Wiley
Structural Control and Health Monitoring
Volume 2025, Article ID 9952459, 18 pages
https://doi.org/10.1155/stc/9952459

WILEY

Research Article

YOLODF: A Concrete Bridge Surface Damage Detection Model Based on Multiscale Feature Fusion in Complex Environments

Lingyun Li[1,2] **Maria Rashidi**[2,3] **Yang Yu**[2,4] **Behruz Bozorg**[3] **and Hamed Kalhori**[5,6]

[1]*School of Civil Engineering, Changsha University of Science and Technology, Changsha 410114, China*
[2]*Centre for Infrastructure Engineering, Western Sydney University, Penrith 2751, New South Wales, Australia*
[3]*Urban Transformations Research Centre, Western Sydney University, Penrith 2751, New South Wales, Australia*
[4]*Centre for Infrastructure Engineering and Safety, School of Civil and Environmental Engineering, The University of New South Wales, Sydney 2052, New South Wales, Australia*
[5]*Department of Mechanical Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran*
[6]*School of Mechanical and Mechatronic Engineering, University of Technology Sydney, Ultimo 2007, New South Wales, Australia*

Correspondence should be addressed to Maria Rashidi; m.rashidi@westernsydney.edu.au

Received 4 September 2025; Revised 30 October 2025; Accepted 11 November 2025

Academic Editor: Young-Jin Cha

Timely and efficient real-time surface damage detection is essential for maintaining the healthy operation of concrete bridges and has become a critical research focus. However, existing deep learning–based damage detection methods still face challenges such as low detection accuracy, poor adaptability, and limited applicability to diverse scenarios. To address these issues and enhance surface damage detection performance in complex environments, this study proposes an improved YOLODF model based on You Only Look Once, Version 5 (YOLOv5). The improvements include replacing the C3 module with the C2f structure with depthwise separable convolutions and inverted bottlenecks (DSIBC2f) module to build a new backbone network, DSIBCSPDarknet, which strengthens feature extraction capabilities. The SPPFCSPC structure is introduced to replace the spatial pyramid pooling fast (SPPF) module, enabling more effective multiscale feature fusion. Furthermore, the Enhanced Multidimensional Collaborative Attention (EMCA) is combined with the DSIBC2f module to construct a fused neck, FNeck, further optimizing feature fusion. Experimental results show that YOLODF significantly outperforms YOLOv5 in terms of precision, recall, F1 score, and $mAP_{0.5}$ and also surpasses the latest YOLOv12. Additionally, it demonstrates excellent damage detection capabilities in challenging scenarios, such as adverse weather, noise interference, and color variations. Despite a slight increase in computational load, YOLODF achieves a detection speed of 118 frames per second, demonstrating its high practicality for surface damage detection on bridges in complex environments.

Keywords: attention mechanism; concrete bridge; deep learning; surface damage detection; YOLOv5

1. Introduction

Bridge structures are a crucial part of highway networks, and as urban development progresses, the number of concrete bridges in cities has rapidly increased. Factors such as material aging, increased traffic volume, accidents, and rainwater erosion have caused surface damage to critical components of concrete bridges (such as main beams, cap beams, and piers), including cracks, concrete spalling, and rebar exposure [1–3]. According to the ASCE Infrastructure Report Card (2021), 42% of bridges in the United States are at least 50 years old and 7.5% are classified as structurally deficient, meaning they are in "poor" condition [4]. These deficient bridges not only require substantial repair costs but also pose significant risks to public safety. Therefore, timely and accurate bridge structure health monitoring (BSHM) is an urgent and challenging task [5–10].

Traditional BSHM typically involves the use of specialized equipment, such as bridge inspection vehicles, with technicians conducting manual inspections and evaluating the condition of bridge structure. This approach is not only inefficient but also highly subjective and potentially hazardous [11]. In recent years, computer vision methods have increasingly replaced manual inspections. Techniques such as edge detection, threshold segmentation, and image processing have been widely used for detecting cracks and measuring dynamic response on concrete bridge surfaces [12–14]. However, traditional computer vision methods rely on manually designed features, which often perform poorly under varying lighting conditions, complex backgrounds, or noise interference, making them unsuitable for handling the variability of real-world environments [15].

With the rapid development of deep learning (DL) [16] methods, deep convolutional neural networks (DCNNs) [17], known for their automatic feature extraction capabilities and robustness, have been widely applied in the field of structural health monitoring. These techniques have demonstrated significant potential and advantages in tasks such as structural displacement measurement [18], bolt damage detection [19], cable force assessment [20], structural safety evaluation [21], structural health diagnosis [22], and surface damage detection [23]. Among these, DL–based object detection algorithms are some of the most widely used technologies, mainly including two-stage detection methods represented by R-CNN [24–26] and one-stage detection methods represented by Single Shot MultiBox Detector (SSD) [27] and You Only Look Once (YOLO) [28–38]. Zhang et al. [39], through transfer learning, applied YOLOv3 to the detection of more complex and diverse concrete bridge surface damages, such as cracks, rebar exposure, concrete spalling, and separation, and compared it with Faster R-CNN. The results showed that, under the same detection accuracy, YOLOv3 was 3–5 times faster than Faster R-CNN, making it more suitable for real-time surface damage detection tasks.

In recent years, advanced DL algorithms and computer vision technologies continue to make progress in the fields of research and application related to infrastructure health monitoring and performance evaluation. Zhou et al. [40] proposed an improved YOLOv4-ED, using the lightweight EfficientNet as the backbone combined with depthwise separable convolutions, for detecting cracks and rebar exposure in tunnel linings. Chen et al. [41] combined diffusion model with segmentation model for repairing missing crack information, thereby improving the detection accuracy of cracks. Zou et al. [42] utilized the YOLOv4 [31] to develop a model for post-earthquake damage detection and safety assessment of concrete structures. Zhao et al. [43] enhanced the feature extraction capabilities of the YOLOv5 [32] by incorporating Transformer modules and attention mechanisms, enabling the detection and 3D reconstruction of surface damage on concrete dams using UAV. Ye et al. [44] integrated the Swin Transformer module into the feature extraction network of YOLOv7 [34], significantly improving the accuracy of road crack detection. Xu et al. [45] established a large general visual model for structural damage segmentation based on Transformer. Jiang et al. [46], using the YOLOv7 as the detection framework, improved the classification and detection of steel bridge rivet defects through a multiscale moving window search technique. Niu et al. [47] integrated the multiscale feature extraction module C2f-DWR into the backbone of the YOLOv8 and adopted a simplified bidirectional feature pyramid network in the neck structure to optimize the feature fusion network, thereby improving the detection accuracy of small objects. Rakesh et al. [48] used the YOLO network family (v3-v10) for concrete structure damage identification in multifeature backgrounds and found that the YOLOv4 showed the best detection accuracy.

Although YOLO models have demonstrated excellent performance in many object detection tasks, they face several challenges in detecting damage. These challenges include limited capability in identifying small-scale damage and weak resistance to interference in complex environmental conditions such as varying lighting, adverse weather, occlusion, and background noise. For example, Saúl et al. [49] constructed a diverse pavement defect dataset and applied the YOLOv5 for detection, yet limitations in robustness across various scenarios were still observed. Zhang et al. [50] proposed ARD-YOLO by integrating multidimensional attention mechanisms and lightweight upsampling operators into the YOLOv5, aiming to improve detection in complex road environments. Li et al. [51] introduced RDD-RGNet, a region-guided damage detection network based on the YOLOv8, to strike a balance between accuracy and computational cost. Rong et al. [52] constructed a super-resolution reconstruction network using dense residual blocks and spatial attention modules, providing support for structural damage detection. Wang et al. [53] developed a crack detection model based on the Swin Transformer for scenarios with small training datasets, aiming to reduce the model's heavy reliance on large datasets. While these approaches contribute valuable improvements, there remains a gap in achieving both high precision and strong generalization for concrete bridge surface damage detection under real-world, complex conditions.

The surface damage on concrete bridges is diverse in type, varies in size, and is often set against complex and dynamic backgrounds, making it difficult for existing detection models to extract subtle damage features hidden in such environments. Moreover, the generalization ability of these models under different environments and conditions remains inadequate [54]. To address these issues, this study proposes an improved YOLODF network, based on the YOLOv5, designed for real-time damage detection on concrete bridge surfaces in complex environments. The main contributions of this paper are as follows.

1. Constructed a database containing seven common types of damage, including cracks, spalling, rebar exposure, separation, corrosion, void pits, and holes. Data augmentation techniques were employed to simulate damage scenarios under various environmental conditions such as motion blur, rain, snow, fog, noise interference, and color variations.

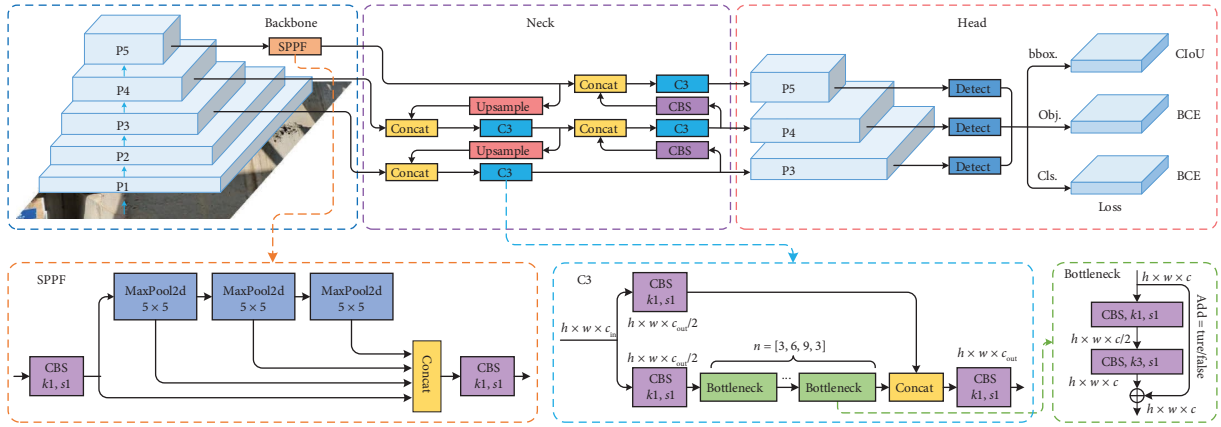

Figure 1: YOLOv5 architecture.

2. Replaced the C3 module in the backbone network of the YOLOv5 with the DSIBC2f structure, creating a new backbone network called DIBCSPDarknet, which enhances the ability of the model to extract subtle damage features.
3. Introduced an Enhanced Multidimensional Collaborative Attention (EMCA) mechanism in the feature fusion path of the neck network and replaced the C3 module with the DSIBC2f structure to form the new fusion neck, FNeck. This improves the multiscale feature fusion capability of the model in complex scenarios.
4. Replaced the spatial pyramid pooling fast (SPPF) module in the YOLOv5 with the SPPFCSPC structure, which provides stronger multiscale feature fusion capability. This improves the effectiveness and adaptability of the model in detecting damage at different scales and in diverse environmental conditions.

2. Damage Identification Network for Concrete Bridges

2.1. Overview of YOLOv5. The YOLOv5 [32] consists of three main components: the backbone, neck, and head, as shown in Figure 1. The backbone uses cross stage partial network (CSPNet) [55], which extracts multilevel, multiscale feature maps from the input image. This is the core part of the model, determining its feature representation capability. The feature extraction process in the backbone includes four stages, each utilizing CBS (convolution + batch normalization + SiLU activate function) modules and C3 modules to extract features effectively. The CBS module downsamples feature maps, condensing feature information, while the C3 module, consisting of multiple bottleneck convolution blocks, extracts both local and global features across different network layers.

The path aggregation network (PANet) structure [56] is used in the neck network, which builds a feature pyramid by performing upsampling and downsampling on features from different layers of the backbone. It is responsible for multiscale fusion of features, enhancing the detection capability of objects of various sizes.

The head network generates anchor boxes on feature maps at different scales. It predicts objects of three sizes: large, medium, and small (20×20, 40×40, 80×80) based on features from the neck network. These predictions include the bounding box location, class, and confidence score of the object.

The C3 module in YOLOv5, which extracts features by stacking multiple bottleneck blocks, ensures effective information flow at higher-level features. However, the dimensionality reduction (1×1 convolution) in the bottleneck blocks may result in feature loss, particularly when dealing with high-resolution detail features. Additionally, shallow layers are more susceptible to background noise and irrelevant information, leading to inefficient feature extraction. This limitation is especially evident in complex environments, where shallow layers often contain large amounts of irrelevant background information. Critical damage features may be weakened or lost, failing to provide sufficient support to deeper layers.

Although YOLOv5 has demonstrated excellent performance in detection tasks on public datasets like PASCAL VOC and MS COCO, it still requires optimization for tasks in specialized domains. The identification of surface damage on concrete bridges, as examined in this paper, presents distinct challenges such as complex backgrounds, low contrast between the damage and the surrounding environment, uneven object size distribution, and low image resolution. These characteristics cause YOLOv5 to struggle with insufficient feature extraction and blurred object boundaries when handling damage detection tasks in complex environments.

2.2. Proposed Method for Concrete Bridge Surface Damage Detection. To achieve high-precision detection of surface damage of concrete bridges in complex environments, this section introduces a model called YOLODF for concrete bridge surface damage detection, as shown in Figure 2. The YOLODF model addresses the limitations of the original YOLOv5 by introducing four efficient feature processing modules: DSIBC2f, SPPFCSPC, EMCA, and FNeck. The

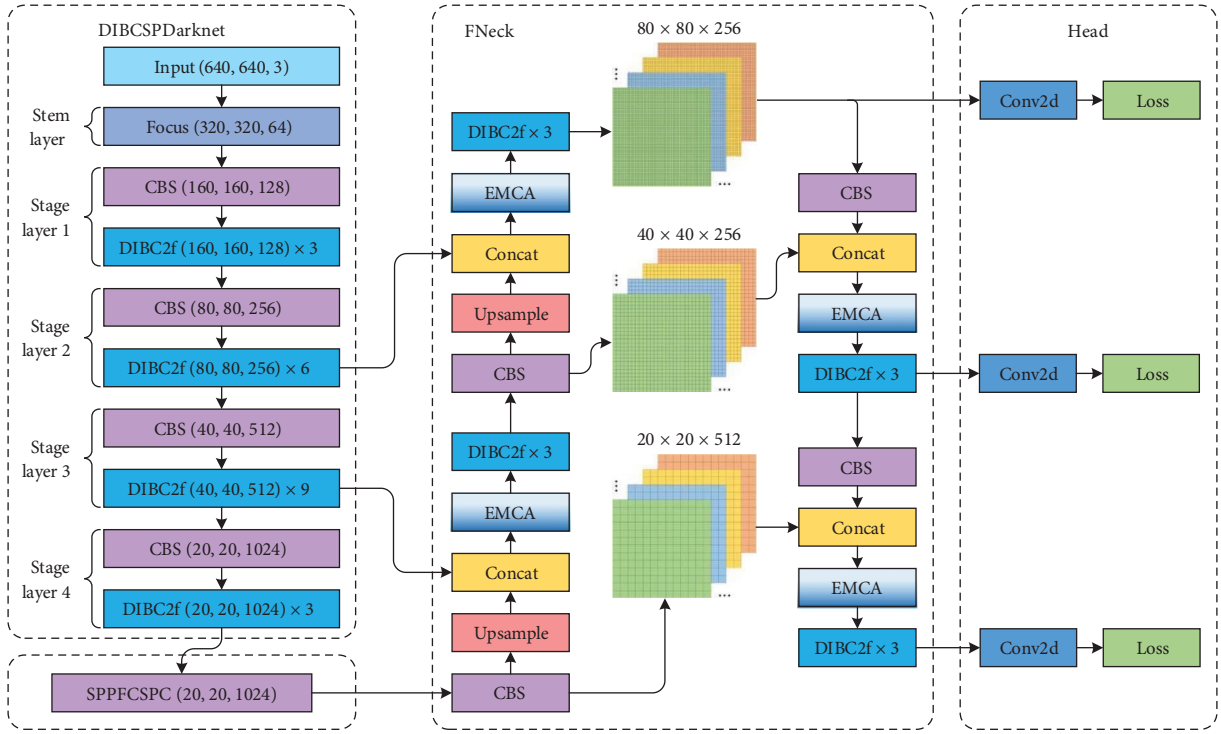

Figure 2: YOLODF architecture.

DSIBC2f module replaces the C3 module in both the backbone and neck networks. This modification creates a new backbone network, DIBCSPDarknet, which improves feature extraction capabilities by incorporating more advanced feature processing techniques. Additionally, the module is integrated into the neck network, where it is combined with the EMCA module to form a new fusion neck (FNeck), which further enhances feature fusion across multiple scales.

2.2.1. DSIBC2f Structure. To enhance the feature extraction capabilities of the backbone and neck network of YOLOv5, this study proposes a C2f structure [35] with depthwise separable convolutions and inverted bottlenecks (DSIBC2f) to replace the original C3 module in YOLOv5. C2f consists of two 1×1 feature fusion convolutions and a set of bottleneck modules in the middle. Its design follows the principles of CSPNet, focusing on improving the feature extraction performance and efficiency of the model. The C2f structure first splits the input feature map into a trunk and a shortcut. The trunk processes feature through several bottleneck structures to extract deeper-level features. The bottleneck structure learns residual features, stabilizing gradients to prevent vanishing or exploding gradients. The shortcut preserves input information from each stage of the trunk and concatenates it with the deep features processed by the trunk. This structure ensures information diversity, helping the model learn both global and detailed information, thereby enhancing its representational power.

It is worth noting that the bottleneck module in the C2f structure uses the same feature extraction strategy as the bottleneck module in the C3 structure, namely, "dimensionality reduction–convolution–dimensionality expansion." This approach relies heavily on low-dimensional features, which may lead to the loss of important information when processing high-resolution features. Research on the MobileNets models [57, 58] shows that the inverted bottleneck, which employs a "dimensionality expansion–convolution–dimensionality reduction" strategy, can process more features in high-dimensional space. This method is better suited for extracting finer and more complex features, and it avoids using an activation function at the end to prevent information loss.

In this study, the inverted bottleneck with depthwise separable convolution is used as the bottleneck module for the DSIBC2f structure, as shown in Figure 3. First, Specialized Figure $X \in \mathbb{R}^{C \times H \times W}$ employs a 3×3 CBS module for initial feature extraction. Subsequently, a 1×1 pointwise CBS (PWCBS) is used to expand the feature dimension (where r in the figure is the channel expansion coefficient). The expanded features are then processed by a 3×3 depthwise separable CBS to extract spatial features in a high-dimensional space. Finally, a 1×1 pointwise CB is applied for dimensionality reduction, thereby obtaining an effective feature layer with enhanced spatial characteristics.

2.2.2. SPPFCSPC Structure. The SPP structure [59] in the YOLO series plays a crucial role in extracting multiscale features, significantly enhancing the detection performance of the model. To make the object detection model more suitable for identifying irregularly shaped and variably sized targets, such as surface damage on concrete bridges, this study replaces the original SPPF structure in YOLOv5 with

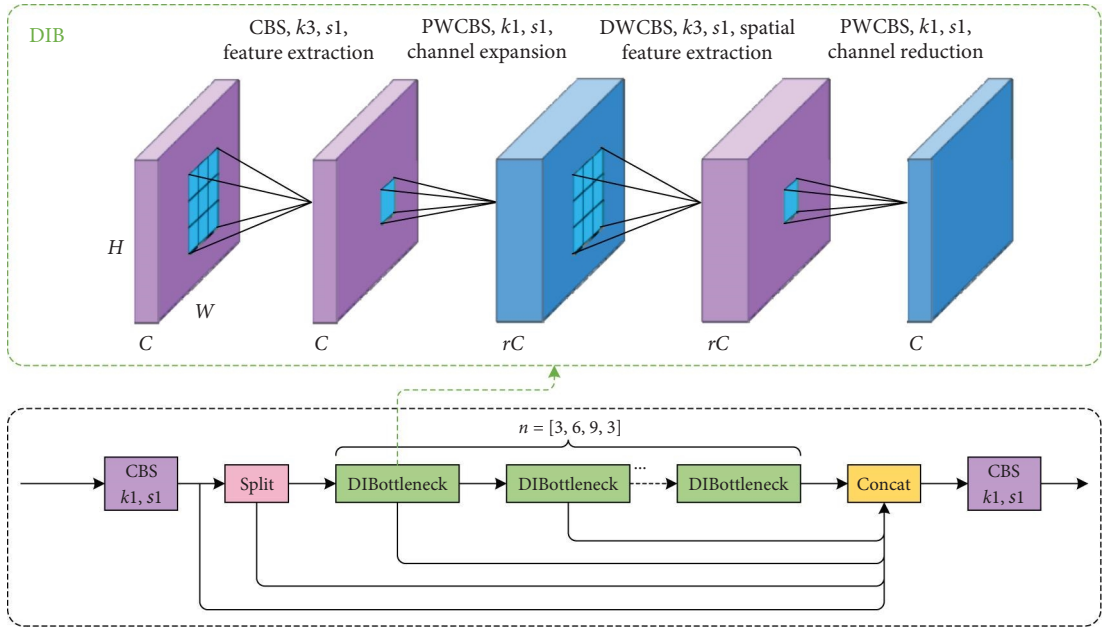

Figure 3: DSIBC2f structure.

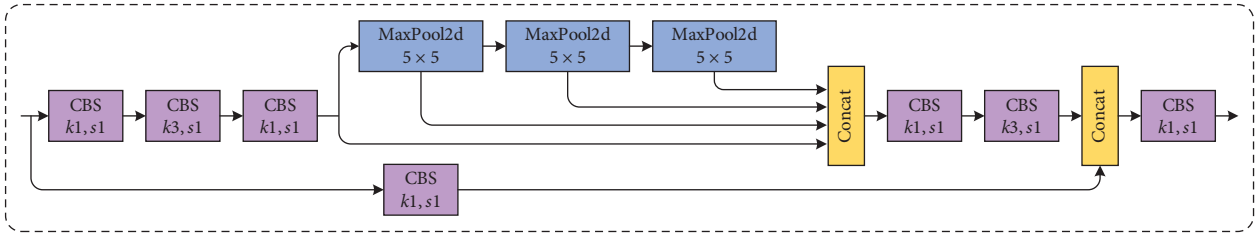

Figure 4: SPPFCSPC structure.

the SPPFCSPC structure [60]. The SPPFCSPC structure builds on the SPPF design by incorporating the CSPNet architecture, as shown in Figure 4.

Specifically, SPPFCSPC splits the feature map into two branches: one branch passes through a bottleneck structure to extract features before entering the SPPF module for multiscale pooling, while the other branch undergoes direct convolution. Finally, the pooled features and the directly convolved features are fused in the channel dimension within the second bottleneck structure. This design enables the network to better preserve multilevel features, resulting in improved object detection performance.

2.2.3. Enhanced Multidimensional Cooperation Attention. To enhance the fine-grained extraction and fusion of damage features in the neck network of YOLOv5, this study proposes an EMCA mechanism. Unlike conventional channel attention, spatial attention, or channel–spatial hybrid attention mechanisms, EMCA captures more comprehensive damage features from the channel dimension, height spatial dimension, width spatial dimension, and local 3D space of the input features. It then fuses damage features at different scales to improve the fine-grained representation of the model for these features. The detailed feature processing flow is shown in Figure 5(a).

In EMCA, the input features $F \in \mathbb{R}^{C\times H\times W}$ are first equally divided into four parts, each entering one of four different scale feature processing branches. In the first, bottom branch, the spatial dimensions of the input features are preserved to compute the interaction between channels. This process is similar to SENet [61], where the features are passed through a squeeze–excitation module to generate the feature map $F_C \in \mathbb{R}^{C\times H\times W}$, resulting in feature maps $\overline{F}_C \in \mathbb{R}^{C\times H\times W}$ enhanced by the channel attention weights $C_W \in \mathbb{R}^{C\times 1\times 1}$. The process is summed up as follows:

$$F_C = IdeO_C(F), \tag{1}$$

$$C_W = \sigma\left(T_{ex}\left(T_{sq}(F_C)\right)\right), \tag{2}$$

$$\overline{F}_C = C_W \otimes F_C, \tag{3}$$

where $IdeO_C(\cdot)$ refers to the identity mapping function. The $\sigma(\cdot)$ stands for the sigmoid activation function. The $T_{sq}(\cdot)$ and $T_{ex}(\cdot)$ refer to the squeeze transformation and excitation transformation, respectively.

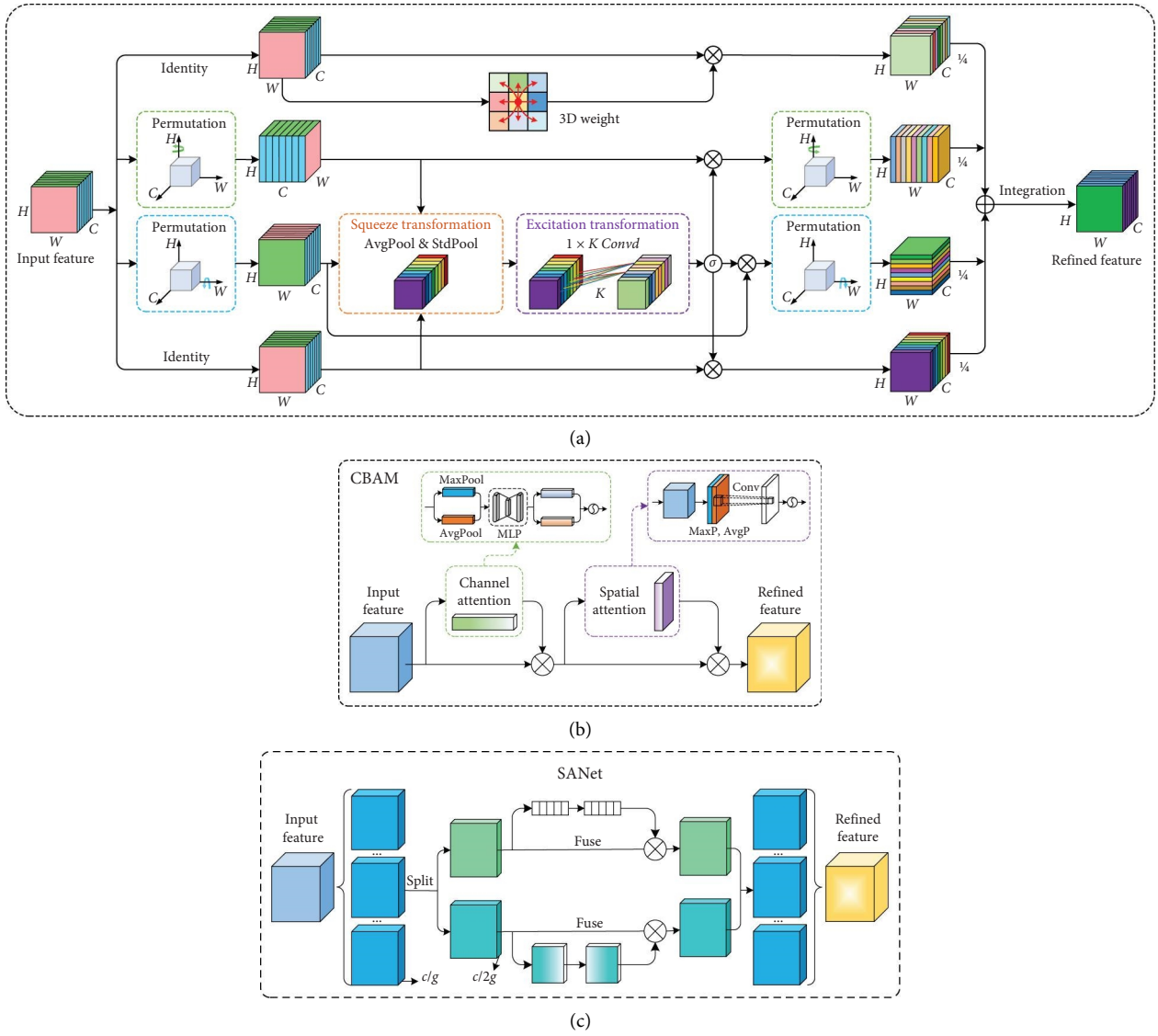

Figure 5: Attention structure. (a) EMCA, (b) CMAB, and (c) SANet.

The two middle branches capture feature interactions in the height and width spatial dimensions. First, the feature map is transformed along the height and width dimensions to obtain the corresponding spatial feature maps $\overrightarrow{F}_W \in \mathbb{R}^{W\times H\times C}$ and $\overrightarrow{F}_H \in \mathbb{R}^{H\times C\times W}$. These transformed maps are also passed through the squeeze–excitation module (SENet), resulting in feature maps $\overline{F}_W \in \mathbb{R}^{W\times H\times C}$ and $\overline{F}_H \in \mathbb{R}^{H\times C\times W}$ enhanced by width spatial attention weights $W_W \in \mathbb{R}^{W\times 1\times 1}$ and height spatial attention weights $H_W \in \mathbb{R}^{H\times 1\times 1}$, respectively. To maintain consistent spatial dimensions for multidimensional feature fusion, the enhanced feature maps are inverse-transformed, resulting in enhanced feature maps $\overline{F}_W' \in \mathbb{R}^{C\times H\times W}$ and $\overline{F}_H' \in \mathbb{R}^{C\times H\times W}$ with dimensions matching those of the original feature map. Similarly, this process can be summarized as the following equations:

$$\begin{aligned}\overrightarrow{F}_W &= PerO_H(F),\\ \overrightarrow{F}_H &= PerO_W(F),\end{aligned} \tag{4}$$

$$\begin{aligned}W_W &= \sigma\left(T_{ex}\left(T_{sq}\left(\overrightarrow{F}_W\right)\right)\right),\\ W_H &= \sigma\left(T_{ex}\left(T_{sq}\left(\overrightarrow{F}_H\right)\right)\right),\end{aligned} \tag{5}$$

$$\begin{aligned}\overline{F}_W &= W_W \otimes \overrightarrow{F}_W,\\ \overline{F}_H &= H_W \otimes \overrightarrow{F}_H,\end{aligned} \tag{6}$$

$$\begin{aligned}\overline{F}_W' &= PerO_H^{-1}\left(\overline{F}_W\right),\\ \overline{F}_H' &= PerO_W^{-1}\left(\overline{F}_H\right).\end{aligned} \tag{7}$$

Here, $PerO_H(\cdot)$ and $PerO_W(\cdot)$ represent a 90° anticlockwise rotation along the H and W axes, while $PerO_H^{-1}(\cdot)$ and $PerO_W^{-1}(\cdot)$ represent the inverse, respectively.

To enhance the representation of local features across different feature dimensions, a parameter-free attention mechanism, SimAM [62], is introduced in the top branch, which also retains the spatial dimensions of the input features. The core concept of SimAM is based on local self-similarity in images. In an image, adjacent pixels typically exhibit strong similarity, while distant pixels show weaker similarity. SimAM leverages this property by calculating the similarity between each pixel and its neighboring pixels within the feature map to obtain the 3D attention weights $S_W \in \mathbb{R}^{C\times H\times W}$, producing a feature map $\overline{F}_S \in \mathbb{R}^{C\times H\times W}$. Specifically, for each position (neuron) in the feature map, SimAM evaluates its importance using an energy function, where neurons with lower energy values are considered more significant. The energy function and the computation of the enhanced feature map are defined as follows:

$$e_t = \frac{1}{M-1}\sum_{i=1}^{M-1}\left((x_i - t)^2 + \lambda\right) + (t-\mu_t)^2, \tag{8}$$

$$S_W = \sigma\left(\frac{1}{e_t}\right), \tag{9}$$

$$\overline{F}_S = S_W \otimes F. \tag{10}$$

Here, t denotes the value of the target neuron, x_i represents the values of the neighboring neurons within the same region, and M is the size of the neighborhood (typically the spatial dimensions of the feature map, i.e., $H\times W$). The μ_t indicates the mean value of the target neuron, and λ (empirically set to 1×10^{-4}) is a small regularization constant introduced to avoid division by zero.

Finally, the feature maps enhanced across the four different dimensions are averaged, resulting in the final fine-grained feature map $\overline{F} \in \mathbb{R}^{C\times H\times W}$. Formally, the process is generalized as follows:

$$\overline{F} = \frac{1}{4}\otimes\left(\overline{F}_C \oplus \overline{F}'_W \oplus \overline{F}'_H \oplus \overline{F}_S\right). \tag{11}$$

3. Damage Dataset of Concrete Bridge

3.1. Damage Dataset Preparation. Concrete bridge damage detection relies on a sufficient amount of labeled sample data, which is crucial for improving the generalization ability of damage detection models. However, building a comprehensive concrete bridge damage dataset is challenging and typically requires significant time and labor. Current researches often focus on a few damage types, such as cracks, rebar exposure, and spalling, but in practice, bridge damage is complex and varied. Detecting only one type of damage does not fully reflect the structural health of concrete bridges. Therefore, this paper aims to expand the diversity and scope of the dataset to include a wider variety of damage types, thereby improving the accuracy and reliability of concrete bridge damage detection.

In this study, the dataset used for training the concrete bridge surface damage detection model includes over 90% of the damage images collected in the field using a drone cloud-control platform and smartphones. The remaining images were obtained through online searches and carefully selected based on reliable resolution (if the resolution of the damaged image is greater than 640×480 or 480×640). All datasets used for detecting concrete bridge damage follow the PASCAL VOC format. A concrete bridge damage dataset containing 2000 original images was established through data collection, covering seven common damage types: cracks, spalling, rebar exposure, separation, corrosion, void pits, and holes. Figure 6 provides detailed descriptions of the common forms of surface damage found in concrete bridges, the causes of these damages, and their potential impact on the structural integrity of the bridges.

3.2. Dataset Augmentation. The progression of surface damage on concrete bridges is often unpredictable, influenced by severe weather, geological disasters, and traffic accidents, which can cause varying degrees of surface damage. Under these complex conditions, on-site inspections and damage data collection face numerous challenges, and obtaining effective damage images can be difficult. To realistically simulate these complex scenarios, this study employs augmentation algorithms such as motion blur, rain, snow, fog, noise interference, and color variation provided by the image augmentation library imgaug to randomly combine and enhance the original dataset, generating degraded images under common environmental conditions. This approach provides more comprehensive and diverse data for future damage detection research, enhancing the adaptability to varied environments of the model.

By augmenting the original damage dataset, a new dataset of 8000 damage images was created to simulate concrete bridge damage under complex conditions. The original damage dataset was annotated using LabelImg software, and image augmentation was performed using the imgaug library. During model training, the annotated damage dataset was randomly divided into a training set and a validation set at a ratio of 9:1. In addition, a separate set of unannotated damage images, collected independently from different bridge scenes, was used as the test set to evaluate the performance of the model in real-world conditions. The same training set, validation set, and test set were used in training both the improved model and the comparison model to ensure fair comparison. Due to limitations in training equipment and time, the model training metrics presented in this paper are derived from a single complete training run.

4. Experiment Preparation

4.1. Experimental Environment. The damage recognition models in this experiment were implemented on a personal computer running Ubuntu 20.04.5 LTS. The system features an Intel i9-13900KF CPU with 24 cores and 32 threads,

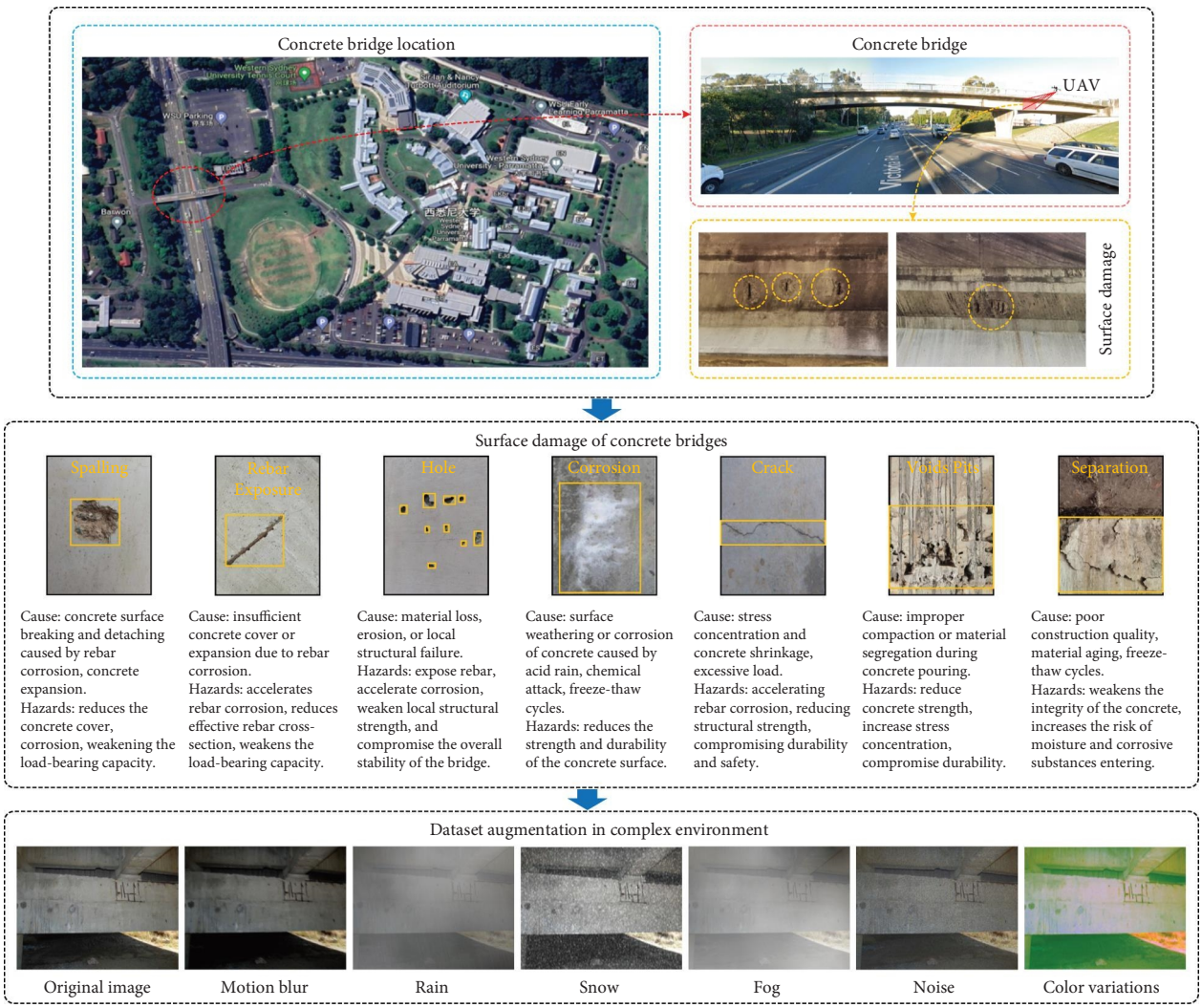

Figure 6: Common surface damage of concrete bridges.

alongside two NVIDIA RTX 4090 GPUs, each with 24 GB of video memory. To optimize performance, CUDA 11.3 and cuDNN 8.2.0 were employed. The software environment included Python 3.7 and PyTorch 1.11 as the primary programming framework.

4.2. Model Performance Evaluation Metrics. To objectively and effectively evaluate the performance of the damage recognition models used in the experiments, the following metrics were employed: average precision (AP), mean AP (mAP), F1 score, frames per second (FPS), parameter count, and computational workload (giga floating-point operations per second [GFLOPs]).

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{12}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{13}$$

$$F1 = \frac{1}{n}\sum\left(\frac{2P \cdot R}{P + R}\right), \tag{14}$$

$$\mathrm{AP} = \int_0^1 P(R)dR, \tag{15}$$

$$\mathrm{mAP} = \frac{1}{n}\sum \mathrm{AP}, \tag{16}$$

$$\mathrm{FPS} = \frac{N_{\mathrm{image}}}{T_{\mathrm{totol}}}. \tag{17}$$

In this context, precision (P) refers to the proportion of correctly identified damage instances among all instances predicted as damage, reflecting the ability to avoid false positives of the model, while recall (R) measures the proportion of actual damage instances that are correctly detected by the model, indicating its ability to minimize false negatives. Specifically, P and R are calculated based on the components of the confusion matrix (Figure 7): true positives (TPs) refer to correctly identified damage instances, false positives (FPs) refer to nondamage instances incorrectly classified as damage, and false negatives (FN) refers to damage instances that were missed by the model.

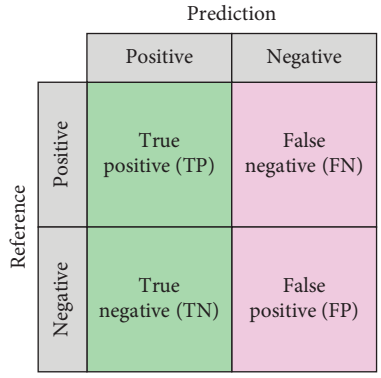

Figure 7: Confusion matrix.

The F1 score is the harmonic mean of precision and recall, ranging from 0 to 1, with 1 indicating the *best performance*. The AP measures the area under the precision–recall curve for a single class, reflecting the trade-off between precision and recall across different confidence thresholds. The mAP is the mean of AP values over all object classes, providing a comprehensive evaluation of detection performance. In this study, we report mAP at an IoU threshold of 0.5 (mAP@0.5), which is a widely adopted metric in object detection tasks. FPS measures the number of image frames processed per second and is primarily used to evaluate the inference speed of the model.

4.3. Model Training. The proposed YOLODF model utilizes a new backbone network, DIBCSPDarknet, for image feature extraction. Due to this change in the backbone, the original YOLOv5-s pretrained weights are no longer applicable to the YOLODF model. To avoid training the model from scratch and to ensure training stability, we first pretrained the baseline model, based on the new DIBCSPDarknet backbone, on the PASCAL VOC dataset. This pretraining allows the model to learn basic features such as edges, shapes, and textures, which are then transferred to the task of detecting damage on concrete bridge surfaces. As a result, the adaptability of the model is enhanced, and the risk of overfitting is reduced.

After the baseline model demonstrated strong detection capabilities, we further fine-tuned it on the concrete bridge surface damage dataset to adapt it to the task of identifying surface damage of concrete bridge. Subsequently, we enhanced the feature extraction of the model by introducing optimization modules such as DSIBC2f, SPPFCSPC, and EMCA. This resulted in the final YOLODF model, which offers higher damage detection accuracy and stronger adaptability.

In the training process of DL algorithms, an optimized set of hyperparameters is essential for the model to successfully complete the intended task. For both the pretraining and fine-tuning of the YOLODF model, the input image resolution was set to 640×640 pixels. The training spanned 300 epochs, with a batch size of 8 per epoch. We employed the Adam optimizer, setting its momentum parameter to 0.937 and the initial learning rate to 0.001. A cosine annealing strategy was used to gradually reduce the learning rate, with a minimum value of 0.00001. These hyperparameters were selected based on the default settings of YOLOv5 and were validated through preliminary experiments to ensure stable convergence and optimal detection performance. Subsequent comparative experiments are conducted under the same training conditions and hyperparameters.

5. Experiment Results and Analysis

5.1. Ablation Experiments

5.1.1. YOLODF Model Training Process. The YOLODF model proposed in this study is an improvement based on the YOLOv5-s model. The most notable modification is the replacement of the C3 module in the YOLOv5-s backbone with the DSIBC2f module, creating a new backbone network, DIBCSPDarknet, which forms the baseline model Baseline for this study. Subsequently, DSIBC2f, SPPFCSPC, and EMCA feature optimization modules were introduced into the baseline model to develop the final model, YOLODF. The model parameters of YOLOv5-s, the baseline model, and YOLODF are shown in Table 1. There is no significant difference in the amount of computation and parameters between the baseline model and YOLOv5-s, and GFLOPS and Params increased by 0.38 GB and 0.05 MB, respectively. However, the introduction of the multiscale feature fusion module SPPFCSPC has increased the amount of computation and parameters of the model, which is almost doubled compared with YOLOv5-s and the baseline model. The neck network (FNeck) with a multiscale feature fusion attention mechanism is relatively lightweight, and GFLOPS and Params increased by 0.16 GB and 0.03 MB, respectively.

Figure 8 shows the loss curves during the fine-tuning of the YOLODF model on the damage dataset, as well as the $mAP_{0.5}$ performance of YOLODF, YOLOv5-s, and the baseline model on the validation set. It can be observed that the YOLODF model demonstrated strong convergence speed during fine-tuning, with both training and validation losses rapidly decreasing in the early stages and stabilizing after the 220th epoch. The damage detection accuracy of the baseline model and YOLODF was similar in the early stages of training. However, starting from the 40th epoch, the accuracy of YOLODF quickly surpassed that of the baseline model and continued to outperform it throughout the entire training process.

Additionally, in the early stages of training, both the baseline model and YOLODF, due to the introduction of the DIBCSPDarknet backbone, achieved nearly double the damage detection accuracy compared to YOLOv5-s. Although YOLOv5-s gradually narrowed the accuracy gap with the baseline model in the later stages of training, its damage detection capability remained significantly lower than that of the YOLODF model. This indicates that, while the computational load and model complexity of YOLODF are higher than those of YOLOv5-s and the baseline model, the inclusion of a series of feature optimization modules in

Table 1: Parameter comparison of the YOLODF model building process.

Network	Backbone	GFLOPS (GB)	Params (MB)
YOLOv5-s	CSPDarknet	16.53	7.08
Baseline	DSIBCSPDarknet	16.95	7.13
Baseline + SPPFCSPC	DSIBCSPDarknet	22.10	13.56
Baseline + SPPFCSPC + FNeck (YOLODF)	DSIBCSPDarknet	22.26	13.59

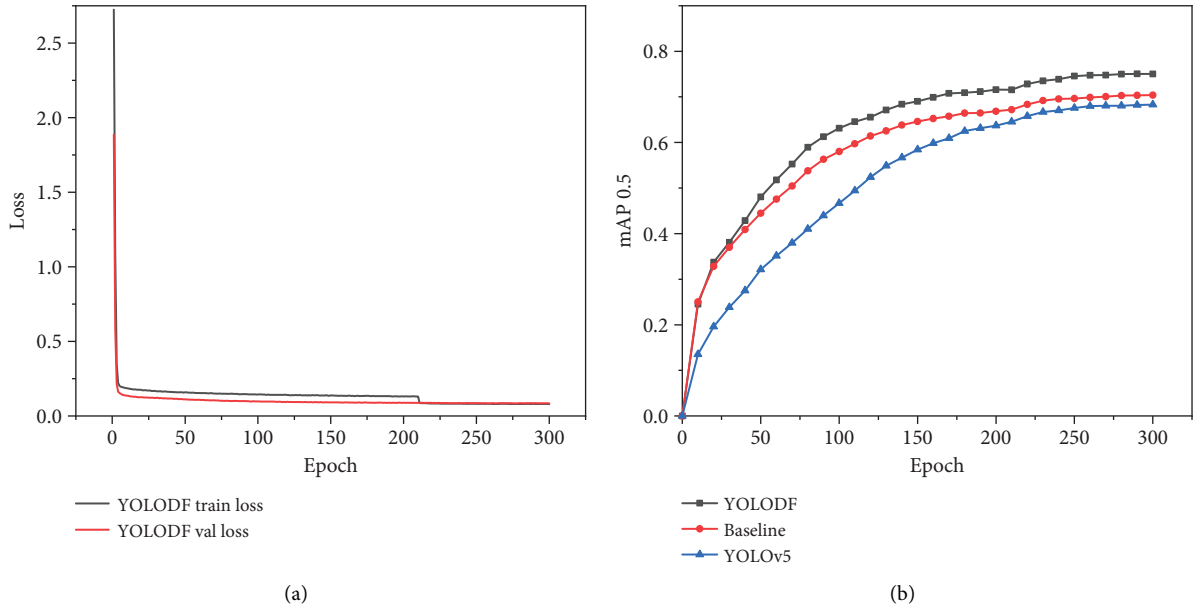

Figure 8: The model training curves of YOLODF. (a) Loss curve and (b) $mAP_{0.5}$ on the validation set.

Table 2: Ablation verification results of YOLODF.

Network	Precision (%)	Recall (%)	F1 (%)	$mAP_{0.5}$ (%)
YOLOv5-s	85.81	58.33	68.88	72.56
Baseline	88.89 (↑3.08)	58.14 (↑−0.19)	69.85 (↑0.97)	73.24 (↑0.68)
Baseline + SPPFCSPC	90.89 (↑5.08)	66.76 (↑8.43)	76.39 (↑7.51)	79.39 (↑6.83)
Baseline + SPPFCSPC + FNeck (YOLODF)	91.18 (↑5.37)	68.68 (↑10.35)	77.60 (↑8.72)	80.78 (↑8.22)

YOLODF leads to a substantial improvement in performance for the damage detection task.

5.1.2. Ablation Study of Feature Optimization Module. To validate the impact of DSIBC2f, SPPFCSPC, and EMCA in the proposed YOLODF model for concrete bridge surface damage recognition, this section presents a performance comparison of YOLOv5-s, the baseline model, Baseline + SPPFCSPC, and Baseline + SPPFCSPC + FNeck. The results are shown in Table 2.

As shown in Table 2, the baseline model, which uses DIBCSPDarknet as the backbone for damage feature extraction, achieved an improvement in overall damage recognition performance. The $mAP_{0.5}$ and F1 scores increased by 0.68% and 0.97%, respectively, compared to YOLOv5-s. The most significant improvement was in precision, which rose by 3.08%. However, the recall of the baseline model slightly decreased by 0.19%, likely because the new DIBCSPDarknet backbone focuses more on extracting high-quality features from the damage areas, thereby improving precision (i.e., the accuracy of the predicted results). This feature extraction strategy, however, may make the model more conservative, leading to a slight reduction in the recall (i.e., the ability to detect some more challenging or unclear damage areas).

When the SPPFCSPC module was introduced into the baseline model, all performance metrics for damage recognition improved significantly, with mAP0.5 reaching 79.39%. The issue of decreased recall was also notably mitigated, with an 8.43% improvement over YOLOv5-s. Precision and F1 score also continued to improve, with increases of 5.08% and 7.51%, respectively. This indicates that DIBCSPDarknet and SPPFCSPC played a crucial role in enhancing the damage recognition performance of the model. DIBCSPDarknet, through the DSIBC2f structure, improved feature extraction, boosting the ability of the model to detect subtle damage. The SPPFCSPC structure combined multiscale feature fusion with the CSP structure, effectively capturing damage features at different scales while optimizing gradient flow and feature

Table 3: Ablation study of different attention mechanisms.

Attention	GFLOPs (GB)	Params (MB)	Precision (%)	Recall (%)	F1 (%)	$mAP_{0.5}$ (%)
Baseline + SPPFCSPC	22.10	13.56	90.89	66.76	76.39	79.39
+ CBAM [63]	22.25	13.76	90.75 (↑−0.14)	68.56 (↑1.80)	76.16 (↑−0.23)	79.24 (↑−0.15)
+ SANet [64]	22.25	13.59	90.94 (↑0.05)	68.81 (↑2.05)	76.82 (↑0.43)	79.63 (↑0.24)
+ EMCA	22.26	13.59	91.18 (↑0.29)	68.68 (↑1.92)	77.60 (↑1.21)	80.78 (↑1.39)

Note: The bold values in Table 4 represent the optimal values of the evaluation metrics: GFLOPs (GB), Params (MB), F1, mAP0.5 (%), and FPS (f/s).

reuse. These improvements increased the robustness and adaptability of the model in complex backgrounds, ensuring high-precision damage detection across varying scales and complex scenarios.

Table 3 presents the impact of the proposed EMCA, as well as the CBAM and SANet attention mechanisms, on the damage recognition performance when added to the neck of the model. The structures of CBAM and SANet are shown in Figures 5(b) and 5(c). CBAM is a typical hybrid attention that combines channel and spatial attention in series. The feature map first passes through the channel attention mechanism to enhance the weights of important channels, and the channel-enhanced feature map is then passed through spatial attention to enhance spatial position information, resulting in the final enhanced feature map. In contrast, SANet adopts a different feature enhancement method from the sequential mechanism of CBAM. It first divides the feature map into smaller units, then performs parallel channel and spatial feature enhancement for each feature map unit, and finally, in the feature output stage, fuses the parallel channel–spatial enhanced features from each feature map unit.

The proposed EMCA considers the channel dimension, height, and width spatial dimension, and calculates the local similarity of each pixel to generate the three-dimensional feature weight. By combining the attention mechanism of these four branches, the multiscale feature fusion capability of EMCA is enhanced. After integrating EMCA into the neck of the YOLODF model, the precision, recall, F1, and mAP0.5 are improved by 0.29%, 1.92%, 1.21%, and 1.39%, respectively, compared with the Baseline + SPPFCSPC model, showing excellent multiscale feature fusion capability. In contrast, the introduction of CBAM and SANet improves the mAP0.5 of YOLOv5-s by 6.68% and 7.07%, respectively, but the performance of CBAM is only improved by 1.80% in recall compared with the Baseline + SPPFCSPC model, while the precision, F1, and AP0.5 are reduced by 0.14%, 0.23%, and 0.15%, indicating that the sequential hybrid feature enhancement method may lead to the loss of some damaged features. Although the parallel hybrid feature enhancement method of SANet also improved performance over the Baseline + SPPFCSPC model, it was still less effective than the EMCA proposed in this study.

5.2. Comparison and Discussion YOLODF With Classical Object Detection Models. In this section, the performance of the YOLODF model is compared with classic object detection models in recognizing surface damage on concrete bridges. Table 4 presents the GFLOPs, parameters, F1 score, $mAP_{0.5}$, and FPS for each damage recognition model, including Faster R-CNN, SSD, YOLOv4, YOLOv5-s, YOLOX-s, YOLOv7-tiny, YOLOv8-s, YOLOv10-s, and YOLOv12-s. The proposed YOLODF network was progressively evolved from YOLOv5-s (GFLOPs = 16.53 GB, Params = 7.08 MB) to its final configuration (GFLOPs = 22.26 GB, Params = 13.59 MB). The increases in computational complexity and parameters were maintained within a reasonable range, while the F1 score improved markedly from 68.88% to 77.60% (+12.6%) and mAP@0.5 increased from 72.56% to 80.78% (+11.3%). These results demonstrate the synergistic effectiveness of the DSIBC2f lightweight feature extraction module, the SPPFCSPC multiscale fusion structure, and the EMCA attention mechanism in capturing fine-grained semantic information of surface damage.

Comparative analyses further confirmed the superiority of YOLODF over conventional object detection architectures. YOLODF outperformed YOLOv12-s, the state-of-the-art (SOTA) baseline, by 1.32% and 1.35% in both F1 (77.60) and mAP@0.5 (80.78%). When compared with YOLOv10-s, YOLODF achieved 2.4% and 1.55% higher F1 and mAP values, respectively, while maintaining comparable computational cost (+2.7% GFLOPs) despite an 87.6% increase in parameters. This indicates that the attention mechanism of YOLODF effectively enhances its task-specific discriminability in damage detection.

Relative to YOLOv8-s ($F1 = 69.42$, $mAP_{0.5} = 73.69$, FPS = 151), YOLODF achieved substantial gains of +11.8% in F1 and +9.6% in $mAP_{0.5}$, with 22.3% fewer GFLOPs and only 22% more parameters. Although the inference speed decreased by 33 FPS, YOLODF exhibited a superior balance between accuracy and efficiency. Furthermore, YOLODF substantially outperformed Faster R-CNN (+97% F1) and YOLOv4 (+22.4% F1), while requiring only 15.7% and 21.2% of YOLOv4's GFLOPs and parameters, respectively. It also achieved an inference speed of 118 FPS, representing a 78.8% improvement over YOLOv4. Despite a minor trade-off in speed compared with the extremely lightweight YOLOv7-tiny (225 FPS), YOLODF attained the most favorable overall balance among accuracy, computational efficiency, and inference speed, making it particularly suitable for semi-real-time structural health monitoring and industrial damage inspection scenarios.

Figure 9 presents the statistical results of precision, recall, and AP for the YOLODF model and other comparison models across various types of damage detection tasks. The results indicate that the YOLOv4 model, despite having huge parameters, still maintains competitive detection precision for most damage categories. However, its recall is relatively low, leading to frequent missed detections. The YOLOv5-s

Table 4: Damage identification results of YOLODF and classic models.

Network	GFLOPs (GB)	Params (MB)	F1	$mAP_{0.5}$ (%)	FPS (f/s)
Faster R-CNN [26]	401.84	136.81	39.39	38.03	50
SSD [27]	62.75	26.29	16.64	27.04	120
YOLOv4 [31]	142.00	63.97	63.36	70.76	66
YOLOv5-s [32]	16.53	7.08	68.88	72.56	150
YOLOX-s [33]	26.77	8.94	75.28	76.89	131
YOLOv7-tiny [34]	**13.23**	**6.03**	57.19	62.19	**225**
YOLOv8-s [35]	28.66	11.14	69.42	73.69	151
YOLOv10-s [37]	21.68	7.24	75.76	79.23	162
YOLOv12-s [38]	21.76	9.28	76.28	79.43	154
YOLODF	22.26	13.59	**77.60**	**80.78**	118

model achieves a notable improvement in recall while maintaining high precision, particularly in detecting small-scale damage such as hole. Nevertheless, its overall detection capability still leaves room for improvement. Although the lightweight YOLOv7-tiny model offers faster inference speed, it performs poorly in identifying various types of damage, resulting in subpar comprehensive detection performance.

YOLOv8-s demonstrates high precision in detecting hole, spalling, and void pits. However, its relatively low recall leads to a substantial drop in average precision. This suggests that YOLOv8-s adopts a more conservative approach in making positive predictions, resulting in fewer false positives but more false negatives. Similarly, YOLOX-s achieves the highest recall in detecting rebar exposure and spalling, but its lower precision causes a decline in overall AP. These observations reflect a common trade-off in object detection: Overly strict decision thresholds may improve precision at the cost of reduced recall, thereby negatively affecting overall performance.

The latest YOLOv12-s demonstrated strong detection performance across various types of surface damage, particularly achieving high precision and recall in identifying small-scale defects such as holes, rebar exposure, and voids and pits, where object boundaries tend to be ambiguous. In contrast, the proposed YOLODF achieved a more balanced trade-off between precision and recall across multiple damage categories, with only minor deficiencies observed in specific cases. Specifically, the precision of YOLODF in detecting exposed rebar was slightly lower than that of YOLOv12-s, while its recall in the same category was marginally inferior to that of YOLOX-s. Apart from these instances, YOLODF consistently outperformed other models across most damage types. These results clearly indicate that YOLODF possesses strong generalization capability and robustness, making it highly suitable for detecting diverse surface damage types under complex real-world conditions.

Figure 10 presents the visualization results of damage detection using classic YOLO series models on four typical images of concrete bridge surface damage. As shown in Figure 10(a), both YOLOv12-s and YOLODF exhibited outstanding performance in detecting cross-shaped cracks, accurately identifying both major and minor cracks. However, the crack detection results of YOLOv12-s were more refined, whereas YOLODF, despite successfully detecting the primary cracks, still showed certain limitations in capturing fine structural details. YOLOv8-s also demonstrated relatively strong crack detection capability but failed to identify the cracks located near the image boundaries. In contrast, YOLOv5-s performed the worst, detecting only a small portion of the cracks present in the image.

Figure 10(b) highlights the detection of small-scale damage (specifically, holes), which remains a persistent challenge for object detection algorithms. The similarity in shape and color between holes and spalling often leads to false detections across all models. However, YOLOv12-s and YOLODF demonstrated clear advantages by successfully identifying a larger number of holes, whereas YOLOv5-s and YOLOv8-s suffered from severe missed detections. In Figure 10(c), the image contains rebar exposure, separation, and corrosion. Due to the relatively blurred edges of the separation regions, only YOLOv12-s and YOLODF successfully identified part of the separation region, while the other models completely failed in this task. Corrosion, characterized by its distinct color features, was effectively recognized by all models, though YOLOv5-s and YOLOv8-s still exhibited partial omission. For smaller instances of rebar exposure, YOLOv8-s again showed significant missed detections. Figure 10(d) presents damage under ideal imaging conditions, where the structural defects are clear and visually distinct. Under such circumstances, all models performed well and achieved accurate detections.

These results demonstrate that the introduction of feature optimization modules like DIBCSPDarknet, SPPFCSPC, and FNeck in YOLODF significantly enhances the adaptability and detection accuracy of the model, underscoring its superiority in real-world applications.

5.3. Damage Identification Analysis of YOLODF in Complex Environments. To verify the performance of the proposed YOLODF model in concrete bridge surface damage detection in real complex environments, this paper selected four types of complex scene images commonly used in bridge inspection for testing and analysis. These scenes include long-distance shooting (involving small and blurred targets), complex backgrounds, low-light environments, and situations with shadow interference.

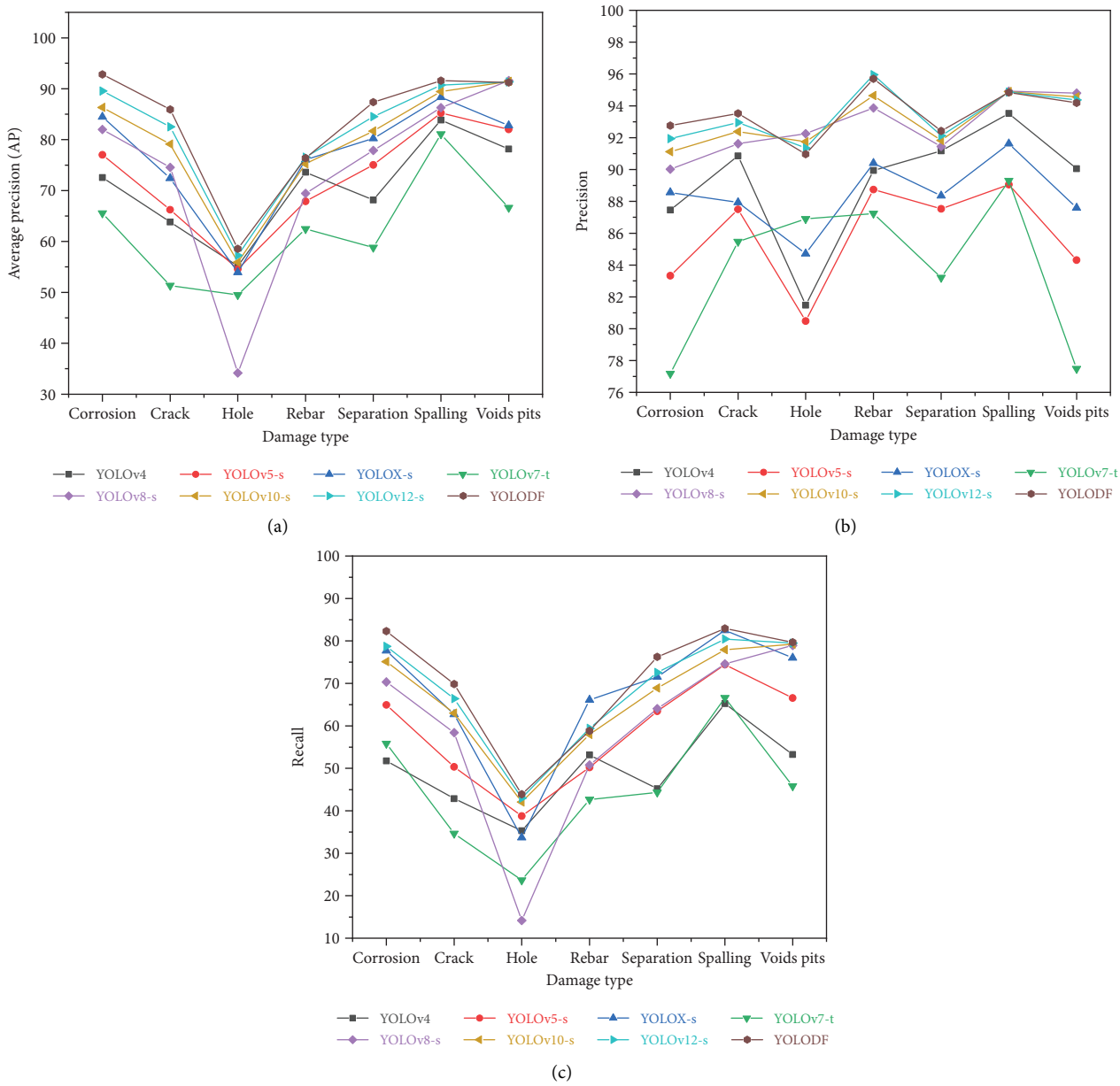

Figure 9: Detection performance of YOLODF and comparison models for each damage type: (a) precision, (b) recall, and (c) AP.

Figure 11 shows the comparison of the detection results of YOLOv5-s, YOLOv8-s, YOLOv12-s, and YOLODF in the above four scenarios. In long-distance imaging scenarios, both YOLOv5-s and YOLOv12-s demonstrated superior capability in identifying most types of damage, with YOLOv12-s achieving the best performance in detecting small-scale rebar exposure. In contrast, YOLOv8-s exhibited significant missed detections for small rebar exposure, indicating its limited sensitivity to fine-scale targets. Compared with these models, the proposed YOLODF showed higher stability in detecting both small and multiple targets, achieving greater accuracy and completeness in its detection results.

In complex background scenarios, the damaged regions may be affected by background interference or exhibit blurred boundaries, leading to varying degrees of false and missed detections in YOLOv5-s, YOLOv8-s, and YOLOv12-s. For instance, YOLOv12-s occasionally misclassified separation areas with indistinct edges as spalling damage. In contrast, the proposed YOLODF maintained reliable detection stability under such complex background conditions, demonstrating its strong adaptability and robustness against background noise.

Under low-light conditions, all four models exhibited varying degrees of missed detections when identifying rebar exposure. Although YOLODF achieved relatively better performance, there remains room for improvement in feature extraction from low-resolution damage regions. In scenarios affected by shadow interference, all models performed well in detecting prominent damage types such as rebar exposure and concrete spalling. However, YOLOv5-s and YOLOv12-s showed limited capability in detecting fine cracks obscured

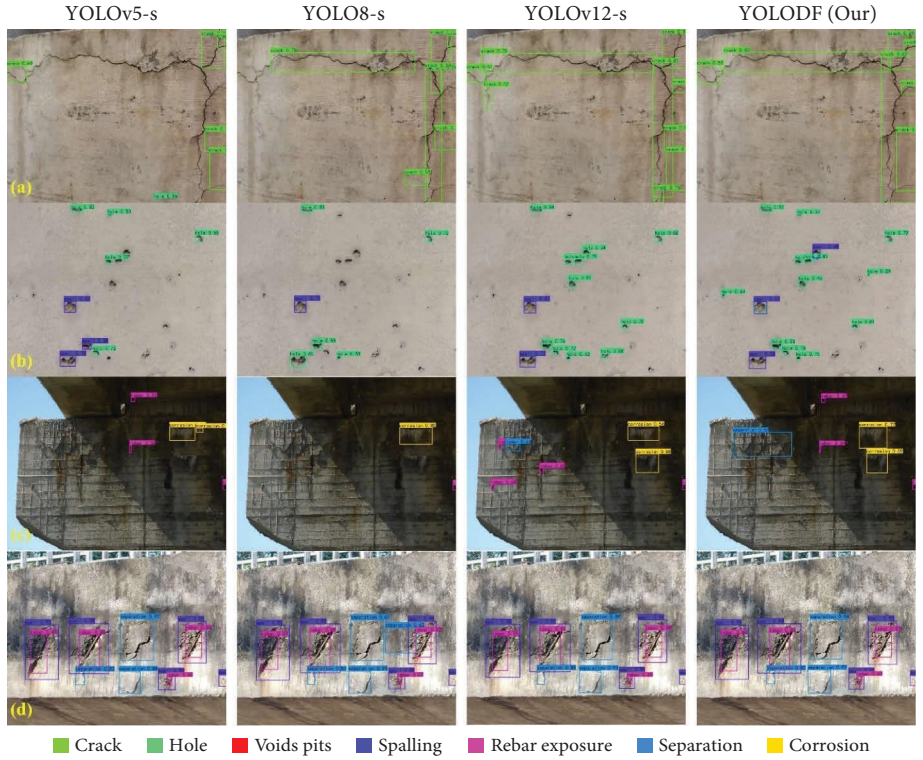

FIGURE 10: Damage recognition results of YOLODF and classic models.

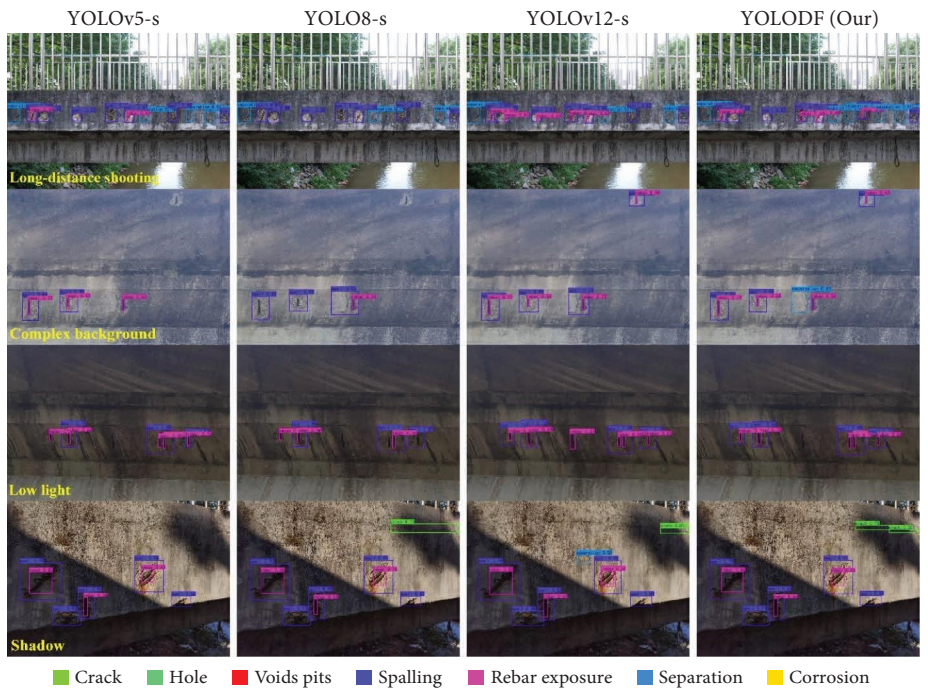

FIGURE 11: Damage recognition results of YOLODF and classic models in simulation complex environments.

by shadows, with YOLOv12-s even misclassifying alternating light–dark background regions as separation damage. While both YOLOv8-s and YOLODF were able to identify part of the fine cracks, their detection accuracy was still affected by sudden illumination variations, indicating that such complex lighting conditions can significantly interfere with the recognition of minor surface damages.

To validate the practicality of the proposed YOLODF model for recognizing concrete bridge surface damage under a wider range of complex environments, this section selects

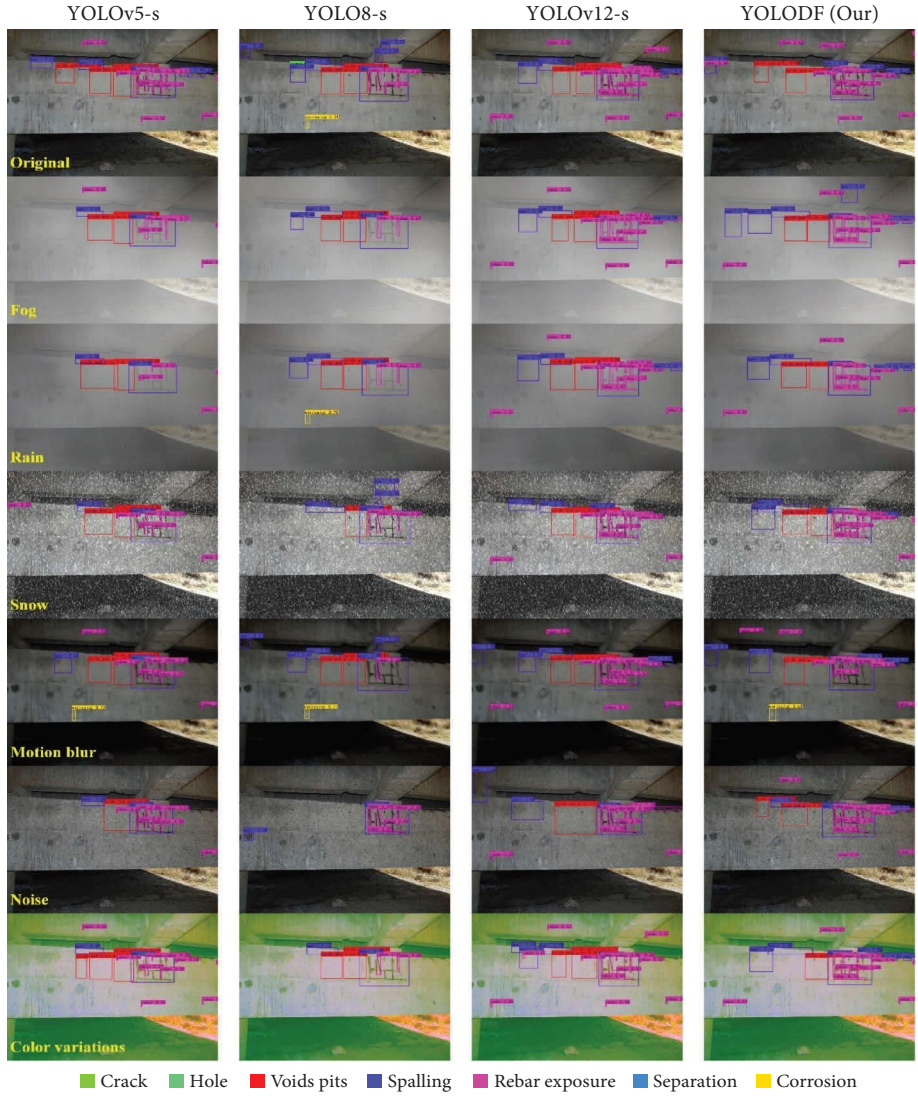

FIGURE 12: Damage recognition results of YOLODF and classic models in simulation complex environments.

an image of a concrete bridge with multiple types of damage and a complex detection background as the damage detection target. The image augmentation methods from Section 3.1 are used to simulate damage images under scenarios such as fog, rain, snow, motion blur, noise interference, and color variations. Figure 12 shows the visualized damage detection results of YOLOv5-s, YOLOv8-s, YOLOv12-s, and YOLODF under these various complex environments.

Given the diverse damage types and complex backgrounds in the original images, all models exhibited certain detection deficiencies. YOLOv8-s demonstrated strong recognition capability for spalling, yet it frequently produced false detections for cracks and corrosion. Both YOLOv5-s and YOLOv8-s struggled to identify small-scale rebar exposure, whereas YOLOv12-s maintained good detection performance for exposed rebar. In contrast, the proposed YOLODF showed excellent performance in detecting small rebar exposure, but it exhibited some limitations in identifying spalling damage, with occasional missed detections.

Under foggy and rainy conditions, the background features of the images were highly similar, resulting in comparable detection performance among all models. Due to the blurring effects of damage regions, both YOLOv5-s and YOLOv8-s experienced a significant decline in their ability to detect rebar exposure, particularly under the uneven backgrounds of rainy scenes, where this limitation became more pronounced. In contrast, YOLOv12-s and the proposed YOLODF exhibited strong robustness in recognizing exposed rebar under these adverse conditions.

In snowy environments, the presence of snowflakes introduced numerous invalid pixels, causing a substantial loss of damage information. This effect was especially evident in small rebar exposure areas, which were often obscured by snow. Under such circumstances, YOLOv5-s performed better than YOLOv8-s, but its detection capability was still inferior to YOLOv12-s and YOLODF, which achieved comparable and stable performance in successfully identifying most surface damage types.

Motion blur causes the damage edges to become indistinct, leading to confusion with background textures. Under such conditions, YOLOv5-s, YOLOv8-s, and YOLODF exhibit corrosion-related false detections, whereas YOLOv12-s shows stronger adaptability to motion blur. The damage recognition capability of YOLODF is notably affected, resulting in missed detections of rebar exposure and honeycomb surfaces; however, its overall detection performance still surpasses that of YOLOv8-s.

Noise interference also leads to partial loss of damage information, yet YOLODF demonstrates robust noise resistance and achieves a clear advantage in the number of detected damages. In contrast, the other three models struggle to detect small-scale damages, and YOLOv12-s tends to generate spalling false positives. Under color variation scenes, YOLOv8-s performs poorly, detecting only limited damages, while YOLOv5-s, YOLOv12-s, and YOLODF remain largely unaffected. Among them, YOLODF achieves the best overall performance.

The proposed YOLODF model demonstrates significant advantages in detecting concrete bridge surface damage under complex environments. While all models exhibit some deficiencies in handling complex backgrounds and diverse damage types, YOLODF stands out in detecting small damage, especially rebar exposure and holes, maintaining high detection accuracy even in extreme weather conditions such as rain and snow. In contrast, the damage recognition capabilities of YOLOv5-s and YOLOv8-s degrade significantly in these challenging environments, particularly in detecting rebar exposure.

Moreover, YOLODF exhibits exceptional robustness under noisy conditions and maintains stable performance in both motion blur and color variation scenarios. Overall, YOLODF outperforms YOLOv8-s, YOLOv5-s, and YOLOv12-s across diverse complex environments, demonstrating superior damage recognition capability and adaptability. These findings highlight YOLODF's great potential as an efficient model for bridge damage detection under diverse and challenging conditions.

6. Conclusion

This study presents YOLODF, an improved YOLOv5-s-based model optimized for detecting concrete bridge surface damage in complex environments through the integration of DIBCSPDarknet, SPPFCSPC, and EMCA modules. Compared to YOLOv5-s, YOLODF achieves substantial gains in detection accuracy, with increases in precision (+5.37%), recall (+10.35%), F1 score (+8.72%), and mAP@0.5 (+8.22%). It also surpasses YOLOv12-s in overall recognition performance while maintaining real-time detection speeds of 118 FPS, demonstrating strong adaptability to challenging conditions.

YOLODF demonstrates strong robustness and adaptability across various challenging environments, such as fog, rain, snow, motion blur, and noise interference. It excels particularly in detecting small damage, such as rebar exposure and holes. Compared to other well-known models, including YOLOv5-s, YOLOv8-s, and YOLO v12-s, YOLODF not only shows significant improvements in metrics like overall $mAP_{0.5}$ and F1 score but also exhibits superior resistance to interference. It maintains high detection accuracy even in adverse conditions involving snow, noise, and color variations.

Despite its advancements, the YOLODF model still has certain limitations. For instance, it may experience missed detections in highly dynamic scenarios, such as severe motion blur or intricate surface textures, particularly for complex damage types like spalling and void pits. Furthermore, after incorporating the SPPFCSPC module, the detection performance of the model was significantly improved due to its enhanced multiscale feature fusion capability. However, this improvement also resulted in a noticeable increase in computational complexity and the number of parameters, which may limit its applicability in real-time or resource-constrained applications.

Future research will focus on enhancing YOLODF by integrating super-resolution detection modules to strengthen the representation of small-scale and texture-blurred damage features. In addition, lightweight convolutional architectures such as ShuffleNet, GhostNet, or MobileNet will be explored to further reduce computational cost and model parameters while maintaining or improving detection accuracy. These improvements are expected to enhance the scalability, efficiency, and real-world applicability of YOLODF in bridge health monitoring and other industrial damage detection scenarios.

Data Availability Statement

Data will be made available on request.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding

The authors express their deep appreciation for the support provided by the Australian Research Council through the Industry Transformation Research Hub for Resilient and Intelligent Infrastructure Systems (RIIS) in Urban, Resources, and Energy Sectors (Grant No. IH210100048).

Acknowledgments

The authors express their deep appreciation for the support provided by the Australian Research Council through the Industry Transformation Research Hub for Resilient and Intelligent Infrastructure Systems (RIIS) in Urban, Resources, and Energy Sectors (Grant No. IH210100048).

References

[1] Y. Ma, Z. Guo, L. Wang, and J. Zhang, "Probabilistic Life Prediction for Reinforced Concrete Structures Subjected to Seasonal corrosion-fatigue Damage," *Journal of Structural Engineering* 146, no. 7 (2020): 04020117, https://doi.org/10.1061/(asce)st.1943-541x.0002666.

[2] Y. Yu, B. Samali, M. Rashidi, M. Mohammadi, T. N. Nguyen, and G. Zhang, "Vision-Based Concrete Crack Detection Using a Hybrid Framework Considering Noise Effect," *Journal of Building Engineering* 61 (2022): 105246, https://doi.org/10.1016/j.jobe.2022.105246.

[3] J. H. Chen, M. C. Su, S. K. Lin, W. J. Lin, and M. Gheisari, "Smart Bridge Maintenance Using Cluster Merging Algorithm Based on self-organizing Map Optimization," *Automation in Construction* 152 (2023): 104913, https://doi.org/10.1016/j.autcon.2023.104913.

[4] *Report Card for America's Infrastructure* (American society of civil engineers (ASCE), 2021), https://infrastructureusa.org/2021-report-card-for-americas-infrastructure.

[5] Y. Yu, M. Rashidi, B. Samali, M. Mohammadi, T. N. Nguyen, and X. Zhou, "Crack Detection of Concrete Structures Using Deep Convolutional Neural Networks Optimized by Enhanced Chicken Swarm Algorithm," *Structural Health Monitoring* 21, no. 5 (2022): 2244–2263, https://doi.org/10.1177/14759217211053546.

[6] Z. Li, J. Yang, X. Li, and X. Xu, "A Novel Bridge Deflection Missing Data Repair Model Based on Two-Stage Modal Decomposition and Deep Learning," *Structural Control and Health Monitoring* 2025, no. 1 (2025): 5458862, https://doi.org/10.1155/stc/5458862.

[7] S. Hartlieb, A. Zeller, T. Haist, A. Reichardt, C. Tarín Sauer, and S. Reichelt, "Advanced Imaging-Based Metrology for Precise Deformation Monitoring: Railway Bridge Case Study," *Structural Control and Health Monitoring* 2025, no. 1 (2025): 5603393, https://doi.org/10.1155/stc/5603393.

[8] M. Rashidi, B. Samali, and P. Sharafi, "A New Model for Bridge Management: Part B: Decision Support System for Remediation Planning," *Australian Journal of Civil Engineering* 14, no. 1 (2016): 46–53, https://doi.org/10.1080/14488353.2015.1092642.

[9] Y. Yu, M. Rashidi, S. Dorafshan, et al., "Ground Penetrating Radar-based Automated Defect Identification of Bridge Decks: a Hybrid Approach," *Journal of Civil Structural Health Monitoring* 15, no. 2 (2025): 521–543, https://doi.org/10.1007/s13349-024-00895-6.

[10] Z. Hu, C. Dang, D. Wang, M. Beer, and L. Wang, "Error-Informed Parallel Adaptive Kriging Method for Time-dependent Reliability Analysis," *Reliability Engineering & System Safety* 262 (2025): 111194, https://doi.org/10.1016/j.ress.2025.111194.

[11] H. Zhang, Y. Chen, B. Liu, X. Guan, and X. Le, "Soft Matching Network with Application to Defect Inspection," *Knowledge-Based Systems* 225 (2021): 107045, https://doi.org/10.1016/j.knosys.2021.107045.

[12] Y. J. Cha, W. Choi, and O. Büyüköztürk, "Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks," *Computer-Aided Civil and Infrastructure Engineering* 32, no. 5 (2017): 361–378, https://doi.org/10.1111/mice.12263.

[13] Y. F. Liu, S. Cho, B. F. Spencer Jr, and J. S. Fan, "Concrete Crack Assessment Using Digital Image Processing and 3D Scene Reconstruction," *Journal of Computing in Civil Engineering* 30, no. 1 (2016): 04014124, https://doi.org/10.1061/(asce)cp.1943-5487.0000446.

[14] D. Feng and M. Q. Feng, "Computer Vision for SHM of Civil Infrastructure: from Dynamic Response Measurement to Damage detection–A Review," *Engineering Structures* 156 (2018): 105–117, https://doi.org/10.1016/j.engstruct.2017.11.018.

[15] B. F. Spencer Jr, V. Hoskere, and Y. Narazaki, "Advances in Computer Vision-based Civil Infrastructure Inspection and Monitoring," *Engineering* 5, no. 2 (2019): 199–222, https://doi.org/10.1016/j.eng.2018.11.030.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature* 521, no. 7553 (2015): 436–444, https://doi.org/10.1038/nature14539.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems* 25 (2012).

[18] Q. Song, J. Wu, H. Wang, Y. An, and G. Tang, "Computer Vision-based illumination-robust and Multi-point Simultaneous Structural Displacement Measuring Method," *Mechanical Systems and Signal Processing* 170 (2022): 108822, https://doi.org/10.1016/j.ymssp.2022.108822.

[19] Y. Zhang and K. V. Yuen, "Bolt Damage Identification Based on orientation-aware Center Point Estimation Network," *Structural Health Monitoring* 21, no. 2 (2022): 438–450, https://doi.org/10.1177/14759217211004243.

[20] T. Jiang, C. Hu, and L. Li, "Complex Background Segmentation for Noncontact Cable Vibration Frequency Estimation Using Semantic Segmentation and Complexity Pursuit Algorithm," *Journal of Civil Structural Health Monitoring* 14, no. 6 (2024): 1533–1554, https://doi.org/10.1007/s13349-024-00798-6.

[21] J. Liu, H. Luo, and H. Liu, "Deep Learning-based Data Analytics for Safety in Construction," *Automation in Construction* 140 (2022): 104302, https://doi.org/10.1016/j.autcon.2022.104302.

[22] X. U. Yang, F. A. N. Yunlei, B. A. O. Yuequan, and L. I. Hui, "Few-Shot Learning for Structural Health Diagnosis of Civil Infrastructure," *Advanced Engineering Informatics* 62 (2024): 102650, https://doi.org/10.1016/j.aei.2024.102650.

[23] S. Xu, R. Shen, Y. Liu, et al., "Cross-Domain Coupled Convolutional Transformer Network for Concrete Damage Detection," *Structural Control and Health Monitoring* 2025, no. 1 (2025): 6547856, https://doi.org/10.1155/stc/6547856.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), 580–587, https://doi.org/10.1109/cvpr.2014.81.

[25] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision* (2015), 1440–1448, https://doi.org/10.1109/iccv.2015.169.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time Object Detection with Region Proposal Networks," *Advances in Neural Information Processing Systems* 28 (2015).

[27] W. Liu, D. Anguelov, D. Erhan, et al., "Single Shot Multibox Detector. Ssd: Single Shot Multibox Detector," in *European Conference on Computer Vision* (Springer International Publishing, 2016).

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, real-time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), 779–788, https://doi.org/10.1109/cvpr.2016.91.

[29] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), 7263–7271.

[30] J. Redmon and A. Farhadi, "Yolov3: an Incremental Improvement. Computer Vision and Pattern Recognition" (2018).

[31] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "Yolov4: Optimal Speed and Accuracy of Object Detection" (2020).

[32] G. Jocher, "YOLOv5 by Ultralytics," (2020), https://github.com/ultralytics/yolov5.

[33] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding Yolo Series in 2021" (2021).

[34] C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "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* (2023), 7464–7475, https://doi.org/10.1109/cvpr52729.2023.00721.

[35] G. Jocher and A. Chaurasia, "Qiu, J. YOLO by Ultralytics," (2023), https://github.com/ultralytics/ultralytics.

[36] C. Y. Wang, I. H. Yeh, and H. Y. Mark Liao, "Yolov9: Learning what You Want to Learn Using Programmable Gradient Information," in *European Conference on Computer Vision* (Springer Nature Switzerland, 2024).

[37] H. Chen, K. Chen, G. Ding, et al., "Yolov10: Real-Time end-to-end Object Detection," *Advances in Neural Information Processing Systems* 37 (2024): 107984–108011, https://doi.org/10.52202/079017-3429.

[38] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-Centric real-time Object Detectors" (2025).

[39] C. Zhang, C. C. Chang, and M. Jamshidi, "Concrete Bridge Surface Damage Detection Using a Single-Stage Detector," *Computer-Aided Civil and Infrastructure Engineering* 35, no. 4 (2020): 389–409, https://doi.org/10.1111/mice.12500.

[40] Z. Zhou, J. Zhang, and C. Gong, "Automatic Detection Method of Tunnel Lining Multi-Defects via an Enhanced You Only Look once Network," *Computer-Aided Civil and Infrastructure Engineering* 37, no. 6 (2022): 762–780, https://doi.org/10.1111/mice.12836.

[41] L. Chen, L. Zhou, L. Li, and M. Luo, "Crackdiffusion: Crack Inpainting with Denoising Diffusion Models and Crack Segmentation Perceptual Score," *Smart Materials and Structures* 32, no. 5 (2023): 054001, https://doi.org/10.1088/1361-665x/acc624.

[42] D. Zou, M. Zhang, Z. Bai, et al., "Multicategory Damage Detection and Safety Assessment of Post-Earthquake Reinforced Concrete Structures Using Deep Learning," *Computer-Aided Civil and Infrastructure Engineering* 37, no. 9 (2022): 1188–1204, https://doi.org/10.1111/mice.12815.

[43] S. Zhao, F. Kang, and J. Li, "Concrete Dam Damage Detection and Localisation Based on YOLOv5s-HSC and Photogrammetric 3D Reconstruction," *Automation in Construction* 143 (2022): 104555, https://doi.org/10.1016/j.autcon.2022.104555.

[44] G. Ye, J. Qu, J. Tao, W. Dai, Y. Mao, and Q. Jin, "Autonomous Surface Crack Identification of Concrete Structures Based on the YOLOv7 Algorithm," *Journal of Building Engineering* 73 (2023): 106688, https://doi.org/10.1016/j.jobe.2023.106688.

[45] Y. Xu, C. Zhang, and H. Li, "Transformer-Based Large Vision Model for Universal Structural Damage Segmentation," *Automation in Construction* 176 (2025): 106256, https://doi.org/10.1016/j.autcon.2025.106256.

[46] T. Jiang, G. T. Frøseth, A. Rønnquist, X. Kong, and L. Deng, "A Visual Inspection and Diagnosis System for Bridge Rivets Based on a Convolutional Neural Network," *Computer-Aided Civil and Infrastructure Engineering* 39, no. 24 (2024): 3786–3804, https://doi.org/10.1111/mice.13274.

[47] G. Niu, G. Li, C. Wang, and K. Hui, "LBN-YOLO: a Lightweight Road Damage Detection Model Based on Multiscale Contextual Feature Extraction and Fusion," *Structural Control and Health Monitoring* 2025, no. 1 (2025): 5595809, https://doi.org/10.1155/stc/5595809.

[48] R. Raushan, V. Singhal, and R. K. Jha, "Damage Detection in Concrete Structures with multi-feature Backgrounds Using the YOLO Network Family," *Automation in Construction* 170 (2025): 105887, https://doi.org/10.1016/j.autcon.2024.105887.

[49] S. Cano-Ortiz, L. Lloret Iglesias, P. Martinez Ruiz del Árbol, P. Lastra-González, and D. Castro-Fresno, "An end-to-end Computer Vision System Based on Deep Learning for Pavement Distress Detection and Quantification," *Construction and Building Materials* 416 (2024): 135036, https://doi.org/10.1016/j.conbuildmat.2024.135036.

[50] Z. Zhang, J. Wu, W. Song, et al., "ARDs-YOLO: Intelligent Detection of Asphalt Road Damages and Evaluation of Pavement Condition in Complex Scenarios," *Measurement* 242 (2025): 115946, https://doi.org/10.1016/j.measurement.2024.115946.

[51] J. Li, Z. Qu, S. Wang, and S. Xia, "A Method of Road Damage Detection for Complex Background Images Based on Region Guidance Network," *Pattern Recognition* 168 (2025): 111780, https://doi.org/10.1016/j.patcog.2025.111780.

[52] Y. Rong, M. Jia, Y. Zhan, and L. Zhou, "SR-RDFAN-LOG: Arbitrary-Scale Logging Image Super-resolution Reconstruction Based on Residual Dense Feature Aggregation," *Geoenergy Science and Engineering* 240 (2024): 213042, https://doi.org/10.1016/j.geoen.2024.213042.

[53] W. Wang and L. Zhou, "Fracture Extraction from Logging Image Using a Dual encoder-decoder Architecture with Swin Transformer," *Petrophysics* 64, no. 01 (2023): 38–49.

[54] T. Jiang, L. Li, B. Samali, et al., "Lightweight Object Detection Network for Multi-Damage Recognition of Concrete Bridges in Complex Environments," *Computer-Aided Civil and Infrastructure Engineering* 39, no. 23 (2024): 3646–3665, https://doi.org/10.1111/mice.13219.

[55] C. Y. Wang, H. Y. M. Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, and I. H. Yeh, "Cspnet: a New Backbone that Can Enhance Learning Capability of CNN," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (2020), 390–391.

[56] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), 8759–8768, https://doi.org/10.1109/cvpr.2018.00913.

[57] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "Mobilenetv2: Inverted Residuals and Linear Bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), 4510–4520, https://doi.org/10.1109/cvpr.2018.00474.

[58] A. Howard, M. Sandler, B. Chen, et al., "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision* ((2019), 1314–1324, https://doi.org/10.1109/iccv.2019.00140.

[59] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, no. 9 (2015): 1904–1916, https://doi.org/10.1109/tpami.2015.2389824.

[60] C. Li, L. Li, Y. Geng, et al., "Yolov6 v3. 0: a full-scale Reloading" (2023).

[61] J. Hu, L. Shen, and G. Sun, "Squeeze-And-Excitation Networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), 7132–7141, https://doi.org/10.1109/cvpr.2018.00745.

[62] L. Yang, R. Y. Zhang, L. Li, and X. Xie, "Simam: a Simple, Parameter-free Attention Module for Convolutional Neural Networks," in *International Conference on Machine Learning* (PMLR, 2021).

[63] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "Cbam: Convolutional Block Attention Module," in *Proceedings of the European Conference on Computer Vision* (ECCV, 2018).

[64] Q. L. Zhang and Y. B. Yang, "Sa-net: Shuffle Attention for Deep Convolutional Neural Networks," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2021).