Structural Durability & Health Monitoring

Tech Science Press

Doi:10.32604/sdhm.2025.063887

ARTICLE

Investigation of Attention Mechanism-Enhanced Method for the Detection of Pavement Cracks

Tao Jin[1,*], Siqi Gu[1], Zhekun Shou[1], Hong Shi[2] and Min Zhang[2]

[1]College of Civil Engineering, Zhejiang University of Technology, Hangzhou, 310014, China
[2]Road & Bridge International Co., Ltd., Beijing, 100027, China

*Corresponding Author: Tao Jin. Email: jintao@zjut.edu.cn

Received: 27 January 2025; Accepted: 24 March 2025; Published: 30 June 2025

ABSTRACT: The traditional You Only Look Once (YOLO) series network models often fail to extract satisfactory features for road detection, due to the limited number of defect images in the dataset. Additionally, most open-source road crack datasets contain idealized cracks that are not suitable for detecting early-stage pavement cracks with fine widths and subtle features. To address these issues, this study collected a large number of original road surface images using road detection vehicles. A large-capacity crack dataset was then constructed, with various shapes of cracks categorized as either cracks or fractures. To improve the training performance of the YOLOv5 algorithm, which showed unsatisfactory results on the original dataset, this study used median filtering to preprocess the crack images. The preprocessed images were combined to form the training set. Moreover, the Coordinate Attention (CA) attention module was integrated to further enhance the model's training performance. The final detection model achieved a recognition accuracy of 88.9% and a recall rate of 86.1% for detecting cracks. These findings demonstrate that the use of image preprocessing technology and the introduction of the CA attention mechanism can effectively detect early-stage pavement cracks that have low contrast with the background.

KEYWORDS: Road detection vehicle; pavement crack detection; deep learning; attention mechanism

1 Introduction

The road structure is a critical component of traffic infrastructure, providing a smooth and comfortable driving environment for vehicle traffic [1]. However, due to factors such as severe weather, freeze-thaw cycles, overweight vehicles, and construction defects, road structures may develop diseases [2]. These issues can reduce their service life, increase maintenance costs, and compromise driving safety. Cracks are a common type of pavement defect that is easily induced by the aforementioned factors. Cracks allow rainwater to penetrate the pavement and weaken the local strength. Moreover, without proper and timely maintenance, cracks can grow and develop into potholes, posing risks to traffic comfortableness [3]. Therefore, to ensure driver safety and maintain pavement service levels, frequent crack detection is necessary. Over the past few decades, researchers have proposed a variety of methods for detecting road cracks. In terms of the inspection process, these methods can be mainly categorized into manual inspection [4] and computer vision methods [5–7]. Traditional detection methods largely rely on manual inspection, which often has significant drawbacks. They are usually labor-intensive, inefficient, subjective, prone to high rates of missed detection, and will cause traffic interruption.

Since pavement surface cracks are visible features that can be captured through images, engineers and scholars have explored methods for detecting cracks via image analysis. Conventional image recognition methods primarily identify faulty regions based on picture characteristics such as grayscale values [8]. The most common techniques include the Otsu method [9], edge detection [10], and region-growing algorithms [11]. These methods can achieve satisfactory recognition performance for crack detection in images with simple backgrounds and strong contrast. However, the recognition error rate is high for images with complex backgrounds. While segmentation methods offer relatively high accuracy in identifying pavement crack faults, their processing speed is slow [12]. Additionally, methods that rely on picture textures, brightness, and connectivity to identify cracks, as well as those using dynamic thresholding algorithms, can only produce coarse fracture patterns [13]. Moreover, crack recognition based on Canny edge detection is prone to erroneous recognition. Overall, traditional image recognition methods require high standards for image acquisition and processing in complex environments and diverse crack patterns. Their algorithms may not be adaptable enough to handle the complex and variable urban road environment, leading to false positives or missed detection. As a result, they are less effective in accurately identifying cracks, necessitating manual feature extraction due to their low accuracy and high false-positive rates. Additionally, the preprocessing approach has a direct impact on recognition performance [14,15]. These methods are not superior at identifying intricate cracks and can be fooled by noise patterns. They are also more sensitive to interference and ambient light.

To overcome the limitations and difficulties associated with manually extracting damage-sensitive features and using traditional machine learning methods for classification, there has been a surge of research on structural health monitoring based on deep learning. In 2006, Hinton et al. [16] introduced the concept of deep learning, leveraging the multi-layer abstraction process of the human brain to generate abstract representations of data such as images and language. Deep neural networks do not require feature engineering; once adequately trained, they can effectively respond to the properties of the data, although it may be challenging to understand and articulate how each layer is computed. Thus, deep learning is a superior solution for the multi-scale nature of pavement crack identification and the complex environment with diverse backgrounds. In recent years, deep learning algorithms have made significant progress in detecting pavement faults. Zhang et al. [17] proposed a road fracture detection system based on deep convolutional neural networks (CNN), using a rectified linear unit (ReLU [18]) as the activation function to accelerate model convergence during training. Jiang et al. [19] employed drones to capture images of cracks and identify them in real time, enabling crack width measurement. Deep learning is frequently combined with CNN techniques to perform detection tasks [20]. Geetha et al. [21] developed a concrete crack detection and classification method using dynamic threshold image binarization and 1D-DFT-CNN (1D Discrete Fourier Transform Convolutional Neural Network) to enhance real-time crack detection and classification efficiency. Yang et al. [22] introduced a Feature Pyramid and Hierarchical Boosting Network (FPHBN)-based crack detection method for concrete, improving boundary localization and accuracy. This method demonstrated superior accuracy, speed, and robustness over other approaches, especially in challenging images with shadows and low contrast. Lu et al. [23] presented an advanced road crack detection method using a U-Net framework combined with a pyramid-assisted supervision module and a spatial-channel dual attention module, achieving high accuracy and robustness under various conditions. Cha et al. [24] proposed a vision-based concrete crack detection method that utilized a deep CNN structure to automatically learn image features without manually computing defect features.

The YOLO series, as an advanced single-stage target detection algorithm, is known for its fast detection speed and high accuracy, evolving up to YOLOv11. While YOLOv11 has seen significant improvements in architecture and performance, YOLOv5 retains several key advantages and remains highly relevant for

specific applications. Since its release in 2020, YOLOv5 has undergone multiple iterations and optimizations. Its maturity and stability make it a widely used and reliable target detection model in both industry and academia. YOLOv5 offers a variety of model sizes, including YOLOv5n, YOLOv5s, and YOLOv5m, which are suitable for a range of scenarios from embedded devices to high-performance servers. The lightweight design of YOLOv5 also performs well in resource-constrained environments. YOLOv5 features a simple training process [25], low training and inference costs, and supports a variety of data augmentation techniques and hyperparameter optimization methods. Additionally, it offers a rich model export format, facilitating deployment on different platforms. In specific tasks, YOLOv5's transfer learning performance is slightly better compared to YOLOv11, despite YOLOv11's more stable mean Average Precision (mAP). Khanam et al. [26] compared three target detection models—YOLOv5, YOLOv8, and YOLOv11—and found that YOLOv5 offers satisfactory computational efficiency while maintaining high accuracy.

In recent years, an increasing number of studies have combined attention mechanisms with YOLOv5. By introducing different attention mechanisms, such as SE (Squeeze-and-Excitation), CBAM (Convolutional Block Attention Module), ECA (Efficient Channel Attention), and CA (Coordinate Attention), the performance of YOLOv5 in the target detection task has been significantly improved. Zhou et al. [27] optimized the YOLOv5s model by introducing a lightweight coordinate attention module, enabling the network to more accurately locate targets and improve detection accuracy. The experimental results indicated that the improved YOLOv5 model can effectively identify road cracks. Liu et al. [28] proposed a lightweight object detection algorithm based on the attention mechanism and YOLOv5. Experimental results on remote sensing datasets, such as RSOD (Remote Sensing Object Detection Dataset) and DIOR (Detection in Optical Remote), showed that the algorithm improved average accuracy by 1.4% and 1.2%, respectively, compared to the YOLOv5s algorithm.

Aside from the development of detection models, crack datasets are of great importance for crack detection tasks. Many scholars have created their own crack sample sets, but the sample sizes are usually small, ranging from a few hundred to a thousand photos [29]. For example, the Crack Forest Dataset (CFD) contains 118 pavement photos, while the Crack Water Hole Dataset (CrackWH100) includes 100 pavement images. However, training deep neural networks requires large datasets. To address the challenge of crack identification using deep learning, a significant number of pavement crack photos need to be gathered and labeled. To compensate for the lack of sample sets, many researchers frequently employ data augmentation techniques. Yet, the actual crack condition in real-world contexts still plays a crucial role in the model training process. Therefore, a dataset containing a large number of labeled pavement fractures is essential for deep learning-based pavement crack identification.

Moreover, crack quantification is a crucial step in structural health monitoring, providing a basis for structural damage assessment by obtaining the geometric characteristics of cracks [30]. After deep learning-based crack detection, quantification is achieved through image processing techniques. These include using edge detection algorithms (such as Sobel and Canny) to extract edge information, applying global or adaptive thresholding methods for segmentation, and expanding the crack area using region growing algorithms. The geometric features of the crack region are then calculated to obtain crack characteristics. One approach is to use bounding boxes to isolate the crack region and apply image processing for segmentation. Kang et al. [31] used an object detection model to locate cracks and proposed an improved distance transformation method (DTM) to measure crack thickness and length in pixels. This method achieved an accuracy of 93% for crack length and width measurements across 100 tested images. Another approach involves using semantic segmentation networks, followed by image processing to fix the crack region and carry out quantification. Quantitative indicators for cracks include length (obtained by extracting crack skeleton lines), width (calculated from the pixel distance between crack edges), and area (estimated by counting pixels

in the crack region). Other metrics include crack density and geometric shape. New methods for crack quantification are rapidly emerging. Peng et al. [32] introduced an automatic detection and quantification method for bridge cracks, quantifying crack width by measuring the minimum distance between crack edges in segmented images. Guo et al. [33] proposed estimating crack width by calculating the crack angle using the cosine function, achieving a crack detection accuracy of 0.992.

Regarding the practical application of computer vision-based pavement crack detection, developing an integrated detection system is essential for rapid image collection. These techniques address the disadvantages of traditional manual detection, saving personnel and material resources while eliminating the influence of human subjectivity [34]. For example, the Canadian company Roadware developed the ARAN (Automatic Road Analyzer) detection vehicle, equipped with two charge-coupled device (CCD) cameras for image data collection [35]. The ARAN uses a synchronized, high-intensity flash to reduce shadows from objects like trees and fences, even in direct sunlight, and can theoretically capture road surface images in the absence of natural light. Similarly, the University of Arkansas developed the Digital Highway Data Vehicle (DHDV), a real-time pavement defect detection system that uses a CCD camera mounted on a test vehicle to capture images of pavement degradation [36]. It employs a Global Positioning System (GPS) to locate fractures and a distance measurement instrument [37] (DMI) to collect distance information. The data is processed by a dual-CPU microprocessor and sent in real time to a multi-CPU computer for damage analysis. This system integrates digital image acquisition and processing components to enable rapid data collection, detection, and classification of pavement cracks.

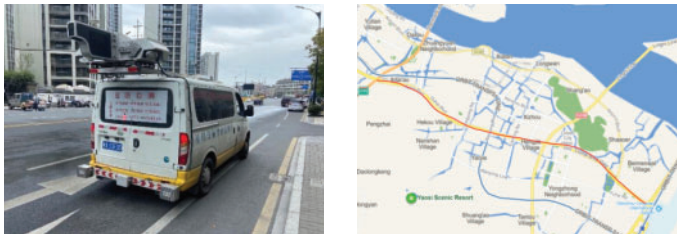
This paper proposes a method for detecting early pavement cracks based on deep learning and image processing. To collect sufficient training image samples in real-world scenarios, a road detection vehicle equipped with a camera module was used. Modifications, including image processing and the incorporation of an attention mechanism into YOLO, were investigated to enhance crack detection capabilities. The results indicated that training with annotated raw images alone was insufficient for detecting early pavement cracks with low contrast against the background. Enhancing YOLOv5's detection ability through image processing and an attention module significantly improved performance. This study provides a reference for detecting early pavement cracks, aiding in preventive maintenance.

2 Establishment of Pavement Crack Dataset

Deep learning is a presentation learning method that enables a network model to learn relevant features and representations directly from raw data and perform classification tasks. However, deep CNN models typically involve a large number of parameters that need to be optimized and adjusted. Therefore, in target detection tasks involving pavement cracks, a substantial number of crack images are often required to train neural networks effectively. Additionally, in real-world scenarios, pavement cracks vary in length, width, and shape, and many crack images contain various interference factors, including uneven illumination, water stains, shadows, manhole covers, etc. Consequently, an ideal pavement crack dataset should include a large number of crack images with diverse geometries and different types of noise patterns.

The image acquisition experiment collected pavement images from different sections and under varying weather conditions at different times. This study captured approximately 60 km of road surface images in cities such as Wenzhou and Jinhua in Zhejiang Province. A Road detection vehicle equipped with a camera module was used for image collection, resulting in a dataset of 10,000 road surface images with a resolution of 4000×2000 pixels. The pavement images collected in Jinhua, Zhejiang Province, exhibit moderate brightness with minimal interference, such as leaf shadows. Linear cracks are the predominant type, accompanied by some mesh cracks. These linear cracks tend to be longer in length but narrower in width. In contrast, the road surface images collected from Ouhai Avenue in Wenzhou are generally darker and contain more interference,

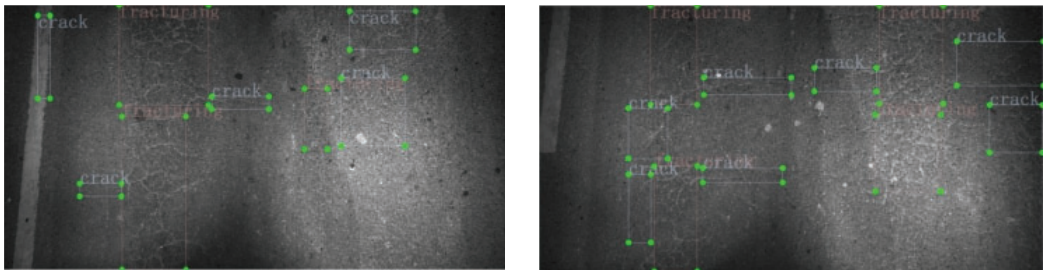
such as leaf shadows. Additionally, some images from Ouhai Avenue feature wet surfaces with more complex texture patterns. The road inspection vehicles and route maps are shown in Fig. 1.



Road detection vehicle Detection roadmap

Figure 1: Road detection vehicle and detection roadmap

In this paper, the LabelImg software was employed to further annotate the acquired images. A total of 1400 images containing cracks were annotated, and the pavement cracks were categorized into two types: cracks and fracturing, as illustrated in Fig. 2. The number of original images and processed datasets, the resolution of the images, and the sample distribution are summarized in Table 1.



(a) Crack with marking line interference (b) Densely cracked pavement surface

Figure 2: Typical label examples

Table 1: Dataset parameters

Raw image quantity	Crack image quantity	Image resolution	Training ratio setting
10,000	1400	4000 × 2000	8:1:1

3 Model Training and Evaluation

YOLOv5 [38] is an upgrade from YOLOv4 [39,40], and it offers four publicly available network architectures: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Among these, the YOLOv5s network has the shallowest depth and narrowest feature map width. The other three models are based on YOLOv5 and have been expanded and developed accordingly.

YOLOv5s consists of two major components: the backbone and the neck. The backbone of YOLOv5 employs the CSPDarknet (Cross Stage Partial Darknet) [41] and SPP (Spatial Pyramid Pooling) [42] framework. The Neck component of YOLOv5 utilizes the PANet (Path Aggregation Network) structure, whose primary function is to generate feature pyramids [43]. These feature pyramids enhance the detection of objects at multiple scales, enabling the recognition of the same object at different sizes and scales. The structure of the YOLOv5s model is shown in Fig. 3.

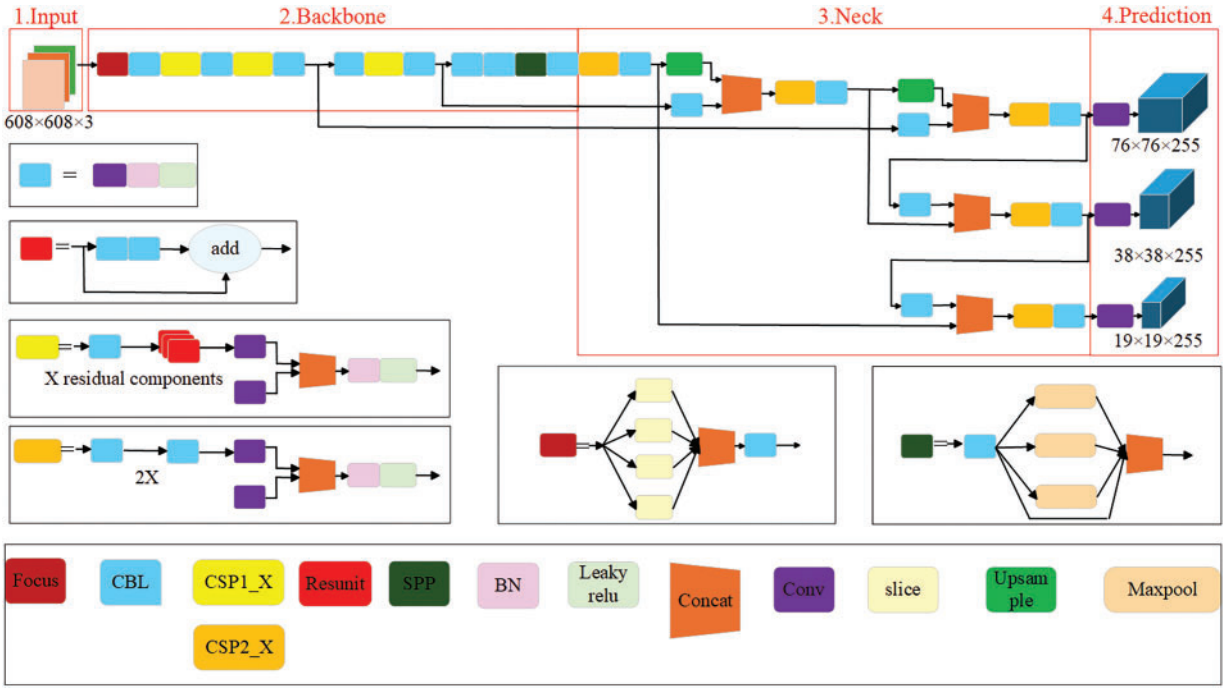


Figure 3: YOLOv5s model

Target detection algorithms are typically evaluated using metrics such as the F1 score, mAP, Recall, Precision. In deep learning, prediction results can be categorized into four types: True Positive (TP), when both the actual and predicted values are positive; False Negative (FN), when the actual value is positive but the prediction is negative; False Positive (FP), when the actual value is negative but the prediction is positive; and True Negative (TN), when both the actual and predicted values are negative. The calculation process is as follows:

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

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

The F1 score is a balance between Precision and Recall. It considers both the Precision rate and the completeness rate, delivering a proper balance between the two indicators. It is computed as follows:

$$F1 = \frac{2 * P * R}{P + R} \tag{3}$$

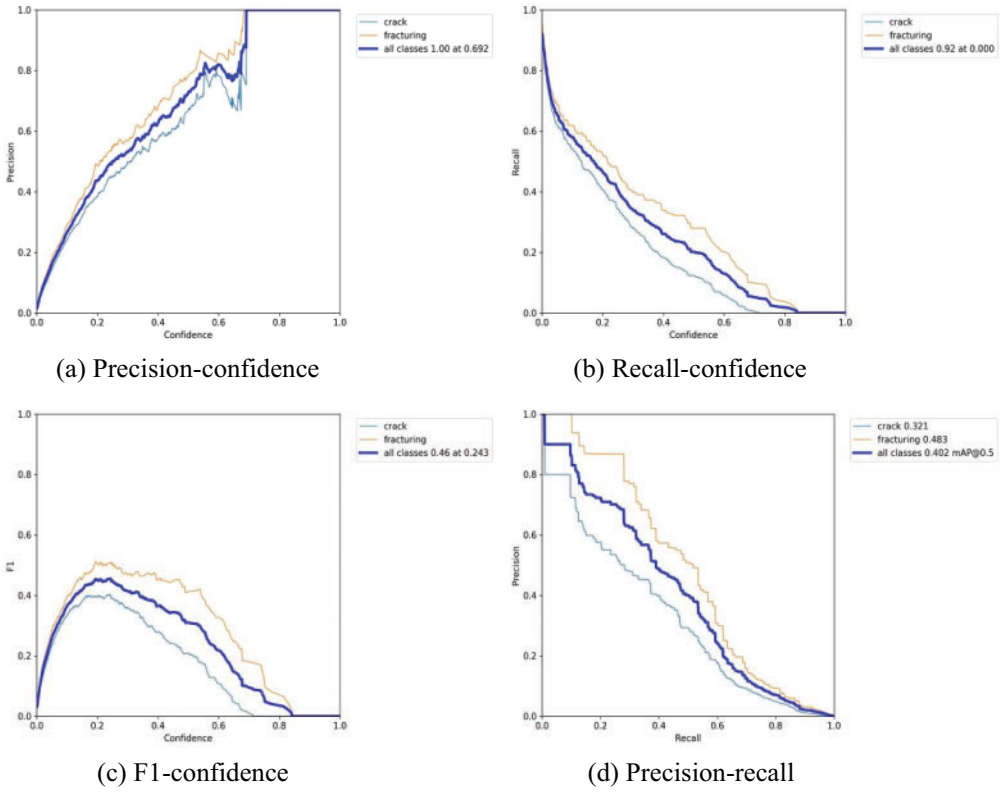
Precision-Recall Curve: The P-R curve is generated by computing Precision and Recall for each prediction and plotting the curve based on their relationship. Different computer vision tasks have varying tolerances for the two types of errors. Usually, efforts are made to reduce one type of error without exceeding a specific threshold for the others. In target detection, AP (Average Precision) serves as a comprehensive metric to balance both. The AP is calculated as the area under the interpolated Precision-Recall curve, bounded by the X-axis. It is calculated as follows:

$$\mathrm{AP} = \sum_{i=1}^{n-1} \left(r_{i+1} - r_i\right) p_{interp}\left(r_{i+1}\right) \tag{4}$$

The AP value is determined for only one category. After the AP is determined, calculating the mAP is straightforward. It is necessary to compute the average precision since AP is calculated for each category. By determining the mean value, the mAP assesses how well the trained model detects all categories. In this paper, the datasets were divided into training sets, test sets, and validation sets with a ratio of 8:1:1, and the training epoch was set to 100. A GPU server was used for training, and the parameters are presented in Table 2. The training results are illustrated in Fig. 4.

Table 2: GPU server parameters

Name	Version
CPU	Inter(R) Xeon(R) Silver 4215R CPU
GPU	NAVIDA GeForce RTX 3090
Cuda	11.5
PyTorch	11.0
PyCharm	pyCharm2022
Python	3.8



(a) Precision-confidence (b) Recall-confidence

(c) F1-confidence (d) Precision-recall

Figure 4: YOLOv5s training results

As can be seen, the training mAP is only 0.323. To improve the training effect, this paper used the median filtering method to enhance the images and improve the dataset. Median filtering [44,45] is primarily used to remove image noise. It is a nonlinear filtering technique and a statistical method that assigns each pixel's gray value as the median of the gray values of all pixels within a local neighborhood window. This technique

replaces individual pixel values in a digital image with the median value of the neighboring points, aiming to bring the neighboring pixel values closer to the actual values and thereby remove isolated noise points.

The core mechanism is as follows: Firstly, a square-shaped region centered on a specific pixel is identified. Then, the gray values of all pixels in this region are sorted, and the median value is chosen as the new gray value for the central pixel. This region is commonly referred to as the window. As the window scans across the image from top to bottom, the median filtering technique effectively smooths the image, particularly at the edges of image regions where gray values can fluctuate significantly and rapidly. The filter can remove these components and smooth the image. The schematic diagram of the median filtering is shown in Fig. 5. Typical examples of using median filtering for preprocessing are presented in Fig. 6. It can be seen that the filtered images are not very different from the original images. The reason might be that median filtering is used to remove isolated noise points. Due to the human eye's limited perception of detailed changes in the images, it may be difficult to observe significant differences with the naked eye. After median filtering, the number of crack images was increased to 2800.

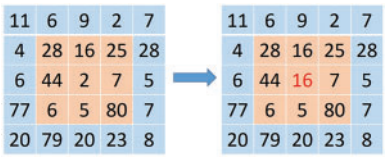
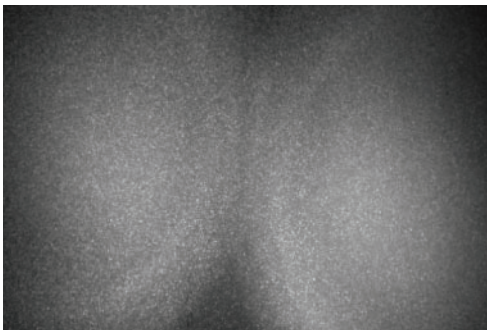
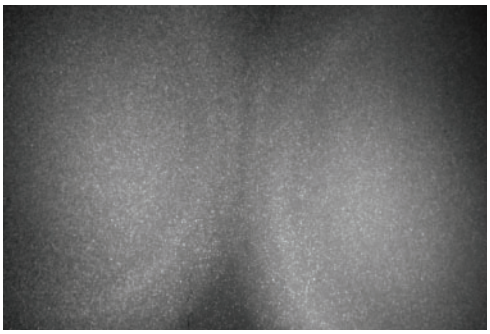


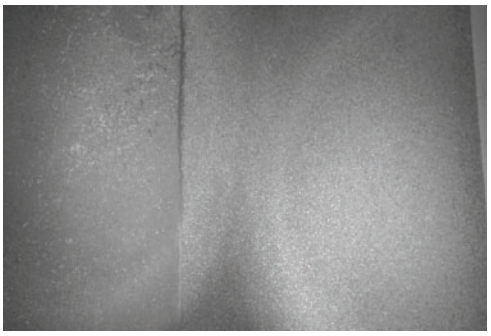
Figure 5: Median filtering sketch map



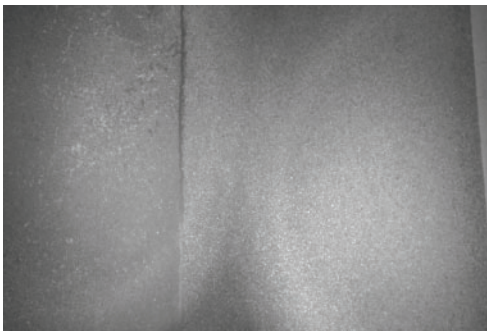
(a) Before median filtering



(b) After median filtering

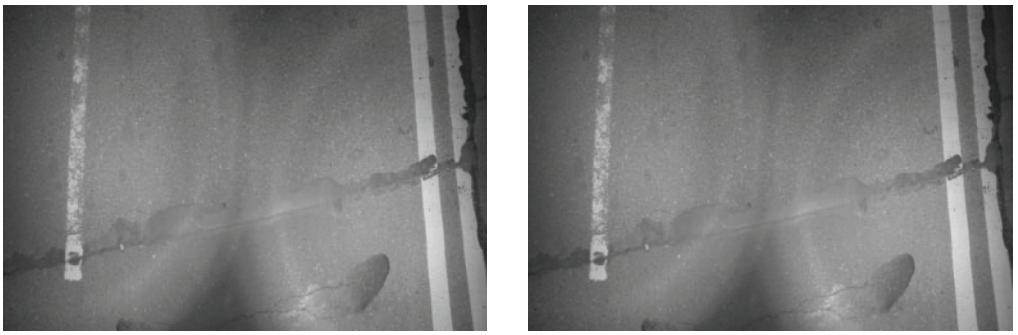


(c) Before median filtering



(d) After median filtering

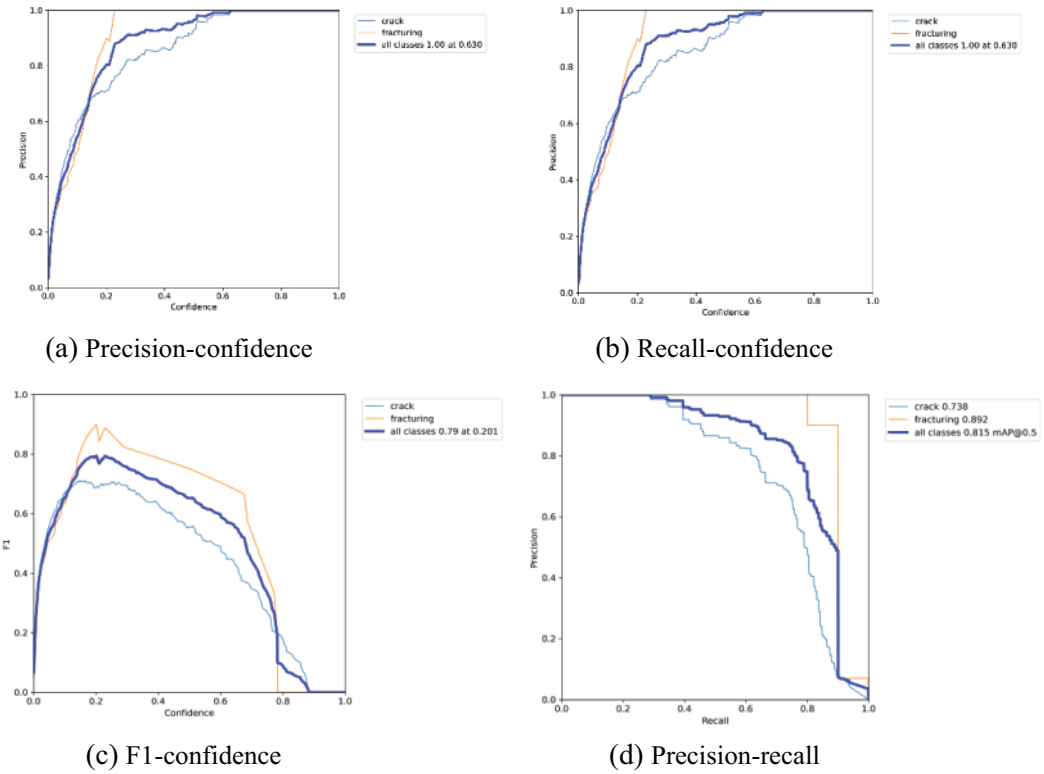
Figure 6: (Continued)



(e) Before median filtering (f) After median filtering

Figure 6: Comparison of effect before and after median filtering of road crack image

The dataset after median filtering and the training results of YOLOv5 are shown in Fig. 7. It is evident that median filtering significantly enhances the training performance of YOLOv5s. All four metrics show marked improvement: Precision increases from 0.381 to 0.816, Recall value rises from 0.370 to 0.753, the F1 value jumps from 0.375 to 0.873, and mAP climbs from 0.323 to 0.794. The primary reason for this improvement is the reduction of noise patterns in the training images and the increased number of training samples.



(a) Precision-confidence (b) Recall-confidence

(c) F1-confidence (d) Precision-recall

Figure 7: YOLOv5s training results after median filtering

However, there are still some limitations in the training effect of the dataset after median filtering. Therefore, this paper further enhances the YOLOv5s model by incorporating the CA mechanism [46].

The CA mechanism splits channel attention into two separate one-dimensional feature encoding processes, focusing on spatial coordinates and attentional focus. By capturing long-range dependencies along one spatial axis while retaining precise location information along another, it improves the accuracy of crack detection and the generalization capability of the model. The schematic of the CA mechanism is shown in Fig. 8. For the height coordinate operation, the output in the height direction can be expressed as:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i < W} x_c(h, i) \tag{5}$$

As for the width coordinate operation, the output in the width direction can be expressed as:

$$z_c^\omega(\omega) = \frac{1}{H} \sum_{0 \le i < H} x_c(j, \omega) \tag{6}$$

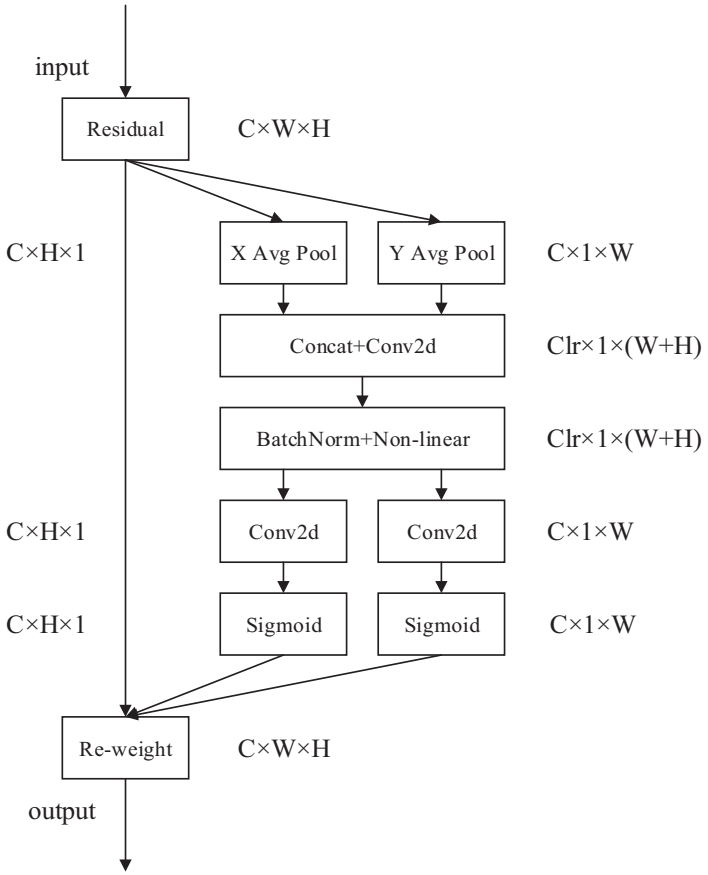


Figure 8: The schematic of CA attention mechanism

The transformation along the vertical and horizontal axes independently merges features across two distinct spatial dimensions, resulting in a pair of directional sensing feature maps. Transforming the vertical and horizontal dimensions also enables the attention module to capture long-range dependencies along a single spatial direction, while concurrently maintaining exact spatial location details along the other. This enhances the network's precision in target localization. The CA attention mechanism concatenates the transformation of the vertical and horizontal dimensions and applies a convolution function for the

transformation operation. It then decomposes the transformation along the dimension, segregating into two distinct components. The calculation process is as follows:

$$f = \delta\left(F_1\left(\left[z^h, z^\omega\right]\right)\right) \tag{7}$$

$$g^h = \sigma\left(F_h\left(f^h\right)\right) \tag{8}$$

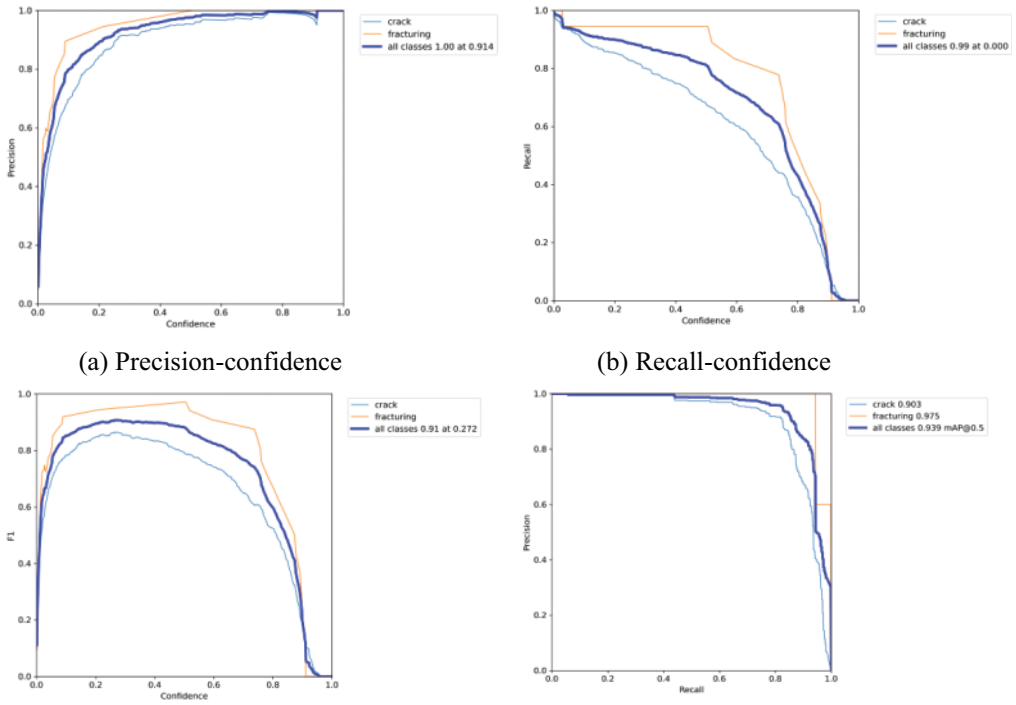
$$g^\omega = \sigma\left(F_\omega\left(f^\omega\right)\right) \tag{9}$$

The final output of the CA attention mechanism is obtained:

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^\omega(j) \tag{10}$$

The attention mechanism can be incorporated into various locations within the YOLOv5s network. For instance, it can be added to every CBL module to implement a full network attention mechanism. Alternatively, it can be integrated into the final layer of the backbone network. To reduce computational load, this paper adds the attention mechanism to the last layer of the backbone network.

The training results of YOLOv5s with CA after median filtering are shown in Fig. 9. It can be observed that integrating the CA attention mechanism further improves the performance of YOLOv5s. As shown in Table 3, the Precision increased from 0.816 to 0.889, the Recall increased from 0.753 to 0.861, the F1 increased from 0.783 to 0.875, and the mAP value increased from 0.794 to 0.920. Although the improvement from the CA attention mechanism is not as substantial as that from median filtering, it still effectively enhances the training outcomes. The results of the three experimental groups are summarized in the table below.



(a) Precision-confidence (b) Recall-confidence

(c) F1-confidence (d) Precision-recall

Figure 9: YOLOv5s-CA training results after median filtering

Table 3: YOLOv5 training results with CA attention mechanism

Dataset	Precision	Recall	F1	mAP
Before median filtering	0.381	0.370	0.375	0.323
After median filtering	0.816	0.753	0.783	0.794
YOLOv5s-CA and median filtering	0.889	0.861	0.875	0.920

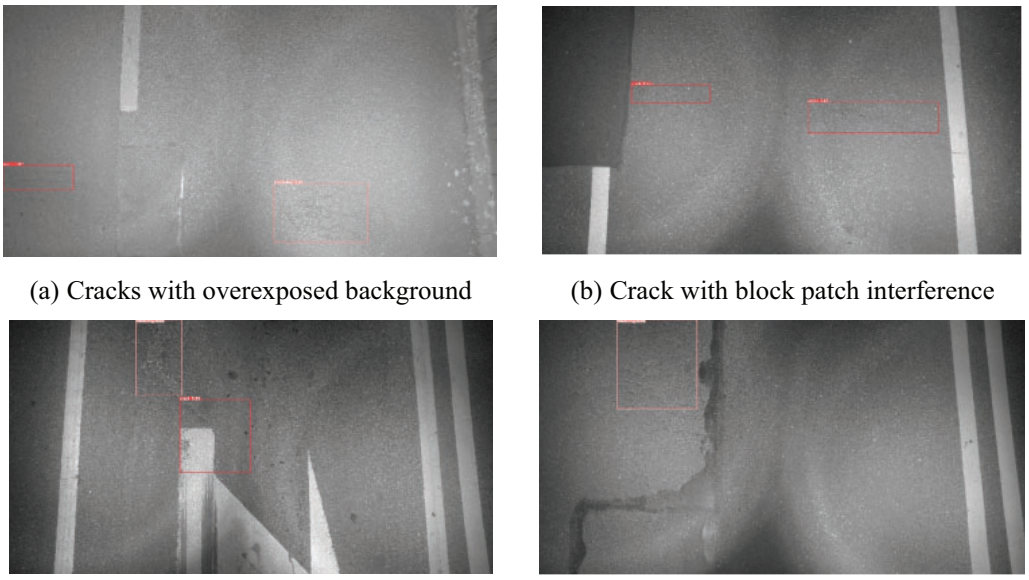
4 Model Visualization Test

The head part of YOLOv5s contains three detection heads, which scan the entire image at once and output several candidate boxes. Each candidate box has a confidence value, indicating the likelihood of a target being present in that region. This confidence value is a probability between 0 and 1. During detection, a fixed threshold screening method is first used to eliminate candidate boxes with confidence values below the threshold. To address the issue of multiple prediction boxes for the same object, the Non-Maximum Suppression (NMS) algorithm is employed to remove highly overlapping candidate boxes.

The NMS method selects prediction box B1 with the highest confidence as the baseline and removes all other prediction boxes whose Intersection over Union (IoU) values with B1 exceed the threshold. The second NMS method selects the prediction box B2 with the second highest confidence as the baseline and removes all other prediction boxes whose IoU values with B2 exceed the threshold. This process is repeated until all prediction boxes have been used as baselines. The IoU is an intersection ratio reflects the overlap of two rectangular boxes. The formula is as follows:

$$IoU = \frac{Box_A \bigcap Box_B}{Box_A \bigcup Box_B} \tag{11}$$

The process of target detection typically generates numerous highly overlapping frames for the same object. To reduce the number of prediction frames, NMS with overlap suppression is commonly used. When the overlap, measured by the IoU, exceeds the NMS threshold, redundant frames are removed, retaining only the most confident prediction. In this case, the YOLOv5s-CA model is also applicable for the detection task. The NMS threshold is set to 0.9, and the confidence threshold is established at 0.5. Fig. 10 below shows the crack detection results for typical pavement images. It can be seen that the YOLOv5s-CA model can detect various types of pavement cracks on different roads. For example, in Fig. 10a, the proposed model can detect cracks that are imperceptible to the human eye. In Fig. 10d, the proposed model can detect fracturing that is also imperceptible to the human eye. For pavement images with different brightness and texture characteristics, the model can detect cracks effectively. The proposed model can better detect actual pavement cracks and has potential practical engineering application value. Moreover, the proposed model eliminates many noise patterns, such as the white road marker lines and shadows in all the sub-images, and the patch mark in Fig. 10b,d. However, it is noted that the proposed model might face challenges when the background of the pavement images is complicated.



(a) Cracks with overexposed background

(b) Crack with block patch interference

(c) Crack with marking line interference

(d) Crack with line patch interference

Figure 10: Typical examples of crack detection with YOLOv5s-CA

5 Conclusions

This study introduces a novel approach that combines deep learning and image processing techniques to detect early-stage pavement cracks characterized by low contrast. The raw crack images were preprocessed using median filtering, and the YOLOv5s network was enhanced by incorporating the CA attention mechanism. This research provides a reference for identifying fine-width and low-contrast pavement cracks. The main findings of this investigation are as follows:

(i) To address the scarcity of road crack datasets, a collection of 10,000 raw pavement images with a resolution of 4000 × 2000 was assembled. Among them, 1400 crack images were selected and annotated. The resulting crack dataset encompasses images featuring diverse background conditions, such as varying brightness levels, water stains, and shadows.

(ii) When trained on the dataset of raw images, YOLOv5s produced suboptimal detection results, with a mAP value of just 0.323. However, by applying median filtering to remove noise from the original images and increase image numbers, the YOLOv5s achieved significantly improved results, with a mAP value of 0.794 and an F1 score of 0.783. This indicates that training directly on annotated sample images (early pavement crack images with low-contrast features) may lead to unsatisfactory outcomes. Therefore, image processing techniques can be effective in addressing this challenge.

(iii) In addition to improving the crack detection capacity, the YOLOv5s model was enhanced by incorporating the CA attention mechanism. This modification led to further performance improvements, resulting in a mAP value of 0.920 and an F1 score of 0.875.

Acknowledgement: Not applicable.

Funding Statement: The work described in this paper was jointly supported by the National Natural Science Foundation of China (No. 52308332), and the China Postdoctoral Science Foundation (Grant No. 2022M712787).

Author Contributions: Conceptualization: Tao Jin, Siqi Gu and Zhekun Shou. Methodology: Tao Jin and Siqi Gu. Validation: Zhekun Shou, Hong Shi and Min Zhang. Investigation: Tao Jin, Siqi Gu, Zhekun Shou and Hong Shi. Data curation: Siqi Gu, Zhekun Shou, Hong Shi and Min Zhang. Writing-original draft preparation: Tao Jin and Siqi Gu. Writing—review and editing: Tao Jin, Siqi Gu and Zhekun Shou. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data used or analyzed during the current study are available from the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Abdel-Qader I, Abudayyeh O, Kelly ME. Analysis of edge-detection techniques for crack identification in bridges. J Comput Civ Eng. 2003;17(4):255–63. doi:10.1061/(ASCE)0887-3801(2003)17:4(255).
2. Zhang H, Li JJ, Kang F, Zhang JA. Monitoring depth and width of cracks in underwater concrete structures using embedded smart aggregates. Measurement. 2022;204(4):112078. doi:10.1016/j.measurement.2022.112078.
3. Deng Y, Gui JY, Zhang HX, Taliercio A, Zhang P, Wong SHF, et al. Study on crack width and crack resistance of eccentrically tensioned steel-reinforced concrete members prestressed by CFRP tendons. Eng Struct. 2022;252:113651. doi:10.1016/j.engstruct.2022.113651.
4. Zhang HD, Chen YQ, Liu B, Guan XP, Le XY. Soft matching network with application to defect inspection. Knowl-Based Syst. 2021;225(1):107045. doi:10.1016/j.knosys.2021.107045.
5. Tang YC, Huang ZF, Chen Z, Chen MY, Zhou H, Zhang HX, et al. Novel visual crack width measurement based on backbone double-scale features for improved detection automation. Eng Struct. 2023;274:115158. doi:10.1016/j.engstruct.2023.115158.
6. Shibano K, Morozova N, Shimamoto Y, Alver N, Suzuki T. Improvement of crack detectivity for noisy concrete surface by machine learning methods and infrared images. Case Stud Constr Mater. 2024;20:e02984. doi:10.1016/j.cscm.2024.102984.
7. Yu TT, Zhu AX, Chen YY. Efficient crack detection method for tunnel lining surface cracks based on infrared images. J Comput Civ Eng. 2017;31(3):04016067. doi:10.1061/(ASCE)CP.1943-5487.0000645.
8. Wang KCP, Li Q, Gong WG. Wavelet-based pavement distress image edge detection with a trous algorithm. Transp Res Rec. 2007;2024(1):73–81. doi:10.3141/2024-09.
9. Goh TY, Basah SN, Yazid H, Safar MJA, Saad FSA. Performance analysis of image thresholding: otsu technique. Measurement. 2018;114(1):298–307. doi:10.1016/j.measurement.2017.10.036.
10. Yuan WZ, Yang Q. Identification of asphalt pavement transverse cracking based on 2D reconstruction of vehicle vibration signal and edge detection algorithm. Constr Build Mater. 2023;408(4):133788. doi:10.1016/j.conbuildmat.2023.133788.
11. Ying L, Salari E. Beamlet transform-based technique for pavement crack detection and classification. Comput Aided Civ Infrastruct Eng. 2010;25(8):572–80. doi:10.1111/j.1467-8667.2010.00674.x.
12. Tsai YC, Kaul V, Mersereau RM. Critical assessment of pavement distress segmentation methods. J Transp Eng. 2010;136(1):11–9. doi:10.1061/(ASCE)TE.1943-5436.0000051.
13. Nguyen TS, Begot S, Duculty F, Avils M. Free-form anisotropy: a new method for crack detection on pavement surface images. In: Proceedings of the 2011 18th IEEE International Conference on Image Processing (ICIP); 2011 Sep 11–14; Brussels, Belgium. p. 1069–72. doi:10.1109/ICIP.2011.6115610.
14. Amhaz R, Chambon S, Idier J, Baltazart V. Automatic crack detection on two-dimensional pavement images: an algorithm based on minimal path selection. IEEE Trans Intell Transp Syst. 2016;17(10):2718–29. doi:10.1109/TITS.2015.2477675.
15. Yeum CM, Dyke SJ. Vision-based automated crack detection for bridge inspection. Comput Aided Civ Infrastruct Eng. 2015;30(10):759–70. doi:10.1111/mice.12141.

16. Hinton GE, Osindero S, Teh YW. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18(7):1527–54. doi:10.1521/neco.2006.18.7.1527.
17. Zhang L, Yang F, Zhang YD, Zhu YJ. Road crack detection using deep convolutional neural network. In: Proceeding IEEE International Conference on Image Processing (ICIP); 2016 Sep 25–28; Phoenix, AZ, USA. p. 3708–12. doi:10.1109/ICIP.2016.7533052.
18. Chen YP, Dai XY, Liu MC, Chen DD, Yuan L, Liu ZC. Dynamic ReLU. In: Proceedings of the 16th European Conference on Computer Vision (ECCV); 2020 Aug 23–28; Glasgow, UK; 2020. p. 351–67.
19. Jiang S, Zhang J. Real-time crack assessment using deep neural networks with wall-climbing unmanned aerial system. Comput Aided Civ Infrastruct Eng. 2020;35(6):549–64. doi:10.1111/mice.12519.
20. Bouvrie J. Notes on convolutional neural networks. [cited 2025 Jan 1]. Available from: http://cogprints.org/5869/.
21. Geetha GK, Sim SH. Fast identification of concrete cracks using 1D deep learning and explainable artificial intelligence-based analysis. Autom Constr. 2022;143(11):104572. doi:10.1016/j.autcon.2022.104572.
22. Yang F, Zhang L, Yu SJ, Prokhorov D, Mei X, Ling HB. Feature pyramid and hierarchical boosting network for pavement crack detection. IEEE Trans Intell Transp Syst. 2019;21(4):1525–35. doi:10.1109/TITS.2019.2910595.
23. Lu YX, Zhang GY, Duan SK, Chen F. A pyramid auxiliary supervised U-Net model for road crack detection with dual-attention mechanism. Displays. 2024;84(2):102787. doi:10.1016/j.displa.2024.102787.
24. Cha YJ, Choi W, Büyüköztürk O. Deep learning-based crack damage detection using convolutional neural networks. Comput Aided Civ Infrastruct Eng. 2017;32(5):361–78. doi:10.1111/mice.12263.
25. Saidani T. Deep learning approach: YOLOv5-based custom object detection. Eng Technol Appl Sci Res. 2023;13(6):12158–63. doi:10.48084/etasr.6397.
26. Khanam R, Asghar T, Hussain M. Comparative performance evaluation of YOLOv5, YOLOv8, and YOLOv11 for solar panel defect detection. Solar. 2025;5(1):6. doi:10.3390/solar5010006.
27. Zhou SX, Yang D, Zhang ZY, Zhang JW, Qu FL, Punetha P, et al. Enhancing autonomous pavement crack detection: optimizing YOLOv5 algorithm with advanced deep learning techniques. Measurement. 2025;240(3):115603. doi:10.1016/j.measurement.2024.115603.
28. Liu PF, Wang Q, Zhang H, Mi J, Liu YC. A lightweight object detection algorithm for remote sensing images based on attention mechanism and YOLOv5s. Remote Sens. 2023;15(9):2429. doi:10.3390/rs15092429.
29. Fan LL, Li S, Li Y, Li B, Cao DP, Wang FY. Pavement cracks coupled with shadows: a new shadow-crack dataset and a shadow-removal-oriented crack detection approach. IEEE/CAA J Autom Sinica. 2023;10(7):1593–607. doi:10.1109/JAS.2023.123447.
30. Liu LQ, Shen B, Huang SC, Liu RL, Liao WZ, Wang B, et al. Binocular video-based automatic pixel-level crack detection and quantification using deep convolutional neural networks for concrete structures. Buildings. 2025;15(2):258. doi:10.3390/buildings15020258.
31. Kang D, Benipal SS, Gopal DL, Cha YJ. Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning. Autom Constr. 2020;118(4):103291. doi:10.1016/j.autcon.2020.103291.
32. Peng X, Zhong XG, Zhao C, Chen AH, Zhang TY. A UAV-based machine vision method for bridge crack recognition and width quantification through hybrid feature learning. Constr Build Mater. 2021;299(5):123896. doi:10.1016/j.conbuildmat.2021.123896.
33. Guo PW, Meng WN, Bao Y. Automatic identification and quantification of dense microcracks in high-performance fiber-reinforced cementitious composites through deep learning-based computer vision. Cem Concr Res. 2021;148(1):106532. doi:10.1016/j.cemconres.2021.106532.
34. Fu Y, Yin HL, Chen TY, Chen ZB. Long-distance measurement-device-independent multiparty quantum communication. Phys Rev Lett. 2015;114(9):090501. doi:10.1103/PhysRevLett.114.090501.
35. Shi Y, Cui LM, Qi ZQ, Meng F, Chen ZS. Automatic road crack detection using random structured forests. IEEE Trans Intell Transp Syst. 2016;17(12):3434–45. doi:10.1109/TITS.2016.2552248.
36. Zou Q, Zhang Z, Li QQ, Qi XB, Wang Q, Wang S. DeepCrack: learning hierarchical convolutional features for crack detection. IEEE Trans Image Process. 2019;28(3):1498–512. doi:10.1109/TIP.2018.2878966.
37. Maharana K, Mondal S, Nemade B. A review: data pre-processing and data augmentation techniques. Global Transitions Proc. 2022;3(1):91–9. doi:10.1016/j.gltp.2022.04.020.

38. Wang CY, Liao H, Wu YH, Chen PY, Hsieh JW, Yeh IH. 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 (CVPR Workshops); 2020 Jun 14–19; Seattle, WA, USA. p. 390–1.
39. He KM, Zhang XY, Ren SQ, Sun J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell. 2015;37(9):1904–16. doi:10.1109/TPAMI.2015.2389824.
40. Liu S, Qi L, Qin HF, Shi JP, Jia JY. Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2018 Jun 18–23; Salt Lake City, UT, USA. p. 8759–68.
41. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. p. 2117–25.
42. Sharma S, Ravi H, Subramanyam AV, Emmanuel S. Anti-forensics of median filtering and contrast enhancement. J Vis Commun Image Represent. 2020;66(6):102682. doi:10.1016/j.jvcir.2019.102682.
43. Hou Q, Zhou D, Feng J. Coordinate attention for efficient mobile network design. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 19–25; Nashville, TN, USA. p. 13713–22.
44. Neubeck A, Van GL. Efficient non-maximum suppression. In: Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06); 2006 Aug 20–24; Hong Kong, China. p. 850–5.
45. Cai Zw, Vasconcelos N. Cascade R-CNN: delving into high quality object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2018 Jun 18–22; Salt Lake City, UT, USA. p. 6154–62.
46. Guo GG, Zhang ZY. Road damage detection algorithm for improved YOLOv5. Sci Rep. 2022;12(1):15523. doi:10.1038/s41598-022-19674-8.