https://doi.org/10.37965/jdmd.2026.1675

Physics-guided graph Transformer for composite pipeline damage detection

Xuan Li[a], Yue Hu[a*], Pugen Zhang[b], Jinsha Xu[b], Yi Chen[b], Haibin Miao[c], Xinguang Ren[c], and Fucai Li[d]

[a] School of Mechanical and Power Engineering, East China University of Science and Technology, Shanghai, China

[b] Shanghai Institute of Special Equipment Inspection and Technical Research Co., Ltd, Shanghai, China

[c] PETROCHINA DAGANG PETROCHEMICAL COMPANY, Tianjin, China

[d] State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai, China

Received 15 July, 2026; Revised 24 August,2026; Accepted 29 October, 2026; Published online 3 November , 2026.

https://doi.org/10.37965/jdmd.2026.1675

Abstract: Composite pipelines are widely used in oil and gas transportation owing to their high specific strength and corrosion resistance. However, guided-wave-based damage localization remains challenging due to multimodal wave propagation, noise interference, complex propagation-path relationships, and limited labeled damage locations. To address these challenges, this study proposes a physics-guided graph Transformer (PGGT) framework for composite pipeline damage localization.

In the proposed method, guided-wave signals are first preprocessed, and actuator–receiver propagation paths are represented as graph nodes. A physics-guided radius graph is constructed by incorporating the spatial configuration of piezoelectric transducers and guided-wave propagation characteristics, enabling multi-channel guided-wave signals to be mapped into a physically meaningful non-Euclidean graph structure. A convolutional module is employed to extract local temporal features from guided-wave responses, a graph attention module is utilized to model topology relationships among signal-path nodes, and a Transformer module is introduced to capture long-range global dependencies among propagation paths.

Comparative experiments are conducted under four training–testing split strategies with unseen damage locations. The proposed PGGT is compared with VGG16, ResNet18, ChebyNet, GCN, GAT, Transformer, and PGGDL in terms of mean absolute percentage error (MAPE), mean absolute error (MAE), and root mean square error (RMSE). The results demonstrate that PGGT achieves superior localization accuracy and generalization capability under limited labeled data. In the most challenging case, PGGT achieves a MAPE of 0.035 (that is, 3.5%), a MAE of 5.38 mm, and an RMSE of 6.23 mm. The proposed method provides an effective physics-informed learning framework for guided-wave-based damage localization of composite pipelines.

Index Terms: Composite pipeline, ultrasonic guided waves, graph neural network, Transformer; physical rules, damage detection

https://doi.org/10.37965/jdmd.2026.1675

1.Introduction

Composite pipelines are increasingly used in oil and gas transportation and other engineering applications because of their high specific strength, corrosion resistance, and lightweight characteristics [1]. However, long-term exposure to complex service environments and accidental impacts may induce structural damage [2][3], threatening the safe operation of pipeline systems [5]. Therefore, reliable structural health monitoring [4] and nondestructive testing techniques are required for early damage identification. Ultrasonic guided waves are particularly attractive for pipeline inspection because of their long propagation distance and large inspection coverage [6].

In guided-wave-based damage detection, elliptical imaging [7][8] and time-of-flight-based [9] methods are widely investigated. Nevertheless, these approaches often rely on reference signals and accurate wave velocity estimation. In practical measurements, noise, boundary reflections, and multimodal wave superposition considerably increase the difficulty of signal interpretation [10][11]. This issue becomes more pronounced in composite pipelines [12] because anisotropy and structural complexity further complicate guided-wave propagation.

Deep learning methods, especially convolutional neural networks [13][14], show strong feature extraction capability for guided-wave signals [15][16]. However, conventional deep learning models are primarily designed for regular Euclidean data [17], such as images and sequences, and thus have limited ability to explicitly represent spatial dependencies among sensors. For multi-sensor guided-wave inspection, the sensor layout, propagation paths, and inter-sensor relationships are highly relevant to damage detection. Graph neural networks [18] provide a natural framework for modeling such non-Euclidean structured data by treating sensors as nodes and propagation-related connections as edges.

Although graph neural networks can capture local topological dependencies through neighborhood aggregation [19], their information propagation is generally restricted to finite-order neighborhoods. As a result, long-range correlations and global dependencies in guided-wave signals may not be sufficiently represented [20]. The Transformer architecture, based on the self-attention mechanism, provides an effective tool for modeling global dependencies [21]. Motivated by this, this study proposes the physics-guided graph Transformer, termed PGGT, for composite pipeline damage detection and localization. The method first normalizes and filters guided wave data to enhance signal-to-noise ratio and constructs a non-Euclidean graph structure using the radius graph strategy, combining sensor spatial distribution and guided wave propagation mechanism. Subsequently, it integrates node feature extraction, spatial feature extraction, and global feature modeling [23], achieving collaborative learning of local patterns, graph topological relationships, and global dependencies. This enables precise prediction of composite pipeline damage locations even under limited training labels.

The main contributions of this article are as follows.

(1) A physics-guided radius graph construction strategy is proposed for guided-wave-based composite pipeline inspection. By incorporating sensor spatial distribution and guided-wave propagation characteristics, multi-sensor signals are represented as physically consistent non-Euclidean graph data.

(2) A physics-guided graph Transformer framework is developed by integrating convolutional neural networks, graph attention networks, and Transformer modules. The proposed model jointly learns local

https://doi.org/10.37965/jdmd.2026.1675

time-domain features, sensor-topology dependencies, and long-range global correlations.

(3) The proposed PGGT is validated under four training–testing split strategies involving unseen damage locations. Comparative results against CNN-, GNN-, and Transformer-based methods demonstrate its improved localization accuracy and generalization capability under limited labeled data.

The rest of this article is organized as follows: Section 2 presents the structure and implementation details of the PGGT model, Section 3 discusses the experimental setup, comparison methods, and results analysis; and finally, conclusions and prospects are provided.

2.Physics-Guided Graph Transformer Model

This study proposes the PGGT method for ultrasonic-guided-wave-based damage detection and localization in composite pipelines. The model integrates convolutional neural networks, graph attention networks, and Transformer architectures to establish an end-to-end mapping from multi-channel guided-wave signals to the axial damage coordinates. The aim is to enhance the detection accuracy and generalization ability under limited training labels, as illustrated in the overall process shown in Fig. 1.

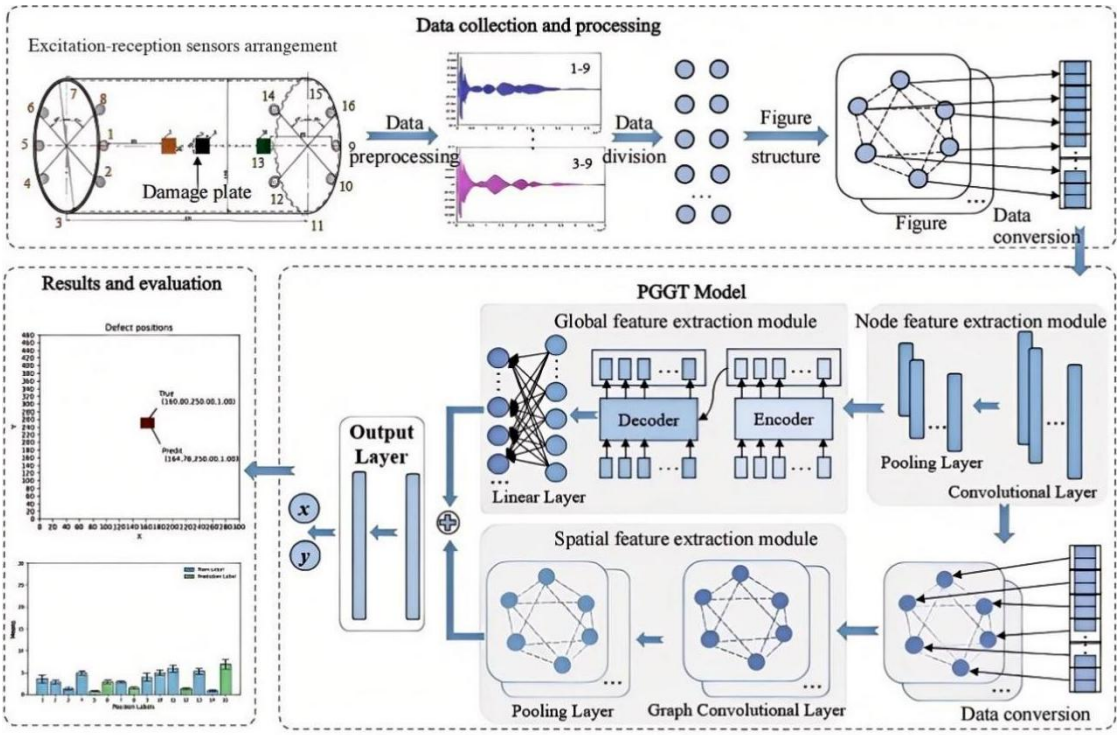


Fig. 1. Flowchart of damage detection based on the PGGT model

The PGGT model comprises four parts: physics-guided graph construction, local feature extraction, topological relationship modeling, and global dependency aggregation. In the graph representation, the graph nodes correspond to the measured pitch-catch signal paths rather than directly to the individual PZT transducers. Since PZT1–PZT8 act as actuators and PZT9–PZT16 as receivers, the 8×8 actuator–receiver combinations produce 64 guided-wave channels; accordingly, each measurement is represented by a graph containing 64 signal-path nodes. Each signal-path node represents a unique ultrasonic guided-wave propagation route between an actuator and a receiver. The corresponding guided-wave waveform contains the combined effects of excitation conditions, propagation characteristics, structural interactions, and damage-induced scattering responses. Therefore,

representing actuator–receiver propagation paths as graph nodes enables the graph structure to preserve the inherent physical information of guided-wave inspection.

Firstly, the collected multi-channel guided-wave data undergoes normalization and filtering preprocessing to enhance the signal-to-noise ratio and optimize data quality. Subsequently, based on the spatial distribution of the PZT transducers and the propagation characteristics of guided waves, a physics-guided radius graph is constructed to represent the signal-path nodes and their physically meaningful connections.

Specifically, the edges between signal-path nodes are constructed according to the physical propagation characteristics of guided waves. The spatial relationship between different actuator–receiver propagation paths is considered to determine the connectivity of the graph. For two signal-path nodes, a connection is established when the distance between their propagation-path indices satisfies the predefined neighborhood constraint $d_{ij} \leq r$, indicating that the corresponding guided-wave propagation paths have meaningful physical interactions. In this way, the graph topology is not generated only from data similarity, but incorporates prior knowledge of guided-wave propagation behavior.

$$A_{ij} = \begin{cases} exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right), d_{ij} \leq r \\ 0, d_{ij} > r \end{cases} \quad (1)$$

where A_{ij} denotes the edge weight between node i and node j, d_{ij} denotes the distance between the propagation-path indices of nodes i and j, which is defined in Eq. (2), r is the radius threshold that defines the neighborhood in this normalized index space, and σ is a scaling parameter controlling the attenuation rate; both r and σ are dimensionless quantities defined in the same normalized index space. Only nodes satisfying $d_{ij} \leq r$ are connected, while the edge weight decreases with increasing propagation-path distance. A larger value of A_{ij} indicates a stronger physical correlation between two signal-path nodes. Compared with purely data-driven graph construction methods, the proposed physics-guided graph incorporates the actual sensing configuration and guided-wave propagation characteristics, thereby improving the physical interpretability of the learned graph structure.

$$d_{ij} = \sqrt{(a_i - a_j)^2 + (b_i - b_j)^2} \quad (2)$$

The distance in Eq. (2) is evaluated in the propagation-path index space. Each signal-path node is indexed by the ordered pair (a_i, b_i), where a_i denotes the actuator index (1-8, corresponding to PZT1-PZT8) and b_i denotes the receiver index (1-8, corresponding to PZT9-PZT16). Since the sixteen transducers are uniformly distributed around the pipe circumference with an angular spacing of 45°, one index step corresponds to 45° of transducer spacing, which is approximately 28.7 mm at the level of the propagation-path midpoint. The radius threshold r and the scaling parameter σ are therefore dimensionless quantities defined in this normalized propagation-path index coordinate system rather than physical lengths; for reference, r = 3.5 and σ = 1.0 correspond to path-midpoint separations of approximately 100 mm and 29 mm, respectively. With r = 3.5, the physics-guided graph contains 64 nodes, 1532 directed edges and an average node degree of 23.9, whereas a fully connected graph would contain 4032 directed edges.

https://doi.org/10.37965/jdmd.2026.1675

In the feature extraction phase, the PGGT model performs collaborative feature learning through three core modules: the node feature extraction module utilizes convolutional layers to capture local time-domain patterns from the guided-wave signals; the spatial feature extraction module employs graph attention networks to model topological dependencies among the signal-path nodes; and the global feature extraction module introduces the Transformer attention mechanism to capture long-range dependencies among the node representations.

For each measurement, the original data contain 64 guided-wave channels with 10,000 sampling points per channel. After preprocessing, 4,000 temporal sampling points from each channel are retained as the model input, resulting in a 64×4000 input matrix. The node feature extraction module independently processes the one-dimensional waveform associated with each signal-path node and transforms it into a 400-dimensional feature vector, producing a 64×400 node-feature matrix. This node-feature matrix is subsequently fed in parallel into the GAT-based spatial feature branch and the Transformer-based global feature branch. The GAT branch generates a 128-dimensional graph-level spatial representation after successive feature transformations and graph-level pooling, while the Transformer branch receives the 64×400 node-feature sequence and generates a 128-dimensional global representation. The two representations are subsequently fused and used for damage-location regression. The detailed architecture of the PGGT model is shown in Fig. 2.

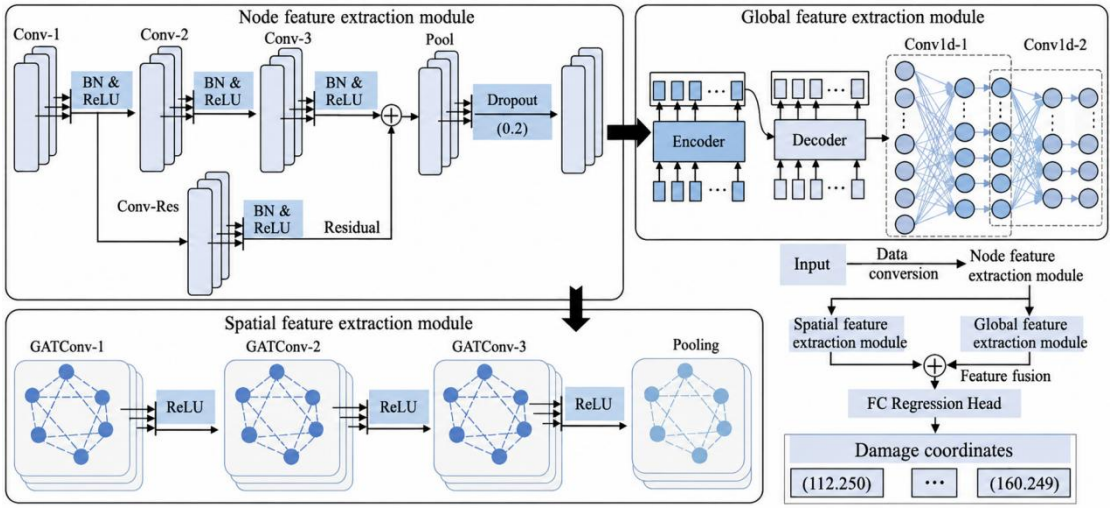


Fig. 2. Schematic of the detailed structure of the PGGT model

The node-feature extraction module processes the one-dimensional temporal signal associated with each graph node independently. For a batch containing B measurements, the input tensor has dimensions B×64×4000, where 64 denotes the number of actuator–receiver signal-path nodes and 4,000 denotes the retained temporal sampling points for each node. Since each signal-path node corresponds to an individual guided-wave time-domain waveform, the node-feature extraction module adopts one-dimensional convolution operations along the temporal dimension. The convolution kernels slide only along the time axis of each waveform to capture local temporal patterns, while no convolution operation is performed across different signal-path nodes. The same convolutional feature extractor is independently applied to each node signal, producing a B×64×400 node-feature tensor. Therefore, each graph node is represented by a 400-dimensional temporal feature vector before being passed to the subsequent GAT and Transformer branches.

Table 1. Structure and Parameter Configuration of Node Feature Extraction Module

Name	Parameter settings	Description
1D convolutional layer 1	Kernel size: 10; Input channels: 1; Output channels: 16; Stride: 10; Padding: 4	Extract initial local features
Batch normalization 1 and activation function	Channel: 16; ReLU	Stabilize the training process, enhance convergence speed, and introduce non-linear features
1D convolutional layer 2	Kernel size: 3; Input channels: 16; Output channels: 32; Stride: 1; Padding: 1	Deep-level feature extraction
Batch Normalization 2 and Activation Function	Channel: 32; ReLU	Stabilize the training process, enhance convergence speed, and introduce non-linear features
1D convolutional layer 3	Filter kernel size: 3; Input channels: 32; Output channels: 64; Stride: 1; Padding: 1	Capture more complex high-order features
Batch normalization 3 and activation function	Channel: 64; ReLU	Stabilize the training process, enhance convergence speed, and introduce non-linear features
Convolutional layer Res	Kernel size: 3; Input channels: 16; Output channels: 64; Stride: 1; Padding: 1	Achieve dimension matching, preparing for the addition of the main path and the residual path.
Batch normalization of the Res and activation functions	Channel: 64; ReLU	Stabilize the training process, enhance convergence speed, and introduce non-linear features
Pooling layer	Pooling Type: Max Pooling; Kernel Size: 2; Stride: 2	Dimension reduction, retain key information, reduce computational load
Dropout layer	Dropout rate: 0.2	Reduce overfitting and improve model generalization

The B×64×400 node-feature tensor is subsequently fed into the GAT-based spatial feature extraction branch together with the graph connectivity information. For each graph, the 400-dimensional node features are successively transformed through three GAT layers with feature dimensions 400→512→256→128. During this process, the graph attention mechanism aggregates information from physically connected signal-path nodes and updates their feature representations according to the graph topology. After the final GAT layer, global average pooling is applied over the 64 signal-path nodes to obtain a 128-dimensional graph-level spatial representation for each measurement. The detailed structure and parameter settings of the GAT-based spatial feature extraction branch are summarized in Table 2.

Table 2. Structure and Parameter Configuration of Spatial Feature Extraction Module

Name	Parameter settings	Description
GAT convolution layer 1	Input feature dimension: 400; output feature dimension: 512; number of attention heads: 4; average output	Utilizing multi-head attention mechanisms to aggregate neighbor node information, extracting rich graph structural features through 4 attention heads, and averaging the results of each head to enhance stability
Activation function	LeakyReLU	Introduce nonlinear transformations to enhance the model's expressive ability while mitigating the 'neuron death' issue of ReLU

GAT Convolution Layer 2	Input feature dimension: 512; Output feature dimension: 256; Number of attention heads: 4; Average output	In a higher-dimensional feature space, selectively aggregate neighbor information to further refine key graph features
Activation function	LeakyReLU	Introduce nonlinear transformations to enhance the model's expressive ability while mitigating the 'neuron death' issue of ReLU
GAT Convolution Layer 3	Input feature dimension: 256; output feature dimension: 128; number of attention heads: 2; average output	Reduce feature dimensions while preserving the core information of the graph structure, providing a concise embedding representation for subsequent damage-location regression
Activation function	LeakyReLU	Introduce nonlinear transformations to enhance the model's expressive ability while mitigating the 'neuron death' issue of ReLU
Pooling layer	Global Average Pooling	Aggregate all node features at the graph level to achieve unified vector output for graph representation, which is obtain a graph-level representation for subsequent damage-location regression

In parallel with the GAT branch, the same B×64×400 node-feature tensor is fed into the Transformer-based global feature extraction branch. In this representation, the 64 signal-path nodes constitute the sequence dimension, while each node is represented by a 400-dimensional feature vector. Therefore, the Transformer input for each measurement has dimensions 64×400. The encoder employs the self-attention mechanism to model long-range dependencies among different signal-path nodes and generate a globally contextualized feature representation. The decoder in PGGT is not used for sequence generation. It takes the same 64×400 node-feature matrix as its query, while the output of the encoder provides the key and value of the cross-attention layer, so that the original node representations are refined by the global context extracted by the encoder. The refined 64×400 representation is then aggregated into the 128-dimensional global feature vector by treating the 64 signal-path nodes as the channel dimension and applying two one-dimensional convolutions along the feature axis, which perform a learnable aggregation over the nodes; the corresponding kernel sizes and strides are listed in Table 3. The resulting 128-dimensional global representation is then fused with the spatial representation extracted by the GAT branch.

After feature extraction, the GAT branch and the Transformer branch each produce a 128-dimensional representation for a given measurement. The GAT representation $\mathbf{g} \in R^{128}$ characterizes the graph-topological relationships among the signal-path nodes, whereas the Transformer representation $\mathbf{t} \in R^{128}$ captures their long-range global dependencies. Since the two representations have the same feature dimension, they are fused by element-wise addition:

$$\mathbf{z} = \mathbf{g} + \mathbf{t} \tag{3}$$

where $\mathbf{z} \in R^{128}$ denotes the fused feature representation. The fused representation is subsequently passed through a fully connected regression head with dimensions 128→64→1, which yields the predicted axial damage coordinate. Since the proposed method is formulated as a continuous regression problem rather than a classification task, no Softmax activation is applied to the output layer, and the single output neuron directly corresponds to the predicted axial damage coordinate. In the present experimental configuration, however, the artificial damage locations vary only along the axial direction, while the circumferential coordinate is kept fixed. Therefore, the proposed coordinate-regression framework is evaluated primarily based on its capability to localize unseen axial damage locations.

Table 3. Structure and Parameter Configuration of Global Feature Extraction Module

Name	Parameter settings	Description
Encoding layer	Feature dimensions: 400; number of heads: 8	Employ multi-head self-attention to model long-range dependencies among signal-path nodes and generate globally contextualized node representations
Decoding layer	Feature dimensions: 400; number of heads: 8; stack layers: 2	Serve as a feature-refinement module that integrates the encoder-derived global contextual information with the original node representations through attention mechanisms
1D convolutional layer over the feature axis	Input channels: 64; output channels: 64; kernel size: 5; stride: 3; padding: 1	Aggregate the 64 node representations as the channel dimension; reduce the feature dimension from 400 to 133
1D convolutional layer over the feature axis	Input channels: 64; output channels: 1; kernel size: 6; stride: 1; padding: 0	Reduce the feature dimension from 133 to 128 and generate the 128-dimensional global representation for fusion with the GAT branch

3.Experimental Validation

A. Experimental Setup

To evaluate the effectiveness of the proposed PGGT, comparative experiments are conducted on the constructed guided-wave dataset. Three evaluation metrics, namely mean absolute percentage error (MAPE), mean absolute error (MAE), and root mean square error (RMSE), are used to quantify localization performance, and several classic network structures are selected as comparison methods, including convolutional neural networks (VGG16 [24] , ResNet18 [25]), graph neural networks (ChebyNet [26], GCN [27], GAT[28]), a sequence modeling network (Transformer) and a graph-attention-based comparison model (PGGDL [23]). Under a unified experimental setting, four cases are constructed using different splits between training locations and unseen test locations to evaluate model generalization under limited training labels and varying spatial distributions, thereby systematically assessing the prediction accuracy and stability of each model at unseen damage locations.

The experimental setup is shown in Fig. 3, consisting of a NI PXIe-1071 four-slot chassis unit, NI controller PXIe-8840 expansion card, NI PXIe-5413 arbitrary waveform generator, NI PXI-6115 data acquisition card, NI PXIe-2737 switch matrix, NI BNC-2110 shielded terminal box, and NTBX-50B threaded terminal strip. The arbitrary waveform generator outputs an excitation signal with a frequency set to 200 kHz and a periodic pulse signal with an output voltage of 1 V. After power amplification, the voltage gain is approximately 10 times, thereby increasing the excitation voltage to 10 V to meet the energy requirements of guided wave excitation. The experimental specimen is a composite pipeline with a diameter of 150 mm, a thickness of 2 mm, and a length of 300 mm. Sixteen annular piezoelectric ceramic (PZT) transducers, each with a diameter of 10 mm and a thickness of 1 mm, are arranged on the outer surface of the pipeline. PZT1–PZT8 are used as actuators, whereas PZT9–PZT16 serve as receivers, forming a pitch-catch sensing network with eight actuators and eight receivers. During each measurement, PZT1–PZT8 are excited sequentially. For each actuator excitation, the response signals from all eight receivers (PZT9–PZT16) are recorded. Therefore, each actuator produces eight actuator–receiver pitch-catch signals, and a complete measurement contains 8×8=64 guided-wave channels. Specifically, each channel corresponds to one unique actuator–receiver

propagation path. For example, the first eight channels correspond to PZT1–PZT9 through PZT1–PZT16, the next eight channels correspond to PZT2–PZT9 through PZT2–PZT16, and the same organization is followed for the remaining actuators up to PZT8–PZT16. Thus, the 64 channels represent 64 distinct guided-wave propagation paths generated by the 16 physical PZT transducers, rather than 64 physical sensors.

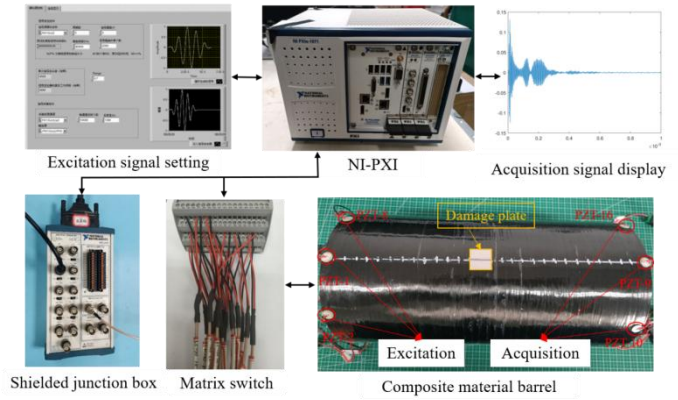


Fig. 3. The experimental set-up

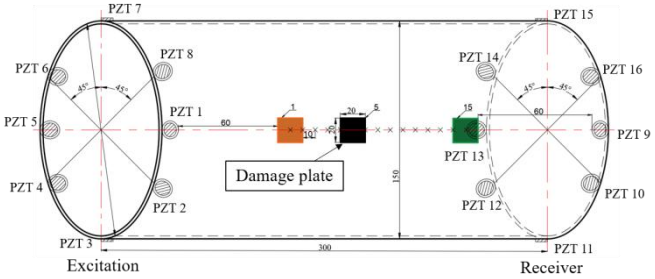


Fig. 4. Arrangement diagram of piezoelectric sensors on the pipeline surface

The experiment collects guided-wave signals under both intact and damaged conditions. Damage is simulated by attaching a 20 mm (L) × 20 mm (W) × 1 mm (H) aluminum block to different positions on the composite pipeline, as illustrated in Fig. 4. During all damage-state measurements, the centerline of the aluminum block is aligned with the centerline between PZT1 and PZT9 so that the circumferential position of the artificial damage remains unchanged. The aluminum block is moved only along the axial direction of the pipeline in increments of 10 mm, resulting in 15 discrete axial damage locations. Therefore, although the damage location is represented by spatial coordinates in the visualization, the quantity predicted by the model and evaluated in this study is the axial damage coordinate, while the circumferential coordinate is fixed. Accordingly, the experimental validation in this study primarily evaluates localization and generalization to unseen axial damage locations, rather than arbitrary two-dimensional damage locations over the entire pipeline surface.

The complete experimental dataset consists of guided-wave signals from 15 damaged locations and one intact condition, with 50 samples collected for each condition. The intact-state signals were collected as part of the complete experimental acquisition but were not included in the damage-location regression experiments. Since the intact condition does not correspond to a physical damage coordinate, no artificial coordinate label is assigned to these samples. Therefore, only samples from the 15 damaged locations are used for the training, validation, and unseen-location test partitions described below, and no separate intact-versus-damaged classification task is performed in this study. Each sample contains 64 guided-wave channels, with a sampling rate of 10 MHz and 10,000 data points

recorded for each channel.

Each raw record contains 10,000 sampling points per channel at a sampling rate of 10 MHz. The first 4,000 sampling points of each channel, corresponding to the interval 0-400 μs, are retained as the network input; this window contains the excitation pulse and the early-arriving direct and damage-scattered wave packets, and it is fixed and identical for all channels, damage locations and repeated measurements. The retained signals are then band-pass filtered with a fourth-order Butterworth filter with a pass band of 10kHz-600kHz, implemented as a zero-phase forward-backward filter, and normalized to the interval [0, 1] by min-max normalization.

To evaluate the generalization capability of the models at unseen damage locations, the 15 damage locations are first divided into training locations and unseen test locations according to the four cases listed in Table 4. For each training location, 80% of the corresponding samples are used to form the training set, while the remaining 20% are used as the validation set. The training set is used for model parameter optimization, whereas the validation set is used for model selection. All samples from the unseen locations are reserved exclusively for the final test and are not involved in model training, hyperparameter selection, learning-rate adjustment, or model selection. Therefore, the unseen-location test set provides an independent evaluation of the model's generalization capability to damage locations that are not observed during training or validation.

Table 4. Dataset division

Case	Training locations	Unseen test locations
Case 1	1, 2, 3, 4, 7, 9, 10, 11, 13, 14	5, 6, 8, 12, 15
Case 2	1, 2, 3, 4, 5, 11, 12, 13, 14, 15	6, 7, 8, 9, 10
Case 3	2, 3, 4, 5, 6, 10, 11, 12, 13, 14	1, 7, 8, 9, 15
Case 4	3, 4, 5, 6, 10, 11, 12, 13	1, 2, 7, 8, 9, 14, 15

Because the circumferential coordinate is fixed for all 15 artificial damage locations, the differences among the investigated damage locations are determined by their axial coordinates. Therefore, the reported localization results primarily characterize the prediction accuracy for the axial damage position.

Three quantitative indicators, namely Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE), are used to evaluate the damage localization performance of these methods.

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|v_i - \hat{v}_i|}{v_i} \tag{4}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|v_i - \hat{v}_i| \tag{5}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|v_i - \hat{v}_i|^2} \tag{6}$$

where v_i and $\hat{v}_i$ denote the actual and predicted values of the axial damage coordinate of the i-th

sample, respectively, and n is the number of testing samples. In Eq. (4), MAPE is a dimensionless normalized error, defined as the mean ratio of the localization error to the true axial damage coordinate, so that the values reported in Tables 5-10 are fractions rather than percentages: a reported value of 0.014 corresponds to a relative error of 1.4%. MAE and RMSE are the mean and root-mean-square values of the localization error and are expressed in millimetres, the unit of the axial damage coordinate.

Based on the dataset division in the aforementioned four cases, to ensure the fairness and reproducibility of the comparison results, each model uses a unified training setup (batch size=32, epoch=100, initial learning rate=0.01), and an adaptive learning rate decay strategy is introduced: when the training loss enters a plateau, the learning rate is reduced by 50%, and if the loss continues to decrease, the learning rate remains unchanged. Subsequently, according to the division in Cases 1-4, the damage localization performance of different models is compared and analyzed. To account for the randomness arising from network initialization and data shuffling, all experiments reported in this work were repeated three times with different random seeds, and the results are reported as the mean and standard deviation over these runs. No statistical significance test was performed, and the comparisons in the following sections are therefore stated in terms of the mean values obtained over the repeated runs.

For all four cases, the same three-way data-management protocol is adopted. Specifically, 80% of the samples from the designated training locations are used for parameter optimization, and the remaining 20% are retained as a validation set. The validation set contains only samples from locations already represented in the training-location group and is used for model selection during training. In contrast, all samples from the designated unseen locations form an independent test set and are evaluated only after model selection has been completed. No unseen-location samples are used during model training or validation.

B. Comparison with Existing Methods

1) Case 1

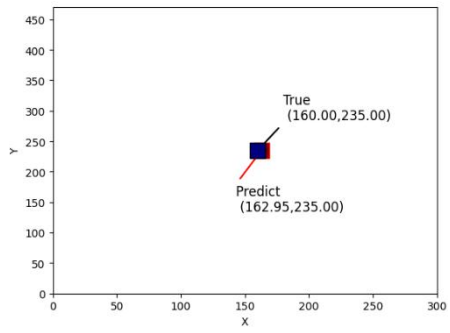
In Case 1, damage locations 5, 6, 8, 12, and 15 are designated as the unseen test locations, while locations 1, 2, 3, 4, 7, 9, 10, 11, 13, and 14 constitute the training-location group. For each training location, 80% of the samples are used for training and the remaining 20% for validation. All samples from the five unseen locations are reserved exclusively for the final test. This case evaluates the model's ability to generalize to spatially distributed damage locations that are absent from both the training and validation sets. The quantitative results on the unseen-location test set are presented in Table 5.

Table 5. Comparison of the three quantitative indicators for Case 1

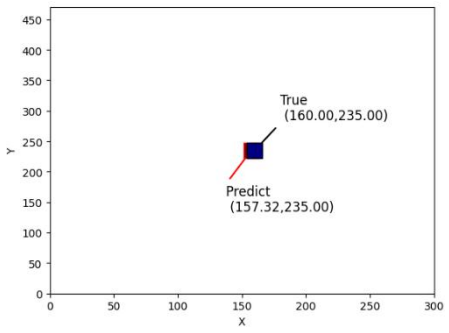
Model	MAPE	MAE (mm)	RMSE (mm)
VGG16	0.025±0.001	4.46±0.11	6.18±0.13
ResNet18	0.017±0.001	2.92±0.08	3.78±0.09
ChebyNet	0.018±0.001	2.88±0.08	3.45±0.10
GCN	0.031±0.001	4.75±0.14	5.49±0.14
GAT	0.015±0.001	2.52±0.06	3.61±0.09

Transformer	0.034±0.001	5.13±0.12	6.79±0.13
PGGDL	0.023±0.001	3.94±0.12	4.91±0.12
PGGT	0.014±0.001	2.27±0.07	2.91±0.09

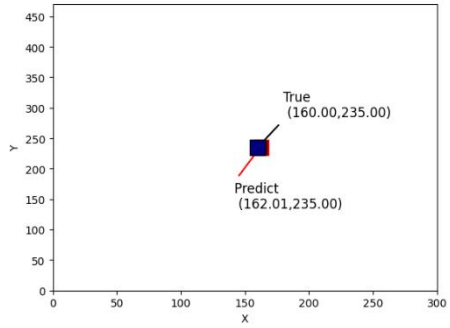
The table shows that the PGGT model achieves the lowest errors among all compared models, with a MAPE of 0.014, a MAE of 2.27 mm, and an RMSE of 2.91 mm, which are the lowest values among all compared models. This excellent performance may be attributed to its combination of the advantages of graph neural networks and Transformers, which allows it to better capture the complex dependency relationships in the data. The GAT model follows closely behind, enhancing the expressiveness of node features through the attention mechanism. The ResNet18 and ChebyNet models also perform well, respectively benefiting from the residual structure's ability to extract deep features and the Chebyshev polynomial's local filtering effect on graph data. In comparison, the standalone Transformer yields relatively large errors, possibly due to its overfitting problem with small-scale data and the lack of explicit modeling of graph structural information. Traditional models like VGG16 and GCN are limited by their simple structure or insufficient feature extraction capabilities, resulting in relatively average performance. To visually demonstrate the prediction results, Fig. 5(a-h) compares the predictions of each model with the actual values, with blue rectangles representing the actual values and red rectangles representing the predicted values.



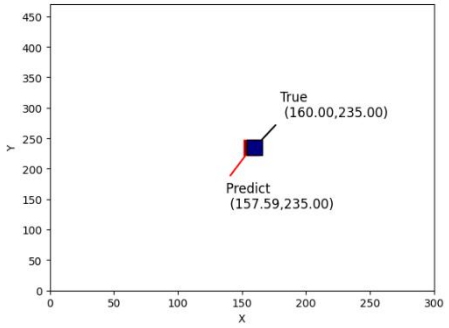
(a) VGG16



(b) ResNet18

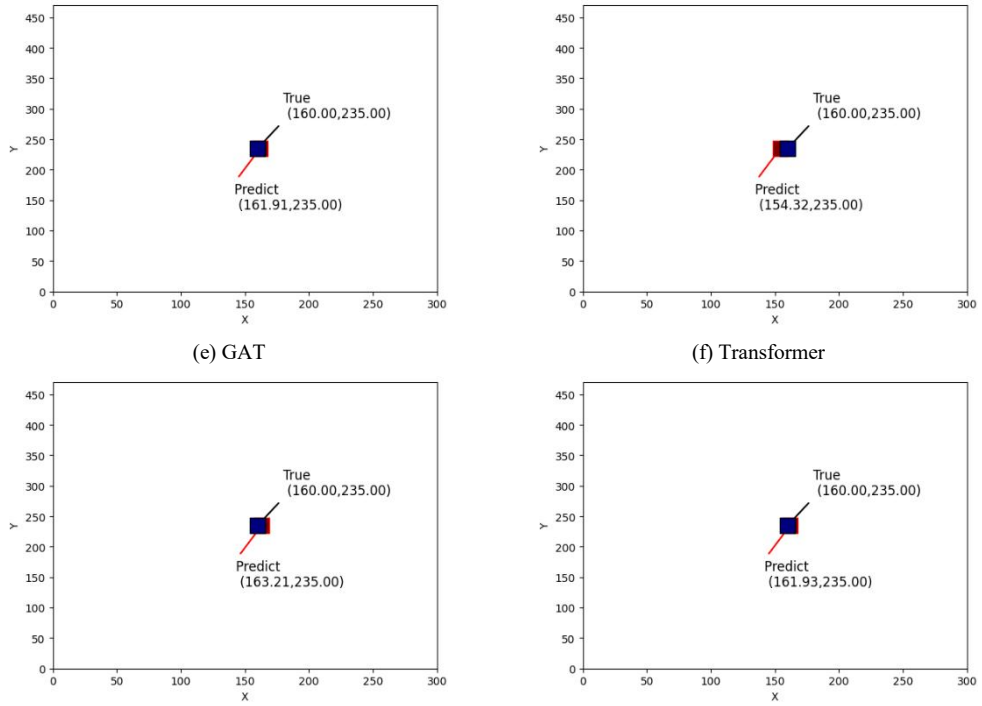


(c) ChebyNet



(d) GCN

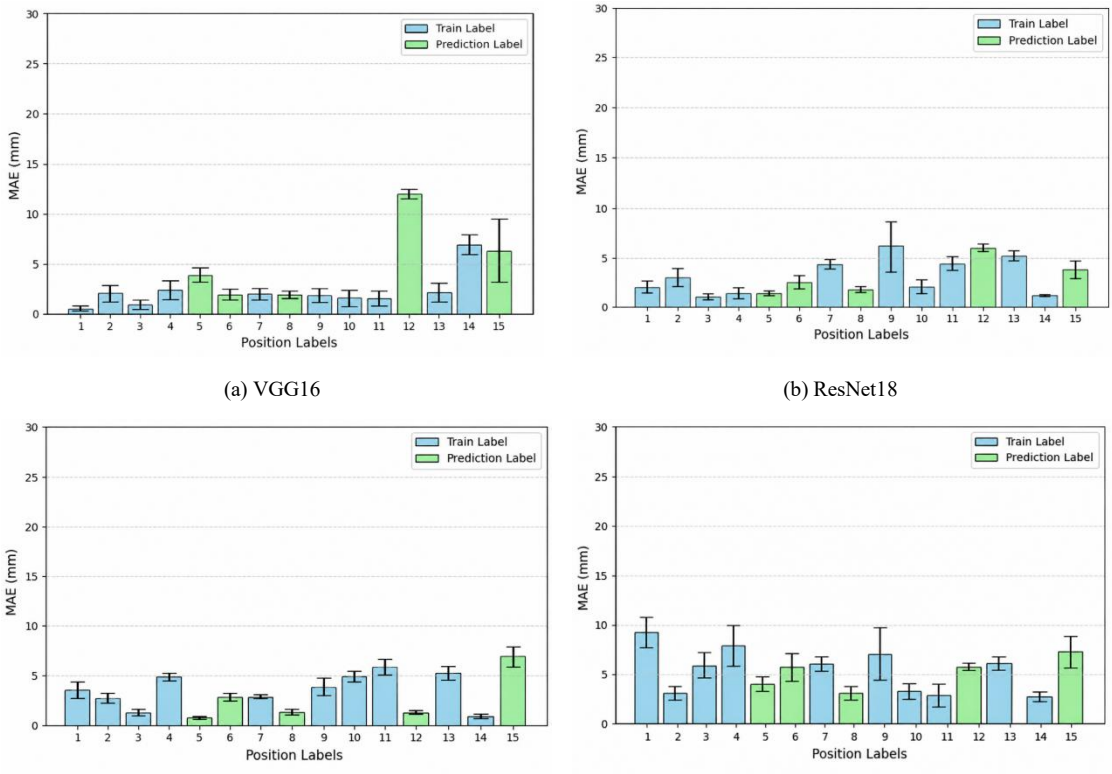
https://doi.org/10.37965/jdmd.2026.1675



(e) GAT (f) Transformer

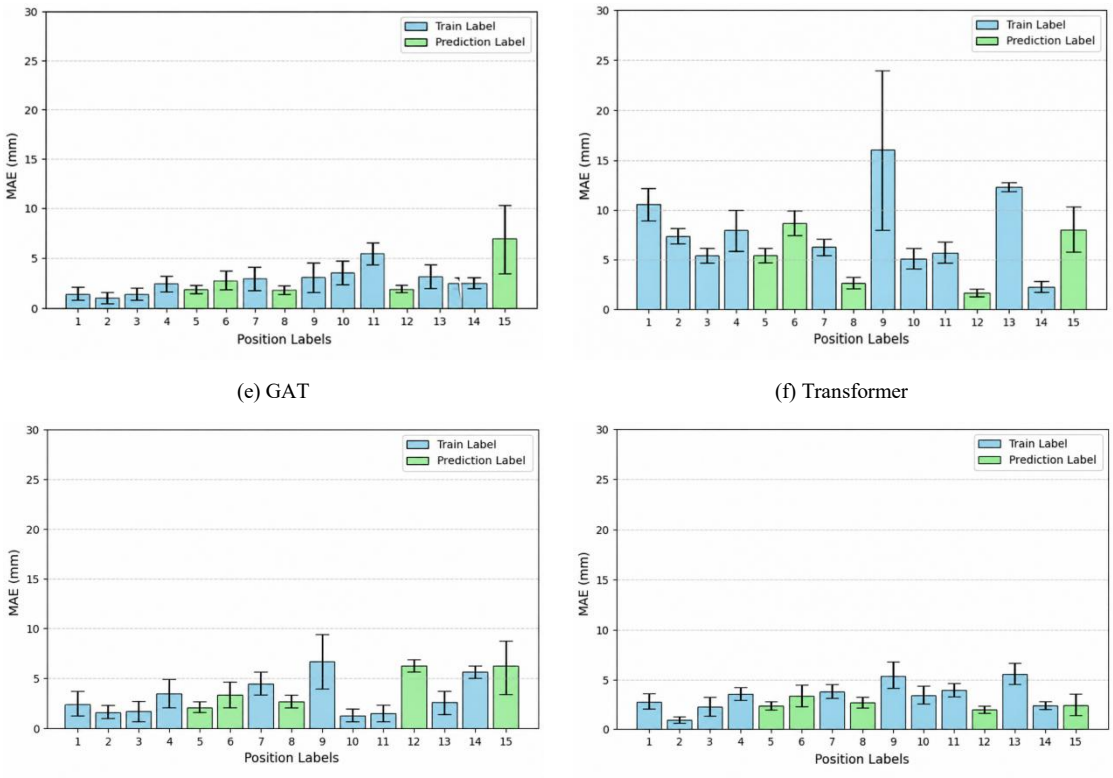
(g) PGGDL (h) PGGT

Fig. 5. Representative localization results for Case 1 using (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, and (h) PGGT. The blue and red rectangles denote the ground truth and predicted results, respectively



(a) VGG16 (b) ResNet18

(c) ChebyNet (d) GCN



(e) GAT (f) Transformer

(g) PGGDL (h) PGGT

Fig. 6. Mean absolute localization error at each damage position for the different models in Case 1: (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, (h) PGGT. Bars denote the mean absolute error of the predicted axial damage coordinate at each position (unit: mm), and the error bars denote the 95% confidence interval of the mean; blue and green bars correspond to the training locations and to the locations held out for testing, respectively.

Through the comparative analysis of the error limits at all positions, we can further analyze the performance of each model at each data position. The mean absolute localization error at each damage position is shown in Fig. 6. VGG16 exhibits an unusually high unseen-location test error at position 12, indicating poor generalization ability at this position. Moreover, the training-location error at position 14 is relatively high, further demonstrating the instability of VGG16 in handling position-sensitive data. The Transformer model shows pronounced imbalance on most positions, especially at positions 9 and 13, where the training-location error is extremely high and fluctuates greatly, limiting its applicability at these positions. ResNet18 performs moderately, with a relatively high training-location error and pronounced fluctuations at position 9, suggesting high prediction uncertainty at this position. The unseen-location test error at position 12 is higher than the training-location error, hinting at insufficient adaptability of the model at this position. GCN's performance is relatively unstable, especially at positions 1, 4, 9, and 15, where the errors are high, and the unseen-location test errors at multiple positions are markedly higher than the training-location errors, indicating insufficient generalization ability at these positions. The performance of PGGDL and ChebyNet models is at a moderate level. PGGDL has a relatively high training-location error at position 9, and larger unseen-location test errors at positions 12 and 15. In comparison, PGGT has a more uniform error distribution, although the unseen-location test error at position 15 is the highest.

Among all the models, PGGT demonstrates the best stability and generalization ability. Its error

distribution remains consistent across all test locations, and the discrepancy between the errors at training locations and unseen test locations is relatively small.

2) Case 2

In Case 2, damage locations 6–10 are designated as the unseen test locations, whereas locations 1–5 and 11–15 constitute the training-location group. Within the training-location group, 80% of the samples at each location are used for model training and the remaining 20% for validation. All samples from locations 6–10 are reserved exclusively for final testing. This partition creates a continuous unseen region in the central part of the investigated damage range and is used to evaluate interpolation-like generalization across a spatial gap not represented during training or validation. The quantitative results on the unseen-location test set are summarized in Table 6.

Table 6 summarizes the specific results of quantitative indicators for different models on the unseen-location test set of this case, and the results are averaged over three independent runs. The PGGT model performs best on all three indicators (MAPE, MAE, and RMSE), with values of 0.024, 4.04 mm, and 5.01 mm, respectively, which are the lowest values of the three indicators among all compared models. Although it is slightly inferior to PGGT, the PGGDL model still possesses strong predictive power. The GCN model also shows good performance by effectively capturing spatial relationships between features. Although it is not as good as PGGT and PGGDL, it still exhibits good robustness.

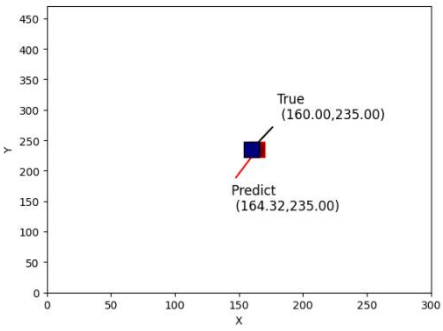
Compared with other models, the VGG16 model performs poorly, especially in terms of the RMSE metric, indicating its insufficient performance in handling complex data. The ResNet18, ChebyNet, and GAT models exhibit similar performance, with higher MAPE and MAE values, failing to reach the precision of PGGT and PGGDL. The Transformer model performs the worst, yielding larger errors and failing to extract key features effectively. Overall, the PGGT model shows the best performance in Case 2, with other models such as PGGDL and GCN also demonstrating good performance. However, models like VGG16 and Transformer fail to meet the challenges of the case. To visually demonstrate the prediction results, Fig. 7(a-h) compares the predictions of various models with the actual values.

Table 6. Comparison of the three quantitative indicators for Case 2.

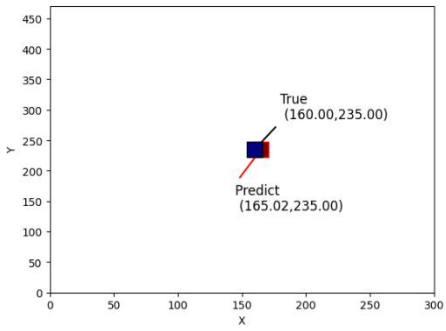
Model	MAPE	MAE (mm)	RMSE (mm)
VGG16	0.038±0.002	5.96±0.21	10.2±0.34
ResNet18	0.048±0.003	7.69±0.29	10.6±0.41
ChebyNet	0.049±0.003	7.46±0.27	9.65±0.36
GCN	0.040±0.002	6.15±0.24	7.76±0.28
GAT	0.047±0.003	7.51±0.26	9.19±0.36
Transformer	0.048±0.003	7.43±0.31	11.5±0.45
PGGDL	0.037±0.002	5.75±0.20	7.49±0.27
PGGT	**0.024±0.001**	**4.04±0.15**	**5.01±0.18**

The mean absolute localization error at each damage position is used to systematically analyze the prediction accuracy and generalization ability of each model, as shown in Fig. 8. The error distribution of the GAT model is relatively balanced, but the training-location errors at positions 4, 5, 11 and 15 are

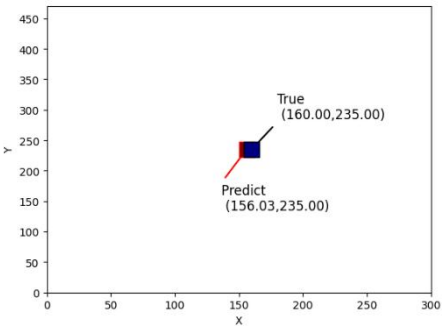
higher, especially the error variance at position 15 increases markedly, indicating that the prediction at this position is unstable, and the unseen-location test errors at positions 7 and 8 are higher than the training-location errors, showing insufficient generalization ability. The training-location error of the GCN model is higher at positions 1 and 4, and the unseen-location test error in the interval from positions 6 to 10 is generally higher than the training-location error, especially position 9, which shows poor generalization ability. The PGGDL model has low errors in most positions and shows certain stability, but the training-location errors at positions 1 and 15 are higher, and the unseen-location test errors in the interval from positions 6 to 9 are higher. The training-location error of ResNet18 is close to zero, but the interval unseen-location test error increases markedly at the "center" position, especially position 9, indicating that there may be a serious overfitting problem and insufficient generalization ability. The training-location error of the Transformer model is higher at positions 1 to 5 and 11 to 15, and the unseen-location test error is concentrated at positions 6 to 9, indicating a structural adaptation problem. The error level of VGG16 is medium, with higher training-location errors at positions 1 and 3, and larger unseen-location test errors at positions 7 and 8, but overall, it shows good generalization ability. The PGGT model shows a more balanced error distribution, and the training-location error remains at a low level (about 2-10) in most positions. Compared with other models, the unseen-location test errors at the five "center" locations are the lowest, and the error variances are relatively small, indicating that the model's performance at these locations is relatively stable.



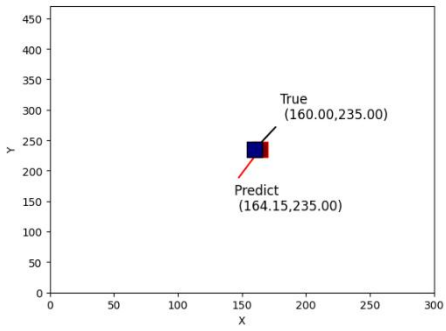
(a) VGG16



(b) ResNet18

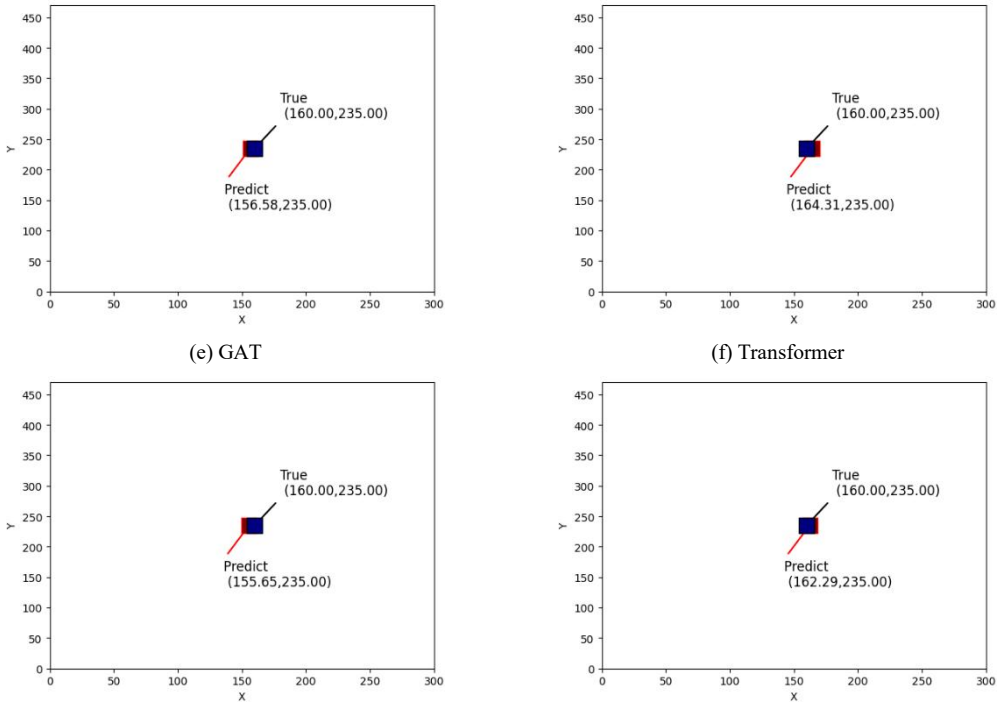


(c) ChebyNet



(d) GCN

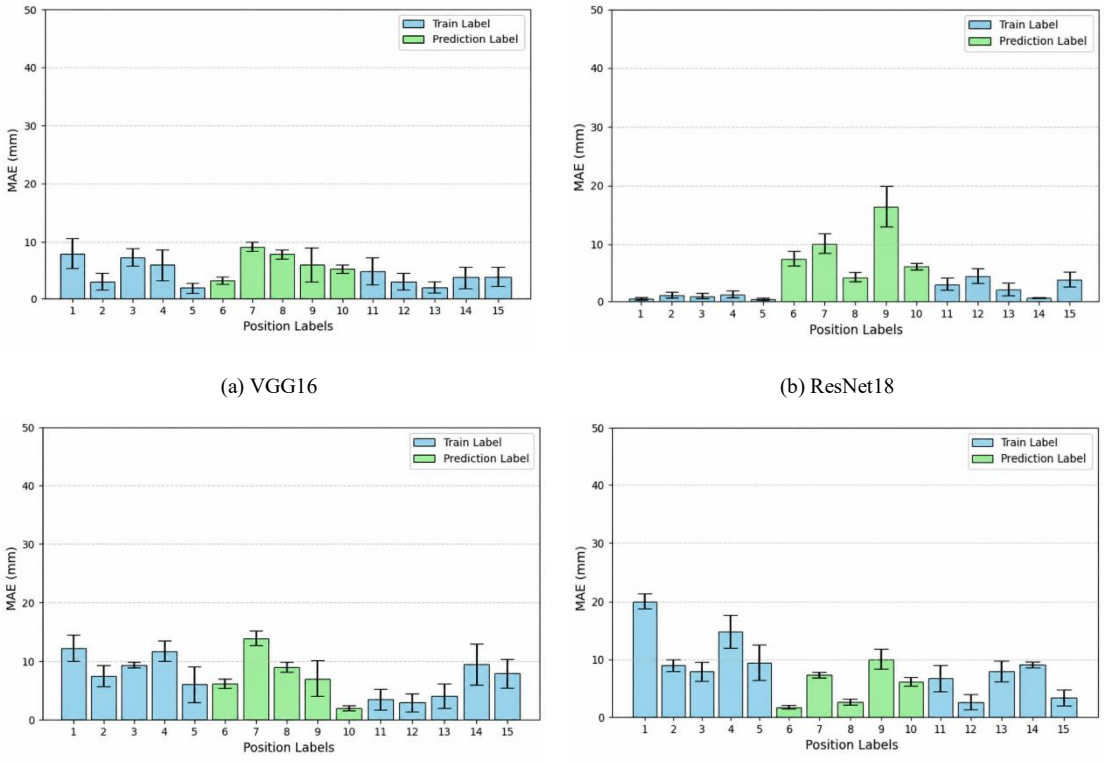
https://doi.org/10.37965/jdmd.2026.1675



(e) GAT (f) Transformer

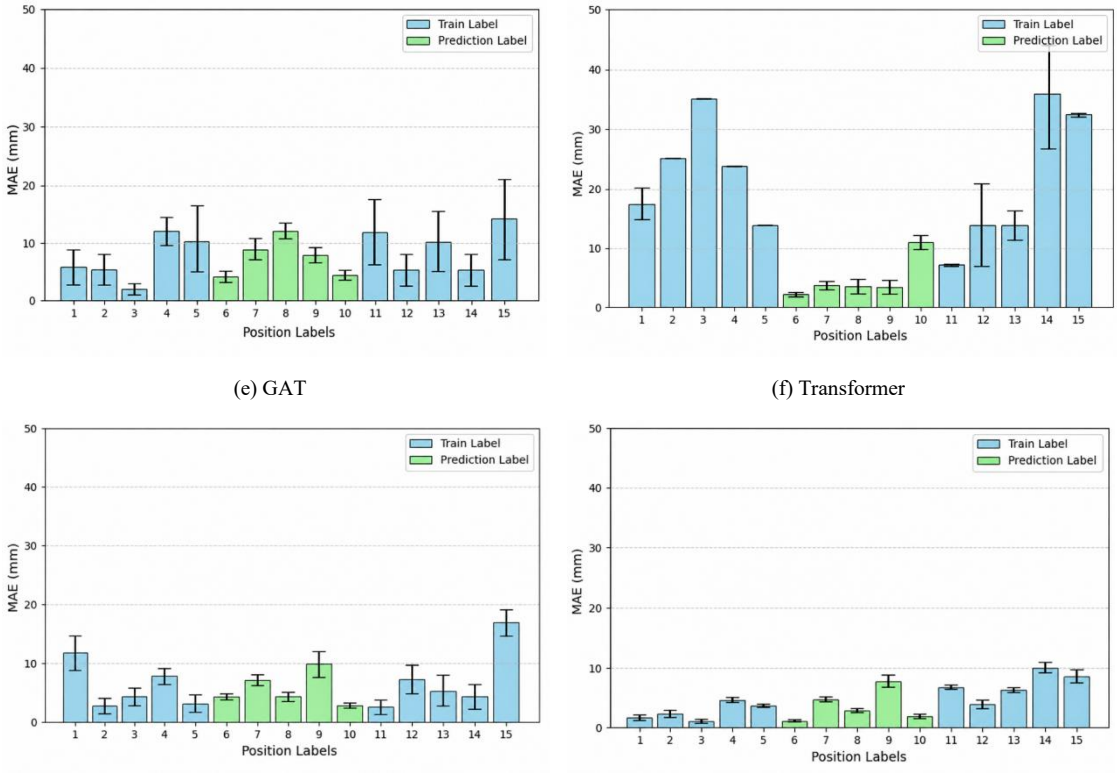
(g) PGGDL (h) PGGT

Fig. 7. Representative localization results for Case 2 using (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, and (h) PGGT. The blue and red rectangles denote the ground truth and predicted results, respectively



(a) VGG16 (b) ResNet18

(c) ChebyNet (d) GCN



(e) GAT (f) Transformer

(g) PGGDL (h) PGGT

Fig. 8. Mean absolute localization error at each damage position for the different models in Case 2: (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, (h) PGGT. Bars denote the mean absolute error of the predicted axial damage coordinate at each position (unit: mm), and the error bars denote the 95% confidence interval of the mean; blue and green bars correspond to the training locations and to the locations held out for testing, respectively.

3) Case 3

In Case 3, damage locations 1, 7, 8, 9, and 15 are designated as the unseen test locations, while locations 2–6 and 10–14 constitute the training-location group. For each training location, 80% of the samples are used for training and the remaining 20% for validation. All samples from the unseen locations are used only for the final test. This partition evaluates the model's generalization capability simultaneously at boundary and central unseen locations under spatially separated training and testing conditions. The quantitative results on the unseen-location test set are presented in Table 7.

As shown in Table 7, PGGT achieves the best performance among all compared models, with a MAPE of 0.028, an MAE of 3.97, and an RMSE of 4.80. These results indicate that PGGT can provide more accurate and stable damage detection under this challenging partition strategy. The GAT model ranks second, which suggests that attention-based graph modeling is effective in capturing spatial dependencies among sensors. However, compared with GAT, PGGT further benefits from the integration of local feature extraction, topological relationship modeling, and global dependency aggregation, leading to improved localization accuracy.

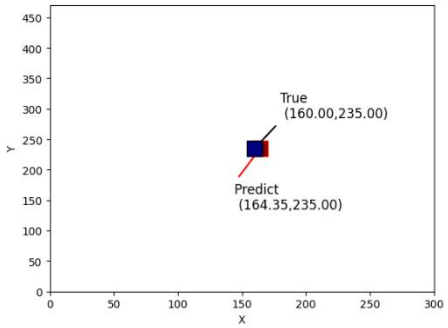
Table 7. Comparison of the three quantitative indicators for Case 3

Model	MAPE	MAE (mm)	RMSE (mm)
VGG16	0.035±0.002	5.02±0.17	6.41±0.21

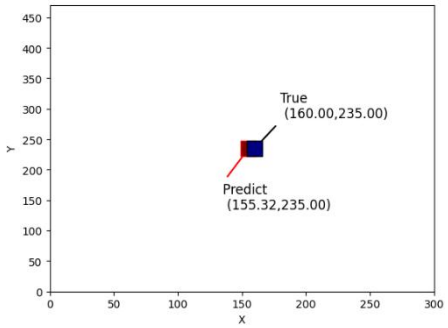
ResNet18	0.041±0.002	5.82±0.19	7.48±0.24
ChebyNet	0.032±0.002	4.22±0.14	5.63±0.26
GCN	0.029±0.001	4.31±0.15	5.53±0.18
GAT	0.030±0.001	4.18±0.16	5.24±0.17
Transformer	0.041±0.002	5.81±0.20	7.48±0.24
PGGDL	0.034±0.002	4.88±0.16	5.79±0.17
PGGT	**0.028±0.001**	**3.97±0.12**	**4.80±0.15**

Although it is slightly inferior to GAT and PGGT, the GCN model still demonstrates good robustness and accuracy. The performance of the ChebyNet model is similar to that of GCN, with certain predictive capabilities, but it falls short in terms of precision. In comparison, the VGG16 model performs poorly in this case. Although it shows a relatively small error, it still fails to surpass PGGT and other graph neural network models, indicating that its overall performance is not satisfactory. The ResNet18 and Transformer models also perform poorly, failing to effectively extract key information from the data, which limits the overall performance of the models. To intuitively demonstrate the predictive effect, Fig. 9 (a-h) compares the predictive results of various models with the true values, where the blue rectangles represent the true values and the red rectangles represent the predicted values.

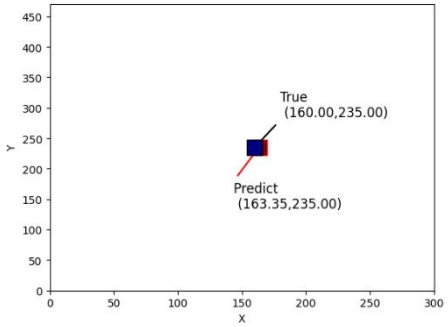
As shown in Fig. 10, the distribution characteristics of the mean absolute localization error of various models reflect the performance differences of these models in the location-sensitive prediction experiment. PGGT exhibits a relatively concentrated error distribution, with the training-location errors at positions 3 and 10 showing peaks, and the unseen-location test errors at positions 7-8 being relatively high. However, the unseen-location test errors at most positions are lower than the training-location errors, indicating that it has good generalization ability on new datasets.



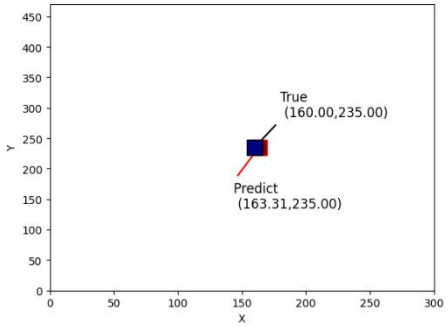
(a) VGG16



(b) ResNet18

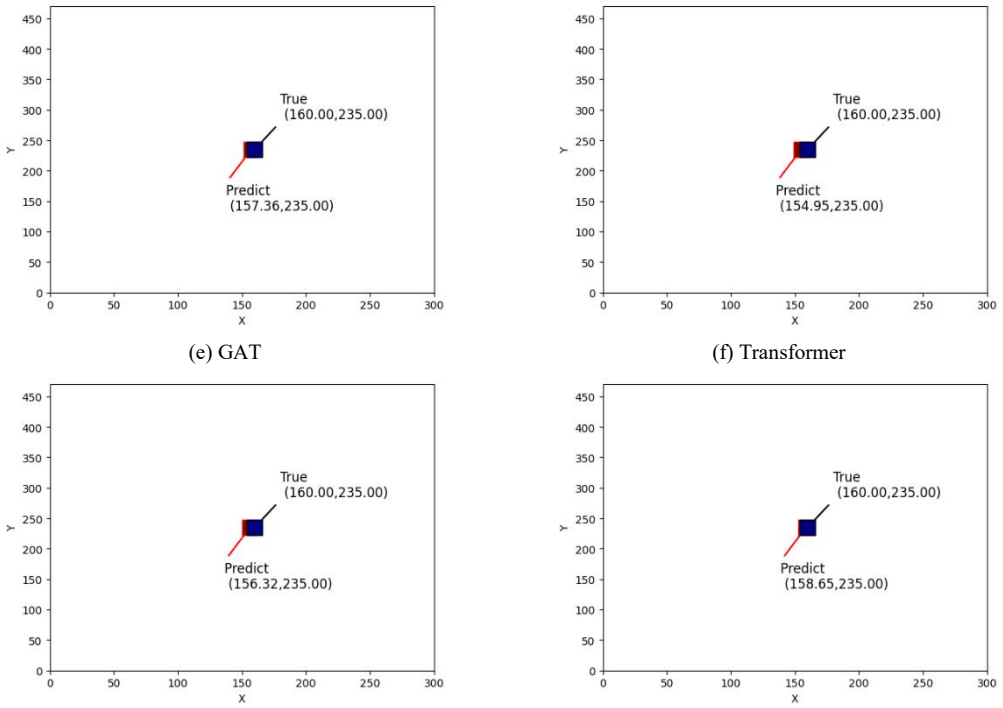


(c) ChebyNet



(d) GCN

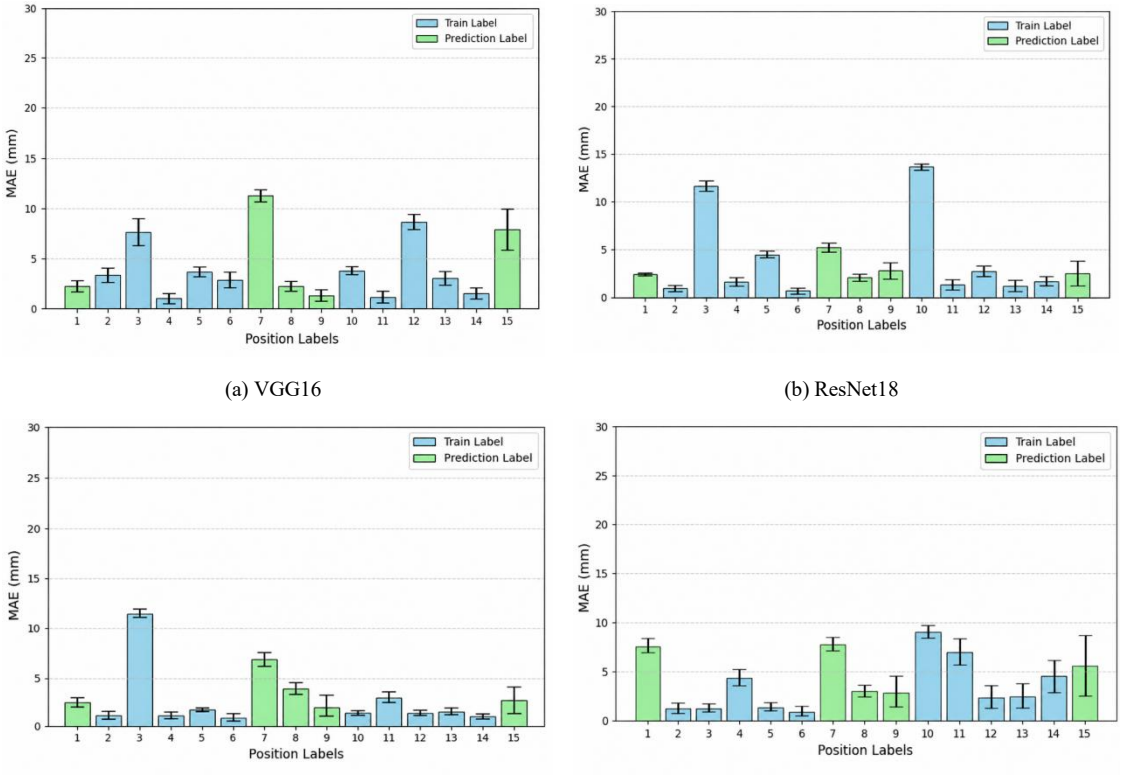
https://doi.org/10.37965/jdmd.2026.1675



(e) GAT (f) Transformer

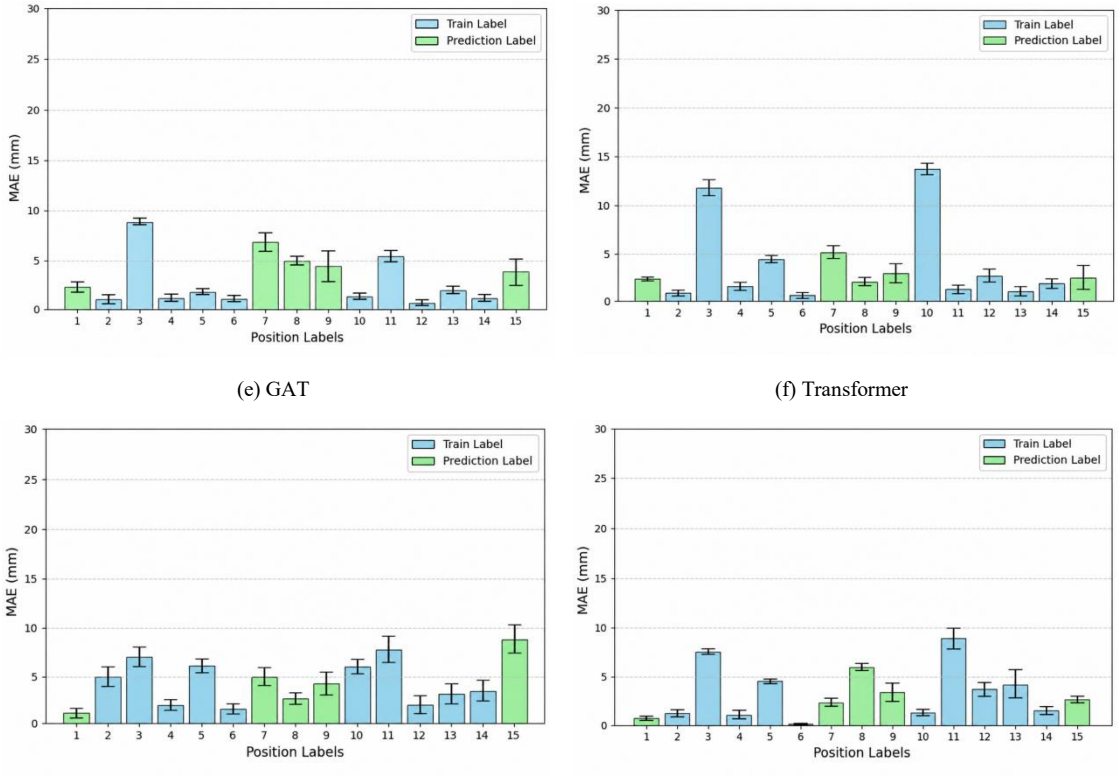
(g) PGGDL (h) PGGT

Fig. 9. Representative localization results for Case 3 using (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, and (h) PGGT. The blue and red rectangles denote the ground truth and predicted results, respectively



(a) VGG16 (b) ResNet18

(c) ChebyNet (d) GCN



(e) GAT (f) Transformer

(g) PGGDL (h) PGGT

Fig. 10. Mean absolute localization error at each damage position for the different models in Case 3: (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, (h) PGGT. Bars denote the mean absolute error of the predicted axial damage coordinate at each position (unit: mm), and the error bars denote the 95% confidence interval of the mean; blue and green bars correspond to the training locations and to the locations held out for testing, respectively.

ChebyNet on this dataset shows marked error imbalance, with the training-location error at position 3 being particularly high (about 11.5), while the unseen-location test error is mainly concentrated in the 6-7 range, especially at position 7 (about 7.5), indicating the limitations of the model's adaptability. The GAT model is similar, with a higher training-location error at position 3, and larger unseen-location test errors in the 7-9 range, especially at position 15, where the variance of the error is large, suggesting poor prediction stability at this position. The error distribution of the GCN model is relatively unique, with higher unseen-location test errors at positions 1 and 7, and a peak in the training-location error at positions 9-10, reflecting the inconsistency in the model's handling of different positions.

The error distribution of ResNet18 is relatively concentrated, with the training-location errors at positions 3 and 10 reaching peaks, and the unseen-location test error at position 7 being the highest (about 5.5). Overall, it avoids overfitting and demonstrates strong generalization ability. The error distribution of the Transformer model is highly similar to that of ResNet18, with pronounced training-location errors at positions 3 and 10 and similar unseen-location test error distribution, which may reflect the similar feature extraction strategies that the model adopts when processing this dataset. The error distribution of VGG16 is relatively balanced, with higher training-location errors at positions 3 and 11 and larger unseen-location test errors at positions 7 and 15, especially at position 7, indicating insufficient generalization ability at that location.

PGGDL then shows a relatively scattered error distribution, with training-location errors forming peaks at multiple positions (such as 2-5, 9-10, and 14), and the unseen-location test errors reaching the highest at position 15, indicating that the model is relatively unstable on the new dataset.

Overall, the PGGT model performs best in Case 3, with GAT, GCN, and ChebyNet also demonstrating strong capabilities. However, VGG16 and Transformer show relatively poor performance in this case, failing to fully realize their potential.

4) Case 4

In Case 4, damage locations 1, 2, 7, 8, 9, 14, and 15 are designated as the unseen test locations, whereas locations 3, 4, 5, 6, 10, 11, 12, and 13 constitute the training-location group. As in the other cases, 80% of the samples at each training location are used for model training and the remaining 20% for validation. All samples from the seven unseen locations are reserved exclusively for final testing. Compared with the other cases, Case 4 contains fewer training locations and simultaneously includes unseen locations at both the boundaries and the central region, thereby providing the most challenging evaluation of generalization under limited spatial coverage.

In Case 4, the performance of different models varies in various quantitative indicators, as detailed in Table 8. The PGGT model continues to excel in Case 4, with a MAPE of 0.035, a MAE of 5.38 mm, and an RMSE of 6.23 mm, which are the lowest values of the three indicators among the compared models. The VGG16 model performs relatively weakly, with a MAPE of 0.043, a MAE of 6.21, and an RMSE of 8.47. Although the error is relatively small, it fails to reach the precision of the PGGT model, indicating its shortcomings in complex tasks.

Table 8. Comparison of the three quantitative indicators for Case 4

Model	MAPE	MAE (mm)	RMSE (mm)
VGG16	0.043±0.002	6.21±0.24	8.47±0.31
ResNet18	0.064±0.004	11.12±0.42	18.7±0.68
ChebyNet	0.068±0.004	9.91±0.44	14.3±0.37
GCN	0.081±0.005	11.20±0.37	15.3±0.45
GAT	0.070±0.004	10.41±0.39	13.6±0.50
Transformer	0.048±0.002	7.42±0.35	11.47±0.35
PGGDL	0.059±0.003	8.45±0.29	12.2±0.46
PGGT	**0.035±0.002**	**5.38±0.21**	**6.23±0.27**

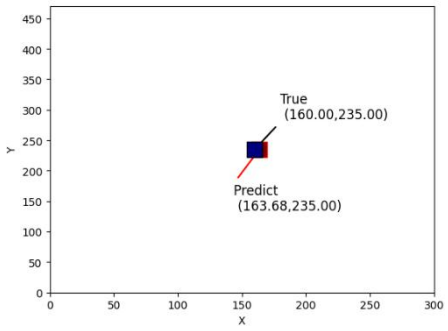
The Transformer model performs relatively well on MAPE and MAE, but has a larger error in terms of RMSE, failing to effectively capture the complex features in the data. The PGGDL model shows a more stable performance, with a MAPE of 0.059, an MAE of 8.45, and an RMSE of 12.2, indicating a certain predictive capability. However, it is slightly inferior in accuracy compared to PGGT and VGG16.

The GCN and ChebyNet models show relatively close performance, with MAPE values of 0.081 and 0.068, MAE values of 11.2 and 9.91, and RMSE values of 15.3 and 14.3, respectively, all showing considerable errors and failing to effectively enhance predictive power. This indicates that graph neural networks may be limited by data partitioning and feature learning methods when dealing with this case,

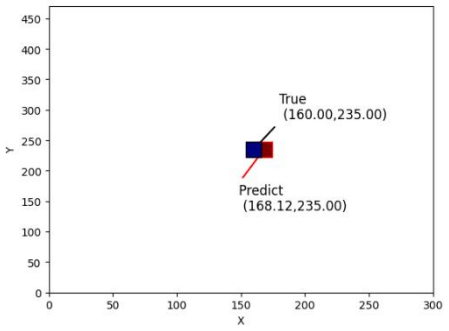
leading to unsatisfactory performance.

In Case 4, the ResNet18 model performs the worst, with a MAPE of 0.064, an MAE of 11.12, and an RMSE of 18.7, indicating a considerable error. It fails to effectively capture the key information in the data, resulting in its overall predictive ability being far inferior to that of other models. To visually demonstrate the predictive results, Fig. 11(a-h) compares the predicted results of various models with the actual values, where the blue rectangles represent the actual values and the red rectangles represent the predicted values.

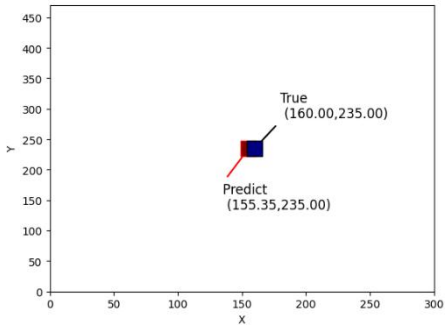
As shown in Fig. 12, the mean absolute localization error of various network models on this dataset is as follows. The PGGT model exhibits a relatively stable error distribution on this dataset, with the highest training-location error at position 11, and the unseen-location test error concentrated mainly at positions 1 and 2. Unlike other models, PGGT does not show an abnormally high unseen-location test error at position 15, demonstrating its advantage in handling edge positions. Overall, PGGT shows strong generalization ability, avoids overfitting issues, and exhibits good stability and adaptability in processing edge position data, thereby providing a more consistent performance.



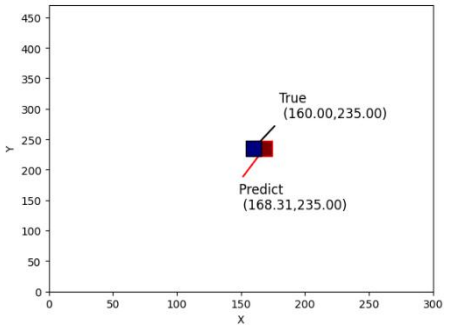
(a) VGG16



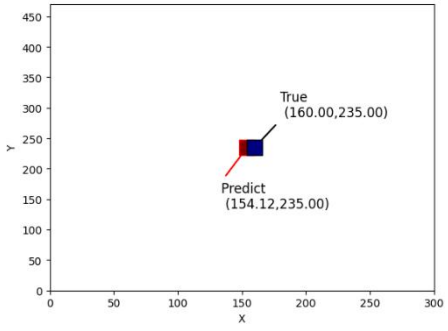
(b) ResNet18



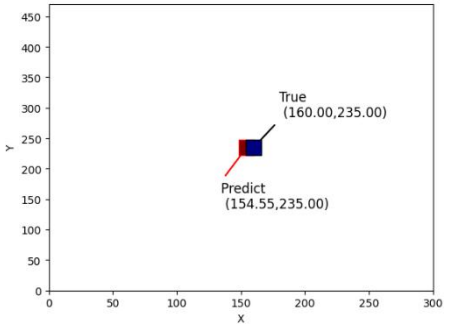
(c) ChebyNet



(d) GCN

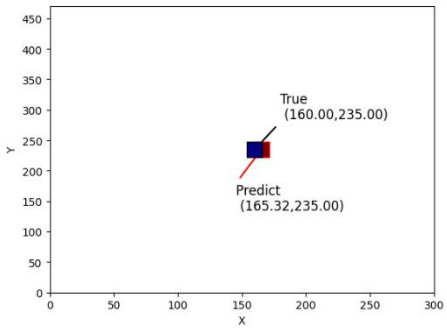


(e) GAT

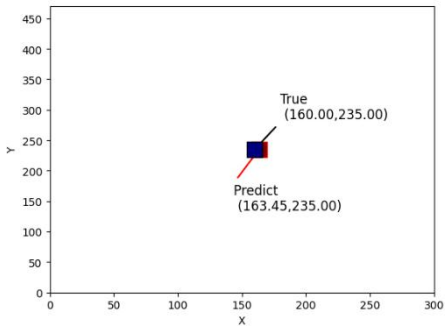


(f) Transformer

https://doi.org/10.37965/jdmd.2026.1675

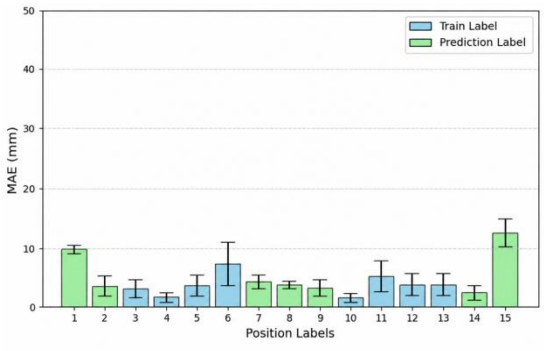


(g) PGGDL

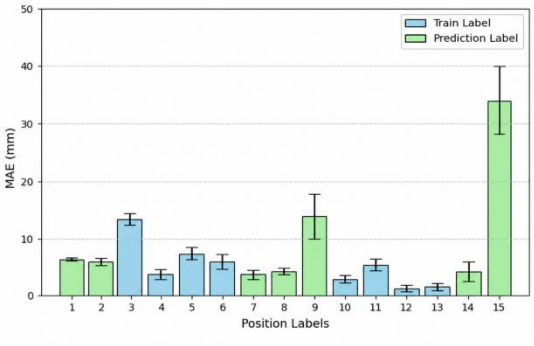


(h) PGGT

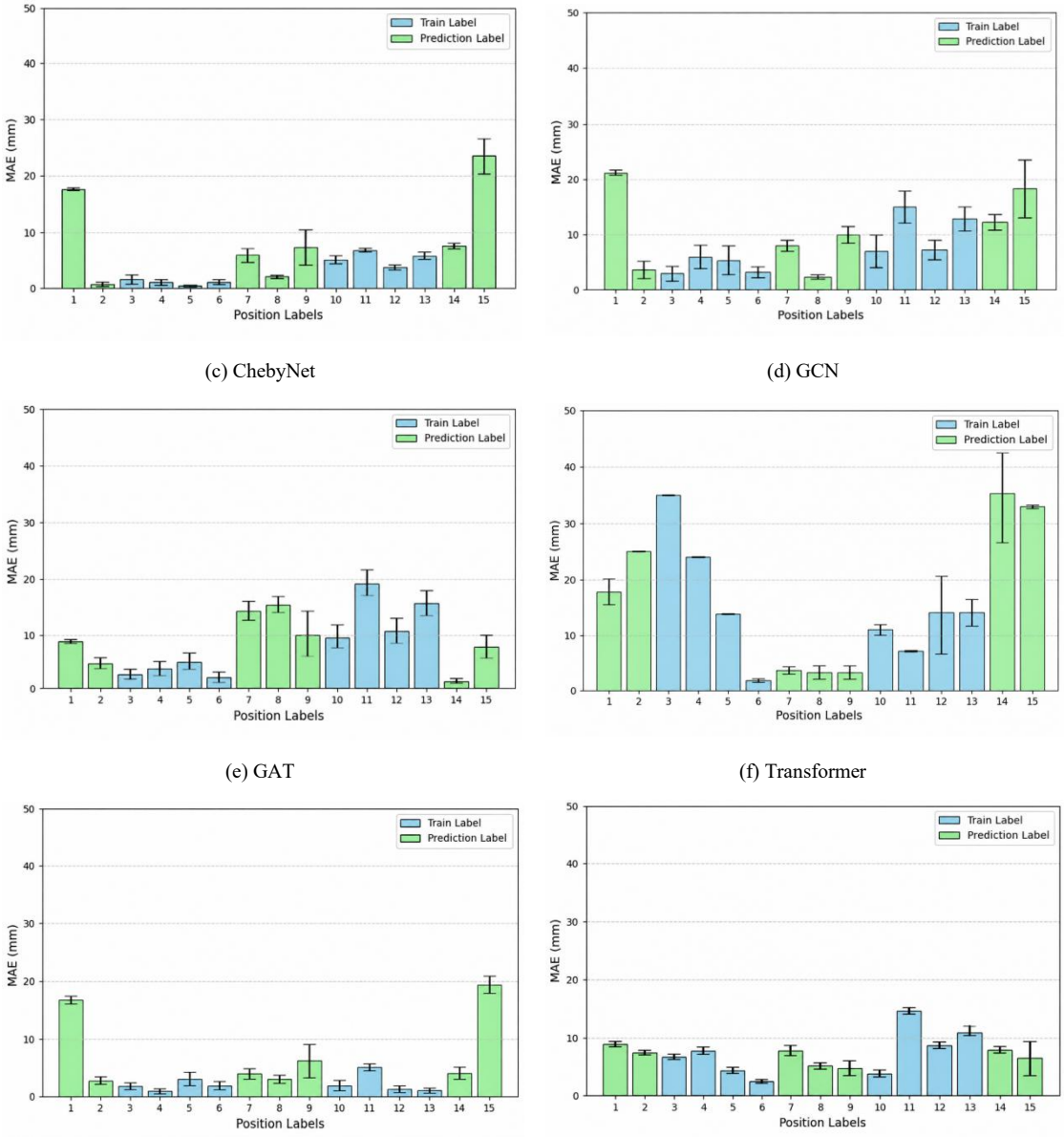
Fig. 11. Representative localization results for Case 4 using (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, and (h) PGGT. The blue and red rectangles denote the ground truth and predicted results, respectively



(a) VGG16



(b) ResNet18



(c) ChebyNet (d) GCN

(e) GAT (f) Transformer

(g) PGGDL (h) PGGT

Fig. 12. Mean absolute localization error at each damage position for the different models in Case 4: (a) VGG16, (b) ResNet18, (c) ChebyNet, (d) GCN, (e) GAT, (f) Transformer, (g) PGGDL, (h) PGGT. Bars denote the mean absolute error of the predicted axial damage coordinate at each position (unit: mm), and the error bars denote the 95% confidence interval of the mean; blue and green bars correspond to the training locations and to the locations held out for testing, respectively.

Compared with PGGT, other models generally suffer from insufficient generalization ability in handling edge positions. For instance, ChebyNet shows a 'high at both ends, low in the middle' distribution in unseen-location test error, with markedly higher unseen-location test errors at positions 1 and 15 (about 17 and 23, respectively), while the training-location errors are generally low, indicating poor generalization ability at edge positions. The error distribution of the GAT model is more complex, with the highest training-location error at position 11 (about 19), and higher unseen-location test errors at positions 1, 7, and 8 (about 9-14), with an unusually low error at position 14 (about 1), showing obvious position sensitivity. The GCN model is similar to ChebyNet, with higher unseen-location test

errors at positions 1 and 15, and training-location errors peaking in the range of positions 10 to 13, reflecting difficulties in generalizing to edge positions. PGGDL also shows higher unseen-location test errors at positions 1 and 15, with lower training-location errors, further verifying the particularity of edge positions. ResNet18 exhibits extremely uneven unseen-location test errors, with the training-location error at position 3 being relatively high, and the unseen-location test error at position 15 being unusually high, indicating serious adaptation issues. The Transformer model shows markedly higher training-location errors at positions 3, 4, and 14, and higher unseen-location test errors at positions 1, 2, and 14 as well, suggesting challenges in fitting and generalization. VGG16 performs relatively evenly, with moderate training-location error at position 6 and higher unseen-location test errors at positions 1 and 15, with a stable overall error distribution. The PGGT model shows a clear advantage in handling edge positions and provides more consistent predictions than the other models.

C. Ablation Study

To further investigate the effectiveness of the proposed PGGT framework, two types of ablation experiments were conducted. The first experiment analyzes the contribution of different network components, including CNN-based temporal feature extraction, GAT-based spatial dependency modeling, and Transformer-based global feature aggregation. The second experiment evaluates the influence of graph construction strategies by replacing the proposed physics-guided graph with alternative graph structures.

All ablation experiments were conducted under identical optimization settings and evaluation metrics. Only the network components or the graph construction strategies were modified, so that their individual contributions to the localization performance can be evaluated independently. For both ablation studies, the experiments were carried out on a subset of three representative damage locations (locations 7, 8 and 9 in Table 4), and the measurements of these locations were divided by a sample-wise split into a training subset and an evaluation subset, with approximately 70% of the measurements used for training and 30% for evaluation. Since all ablation variants were trained and evaluated on exactly the same subsets under identical optimization settings, the relative comparison among them is unaffected by this reduced protocol. Accordingly, the absolute error values reported in Tables 9 and 10 are not directly comparable with the accuracies given in Section 3.B, which were obtained with the complete four-case protocol described in Section 3.A.

1) Ablation of Network Components

The proposed PGGT framework consists of three key components: CNN-based local temporal feature extraction, GAT-based spatial topology modeling, and Transformer-based global dependency aggregation. To evaluate the contribution of each component, four degraded variants were constructed by removing or combining specific modules, including CNN_only, CNN_GAT, CNN_Transformer, and GAT_Transformer. The complete PGGT integrates all three components.

Table 9. Ablation results of different network components

Model	MAPE	MAE (mm)	RMSE (mm)
CNN_only	0.024±0.002	6.53±0.31	7.95±0.37
CNN_Transformer	0.023±0.002	6.12±0.27	7.62±0.32
CNN_GAT	0.022±0.001	5.86±0.24	7.13±0.35

GAT_Transformer	0.020±0.001	5.47±0.21	6.68±0.25
PGGT	**0.017±0.001**	**4.85±0.18**	**5.96±0.20**

As shown in Table 9, the complete PGGT model achieves the best localization performance among all variants. Compared with CNN_only, introducing GAT or Transformer improves the representation capability by incorporating spatial topology information or global dependency modeling.

The CNN_GAT and CNN_Transformer variants demonstrate that spatial topology information and global dependency modeling provide complementary improvements beyond local temporal feature extraction. However, individual spatial or global modeling alone cannot fully characterize the complex guided-wave propagation relationships.

The GAT_Transformer model further improves localization performance by integrating spatial topology modeling and global dependency learning. However, the absence of CNN limits its capability to extract local temporal characteristics from guided-wave signals.

The complete PGGT model achieves the best performance by jointly incorporating temporal, spatial, and global representations, demonstrating the complementary contributions of the three modules.

2) Ablation of Graph Construction Strategies

Although GAT can automatically learn the importance of node interactions, the quality of graph topology directly affects the propagation of information between signal-path nodes.

Therefore, additional experiments were conducted to investigate the influence of graph topology and neighborhood selection strategies. The proposed physics-guided graph was replaced by two alternative graph structures:

(1) kNN graph: each node is connected to its k = 5 nearest neighbors in the normalized propagation-path index space defined in Section 2, whereas the radius criterion and the Gaussian attenuation weighting adopted by the physics-guided graph are not applied; the edge weight is defined as $w_{ij} = \exp(-d_{ij})$, where d_{ij} is the distance between the propagation-path indices of nodes i and j given in Eq. (2).

(2) Fully connected graph (FC): all nodes are connected without topology restrictions.

For a fair comparison, the network architecture and training parameters were kept unchanged, and only the graph construction strategy was modified.

For the physics-guided graph, the radius threshold was determined according to the spatial relationship among signal-path nodes, where different neighborhood ranges were evaluated to balance local physical interactions and graph connectivity; the configurations examined were r = 2.0, 2.5, 3.0, 3.5, 4.0 and 5.0 combined with σ= 1.0, 1.5 and 2.0. Based on the validation results of different radius configurations, the configuration r=3.5 and σ=1.0, expressed in the normalized propagation-path index coordinate system defined in Section 2, was selected as the optimal setting and used for subsequent graph comparison experiments. The radius parameter defines the neighborhood range among signal-path nodes. A smaller radius may restrict information exchange between physically related propagation paths, whereas an excessively large radius may introduce redundant connections and weaken the physical interpretability of the graph structure. Therefore, the selected radius provides a balance between local guided-wave propagation relationships and sufficient graph connectivity.

Table 10. Comparison of different graph construction strategies

Graph Construction	Localization Error (mm)	Localization RMSE (mm)

FC graph	2.40±0.56	2.89±0.34
kNN graph	2.27±0.50	2.77±0.47
Physics-guided graph	**1.46±0.37**	**2.17±0.35**

Localization Error represents the mean absolute localization distance between the predicted and actual damage coordinates and is expressed in millimetres; Localization RMSE is the corresponding root-mean-square value.

As shown in Table 10, the proposed physics-guided graph achieves the best localization performance among all graph construction strategies.

Compared with the kNN graph, the physics-guided graph reduces the localization error from 2.27 to 1.46. This improvement indicates that fixing the number of neighbors without the radius criterion and the attenuation weighting derived from guided-wave propagation cannot fully represent the propagation relationships of guided waves. The improvement verifies that incorporating prior knowledge of actuator–receiver propagation paths enables the graph structure to better preserve physically meaningful interactions among signal-path nodes.

Compared with the fully connected graph, the proposed graph achieves a lower localization error, indicating that unrestricted connections may introduce irrelevant interactions between signal-path nodes.

The superior performance of the physics-guided graph verifies that incorporating physical prior knowledge into graph construction can provide more meaningful node relationships and improve the localization capability of PGGT.

These results demonstrate that the proposed graph construction strategy does not simply increase the connectivity of the graph, but introduces physically meaningful relationships derived from guided-wave propagation characteristics, thereby improving localization performance.

4.Conclusion

This paper presents a physics-guided graph Transformer (PGGT) framework for guided-wave-based damage detection and localization in composite pipelines under limited labeled damage locations. Unlike conventional data-driven approaches that directly learn from signal responses, the proposed method incorporates physical propagation information by representing actuator–receiver guided-wave propagation paths as graph nodes and constructing a physics-guided graph according to the sensor configuration and guided-wave propagation characteristics. Within this framework, the CNN-based module extracts local temporal characteristics from guided-wave responses, the GAT-based module captures topology relationships among signal-path nodes, and the Transformer-based module models long-range global dependencies among propagation paths. The complementary fusion of temporal, spatial, and global representations enables accurate damage localization.

The proposed PGGT framework is evaluated under four training–testing split strategies to investigate its generalization capability for unseen damage locations. Compared with VGG16, ResNet18, ChebyNet, GCN, GAT, Transformer, and PGGDL, the proposed method achieves superior damage localization performance in terms of MAPE, MAE, and RMSE. In the most challenging case with fewer training locations and unseen central and boundary damage positions, PGGT achieves a MAPE of 0.035 (a relative error of 3.5%), an MAE of 5.38 mm, and an RMSE of 6.23 mm. Furthermore, the ablation studies demonstrate the complementary contributions of temporal feature

extraction, spatial topology modeling, and global dependency learning. The graph construction comparison further verifies that the physics-guided graph provides more meaningful propagation relationships than purely data-driven graph structures.

Although the proposed method demonstrates promising performance for guided-wave-based damage detection and localization, several limitations remain. The current validation is conducted on a controlled composite pipeline specimen with a fixed sensor arrangement and single-damage scenarios. Future studies will focus on extending the proposed framework to more complex structural configurations, different sensor layouts, multiple-damage conditions, and practical engineering environments with stronger noise interference.

Acknowledgments

This work is supported by Research on Structural Damage Monitoring Methods and Safety Warning of Mobile Pressure Vessels for High-Pressure Hydrogen (Grant No. 2025KJ60) and Digital Twin-Based Health Monitoring and Life Assessment of Coke Drums (Grant No. F80025S006).

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

References

[1] D. Li, Y. Bai, C. Wang, N. Su, L. Wu, D. Gu, Y. Zhai, and P. Luo, “Graphene nanomaterials in oil and gas industry: Current status and future perspectives,” Renewable and Sustainable Energy Reviews, vol. 226, p. 116342, 2026, doi: 10.1016/j.rser.2025.116342.

[2] M. Cheng, W. Zhang, H. Li, J. Tian, M. Zheng, and H. Ma, “Quantitative detection method for small defects in high-attenuation polyethylene pipes based on ultrasonic guided waves and chaos sensitivity,” Thin-Walled Structures, vol. 224, p. 114700, 2026, doi: 10.1016/j.tws.2026.114700.

[3] F. Hou, Y. Chen, Y. Yan, R. Liu, R. Li, Y. Liu, and R. Zhong, “Structural response and damage assessment method for subsea pipe-in-pipe subjected to anchor impact,” Marine Structures, vol. 99, p. 103714, 2025, doi: 10.1016/j.marstruc.2024.103714.

[4] Y. Zhang, X. Wu, Q. Guo, D. Zhang, C. Li, D. Li, Y. Liu, J. Zhang, P. Zhang, Y. Yan, L. An, J. Guo, and L. Chen, “Advances in sensors technologies for composites structural health monitoring,” Composite Structures, vol. 370, p. 119448, 2025, doi: 10.1016/j.compstruct.2025.119448.

[5] M. Waqar, A. M. Memon, M. Sabih, and L. M. Alhems, “Composite pipelines: Analyzing defects and advancements in non-destructive testing techniques,” Engineering Failure Analysis, vol. 157, p. 107914, 2024, doi: 10.1016/j.engfailanal.2023.107914.

[6] X. Zang, Z.-D. Xu, H. Lu, C. Zhu, and Z. Zhang, “Ultrasonic guided wave techniques and applications in pipeline defect detection: A review,” International Journal of Pressure Vessels and Piping, vol. 206, p. 105033, 2023, doi: 10.1016/j.ijpvp.2023.105033.

[7] H. Zhang, F. Teng, J. Wei, S. Lv, L. Zhang, F. Zhang, and M. Jiang, “Damage Location Method of Pipeline Structure by Ultrasonic Guided Wave Based on Probability Fusion,” IEEE Transactions on Instrumentation and Measurement, vol. 73, p. 9504914, 2024, doi: 10.1109/TIM.2024.3365164.

[8] A. De Luca, D. Perfetto, A. De Fenza, G. Petrone, and F. Caputo, “Guided wave SHM system for damage detection in complex composite structure,” Theoretical and Applied Fracture Mechanics, vol. 105, p. 102408, 2020, doi: 10.1016/j.tafmec.2019.102408.

[9] A. Diab and T. Nestorović, "Numerical Investigation of the Time-of-Flight and Wave Energy Dependent Hybrid Method for Structural Damage Detection," Journal of Vibration Engineering & Technologies, vol. 11, pp. 2689–2707, 2023, doi: 10.1007/s42417-023-01025-0.

[10] Y. K. Chen, N. Bakhary, K. H. Padil, M. F. Shamsudin, A. Ahmad, N. H. Noh, and N. Norazahar, "Efficient residual reliability criterion index in a permanent guided wave monitoring system," Measurement, vol. 197, p. 111292, 2022, doi: 10.1016/j.measurement.2022.111292.

[11] X. Zhang, Z. Tang, F. Lv, and X. Pan, "Helical comb magnetostrictive patch transducers for inspecting spiral welded pipes using flexural guided waves," Ultrasonics, vol. 74, pp. 1–10, 2017, doi: 10.1016/j.ultras.2016.08.024.

[12] R. K. Munian, D. R. Mahapatra, and S. Gopalakrishnan, "Ultrasonic guided wave scattering due to delamination in curved composite structures," Composite Structures, vol. 239, p. 111987, 2020, doi: 10.1016/j.compstruct.2020.111987.

[13] X. Xu and C. Liu, "Physics-guided deep learning for damage detection in CFRP composite structures," Composite Structures, vol. 331, p. 117889, 2024, doi: 10.1016/j.compstruct.2024.117889.

[14] R. Tang, S. Zhang, W. Wu, S. Zhang, and Z. Han, "Explainable deep learning based ultrasonic guided wave pipe crack identification method," Measurement, vol. 206, p. 112277, 2023, doi: 10.1016/j.measurement.2022.112277.

[15] J. Wu, Y. Yang, Z. Lin, Y. Lin, Y. Wang, W. Zhang, and H. Ma, "Weak ultrasonic guided wave signal recognition based on one-dimensional convolutional neural network denoising autoencoder and its application to small defect detection in pipelines," Measurement, vol. 242, p. 116234, 2025, doi: 10.1016/j.measurement.2024.116234.

[16] G. R. B. Ferreira, M. G. de Castro Ribeiro, A. C. Kubrusly, and H. V. H. Ayala, "Improved feature extraction of guided wave signals for defect detection in welded thermoplastic composite joints," Measurement, vol. 198, p. 111372, 2022, doi: 10.1016/j.measurement.2022.111372.

[17] S. Ning, Y. Ren, and Y. Wu, "Intelligent fault diagnosis of rolling bearings based on the visibility algorithm and graph neural networks," Journal of the Brazilian Society of Mechanical Sciences and Engineering, vol. 45, p. 72, 2023, doi: 10.1007/s40430-022-03913-0.

[18] S. Wang, Z. Luo, P. Shen, H. Zhang, and Z. Ni, "Graph-in-Graph Convolutional Network for Ultrasonic Guided Wave-Based Damage Detection and Localization," IEEE Transactions on Instrumentation and Measurement, vol. 71, p. 2502011, 2022, doi: 10.1109/TIM.2022.3144732.

[19] M. Xu, B. Zhang, M. Cao, H. Yu, and C. Wang, "LSEGNN: Encode Local Topology Structure in Graph Neural Networks," in 2022 IEEE International Performance, Computing, and Communications Conference (IPCCC), 2022, pp. 49–56, doi: 10.1109/IPCCC55026.2022.9894311.

[20] P. Zhan, X. Qin, Q. Zhang, and Y. Sun, "A Novel Structural Damage Detection Method via Multisensor Spatial–Temporal Graph-Based Features and Deep Graph Convolutional Network," IEEE Transactions on Instrumentation and Measurement, vol. 72, p. 2504814, 2023, doi: 10.1109/TIM.2023.3238048.

[21] Y. Ding and M. Jia, "Convolutional Transformer: An Enhanced Attention Mechanism Architecture for Remaining Useful Life Estimation of Bearings," IEEE Transactions on Instrumentation and Measurement, vol. 71, p. 3515010, 2022, doi: 10.1109/TIM.2022.3181933.

[22] S. Khan, F. S. Khan, A. Vaswani, N. Parmar, M.-H. Yang, and M. Shah, "Introduction to the Special Section on Transformer Models in Vision," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 11, pp. 12721–12725, Nov. 2023, doi: 10.1109/TPAMI.2023.3306164.

[23] X. Jiang, Y. Hu, S. Cao, F. Cui, F. Li, Y. Gao, and F.-Z. Xuan, "Physical-guided graph deep learning for composite pipelines structural health monitoring," Structural Health Monitoring, published online Sep. 12, 2025, doi: 10.1177/14759217251368998.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on

Computer Vision and Pattern Recognition (CVPR), Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[26] M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” in Advances in Neural Information Processing Systems 29 (NIPS 2016), 2016, pp. 3844–3852.

[27] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” arXiv preprint arXiv:1609.02907, 2017.

[28] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” arXiv preprint arXiv:1710.10903, 2018.