

Few-shot road surface defect detection using enhanced multi-level multi-scale graph neural network

Muwang Cao^a, Yang Xing^{b,*}

^a Advanced Mechanical Engineering, Cranfield University, Bedford MK43 0AL, UK

^b Center for Connected and Assured Autonomy, Cranfield University, Bedford MK43 0AL, UK

ARTICLE INFO

Keywords:

Road surface defects
Graph neural networks
Kolmogorov-Arnold networks
Data augmentation
Attention mechanism

ABSTRACT

Detecting road surface defects in complex environments remains challenging due to and data scarcity and insufficient detail perception. In this paper, a stable diffusion and fractal texture hybrid data enhancement algorithm (DFH-Enhancement) is proposed to generate images of harsh environments that retain key semantic information for model training. An enhanced multi-scale graph neural network (MSKA-GNN) is also proposed for few-shot scenarios. This model converts images into graph data containing multi-scale information through graph neural network (CNN) and superpixel region aggregation, employs kolmogorov-arnold network (KAN) in place of multi-layer perceptron (MLP) to enhance the non-linear fitting capability of graph convolution, and combines multi-scale feature fusion (MSFF) and dual attention mechanism (DAM) to optimise feature extraction. Our experiments on two public datasets show that MSKA-GNN performs better than seven existing state-of-the-art models, achieving an accuracy of 81.1%.

1. Introduction

Autonomous driving technology is shifting vehicle suspension systems from passive response to fully active control, adjusting damping in real time based on road surface detection to improve ride comfort and stability [1–4]. The effectiveness of the fully active suspension system critically depends on the accurate detection of road surface defects (such as cracks and potholes) [5]. Thus, research into road defect detection methods holds significant value for the vehicle suspension systems [6].

Early defect detection technologies mainly rely on physical sensors such as lidar to construct 3D point cloud models of the road surface [7,8]. Although LiDAR is effective at capturing geometric variations, it struggles to detect subtle textural defects. Additionally, harsh weather, such as rain, snow and fog, can cause significant degradation of the point cloud due to laser scattering [9,10]. Making it difficult for physical sensors to meet the high-precision requirements of fully active suspension systems in all weather conditions.

In recent years, vision-based deep learning has become the mainstream solution for defect detection [11]. Relevant detection models can be divided into three main types: convolutional neural networks (CNNs), graph neural networks (GNNs) and Vision Transformers (ViTs) [12]. CNNs extract features from images through convolution and pooling,

this process inevitably results in the loss of spatial detail, resulting in inadequate recognition capabilities for subtle defects such as fine cracks [13]. Although Li et al. proposed CarNet by incorporating an upsampling feature pyramid and Chen et al. developed the Multi-scale Mobile Attention Network (MANet) with a hybrid attention mechanism, both methods improved the detection accuracy of crack and other defect types to some extent. However, neither fundamentally addresses the loss of fine-grained details and topological information. [14,15].

Graph neural networks (GNNs) convert images into graph data consisting of nodes and edges, enabling better alignment with the morphology of defects such as cracks [16]. Recent studies have demonstrated that effective graph construction and graph representation learning are crucial for improving feature discrimination in graph-based image analysis. For example, progressive graph contrastive clustering and lightweight self-supervised graph contrastive learning have significantly enhanced the robustness of graph representations by optimising graph topology and exploiting contrastive learning strategies [17,18]. In addition, adaptive multi-receptive field graph aggregation and graph convolution-based feature extraction have further improved the capability of GNNs to capture multi-scale spatial relationships and discriminative features, providing valuable insights for graph-based image understanding [19,20]. The ability of neural networks to

* Corresponding author.

E-mail addresses: Muwang.Cao.582@cranfield.ac.uk (M. Cao), Yang.x@cranfield.ac.uk (Y. Xing).

<https://doi.org/10.1016/j.imavis.2026.106127>

Received 3 March 2026; Received in revised form 13 July 2026; Accepted 16 July 2026

Available online 18 July 2026

0262-8856/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

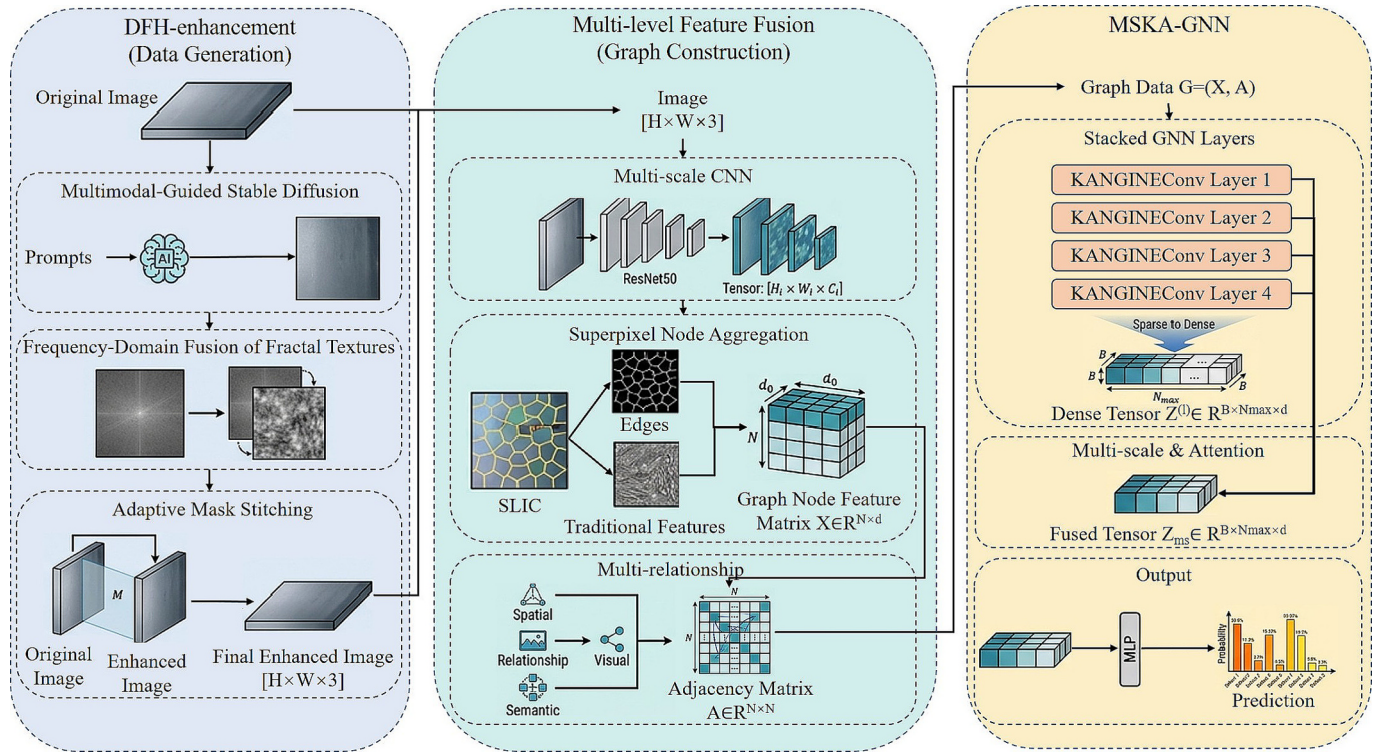


Fig. 1. Overview of the End-to-end small-sample defect recognition framework. Using diffusion models to augment sparse data, and constructing a multi-scale graph input MSKA-GNN to achieve classification of pavement defects in complex environments.

extract information from graph data is also a key factor in determining the performance. For example, Ma et al. introduced an attention mechanism in the spatial dimension and a Transformer in the temporal dimension, thereby improving the model's stability [21]. But these improvements have not resolved the problem that GNNs are unable to adequately represent features with vastly different scales in complex environments.

Visual Transformers (ViTs) use self-attention mechanisms to simulate long-range dependencies across the whole image, exhibiting exceptional ability to capture global contextual information [22]. This also result in the loss of local information, making the detection of minute cracks or small potholes challenging [23]. Although researchers have made some attempts. For example, Zhang et al. propose a hybrid model named PoFormer, where a Transformer acts as the encoder to capture long-range contextual dependencies, and a CNN acts as the decoder to recover fine-grained spatial features, significantly improving the accuracy of pothole segmentation in complex environments [24]. Wang et al. designed RCP-YOLO, which combines CNNs and Transformers while introducing a feature pyramid and an aggregated multi-coordinate attention mechanism, improving the model's ability to localise defects [25]. These also result in extremely high computational costs.

Beyond architectural limitations, models trained under ideal conditions may experience a decline in performance when deployed in adverse environments. Recent studies have demonstrated that advanced visual preprocessing techniques can improve the robustness of downstream vision tasks by preserving structural information and enhancing feature representations. Cong et al. proposed a deep homography-based image stitching framework that combines multi-scale feature extraction with attention mechanisms to improve small-object preservation, while Qin et al. further developed efficient unsupervised image stitching frameworks based on deep homography estimation and feature frequency awareness to improve image alignment and structural consistency [26–28]. However, these approaches primarily focus on image alignment and enhancement rather than addressing the scarcity of

adverse-weather training data. To alleviate this limitation, Qian et al. proposed WeatherEdit, which employs a diffusion-model adapter to edit weather backgrounds while modelling rain, snow, and fog particles using 4D Gaussian fields to synthesise diverse weather conditions [29]. Although these methods improve either image quality or data diversity, they still suffer from limitations such as considerable computational overhead or an insufficient ability to generate sufficiently challenging adverse-weather samples.

Deploying real-world road defect detection therefore requires solving two challenges: achieving a structural balance between fine detail perception and global topological understanding, and maintaining robust generalisation in complex environments without relying on massive annotated datasets. The MSKA-GNN proposed in this paper, designed specifically for road defect detection using a small number of images, solves both the data scarcity and feature extraction limitations. The main contributions of our work are as follows:

- (1) Proposing the DFH-Enhancement, a hybrid data augmentation strategy combining stable diffusion and fractal texture synthesis. This approach generates visually diverse, semantically consistent defect samples under simulated adverse weather, substantially improving model generalisation in few-shot scenarios.
- (2) Proposing the MSKA-GNN architecture, which employs multi-level graph construction modules to capture structural information ranging from local edge textures to global semantics. By embedding KAN and attention mechanisms into multi-scale graph convolutions, it enhances the ability to extract multi-scale information, thereby enabling the accurate recognition of various defects in complex environments.
- (3) We tested MSKA-GNN and seven other advanced models on two publicly available datasets, extensively validating the performance of the models.

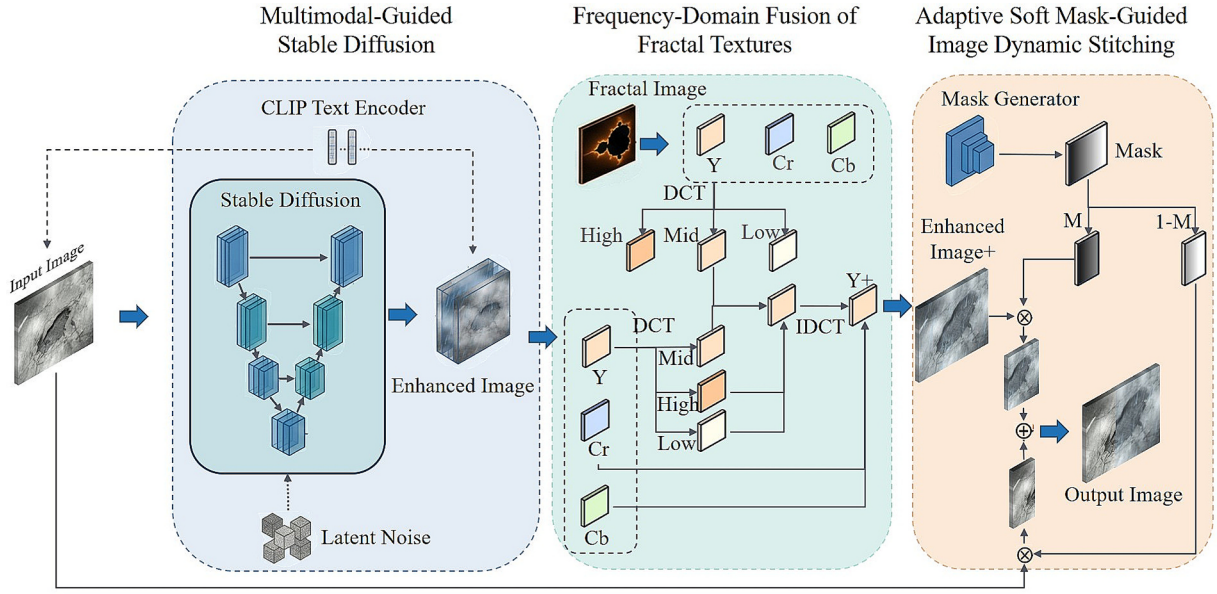


Fig. 2. Overview of the DFH-Enhancement framework. Using Stable Diffusion with frequency domain fusion to enhance texture details, with adaptive masking enabling differentiated enhancement of diseased areas and background.

2. Method

This study develops an end-to-end small-sample defect recognition framework from data to features. In Section 2.1, the DFH-Enhancement is designed, which expands road defect datasets featuring visual characteristics such as rain through environmental transformation, texture fusion, and adaptive masking stitching, while preserving key semantic information. In Section 2.2, a multi-scale graph construction method is designed that integrates CNN multi-level features with traditional features, transforming the enhanced images into a multi-scale graph structure encompassing spatial, visual, and semantic relationships. In Section 2.3, the MSKA-GNN is constructed. This model enhances nonlinear modelling capabilities through KAN convolutional layers, dynamically integrating local and global features via node-level and scale-level attention mechanisms. Achieving accurate identification and classification of various road surface defects. The overall operating principle is shown in Fig. 1.

2.1. Sample generation method based on DFH-enhancement

To address the challenges of sparse data and high annotation costs in extreme environments, this paper proposes a data enhancement method (DFH-Enhancement) based on stable diffusion and fractal texture fusion. Through multi-level image synthesis and semantic control, this method generates enhanced images for extreme environments while preserving the semantic consistency of key targets. DFH-Enhancement comprises three core components: diffusion-based image transformation, fractal texture fusion, and adaptive mask integration. The overall workflow is illustrated in Fig. 2.

(1) Multimodal-Guided Stable Diffusion Model Image Environment Transformation.

To achieve visual transfer of images under extreme conditions such as heavy rain and extreme lighting conditions, This paper employs the pre-trained Stable Diffusion model v1.5 for image environment transformation. For the input original image I_0 , by constructing a multimodal prompt P encompassing three dimensions: weather, scene, and viewpoint, guiding diffusion models to generate enhanced images I_{diff} . To balance the intensity of changes with semantic preservation, a diffusion intensity parameter σ is introduced to constrain the diffusion process. Analysis of the diffusion intensity can be found in Section 4.1.1. The

complete mathematical expression for this process is as follows:

$$I_{diff} = G(I_0, P, \sigma) \quad (1)$$

To ensure enhanced semantic consistency between the augmented images and the original images, the Contrastive Language-Image Pre-training model (CLIP) is introduced as a semantic supervisor. The similarity between the enhanced image I_{diff} and label L is set to be no less than τ times the similarity between the original image I_0 and label L . The sim is a cosine similarity function based on image text information and graphical information, used to measure the semantic consistency of two feature categories within the embedding space. Analysis of the CLIP similarity threshold can be found in Section 4.1.1.

$$sim(\text{CLIP}(I_{diff}), \text{CLIP}(L)) \geq \tau \times sim(\text{CLIP}(I_0), \text{CLIP}(L)) \quad (2)$$

(2) Frequency-domain fusion of fractal textures for simulating extreme visual effects.

To accurately simulate special visual effects such as reflections and texture distortions under extreme conditions, this paper introduces a fractal texture fusion module. Images are generated based on the Mandelbrot set. Then Gaussian blurring is applied to reduce sharpness, perform random colour mapping to enhance naturalism, and introduce Gaussian noise to simulate environmental interference. This process ultimately yields the fractal image F . To achieve a refined fusion of the fractal texture and the enhanced image, this paper will fuse the fractal image F with the enhanced image I in the frequency domain. The fusion process first converts the image to the YCrCb colour space, which separates the image's luminance and chrominance information into the Red-Difference Chrominance Channel (Cr), Blue-Difference Chrominance Channel (Cb), and Luminance Channel (Y). Subsequently, the luminance channels Y_{diff} and $Y_{fractal}$ from the I_{diff} and F frames undergo two-dimensional DCT transformations to yield their frequency domain representations D_{diff} and $D_{fractal}$. The frequency domain comprises low, medium, and high frequency components. In the low and high frequency bands, only $D_{diff,low}$ and $D_{diff,high}$ is retained to ensure image structural stability and prevent detail loss. In the medium frequency band, $D_{diff,mid}$ and $D_{fractal,mid}$ are proportionally blended. Analysis of the fractal fusion ratio can be found in Section 4.1.1. Finally, the merged frequency domain D_{fused} undergoes inverse DCT to reconstruct the merged luminance channel Y_{fused} . This is then combined with the original Cr and Cb

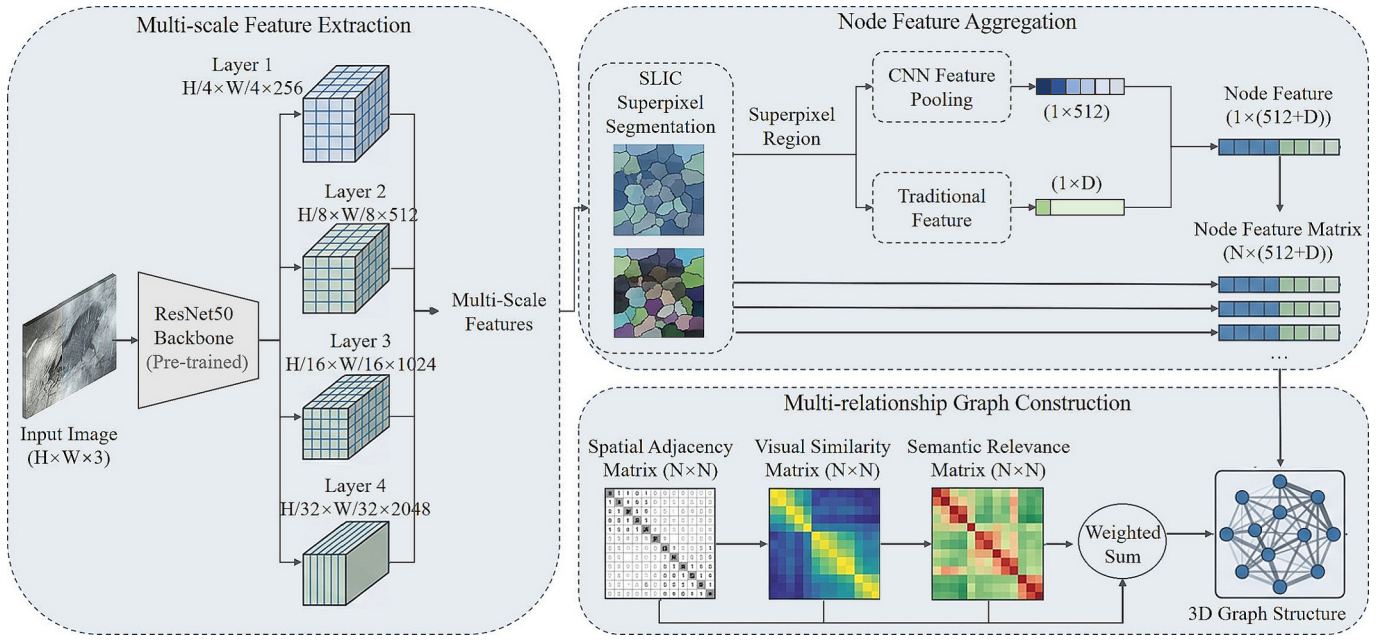


Fig. 3. Overview of the Multi-scale graph construction framework. Using ResNet50 to extract multi-scale deep features, then aggregating them into graph nodes via superpixel aggregation to construct a multi-relational graph incorporating spatial, feature, and semantic associations.

channels, converted back to RGB colour space, yielding the final merged fractal texture-enhanced image I_{fused} . The spatial transformations between RGB and YCrCb, along with the *DCT* transformation, are implemented using the `cv2.cvtColor(CC)` and `np.idct(IDCT)` functions within the OpenCV library. And the mathematical expression for this fusion process is as follows:

$$I_{fused} = CC \left(IDCT \left(D_{diff,low} + \alpha \bullet D_{fractal,mid} + (1 - \alpha) \bullet D_{diff,mid} + D_{diff,high} \right), Cr, Cb \right) \quad (3)$$

(3) Adaptive Soft Mask-Guided Image Dynamic Stitching.

The original image and the fused image are dynamically concatenated using an adaptive mask M . The network responsible for mask generation employs a lightweight encoder-decoder architecture. This network takes as input a 6-channel tensor formed by concatenating the original image and the fused image along the channel dimension. The encoder extracts features via two consecutive convolutional layers, which are then fed to the decoder via activation functions and pooling layers. The decoder processes the features through a 32-channel convolutional layer, applies bilinear interpolation to restore the original spatial resolution, and finally outputs a continuous mask using a convolutional layer with a sigmoid activation function. The hybrid enhanced image is obtained as follows:

$$I_{end} = (1 - M) \odot I_0 + M \odot I_{fused} \quad (4)$$

Where $\odot$ represents Hadama's product (element-wise multiplication).

The training process for the mask generator uses the Adam optimiser, with a learning rate set to 0.01, and two epochs of training for each pair of images. The loss function ζ aims to maximise CLIP-based semantic confidence whilst minimising mask variance to encourage spatial continuity:

$$\zeta = -\mathbb{E}[\|\mathcal{F}_{CLIP}(I_{end})\|_2 + \lambda_{reg} Var(M)] \quad (5)$$

Where $\mathcal{F}_{CLIP}(I_{end})$ represents the visual features extracted by the CLIP image encoder. $\mathbb{E}[\|\bullet\|_2]$ calculates the mean of the feature norm, representing semantic confidence. $Var(M)$ calculates the variance of the generated mask matrix, serving as a regularisation term. λ_{reg} is a weight hyperparameter.

2.2. Graph neural networks with multi-level feature fusion

To fully exploit the multi-level visual semantics of images and accurately model complex inter-region relationships, this paper proposes a graph structure construction method that integrates CNN multi-level features with traditional image features. This approach encompasses multi-scale feature extraction, superpixel region feature aggregation, and multi-relationship graph construction, with the overall workflow illustrated in Fig. 3.

(1) Multi-scale Feature Extraction Module.

For the image I with input dimensions height (H), width (W) and number of channels (C), a pre-trained ResNet50 model is used as the feature extractor, extracting four key feature levels from the final convolutional layers (third convolutional layer) of the four residual blocks, respectively. Layer 1 is highly sensitive to fine-grained edges and textures (such as minute cracks) and is responsible for extracting shallow-level features that retain high spatial resolution. Layer 2 is responsible for capturing local geometric patterns, such as pits or crack networks. Layers 3 and 4 are responsible for extracting global structure and contextual information, and for distinguishing between noise and defects. The mathematical expression for this process is as follows:

$$F_l = \phi_l(I) \in \mathbb{R}^{H_l \times W_l \times C_l} \quad (6)$$

(2) Node Feature Aggregation Module.

To aggregate pixel-level features into region-level representations with semantic integrity, this module employs Simple Linear Iterative Clustering (SLIC) to partition the image into multiple superpixel regions. SLIC can efficiently generate superpixel regions R_k with semantic consistency. For each superpixel region R_k , perform feature aggregation at each feature level to generate a multi-scale feature representation for that region. Specifically, for each feature map F_l , statistical features (mean, standard deviation, maximum, and minimum values) are computed within the superpixel region R_k , forming the feature aggregation function $\psi(F_l, R_k)$ for that region at that level. Subsequently, $\psi(F_l, R_k)$ is aggregated to yield shallow, medium, and deep CNN features respectively, along with semantic features incorporating global characteristics. Concurrently, traditional features encompassing colour, texture, and shape are extracted. Finally, the multi-level features and traditional features are concatenated by channel to obtain the feature

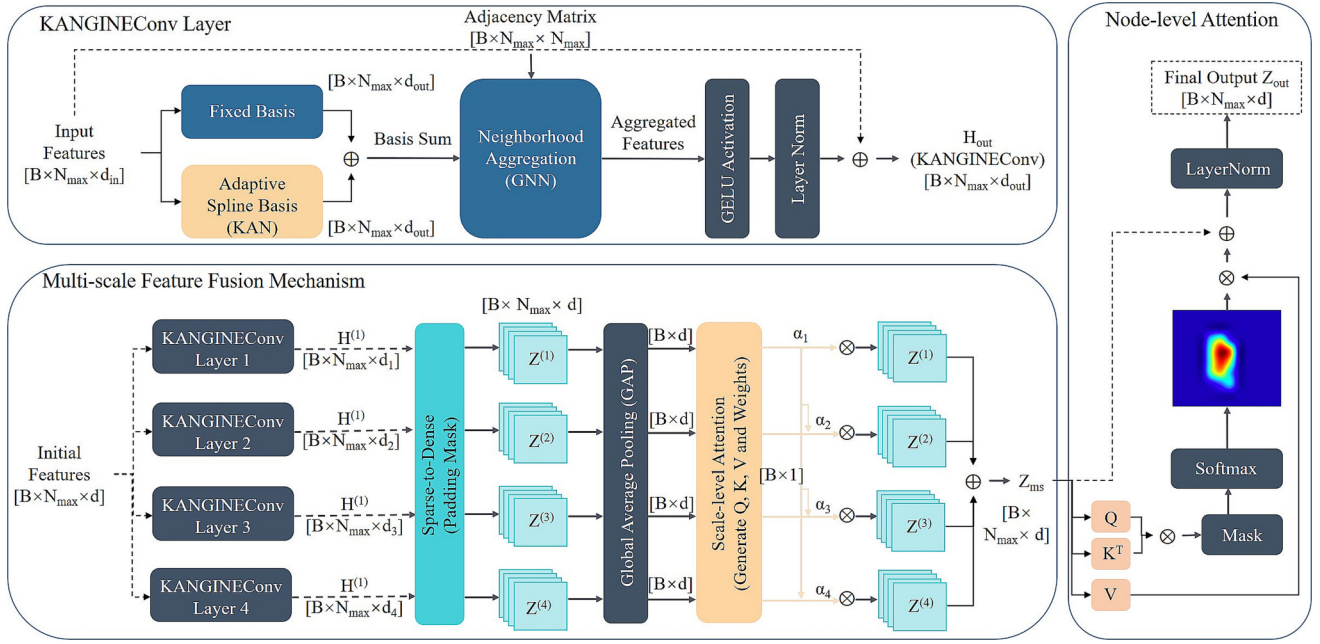


Fig. 4. Overview of the MSKA-GNN framework. Employing KANGINEConv to extract multi-scale image features, and following sparse-to-dense transformation, feature adaptation is achieved through dual attention mechanisms at both node and scale levels.

representation of the superpixel node. In this process, the CNN's feature vectors are fixed at 512 dimensions through uniform sampling and padding. These are then concatenated with traditional shape and texture descriptors to form a dense, fixed-length 1D feature vector for each graph node. The mathematical expression for this process is as follows:

$$\mathbf{x}_k = \left[\mathbf{f}_k^{\text{trad}} \oplus \mathbf{f}_k^{\text{shallow}} \oplus \mathbf{f}_k^{\text{mid}} \oplus \mathbf{f}_k^{\text{deep}} \oplus \mathbf{f}_k^{\text{semantic}} \right] \quad (7)$$

(3) Multi-relationship Graph Construction.

To comprehensively model spatial, visual and semantic relationships between image regions, this module integrates spatial adjacency, traditional feature similarity and semantic feature similarity to construct a triple-relationship modelling framework.

First, construct the spatial adjacency matrix $A_{ij}^{\text{spatial}} = \mathbb{I}(\|P_i - P_j\|_2 < \tau_s)$ based on the spatial positions centered on superpixels. This matrix represents the spatial adjacency relationship between superpixel nodes i and j , with values of 1 (adjacent) or 0 (non-adjacent). The decision criterion for the indicator function $\mathbb{I}$ is whether the Euclidean distance between the center coordinates P_i and P_j (the mean pixel coordinates within the region) of superpixel nodes i and j is less than the spatial adjacency threshold (15% of the image diagonal length).

Secondly, feature similarity matrices $A_{ij}^{\text{appearance}} = \exp\left(-\frac{\|\mathbf{f}_i^{\text{trad}} - \mathbf{f}_j^{\text{trad}}\|_2^2}{\sigma_a^2}\right)$ and semantic relevance matrices $A_{ij}^{\text{semantic}} = \exp\left(-\frac{\|\mathbf{f}_i^{\text{semantic}} - \mathbf{f}_j^{\text{semantic}}\|_2^2}{\sigma_s^2}\right)$ are constructed based on traditional and semantic

features respectively. The visual similarity matrix represents the traditional feature similarity between superpixel nodes i and j . This is calculated by first determining the squared Euclidean distance between the traditional feature vectors of superpixels i and j , then dividing this by the median squared distance of all superpixel pairs' traditional features. Finally, the result is inverted and applied to a natural exponential function to yield a similarity value within the range 0 to 1. The semantic

correlation matrix represents the semantic feature similarity between superpixel nodes i and j . First, the squared Euclidean distance between the semantic feature vectors of superpixels i and j is calculated. This value is then divided by the median squared semantic feature distance across all superpixel pairs. Taking the negative of this quotient and applying it to a natural exponential function yields a semantic similarity score ranging from 0 to 1.

Finally, the final adjacency matrix A is obtained through weighted averaging.

2.3. Multi-scale attention graph neural network enhanced with KAN

This section develops an enhanced graph neural network algorithm based on KAN-enhanced graph convolutions and multi-scale fusion (MSKA-GNN), comprising three key modules: First, designing the KAN-enhancing graph convolutional layer (KANGINECONV) by adopting trainable adaptive basis functions to replace fixed activation functions, improving the model's ability to capture complex nonlinear relationships within graphs; Second, building an attention-weighted multi-scale feature fusion mechanism. This achieves comprehensive capture of hierarchical information within the graph structure by stacking graph convolutional layers of varying depths and dynamically integrating local-to-global features via cross-scale attention. Third, introducing a dual-level attention module. This combines node-level self-attention with scale-level attention to optimise feature representation and fusion strategies at both the feature dimension and network depth levels. The overall workflow is illustrated in Fig. 4.

2.3.1. KAN-enhanced graph convolutional layer

To address the limitations of multi-layer perceptions in representing non-linear relationships, this section proposes a graph convolutional layer enhanced by the Kolmogorov-Arnold network (KAN) to improve the modelling capability of graph neural networks for complex non-linear associations.

The KAN network is based on the Kolmogorov-Arnold Representation Theorem, which states that any multivariate continuous function can be represented as a finite combination of univariate functions. Unlike traditional MLP networks that employ fixed activation functions, the KAN network approximates complex functional relationships by

learning adaptive basis functions, thereby achieving superior nonlinear fitting capabilities [30].

KANLinear constitutes the core component of KAN, with its forward propagation process achieved through a combination of fixed basis function terms and adaptive spline basis function terms. The fixed basis function is derived by multiplying the mapping of the linear layer input vector x by the corresponding weight matrix W_{base} . The adaptive spline basis function is the sum of K adaptive spline basis functions multiplied by their individual weights. The mathematical expression for this process is as follows:

$$\text{KANLinear}(x) = \phi_{base}(x)W_{base} + \sum_{k=1}^K \phi_{spline,k}(x)W_{spline,k} \quad (8)$$

Then integrate KAN into the graph convolutional layer to construct KANGINEConv. The key is to first aggregate neighbouring information, then update node features. During the aggregation step, for each target node i in network layer l , calculate the feature vector $h_j^{(l)}$ for each neighbouring node j , along with the nonlinear feature of the edge feature vector e_{ij} relative to the node feature vector $h_j^{(l)}$ in the same dimensionality. Finally, sum these values to obtain the aggregated information for the relevant network layer. The mathematical expression for this process is as follows:

$$m_i^{(l)} = \sum_{j \in N(i)} (h_j^{(l)} + \text{KAN}_{edge}(e_{ij})) \quad (9)$$

During the node update step, update the feature representation of the target node based on the aggregated results. Specifically, we concurrently perform aggregated information processing and residual feature fusion to obtain the updated feature vector for target node i at layer $l + 1$. The aggregated results undergo a nonlinear transformation via the node aggregation information encoder to capture complex correlations among local neighbour information. Subsequently, the nonlinear activation function *GELU* and layer normalisation *LayerNorm* are successively applied to yield the processed aggregated information. The residual features are obtained by multiplying the original feature vector $h_i^{(l)}$ of target node i at layer l by the linear projection matrix W_{res} , which incorporates learnable scaling factors λ . The mathematical expression for this process is as follows:

$$h_i^{(l+1)} = \text{LayerNorm}\left(\text{GELU}\left(\text{KAN}_{node}\left(m_i^{(l)}\right)\right)\right) + \lambda \cdot W_{res} h_i^{(l)} \quad (10)$$

2.3.2. Multi-scale feature fusion mechanism

The structural information within graph data exhibits multi-scale characteristics, displaying a hierarchical distribution from node-level local associations to graph-level global topologies. Traditional single-scale feature extraction struggles to comprehensively capture this cross-scale structural information. Therefore, this paper designs an attention-weighted multi-scale feature fusion mechanism to fully capture such information.

First, a multi-scale feature extractor, *GNN-Layers*, is constructed by stacking L consecutive graph convolutional layers (KANGINEConv). The shallow layers capture direct adjacency information between nodes, forming local feature representations; the middle layers integrate neighbourhood information to form community-level features; the deep layers aggregate global information to form graph-level macroscopic features. This yields feature representations with distinct receptive fields.

$$H^{(1)}, H^{(2)}, \dots, H^{(L)} = \text{GNN-Layers} \quad (11)$$

Secondly, as the number of nodes N may vary across different graphs and the original $H^{(l)}$ features are in sparse format, it is necessary to convert the sparse node features $H^{(l)}$ from each layer into a unified dense

representation format $G^{(l)}$ (dimension $B \times N_{max} \times d$, where B denotes the training batch size, N_{max} represents the maximum number of nodes across all graphs in the batch, and d is the feature dimension) via appropriate functions. This preserves the original graph's topological structural information. This transformation first zero-pads $H^{(l)}$ for each graph within the batch, expanding the node count to N_{max} . Subsequently, a $B \times N_{max}$ -dimensional mask vector b is generated to mark valid nodes. The final output consists of the concatenation of the dense tensor and the mask. The mathematical expression for this process is as follows:

$$G^{(l)} = \text{ToDenseBatch}(H^{(l)}, b), G^{(l)} \in \mathbb{R}^{B \times N_{max} \times d} \quad (12)$$

In particular, graphs in real-world datasets vary in size. To perform parallel matrix operations within a mini-batch, N_{max} is dynamically defined as the maximum number of nodes strictly present in all graphs within the current mini-batch. Graphs with fewer nodes are augmented with zero-padded virtual nodes, whilst a binary mask vector b is generated, where real nodes are assigned a value of 1 and virtual nodes a value of 0. During pooling, the aggregated graph representation is computed by dividing the sum of the features of the valid nodes by the number of real nodes, rather than by the padded dimension N_{max} , thereby ensuring scale invariance. In the multi-head attention module, the vector b is applied directly to the attention score matrix prior to normalisation. Virtual nodes are assigned a value of negative infinity, thereby ensuring that their attention weights are zero, which strictly guarantees that padding nodes do not participate in information aggregation.

Finally, we introduce an attention mechanism during the multi-scale feature fusion stage. The first step is to conduct global average pooling on the effective dense features $G^{(l)}$ at each scale within each batch (where validity is confirmed via the mask vector b_{kj}), to obtain a summarised representation for each scale. Then transpose it and sum the elements to obtain the global summary feature matrix G_{mean} .

$$G_{mean}^{(l)} = \frac{1}{\sum_{j=1}^{N_{max}} b_{kj}} \sum_{j=1}^{N_{max}} b_{kj} \bullet G_{kj}^{(l)}, k = 1, 2, \dots, B \quad (13)$$

$$G_{mean} = [G_{mean}^{(1)}, G_{mean}^{(2)}, \dots, G_{mean}^{(L)}]^T \quad (14)$$

Step two involves mapping the global summary feature G_{mean} onto the query matrix Q , key matrix K , and value matrix V respectively through the learnable projection matrices W_q , W_k , and W_v . Subsequently, a normalisation function transforms these into a similarity matrix, which serves as the scale attention weight matrix A_{scale} . This is then multiplied by the value matrix V to yield the attention-weighted scale summary feature matrix $G_{attended}$.

$$Q = G_{mean} W_q, K = G_{mean} W_k, V = G_{mean} W_v \quad (15)$$

$$A_{scale} = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right), G_{attended} = A_{scale} V \quad (16)$$

Step three involves normalising the summary features of the scale l within $G_{attended}$ to obtain normalised attention weights. These are then multiplied by the corresponding feature $H^{(l)}$, and after weighted summation, the multi-scale fused node feature matrix H_{fused} is obtained.

$$H_{fused} = \sum_{l=1}^L a_l \cdot H^{(l)}, a_l = \text{Softmax}(G_{attended,l}) \quad (17)$$

2.3.3. Dual attention mechanism

To improve feature space representation and cross-scale feature extraction, this paper proposes a dual-layer attention module consisting of node-level and scale-level attention. Node-level attention optimises feature dimension modelling, while scale-level attention dynamically distributes weights for features of different scales on the network's deeper dimensions. This combined approach enhances the model's

Table 1
Network architecture configuration.

Module name		Layer type	Input dimensions	Output dimensions	Key parameters
Input projection module		Linear	512	256	bias = True
Multi-scale feature extraction module	Level 1	KANGINEConv	256	256	grid_size = 5 spline_order = 3
	Level 2	KANGINEConv	256	256	grid_size = 5 spline_order = 3
	Level 3	KANGINEConv	256	256	grid_size = 5 spline_order = 3
	Level 4	KANGINEConv	256	256	grid_size = 5 spline_order = 3
Dual attention fusion module	Scale-level	MultiScaleFusion	4×256	256	num_scales = 4 attention_dim = 64
	Node-level	NodeLevelAttention	256	256	num_heads = 8 head_dim = 32
Classification output module	Level 1	Linear + GELU	256	128	bias = True dropout = 0.2
	Level 2	Linear	128	3	bias = True

ability to capture key features in complex graphs.

Node-level attention adopts a multi-head self-attention architecture, enabling each node to simultaneously focus on all other nodes in the graph and dynamically adjust information flow based on relevance strength. After attention-weighted aggregation, node features are integrated through residual connections and layer normalisation, retaining original feature information while incorporating global contextual awareness. First, the method described in Section 2.3.2 generates the node-level multi-head self-attention output feature matrix H_{attn} from the input feature H_{fused} . Subsequently, H_{fused} and H_{attn} undergo residual connection before being fed into a normalisation layer to yield the final output feature matrix of the two-stage attention module. The mathematical formulation is as follows:

$$H_{attn} = \text{MultiHeadAttention}(H_{fused}, H_{fused}, H_{fused}) \quad (18)$$

$$H_{out} = \text{LayerNorm}(H_{fused} + H_{attn}) \quad (19)$$

Scale-wise attention dynamically adjusts fusion strategies according to the specific structure of the input graph. This mechanism treats features from each layer as independent views, allocating fusion weights by analysing their contribution. Based on the statistical properties and task relevance of features across layers, a learnable transformation network maps features to importance scores. These are subsequently normalised via softmax to form a probability distribution (Section 2.3.2 for details).

3. Experimental preparation

3.1. Experimental dataset

To comprehensively evaluate the effectiveness and generalisation capability of the proposed model in the task of road defect detection, this paper conducts experimental validation using two representative road defect datasets: the private dataset RDD.v2i and the public dataset RDD2022.

- (1) RDD.v2i. This dataset consists of real road images photographed by Dian et al. in various weather and lighting conditions. It mainly includes three types of defects: potholes, conventional cracks, and alligator cracking. This dataset reliably retains the real-world environment of road surface defects under different conditions, and also includes disruptive background elements such as pedestrians. In experiments conducted on this dataset, a total of 300 images is randomly selected for training, validation, and testing purposes.
- (2) RDD2022. This dataset is a large-scale public dataset [31] consisting of four defect types: transverse cracks, longitudinal cracks,

alligator cracks, and potholes. It is useful for evaluating a model's generalisation performance in multi-class classification tasks. In experiments conducted on this dataset, a total of 300 images is randomly selected for training, validation, and testing purposes.

To standardise the input format, all images were unified to 640×640 pixels prior before the experiment. Each dataset was partitioned into training, validation, and test sets in a 7:1.5:1.5 ratio [32].

3.2. Implementation details

The proposed MSKA-GNN consists of four key modules: the input projection module, the multi-scale feature extraction module, the dual attention fusion module, and the classification output module. The details of each module's structural configuration are provided in Table 1. The input dimension for the Input Projection Module is set to 512. For the output dimension, we compare 128, 256, and 512. Among these, 128 dimensions result in slow training convergence and underfitting, 512 dimensions exhibit overfitting, while 256 dimensions demonstrate optimal performance on the small dataset of 600 samples. For the multi-scale feature extraction module, we test four network architectures from 3 to 6 layers. The 4-layer architecture achieve the highest accuracy, and adding further layers doesn't improve performance. For KAN parameters, smaller grid 3 produced insufficiently resolve basis functions, while larger grids don't enhance performance. Second-order splines exhibit poor continuity, while fourth-order splines incur excessive computational costs. For the attention fusion module, we compare 32, 64, and

Table 2

Results of sensitivity analysis for diffusion intensity, fractal fusion ratio and CLIP similarity threshold.

Dataset	Parameter	Value	CLIP	SSIM	LPIS	Accuracy
RDD.v2i	Diffusion intensity	0.2	0.9809	0.94	0.04	0.5326
		0.3	0.9556	0.9	0.11	0.605
		0.5	0.85	0.88	0.13	0.61
		0.6	0.7684	0.81	0.18	0.5769
		0.06	0.833	0.86	0.15	0.651
	Fractal fusion ratio	0.08	0.8196	0.84	0.17	0.657
		0.15	0.75	0.8	0.19	0.66
		0.16	0.7392	0.76	0.24	0.652
		0.2	0.986	0.95	0.04	0.5405
		0.3	0.963	0.91	0.1	0.612
RDD2022	Diffusion intensity	0.5	0.87	0.89	0.11	0.616
		0.6	0.795	0.82	0.17	0.5702
		0.06	0.8549	0.87	0.15	0.655
		0.08	0.8336	0.85	0.17	0.668
	Fractal fusion ratio	0.15	0.76	0.81	0.19	0.67
		0.16	0.7422	0.76	0.25	0.662

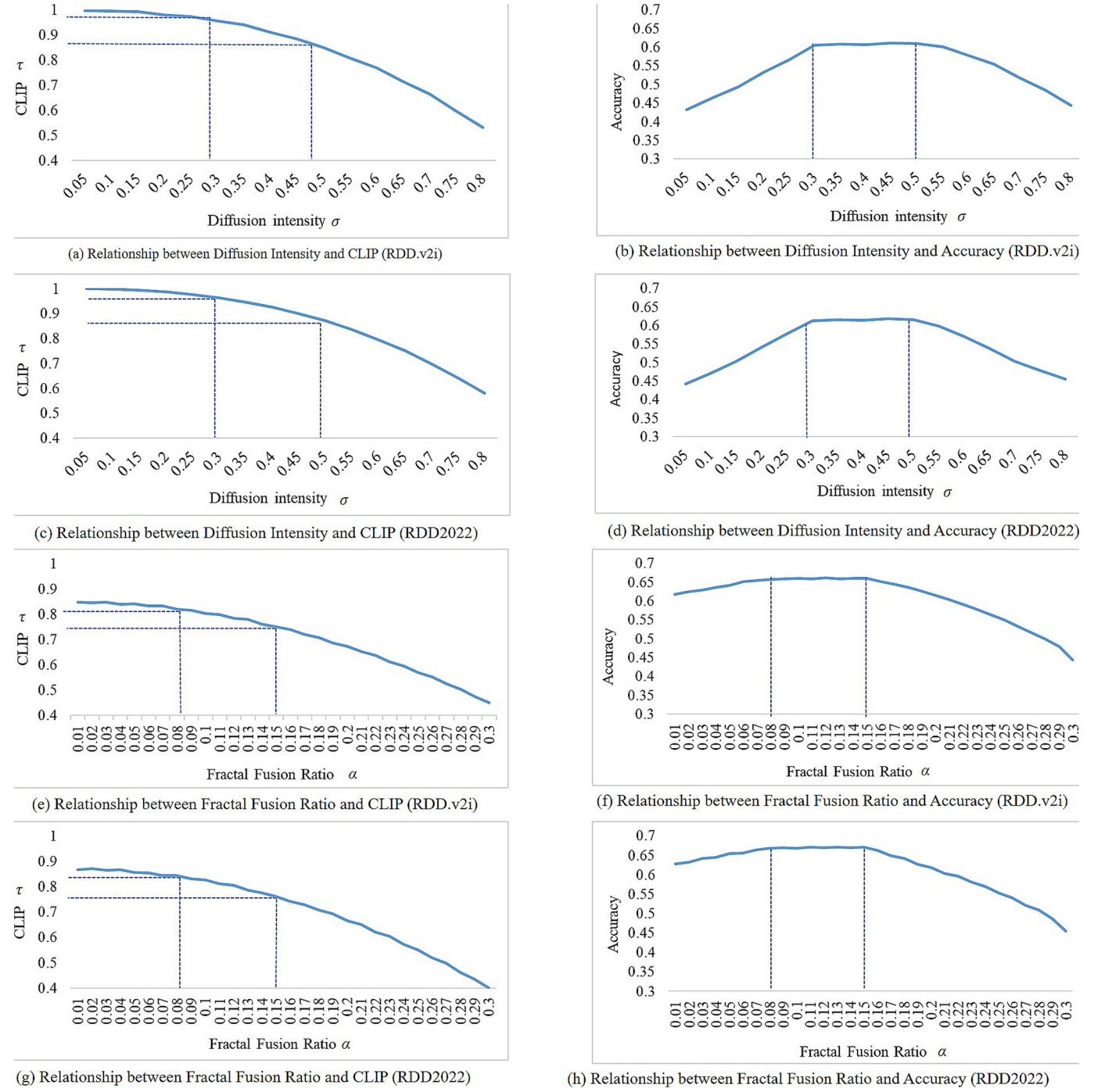


Fig. 5. Results of Sensitivity Analysis for diffusion intensity, fractal fusion ratio and CLIP similarity threshold.

Table 3

Results of Ablation Study for DFH-enhancement.

Dataset	Model	SSIM	LPIS	Accuracy	Precision	Recall	F1-Score
RDD.v2i	X	1	0	0.411	0.417	0.444	0.43
	A	0.95	0.11	0.611	0.6	0.667	0.632
	B	0.91	0.17	0.656	0.64	0.711	0.674
	C	0.9	0.2	0.678	0.654	0.756	0.701
RDD2022	X	0.94	0	0.422	0.419	0.4	0.409
	A	0.91	0.12	0.622	0.608	0.689	0.646
	B	0.89	0.16	0.671	0.647	0.733	0.687
	C	0.88	0.19	0.689	0.667	0.756	0.708

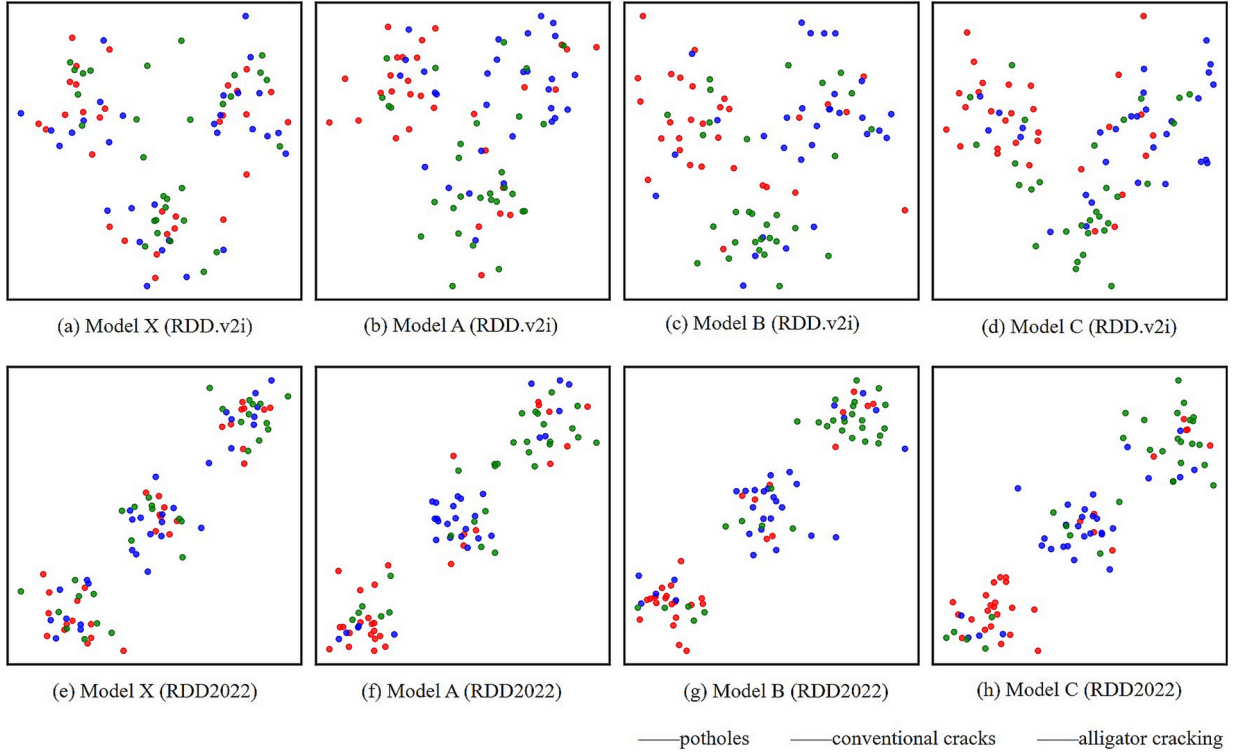


Fig. 6. Results of Ablation Study for DFH-enhancement.

Table 4

Results of Partitioned Set Study for DFH-enhancement.

Dataset	Model	Subset	Accuracy	Precision	Recall	F1-Score
RDD.v2i	X	Low light	0.467	0.467	0.467	0.467
		Rain	0.4	0.414	0.407	0.4
		Complex	0.367	0.379	0.474	0.367
	A	Low light	0.667	0.667	0.667	0.667
		Rain	0.6	0.621	0.6	0.61
		Complex	0.567	0.586	0.567	0.576
	B	Low light	0.7	0.7	0.7	0.7
		Rain	0.667	0.667	0.667	0.667
		Complex	0.6	0.621	0.6	0.61
	C	Low light	0.733	0.733	0.733	0.733
		Rain	0.667	0.667	0.667	0.667
		Complex	0.633	0.655	0.633	0.644
RDD2022	X	Low light	0.467	0.467	0.467	0.467
		Rain	0.433	0.448	0.433	0.44
		Complex	0.367	0.393	0.367	0.379
	A	Low light	0.667	0.667	0.667	0.667
		Rain	0.633	0.655	0.633	0.644
		Complex	0.567	0.607	0.567	0.586
	B	Low light	0.7	0.724	0.7	0.712
		Rain	0.667	0.667	0.667	0.667
		Complex	0.633	0.613	0.633	0.623
	C	Low light	0.733	0.733	0.733	0.733
		Rain	0.7	0.7	0.7	0.7
		Complex	0.633	0.679	0.633	0.655

128 dimensions for scale attention. The 64-dimension configuration prove computationally efficient while adequately capturing scale dependencies. For node-level attention, we find that 4 heads inadequately model global dependencies, while 16 heads exhibit limited representational capacity.

3.3. Evaluation metric

This paper selects classification metrics and clustering metrics to systematically evaluate model performance. These metrics are widely applied in the field of image recognition and can fully assess the

performance of the MSKA-GNN in the task of road defect detection.

In the multi-classification task involving three defect types (alligator cracks, conventional cracks, and pits), we decompose it into three binary sub-tasks based on the One vs Rest strategy: Sub-task 1: Alligator cracks as positive class, all others as negative class; Sub-task 2: Conventional cracks as positive class, all other cracks as negative class; Sub-task 3: Pits as positive class, all others as negative class. For these three sub-tasks, we define the following: - True Positive (TP): Samples where the ground truth label is the target defect and the model prediction also indicates that defect. - True Negative (TN): Samples where the ground truth label is a non-target defect and the model prediction also indicates a non-target defect. - False Positive (FP): Samples where the ground truth label is a non-target defect but the model incorrectly predicts it as a target defect. The number of samples with true labels as target defects but misclassified as non-target defects by the model is recorded as False Negatives (FN). The formulas for Accuracy, Precision, Recall, and F1-Score are as follows:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (19)$$

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (20)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (21)$$

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (22)$$

The clustering metric focuses on the structural consistency of defect features. Its mapping process first extracts high-dimensional node embedding vectors from 600 pavement defect images encoded via the MSKA-GNN model. Subsequently, the t-Distributed Stochastic Neighbour Embedding (t-SNE) algorithm is employed to reduce these high-dimensional vectors to a 2D space. Colour differentiation indicates defect types, with the horizontal axis (t-SNE1) and vertical axis (t-SNE2) representing the two feature dimensions after dimensionality reduction.

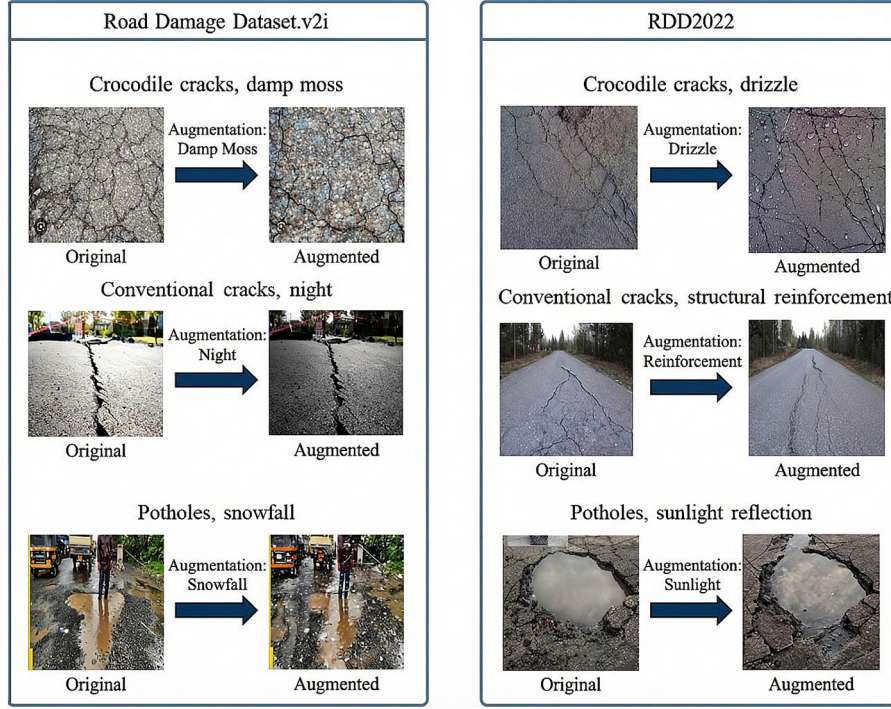


Fig. 7. Data augmentation examples.

Table 5

Test Results of Different convolutional layers.

Dataset	Layers	Accuracy	Precision	Recall	F1-score
RDD.v2i	2	0.71	0.62	0.58	0.599
	3	0.755	0.75	0.75	0.75
	4	0.811	0.824	0.822	0.823
	5	0.765	0.77	0.76	0.765
	2	0.7	0.6	0.56	0.579
RDD2022	3	0.74	0.735	0.735	0.735
	4	0.788	0.8	0.798	0.798
	5	0.75	0.755	0.745	0.75

Table 6

Ablation Study Results for MSKA-GNN.

Dataset	Module	Accuracy	Precision	Recall	F1-Score
RDD.v2i	GNN	0.667	0.683	0.678	0.678
	GNN + MSGC	0.644	0.666	0.663	0.663
	GNN + MSFF	0.689	0.702	0.698	0.699
	GNN + MSGC + MSFF1	0.689	0.700	0.689	0.689
	GNN + MSGC + MSFF	0.700	0.708	0.700	0.701
	GNN + MSGC + MSFF + DAM	0.767	0.781	0.778	0.779
	MSKA-GNN	0.811	0.824	0.822	0.823
	GNN	0.678	0.695	0.690	0.691
	GNN + MSGC	0.656	0.675	0.672	0.672
	GNN + MSFF	0.682	0.701	0.698	0.700
RDD2022	GNN + MSGC + MSFF1	0.678	0.691	0.692	0.692
	GNN + MSGC + MSFF	0.685	0.709	0.701	0.708
	GNN + MSGC + MSFF + DAM	0.748	0.762	0.759	0.759
	MSKA-GNN	0.788	0.800	0.798	0.798

4. Experimental process and result analysis

4.1. Study for DFH-enhancement

4.1.1. Sensitivity analysis for DFH-enhancement

In DFH enhancement, the values of diffusion intensity σ , fractal fusion ratio α , and CLIP similarity threshold τ need to be confirmed. Two tests were set on each dataset. The dataset was randomly divided into training, validation, and test sets containing 210, 45, and 45 images. Training and validation sets were augmented using multimodal-guided stable diffusion in first test (for σ), and adding fractal fusion module on this basis in second test (for α and τ). The test set was augmented using the Albumentations library. The performance of the GNN model was evaluated. The structural similarity index measure (SSIM) and the Learned Perceptual Image Patch Similarity (LPIPS) between the original and augmented images were also used to quantify the semantic destruction degree. The sensitivity analysis results are shown in Table 2 and Fig. 5 (The complete results are provided in supplementary material table s1). As for diffusion intensity, images generated within the range of 0.3 to 0.5 enable the model to achieve optimal accuracy, whilst higher diffusion intensities disrupt the semantic content of the images (SSIM lower than 0.8). As for the fractal fusion ratio, images generated within the range of 0.08 to 0.15 result in a further improvement in recognition accuracy, whilst higher fusion ratios reduce accuracy. Within this range, the lowest CLIP similarity is 0.75, implying that lower similarity no longer contributes to any further improvement in accuracy.

4.1.2. Ablation study for DFH-enhancement

Based on the results of the sensitivity analysis, the diffusion intensity was set to 0.3–0.5, the fractal fusion ratio was set to 0.08–0.15, and the CLIP similarity threshold was set to 0.75. Ablation study was then conducted. Four sets of models were set to verify the role played by each module in the enhancement process:

- (1) Model X: No enhancement;
- (2) Model A: Diffusion;
- (3) Model B: Diffusion + fractal fusion;
- (4) Model C: Diffusion + fractal fusion + masked stitching (DFH-

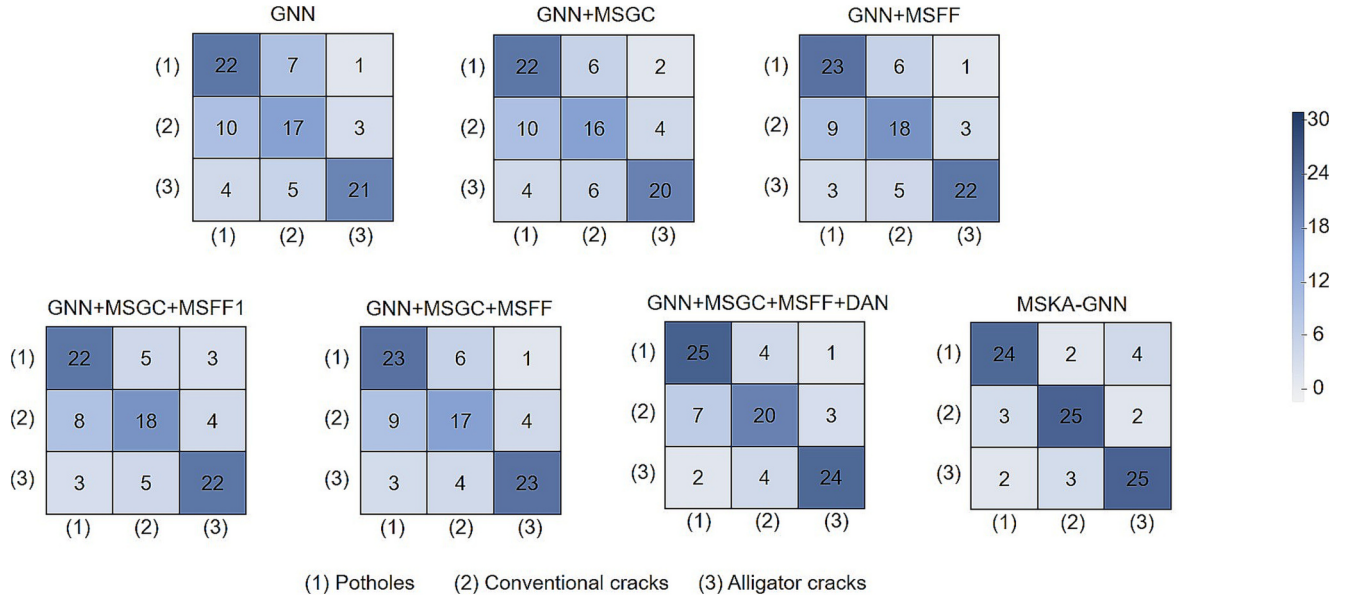


Fig. 8. Confusion Matrix of Ablation Study for RDD.v2i.

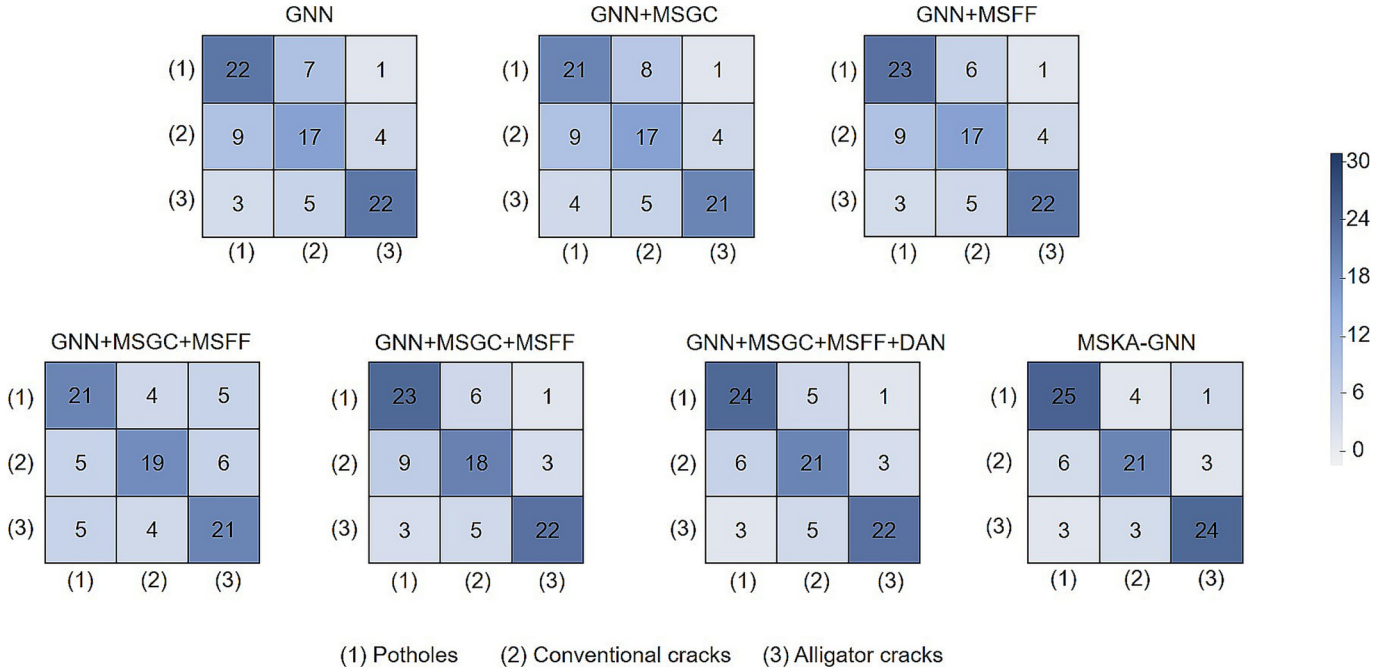


Fig. 9. Confusion Matrix of Ablation Study for RDD2022.

enhancement)

Performance evaluation is divided into two parts. The first part assesses the extent to which the enhanced images distort the semantics of the original images, specifically by measuring structural similarity between image edges (SSIM), Learned Perceptual Image Patch Similarity (LPIPS) and T-distributed stochastic neighbour embedding (t-SNE). The second part assesses whether the enhanced images help improve the detection accuracy of defects under adverse weather conditions. At the same time, the test set were divided into three subsets, low light, rain and complex backgrounds, to verify whether the model was effective in each of these adverse conditions. The results are shown in Table 3, Fig. 6 and Table 4. The results show that, as modules were gradually added, the SSIM remained above 0.85 whilst the LPIPS remained below 0.2. Results from the test set indicate that each module in DFH-enhancement

made a contribution, significantly enhancing the model's ability to recognise defects under harsh weather conditions.

4.1.3. Data augmentation and splitting

To validate MSKA-GNN, we designed two experiments based on two datasets. For each experiment, the 300 images in the corresponding dataset were first randomly divided into training, validation, and test sets at a ratio of 7:1.5:1.5. Data augmentation, including random transformations, horizontal/vertical flipping, and brightness adjustment, was then performed independently within each subset, resulting in 420 training images, 90 validation images, and 90 test images. By applying augmentation only after dataset partitioning and exclusively within each subset, the risk of data leakage between the training, validation, and test sets was eliminated. Examples of the augmented images

Table 7
Optimal Parameters for the Comparison Models.

Model	Optimal parameters
GAL	Contrast learning temperature coefficient (0.1)
	Contrast loss weight (0.5)
GIN	Initial learning rate (0.0001)
	Dropout rate (0.5)
GCN	Initial learning rate (0.001)
	Graph pooling rate (0.5)
Graph-Mamba	Initial learning rate (0.001)
	Loss weight (0.5)
GGMNet	Initial learning rate (0.0002)
	Loss weight (0.4)
HG-DCN	Initial learning rate (0.0005)
	Contrast learning temperature (0.07)
	Initial learning rate (0.00003)

Table 8
Comparison of Results Using the Fixed-Proportion Allocation Method.

Dataset	Net	Accuracy	Precision	Recall	F1-Score
Road Damage Dataset.v2i	MSKA-GNN	0.811	0.824	0.822	0.823
	GNN	0.667	0.683	0.678	0.678
	GAL	0.718	0.732	0.725	0.726
	GIN	0.715	0.730	0.722	0.723
	GCN	0.722	0.735	0.728	0.729
	Graph-Mamba	0.782	0.798	0.825	0.810
	GGMNet	0.768	0.785	0.778	0.780
	HG-DCN	0.792	0.822	0.788	0.803
	MSKA-GNN	0.788	0.800	0.798	0.798
	GNN	0.678	0.695	0.690	0.691
RDD2022	GAL	0.708	0.722	0.715	0.716
	GIN	0.705	0.720	0.712	0.713
	GCN	0.712	0.725	0.718	0.719
	Graph-Mamba	0.765	0.782	0.785	0.782
	GGMNet	0.752	0.768	0.798	0.780
	HG-DCN	0.775	0.805	0.772	0.786

are shown in Fig. 7.

4.2. Ablation study for MSKA-GNN

4.2.1. Influence of convolutional layers on model performance

This section discusses the impact of selecting the number of convolutional layers on model performance to validate the rationality of four layer convolution. All other model architectures and parameter settings were kept unchanged, while the number of convolutional layers was set to 2, 3, 4, and 5, respectively. The classification performance of each model was then evaluated on the test set, and the results are shown in Table 5. In both datasets, the model performance improved as the number of convolutional layers increased from two to four, indicating that a deeper convolutional feature extractor generated more informative features for graph construction. However, when the depth was further increased to five layers, the performance declined, suggesting that the additional layer introduced excessive redundant features that adversely affected the model's discriminative capability. These results indicate that a four-layer convolutional architecture provides the best trade-off between feature representation and redundancy, yielding the optimal performance.

4.2.2. Ablation study for MSKA-GNN

To ensure the effectiveness of each key module of MSKA-GNN, ablation study is designed in this paper. We use a classical GNN as the baseline model, then step-by-step add the MSGC, MSFF, DAM, and KAN modules, comparing the performance of models with different configurations.

(1) GNN;

(2) GNN + MSGC;

(3) GNN + MSGC + MSFF1 (without scale-level attention);

(4) GNN + MSGC + MSFF (with scale-level attention);

(5) GNN + MSGC + MSFF + DAM;

(6) GNN + MSGC + MSFF + DAM + KAN (MSKA-GNN).

Results are shown in Table 6, Fig. 8 and Fig. 9. In two datasets, MSKA-GNN achieves superior performance across all evaluation metrics (Accuracy reaches 0.811 in RDD.v2i and 0.788 in RDD2022). Compared to the baseline, the introduction of almost every module improves the performance of the model. Introducing the MSGC module alone enriches feature representations but also introduces redundant features. As the baseline GNN is unable to effectively exploit or suppress these redundant features, its classification performance deteriorates. After adopting the MSFF module, all metrics of the model are improved. The MSFF module is further evaluated independently. Although it improves the performance compared with the baseline model, its performance remains inferior to that of the combined MSGC + MSFF configuration. In this setting, the two modules complement each other effectively: the MSGC module generates richer and more informative graph representations, while the MSFF module progressively exploits these representations through multi-scale feature fusion. As a result, their combination fully realises the advantages of both modules and yields the most significant performance improvement.

The results of confusion matrixes reveal that introducing the MSGC module independently led to a decline in recognition accuracy for both crack types. Subsequently, incorporating the MSFF module improve recognition accuracy for both potholes and cracks, further corroborating the synergistic effect between the MSGC and MSFF modules. The subsequent introduction of the DAM and KAN modules result in a sustained increase in the accurate recognition rate for all three defect types. At the same time, we discover that during the training process, as modules are progressively stacked, the training loss gradually decreases, accuracy steadily increases, and the process becomes more stable, effectively enhancing the model's performance.

In summary, the research confirms that the proposed MSKA-GNN and its modules positively influence the classification of road surface defects, proving it is an effective graph neural network model.

4.3. Comparative study for MSKA-GNN

To further validate the superiority of the proposed MSKA-GNN network in the field of road defect recognition, comparative experiments are conducted against several advanced graph neural networks—GAL [12], GIN [33], GCN [34], Graph-Mamba [35], GGMNet [36], and HG-DCN [22] for comparative experiments. To ensure the fairness of the experiments, the datasets (training, validation and test sets), evaluation metrics, random number seeds, optimiser types (i.e. scheduling strategies), batch sizes and number of training iterations were kept identical across all models under comparison. The structural parameters of these models remained unchanged; the settings described in their respective papers were adopted directly. Hyperparameters such as learning rate, weight decay and gradient clipping were fine-tuned to suit our dataset, and the most suitable parameters were selected for each model. The specific settings are shown in Table 7.

The results of the parameterised classification metrics are shown in Table 8, while the clustering results are shown in Fig. 10 and Fig. 11.

For the RDD.v2i, the proposed MSKA-GNN model outperforms current mainstream graph neural network models across all three key metrics: Accuracy, Precision, and F1-Score. Specifically, it achieves improvements of 1.9%, 0.2%, and 1.3% respectively compared to the next best models. Although the proposed MSKA-GNN (0.822) slightly underperforms Graph-Mamba (0.825) in Recall, its performance remains among the top in the network models. Compared to the baseline GNN, MSKA-GNN achieves improvements of 14.4%, 14.1%, 14.4%, and 14.5% in Accuracy, Precision, Recall, and F1-Score respectively. Furthermore, the clustering result (Fig. 8) demonstrates that MSKA-GNN

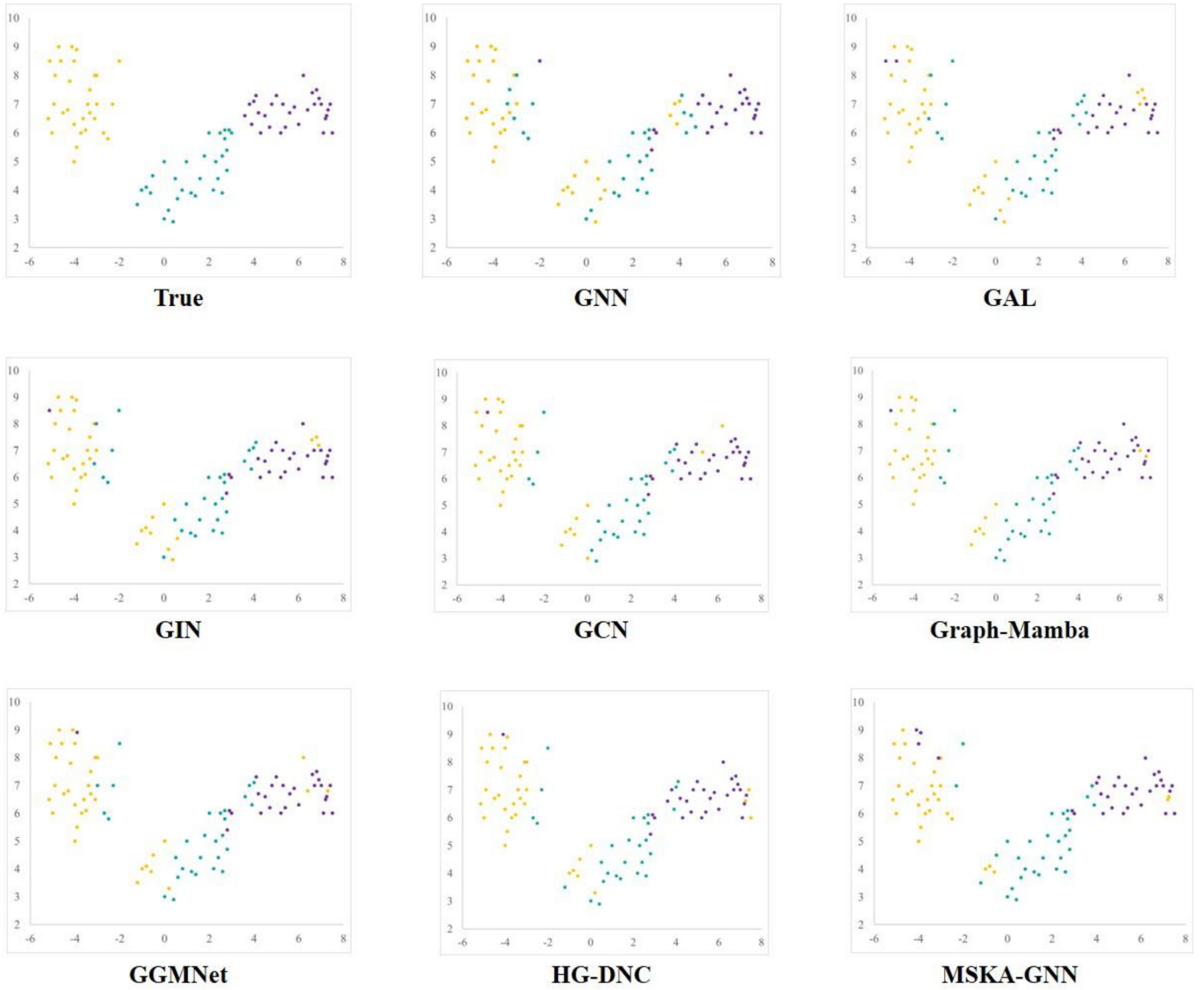


Fig. 10. Clustering Results for the RDD.v2i.

achieves more precise category separation during classification tasks, particularly exhibiting superior performance compared to other network models when handling easily confused defects such as cracking. This indicates that the suboptimal performance of mainstream models on road-snapshot datasets with prominent interference factors stems from two primary causes: firstly, graph data focuses on local node aggregation, struggling to capture the global spatial correlations of road defects; secondly, the absence of targeted feature selection mechanisms prevents precise differentiation between damage and background features, rendering models susceptible to irrelevant information interference. However, MSKA-GNN achieves superior performance by integrating local and global features through its multi-scale module, and adopting attention mechanisms to suppress noise and amplify key features.

On the RDD2022 dataset, similar to results on the RDD.v2i, it demonstrates outstanding performance. It outperforms current mainstream graph neural network models on both Accuracy and F1-Score metrics, achieving respective improvements of 1.3% and 1.2% over the next best model, with the most significant gain observed in accuracy. In terms of Recall, it matches the high-performing GGMNet model at 79.8%. While Precision is marginally lower than the HG-DCN network, it still ranks among the top in the graph neural network domain. Compared to basic

graph neural networks, improvements across metrics reach 11%, 10.5%, 10.8%, and 10.7% respectively. Clustering result (Fig. 9) visually demonstrates MSKA-GNN's superiority in classification tasks, particularly when handling highly similar and complex pavement defects. This indicates that mainstream models' limited performance on standard datasets fundamentally stems from structural mismatches with road damage data characteristics. Traditional GNNs struggle to model the global interdependencies inherent in defects due to their local dependency modelling. Furthermore, existing models lack sufficient capability to extract fine-grained defects, rendering them ineffective at classifying easily confused categories. MSKA-GNN achieves a performance breakthrough through two key optimisation mechanisms: its multi-scale feature module balances the correlation between local details and global features, while its dual-layer attention mechanism effectively identifies key features and reduces background interference, resulting in enhanced performance.

Meanwhile, based on the detection results of the three defects, we find that across both datasets, MSKA-GNN achieves the best classification performance for both crack types. In the pothole classification task, it still outperforms the industry benchmark algorithm, maintaining above-average performance.

To verify that the experimental results are genuine, stable and

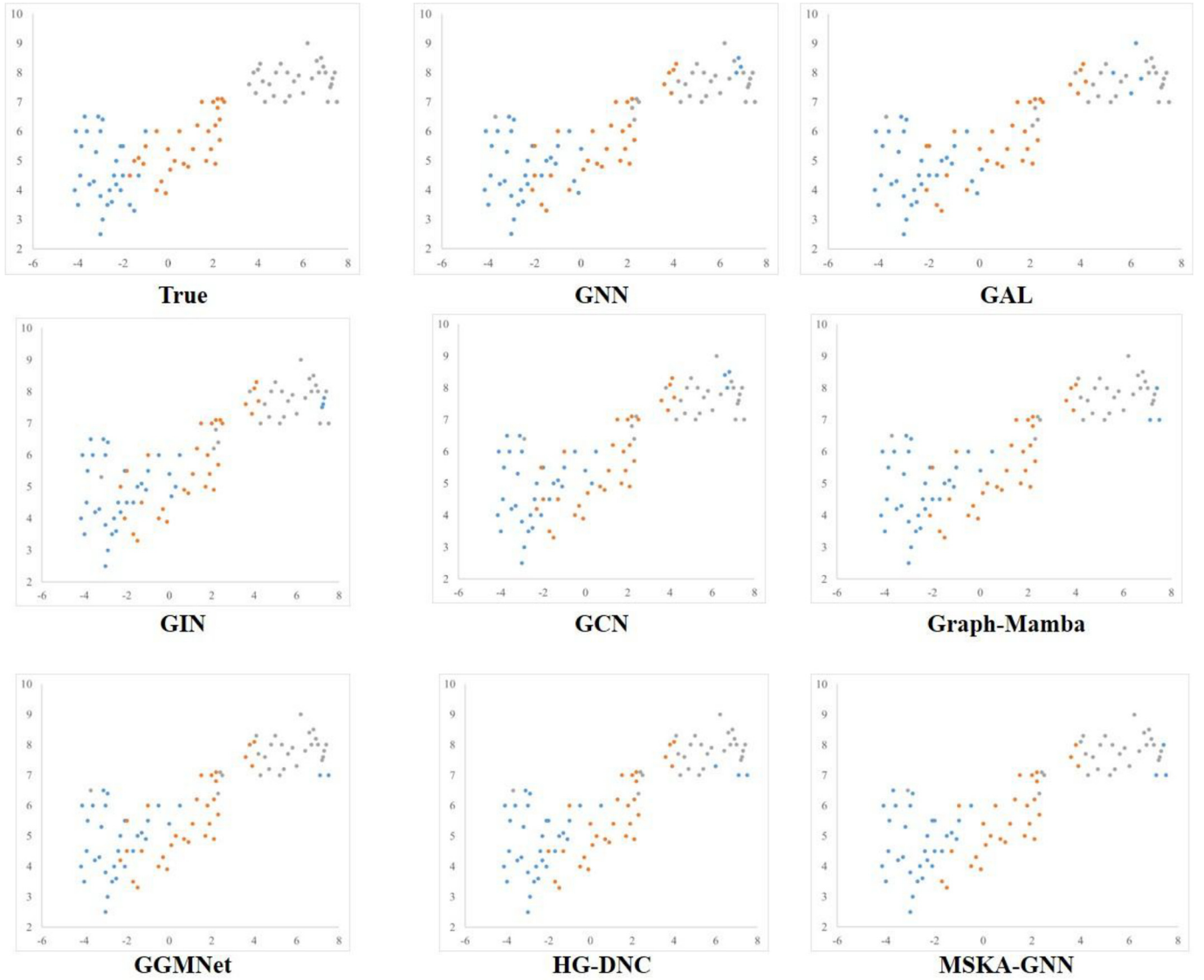


Fig. 11. Clustering Results for the RDD2022.

statistically significant, rather than merely coincidental, we performed 5-fold cross-validation on each set of models. We merged the training and validation sets and divided them into five equal parts; for each iteration, one part was selected as the validation set and the remaining four as the training set, repeating this process five times. The results of the experiment are shown in Table 9.

As can be seen from the results, compared with the results obtained using the holdout method with fixed partitioning, there was a slight decrease in overall accuracy (less than 0.01), which falls within the normal range of variation; the results are therefore reliable. Meanwhile, MSKA-GNN continued to achieve the best mean and standard deviation. Furthermore, the standard deviations of all models remained within a narrow range (less than 0.025), indicating that the methods are insensitive to the training/validation split and that the evaluation results are reliable; the standard deviation of MSKA-GNN (approximately 0.011–0.013) further demonstrates its robustness.

The above experimental results prove that MSKA-GNN not only is a viable approach but also shows distinct advantages in the field of road defect recognition. Even when faced with complex, small-sample datasets, it can achieve high-precision recognition.

5. Conclusions

This paper proposes an end-to-end defect detection framework (MSKA-GNN) designed to address the challenges of data scarcity and poor detection performance in complex road environments. By developing a DFH enhancement strategy, we demonstrate that synthesising semantically consistent defect samples under adverse conditions such as rain can effectively improve the model's robustness.

Furthermore, the experiments demonstrate that the synergistic combination of multi-level graph construction and multi-scale feature extraction can effectively enhance the ability of GNNs to recognise defects at different scales. By incorporating KAN and a dual-attention mechanism into the graph convolution process, we have effectively bridged the gap between shallow local textures and deep global semantics, thereby enhancing the model's non-linear fitting capabilities. Ultimately, MSKA-GNN achieved accuracy rates of 81.1% and 78.8% on the RDD.v2i and RDD2022 datasets respectively, which comprise only 600 images and feature complex, extreme backgrounds, outperforming the mainstream models (GNN, GAL, GIN, GCN, Graph-Mamba and GGMNet), demonstrating its robust perceptual capabilities.

In future research, we will focus on attention-guided adversarial training strategies to improve the detection of highly sparse defects.

Table 9

Result of 5-Fold Cross-Validation.

Dataset	Net	Accuracy	Precision	Recall	F1-Score
RDDv2i	MSKA-GNN	0.802 ± 0.012	0.815 ± 0.013	0.813 ± 0.011	0.814 ± 0.012
	GNN	0.658 ± 0.023	0.674 ± 0.025	0.669 ± 0.024	0.669 ± 0.024
	GAL	0.709 ± 0.019	0.723 ± 0.021	0.716 ± 0.020	0.717 ± 0.020
	GIN	0.706 ± 0.020	0.721 ± 0.022	0.713 ± 0.021	0.714 ± 0.021
	GCN	0.713 ± 0.018	0.726 ± 0.020	0.719 ± 0.019	0.720 ± 0.019
	Graph-Mamba	0.773 ± 0.013	0.789 ± 0.015	0.816 ± 0.012	0.801 ± 0.014
	GGMNet	0.759 ± 0.016	0.776 ± 0.018	0.769 ± 0.017	0.771 ± 0.017
	HG-DCN	0.783 ± 0.013	0.813 ± 0.014	0.779 ± 0.015	0.794 ± 0.013
	MSKA-GNN	0.779 ± 0.012	0.791 ± 0.013	0.789 ± 0.012	0.789 ± 0.012
	GNN	0.669 ± 0.024	0.686 ± 0.026	0.681 ± 0.025	0.682 ± 0.025
	GAL	0.699 ± 0.020	0.713 ± 0.022	0.706 ± 0.021	0.707 ± 0.021
	GIN	0.696 ± 0.021	0.711 ± 0.023	0.703 ± 0.022	0.704 ± 0.022
	GCN	0.703 ± 0.019	0.716 ± 0.021	0.709 ± 0.020	0.710 ± 0.020
	Graph-Mamba	0.756 ± 0.014	0.773 ± 0.015	0.776 ± 0.013	0.773 ± 0.014
RDD2022	GGMNet	0.743 ± 0.016	0.759 ± 0.018	0.789 ± 0.015	0.771 ± 0.016
	HG-DCN	0.766 ± 0.013	0.796 ± 0.014	0.763 ± 0.015	0.777 ± 0.013

Additionally, we will explore lightweight network architectures via knowledge distillation to reduce computational overhead for real-time edge deployment on smart vehicle chassis systems.

CRedit authorship contribution statement

Muwang Cao: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Yang Xing:** Writing – review & editing, Supervision, Project administration.

Declaration of competing 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.

Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.imavis.2026.106127>.

Data availability

Relevant content links are as follows: <https://github.com/maverickgreen856-collab/Few-shot-Road-Surface-Defect-Detection-Using-Enhanced-Mul-ti-level-Multi-scale-Graph-Neural-Network.git>

References

- [1] S. Shao, H. Zhou, H. Liu, Distributed model predictive control and implementation for vehicle active suspensions, IFAC-PapersOnLine 51 (2018) 961–966, <https://doi.org/10.1016/j.ifacol.2018.10.133>.
- [2] C. Wang, Mechanism-data-driven control strategy for active suspension systems: integrating deep reinforcement learning with differential geometry to enhance

- vehicle ride comfort, Adv. Eng. Inform. 65 (2025) 103326, <https://doi.org/10.1016/j.aei.2025.103326>.
- [3] V. Mazzilli, S. De Pinto, L. Pascali, M. Contrino, F. Bottiglione, G. Mantriota, P. Gruber, A. Sornioti, Integrated chassis control: classification, analysis and future trends, Annu. Rev. Control. 51 (2021) 172–205, <https://doi.org/10.1016/j.arcontrol.2021.03.008>.
- [4] H. Zhu, T. Ding, M. Yang, Y. Wu, T. Du, Research on robust fault-tolerant control of the controllable suspension based on knowledge-data fusion driven, Sci. Rep. 13 (2023) 22723, <https://doi.org/10.1038/s41598-023-49839-6>.
- [5] N. Senel, K. Kefferpütz, K. Doycheva, G. Elger, Multi-sensor data fusion for real-time multi-object tracking, Processes 11 (2023) 501, <https://doi.org/10.3390/pr11020501>.
- [6] K.A. Sattar, M. Abdel-Nasser, S. El Ferik, A.E. Taha, Detecting road defects and hazards in metropolitan environments using optimized deep learning techniques, Transp. Res. Procedia 84 (2025) 528–533, <https://doi.org/10.1016/j.trpro.2025.03.105>.
- [7] J. Han, J. Liu, H. Chang, The key technologies of road elevation detection based on sensor fusion, Sensors 22 (2022) 5756, <https://doi.org/10.3390/s22155756>.
- [8] S. Liu, T. Cao, L. Du, Y. Zhu, LU-mamba: LiDAR upsampling via bidirectional state space modeling on range images, Neural Netw. 197 (2026) 108480, <https://doi.org/10.1016/j.neunet.2025.108480>.
- [9] J. Fan, W. Song, J. Zhang, S. Sun, G. Jia, G. Jin, PAN: improved PointNet++ for pavement crack information extraction, Electronics 13 (2024) 3340, <https://doi.org/10.3390/electronics13163340>.
- [10] X. Huang, Z. Xu, H. Wu, J. Wang, Q. Xia, Y. Xia, et al., L4DR: LiDAR-4DRadar fusion for weather-robust 3D object detection, in: Proc. AAAI Conf. Artif. Intell., 2025, pp. 3806–3814, 39 (4).
- [11] L. Fan, D. Wang, J. Wang, Y. Li, Y. Cao, Y. Liu, et al., Pavement defect detection with deep learning: a comprehensive survey, IEEE Trans. Intell. Veh. 9 (2024) 4292–4311, <https://doi.org/10.1109/TIV.2023.3326136>.
- [12] J. Wu, J. Ma, H. Xi, J. Li, J. Zhu, Multi-scale graph harmonies: unleashing U-net's potential for medical image segmentation through contrastive learning, Neural Netw. 182 (2025) 106914, <https://doi.org/10.1016/j.neunet.2024.106914>.
- [13] X. Hua, Z. Du, H. Yu, J. Ma, F. Zheng, C. Zhang, Q. Lu, H. Zhao, A lightweight multi-scale multi-angle dynamic interactive transformer-CNN fusion model for 3D medical image segmentation, Neurocomputing 608 (2024) 128417, <https://doi.org/10.1016/j.neucom.2024.128417>.
- [14] K. Li, J. Yang, S. Ma, B. Wang, S. Wang, Y. Tian, Z. Qi, Rethinking lightweight convolutional neural networks for efficient and high-quality pavement crack detection, IEEE Trans. Intell. Transp. Syst. 25 (2024) 237–250, <https://doi.org/10.1109/TITS.2023.3325954>.
- [15] J. Chen, Y. Wen, Y.A. Nanekharan, D. Zhang, A. Zeb, Multiscale attention networks for pavement defect detection, IEEE Trans. Instrum. Meas. 72 (2023) 1–12, <https://doi.org/10.1109/TIM.2023.3254204>.
- [16] S. Yun, M. Jeong, S. Yoo, S. Lee, S.S. Yi, R. Kim, J. Kang, H.J. Kim, Graph transformer networks: learning meta-path graphs to improve GNNs, Neural Netw. 153 (2022) 104–119, <https://doi.org/10.1016/j.neunet.2022.05.020>.
- [17] Y. Ding, Z. Zhang, W. Kang, A. Yang, J. Zhao, J. Feng, et al., Adaptive homophily clustering: structure homophily graph learning with adaptive filter for hyperspectral image, IEEE Trans. Geosci. Remote Sens. 63 (2025) 1–13, <https://doi.org/10.1109/TGRS.2025.3556276>.
- [18] Y. Ding, Z. Zhang, A. Yang, Y. Cai, X. Xiao, D. Hong, et al., SLCGC: a lightweight self-supervised low-pass contrastive graph clustering network for hyperspectral images, IEEE Trans. Multimed. 27 (2025) 8251–8262, <https://doi.org/10.1109/TMM.2025.3604954>.
- [19] Y. Ding, Z. Zhang, X. Zhao, D. Hong, W. Li, W. Cai, et al., AF2GNN: graph convolution with adaptive filters and aggregator fusion for hyperspectral image classification, Inf. Sci. 602 (2022) 201–219, <https://doi.org/10.1016/j.ins.2022.04.006>.
- [20] Y. Ding, Z. Zhang, H. Hu, F. He, S. Cheng, Y. Zhang, Multi-receptive field: An adaptive path aggregation graph neural framework for hyperspectral image classification, in: Y. Ding, Z. Zhang, H. Hu, F. He, S. Cheng, Y. Zhang (Eds.), Graph Neural Network for Feature Extraction and Classification of Hyperspectral Remote Sensing Images, Springer Nature Singapore, Singapore, 2024, pp. 69–98.
- [21] Z.C. Ma, X.M. Liu, H.Q. Xia, W.Q. Wang, J.Z. Wang, H.T. Shen, Pavement distress situation prediction method based on graph neural network, J. Zhejiang Univ. (Eng. Sci.) 58 (2024) 2596–2608 (in Chinese).
- [22] K.I. Rashid, C. Yang, ViT-CAPS: vision transformer with contrastive adaptive prompt segmentation, Neurocomputing 625 (2025) 129578, <https://doi.org/10.1016/j.neucom.2025.129578>.
- [23] B. Zhang, Y. Zhang, MSCViT: a small-size ViT architecture with multi-scale self-attention mechanism for tiny datasets, Neural Netw. 188 (2025) 107499, <https://doi.org/10.1016/j.neunet.2025.107499>.
- [24] T. Zhang, Z. Liu, B. Cui, X. Gu, Y. Lu, Transformer-CNN hybrid framework for pavement pothole segmentation, Sensors 25 (2025) 6756, <https://doi.org/10.3390/s25126756>.
- [25] X. Wang, H. Gao, Z. Jia, J. Zhao, A road defect detection algorithm incorporating partially transformer and multiple aggregate trail attention mechanisms, Meas. Sci. Technol. 36 (2025) 026003, <https://doi.org/10.1088/1361-6501/adb0f3>.
- [26] J. Cong, X. Ran, C. Qin, Deep homography-based image stitching with enhanced small object detection, Sci. Rep. 16 (2026) 10127, <https://doi.org/10.1038/s41598-026-58267-7>.
- [27] C. Qin, X. Ran, Efficient unsupervised image stitching using attention mechanism with deep homography estimation, Comput. Mater. Contin. 79 (1) (2024) 1319–1334, <https://doi.org/10.32604/cmc.2024.048850>.

- [28] C. Qin, X. Ran, D. Zhang, Unsupervised image stitching based on generative adversarial networks and feature frequency awareness algorithm, *Appl. Soft Comput.* 183 (2025) 113466, <https://doi.org/10.1016/j.asoc.2025.113466>.
- [29] C. Qian, W. Li, Y. Guo, G. Markkula, WeatherEdit: Controllable Weather Editing With 4D Gaussian Field, arXiv:2505.20471, 2025, <https://doi.org/10.48550/arXiv.2505.20471>.
- [30] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljacić, et al., KAN: Kolmogorov–Arnold networks, in: *Proc. 13th Int. Conf. Learn. Represent. (ICLR)*, 2025.
- [31] D. Arya, H. Maeda, S.K. Ghosh, D. Toshniwal, Y. Sekimoto, RDD2022: a multi-national image dataset for automatic road damage detection, *Geosci. Data J.* 11 (2024) 846–862, <https://doi.org/10.1002/gdj3.230>.
- [32] H. Huang, M.Y. Ma, S.L. Bai, L. Yang, Y.H. Liu, Automatic crack defect detection via multiscale feature aggregation and adaptive fusion, *Autom. Constr.* 170 (2025) 105934, <https://doi.org/10.1016/j.autcon.2025.105934>.
- [33] Y. Djenouri, A. Belhadi, E.H. Houssein, G. Srivastava, J.C.-W. Lin, Intelligent graph convolutional neural network for road crack detection, *IEEE Trans. Intell. Transp. Syst.* 24 (2023) 8475–8482, <https://doi.org/10.1109/TITS.2022.3224248>.
- [34] O. Sidorov, J.Y. Hardeberg, Craquelure as a graph: Application of image processing and graph neural networks to the description of fracture patterns, in: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 1429–1436.
- [35] Z. Tian, X. Shao, Y. Bai, Graph-MambaRoadDet: a symmetry-aware dynamic graph framework for road damage detection, *Symmetry* 17 (2025) 1654, <https://doi.org/10.3390/sym17091654>.
- [36] Y. Wang, Z. He, X. Zeng, J. Zeng, Z. Cen, L. Qiu, et al., GGMNet: pavement-crack detection based on global context awareness and multi-scale fusion, *Remote Sens.* 16 (2024) 1797, <https://doi.org/10.3390/rs16091797>.



DSPACE

<https://dspace.org/>

Few-shot road surface defect detection using enhanced multi-level multi-scale graph neural network

Cao, Muwang; Xing, Yang

2026-10

Elsevier

<https://doi.org/10.1016/j.imavis.2026.106127>

Cao M, Xing Y. (2026) Few-shot road surface defect detection using enhanced multi-level multi-scale graph neural network. Image and Vision Computing, Volume 174, October 2026, Article number 106127