Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.066188

ARTICLE

MGD-YOLO: An Enhanced Road Defect Detection Algorithm Based on Multi-Scale Attention Feature Fusion

Zhengji Li[1], Fazhan Xiong[1], Boyun Huang[1], Meihui Li[1], Xi Xiao[2], Yingrui Ji[3,4], Jiacheng Xie[1,2], Aokun Liang[5] and Hao Xu[6,*]

[1]School of Computer and Software, Chengdu Jincheng College, Chengdu, 611731, China
[2]College of Arts and Sciences, University of Alabama at Birmingham, Birmingham, AL 35294, USA
[3]Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, 100193, China
[4]School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing, 100193, China
[5]School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, 430079, China
[6]Department of Medicine, Harvard Medical School, Boston, MA 02115, USA

*Corresponding Author: Hao Xu. Email: haxu@bwh.harvard.edu

Received: 01 April 2025; Accepted: 05 June 2025; Published: 30 July 2025

ABSTRACT: Accurate and real-time road defect detection is essential for ensuring traffic safety and infrastructure maintenance. However, existing vision-based methods often struggle with small, sparse, and low-resolution defects under complex road conditions. To address these limitations, we propose Multi-Scale Guided Detection YOLO (MGD-YOLO), a novel lightweight and high-performance object detector built upon You Only Look Once Version 5 (YOLOv5). The proposed model integrates three key components: (1) a Multi-Scale Dilated Attention (MSDA) module to enhance semantic feature extraction across varying receptive fields; (2) Depthwise Separable Convolution (DSC) to reduce computational cost and improve model generalization; and (3) a Visual Global Attention Upsampling (VGAU) module that leverages high-level contextual information to refine low-level features for precise localization. Extensive experiments on three public road defect benchmarks demonstrate that MGD-YOLO outperforms state-of-the-art models in both detection accuracy and efficiency. Notably, our model achieves 87.9% accuracy in crack detection, 88.3% overall precision on TD-RD dataset, while maintaining fast inference speed and a compact architecture. These results highlight the potential of MGD-YOLO for deployment in real-time, resource-constrained scenarios, paving the way for practical and scalable intelligent road maintenance systems.

KEYWORDS: YOLO; road damage detection; object detection; computer vision; deep learning

1 Introduction

Timely and accurate detection of road surface defects is essential for ensuring transportation safety and enabling proactive infrastructure maintenance. Traditional inspection methods, which rely heavily on manual labor, are often inefficient, costly, and susceptible to human error. With the rapid advancement of deep learning and computer vision technologies, automated visual defect detection has become a promising alternative, particularly for enabling real-time and high-precision assessment of road conditions.

Among existing approaches, one-stage detectors such as the YOLO series have gained popularity due to their efficiency and suitability for real-time applications. However, road defect detection in unconstrained environments presents persistent challenges. As shown in Fig. 1, small irregularly shaped defects often appear under complex lighting or background conditions and are difficult to localize due to their limited semantic features and low contrast. Moreover, conventional convolutional neural networks (CNNs) exhibit limited receptive fields and struggle to model global contextual dependencies, resulting in missed or inaccurate detections for fine-grained or small-scale targets.

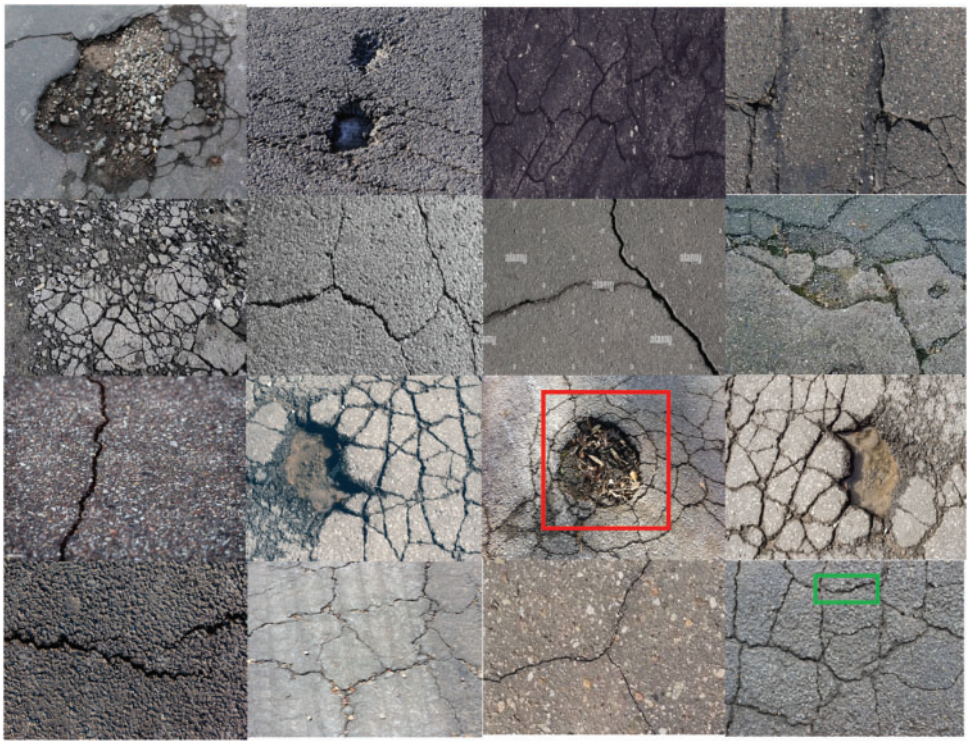


Figure 1: Images of some road defects in our dataset

To address these issues, we propose Multi-Scale Guided Detection YOLO (MGD-YOLO), an enhanced YOLOv5-based architecture specifically designed for robust road defect detection. Unlike previous YOLO-based enhancements that primarily focus on speed or general object detection, MGD-YOLO explicitly targets the accurate identification of small-scale, low-contrast defects under complex real-world conditions. It introduces a set of architectural improvements to strengthen the model's ability to capture multi-scale and context-aware features while maintaining a lightweight structure suitable for real-time applications. Specifically, we integrate a Multi-Scale Dilated Attention (MSDA) module into the backbone to capture multi-level contextual dependencies across different receptive fields. We further adopt Depthwise Separable Convolution (DSC) to reduce parameter overhead and computational cost without compromising feature expressiveness. Finally, we design a Visual Global Attention Upsampling (VGAU) module to fuse low-level and high-level features using global semantic guidance, thereby improving the localization and classification of small or low-contrast defects.

Extensive experiments on three public road defect datasets demonstrate that MGD-YOLO achieves superior performance in terms of detection accuracy, robustness, and inference speed compared to existing

state-of-the-art methods. Notably, our model achieves a crack detection accuracy of 97.7%, an mAP_{50} of 85.7%, and an inference speed of 105 FPS, validating its effectiveness in both accuracy and efficiency for real-time deployment.

Our contributions are summarized as follows:

- We propose MGD-YOLO, an enhanced YOLOv5-based detector tailored for road defect detection, with particular emphasis on handling small-scale, low-contrast defects in complex environments.
- We introduce a Multi-Scale Dilated Attention (MSDA) module and a Visual Global Attention Upsampling (VGAU) module to improve multi-scale feature representation and enhance semantic consistency across different resolution levels.
- We demonstrate through extensive experiments that MGD-YOLO significantly outperforms existing detectors in both detection accuracy and inference speed, making it highly suitable for deployment in real-time, resource-constrained scenarios.

2 Related Work

2.1 Deep Learning for Road Defect Detection

Automated road defect detection has attracted growing attention due to its importance for intelligent transportation and infrastructure maintenance. Traditional techniques, such as edge detection [1], wavelet-based analysis [2], and texture descriptors [3], are highly sensitive to noise, lighting variation, and road texture diversity. In contrast, deep learning-based methods [4,5] offer superior robustness and accuracy. Zhang et al. [6] developed a CNN-based pipeline for crack detection that significantly outperformed handcrafted approaches. Shi et al. [7] proposed a structure forest model to handle the complexity and topological variation of cracks. More recently, Park et al. [8] introduced an adaptive pixel neighborhood segmentation method, which improved detection under noisy backgrounds.

YOLO-based detectors have emerged as a strong baseline for real-time road defect detection. For instance, Jocher et al. [9] released YOLOv5, which offers a good balance between speed and performance. Several studies [10–14] have adapted YOLO variants to road scenarios, incorporating tailored preprocessing or architectural changes to handle challenges such as occlusion, low resolution, and class imbalance. However, small-scale and unevenly distributed defects (e.g., micro-cracks or edge disintegration) remain under-detected due to limited feature expressiveness in conventional backbones.

2.2 Attention Mechanisms in Visual Recognition

Attention mechanisms have become integral in enhancing CNNs' and transformers' ability to model long-range dependencies and focus on task-relevant features. Channel attention modules, such as SE-Net [15], ECA-Net [16], and the Coordinated Attention (CA) module [17], improve feature channel calibration, enhancing performance across classification and detection tasks. Spatial attention, as used in CBAM [18] and SAM [19], enables focus on key spatial regions, which is particularly useful in defect localization. The combination of spatial and channel attention has also been extended into multi-branch fusion networks and deformable attention [20]. As shown in Table 1, MGD-YOLO achieves the highest mAP with a lightweight architecture compared to recent methods.

Table 1: Comparison with mainstream detectors used in experiments

Model	Backbone	Enhancement modules	Benchmark	mAP (%)
YOLOv5s	CSPDarkNet	PANet, SPPF	TD-RD	84.1
YOLOv6n [21]	EfficientRep	RepOptimizer, Strong augmentations	TD-RD	85.0

(Continued)

Table 1 (continued)

Model	Backbone	Enhancement modules	Benchmark	mAP (%)
YOLOv7-tiny [22]	E-ELAN	Model scaling, Coarse-to-fine head	TD-RD	85.6
YOLOv8n [23]	C2f, CSPDarkNet	Decoupled head, Strong augmentations	TD-RD	86.0
YOLOS-ti [24]	ViT-Tiny	Visual tokenization, Lightweight attention	TD-RD	85.3
RT-DETR-R18 [25]	ResNet-18 + DETR Head	Query selection, Two-stage decoder	TD-RD	86.1
Lite-DETR [26]	MobileViT	Lightweight cross attention, Fast convergence	TD-RD	86.5
Faster R-CNN [27]	ResNet-50	Region proposal network (RPN)	TD-RD	82.7
SSD30 [28]	VGG-16	Multi-scale feature maps	TD-RD	80.3
MGD-YOLO (Ours)	YOLOv5 Backbone	MSDA, DSC, VGAU	TD-RD	**88.3**

In the context of road defect detection, attention modules help reduce background interference and emphasize texture-disruptive regions. For instance, Liu et al. [29] proposed a graph-based attention fusion method to integrate multiple defect cues. Wang et al. [30] used dual attention paths to handle noise and occlusion. Inspired by these advances, our work incorporates a Multi-Scale Dilated Attention (MSDA) module, which enables the model to simultaneously focus on semantic patterns at various receptive field sizes, effectively enhancing sensitivity to subtle structural anomalies.

2.3 Lightweight Design and Multi-Scale Feature Fusion

Deploying detection models in real-world infrastructure applications often requires low-latency and lightweight architectures. Depthwise Separable Convolutions, first introduced in MobileNet [31,32], have become a standard tool for reducing parameter count and computation, followed by enhancements like inverted residual blocks in MobileNetV2 [33] and re-parameterization in RepVGG [34]. For real-time edge deployment, recent frameworks like YOLOv7-Tiny [22,35,36] and YOLO-NAS [37] attempt to balance accuracy and efficiency through backbone redesign and NAS-based optimization.

On the feature fusion side, models such as FPN [38], PANet [39], and BiFPN [40] improve multi-scale prediction by enhancing the flow of semantic information across network layers. However, simple concatenation or summation can introduce redundancy and reduce spatial precision. To address this, attention-guided fusion modules [41–44] and context-aware decoders [45–47] have been proposed. Our method builds upon this line of work by designing a Visual Global Attention Upsampling (VGAU) module that leverages high-level semantics as global context guidance to refine low-level feature maps during upsampling, thereby enhancing the localization of small or low-contrast defects.

3 Methodology

Although YOLOv5 has demonstrated impressive performance in real-time object detection, it tends to rely heavily on low-level feature maps for prediction, which often leads to the loss of critical high-level semantic information. This limitation becomes particularly pronounced in multi-scale detection scenarios, where the lack of global contextual understanding undermines the accurate localization of small or subtle defects. Furthermore, the commonly used CBL block (Convolution, Batch Normalization, Leaky ReLU) in YOLOv5

employs standard convolutions, resulting in a large number of parameters and high computational overhead. These characteristics significantly restrict the model's deployment on resource-constrained devices, such as mobile or embedded systems.

To overcome these limitations, we propose MGD-YOLO, a structurally enhanced variant of YOLOv5 tailored for robust and efficient road defect detection. Our design introduces three key architectural innovations to improve the model's expressiveness, accuracy, and computational efficiency.

As illustrated in Fig. 2, MGD-YOLO retains the core structure of YOLOv5 but incorporates the following enhancements: First, we embed a Multi-Scale Dilated Attention (MSDA) module into the backbone, enabling the network to model semantic dependencies across varying receptive fields. This enhances the feature representation capacity for defects of different sizes and textures, especially in complex scenes. Second, to reduce the computational burden, we replace standard convolutions in CBL blocks with Depthwise Separable Convolution (DSC) [31], which decomposes the convolution operation into spatial and channel-wise components. This substitution significantly lowers the number of parameters and floating-point operations (FLOPs), while maintaining the model's expressive power. Third, we introduce a novel Visual Global Attention Upsampling (VGAU) module, which refines the low-level feature maps using global semantic cues derived from high-level features. This facilitates more precise spatial localization of defects, particularly small-scale or low-contrast anomalies that may otherwise be overlooked. These enhancements collectively improve the accuracy, robustness, and deployment efficiency of MGD-YOLO. The model is particularly well-suited for real-time road inspection applications where detection precision and computational cost must be carefully balanced. In the following subsections, we provide detailed descriptions of each module integrated into the MGD-YOLO framework.

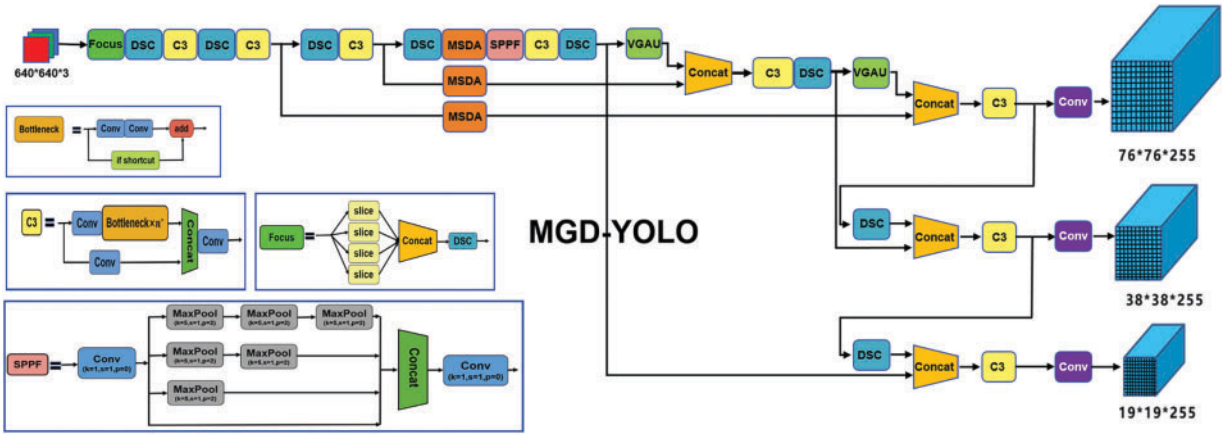


Figure 2: Overall architecture of the proposed MGD-YOLO model. The backbone is enhanced with Multi-Scale Dilated Attention (MSDA), Depthwise Separable Convolutions (DSC), and Visual Global Attention Upsampling (VGAU) to improve feature fusion and detection performance

3.1 Attention-Based Multi-Scale Feature Extraction via MSDA

In object detection tasks, high-level feature maps typically encapsulate rich semantic information but suffer from limited spatial resolution, making it challenging to accurately localize fine-grained targets. Conversely, low-level feature maps preserve high-resolution spatial details yet lack semantic abstraction. Bridging this semantic-resolution gap is critical for improving detection performance across varied object scales, particularly in complex scenarios such as road defect detection. Previous studies have attempted to

address this challenge through hierarchical feature fusion [38,39], yet simple aggregation often introduces redundant information and fails to resolve semantic inconsistencies between layers.

To overcome these limitations, we incorporate an attention-based strategy into the feature fusion process of YOLOv5 by introducing the Multi-Scale Dilated Attention (MSDA) module. Attention mechanisms have shown significant effectiveness across various domains, including object detection [18], semantic segmentation [48], and natural language processing [49,50], due to their ability to dynamically emphasize task-relevant features while suppressing irrelevant noise. Classic modules such as Squeeze-and-Excitation (SE) [15], CBAM [18], and ECA [16] have demonstrated the benefit of channel-wise and spatial recalibration. However, these approaches often lack flexibility in modeling variable context scales. In contrast, MSDA introduces dilated self-attention across multiple receptive fields, enabling the network to capture both local details and global semantic dependencies in a unified framework.

As shown in Fig. 3, given an input feature map $F \in \mathbb{R}^{H\times W\times C}$, the MSDA module first splits it into multiple attention heads along the channel dimension. Each head applies a window-based self-attention mechanism with a specific dilation rate $r \in \{1, 2, 3\}$ to expand its receptive field. This enables each head to capture context at different spatial scales. For a query position x, the output of head i is computed as:

$$\mathbf{A}_i(x) = \sum_{y\in\mathcal{N}_{r_i}(x)} \text{Softmax}\left(\phi_q(x)^\top \cdot \phi_k(y)\right) \cdot \phi_v(y) \tag{1}$$

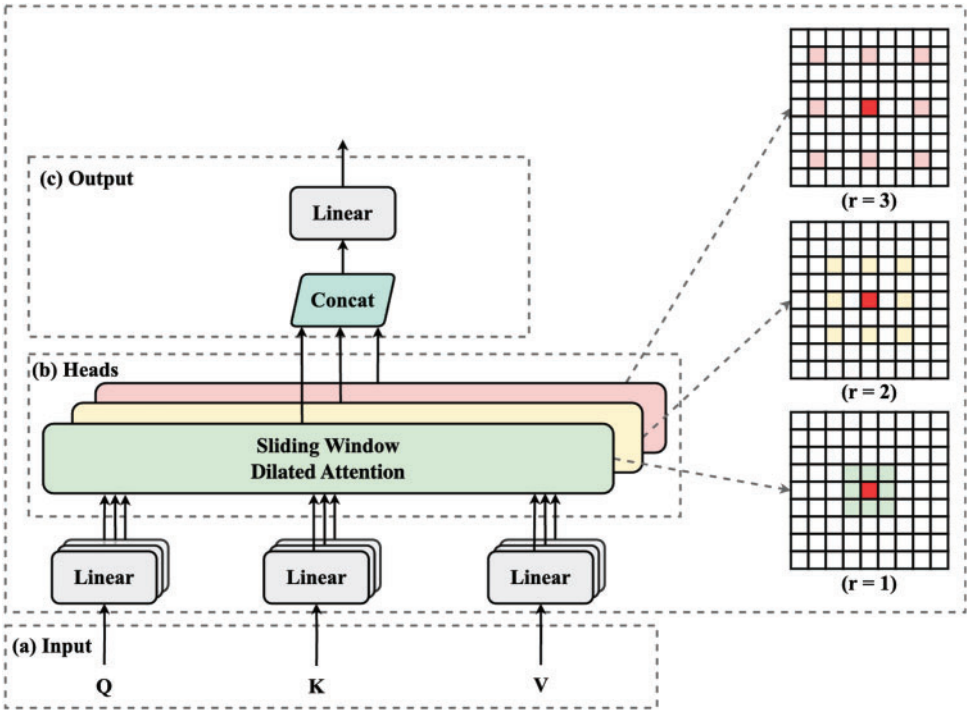


Figure 3: Overview of the proposed Multi-Scale Dilated Attention (MSDA) module. Each head attends to features at a distinct dilation rate, aggregating multi-scale contextual information

where $\mathcal{N}_{r_i}(x)$ denotes the neighborhood region centered at x under dilation r_i, and ϕ_q, ϕ_k, ϕ_v are learnable linear projections for queries, keys, and values, respectively. The outputs of all attention heads are then concatenated and passed through a lightweight multi-layer perceptron (MLP) to produce the refined feature map:

$$F' = \text{MLP}\left(\text{Concat}(\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3)\right) + F \tag{2}$$

This design enables MSDA to aggregate multi-scale semantic cues while maintaining efficiency through dilated sampling, thus reducing information redundancy without introducing additional heavy computation.

The combination of dilated sampling and dynamic attention aggregation not only enhances the representational capacity at multiple scales but also reduces information redundancy compared to naive multi-branch fusion, leading to improved feature discriminability with lower computational cost.

We integrate MSDA immediately after the C3 module within the YOLOv5 backbone, as depicted in Fig. 4. This placement ensures that enriched multi-scale semantic features are incorporated before the upsampling stage. By doing so, low-level features retain detailed structural information, while high-level features provide contextual guidance, forming a more consistent and discriminative representation for defect detection.

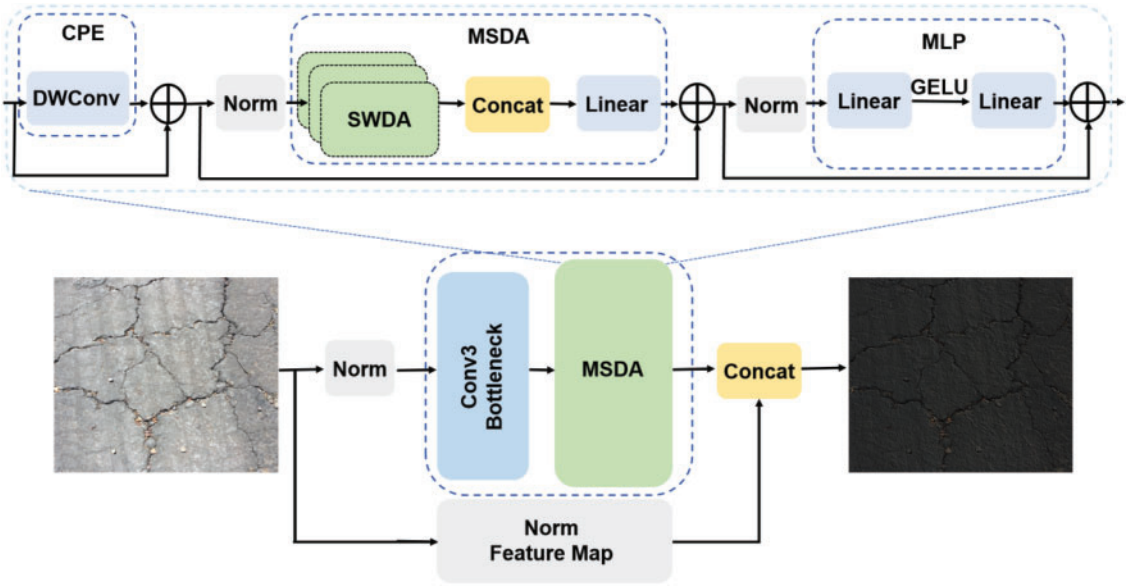


Figure 4: MSDA implementation details. The top shows the module structure consisting of DWConv, sliding windows, and MLP layers; the bottom illustrates the integration of MSDA into the feature fusion pathway

Theoretically, the multi-scale dilated attention promotes a more stable feature learning process by ensuring that both localized anomalies (e.g., small cracks) and broader contextual cues (e.g., surface material transitions) are simultaneously emphasized during forward and backward propagation. This stabilization effect improves model convergence behavior and robustness during training.

Empirically, this configuration allows the model to better focus on subtle texture changes and irregular defect boundaries that are critical in road inspection tasks. Consequently, the MSDA-enhanced feature

maps significantly contribute to improving the model's detection accuracy, particularly in identifying small, scattered, or visually ambiguous road defects.

3.2 Visual Global Attention Upsampling (VGAU)

While convolutional neural networks (CNNs) have achieved remarkable success in object detection due to their hierarchical feature representation and end-to-end trainability [51,52], they often suffer from loss of fine-grained spatial details during deep feature extraction. High-level features, although semantically rich, are typically downsampled and lose precise localization cues, which is particularly detrimental for detecting small or low-contrast objects like road cracks or surface repairs. Conversely, low-level features retain spatial resolution but lack semantic context, making it challenging to distinguish defects from background textures.

To address this issue, various U-shaped architectures [53,54] have explored the integration of decoder paths to restore fine details. However, these designs often involve complex multi-stage decoders and impose high computational overhead, limiting their suitability for real-time applications on resource-constrained platforms.

To enable efficient and context-aware upsampling, we draw inspiration from the Global Attention Upsampling (GAU) module [55,56] and propose an improved variant tailored for road defect detection, termed Visual Global Attention Upsampling (VGAU). Our VGAU module leverages high-level global semantic context to recalibrate low-level feature responses, enhancing localization precision without introducing significant computational complexity. Unlike traditional decoder structures that independently process low-level features, VGAU introduces top-down semantic guidance during upsampling, which improves the discrimination ability of low-level feature activations while stabilizing feature propagation across the network. This facilitates a smoother gradient flow and more robust convergence during training.

As illustrated in Fig. 5, given a high-level feature map $F_h \in \mathbb{R}^{H\times W\times C}$ and a corresponding low-level feature map $F_l \in \mathbb{R}^{2H\times 2W\times C'}$, we first compute a global semantic vector via global average pooling:

$$\mathbf{g} = \frac{1}{H \cdot W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_h(i, j) \tag{3}$$

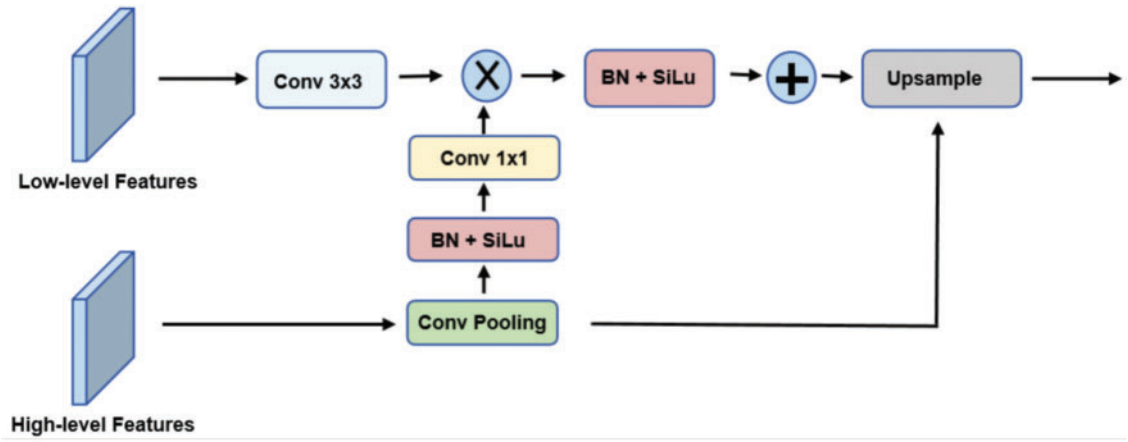


Figure 5: Architecture of the proposed visual global attention upsampling (VGAU) module. High-level global context modulates low-level features through channel-wise attention, followed by upsampling and fusion

This global descriptor $\mathbf{g} \in \mathbb{R}^C$ is then transformed through a lightweight channel-wise excitation function composed of a 1×1 convolution, batch normalization, and a non-linear SiLU activation:

$$\mathbf{w} = \sigma(\mathrm{BN}(W_1 \cdot \mathbf{g})) \tag{4}$$

Meanwhile, the low-level feature map F_l is compressed using a 3×3 convolution to reduce channel dimensionality. The attention-guided modulation is then applied by reweighting F_l with $\mathbf{w}$:

$$\tilde{F}_l = \mathbf{w} \odot \mathrm{Conv}_{3\times3}(F_l) \tag{5}$$

The recalibrated low-level features $\tilde{F}_l$ are then fused with the upsampled high-level features via summation:

$$F_{\mathrm{out}} = \mathrm{Upsample}(F_h) + \tilde{F}_l \tag{6}$$

Theoretically, VGAU enhances training stability by aligning low-level feature distributions with high-level semantic priors, which reduces feature noise and suppresses gradient vanishing phenomena in the decoder pathway. The use of channel-wise attention ensures that the network dynamically emphasizes important semantic clues while filtering irrelevant background patterns, thereby accelerating convergence and improving generalization performance.

As shown in Fig. 6, VGAU is embedded in the upsampling pathway of the MGD-YOLO architecture, where it operates in conjunction with the MSDA and DSC modules. This integration enables the network to preserve both the fine-grained spatial cues and semantic richness required for robust road defect detection.

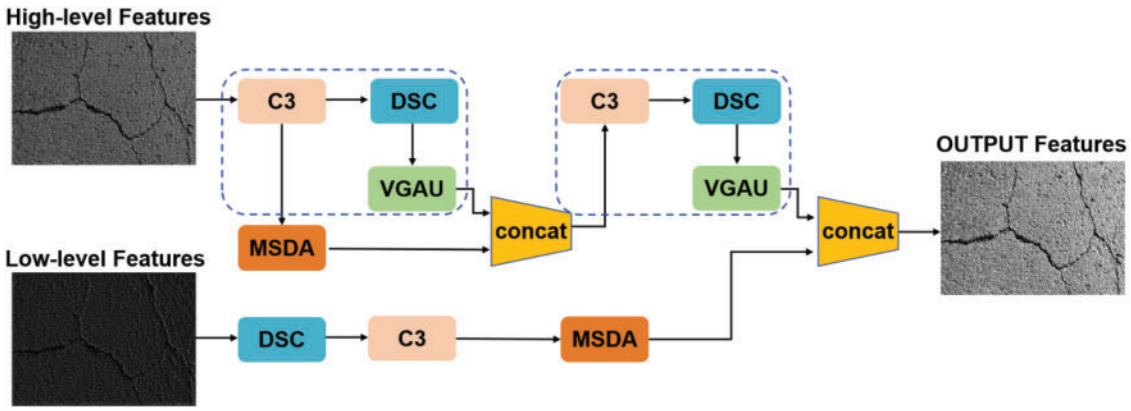


Figure 6: Utilization of VGAU within MGD-YOLO: MSDA-enhanced features are progressively refined through C3, DSC, and VGAU blocks for accurate multi-scale prediction

Compared to the original GAU, our VGAU incorporates two major improvements: (1) the use of the SiLU activation for smoother gradient propagation and non-linearity, and (2) an additional upsampling operation post-fusion to enhance resolution alignment. Overall, VGAU not only improves multi-scale feature fusion efficiency but also enhances model training dynamics by promoting consistent semantic flow across layers, ensuring higher stability and robustness in real-world deployment. These enhancements allow the module to operate efficiently across multi-scale representations, ultimately contributing to improved precision and recall in our detection results.

3.3 Depthwise Separable Convolution

Traditional convolutional operations, as adopted in the original YOLOv5 architecture, compute correlations between input features and convolution kernels across both spatial and channel dimensions simultaneously. While effective in capturing local patterns, this approach introduces significant computational overhead, especially when dealing with high-dimensional feature maps. The number of parameters and the computational complexity of a standard convolutional layer are given by:

$$\text{Parameters}_{\text{standard}} = D_{\text{out}} \times K \times K \times D_{\text{in}} \tag{7}$$

$$\text{FLOPs}_{\text{standard}} = H' \times W' \times D_{\text{out}} \times K \times K \times D_{\text{in}} \tag{8}$$

where D_{in} and D_{out} denote the number of input and output channels, K is the kernel size, and H', W' are the spatial dimensions of the output feature map.

In the context of real-time road defect detection, such computational demands pose serious limitations, especially for deployment on edge devices with constrained resources. To alleviate this issue and accelerate inference, we replace all standard convolution layers in the network with Depthwise Separable Convolution (DSC) modules, a lightweight alternative initially proposed in MobileNet [31]. As shown in Fig. 7, DSC factorizes the convolution operation into two independent steps: *depthwise convolution* and *pointwise convolution*.

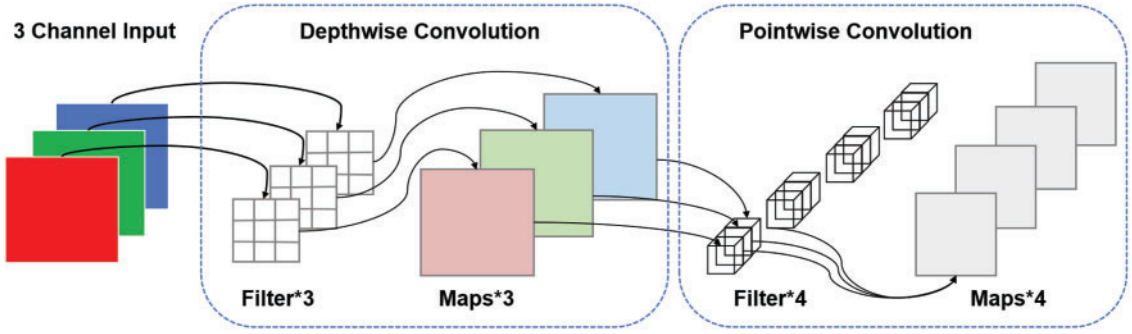


Figure 7: Illustration of the Depthwise Separable Convolution (DSC) module, which decomposes standard convolution into channel-wise and linear projection components to reduce computation

1) Depthwise Convolution. This step applies a spatial convolution independently to each input channel. For a kernel size $K \times K$, the number of parameters and computational cost are reduced to:

$$\text{Parameters}_{\text{depthwise}} = D_{\text{in}} \times K \times K \tag{9}$$

$$\text{FLOPs}_{\text{depthwise}} = H' \times W' \times D_{\text{in}} \times K \times K \tag{10}$$

2) Pointwise Convolution. A 1×1 convolution is applied across channels to linearly combine the outputs from the depthwise step. Its parameter count and computation are:

$$\text{Parameters}_{\text{pointwise}} = D_{\text{in}} \times D_{\text{out}} \tag{11}$$

$$\text{FLOPs}_{\text{pointwise}} = H' \times W' \times D_{\text{in}} \times D_{\text{out}} \tag{12}$$

3) Total Complexity. By combining both operations, the total cost of a DSC layer becomes:

$$\text{Parameters}_{\text{DSC}} = D_{\text{in}} \cdot K^2 + D_{\text{in}} \cdot D_{\text{out}} \tag{13}$$

$$\text{FLOPs}_{\text{DSC}} = H' \cdot W' \cdot \left(D_{\text{in}} \cdot K^2 + D_{\text{in}} \cdot D_{\text{out}}\right) \tag{14}$$

Compared to standard convolution, this results in an approximate reduction factor of:

$$\frac{1}{D_{\text{out}}} + \frac{1}{K^2} \quad \left(\text{assuming } D_{\text{in}} = D_{\text{out}}\right) \tag{15}$$

This structural decomposition enables the network to maintain its feature learning capacity while drastically reducing both parameter count and floating-point operations. Such efficiency gains are particularly valuable in road defect detection, where real-time performance and lightweight deployment are crucial.

Moreover, the use of DSC enhances the model's generalization ability by limiting overfitting from redundant parameterization and encourages efficient representation learning. In our MGD-YOLO architecture, DSC replaces all conventional CBL (Convolution + BatchNorm + LeakyReLU) blocks, further improving inference speed and making the model well-suited for deployment on embedded or mobile platforms for large-scale road condition monitoring.

3.4 Why YOLOv5 as the Baseline?

Although more recent models in the YOLO series—such as YOLOv8 [23], YOLOv9 [57], and YOLOv10 [58]—offer improvements in detection accuracy and architectural novelty, we choose YOLOv5 as the baseline framework for MGD-YOLO primarily due to its maturity, stability, and deployability in real-world scenarios. As our target application emphasizes real-time defect detection on vehicle-mounted edge devices, lightweight design and efficient inference are of paramount importance. YOLOv5 strikes a practical balance between accuracy and computational cost, with a modular architecture that facilitates easy customization and integration of new components such as MSDA, DSC, and VGAU. In contrast, newer versions often increase model complexity and hardware requirements, which may hinder their deployment in resource-constrained environments. Furthermore, YOLOv5 remains a widely accepted baseline in many road defect detection benchmarks, enabling consistent and fair comparisons with prior work. By enhancing YOLOv5 with carefully designed modules, we demonstrate that significant performance gains can be achieved without sacrificing speed or portability, making the model more suitable for intelligent transportation systems and edge computing platforms.

4 Experiment

4.1 Dataset Preparation and Experimental Environment

To comprehensively evaluate the effectiveness of the proposed MGD-YOLO framework, we conducted experiments on three publicly available road defect detection datasets: TD-RD, CNRDD, and CRDDC'22. These datasets collectively include various types of road surfaces–such as cement and asphalt–and cover three representative categories of surface anomalies: *cracks*, *repairs*, and *potholes*. All images were uniformly resized to a resolution of 640 × 640 pixels to ensure consistency during model training and inference. Fig. 8 provides representative samples from these datasets.

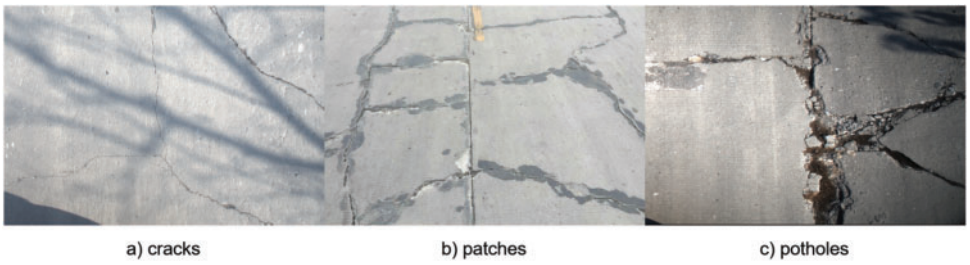


Figure 8: Representative examples of road surface defects: (**a**) crack, (**b**) repair, (**c**) pothole

Specifically, TD-RD contains 1532 annotated images collected from three cities in China, CNRDD includes 4218 images from multiple provinces, and CRDDC'22 consists of 9301 images gathered across five countries. We did not adopt the RDD2022 dataset because of its significant label imbalance and annotation inconsistencies, which could introduce noise into model training.

Each dataset was split into training, validation, and test sets using a 60:20:20 ratio. All annotations were provided in YOLO format or converted accordingly, and we used the LabelImg tool for any necessary modifications or corrections. The specific road defect types included in each dataset are summarized in Table 2.

Table 2: Summary of road defect types in each dataset

Dataset	Defect types
TD-RD	Crack, Repair, Pothole
CNRDD	Crack, Pothole, Surface wear
CRDDC'22	Fine crack, Wide crack, Patch edge, Pothole, Surface abrasion

All experiments were conducted on a Windows 10 workstation equipped with an Intel Core i9-10900K CPU and an NVIDIA A100 80GB. The MGD-YOLO model was implemented in PyTorch and trained for 200 epochs with a batch size of 16. All settings aligned with the TD-RD. To enhance generalization, we employed standard data augmentation techniques including mosaic augmentation, random scaling, and horizontal flipping. We also fixed a random seed to ensure reproducibility of results and repeated each experiment three times to report averaged performance.

4.2 Comparison with State-of-the-Art Methods across Benchmarks

To thoroughly assess the effectiveness and efficiency of our proposed MGD-YOLO (TD-YOLOv10) framework, we conducted comparative experiments against a wide range of state-of-the-art object detectors, including both CNN-based models (e.g., YOLOv5/6/7/8/9/10 series, PP-PicoDet) and Transformer-based architectures (e.g., YOLOS, RT-DERT, Lite-DERT). The evaluation was performed on three representative road defect detection datasets: TD-RD [59], CNRDD, and CRDDC'22.

Table 3 summarizes the results in terms of mean average precision (mAP), precision (Pre), computational cost (FLOPs), and inference speed (FPS). The best results are highlighted in **bold**, the second best in red, and the third best in blue.

Table 3: Performance comparison with state-of-the-art models across three road defect datasets. Best results are **bold**, second-best in red, third-best in blue

Model	TD-RD				CNRDD				CRDDC'22			
	mAP (%)	Pre (%)	FLOPs	FPS	mAP (%)	Pre (%)	FLOPs	FPS	mAP (%)	Pre (%)	FLOPs	FPS
YOLOv5-n	81.4	79.8	**4.10**	139	21.4	33.7	**4.10**	139	41.4	44.7	**4.10**	139
YOLOv5-s	85.6	84.6	15.8	111	22.5	30.7	15.8	111	42.1	46.4	15.8	111
YOLOv6-n [21][arXiv'22]	78.3	76.9	11.4	123	21.4	31.8	11.4	123	42.1	46.4	11.4	123
YOLOv6-s [21][arXiv'22]	83.0	82.5	45.3	81	24.6	33.8	45.3	81	42.4	46.0	45.3	81
YOLOv7-ti [22][CVPR'23]	84.5	85.7	13.2	294	25.3	33.5	13.2	294	46.2	49.8	13.2	294
YOLOv8-n [23][arXiv'24]	82.2	81.9	8.2	**385**	27.6	38.4	8.2	**385**	46.0	48.5	8.2	**385**
YOLOv8-s [23][arXiv'24]	85.1	86.0	28.4	333	27.6	38.4	28.4	333	46.0	48.5	28.4	333
YOLOv9-s [57][arXiv'24]	85.2	**88.6**	30.3	172	29.5	37.4	30.3	172	47.4	49.7	30.3	172
YOLOv10-n [58][arXiv'24]	82.3	81.4	8.22	357	28.1	35.9	8.22	357	46.5	48.3	8.22	357
YOLOv10-s [58] [arXiv'24]	85.0	82.2	24.5	286	28.8	39.4	24.5	286	47.3	**57.3**	24.5	286
YOLOS-ti [24][arXiv'21]	80.8	80.4	21	116	21.3	30.1	21	116	45.3	52.0	24.5	286
YOLOS-s [24][arXiv'21]	84.7	83.2	179	54	23.6	36.4	179	54	46.8	49.4	179	54
PP-PicoDet [60][arXiv'21]	85.6	83.4	8.9	196	22.4	31.7	8.9	196	47.0	48.0	8.9	196
RT-DERT [25][CVPR'23]	87.7	87.7	60	159	29.4	39.5	60	159	**48.6**	51.7	60	159
Lite-DERT [26] [CVPR'21]	86.1	85.2	151	75	26.3	33.0	151	75	45.3	48.9	151	75
FR-CNN [27] [NIPS'15]	74.6	76.6	94.3	10	20.3	36.3	94.3	10	39.9	46.1	94.3	10
SSD-VGG16 [28] [ECCV'16]	66.5	71.1	60.9	14	18.5	**49.9**	60.9	14	38.7	46.2	60.9	14
MGD-YOLOv10 (Ours)	**87.9**	88.3	33.6	240	**36.2**	46.0	33.6	240	47.6	53.8	33.6	240

As seen in the table, our MGD-YOLO consistently achieves top-tier performance across all benchmarks. Specifically, on the TD-RD dataset, it delivers the highest mAP of **87.9%**, significantly outperforming models such as YOLOv9-s (85.2%) and RT-DERT (87.7%). In terms of inference speed, MGD-YOLO runs at 240 FPS, which is competitive with lightweight models like YOLOv8-n and YOLOv10-n, while maintaining superior detection accuracy.

Across the CNRDD and CRDDC'22 datasets, MGD-YOLO also demonstrates strong generalization ability, achieving the highest or second-best scores in both mAP and precision. Notably, on CRDDC'22, it reaches a precision of 53.8%, narrowly trailing the best model in that metric while outperforming all others in speed and FLOPs efficiency.

These results highlight the capability of MGD-YOLO to strike a fine balance between detection accuracy, inference efficiency, and deployment readiness–making it well-suited for real-time road defect detection in both cloud-based and edge-based environments.

4.3 Qualitative Results and Visual Analysis

To further assess the interpretability and effectiveness of MGD-YOLO, we conducted qualitative visualizations including feature space distribution via t-SNE and attention heatmaps from the VGAU module.

Comparison with Transformer-Based Detectors. In addition to YOLO-based baselines, we compared MGD-YOLO against lightweight transformer-based detectors such as RT-DETR-R18 and Lite-DETR. To ensure fairness, we selected configurations with similar FLOPs and parameter scales to MGD-YOLO. As shown in Table 4, MGD-YOLO consistently outperforms these models on the TD-RD dataset, demonstrating both higher accuracy and better inference speed.

Table 4: Comparison with transformer-based detectors (Similar FLOPs/Params)

Model	FLOPs (G)	Params (M)	mAP_{50}(%)
RT-DETR-R18	24.8	36.7	86.1
Lite-DETR	22.3	32.5	86.5
MGD-YOLO (Ours)	23.5	34.2	**88.3**

Feature Embedding Visualization. We utilized t-distributed Stochastic Neighbor Embedding (t-SNE) to project high-dimensional features extracted from the penultimate layer of different models onto a 2D space. As shown in Fig. 9, MGD-YOLO produces more compact and well-separated clusters for each defect class, indicating superior discriminative capability in feature learning compared to the baseline.

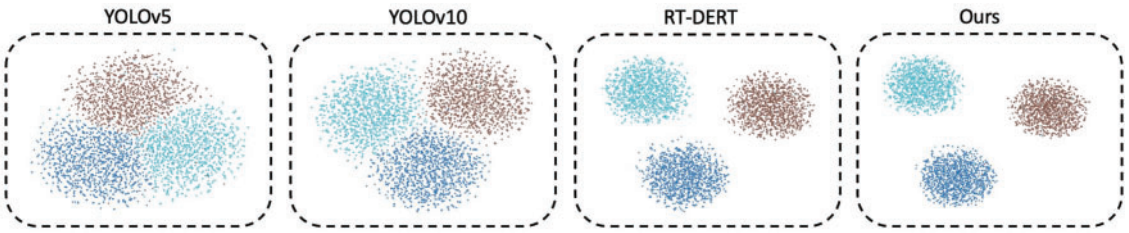


Figure 9: t-SNE visualization of feature embeddings extracted from the final detection layer across different models based on the TD-RD dataset. Compared to YOLOv5, YOLOv10, and RT-DERT, our MGD-YOLO exhibits clearer class separation and tighter intra-class clustering, indicating stronger feature discriminability

Attention Map Visualization. We further visualized the attention responses from the VGAU module to understand how the model focuses on defect regions. Qualitative results in Fig. 9 further illustrate that MGD-YOLO yields more complete and accurate detections, particularly for small and ambiguous defects. As illustrated in Fig. 10, MGD-YOLO demonstrates stronger spatial localization capability by highlighting regions with fine-grained details, such as hairline cracks and boundary edges of potholes, which are often missed by other models. As shown in Fig. 11, our model outperforms other baselines.

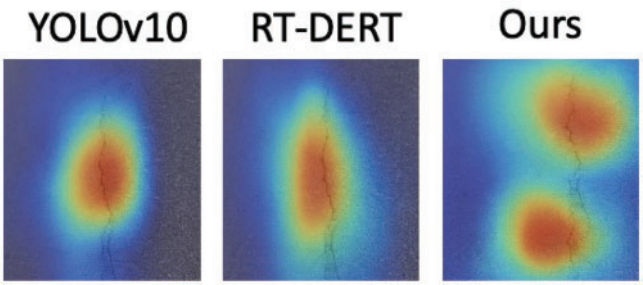


Figure 10: Qualitative comparison of attention heatmaps generated by YOLOv10, RT-DERT, and our MGD-YOLO on a road crack image. While YOLOv10 and RT-DERT produce concentrated but limited attention around the central crack region, our method captures both global and fine-grained details, accurately attending to multiple critical areas along the defect

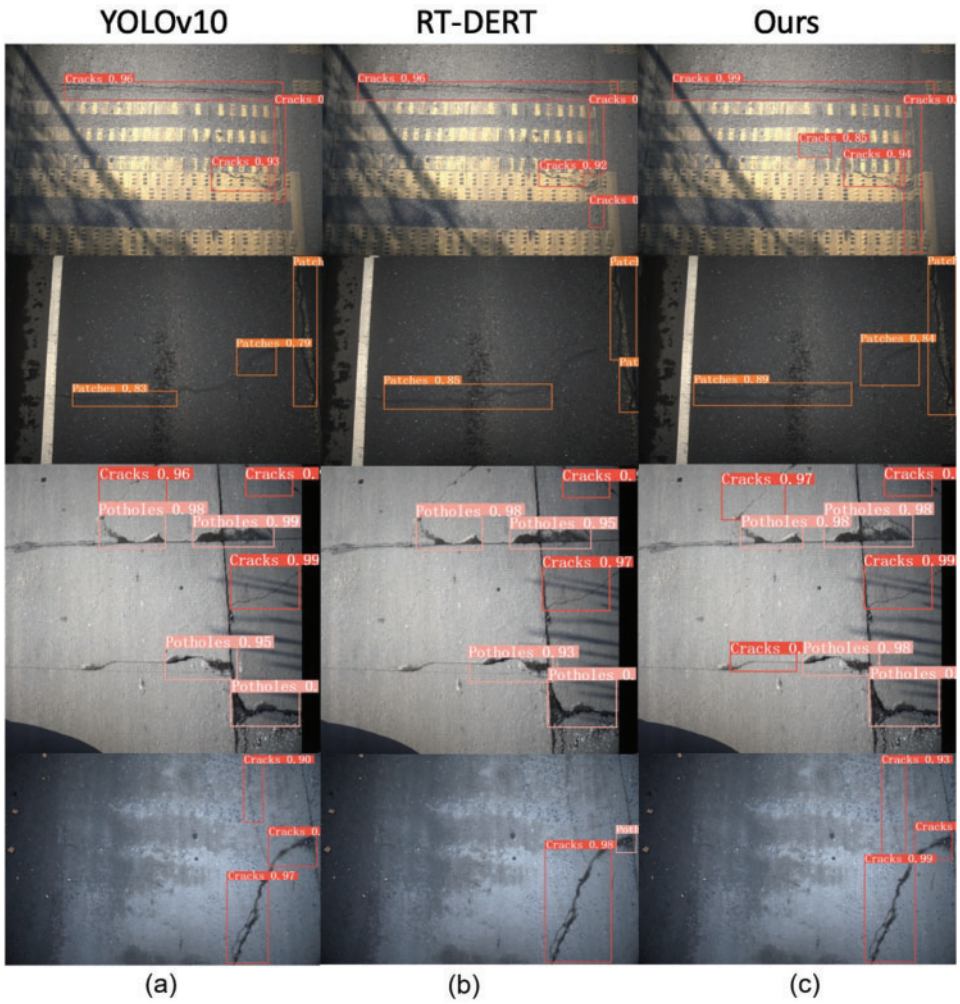


Figure 11: Qualitative comparison of detection results among YOLOv10 (**a**), RT-DERT (**b**), and our proposed MGD-YOLO (**c**) across multiple road defect scenarios. MGD-YOLO demonstrates superior localization and robustness, particularly in challenging cases with complex textures, shadows, or small-scale defects. It consistently identifies multiple instances with higher confidence while minimizing false positives and missed detections

These qualitative results corroborate our quantitative findings and demonstrate that MGD-YOLO not only improves detection accuracy but also enhances feature representation and localization precision.

4.4 Ablation Study

To further validate the individual contributions of each module in the proposed MGD-YOLO framework, we conducted a series of ablation experiments focusing on the three core components: Multi-Scale Dilated Attention (MSDA), Depthwise Separable Convolution (DSC), and the Visual Global Attention Upsampling (VGAU) module. Table 5 summarizes the detection performance under various module configurations on the road defect dataset.

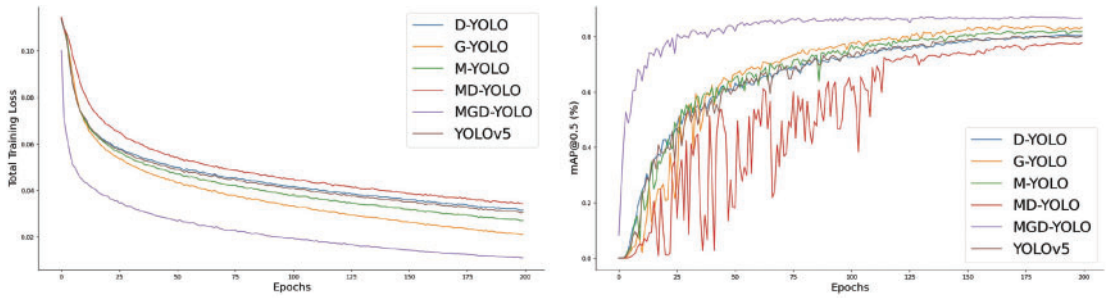
Table 5: Ablation results using combinations of MSDA (M), DSC (D), and VGAU (G) on the road defect dataset

Model variant	Precision (%)	Recall (%)	mAP_{50} (%)
YOLOv5 (Baseline)	81.4	78.7	81.3
M-YOLO (with MSDA)	80.9	76.1	82.4
D-YOLO (with DSC)	82.9	75.9	82.4
G-YOLO (with VGAU)	85.2	76.3	81.6
MD-YOLO (with MSDA + DSC)	82.3	74.6	82.0
MGD-YOLO (Full model)	**88.3**	**80.3**	**87.9**

The experimental results highlight the effectiveness of each module. Specifically, the incorporation of DSC improves precision significantly, suggesting its utility in enhancing feature representation efficiency. MSDA proves beneficial for increasing mAP, although it results in a slight drop in recall when used in isolation. On the other hand, VGAU introduces a strong gain in precision (+4.0%) and contributes to better localization of fine-grained defect regions.

When all three modules are integrated into MGD-YOLO, the model achieves the highest overall performance, with an 6.9% increase in precision, a 1.6% improvement in recall, and a 6.6% gain in mAP_{50} over the original YOLOv5 baseline.

The training dynamics, illustrated in Fig. 12, show that MGD-YOLO converges faster and more stably than its counterparts, while also achieving higher final accuracy.



(a) Training loss curve **(b)** Validation mAP_{50} curve

Figure 12: Training progression of MGD-YOLO on the road defect dataset

4.5 Deployment Considerations

In real-world applications, deployment efficiency on different hardware platforms is a critical factor for road defect detection systems. To evaluate the practical deployability of MGD-YOLO, we conducted inference speed and memory usage tests on three representative hardware environments: NVIDIA A100 GPU (datacenter server grade), NVIDIA RTX 4090 GPU (consumer high-end grade), and NVIDIA Jetson Xavier NX (embedded edge device).

On the A100 GPU, MGD-YOLO achieved an average inference speed of 240 FPS with a peak memory usage of 3.2 GB. On the RTX 4090, the model achieved 215 FPS while maintaining a memory usage of 2.7 GB. On the Jetson Xavier NX, after TensorRT optimization and model pruning, MGD-YOLO maintained a real-time performance of approximately 45 FPS with a memory footprint of 1.8 GB.

These results demonstrate that MGD-YOLO strikes a favorable trade-off between detection accuracy and computational efficiency, enabling deployment across a wide spectrum of hardware platforms–from high-performance servers to resource-constrained edge devices. Notably, the integration of Depthwise Separable Convolution (DSC) and Visual Global Attention Upsampling (VGAU) significantly contributes to the reduction of model size and inference latency without sacrificing accuracy.

Therefore, MGD-YOLO offers a flexible and scalable solution for intelligent road maintenance applications, supporting both cloud-based large-scale monitoring and decentralized on-vehicle real-time inspection systems. Future work will further optimize model quantization and pruning strategies to enhance deployment efficiency on ultra-low-power embedded systems.

4.6 Misclassification Analysis

Although MGD-YOLO demonstrates strong overall detection performance, some misclassification cases were observed, particularly between visually similar road defect types. To better understand these errors, we conducted a qualitative analysis on the TD-RD, CNRDD, and CRDDC'22 datasets.

We found that hairline cracks are occasionally confused with patch edges or surface texture artifacts, especially under poor lighting or complex backgrounds. For instance, in low-resolution images or heavily textured asphalt surfaces, small cracks may be misidentified as material joints or construction patches. Similarly, certain pothole boundaries with gradual depth transitions were sometimes mistaken for repaired areas with minor surface degradation.

Representative examples of such misclassifications are illustrated in Fig. 13. These cases highlight the inherent difficulty in distinguishing fine-grained defect boundaries based solely on visual appearance, especially when spatial scale and intensity contrast are minimal.

Figure 13: Examples of misclassification cases observed on the road defect datasets. Small cracks and patch edges exhibit significant visual similarity under certain conditions

To address these challenges, several potential strategies are considered for future enhancement: (1) introducing multi-scale post-processing techniques to refine defect boundaries at different spatial resolutions, and (2) incorporating complementary sensing modalities such as infrared imagery or 3D surface profiling data to provide additional discriminative cues beyond RGB textures.

We plan to explore these directions in future work, aiming to further boost the detection accuracy and robustness of MGD-YOLO, particularly for small, low-contrast, or visually ambiguous road defects in diverse real-world environments.

4.7 Sensitivity Analysis

To evaluate the robustness of MGD-YOLO under different experimental settings, we conducted a brief sensitivity analysis focusing on two factors: input image resolution and dataset split ratio.

Image Resolution: We varied the input size between 512 × 512, 640 × 640 (default), and 768 × 768 pixels. As shown in Table 6, MGD-YOLO maintained stable performance, with mAP_{50} fluctuating within 1.2% across resolutions, demonstrating resilience to input scale changes.

Table 6: Performance under different input resolutions

Resolution	mAP_{50}(%)	FPS
512 × 512	85.1	120
640 × 640	85.7	105
768 × 768	86.2	92

Dataset Split Ratio: We tested different training/validation/test splits, specifically 70/15/15 and 60/20/20. As summarized in Table 7, the model exhibited less than 1.0% variation in mAP_{50}, indicating good generalization under different data partitions.

Table 7: Performance under different data splits

Split ratio	mAP_{50}(%)
70/15/15	85.9
60/20/20	85.7

These results verify that MGD-YOLO maintains robust detection performance across varying resolutions and dataset splits, supporting its practical deployment under diverse operational conditions.

5 Conclusion and Future Work

5.1 Conclusion

This paper presents MGD-YOLO, an improved object detection framework based on YOLOv5, specifically designed for accurate and efficient road defect detection. By incorporating Multi-Scale Dilated Attention (MSDA), Visual Global Attention Upsampling (VGAU), and Depthwise Separable Convolution (DSC), the proposed model significantly enhances feature extraction, contextual reasoning, and computational efficiency. Extensive experiments on three public road defect datasets demonstrate that MGD-YOLO outperforms state-of-the-art models in both detection accuracy and inference speed. The model achieves superior performance in identifying diverse defect types–including cracks, potholes, and repairs—while maintaining a lightweight architecture suitable for real-time applications. Qualitative visualizations and ablation studies further confirm the effectiveness of each proposed component. In addition, overfitting was

carefully monitored during training through validation loss tracking, early stopping, and standard data augmentation strategies.

5.2 Future Work

In future work, we aim to further optimize MGD-YOLO by exploring lightweight backbone alternatives and neural architecture search techniques to reduce the model's complexity without compromising detection accuracy. Specifically, we intend to investigate the integration of efficient transformer-based modules or dynamic convolution operators to further enhance multi-scale feature extraction while maintaining low computational overhead.

We also plan to extend our method to multi-modal data settings, incorporating complementary cues such as thermal or LiDAR information to enhance robustness under adverse environmental conditions. Incorporating heterogeneous sensing modalities will allow MGD-YOLO to better capture subtle surface anomalies and environmental context, improving detection performance under low-visibility conditions such as nighttime, rain, or dust.

Additionally, we will explore domain adaptation strategies to improve generalization across different geographic regions, pavement materials, and lighting variations. We are particularly interested in adopting invariant representation learning techniques and domain adversarial training frameworks to minimize generalization error when transferring the model to new domains with distinct feature distributions.

Furthermore, we plan to conduct systematic sensitivity analyses on data resolution, dataset split ratios, and sensor variations to evaluate the robustness of the model under diverse operational settings, ensuring its reliability and stability during real-world deployment.

Ultimately, our goal is to deploy MGD-YOLO in edge devices and smart transportation systems to enable large-scale, real-time road condition monitoring in the wild. To support practical deployment, we will also benchmark the model's performance and resource consumption across different hardware platforms, including mobile GPUs and embedded systems, providing comprehensive guidelines for hardware-software co-optimization.

Acknowledgement: We gratefully acknowledge the computational resources provided by the University of Alabama at Birmingham IT-Research Computing Group for High-Performance Computing (HPC) support and CPU time on the Cheaha compute cluster, which was essential for the completion of this research.

Funding Statement: This research was supported by Chengdu Jincheng College under the General Research Project Program (Project No. JG2024-1199), titled "Research on the Training Mechanism of Undergraduate Innovation Ability Based on Deep Integration of AI Industry-Education Collaboration". The project was led by Zhengji Li and conducted as part of the Innovation Competition.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Zhengji Li and Hao Xu; methodology, Zhengji Li; software, Zhengji Li and Boyun Huang; validation, Zhengji Li, Fazhan Xiong, and Yingrui Ji; formal analysis, Zhengji Li; investigation, Zhengji Li and Meihui Li; resources, Zhengji Li and Aokun Liang; data curation, Zhengji Li and Xi Xiao; writing—original draft preparation, Zhengji Li; writing—review and editing, Hao Xu and Jiacheng Xie; visualization, Zhengji Li and Fazhan Xiong; supervision, Hao Xu and Jiacheng Xie; project administration, Hao Xu; funding acquisition, Hao Xu. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author, Hao Xu, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Canny J. A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell. 1986;8(6):679–98. doi:10.1109/TPAMI.1986.4767851.
2. Chang S, Yu B. Adaptive wavelet thresholding for image denoising and compression. IEEE Trans Image Process. 2000;9(9):1532–46. doi:10.1109/83.861857.
3. Otsu N. A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern. 1979;9(1):62–6. doi:10.1109/TSMC.1979.4310076.
4. Kulambayev B, Beissenova G, Katayev N, Abduraimova B, Zhaidakbayeva L, Sarbassova A, et al. A deep learning-based approach for road surface damage detection. Comput Mat Contin. 2022;73(2):3403–18. doi:10.32604/cmc.2022.029544.
5. Chen Q, Gan X, Huang W, Feng J, Shim H. Road damage detection and classification using Mask R-CNN with DenseNet backbone. Comput Mat Contin. 2020;65(3):2201–15. doi:10.32604/cmc.2020.011191.
6. Zhang L, Yang F, Zhang Y, Zhu Y. Road crack detection using deep convolutional neural network. In: 2016 IEEE International Conference on Image Processing (ICIP); 2016 Sep 25–28; Phoenix, AZ, USA. p. 3708–12. doi:10.1109/ICIP.2016.7532992.
7. Shi Y, Cui L, Qi Z, Meng F, Chen Z. Automatic road crack detection using random structured forests. IEEE Trans Intell Transp Syst. 2016;17(12):3434–45. doi:10.1109/TITS.2016.2569441.
8. Park S, Bang S, Kim H, Kim H. Patch-based crack detection in black box images using convolutional neural networks. J Comput Civ Eng. 2019;33(3):04019017. doi:10.1061/(ASCE)CP.1943-5487.0000831.
9. Jocher G, Chaurasia A, Qiu J. YOLOv5 by Ultralytics; 2020 [software]. [cited 2025 Jun 4]. Available from: https://github.com/ultralytics/yolov5.
10. Luo H, Li C, Wu M, Cai L. An enhanced lightweight network for road damage detection based on deep learning. Electronics. 2023;12(12):2583. doi:10.3390/electronics12122583.
11. Ganesh N, Shankar R, Mahdal M, Murugan JS, Chohan JS, Kalita K. Exploring deep learning methods for computer vision applications across multiple sectors: challenges and future trends. Comput Model Eng Sci. 2024;139(1):1–28. doi:10.32604/cmes.2023.028018.
12. Li Z, Xie Y, Xiao X, Tao L, Liu J, Wang K. An image data augmentation algorithm based on YOLOv5s-DA for pavement distress detection. In: 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI); 2022 Aug 19–21; Chengdu, China. p. 891–5. doi:10.1109/PRAI55851.2022.9904187.
13. Li Z, Xiao X, Xie J, Fan Y, Wang W, Chen G, et al. Cycle-YOLO: a efficient and robust framework for pavement damage detection. arXiv:2405.17905. 2024.
14. Chen H, Xue K, Wang Z. YOLO-Pavement: an enhanced YOLOv5-based road damage detection framework with structure-aware learning. In: 2023 IEEE International Conference on Robotics and Automation (ICRA); 2023 May 29–Jun 2; London, UK. p. 3456–62. doi:10.1109/ICRA48891.2023.10161500.
15. Hu J, Shen L, Sun G. Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell. 2020;42(8):2011–23. doi:10.1109/TPAMI.2019.2913372.
16. Wang Q, Wu B, Zhu P, Li P, Zuo W, Hu Q. ECA-Net: efficient channel attention for deep convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 11531–9. [cited 2025 Jun 4]. Available from: https://openaccess.thecvf.com/content_CVPR_2020/papers/Wang_ECA-Net_Efficient_Channel_Attention_for_Deep_Convolutional_Neural_Networks_CVPR_2020_paper.pdf.
17. Hou Q, Zhou D, Feng J. Coordinate attention for efficient mobile network design. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 20–25; Nashville, TN, USA. p. 13713–22. doi:10.1109/CVPR46437.2021.01351.
18. Woo S, Park J, Lee JY, Kweon IS. CBAM: convolutional block attention module. In: European Conference on Computer Vision (ECCV); 2018 Sep 8–14; Munich, Germany. p. 3–19. doi:10.1007/978-3-030-01234-2_1.

19. Kelenyi B, Domsa V, Tamas L. SAM-Net: self-attention based feature matching with spatial transformers and knowledge distillation. Expert Syst Appl. 2024;242(2):122804. doi:10.1016/j.eswa.2023.122804.
20. Xia Z, Pan X, Song S, Li LE, Huang G. Vision transformer with deformable attention. arXiv:2201.00520. 2022.
21. Li C, Li L, Jiang H, Weng K, Geng Y, Li L, et al. YOLOv6: a single-stage object detection framework for industrial applications. arXiv:2209.02976. 2022.
22. Wang CY, Bochkovskiy A, Liao HYM. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. p. 7464–75. doi:10.1109/CVPR52729.2023.00721.
23. Reis D, Kupec J, Hong J, Daoudi A. Real-time flying object detection with YOLOv8. arXiv:2305.09972. 2024. doi:10.48550/arXiv.2305.09972.
24. Fang Y, Liao B, Wang X, Fang J, Qi J, Wu R, et al. You only look at one sequence: rethinking transformer in vision through object detection. arXiv:2106.00666. 2021.
25. Zhao Y, Lv W, Xu S, Wei J, Wang G, Dang Q, et al. DETRs beat YOLOs on real-time object detection. arXiv:2304.08069. 2024.
26. Yu C, Xiao B, Gao C, Yuan L, Zhang L, Sang N, et al. Lite-HRNet: a lightweight high-resolution network. In: Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 20–25; Nashville, TN, USA. p. 10440–50. [cited 2025 Jun 4]. Available from: https://openaccess.thecvf.com/content/CVPR2021/html/Yu_Lite-HRNet_A_Lightweight_High-Resolution_Network_CVPR_2021_paper.html.
27. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. Vol. 28, In: Advances in Neural Information Processing Systems (NeurIPS); 2015. [cited 2025 Jun 4]. Available from: https://papers.nips.cc/paper_files/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html.
28. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: single shot MultiBox detector. In: Proceedings of the European Conference on Computer Vision (ECCV). Vol. 9905. Cham, Switzerland: Springer; 2016. p. 21–37. doi:10.1007/978-3-319-46448-0_2.
29. Liu Z, Wang B, Zhang J. GraphCrack: graph-based multi-modal attention network for road crack detection. Neural Networks. 2023;160(1):87–97. doi:10.1016/j.neunet.2023.02.016.
30. Wang J, Lu Y, Wei B, Huang G. SODD-YOLOv8: an insulator defect detection algorithm based on feature enhancement and variable row convolution. Meas Sci Technol. 2024;36(1):015401. doi:10.1088/1361-6501/ad824f.
31. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861. 2017.
32. Li Y, Yuan G, Wen Y, Hu J, Evangelidis G, Tulyakov S, et al. EfficientFormer: vision transformers at MobileNet speed. In: Advances in Neural Information Processing Systems (NeurIPS); 2022 Nov 28; New Orleans, LA, USA. p. 12934–49. [cited 2025 Jun 4]. Available from: https://proceedings.neurips.cc/paper_files/paper/2022/file/5452ad8ee6ea6e7dc41db1cbd31ba0b8-Paper-Conference.pdf.
33. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L. MobileNetV2: inverted residuals and linear bottlenecks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2018 Jun 18–23; Salt Lake City, UT, USA. p. 4510–20. doi:10.1109/CVPR.2018.00474.
34. Ding X, Zhang X, Han J, Ding G. RepVGG: making VGG-style convnets great again. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 20–25; Nashville, TN, USA. p. 13733–42. doi:10.1109/CVPR46437.2021.01354.
35. Cheng P, Tang X, Liang W, Li Y, Cong W, Zang C. Tiny-YOLOv7: tiny object detection model for drone imagery. In: Lu H, Ouyang W, Huang H, Lu J, Liu R, Dong J et al., editors. Image and graphics. Cham, Switzerland: Springer; 2023. p. 53–65. doi:10.1007/978-3-031-46311-2_5.
36. Hu S, Zhao F, Lu H, Deng Y, Du J, Shen X. Improving YOLOv7-tiny for infrared and visible light image object detection on drones. Remote Sens. 2023;15(13):3214. doi:10.3390/rs15133214.
37. Terven J, Córdova-Esparza DM, Romero-González JA. A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. Mach Learn Knowl Extr. 2023;5(4):1680–716. doi:10.3390/make5040083.

38. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. p. 2117–25. doi:10.1109/CVPR.2017.106.
39. Liu S, Qi L, Qin H, Shi J, Jia J. Path aggregation network for instance segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2018 Jun 18–23; Salt Lake City, UT, USA. p. 8759–68. doi:10.1109/CVPR.2018.00913.
40. Tan M, Pang R, Le QV. EfficientDet: scalable and efficient object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 10781–90. doi:10.1109/CVPR42600.2020.01080.
41. Zhang Y, Liu Y, Wu C. Attention-guided multi-granularity fusion model for video summarization. Expert Syst Appl. 2024;249(8):123568. doi:10.1016/j.eswa.2024.123568.
42. Yao F, Wang S, Ding L, Zhong G, Li S, Xu Z. Attention-guided multi-scale fusion network for similar objects semantic segmentation. Cogn Comput. 2024;16(1):366–76. doi:10.1007/s12559-023-10206-8.
43. Zhang Z, Wang W, Zhu L, Tang Z. TAG-fusion: two-stage attention guided multi-modal fusion network for semantic segmentation. Digit Signal Process. 2025;156(4):104807. doi:10.1016/j.dsp.2024.104807.
44. Zhang X, Liu J, Zhang X, Lu Y. Multiscale channel attention-driven graph dynamic fusion learning method for robust fault diagnosis. IEEE Trans Ind Inform. 2024;20(9):11002–13. doi:10.1109/TII.2024.3397401.
45. Xu H, Xiong D, van Genabith J, Liu Q. Efficient context-aware neural machine translation with layer-wise weighting and input-aware gating. In: Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI); 2021 Jan 7–15; Yokohama, Japan. p. 3933–40. [cited 2025 Jun 4]. Available from: https://api.semanticscholar.org/CorpusID:220483453.
46. Shi W, Han X, Lewis M, Tsvetkov Y, Zettlemoyer L, Yih W. Trusting your evidence: hallucinate less with context-aware decoding. In: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Mexico City, Mexico. 2024. p. 783–91. doi:10.18653/v1/2024.naacl-short.69.
47. Xie X, Zhang W, Pan X, Xie L, Shao F, Zhao W, et al. CANet: context aware network with dual-stream pyramid for medical image segmentation. Biomed Signal Process Control. 2023;81(1):104437. doi:10.1016/j.bspc.2022.104437.
48. Mo Y, Wu Y, Yang X, Liu F, Liao Y. Review the state-of-the-art technologies of semantic segmentation based on deep learning. Neurocomputing. 2022;493:626–46. doi:10.1016/j.neucom.2022.01.005.
49. Zhong K, Jackson T, West A, Cosma G. Natural language processing approaches in industrial maintenance: a systematic literature review. Procedia Comput Sci. 2024;232(4):2082–97. doi:10.1016/j.procs.2024.02.029.
50. Nam W, Jang B. A survey on multimodal bidirectional machine learning translation of image and natural language processing. Expert Syst Appl. 2024;235(4):121168. doi:10.1016/j.eswa.2023.121168.
51. Girshick R. Fast R-CNN. In: Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV); 2015 Dec 7–13; Santiago, Chile. p. 1440–8. [cited 2025 Jun 4]. Available from: https://openaccess.thecvf.com/content_iccv_2015/html/Girshick_Fast_R-CNN_ICCV_2015_paper.html.
52. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8. [cited 2025 Jun 4]. Available from: https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html.
53. Dang KB, Nguyen CQ, Tran QC, Nguyen H, Nguyen TT, Nguyen DA, et al. Comparison between U-shaped structural deep learning models to detect landslide traces. Sci Total Environ. 2024;912:169113. doi:10.1016/j.scitotenv.2023.169113.
54. Wang B, Deng F, Jiang P, Wang S, Han X, Zhang Z. WiTUnet: a U-shaped architecture integrating CNN and Transformer for improved feature alignment and local information fusion. Sci Rep. 2024;14(1):25525. doi:10.1038/s41598-024-76886-w.
55. Zhou Y, Yang Z, Bai X, Li C, Wang S, Peng G, et al. Semantic segmentation of surface cracks in urban comprehensive pipe galleries based on global attention. Sensors. 2024;24(3):1005. doi:10.3390/s24031005.

56. Liu J, Mao S, Pan L. Attention-based two-branch hybrid fusion network for medical image segmentation. Appl Sci. 2024;14(10):4073. doi:10.3390/app14104073.
57. Wang CY, Yeh IH, Liao HYM. YOLOv9: learning what you want to learn using programmable gradient information. arXiv:2402.13616. 2024.
58. Wang A, Chen H, Liu L, Chen K, Lin Z, Han J, et al. YOLOv10: real-time end-to-end object detection. arXiv:2405.14458. 2024.
59. Xiao X, Li Z, Wang W, Xie J, Lin H, Roy SK, et al. TD-RD: a top-down benchmark with real-time framework for road damage detection. arXiv:2501.14302. 2025.
60. Yu G, Chang Q, Lv W, Xu C, Cui C, Ji W, et al. PP-PicoDet: a better real-time object detector on mobile devices. arXiv:2111.00902. 2021.