Road Surface Crack Detection Based on Improved YOLOv9 Image Processing

Quanwu Li*, Shaopeng Duan
School of Information Engineering and Artificial Intelligence, Zhengzhou Vocational University of Information and Technology, Zhengzhou 450008, China

***Abstract*—Road surface crack detection is a critical task in road maintenance and safety management. Cracks in road surfaces are often the early indicators of larger structural issues, and if not detected and repaired in time, they can lead to more severe deterioration and increased maintenance costs. Effective and timely crack detection is essential to prolong road lifespan and ensure the safety of road users. This paper introduces CrackNet, an advanced crack detection model built upon the YOLOv9 architecture, which integrates a fusion attention module and task space disentanglement to enhance the accuracy and efficiency of road surface crack detection. Traditional methods often struggle with the complex and irregular nature of road cracks, as well as the challenge of distinguishing cracks from their backgrounds. CrackNet overcomes these challenges by leveraging an attention mechanism that highlights relevant features in both the channel and spatial dimensions while separating the tasks of classification and regression. This approach significantly reduces false negatives and improves localization accuracy. The effectiveness of CrackNet is validated through comparative analysis with other segmentation models, including Unet, SOLO v2, Mask R-CNN, and Deeplab v3+. CrackNet consistently outperforms these models in terms of F1 and Jaccard coefficients. This study highlights the critical role of accurate crack detection in minimizing maintenance costs and enhancing road safety.**

***Keywords*—Road crack; YOLOv9; deep learning; surveillance**

I. Introduction

Road surface damage refers to the occurrence of deterioration, cracks, and potholes in the road surface layer, which are major factors affecting road performance. Therefore, timely and accurate detection of road surface damage is a crucial aspect of road maintenance. Cracks are the initial manifestation of various types of road surface diseases, making the detection and repair of road cracks particularly important [1-3]. Not only do road surface cracks affect the appearance and comfort of driving, but if they are not repaired promptly, they can widen and worsen, leading to structural damage and reducing the overall performance and lifespan of the road. Thus, early detection and timely repair of cracked roads not only reduce the economic cost of road repairs but also ensure the safety of vehicles and drivers on the road. Moreover, with the increasing number of traffic accidents, road safety has become a global challenge. Therefore, the detection and repair of road surface cracks should be prioritized to ensure road safety and longevity [4-6].

Historically, road surface detection and maintenance primarily relied on manual inspection, which was not only time-consuming and labor-intensive but also low in accuracy and fraught with risks. Scholars around the world have utilized the latest scientific and technological advancements to conduct extensive and in-depth studies to accurately and effectively extract crack information from images. In 2014, Wang et al. [7] proposed a crack extraction method based on the valley boundary, employing a series of image processing algorithms to achieve crack detection results. In 2015, Liang et al. [8] introduced a road crack connection algorithm based on Prim's minimum spanning tree, which involves filling cracks to create a structured crack map.

However, these traditional methods of crack detection have obvious disadvantages. Each method is designed for specific databases or scenarios, and the crack detectors fail if there are changes in the dataset or scenario. This highlights a significant gap between conventional methods and current demands in real-world applications, where crack patterns, lighting conditions, and environmental variations pose challenges for traditional models. In particular, many existing models struggle with detecting fine or irregular cracks under diverse conditions, which leads to false positives or missed detections. This gap emphasizes the need for more robust, adaptable models that can address these challenges and ensure accurate detection in real-world settings.

In recent years, deep learning methods have been increasingly applied to road crack detection and segmentation, integrating deep learning techniques with road crack detection technologies, significantly enhancing the efficiency and accuracy of road crack detection. Lee et al. [9] researched a CNN-based road-surface crack detection model that responds to changes in brightness. They discovered that a preprocessing model, which adjusts the image brightness before inputting it into the crack detection model, enhances the consistency of road-surface crack detection, maintaining stable performance under varying brightness conditions.

Hammouch et al. [10] and colleagues studied an automated methodology for crack detection and classification in Moroccan flexible pavements using Convolutional Neural Networks (CNN). They found that good crack detection and classification are achieved on the dataset using both the CNN and a pre-trained Visual Geometry Group 19 (VGG-19) model. However, the accurate identification of road cracks remains a challenging issue due to their high similarity to the background, small size, and irregular shape in real-world scenarios. Enhancing the precision and timeliness of image-based crack extraction has become a focal point of current research. Originally utilized in

*Corresponding Author

the medical field for cell segmentation, YOLO networks handle complex noise interference better than road surface cracks and can extract both high and low-level features from objects. Despite their simplicity and high accuracy in cell segmentation, these networks have limitations in shallow feature extraction layers and the introduction of irrelevant features through the fusion of different feature levels. Consequently, this paper proposes an improved YOLO method for crack detection. During the feature extraction phase, a Fusion Attention Module (FAM) is embedded, which applies non-uniform weighting across channel and spatial dimensions to highlight useful information. Additionally, a Task-Aware Spatial Disentanglement Head (TSDHead) decouples classification and regression tasks, effectively addressing issues of crack misdetection and inaccurate localization, thus ensuring real-time detection while enhancing the accuracy of road crack detection.

At the end of this introduction, the structure of the paper is outlined as follows: Section II provides a detailed explanation of the improved YOLOv9 architecture and the modifications made to enhance crack detection accuracy. Section III describes the experimental setup, including the dataset used and the evaluation metrics employed to assess the model's performance. Section IV presents the results and a comparative analysis with other segmentation models, highlighting the strengths of CrackNet. Finally, Section V concludes the paper with a discussion on the potential applications of the model in real-world road maintenance systems and suggestions for future research directions. This structure is designed to guide readers through the study and provide a clear understanding of the proposed methodology and its practical implications.

II. Improved YOLOv9 Model

In this study, we conducted a comparative analysis of several segmentation models (including Unet, SOLO v2, Mask R-CNN, and Deeplab v3+) in the context of road crack detection tasks. The strengths and weaknesses of each model are summarized as follows:

Unet: The U-shaped architecture of Unet allows it to perform well on smaller datasets and enables end-to-end training, making it suitable for crack detection tasks. However, due to its reliance on a symmetrical encoder-decoder structure, Unet may suffer from over-segmentation or under-segmentation when detecting complex crack patterns, particularly in cases where crack boundaries are unclear.

SOLO v2: As an instance segmentation model, SOLO v2 transforms the segmentation task into a pixel classification problem, eliminating the need for proposal generation. As a result, it can deliver good segmentation results in crack detection, especially in complex background scenarios. However, SOLO v2 still struggles with accurately detecting fine and low-contrast cracks.

Mask R-CNN: Mask R-CNN uses a Region Proposal Network (RPN) to precisely localize crack regions and generate instance masks. This makes it highly accurate for detecting wide cracks. However, it comes with a high computational cost and tends to over-segment or miss finer cracks during detection.

Deeplab v3+: Combining atrous convolution with an encoder-decoder structure, Deeplab v3+ is well-suited for extracting features at large scales and handling crack images with complex backgrounds. However, its ability to recover fine details is limited, resulting in less effective performance when detecting small cracks compared to other models.

In contrast, CrackNet introduces a Fusion Attention Module and Task Space Disentanglement mechanism, which effectively enhances crack feature extraction, reduces false detections, and improves the localization of cracks. This is especially true when handling fine and irregular cracks. Therefore, CrackNet consistently outperforms the aforementioned models in terms of F1 and Jaccard coefficients, demonstrating superior overall performance [11-13].

A. YOLOv9 Model

The YOLO network has undergone multiple iterations to overcome the limitations of previous versions and enhance performance, achieving a good balance between speed and accuracy. The latest version, YOLOv9-Seg, comprises three components: Backbone, Neck, and Head, as illustrated in Fig. 1. The Backbone extracts features from the input image, while the Neck further processes these features and integrates information across different levels. Finally, the Head layer and subsequent post-processing steps generate the classification, location, and pixel segmentation results of detected objects.

The Backbone of YOLOv9-Seg is composed of three key modules: Conv, ADown, and RepNCSPELAN4. The Conv module, a standard component in convolutional neural networks (CNNs), utilizes convolution, batch normalization, and activation functions to extract features from the input image. The ADown module applies pooling operations to downsample the feature matrix. The RepNCSPELAN4 module plays a critical role in the YOLOv9-Seg network by segmenting and merging the feature matrix through layer aggregation, thereby reducing redundant computations and enhancing feature extraction efficiency.

The Neck component consists of a feature pyramid structure that integrates a Feature Pyramid Network (FPN) and a Path Aggregation Network (PAN). Lower-level convolutional features, despite having less semantic information and more noise, offer higher resolution and more detailed location data. On the other hand, higher-level features provide richer semantic information but compromise resolution and detail. FPN combines high-level and low-level features through a top-down upsampling approach, creating feature maps that are rich in semantic information. PAN further enhances the location accuracy across levels by propagating location information from the bottom to the top, strengthening the overall feature pyramid based on FPN.

SPPELAN combines the advantages of Spatial Pyramid Pooling Fast (SPPF) and Efficient Local Aggregation Network (ELAN). SPPF captures spatial information across multiple scales, improving the model's robustness, while ELAN is a lightweight network structure that enhances feature extraction through local aggregation and global integration. The combination of SPPF and ELAN further boosts feature extraction capabilities.

The head network includes three segment detectors, each operating on feature matrices at different scales to locate and

segment target objects, improving detection performance through multi-scale integration. In our implementation, we utilized pre-trained weights on a large dataset for transfer learning. Furthermore, we incorporated Programmable Gradient Information (PGI) and Generalized Efficient Layer Aggregation Network (GELAN) architectures to optimize both the model's performance and efficiency, as shown in Fig. 1.

B. Fusion Attention Module

Drawing from the design concepts of FcaNet (Frequency Channel Attention Networks) and CBAM (Convolutional Block Attention Module), this paper introduces the Fusion Attention Module (FAM). This module comprises a multispectral channel attention module and a spatial attention module, as illustrated in Fig. 2. The multispectral channel attention module, based on the multispectral frequency information of feature maps, adaptively models the importance of each channel, highlighting significant channel features. The spatial attention module focuses on areas within the feature map rich in detail, addressing the loss of some spatial information in the multispectral channel attention module [14, 15]. By chaining these two modules, the design retains both critical channel features and detailed features around the cracks.

1) Multispectral channel attention module: As shown in Fig. 2, the multispectral channel attention module utilizes the Discrete Cosine Transform (DCT) to extract multispectral frequency features from the feature maps. These features are used to model the importance of each channel, and subsequently, different weights are assigned to these channels to implement an attention mechanism along the channel dimensions. This paper employs 2D-DCT technology to transform the feature extraction process for different channels into a feature compression process using multispectral frequency components. Specifically, 2D-DCT maps time-domain signals from the spatial domain to the frequency domain, transforming energy dispersion in the time domain into relatively concentrated energy forms in the frequency domain.

The specific data processing flow of the multispectral channel attention module is as follows:

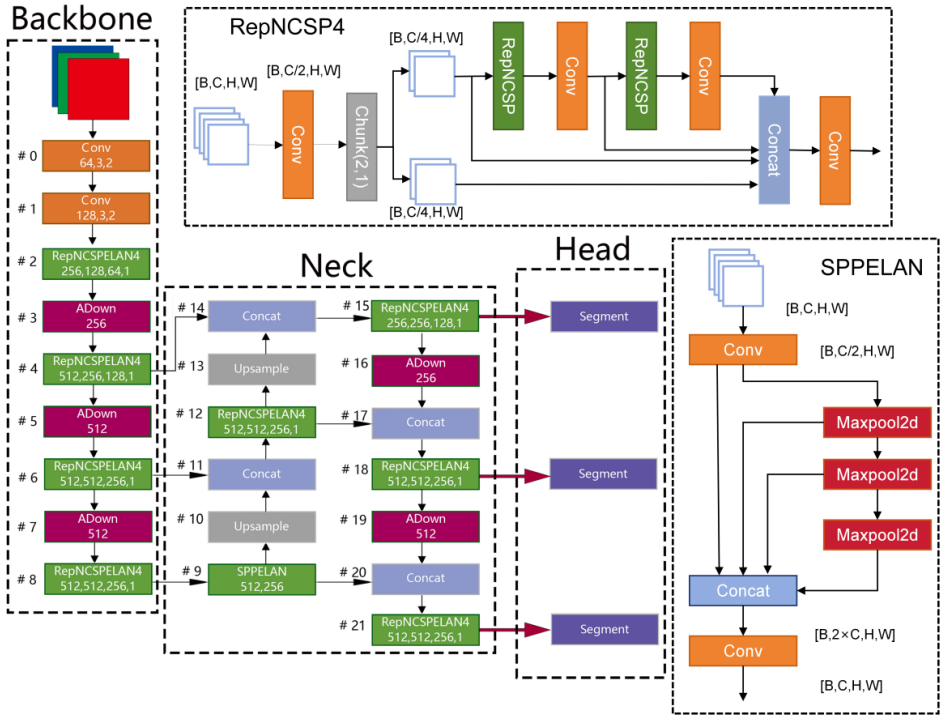


Fig. 1. Architecture of YOLOv9.

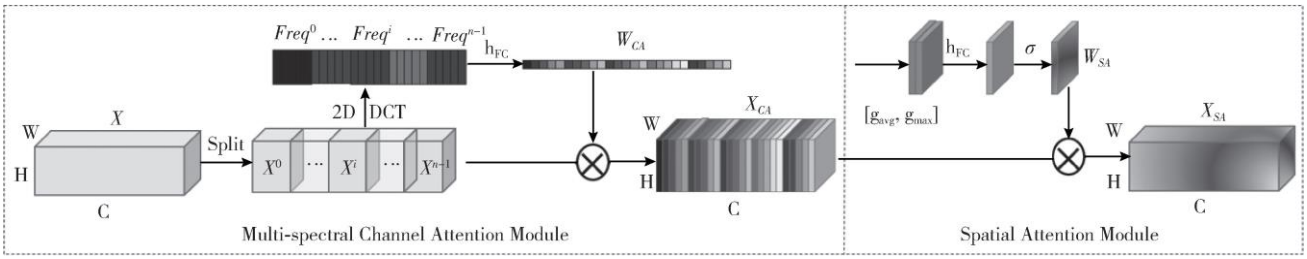


Fig. 2. Structure of the fusion attention module.

Step 1: As shown in Fig. 2, the input feature map $X \in R^{C\times H\times W}$ is split along the channel dimension into n feature map blocks $X^i \in R^{\frac{C}{n}\times H\times W}$.

Step 2: For each feature map block Xi, utilize the two-dimensional Discrete Cosine Transform (2D-DCT) to extract the corresponding multispectral frequency information, obtaining the feature vector $\mathrm{Freq}^i \in R^{\frac{\mathrm{C}}{\mathrm{n}}\times 16}$ as described by Eq. (1) and Eq. (2). Here, H and W represent the height and width of the feature map, respectively, and u, v are the two-dimensional indices for the feature map block Xi.

$$\mathrm{Freq}^i = 2DDCT^{\mathrm{u,v}}\left(X^i\right) = \sum_{h=0}^{\mathrm{H}-1}\sum_{w=0}^{\mathrm{W}-1} X^i_{:,h,w} B^{\mathrm{u,v}}_{h,w} \tag{1}$$

$$\mathrm{B}^{u,v}_{h,w} = \cos\left(\frac{h}{\mathrm{H}}\left(\mathrm{u}+\frac{1}{2}\right)\right)\cos\left(\frac{w}{\mathrm{W}}\left(\mathrm{v}+\frac{1}{2}\right)\right) \tag{2}$$

Step 3: Concatenate the feature vector Freqi along the channel dimension to form the multispectral frequency information Freq∈R C×16. As indicated in Eq. (3), after a fully connected and Sigmoid operation, the channel weight matrix WCA∈R 1×1×C is obtained. This matrix is then multiplied by the input feature map to produce the channel-attention-weighted feature map XCA.

$$W_{CA} = \sigma\left(h_{\mathrm{RC}}(\mathrm{Freq})\right) \tag{3}$$

Eq. (3) where σ represents the Sigmoid function and hRC denotes the fully connected operation.

2) Spatial attention module: As depicted on the right side of Fig. 2, the spatial attention module focuses on the cracks and their surrounding areas within the image based on the input feature map. Specifically, according to Eq. (4), the feature map XCA undergoes both max pooling and average pooling. Subsequently, these pooled features are concatenated along the channel dimension, and a fully connected operation is used to generate the spatial weight matrix WSA ∈ RH×W×1. This matrix is then multiplied by the feature map XCA to produce the output feature map XSA.

$$W_{SA} = \sigma\left(h_{\mathrm{FC}}\left(\left[\mathrm{g}_{\mathrm{avg}}\left(X_{CA}\right), \mathrm{g}_{\mathrm{max}}\left(X_{CA}\right)\right]\right)\right) \tag{4}$$

Eq. (4) where hFC represents the fully connected operation, gavg and gmax denote average pooling and max pooling, respectively, and σ is the Sigmoid function.

C. Task Space

In deep learning, classification tasks focus on capturing the overall information of the target, while regression tasks are more dependent on the edges and finer details of the object. Drawing inspiration from TSD and RetinaNet, this paper introduces a Task-Space Disentanglement Head (TSDHead), which separates the classification and regression tasks during the multi-dimensional prediction phase. This decoupling allows the model to optimize each task independently, without the need to balance between them. The classification branch optimizes weights by following the steepest descent of the classification loss gradient, while the regression branch optimizes in a similar manner for its own specific task.

As shown in Fig. 3, the TSDHead processes the fused feature map to perform classification and regression predictions, and then inputs the results into the Non-Maximum Suppression (NMS) module for further processing. Specifically, the TSDHead comprises both a classification and a regression branch. The classification branch, used for predicting object categories, includes four structurally identical depthwise separable convolutional layers and one category prediction layer. Each depthwise separable convolution layer consists of a depthwise convolution layer (kernel size of 3×3 and the same number of channels as the input feature map) and a pointwise convolution layer (kernel size of 1×1, with 256 channels). The category prediction layer uses a kernel size of 3×3 and a channel count of K×A, where K represents the number of categories (set to 5 in this paper, including four types of cracks and background) and A represents the number of preset anchor boxes per spatial position on the feature map, set to 3. The structure of the regression branch mirrors the classification branch, except that the K in the prediction layer of the regression branch is 4, indicating the offsets for the center position and dimensions of the bounding boxes. After processing through the TSDHead, the feature map yields the crack categories and bounding box coordinates, which are refined by the NMS module to produce the final prediction results.

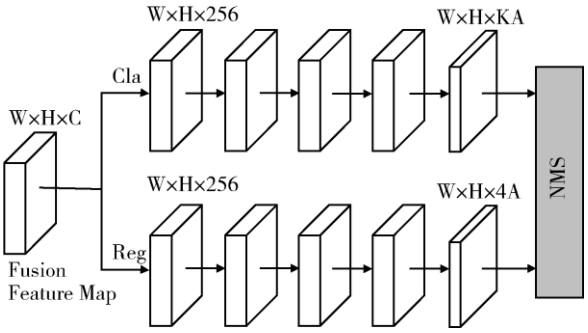


Fig. 3. Structure of the task-space disentanglement head.

1) K-means clustering of crack anchor box sizes: To address the diversity of road crack shapes and extreme aspect ratios, this paper employs the K-means clustering algorithm to cluster the sizes of bounding boxes in a constructed road crack dataset. Following the design philosophy of YOLOv5, the paper clusters large, medium, and small target sizes at three down sampling scales (8x, 16x, and 32x). Each down sampling scale is preset with three anchor boxes, with the clustering results presented in Table I.

TABLE I. CLUSTERING SIZES OF ANCHOR BOXES

	Scale_D32	**Scale_D16**	**Scale_D8**
Anchor_1	23,11	57,40	335,25
Anchor_2	76,8	212,13	115,89
Anchor_3	24,46	46,78	216,86

III. Experiments and Analysis

A Model Training and Testing Trials

1) Experimental data: The research dataset was constructed in two parts. The first part originated from the Crack500 dataset [11], where Yang and colleagues [11] used smart digital devices to capture 500 images of road surface cracks at Temple University with a resolution of 2000×1500, 24-bit RGB, creating the Crack500 dataset for crack detection. To enrich the experimental data, a photographic collection platform was established using smart digital devices to capture an additional 300 images of road surface cracks at a resolution of 1920×1080, 24-bit RGB, forming the second part. The digital devices used were equipped with three cameras, capable of capturing images up to 48 million pixels.

For ease of model training, a total of 800 images from both parts were cropped and filtered to produce 1600 images with a resolution of 320×320, 24-bit RGB. Of these, 1350 images were used as the training and validation set, which was randomly divided in a 9:1 ratio, and 250 images served as the test set. The test set was used solely for testing and did not participate in network training. Each collection was divided into two categories: fine narrow cracks and clearly visible wide cracks. Table I presents the number of each type of crack, and Fig. 4 provides examples of the crack images. The crack images were annotated using the LabelMe tool and further formatted into the VOC dataset structure.

TABLE II. Experimental Data

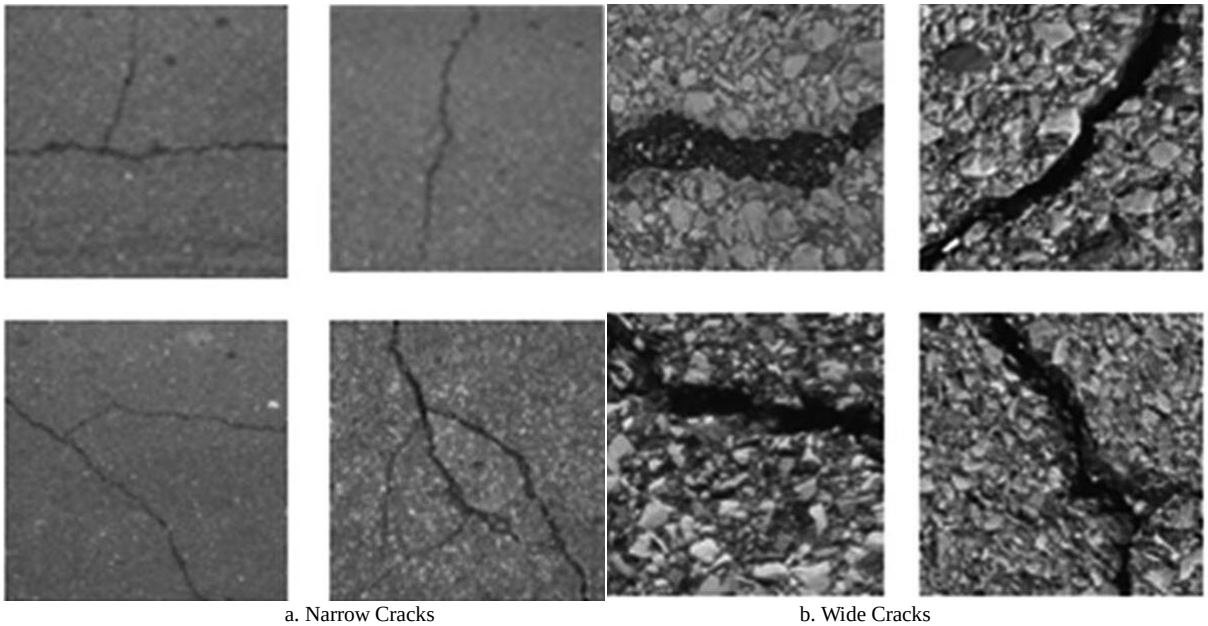
Training	Number of Images	Validation Set	Number of Images
Narrow Cracks	498	Narrow Cracks	91
Wide Cracks	852	Wide Cracks	159

2) Comparison of segmentation network models: This study compares the Improved YOLOv9 with Unet, SOLO v2 [12], Mask R-CNN (Mask Recycle Convolutional Neural Network), and Deeplab v3+ (Deep Convolutional Neural Networks v3 Plus) [13]. SOLO v2 and Mask R-CNN are instance segmentation algorithms, whereas Improved YOLOv9, Unet, and Deeplab v3+ are semantic segmentation models.

a) Unet: This network architecture features a clear U-shaped structure with symmetrical encoding on the left and decoding structures on the right, enhancing the extraction of feature map information. The Unet structure has low dependency on the number of images and can complete end-to-end training with only a small set of images, making it suitable for medical image segmentation.

b) Mask R-CNN: This network builds on the Faster Recycle Convolutional Neural Network (Faster R-CNN), adding a mask branch that runs in parallel with the classification and bounding box regression branches to predict segmentation masks. It employs a top-down, detection-based method that detects regions of each instance first and then segments the instance masks within these areas. Detection-based methods are generally highly accurate and rely on precise bounding box detection, which requires substantial computational resources [14].

c) SOLO v2: Unlike Mask R-CNN, this network transforms the segmentation task into a pixel classification problem, eliminating the need for proposal generation. The network has two branches: a category prediction branch that predicts the semantic category of the target, and a mask branch that predicts the instance mask of the target [15].



a. Narrow Cracks b. Wide Cracks

Fig. 4. Crack image of pavement.

d) Deeplab v3+: This network represents the latest generation of Deeplab models, using Deeplab v3 as the encoding structure and incorporating a decoder to address the loss of fine detail information caused by direct up-sampling of feature maps in Deeplab v3, thereby achieving advanced semantic segmentation performance.

3) Testing trial setup: After training, the models load the optimally saved weights from the training process to predict the test set, which consists of 250 images with a resolution of 320×320, 24-bit RGB. The test hardware platform includes an AMD Ryden 5 3600 CPU and NVIDIA GeForce GTX 2060 GPU, running on Windows 10 with Python version 3.6. Except for Mask R-CNN, which is tested using TensorFlow version 1.13, the other models are tested using PyTorch version 1.4. The test results are RGB three-channel images, which are then binarized and compared with the true images of the test set to compute evaluation metrics.

4) Evaluation metrics: To assess the segmentation performance of Improved YOLOv9 and the comparison models, this study employs the Jaccard coefficient and F1 score as evaluation metrics. Precision and recall are crucial parameters for binary classification problems and are important indicators of model segmentation performance. Calculations for precision and recall are provided in Eq. (5) and Eq. (6).

$$P = \frac{n_{\mathrm{TP}}}{n_{\mathrm{TP}} + n_{\mathrm{FP}}} \tag{5}$$

$$R = \frac{n_{\mathrm{TP}}}{n_{\mathrm{TP}} + n_{\mathrm{FN}}} \tag{6}$$

For a single image, segmenting the crack regions essentially means performing binary classification for each pixel, where nTP (true positives) represents pixels correctly identified as cracks, nFP (false positives) represents non-crack pixels predicted as cracks, nFN (false negatives) represents crack pixels predicted as non-cracks, and nTN (true negatives) represents non-crack pixels correctly identified as non-cracks.

A higher precision indicates a larger number of correctly identified crack pixels among those predicted as cracks by the model. Relying solely on either precision or recall to evaluate model performance is not advisable. For example, if all pixels in a test image are predicted as cracks, the recall would be 1, but the precision might be low.

Therefore, the harmonic mean of precision and recall, known as the F1 score, is used to measure model performance. The F1 score reflects the similarity between the predicted crack pixel set and the true crack pixel set. The F1 score ranges from 0 to 1, with higher values indicating better crack segmentation effectiveness. The calculation is shown in Eq. (7).

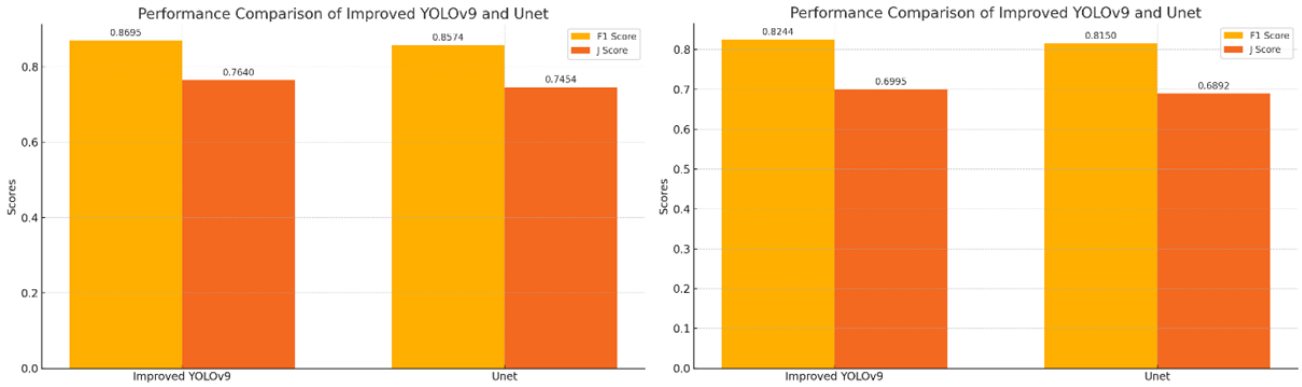
$$F_1 = \frac{2PR}{P + R} \tag{7}$$

The Jaccard coefficient measures the similarity between the predicted crack region and the actual crack region. It is calculated as the percentage of the intersection of the predicted and actual regions relative to the union of these regions. The value of the Jaccard coefficient ranges from 0 to 1, with higher values indicating a greater overlap between the predicted and actual areas, meaning that the predicted crack regions more closely match the actual regions. The calculation of the Jaccard coefficient is shown in Eq. (8).

$$J = \frac{n_{TP}}{n_{TP} + n_{FP} + n_{FN}} \tag{8}$$

B Experimental Results

1) Comparison of evaluation metrics between improved YOLOv9 and Unet models: This study first analyzes the enhancements made in Improved YOLOv9. The road surface crack test dataset includes two types of images: (1) fine narrow cracks with low contrast and narrow width, and (2) clear images of wider cracks. The F1 and Jaccard coefficients for Improved YOLOv9 and Unet under these two categories are shown in Fig. 5. As seen from Fig. 5, the metrics for Improved YOLOv9 are higher than those for Unet in both types of cracks, indicating that Improved YOLOv9 performs better in segmenting both narrow and wide cracks. Specifically, the F1 and Jaccard coefficients for narrow cracks are 4% to 6% lower than those for wide cracks, suggesting that crack width impacts the segmentation performance of the models.



a. Wide cracks b. Narrow cracks

Fig. 5. F1 and Jaccard coefficients of YOLOv9 and Unet.

Fig. 6 and Fig. 7 show the real and predicted segmentation results for narrow and wide cracks using Improved YOLOv9 and Unet models. The first column is the original image, the second column is the ground truth, and the third and fourth columns are the predictions from Unet and Improved YOLOv9, respectively, with white areas representing the crack regions.

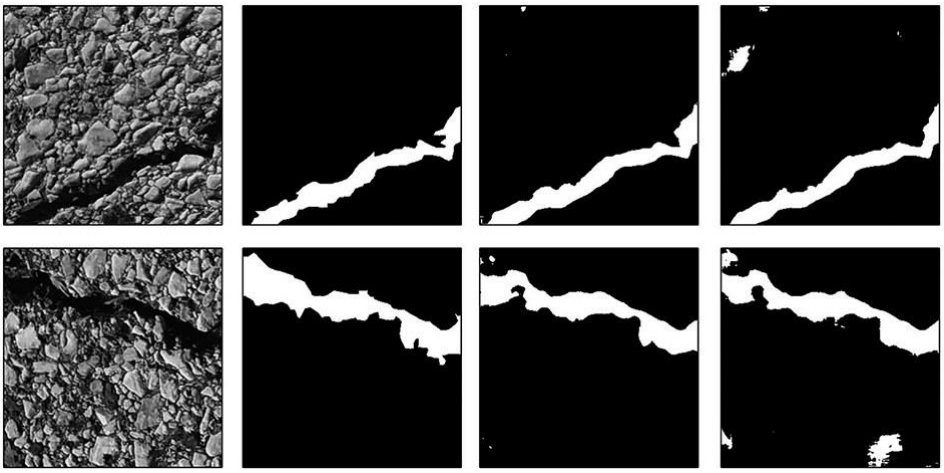


Fig. 6. Long and narrow crack segmentation result.

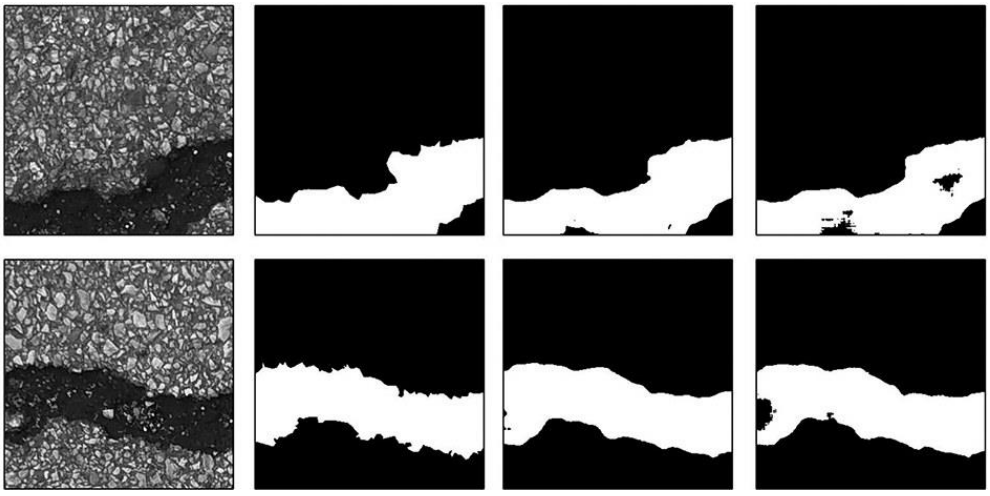


Fig. 7. Non-narrow crack segmentation result.

From the Figures, it is evident that Unet suffers from issues of over-segmentation and under-segmentation, particularly severe over-segmentation for narrow cracks (as shown in column 3 of Fig. 7) and under-segmentation for wide cracks (column 3 of Fig. 8). Compared to Unet, the Improved YOLOv9 proposed in this paper demonstrates better segmentation performance, with enhancements in feature extraction and the application of crack attention units contributing to more accurate crack image segmentation.

2) Real-time analysis of improved YOLOv9 and comparative models: This section of the study focuses on analyzing both the real-time performance and computational costs of the Improved YOLOv9 model compared to other segmentation models (Unet, SOLO v2, Mask R-CNN, and Deeplab v3+). The key metrics evaluated include single-frame image inference times, model complexity, and the balance between speed and accuracy.

As illustrated in Fig. 8, the inference time comparison shows that Improved YOLOv9 has a slightly longer inference time per frame (0.089 seconds) compared to Unet (0.084 seconds), but it offers significantly higher segmentation accuracy. This indicates that Improved YOLOv9 strikes an effective balance between speed and precision, which is essential for tasks requiring both real-time performance and high reliability, such as crack detection in road surfaces.

Other comparative models, such as SOLO v2 and Mask R-CNN, have considerably longer inference times (0.130 and 0.162 seconds per frame, respectively), making them less suitable for real-time applications where quick response is crucial. These models, while offering strong segmentation capabilities, suffer from higher computational costs and slower processing times, which could be a disadvantage in large-scale, real-time crack detection tasks.

Deeplab v3+ performs more closely to Improved YOLOv9, with an inference time of 0.093 seconds per frame. While this

model is competitive in terms of speed, it does not match the segmentation accuracy of Improved YOLOv9, especially in detecting fine and irregular cracks. Thus, for applications requiring both high accuracy and efficient real-time performance, Improved YOLOv9 proves to be the more optimal choice.

In terms of model complexity, the architectural advancements in Improved YOLOv9, such as the Fusion Attention Module and Task Space Disentanglement, contribute to its slight increase in computational cost compared to Unet. However, these enhancements also lead to more accurate feature extraction and better localization, particularly in complex road conditions. As a result, the minimal trade-off in processing time is justified by the superior detection performance in real-world applications.

This analysis highlights the strengths and weaknesses of each model regarding both real-time performance and computational efficiency. While Improved YOLOv9 may have a slightly higher computational cost compared to Unet, its improved accuracy and relatively low inference time make it the best choice for practical road maintenance operations where detection quality and speed are both critical. In contrast, models such as SOLO v2 and Mask R-CNN, despite their strong segmentation capabilities, exhibit slower processing times, making them less suitable for real-time deployments in large-scale applications.

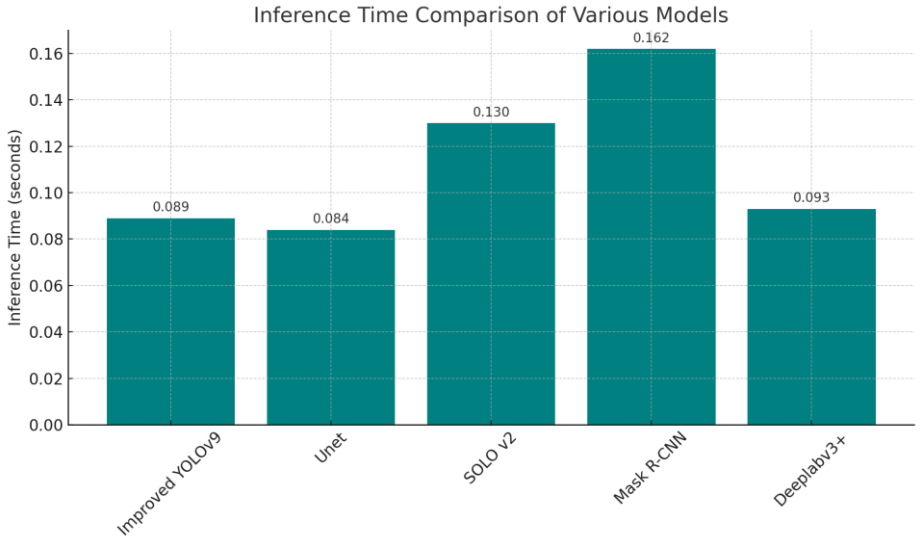


Fig. 8. Inference time of different models.

3) F1 and Jaccard indices of improved YOLOv9 and comparative models: In order to evaluate the segmentation performance of the Improved YOLOv9 model in detail, this research compares its results against those of Unet, SOLO v2, Mask R-CNN, and Deeplab v3+ on a test set consisting of 250 images. These images include two distinct types of cracks, providing a diverse basis for assessment. The average F1 and Jaccard coefficients obtained from the testing are graphically represented in Fig. 9.

The analysis of the results demonstrates that Mask R-CNN and Deeplab v3+ score significantly lower on both F1 and Jaccard indices compared to Improved YOLOv9 and Unet. This lower performance highlights the challenges these models face in accurately segmenting fine and narrow cracks, as well as broad and distinct cracks, under the testing conditions. Specifically, the metrics for Improved YOLOv9 are slightly higher than those for Unet, marking it as the superior model among the four evaluated. With F1 and Jaccard coefficients of 0.8403 and 0.7221, respectively, Improved YOLOv9 demonstrates the highest performance in terms of set evaluation metrics, indicating its enhanced capability in image segmentation and crack detection accuracy across diverse road surfaces.

These findings underscore the effectiveness of Improved YOLOv9 in handling varying crack types and conditions, potentially leading to more reliable and robust road maintenance and safety protocols. The integration of advanced feature extraction and attention mechanisms within Improved YOLOv9 likely contributes to its elevated performance, suggesting avenues for future enhancements in similar segmentation models.

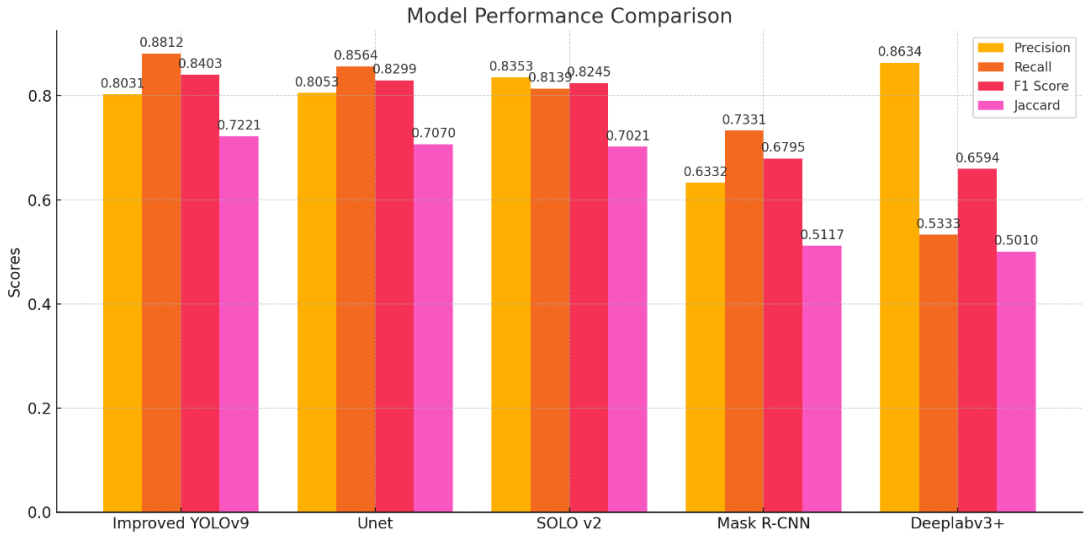


Fig. 9. F1 and Jaccard coefficients of different models.

4) Segmentation results on test data of improved YOLOv9 and comparative models: This study further analyzes the segmentation capabilities of the four models using the road surface crack test set. Three images of wide cracks with significant differences between the crack and the background were randomly selected from the test set for comparative analysis of predictions from the four models, as shown in Fig. 10. The segmentation results and evaluation metrics are consistent, with the Deeplab v3+ model performing poorly in actual segmentation, exhibiting issues such as excessive segmentation area and discontinuity. This indicates that Deeplab v3+ struggles with accurate crack image segmentation under conditions of limited image quantity. SOLO v2 and Mask R-CNN perform better than Deeplab v3+ but still show noticeable issues with crack misdetection. Improved YOLOv9 and Unet perform well in crack detection, with Improved YOLOv9 showing more precise crack segmentation, fewer misdetections, and better continuity.

To further investigate the performance of each model in segmenting fine narrow cracks with low contrast and narrow width, several such cracks were selected for comparison. Fig. 11 displays the segmentation results under conditions of narrow crack width and low contrast, where Mask R-CNN shows imprecise edge detection and misdetection issues (from left to right: original image, ground truth, Improved YOLOv9, Unet, SPLOv2, Mask R-CNN, Deeplabv3+). Deeplab v3+ not only has misdetection issues but also incorrectly identifies non-crack areas as cracks, particularly in cases of narrow longitudinal cracks, where misdetection is especially evident. Unet generally performs better than Mask R-CNN and Deeplabv3+ but also shows some misdetection. Improved YOLOv9, under conditions of narrow and low-contrast cracks, still clearly segments crack edges without misdetection or false detection, achieving the best segmentation results.

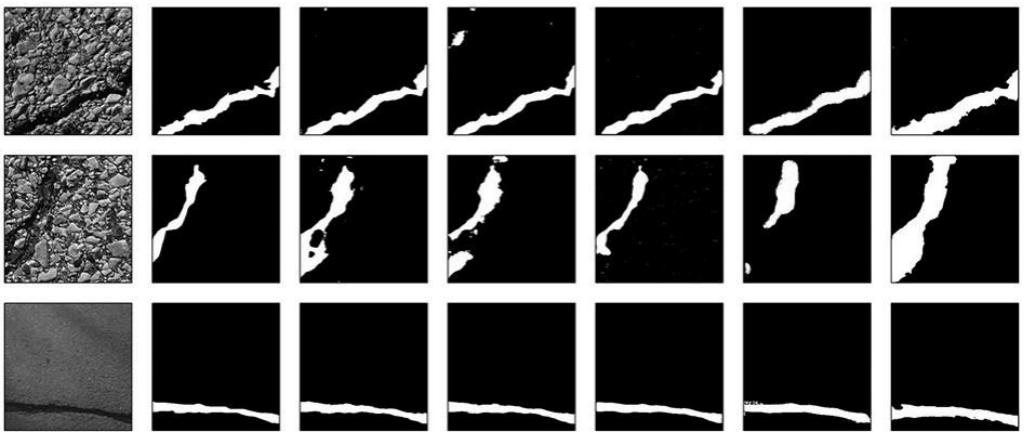


Fig. 10. Test results of pavement cracks under multiple models.

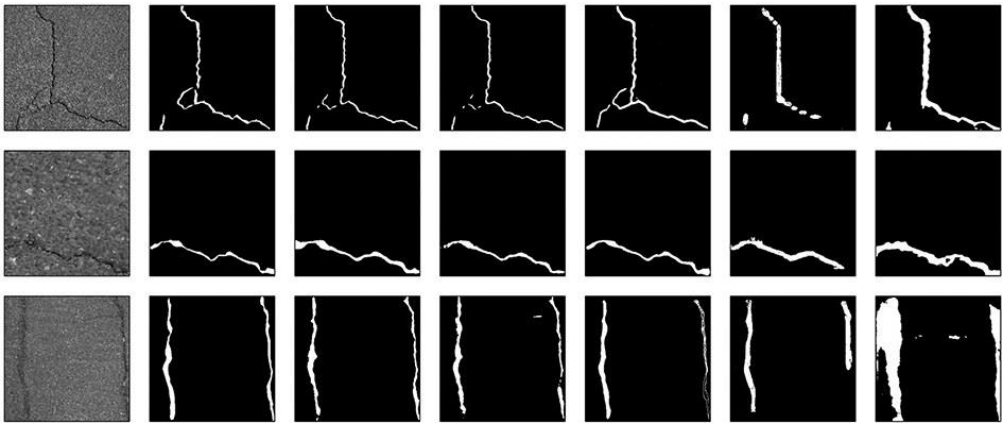


Fig. 11. Segmentation results of multiple models in the case of narrow crack width and low contrast.

IV. Discussion

To ensure the applicability of CrackNet in real-world road maintenance systems, deployment optimizations such as model pruning and quantization can be considered in the future.

Model Pruning: By removing less important connections or neurons, pruning can significantly reduce the model size and inference time without sacrificing much accuracy. This is particularly useful for deployment on mobile or edge devices with limited computational resources. Pruning can make CrackNet more efficient and suitable for resource-constrained platforms (such as drones or vehicle-mounted systems), enabling real-time crack detection in large-scale road inspections.

Quantization: Another potential optimization is model quantization, which converts high-precision weights (e.g., 32-bit floats) into lower precision (e.g., 8-bit integers). Quantization helps reduce the model size and speeds up inference, allowing faster computations while maintaining acceptable accuracy. This will make CrackNet more suitable for deployment in embedded systems and mobile devices, where memory and energy efficiency are critical.

V. Conclusion

Addressing the issues of imprecise edge segmentation and slow detection speed in traditional crack detection algorithms, this paper proposes a road crack detection method based on improved YOLOv9. By suppressing useless features extracted during high-low order feature fusion and enhancing the model's ability to extract crack features, the method achieves segmentation of both narrow and wide crack images.

F1 score and Jaccard coefficient are selected as evaluation metrics. A comparison between improved YOLOv9 and the basic Unet model in segmenting narrow and wide cracks demonstrates the superiority of the proposed method over the basic Unet algorithm, both quantitatively and qualitatively.

The real-time performance of the model is evaluated based on the inference time of a single-frame image. While the improved YOLOv9 outperforms Unet in segmentation performance, its inference speed is 0.089 seconds per frame, only 0.005 seconds slower than Unet, striking a balance between real-time performance and segmentation accuracy.

Further comparisons are made with three other classic segmentation networks. The results show that the evaluation metrics of the improved YOLOv9, with an F1 score of 0.8403 and a Jaccard coefficient of 0.7221, surpass those of classic segmentation models such as SOLO v2, Mask R-CNN, and Deeplabv3+. Compared to other models, the improved YOLOv9 achieves the highest evaluation metrics and the best segmentation performance, effectively extracting road cracks.

While the proposed model's segmentation performance on subtle narrow cracks is inferior to that on clear wide cracks, it does not account for the interference caused by different lighting conditions at different times. Future research will focus on adjusting the network structure to improve the segmentation performance on subtle narrow cracks and preprocessing images using image enhancement algorithms to eliminate the influence of lighting conditions, enabling high-precision crack detection under various lighting conditions. The proposed CrackNet model has significant potential for real-world applications in road maintenance systems. Its ability to accurately detect cracks in road surfaces, including fine and irregular cracks, positions it as a valuable tool for improving the efficiency and precision of road maintenance operations. By incorporating the CrackNet model into automated inspection systems, road maintenance departments can significantly reduce the time and labor costs associated with manual inspections, while ensuring more timely repairs, which are critical for preventing further road deterioration. Moreover, the model's real-time detection capability allows for continuous monitoring of road conditions, enhancing the safety of drivers and reducing the risks of accidents caused by undetected surface damage.

Funding

Key R&D and Promotion Special Project (Science and Technology Research) in Henan Province: Research on key technologies of road damage detection based on deep learning (232102210108).

Statement of Interest

The author declares that there is no conflict of interest in the manuscript.

REFERENCES

[1] Praticò F G, Fedele R, Naumov V, et al. Detection and monitoring of bottom-up cracks in road pavement using a machine-learning approach[J]. Algorithms, 2020, 13(4): 81.

[2] Ha J, Kim D, Kim M. Assessing severity of road cracks using deep learning-based segmentation and detection[J]. The Journal of Supercomputing, 2022, 78(16): 17721-17735.

[3] Feng X, Xiao L, Li W, et al. Pavement crack detection and segmentation method based on improved deep learning fusion model. Mathematical Problems in Engineering, 2020, 2020: 1-22.

[4] Wang S J, Zhang J K, Lu X Q. Research on Real-Time Detection Algorithm for Pavement Cracks Based on SparseInst-CDSM. Mathematics, 2023, 11(15): 3277.

[5] Bhat S, Naik S, Gaonkar M, et al. A survey on road crack detection techniques. 2020 international conference on emerging trends in information technology and engineering (ic-ETITE). IEEE, 2020: 1-6.

[6] Gavilán M, Balcones D, Marcos O, et al. Adaptive Road crack detection system by pavement classification. Sensors, 2011, 11(10): 9628-9657.

[7] Zhang L, Yang F, Zhang Y D, et al. Road crack detection using deep convolutional neural network. 2016 IEEE international conference on image processing (ICIP). IEEE, 2016: 3708-3712.

[8] Wang W, Wu L. Pavement crack extraction based on fractional integral valley bottom boundary detection. Journal of South China University of Technology (Natural Science Edition), 2014, 42(1): 117-122.

[9] Lee T, Yoon Y, Chun C, et al. Cnn-based road-surface crack detection model that responds to brightness changes. Electronics, 2021, 10(12): 1402.

[10] Hammouch W, Chouiekh C, Khaissidi G, et al. Crack detection and classification in moroccan pavement using convolutional neural network. Infrastructures, 2022, 7(11): 152.

[11] Fan R, Bocus M J, Zhu Y, et al. Road crack detection using deep convolutional neural network and adaptive thresholding. 2019 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2019: 474-479.

[12] Zhang L, Yang F, Zhang Y D, et al. Road crack detection using deep convolutional neural network. 2016 IEEE international conference on image processing (ICIP). IEEE, 2016: 3708-3712.

[13] Carr T A, Jenkins M D, Iglesias M I, et al. Road crack detection using a single stage detector based deep neural network. IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems. IEEE, 2018: 1-5.

[14] Siriborvornratanakul T. Pixel-level thin crack detection on road surface using convolutional neural network for severely imbalanced data. Computer-Aided Civil and Infrastructure Engineering, 2023, 38(16): 2300-2316.

[15] Sy N T, Avila M, Begot S, et al. Detection of defects in road surface by a vision system. MELECON 2008-The 14th IEEE Mediterranean Electrotechnical Conference. IEEE, 2008: 847-851.