Measurement Science
and Technology

PAPER

An improved cross-dimensional collaborative bridge defect detection model based on UAV images

To cite this article: Jianjun Ni *et al* 2025 *Meas. Sci. Technol.* **36** 105007

IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **36** (2025) 105007 (16pp) https://doi.org/10.1088/1361-6501/ae09c0

An improved cross-dimensional collaborative bridge defect detection model based on UAV images

Jianjun Ni*, Qibo Ji, Yonghao Zhao, Weidong Cao and Pengfei Shi

College of Artificial Intelligence and Automation, Hohai University, Changzhou 213200, Jiangsu, People's Republic of China

E-mail: njjhhuc@gmail.com

Received 19 July 2025
Accepted for publication 22 September 2025
Published 13 October 2025

Abstract

Concrete bridges are critical components of urban infrastructure, and their structural health directly influences the safety and efficiency of urban transportation. However, existing bridge defect detection methods often focus on single defect type and require manual data collection, which is both time-consuming and labor-intensive. Although recent advancements in unmanned aerial vehicle (UAV) technology have significantly improved the efficiency of image acquisition, challenges such as varying viewing angles, illumination conditions, and complex environmental backgrounds in captured images continue to hinder the accuracy of existing methods for bridge defect detection. To address these limitations, we propose a cross-dimensional collaborative You Only Look Once model (CDC-YOLO), an improved defect detection network designed for multi-type defect detection in concrete bridges based on UAV images. In this proposed CDC-YOLO model, a multi-dimensional feature extraction module is presented to capture both shallow and detailed features, ensuring the accurate identification of fine crack defects in real-world scenarios. Then, a dynamic feature recombination module is proposed to improve the adaptability of the network in detecting irregular defect shapes and distributions in complex scenes. In addition, an adaptive feature fusion module is designed, which overcomes the limitations of traditional fusion methods by effectively mitigating false positives and missed detections caused by complex environmental interference, illumination changes, and varying viewing angles. Finally, various experiments are conducted, and the quantitative and qualitative results demonstrate the superior performance of the proposed model over state-of-the-art defect detection methods, particularly in detecting small defects with low contrast against their environmental backgrounds.

Keywords: defect detection, concrete bridges, deep learning, YOLO, UAV images

1. Introduction

Concrete infrastructure forms the backbone of urban development, with its longevity and reliable operation hinging on systematic and rigorous maintenance [1]. In transportation, for instance, urban infrastructure such as bridges, overpasses, and tunnels, predominantly constructed with concrete, is prone to fatigue, aging, and damage over extended use [2, 3]. These issues arise from the combined effects of natural factors, including adverse weather conditions and geological disasters, as well as the constant load of vehicular traffic. Without real-time monitoring, structural defects in bridges may escalate to failures or collapses, resulting in significant economic losses,

* Author to whom any correspondence should be addressed.

infrastructure damage, and threats to public safety. Hence, regular, effective, and precise inspections of bridge structures are paramount for identifying defects early and enabling targeted maintenance to enhance their lifespan and ensure safety [4, 5].

Traditional concrete bridge inspections largely rely on experienced personnel to visually assess structural conditions during on-site patrols, sometimes with assistance from acoustic or laser-based equipment [6]. However, these methods are labor-intensive, time-consuming, and often impractical in scenarios where traffic disruptions or difficult-to-access areas hinder close inspections. In some cases, such visual inspections can be cumbersome or pose safety risks. The limitations of traditional methods, coupled with advances in computer vision and sensor technologies such as digital cameras, have driven the adoption of automated defect detection approaches [7]. By integrating computer vision with UAVs, remote inspections of critical bridge components can be conducted efficiently [8, 9]. In addition, intelligent detection technology based on UAVs has been widely applied in the field of transportation infrastructure. Through automatic cruise, high-definition image collection and artificial intelligence analysis, it has significantly improved the efficiency and safety of inspection [10]. For example, this technology can be applied in construction supervision and maintenance of cross-sea bridges, promoting the transformation of transportation infrastructure towards intelligent and digital operation and maintenance [11].

Most existing defect detection models are tailored to specific defect types, such as cracks or spalling, and typically assume that the detection images contain isolated defects against clean, uniform backgrounds [12–14]. These methods often rely on preprocessing steps to remove irrelevant portions of the image. However, real-world scenarios involving UAVs introduce considerable variability in defect appearance and size due to differences in shooting angles, resolutions, and camera coverage. This variability complicates accurate detection. Additionally, small-sized defects in UAV images with low contrast against uneven backgrounds further challenge detection systems. Overlapping defects of similar types in the same location exacerbate the risk of false positives and missed detections [15, 16]. These challenges severely limit the applicability of traditional methods, which are more suited to simplified, structured scenarios [17]. Thus, improvements in detection models are essential to meet the demands of real-world applications characterized by complex conditions and diverse defect distributions for UAV images.

Early approaches to concrete defect detection typically focused on designing handcrafted features to describe specific defects [18]. While effective for single defect types, their performance diminishes significantly in scenarios involving multiple defect types [19, 20]. In recent years, deep learning (DL) has achieved state-of-the-art results in defect detection, classification, and segmentation, driven by advances in large-scale dataset annotation and computational capabilities [21]. Convolutional neural networks (CNNs) have become the dominant approach in visual tasks. Through alternating layers of convolution, pooling, and nonlinear activation, CNNs can directly learn robust hierarchical features from input images. Existing CNN-based defect detection models are broadly classified into single-stage and two-stage approaches.

Single-stage methods, represented by models like the single-shot multi-box detector (SSD) and RetinaNet [22], directly predict object categories and locations from input images, bypassing additional candidate region generation steps. These models offer superior detection speed but often trade off accuracy. In contrast, two-stage methods, such as regions with CNN (R-CNN) and mask R-CNN, generate high-quality candidate regions in the first stage, followed by precise classification and localization in the second stage, resulting in higher accuracy but at the cost of increased computational complexity and training time [23]. In addition, transformer-based architectures have emerged as a powerful alternative, enabling efficient feature extraction through multi-head attention mechanisms [24]. Their flexibility and high performance have made them increasingly popular in object detection tasks [25]. For applications requiring both efficiency and precision, single-stage methods are more suitable.

The You Only Look Once (YOLO) series, as a prominent example of single-stage detection models, have been widely adopted for defect detection. YOLOv11 (the latest version YOLO model) achieves state-of-the-art results in object detection, delivering superior accuracy, faster inference speeds, and improved adaptability to small object detection. YOLOv11 employs an anchor-free strategy. Compared with the traditional anchor box method that requires predefined anchor box sizes and proportions for each scale, the anchor-free box detection method directly regresses the center point and size information of the object, significantly reducing the model's reliance on data distribution and avoiding the complexity of prior box calculations. It offers faster inference speed and more efficient calculations. YOLOv11 is better suited to meet the growing demands of modern industrial applications and is easier to deploy in real-world scenarios [26]. However, the research on concrete bridge defect detection based on UAVs is confronted with challenges such as small defect comparison dimensions, low discrimination between defects and the background environment, and single detection categories.

To address these challenges, we propose an cross-dimensional collaborative YOLO (CDC-YOLO) framework for bridge defect detection based on UAV images, building on the foundation of YOLOv11. CDC-YOLO introduces a multi-dimensional feature extraction module (MDFEM) within the backbone network, enabling the cross-fusion of global information through pointwise convolution (PWConv) and inter-dimensional collaboration mechanisms. Additionally, the proposed model incorporates an adaptive feature fusion module (ADFFM) and a dynamic feature recombination module (DRFEM) in the neck. ADFFM enhances traditional feature fusion methods by integrating global average pooling and fully connected layers to generate adaptive feature weight matrices, facilitating more effective feature integration via element-wise matrix multiplication. Inspired by content-aware sampling, DRFEM employs dynamic offset sampling and bilinear interpolation to capture finer-grained feature information, ensuring smoother feature mapping. In summary, the contributions of this paper include:

- This paper presents a novel MDFEM module, designed to achieve detailed feature representation through dimensional displacement and mixed pooling. MDFEM jointly analyzes the three dimensions of channel, height and width, achieving the comprehensive integration of features from different dimensions and solving the problems of strong background interference and complex and diverse defect types faced in the inspection of concrete bridges.
- This paper designs an improved DRFEM module, to overcome the shortcomings of traditional sampling methods faced with complex scenes. DRFEM dynamically predicts a recombination kernel based on low-level feature information and utilizes an offset prediction mechanism to calculate pixel-wise offsets within the feature map. Dynamic sampling is performed using these offsets to generate recombination weights. The dynamic migration mechanism flexibly adjusts the sampling points' positions according to the input features' spatial structures, enhancing the module's ability to capture fine-grained details of irregular defects and accurately reconstruct regional details.
- This paper proposes a novel ADFFM module, to improve feature integration capability. By leveraging a combination of global average pooling and fully connected layers, ADFFM generates adaptive feature weights, thereby mitigating the risk of overfitting. Through element-wise multiplication, ADFFM achieves cross-scale weighted fusion through element-by-element multiplication, effectively integrating and enhancing the detailed and structural information in the original features, and improving the model's ability to pay attention to minor defects in complex backgrounds.

The rest of this paper is structured as follows: section 2 reviews the related work; section 3 details the proposed model for defect detection of the concrete bridge; section 4 presents the experimental results; section 5 discusses the results of the ablation experiments and the generalization of the proposed model; section 6 concludes the paper.

2. Related work

This section reviews the mainstream models for concrete defect detection developed in recent years, with a particular focus on the application of two-stage and single-stage object detection approaches in the field of defect identification.

2.1. Two-stage defect detection

Two-stage defect detection models operate in two distinct phases: candidate region generation and defect classification with precise localization. In the first phase, the model employs a region proposal algorithm to identify a set of candidate regions that potentially contain defects. These candidate regions are represented as rectangular bounding boxes within the image, marking areas where defects are likely to exist. In the second phase, the candidate regions generated in the first phase are further analyzed to determine whether they indeed contain defects. This phase involves classifying the detected defects and accurately localizing them within the candidate regions, ensuring a refined and detailed defect detection process.

R-CNN and its variants represent classical and widely adopted DL models in two-stage defect detection. To further enhance detection performance, numerous studies have proposed improvements based on R-CNN and its derivatives. For example, Kim *et al* [27] employed R-CNN to detect cracks in concrete bridges, integrating candidate regions generated through selective search with feature information extracted by CNN. Additionally, the model was adapted for UAV platforms to improve detection efficiency. Haciefendioglu and Basaga [28] applied faster R-CNN to detect cracks in concrete roads. Wei *et al* [29] introduced a concrete surface pit detection method based on mask R-CNN, which combines target detection and instance segmentation. This approach significantly facilitates advanced visual tasks that follow defect identification. Similarly, Xu *et al* [30] developed a method for detecting and localizing seismic damage in reinforced concrete. Their method uses a region proposal network to generate initial bounding boxes for damage, which are then refined through faster R-CNN, enabling highly accurate defect detection and localization. However, these models still have some limitations. For example, due to the reliance of DL models on a single feature map for prediction, their performance in detecting multi-scale defects remains limited, and they are generally restricted to single-defect detection scenarios.

In addition to directly utilizing R-CNN and its variants for defect detection tasks, other algorithms have also demonstrated effectiveness in accurately identifying bridge defects. For instance, Yao *et al* [31] proposed a pothole detection method based on GoogleNet, which is capable of detecting defects even when trained on a limited dataset and under challenging conditions such as uneven lighting and shading. Mishra *et al* [32] developed a two-stage automated inspection method using YOLOv5 for identifying, locating, and quantifying cracks in general concrete structures. Similarly, Pang *et al* [33] introduced a bridge defect detection method, which enhances the accuracy of small-target detection by incorporating fuzzy Petri nets, effectively addressing the issue of defect misdetection. Wan *et al* [34] proposed a method based on Vision Transformers for concrete defect detection, demonstrating superior performance in feature extraction and representation, particularly in complex scenarios.

The aforementioned bridge defect detection tasks based on two-stage detection models perform well when dealing with single defect type. However, as the number of defect categories increases, the performance of these models tends to degrade significantly. Additionally, since two-stage detection algorithms rely on generating defect candidate regions in the initial stage, the computational complexity of the overall model is substantially higher. This results in elevated computational costs, making it challenging for such methods to meet the requirements of real-time bridge defect detection tasks.

2.2. Single-stage defect detection

Unlike two-stage defect detection, single-stage defect detection integrates candidate region generation and defect classification into a single inference process. By utilizing a unified network structure, it directly predicts the location and category of defects from the input images. This streamlined approach offers higher detection efficiency and significantly lower computational cost compared to two-stage detection methods.

The YOLO-based models and the SSD-based models are widely utilized in single-stage defect detection. However, SSD often underperforms in small-target detection tasks, making it less suitable for bridge defect detection applications. In contrast, YOLO-based models have demonstrated superior performance in this domain. For instance, Teng *et al* [35] proposed crack detection methods based on YOLOv2, achieving impressive results in near real-time image detection tasks. Cui *et al* [36] further improved YOLOv3 for detecting concrete corrosion by employing Darknet53 as the backbone network and incorporating the Mish activation function. Wu *et al* [37] addressed these limitations by employing the YOLOv4 model for crack detection. They introduced a pruning strategy to reduce the excessive number of neural network parameters, significantly enhancing detection speed while maintaining high accuracy. Despite these models having been used widely, they still faced challenges. Some of these models exhibited limitations when handling images with varying aspect ratios and were restricted to single-defect detection scenarios. Others have limitations in detecting defects of varying scales.

To deal with these problems, some improved models have been presented. For example, Wang *et al* [38] proposed an automated one-stage concrete defect detection method utilizing EfficientNetB0 as the backbone network. To enhance detection accuracy, this method extracts feature information from three scales as input to the detector and employs upsampling to fuse low-level and high-level features effectively. Similarly, Kumar *et al* [39] applied YOLOv3 to detect defects such as flaking and cracking. In another study, Zou *et al* [40] integrated YOLOv4 with separable convolution to detect various defects, including cracks, flaking, and exposed rebar. This approach significantly reduced computational costs while improving detection speed, offering a practical solution for efficient and real-time defect detection tasks.

Although the aforementioned single-stage detection models demonstrate strong performance in identifying a single type of defect, their effectiveness diminishes when addressing multiple defect types. This limitation becomes more pronounced in drone-captured images of bridge defects, which are often affected by varying lighting conditions, changes in viewpoints, and differences in scale. Furthermore, bridge defect detection frequently encounters overlapping defects of different types or defects that are poorly distinguishable from complex environmental backgrounds. These challenges can lead to false positives or missed detections, significantly compromising detection accuracy. Furthermore, if the batch analysis method of 'flight—unloading—offline processing' is adopted, the missed inspection areas cannot be detected in real time during the flight route execution. Once the omission is discovered at the back end, an additional return flight will be required, which not only wastes the limited flight time but also increases the labor and logistics costs.

To address these issues, this study proposes an improved defect detection model based on YOLOv11, which is designed for multi-type defect detection. The proposed model aims to enhance the accuracy and robustness of detection under challenging conditions, providing a reliable solution for real-world bridge defect identification on UAV images.

3. Proposed model

This section introduces the proposed CDC-YOLO, a concrete bridge defect detection model developed based on YOLOv11, as illustrated in figure 1. The model incorporates three primary innovations: the MDFEM, the DRFEM, and the ADFFM.

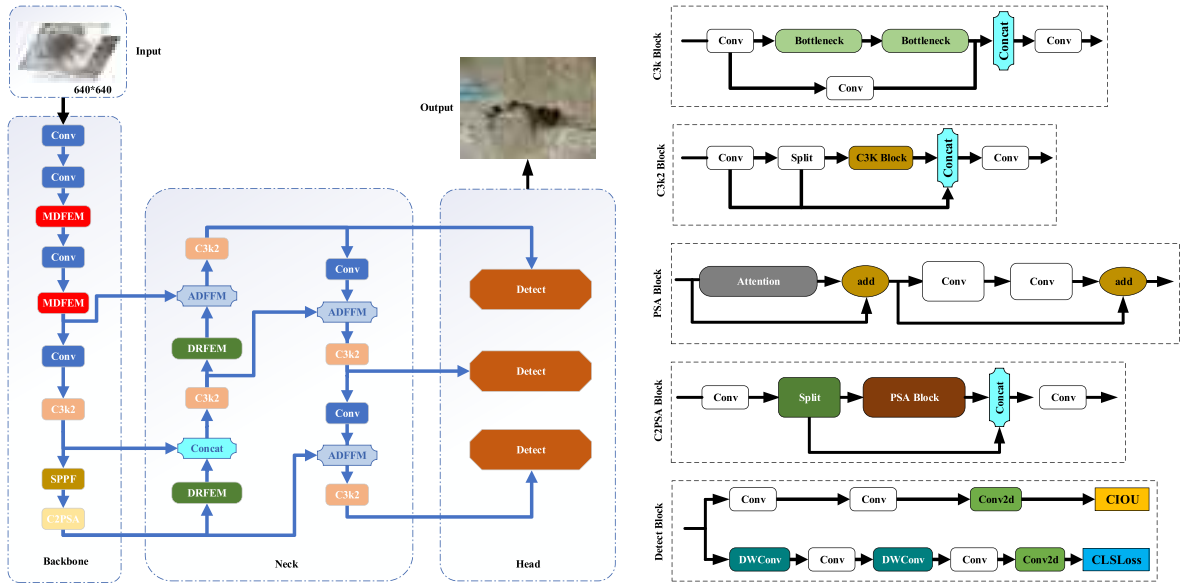
3.1. Multi-dimensional feature extraction module

In UAV-based bridge defect detection tasks, defects typically occupy only a small portion of the image background, and certain defect types are difficult to distinguish from complex surroundings [41]. Therefore, it is very important to effectively locate the defect area and extract feature information in the backbone network. Although increasing the depth of the network or the complexity of the module can improve the detection performance, it will significantly increase the computational overhead of the network. In response, this paper proposes a MDFEM module (see figure 2), specifically designed to enhance the model's performance in bridge defect detection.

As shown in figure 2, the proposed MDFEM module integrates the PWConv, global maximum pooling (GMP), and an inter-dimensional collaboration mechanism (IDCM) to perform multi-dimensional feature extraction. It is well known that the excessive stacking of convolutional layers can lead to the excessive expansion of the receptive field in feature extraction. In the bridge defect detection task based on UAVs, this excessive expansion may lead to the network focusing on irrelevant areas. Compared with other lightweight convolutions, PWConv can effectively enhance the discriminative power of the defect area by fusing features in the channel dimension, suppress background interference, keep the spatial information unchanged, and will not increase excessive computational overhead. Therefore, PWConv is adopted instead of the traditional convolution operation in this study [42].

Specifically, for the given input feature map $X \in R^{C\times H\times W}$, X is first processed using PWConv to adjust the channel dimensions, resulting in an output with C_{out} channels, $X \in R^{C_{\text{out}}\times H\times W}$. The feature map is then divided along the channel dimension into two branches: $X_1 \in R^{\frac{1}{2}C_{\text{out}}\times H\times W}$ and $X_2 \in R^{\frac{1}{2}C_{\text{out}}\times H\times W}$, with each branch allocated to a distinct processing pathway.

Existing detection networks often emphasize channel information while neglecting the spatial dimension's feature information [43]. To address this limitation, we introduce



(a) (b)

Figure 1. The structure of CDC-YOLO, where MDFEM means a multi-dimensional feature extraction module, ADFFM means an adaptive feature fusion module, and DRFEM denotes a dynamic feature recombination module: (a) the overall architecture of CDC-YOLO; and (b) the details of some blocks in the model.

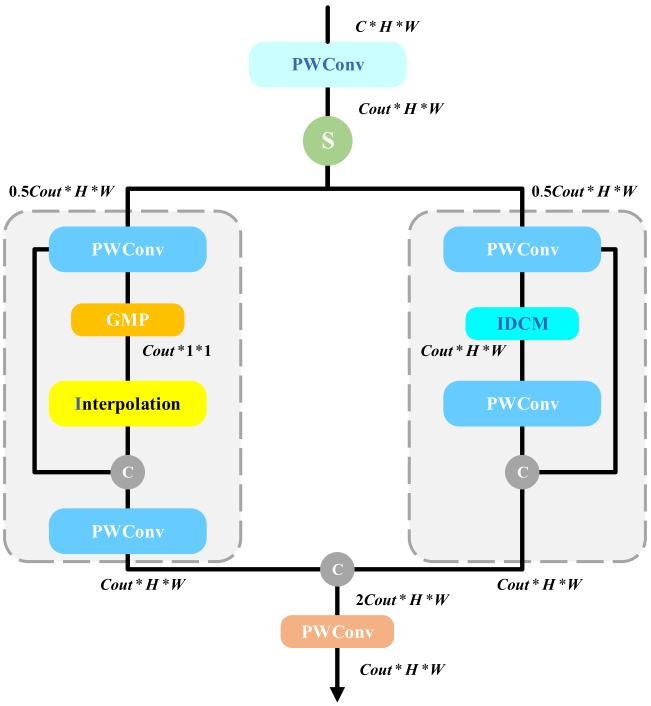


Figure 2. Details of the proposed MDFEM module, where PWConv denotes the pointwise convolution, GMP is the global maximum pooling, and IDCM is an inter-dimensional collaboration mechanism.

IDCM (see figure 3), to effectively capture defect information in the spatial dimension. In the second branch, the feature map is passed through a PWConv to expand the channels before being input into IDCM. Within IDCM, mixed pooling and standard differential operations are applied to the feature map across three dimensions-channel, height, and width. The weighted outputs from these operations are concatenated along the channel dimension and then subjected to another PWConv to compress the channels. To enhance feature representation, MDFEM incorporates residual connections in both branches, facilitating the fusion of multi-scale defect information.

Specifically, given an input feature map $F \in R^{C\times H\times W}$, it is decomposed into three components: $F_1 \in R^{C\times H\times W}$, $F_2 \in R^{H\times C\times W}$, and $F_3 \in R^{W\times C\times H}$. These components are individually subjected to a squeeze transformation using a combination of mixed pooling and standard differential pooling, yielding $\tilde{F}_1 \in R^{C\times 1\times 1}$, $\tilde{F}_2 \in R^{H\times 1\times 1}$, and $\tilde{F}_3 \in R^{W\times 1\times 1}$.

Each transformed feature, $\tilde{F}_1$, $\tilde{F}_2$, and $\tilde{F}_3$, is then processed through a Conv module designed to capture specific dimensional features. The Conv module consists of a $1 \times K$ convolution kernel ($K = 3$ in this study), followed by a standard 2D convolution layer, batch normalization, and an activation function.

After normalization through the Sigmoid activation function, the enhanced feature maps are obtained by element-wise multiplication with the original input features F_1, F_2, and F_3. The enhanced feature maps, $\hat{F}_1$, $\hat{F}_2$, and $\hat{F}_3$, are then permuted back to their original dimensions. Finally, a weighted averaging operation combines these features to produce the output F_{out}.

This process can be summarized by:

$$F_* = \text{PM}_i(F) \tag{1}$$

$$\tilde{F}_* = \text{PL}(F_*) \tag{2}$$

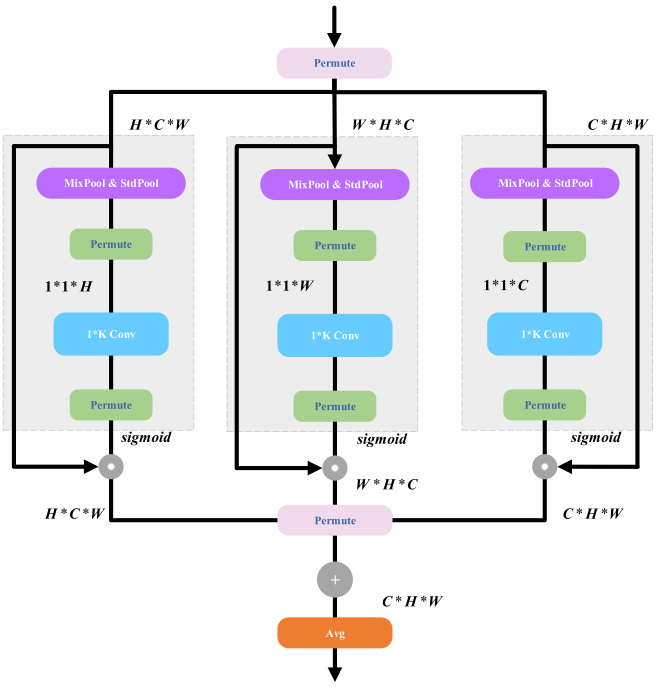


Figure 3. Details of the inter-dimensional collaboration mechanism (IDCM).

$$\hat{F}_* = \sigma\left(\mathrm{Conv}\left(\tilde{F}_*\right)\right) \otimes F_* \tag{3}$$

$$F_{\mathrm{out}} = \mathrm{Avg}\left(\sum_{i=1}^{3} \mathrm{PM}_i^{-1}\left(\hat{F}_*\right)\right) \tag{4}$$

where, i represents the channel direction C, the height direction H, and the width direction W; the $\mathrm{PM}_i(\cdot)$ denotes the replacement operations along the channel, height, and width; $\mathrm{PM}_i{}^{-1}(\cdot)$ is the inverse permutation operation; $\mathrm{PL}(\cdot)$ represents both mixed pooling and standard difference pooling operations. Among them, the hybrid pooling is the weighted average sum of the GMP and the global average pooling. $\sigma(\cdot)$ indicates the excitation transformation operation; and $\otimes$ represents element-by-element multiplication.

The proposed MDFEM module achieves cross-scale feature fusion within a single path, maintaining or even enhancing the detection performance of small targets without the need for a multi-branch structure. In addition, this module introduces a collaborative mechanism of channel splitting and separable convolution: channel splitting ensures the independent expression of different feature channels, while separable convolution further reduces the computational load and the number of parameters.

3.2. Adaptive feature fusion module

In the original YOLOv11 network, feature fusion in the neck network is achieved through concatenation along the channel dimension. However, this fusion strategy may inadvertently amplify irrelevant or redundant features. Complex backgrounds, variations in lighting, and other non-defect-related

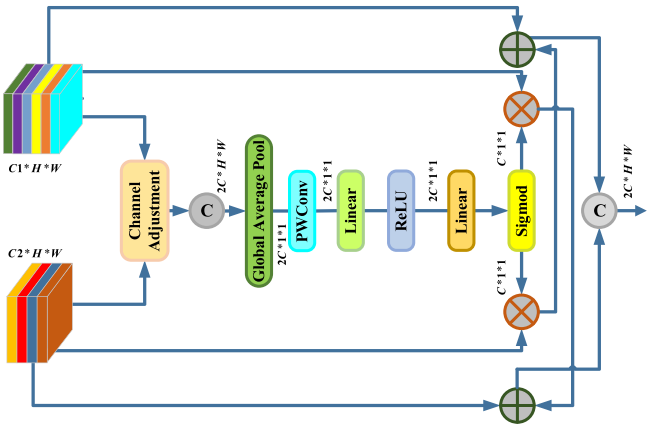


Figure 4. Details of the proposed ADFFM module.

elements may become overly emphasized, thereby interfering with the model's ability to accurately classify defects and significantly increasing the false detection rate [44]. Meanwhile, simple cascading often fails to achieve full complementarity between local and global features. This limitation weakens the model's ability to effectively handle detailed information and global structural features. To address these challenges, this paper proposes an ADFFM, as illustrated in figure 4.

Specifically, in the proposed ADFFM module, the feature maps of different scales, denoted as F_i and F_j, are first adjusted to have the same number of channels through channel adjustment operations. Then, the feature weight $\tilde{F}$ is computed using convolution, pooling, and fully connected layers.

Based on the bidirectional feature enhancement mechanism, $\tilde{F}$ is applied to the input features via element-wise multiplication. This results in the enhanced feature maps F_i' and F_j', which are obtained by element-wise addition of $\tilde{F}$ to the input features from another branch. By jointly learning the attention weights, the model automatically selects the most helpful channel for the other one and suppresses the interfering channels. During the integration process, each path only borrows the information it needs from the other party to supplement itself, thereby maximizing two-way integration and enhancing complementarity. In this way, the deep features are cross-superimposed and fused with the shallow features of different dimensions through a bidirectional feature enhancement mechanism [45]. Finally, the enhanced feature maps of different scales are spliced and fused.

This bidirectional feature enhancement fusion process of the proposed ADFFM module can be summarized as follows:

$$F_i = \mathrm{CT}\left(\mathrm{re}\left(F_i\right), \mathrm{re}\left(F_j\right)\right) \tag{5}$$

$$\bar{F} = \mathrm{GAP}\left(\mathrm{PW}\left(F_i\right)\right) \tag{6}$$

$$\tilde{F} = \sigma\left(\mathrm{FC}\left(\bar{F}\right)\right) \tag{7}$$

$$F_i' = F_i \oplus \left(F_j \otimes \tilde{F}\right),\ F_j' = F_j \oplus \left(F_i \otimes \tilde{F}\right) \tag{8}$$

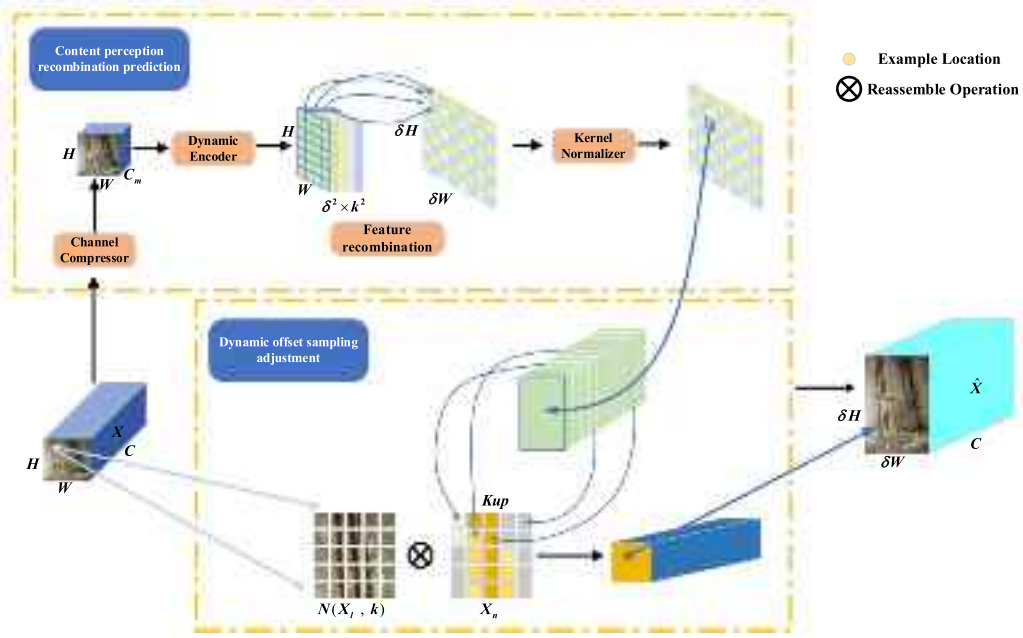


Figure 5. Details of the DRFEM module.

where, $\mathrm{CT}(\cdot)$ represents concatenation operations in the channel dimension; $\mathrm{re}(\cdot)$ indicates feature remodeling, ensuring the same number of feature channels for different dimensions of the input; $\mathrm{PW}(\cdot)$ and $\mathrm{GAP}(\cdot)$ represent point-by-point convolution and global averaging pooling, respectively; $\sigma(\cdot)$ and $\mathrm{FC}(\cdot)$ represent the *Sigmoid* function and the fully connected layer, respectively; $\oplus$ and $\otimes$ table element-by-element summation and element-by-element multiplication, respectively.

Finally, we get the output feature F_{out}, which combines semantic information of different scales, namely:

$$F_{\mathrm{out}} \quad = \mathrm{CT}\left(F_i', F_j'\right). \tag{9}$$

By assigning different weights to features of various scales, the proposed ADFFM module optimizes the feature fusion process. Furthermore, ADFFM incorporates a bidirectional feature enhancement mechanism to further integrate contextual information. This mechanism not only alleviates the limitations of single-feature representation but also significantly enhances the model's comprehensive perception of both local details and global textures.

3.3. Dynamic feature recombination module

Feature upsampling is a crucial operation in many modern convolutional network architectures, and it plays a vital role in bridge defect detection tasks. Traditional fixed interpolation not only has a large amount of calculation, but also often blurs the edge details of irregular defects, which is not robust for complex textures or changes in viewing angles. To address these issues, this paper proposes a novel DRFEM, which combines dynamic migration sampling with content-aware recombination [46, 47], as shown in figure 5. The operation of DRFEM consists of two key components:

(1) Content-aware recombination prediction. For the input feature map $X \in R^{C\times H\times W}$, where C, H, and W represent the number of channels, height, and width respectively. DRFEM first maps the output spatial coordinates $l = (I, J)$ to the low-resolution source points $l = (I', J')$ according to the upsampling multiple δ, namely, $(I' = I/\delta)$ and $(J' = J/\delta)$. Then extract the $k \times k$ neighborhood block of this position $X_{k\times k} \in R^{C\times k\times k}$.

Then, the recombination kernel is generated through the convolutional encoder $\mathrm{Conv}_{\mathrm{enc}}$. The weights of neighborhood pixels are dynamically allocated for each output pixel based on its local content

$$W_l = \mathrm{Conv}_{\mathrm{enc}}\left(X_{k\times k}\right) \in R^{k^2} \tag{10}$$

where, W_l represents the normalized weight of the nth pixel in the neighborhood. This adaptive weight can automatically adjust the contribution of each point to the reconstruction according to the local texture, strengthen the high-frequency information of the edge of irregular defects, enhance the retention ability of the fracture edge and the robustness to changes in viewing angles and textures.

(2) Dynamic offset sampling adjustment. DRFEM contains an offset predictor $\Psi(X)$, which predicts the feature map of the output of the content-aware recombination prediction part:

$$X_{\mathrm{carafe}} \in R^{C\times \delta H\times \delta W}. \tag{11}$$

Then, perform a 1×1 convolution to generate the offset for each resampling point:

$$\Delta p_l = \Psi\left(X_l\right) \in R^2. \tag{12}$$

Algorithm 1. The pseudocode of the proposed DRFEM module.

Input: $\mathbf{X}$; % $\mathbf{X}$ is the input feature map;
Output: $\mathbf{X}_{\text{out}}$;
Initialize parameters:
scale $\leftarrow$ 2 % sampling scale factor
groups $\leftarrow$ 4 % number of grouped convolutions
k_up $\leftarrow$ 5 % upsampling kernel size
k_enc $\leftarrow$ 3 % encoding convolution kernel size
Content perception recombination prediction:
$\mathbf{W}_{\text{comp}} \leftarrow \text{Conv}(\mathbf{X}, c_{\text{mid}})$ % Convolution operation on input features;
$\mathbf{W}_{\text{enc}} \leftarrow \text{Conv}(\mathbf{W}_{\text{comp}}, (\text{scale} \cdot \text{k\_up})^2)$ % Convolution to expand the features;
$\mathbf{W} \leftarrow \text{PixelShuffle}(\text{Softmax}(\mathbf{W}_{\text{enc}}, \text{dim} = 1))$ % Pixel shuffle and softmax activation;
$\mathbf{X}_{\text{unfold}} \leftarrow \text{Unfold}(\mathbf{X}, \text{k\_up}, \text{k\_up}//2 \cdot \text{scale})$ % Unfolding the input features;
$\mathbf{X}_{\text{reshaped}} \leftarrow \text{Reshape}(\mathbf{X}_{\text{unfold}}, (b, c, \text{k\_up}^2, h, w))$ % Reshaping the unfolded features;
$\mathbf{X}_{\text{Cprp}} \leftarrow \text{Einsum}(bchw', [\mathbf{W}, \mathbf{X}_{\text{reshaped}}])$ % Content perception recombination prediction;
Dynamic offset sampling adjustment:
$\text{offset} \leftarrow \text{Offset}(\mathbf{X}_{\text{Cprp}}) \cdot \alpha + \text{Init\_Pos} \cdot \beta$ % Offset adjustment based on initial position;
if dyscope **then**
    $\text{scope} \leftarrow \text{Sigmoid}(\text{Scope}(\mathbf{X}_{\text{Cprp}}))$ % Apply sigmoid to scope function;
    $\text{offset} \leftarrow \text{offset} \cdot \text{scope} \cdot \gamma$ % Adjust the offset with scope;
end if
$\mathbf{X}_{\text{out}} \leftarrow \text{Sample}(\mathbf{X}_{\text{Cprp}}, \text{offset})$ % Sampling based on the adjusted offset;
return $\mathbf{X}_{\text{out}}$ % Return the output feature map;

Simultaneously, the scale factor controls the range of the offset, which is then applied to the initial sampling location, resulting in the final sampling neighborhood. This process can be expressed as follows:

$$N_{(X_l,k)} = \Psi\left(X_l\right) \alpha \ + \ \text{init\_pose}\ \beta \tag{13}$$

where, α is the offset scaling factor, β is the initial grid weight, and init_pose $\in R^2$ is the standard grid coordinate with the pixel center as the origin. In this study, α and β are set to 0.25 and 1.0 respectively based on empirical values. Finally, the grid is normalized to $[-1, 1]$ and dynamic sampling is completed through bilinear interpolation to adapt to the geometric changes of irregular structures such as cracks.

Combining the above two steps, the final upsampling result of DRFEM at the output position l is

$$\hat{X}_l \ = \ \sum_{n \in N_{(X_l,k)}} W_l(n)\ X_n \tag{14}$$

where, X_n represents the sampling feature of the original image on the dynamic neighborhood $N_{(X_l,k)}$. Through content-aware weighted recombination and spatial-adaptive dynamic offset, DRFEM effectively enhances the robustness against UAV view distortion and complex backgrounds while retaining the microstructure details of the bridge.

In summary, the pseudocode of the DRFEM module can be represented by algorithm 1.

In the proposed DRFEM module, each location not only utilizes the underlying structural information to predict an adaptive recombination kernel, but also provides offset adjustments for each sampling point through dynamic offset prediction. This enables optimized sampling and recombination of defect areas. This capability is particularly crucial for handling complex details, such as cracks or irregular corrosion areas, in bridge defect detection.

4. Experiments and results analyses

4.1. Implementation details

To evaluate the performance of the proposed CDC-YOLO model, we conducted some comparison experiments on two datasets:

(1) The publicly available BRidge DEfect BRidge IMage dataset (CODEBRIM) [48]. This dataset, which is based on drone photography, contains 1590 high-resolution images of concrete bridge defects across six categories: background, crack, spallation, exposed bars, efflorescence, and corrosion. As shown in figure 6, the environmental background outside the target defect is of no help to our task. Therefore, we excluded background category and focused on the defect categories most relevant to the safety of concrete bridges: cracks, efflorescence, spallation, exposed bar, corrosion. This reduction allowed us to better align the dataset with the primary objectives of this study.

To ensure the model performs well in bridge defect detection, it is crucial to train on a diverse set of data. In this study, we applied some image enhancement methods to the remaining 1018 labeled images, after eliminating any unlabeled and poor-quality samples from 1590

Figure 6. Five types of bridge defects in the CODEBRIM dataset: (a) exposed bars; (b) crack; (c) efflorescence; (d) spallation; (e) corrosion.

Figure 7. Eight types of bridge defects in the BDD_2K dataset: (a) shrinkage crack; (b) crack; (c) corrosion; (d) road deterioration; (e) void; (f) subfloor shrinkage crack; (g) dampness; (h) degraded concrete.

original images containing defects. The image enhancement methods include image translation, horizontal flipping, random rotation and the addition of Gaussian noise. Through random application of these methods, we generate a total of 5090 defect images. We partition the dataset with a training-to-validation ratio of 9:1 to ensure that the model's performance could be thoroughly evaluated during the training and validation process.

(2) BDD_2K dataset: The CODEBRIM dataset contains a limited number of defect categories, so evaluation experiments on a single dataset are insufficient to establish the reliability of the proposed model. We created a BDD_2K dataset for further validation. This dataset, built by combining a subset of images extracted from the DACL10K dataset [49] with bridge defect images collected from the Internet, contains 2000 images representing 8 categories of bridge defects: corrosion, cracks, spalling, voids, weathering, shrinkage cracks, floor shrinkage cracks, and road deterioration (see figure 7). In the experiment, the data set was also divided into the training set and the verification set with a ratio of 9:1.

In this study, all experiments are implemented using the PyTorch framework on a single NVIDIA GeForce RTX 3080 Ti graphics processor with 8GB of memory. In all experiments, the resolution of all images is adjusted to 640 × 640. The specific parameter settings are shown in table 1, using the Stochastic Gradient Descent (SGD) optimizer, and

Table 1. Hyperparameter settings.

Parameter	Value
Learning rate	0.01
Momentum	0.937
Weight decay	0.0004
Epochs	400
Optimizer	SGD
Mosaic	10
Warmup-epochs	4.0

python version 3.9.19. The loss function uses the sum of BCEWithLogitsLoss (L_{BCE}), CIOU (L_{CIoU}) and distributed focal loss (L_{DFL}), namely

$$L_{\mathrm{total}} = L_{\mathrm{BCE}} + L_{\mathrm{CIoU}} + L_{\mathrm{DFL}}. \tag{15}$$

To accurately evaluate the performance of the proposed model, we employ five widely used metrics in this study: $F1$ score, mAP@50, flops (floating point of operations), and parameters. These indicators are calculated as follows:

$$F1 = \frac{2 \times P \times R}{P + R} \tag{16}$$

$$\mathrm{mAP50} = \frac{1}{C} \sum_{i=1}^{C} \mathrm{AP}_i^{0.5} \tag{17}$$

Table 2. Experimental results on the CODEBRIM and BDD_2K datasets.

Model	Parameters (M)	Flops (G)	CODEBRIM		BDD_2K	
			*F*1	mAP@50	*F*1	mAP@50
EfficientDET	6.600	6.1	0.66	0.608	0.39	0.304
SSD	45.020	39.5	0.71	0.702	0.45	0.398
Faster-RCNN	41.384	195.0	0.73	0.671	0.49	0.412
Gold-YOLO	21.502	46.0	0.81	0.788	0.43	0.378
RT-DETR	31.994	103.5	0.79	0.790	0.40	0.362
YOLOv8	9.830	23.4	0.85	0.843	0.57	0.519
YOLOv11	9.692	27.0	0.87	0.855	0.55	0.507
Deformable-DETR	40.000	173.0	0.81	0.802	0.39	0.351
YOLO-NAS	19.000	32.0	0.80	0.791	0.41	0.402
CDC-YOLO (Ours)	9.867	32.5	0.89	0.893	0.58	0.536

where, C represents the number of defect classes, $AP_i^{0.5}$ represents the average precision of class under the IoU threshold of 0.5, the accuracy (P) and recall rate (R) are calculated as follows:

$$P = \frac{TP}{TP + FP} \tag{18}$$

$$R = \frac{TP}{TP + FN} \tag{19}$$

where, TP means true positive, FP means false positive, and FN means false negative.

4.2. Comparison experiments

To evaluate the performance of the proposed method on the enhanced bridge defect dataset, we select several high-performance target detection models for comparison, including RT-DETR [50], Gold-YOLO [51], SSD [22], faster-RCNN [52], EfficientDET [53], YOLOv8 [54], YOLOv11 [26], Deformable-DETR [55], and YOLO-NAS [56]. Among these models, SSD and faster-RCNN are widely adopted models in defect detection tasks, while RT-DETR utilizes a transformer-based architecture. YOLOv8 and YOLOv11 are two prominent single-stage detection models in the YOLO series, and EfficientDET and Gold-YOLO are among the most recent advancements in this field. In these comparison experiments, all models use uniform image size, training number, and learning rate during training to ensure consistency. In addition, we trained all comparison models with the same data augmentation strategies and optimizers as the proposed CDC-YOLO model to make the comparison fair. The experimental results on the two datasets are shown in table 2.

(a) Results on the CODEBRIM dataset
As can be seen from table 2, CDC-YOLO stands out on mAP@50, reaching 0.893, which is significantly higher than other models. Compared with Gold-YOLO, YOLOv8, and YOLOv11, its accuracy is improved by about 13.32%, 5.93%, and 4.44%, respectively (relative value). Compared with faster-RCNN and EfficientDET, the accuracy is much higher than the two models, improving by about 33.07% and 28.5%, respectively. In terms of computational power, the parameters of CDC-YOLO are only 0.39% and 1.81% less than those of YOLOv8 and YOLOv11, respectively, but reduced by about 95% compared with faster-RCNN and increased by about 49.5% compared with EfficientDET. Compared with deformable-DETR, CDC-YOLO outperforms deformable-DETR by 11.3% in the mAP@50 metric, while the parameter count and flops are only 51.9% and 18.7% of those of deformable-DETR, respectively. Facing the lightweight model YOLO-NAS, our proposed model increases mAP@50 by 12.9% at the cost of adding 1.5% Flops, and the number of parameters is only 51.9% of that of YOLO-NAS. This difference may be due to CDC-YOLO's use of more efficient feature fusion and multi-scale feature extraction strategies to optimize the balance between precision and computational effort. In contrast, Gold-YOLO, YOLOv8, YOLOv11, and YOLO-NAS focus on lightweight design and real-time performance, sacrificing some accuracy to reduce computing requirements, while SSD, faster-RCNN, RT-DETR, and deformable-DETR improve accuracy through more complex regional suggestion networks and full connectivity layers, but with higher computing costs. EfficientDET performs relatively poorly in the defect detection of concrete bridges, mainly due to its excessive emphasis on lightweight design, which sacrifices detection accuracy in real scenarios such as complex background interference and multi-scale small targets. Based on the performance of existing embedded platforms, such as Jetson with TensorRT, the proposed model is able to be deployed on UAVs for real-world applications [57]. In general, CDC-YOLO significantly improves the accuracy by appropriately increasing the amount of calculation and parameters, and can well complete the task of concrete bridge detection based on UAVs.

(b) Results on the BDD_2K dataset
As can be seen from the data in table 2, CDC-YOLO also performs well in mAP@50 and detection performance when facing different data sets, reaching 0.536, which is the best performance among all comparison models. Compared with other models, CDC-YOLO improved by about 3.27% and 5.72% respectively compared with the baseline model and YOLOv8, and 30.10% compared with Faster-RCNN. Compared with

Gold-YOLO, EfficientDET and RT-DETR, mAP@50 indicator increased 41.80%, 76.32%, and 48.07%, respectively. Compared with Deformable-DETR and YOLO-NAS, our model also has improvements of 52.7% and 33.3% respectively. In $F1$ scores, CDC-YOLO improves by about 2%–49% compared to other models. In terms of parameters, CDC-YOLO is slightly higher than YOLOv8 and YOLOv11, which are also YOLO series, and also EfficientDET, increasing by about 0.38%, 1.81%, and 49.50%. Compared with other models, the number of parameters in our proposed model is reduced by about 70% on average, indicating that CDC-YOLO achieves efficient detection performance through the designed feature fusion and extraction module while maintaining a low number of parameters.

The BDD_2K dataset has added defect categories, and at the same time, the interfering factors in the data have further increased, which has led to a decline in the performance of all models in the comparative experiments. However, compared with other models, the model we proposed still achieved the best detection effect, indicating that CDC-YOLO has good generalization in the detection task.

5. Discussions

5.1. About the ablation study

Firstly, the effectiveness of the inter-dimensional collaboration mechanism (IDCM) in the MDFEM is studied. To evaluate its performance, we compared IDCM [43] to convolutional block attention module (CBAM) [58], coordinate attention (CA) [59], efficient channel attention (ECA) [60], and Simple Attention Module (SimAM) [61] on the CODEBRIM dataset. As can be seen from the data in table 3, IDCM has the highest value of mAP@50 (0.893) among all attention mechanisms, which is superior to other methods. The increases were 2.5% (compared to CBAM), 3.1% (compared to CA), 2.4% (compared to ECA) and 2.9% (compared to SimAM). In terms of Flops and parameter count, IDCM is not much different from other methods, and the calculation cost and parameter count remain at the same order of magnitude, indicating that it does not bring too much computation burden while improving accuracy. In terms of Flops and parameter count, IDCM is not much different from other methods, and the calculation cost and parameter count remain at the same order of magnitude, indicating that it does not bring too much computation burden while improving accuracy. This difference may be due to the fact that IDCM optimizes the feature representation of the model by integrating dynamic context information more effectively, thus improving the accuracy of target detection and location.

To further investigate the impact of the IDCM, we utilize GradCAM to visualize the defect images processed with different attention mechanisms. The results, as shown in figure 8, clearly illustrate that the proposed MDFEM module effectively filters out irrelevant background and noise, allowing the network to focus more accurately on the target regions. This enhanced focus underscores the capability of the model to prioritize relevant defect features, thereby improving its performance in defect detection tasks.

Secondly, to assess the contribution of each component in our proposed model, we conduct incremental ablation experiments on the enhanced CODEBRIM dataset. These experiments compared the performance of the overall network as different modules are progressively added. In this study, YOLOv11 serves as the baseline model, and the modules MDFEM, ADFFM, and DRFEM are sequentially integrated. The results of the ablation experiments are presented in table 4.

As can be seen from the ablation experimental data in table 4, with the gradual addition of MDFEM, ADFFM, and DRFEM, mAP@50 presents a gradually increasing trend. But it brings about an overhead of approximately 0.039 M parameters and 2.9 G Flops. By introducing ADFFM, the mAP@50 reaches 0.868, with parameters and Flops increasing by only 0.020 M and 1.2 G, respectively. DRFEM retains more high-frequency details by content-aware offset sampling points and reorganizing weights, increasing mAP@50 to 0.870, with parameters and Flops increasing by 0.065 M and 0.6 G, respectively. The combination of MDFEM and ADFFM made mAP@50 rise to 0.873, the parameters increased by 0.120 M, and Flops increased by 3.3 G. The combination of MDFEM and DRFEM made mAP@50 reach 0.876, parameters and Flops increased by 0.151 M and 3.3G respectively. The combined effect of the three modules (namely the proposed model) increased mAP@50 to 0.893, with an overall increase of only approximately 0.175 M parameters and 5.5 G Flops, verifying the significant complementarity and cumulative gain of each module in enhancing the detection capabilities of small targets, complex backgrounds, and irregular defects.

5.2. About the detection results on different defect classes

To further demonstrate the high performance of our proposed model in the task of concrete bridge defect inspection, we selected some of the most advanced models in the field of defect detection in the past two years from all the comparison models to analyze the performance differences between our proposed CDC-YOLO model and their specific defects, as shown in table 5, and superimposed with the original image to give the actual visual detection results, as shown in figure 9. The results of the PR curve and the confusion matrix based on the proposed model is shown in figure 10.

According to the results in table 5, CDC-YOLO has performed very well in the defects detection task of concrete Bridges, showing high precision in crack, spallation, efflorescence, exposed bars, corrosion, and other defects. Compared with other models, CDC-YOLO improves significantly, especially in the detection of defects such as crack, spallation, and efflorescence, which respectively increase by 13.9%, 6.78%, and 10.18% compared with other models. This indicates that CDC-YOLO has a strong performance in the processing of complex defects, indicating that the feature extraction and fusion module designed by us has fully played its role, reducing the impact of complex environmental background on

Table 3. Performance comparison of different attention mechanisms.

Attention mechanism	$F1$	Parameters (M)	Flops (G)	mAP@50
CBAM	0.88	9.895	32.6	0.871
CA	0.85	9.897	32.7	0.866
ECA	0.88	9.892	32.5	0.872
SimAM	0.86	9.892	32.5	0.868
IDCM (Ours)	0.89	9.867	32.5	0.893

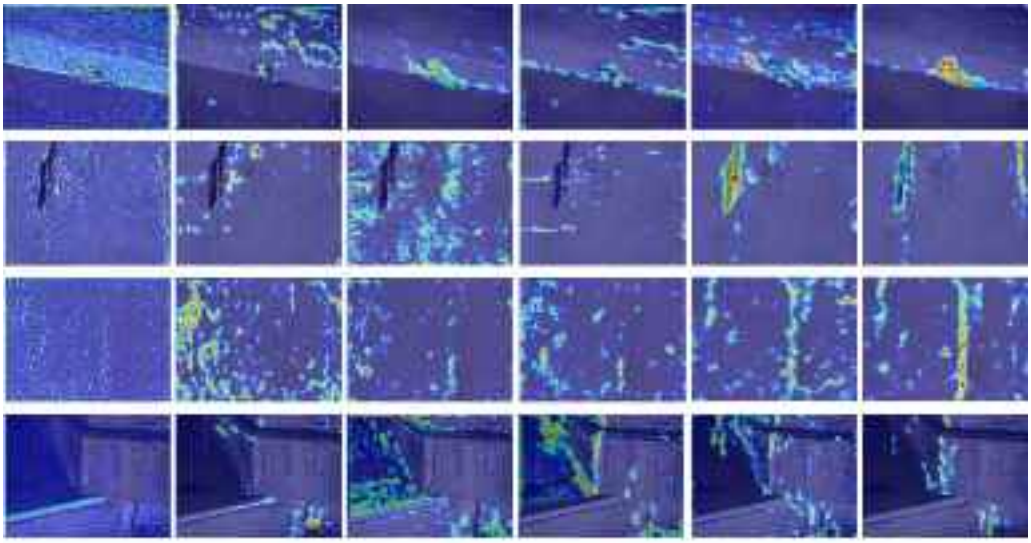


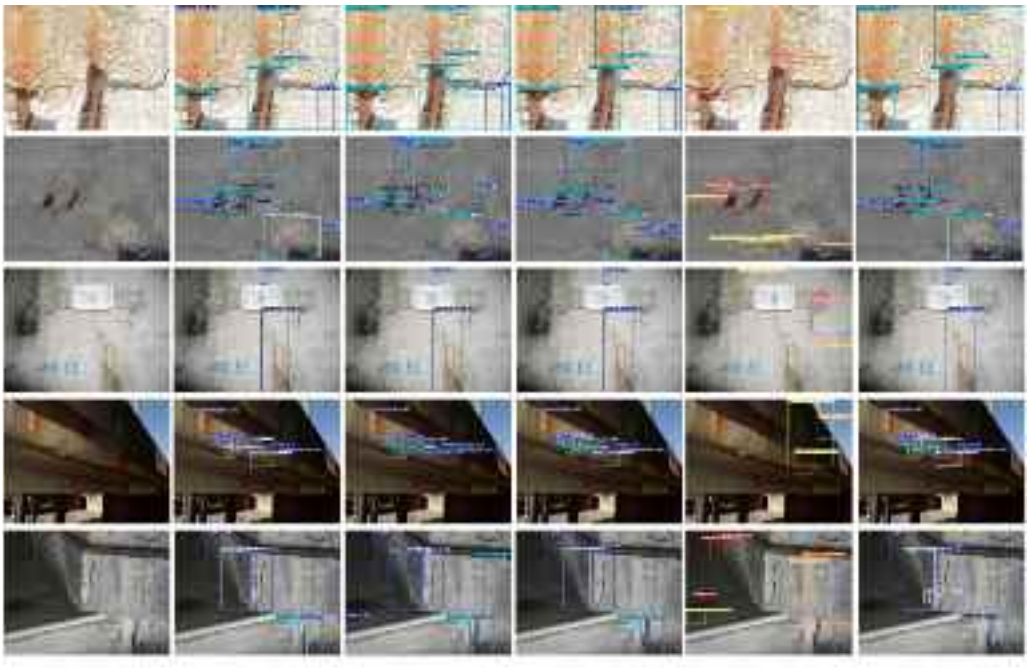
Figure 8. Visualization of heat maps with different attention mechanisms. (a) Initial images; (b) CBAM; (c) CA; (d) ECA; (e) SimAM; (f) IDCM.

Table 4. Ablation study of different modules in the proposed model.

Baseline	MDFEM	ADFFM	DRFEM	$F1$	Parameters (M)	Flops (G)	mAP@50
✓				0.87	9.692	27.0	0.855
✓	✓			0.87	9.731	29.9	0.865
✓		✓		0.86	9.712	28.2	0.868
✓			✓	0.88	9.757	27.6	0.870
✓	✓	✓		0.87	9.812	30.3	0.873
✓	✓		✓	0.88	9.843	30.3	0.876
✓		✓	✓	0.89	9.859	29.8	0.881
✓	✓	✓	✓	0.89	9.867	32.5	0.893

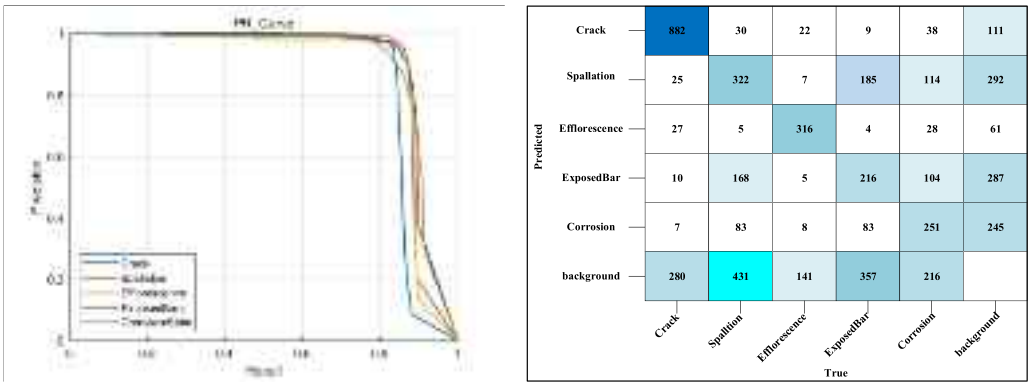
Table 5. Performance comparison of state-of-the-art models on different defect classes.

	mAP@50 of different models				
Defect class	YOLOv8	RT-DETR	YOLOv11	Gold-YOLO	CDC-YOLO
Crack	0.805	0.731	0.820	0.755	0.884
Spallation	0.861	0.815	0.880	0.824	0.901
Efflorescence	0.828	0.765	0.856	0.781	0.888
Exposed bars	0.866	0.839	0.883	0.829	0.901
Corrosion	0.854	0.800	0.870	0.780	0.890



(a) (b) (c) (d) (e) (f)

Figure 9. The visualization results based on different state-of-the-art models: (a) real image; (b) YOLOv8; (c) RT-DETR; (d) YOLOv11; (e) Gold-YOLO; (f) CDC-YOLO.



(a) (b)

Figure 10. The PR curve and the confusion matrix of the proposed CD-YOLO model: (a) PR curve; (b) confusion matrix.

the detection task, and improving the adaptability to irregular defect detection.

The actual visualization results presented in figure 9 further support this conclusion. While other models struggle with missed and false detections, particularly in low-light environments or when defects overlap, CDC-YOLO excels at accurately detecting and localizing these defects. The superior detection accuracy demonstrated by CDC-YOLO confirms its ability to effectively overcome the challenges associated with complex real-world scenarios in bridge defect detection.

As shown in the PR curve of figure 10(a), the model demonstrates relatively stable and excellent detection performance in four types of defects: spalling, weathering, exposed steel bars, and corrosion. Within the recall rate range not exceeding 80%, its precision can be maintained above 99%, indicating that the model has a good effect in feature extraction and classification discrimination. In contrast, when the recall rate of the crack category detection curve increased to over 90%, the accuracy rate dropped to a certain extent. However, in the low recall rate range, the accuracy rate remained stable at over 99%, indicating that the model can accurately identify key crack areas and also has certain robustness against small targets.

The confusion matrix in figure 10(b) shows that cracks and weathering defects exhibit a relatively high TP and accuracy rate. There is a certain degree of confusion among the other three types of defects, which is related to the characteristics of the three types of defects themselves. The three types of defects, namely spalling, exposed steel bars and corrosion, are associated when they occur. Only when concrete spalling occurs will exposed steel bars appear. Due to natural causes,

(a) (b) (c)

Figure 11. Failed cases of CDC-YOLO detection.

exposed steel bars will show a certain degree of corrosion. For the interfering factor of the environmental background, the proposed model effectively avoids the interference and prevents false detections and missed detections that occur within irrelevant environmental backgrounds.

5.3. About the detection failed case analysis

In order to better understand the challenges faced by the CDC-YOLO model and determine the subsequent in-depth research, some detection failed cases are analyzed (see figure 11). As shown in figure 11(a), the model wrongly identifies the environmental background as cracks and weathering defects. The potential cause of this false detection is the shadow and pollution in the background environment. In complex situations, such as jitter or focal length misalignment generated during the image acquisition process (as shown in figure 11(b)), the CDC-YOLO model may identify water pipes and inherent concave areas existing in concrete Bridges as bridge defects. Furthermore, as shown in figure 11(c), there will be certain false detections and missed detections in the low-light areas of the night environment. Despite some failure cases, CDC-YOLO demonstrated good performance on two datasets. To further improve the model performance, future work will focus on increasing the types of defects, the number of datasets and optimizing the feature extraction of the model.

6. Conclusions

To address the challenges of detecting overlapping and hard-to-distinguish defects in concrete bridges using UAV images, we propose a defect detection algorithm based on YOLOv11. The key improvements of the proposed model include the multi-dimensional feature extraction, the adaptive feature fusion, and the DRFEM. The proposed model has been validated using two datasets. Both theoretical analysis and experimental results yield satisfactory outcomes, demonstrating the effectiveness and reliability of the proposed model. Furthermore, the ablation study and the performance of the proposed model on different defect classes are discussed. Overall, the accuracy of CDC-YOLO in concrete bridge defect detection has significantly improved, underscoring its potential for practical applications in bridge health monitoring. In future work, we will focus on enhancing robustness in complex environments (such as optimizing adaptability to low light at night, dynamic blurring, and strong shadow interference), exploring the integration of multimodal data such as infrared and Lidar to solve the problem of false background detection, and developing lightweight deployment solutions to achieve real-time detection and autonomous trajectory adjustment at the unmanned aerial vehicle.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgment

This work was supported by the National Key R&D Program of China (2022YFB4703402), and the National Natural Science Foundation of China (61873086).

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author contributions

Jianjun Ni 0000-0002-7130-8331
Conceptualization (equal), Funding acquisition (equal)

Qibo Ji
Conceptualization (equal), Methodology (equal), Writing – original draft (equal)

Yonghao Zhao
Validation (equal), Writing – review & editing (equal)

Weidong Cao
Writing – review & editing (equal)

Pengfei Shi
Writing – review & editing (equal)

References

[1] Khan S M, Atamturktur S, Chowdhury M and Rahman M 2016 Integration of structural health monitoring and intelligent transportation systems for bridge condition assessment: current status and future direction *IEEE Trans. Intell. Transp. Syst.* **17** 2107–22

[2] Koch C, Georgieva K, Kasireddy V, Akinci B and Fieguth P 2015 A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure *Adv. Eng. Inf.* **29** 196–210

[3] Wang C, Chen B, Li Y, Wang H, Tan L, Zhang Y and Zhang H 2025 YOLOv8-CDD: an improved concrete defect detection method combined CNN with transformer *Meas. Sci. Technol.* **36** 015409

[4] Amirkhani D, Allili M S, Hebbache L, Hammouche N and Lapointe J-F 2024 Visual concrete bridge defect classification and detection using deep learning: a systematic review *IEEE Trans. Intell. Transp. Syst.* **25** 10483–505

[5] Ai D, Jiang G, Lam S-K, He P and Li C 2023 Computer vision framework for crack detection of civil infrastructure-a review *Eng. Appl. Artif. Intell.* **117** 105478

[6] Nishimura Y, Takahashi S, Mochiyama H and Yamaguchi T 2022 Automated hammering inspection system with multi-copter type mobile robot for concrete structures *IEEE Robot. Autom. Lett.* **7** 9993–10000

[7] Spencer B F, Hoskere V and Narazaki Y 2019 Advances in computer vision-based civil infrastructure inspection and monitoring *Engineering* **5** 199–222

[8] Dong C-Z and Catbas F N 2021 A review of computer vision-based structural health monitoring at local and global levels *Struct. Health Monit.* **20** 692–743

[9] Jiang S, Zhang Y, Wang F and Xu Y 2025 Three-dimensional reconstruction and damage localization of bridge undersides based on close-range photography using UAV *Meas. Sci. Technol.* **36** 015423

[10] Feroz S and Dabous S A 2021 UAV-based remote sensing applications for bridge condition assessment *Remote Sens.* **13** 1809

[11] Liang H, Lee S-C, Bae W, Kim J and Seo S 2023 Towards UAVs in construction: advancements, challenges and future directions for monitoring and inspection *Drones* **7** 202

[12] Qiao W, Ma B, Liu Q, Wu X and Li G 2021 Computer vision-based bridge damage detection using deep convolutional networks with expectation maximum attention module *Sensors* **21** 1–18

[13] Zhang L, Yang F, Daniel Zhang Y and Zhu Y J 2016 Road crack detection using deep convolutional neural network *Proc. - Int. Conf. on Image Processing, ICIP* pp 3708–12

[14] Xin G, Fan X, Shi P, Luo C, Ni J and Cao Y 2023 A fine extraction algorithm for image-based surface cracks in underwater dams *Meas. Sci. Technol.* **34** 035402

[15] Bhattacharya G, Mandal B and Puhan N B 2021 Interleaved deep artifacts-aware attention mechanism for concrete structural defect classification *IEEE Trans. Image Process.* **30** 6957–69

[16] Tang G, Ni J, Zhao Y, Gu Y and Cao W 2024 A survey of object detection for UAVs based on deep learning *Remote Sens.* **16** 149

[17] Jiang Y, Pang D and Li C 2021 A deep learning approach for fast detection and classification of concrete damage *Autom. Constr.* **128** 103785

[18] Yapi D, Allili M S and Baaziz N 2018 Automatic fabric defect detection using learning-based local textural distributions in the contourlet domain *IEEE Trans. Autom. Sci. Eng.* **15** 1014–26

[19] Chen C, Seo H, Jun C and Zhao Y 2022 A potential crack region method to detect crack using image processing of multiple thresholding *Signal Image Video Process.* **16** 1673–81

[20] Shi Y, Cui L, Qi Z, Meng F and Chen Z 2016 Automatic road crack detection using random structured forests *IEEE Trans. Intell. Transp. Syst.* **17** 3434–45

[21] Zheng X, Zheng S, Kong Y and Chen J 2021 Recent advances in surface defect inspection of industrial products using deep learning techniques *Int. J. Adv. Manuf. Technol.* **113** 35–58

[22] Ni J, Shen K, Chen Y and Yang S X 2023 An improved SSD-like deep network-based object detection method for indoor scenes *IEEE Trans. Instrum. Meas.* **72** 5006915

[23] Liu Y, Zhang C and Dong X 2023 A survey of real-time surface defect inspection methods based on deep learning *Artif. Intell. Rev.* **56** 12131–70

[24] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L and Polosukhin I 2017 Attention is all you need *Advances in Neural Information Processing Systems* vol 2017 pp 5999–6009

[25] Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A and Zagoruyko S 2020 End-to-end object detection with transformers *Computer Vision - ECCV 2020* (*Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*) vol 12346 pp 213–29

[26] Li Y, Yan H, Li D and Wang H 2024 Robust miner detection in challenging underground environments: an improved YOLOv11 approach *Appl. Sci.* **14** 11700

[27] Kim I-H, Jeon H, Baek S-C, Hong W-H and Jung H-J 2018 Application of crack identification techniques for an aging concrete bridge inspection using an unmanned aerial vehicle *Sensors* **18** 1881

[28] Haciefendioglu K and Basaga H B 2022 Concrete road crack detection using deep learning-based faster R-CNN method *Iran. J. Sci. Technol.* **46** 1621–33

[29] Wei F, Yao G, Yang Y and Sun Y 2019 Instance-level recognition and quantification for concrete surface bughole based on deep learning *Autom. Constr.* **107** 102920

[30] Xu Y, Wei S, Bao Y and Li H 2019 Automatic seismic damage identification of reinforced concrete columns from images by a region-based deep convolutional neural network *Struct. Control Health Monit.* **26** e2313

[31] Yao G, Wei F, Yang Y and Sun Y 2019 Detection of bughole on concrete surface with convolutional neural network *Proc. - 2019 4th Int. Conf. on Control, Robotics and Cybernetics, CRC 2019* pp 184–8

[32] Mishra M, Jain V, Singh S K and Maity D 2022 Two-stage method based on the you only look once framework and image segmentation for crack detection in concrete structures *Arch. Struct. Constr.* **3** 429–46

[33] Pang R, Yang Y, Huang A, Liu Y, Zhang P and Tang G 2024 Multi-scale feature fusion model for bridge appearance defect detection *Big Data Mining Anal.* **7** 1–11

[34] Wan H, Gao L, Yuan Z, Qu H, Sun Q, Cheng H and Wang R 2023 A novel transformer model for surface damage detection and cognition of concrete bridges *Expert Syst. Appl.* **213** 119019

[35] Teng S, Liu Z, Chen G and Cheng L 2021 Concrete crack detection based on well-known feature extractor model and the YOLOv2 network *Appl. Sci.* **11** 813

[36] Cui X, Wang Q, Dai J, Zhang R and Li S 2021 Intelligent recognition of erosion damage to concrete based on improved YOLO-v3 *Mater. Lett.* **302** 130363

[37] Wu P, Liu A, Fu J, Ye X and Zhao Y 2022 Autonomous surface crack identification of concrete structures based on an improved one-stage object detection algorithm *Eng. Struct.* **272** 114962
[38] Wang W, Su C and Fu D 2022 Automatic detection of defects in concrete structures based on deep learning *Structures* **43** 192–9
[39] Kumar P, Batchu S, Swamy S N and Kota S R 2021 Real-time concrete damage detection using deep learning for high rise structures *IEEE Access* **9** 112312–31
[40] Zou Z, Chen K, Shi Z, Guo Y and Ye J 2023 Object detection in 20 years: a survey *Proc. IEEE* **111** 257–76
[41] Zhang C, Zou Y, Wang F, del Rey Castillo E, Dimyadi J and Chen L 2022 Towards fully automated unmanned aerial vehicle-enabled bridge inspection: where are we at? *Constr. Build. Mater.* **347** 128543
[42] Zhang X, Zhou X, Lin M and Sun J 2018 Shufflenet: an extremely efficient convolutional neural network for mobile devices *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* vol 112 pp 6848–56
[43] Yu Y, Zhang Y, Cheng Z, Song Z and Tang C 2023 MCA: multidimensional collaborative attention in deep convolutional neural networks for image recognition *Eng. Appl. Artif. Intell.* **126** 107079
[44] Wang F, Zou Y, Chen X, Zhang C, Hou L, del Rey Castillo E and Lim J B 2024 Rapid in-flight image quality check for UAV-enabled bridge inspection *ISPRS J. Photogramm. Remote Sens.* **212** 230–50
[45] Tang L, Zhang H, Xu H and Ma J 2023 Rethinking the necessity of image fusion in high-level vision tasks: a practical infrared and visible image fusion network based on progressive semantic injection and scene fidelity *Inf. Fusion* **99** 101870
[46] Wang J, Chen K, Xu R, Liu Z, Loy C C and Lin D 2019 Carafe: content-aware reassembly of features *2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV)* pp 3007–16
[47] Liu W, Lu H, Fu H and Cao Z 2023 Learning to upsample by learning to sample *2023 IEEE/CVF Int. Conf. on Computer Vision (ICCV)* pp 6004–14
[48] Mundt M, Majumder S, Murali S, Panetsos P and Ramesh V 2019 Meta-learning convolutional neural architectures for multi-target concrete defect classification with the concrete defect bridge image dataset *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 11188–97
[49] Flotzinger J, Rosch P J, Benz C, Ahmad M, Cankaya M, Mayer H, Rodehorst V, Oswald N and Braml T 2024 Dacl-challenge: semantic segmentation during visual bridge inspections *Proc. - 2024 IEEE Winter Conf. on Applications of Computer Vision Workshops, WACVW 2024* pp 716–25
[50] Zhao Y, Lv W, Xu S, Wei J, Wang G, Dang Q, Liu Y and Chen J 2023 DETRs beat YOLOs on real-time object detection *2024 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 16965–74
[51] Wang C, He W, Nie Y, Guo J, Liu C, Han K and Wang Y 2023 Gold-YOLO: efficient object detector via gather-and-distribute mechanism *Adv. Neural Inf. Process. Syst.* **36** 51094–112
[52] Ren S, He K, Girshick R and Sun J 2017 Faster R-CNN: towards real-time object detection with region proposal networks *IEEE Trans. Pattern Anal. Mach. Intell.* **39** 1137–49
[53] Tan M, Pang R and Le Q V 2019 EfficientDet: scalable and efficient object detection *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 10778–87
[54] Ni J, Zhu S, Tang G, Ke C and Wang T 2024 A small-object detection model based on improved YOLOv8s for UAV image scenarios *Remote Sens.* **16** 2465
[55] Zhu X, Su W, Lu L, Li B, Wang X and Dai J 2021 Deformable DETR: deformable transformers for end-to-end object detection *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (*Virtual, Online*) pp 10247–56
[56] Pandey N N, Pati A and Maurya R 2024 DriSm_YNet: a breakthrough in real-time recognition of driver smoking behavior using YOLO-NAS *Neural Comput. Appl.* **36** 18413–32
[57] Ma M-Y, Shen S-E and Huang Y-C 2023 Enhancing UAV visual landing recognition with YOLO's object detection by onboard edge computing *Sensors* **23** 8999
[58] Woo S, Park J, Lee J Y and Kweon I S 2018 CBAM: convolutional block attention module *Proc. European Conference on Computer Vision (ECCV)* (*Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*) vol 11211 pp 3–19
[59] Hou Q, Zhou D and Feng J 2021 Coordinate attention for efficient mobile network design *2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Virtual, Online*) pp 13708–17
[60] Wang Q, Wu B, Zhu P, Li P, Zuo W and Hu Q 2020 ECA-Net: efficient channel attention for deep convolutional neural networks *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (*Virtual, Online*) pp 11531–9
[61] Yang L, Zhang R Y, Li L and Xie X 2021 SimAM: a simple, parameter-free attention module for convolutional neural networks *Proc. Machine Learning Research* (*Virtual, Online*) vol 139 pp 11863–74