuable information and increase detection accuracy. Furthermore, the GAM attention mechanism is added to both the YOLOv5s backbone network and the feature fusion network to increase the model's focus on crack information and reduce the false-negative rate for small cracks. Adding dynamic snake convolution to the feature fusion network enables the model's receptive field to adaptively change with the size of road cracks, which is advantageous for improving the accuracy of road crack detection. The paper also introduces label smoothing, setting the label smoothing value to 0.1, to enhance the model's generalization. This model achieves a balance between detection accuracy and speed while controlling the model's parameter size.

To enhance the model's accuracy and robustness, future research should consider collecting more samples of different types of road cracks and using data augmentation techniques to increase dataset diversity. Furthermore, considering real-time road crack detection and application on mobile devices would enhance the algorithm's practicality.

Use of AI tools declaration

The author declares they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported in part by the Shandong Provincial Department of Science and Technology Research Project on Key Technologies for the Development of All-Terrain Intelligent Orchard Platform. (2019GNC106032).

Conflict of interest

The author declares there is no conflict of interest.

References

1. C. .P. Meng, J. P. Li, J. Guo, C. L. Li, Analysis of common problems in ecological impact investigation of highway environmental protection acceptance, *Res. Cons. Envir. Prot.*, **4** (2023), 121–124. https://doi.org/10.16317/j.cnki.12-1377/x.2023.04.015
2. W. Zhou, Y. He, J. Li, Dangerous behavior detection in gas stations based on deep learning, in *2023 IEEE 6th International Conference on Electronic Information and Communication Technology*, (2023), 935–939. https://doi.org/10.1109/ICEICT57916.2023.10245093
3. N. Sholevar, A. Golroo, S. R. Esfahani, Machine learning techniques for pavement condition evaluation, *Autom. Constr.*, **136** (2022), 104190. https://doi.org/10.1016/j.autcon.2022.104190

4. H. Bello-Salau, A. M. Aibinu, E. N. Onwuka, J. J. Dukiya, A. J. Onumanyi, Image processing techniques for automated road defect detection: A survey, in *International Conference on Electronics*, (2014), 1–4. https://doi.org/10.1109/ICECCO.2014.6997556
5. S. Chatterjee, P. Saeedfar, S. Tofangchi, L. M. Kolbe, Intelligent road maintenance: a machine learning approach for surface defect detection, in *European Conference on Information Systems*, 2018.
6. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016), 779–788.
7. S. Park, S. Bang, H. Kim, H. Kim, Patch-based crackdetection in black box images using convolutional neural net-works, *J. Comput. Civil Eng.*, **33** (2019). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000831
8. X. B. Su, Research on pavement crack detection based on improved YOLOv4, *Henan Sci. Technol.*, **41** (2022), 62–67. https://doi.org/10.19968/j.cnki.hnkj.1003-5168.2022.18.012
9. M. M. Wang, Q. D. Huang, S. N. Liu, Pavement damage detection based on improved YOLOv5s, *J. Lasers*, **44** (2023), 66–71. https://doi.org/10.14016/j.cnki.jgzz.2023.05.066
10. J. Terven, D. Cordova-Esparza, A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond, preprint, arXiv: 2304.00501.
11. J. Lu, M. Zhu, X. Ma, K. Wu, Steel strip surface defect detection Metho based on improved YOLOv5s, *Biomimetics*, **9** (2024), 28. https://doi.org/10.3390/biomimetics9010028
12. Y. Zhou, W. Zhu, Y. He, Y. Li, Yolov8-based spatial target part recognition, in *2023 IEEE 3rd International Conference on Information Technology, Big Data and Artificial Intelligence*, (2023), 1684–1687. https://doi.org/10.1109/ICIBA56860.2023.10165260
13. H. Liu, F. Sun, J. Gu, L. Deng, Sf-yolov5: A lightweight small object detection algorithm based on improved feature fusion mode, *Sensors*, **22** (2022), 5817. https://doi.org/10.3390/s22155817
14. J. Zhou, Z. Xi, S. Wang, B. Yang, Y. Zhang, Y. Zhang, A real spatial–temporal attention denoising network for nugget quality detection in resistance spot weld, *J. Intell. Manuf.*, **2023** (2023), 1–22. https://doi.org/10.1007/s10845-023-02160-x
15. C. Y. Wang, H. Y. M. Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, IH Yeh CSPNet: A new backbone that can enhance learning capability of CNN, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, (2020), 390–391.
16. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, preprint, arXiv: 1409.1556.
17. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, et al., Going deeper with convolutions, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2015), 1–9.
18. X. Jiang, H. Hu, Y. Qin, Y. Hu, R. Ding, A real-time rural domestic garbage detection algorithm with an improved YOLOv5s network model, *Sci. Rep.*, **12** (2022), 16802. https://doi.org/10.1038/s41598-022-20983-1
19. S. H. Gao, M. M. Cheng, K. Zhao, X. Y. Zhang, M. H. Yang, P. Torr, Res2Net: A New Multi-Scale Backbone Architecture, *IEEE Trans. Pattern Anal. Mach. Intell.*, **2** (2021), 43. https://doi.org/10.1109/TPAMI.2019.2938758

20. U. Batool, M. I. Shapiai, S. A. Mostafa, M. Z. Ibrahim, An attention-augmented convolutional neural network with focal loss for mixed-type wafer defect classification, *IEEE Access*, **11** (2023), 108891–108905. https://doi.org/10.1109/ACCESS.2023.3321025
21. Z. Yu, H. Huang, W. Chen, Y. Su, Y. Liu, X. Wang, Yolo-facev2: A scale and occlusion aware face detector, preprint, arXiv: 2208.02019.
22. J. Cai, J. Hu, 3D RANs: 3D residual attention networks for action recognition, *Vis. Comput.*, **36** (2020), 1261–1270. https://doi.org/10.1007/s00371-019-01733-3
23. J. Hu, L. Shen, G. Sun, Squeeze-and-Excitation networks, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2018), 7132–7141. https://doi.org/10.1109/CVPR.2018.00745
24. S. Woo, J. Park, J. Y. Lee, I. S. Kweon, Cbam: Convolutional block attention module, in *Proceedings of the European conference on computer vision (ECCV)*, (2018), 3–19.
25. Y. Liu, Z. Shao, N. Hoffmann, Global attention mechanism: Retain information to enhance channel-spatial interactions, preprint, arXiv: 2112.05561.
26. Z. Guo, Y. Li, Y. Tian, H. Liu, S. Yuan, C. Hou, Global attention-based approach for substation devices classification and localization, in *2023 IEEE/IAS Industrial and Commercial Power System Asia*, (2023), 990–995. https://doi.org/10.1109/ICPSAsia58343.2023.10294513
27. Y. Qi, Y. He, X. Qi, Y. Zhang, G. Yang, Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2023), 6070–6079. https://doi.org/10.1109/ICCV51070.2023.00558
28. J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, et al., Deformable convolutional networks, in *Proceedings of the IEEE international conference on computer vision*, (2017), 764–773. https://doi.org/10.1109/ICCV.2017.89
29. Z. Wu, R. Xue, H. Li, Real-time video fire detection via modified YOLOv5 network model, *Fire Technol.*, **58** (2022), 2377–2403. https://doi.org/10.1007/s10694-022-01260-z
30. Y. Wang, G. Fu, A novel object recognition algorithm based on improved YOLOv5 model for patient care robots, *Int. J. Hum. Robot.*, **19** (2022). https://doi.org/10.1142/S0219843622500104
31. L. Shi, S. Zhao, W. Niu, A welding defect detection method based on multiscale feature enhancement and aggregation, *Nond. Testing and Eval.*, (2023), 1–20. https://doi.org/10.1080/10589759.2023.2253494
32. E. R. Daniel, Wildfire smoke detection with computer vision, preprint, arXiv: 2301.05070.