



Research Paper

An Effective Damage Identification Method Combining Double-Window Principal Component Analysis with AutoGluon

Ge Zhang¹, Neng Wei², Ying Zhou², Licheng Zhou², Gongfa Chen¹, Zejia Liu², Bao Yang²,
Zhenyu Jiang², Yiping Liu², Liqun Tang²

¹ Guangdong University of Technology, No. 161 Yinglong Road, Tianhe District, Guangzhou 510006, P.R. China
² School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510640, China

Received December 14 2023; Revised February 24 2024; Accepted for publication February 29 2024.
Corresponding authors: Licheng Zhou (tlczhou@scut.edu.cn); Gongfa Chen (gongfa.chen@gdut.edu.cn)
© 2024 Published by Shahid Chamran University of Ahvaz

Abstract. In recent years, Double Window Principal Component Analysis (DWPCA) has been proposed. The spatial windows exclude damage-insensitive data from the analysis, while the temporal window improves the discrimination between healthy and damaged states. As a result, the DWPCA method exhibits higher sensitivity and resolution in damage identification compared to traditional PCA methods, as well as other traditional signal processing methods such as wavelet analysis. However, existing research on DWPCA has mainly focused on using the first-order eigenvector for damage identification, while the potential of higher order DWPCA eigenvectors remains unexplored. Therefore, the objective of this paper is to investigate the damage identification capabilities of higher-order DWPCA eigenvectors. Furthermore, we propose three types of damage-sensitive features based on DWPCA eigenvectors and use them as inputs to artificial intelligence (AI) algorithms for damage localization and quantification. The AI algorithms considered include AutoGluon and Transformer, which are powerful machine learning (ML) and deep learning (DL) algorithms proposed in recent years, respectively. In addition, classical ML algorithms such as Decision Tree (DT), Random Forest (RF) and Extreme Gradient Boost (XGBoost) are considered for comparison. Extensive benchmark experiments are performed and the numerical results obtained show that the combination of AutoGluon with DWPCA features achieves remarkable performance in terms of damage localization and quantification. This performance exceeds that of DT, RF, XGBoost and Transformer algorithms. Specifically, the prediction accuracies for damage localization and quantification exceed 90%. These results highlight the great potential of integrating AutoGluon with DWPCA features, particularly by combining AutoGluon with the first and second DWPCA eigenvectors, for real-world applications in structural health monitoring.

Keywords: Structural health monitoring; damage detection; machine learning algorithm; principal component analysis.

1. Introduction

Structural Health Monitoring (SHM) technology aims to collect real-time structural responses by installing sensors on structures and to evaluate the condition of the bridge by analyzing these responses using data-driven methods [1-3]. However, despite the installation of SHM systems on most long-span bridges and the collection of numerous structural responses, practical bridges still face challenges in damage detection. A major factor is that the structural responses may not show significant changes after damage has occurred, or the damage information may be masked by environmental noise. As a result, structural responses become insensitive to damage, making damage detection difficult. Therefore, it is crucial to extract damage-sensitive features from the original responses and apply appropriate artificial intelligence algorithms to establish the relationship between inputs and structural states.

PCA is a versatile and powerful technique that plays a crucial role in exploratory data analysis and feature engineering, enabling efficient data representation and interpretation [4-7]. The primary goal of PCA is to identify a set of orthogonal vectors, known as principal components, that best represent the data by maximizing the variance along each component. However, traditional PCA analysis considers the entire historical data set, including responses from both healthy and damaged states, making it difficult to observe structural changes in the time domain promptly after damage occurrence [4, 5, 8]. To address this issue, Smith et al. introduced the Moving Principal Component Analysis (MPCA) method by incorporating a moving time window into PCA [9-13]. In each step, responses within a fixed-size window are subjected to PCA analysis, yielding corresponding eigenvectors. As the window moves forward, a time series of eigenvectors is generated. Different structural states can be effectively distinguished by following the evolution of the eigenvector series. MPCA eigenvectors have been successfully applied in practical engineering for structural health monitoring and damage detection, as in the case of the Ricciolo Viaduct Bridge in Switzerland [10, 12, 14]. Previous studies



have also shown that MPCA outperforms other methods, including autoregressive integrated moving average, continuous wavelet transform, robust regression analysis and instance-based methods, in terms of damage localization [10, 12, 14]. However, in the case of localized damage, such as cracks, only a few sensors in the vicinity may show changes in their responses. Both PCA and MPCA methods consider the responses of all sensors, including a large number of damage-insensitive responses in the calculation. Therefore, the simultaneous application of spatial windows in MPCA can potentially improve the effectiveness of damage identification.

To address this problem, our previous study proposed the DWPCA method, which includes both spatial and temporal windows [15, 16]. The spatial windows exclude damage-insensitive data from the analysis, while the temporal window improves the discrimination between healthy and damaged states. As a result, the DWPCA method exhibits higher sensitivity and resolution in damage identification compared to previous methods. Numerical results from previous research have shown that DWPCA can detect damage 31 to 59 days earlier than MPCA for a concrete beam [15]. However, the DWPCA method still has some limitations. Previous research on DWPCA has mainly focused on using the first-order eigenvector for damage identification, and the damage identification capability of higher order eigenvectors has yet to be investigated.

The relationship between sensor responses and structural state is complex, especially for practical bridges. In recent decades, ML algorithms have proven to be effective and efficient in dealing with complex nonlinear problems and high-dimensional data [17-19]. In recent years, they have shown potential in structural damage detection [20-23]. Previous studies have shown that ML algorithms perform better when given damage-sensitive features as inputs [11, 24-28]. Consequently, derivative features such as curvature modes, autoregressive (AR) parameters and MPCA eigenvectors have been used as inputs to ML algorithms for damage detection, resulting in improved performance in damage localization and quantification. In addition, it has been shown that using damage-sensitive features as inputs to ML algorithms can reduce the time required for model optimization and computation. In our recent study, DWPCA eigenvectors were found to be more sensitive to damage than MPCA eigenvectors [10, 12, 14, 16]. However, the performance of ML algorithms using DWPCA eigenvectors as inputs remains unknown. In addition, ensemble learning methods using different types of techniques and classifiers have been applied to damage identification [29-31]. By using multiple ML algorithms, ensemble learning methods can achieve better predictions than single learning methods. Four tree-based ensemble classifiers, namely random forest (RF), extremely randomized tree (ERT), gradient boosting decision trees (GBDT) and extreme gradient boosting decision trees (XGBoost), have been used for structural damage detection and other signal anomaly detection [31-33]. Additionally, AutoGluon, an automated machine learning (AutoML) framework, has been utilized in signal anomaly detection scenarios, such as landslide hazard analysis, demonstrating exceptional predictive capability with highly accurate results [34]. AutoGluon offers a Tabular module that excels in handling structured data for predictive modeling tasks. It employs a multi-layered approach to ensemble multiple models through stacking/bagging techniques [35]. However, AutoGluon has not yet been applied to structural damage identification. Currently, there is no relevant research comparing the damage detectability of AutoGluon with other ensemble machine learning algorithms and newly proposed deep learning methods. Considering that DWPCA features exhibit higher damage sensitivity compared to the original signals and other traditional damage-sensitive features, this study aims to combine AutoGluon and DWPCA features for damage localization and quantification. Furthermore, this study will conduct a comparative analysis of the damage identification capabilities among the AutoGluon algorithm, ensemble algorithms, and the Transformer, which serves as a representative deep learning algorithm.

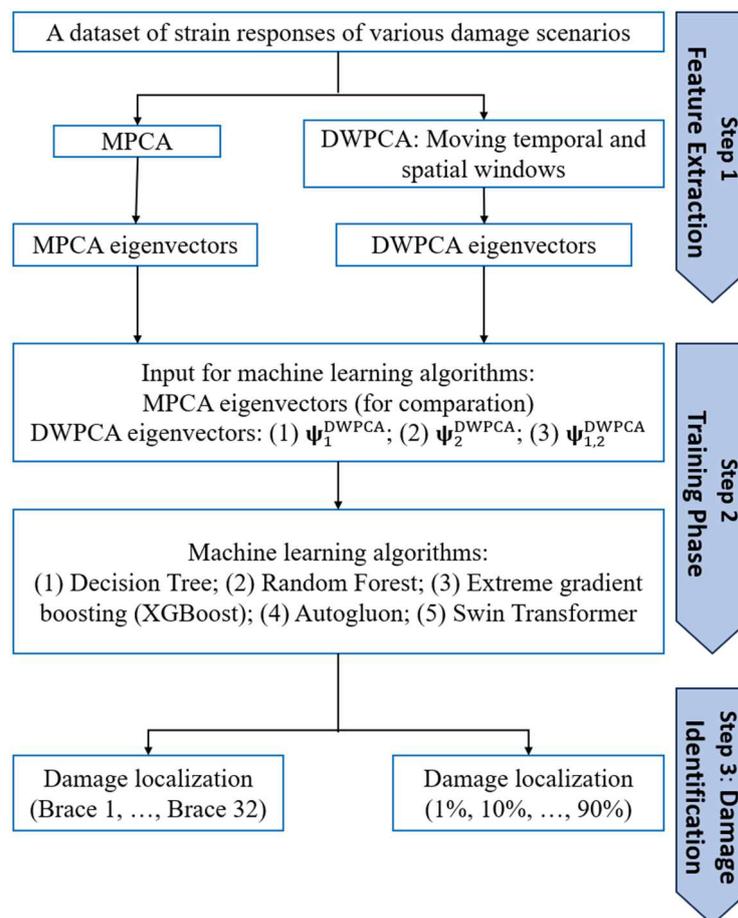


Fig. 1. The flowchart of the proposed method for damage identification.



This paper investigates the damage identification performance of the method that combines DWPCA with ML algorithms, as illustrated in Fig. 1. Firstly, we investigate the damage sensitivity and robustness of the first-order and second-order DWPCA eigenvectors after damage occurs. Then, three types of damage-sensitive features based on DWPCA eigenvectors are proposed and applied as inputs to ML algorithms. The damage performance of a representative ML algorithm with DWPCA features is then evaluated in terms of damage localization and quantification. In addition, AutoGluon is introduced and combined with the three types of DWPCA features for damage identification. Finally, a comparative study is performed between AutoGluon and several representative ML algorithms and novel DL algorithms for damage identification.

2. Methods

2.1. Feature extraction

Both MPCA and DWPCA have been shown to be effective techniques for extracting damage-sensitive features from structural responses. Previous studies [15, 16] have shown that DWPCA outperforms MPCA in terms of sensitivity and resolution. Previous research has suggested that the first-order and second-order eigenvectors capture most of the structural information[5, 8, 28]. However, the exploration of eigenvectors beyond the first-order eigenvector remains incomplete in DWPCA. Therefore, it is crucial to investigate the damage identification capability of higher order eigenvectors in DWPCA. In this section, the principles of MPCA and DWPCA are described sequentially as follows.

2.1.1. MPCA

The MPCA method introduces a moving time window to the PCA method. At each step, the structural responses within a fixed-sized time window are analyzed, and eigenvectors are obtained. As the time window moves forward, a time series of eigenvectors is generated. Instead of directly analyzing the structural responses, damage alarms are triggered based on the evolution of the eigenvector series. The detailed steps of the MPCA method are presented below.

As the time window moves forward, the response matrix \mathbf{U} at the k^{th} time step can be represented as:

$$\mathbf{U}(t_k) = \begin{bmatrix} u_1(t_k) & \cdots & u_m(t_k) \\ \vdots & \ddots & \vdots \\ u_1(t_{k+T_w-1}) & \cdots & u_m(t_{k+T_w-1}) \end{bmatrix} \tag{1}$$

where $k = 1, 2, \dots, N_s - T_w + 1$, m represents the total number of sensors, and t represents time. $u_j(t_k)$ denotes the response of the j^{th} sensor at the k^{th} time step. T_w represents the size of the time window, which is equal to the duration of the observations within the time window. N_s represents the total number of sensor measurements. If the response time series have periodic characteristics, the time window size should be set to the length of the longest period.

Next, the time series responses for each column are normalized by subtracting the mean value calculated using Eq. (2):

$$\bar{\mu}_j(k) = \frac{1}{T_w} \sum_{i=k}^{k+T_w-1} \mu_j(t_i) \tag{2}$$

Then, the normalized matrix can be expressed by Eq. (3):

$$\mathbf{U}'(t_k) = \begin{bmatrix} u_1(t_k) - \bar{u}_1(t_k) & \cdots & u_m(t_k) - \bar{u}_m(t_k) \\ \vdots & \ddots & \vdots \\ u_1(t_{k+T_w-1}) - \bar{u}_1(t_{k+T_w-1}) & \cdots & u_m(t_{k+T_w-1}) - \bar{u}_m(t_{k+T_w-1}) \end{bmatrix} \tag{3}$$

Subsequently, the covariance matrix is obtained as:

$$\mathbf{C}(t_k) = \frac{1}{m} \mathbf{U}'(t_k)^T \mathbf{U}'(t_k) \tag{4}$$

Finally, the eigenvalues and eigenvectors of the covariance matrix at k^{th} time step satisfy:

$$[\mathbf{C}(t_k) - \lambda_i^P(t_k) \mathbf{I}^P] \boldsymbol{\psi}_i(t_k) = \mathbf{0} \tag{5}$$

where $\mathbf{C}(t_k)$ is the covariance matrix at the k^{th} time step, \mathbf{I}^P is the $m \times m$ identity matrix, $\boldsymbol{\psi}_i(t_k)$ is the i^{th} eigenvector at the k^{th} time step corresponding to the eigenvalue $\lambda_i^P(t_k)$, and $\boldsymbol{\psi}_i(t_k) = [\psi_{i_1}(t_k), \dots, \psi_{i_2}(t_k), \dots, \psi_{i_m}(t_k)]^T$ is the component corresponding to the j^{th} sensor.

2.1.2. DWPCA

To overcome the limitations of PCA and MPCA, the DWPCA method integrates spatial and temporal windows into PCA. The spatial window is used to exclude sensors that are insensitive to damage prior to analysis, while the temporal window is used to distinguish between different structural states. The combined use of spatial and temporal windows aims to improve the ability to identify damage.

The first step is to determine the spatial windows. These windows should include a set of sensors that show significant changes after the occurrence of damage. If the finite element (FE) model of the structure is available, damage-sensitive sensors can be identified by simulating different damage scenarios. In cases where FE models are not available, empirical experience can be used to determine the spatial windows. For example, grouping neighbouring sensors to form each spatial window is a feasible approach, with all sensors distributed in a total of w spatial windows. In this paper, the spatial window is defined as $SW = \{m_1, m_2, \dots, m_s\}$ where s represents the size of the spatial window and m_1, m_2, \dots, m_s denotes the sensor serial numbers. For example, $\{2, 4, 6\}$ denotes that Sensors 2, 4 and 6 are grouped within the spatial window and is equal to 3.

Once the spatial window has been determined, the matrix \mathbf{U} representing the structural responses of the sensors within the spatial and temporal windows can be expressed as follows:

$$\mathbf{U}(t_k) = \begin{bmatrix} u_{m_1}(t_k) & \cdots & u_{m_s}(t_k) \\ \vdots & \ddots & \vdots \\ u_{m_1}(t_{k+T_w-1}) & \cdots & u_{m_s}(t_{k+T_w-1}) \end{bmatrix} \tag{6}$$



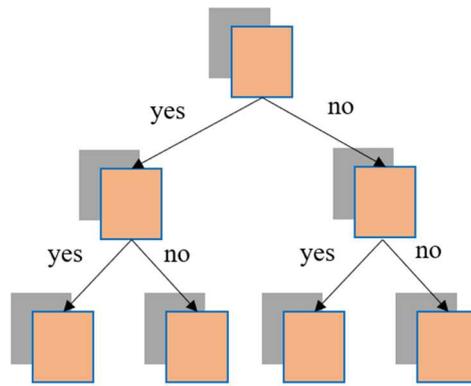


Fig. 2. Architecture of the Decision Tree.

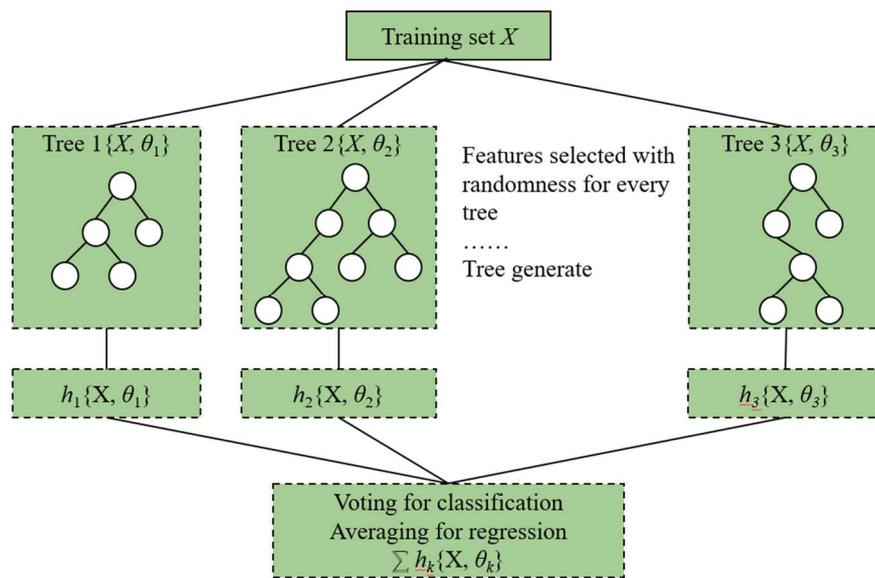


Fig. 3. Architecture of the Random Forest.

Subsequently, by repeating the steps of MPCA, we can obtain the time-variant eigenvectors $\psi_i(t_k) = [\psi_{i_{m_1}}(t_k), \dots, \psi_{i_{m_2}}(t_k), \dots, \psi_{i_{m_s}}(t_k)]^T$. Previous research has shown that the first-order and second-order eigenvectors of PCA capture more than 90% of the structural information. Therefore, this study investigates damage detection using the first-order and second-order eigenvectors by DWPCA. In addition, several damage-sensitive features based on DWPCA eigenvectors are proposed and applied as inputs to ML and DL algorithms for damage localization and quantification.

2.2. Machine learning algorithms

As mentioned in the previous section, the detection of damage located far from sensors or of minimal severity remains a challenge for both the MPCA and DWPCA methods. If novel machine learning (ML) algorithms can be applied to establish the relationship between DWPCA eigenvectors and structural state, it is expected to further improve the detectability of damage. In this paper, several novel and representative AI algorithms are incorporated and combined with DWPCA features for damage identification. These algorithms include Decision Tree (DT), Random Forest (RF), Extreme Gradient Boosting (XGBoost), AutoGluon and Transformer. The detailed principles of these algorithms are presented below.

2.2.1. Decision tree

The Decision Tree (DT) algorithm is a supervised ML technique commonly used for classification and regression tasks [36]. It constructs a hierarchical structure in the form of a tree, where internal nodes represent features or attributes, and leaf nodes correspond to class labels or predicted values, as shown in Fig. 2. The decision tree algorithm recursively partitions the data set based on feature values, aiming to optimize information gain or minimize impurity at each partition. At each internal node, a specific feature is selected as the splitting criterion. Several measures, including information gain, entropy and Gini impurity, are used to assess the quality of a split. Information gain measures the reduction in entropy or impurity achieved by splitting the data based on a particular feature. Entropy quantifies the disorder or uncertainty within a set of class labels, while Gini impurity measures the probability of misclassifying a randomly selected element from the data set. The decision tree algorithm continues to partition the data until a pre-defined stopping criterion is met, such as reaching a maximum depth limit, having a minimum number of samples at a node, or encountering diminishing improvements from further partitioning. To combat overfitting, pruning techniques can be used to remove or merge parts of the tree to improve the generalization performance of the model. However, decision trees are susceptible to overfitting, particularly if they become too deep and complex. Ensemble methods, such as random forests and boosting algorithms, are often used to mitigate this problem and improve the predictive accuracy of decision trees.



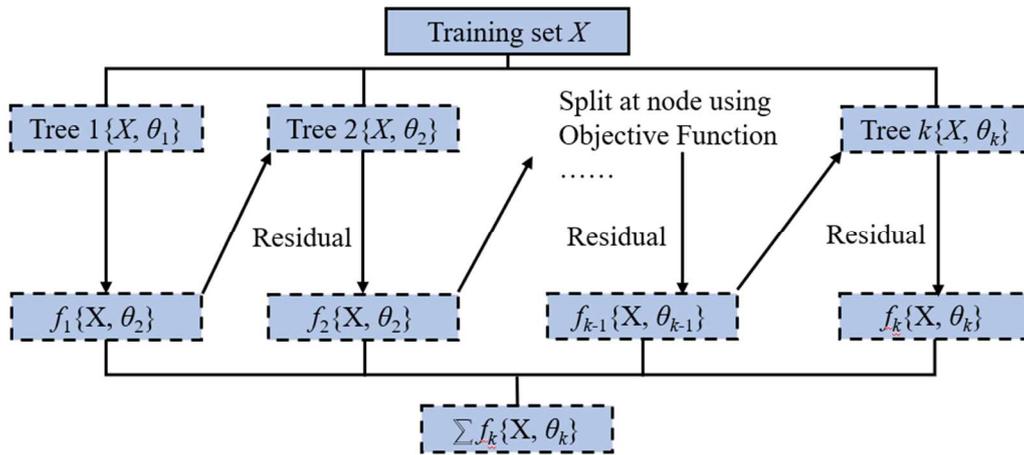


Fig. 4. Architecture of XGBoost.

2.2.2. Random forest

Random forest is an ensemble machine learning method that constructs numerous decision trees during training and outputs the class that is averaged or voted by each individual tree [37]. It was proposed by Breiman in 2001 as an improvement over the bagging method, with an additional layer of randomness. RF can be applied in regression and classification. A schematic diagram of RF is presented in Fig. 3.

Bagging, short for bootstrap aggregating, is an ensemble algorithm designed to enhance the stability and accuracy of individual predictive models such as tree. Bagging helps decision trees to reduce their variance and the influence of overfitting. Given a training set $X = x_{1,2,\dots,n}$ with responses $Y = y_{1,2,\dots,n}$, bagging repeats K times to randomly select a sample with replacement from the training set and fits trees to these samples. In our research problem, X represents the eigenvectors extracted from the strains of the benchmark model, as depicted in Fig. 4. Y represents the labels indicating damage locations or severities of the benchmark model. A tree $h(x, \theta_k)(k = 1, 2, \dots, K)$ is trained every time, where the θ_k are independent and identically distributed random vectors. After training, the resulting prediction model can be established by averaging the predictions from K regression trees or by taking the majority vote from K decision trees.

2.2.3. Extreme gradient boosting

Extreme Gradient Boosting (XGBoost) is a scalable machine learning system for tree boosting that was proposed by Chen and Guestrin in 2016. The superior performance of XGBoost in supervised machine learning is the reason it was chosen to train the structural damages classifier in this study. Gradient boosting serves as the foundation for XGBoost, where weak base learning models are combined into a stronger learner in an iterative manner [31]. As illustrated in Fig. 4, during each iteration of gradient boosting, the residual is utilized to refine the previous predictor so that the specified loss function can be optimized. In an enhancement, regularization is incorporated into the loss function to define the objective function in XGBoost for evaluating model performance, which is formulated as:

$$UJ(\theta) = L(\theta) + \Omega(\theta) \tag{7}$$

The parameters trained with the given data are denoted as θ . Here, L represents the training loss function, such as square loss or logistic loss, which evaluates how effectively the model fits the training data. Ω denotes the regularization term, such as L_1 norm or L_2 norm, which assesses the complexity of the model. Simpler models typically exhibit improved performance in mitigating overfitting issues. Given that the base model is a decision tree, the output of model \hat{y}_i is combined through voting or averaging by an ensemble F of k trees:

$$\hat{y}_i = \sum_{k=1}^k f_k(x_i), f_k \in F \tag{8}$$

The objective function at the t time iteration can be explicitly formulated as:

$$J^{(t)} = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^t \Omega(f_k) \tag{9}$$

where n represents the number of predictions. Here, $\hat{y}_i^{(t)}$ can be expressed as:

$$\hat{y}_i^{(t)} = \sum_{k=1}^t f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{10}$$

In reference [31], the regularization term $\Omega(f_k)$ for a decision tree is defined as:

$$\Omega(f_k) = \gamma_{xgbt} T_{tr} + \frac{1}{2} \lambda_{xgbt} \sum_{j=1}^T w_j^2 \tag{11}$$

Here, γ_{xgbt} denotes the complexity of each leaf, while T_{tr} represents the number of leaves in a decision tree. λ_{xgbt} is a scaling parameter for the penalty, and w is the vector of scores assigned to the leaves. To enhance the performance of gradient boosting, XGBoost employs a second-order Taylor expansion instead of the first-order approach utilized by typical gradient boosting. For instance, if the loss function is mean square error (MSE), the objective function can be derived as follows:



$$J^{(t)} \approx \sum_{i=1}^n [g_i w_{q(x_i)} + \frac{1}{2} (h_i w_{q(x_i)}^2)] + \gamma_{xgbt} T_{tr} + \frac{1}{2} \lambda_{xgbt} \sum_{j=1}^{T_{tr}} w_j^2 \tag{12}$$

with the constants removed. Here, $q(\cdot)$ is a function that assigns each data point to its corresponding leaf. g_j and h_i represent the first and second derivatives of the MSE loss function, respectively. In Eq. (12), the loss function is determined by summing the loss values for each data sample. As each data sample corresponds to only one leaf node, the loss function can also be expressed as the sum of loss values for each leaf node. Therefore, the objective function can be further simplified as:

$$J^{(t)} \approx \sum_{j=1}^{T_{tr}} [(\sum_{i \in P_j} g_i) w_j + \frac{1}{2} (\sum_{i \in P_j} h_i + \lambda_{xgbt}) w_j^2] + \gamma_{xgbt} T_{tr} \tag{13}$$

According to (13), G_j and H_j can be defined as:

$$G_j = \sum_{i \in P_j} g_i, H_j = \sum_{i \in P_j} h_i \tag{14}$$

where P_j represents all the data samples in leaf node j . Consequently, the optimization of the objective function can be reformulated as a problem of minimizing a quadratic function. In other words, following a split at a specific node in the decision tree, the alteration in model performance can be assessed based on the objective function. If the decision tree model performance improves after this node split, the change is incorporated; otherwise, the split is halted. Furthermore, due to the regularization applied during the optimization of the objective function, a predictive classifier can be trained to mitigate overfitting.

2.2.4. AutoGluon

AutoGluon is an advanced AutoML framework that represents a paradigm shift towards AutoML, as opposed to traditional approaches of algorithm selection and hyper-parameter optimization [35].

AutoGluon employs a groundbreaking multi-layer stacking strategy, as shown in Fig. 5. In this strategy, several layers are arranged in a stack. The first layer consists of multiple models whose outputs are concatenated to serve as the basis for subsequent layers. AutoGluon uses these layers of model types iteratively, allowing higher layers to access the original data features. The final layers of the stack incorporate an innovative ensemble selection mechanism to address overfitting. In addition, AutoGluon employs a k-fold bagging method to further improve stacking performance and implements repeated bagging processes to mitigate overfitting. In particular, AutoGluon adopts a unique training strategy by immediately saving trained models to disk. This approach allows it to meet time constraints and skip potentially unsuccessful models based on their results. In addition, AutoGluon employs a unique workflow strategy that trains models sequentially, using multi-task optimization to mitigate potential memory management errors.

AutoGluon's pioneering design demonstrates that AutoML frameworks should focus on more than just the CASH (Classification, Accuracy, Speed and Hyperparameters) problem. It offers several features that other frameworks do not. For example, its preprocessing model can handle heterogeneous datasets. It also has a powerful multi-layer stacking design integrated with advanced neural network structures that outperform conventional Bayesian optimization approaches. Importantly, AutoGluon maintains flexibility without introducing unnecessary complexity. Users can import their own models and use a simple API that supports training resumption.

2.2.5. Swin-transformer

The Swin Transformer block is an essential component in the encoder block, decoder block, or bottleneck [38]. Unlike the traditional multi-head self-attention module, the Swin Transformer block is constructed by the multi-head attention module based on a fixed window and a shifted window. Figure 6 shows the structure of the Swin Transformer block, which consists of four LayerNorm (LN) layers, a window-based multi-head self-attention module (W-MSA), a shifted window-based multi-head self-attention module (SW-MSA), and two multilayer perceptron (MLP) layers with a nonlinear activation function GeLU.

The W-MSA divides the feature map into several non-overlapping windows and computes the local self-attention within each window. This approach significantly reduces computational complexity while preserving global attention. Meanwhile, SW-MSA is proposed to address the lack of cross-window connection in W-MSA and improve the overall performance of the model. Based on SW-MSA, cross-window connection is introduced while maintaining computational efficiency. The main difference between SW-MSA and W-MSA is the inclusion of a window shift operation prior to W-MSA. This minor modification allows the network model to better handle global information. In practical implementation, shifting the feature map achieves the same effect. W-MSA and SW-MSA are used alternately to form a Swin Transformer block, which can be represented by the following formulas:

$$\hat{x}^l = \text{W-MSA}(\text{LN}(x^{l-1})) + x^{l-1} \tag{15}$$

$$x^l = \text{MLP}(\text{LN}(\hat{x}^l)) + \hat{x}^l \tag{16}$$

$$\hat{x}^{l+1} = \text{SW-MSA}(\text{LN}(x^l)) + x^l \tag{17}$$

$$x^{l+1} = \text{MLP}(\text{LN}(\hat{x}^{l+1})) + \hat{x}^{l+1} \tag{18}$$

$$\text{LN}(x) = \frac{x - E[x]}{\sqrt{\text{Var}(x) + \varepsilon}} \cdot \gamma_{trsfm} + \beta \tag{19}$$

Here, \hat{x}^l and x^l denote the feature maps output by the i^{th} SW-MSA module and MLP module, respectively. $E[x]$ represents the mean of x , $\text{Var}(x)$ represents the variance of x , ε is typically set to 1×10^{-5} , and γ_{trsfm} and β are learnable affine transform parameters. When computing self-attention, following previous studies, we first use a MLP layer to map the input embedding feature vector into three vectors of equal size: query, key and value. We then compute the similarity between the query and the key, obtain an attention weight matrix through a softmax layer, and finally multiply the attention weight matrix by the value to obtain the final output. The self-attention formula is given by:

$$\text{Attention}(Q, K, V) = \text{SoftMax}(\frac{QK^T}{\sqrt{d}} + B)V \tag{20}$$



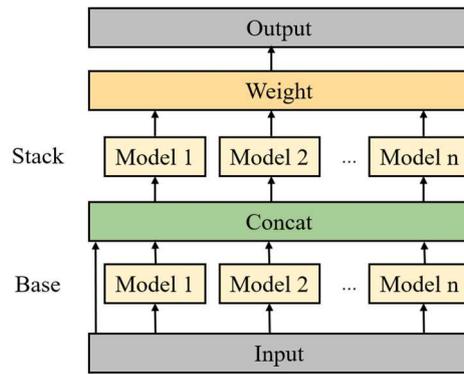


Fig. 5. AutoGluon's multi-layer stacking strategy.

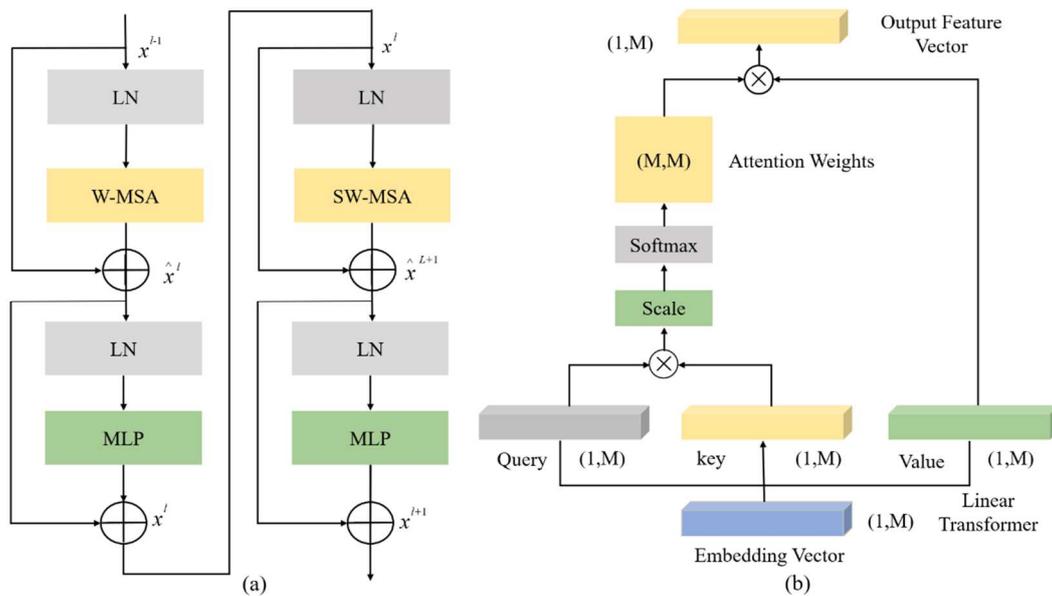


Fig. 6. Overview of swin transformer block and the multi-head self-attention (MSA) module. (a) Architecture of swin transformer block; (b) Calculation flow of multi-head self-attention (MSA) module.

where $Q, K, V \in R^{M^2 \times d}$ are the query, key, and value vectors, respectively. M denotes the window size for calculating local self-attention, d denotes the dimension size of the query or key, and $B \in R^{M^2 \times M^2}$ denotes the offset matrix. Since storing a $M^2 \times M^2$ matrix requires a significant amount of memory, and the relative position of the patch in the window is typically in the range of $[-M + 1, M - 1]$, $\hat{B} \in R^{(2M-1) \times (2M-1)}$ is commonly used to store the value of B .

3. Finite Element Model and Simulated SHM Dataset

Firstly, in order to investigate the damage identification capability of ML algorithms using DWPCA eigenvectors, it is crucial to construct a comprehensive and complete dataset that includes sensor responses under different damage scenarios. However, due to safety concerns, it is not feasible to induce actual damage to obtain responses in practical engineering applications. Therefore, an accurate finite element model has been developed to generate a larger dataset of responses that includes numerous damage scenarios. For this study, we used a benchmark model developed by the Structural Health Monitoring (SHM) Task Group of the International Association for Structural Control (IASC) and the American Society of Civil Engineers (ASCE). It was constructed in the University of British Columbia (UBC) laboratory. It comprises both a physical laboratory model and an open-source finite element model. However, the laboratory benchmark model lacks temperature-induced strain data, particularly a dataset of strain responses with damage labels under various damage scenarios. Considering that the open-source finite element model has been calibrated using measured acceleration data [3, 4] in this paper, we employed the refined benchmark model in this paper to generate a comprehensive dataset of strain responses under different damage scenarios, taking temperature loads into account.

The benchmark model represents a four-story, two-by-two bay steel braced frame as shown in Fig. 7. The plan dimensions of the model are 2.5 m \times 2.5 m, with a total height of 3.6 m. Detailed properties of the structural members are documented in Table 1.

In practical engineering applications, strain measurements are commonly used to monitor responses in SHM. Variations in strain are mainly due to seasonal and daily temperature variations. Therefore, a temperature load is applied to the benchmark model to obtain strain responses. Temperature data collected over four years from the Xijiang Bridge in Zhaoqing, China are used for this purpose. The sampling frequency is set at 4 measurements per day. The temperature loading includes both seasonal and daily temperature variations as shown in Fig. 8a. In addition, other factors such as traffic and wind are included as noise sources in the finite element model. In this paper, within the virtual monitoring system, sensors are placed at the center point of each brace in the benchmark model. The sensors are numbered sequentially from the bottom to the top layer. For example, the sensors on the bottom layer are labelled as Sensor 1, Sensor 2, ..., Sensor 8, while the sensors on the top layer are labelled as Sensor 25, Sensor 26, ..., Sensor 32.



Table 1. The properties of the structural members in the benchmark model.

Property	Columns	Floor Beams	Braces
Section type	B100×9	S75×11	L25×25×3
Cross-sectional area A [m ²]	1.133×10^{-3}	1.43×10^{-3}	0.141×10^{-3}
Moment of inertia I_x [m ⁴]	1.97×10^{-6}	1.22×10^{-6}	0
Moment of inertia I_y [m ⁴]	0.664×10^{-6}	0.249×10^{-6}	0
St. Venant torsion constant J [m ⁴]	8.01×10^{-9}	38.2×10^{-9}	0
Young's modulus E [Pa]	2×10^{11}	2×10^{11}	2×10^{11}
Shear modulus G [Pa]	$E/2.6$	$E/2.6$	$E/2.6$
Mass per unit volume ρ [kg/m ³]	7800	7800	7800

To simulate damage, the stiffness of the braces is reduced to replicate real world conditions. The model consists of a total of 32 braces representing 32 different damage locations. Damage severity is assumed to be a reduction in stiffness from 10% to 90% in 10% increments, and damage with a 1% reduction in stiffness is also included. Consequently, there are 11 categories of damage severity, including 10 types of damage severity and one healthy category. This results in a total of 321 scenarios, comprising 320 damage scenarios with 32 different damage locations and 10 different damage severities, and one health scenario. Permanent damage at a given location is introduced at the beginning of the third year, specifically after the 2920th measurement. Strain responses are collected from these sensors over a period of four years, the first two years in the undamaged state and the last two years in the damaged state. A total of 5840 measurements are obtained, consisting of four daily measurements over four years. Fig. 8b shows an example of time-varying strain acquired by Sensor 1 when Brace 1 experiences a 10% stiffness reduction under noise with a signal-to-noise ratio (SNR) of 20dB. It can be observed that the strain levels increase after the occurrence of damage. However, the strain changes associated with damage can be masked by ambient noise or when the damage is minimal. It is therefore essential to derive features that are sensitive to damage while remaining insensitive to environmental changes.

In this paper, we apply DWPCA and MPCA to extract damage-sensitive features as inputs for AI algorithms to localize and quantify damage. During the computation, MPCA uses all sensors to obtain the corresponding eigenvectors. In contrast, DWPCA uses only a subset of sensors within spatial windows to derive its eigenvectors. Specifically, for DWPCA, we consider the simulation results under different damage scenarios, where each spatial window includes three neighbouring sensors. Thus, there are a total of 30 spatial windows in DWPCA. The spatial windows used in DWPCA are listed below:

Spatial Window 1: [1 2 3]; Spatial Window 2: [2 3 4]; ..., Spatial Window 30: [30 31 32].

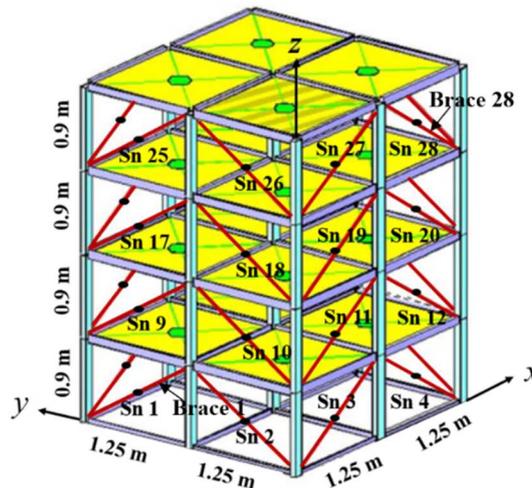


Fig. 7. Benchmark model proposed by the IASC-ASCE Structural Health Monitoring Task Group.

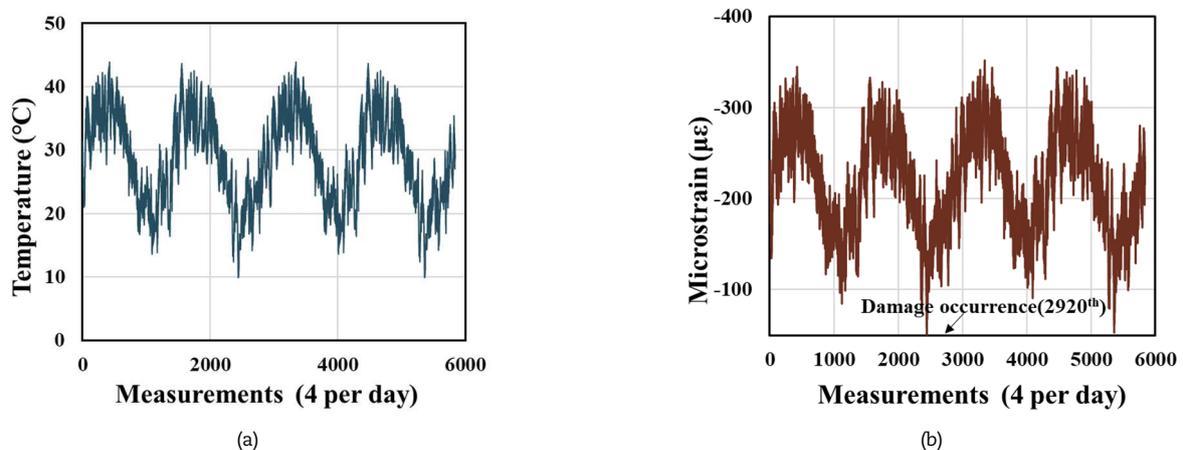


Fig. 8. Response evolution of the benchmark model over four years: (a) Temperature variation; (b) Strain responses in a damage scenario with 10% stiffness reduction in Brace 1 under 20dB noise.



4. Results and Discussion

Section 4.1 investigates the damage detectability of DWPCA eigenvectors, focusing on the sensitivity and robustness of the first and second DWPCA eigenvectors. Sections 4.2 and 4.3 propose different types of DWPCA features that are integrated with the widely recognized XGBoost algorithm for damage localization and quantification, respectively. The effectiveness of these different DWPCA features combined with the XGBoost algorithm is evaluated in terms of damage localization and quantification.

Furthermore, Section 4.4 highlights the emergence of AutoGluon, which is an excellent exceptional AutoML framework, and explores its combination with DWPCA features for damage localization and quantification. In addition, a comprehensive comparison is presented between AutoGluon and a number of representative ML algorithms, including DT and RF, as well as novel DL algorithms such as Transformer. This comprehensive comparison aims to assess the performance of AutoGluon in damage identification using DWPCA features.

4.1. A comparative study between MPCA and DWPCA

As shown in Fig. 8b, the strain changes subtly when the damage is minimal. If the original strains are directly used as inputs for ML algorithms, our previous studies have demonstrated that ML performs poorly in damage identification [23]. Therefore, it is necessary to extract damage-sensitive features from strains using DWPCA, which will serve as the inputs for ML algorithms. In this section, we first evaluate the damage sensitivity and robustness of the first-order and second-order eigenvectors obtained through DWPCA. For ease of comparison, the eigenvectors derived from MPCA are also included in the discussion.

Figure 9 shows the first-order eigenvectors of MPCA and DWPCA using Spatial Window 1 under 40dB and 20dB noise, respectively. As can be seen from the first-order eigenvectors obtained by MPCA and DWPCA using Spatial Window 1, both show significant shifts after the occurrence of damage. This phenomenon is attributed to the unstable phase between the 2920th and the 4380th measurement after the occurrence of the damage. During this phase, the time window includes strains from both undamaged and damaged states, with an increasing proportion of strains from the damaged state as the time window moves forward. After the 4380th measurement, the first-order eigenvector stabilizes as strain responses within the time window are exclusively from the damaged state. Notably, the first-order eigenvector derived from DWPCA shows more pronounced changes compared to MPCA. This difference can be attributed to the exclusion of insensitive sensors and the improved sensitivity of the damage identification features achieved by the DWPCA method.

In contrast to the original strains and the first-order eigenvector, the second-order eigenvector undergoes the most significant change under 40dB noise after damage occurrence, as shown in Fig. 10a. Between the 2920th and 4380th measurements, the second-order eigenvector undergoes a sharp transition, while the first-order eigenvector changes gradually. When the strain responses within the analyzed time window originate exclusively from either undamaged or damaged states, the first-order eigenvector exhibits stability with minimal variation, while the noise influences the second-order eigenvector, resulting in its random state. However, when the time window includes data from both healthy and damaged states, the second-order eigenvector is predominantly influenced by the data from the damaged state. Consequently, the second-order eigenvector remains stable within the 2920th and 4380th measurements. Despite the higher sensitivity to damage in an ideal environment, the second-order eigenvector shows poor robustness to noise, as shown in Fig. 10b. Tracking the evolution of the second-order eigenvector becomes difficult under high levels of noise, thereby hindering damage identification.

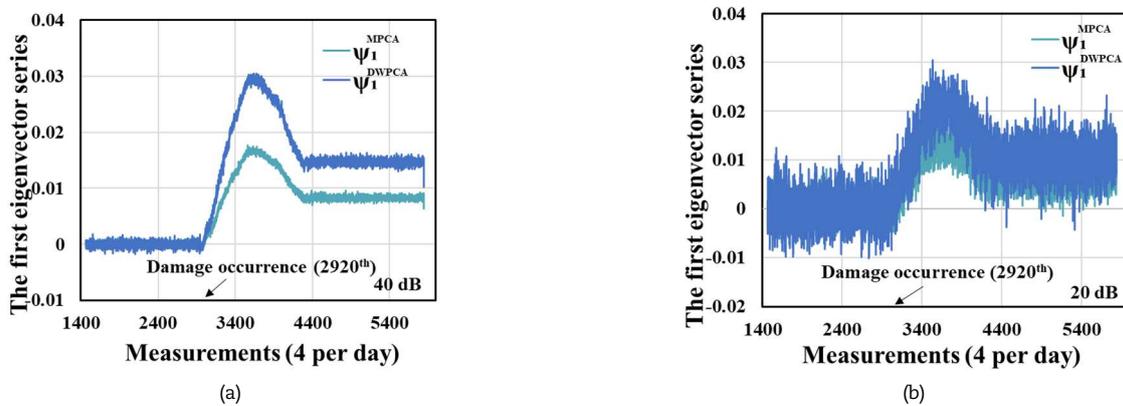


Fig. 9. Evolution of the first-order eigenvectors by MPCA and DWPCA for the damage located in Brace 1 with 10% stiffness reduction at different SNRs: (a) 40dB; (b) 20dB.

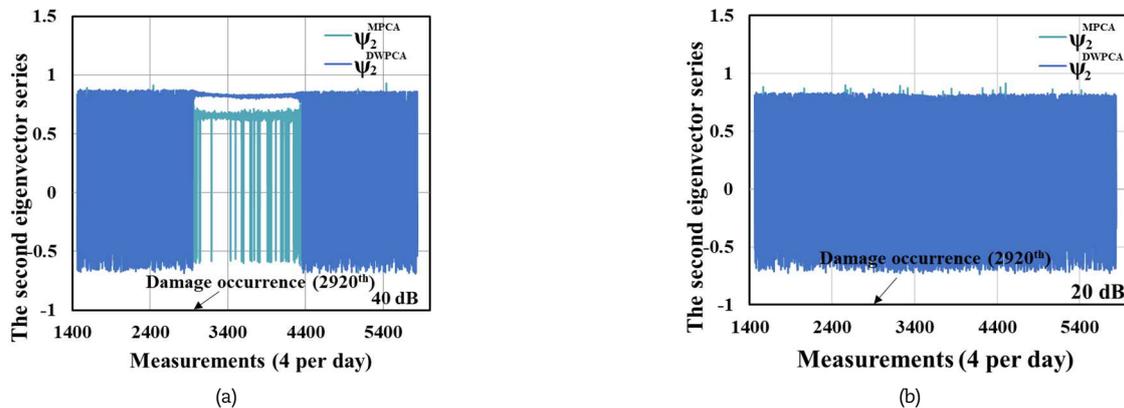


Fig. 10. Evolution of the second-order eigenvectors by MPCA and DWPCA for the damage located in Brace 1 with a 10% stiffness reduction at different SNRs: (a) 40dB; (b) 20dB.



In summary, both the first-order and second-order eigenvectors show improved sensitivity to small damage. In addition, DWPCA consistently shows higher sensitivity to damage than MPCA. This discrepancy is due to DWPCA's use of a spatial window to exclude insensitive sensors, thereby enhancing its ability to identify damage. Notably, the second-order eigenvector also shows significant sensitivity to damage, albeit with poor robustness. Consequently, the following section focuses on the use of the first-order and second-order eigenvectors derived from both MPCA and DWPCA as inputs to ML algorithms, enabling the evaluation and demonstration of the damage identification capability of these novel features.

4.2. Damage localization

The XGBoost algorithm is considered to be a typical ensemble learning method with excellent performance in classification problems. It is a modification of the traditional gradient boosting decision tree (GBDT) algorithm. In this section, we evaluate the detectability of MPCA and DWPCA features for damage localization using the XGBoost algorithm. The MPCA and DWPCA methods transform the strain responses into corresponding eigenvector time series. Each damage scenario contains 2920 eigenvectors ranging from the 2921st to the 5840th sampling step due to the moving time window applied by MPCA or DWPCA. The data set consists of 321 scenarios and 937,320 records. Note that 80% of the dataset is used for training and the remaining 20% for testing. For the damage localization problem, there are 33 prediction classes, including 32 locations of all braces in the model and one healthy scenario, as shown in Fig. 7. The six input types are listed as follows:

- (1) ψ_1^{MPCA} : the first-order eigenvector extracted by MPCA;
- (2) ψ_2^{MPCA} : the second-order eigenvector extracted by MPCA;
- (3) $\psi_{1,2}^{MPCA}$: the combination of first-order and second-order eigenvectors extracted by MPCA, as represented by $[\psi_1^{MPCA}, \psi_2^{MPCA}]$;
- (4) ψ_1^{DWPCA} : the first-order eigenvector extracted by DWPCA;
- (5) ψ_2^{DWPCA} : the second-order eigenvector extracted by DWPCA;
- (6) $\psi_{1,2}^{DWPCA}$: the combination of first-order and second-order eigenvectors extracted by DWPCA, as represented by $[\psi_1^{DWPCA}, \psi_2^{DWPCA}]$.

The accuracy of the ML algorithms is defined as the percentage of correctly classified records over all samples in the test dataset. Fig. 11 illustrates the prediction accuracy of the XGBoost algorithm using different inputs for damage localization at different SNRs. It can be seen that regardless of the input, the prediction accuracy of the XGBoost algorithm decreases as the noise intensity increases. This can be attributed to the increased overlap of the input curves from different damage scenarios as the noise intensity increases.

With respect to the three inputs including ψ_1^{DWPCA} , ψ_2^{DWPCA} and $\psi_{1,2}^{DWPCA}$, extracted by DWPCA, the results shown in Fig. 11a indicate that $\psi_{1,2}^{DWPCA}$ outperforms the other two features. In addition, it shows a better performance in terms of damage localization accuracy. The input $\psi_{1,2}^{DWPCA}$, which combines first-order and second-order eigenvectors, shows the best performance among the three DWPCA features. This is because the two eigenvectors provide complementary information for damage localization. The first-order eigenvector ψ_1^{DWPCA} have good robustness, while the second-order eigenvector ψ_2^{DWPCA} have higher sensitivity to damage. Consequently, it achieves a better performance compared to ψ_1^{DWPCA} and ψ_2^{DWPCA} . Fig. 11b illustrates the damage localization accuracy of XGBoost using three MPCA features at different SNRs. The accuracy of ψ_1^{MPCA} is similar to that of $\psi_{1,2}^{MPCA}$, with the two curves nearly overlapping. However, the prediction accuracy of ψ_2^{MPCA} is the lowest. It is worth noting that, regardless of whether using the first-order eigenvector, the second-order eigenvector, or the combination of first-order and second-order eigenvectors, DWPCA consistently demonstrates higher prediction accuracy compared to MPCA at different SNRs. It is worth noting that the eigenvectors obtained by DWPCA consistently show better performance than those obtained by MPCA over different noise intensities. This can be attributed to the effectiveness of DWPCA in excluding damage-insensitive sensors and enhancing the damage sensitivity of the inputs. As a result, the XGBoost algorithm shows improved accuracy in discriminating and localizing different damage states. In other words, in addition to $\psi_{1,2}^{DWPCA}$, extracted by the DWPCA method, which combines ψ_1^{DWPCA} and ψ_2^{DWPCA} , the XGBoost algorithm achieves the best performance in damage localization.

Figure 12 shows the spatial distribution of the prediction accuracy of the XGBoost algorithm using the three DWPCA features for the benchmark model under 20dB noise, focusing on damage localization. Similar to the result shown in Fig. 11a, in each damaged scenario, $\psi_{1,2}^{DWPCA}$ as inputs has the most stable performance and high accuracy in damage localization, followed by ψ_2^{DWPCA} and ψ_1^{DWPCA} . Furthermore, it can be seen that XGBoost with DWPCA features slightly outperforms the other two features when the damage is located at the bottom. The strain responses and DWPCA eigenvectors undergo significant changes when the damage is located in the bottom layer or near the fixed supports. For each damage scenario, $\psi_{1,2}^{DWPCA}$ extracted by DWPCA shows the best performance in damage localization. This is because DWPCA effectively excludes the damage-insensitive sensors and improves the damage sensitivity of the inputs.

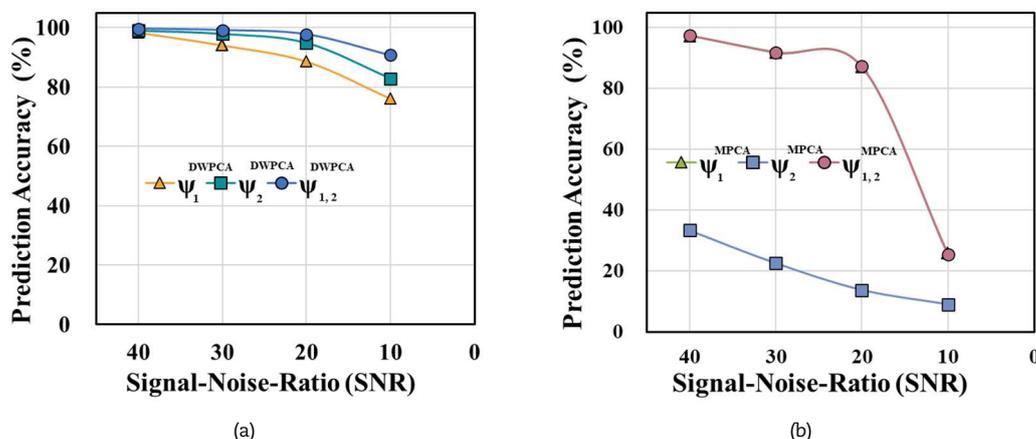


Fig. 11. Damage localization prediction accuracy of XGBoost using six types of inputs extracted by DWPCA and MPCA at different SNRs: (a) DWPCA; (b) MPCA.



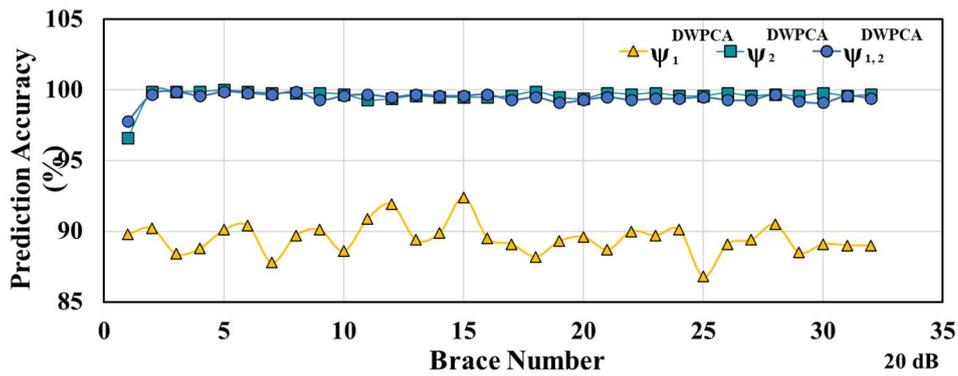


Fig. 12. Distribution of prediction accuracy of XGBoost for damage localization with three types of DWPCA features as inputs under 20 dB noise.

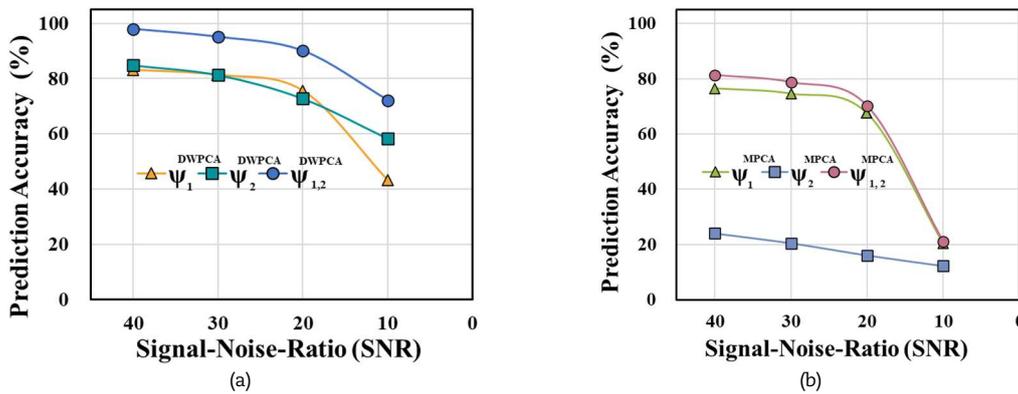


Fig. 13. Damage quantification prediction accuracy of XGBoost using six types of inputs extracted by DWPCA and MPCA at different SNRs: (a) DWPCA; (b) MPCA.

4.3. Damage quantification

This section evaluates the performance of the XGBoost algorithm using DWPCA and MPCA features for damage quantification. The classification system consists of 11 classes representing different levels of damage severity and one healthy scenario. A comparative study is conducted to evaluate the effectiveness of six types of DWPCA and MPCA inputs for damage quantification.

Figure 13 illustrates the accuracy of XGBoost with DWPCA and MPCA features as inputs at different SNRs. Similar to the damage localization results discussed in Section 4.2, the use of DWPCA eigenvectors significantly improves the accuracy of damage quantification compared to MPCA features. Regardless of the noise level, DWPCA features perform better in damage quantification than MPCA features. Specifically, at 20dB noise, the accuracy of $\psi_{1,2}^{DWPCA}$ improves by 19% compared to $\psi_{1,2}^{MPCA}$, the accuracy of ψ_1^{DWPCA} improves by 8% compared to ψ_1^{MPCA} , and the accuracy of ψ_2^{DWPCA} improves by 56% compared to ψ_2^{MPCA} . This improvement can be attributed to the preservation of damage sensitivity responses in DWPCA while eliminating insensitivity, thereby increasing the sensitivity of the DWPCA eigenvectors. Among the three DWPCA features, combining the first-order and second-order eigenvectors as inputs yields higher prediction accuracy than using either eigenvector alone, as shown in Fig. 13a. For example, at 20dB noise, the corresponding accuracy of $\psi_{1,2}^{DWPCA}$ is 14% higher than ψ_1^{DWPCA} and 17% higher than ψ_2^{DWPCA} . The first-order eigenvector shows robustness, while the second-order eigenvector exhibits greater sensitivity to damage. Therefore, the combination of these features $\psi_{1,2}^{DWPCA}$ is better than using ψ_1^{DWPCA} or ψ_2^{DWPCA} individually as inputs.

Furthermore, we investigate the performance of XGBoost using DWPCA features for damage quantification across various damage severities under 20dB noise, as illustrated in Fig. 14. As the damage intensity increases, there is an upward trend in accuracy for both $\psi_{1,2}^{DWPCA}$ and ψ_2^{DWPCA} , whereas the accuracy of ψ_1^{DWPCA} fluctuates. Additionally, when compared to using ψ_1^{DWPCA} or ψ_2^{DWPCA} independently, $\psi_{1,2}^{DWPCA}$ achieves superior accuracy in damage quantification.

Figure 15 shows the improved accuracy of the combined feature $\psi_{1,2}^{DWPCA}$ compared to the other two types of features for damage quantification. Specifically, Fig. 15a illustrates the improved accuracy of $\psi_{1,2}^{DWPCA}$ for damage quantification in comparison to using ψ_1^{DWPCA} as input. It is evident that $\psi_{1,2}^{DWPCA}$ significantly improves the accuracy of predicting damage severities, except for a damage severity of 1%. In the range of damage severities from 20% to 80%, $\psi_{1,2}^{DWPCA}$ outperforms ψ_1^{DWPCA} with an improvement in accuracy of more than 10%. Due to the variability of the included damaged information, the combination of ψ_1^{DWPCA} and ψ_2^{DWPCA} synergistically improves the accuracy of damage severity prediction. However, for damages severity below 20% or above 80%, the improvement in accuracy is less than 10%. When the damage severity is below 10%, although ψ_2^{DWPCA} demonstrates higher sensitivity to damage, it is also susceptible to noise interference, which renders the contained damage information ineffective. For a damage severity of 90%, both ψ_1^{DWPCA} and ψ_2^{DWPCA} show comparable performance in predicting damage severity, suggesting the insignificance of the advantage offered by $\psi_{1,2}^{DWPCA}$. On the other hand, Fig. 15b shows the improved accuracy of $\psi_{1,2}^{DWPCA}$ for damage quantification in contrast to using ψ_2^{DWPCA} as input. Similar to the results in Fig. 15a, when the damage intensity reaches 90%, both ψ_1^{DWPCA} and ψ_2^{DWPCA} experience notable changes, resulting in diminished advantages of the combined features. However, unlike the results in Fig. 15a, the combined features $\psi_{1,2}^{DWPCA}$ show their advantages particularly for minor damages, especially for damage 1% and 10% damages. This can be attributed to the minimal variations observed in the second-order eigenvector, where the damage information it contains might be obscured by noise. By combining the robustness of ψ_1^{DWPCA} , the shortcomings of ψ_2^{DWPCA} are compensated, consequently improving the damage localization capability. In summary, $\psi_{1,2}^{DWPCA}$ exhibits the best performance and highest accuracies in damage quantification.



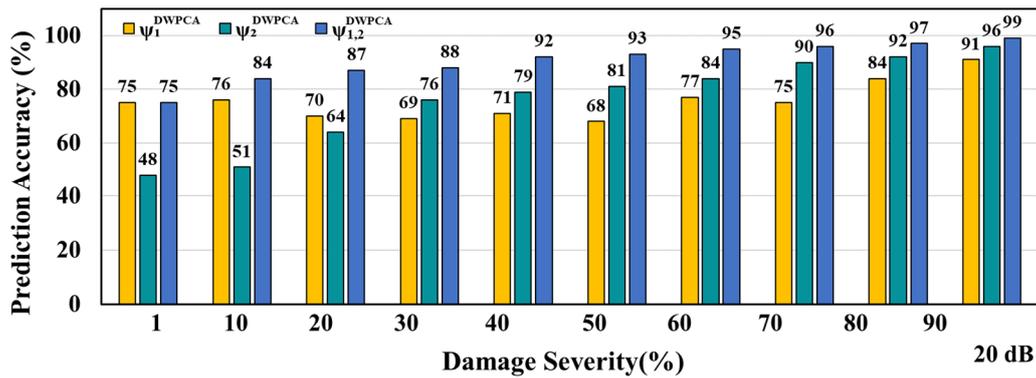


Fig. 14. Prediction accuracy of XGBoost for damage quantification with three types of DWPCA features under 20 dB noise at different damage severities.

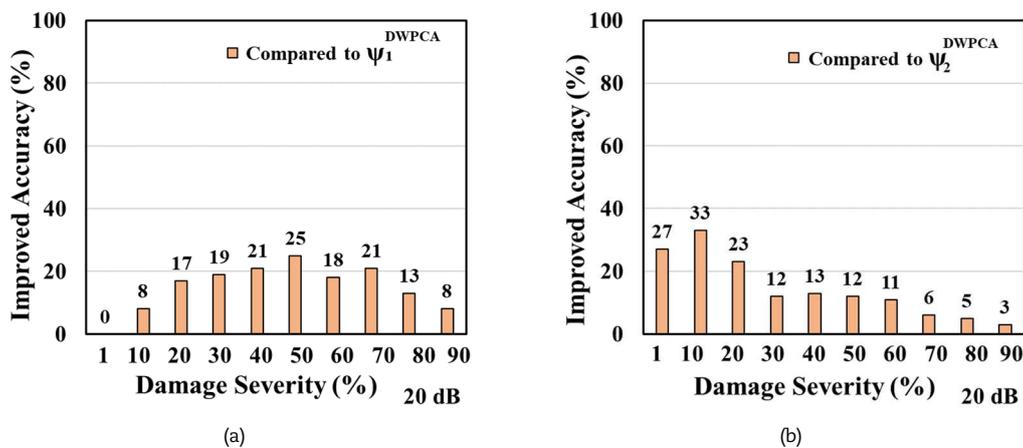


Fig. 15. Improved accuracy of the combined feature $\psi_{1,2}^{DWPCA}$ compared to the other two types of features for damage quantification: (a) ψ_1^{DWPCA} ; (b) ψ_2^{DWPCA} .

4.4. Different machine learning algorithms with DWPCA features

In the previous sections, we have concluded that the XGBoost algorithm shows excellent damage identification performance when using the combined DWPCA features $\psi_{1,2}^{DWPCA}$ compared to using individual DWPCA features (ψ_1^{DWPCA} or ψ_2^{DWPCA}) or the other three MPCA features. In this section, we further evaluate the damage detectability of the DWPCA features by combining them with representative and novel ML algorithms. The ML algorithms considered are DT and RF, which are representative and classical, and AutoGluon, which is a novel and highly regarded ML algorithm that has gained attention in recent years. These algorithms are combined with DWPCA features for damage identification.

Figure 16 shows the accuracy of the four ML algorithms using three types of DWPCA features for damage localization and quantification. Similar to the results in Section 4.1, when using DT, RF, and XGBoost algorithms, the combined DWPCA features $\psi_{1,2}^{DWPCA}$ achieve the best performance and highest prediction accuracy for damage identification compared to using individual DWPCA features. In addition, for AutoGluon, both $\psi_{1,2}^{DWPCA}$ and ψ_2^{DWPCA} achieve high prediction accuracies of over 98% for damage localization and quantification. Importantly, regardless of the inputs used, AutoGluon consistently achieves the highest prediction accuracies in damage localization and quantification compared to the other three ML algorithms. For example, when using $\psi_{1,2}^{DWPCA}$ as inputs, AutoGluon achieves an improved accuracy of 1%, 5% and 10%, compared to XGBoost, RF, and DT, respectively, for damage localization. Similarly, for damage quantification using $\psi_{1,2}^{DWPCA}$, AutoGluon achieves an improved accuracy of 8%, 8%, and 16%, compared to XGBoost, RF, and DT, respectively. This demonstrates that AutoGluon can effectively mitigate the effects of strong noise and exploit the structural damage information contained in the second-order eigenvector, resulting in a high level of accuracy in damage localization and quantification. Additionally, AutoGluon's combination of multiple models leads to higher accuracy and more reliable predictions.

Furthermore, Fig. 17 also shows the spatial distribution of prediction accuracy for the four ML algorithms using $\psi_{1,2}^{DWPCA}$, under 20dB noise. It is evident that AutoGluon consistently achieves an accuracy above 95% regardless of the damaged braces, followed by XGBoost, RF, and DT. Additionally, AutoGluon shows stable performance in predicting damage localization, followed by XGBoost and RF, while DT shows the most unstable performance. This indicates that AutoGluon leverages the strengths of different ML algorithms to maximize their advantages, achieving the highest prediction accuracy and stable performance. Meanwhile, XGBoost, as an ensemble learning method and based on a combination of multiple decision trees, improves the generalization performance of classification algorithms by combining multiple learners to vote on the classification results.

Figure 18 provides a detailed representation of the accuracy of each algorithm for each level of damage severity. Similar to Fig. 16, AutoGluon achieves the highest accuracy at all levels of damage severity, followed by XGBoost, RF, and DT. When the damage severity exceeds 10%, AutoGluon maintains an accuracy above 95%. Additionally, AutoGluon's advantage becomes more pronounced as the damage severity decreases. For example, compared to other algorithms, AutoGluon exhibits improvements of 11%, 11%, and 10% for a damage severity of 1%, compared to XGBoost, RF, and DT, respectively.



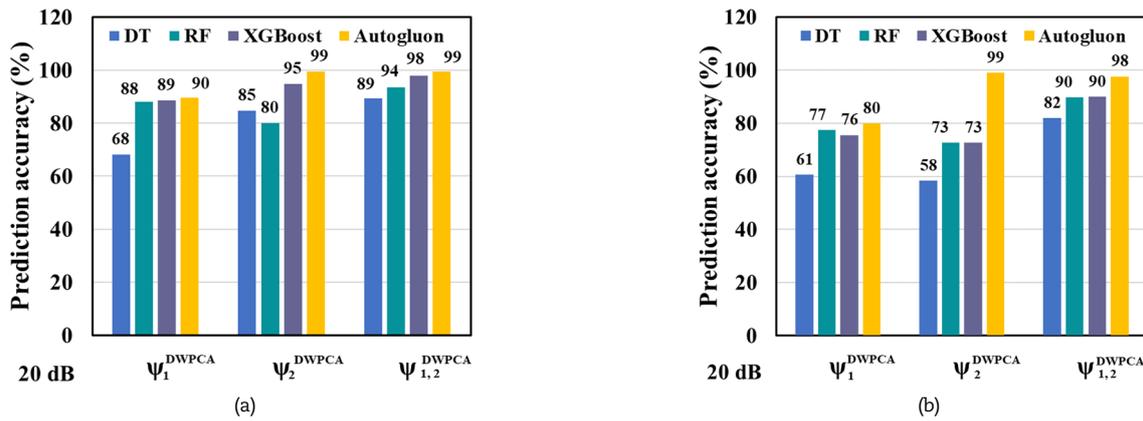


Fig. 16. The prediction accuracy of the four ML algorithms using three types of DWPCA features as inputs for damage localization under 20dB noise: (a) damage localization; (b) damage quantification.

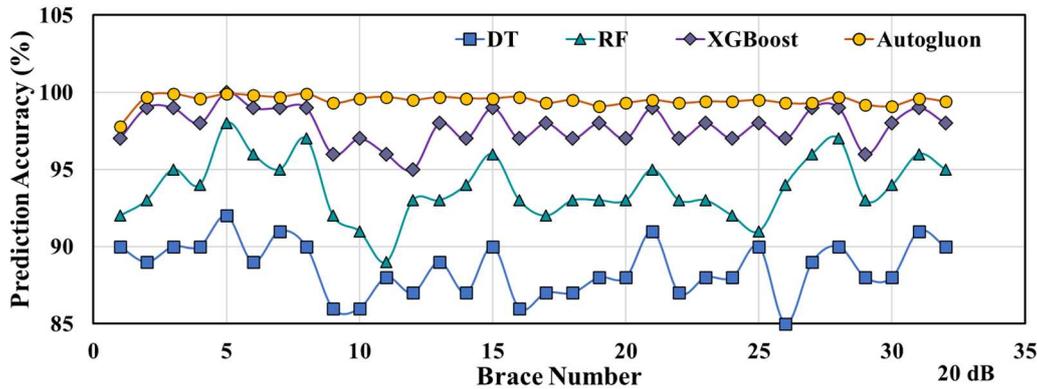


Fig. 17. Distribution of prediction accuracy of the four ML algorithms using $\psi_{1,2}^{DWPCA}$ for damage localization under 20dB noise.

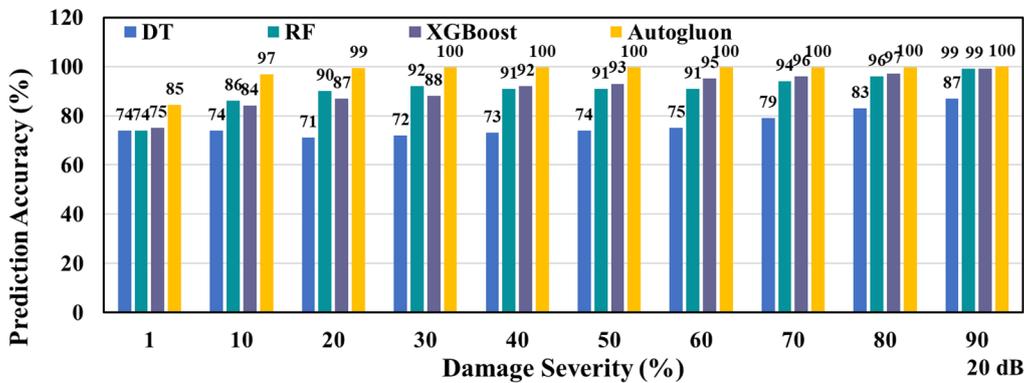


Fig. 18. Distribution of prediction accuracy of the four ML algorithms using $\psi_{1,2}^{DWPCA}$ for damage quantification under 20dB noise.

In conclusion, AutoGluon stands out as the best performer in damage localization and quantification regardless of the types of DWPCA features used as inputs. Its advanced techniques and ability to combine multiple models result in higher accuracy and more reliable predictions. The automated hyperparameter tuning feature also contributes to its superior model performance. In addition, AutoGluon's strength becomes more pronounced when dealing with minor damages. It should be noted, however, that apart from AutoGluon, XGBoost and RF demonstrate significant performance in damage identification when used with the combined DWPCA features $\psi_{1,2}^{DWPCA}$. These methods exploit the construction of multiple learners to vote on classification results, improving the overall performance and generalization capabilities of the classification algorithm. This advantage is particularly noticeable when dealing with strong noise in the data.

4.5. A Comparative study between AutoGluon and Transformer

Based on the previous study, it can be concluded that when using DWPCA features as inputs, AutoGluon achieves an accuracy of over 90% in both damage localization and damage quantification. Particularly, when $\psi_{1,2}^{DWPCA}$ are used, AutoGluon achieves accuracies of 99% and 98% in damage localization and quantification under 20dB noise, respectively. In this section, we introduce Transformer, a model known for its excellent performance in various intelligent recognition tasks, to address the problem of damage identification. For this purpose, we combine Transformer with DWPCA features.

Figure 19 shows the accuracy of AutoGluon and Transformer with three types of DWPCA features for damage localization and quantification. From the figure, it can be seen that for each type of DWPCA feature, AutoGluon outperforms Transformer in both damage localization and quantification tasks. In particular, in the case of damage quantification, AutoGluon improves the prediction accuracy by 19%, 66%, and 16% for ψ_1^{DWPCA} , ψ_2^{DWPCA} and $\psi_{1,2}^{DWPCA}$ as inputs, respectively.



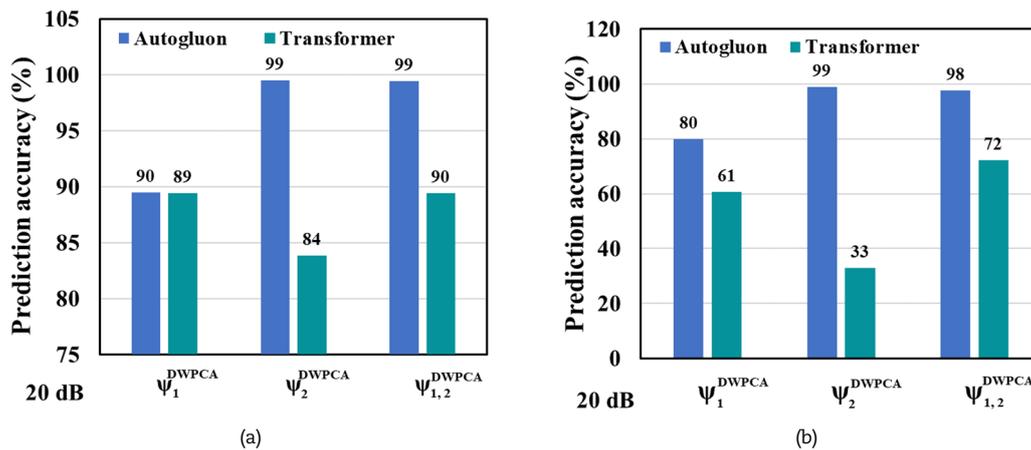


Fig. 19. Prediction accuracy of AutoGluon and Transformer with three types of DWPCA features as inputs for damage identification under 20dB noise: (a) damage localization; (b) damage quantification.

5. Conclusion

In this paper, we investigate the characteristics of the first-order and second-order eigenvectors obtained by DWPCA and propose three types of DWPCA features. These features are combined with various ML and DL algorithms for damage localization and quantification. Based on the observations and discussions in the previous section, the following conclusions can be drawn:

- (1) Regarding MPCA and DWPCA, it is found that the second-order eigenvector also shows considerable sensitivity to damage, although it lacks robustness.
- (2) For ML and DL algorithms, the use of DWPCA eigenvectors significantly improves the accuracy of damage quantification compared to MPCA features. This improvement can be attributed to the effectiveness of the spatial windows of DWPCA in excluding damage-insensitive sensors and enhancing the damage sensitivity of the inputs. Furthermore, the combined DWPCA feature $\psi_{1,2}^{DWPCA}$ results in the highest prediction accuracy and best performance in damage localization and quantification. The first-order eigenvector shows robustness, while the second-order eigenvector shows greater damage sensitivity. Therefore, the combination of these features outperforms using either the first-order or second-order eigenvector individually as inputs.
- (3) Regardless of the three types of DWPCA features, AutoGluon achieves the highest accuracy in damage localization and quantification, followed by XGBoost, RF, and DT. It suggests that ensemble learning using multiple ML algorithms, often leads to higher accuracy than using a single algorithm. Furthermore, with DWPCA features as input, AutoGluon outperforms Transformer in both damage localization and quantification tasks. In particular, in the case of damage quantification, compared to Transformer, AutoGluon improves the prediction accuracy by 19%, 66%, and 16% for ψ_1^{DWPCA} , ψ_2^{DWPCA} and $\psi_{1,2}^{DWPCA}$ as inputs, respectively.

In summary, the method proposed in this paper, which combines AutoGluon with the aforementioned features, exhibits higher sensitivity and better resolution in damage identification compared to other traditional methods. This demonstrates its potential for practical engineering applications. Validation using experimental data scenarios or real damage scenarios is essential. The next step in the research is to apply the proposed methodology to a database of measurements obtained from full-scale structures.

Author Contributions

Ge Zhang contributed to the final version of the article. She principally designed and generated this research paper. Ge Zhang, Wei Neng and Ying Zhou devised mathematical modeling, analyzed the theory validation, and provided the technical interpretation of numerical analysis results. Licheng Zhou and Gongfa Chen provided expertise and guidance throughout the research process. The detailed review carried out by Zejia Liu, Bao Yang, Zhenyu Jiang and Liqun Tang, extensively improved the quality of the work. The manuscript was written through the contribution of all authors. All authors discussed the results, reviewed the methodology, and approved the final version of the manuscript.

Acknowledgements

Not applicable.

Conflict of Interest

The authors declare no conflicts of interest regarding the research, authorship, or publication of this article.

Fundings

This work supported in part by the National Natural Science Foundation of China (Grant Nos. 11972162, 11932007, 12072115, and 12072116), the Natural Science Foundation of Guangdong Province, China (Grant No. 2023A1515012942), Postdoctoral Research Foundation of China (2021M700886), Guangzhou Science and Technology Project (SL2023A04J01633), and the State Key Laboratory of Subtropical Building and Urban Science, South China University of Technology (Grant No. 2022KA05).

Data Availability Statements

Not applicable.



Nomenclature

U	The response matrix	u	The response of one sensor
t_k	The time step	T_w	The size of a time window
N_S	The total number of sensor measurements	C	The covariance matrix
λ^P	The eigenvalues	ψ	The eigenvectors
I^P	The identity matrix	ψ	The eigenvector component
m	The total number of sensors	m_s	The sensor serial number in a spatial window
SW	Spatial windows	s	The size of a spatial window
X	The training data set in ML	Y	The response corresponding to X
x	A component in training data set X	y	A component in response data set Y
K	The total number of trees in RF	θ	The independent and identically distributed random vectors
L	The training loss function	Ω	The regularization term
\hat{y}_i	The output of the model	J	The objective function
F	An ensemble of trees	n	The prediction number
γ_{xobt}	The complexity of each leaf in XGBoost	T_{tr}	The number of leaves in a decision tree
λ_{xobt}	A scaling parameter for the penalty in XGBoost	w	The vector of scores assigned to the leaves
$q(\cdot)$	A function that assigns each data point to its corresponding leaf	g_i	The first derivatives of the MSE loss function
h_i	The second derivatives of the MSE loss function	P_i	All the data samples in leaf node j
\hat{x}^l, x^l	The feature maps output by the SW-MSA module and MLP module	$E[x]$	The mean of x in Transformer
$Var(x)$	The variance of x in Transformer	γ_{trsfm}	The learnable affine transform parameter
β	The learnable affine transform parameter	Q	The query vectors
V	The value vectors	K	The key vectors
M	The window size for calculating local self-attention	d	The dimension size of the query or key vectors
B	The offset matrix		

References

- [1] An, Y., Chatzi, E., Sim, S.H., Laflamme, S., Blachowski, B., Ou, J., Recent progress and future trends on damage identification methods for bridge structures, *Structural Control and Health Monitoring*, 26(10), 2019, e2416.
- [2] Sun, L., Shang, Z., Xia, Y., Bhowmick, S., Nagarajaiah, S., Review of Bridge Structural Health Monitoring Aided by Big Data and Artificial Intelligence: From Condition Assessment to Damage Detection, *Journal of Structural Engineering*, 146(5), 2020, 04020073.
- [3] Diao, Y., Lv, J., Wang, Q., Li, X., Xu, J., Structural damage identification based on variational mode decomposition–Hilbert transform and CNN, *Journal of Civil Structural Health Monitoring*, 13, 2023, 1415–1429.
- [4] Kumar, K., Biswas, P.K., Dhang, N., Time series-based SHM using PCA with application to ASCE benchmark structure, *Journal of Civil Structural Health Monitoring*, 10(5), 2020, 899–911.
- [5] Azim, M.R., Gül, M., Data-driven damage identification technique for steel truss railroad bridges utilizing principal component analysis of strain response, *Structure and Infrastructure Engineering*, 17(8), 2021, 1019–1035.
- [6] Jin, S.-S., Jung, H.-J., Vibration-based damage detection using online learning algorithm for output-only structural health monitoring, *Structural Health Monitoring*, 17(4), 2018, 727–746.
- [7] Sen, D., Erazo, K., Zhang, W., Nagarajaiah, S., Sun, L., On the effectiveness of principal component analysis for decoupling structural damage and environmental effects in bridge structures, *Journal of Sound and Vibration*, 457, 2019, 280–298.
- [8] Tibaduiza, D.A., Mujica, L.E., Rodellar, J., Güemes, A., Structural damage detection using principal component analysis and damage indices, *Journal of Intelligent Material Systems and Structures*, 27(2), 2016, 233–248.
- [9] Zhu, Y., Ni, Y.-Q., Jin, H., Inaudi, D., Laory, I., A temperature-driven MPCA method for structural anomaly detection, *Engineering Structures*, 190, 2019, 447–458.
- [10] Posenato, D., Kripakaran, P., Inaudi, D., Smith, I.F.C., Methodologies for model-free data interpretation of civil engineering structures, *Computers and Structures*, 88(7–8), 2010, 467–482.
- [11] Laory, I., Trinh, T.N., Posenato, D., Smith, I.F.C., Combined model-free data-interpretation methodologies for damage detection during continuous monitoring of structures, *Journal of Computing in Civil Engineering*, 27(6), 2013, 657–666.
- [12] Laory, I., Trinh, T.N., Smith, I.F.C., Evaluating two model-free data interpretation methods for measurements that are influenced by temperature, *Advanced Engineering Informatics*, 25(3), 2011, 495–506.
- [13] Nie, Z., Guo, E., Li, J., Hao, H., Ma, H., Jiang, H., Bridge condition monitoring using fixed moving principal component analysis, *Structural Control and Health Monitoring*, 27(6), 2020, e2535.
- [14] Malekzadeh, M., Gul, M., Kwon, I.B., Catbas, N., An integrated approach for structural health monitoring using an in-house built fiber optic system and non-parametric data analysis, *Smart Structures and Systems*, 14(5), 2014, 917–942.
- [15] Zhang, G., Tang, L., Liu, Z., Zhou, L., Sun, S., Enhanced features in principal component analysis with spatial and temporal windows for damage identification, *Inverse Problems in Science and Engineering*, 29(13), 2021, 2877–2894.
- [16] Zhang, G., Tang, L., Zhou, L., Liu, Z., Liu, Y., Jiang, Z., Principal Component Analysis Method with Space and Time Windows for Damage Detection, *Sensors*, 19(11), 2019, 2521.
- [17] Avci, O., Abdeljaber, O., Kiranyaz, S., Hussein, M., Gabbouj, M., Inman, D.J., A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications, *Mechanical Systems and Signal Processing*, 147, 2021, 107077.
- [18] Hou, R.X., Yong, Review on the new development of vibration-based damage identification for civil engineering structures: 2010–2019, *Journal of Sound and Vibration*, 491(1), 2021, 115741.
- [19] Tian, Y., Xu, Y., Zhang, D., Li, H., Relationship modeling between vehicle-induced girder vertical deflection and cable tension by BiLSTM using field monitoring data of a cable-stayed bridge, *Structural Control and Health Monitoring*, 28(2), 2021, e2667.
- [20] Ruiz, D.V., Bragança, C.S.C.D., Poncetti, B.L., Bittencourt, T.N., Futai, M.M., Vibration-based structural damage detection strategy using FRFs and machine learning classifiers, *Structures*, 59, 2024, 105753.
- [21] Le-Xuan, T., Bui-Tien, T., Tran-Ngoc, H., A novel approach model design for signal data using 1DCNN combing with LSTM and ResNet for damaged detection problem, *Structures*, 59, 2024, 105784.
- [22] Zhou, W., Xu, Y.F., Damage identification for plate structures using physics-informed neural networks, *Mechanical Systems and Signal Processing*, 209, 2024, 111111.
- [23] Chen, G., Yan, Z., Teng, S., Cui, F., Bassir, D., A Bridge Vibration Measurement Method by UAVs based on CNNs and Bayesian Optimization, *Journal of Applied and Computational Mechanics*, 9(3), 2023, 749–762.
- [24] Zhang, G., Lianliu, Zejiazhou, Lichengliu, Yipingjiang, Zhenyu, Machine-learning-based damage identification methods with features derived from moving principal component analysis, *Mechanics of Advanced Materials and Structures*, 27(10), 2020, 1789–1802.
- [25] Li, H., Spencer, B.F., Mao, J., Wang, H., Spencer, B.F., Toward data anomaly detection for automated structural health monitoring: Exploiting



- generative adversarial nets and autoencoders, *Structural Health Monitoring*, 20(4), 2021, 1609-1626.
- [26] Lei, X., Xia, Y., Wang, A., Jian, X., Zhong, H., Sun, L., Mutual information based anomaly detection of monitoring data with attention mechanism and residual learning, *Mechanical Systems and Signal Processing*, 182, 2023, 109607.
- [27] Tang, Z., Chen, Z., Bao, Y., Li, H., Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring, *Structural Control and Health Monitoring*, 26(1), 2019, e2296.
- [28] Silva, M., Santos, A., Santos, R., Figueiredo, E., Sales, C., Costa, J.C., Deep principal component analysis: An enhanced approach for structural damage identification, *Structural Health Monitoring*, 18(5-6), 2019, 1444-1463.
- [29] Nguyen, V.T., Tran, T.H., Ha, N.A., Ngo, V.L., Pham, B.T., GIS Based Novel Hybrid Computational Intelligence Models for Mapping Landslide Susceptibility: A Case Study at Da Lat City, Vietnam, *Sustainability*, 11(24), 2019, 7118.
- [30] Hong, H., Liu, J., Zhu, A.X., Modeling landslide susceptibility using LogitBoost alternating decision trees and forest by penalizing attributes with the bagging ensemble, *Science of The Total Environment*, 718, 2020, 137231.
- [31] Song, J., Wang, Y., Fang, Z., Peng, L., Hong, H., Potential of Ensemble Learning to Improve Tree-Based Classifiers for Landslide Susceptibility Mapping, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 2020, 4642-4662.
- [32] Hoang, N.-D., Tran, V.-D., Tran, X.-L., Image processing-based classification of pavement fatigue severity using extremely randomized trees, deep neural network, and convolutional neural network, *International Journal of Pavement Engineering*, 24(1), 2023, 2201902.
- [33] Leon-Medina, J.X., Anaya, M., Parés, N., Tibaduiza, D.A., Pozo, F., Structural Damage Classification in a Jacket-Type Wind-Turbine Foundation Using Principal Component Analysis and Extreme Gradient Boosting, *Sensors*, 21(8), 2021, 2748.
- [34] Qi, W., Xu, C., Xu, X., AutoGluon: A revolutionary framework for landslide hazard analysis, *Natural Hazards Research*, 1(3), 2021, 103-108.
- [35] Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Smola, A. AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data, *7th ICML Workshop on Automated Machine Learning*, 2020.
- [36] Song, Y.-Y., Ying, L., Decision tree methods: applications for classification and prediction, *Shanghai Archives of Psychiatry*, 27(2), 2015, 130.
- [37] Zhao, Y. Chapter 4 - Decision Trees and Random Forest. *R and Data Mining*: Academic Press; 2013. p. 27-40.
- [38] Al-Ameri, S., Alawady, A.A., Yousof, M.F.M., Ahmad, H., Salem, A.A., Talib, M.A., Frequency response analysis for transformer tap changer damage detection, *International Journal of Power Electronics and Drive Systems*, 11(1), 2020, 350-380.

ORCID iD

Ge Zhang  <https://orcid.org/0009-0006-5044-5413>
 Licheng Zhou  <https://orcid.org/0000-0002-8145-5124>
 Gongfa Chen  <https://orcid.org/0000-0001-7232-9583>
 Zejia Liu  <https://orcid.org/0000-0001-9420-4896>
 Bao Yang  <https://orcid.org/0000-0003-3638-0212>
 Zhenyu Jiang  <https://orcid.org/0000-0001-8497-3405>
 Yiping Liu  <https://orcid.org/0000-0003-2900-3435>
 Liqun Tang  <https://orcid.org/0000-0001-6037-1733>



© 2024 Shahid Chamran University of Ahvaz, Ahvaz, Iran. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0 license) (<http://creativecommons.org/licenses/by-nc/4.0/>).

How to cite this article: Zhang G., et al. An Effective Damage Identification Method Combining Double-Window Principal Component Analysis with AutoGluon, *J. Appl. Comput. Mech.*, 10(3), 2024, 531-546.
<https://doi.org/10.22055/jacm.2024.45565.4385>

Publisher's Note Shahid Chamran University of Ahvaz remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

