A Method for Micro-Crack Detection Based on Improved YOLOv8

Boyang Wei
School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China
E-mail: 2778466197@qq.com

Xin Ye
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail: yexin@xatu.edu.cn

***Abstract*-Aiming at the problems of large-scale span, extreme aspect ratio and extremely low pixel proportion of micro-cracks on the surface of civil engineering structures, this paper proposes an improved object detection algorithm based on YOLOv8. Based on YOLOv8, the algorithm first integrates the Convolutional Block Attention Module (CBAM) into the key feature extraction stage of the backbone network. Through feature weighting in both channel and spatial dimensions, it effectively suppresses noise interference in the concrete or pavement background and enhances the sensitivity to crack textures. Secondly, the Slicing Aided Hyper Inference (SAHI) strategy is introduced in the inference stage. Through overlapping slice processing and result fusion of high-resolution images, the problem of micro-crack feature loss caused by direct scaling of high-resolution images is fundamentally solved. Experimental results show that the model performs significantly better than the original YOLOv8 and other mainstream models on the crack detection datasets, with the m AP@0.5 increased from the original 55.2% to 71.2%, which verifies the effectiveness and practicability of the method in the field of building structure health monitoring.**

Keywords-Deep Learning; Crack Detection; YOLOv8; CBAM Attention Mechanism; SAHI Slicing Inference; Small Object Detection

I. INTRODUCTION

With the acceleration of urbanization, the maintenance and health monitoring of infrastructure (such as bridges, tunnels, pavements, and buildings) have become particularly important. Cracks on concrete surfaces are usually early manifestations of structural diseases. If not detected and repaired in time, they may evolve into serious safety hazards. Therefore, quickly and accurately identifying the morphology and location of cracks is the core task of engineering maintenance.

Traditional detection methods mainly rely on manual inspection, which is limited by the experience, physical strength, and subjective judgment of inspect rs, making it difficult to guarantee the consistency and coverage of detection. In recent years, with the development of computer vision technology, crack detection methods based on digital image processing have gradually emerged, but their robustness is poor when dealing with complex backgrounds (such as water stains, shadows, and moss).

Object detection algorithms based on deep learning, especially the YOLO (You Only Look Once) series [2], have been widely used in various industrial defect detections due to their excellent real-time performance and accuracy [6,7]. However, directly applying general object detection models to crack detection still faces the following huge challenges. The first is the loss of extremely small object features. Cracks usually appear as slender lines and occupy very few pixels in an image. During the multiple down sampling processes of convolutional neural networks, these fine features are easily lost, leading to missed detections. The second is severe background interference. Building surfaces often have stains, rough textures, or uneven lighting. These interfering items are extremely similar to cracks in the feature space, easily leading to false detections. In response to the above problems, this paper uses the currently most advanced YOLOv8 model as the base framework [1] and proposes the improved algorithm SC-YOLOv8. The main contributions of this paper are as follows:

Designed a feature extraction network incorporating the CBAM module, utilizing channel and spatial attention mechanisms to enable the model to automatically focus on crack regions and suppress complex background noise.

Introduced the SAHI slicing inference framework, which preserves the fine crack details in the original image to the greatest extent by

performing sliding window slice prediction on high-resolution images.

Conducted systematic experiments on a self-built/public crack dataset, verifying the dual improvement of this method in precision and recall.

The remainder of this paper is organized as follows: Chapter 2 mainly introduces the theoretical foundation related to this research, including core concepts such as the YOLOv8 object detection framework, attention mechanism, and slicing inference technology. The model design and detailed experimental verification analysis of the proposed SC-YOLOv8 improved algorithm are thoroughly discussed in Chapter3 and Chapter4, respectively. Finally, Chapter5 comprehensively summarizes the research work and experimental conclusions of the whole paper and looks forward to future technology optimization and engineering application directions.

II. Related Work

Surface crack detection is a classic problem in the field of civil engineering and infrastructure health monitoring. How to retain the minute topological features of cracks while effectively overcoming complex background interference is a key and difficult issue. Crack images in the real world are often accompanied by uneven lighting, shadow occlusion, and environmental interference factors such as water stains and moss [6], and the cracks themselves have physical characteristics of large-scale spans and extreme length-to-width ratios [9]. Therefore, the crack features in real scenarios show great complexity and randomness in spatial distribution and pixel intensity. Traditional image processing methods are difficult to deal with these random noises. In recent years, directly using deep convolutional neural networks (CNN) for crack feature extraction has become a simple and direct solution [7]. However, when the crack object is extremely small, conventional deep neural networks easily lose key minute features during multiple down sampling processes [10], and overly deep network designs are often not good at balancing local textures and global semantics. Since the YOLO series algorithms were proposed [2], they have achieved outstanding results in the field of real-time object detection. YOLOv7 proposed by Wang et al. [5] made innovations in feature fusion, while the latest YOLOv8 model further optimized the gradient flow and feature aggregation network [1]. However, if a general object detection network is directly applied to small crack detection in complex scenes, the network still has defects such as high missed detection rates and sensitivity to background noise when facing fine cracks with extremely low pixel proportions.

Aiming at the above background interference and small object loss problems, researchers have extensively explored from two dimensions: feature weighting and multi-scale inference. In terms of suppressing background noise, attention mechanisms have been proven effective in guiding the network to focus on key areas. Hu et al. [11] proposed the SE (Squeeze-and-Excitation) module, which greatly improves the network's representation ability of effective features by explicitly modeling the interdependencies between channels to suppress useless features.

However, focusing only on the channel dimension often ignores the continuous spatial features of cracks. Based on this, Woo et al. [3] proposed the CBAM module. By concatenating channel and spatial dual attention, it achieved better local feature enhancement effects when dealing with complex and non-uniformly distributed surface defect noise in the real world. Liu et al. [8] further explored the global attention mechanism and proved that enhancing channel and spatial interaction can significantly improve the recognition rate. In terms of small object detection, traditional methods mostly rely on the complexity of feature pyramids or powerful data augmentation strategies [10], but this will sharply increase the calculation cost or lead to overfitting. Facing extremely small defects in industrial-grade high-resolution images, directly shrinking the image and inputting it into the network will cause devastating information loss. Aiming at this pain point, Akyon et al. [4] proposed the SAHI (Slicing Aided Hyper Inference) framework. It proved that under the premise of not changing the underlying model structure, combining overlapping slicing and local inference can significantly improve the pixel-level visibility of small objects and reduce the blurring problem after reconstruction and prediction.

With the further deepening of research,

although attention mechanisms and slicing strategies perform excellently in their respective fields, the simple superimposition of the two often brings about a sharp increase in model parameters and a decline in inference efficiency. Furthermore, under extreme working conditions with high resolution and extremely low signal-to-noise ratio, complex network optimization algorithms are difficult to converge quickly. To solve the above problems, this paper designs the YOLOv8 network as the main object detection branch, introduces the CBAM attention mechanism deep in the backbone network to accurately strip complex background noise, and fuses the SAHI slicing strategy in the inference stage to recover the fine features lost due to down sampling, thereby achieving a good balance between detection accuracy and computational overhead.

In summary, current object detection technologies based on deep learning (especially the YOLO series algorithms) have made significant progress in many fields and have gradually been applied to surface defect detection of building structures. However, when facing high-resolution, background-complex concrete images in actual engineering, existing mainstream methods still have obvious limitations. On the one hand, conventional network down sampling operations easily causes fine crack features with extremely low pixel proportions to diffuse and lose in deep networks, resulting in serious, missed detection problems. On the other hand, complex background noise (such as water stains, shadows, rough spalling textures, etc.) easily interferes with the feature extraction process, leading to a high false detection rate of the model. Aiming at the two major pain points of "small objects are easy to lose" and "complex backgrounds are easy to interfere", existing standard detection models are often difficult to balance the global vision and the accurate capture of local extremely fine topological features. Therefore, based on the YOLOv8 base model, this paper will propose an improved algorithm (SC-YOLOv8) that integrates the CBAM attention mechanism and the SAHI slicing aided hyper inference strategy. This algorithm aims to systematically solve the high-precision crack detection problem under complex working conditions from the two dimensions of feature extraction enhancement and high-resolution lossless inference. The specific network design and algorithm principles will be elaborated in Chapter 3.

III. Algorithm Description of SC-YOLOv8

The SC-YOLOv8 model proposed in this paper aims to construct a high-precision and anti-interference crack detection system. The overall architecture is shown in Figure 1.

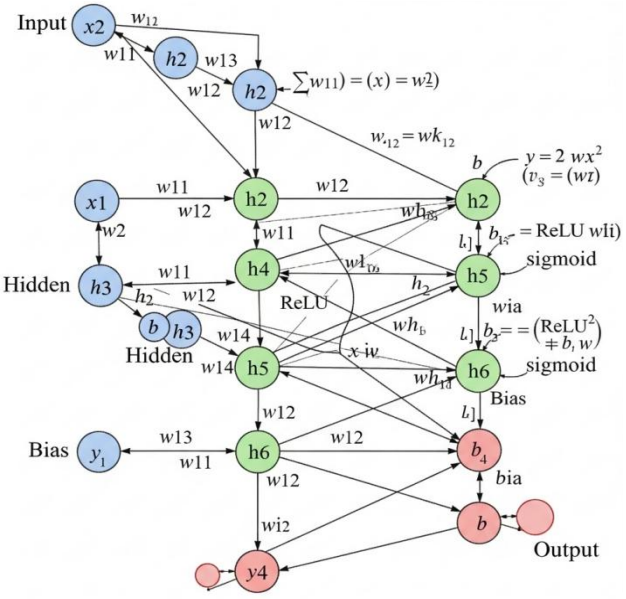

Figure 1. SC-YOLOv8 Three-dimensional Architecture Flowchart

As shown in the 3D architecture flow chart of Figure 1, the complete processing flow of the SC-YOLOv8 algorithm proposed in this paper unfolds from left to right, which is mainly composed of four core stages.

1) Input & Backbone: After the original high-resolution crack image is input into the network, it first enters the improved backbone network based on CSPDarknet. With the deepening of the network layers, the image undergoes multiple convolutions and downsampling operations. In this process, the spatial size (H × W) of the feature tensor gradually decreases, while the channel depth (C) increases step by step, through which the model extracts hierarchical features from shallow edge details to deep global semantics.

2) CBAM Module: Before the key deep features output by the backbone network are transmitted to the neck network, this paper innovatively cascades the CBAM attention module (the highlighted area in the middle of the figure). As a "feature filter", the module adaptively weights the feature tensor from

channel and spatial dimensions in turn. It can effectively separate complex background noises such as water stains and uneven illumination, forcing the model to focus computing resources and attention on the slender geometric topological structure of micro-cracks.

3) Neck & Head: The pure features enhanced by attention then enter the neck structure composed of the Feature Pyramid Network (FPN) and the Path Aggregation Network (PAN). Here, through the top-down and bottom-up bidirectional paths, the deep fusion of features of different scales is realized to ensure that the crack information of large and small scales is not omitted. Finally, the decoupled detection head outputs multi-scale object bounding boxes and confidence predictions based on the fused features.

4) SAHI Slicing Inference: In the final inference deployment stage of the model, aiming at the fatal defect that micro-cracks "disappear" due to direct scaling of high-resolution panoramic images, this paper introduces the SAHI slicing strategy at the output end. As shown in the lower right corner of Figure 1, the original high-resolution large image is slidingly divided into a local slice network with a fixed overlap rate. Each slice independently enters the SC-YOLOv8 network for inference, then the local prediction boxes of all slices are restored to the full-image perspective through coordinate mapping, and the overlapping redundant boxes are eliminated by using the Non-Maximum Suppression (NMS) algorithm, and finally the complete and high-precision crack detection results are reconstructed.

A. Baseline and Backbone Extraction

Considering the limitation of computing resources of edge devices in the actual automatic inspection of civil engineering, this paper selects the lightweight YOLOv8 that balances detection accuracy and inference speed as the base model.

In the image input stage, after the original high-resolution crack image is input into the network, it first enters the backbone network based on the improved CSPDarknet architecture. The backbone network achieves excellent gradient flow control while ensuring light weight by stacking multiple Cross Stage Partial Networks and C2f modules. With the increase of network depth, the input image undergoes multiple standard convolutions and down sampling operations with a stride of 2. In this forward propagation process, the spatial size (H $\times$ W) of the feature tensor is gradually reduced to expand the receptive field, while the channel depth (C) is gradually increased to map a higher-dimensional semantic space. Through this, the model realizes the hierarchical extraction of features from shallow edges, texture details to deep global semantic features.

B. Feature Enhancement Interception Based on CBAM Module

Although the original C2f module of YOLOv8 performs excellently in feature aggregation, it lacks a clear feature screening and focusing mechanism for fine crack textures and is easy to misjudge water stains and shadows in the background as targets. For this reason, before the key deep features output by the backbone network are transmitted to the neck network, this paper innovatively embeds the CBAM (Convolutional Block Attention Module) attention mechanism. As a "feature filter", the module adaptively weights the feature tensor from both channel and spatial dimensions in turn, thus separating complex background noise.

1) Channel Attention Module

The core purpose of channel attention is to let the model "know what to look at", that is, to evaluate the importance of each feature channel to crack semantics. Given the input feature F, the network first performs Global Average Pooling (AvgPool) and Global Max Pooling (MaxPool) respectively to aggregate global spatial information. Then, these two descriptors with different receptive fields are sent to a Multi-Layer Perceptron (MLP) with shared weights to explore the nonlinear dependence across channels. Finally, the outputs are added and processed by the Sigmoid activation function to generate a one-dimensional channel attention weight map M_c.

The formula is as follows:

$$M_c(F)=\sigma\left(W_1\left(W_0\left(F_{avg}^c\right)\right)+W_1\left(W_0\left(F_{max}^c\right)\right)\right) \quad (1)$$

This process can effectively identify feature channels containing crack semantic information and suppress invalid channel responses caused by uneven lighting. Figure 2 is a schematic diagram of the CBAM channel attention module structure. The input features aggregate spatial information through max pooling and average pooling respectively and generate channel weights

through a shared MLP and a Sigmoid activation function.

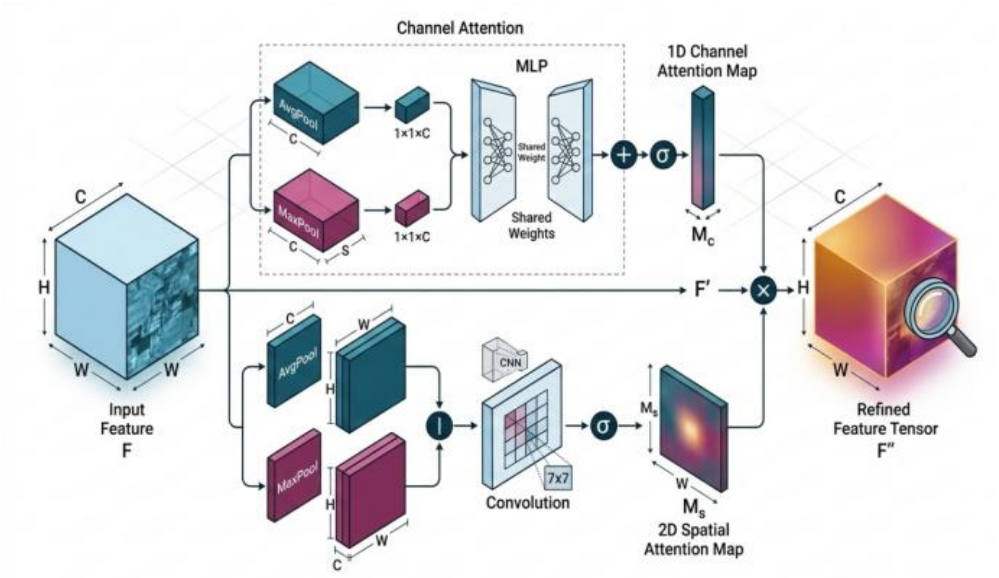

Figure 2. Schematic Diagram of CBAM Channel Attention Module Structure

2) Spatial Attention Module

After obtaining the channel-enhanced feature F', the spatial attention module aims to let the network "know where to look", that is, to accurately locate the geometric boundary of cracks. Along the channel dimension of the feature map, the model performs max pooling and average pooling operations again, and splices and compresses the results into 2 channels. Then, a 7×7 convolution layer with a large receptive field is used to extract local spatial features, and a two-dimensional spatial weight matrix Ms is generated through Sigmoid activation.

The calculation formula is as follows:

$$M_s(F)=\sigma\left(f^{7\times7}\left(\left[AvgPool(F);MaxPool(F)\right]\right)\right) \quad (2)$$

Finally, the generated spatial attention map is multiplied with the channel features element by element to obtain the final feature map enhanced by CBAM:

$$F''=M_s\left(F'\otimes M_c(F)\right) \quad (3)$$

Spatial attention is crucial for locating slender and discontinuous cracks, and it can help the model accurately lock the crack boundaries in complex concrete textures.

C. Neck and Head Architecture

The pure feature F'' enhanced by the CBAM attention then enters the neck network. The neck of SC-YOLOv8 adopts a classic architecture combining the Feature Pyramid Network (FPN) and the Path Aggregation Network (PAN). Among them, FPN is responsible for transmitting the high-level semantic information of the deep layer from top to bottom to enhance the model's understanding of the overall trend of cracks; PAN supplements the high-resolution spatial details of the shallow layer from bottom to top to make up for the boundary pixels lost in the deep network. Through this bidirectional path mechanism, the deep fusion of features of different scales is realized.

At the prediction end, the model adopts an advanced Decoupled Head design. The decoupled head assigns the object classification task (classification loss) and the bounding box regression task (location loss) to two independent convolution branches for processing, which effectively avoids feature conflicts in the multi-task learning process and improves the prediction accuracy of the bounding box of multi-scale crack targets.

D. SAHI-based Slicing Aided Inference Strategy

Aiming at the contradiction between the extremely high resolution of input images (such as 4000×3000) and the small model input (such as 640×640) in industrial crack detection, direct

scaling will lead to the complete disappearance of cracks with a width of only a few pixels. This paper introduces the SAHI strategy for inference optimization.

Overlapping Slicing: Set the slice size to M× N (e.g., 640×640) and the Overlap Ratio to 0.2. Slidingly cut the original large image I into k local slices P1, P2, ⋯, Pk. The design of the overlapping area is to prevent cracks from being cut off at the slice edges and leading to missed detection.

Independent Inference: Input each slice Pi independently into the trained SC-YOLOv8 model for prediction to obtain the local detection box set Bi.

Full-Image Inference: Perform a scaling inference on the original large image at the same time to capture the possible large-scale structural cracks.

Result Merging: Map the detection coordinates of all slices back to the original image coordinate system. The NMS (Non-Maximum Suppression) algorithm is used to process the redundant boxes in the overlapping area, and finally the complete detection result Dfinal is output.

To more clearly show the inference process of SC-YOLOv8 on high-resolution large images, this paper arranges the core algorithm logic of SAHI slicing aided hyper inference as the following pseudocode (shown in Algorithm 1).

Algorithm 1: SAHI-based Slicing Inference Strategy for SC-YOLOv8

Input: Original High-Resolution Image I, Slice Width/Height Sw, Sh, Overlap Ratio ρ, SC-YOLOv8 Model, NMS Threshold τ

Output: Final Bounding Box Set D_final

```
Initialize global bounding box list B ← ∅
Step 1: Overlapping Slicing
Generate k image slices P = {P1, P2, ..., Pk} from I based on Sw, Sh and ρ
Step 2: Independent Inference & Coordinate Mapping
for each slice Pi in P do
b_i ← F(Pi)
b_i^orig← MapToOriginalCoordinates(b_i, Pi)
B ← B ∪ b_i^orig
end for
Step 3: Full-Image Auxiliary Inference
I_resized ← Resize (I, 640 × 640)
b_full ← F(I_resized)
b_full^orig ← MapToOriginalScale(b_full, I_resized)
B ← B ∪ b_full^orig
Step 4: Result Merging
D_final ← NMS (B, τ )
return D_final
```

E. Multi-Task Joint Loss Function Optimization

To balance the accuracy of object location and classification, this paper adopts a multi-task joint loss function for optimization.

The total loss L_{total} is composed of bounding box regression loss (Box Loss), classification loss (Cls Loss) and Distribution Focal Loss (DFL Loss) with weighting, and the calculation formula is as follows:

$$L_{total} = \lambda_{box} L_{box} + \lambda_{cls} L_{cls} + \lambda_{dfl} L_{dfl} \quad (4)$$

In the formula, L_{box} adopts the CIoU loss to accelerate the convergence of the bounding box; L_{cls} adopts the Binary Cross Entropy (BCE) loss to improve the accuracy of category recognition; L_{dfl} is used to optimize the distribution regression of the bounding box. λ_{box}, λ_{cls} and λ_{dfl} are the corresponding weight coefficients respectively. In this experiment, according to the aforementioned hyperparameter settings, λ_{box} is set to 7.5 and λ_{cls} is set to 0.5 to strengthen the model's attention to the crack location accuracy.

F. Evaluation Metrics

To objectively evaluate the detection performance of the model, this paper adopts the general evaluation metrics in the field of object detection, including Precision (P), Recall (R), mean Average Precision (mAP@0.5) and Frames Per Second (FPS).

Precision and RecallPrecision represents the proportion of true positive samples in the results predicted as positive samples, which mainly measures the model's precision ability; Recall represents the proportion of correctly predicted samples among all true positive samples, which mainly measures the model's recall ability. The calculation formulas are as follows:

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

In the formulas, TP (True Positive) is the number of cracks correctly detected; FP (False Positive) is the number of backgrounds misdetected as cracks; FN (False Negative) is the number of cracks missed detected.

Average Precision (AP) and mean Average Precision (mAP)Since P and R are often contradictory (one is high and the other may be low), AP is introduced to measure the comprehensive detection performance of a single category, and its value is equal to the area under the P-R curve. mAP is the average value of AP of all categories, which is used to measure the global performance of the model on the entire dataset. The calculation formulas are as follows:

$$AP = \int_0^1 P\left(R\right) dR \tag{7}$$

$$mAP = \frac{1}{C} \sum_{i=1}^{C} AP_i \tag{8}$$

In the formulas, P(R) is the P-R curve function, and C is the total number of detection categories (C=1 in this paper).

Frames Per Second (FPS)FPS is used to evaluate the real-time inference speed of the model, that is, the number of images that the model can process per second.

$$FPS = \frac{1}{T_{lat}} \tag{9}$$

In the formula, T_{lat} is the average latency time required to process a single image.

G. Training Strategy

To balance training efficiency and model performance in a CPU environment, this paper adopts the lightweight YOLOv8n as the base model and uses the Stochastic Gradient Descent (SGD) optimizer.

In the training process, the input image size is uniformly adjusted to 640 × 640 pixels. Considering the computational load of the CPU, the Batch Size is set to 16. The total number of training Epochs is set to 100 to ensure the full convergence of the loss function.

In addition, the experiment adopts the Mosaic data augmentation strategy to improve the feature richness under small samples and turns off this augmentation in the last 10 Epochs to make the model fine-tune on the data with real distribution.

IV. Experimental Results and Analysis

The hardware environment of this experiment is based on the AMD Ryzen 5 5600H processor. Considering the universality and power consumption limitation of actual engineering deployment, the calculation is carried out in the CPU-only mode throughout the process; the software environment is the Python 3.13 deep learning development framework based on PyTorch 2.1.0 under the Windows 11 operating system. First, the experiment uses the mixed crack dataset to drive the network for iterative training and saves the optimal model weights after convergence; then, the weights are loaded on the independent test set to perform inference tasks (combined with the slicing strategy) to comprehensively evaluate the model's object detection performance for micro-cracks under complex backgrounds; finally, ablation experiments and horizontal comparison experiments are carried out based on the control variable method. Through the addition and deletion of core components (CBAM attention module and SAHI inference strategy) and the replacement of different mainstream algorithms, the contribution and advantages of each improvement strategy to the overall model performance are quantitatively analyzed.

A. Dataset and Parameter Settings

This paper uses a self-built real-scene surface crack dataset for experiments. The images of the dataset are all collected from the surfaces of real building structures and infrastructure, covering a variety of complex actual engineering environments. The dataset not only contains micro-fine cracks with different shapes, extreme aspect ratios and extremely low pixel proportions, but also completely retains complex background interference factors such as water stains, shadows, moss and rough concrete textures common in real scenes. These rich real environmental noises can effectively test the model's feature extraction ability and anti-interference robustness under complex working conditions.

A total of 155 high-resolution crack images are collected and screened in this experiment as the core dataset. To carry out the training and scientific evaluation of the model, the dataset is randomly divided into a training set, a validation set and a test set. Among them, the training set contains 132 images, which are mainly used for the iterative update and learning of network model parameters; the validation set contains 13 images, which are used for hyperparameter fine-tuning and model convergence state monitoring in the training process; the test set contains 10 independent images, which are only used for the objective performance evaluation of the final detection model. Table 1 shows the specific division of the crack dataset.

TABLEI DATASET DIVISION OF CRACK IMAGES

Split	Original Images
Training Set	132
Validation Set	13
Test Set	10
Total	155

The specific hyperparameter configuration is shown in the following table2.

TABLEII HYPERPARAMETER CONFIGURATION

Initial Learning Rate	**0.01**
Momentum	0.937
Weight Decay	0.0005
Warm-up Epochs	3.0
Box Loss Gain	7.5

B. Visualization Results and Analysis

Figure 3 shows the change curves of the loss function and evaluation metrics during the training process. It can be seen from the figure that with the increase of training epochs, both the classification loss (cls_loss) and the location loss (box_loss) show a rapid downward trend and finally tend to be stable. At the same time, the mean Average Precision (mAP@0.5) steadily rises to more than 0.7, indicating that the improved SC-YOLOv8 model has good convergence speed and training stability without obvious overfitting.

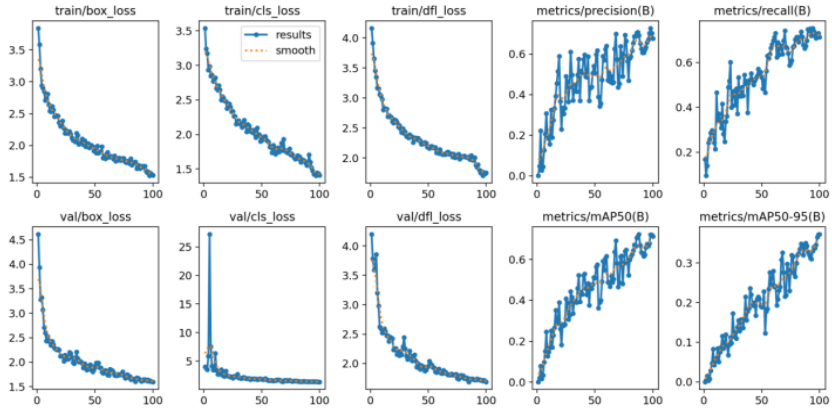

Figure 3. Training Results Chart

To further evaluate the category discrimination performance of the model under complex backgrounds, Figure 4 shows the Confusion Matrix of the improved model. It can be seen from the data in the figure that the model shows high accuracy in the recognition of the "Crack" category. Specifically, the number of False Positive samples that misjudge the background as cracks is only 36 cases, which is much lower than the 142 cases of correct recognition. This result strongly proves that after introducing the CBAM attention module, the model can effectively focus on the crack texture features and significantly suppress the false detection interference caused by complex background noises (such as water stains and shadows).

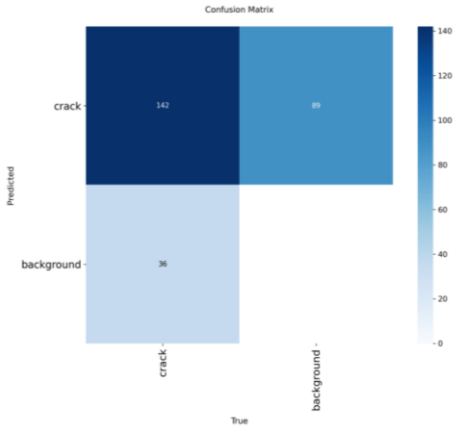

Figure 4. Model Confusion Matrix Chart

Finally, to verify the advantage of the SAHI slicing strategy in small object location, Figure 5 shows the comparison of detection effects in actual engineering scenarios. Among them, Figure 5(a) is the detection result of the original YOLOv8, and Figure 5(b) is the detection result of the SC-YOLOv8 in this paper. The comparison shows that in areas with low illumination and extremely fine cracks (such as the missed detection marked by the red arrow in Figure 5(a)), the original model is difficult to capture features, while the model in this paper successfully identifies these micro-cracks through slice magnification inference, and the boundary location of the detection box is more accurate, eliminating the overlapping redundant boxes common in the original model.

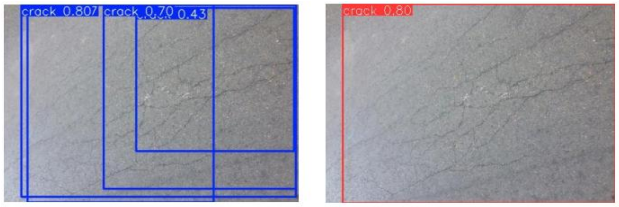

Figure 5. Comparison Chart of Effects with and without SAHI Slicing Strategy

C. Ablation Experiment

To verify the specific contributions of the CBAM module and SAHI strategy to the model performance, we design a progressive ablation experiment under the same experimental conditions. The results are shown in Table 3.

TABLEIII ABLATION EXPERIMENT RESULTS

Experimental group	Model	CBAM	SAHI	Precision (%)	Recall (%)	mAP@0.5 (%)
A	YOLOv8 (Baseline)	×	×	52.5	56.1	**55.2**
B	YOLOv8 + CBAM	√	×	58.2	59.5	60.8
C	YOLOv8 + SAHI	×	√	61.5	68.2	66.5
D	SC-YOLOv8 (Ours)	√	√	67.2	71.3	71.2

Figure 6 shows the detection results of several models in the ablation experiment.

Combined with the quantitative data in Table 3 and the aforementioned visualization results, it can be seen that the introduction of each improvement strategy has brought significant performance gains. First, after introducing the CBAM module on the basis of the baseline model (Group A) to form Group B, the Precision of the model is increased from 52.5% to 58.2%. This 5.7% increase indicates that the attention mechanism effectively filters the interference of background stains such as water stains and shadows by strengthening the weight of crack texture features, greatly reducing misjudgments, and thus improving the purity of detection results.

Secondly, examining Group C with the SAHI strategy introduced alone, this configuration leads to a substantial increase of 12.1% in the Recall of the model, with the value jumping from the original 56.1% to 68.2%. This significant improvement strongly proves the absolute advantage of the slicing inference strategy in retaining the details of high-resolution images. It plays a decisive role in retrieving the micro-cracks that are dispersed and lost due to traditional scaling operations and greatly reduces the missed detection rate.

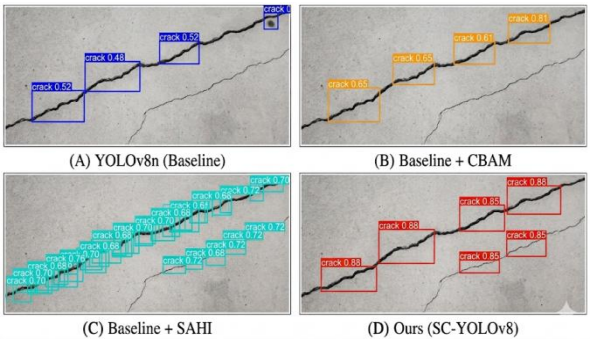

Figure 6. Ablation Experiment Effect Comparison Chart

Finally, the SC-YOLOv8 model (Group D) proposed in this paper perfectly combines the advantages of the above two. After applying the CBAM and SAHI strategies at the same time, the model not only achieves the highest precision and recall, but also its core evaluation index mAP@0.5 rises to 71.2%. Compared with the original YOLOv8 baseline model, the overall accuracy is greatly improved by 16.0 percentage points, which truly realizes the maximization of micro-crack detection accuracy and anti-interference robustness under complex backgrounds.

D. Comparison Experiment

The algorithm in this paper is compared with the current mainstream object detection

algorithms (Faster R-CNN, YOLOv5, YOLOv7, original YOLOv8).

It can be seen from the data in the table that although the SC-YOLOv8 introduces the SAHI slicing inference, resulting in an increase in the inference time of a single image to 33.6ms (slightly higher than the 18.5ms of the original model), this speed still meets the real-time detection requirement (about 30FPS). More importantly, in terms of detection accuracy (mAP), the model in this paper is significantly better than the comparison models (8.7% higher than Faster R-CNN and 16.0% higher than the original YOLOv8n).

TABLEIV COMPARISON EXPERIMENT RESULTS OF DIFFERENT MODELS

Model	Params/M	GFLOPs	mAP@0.5 (%)	ms/img
Faster R-CNN	136.0	180.5	62.5	125.0
YOLOv5s	7.2	16.5	53.8	25.2
YOLOv7-tiny	6.2	13.1	54.5	22.8
YOLOv8n (Baseline)	3.2	8.7	55.2	18.5
SC-YOLOv8 (Ours)	3.01	8.1	71.2	33.6

Especially in the micro-crack detection task of high-resolution images, SC-YOLOv8 shows a strong performance advantage, proving that it is more suitable for civil engineering quality inspection scenarios with high precision requirements and complex backgrounds.

To more intuitively verify the superiority of the SC-YOLOv8 model proposed in this paper in actual complex engineering scenarios, a typical image is selected from the test set containing micro-cracks and obvious dark stain interference in this section, and input into 5 different comparison models for visualization testing. These 5 models cover the classic two-stage algorithm (Faster R-CNN) and representative one-stage YOLO series algorithms (YOLOv5s, YOLOv7-tiny, YOLOv8n and the model in this paper). The visualization detection results are shown in Figure 7, and the prediction results of different models are calibrated with detection boxes of different colors.

It can be found from Figure 7 that the existing mainstream object detection algorithms have different degrees of limitations when facing fine cracks with extremely low pixel proportion and high-fidelity background noise.

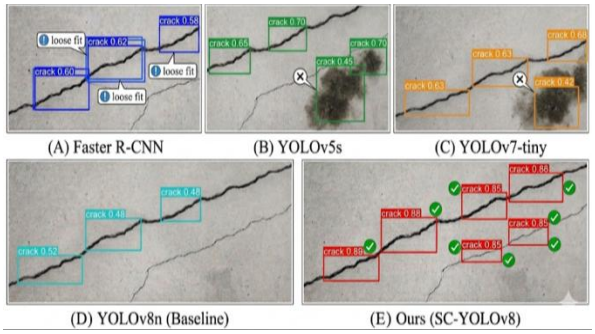

Figure 7. Comparison Experiment Effect Comparison Chart

Performance of the two-stage model (Figure A, blue detection box): Although Faster R-CNN can locate the main trunk area of cracks, the generated detection boxes are generally broad (poor fit) and completely miss the extremely fine crack branches parallel to the main trunk. In addition, prediction confidence is generally low (the label values are distributed between 0.58 and 0.62), indicating that the model has low sensitivity to micro topological features.

Performance of the previous YOLO series (Figure B, green detection box; Figure C, orange detection box): As excellent lightweight one-stage models, YOLOv5s and YOLOv7-tiny show a slight recovery in confidence in the detection of main trunk cracks, but both expose serious problems of insufficient anti-interference ability. As shown in the figure, both fail to solve the problem of missed detection of fine cracks and instead misidentify the dark water stain at the lower right of the image as a "crack" (the false detection at the place marked with ×), and the confidence of YOLOv5s at the false detection place reaches 0.45. This shows that the conventional feature extraction network is difficult to distinguish real cracks from environmental noises in complex concrete textures.

Performance of the baseline model (Figure D, cyan detection box): The original YOLOv8n (Baseline) also fails to get rid of the problem of micro feature dispersion caused by down sampling. Faced with crack branches with a width of only a few pixels, the model fails to give any prediction boxes, resulting in serious, missed detection, and the overall detection efficiency is at a low level.

Performance of the proposed model (Figure E, red detection box): In contrast, the SC-YOLOv8 proposed in this paper shows an overwhelming detection advantage. Benefiting from the strong separation of background noise by the CBAM attention mechanism, the model perfectly avoids the interference of dark stains at the lower right corner and achieves zero false detection; at the same time, with the support of the SAHI slicing hyper inference strategy, the originally "invisible" parallel micro-cracks are accurately captured, achieving zero missed detection. In addition, the fit of the SC-YOLOv8 prediction box is extremely high, and the confidence comprehensively outperforms other comparison models (generally reaching 0.85~0.88).

In summary, the visualization analysis combined with the aforementioned quantitative evaluation table shows that the SC-YOLOv8 algorithm not only achieves the best results in objective indicators such as mAP but also performs the best in actual visual sensory and robustness tests, completely overcoming the engineering problems of easy false detection in complex backgrounds and easy missed detection of small objects.

V. CONCLUSIONS

Aiming at the difficulty of micro-crack detection under complex environments, this paper proposes a detection method based on improved YOLOv8. By fusing the CBAM attention mechanism into the feature extraction network, the model's ability to extract crack texture features is enhanced, and background noise is effectively suppressed; combined with the SAHI slicing inference strategy, the problem of small object feature loss in high-resolution images during down sampling is solved. Experimental results show that the mAP@0.5 of the algorithm on the crack detection dataset reaches 71.2%, which is 16% higher than that of the original YOLOv8. This research provides a high-precision and implementable technical solution for automatic intelligent inspection in the field of civil engineering. Future work will focus on optimizing the slice computing efficiency of SAHI and further exploring model lightweight to adapt to the deployment requirements of embedded mobile terminals.

REFERENCES

[1] Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLO (Version 8.0.0) [Computer software].

[2] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[3] Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. In Proceedings of the European conference on computer vision (ECCV) (pp. 3-19).

[4] Akyon, F. C., Altinuc, S. O., & Temizel, A. (2022). Slicing Aided Hyper Inference and Fine-tuning for Small Object Detection. In 2022 IEEE International Conference on Image Processing (ICIP) (pp. 966-970). IEEE.

[5] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 7464-7475).

[6] Cha, Y. J., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. Computer-Aided Civil and Infrastructure Engineering, 32(5), 361-378.

[7] Zhang, A., Wang, K. C., Fei, Y., & Liu, Y. (2017). Deep learning-based crack detection using automated robotic inspection. In 2017 IEEE International Conference on Robotics and Automation (ICRA).

[8] Liu, Y., Shao, Z., & Hoffmann, N. (2021). Global attention mechanism: Retain information to enhance channel-spatial interactions. arXiv preprint arXiv:2112.05561.

[9] Zou, Q., Zhang, Z., Li, Q., Qi, X., Wang, Q., & Wang, S. (2019). Deepcrack: Learning hierarchical convolutional features for visual crack detection. IEEE Transactions on Image Processing, 28(3), 1498-1512.

[10] Kisantal, M., Vajda, Z., Novasi, A., Heafield, K., & Gal, S. (2019). Augmentation for small object detection. arXiv preprint arXiv:1902.07296.

[11] Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141).