Enhanced YOLOv8 for Detecting Multiple Defects on Bridge Surfaces

Ruiping Li, *Student Member, IEEE,* Linchang Zhao, *Member, IEEE,* Hao Wei, *Student Member, IEEE,* Bocheng OuYang, *Member, IEEE,* Bing Fang, *Member, IEEE,* Yongchi Xu, *Member, IEEE,* and Guoqing Hu, *Member, IEEE*

***Abstract*—With the advancement of machine vision, numerous models have been created to detect imperfections in bridges. However, the bulk of these models are designed for single defect detection and are not adept at managing cases with concurrent multiple defects. As a result, quickly recognizing the array of defects on bridge surfaces is still a major obstacle. In response to this challenge, the current research introduces the YOLOv8-CBAM-Wise-IoU model, specifically crafted for the detection of seven distinct bridge surface defect categories. This model integrates the CBAM mechanism for focusing attention and the Wise-IoU for calculating loss, with its effectiveness measured by metrics including accuracy, retrieval rate, F1 measure, and mAP50. Rigorous ablation analyses and benchmarking against both single-tier and multi-tier deep learning frameworks were performed to substantiate the model's utility. The YOLOv8-CBAM-Wise-IoU model exhibited formidable performance, recording an accuracy rate of 97.9%, a retrieval rate of 76%, an F1 measure of 58%, an mAP50 of 55.4%, and an mAP50-95 of 32.4%. These results outstrip those of standard models and other ablation variations, emphasizing the model's ability to boost the precision and robustness of detecting various defect types on bridge surfaces. Code is available at https://github.com/IamSunday/Enhanced-YOLOv8-for-Detecting-Multiple-Defects-on-Bridge-Surfaces.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/9466

***Index Terms*—CBAM; Wise-IoU; YOLOv8-CBAM-Wise-IoU; Bridge Surface Defect.**

I. Introduction

BRIDGES are vital to urban infrastructure, enabling road traffic essential for growth and economic prosperity [1]. However, prolonged exposure to environmental conditions can cause damage such as cracks, water seepage, and concrete deterioration, compromising structural integrity and posing safety risks to urban traffic [2].

Traditional bridge inspections, relying on manual evaluations and targeted inspections, are inefficient, costly, and prone to missed defects [3], [4], limiting large-scale surface issue detection [5]. Thus, leveraging AI and computer vision for rapid defect detection has become a critical challenge requiring innovative solutions [6], [7].

In recent years, deep learning has shown great potential in bridge surface defect detection, gradually replacing manual and traditional image processing methods such as Sobel and Canny [8], [9]. Early studies, however, struggled with complex backgrounds and noise sensitivity [10]. The advent of CNNs improved localization accuracy through two-stage models like Faster R-CNN, but computational redundancy hindered real-time performance [11], [12].

The YOLO series of single-stage models balance speed and accuracy with end-to-end detection. For example, Tran et al. [13] combined YOLOv7 with an improved U-Net to create a joint detection-segmentation network, achieving 92.38% accuracy in crack length detection, 91% in width classification, and mAP@0.5 of 0.748, with 8x higher efficiency than Faster R-CNN. Zhang et al. [14] enhanced YOLOv5 by integrating the SPPCSPC module for adaptive image output and multi-scale receptive field capture, while transposed convolution improved feature learning and reduced information loss. Although ResNet-101 and Mask R-CNN have been used for classification and quantification, their generalization in multi-defect scenarios remains limited.

Despite promising results, these studies face two key challenges: focusing solely on single defects and overlooking feature interference from multiple defects, and the sensitivity of the loss function to low-quality anchor boxes, affecting localization accuracy.

To address these challenges, this research introduces the YOLOv8-CBAM-Wise-IoU model, integrating CBAM attention and Wise-IoU loss into YOLOv8 for more accurate and efficient bridge defect detection. The main contributions are:

(1)Curate a database for detecting various imperfections on bridge surfaces in Guizhou, including seven defect categories and addressing common practical challenges.

(2)CBAM in YOLOv8 fine-tunes channel and spatial weights, enhancing perception and generalization for defect detection.

(3)Wise-IoU in YOLOv8 reduces the impact of high-quality anchors and prioritizes average-quality ones, improving detection accuracy.

(4)This study enhances YOLOv8 with CBAM and Wise-IoU, demonstrating its effectiveness in detecting multiple bridge surface defects.

The associate editor coordinating the review of this manuscript and approving it for publication was Eduardo José da Luz (*Corresponding author: Linchang Zhao*).

R. Li, Linchang Zhao, H. Wei, B. OuYang, B. Fang, and Y. Xu are with the Guiyang University, Guiyang, China (e-mails: liruiping@gyu.edu.cn, anilue@alu.cqu.edu.cn, weihao@gyu.edu.cn, Ouayng@gyu.edu.cn, Fangb@gyu.edu.cn, and yongXu@gyu.edu.cn).

G. Hu is with the PKU-HKUST Shenzhen-Hong Kong Institution and Shenzhen Institute of Peking University, Shenzhen, China (e-mail: huking@pku.edu.cn).

The paper is structured as follows: Section 2 reviews related work, Section 3 presents the methodology, Section 4 details the experimental setup, Section 5 analyzes the results, and Section 6 concludes with key findings.

II. Related Work

Early bridge surface defect detection relied on image processing techniques. A 2003 study [15] compared four edge detection methods—FHT, FFT, Sobel, and Canny—for crack detection, with FHT achieving the highest accuracy of 86%, while Sobel was significantly affected by noise. A 2008 study [16] introduced an automated machine vision system with 94.1% accuracy, outperforming traditional methods and validating its performance on 100 noisy images.

Subsequent research refined these techniques. V et al. [17] improved Otsu segmentation using a maximum-minimum gray level algorithm, reducing noise and highlighting crack pixels. Study [18] applied the Hilbert-Huang transform to analyze post-tensioned prestressed concrete bridges and proposed a probabilistic model for feature extraction and anomaly detection, validated through extensive parameter evaluation.

In recent years, deep learning and object detection have significantly improved bridge defect detection. Study [19] retrained ResNet-101 for defect classification, outperforming VGG-16. Study [20] applied Faster R-CNN with VGG-16 transfer learning to detect surface defects on a bridge in Hunan Province, China. Study [21] enhanced YOLOv5s crack detection by refining feature extraction, incorporating attention mechanisms, and applying transfer learning, while K-means clustering optimized anchor boxes. Study [22] combined DCGAN with YOLOv5s for crack classification and used Otsu and mid-axis algorithms for quantitative analysis, significantly boosting accuracy and mAP. Study [23] proposed an improved YOLOv3 for detecting cracks and exposed steel bars on bridge surfaces.

Although the above studies yield promising results, flaws still exist. Table I outlines the main innovations of this study compared with recent works.

III. Methodology

A. Process Framework

Fig. 1 depicts the developed bridge defect detection framework, comprising three key components: a systematic review, data collection, and defect identification. This study built a dataset of over 3,700 bridge defect images from Guizhou Province, annotated using LabelImg for rapid identification. Subsequently, using the Roboflow tool, we applied flipping, cropping, translation, rotation, brightness adjustment, and noise addition to generate 700 augmented images, expanding the dataset to over 4,400 images [24]–[26]. Simultaneously, an innovative bridge defect detection model, YOLOv8-CBAM-Wise-IoU, is proposed and compared with leading one-stage and two-stage models (YOLOv8x, RetinaNet, and Faster R-CNN). Its performance is evaluated by precision, recall, and F1 score.

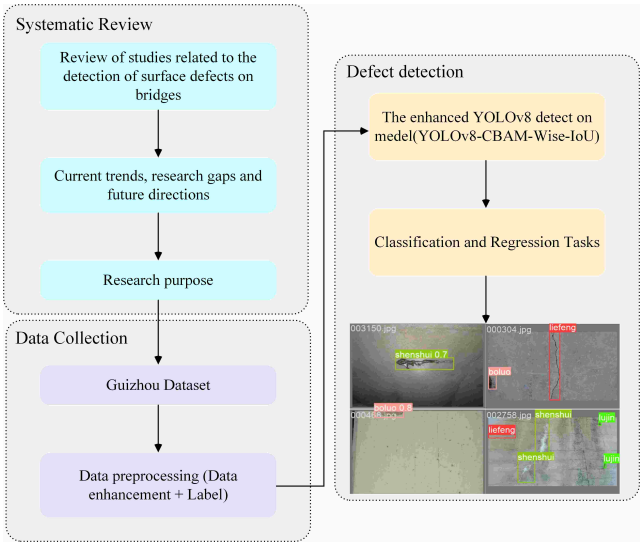

Fig. 1. Designed a framework to identify defects in bridges.

B. YOLOv8 Improvements

This study adopts YOLOv8 as its primary framework due to its simplified design, fewer hyperparameters, and enhanced training and inference efficiency [27], [28]. In real-world applications, bridge defect images often have complex backgrounds (e.g., textures, shadows, water stains) that introduce significant noise, which can result in false alarms or missed defects. To address these issues, YOLOv8 is enhanced with a CBAM attention mechanism and Wise-IoU loss, yielding the YOLOv8-CBAM-Wise-IoU model for multi-type surface defect detection. The model's architecture is shown in Fig. 2.

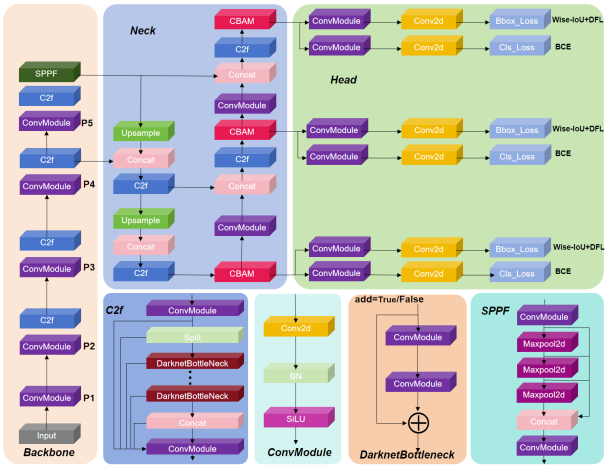

Fig. 2. Design framework for the YOLOv8-CBAM-Wise-IoU model.

The proposed model enhances accuracy and performance by replacing the conventional CIoU with Wise-IoU, which uses a gradient gain allocation strategy to reduce the impact of low-quality instances and mitigate competition among high-quality anchors, focusing on average-quality anchors. To further improve feature extraction and fusion, CBAM is integrated into the Neck module. As a novel attention mechanism, CBAM dynamically adjusts weights across spatial

TABLE I
MAIN INNOVATIONS OF THIS PAPER COMPARED TO RELATED LITERATURE

Feature	Traditional Methods	DL Methods	Proposed Methods
Defect Type	Single defect(crack)	More single defects(crack), fewer multiple defects	Multiple defects (7 types)
Generalization Ability	Weak	Moderate	Strong (complex backgrounds)
Feature Extraction	Relying on Handcrafted Features	Automatic feature extraction using CNN	Enhanced by CBAM (channel + spatial attention)
Data Diversity	Limited	Moderate	High (augmented, diverse datasets)

and channel dimensions of feature maps, enhancing CNN performance, perceptual accuracy, and generalization [29]. The combined use of CBAM and Wise-IoU significantly improves the model's ability to detect, identify, and locate bridge surface defects.

C. The CBAM Module

The module is composed of two sequential components: channel attention and spatial attention. As shown in Fig. 3.

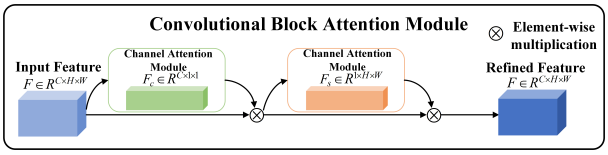

Fig. 3. Structure of the Convolutional Block Attention Module Integrating Channel and Spatial Attention.

This CBAM module adaptively improves the intermediate feature maps at each convolutional block in deep neural networks. Taking the intermediate feature map $\mathrm{F} \in \mathrm{R}^{\mathbb{C}\times \mathrm{H}\times \mathrm{W}}$ as input, CBAM calculates the 1D channel attention map $F_c \in R^{C\times 1\times 1}$ followed by the 2D spatial attention map $F_s \in R^{1\times H\times W}$ in a sequential manner. The entire attention process is depicted in equation (1):

$$\begin{aligned} F' &= F_c(F) \otimes F \\ F'' &= F_s(F') \otimes F' \end{aligned} \tag{1}$$

Where $\otimes$ represents the operation of element-wise multiplication. This process involves element-wise multiplication, where attention values are distributed according to their dimensions: channel attention is expanded spatially, and vice versa. The final output, F'' , is the refined result.

The channel attention mechanism module aims to improve the feature representation of each channel. As depicted in Fig. 4, for the input feature map, spatial information is first captured through average pooling and max pooling operations. These operations produce two separate spatial context descriptors, referred to as F^c_{avg} and F^c_{max}.

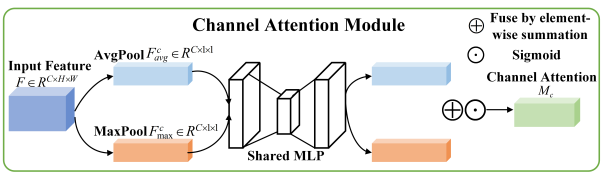

Fig. 4. Channel Attention Mechanism: Using Global Pooling and Shared MLP to Generate Attention Weights.

Then, through the Sigmoid activation, the output of the convolutional layer is transformed into the weight factor $M_c(F)$, within the range of (0-1). Ultimately, the weight factors are applied element-wise to the original input, resulting in a weighted feature map. This process enables the model to give greater attention to the more informative channels. The channel attention calculation formula is:

$$\begin{aligned} M_c(F) &= \sigma\left(\mathrm{MLP}\left(\mathrm{AvgPool}(F)\right) + \mathrm{MLP}\left(\mathrm{MaxPool}(F)\right)\right) \\ &= \sigma\left(\mathrm{Conv}(F^c_{avg}) + \mathrm{Conv}(F^c_{max})\right) \end{aligned} \tag{2}$$

Where σ is the Sigmoid activation and Conv is a convolution operation. The MLP consists of a hidden layer in its architecture.

The spatial attention mechanism module focuses on emphasizing or diminishing the significance of specific spatial regions by assigning weights to the input feature map. As shown in Fig. 5, to compute spatial attention, this study combines the channel information from the feature map using two pooling operations, resulting in two 2D maps: $F^s_{avg} \in R^{1\times H\times W}$ and $F^s_{max} \in R^{1\times H\times W}$, they reflect the features obtained by average pooling and max pooling across the channels.

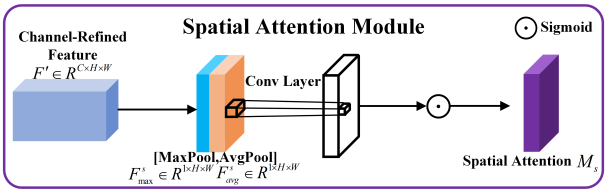

Fig. 5. Spatial Attention Mechanism: Highlighting Informative Regions in Feature Maps.

The extracted features are subsequently merged and processed through standard convolutional layers, resulting in the generation of a 2D spatial attention map, $M_s \in R^{H\times W}$. The formula for the spatial attention operation is:

$$\begin{aligned} M_s(F) &= \sigma\left(\mathrm{Conv}\left([\mathrm{AvgPool}(F); \mathrm{MaxPool}(F)]\right)\right) \\ &= \sigma\left(\mathrm{Conv}\left([F^s_{avg}; F^s_{max}]\right)\right) \end{aligned} \tag{3}$$

Where σ is the Sigmoid activation and Conv is a convolution operation.

D. The Wise-IoU Loss Function

This research uses the Wise-IoU loss function [30] for bounding box regression, offering two key benefits. Firstly, it addresses poor-quality training samples by amplifying penalties for geometric factors like distance and aspect ratio, thereby improving the model's generalization ability.

To accomplish this, the authors propose a distance-based attention mechanism, which serves as the foundation for creating a Wise-IoU loss function that incorporates dual attention

layers. The mathematical formulation of the Wise-IoU loss is given in (4).

$$\begin{aligned} L_{\text{WIoU}} &= R_{\text{WIoU}} L_{\text{IoU}} \\ R_{\text{WIoU}} &= \exp\left(\frac{(x-x_{\text{gt}})^2+(y-y_{\text{gt}})^2}{(W_g^2+H_g^2)^*}\right) \\ f_{\text{loss}} &= \lambda_1 f_{\text{BCEL}} + \lambda_2 f_{\text{DFL}} + \lambda_3 f_{\text{WIoU}} \end{aligned} \tag{4}$$

In this context, W_g and H_g denote the measurements of the smallest enclosing box. To avoid R_{WIoU} from slowing down rate of convergence, the computation graph excludes W_g and H_g (indicated using * as a superscript), effectively eliminating factors that impede convergence. $R_{WIoU} \in [1, e$ greatly increases the L_{IoU} of standard quality anchor boxes. $L_{IoU} \in [1, e)$, in turn, reduces the R_{WIoU} of high-quality anchor boxes significantly, especially when their alignment with the target box is close to the center point. The total loss equation incorporates the Wise-IoU loss function to enhance bounding box regression by addressing geometric challenges and reducing the impact of poor-quality instances during training.

IV. Experiments

A. Experimental Condition

The model was trained on a workstation with a GeForce RTX 4090 GPU, Intel Core i9-13900K processor (3.00 GHz), 31.8 GB RAM, and Windows 10. Full hardware and software details are in Table II.

TABLE II
HARDWARE AND SOFTWARE SETUP

Hardware	Description	Software	Description
GPUs	GeForce RTX 4090	Operating System	Windows 10
CPU	Intel(R) Core(TM) i9-13900K 3.00 GHz	Deep Learning Framework	Pytorch
RAM	31.8G	CUDA Version	12.1

B. Data Collection

To achieve rapid identification of bridge surface defects, this study collected over 4,400 images of bridges from Guizhou Province and annotated them using LabelImg. The materials in the data collection are primarily concrete, with a smaller portion consisting of asphalt and steel. The dataset comprises 3,527 training images, 440 validation images, and 440 test images, covering seven defect categories: crack, exposed reinforcement, honeycomb, hole, hungry spots, breakage and seepage.

To improve the model's generalization ability, the Roboflow data augmentation tool was used to expand the dataset through various strategies such as flipping, cropping, translation, rotation, brightness adjustment, and noise addition, generating a total of 700 augmented images. The dataset also includes images captured under different lighting conditions, featuring elements such as paint, expansion joints, water stains, and dry moss, which further increase the complexity and diversity of the data.

Fig. 6 shows examples of images after data augmentation. Specifically, the top row illustrates three examples: the original image, the result after applying horizontal mirroring, and the result after applying vertical mirroring combined with brightness adjustment and noise addition. The bottom row displays another set of images: the original image, the image after brightness adjustment, and the image after both cutout and brightness adjustment. These augmentation techniques effectively simulate various real-world scenarios, enhancing the robustness of the model against environmental variations. Additionally, Fig. 9 provides image samples for each analyzed class.

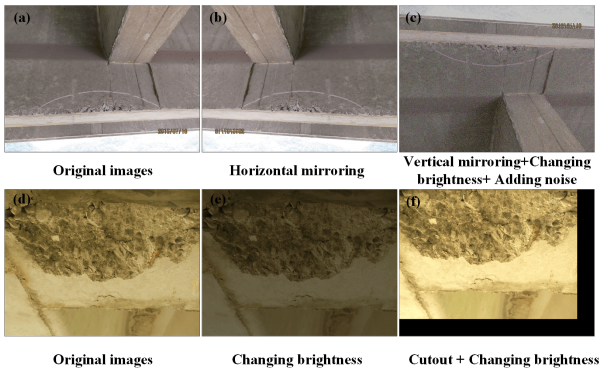

Fig. 6. Examples of data enhancement techniques applied to concrete damage images. (a) Original image, (b) horizontal mirroring, (c) vertical mirroring with brightness adjustment and added noise, (d) original image, (e) brightness adjustment, and (f) cutout with brightness adjustment.

C. Parameter Setting

In this study, extensive pre-experiments and hyperparameter tuning were conducted to compare the model's convergence speed, stability, and final performance under various settings. The final hyperparameter configuration was selected as follows: an SGD optimizer with a learning rate of 0.01 to prevent oscillation while maintaining stability and speed; momentum set to 0.937 for smoother updates; weight decay of 0.005 to reduce overfitting and enhance generalization; a batch size of 16 as a balance between efficiency and stability; and an input size of 640×640 to balance accuracy and computational cost. This configuration achieved the best performance across multiple experiments.

D. Experimental Comparison

Ablation experiments evaluated different module combinations in the YOLOv8-CBAM-Wise-IoU model using the same dataset, with YOLOv8m as the baseline for comparing attention mechanisms and IoU loss functions. Finally, the YOLOv8-CBAM-Wise-IoU model was compared with popular one-stage and two-stage models to assess its performance.

E. Evaluation Metrics

This paper evaluates the YOLOv8-CBAM-Wise-IoU model using Precision, Recall, F1-score, and mAP.

Before calculating mAP, precision, recall, and F1-score must be computed, as shown in (5). Precision measures detection

accuracy, recall evaluates the model's ability to identify relevant instances, and the F1-score balances precision and recall as their harmonic mean [31]–[33].

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN} \\ F1-score &= \frac{2 \times Precision \times Recall}{Precision + Recall} \end{aligned} \tag{5}$$

TP (True Positive) refers to correctly detected defect bounding boxes, FP (False Positive) to incorrectly identified defects, and FN (False Negative) to missed defects.

Average Precision (AP) and Mean Average Precision (mAP) are defined as demonstrated in (6).

$$\begin{aligned} AP &= \int_0^1 P(R)\, dr \\ MAP &= \frac{1}{N} \sum_{i=1}^{N} AP \end{aligned} \tag{6}$$

P denotes precision, R represents recall, and N is the total number of classes. The model's performance is assessed using the IoU metric, which measures how well the predicted bounding box overlaps with the ground truth. The IoU is calculated as the ratio of the overlapping area to the combined area of the bounding boxes. A threshold is set to determine detection validity: if IoU exceeds the threshold, the detection is valid; otherwise, it is invalid [34]. The IoU computation formula is shown in (7).

$$IoU = \frac{areaofoverlap}{areaofunion} \tag{7}$$

mAP is a metric that averages AP across various IoU thresholds. For example, mAP@[0.5:0.05:0.95] computes AP at IoU thresholds from 0.5 to 0.95 in increments of 0.05, providing a comprehensive evaluation of detection performance across thresholds [35], [36]. Additionally, mAP is often calculated for specific IoU thresholds, such as mAP@0.5, which assesses average precision at an IoU of 0.5, offering targeted insights into algorithm performance for specific applications.

V. Results

A. Ablation Experiments

In this paper, YOLOv8m serves as the baseline model, with different modules tested on the same dataset and parameters. As shown in Table III, adding the CBAM attention mechanism increases recall to 0.71 but reduces precision, mAP50, and mAP50-95, indicating a higher defect detection likelihood. The addition of Wise-IoU improves precision and mAP50-95 to 0.981 and 0.314, respectively, but reduces mAP50, suggesting enhanced accuracy and fewer false positives. Combining CBAM and Wise-IoU in YOLOv8-CBAM-Wise-IoU boosts precision by 5% over YOLOv8m and 1.8% over CBAM alone, achieving the highest recall(0.76),F1-score (0.58), mAP50(0.554), and mAP50-95(0.324).

Additionally, this paper includes statistics on mAP50 and mAP50-95 for each training epoch, illustrated in Fig. 7. The

TABLE III
Ablation Experiments

YOLOv8m	CBAM	Wise-IoU	P	R	F1-score	mAP50	mAP50-95
✓	×	×	0.929	0.66	0.55	0.531	0.296
✓	✓	×	0.961	0.71	0.52	0.479	0.252
✓	×	✓	0.981	0.62	0.54	0.537	0.314
✓	✓	✓	0.979	0.76	0.58	0.554	0.324

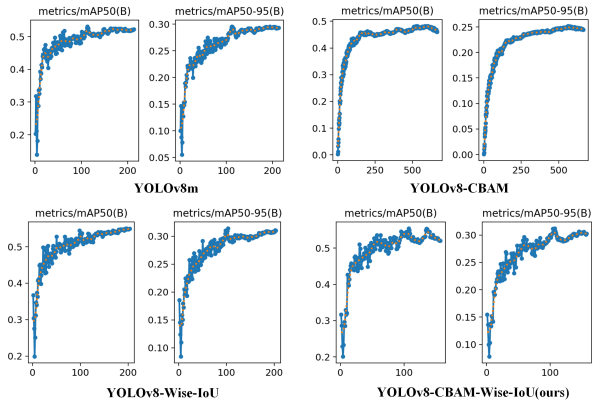

Fig. 7. mAP50 and mAP50-95 training curves for YOLOv8m, YOLOv8-CBAM, YOLOv8-Wise-IoU, and YOLOv8-CBAM-Wise-IoU (ours). Each model shows mAP50 (left) and mAP50-95 (right).

figure is divided into four groups for four models: YOLOv8m, YOLOv8-CBAM, YOLOv8-Wise-IoU, and YOLOv8-CBAM-Wise-IoU (ours). Each group has two subplots: the left subplot shows mAP50 trends across epochs, and the right subplot shows mAP50-95 trends. For YOLOv8m, both mAP50 and mAP50-95 rise quickly and stabilize around 200 epochs. YOLOv8-CBAM shows smoother curves but requires over 500 epochs to stabilize, prolonging training time. YOLOv8-Wise-IoU converges swiftly within about 200 epochs but with slight fluctuations compared to YOLOv8m. Finally, YOLOv8-CBAM-Wise-IoU achieves the most significant improvement: both mAP50 and mAP50-95 reach peak values and converge rapidly within 200 epochs, demonstrating superior detection performance and the fastest convergence speed among all variants.

Thus, it is evident that the proposed YOLOv8-CBAM-Wise-IoU model not only achieved the highest mAP50 and mAP50-95 in ablation experiments but also required the least training time.

This research evaluates the proposed model using additional metrics, including P-C, R-P, F1-C, and R-C curves, providing deeper insights into defect detection and categorization. The P-R and F1-C curves illustrate the trade-off between precision and recall at varying thresholds, while the P-C curve demonstrates that precision generally improves with increasing confidence levels.

Fig. 8 shows the performance diagnostic curve for YOLOv8-Wise-IoU. In Subfigure (a), precision increases steadily with confidence, reaching nearly 1.0 at a score of 0.979. Subfigure (b) indicates that recall gradually decreases as the confidence threshold rises, peaking at 0.76 for lower thresholds. Subfigure (c) reveals the model's highest F1 score of 0.58 at a confidence threshold of 0.276. Subfigure (d) demonstrates a comprehensive P-R curve, indicating strong de-

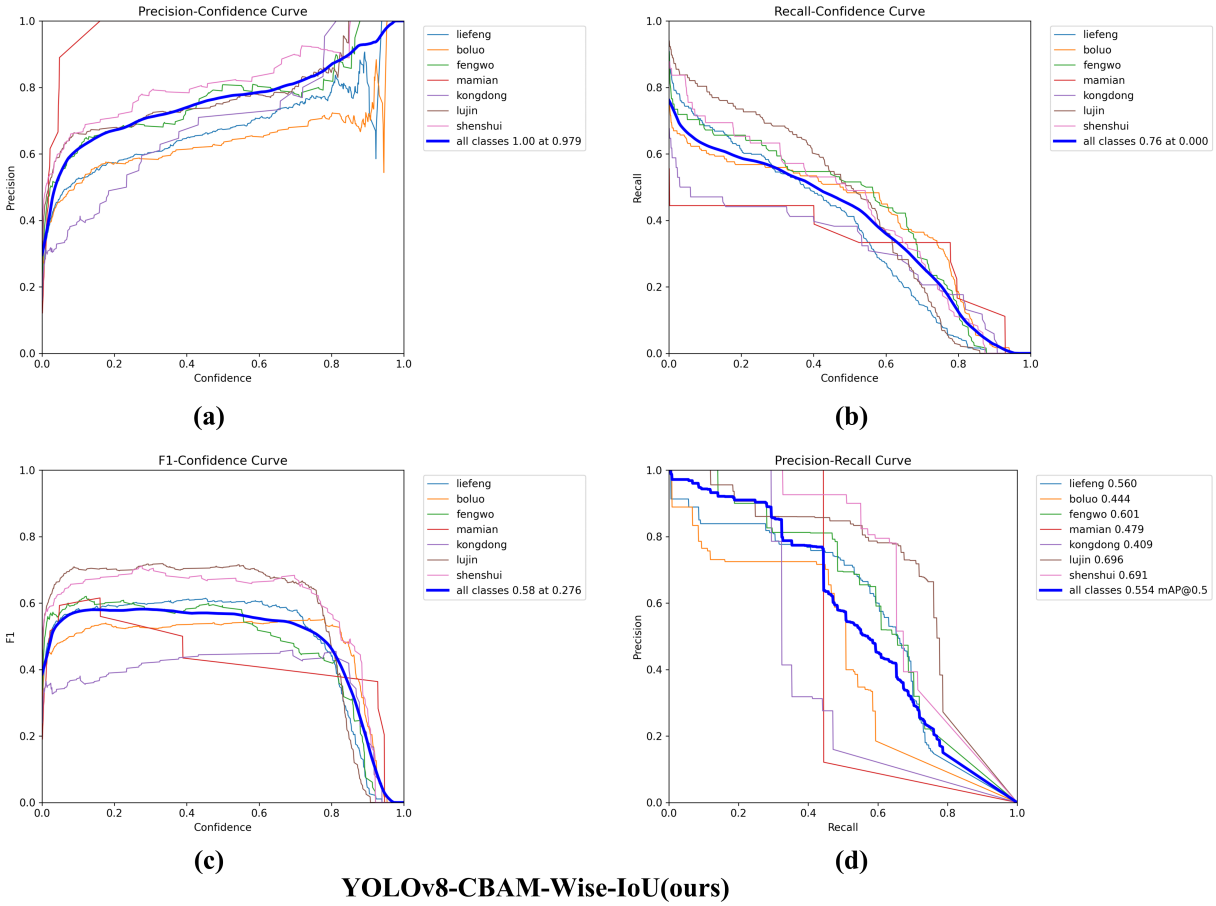

Fig. 8. Performance diagnostic curves of YOLOv8-CBAM-Wise-IoU (ours): (a) precision–confidence curve, (b) recall–confidence curve, (c) F1–confidence curve, and (d) precision–recall curve.

tection accuracy and a balanced precision-recall relationship. These results confirm the effectiveness and robustness of the proposed method.

Additionally, this study uses Gradient-weighted Class Activation Mapping (Grad-CAM) [37] to visualize features through heatmaps based on class activation. Grad-CAM computes classification gradients from the final convolutional feature map, highlighting critical features for classification, with stronger gradients shown in red and weaker ones in blue. Larger gradients correspond to redder areas, indicating greater impact on classification. Fig. 9 displays the heatmaps for the YOLOv8m and YOLOv8-CBAM models with the CBAM attention mechanism.

Fig. 9 illustrates how the YOLOv8-CBAM model prioritizes defect regions by focusing on key areas to extract crucial information. The figure displays heatmaps for seven types of defects: crack, reinforcement, comb, hole, hungry spots, breakage, and seepage. Each defect type is shown with the input image in the first row, followed by attention heatmaps from both the YOLOv8-CBAM and YOLOv8m models. YOLOv8-CBAM generates more concentrated and accurate heatmaps, with stronger attention precisely on defect regions, whereas YOLOv8m produces more dispersed attention areas.

The heatmap comparison highlights that the CBAM module shifts the model's focus towards defects. In conclusion, the CBAM module enhances defect detection by leveraging both local and global features, improving the representation of critical regions.

B. Comparative Study of Attention and Loss Functions

To validate the performance improvement of the CBAM attention mechanism with the Wise-IoU loss function, this paper combined Channel Attention (CA) [38], Squeeze-and-Excitation (SE) [39], and Global Attention Mechanism (GAM) [40] with CIoU, DIoU, and Wise-IoU for comparative experiments. The results are shown in Table IV.

As shown in Table IV, the YOLOv8-CBAM-Wise-IoU model ranks highest in F1-score, mAP50, and mAP50-95, demonstrating the effectiveness of the proposed approach.

The CBAM attention mechanism enhances representation in both the channel and spatial dimensions, improving local information capture with minimal computational cost. Compared to other global attention mechanisms, CBAM offers a better balance between complexity and performance, addressing the limitations of CA, SE, and GAM for more efficient feature enhancement.

The Wise-IoU loss function improves upon CIoU and DIoU by considering center distances, aspect ratios, and bounding box size differences with confidence levels. It balances com-

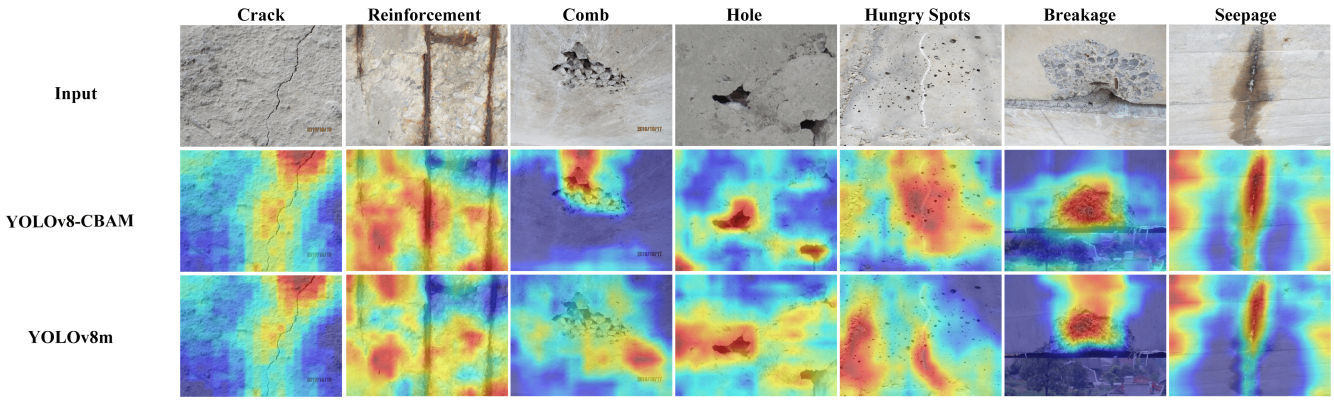

Fig. 9. Heatmap comparison between YOLOv8-CBAM and YOLOv8m across seven defect types. The top row shows input images; the second and third rows show the corresponding Grad-CAM results.

TABLE IV
COMPARATIVE EXPERIMENTS ON DIFFERENT ATTENTION MECHANISMS AND IOU FUNCTIONS

Model	P	R	F1-score	mAP50	mAP50-95
YOLOv8-CA-CIoU	0.910	0.78	0.54	0.521	0.279
YOLOv8-CA-DIoU	0.931	0.66	0.55	0.517	0.286
YOLOv8-CA-Wise-IoU	0.951	0.64	0.56	0.534	0.290
YOLOv8-SE-CIoU	0.957	0.65	0.58	0.538	0.304
YOLOv8-SE-DIoU	0.960	0.63	0.58	0.535	0.306
YOLOv8-SE-Wise-IoU	0.978	0.63	0.57	0.540	0.310
YOLOv8-GAM-CIoU	0.972	0.67	0.55	0.544	0.301
YOLOv8-GAM-DIoU	0.981	0.62	0.54	0.538	0.310
YOLOv8-GAM-Wise-IoU	0.967	0.66	0.57	0.548	0.311
YOLOv8-CBAM-CIoU	0.961	0.71	0.52	0.479	0.252
YOLOv8-CBAM-DIoU	0.952	0.65	0.56	0.535	0.306
ours	0.979	0.76	0.58	0.554	0.324

TABLE V
COMPARATIVE EXPERIMENTS OF BENCHMARK MODELS

Model	P	R	F1-score	mAP50	mAP50-95	parameters	GFLOPs
YOLOv8x	0.978	0.63	0.57	0.539	0.312	130 M	258.1
Faster R-CNN	0.271	0.52	0.36	0.344	0.127	60.37 M	182
Retina Net	0.761	0.33	0.44	0.462	0.214	145 M	97
ours	0.979	0.76	0.58	0.554	0.324	49.6 M	79.5

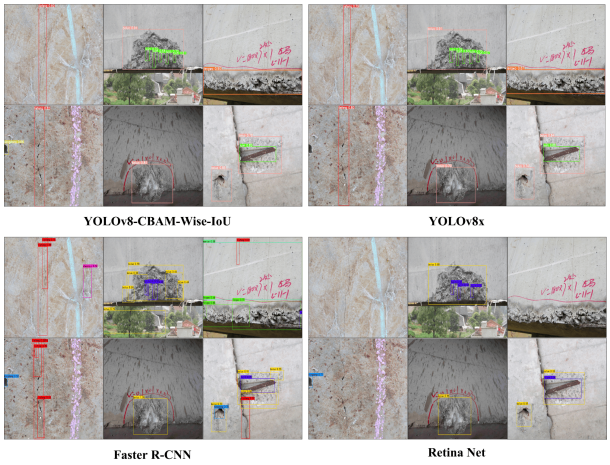

Fig. 10. Detection results of different models on structural defect images. From top left to bottom right: YOLOv8-CBAM-Wise-IoU, YOLOv8x, Faster R-CNN, and RetinaNet.

plex and simple samples, enhancing optimization and training effectiveness, leading to a more robust target detection model.

This study integrates the CBAM module and Wise-IoU loss into YOLOv8, resulting in the YOLOv8-CBAM-Wise-IoU model, which outperforms baseline models, achieving the highest Precision, Recall, F1-score, mAP50, and mAP50-95.

C. Benchmark Comparisons

The proposed model was compared with current mainstream detection models Faster R-CNN, Retina Net and YOLOv8x using the same parameter configuration. Table V presents the results of the experiments.

As shown in Table V, the proposed YOLOv8-CBAM-Wise-IoU model, compared with mainstream models (Faster R-CNN, RetinaNet, and YOLOv8x) under the same parameters, achieves comparable detection performance to YOLOv8x in precision (0.979 vs. 0.978) and F1-score (0.58 vs. 0.57). However, it offers significant advantages in model size, inference speed, and computational complexity, making it more efficient and practical.

With only 49.6M parameters, compared to YOLOv8x (130M) and RetinaNet (145M), the model reduces memory requirements and improves deployability in resource-constrained environments. Its faster inference speed, resulting from a 69% reduction in GFLOPs (79.5 vs. 258.1), ensures real-time performance. The integration of CBAM and Wise-IoU enhances feature extraction and bounding box refinement with minimal overhead.

In summary, while maintaining detection performance close to YOLOv8x, the proposed model's smaller size, faster inference, and lower computational cost make it a superior choice for real-world applications, especially in resource-limited settings.

Fig. 10 shows detection results of bridge surface defects using YOLOv8-CBAM-Wise-IoU, YOLOv8x, Faster R-CNN, and RetinaNet. Each group presents multiple samples across various defect types. YOLOv8-CBAM-Wise-IoU achieves more accurate localization and classification with tighter boxes and fewer misses. In contrast, YOLOv8x and Faster R-CNN show more redundant or imprecise boxes, while RetinaNet occasionally misclassifies small defects.

D. Discussion

The YOLOv8-CBAM-Wise-IoU model significantly improves bridge surface flaw detection. CBAM enhances feature representation across spatial and channel dimensions, capturing localized information and boosting detection precision. Wise-IoU optimizes bounding box comparison by considering center distance, aspect ratio, width, height, and confidence, balancing complex and simple samples to enhance training efficiency.

Extensive experiments validate its effectiveness. Ablation studies show CBAM increases recall by reducing missed detections, while Wise-IoU improves precision by minimizing false positives. The enhanced YOLOv8 achieves 97.9% precision, 76% recall, 58% F1-score, 55.4% mAP50, and 32.4% mAP50-95.

Comparative experiments with Faster R-CNN, RetinaNet, and YOLOv8x confirm the model's superiority, though the 76% recall rate indicates potential missed detections in complex scenarios.

VI. Conclusion

This paper presents the YOLOv8-CBAM-Wise-IoU model for detecting multiple defects on bridge surfaces, achieving impressive results: precision (97.9%), recall (76%), F1-score (58%), mAP50 (55.4%), and mAP50-95 (32.4%). These results outperform other one-stage and two-stage models, including Faster R-CNN, RetinaNet, and YOLO versions, ranking first in performance. The model's success stems from key factors: YOLOv8 ensures fast, efficient detection; CBAM improves generalization by adjusting channel and spatial weights adaptively; and Wise-IoU loss enhances defect detection and bounding box localization, addressing challenges with low-quality training samples. Together, these components enhance precision and robustness.

Future efforts will focus on enhancing detection accuracy and robustness by addressing current limitations, including multi-scale training, advanced data augmentation, and expanding the dataset with high-quality, diverse images and refined annotations to improve defect detection and enhance recall. Further advancements in feature representation through advanced attention mechanisms will be explored, alongside real-world deployment to assess scalability and effectiveness in bridge inspections. Additionally, techniques like KAN networks, GCN, and Mamba networks will be investigated to bolster robustness in complex scenarios. Leveraging semi-supervised and unsupervised learning methods will also be explored to reduce manual labeling dependency and address challenges such as the 76% recall rate and complexities in background variations.

Acknowledgments

This research was partially funded by the Guizhou Science and Technology Plan Project (QKTY[2024]017), the project "Research on the Quality Monitoring System for Master's Degree Programs in Electronic Information Based on Analytic Hierarchy Process" (2024JG01), the Guiyang City Science and Technology Plan Project ([2024]2-22), and the Natural Science Research Foundation of the Education Department of Guizhou Province (QJJ[2024]190). Additional support came from the Scientific Studies of Higher Education Institutions under the Guizhou Province Education Department (QEJ[2022]307, QEJ[2021]005), the Science and Technology Foundation of Guizhou Province (QKHJC-ZK[2023]012, QKHJC-ZK[2022]014), and the Doctoral Research Start-up Fund of Guiyang University (GYU-KY-2025). Further funding was provided by the National Natural Science Foundation of China (U23A20341) and the IER Foundation 2023 (IERF202304, IERF202205).

References

[1] M. Morgese, et al., "Distributed detection and quantification of cracks in operating large bridges," *J. Bridge Eng.*, vol. 29, no. 1, p. 04023101, 2024, doi: 10.1061/JBENF2.BEENG-6454.

[2] R. S. Tozetto, et al., "Development of a vibration measurement system for bridges," *IEEE Latin Amer. Trans.*, vol. 19, no. 5, pp. 790–797, 2021, doi: 10.1109/TLA.2021.9448313.

[3] Q. Liu, et al., "ViTR-Net: An unsupervised lightweight transformer network for cable surface defect detection and adaptive classification," *Eng. Struct.*, vol. 313, p. 118240, 2024, doi: 10.1016/j.engstruct.2024.118240.

[4] E. M. Abdelkader, T. Zayed, and N. Faris, "Synthesized evaluation of reinforced concrete bridge defects, their non-destructive inspection and analysis methods: a systematic review and bibliometric analysis of the past three decades," *Buildings*, vol. 13, no. 3, p. 800, 2023, doi: 10.3390/buildings13030800.

[5] J. P. Liu, et al., "Recent application of and research on concrete arch bridges in China," *Struct. Eng. Int.*, vol. 33, no. 3, pp. 394–398, 2023, doi: 10.1080/10168664.2022.2058441.

[6] D. Su, Y. S. Liu, X. T. Li, X. Y. Chen, and D. H. Li, "Management path of concrete beam bridge in China from the perspective of sustainable development," *Sustainability*, vol. 12, no. 17, p. 7145, 2020, doi: 10.3390/su12177145.

[7] Z. Nie, et al., "Damage Detection in Bridge via Adversarial-Based Transfer Learning," *Struct. Control Health Monit.*, 2024, Art no. 5548218, doi: 10.1155/stc/5548218.

[8] E. Bianchi and M. Hebdon, "Visual structural inspection datasets," *Autom. Constr.*, vol. 139, p. 104299, 2022, doi: 10.1016/j.autcon.2022.104299.

[9] H. Zoubir, et al., "Pixel-level concrete bridge crack detection using Convolutional Neural Networks, gabor filters, and attention mechanisms," *Eng. Struct.*, vol. 314, p. 118343, 2024, doi: 10.1016/j.engstruct.2024.118343.

[10] Y. Wang, et al., "Real-time spatial contextual network based on deep learning for bridge exposed rebar segmentation," *Constr. Build. Mater.*, vol. 449, p. 138379, 2024, doi: 10.1016/j.conbuildmat.2024.138379.

[11] B. Sezen and C. C. Cerasi, "Solar cell busbars surface defect detection based on deep convolutional neural network," *IEEE Latin Amer. Trans.*, vol. 21, no. 2, pp. 242–250, 2023, doi: 10.1109/TLA.2023.10015216.

[12] C. Xiang, A. Chen, and D. Wang, "Real-time stress-based topology optimization via deep learning," *Thin-Walled Struct.*, vol. 181, p. 110055, 2022, doi: 10.1016/j.tws.2022.110055.

[13] T. S. Tran, S. D. Nguyen, H. J. Lee, et al., "Advanced crack detection and segmentation on bridge decks using deep learning," *Constr. Build. Mater.*, vol. 400, p. 132839, 2023.

[14] X. Zhang, et al., "Intelligent surface cracks detection in bridges using deep neural network," *Int. J. Struct. Stab. Dyn.*, vol. 24, no. 5, p. 2450046, 2024, doi: 10.1142/S0219455424500469.

[15] I. Abdel-Qader, O. Abudayyeh, M. Asce, and M. E. Kelly, "Analysis of edge-detection techniques for crack identification in bridges," *J. Bridge Eng.*, vol. 17, pp. 255–263, 2003, doi: 10.1061/(ASCE)0887-3801(2003)17:4(255).

[16] J. H. Lee, J. M. Lee, H. J. Kim, and Y. S. Moon, "Machine vision system for automatic inspection of bridges," in *Proc. 1st Int. Congr. Image Signal Process.*, vol. 3, pp. 363–366, 2008, doi: 10.1109/CISP.2008.672.

[17] V. Vivekananthan, R. Vignesh, S. Vasanthaseelan, E. Joel, and K. S. Kumar, "Concrete bridge crack detection by image processing technique by using the improved OTSU method," *Mater. Today: Proc.*, vol. 74, pp. 1002–1007, 2023, doi: 10.1016/j.matpr.2022.11.356.

[18] F. Scozzese and A. Dall'Asta, "Nonlinear response characterization of post-tensioned RC bridges through Hilbert–Huang transform analysis," *Struct. Control Health Monit.*, 2024, doi: 10.1155/2024/5960162.
[19] G. Tan and Z. Yang, "Autonomous bridge detection based on ResNet for multiple damage types," in *Proc. 2021 IEEE 11th Annu. Int. Conf. CYBER Technol. Autom. Control Intell. Syst. (CYBER)*, pp. 555–559, 2021, doi: 10.1109/CYBER53097.2021.9588187.
[20] R. Li, et al., "Automatic bridge crack detection using unmanned aerial vehicle and Faster R-CNN," *Constr. Build. Mater.*, vol. 362, 2023, doi: 10.1016/j.conbuildmat.2022.129659.
[21] Z. Yu, "YOLO V5s-based deep learning approach for concrete cracks detection," *SHS Web Conf.*, vol. 144, p. 03015, 2022, doi: 10.1051/shsconf/202214403015.
[22] Y. Ni, J. Mao, H. Wang, Z. Xi, and Y. Xu, "Toward high-precision crack detection in concrete bridges using deep learning," *J. Perform. Constr. Facil.*, vol. 37, no. 3, 2023, doi: 10.1061/JPCFEV.CFENG-4275.
[23] S. Teng, Z. Liu, and X. Li, "Improved YOLOv3-based bridge surface defect detection by combining high-and low-resolution feature images," *Buildings*, vol. 12, no. 8, p. 1225, 2022, doi: 10.3390/buildings12081225.
[24] Y. A. S. M. Chen, et al., "Deep learning based underground sewer defect classification using a modified RegNet," *Comput. Mater. Continua*, vol. 75, no. 3, pp. 5455–5473, 2023, doi: 10.32604/cmc.2023.033787.
[25] Y. H. Ni, et al., "Quantitative detection of typical bridge surface damages based on global attention mechanism and YOLOv7 network," *Struct. Health Monit.*, vol. 24, no. 2, pp. 941–962, 2025, doi: 10.1177/14759217241246953.
[26] K. Luo, et al., "Computer vision-based bridge inspection and monitoring: A review," *Sensors*, vol. 23, no. 18, p. 7863, 2023, doi: 10.3390/s23187863.
[27] G. Oh and S. Lim, "One-stage brake light status detection based on YOLOv8," *Sensors*, vol. 23, no. 17, pp. 1–18, 2023, doi: 10.3390/s23177436.
[28] D. Reis, et al., "Real-time flying object detection with YOLOv8," *arXiv preprint arXiv:2305.09972*, 2023, doi: 10.48550/arXiv.2305.09972.
[29] S. Woo, et al., "Cbam: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 618–626, 2018, doi: 10.48550/arXiv.1807.06521.
[30] Z. Tong, Y. Chen, Z. Xu, and R. Yu, "Wise-IoU: Bounding box regression loss with dynamic focusing mechanism," *arXiv preprint arXiv:2301.10051*, 2023, doi: 10.48550/arXiv.2301.10051.
[31] S. M. Park, J. H. Lee, and L. S. Kang, "A framework for improving object recognition of structural components in construction site photos using deep learning approaches," *KSCE J. Civil Eng.*, vol. 27, no. 1, pp. 1–12, 2023, doi: 10.1007/s12205-022-2318-0.
[32] J. Zhang, X. Yang, W. Li, S. Zhang, and Y. Jia, "Automatic detection of moisture damages in asphalt pavements from GPR data with deep CNN and IRS method," *Autom. Constr.*, vol. 113, p. 103119, 2020, doi: 10.1016/j.autcon.2020.103119.
[33] D. Valencia, E. Muñoz España, and M. Muñoz Añasco, "Impact of the preprocessing stage on the performance of offline automatic vehicle counting using YOLO," *IEEE Latin Amer. Trans.*, vol. 22, no. 9, pp. 723–732, 2024, doi: 10.1109/TLA.2024.10669248.
[34] K. Su, et al., "N-IoU: better IoU-based bounding box regression loss for object detection," *Neural Comput. Appl.*, vol. 36, no. 6, pp. 3049–3063, 2024, doi: 10.1007/s00521-023-09133-4.
[35] M. Hussain, "Yolov1 to v8: Unveiling each variant–a comprehensive review of YOLO," *IEEE Access*, vol. 12, pp. 42816–42833, 2024, doi: 10.1109/ACCESS.2024.3378568.
[36] T. Li, G. Liu, and S. Tan, "Superficial Defect Detection for Concrete Bridges Using YOLOv8 with Attention Mechanism and Deformation Convolution," *Appl. Sci.*, vol. 14, no. 13, p. 5497, 2024, doi: 10.20944/preprints202405.1498.v1.
[37] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 618–626, 2017, doi: 10.1007/s11263-019-01228-7.
[38] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, doi: 10.48550/arXiv.2103.02907.
[39] J. Hu, S. Li, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, doi: 10.48550/arXiv.2308.13343.
[40] Y. Liu, Z. Shao, and N. Hoffmann, "Global attention mechanism: Retain information to enhance channel-spatial interactions," *arXiv preprint arXiv:2112.05561*, 2021, doi: 10.48550/arXiv.2112.05561.

Ruiping Li (Student Member, IEEE), received the BS degree in the College of Computer Science, Guiyang University, Guiyang, China, in 2023. He is currently a master's student at the School of Computer Science, Guiyang University. His research interests include machine vision, defect detection, neural network model design, etc.

Linchang Zhao (Member, IEEE), received the Ph.D. degree from the College of Computer Science, Chongqing University, China. He earned the BS degree from the College of Computer Science, Northeast Petroleum University, Daqing, China, in 2013, and the Master's degree from the School of Mathematics and Statistics, Qiannan Normal College for Nationalities, Duyun, China, in 2017. He is currently an associate professor at the College of Computer Science, Guiyang University. His research interests include pattern recognition, machine learning, image processing, and deep learning.

Hao Wei (Student Member, IEEE), received the BS degree in the School of Artificial Intelligence, Chongqing University of Technology Chongqing, China, in 2023. He is currently a master's student at the School of Computer Science, Guiyang University. His research interests include machine vision, defect detection, machine learning, image processing, etc.

Bocheng OuYang (Member, IEEE), professor and master tutor at Guiyang University, is a key figure in Guiyang's "Hundred People Plan" for big data, a high-level innovation talent in Guizhou, and a member of the Guizhou Computer Society's work committee and the Guizhou Big Data Bureau standard committee. He has led and participated in over 20 projects, published more than 20 papers, obtained 10 utility model patents, 3 software copyrights, and edited 2 textbooks.

Bing Fang (Member, IEEE), associate professor at the College of Computer Science, Guiyang University, is a member of the industry-university-research committee of the Guizhou Computer Society. He has led over 3 projects and published more than 5 papers. His research interests include image processing and machine learning.

Yongchi Xu (Member, IEEE), Ph.D. in Engineering from Peking University, is an associate professor and graduate supervisor at Guiyang University's School of Electronics and Communication Engineering. He teaches courses like Introduction to Communication Engineering, Optical Access Technology, and Radio Frequency Communication Circuit Design. His research focuses on Microwave Photonics. Xu has led or participated in five national and provincial projects, published over 10 papers, and holds more than 10 invention patents.

Guoqing Hu (Member, IEEE), holds a Ph.D. and is an associate researcher and senior engineer at the Shenzhen-Hong Kong Institute of Productive Studies, Peking University Shenzhen Graduate School, and the Postgraduate School (Peking University-Hong Kong University of Science and Technology Shenzhen Institute). He leads the 5G & 6G research team and serves as an independent director at Xin Tianxia Technology Co., Ltd.