Structures 88 (2026) 111949

Contents lists available at ScienceDirect

Structures

journal homepage: www.elsevier.com/locate/istruc

Temperature effects on strain retrieval using machine learning: A field study

Zhenkun Li [a,*], Rolands Kromanis [b]

[a] Department of Architecture, Built Environment and Construction Engineering, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milan, Italy
[b] Department of Civil Engineering and Management, Faculty of Engineering Technology, University of Twente, Drienerlolaan 5, 7522 NB Enschede, the Netherlands

ARTICLE INFO

Keywords:
Structural health monitoring
Temperature
Strain
Machine learning
Data retrieval
Footbridge

ABSTRACT

Temperature variations are known to dominate the quasi-static response of bridge structures and complicate structural health monitoring (SHM) tasks. This study investigates the feasibility of retrieving bridge strain responses using temperature measurements through machine learning (ML) techniques. A field dataset collected from a 27.5 m footbridge, including ten strain gauges and eleven temperature sensors, is evaluated. A novel temperature–strain separation method, combining Empirical Mode Decomposition and Moving Average Filtering (EMD–MAF), is proposed to isolate temperature-induced strain components and mitigate measurement noise and operational effects. Five scenarios are designed to assess the influence of thermal information on strain retrieval, considering single/multiple temperature sensors, historical temperature features, strain information from neighboring strain gauges, and their combinations. Four ML regression models: ridge regression (RR), light gradient boosting machine, XGBoost, and artificial neural networks, are trained and optimized using Bayesian hyperparameter tuning. Results show that the proposed EMD–MAF method successfully extracts thermal strain trends and reveals strong temperature–strain correlations. For strain retrieval, RR consistently provides the most reliable performance when using temperature data alone, while nonlinear models overfit and generalize poorly. When strain measurements from neighboring strain gauges are available, RR again yields the best generalization performance, as other strain gauges implicitly encode temperature and operational effects. Incorporating multi-sensor temperature data enhances model robustness, but additional historical temperature features do not significantly improve predictions in field conditions. The findings highlight the critical role of temperature in strain interpretation and demonstrate the practical limitations and capabilities of ML-based strain retrieval for real-world SHM systems.

1. Introduction

Bridge structures are a key component of modern transportation systems worldwide [1–3]. Many bridges were built in the last century and have now been in service for more than 50 years [4,5]. Over time, aging and deterioration can develop, posing risks to public safety. The 2018 collapse of the Morandi Bridge in Italy, where corrosion and damage were later identified, is a clear example of such risks [6]. These incidents have increased public concern regarding the safety of existing bridges.

Since the beginning of this century, researchers have developed reliable methods for assessing the health of bridges. Traditionally, this has relied on periodic visual inspections carried out by experienced engineers. However, as bridges have become longer and taller, visual inspection has grown increasingly difficult, unsafe, and in some cases impractical [7–9]. Structural health monitoring (SHM) offers an alternative by providing continuous information on structural performance, damage detection, safety assessment, maintenance planning, and remaining service life, which has led to growing interest in SHM worldwide [10–12]. Damage detection, a central component of SHM, delivers essential insights into structural condition and guides maintenance decisions [13,14]. To support this task, various sensors are installed on bridges to record structural responses and environmental conditions, enabling continuous and data-driven evaluations [15–17]. Commonly used sensors include accelerometers, strain gauges, displacement sensors, inclinometers, and anemometers [18–20].

Nevertheless, damage detection in practical engineering is challenging because many external factors can mask the effects of actual structural damage [21–23]. For bridges, these factors include temperature changes, wind, and traffic loads. Among them, temperature plays a major role and varies on both daily and seasonal scales [24–26]. Ni

* Corresponding author.
Email addresses: zhenkun.li@polimi.it; zhenkun-li@outlook.com (Z. Li), r.kromanis@utwente.nl (R. Kromanis).

https://doi.org/10.1016/j.istruc.2026.111949
Received 16 December 2025; Received in revised form 6 March 2026; Accepted 22 April 2026

et al. [27] examined the influence of temperature on the natural frequencies of the cable-stayed Ting Kau Bridge in Hong Kong and used a support vector machine to model how frequencies change with temperature. Cross et al. [28] studied the Tamar Bridge in the U.K. and also found that its modal properties were strongly affected by temperature. In some cases, temperature-induced responses can exceed those caused by traffic loads [29]. On the same bridge studied in [28], Koo et al. [30] reported that temperature causes noticeable thermal expansion in the deck, main cables, and stays, which dominates global deformation, while the effects of traffic and wind are secondary.

In addition, sensor faults present a major challenge in engineering applications. Harsh weather can cause sensors to record faulty data or fail completely, and communication errors may also occur in long-term monitoring. When these issues arise, it becomes necessary to recover the missing responses to support engineering decisions [31]. Previous studies have explored the recovery of various structural responses, including accelerations, strains, and displacements. For example, Oh et al. [32] recovered missing strain data from an overpass bridge using a convolutional neural network. They treated one sensor as faulty and used the responses of other sensors as inputs to predict the missing strain. Their approach was validated through numerical tests on a steel beam and experiments on a frame structure. Bao et al. [33] proposed a wireless acceleration recovery method based on data-driven modeling and compressed sensing. The original signal was first converted into a synthetic measurement using a random matrix, and part of this data was assumed lost during wireless transmission. The original response was then reconstructed using an l_1 optimization process. This method was verified on SHM systems installed on the Jinzhou West Bridge and the National Aquatics Center in China. For strain data in particular, temperature variations have a strong influence, making strain closely correlated with temperature. This correlation creates an opportunity to retrieve strain responses directly from temperature measurements.

The importance of temperature measurements for predicting structural responses has been well recognized [34], where auto-regression models were used to forecast future thermal behavior. Building on this idea, Kromanis and Kripakaran [35] developed a method to predict traffic-induced responses using information about vehicle location and weight. In 2024, Glashier et al. [36] introduced the iterative regression-based thermal response prediction (IRBTRP) method, which can remove temperature effects prior to anomaly detection. In the same year, Mariani et al. [37] explored multiple linear regression and deep learning models to predict thermal responses, and validated their approach using a laboratory truss setup. However, their study relied on measurements from only one temperature sensor. In real applications, bridges experience both daily and seasonal temperature changes. These changes are driven mainly by solar radiation, which is unevenly distributed across the bridge surface. Existing studies on temperature-related structural responses can be broadly categorized into three groups: (1) temperature compensation for modal frequency or damage detection, (2) regression-based prediction of thermal response, and (3) laboratory-scale modeling of temperature–strain relationships. However, limited attention has been given to the inverse problem of retrieving missing strain data directly from distributed temperature measurements under long-term field monitoring conditions. Temperature transfer within structural components is generally affected by material properties. As a result, the temperature within a bridge is always non-uniform [38,39]. However, previous studies rarely examine model generalization behavior when multiple temperature sensors and strain-assisted features are jointly considered.

To address this gap, this study formulates strain retrieval as a temperature-driven inverse problem under long-term field monitoring conditions. The main contributions are threefold: (1) A physically motivated hybrid EMD–MAF method is proposed to isolate temperature-induced strain components while preserving quasi-static trends and suppressing operational noise. (2) Five feature scenarios are systematically designed to quantify the relative information content of distributed temperature measurements and neighboring strain gauges for strain retrieval. (3) Using one-year field data, we demonstrate that the temperature–strain relationship is predominantly linear and that simpler linear models provide superior generalization compared with nonlinear approaches under real environmental variability.

The remainder of this paper is organized as follows. Section 2 introduces the ML regression algorithms used to retrieve strain data from temperature measurements. Section 3 describes the field tests conducted on the bridge and outlines the preliminary data processing. Section 4 presents the different scenarios, along with the corresponding results and discussions. Finally, Section 5 concludes the paper.

2. Methodology

In this work, different machine learning (ML) algorithms are employed and compared in strain retrieval using temperature data. In this section, the utilized algorithms, including ridge regression (RR), light gradient boosting machine (LGBM), eXtreme gradient boosting (XGBoost), and artificial neural network (ANN), will be briefly introduced. It is assumed that there are m samples and n features in the dataset, namely $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_m]$, in which $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T$.

2.1. Ridge regression (RR)

Multiple Linear Regression (MLR) is one of the most commonly used machine learning algorithms. It has a solid theoretical foundation and can clearly show the contribution of each feature. Eq. (1) presents the general form of MLR [40],

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_n x_{in} + \epsilon_i \tag{1}$$

where y_i is the observed dependent variable; x_{ij} is the value of the j-th feature for sample i; β_j is the j-th regression weight, and ϵ_i is the random error term. If the predicted dependent variable is denoted as $\hat{y}_i$, then $y_i = \hat{y}_i + \epsilon_i$. However, traditional MLR may suffer from multicollinearity. To address this issue, Ridge Regression (RR, also known as Tikhonov regularization) was developed to reduce errors caused by overfitting on training data [41]. It applies l_2 regularization to the weights, especially those with large values, in the MLR model. RR aims to minimize the objective function shown in Eq. (2),

$$\mathcal{L}_{rr}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right)^2 + \lambda \sum_{j=1}^{n} \beta_j^2 \tag{2}$$

where $\boldsymbol{x}_i$ is the i-th feature vector, and $\boldsymbol{\beta}$ is the weight vector. It can be seen that instead of minimizing $\sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right)^2$ in Ordinary Least Squares (OLS), Eq. (2) adds a regularization term $\lambda \sum_{j=1}^{n} \beta_j^2$ to limit the growth of the weights, in which λ is a hyperparameter to adjust the degree of regularization. This also reduces the sensitivity of RR to noise and multicollinearity.

2.2. Light gradient boosting machine (LGBM)

LGBM is a fast, high-performance implementation of gradient boosting decision trees developed by Microsoft [42]. It is designed for efficiency, scalability, and fast training, especially on large datasets or high-dimensional tabular data. Like other boosting methods, LGBM builds an ensemble of decision trees sequentially, where each new tree aims to correct the errors made by the previous ones. The model prediction can be expressed by Eq. (3),

$$\hat{y}_i = \sum_{k=1}^{K_t} f_k\left(\boldsymbol{x}_i\right), \quad f_k \in \mathcal{F} \tag{3}$$

where f_k is the k-th regression tree; $\mathcal{F}$ is the space of all possible trees; K_t is the total number of trees. LGBM optimizes a regularized objective

shown in Eq. (4),

$$\mathcal{L}_{lg} = \sum_{i=1}^{m} l\left(y_i, \hat{y}_i^{(t-1)} + f_t\left(\boldsymbol{x}_i\right)\right) + \Omega\left(f_t\right) \tag{4}$$

where l is a differentiable loss (e.g., squared error, logistic loss) and $\Omega(f_t)$ controls tree complexity (number of leaves, leaf weights, etc.). LGBM has been widely used in time-series forecasting.

2.3. XGBoost

Extreme Gradient Boosting (XGBoost) is an advanced and efficient implementation of the gradient boosting algorithm [43], widely used for both regression and classification tasks. It is known for its speed, scalability, and strong predictive accuracy, and has become a standard tool in machine learning research. XGBoost builds an ensemble of decision trees sequentially, where each new tree is trained to correct the prediction errors (residuals) made by the previous trees. The final prediction is the weighted sum of all trees, consistent with Eq. (3). It is designed to minimize the regularized objective shown in Eq. (5),

$$\mathcal{L}_{xg} = \sum_{i=1}^{m} l\left(y_i, \hat{y}_i\right) + \sum_{k=1}^{K_t} \Omega\left(f_k\right) \tag{5}$$

where $l(y_i, \hat{y}_i)$ is the loss function; $\Omega\left(f_k\right) = \gamma T + \frac{1}{2}\lambda\|\boldsymbol{w}\|^2$ is the regularization term controlling three complexity with T learning number of leaves and w leaf weights. This helps prevent overfitting and improves generalization. At each boosting iteration, XGBoost adds a new tree that minimizes the second-order Taylor expansion of the loss function, as shown in Eq. (6),

$$\mathcal{L}_{xg}^{(t)} \approx \sum_{i}[g_i f_t\left(\boldsymbol{x}_i\right) + \frac{1}{2}h_i f_t\left(\boldsymbol{x}_i\right)^2] + \Omega\left(f_t\right) \tag{6}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right)$ is the first derivative of the loss, and $h_i = \partial^2_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right)$ represents the second derivative of the loss. By using both first and second derivatives, XGBoost can achieve faster and more accurate optimization than traditional gradient boosting.

2.4. Artificial neural network (ANN)

Artificial Neural Network (ANN) is the basic framework underlying many advanced neural network architectures [44], such as convolutional neural networks (CNN), recurrent neural networks (RNN), and Transformers. It is composed of layers of interconnected neurons that learn to map input features to outputs through weighted connections. An ANN usually includes three main layers: an input layer, one or more hidden layers, and an output layer. A typical ANN with two hidden layers is illustrated in Fig. 1.

In the ANN, the output in the next layer (e.g., hidden layer 1) can be computed using Eq. (7),

$$a^{(1)} = \sigma(\mathbf{W}^{(1)}\boldsymbol{x}_i + \boldsymbol{b}^{(1)}) \tag{7}$$

where $\mathbf{W}^{(1)}$ is the weight matrix in the first hidden layer, and $\boldsymbol{b}^{(1)}$ is the bias vector. $\sigma(\cdot)$ denotes the activation function (such as ReLU, Sigmoid, or Tanh). ANNs learn by adjusting weights and biases to minimize a loss function, for example the mean squared error used in regression, as shown in Eq. (8),

$$\mathcal{L}_{nn} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2 \tag{8}$$

This optimization is carried out through backpropagation, which computes the gradients of the loss with respect to each parameter using the chain rule, and gradient descent, which updates the parameters. The learning rate controls the step size of each update. Adaptive moment estimation (Adam) is one of the most widely used optimization algorithms for training ANNs and is adopted in this work [45].

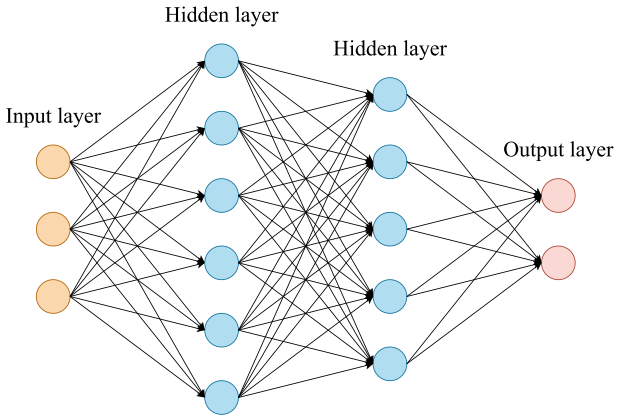


Fig. 1. Illustration of an ANN.

3. Field-test setup and the dataset

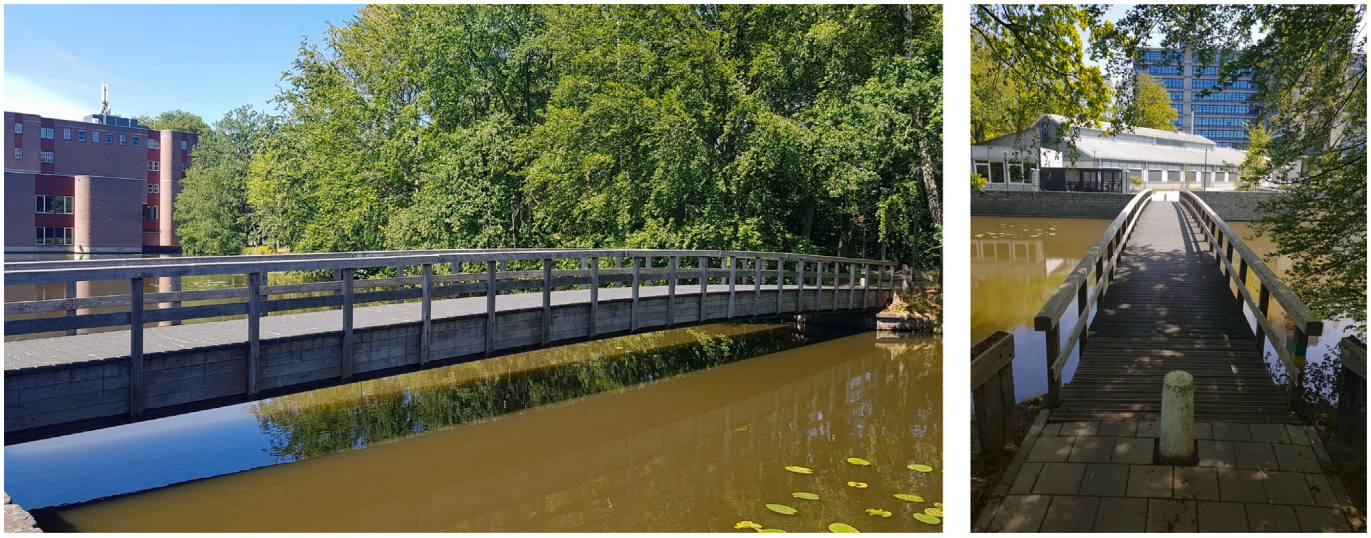
This section introduces the dataset collected from field tests on a real footbridge. The data is then initially analyzed and prepared for model training in the following section.

3.1. Field tests

The field tests were carried out on a footbridge located on the campus of the University of Twente (UT) in Enschede, the Netherlands, as shown in Fig. 2. The side and front views of the UT campus footbridge are provided in Figs. 2(a) and Fig. 2(b), respectively. The drawing of the footbridge, including its plan and elevation views, is presented in Fig. 3. The bridge is simply supported at both ends and has a total length of 27.5 m. The horizontal bracing uses 80×80×5 SHS steel, and the three main girders are IPE600 steel beams. In February 2023, eleven temperature sensors (ten on the surface and one in the air) and ten strain gauges were installed on the bridge to monitor its condition. The sensor locations are shown in Fig. 3. The strain channels were measured using Kyowa weldable full-bridge strain gauges (KCW-5-350-G14S-11) with a chloroprene-coated 4-conductor cable (3 m). The temperature field was recorded using Ahlborn magnetic temperature sensors (X-9/K, Type-K thermocouple) mounted on the steel surface, and the air temperature channel was measured using Ahlborn FHAD46-C41AL10. The Ahlborn ALMEMO data acquisition (DAQ) system was employed.

By the time this study began, the bridge had been monitored for nearly one year (from February 7th, 2023, to January 20th, 2024). Strain responses were recorded by strain gauges installed on the bridge and synchronized with the temperature sensors at a sampling frequency of approximately 0.1 Hz. The collected data from all sensors are shown in Fig. 4.

The ten strain gauges, named S1, S2, …, S10, recorded strain at different positions on the footbridge. Throughout the monitoring period, various types of sensor faults occurred, including “outlier,” “drift,” “missing,” and “square” faults [46]. The fault patterns observed in the long-term strain data are consistent with common sensor and acquisition issues in field monitoring. Drift typically arises from gradual changes in sensor offset/sensitivity caused by temperature cycling, adhesive aging, moisture effects, or slow variations in wiring/contact resistance. Missing data is mainly due to acquisition or communication interruptions (e.g., power off, logger resets, or intermittent connections). Square faults usually appear as step-like or plateau segments and are often related to clipping/saturation or intermittent electrical contact/channel freezing. These faults became more frequent after winter, likely due to harsher conditions such as rapid temperature drops and condensation/icing. Some gauges, such as S2 and S6, developed faults early, while others experienced issues later. Overall, every strain gauge showed some



(a) Side (b) Front

Fig. 2. Photos of the UT campus footbridge.

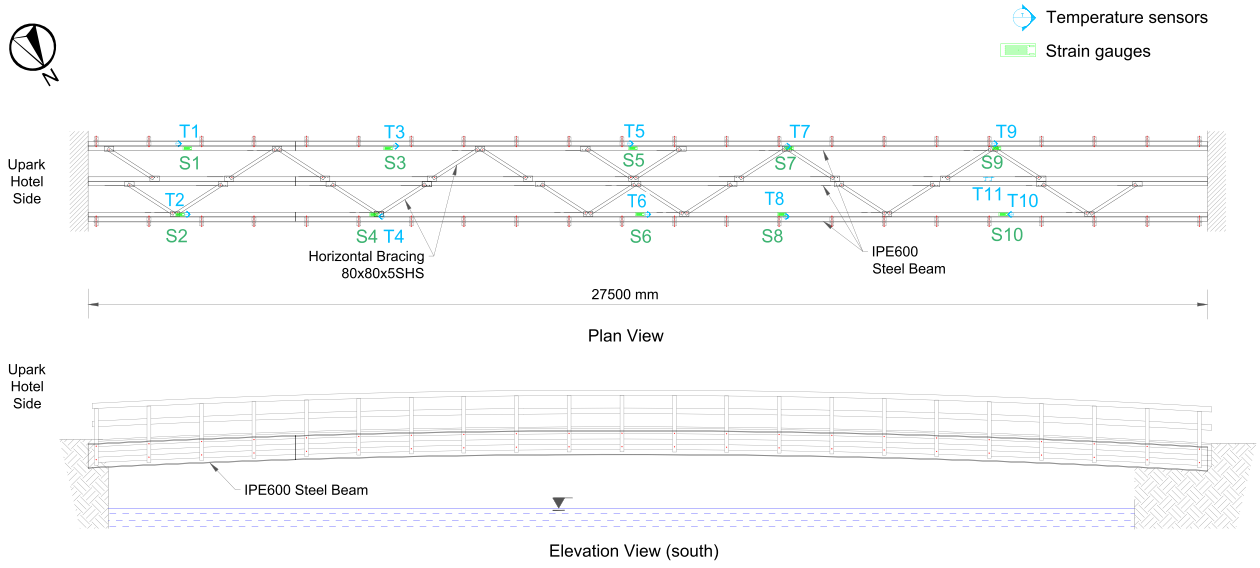


Fig. 3. Drawings of the footbridge.

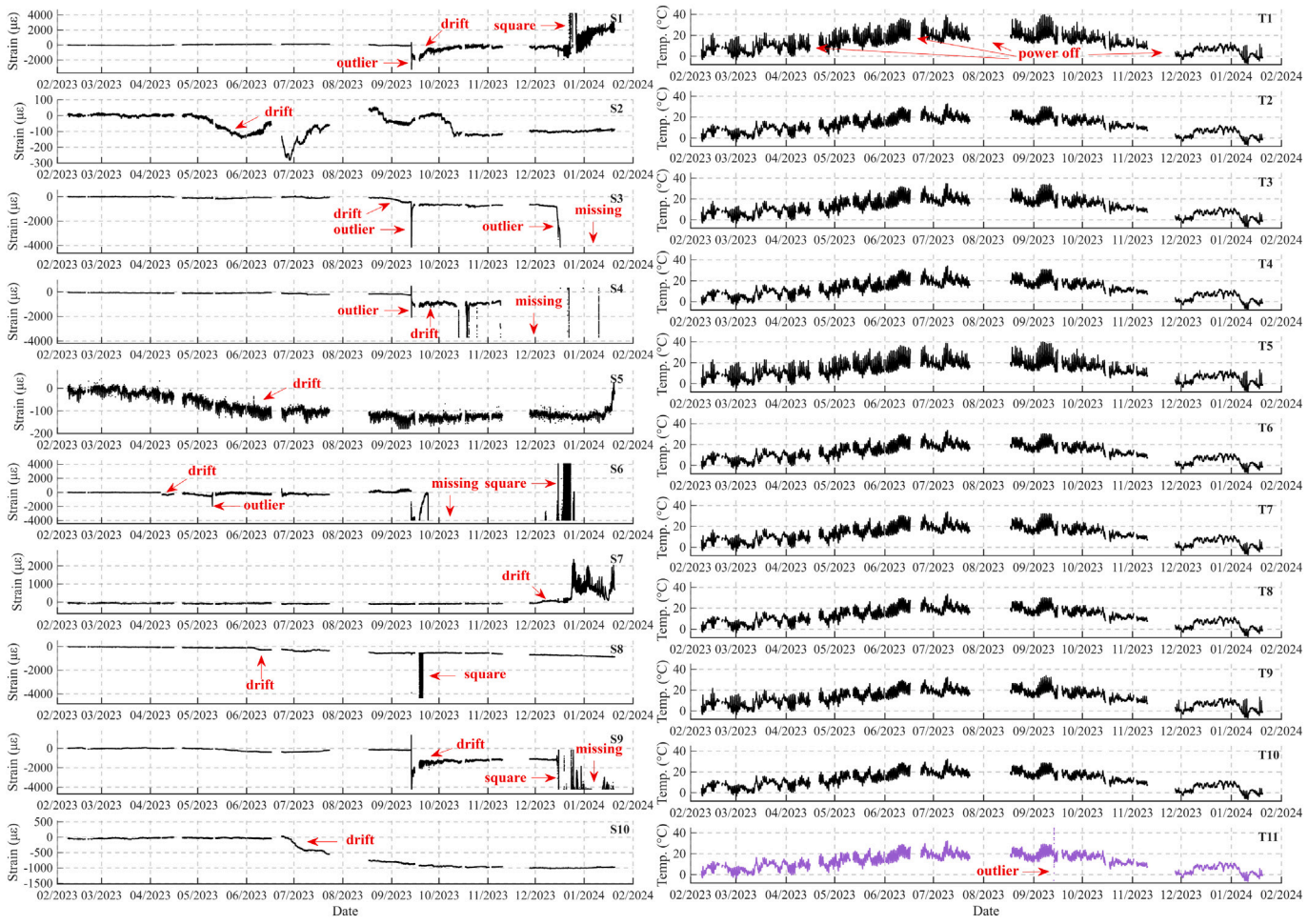
level of faulty behavior during the year-long monitoring period. Notably, when winter began around October, significant sensor faults appeared more frequently, likely due to the sharp drop in temperature, as observed in Fig. 4(b). Certain strain gauges (e.g., S4 and S6) eventually stopped collecting data altogether. These issues highlight the challenges in practical bridge monitoring. When strain data are needed for assessing structural conditions, it becomes necessary to retrieve missing or faulty strain measurements using available information.

The recorded temperature data are shown in Fig. 4(b). Eleven temperature sensors, named T1, T2, …, T11, were installed on the bridge. Sensors T1–T10 were mounted directly on the bridge to measure actual material temperatures, while T11 measured air temperature (it performs in a way similar to a weather station). As seen in Fig. 3, the temperature sensors were positioned close to the strain gauges, and the air temperature sensor was suspended between T9 and T10. From Fig. 4(b), it can be observed that the temperature sensors experienced fewer faults, with the exception of an “outlier” period for T11 between September and October 2023. Overall, the temperature sensors performed reliably, making temperature data a strong reference for strain retrieval. Although occasional system suspensions occurred, such as during a power outage in August 2023, these interruptions did not cause sensor faults.

3.2. Data initial analysis

After introducing the collected strain and temperature data, this section presents the data analysis and separation. The variations in strain data come from three main sources: temperature changes, ongoing traffic (including bicycles, pedestrians, etc.), and other influences such as wind, noise, and humidity. Among these, the first two factors have the greatest impact on the footbridge’s strain response. Therefore, to study temperature effects, the strain caused by temperature must be separated from other components. To examine the data more closely, the temperature and strain records from the first ten days (February 7th 17:32:02 to February 17th 17:32:02, 2023) after monitoring began are shown in Fig. 5.

From Fig. 5(a), although the overall patterns of temperature variation are similar, the peak values differ among sensors. This indicates



(a) Strain (b) Temperature

Fig. 4. Strain and temperature data over one year.

that the surface temperature of the footbridge is not uniform at any given time. To examine this phenomenon in more detail, Fig. 6 presents the temperature data from T1 to T11 in different seasons. Clear differences can be observed between the measurements from the various temperature sensors, especially in winter. The temperature typically peaks around 12:00 to 15:00. The patterns of variation are similar across days in winter, spring, and summer, although the absolute temperature values differ. As a result, the strain in the bridge components should be influenced by temperature measurements from multiple locations rather than from a single sensor. At the same time, temperature data from T1–T10 are clearly related, even though their ranges differ. This means that when temperature data are used as input in machine learning models, multicollinearity may affect training and should be considered.

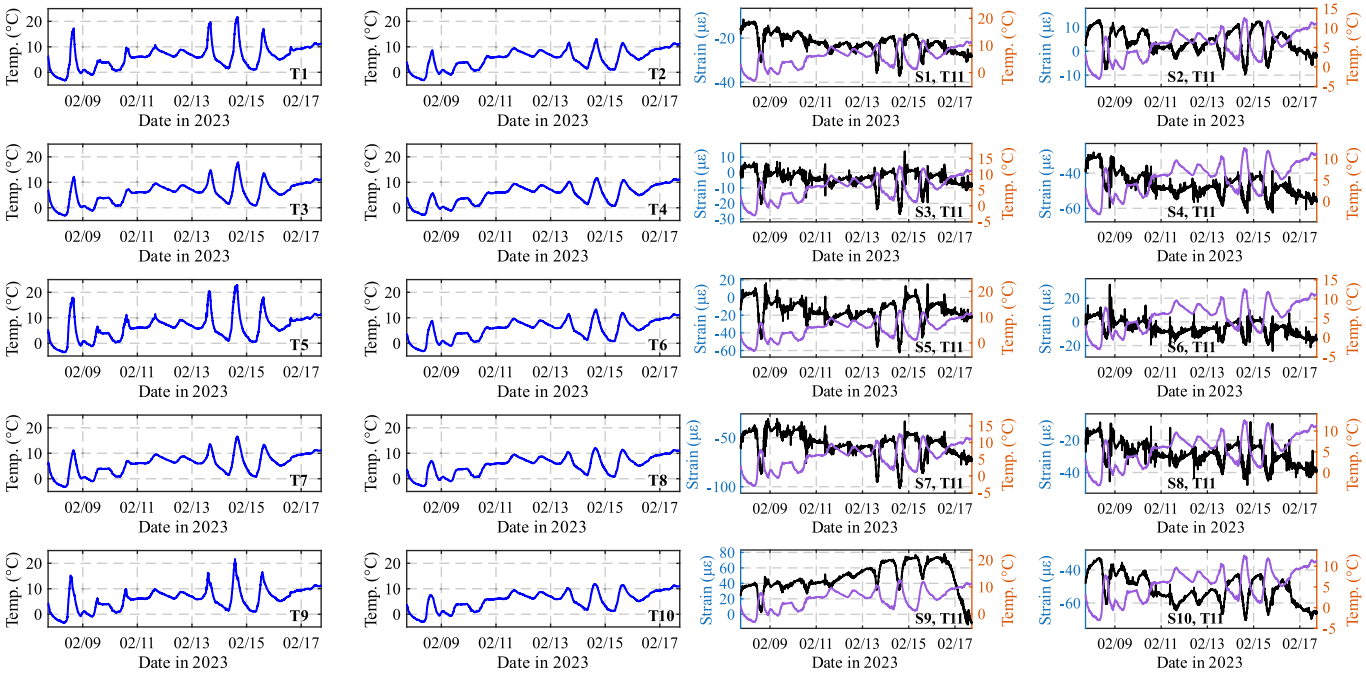
Fig. 5(b) shows the air temperature from T11 together with strain data from the same ten-day period. A clear negative correlation can be observed: as temperature increases, strain decreases. The effects of ongoing traffic are also visible as fluctuations in the strain signal, but temperature remains the dominant factor shaping the strain response. To further study the temperature–strain relationship, correlation analysis, including both self-correlation and cross-correlation, was performed. The results are shown in Fig. 7.

From Fig. 7(a), it can be seen that the temperature data collected from T1–T11 are highly correlated, with most correlation coefficients above 0.8. Although the sensors show strong similarity, the differences among them indicate that each temperature sensor still provides useful information and can contribute to improving strain retrieval. Fig. 7(b) then shows the correlations among the strain measurements from S1–S10. These correlations are generally lower than those of the temperature sensors, especially for S9. This makes the data from S9 less reliable. Overall, the analysis confirms that both temperature and strain data exhibit meaningful correlations, which can help identify and exclude unreliable measurements.

Since this work focuses on the influence of temperature on strain retrieval, the correlation coefficients between all temperature and strain channels are shown as a heatmap in Fig. 7(c). All coefficients are negative, confirming the expected inverse relationship between temperature and strain. However, different temperature sensors show different levels of influence. For instance, T1, T5, and T9 appear to affect all strain gauges. Likewise, S1, S2, S4, S6, S8, and S10 are strongly influenced by most temperature sensors, while S9 shows much weaker correlations. This again indicates that S9 provides unreliable information and should be excluded from model training to avoid degrading predictive performance.

3.3. Data separation

From the above analysis, it is clear that all temperature data are closely related to the strain measurements. The next step is to separate the strain into two components: strain caused by temperature and strain caused by other factors. To achieve this, a method combining Empirical



(a) Temperature data of T1-T10 (b) Temperature data of T11 and strain data

Fig. 5. Temperature and strain data during the first 10 days.

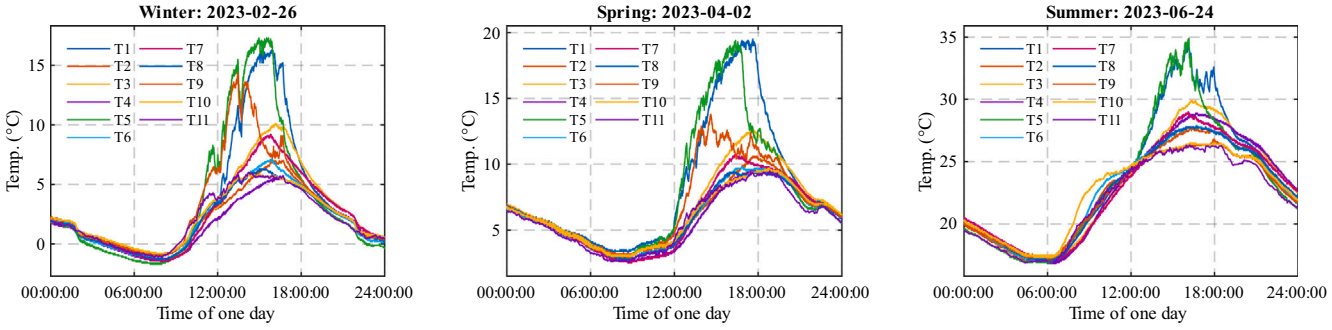
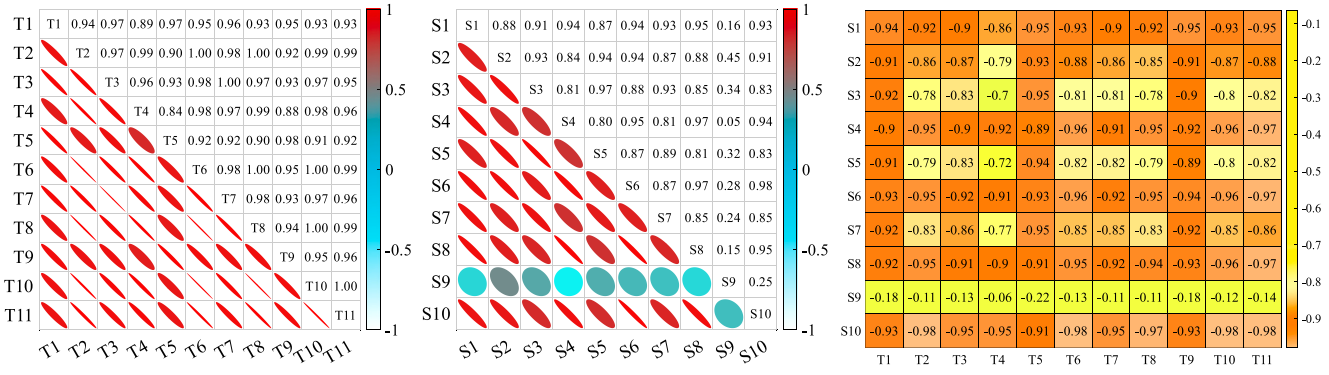


Fig. 6. Temperature variations in one day during winter, spring, and summer.



(a) Temperature (b) Strain (c) Temperature-strain

Fig. 7. Correlation analysis.

Mode Decomposition (EMD) and a Moving Average Filter (MAF) is proposed. From a structural mechanics perspective, temperature-induced strain is characterized by slow-varying, quasi-static behavior driven by thermal expansion and contraction of structural components. In contrast, traffic loads and environmental disturbances introduce high-frequency fluctuations superimposed on the thermal trend. Therefore,

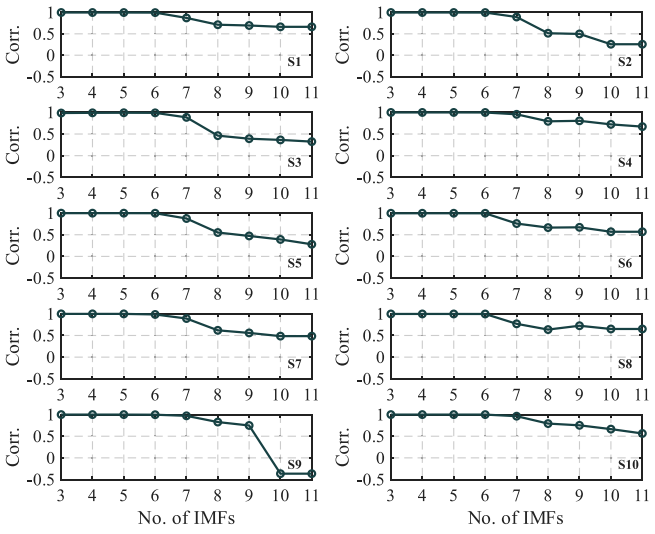


Fig. 8. Correlation analysis after applying EMD.

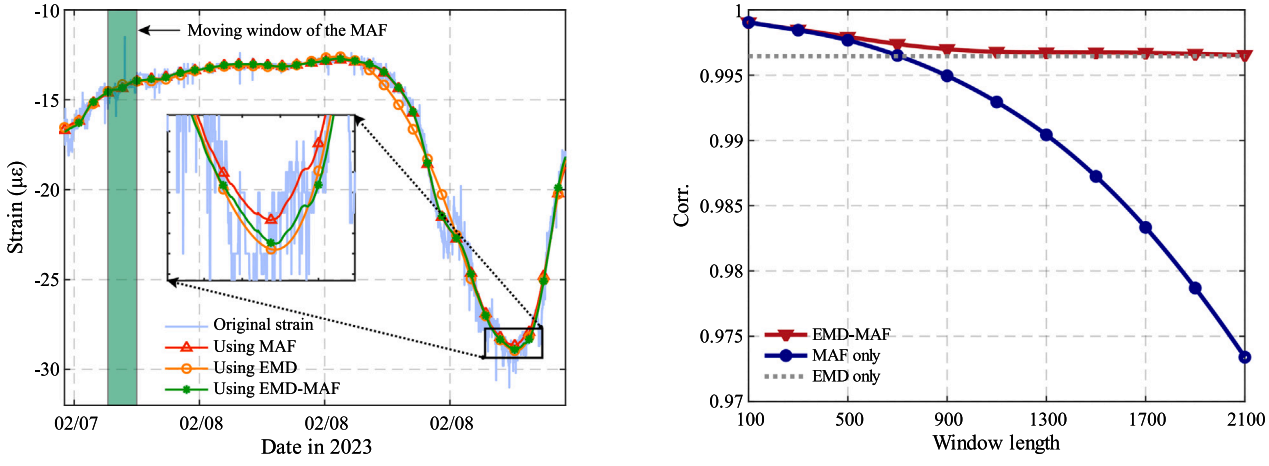
separating low-frequency thermal components from high-frequency operational effects can be interpreted as a time-scale decomposition problem. EMD is particularly suitable for this task because it adaptively decomposes nonstationary signals without requiring predefined basis functions. However, EMD alone may produce residual components sensitive to local extrema and noise. The subsequent application of a moving average filter stabilizes the extracted low-frequency component and mitigates sensitivity to outliers, resulting in a robust estimate of temperature-induced strain.

This approach integrates the strengths of both techniques. The first step is to select an appropriate number of Intrinsic Mode Functions (IMFs). Fig. 8 shows the relationship between the correlation coefficients and the number of IMFs, where the correlation coefficient reflects the consistency between the residual after EMD and the original strain data. The number of IMFs tested ranges from 3 to 11. As shown in Fig. 8, for all strain gauges S1–S10, the correlation remains high (near 1.0) when 3–6 IMFs are used. To maintain a strong correlation between the residual (temperature-related strain) and the original strain while filtering out as much non-temperature-related strain as possible, 6 IMFs are selected in this work.

Nevertheless, if only EMD is applied, the residual may deviate from the original strain data, leading to larger errors when extracting the temperature-induced strain. Fig. 9(a) shows the extracted strain for S1 on the first day (February 7th, 2023) using different methods. When EMD is solely used, the extracted strain departs noticeably from the original signal, especially when the strain decreases sharply. We also tested using MAF alone for temperature-induced strain extraction. However, MAF is sensitive to outliers and strongly depends on the chosen window length. When the window length equals 500, the extracted strain deviates from the original strain at several peak values. To address these issues, this work combines the two methods. After applying EMD, MAF is used to refine the extracted strain. The proposed EMD–MAF method can take the strengths of both techniques, producing results that lie between those obtained from EMD alone and MAF alone while reducing their individual shortcomings, as shown in Fig. 9(a). The “window length” in the EMD–MAF method refers to the MAF applied to the decomposed thermal component to suppress short-term fluctuations. This windowing is only used for signal smoothing during thermal strain extraction and does not generate moving-window training samples.

To further demonstrate the capability of the EMD–MAF method, Fig. 9(b) shows the correlation coefficients between the extracted temperature-induced strain and the original strain for different window lengths. It can be seen that MAF alone is highly sensitive to the window length, with the correlation decreasing sharply as the window length increases. In contrast, the EMD–MAF method maintains a high correlation and gradually approaches the performance of EMD alone when the window length becomes large. This is advantageous in practical engineering, where selecting an appropriate window length based solely on experience can be difficult. The proposed method ensures that the extracted temperature-induced strain remains highly correlated with the original data. Based on the authors’ experience, a window length of 100–300 is recommended when applying the EMD–MAF method. Using a window length of 150, the remaining response after removing the temperature-induced strain is shown in Fig. 10. It should be noted that the response shown in Fig. 10 represents the residual strain after removing the dominant temperature-induced component. Therefore, its amplitude is expected to be much smaller than that of the original strain signal.

From Fig. 10, a clear periodic pattern can be seen in the data after removing the temperature effects. This occurs because fewer pedestrians and bicycles use the footbridge during night hours. In addition, February 11th and 12th, 2023 were a Saturday and Sunday, during which the strain peaks are smaller compared to weekdays. To further verify the effectiveness of the proposed method for extracting temperature-induced strain and to illustrate the relationship between temperature and strain, Fig. 11 is provided, which shows that the temperature-induced strain



(a) Temperature-induced strain (b) Correlation

Fig. 9. Strain extraction using various methods.

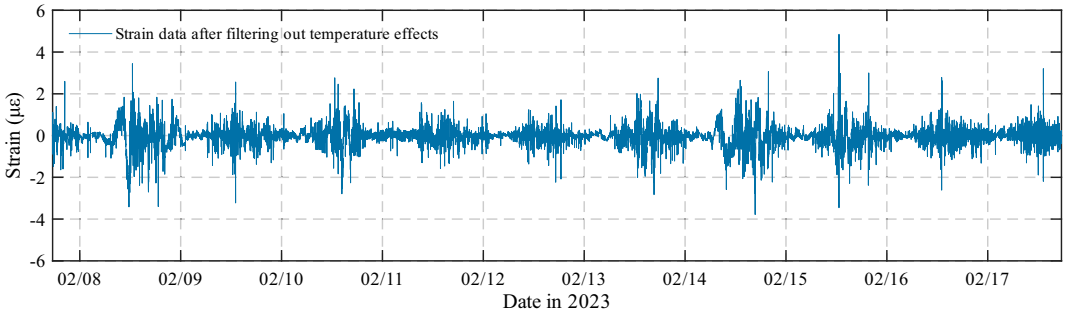


Fig. 10. Residual strain after filtering out the temperature-induced component using the proposed EMD–MAF method.

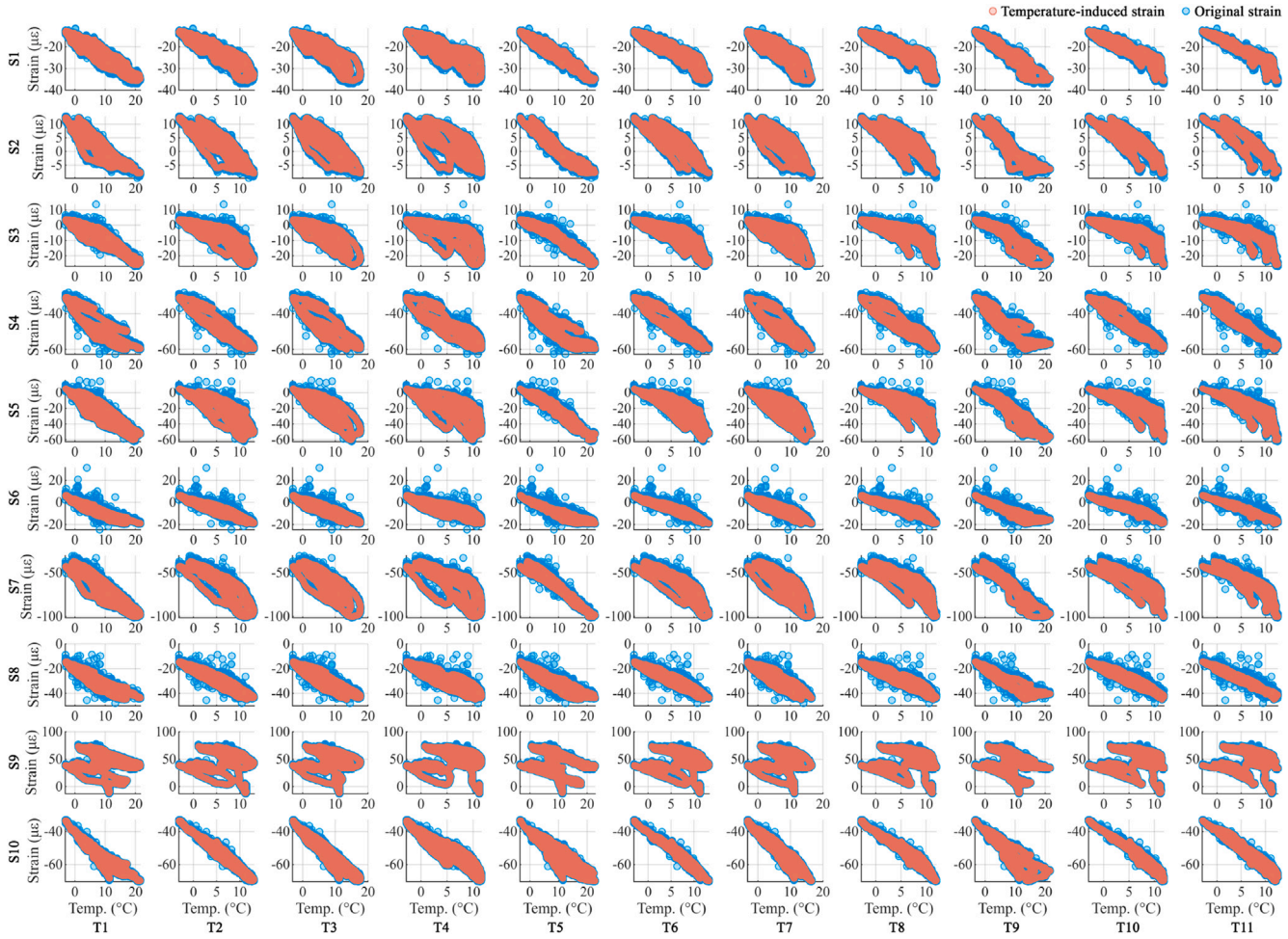


Fig. 11. Relationship between temperature and strain data before and after filtering.

closely covers the original strain, while the remaining fluctuations surround the main temperature-driven trend. This indicates that the temperature-induced strain is well extracted. A clear negative correlation between temperature and strain is also visible, particularly for S6 and S10, which supports the feasibility of predicting strain from temperature data. In contrast, abnormal behavior is again observed for S9.

In addition, the histogram of the strain components caused by ongoing traffic (including bicycles and pedestrians) and other factors (such as wind, noise, and humidity) is shown in Fig. 12 for S1–S10 after removing the temperature-induced strain. The remaining strain data approximately follow normal distributions with a mean close to zero. This aligns with the expectation that these influencing factors typically follow normal patterns in practical engineering. These results further demonstrate the effectiveness of the proposed EMD–MAF method in extracting temperature-induced strain.

In this study, the ML models are developed to retrieve the quasi-static, temperature-induced strain component. Therefore, the extracted low-frequency thermal strain trend obtained by the proposed EMD-MAF procedure is used as the prediction target for model training, validation, and testing, while the remaining high-frequency component (mainly operational effects and measurement noise) is not used as the target in the present work. However, to quantify the effect of the proposed preprocessing, a case without applying EMD-MAF is also considered, where

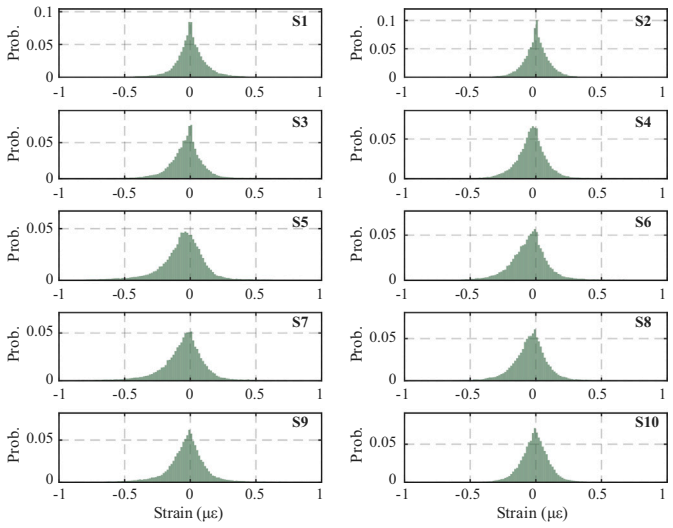


Fig. 12. Histogram of strains excluding temperature effects.

models are trained and tested directly using the originally measured strain signal. This will be discussed in Section 4.4.4.

3.4. Data transformation

For the RR model (linear model), it assigns different weights to the features, as shown in Eqs. (1) and (2). These weights directly indicate the contribution of each feature, so data transformation is not required. However, for the other three ML models (nonlinear models), feature standardization is necessary because the input variables may not share the same scale. Standardization is important since many ML models are sensitive to feature magnitudes. Methods such as ANNs rely on distance-based calculations or gradient-based optimization, where features with large numerical ranges may dominate the model, cause unstable gradient updates, slow convergence, or lead to suboptimal results. Standardization ensures that all features contribute proportionally, improves training stability, and makes optimization efficient. In contrast, tree-based models like LGBM and XGBoost are not scale-dependent because they use threshold-based splits rather than distance or gradient magnitude, so standardization is not strictly necessary, although it can still improve consistency within a processing pipeline. In this work, we use StandardScaler from the Scikit-learn package [47], which is expressed in Eq. (9),

$$x_{i,j}^{\text{std}} = \frac{x_{i,j} - \mu_j}{\sigma_j} \tag{9}$$

where $x_{i,j}$ is the j-th feature of the i-th sample; μ_j is the mean value of the j-th feature computed from the training data and is given by $\mu_j = \frac{1}{N}\sum_{i=1}^{N} x_{i,j}$; and σ_j is the standard deviation of the j-th feature, obtained by $\sigma_j = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_{i,j} - \mu_j\right)^2}$. The standardized samples are then fed into the nonlinear ML models to predict the output.

3.5. Data partitioning

As introduced before, the strain and temperature data were collected at a sampling frequency of approximately 0.1 Hz. After quality control and removal of faulty/missing segments, the original strain and temperature dataset contains 818,645 data points. Although the raw measurements were collected at approximately 0.1 Hz, within relatively short time intervals, the strain and temperature change only slightly. To reduce redundancy and focus on quasi-static behavior, an effective sampling rate of 0.002 Hz (i.e., one sample every 8.3 minutes) was adopted for model development. Under this effective sampling rate, the dataset used for model training and validation (February 7th, 17:32:02 to March 30th, 17:32:02, 2023) contains 16,373 data points, split into 11,461 training data points (70%) and 4912 validation data points (30%). The independent testing dataset (March 30th, 17:32:02 to April 7th, 7:30:58, 2023) contains 2615 data points. To ensure reproducibility, a fixed random seed is assigned to each model so that all models use the same deterministic train-validation-test split.

4. Results and discussions

Based on the dataset described in the previous section, this section explains how the models are trained and outlines the performance evaluation metrics used. Five scenarios with different input feature configurations are then introduced to examine the influence of temperature on strain retrieval, followed by the corresponding prediction results. Finally, several key observations and discussions are provided.

4.1. Scenarios

To investigate the effects of temperature data in strain retrieval, five ML model training scenarios are included in this study. The strain data collected from S4 is utilized as the data to be retrieved. With these scenarios, we aim to evaluate the effects of temperature data in retrieving strain data of one channel.

- **Scenario 1.** The temperature data of one sensor is employed to predict the strain at the same time. Under this condition, there is only one feature to predict the strain. In this scenario, the air temperatures, namely data from T11, are utilized as input.
- **Scenario 2.** Data from multiple temperature sensors are utilized as input to predict the strains. In other words, data from T1, T2, …, T11 are employed. Therefore, there are 11 features in this scenario. It is designed to consider the effects of temperature at different positions of the footbridge.
- **Scenario 3.** This scenario considers historical temperature information to account for possible delay effects between temperature variation and strain response. As illustrated in Fig. 13, a fixed-length look-back window is constructed for each temperature channel by concatenating the current measurement with the previous 59 measurements. Using all 11 sensors, this yields 660 temperature-history features per sample, while the output remains the strain at

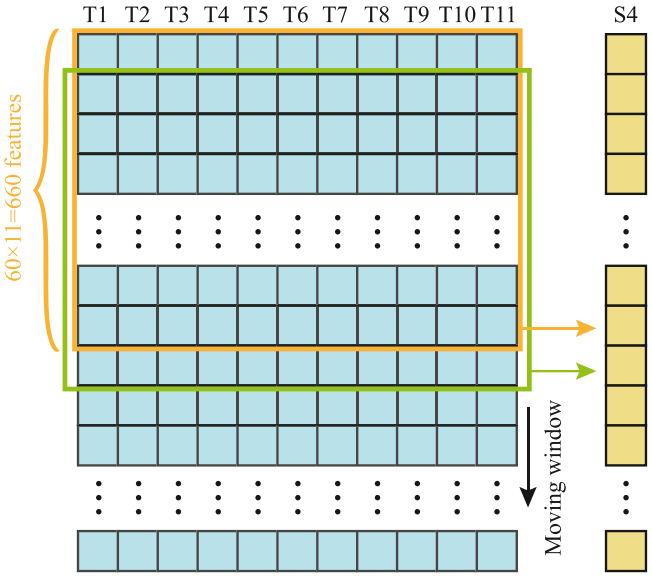


Fig. 13. Illustration of input features in Scenario 3.

the same time step (e.g., S4). Each timestamp therefore corresponds to one training sample with lagged temperature features.

- **Scenario 4.** Apart from the temperature data, another strategy is to retrieve S4 strain using the strain data of other channels. This is because the data from other channels can capture not only the variation of temperature but also other influencing factors, and therefore can be good references.
- **Scenario 5.** In this scenario, we consider all temperature and strain data of other channels to estimate the effects of temperature data in strain retrieval. This scenario is designed to incorporate information associated with thermal effects (distributed temperature sensors) as well as operational and other environmental influences that are implicitly embedded in neighboring strain measurements.

4.2. Hyperparameter tuning

In the training process, proper selection of hyper-parameters is essential for optimizing the performance of all ML models. Common search strategies include grid search and random search, but these approaches often require substantial computational resources and time [48,49]. A more advanced alternative is Bayesian optimization [50], a sequential model-based technique for hyper-parameter tuning, particularly effective when each model evaluation (training and validation) is computationally expensive. Bayesian optimization constructs a probabilistic surrogate model that approximates the relationship between hyper-parameters and performance. An acquisition function then uses this surrogate to choose the next set of hyper-parameters, balancing exploration of uncertain regions with exploitation of promising ones. Because it leverages information from previous evaluations, Bayesian optimization typically needs far fewer iterations than grid or random search, making it suitable for scenarios with limited computational resources or time-consuming model training.

In this work, Bayesian optimization is used to determine the optimal hyper-parameters. The search ranges (or categories) for each model are listed in Table 1. For all models, the Mean Squared Error (MSE) on the validation set is used to select the optimal configuration. After tuning, a prefix “O” is added to each model name to distinguish optimized models from their non-optimized counterparts.

Specifically for O-ANN training, convergence follows the default scikit-learn stopping criterion with a tolerance of $1e^{-4}$: training terminates when the improvement in the loss is less than the tolerance for 10 consecutive iterations, or when the maximum number of iterations is reached. With the maximum iteration number of 3000, the ANN converged well before reaching this limit. The numbers of training epochs required for convergence in Scenarios 1–5 are 35, 191, 155, 89, and 74, respectively. This indicates that the training process converged within a limited number of epochs rather than being terminated by the maximum iteration. Early stopping is not enabled in this study.

4.3. Performance evaluation

To evaluate the performance of all ML models, one of the commonly used statistical metrics, Mean Absolute Error (MAE), is employed in this work. Its mathematical expression is given in Eq. (10). MAE measures the mean absolute difference between the true and predicted strain values. In Eq. (10), y_i denotes the true strain values, and $\hat{y}_i$ represents the predicted values. In MAE, higher metric values indicate worse model performance, and lower values indicate better performance. In addition to MAE, RMSE and R^2 in Eqs. (11) and (12) are employed to provide complementary evaluation perspectives. RMSE penalizes large deviations strongly, while R^2 measures the proportion of variance explained by the model and provides insight into generalization behavior.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right| \tag{10}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2} \tag{11}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N}\left(y_i - \bar{y}\right)^2} \tag{12}$$

4.4. Results

4.4.1. Strain retrieval results

For Scenario 1, to examine the strain retrieval performance, the strain data in the training, validation, and testing phases are plotted over time in Fig. 14. In the training and validation phases, the four models are able to predict the strain data reasonably well at certain periods (e.g., 02/28–03/14). However, in the testing phase, all models show limited accuracy. In particular, the O-LGBM and O-XGBoost models introduce more noise and fluctuations in the predicted strain. Overall, the models exhibit noticeable prediction errors, indicating that the bias learned during training is not captured accurately. This indicates that even during training and validation, predicting strain using only one temperature sensor is challenging. Although a general relationship exists, the correlation is not strong enough to be captured accurately by either linear or nonlinear models. In other words, temperature data from a single sensor do not provide sufficient information for reliable strain retrieval.

In Scenario 2 when temperature data from multiple sensors are used for strain prediction, Fig. 15 shows the predicted strain over time. The O-LGBM, O-XGBoost, and O-ANN models provide improved predictions in the training and validation phases. For the O-RR model, although its performance is better than in Fig. 14(a), noticeable differences remain between 02/07 and 02/23. In the testing phase, the predicted strain

Table 1
Hyperparameters (Sc. : Scenario).

Model	Hyper-parameters	Data type	Tuning range	Sc.1	Sc.2	Sc.3	Sc.4	Sc.5
RR	‘alpha’	Real	[1e-1,1e9]	0.1	0.1	87,215.03	0.1	0.1
LGBM	‘n_estimators’	Integer	[10,2000]	1918	434	1927	1809	1504
	‘max_depth’	Integer	[1,50]	1	50	42	27	33
	‘num_leaves’	Integer	[2,2000]	2	1176	1366	1435	308
	‘learning_rate’	Real	[1e-2,1e0]	1	0.255	0.118	0.075	0.049
	‘min_child_samples’	Integer	[5,50]	10	17	37	37	50
	‘subsamples’	Real	[0.5,1]	0.5	0.537	0.994	0.712	0.62
	‘colsample_bytree’	Real	[0.5,1]	0.569	0.902	0.593	0.9	0.584
XGBoost	‘n_estimators’	Integer	[10,2000]	1443	459	1587	605	1274
	‘max_depth’	Integer	[1,50]	1	37	46	41	24
	‘max_leaves’	Integer	[1,50]	50	42	50	39	17
	‘learning_rate’	Real	[1e-2,1e0]	1	0.274	0.067	0.085	0.34
ANN	‘activation’	Categorical	{‘relu’, ‘tanh’, ‘identity’}	‘relu’	‘relu’	‘relu’	‘relu’	‘relu’
	‘learning_rate’	Categorical	{‘constant’, ‘invscaling’, ‘adaptive’}	‘adaptive’	‘adaptive’	‘adaptive’	‘adaptive’	‘adaptive’
	‘max_iter’	Categorical	{1000, 2000, 3000}	3000	3000	3000	3000	3000

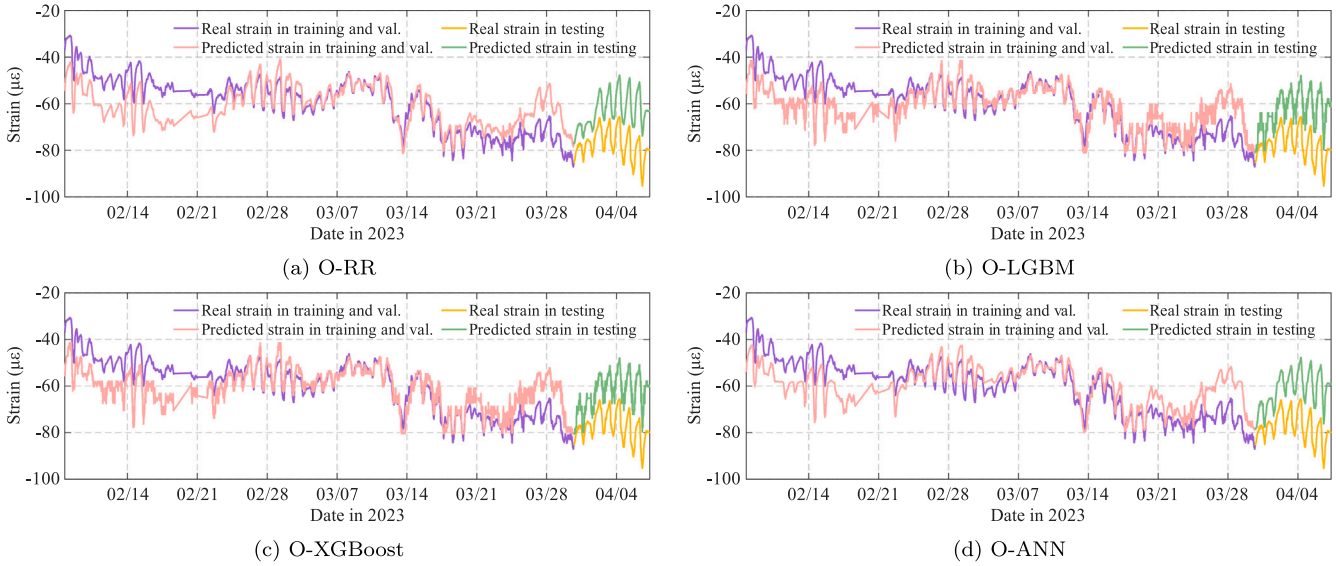


Fig. 14. Scenario 1: true and predicted strain data using different ML models.

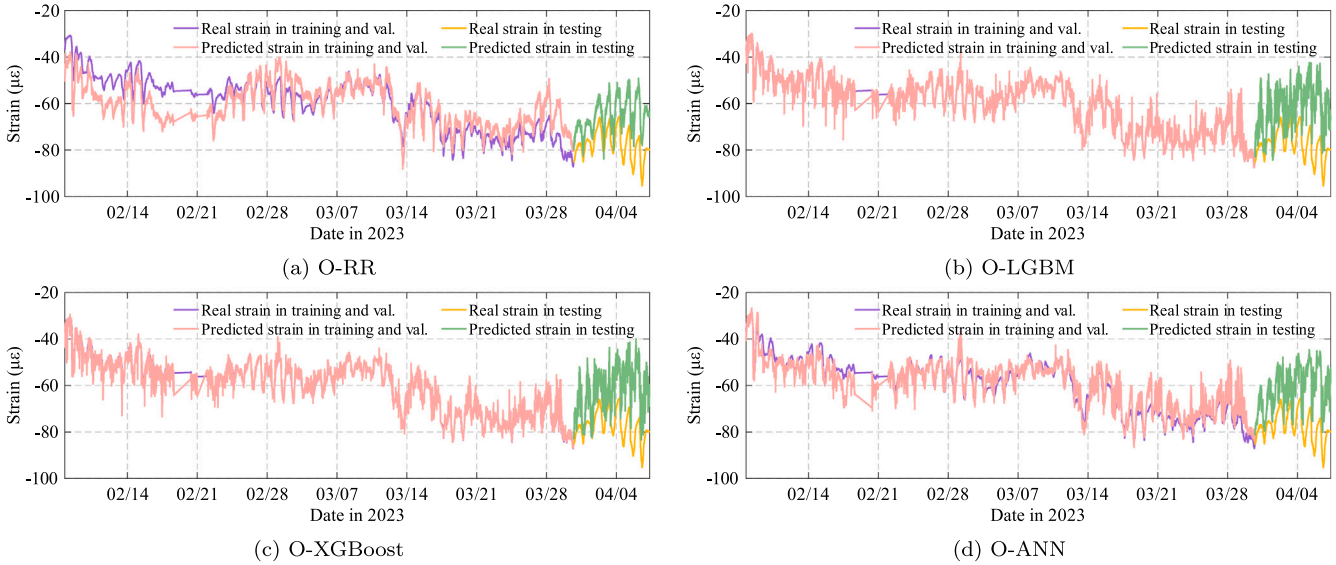


Fig. 15. Scenario 2: true and predicted strain data using different ML models.

still does not align well with the real measurements, and the predictions appear scattered and disordered.

In Scenario 3, the historical temperature data from multiple sensors are included, since delay effects may exist between temperature changes and strain response, as discussed in [37]. Here, 59 past temperature data points from each sensor are incorporated. As in the previous scenarios, the strain retrieval results over time are shown in Fig. 16. Similar patterns to Scenario 2 can be observed. The O-RR and O-ANN models produce stable predictions with a few noisy fluctuations. For the O-LGBM and O-XGBoost models, the predicted strain matches the true values closely during the training and validation phases (the pink and purple curves largely overlap in Figs. 16(b) and 16(c)). However, in the testing phase, their performance remains poor. The predicted strain does not align with the actual measurements, indicating that including historical temperature data does not necessarily improve generalization in real field conditions.

In practical engineering, there can be various influencing factors causing variations in strains, and temperature is an important one among them. As shown in Fig. 7(b), the strain data from all gauges are closely related, which suggests that the strain at one gauge may be retrieved using strain measurements from other gauges. Under this condition, if one sensor produces faulty data, strain data from the remaining gauges can be used instead of relying solely on temperature information.

In the previous scenarios, we observed that using temperature data alone may lead to noticeable errors in strain retrieval. Therefore, Scenario 4 examines the use of strain data from neighboring strain gauges for reconstructing the response at S4. Fig. 17 presents the predicted and actual strains in all phases. We can see that the O-LGBM, O-XGBoost, and O-ANN models match the real strain data well in the training and validation phases, while the O-RR model shows noticeable prediction errors. However, in the testing phase, the O-RR model achieves a prediction accuracy comparable to that of the O-LGBM and O-XGBoost models. The O-ANN model performs the worst in testing, with clear and significant prediction errors.

This improved performance is possible because strain measurements from other gauges contain not only temperature-related information but

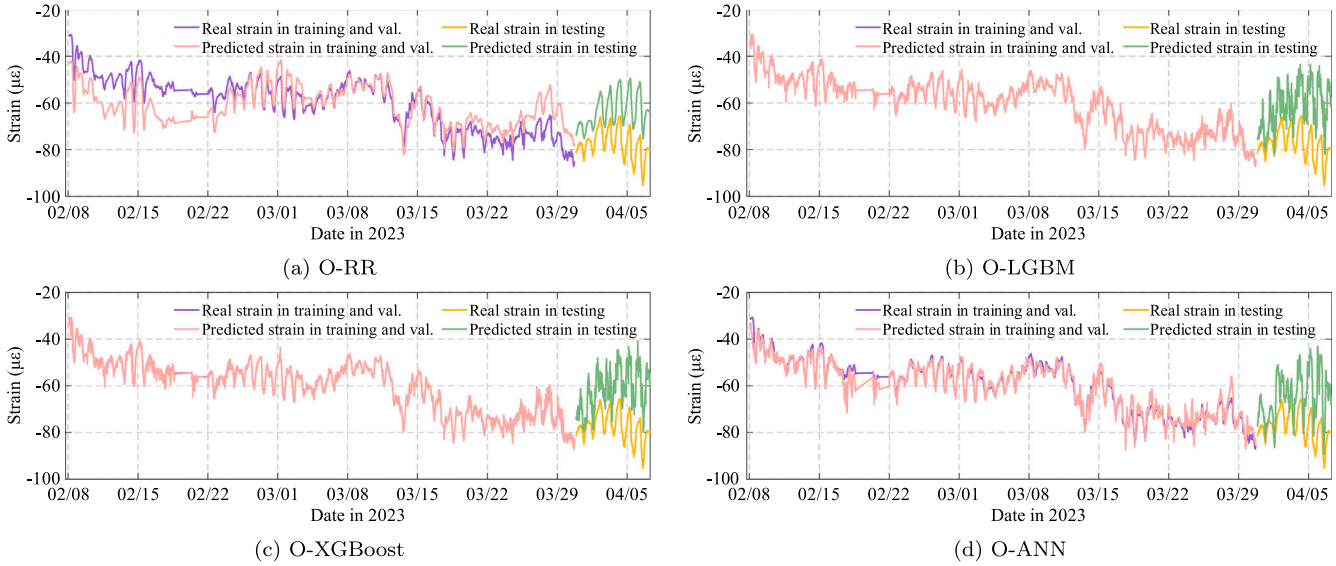


Fig. 16. Scenario 3: true and predicted strain data using different ML models.

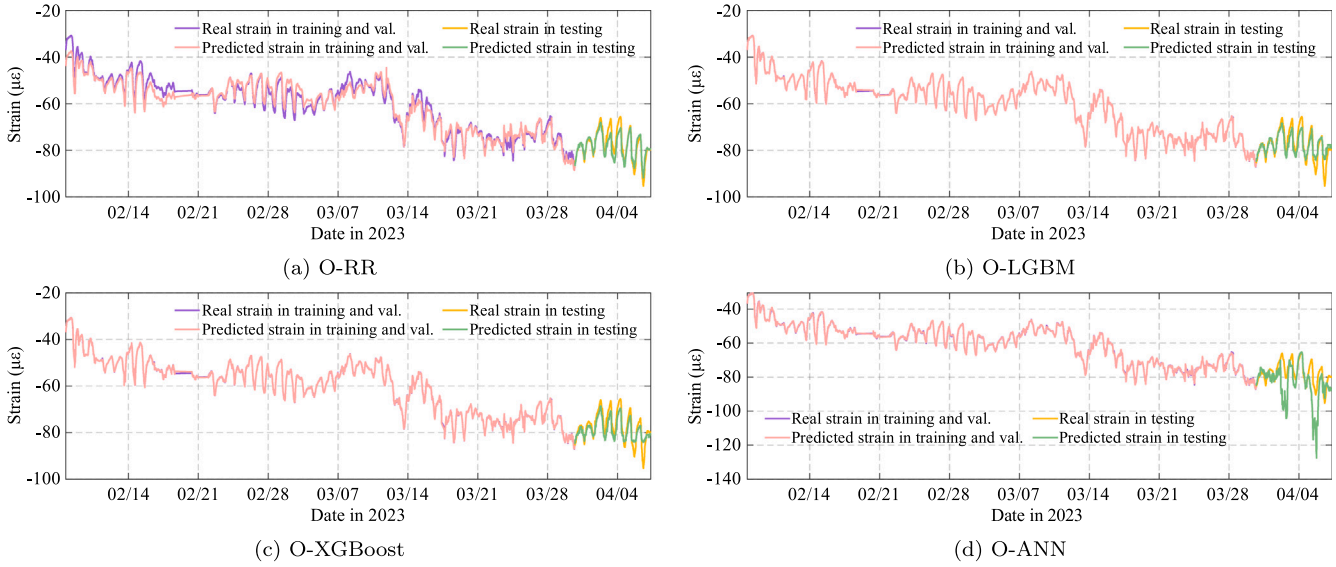


Fig. 17. Scenario 4: true and predicted strain data using different ML models.

also effects from various environmental and operational factors, such as wind, humidity, rainfall, and snow. These factors also influence strain variations. While temperature is a major contributor, other influences in field measurements cannot be ignored, unlike in controlled laboratory settings. Capturing all these factors directly is challenging, but using "healthy" data from other sensors provides an effective way to account for them during strain retrieval.

To further examine the influence of temperature data, Scenario 5 incorporates thermal data from all temperature sensors together with strain data from all gauges except S4 to retrieve the strain at S4, without using historical data. Fig. 18 shows the predicted and actual strains. It can be seen that the strain retrieved by the O-LGBM, O-XGBoost, and O-ANN models matches the real strain very closely in the training and validation phases. However, in the testing phase, the O-ANN model performs the worst, while the O-LGBM and O-XGBoost models produce similar and more stable results. For the O-RR model, noticeable differences remain between the predicted and real strain, but its testing performance is relatively better than in the earlier scenarios.

4.4.2. Model performance evaluation

For evaluating the performance of all models, Fig. 19 summarizes their MAE values in the training, validation, and testing phases for all scenarios. In Scenarios 1–3, where only temperature information is used as input, the nonlinear models (O-LGBM, O-XGBoost, and O-ANN) achieve the relatively low MAEs in the training and validation phases, in particular when multiple sensors and historical temperature records are included (Scenarios 2 and 3). Their testing MAEs are, however, higher than in the training and validation phases, which indicates clear overfitting. The O-RR model yields slightly large MAEs during training and validation but gives the smallest testing MAE in all three temperature-only scenarios. Specifically, when data of multiple temperature sensors are utilized in Scenario 2, the MAE value in testing phase decrease by 15.3%. The testing MAE in these cases remains between 13 and 17 με, which shows that temperature alone is not sufficient for highly accurate strain retrieval, even when data from several sensors or historical temperature features are provided. These results support the earlier observation that the relationship between temperature and strain

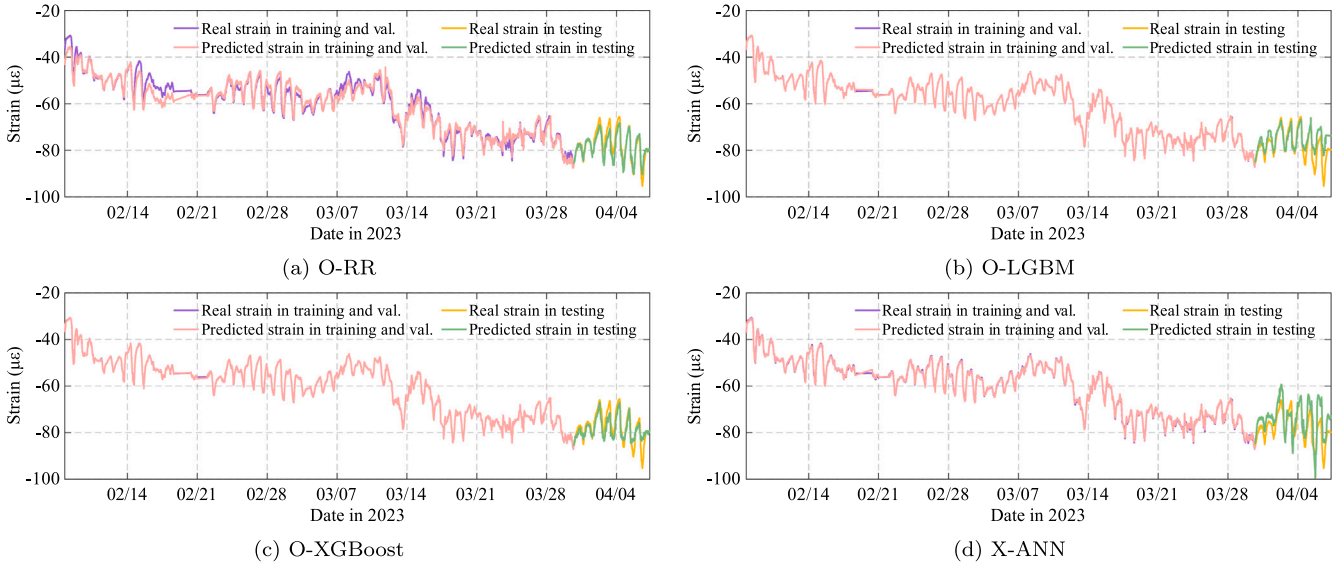


Fig. 18. Scenario 5: true and predicted strain data using different ML models.

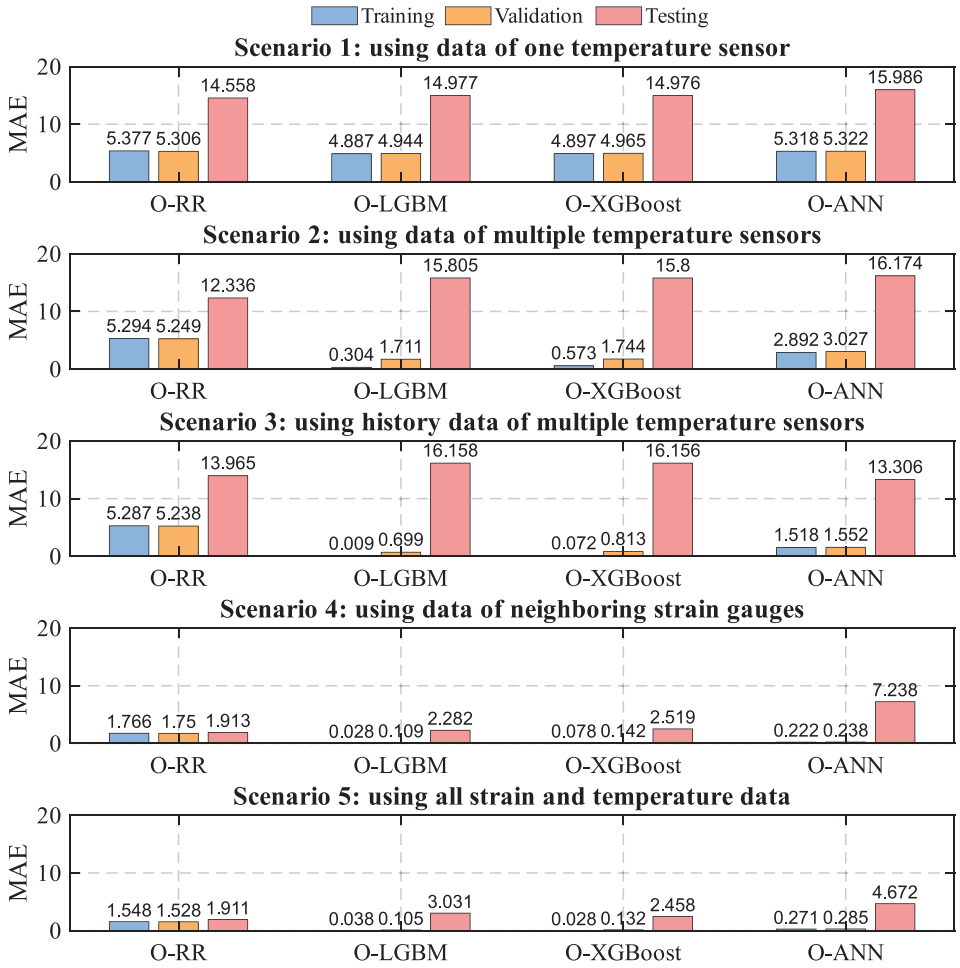


Fig. 19. Performance metrics of all models in different scenarios.

is predominantly linear and that increasing the input dimensionality mainly amplifies the tendency of nonlinear models to overfit.

Scenarios 4 and 5 use strain measurements from other gauges as inputs, with and without additional temperature data. Here, the MAE in the testing phase decreases by approximately one order of magnitude compared with the temperature-only scenarios. When only strain from neighboring gauges is used (Scenario 4), the nonlinear models again attain the smallest MAE in the training and validation phases, but the O-RR

model gives the lowest MAE in the testing phase. This indicates that a simple linear model is sufficient to capture the relationship between S4 and the remaining strain gauges, while the more flexible models tend to fit noise and operational variability. When temperature data are added to these strain inputs (Scenario 5), the training and validation MAE of the nonlinear models is further reduced, but the testing MAE does not improve and in some cases increases. Compared to the best MAE value in Scenarios 1–3 in the testing phase, the MAE decreases by 84.5% when all temperature and strain data are utilized in this scenario.

The good performance of Scenario 5 can be explained by the fact that long-term field strain measurements are governed by multiple sources of variability. Temperature dominates the quasi-static response. However, operational actions and additional environmental effects (e.g., wind and humidity) also introduce fluctuations that are not fully observable from temperature measurements alone. Scenario 5 combines explicit thermal information from distributed temperature sensors with strain measurements from neighboring gauges, which act as proxy signals for operational and environmental influences occurring at the same time. As a result, Scenario 5 reduces the unmodeled variance in the prediction target and improves robustness in the testing phase, whereas temperature-only scenarios (Scenarios 1–3) are more sensitive to noise and operational disturbances and therefore exhibit poorer generalization. However, compared to Scenario 4, the O-RR model in Scenario 5 shows a modest reduction in MAE across all phases, yet the overall gain in testing accuracy is small. This is because the strain measurements from other gauges already include information about temperature effects, as well as other environmental and operational influences. As a result, adding temperature data again does not introduce new information for the ML models and therefore does not lead to meaningful improvements in prediction performance.

To make the evaluation more general, two other evaluation metrics, RMSE and R^2, for models of all scenarios in different phases have been listed in Table 2. The inclusion of RMSE and R^2 further reveals clear overfitting behavior in nonlinear models under temperature-only scenarios (e.g., Scenarios 2 and 3). While O-LGBM and O-XGBoost achieve nearly perfect training and validation R^2 values (approaching 1.0), their testing R^2 becomes negative, indicating that the models perform worse than a simple mean predictor on unseen data, which is a strong indicator of severe overfitting.

This phenomenon can be explained by the high dimensionality and multicollinearity of temperature inputs. When multiple temperature sensors and historical features are introduced (Scenario 3), nonlinear models with high representational capacity can fit noise and short-term fluctuations in the training data. However, due to seasonal variation and distribution shifts between the training and testing periods, these learned nonlinear patterns do not transfer to unseen data.

In contrast, O-RR exhibits moderate but consistent performance across all phases, with significantly small discrepancies between training and testing metrics. The linear structure of ridge regression restricts model variance and prevents excessive sensitivity to minor fluctuations in the training data. This behavior aligns with the predominantly linear nature of temperature-induced strain, suggesting that increasing model complexity mainly increases variance without improving bias under field conditions.

Overall, the combined comparison across all scenarios demonstrates that O-RR consistently provides the most reliable generalization performance, particularly in the testing phase. Nonlinear models can fit the training data very accurately, especially when many thermal or strain features are supplied, but they do not translate this advantage into better performance on unseen data and are therefore less suitable for robust field applications than the linear model. These findings reveal that under temperature-dominated quasi-static strain behavior, increasing model complexity does not necessarily improve predictive robustness. This insight has practical implications for SHM system design, suggesting that model simplification and physically informed feature selection may be more critical than algorithmic sophistication.

The good generalization of the RR model can be explained by both the physical nature of thermal strain and the statistical characteristics of the field dataset. First, the dominant component of long-term strain is quasi-static and temperature-driven. For steel members in this study, the temperature-induced strain primarily follows thermal expansion/contraction and restraint effects, which can be approximated by a linear mapping between strain and a linear combination of temperature measurements at different locations [37]. This is consistent with the strong negative correlations observed between temperature and strain across most channels. Second, the temperature sensors are highly correlated with each other, resulting in strong multicollinearity and feature redundancy. In such cases, flexible nonlinear models (O-LGBM, O-XGBoost, and O-ANN) can exploit small fluctuations and noise patterns during training, producing very low training/validation errors but poor transferability to unseen periods. The RR model, by contrast, applies l_2 regularization, which can stabilize coefficient estimation under multicollinearity and suppress sensitivity to noise. Third, field data are nonstationary: solar radiation, thermal gradients across the deck, and operational effects (e.g., wind and humidity) introduce distribution shifts between the training and testing periods. Nonlinear models are prone to learning training-specific patterns that do not persist under such shifts, leading to severe overfitting. Instead, the RR model tends to capture the dominant, physically meaningful trend and therefore achieves more reliable performance on the testing set.

ANN is one of the most powerful nonlinear models. To further examine model behavior beyond aggregate error metrics, a residual analysis was conducted for the ANN model, where the residual is defined as the

Table 2
Strain retrieval evaluation using RMSE and R^2 in scenarios 1-5.

Dataset	Scenarios	Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
	Models	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2
Training	O-RR	6.886	0.586	6.318	0.651	6.535	0.617	2.238	0.956	2.000	0.965
	O-LGBM	6.486	0.632	0.503	0.998	0.015	1.000	0.039	1.000	0.054	1.000
	O-XGBoost	6.487	0.632	0.923	0.993	0.100	1.000	0.107	1.000	0.038	1.000
	O-ANN	6.772	0.599	3.981	0.862	2.205	0.956	0.300	0.999	0.354	0.999
Validation	O-RR	6.816	0.593	6.273	0.655	6.469	0.629	2.214	0.957	1.972	0.966
	O-LGBM	6.562	0.623	2.920	0.925	1.106	0.989	0.206	1.000	0.189	1.000
	O-XGBoost	6.578	0.621	2.950	0.924	1.370	0.983	0.254	0.999	0.239	1.000
	O-ANN	6.783	0.597	4.155	0.849	2.296	0.953	0.326	0.999	0.378	0.999
Testing	O-RR	15.326	−5.060	13.358	−3.603	14.819	−4.668	2.515	0.837	2.493	0.840
	O-LGBM	15.939	−5.554	17.757	−7.134	17.925	−7.293	3.033	0.763	4.262	0.531
	O-XGBoost	15.923	−5.541	17.859	−7.228	17.681	−7.069	3.089	0.754	3.197	0.736
	O-ANN	16.805	−6.286	17.909	−7.274	15.459	−5.168	11.133	−2.197	5.634	0.181

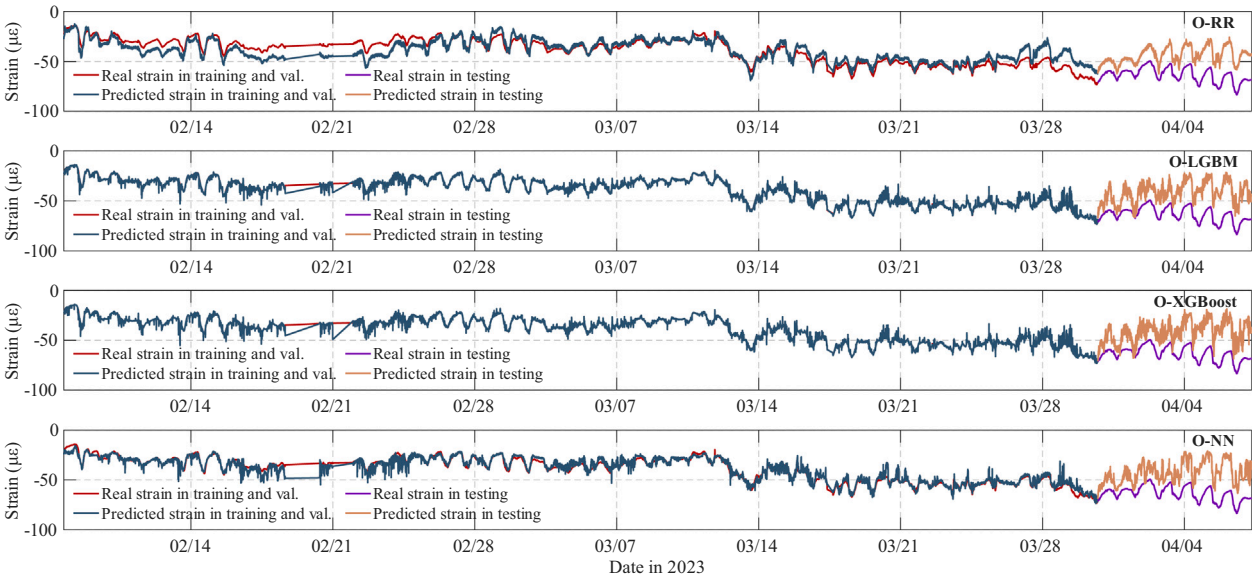
Table 3
Statistics of ANN residuals for all phases under scenarios 1–5.

Phase	Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
Training	−1.200	6.666	−1.034	3.844	−0.202	2.196	−0.006	0.300	−0.059	0.349
Validation	−1.392	6.639	−1.074	4.015	−0.212	2.286	−0.0002	0.326	−0.058	0.374
Testing	−15.986	5.183	−16.172	7.695	−13.194	8.056	5.905	9.440	−3.464	4.445

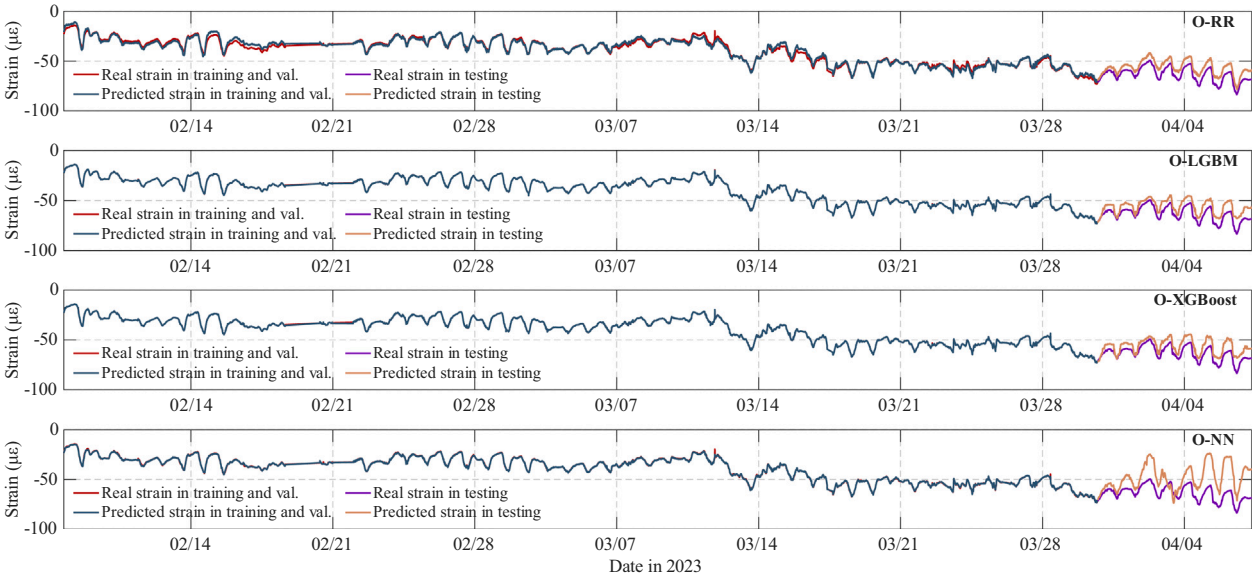
difference between measured and predicted strains. The mean value μ and standard deviation σ of residuals for training, validation, and testing phases are summarized in Table 3. The ANN yields small residual magnitudes in training and validation (especially in Scenarios 2–5), whereas testing residuals become substantially large in the temperature-only scenarios (Scenarios 1–3), which is consistent with the overfitting and generalization issues indicated by the performance metrics. In contrast, when neighboring strain measurements are included (Scenarios 4–5), both the residual mean value and standard deviation in testing are reduced, demonstrating improved robustness under field conditions.

4.4.3. Results of another strain gauge

To further examine the generality of the conclusions derived from S4, the same modeling approach was applied to another strain gauge (S8). The four ML models were retrained using S8 as the target output under Scenarios 2 and 5. Fig. 20 presents the comparison of predicted



(a) Strain retrieval in Scenario 2



(b) Strain retrieval in Scenario 5

Fig. 20. Comparison of scenarios 2 and 5 using O-RR and data from S8.

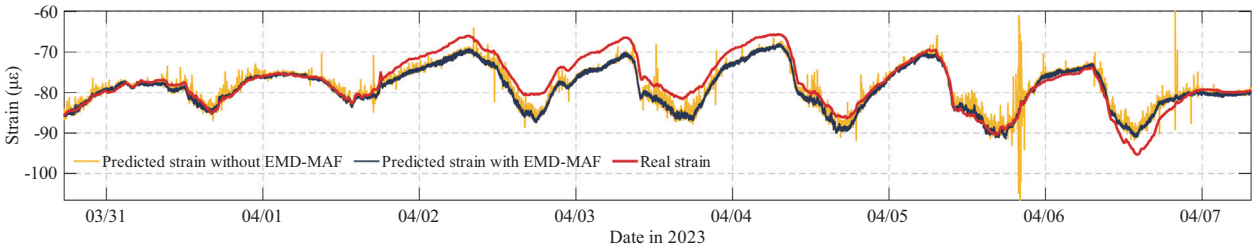


Fig. 21. Strain retrieval results with or without EMD-MAF during testing.

and measured strain for S8. Similar trends to those observed for S4 are evident. Specifically, Scenario 5 consistently provides improved training and validation performance compared to Scenario 2. In addition, the testing error is substantially reduced when strain information from neighboring gauges is included, which indicates that incorporating neighboring strain information can greatly enhance strain retrieval robustness. Moreover, the O-RR model maintains stable generalization behavior in the testing phase. This further supports the observation that the temperature-strain relationship is predominantly linear and that linear models are sufficient for practical field applications. These results demonstrate that the findings drawn in this study are not limited to a single strain channel but are representative of the monitoring system.

4.4.4. Results without EMD-MAF

To evaluate the contribution of the proposed EMD-MAF preprocessing, another study was conducted for the strain retrieval without applying EMD-MAF, i.e., by using the original measured strain data of S4 with a sampling frequency of approximately 0.1 Hz directly, rather than the extracted quasi-static thermal strain component. Based on the results presented in the previous sections, the O-RR demonstrates the most reliable generalization performance under field conditions and therefore is adopted here.

Specifically, we trained O-RR under the same strategy as in Scenario 5, but with or without EMD-MAF preprocessing. Fig. 21 compares the corresponding predictions in the testing phase. Both approaches can capture the overall quasi-static trend of the strain response. However, the prediction without EMD-MAF exhibits noticeably large high-frequency fluctuations and occasional spikes, which deviate from the thermal strain response. In contrast, the prediction with EMD-MAF is smoother and follows the thermal strain response consistently across the testing period. These observations indicate that EMD-MAF improves robustness by suppressing short-term operational disturbances and measurement noise. As a result, the retrieved strain becomes stable and physically interpretable as a quasi-static response, which is beneficial for practical SHM applications.

4.5. Discussion on the applicability to other structural types

Although this study is validated on a steel footbridge, the proposed method is in principle applicable to reinforced concrete (RC) and steel-concrete composite beams. Methodologically, the workflow includes two general steps: extracting the quasi-static thermal component using the EMD-MAF procedure and retrieving missing/faulty strain using data-driven regression models. Both steps can be transferred to other structural systems. However, compared with steel members, RC and composite components may exhibit additional nonlinearity and time-dependence due to creep, shrinkage, cracking, reinforcement restraint, and moisture-related effects. These mechanisms can modify the temperature-strain relationship, potentially reducing the effectiveness of temperature-only retrieval (Scenarios 1–3) and increasing the need for richer environmental inputs (e.g., humidity) and longer training periods spanning different seasons. Nevertheless, the results from Scenarios 4–5 indicate that when neighboring strain measurements are available, retrieval remains robust because these channels implicitly encode thermal, operational, and other environmental influences. Therefore, with appropriate sensor placement and inclusion of relevant environmental variables, the method has the potential to be extended to RC and composite systems for strain retrieval.

5. Concluding remarks and future work

This study investigated the influence of temperature on long-term strain measurements collected from a footbridge and evaluated the feasibility of using machine learning (ML) to retrieve missing or faulty strain data. A hybrid EMD–MAF method was introduced to extract thermal components from strain signals, and five scenarios were tested to assess how different combinations of temperature and strain inputs affect retrieval accuracy. The key findings are summarized below:

(1) The proposed EMD–MAF method effectively extracts temperature-induced strain while suppressing noise and operational fluctuations, maintaining a strong correlation with the original signal. Temperature and strain exhibit a clear, predominantly linear relationship, enabling the retrieval of strains through linear models.
(2) When using temperature-only inputs, ridge regression mostly provides good generalization performance, while nonlinear models overfit the training data and perform poorly on unseen data. Using multiple temperature sensors improves strain retrieval compared to relying on a single thermal measurement, which results in a decrease in MAE by 15.3% in testing. This can be caused by the non-uniform temperature field along the bridge deck. Including historical temperature features does not significantly improve performance in field conditions and increases the model overfitting risk.
(3) When available, strain data from other gauges outperform temperature data for strain retrieval, as they encode thermal, operational, and environmental effects simultaneously. In such cases, ridge regression again delivers the most stable and robust retrieval results. Specifically, when all temperature and strain data are utilized, the MAE value in testing is decreased by 84.5% compared to using temperature data only, whereas nonlinear models add complexity without improving generalization.

Future efforts should focus on extending the method to different bridge types (including RC and composite members) and climates, integrating physics-informed constraints to improve generalization, and developing adaptive, real-time sensor-fault recovery systems. Incorporating additional environmental variables may further enhance stability. Finally, coupling strain retrieval with downstream SHM tasks, such as anomaly detection and condition assessment, will help evaluate its practical impact on long-term monitoring.

CRediT authorship contribution statement

Zhenkun Li: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis,

Conceptualization. **Rolands Kromanis:** Writing – review & editing, Resources, Project administration, Funding acquisition, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the DeSIRE (Designing Systems for Informed Resilience Engineering) programme for funding the sensor network on the UT campus bridge, from which the data used in this study were obtained.

References

[1] 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. J Struct Eng 2020;146(5):04020073. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002535

[2] Lin Y, Fu Y, Sun J, Xie R, He X. Robust seismic response evaluation considering an uncertainty emerging in asymmetric bridges. Reliab Eng Syst Saf 2026;266:111740. https://doi.org/10.1016/j.ress.2025.111740

[3] Zhang Z, Fu Y, Sun Z. Advancements in digital twin-enhanced health monitoring and condition assessment of cable-supported bridges. Structures 2025;79:109448. https://doi.org/10.1016/j.istruc.2025.109448

[4] European Commission, European Commission, Gkoumas K, Balen M, Grosso M, Pekár F, Marques Dos Santos F, Haq G, Ortega Hortelano A, Tsakalidis A. Research and innovation in bridge maintenance, inspection and monitoring: a european perspective based on the transport research and innovation monitoring and information system (TRIMIS). Publications Office; 2019. https://doi.org/10.2760/16174

[5] American Society of Civil Engineers. ASCE's 2021 infrastructure report card: bridges; 2021. https://infrastructurereportcard.org/wp-content/uploads/2020/12/Bridges-2021.pdf[Last accessed on 10 November 2025].

[6] Calvi GM, Moratti M, O'Reilly GJ, Scattarreggia N, Monteiro R, Malomo D, Calvi PM, Pinho R. Once upon a time in italy: the tale of the morandi bridge. Struct Eng Int 2019;29(2):198–217. https://doi.org/10.1080/10168664.2018.1558033

[7] Agdas D, Rice JA, Martinez JR, Lasa IR. Comparison of visual inspection and Structural-Health monitoring as bridge condition assessment methods. J Perform Constr Facil 2016;30(3):04015049. https://doi.org/10.1061/(ASCE)CF.1943-5509.0000802

[8] Spencer BF, Hoskere V, Narazaki Y. Advances in computer vision-based civil infrastructure inspection and monitoring. Engineering 2019;5(2):199–222. https://doi.org/10.1016/j.eng.2018.11.030

[9] Dong CZ, Catbas FN. A review of computer vision–based structural health monitoring at local and global levels. Struct Health Monit 2020;20(2):692–743. https://doi.org/10.1177/1475921720935585

[10] Negi P, Kromanis R, Dorée AG, Wijnberg KM. Structural health monitoring of inland navigation structures and ports: a review on developments and challenges. Struct Health Monit 2023;23(1):605–45. https://doi.org/10.1177/14759217231170742

[11] Farrar CR, Worden K. An introduction to structural health monitoring. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 2006;365(1851):303–15. https://doi.org/10.1098/rsta.2006.1928

[12] Worden K, Manson G. The application of machine learning to structural health monitoring. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 2006;365(1851):515–37.

[13] An Y, Chatzi E, Sim SH, Laflamme S, Blachowski B, Ou J. Recent progress and future trends on damage identification methods for bridge structures. Struct Control Health Monit 2019;26(10):e2416. https://doi.org/10.1002/stc.2416

[14] Li Z, Lan Y, Lin W. Indirect damage detection for bridges using sensing and temporarily parked vehicles. Eng Struct 2023;291:116459. https://doi.org/10.1016/j.engstruct.2023.116459

[15] Giglioni V, Venanzi I, Poggioni V, Milani A, Ubertini F. Autoencoders for unsupervised real-time bridge health assessment. Computer-Aided Civil and Infrastructure Engineerin 2023;38(8):959–74. https://doi.org/10.1111/mice.12943

[16] Li Z, Lin W, Zhang Y. Real-time drive-by bridge damage detection using deep autoencoder. Structures 2023;47:1167–81. https://doi.org/10.1016/j.istruc.2022.11.094

[17] Rainieri C, Fabbrocino G, Cosenza E. Near real-time tracking of dynamic properties for standalone structural health monitoring systems. Mech Syst Signal Process 2011;25(8):3010–26. https://doi.org/10.1016/j.ymssp.2011.04.010

[18] An X, Hou J, Jankowski Ł, Zhang Q. A physics-augmented deep learning framework for structural dynamic load identification with FRF-guided state expansion. Mech Syst Signal Process 2026;247:113965. https://doi.org/10.1016/j.ymssp.2026.113965

[19] Zhang C, Mousavi AA, Masri SF, Gholipour G, Yan K, Li X. Vibration feature extraction using signal processing techniques for structural health monitoring: a review. Mech Syst Signal Process 2022;177:109175. https://doi.org/10.1016/j.ymssp.2022.109175

[20] Tan X, Poorghasem S, Huang Y, Feng X, Bao Y. Monitoring of pipelines subjected to interactive bending and dent using distributed fiber optic sensors. Autom Constr 2024;160:105306. https://doi.org/10.1016/j.autcon.2024.105306

[21] Wang Z, Yang DH, Yi TH, Zhang GH, Han JG. Eliminating environmental and operational effects on structural modal frequency: a comprehensive review. Struct Control Health Monit 2022;29(11):e3073. https://doi.org/10.1002/stc.3073

[22] Sohn H. Effects of environmental and operational variability on structural health monitoring. Philos. Trans. Royal Soc. A 2006;365(1851):539–60. https://doi.org/10.1098/rsta.2006.1935

[23] Jing Q, Shan Y, Zhang L, Yan Y, Xia Z, Xia Y. Typhoon- and temperature-induced responses of a cable-stayed bridge. Adv Struct Eng 2024;27(16):2773–89. https://doi.org/10.1177/13694332241246378

[24] Croxford AJ, Moll J, Wilcox PD, Michaels JE. Efficient temperature compensation strategies for guided wave structural health monitoring. Ultrasonics 2010;50(4):517–28. https://doi.org/10.1016/j.ultras.2009.11.002

[25] Yarnold MT, Moon FL. Temperature-based structural health monitoring baseline for long-span bridges. Eng Struct 2015;86:157–67. https://doi.org/10.1016/j.engstruct.2014.12.042

[26] Han Q, Ma Q, Xu J, Liu M. Structural health monitoring research under varying temperature condition: a review. J Civ Struct Health Monit 2021;11(1):149–73. https://doi.org/10.1007/s13349-020-00444-x

[27] Ni YQ, Hua XG, Fan KQ, Ko JM. Correlating modal properties with temperature using long-term monitoring data and support vector machine technique. Eng Struct 2005;27(12):1762–73. https://doi.org/10.1016/j.engstruct.2005.02.020

[28] Cross EJ, Koo KY, Brownjohn JMW, Worden K. Long-term monitoring and data analysis of the tamar bridge. Mech Syst Signal Process 2013;35(1):16–34. https://doi.org/10.1016/j.ymssp.2012.08.026

[29] Zhou Y, Sun L. Insights into temperature effects on structural deformation of a cable-stayed bridge based on structural health monitoring. Struct Health Monit 2018;18(3):778–91. https://doi.org/10.1177/1475921718773954

[30] Koo KY, Brownjohn JMW, List DI, Cole R. Structural health monitoring of the tamar suspension bridge. Struct Control Health Monit 2013;20(4):609–25. https://doi.org/10.1002/stc.1481

[31] Zhang J, Huang M, Wan N, Deng Z, He Z, Luo J. Missing measurement data recovery methods in structural health monitoring: the state, challenges and case study. Measurement 2024;231:114528. https://doi.org/10.1016/j.measurement.2024.114528

[32] Oh BK, Glisic B, Kim Y, Park HS. Convolutional neural network–based data recovery method for structural health monitoring. Struct Health Monit 2020;19(6):1821–38. https://doi.org/10.1177/1475921719897571

[33] Bao Y, Li H, Sun X, Yu Y, Ou J. Compressive sampling–based data loss recovery for wireless sensor networks used in civil structural health monitoring. Struct Health Monit 2012;12(1):78–95. https://doi.org/10.1177/1475921712462936

[34] Kromanis R, Kripakaran P. Predicting thermal response of bridges using regression models derived from measurement histories. Computers & Structures 2014;136:64–77. https://doi.org/10.1016/j.compstruc.2014.01.026

[35] Kromanis R, Kripakaran P. Data-driven approaches for measurement interpretation: analysing integrated thermal and vehicular response in bridge structural health monitoring. Advanced Engineering Informatics 2017;34:46–59. https://doi.org/10.1016/j.aei.2017.09.002

[36] Glashier T, Kromanis R, Buchanan C. An iterative regression-based thermal response prediction methodology for instrumented civil infrastructure. Advanced Engineering Informatics 2024;60:102347. https://doi.org/10.1016/j.aei.2023.102347

[37] Mariani S, Kalantari A, Kromanis R, Marzani A. Data-driven modeling of long temperature time-series to capture the thermal behavior of bridges for SHM purposes. Mech Syst Signal Process 2024;206:110934. https://doi.org/10.1016/j.ymssp.2023.110934

[38] Hedegaard BD, French CEW, Shield CK. Effects of cyclic temperature on the Time-Dependent behavior of posttensioned concrete bridges. J Struct Eng 2016;142(10):04016062. https://doi.org/10.1061/(ASCE)ST.1943-541X.0001538

[39] Ding Y, Zhou G, Li A, Wang G. Thermal field characteristic analysis of steel box girder based on long-term measurement data. Int J Steel Struct 2012;12(2):219–32. https://doi.org/10.1007/s13296-012-2006-x

[40] Montgomery DC, Peck EA, Vining GG. Introduction to linear regression analysis. John Wiley & Sons; 2021.

[41] McDonald GC. Ridge regression. Wiley Interdiscip Rev Comput Stat 2009;1(1):93–100.

[42] Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y. Lightgbm: a highly efficient gradient boosting decision tree. In: Guyon I, Von Luxburg U, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, editors. Advances in neural information processing systems, vol. 30. Curran Associates, Inc.; 2017. p. 1–9.

[43] Zhang F, Falchetto AC, Wang D, Li Z, Sun Y, Lin W. Prediction of asphalt rheological properties for paving and maintenance assistance using explainable machine learning. Fuel 2025;396:135319. https://doi.org/10.1016/j.fuel.2025.135319

[44] Agatonovic-Kustrin S, Beresford R. Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. Journal of pharmaceutical and biomedical analysis 2000;22(5):717–27.

[45] Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. [arXiv]. 2014. arXiv:1412.6980.

[46] Bao Y, Tang Z, Li H, Zhang Y. Computer vision and deep learning–based data anomaly detection method for structural health monitoring. Struct Health Monit 2018;18(2):401–21. https://doi.org/10.1177/1475921718757405

[47] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E. Scikit-learn: machine learning in Python. J Mach Learn Res 2011;12:2825–30.
[48] Bergstra J, Bengio Y. Random search for hyper-parameter optimization. J Mach Learn Res 2012;13(10):281–305.
[49] Bousquet O, Gelly S, Kurach K, Teytaud O, Vincent D. Critical Hyper-Parameters: No Random, No Cry. [arXiv] arXiv:1706.03200. 2017.
[50] Bischl B, Binder M, Lang M, Pielok T, Richter J, Coors S, Thomas J, Ullmann T, Becker M, Boulesteix A-L, Deng D, Lindauer M. Hyperparameter optimization: foundations, algorithms, best practices, and open challenges. WIREs Data Min. Knowl. Discovery 2023;13(2):e1484. https://doi.org/10.1002/widm.1484