Original Article

Structural Health Monitoring
2026, Vol. 25(2) 1219–1236

DOI: 10.1177/14759217241305050
journals.sagepub.com/home/shm
Sage

Physics-guided neural network for structural health monitoring with lamb waves through boundary reflection elimination

Yang Song[1,2], **Shengbo Shan**[3], **Yuanman Zhang**[1,2] and **Li Cheng**[1,2]

Abstract

Lamb-wave-based structural health monitoring (SHM) technology for damage location in plate-like structures relies on the postprocessing of captured signals after interacting with damage. Traditional methods typically leverage the time of flight (ToF) of scattered waves from damage. However, these methods are prone to reflected waves from structural boundaries which mix with scattered waves from damage. This is a vital problem faced by most ToF-based detection methods, which seriously narrows the inspection area. To tackle this problem, a machine learning framework, consisting of a multiscale spatiotemporal (MSST) fusion network, is proposed to facilitate the accurate extraction of the ToF of scattered waves through eliminating the influence of boundary reflections. Experiments are conducted with the time-domain Lamb wave signals recorded by a tactically designed piezoelectric sensor array on a 2-mm-thick Al-6061 plate. A pair of circle magnets is attached onto the plate as the wave reflectors. Through step-by-step moving of the magnets in the predefined grids, the corresponding Lamb wave signals are measured to construct a database. An MSST is subsequently designed to minimize the error between estimated and theoretical ToFs, with wavelet coefficients of the signals and transducer position as inputs. The model is trained with the Adam algorithm where 80% of samples in the database are used for training and the rest for evaluation. The final validations are conducted with the scatters off the predefined grids. Results demonstrate that the designed neural network architecture can effectively eliminate boundary reflections and enable precise ToF extraction of the scattered waves from damage. This allows the enlargement of the detection area and presents a promising and useful tool for enhancing the detection performance of existing SHM methods in complex structures.

Keywords
Lamb waves, boundary reflection, time of flight, artificial intelligence, structural health monitoring

Introduction

The online and real-time monitoring of engineering structures is of paramount importance to ensure their operational safety, which hatches out the need for structural health monitoring (SHM) techniques.[1–5] Among various existing SHM methodologies, Lamb-wave-based method holds particular promise for thin-walled structures, owing to its high sensitivity to small and even invisible damage,[6,7] conducive to early-stage detection of defects, so as to facilitate proactive and timely maintenance decisions.[8–10]

Lamb-wave-based SHM relies on a network of strategically placed sensors across a structure, with actuators generating Lamb waves that interact with

---

[1] Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong
[2] Research Institute for Artificial Intelligence of Things, The Hong Kong Polytechnic University, Kowloon, Hong Kong
[3] School of Aerospace Engineering and Applied Mechanics, Tongji University, Shanghai, PR China

Corresponding author:
Shengbo Shan, School of Aerospace Engineering and Applied Mechanics, Tongji University, 100 Zhangwu Road, Yangpu District, Shanghai, PR China.
Email: shanshengbo@tongji.edu.cn

Li Cheng, Department of Mechanical Engineering, The Hong Kong Polytechnic University, 11 Yuk Choi Road, Hung Hom, Kowloon, 999077, Hong Kong.
Email: li.cheng@polyu.edu.hk

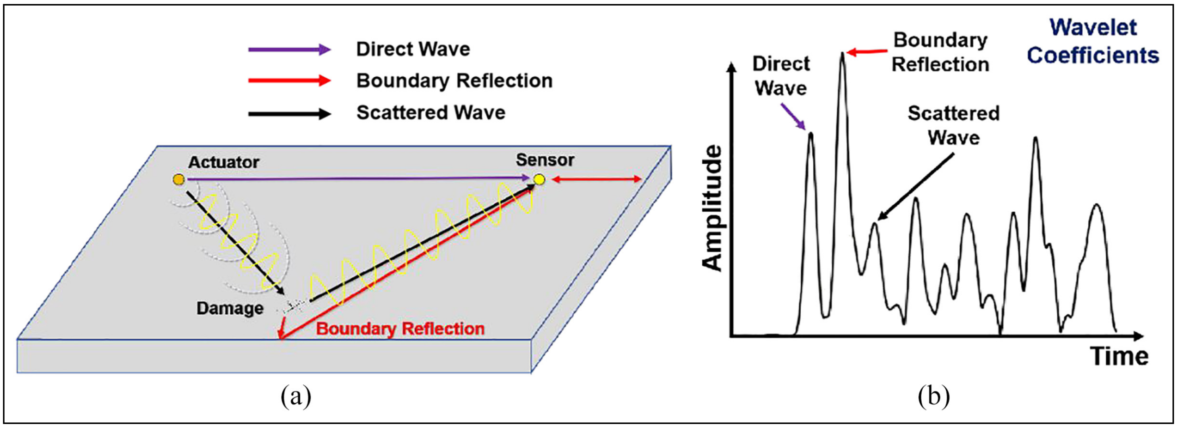

Figure 1. Boundary reflections on scattered waves: (a) sketch of wave propagation paths in a plate with damage and (b) typical wavelet coefficients of a time-domain signal after wavelet transformation.

potential damage and sensors capturing the resultant waves. Detecting and locating damage often involves baseline comparisons to underscore their impact.[11,12] The time of flight (ToF) of Lamb waves scattered by damage is commonly leveraged in various detection and localization algorithms.[13,14] Accordingly, a few classical algorithms, such as delay-and-sum imaging,[15] path imaging,[16] time-inverted-focus imaging,[17,18] spatial wave number domain filtering imaging,[19–23] and multiple signal classification imaging,[24–26] have been developed. All these methods rely on a reliable and accurate estimation of the ToF. However, practical applications face challenges when damage occurs near structural boundaries, causing interference between scattered waves and boundary reflections, as sketched in Figure 1. Specifically, it becomes difficult to discern the ToF of Lamb waves scattered by damage from that caused by boundary reflections. This compromises the accuracy of these methods and limits their effective inspection area.

To tackle this problem, an intuitive solution is to put absorbing materials on structural boundaries. For example, Salmanpour et al.[27] applied the packaging foam to a plate to reduce boundary reflections. However, it is usually difficult to achieve a perfect elimination of boundary reflections with damping materials. In addition, the installation of absorbing materials can be hardly applied in some engineering applications where structural weight needs to be strictly controlled. By contrast, a more realistic way is to scrutinize these complex signals and extract ToF from the signal processing viewpoint.

Separating overlapped boundary reflected and damage scattered waves in time domain often requires an expert construction of a physics-based model, albeit very difficult due to complex types of damage, multiple guided wave modes, and various boundary reflection paths. Recent advances in machine learning, particularly through hybrid approaches like physics-informed neural networks (PINNs),[28–31] have emerged as innovative alternatives in various applications. PINNs uniquely combine physical laws represented by partial differential equations (PDEs) with neural networks, enhancing model accuracy and reducing reliance on extensive datasets. In particular, the work by Shukla et al.[32] demonstrates the practical application of PINNs for the nondestructive quantification of surface-breaking cracks in metal plates. Their approach effectively integrates ultrasonic imaging data with physics-based models, significantly improving detection accuracy while minimizing the need for large datasets. This highlights the transformative potential of physics-informed machine learning in the field of SHM. More generally, deep learning enables the extraction and processing of multiple signal features from the inputs, allowing nonlinear mapping between input time-domain signals and damage locations.[33,34] For example, Zhang et al.[35] used a one-dimensional convolutional neural network (1D CNN) to correlate the time-varying damage index directly with the damage location, which allows an accurate localization of damage in plates with a few transducers. Additionally, a kernel regression method which incorporates deep metric learning was used to locate damage in metallic plates based on linear guided waves.[36] Wang et al.[37] proposed a real-time guided wave imaging method using a CNN for quantitative corrosion evaluation, alongside the establishment of the relationship between the wave signals and the velocity map. Rautela and Gopalakrishnan[38] assessed some typical machine learning frameworks, including dense neural networks, 1D CNNs, recurrent neural networks, and long short-term memory. The study assessed the robustness of the models with noisy datasets and their potential for real-time applications. Moreover, Sbarufatti et al.[39] utilized a multilayer perceptron (MLP) to localize a crack in an aluminum plate. The model was trained using cross-correlations between the simulation data and in-field measurements. By artificially incorporating noise into the training data, the model exhibited strong generalization to experimental data. For composite structures, Lin et al.[40] also applied a CNN model to detect and locate defects in a composite wing. Heesch et al.[41] presented a generative adversarial network architecture to generate guided wave signals with the open guided waves database. Liu et al.[42] presented a novel approach using a structural digital twin for damage detection in carbon fiber-reinforced plastics (CFRP) composites, leveraging a meta transfer learning framework. This method addresses the challenges of limited training data by generating virtual data under various damage conditions, significantly enhancing the accuracy of damage localization and classification. Xu and Liu[43] introduced a physics-guided CNN, which combines physical models and data-driven techniques to achieve over 90% accuracy in detecting damage in CFRP composites across varying structures and conditions, further underscoring the effectiveness of hybrid models in SHM. These studies signify the potential of machine learning as a transformative tool in the field of SHM, offering improved capabilities and new perspectives to damage detection and localization.

Despite these successful attempts of machine learning in SHM, prevalent issues persist in many existing methods. A significant challenge lies in excessive reliance on end-to-end learning, which may overlook some critical information in intermediate processes and lead to a lack of model interpretability. This deficiency could hinder the widespread adoption of machine learning methods in practical engineering applications. This work introduces a novel neural network architecture tailored for SHM, incorporating a preprocessing step with wavelet transforms. This step identifies the appropriate domain for signal analysis, enhancing the extraction of ToF from scattered waves while minimizing boundary reflections. The proposed method eliminates the need for baselines and effectively interrogates both the temporal information of Lamb wave signals and the spatial information of sensor coordinates, facilitating more efficient learning of mapping relationships. A multiscale module is introduced to represent input time-domain signals at different levels comprehensively. Deconvolution is applied to interpret the results of the multiple-scale module, revealing its effectiveness in eliminating unnecessary boundary reflections. Experimental validations are finally carried out to demonstrate the effectiveness of the proposed method for precise ToF extraction, which is subsequently used for damage localization.

This paper is organized as follows. The proposed method is presented in "The MSST fusion network" section in terms of the multiscale spatiotemporal (MSST) fusion network. "Experiment setup and analysis" section describes the experimental setup and analyses the interpretability of the network. The results of damage localization are presented in "Applications" section. Conclusions are summarized in "Conclusions" section.

The MSST fusion network

Prior to the construction of the MSST fusion model, methods for signal generation, acquisition, and preprocessing are briefly introduced. Lamb wave signals are generated and captured using PZTs on a plate, used as an illustrative example to exemplify a thin-walled structure. As the preprocessing of the captured time-domain signals, the wavelet transform is used to extract the arrival time of different wave packets,[44] which is termed as the signal temporal information (STI). To extract the envelope of the response signal, the complex Morlet transform is utilized. The coefficients of the wavelet transform are given by

$$\mathrm{CWT}_f(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t)\psi^*\left(\frac{t-b}{a}\right)\mathrm{d}t \tag{1}$$

where the mother wavelet $\psi(t)$ is formulated as

$$\psi(t) = \frac{1}{\sqrt{\pi\gamma}} e^{j\omega_0 t} e^{-t^2/\gamma} \tag{2}$$

where a and b are the scale and shift parameters, respectively. $\psi^*(t)$ represents the complex conjugate of the Morlet wavelet function. Here, ω_0 is the angular frequency, and γ defines the width of the Gaussian window. This analysis emphasizes the modulus at a constant scale to align with the central frequency of the excitation signal. The narrowband nature of the complex Morlet wavelet can effectively reduce noise interference.

Meanwhile, the coordinates of the PZTs and the structural dimensions are determined a priori, which are called signal spatial information (SSI) hereafter. The STI and SSI are further synthesized in a fully connected module in the designed network. In general, with the input of the extracted temporal wavelet coefficients, the proposed network aims at eliminating the influence of boundary reflections to precisely extract the ToF of the scattered waves induced by the damage.

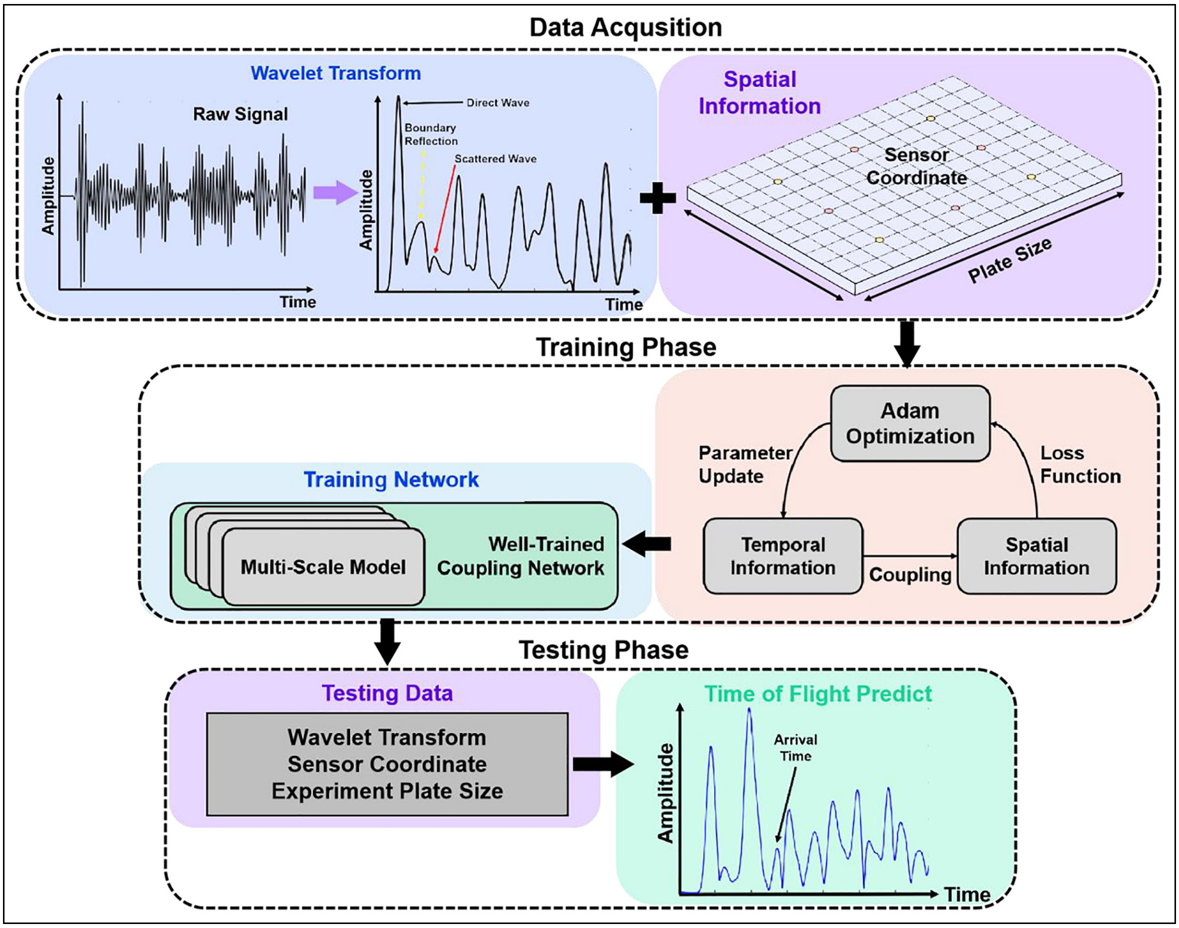

Figure 2. Flowchart of the proposed method.

The flowchart of the proposed method is illustrated in Figure 2.

Network architecture

Assuming that there are m actuator-sensor paths, the corresponding time-domain signals are processed with the wavelet transform to extract their STI. Simultaneously, the coordinates of the actuators and sensors, along with the dimensions of the structure, constitute the SSI. Both STI and SSI form an input feature set $\mathbf{x}^{(i)} \in \mathbb{R}^{1\times m}$. Correspondingly, the ToF of scattered waves from damage is used as the output, which is labeled as $\mathbf{y}^{(i)} \in \mathbb{R}^{1\times 1}$. In the context of machine learning, the model essentially functions as a parametric regression model, trained to approximate a set of parameters representing the nonlinear mapping relationship $F: \mathbf{x}^{(i)} \rightarrow \mathbf{y}^{(i)}$. The training process involves optimizing a mean squared error (MSE) loss function through parameter updates using stochastic gradient descent (SGD) method.[45] The specific expression is given as follows:

$$L(\mathbf{H}) = \frac{1}{M}\sum_{i=1}^{M}\left\|\mathbf{y}^{(i)} - F\left(\mathbf{x}^{(i)};\mathbf{H}\right)\right\|_2^2 \tag{3}$$

where $\mathbf{H}$ represents the set of parameters to be optimized.

The overall network architecture consists of two main components: a multiscale module and a fully connected module, as illustrated in Figure 3. In the first part, the temporal signals undergo a holistic convolutional layer with 256 filters, followed by a normalization and max-pooling layer. Subsequently, four parallel convolutional layers with different kernel sizes, comprising 32, 64, 64, and 128 filters respectively, are applied.

The features extracted from the first part are then flattened and concatenated with spatial information in the second part. This concatenated tensor is fed into two fully connected layers. The first layer consists of 64 neurons with rectified linear unit (ReLU) activation function and a dropout rate of 0.01. The second layer serves as the output layer with a single neuron and linear activation function. The entire network is optimized

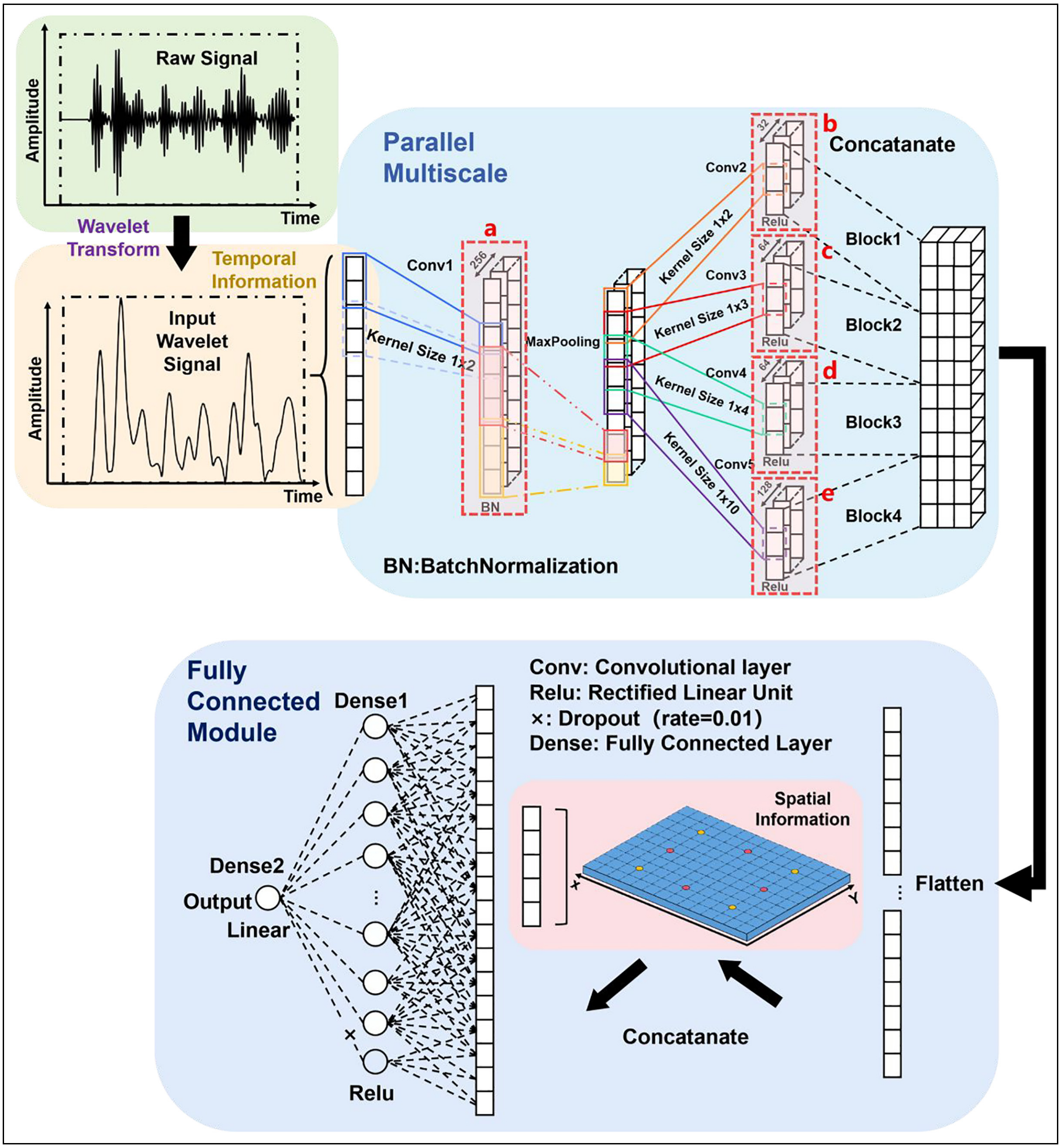

Figure 3. Architecture of the proposed spatiotemporal fusion with a multiscale parallel neural network.

using the Adam optimization algorithm[46] with a learning rate of 2×10^{-6}.

One-dimensional convolutional layers, commonly employed in signal processing and time-series data analysis, are also adopted in the proposed network. These layers apply convolution operations along a singular dimension, adept at capturing temporal dependencies and patterns while concurrently extracting temporal features.[47] The one-dimensional convolution operation is mathematically expressed as follows:

$$(\mathbf{X}\times\boldsymbol{\varphi})_i = \sum\nolimits_{k=1}^{K} \mathbf{X}_{i+k-1}\boldsymbol{\varphi}_k + \tau_i \tag{4}$$

In the equation, $\mathbf{X}$ represents the input sequence, $\boldsymbol{\varphi}$ denotes the one-dimensional convolutional filter, τ is the bias term, and $\sum$ signifies the summation over the convolutional filter size K.

Usually, a single-scale convolutional network faces challenges in capturing diverse and multiscale temporal characteristics. Therefore, within the proposed network architecture, a multiscale module has been meticulously designed specifically for extracting features from time-domain signals after the wavelet transformation, as depicted in Figure 3. This module intends to enhance the network's ability to discern subtle patterns in the input data. By employing convolutional layers with kernel sizes corresponding to scales 2, 3, 4, and 10, the multiscale convolutional module (MCM) facilitates feature extraction at different resolutions. For a given input set $\mathbf{X}$, the MCM processes $\mathbf{X}$ through a set of convolutional layers, each associated with a specific scale. By denoting the convolutional operation at the ith scale as Conv_i and applying the $\mathrm{ReLU}f(x) = \max(0, x)$ as activation function,[48] the module's output O is expressed as:

$$\mathbf{O} = \mathrm{ReLU}(\mathrm{Conv}_2(\mathbf{X})) \oplus \mathrm{ReLU}(\mathrm{Conv}_3(\mathbf{X})) \oplus \mathrm{ReLU}(\mathrm{Conv}_4(\mathbf{X})) \oplus \mathrm{ReLU}(\mathrm{Conv}_{10}(\mathbf{X})). \tag{5}$$

The $\oplus$ operator concatenates the outputs along the channel dimension. These convolutional operations capture features at various scales, enabling the network to distinguish intricate details.

In the lead-up to the multiscale convolutional network, we introduce an initial global convolution operation, succeeded by batch normalization (BN)[49] and max pooling layers.[50] This combination aims at refining the feature extraction and overall performance. BN normalizes convolutional layer outputs to mitigate the vanishing gradient for improved stability and faster convergence as

$$\mathrm{BN}(\mathbf{x}) = \gamma\left(\frac{\mathbf{x}_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}\right) + \beta \tag{6}$$

where $\mathbf{x}_i$ represents the input, μ_B and σ_B are the mean and standard deviation, and γ and β are learnable parameters, with ϵ preventing division by zero. BN accelerates training, augments stability, and improves generalization.

Max pooling reduces spatial dimensions by selecting maximum values within each input window, aiming at decreasing complexity and enhancing model robustness to translations. Each window in Max Pooling selects only one maximum value to focus on crucial information. This integrated design not only elevates network performance but also addresses gradient challenges, enhances generalization, and accelerates training, which provides a robust foundation for diverse tasks and datasets.

Following the multiscale convolutional module, the output undergoes flattening and concatenation with spatial information to merge temporal and spatial features. This concatenated information is then processed through an MLP[51] for precise ToF estimation. The flattening operation transforms temporal features into a one-dimensional vector, enhancing the network's capacity for feature fusion. Algebraically, with flattened temporal features $\mathbf{X}_\mathrm{f}$ and spatial features $\mathbf{X}_\mathrm{s}$, the merged feature is expressed as:

$$\mathbf{X}_\mathrm{merged} = \mathbf{X}_\mathrm{f} \oplus \mathbf{X}_\mathrm{s}. \tag{7}$$

The subsequent MLP performs a nonlinear transformation:

$$\mathrm{MLP}(\mathbf{X}_\mathrm{merged}) = \sigma(\mathbf{W}_2\sigma(\mathbf{W}_1\mathbf{X}_\mathrm{merged} + \mathbf{b}_1) + \mathbf{b}_2) \tag{8}$$

where σ represents the activation function (ReLU), and $\mathbf{W}_1, \mathbf{W}_2$, and $\mathbf{b}_1$, $\mathbf{b}_2$ denote the weight matrices and bias vectors, respectively. This design leverages flattening and concatenation for effective feature fusion to enhance the efficiency of the designed network.

After constructing the architecture and initializing its parameters randomly, we employ SGD to iteratively update the parameters. Following each gradient descent step, the Adam optimizer, known for adaptive learning based on gradients, is applied to update all parameters to achieve swift convergence of results. The updating rule for each parameter is expressed as follows:

$$\theta \leftarrow \theta - \alpha \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t + \epsilon}} \tag{9}$$

where α denotes the learning rate, and ε is a small constant introduced for numerical stability. Adam maintains two moving averages for each parameter θ:

$$\mathbf{m}_t = \beta_1\mathbf{m}_{t-1} + (1 - \beta_1)\mathbf{g}_t \tag{10}$$

$$\mathbf{v}_t = \beta_2 \cdot \mathbf{v}_{t-1} + (1 - \beta_2)\mathbf{g}_t^2 \tag{11}$$

where $\mathbf{m}_t$ is the first moment estimate, $\mathbf{v}_t$ is the second raw moment estimate, $\mathbf{g}_t$ is the gradient at time t, and β_1 and β_2 are the exponential decay rates for the moment estimates. The bias-corrected estimates are calculated as:

$$\hat{\mathbf{m}}_t = \frac{\mathbf{m}_t}{1 - \beta_1^t} \tag{12}$$

$$\hat{\mathbf{v}}_t = \frac{\mathbf{v}_t}{1 - \beta_2^t}. \tag{13}$$

Here, β_1^t and β_2^t represent the tth power of β_1 and β_2, respectively. The adaptive learning rate in the Adam optimizer enables efficient adjustment of step sizes for various parameters, striking a balance between adaptability and stability throughout the optimization process.

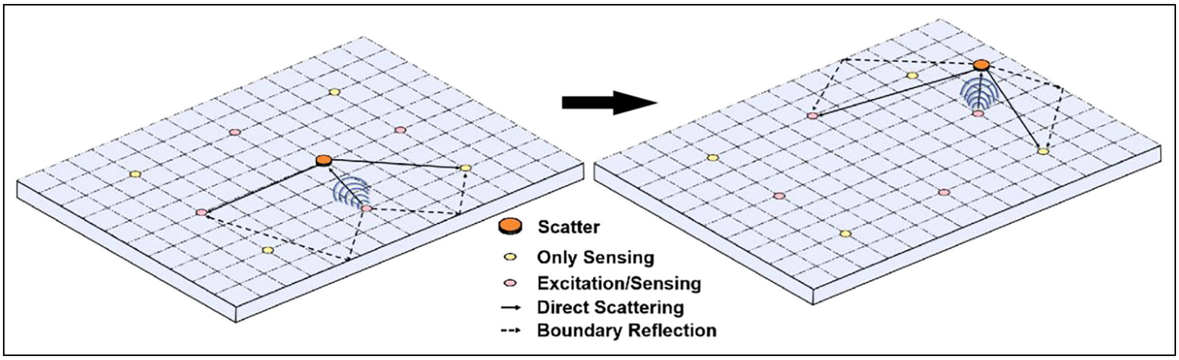

Figure 4. Schematic illustration of the data collection method.

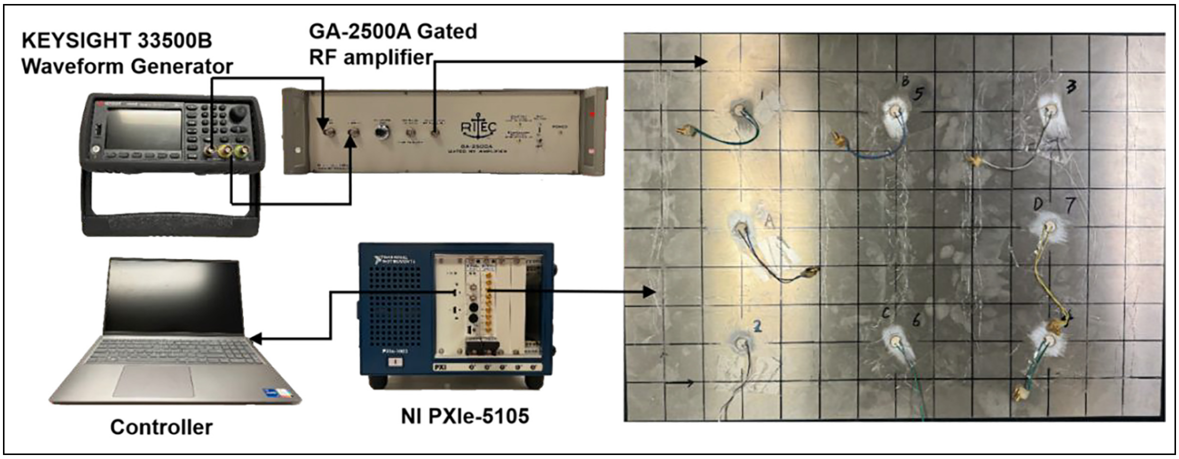

Figure 5. Experimental setup.

In the training process, the parameters in the designed network undergo updates for multiple epochs until convergence. After that, the efficacy of the network is assessed through practical testing. This entails evaluating the network's performance on a dataset that was not part of the training data.

Experimental setup and analysis

Experimental setup

In this work, guided wave data are experimentally measured from a typical aluminum plate with a dimension of 500 × 700 × 2 mm. Eight PZT transducers were mounted on the surface of the plate by epoxy resin to actuate and receive Lamb waves. For ease of presentation, the plate is divided into 140 uniform regions, as shown in Figure 4. The positions for placing scatterers are chosen at the centers of every four-unit region. Aiming at diversifying the datasets, four PZTs marked by the yellow points in Figure 4 are solely used for sensing, while those at the pink points serving as actuators and sensors as needed.

A pair of circular magnets is attached to the plate as the scatterer to simulate the damage. For the same scatterer, different excitation channels produce different boundary reflection paths as a way to increase the diversity of the datasets. Four excitation channels are selected in the present case, resulting in four data sets to assess the proposed method. Each signal in the data sets includes both direct waves and boundary reflected waves. The collected data are divided with 80% for training and 20% for testing purposes.

The experimental system works as follows, as shown in Figure 5: a KEYSIGHT 33500B waveform generator is used to output the excitation signal. A power amplifier is used to amplify the excitation voltage for the PZT actuator. The sensor captures and records the propagating Lamb waves by the NI PXIe-5105 data acquisition module. The excitation signal is a Hann-windowed sinusoidal tone burst for all sets. The central frequency is selected with the following considerations: (1) the amplitude of response signals should be as high as possible, (2) the wavelength should be comparable to the size of the damage so that the damage can serve as a strong scatter, and (3) the waves should be as clean

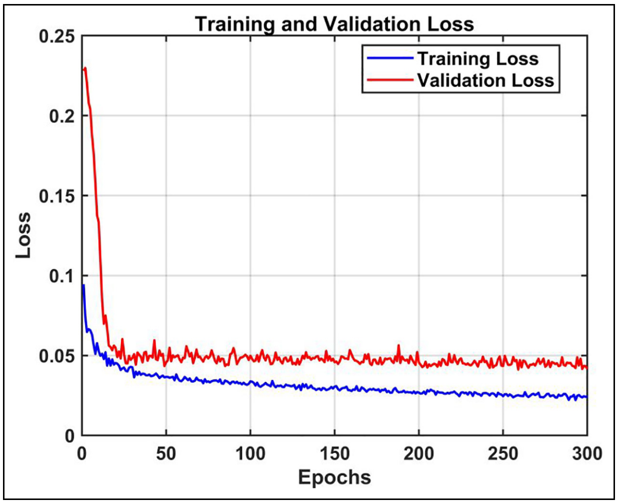

Figure 6. Training and validation loss curves.

as possible, that is, the waves with fewer modes are preferable. Guided by these principles, the excitation frequency is selected as 160 kHz. Examining the waves in the response signals, the A0 mode Lamb waves dominate the responses.

Performance evaluation

The MSE and the mean absolute error (MAE) between the predicted ToF and the actual ToF associated with the scattered waves from damage are used to evaluate the performance of the designed network. The two indices are mathematically expressed as:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|\mathbf{y}^{(i)} - \hat{\mathbf{y}}^{(i)}\right| \tag{14}$$

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{y}^{(i)} - \hat{\mathbf{y}}^{(i)}\right)^2 \tag{15}$$

where N is the total number of samples, $\mathbf{y}^{(i)}$ is the actual arrival time, and $\hat{\mathbf{y}}^{(i)}$ is the predicted arrival time.

Figure 6 illustrates the training and validation loss curves of the proposed model over 300 epochs. The training loss (blue line) decreases steadily, demonstrating that the model is learning effectively throughout the training process. The validation loss (red line) follows a similar trend, indicating that the model generalizes well to unseen data. Following an initial sharp decrease in losses, they start to stabilize around the 50th epoch, with the training loss consistently lower than the validation loss. This consistent behavior further validates the robustness and accuracy of our model.

Table 1 presents a comparative analysis of the performance of our proposed method with traditional machine learning and deep learning approaches. The comparison includes machine learning models such as linear regression,[52] ridge regression,[53] Lasso regression,[54] decision tree,[55] principal component analysis (PCA),[56] support vector regression (SVR),[57], and non-negative matrix factorization (NMF),[58] as well as several deep learning models Long Short-Term Memory (LSTM),[59] Recurrent Neural Network (RNN),[60] Gated Recurrent Unit (GRU),[61] Deep Neural Network

Table 1. Comparison between the performance of the proposed method versus traditional machine learning approaches.

	Model	MAE	MSE
	Our model	2.91981249932E-05	1.28721169000E-09
Machine learning methods	Linear Regression	4.32372880772E-01	5.06988352482E-01
	Ridge	1.53132763600E-01	3.72314677860E-02
	Lasso	1.57350022302E-01	3.73620651276E-02
	Decision Tree	1.63530463634E-01	4.77629257822E-02
	PCA + Linear Regression	1.56491938842E-01	3.72886161208E-02
	PCA + Ridge	1.56491938581E-01	3.72886159445E-02
	PCA + SVR	1.60217224315E-01	3.92246366963E-02
	NMF + Linear Regression	1.60135578253E-01	3.88855283405E-02
	NMF + Ridge	1.58633513046E-01	3.80224650692E-02
	NMF + SVR	1.58911557352E-01	3.86085520174E-02
Deep learning methods	LSTM	3.16880453326E-05	1.57238322832E-09
	RNN	3.14389894018E-05	1.52228323027E-09
	GRU	3.24486383443E-05	1.70872061698E-09
	DNN	3.13540027655E-05	1.45930177241E-09
	Transformer	3.27024178897E-05	1.67044803484E-09
	Densenet	3.18908274611E-05	1.54580940390E-09
	Resnet-18	3.18590071187E-05	1.58138232759E-09
	Alexnet	3.08384310074E-05	1.52713839646E-09

MSE: mean squared error; MAE: mean absolute error; PCA: principal component analysis; SVR: support vector regression; NMF: non-negative matrix factorization.

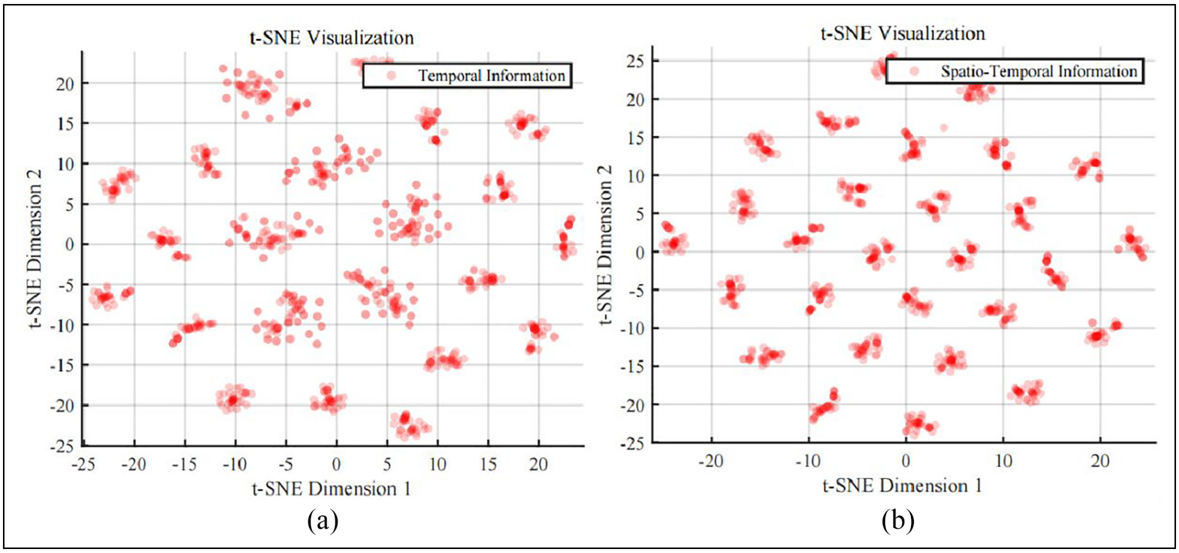

Figure 7. Illustration of the results obtained from the dimensionality reduction algorithm with (a) wavelet coefficients only and (b) both wavelet coefficients and sensor coordinates.

(DNN),[62] Transformer,[63] DenseNet,[64] ResNet-18,[65] and AlexNe.[66]

Our dual-input network processes both temporal and spatial information separately, while in the traditional machine learning models, temporal and spatial data are concatenated into a single input. In the deep learning comparisons, only the temporal part of our network was replaced, keeping the spatial component unchanged to better validate our network's performance.

The experimental results demonstrate that our proposed method achieves superior performance compared to both existing approaches. Specifically, our method yields an MAE that is four orders of magnitude lower and an MSE that is seven orders of magnitude lower than those of traditional machine learning models, while also outperforming the deep learning models included in the comparison. This exceptional performance in both MAE and MSE metrics highlights the effectiveness of our proposed network in integrating temporal and spatial information, underscoring its significant accuracy advantage over existing methods.

Evaluation of network input information

To validate the relevance of integrating STI and SSI into the network, we utilized t-distributed stochastic neighbor embedding (t-SNE)[67] to visualize high-dimensional data. t-SNE transforms data point similarities into joint probabilities, minimizing the Kullback–Leibler (KL) divergence between the joint probabilities of low-dimensional embedding and high-dimensional data. The definition of t-SNE is expressed as follows:

$$\sum_i \mathrm{KL}(P_i||Q_i) = \sum_i \sum_j p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right) \tag{16}$$

where P_i represents pairwise similarities in the original high-dimensional space, and Q_i denotes similarities in the lower-dimensional space. p_{ij} is the conditional probability that point i would pick point j as its neighbor in the original space, and q_{ij} is the conditional probability in the low-dimensional space. The KL divergence in t-SNE measures the variation in the similarity of data distributions in the low-dimensional space after dimensionality reduction with different input signal encoding forms. A more minor KL divergence indicates that data points are closer in the low-dimensional space, suggesting a more apparent similarity in the high-dimensional space.

As shown in Figure 7, after reducing dimensionality, the outcomes derived solely from the STI display a distributed pattern. By contrast, when SSI is integrated, the dimensionality-reduced results show a more concise envelope where temporal information surrounds spatial details. This indicates that this encoding approach effectively captures the reflection path of guided waves.

Interpretability analysis

To explain the reason for the high accuracy achieved by the proposed model, we select a typical input (in Figure 8) and track the signal flow through the model. The actual ToF of the scattered wave from the damage corresponds to the third peak, which would be very difficult to identify by any conventional means. The

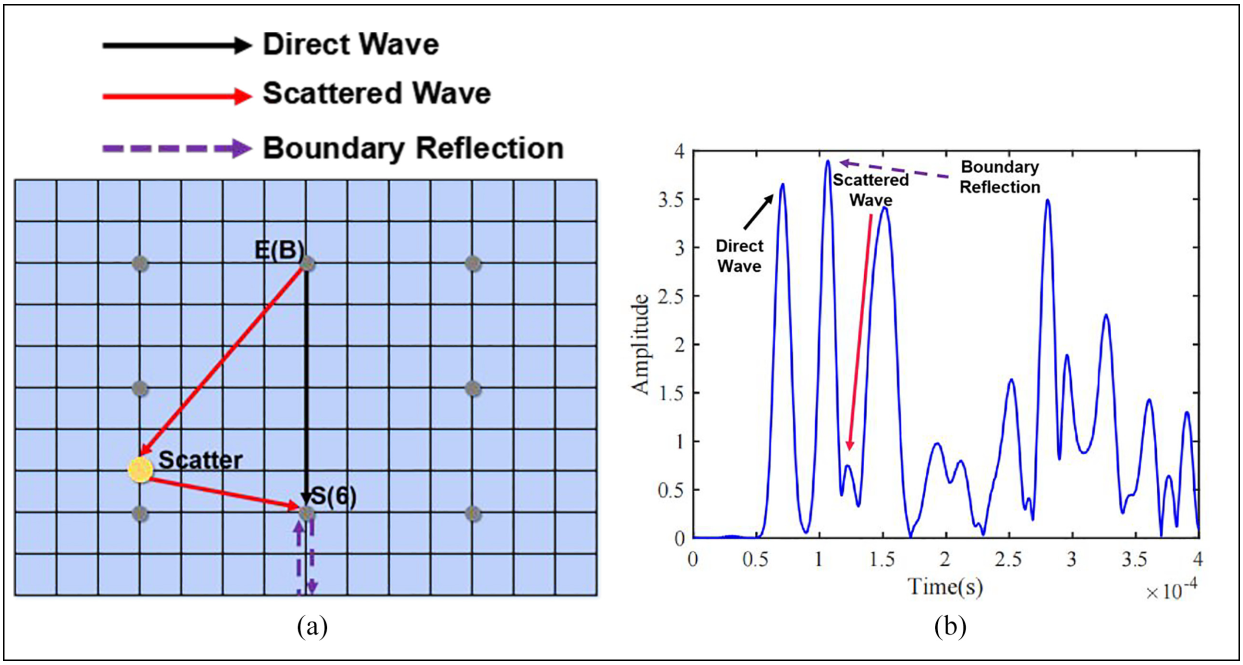

Figure 8. A representative case to showcase the effect of the proposed method: (a) the system configuration and (b) the identified peaks in the wavelet coefficients.

first peak attributes to the direct wave, and remaining peaks associate with boundary reflections. Through deconvolution[68] calculations on the input, we manage to illustrate the knowledge acquired by each network layer.

Five cases, labeled as (a), (b), (c), (d), and (e) in Figure 9, correspond to the output of the convolution operations at different locations depicted in Figure 3. Here, we investigate four distinct patterns observed while applying a neural network model to eliminate boundary reflections in guided wave signals. These patterns signify varied outcomes: Effective Elimination: demonstrating successful removal of boundary reflections while retaining desired signal components; Unprocessed Signal: indicating no suppression of boundary reflections and an unaltered signal; Partial Suppression: illustrating partial mitigation of boundary reflections while preserving desired signal integrity; Misdirected Suppression: highlighting accidental suppression of desired signal components rather than boundary reflections. Understanding these modes is imperative for optimizing the model's performance in guided wave signal processing applications.

The visualization results in Figure 9(a) reveal three notable observations: successful removal of boundary reflections, elimination of unsuccessful instances, and occurrences of ineffective learning. These visualizations underscore the importance of maintaining diversity in the network's learning content during the initial phase. Specifically, to validate the network's ability to eliminate boundary reflections, we performed a deconvolution on the results shown in Figure 8(b). In this figure, the third wave peak represents the desired time of arrival, while the subsequent peaks are attributed to boundary reflections. Our primary focus is to determine whether the network effectively removes these unwanted peaks.

In Figure 9, the x-limits represent time, since the input signal comes from the wavelet transform processing in the time domain. However, the length of the time axis is shortened due to the compression effect of the convolution layers in the multiscale module, the limits of which are determined by the downsampling effect of these convolution operations. After deconvolution is performed to reconstruct the signal and reduce boundary reflections, the x-limits still represents time, although it is compressed compared to the original input.

In the multiscale convolution module, the outcomes with a kernel size of 2 (illustrated in Figure 9(b)) closely resemble those in Figure 9(a). This similarity primarily arises from the small receptive field of the network when using a kernel size of 2, making it susceptible to local minima and excessive focus on detailed information. Upon increasing the kernel size to 3, as depicted in Figure 9(c), the network notably improves its capability to suppress boundary reflections. Compared to the first two scenarios, there is a remarkable enhancement in accurately learning the ToF. This improvement primarily stems from the network's increasing focus on macroscopic signal information with the larger kernel size. Furthermore, when the kernel size is set to 4, the output of network closely resembles that with a size of 3, albeit with a slight increase in failure to eliminate reflections as shown in Figure 9(d). This slight increase is mainly due to both scenarios maintaining a similar-sized receptive field. Subsequently, with a kernel size of 10, a significant improvement is observed in successfully eliminating boundary reflections, as shown in Figure 9(e). This suggests the network prioritizes macroscopic waveform information with a larger kernel size, closely tied to the provided plate size and PZTs spatial coordinates.

In summary, multiscale convolution substantially enhances the learning of signal features, thereby improving the method's accuracy in eliminating boundary reflections. More essentially, the multiscale module considers the mathematical model of neurons and the frequency-domain decay characteristics of activation functions, as shown in Figure 10. For a one-hidden layer ReLU-DNN fitting a 1D function f, the mathematical representation is:

$$h(\mathbf{x}) = \mathbf{a}_j\sigma\left(\mathbf{w}_j\mathbf{x} + \mathbf{b}_j\right) \tag{17}$$

where $\mathbf{w}_j$ and $\mathbf{a}_j$ denote the weights, $\mathbf{x}$ is the input, $\mathbf{b}_j$ represents the bias, and σ is the ReLU activation. The Fourier transform of the output $h(x)$ can be expressed as:

$$F(k) = \int_{-\infty}^{\infty} h(\mathbf{x})e^{-ikx}\mathrm{d}x. \tag{18}$$

The frequency-domain representation of a neural network mathematical model is mainly determined by the activation function. Specifically, a Fourier transform is applied to the $h(x)$ function:

$$F(k) = -\frac{\mathbf{a}_j\mathbf{w}_j}{k^2} - i\frac{\mathbf{b}_j}{k}. \tag{19}$$

From the above equation, it is evident that as the frequency approaches infinity, the amplitude converges toward 0. Consequently, when influenced by the activation function, neurons exhibit a bias toward low frequencies, suggesting that the network prioritizes the acquisition of low-frequency information, which is referred to as the frequency principle.[69]

As shown in Figure 11, the commonly used ReLU activation function becomes less effective in capturing high-frequency information due to exponential decay in the frequency domain during the learning process. This limitation restricts the network's ability to represent detailed features. In the frequency domain, ReLU attenuates high-frequency components through nonlinear clipping, which complicates the network's capacity to learn higher-frequency details of the signal.

To address this limitation, multiscale architectures have been introduced, incorporating layers with different receptive field sizes. This design enables the network to capture information across multiple scales, preserving high-frequency components while integrating them with low-frequency information. Our multiscale convolutional network specifically employs convolutional kernels of sizes 2, 3, 4, and 10, each tuned to capture distinct frequency bands within the input signal. Kernels of sizes 2 and 3 are optimized for capturing high-frequency elements, such as rapid transitions and sharp variations. A kernel size of 4 achieves a balance between high- and mid-frequency features, capturing both fine details and moderate signal changes, while a kernel size of 10 targets low-frequency components, capturing broader trends and smoother variations in the signal.

The architecture achieves comprehensive learning by processing these features in parallel across various kernel sizes, followed by a gradual fusion across layers. This design maintains the integrity of low-frequency information while preventing the total suppression of high-frequency signals due to ReLU's attenuation. Moreover, by dynamically adjusting the significance of different frequency components during training, the multiscale architecture further enhances the network's capacity to accurately represent complex signals.

Ablation experiment

To evaluate the effectiveness of our proposed multiscale network, we conducted a series of ablation experiments summarized in Table 2. These experiments systematically compare different network variants at three scales (small, medium, and large), with and without spatiotemporal fusion. In the "Ablation 2× in Parallel Multiscale" experiment, we used convolutions with a kernel size of 2 across all scales to focus on small-scale features. Similarly, "Ablation 3× in Parallel Multiscale" and "Ablation 10× in Parallel Multiscale" employed kernel sizes of 3 and 10 for medium and large scales, respectively, allowing us to assess the impact of scale selection on model performance. We also conducted "Ablation w/o Multiscale" experiments to explore the effects of removing the multiscale structure. For instance, "Ablation w/o Multiscale (Single 2× Conv)" utilized a single convolution layer with a kernel size of 2, while "Ablation w/o Spatial Info" assessed performance without integrating spatial information. By comparing these results to our multiscale architecture, we can evaluate the contribution of multiscale

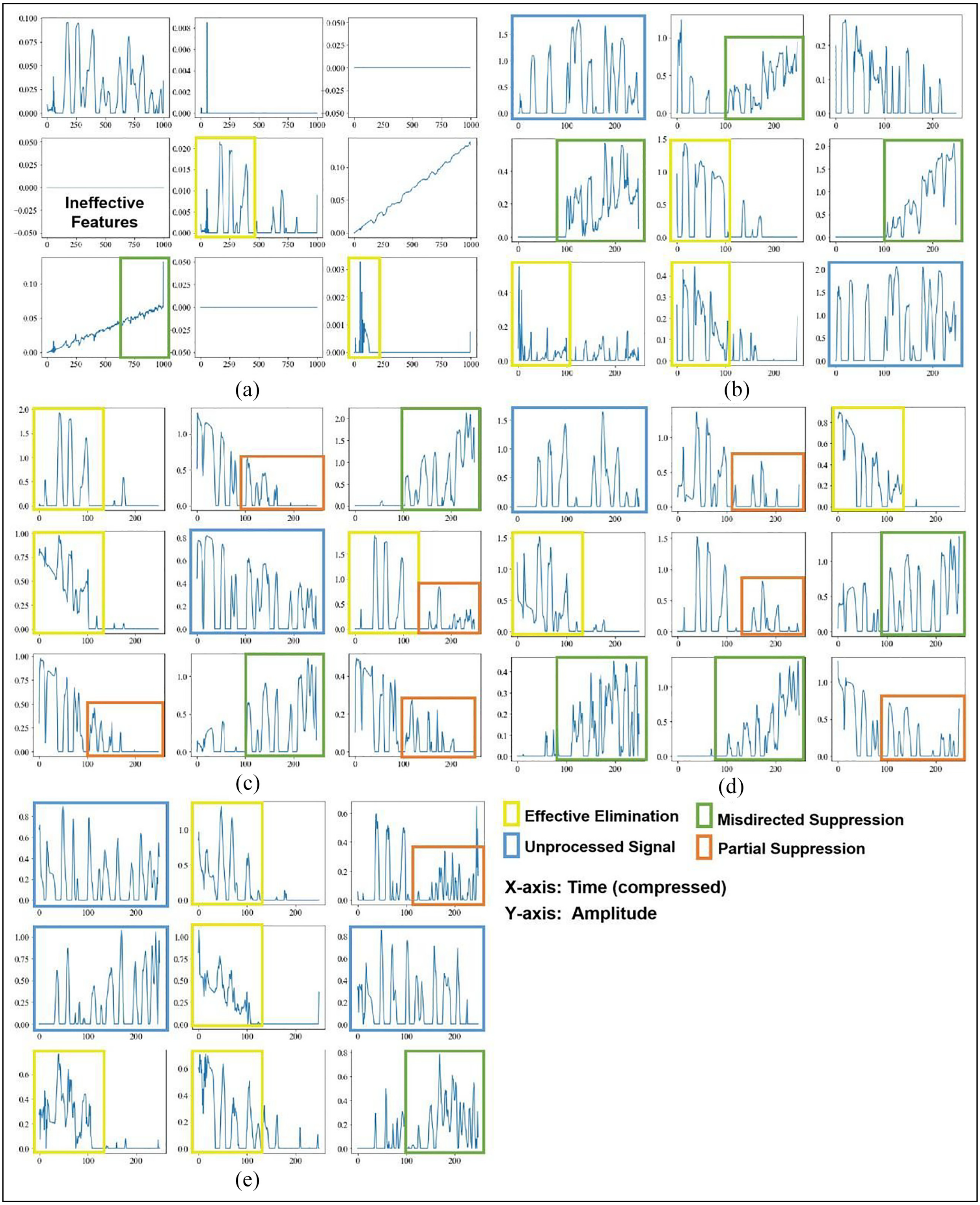

Figure 9. Results of the deconvolution operations at different locations in the proposed network to illustrate the effect of the multiscale scheme for boundary reflection elimination: (a) after deconvolution following global convolution of the input signal, (b) after deconvolution with a 2-point kernel, (c) after deconvolution with a 3-point kernel, (d) after deconvolution with a 4-point kernel, and (e) after deconvolution with a 10-point kernel.

Table 2. Ablation study results.

Method	MAE	MSE
Ablation 2× in Parallel Multiscale	3.03963360E-05	1.35731420E-09
Ablation 3× in Parallel Multiscale	3.01363561E-05	1.36770098E-09
Ablation 4× in Parallel Multiscale	3.04813807E-05	1.46242259E-09
Ablation 10× in Parallel Multiscale	2.92590390E-05	1.30759970E-09
Ablation w/o Spatial Info	2.95890107E-05	1.32726064E-09
Ablation w/o Multiscale (Single 2× Conv)	3.08380499E-05	1.40681359E-09
Ablation w/o Multiscale (Single 3× Conv)	3.09936803E-05	1.45320109E-09
Ablation w/o Multiscale (Single 4× Conv)	3.07001596E-05	1.40353542E-09
Ablation w/o Multiscale (Single 10× Conv)	3.06420031E-05	1.39146826E-09
Our model	2.91981250E-05	1.28721169E-09

MSE: mean squared error; MAE: mean absolute error.

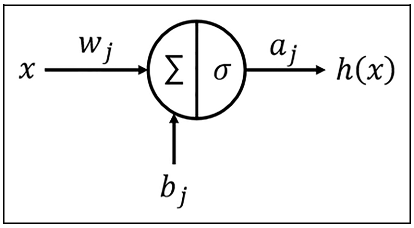

Figure 10. Illustration of the mathematical model of neurons.

fusion for feature extraction and its effectiveness in processing complex data.

This detailed experimental approach allows for a thorough analysis of the contributions of each configuration, as well as a quantified assessment of the impact of spatiotemporal fusion on overall performance. The results of these controlled experiments unequivocally demonstrate the superior effectiveness of our multiscale network. These findings highlight the crucial role of multiscale fusion, which improves the network's ability to process information across various dimensions and underscores the importance of multiscale integration in addressing complex tasks.

Applications

Signal analysis and processing

To confirm the efficiency of our proposed network in eliminating boundary reflections and extracting the ToF of the scattered waves from damage, we place a scatterer near the boundary as a representative case, as shown in Figure 12. It is worth noting that the position of the scatterer has not been used in the previous training process. We collect and test the data from excitation E to receiver S. The black line indicates the direct wave, the red line associates with the path of the scattered wave, and the dashed lines illustrate the boundary reflections.

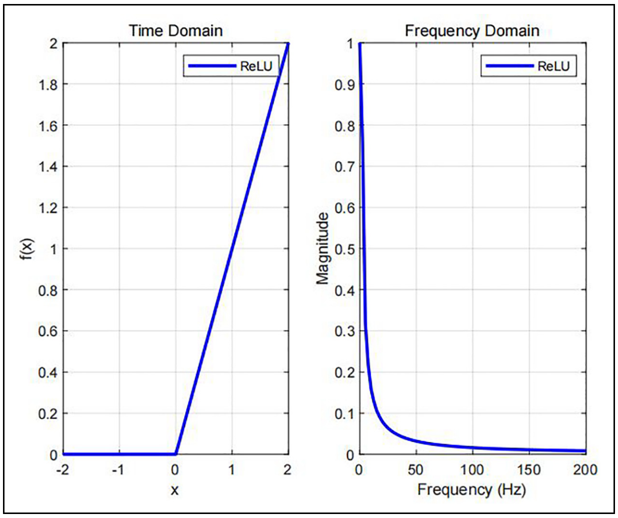

Figure 11. Form of the ReLU activation function in time and frequency domains.

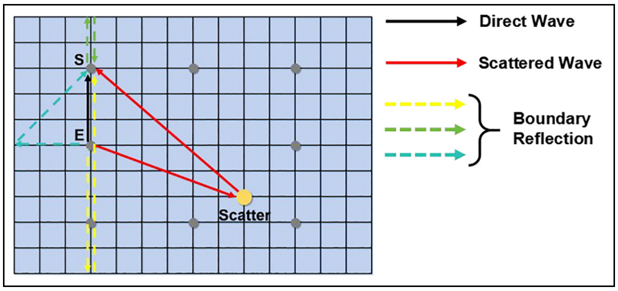

Figure 12. Wave path diagram.

The time-domain signal from the specific path is shown in Figure 13(a), followed by the corresponding

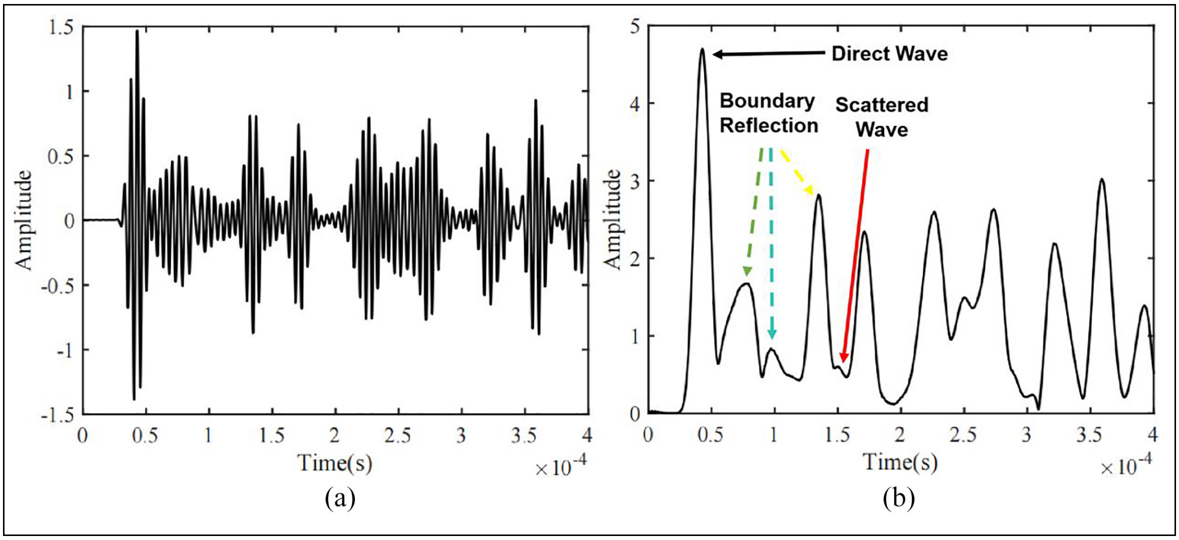

Figure 13. (a) The raw signal and (b) the identified peaks in the wavelet coefficients.

wavelet coefficients in Figure 13(b). The first peak in wavelet coefficients is presumed to be the direct wave. In this specific case, the second peak corresponds to the boundary reflection rather than the scattered wave. Instead, the fifth peak corresponds to the scattered wave from damage. As to the output of the proposed method, the predicted ToF is 1.5e-04, which agrees well with the ground truth. This confirms that the designed network adeptly recognizes and accurately identifies the ToF of scatter waves despite their relatively low amplitude.

Localization results

With the extracted ToFs, damage localization can be achieved by the ellipse positioning method,[70] resulting in the corresponding damage image. Two representative cases are selected to illustrate the efficacy of the method, as shown in Figure 14. In the first case, when damage is located near the center of the plate, the damage image is reconstructed with the ToFs extracted by the proposed method, as depicted in Figure 14(a). It can be seen the accuracy of damage localization is very high with a localization error of merely 1.31 cm. By comparison, a traditional strategy is adopted by using the ToF of the second peaks in the wavelet coefficients. This should be fine when the structure is sufficiently large. However, in the present case, influenced by the boundary reflections, the accuracy suffers, as shown in Figure 14(b), with a localization error of 34.29 cm. In the second case, the scatter is placed very close to the structural boundary. The damage images reconstructed with the proposed and traditional methods are obtained in Figure 14(c) and (d), respectively. It can be seen that the proposed method induces a localization error of 3.14 cm, whereas the traditional method incurs an error of 9.29 cm. These outcomes again underscore the efficacy of the proposed method in advancing damage localization.

Conclusions

In this study, we propose a multiscale CNN with spatiotemporal fusion to accurately extract the ToF of scattered Lamb waves from damage while effectively mitigating the influence of boundary reflections. The effectiveness of our proposed network architecture is validated through experiments, in which we compare its performance with traditional machine learning algorithms. Additionally, we analyze the encoded representations of different network inputs using the t-SNE algorithm. Interpretability analyses are then conducted by tracing data flows through deconvolution operations and analyzing them from a frequency-domain perspective. Moreover, ablation experiments are executed to assess the impact of different scales on the results. Finally, testing is performed to evaluate the performance of our network in eliminating boundary reflections compared with manually identified arrival time localization results.

The proposed network achieves significantly lower MAE and MSE than traditional methods. Specifically, the MAE of our proposed method is four orders of magnitude lower than that of other existing methods, and the MSE is seven orders of magnitude lower, unequivocally demonstrating the superior accuracy of

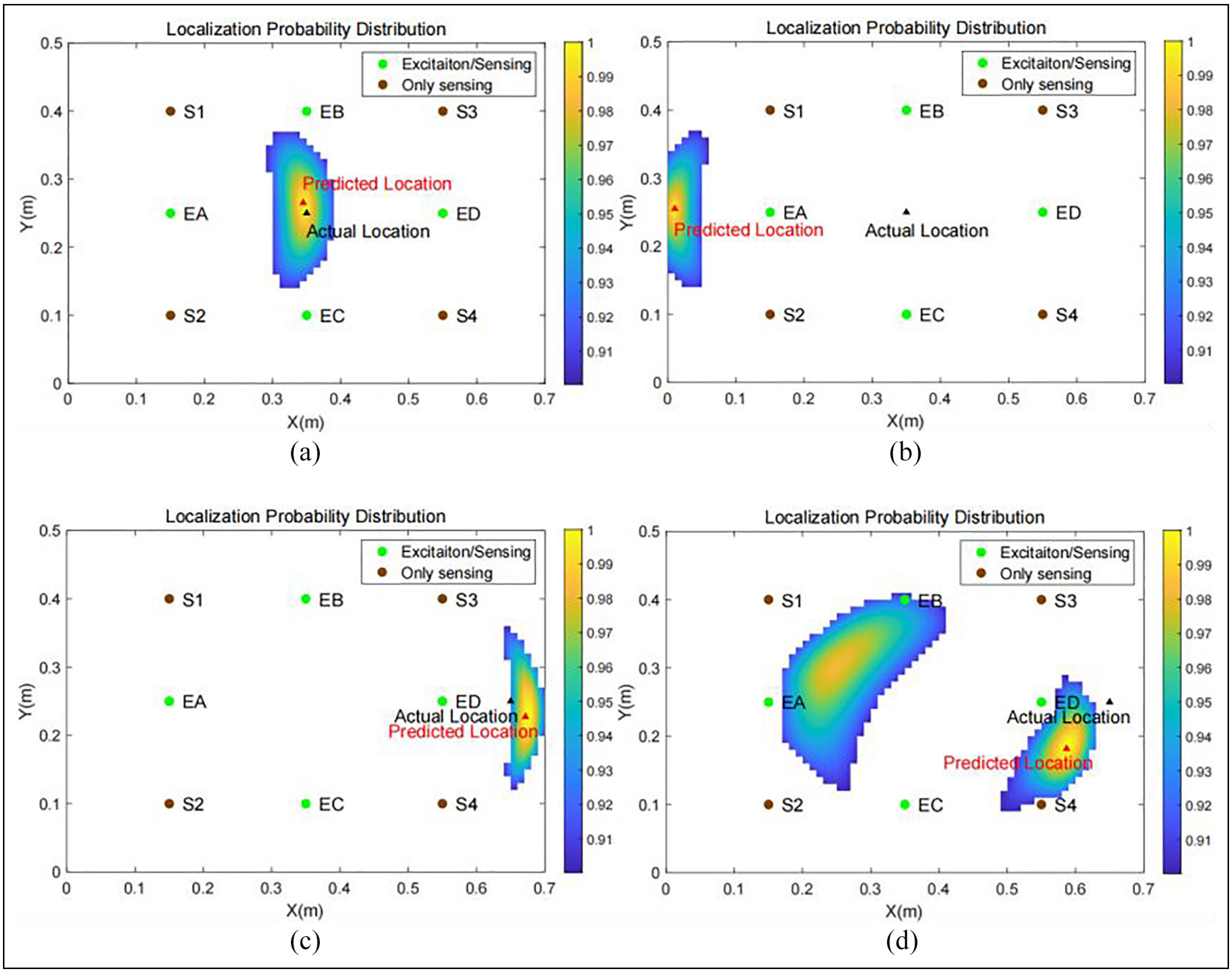

Figure 14. Comparison of location results: (a) and (c) eliminating boundary reflection through the network; (b) and (d) second peak as the default arrival time.

our proposed approach. Furthermore, the t-SNE results indicate that coupling temporal and spatial information as inputs to the network better assists in capturing the reflection paths of guided waves. The effectiveness of our network in eliminating unnecessary boundary reflections during intermediate processing stages is confirmed by deconvolution operations. Additionally, the designed multiscale network avoids the neural network's bias toward low-frequency features, allowing for comprehensive learning of features at different scales. Ablation experiments demonstrate that combining four scales (2, 3, 4, and 10) and the spatiotemporal fusion form currently yields the most effective results. Finally, by comparing the network-identified arrival times with manually identified ToF for damage localization, we find that the localization errors after boundary reflection elimination by the network are only 1.31 and 3.14 cm for two typical scenarios, as opposed to the respective errors of 34.29 and 9.29 cm after manual identification.

The developed CNN with spatiotemporal fusion and multiscale properties leads to significant improvements in the accuracy of the ToF estimation of the scattered Lamb wave for damage localization. It can be used as an enabler to enhance the detection and localization capabilities of existing SHM methods based on ToF. Meanwhile, the study paves the way for understanding and explaining the internal working mechanism of deep learning models in such tasks.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The work was supported by grants from the RIAIoT (Research Institute for Artificial Intelligence of Things) through project P0049625, National Natural Science Foundation of China (12302114), the Research Grants Council of Hong Kong Special Administrative Region (PolyU 152013/21E), the Natural Science Foundation of Shanghai (22ZR1462700), the Fundamental Research Funds for the Central Universities, and the Innovation and Technology Commission of the HKSAR Government to the Hong Kong Branch of National Rail Transit Electrification and Automation Engineering Technology Research Center (K-BBY1).

ORCID iDs

Yang Song https://orcid.org/0009-0001-9606-8055
Yuanman Zhang https://orcid.org/0000-0002-3642-402X

References

1. Chang F-K, Markmiller JF, Yang J, et al. *Structural health monitoring, system health management*. New York, NY: John Wiley and Sons, Inc., 2001.
2. Giurgiutiu V. *Structural health monitoring: with piezoelectric wafer active sensors*. Amsterdam, Netherlands: Elsevier, 2007.
3. Farrar CR and Worden K. An introduction to structural health monitoring. *Philos Trans A Math Phys Eng Sci* 2007; 365(1851): 303–315.
4. Adeagbo MO, Wang S-M and Ni Y-Q. Revamping structural health monitoring of advanced rail transit systems: a paradigmatic shift from digital shadows to digital twins. *Adv Eng Inform* 2024; 61: 102450.
5. Malekloo A, Ozer E, AlHamaydeh M, et al. Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights. *Struct Health Monit* 2022; 21(4): 1906–1955.
6. Su Z and Ye L. *Identification of damage using Lamb waves: from fundamentals to applications*. Heidelberg, Germany: Springer Science & Business Media, 2009.
7. Yu L and Tian Z. Lamb wave structural health monitoring using a hybrid PZT-laser vibrometer approach. *Struct Health Monit* 2013; 12(5–6): 469–483.
8. Kaewniam P, Cao M, Alkayem NF, et al. Recent advances in damage detection of wind turbine blades: a state-of-the-art review. *Renew Sustain Energy Rev* 2022; 167: 112723.
9. Annamdas VGM, Bhalla S and Soh CK. Applications of structural health monitoring technology in Asia. *Struct Health Monit* 2017; 16(3): 324–346.
10. Cawley P. Structural health monitoring: closing the gap between research and industrial deployment. *Struct Health Monit* 2018; 17(5): 1225–1244.
11. Michaels JE. Detection, localization and characterization of damage in plates with an in situ array of spatially distributed ultrasonic sensors. *Smart Mater Struct* 2008; 17(3): 035035.
12. Dawson AJ, Michaels JE and Michaels TE. Isolation of ultrasonic scattering by wavefield baseline subtraction. *Mech Syst Signal Process* 2016; 70: 891–903.
13. Hameed MS, Li Z, Qi J, et al. Lamb wave based multistage damage detection method. *AIP Conf Proc* 2019; 2102(1): 050009.
14. Cantero-Chinchilla S, Chiachío J, Chiachío M, et al. A robust Bayesian methodology for damage localization in plate-like structures using ultrasonic guided-waves. *Mech Syst Signal Process* 2019; 122: 192–205.
15. Hall JS, McKeon P, Satyanarayan L, et al. Minimum variance guided wave imaging in a quasi-isotropic composite plate. *Smart Mater Struct* 2011; 20(2): 025013.
16. Ng C-T and Veidt M. A Lamb-wave-based technique for damage detection in composite laminates. *Smart Mater Struct* 2009; 18(7): 074006.
17. Mori N, Biwa S and Kusaka T. Damage localization method for plates based on the time reversal of the mode-converted Lamb waves. *Ultrasonics* 2019; 91: 19–29.
18. Qiu L and Yuan S. A phase synthesis time reversal impact imaging method for on-line composite structure monitoring. *Smart Struct Syst* 2011; 8(3): 303–320.
19. Purekar A and Pines D. Damage detection in thin composite laminates using piezoelectric phased sensor arrays and guided Lamb wave interrogation. *J Intell Mater Syst Struct* 2010; 21(10): 995–1010.
20. Yu W, Shenfang Y and Lei Q. Improved wavelet-based spatial filter of damage imaging method on composite structures. *Chinese J Aeronaut* 2011; 24(5): 665–672.
21. Bin L, Lei Q and Shenfang Y. Damage imaging and localization method based on multi-dimension arrays and spatial filter without wave velocity. *Acta Materize Compositae Sinica* 2014; 31(3): 835–844.
22. Qiu L, Liu B, Yuan S, et al. Impact imaging of aircraft composite structure based on a model-independent spatial-wavenumber filter. *Ultrasonics* 2016; 64: 10–24.
23. Ren Y, Qiu L, Yuan S, et al. A diagnostic imaging approach for online characterization of multi-impact in aircraft composite structures based on a scanning spatial-wavenumber filter of guided wave. *Mech Syst Signal Process* 2017; 90: 44–63.
24. Engholm M and Stepinski T. Direction of arrival estimation of Lamb waves using circular arrays. *Struct Health Monit* 2011; 10(5): 467–480.
25. Zuo H, Yang Z, Xu C, et al. Damage identification for plate-like structures using ultrasonic guided wave based on improved MUSIC method. *Compos Struct* 2018; 203: 164–171.
26. Bao Q, Yuan S, Wang Y, et al. Anisotropy compensated MUSIC algorithm based composite structure damage imaging method. *Compos Struct* 2019; 214: 293–303.
27. Salmanpour M, Sharif Khodaei Z and Aliabadi M. Transducer placement optimisation scheme for a delay and sum damage detection algorithm. *Struct Control Health Monit* 2017; 24(4): e1898.

28. Karniadakis GE, Kevrekidis IG, Lu L, et al. Physics-informed machine learning. *Nat Rev Phys* 2021; 3(6): 422–440.
29. Tetali HV, Harley JB and Haeffele BD. *Wave physics-informed matrix factorizations*. Piscataway, NJ: IEEE, 2024.
30. Baddoo PJ, Herrmann B, McKeon BJ, et al. Physics-informed dynamic mode decomposition. *Proc R Soc A* 2023; 479(2271): 20220576.
31. Wu Y, Sicard B and Gadsden SAJESwA. Physics-informed machine learning: a comprehensive review on applications in anomaly detection and condition monitoring. *Expert Syst Appl* 2024; 255: 124678.
32. Shukla K, Di Leoni PC, Blackshire J, et al. Physics-informed neural network for ultrasound nondestructive quantification of surface breaking cracks. *J Nondestruct Eval* 2020; 39: 1–20.
33. Farrar CR and Worden K. *Structural health monitoring: a machine learning perspective*. Hoboken, NJ: John Wiley & Sons, 2012.
34. Bao Y and Li H. Machine learning paradigm for structural health monitoring. *Struct Health Monitor* 2021; 20(4): 1353–1372.
35. Zhang S, Li CM and Ye W. Damage localization in plate-like structures using time-varying feature and one-dimensional convolutional neural network. *Mech Syst Signal Process* 2021; 147: 107107.
36. Zhang S, Li CM, Yang J, et al. Effective combination of modeling and experimental data with deep metric learning for guided wave-based damage localization in plates. *Mech Syst Signal Process* 2022; 172: 108979.
37. Wang X, Lin M, Li J, et al. Ultrasonic guided wave imaging with deep learning: applications in corrosion mapping. *Mech Syst Signal Process* 2022; 169: 108761.
38. Rautela M and Gopalakrishnan A. Deep learning frameworks for wave propagation-based damage detection in 1d-waveguides. In: *Proceedings of 11th international symposium on NDT in aerospace*, Toulouse, France, 23–26 September 2019, vol. 2, pp. 1–11.
39. Sbarufatti C, Manson G and Worden K. A numerically-enhanced machine learning approach to damage diagnosis using a Lamb wave sensing network. *J Sound Vib* 2014; 333(19): 4499–4525.
40. Lin M, Guo S, He S, et al. Structure health monitoring of a composite wing based on flight load and strain data using deep learning method. *Compos Struct* 2022; 286: 115305.
41. Heesch M, Dziendzikowski M, Mendrok K, et al. Diagnostic-quality guided wave signals synthesized using generative adversarial neural networks. *Sensors* 2022; 22(10): 3848.
42. Liu C, Chen Y and Xu XJ. Structural digital twin for damage detection of CFRP composites using meta transfer learning-based approach. *Expert Syst Appl* 2024; 261: 125458.
43. Xu X and Liu CJ. Physics-guided deep learning for damage detection in CFRP composite structures. *Compos Struct* 2024; 331: 117889.
44. Liu X, Jiang Z and Yan Z. Improvement of accuracy in damage localization using frequency slice wavelet transform. *Shock Vib* 2012; 19(4): 585–596.
45. Amari S-i. Backpropagation and stochastic gradient descent method. *Neurocomputing* 1993; 5(4–5): 185–196.
46. Zhang Z. Improved adam optimizer for deep neural networks. In: *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*, Banff, AB, Canada, 04–06 June 2018. Piscataway, NJ: IEEE, 2018, pp. 1–2.
47. Ma Q, Chen E, Lin Z, et al. Convolutional multitimescale echo state network. *IEEE Trans Cybern* 2019; 51(3): 1613–1625.
48. Nair V and Hinton GE. Rectified linear units improve restricted boltzmann machines. In: *Proceedings of the 27th international conference on machine learning (ICML-10)*, Haifa, Israel, 21–24 June2010, pp. 807–814. Madison, WI: Omnipress.
49. Ioffe S and Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *International conference on machine learning*. Lille, France, 6–11 July 2015, Cambridge, MA: PMLR, 2015, pp. 448–456.
50. Gholamalinezhad H and Khosravi H. Pooling methods in deep neural networks, a review. arXiv preprint arXiv:2009.07485, 2020.
51. Kruse R, Mostaghim S, Borgelt C, et al. Multi-layer perceptrons. In: *Computational intelligence: a methodological introduction*. Cham, Switzerland: Springer, 2022, pp. 53–124.
52. Montgomery DC, Peck EA and Vining GG. *Introduction to linear regression analysis*. Hoboken, NJ: John Wiley & Sons, 2021.
53. McDonald GC. Ridge regression. *Wiley Interdiscip Rev Comput Stat* 2009; 1(1): 93–100.
54. Ranstam J and Cook JA. LASSO regression. *Br J Surg* 2018; 105(10): 1348–1348.
55. Song Y-Y and Ying LJ. Decision tree methods: applications for classification and prediction. *Shanghai Arch Psychiatry* 2015; 27(2): 130.
56. Maćkiewicz A and Ratajczak WJC. Principal components analysis (PCA). *Comput Geosci* 1993; 19(3): 303–342.
57. Awad M, Khanna R, Awad M, et al. *Efficient learning machines: theories, concepts, and applications for engineers and system designers*. New York, NY: Apress, 2015, pp. 67–80.
58. Lee D and Seung HS. Algorithms for non-negative matrix factorization. *Adv Neural Inf Process Syst* 2000; 13: 556–562.
59. Sherstinsky AJ. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Phys D Nonlinear Phenom* 2020; 404: 132306.
60. Medsker LR and Jain LC. Recurrent neural networks. *J Chem Inf Model* 2001; 5(64–67): 2.
61. Chung J, Gulcehre C, Cho K, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling. In: *Proceedings of the NIPS workshop on deep learning*, Montreal, Canada, 8–13 December 2014. Cambridge, MA: NeurIPS Foundation.

62. LeCun Y, Bengio Y and Hinton GJ. Deep learning. *Nature* 2015; 521(7553): 436–444.
63. Vaswani A. Attention is all you need. *Adv Neural Inform Process Syst* 2017.
64. Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, Hawaii, 21–26 July, 2017, pp. 4700–4708. Piscataway, NJ: IEEE publisher.
65. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, Las Vegas, Nevada, 27–30 June 2016, pp. 770–778. Piscataway, NJ: IEEE publisher.
66. Krizhevsky A, Sutskever I and Hinton GE. ImageNet classification with deep convolutional neural networks. *Adv Neural Inform Process Syst* 2012; 25: 1097–1105.
67. Belkina AC, Ciccolella CO, Anno R, et al. Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. *Nat Commun* 2019; 10(1): 5415.
68. Chen Y, Wang Y, Chen Y, et al. Deep autoencoder for interpretable tissue-adaptive deconvolution and cell-type-specific gene analysis. *Nat Commun* 2022; 13(1): 6735.
69. Ma Y, Xu Z-QJ and Zhang J. Frequency principle in deep learning beyond gradient-descent-based training. arXiv preprint arXiv:2101.00747, 2021.
70. Zheng R, Wang G and Ho K. Accurate semidefinite relaxation method for elliptic localization with unknown transmitter position. *IEEE Trans Wirel Commun* 2020; 20(4): 2746–2760.