Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Probabilistic autoencoder-based bridge damage assessment using train-induced responses

Muhammad Zohaib Sarwar [*], Daniel Cantero

Department of Structural Engineering, Norwegian University of Science and Technology, Trondheim, 7491, Norway

ARTICLE INFO

Communicated by J.E. Mottershead

Keywords:
Train-induced vibration
Damage assessment
EWMA
Probabilistic autoencoder
Structural health monitoring
KW51 bridge

ABSTRACT

Structural health monitoring (SHM) systems have been increasingly employed to continually assess the current state of bridges. However, the vast amounts of sensor data generated by SHM systems, along with constantly changing environmental and operational conditions, make structural damage assessment a computationally demanding and challenging task. Traditional data-driven approaches primarily utilise machine learning methods for pattern recognition and feature extraction to address this issue. This paper introduces a methodology for assessing bridge conditions using a probabilistic temporal autoencoder (PTAE). The proposed approach effectively extracts features and captures temporal relationships in multi-sensor data collected only during train crossings. By calculating the reconstruction loss and KL divergence-based of damage features, the methodology enables the identification of potential damage of a monitored bridge. An Exponentially Weighted Moving Average (EWMA) filter and a control chart-based threshold mechanism are applied to further refine the damage assessment process, facilitating the distinction between healthy and progressively deteriorating damage cases. The proposed method is adaptable to various monitoring scenarios and sensor configurations, and robust to varying operational and environmental conditions. The effectiveness of the methodology is assessed using numerically generated data and validated with real-world data from the KW51 bridge. The results demonstrate that the proposed method can detect damage with a limited number of sensors, making it a valuable approach to enhance bridge safety.

1. Introduction

The rapidly ageing bridge infrastructure has become a growing concern worldwide, as a significant number of bridges in developed countries reach the end of their design life. Increased mobility, traffic volume, and climate change have resulted in deviations from original design assumptions, accelerating the deterioration process [1,2]. Presently, bridge maintenance decisions rely primarily on manual and visual inspections, which are time-consuming, expensive, and partly subjective. As a result, there is an urgent need for more efficient and objective methods to monitor and maintain ageing bridges. Structural Health Monitoring (SHM) techniques present a promising alternative to traditional manual inspections, providing real-time, objective data on bridge conditions [3]. Among various SHM techniques, vibration-based methods have attracted considerable attention due to their ability to capture a structure's global behaviour and detect damage without prior knowledge of the damage location [4].

Vibration-based Structural Health Monitoring (SHM) approaches primarily fall into two categories: model-based and data-driven methods. Model-based methods, though accurate, are computationally intensive, making them less feasible for widespread SHM applications [5]. In contrast, data-driven methods leverage data mining and signal processing techniques, offering a more

[*] Corresponding author.
E-mail address: muhammad.z.sarwar@ntnu.no (M.Z. Sarwar).

https://doi.org/10.1016/j.ymssp.2023.111046
Received 16 October 2023; Received in revised form 4 December 2023; Accepted 12 December 2023
Available online 19 December 2023

computationally efficient approach for real-time assessments in larger structures [6]. Numerous data-driven strategies have been discussed in literature, demonstrating effectiveness in identifying bridge damage under various operational and environmental conditions [7–11]. The incorporation of machine learning (ML) in SHM systems has shown potential in navigating the challenges posed by environmental and operational variabilities. ML models, particularly unsupervised learning, stand out when monitoring data mainly includes the original state of the structure, a common occurrence in civil infrastructures [5,12–16]. Combining data-driven and ML approaches appears promising, merging computational efficiency with the capability to adapt to various structural, operational, and environmental challenges, thereby enhancing the robustness and reliability of SHM systems.

The majority of research conducted on data-driven methods for bridge damage assessment primarily relies on ambient vibration data or static effects. However, recent studies have acknowledged that structural responses generated by vehicle crossings can also be effective for bridge damage assessments [16]. In this context, most research related to monitoring techniques and data interpretation focuses on highway bridges, but these techniques are generally applicable to railway bridges as well [17]. Gonzalez and Karoumi [18] proposed an artificial neural network (ANN)-based damage detection method. Their approach utilises deck acceleration and bridge weigh-in-motion data to train a machine learning model, which is then used in conjunction with statistical processes for classifying the bridge's health and damage states. In [19], the authors validated this idea using real bridge measurements. One of the main drawbacks of this strategy is the need to train an ANN model for each individual sensor. Azim and Gul [20] proposed a time series analysis-based method for the global monitoring of railway bridges using operational data. Their method employs the autoregressive moving average (ARMA) model to analyse the free vibration response of the bridge, in order to extract damage features. Similarly, Meixedo et al. [21] presented the idea of using autoregressive (AR) models to extract damage-sensitive features from traffic-induced vibration responses. Their method effectively removes the influence of temperature, train type, and speed from the damage-sensitive features using Multiple Linear Regression (MLR) and Principal Component Analysis (PCA). However, the main challenge with their proposed method is the manual extraction of damage-sensitive features for each sensor, which requires careful analysis of the AR parameters.

Despite the strong potential, the real-world application of vehicle-induced dynamic response for Structural Health Monitoring (SHM) implementation remains limited. Most of the existing works and methodologies have certain limitations, such as disregarding the influence of environmental and operational conditions, specific loading effects, and validating methods using simple numerical models. Consequently, this study aims to develop a novel methodology for damage assessment that can automatically extract damage-sensitive features from responses generated by trains crossing at varying speeds and loading conditions. To this end, a multivariate temporal convolutional autoencoder is proposed. The proposed autoencoder is first trained using the current, healthy state of the bridge. Then, the trained model is utilised for damage assessment. In practice, it is impossible for training data to encompass all varying conditions, such as temperature and loading conditions. Therefore, it is crucial that the proposed autoencoder can generalise the feature space for the healthy state of the bridge and quantify the uncertainty arising from limited data. This challenge is addressed by developing a temporal convolutional autoencoder as a Bayesian deep learning approach. Moreover, for damage assessment, the features extracted from the autoencoder are combined with an Exponentially Weighted Moving Average (EWMA) filter, a pattern recognition algorithm capable of distinguishing minor damage cases under varying operational conditions. The EWMA control chart is employed for threshold selection in damage detection. Likewise, a one-class Support Vector Machine (OC-SVM) is implemented using the input from the EWMA output for damage detection. The proposed methodology is evaluated across a range of different damage scenarios using both numerical and real-life field data.

The unique contribution of this study can be summarised as the development and implementation of an innovative methodology in Structural Health Monitoring (SHM) system that utilises vehicle-induced dynamic responses for damage assessment. A novel multivariate temporal convolutional autoencoder is introduced, augmented with a Bayesian deep learning approach, enabling the autonomous extraction of damage-sensitive features and effective management of uncertainties arising from limited data. This methodology is particularly adept at generalising the feature space of a bridge's healthy state amidst varied operational conditions. Further enhancing its novelty, the approach integrates these features with an Exponentially Weighted Moving Average (EWMA) filter and a one-class Support Vector Machine (OC-SVM), facilitating the sensitive detection of minor damages under fluctuating conditions.

The remainder of the paper is structured as follows. Section 2 offers an overview of the suggested methodology, encompassing details of the probabilistic deep autoencoder and the damage index. Section 3 presents the specifics of the numerical case study, which includes the train-track bridge model used to generate data, as well as an evaluation of the proposed method's performance for the numerical case study. Section 4 describes the monitoring details of railway bridge KW51 and validates the proposed method by detecting various bridge conditions. Lastly, Section 5 summarises the findings of this research.

2. Proposed methodology

2.1. Overview

The main overview of the proposed methodology is shown in Fig. 1, which involves collecting sensor data as trains cross the bridge. The sensor data for a single event is further divided into n windows. The collected data is then utilised to train a probabilistic temporal autoencoder (PTAE). The architecture of the PTAE consists of multiple convolutional blocks and a Long Short-Term Memory (LSTM) recurrent neural network. For damage assessment, the reconstruction loss of each window is computed first, followed by the calculation of a KL (Kullback–Leibler) divergence-based damage feature between the baseline condition and a subsequent single train crossing. In the final step, an Exponentially Weighted Moving Average (EWMA) filter is applied to the damage feature. The EWMA control chart-based threshold mechanism is developed to distinguish between healthy and subsequent damage states. Further details about each step are discussed in subsequent sections.

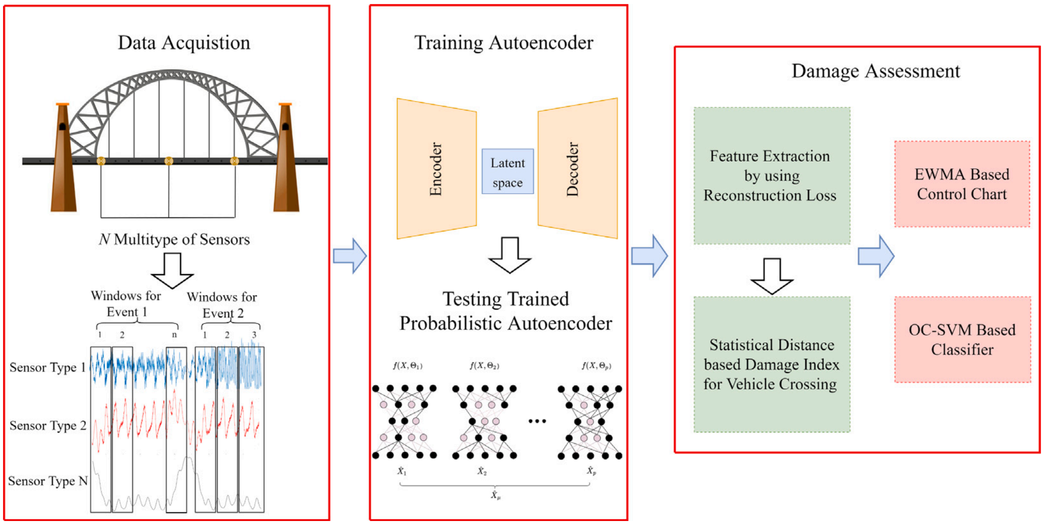


Fig. 1. Overview of proposed damage detection framework.

2.2. Probabilistic deep autoencoder

An autoencoder is a neural network architecture designed to replicate its input X to itself, namely $Y = X$ via a bottleneck structure. A deep autoencoder consists of two main components: an encoder and a decoder. The encoder maps the input data X to a lower-dimensional latent representation z, also known as the bottleneck layer, while the decoder reconstructs the input data from this latent representation. This means that in an autoencoder, we select a model $F^{W,b}(X)$ to concentrate the information required to accurately recreate X. Suppose that we have N input vectors $X = \{x_1, \dots, x_N\} \in \mathbb{R}^{M\times N}$. The transfer function of the encoder and decoder can be expressed by Eqs. (1) and (2) respectively.

$$\begin{aligned} z^{(0)} &= X \\ z^{(l)} &= \phi(W^{(l)} z^{(l-1)} + b^{(l)}), \text{ for } l = 1 \text{ to } L \end{aligned} \tag{1}$$

$$\begin{aligned} y^{(0)} &= z^{(L)} \\ y^{(l-L)} &= \phi'(W'^{(l)} y^{(l-L-1)} + b'^{(l)}), \text{ for } l = L+1 \text{ to } 2L \end{aligned} \tag{2}$$

where ϕ, ϕ' are the activation functions of the encoder and decoder modules, while W, W' and b, b' are the weights and biases of each module, and L is the number of hidden layers in the encoder. In an autoencoder, we fit the model $F^{W,b}(X)$ and optimise the weights W, W' and biases b, b' parameters using the backpropagation algorithm by employing the mean squared error as the loss function, which is expressed as:

$$\mathcal{L} = f(X : \Theta) = \frac{1}{n} \sum_{i=1}^{N} \left(\frac{1}{2} \left\| \hat{x}_i - x_i \right\|^2 \right) + \lambda(\Theta) \tag{3}$$

where λ is a regularisation factor applied to the weights of a specific layer in order to prevent overfitting. In a standard autoencoder, the learning parameters are optimised as single deterministic estimates based on a given dataset. However, in various applications, the ability to represent uncertainty is of paramount importance. Unfortunately, standard autoencoders do not possess the capability to account for or represent model uncertainty.

Bayesian neural networks (BNN) or probabilistic neural networks (PNN) can incorporate uncertainty in a systematic manner, as opposed to standard neural networks. BNNs handle uncertainty by treating weights as stochastic variables with a prior distribution. One major drawback of BNNs is their computational cost. In the case of deep networks, BNNs need to obtain posterior distributions across the network's parameters, which become high-dimensional probability distributions as the model complexity increases. To address this issue, Gal and Ghahramani [22] proposed a more computationally efficient algorithm called Monte Carlo Dropout (MC Dropout). In their work, the authors showed that applying dropout before weight layers in any neural network, regardless of its depth and non-linearities, can be interpreted as a Bayesian approximation of the probabilistic deep Gaussian process. Dropout, which randomly deactivates neurons in a neural network during training, is typically employed as a regularisation technique to reduce overfitting, without being used during prediction. However, when each unit is dropped during prediction with some probability p the output ceases to be probabilistic. Training a neural network with dropout can be considered as training a set of pruned networks with extensive weight sharing. Each dropout configuration corresponds to a sub-network, producing multiple output predictions for a given input, as illustrated in Fig. 2. Uncertainty can be quantified by calculating the variance of numerous predictions across different dropout configurations, while the predictive mean of the output is represented by the mean of these multiple predictions.

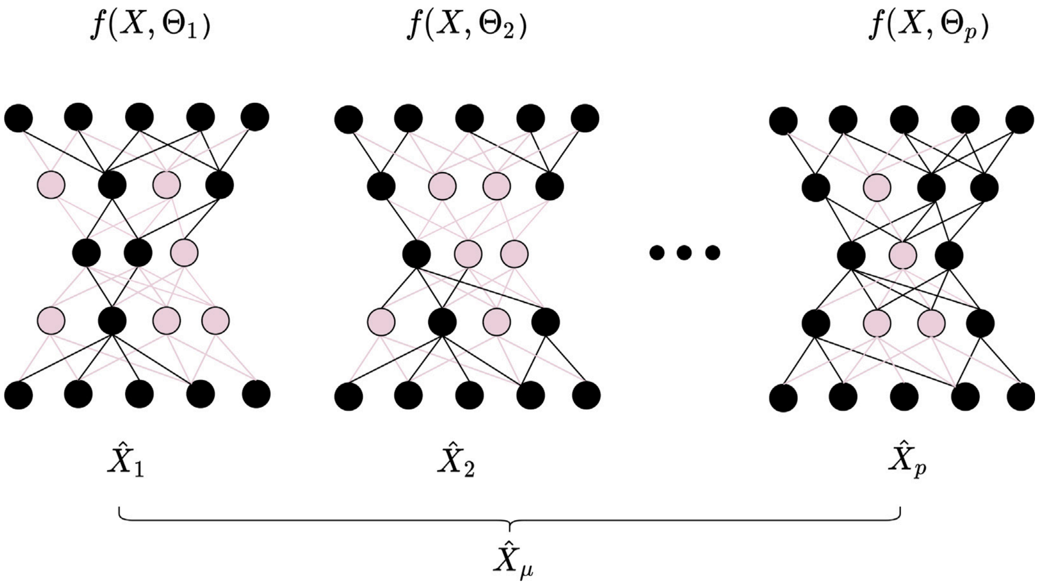


Fig. 2. Different MC-Dropout configurations for autoencoder.

2.3. *Network architecture of probabilistic temporal autoencoder*

Deep learning models have shown great promise in combining information from multiple sensors to perform a range of tasks, including classification and regression [16]. Automatic feature extraction from time series data often relies on the use of 1D convolutional neural networks (1D CNN) and recurrent neural networks (RNN). However, training RNNs for long-term sequences can be challenging due to the vanishing gradient problem encountered during backpropagation [23]. To address this issue, Long Short-Term Memory (LSTM) networks were introduced [24]. In this study, the network architecture of the proposed probabilistic temporal autoencoder (PTAE) is primarily built using 1D CNN and LSTM. The architecture of the proposed PTAE is primarily inspired by the temporal autoencoder proposed by Madiraju [25].

In this study, the input dataset for damage assessment comprises multi-sensor information, including accelerations, displacements, and rotations. Time series data from different sensors exhibit considerable variations in important properties and features on the temporal scale and dimensionality. Considering the nature of multi-sensor information in the time series data, the network architecture is designed to ensure that each time scale contains informative features.

For an autoencoder, it is crucial to have an effective latent representation that can be used to reconstruct the input. In this study, this is achieved by employing PTAE, as shown in Fig. 3. The model's architecture is divided into two stages. The first stage consists of 1D convolutional layers that extract key short-term features, followed by a pooling layer of size P, activation functions namely here as Leaky Relu, and an MC-dropout layer. The first stage reduces the time series data to a more compact representation of the most relevant features. To obtain a latent representation and learn temporal changes, the output of the first stage is fed into an LSTM. The LSTM module learns temporal changes across each time step, collapsing the input signals into all dimensions except the temporal one and casting the inputs into a much smaller latent space. For reconstruction, the decoder is developed by using an upsampling layer of size P followed by a deconvolutional layer to obtain the autoencoder output.

The proposed PTAE is implemented using TensorFlow and Keras. During the initialisation phase, the weights are randomly assigned, and the biases are set to zero. Optimisation of weight and bias parameters within the proposed architecture is carried out using an end-to-end approach, considering the loss function as illustrated in Eq. (3).

2.4. *Data collection and preprocessing*

In the proposed framework, a permanent Structural Health Monitoring (SHM) system is assumed to be installed on the bridge, capable of capturing multiple sensor data such as accelerations, displacements, and rotations at various locations. Sensor measurements are collected exclusively when a train crosses the bridge, with known entry and exit times. It is important to note that due to varying train speeds, each crossing event produces signals of different lengths.

Signal preprocessing is a critical step for both damage assessment and training of the PTAE. This process involves standardising the time series data collected by each sensor, ensuring zero mean and unit variance. Additionally, for every individual train crossing event, the time series signals are divided into appropriate window lengths. Each sliced window's data matrix consists of sensors as rows and sequences of measured responses as columns, as shown in Fig. 4.

To train the autoencoder, numerous train crossing events are gathered when the bridge is in a healthy condition. These events are standardised and divided into fixed-length windows, as previously described. During training, the PTAE focuses on reconstructing the time series sequences present in each data matrix.

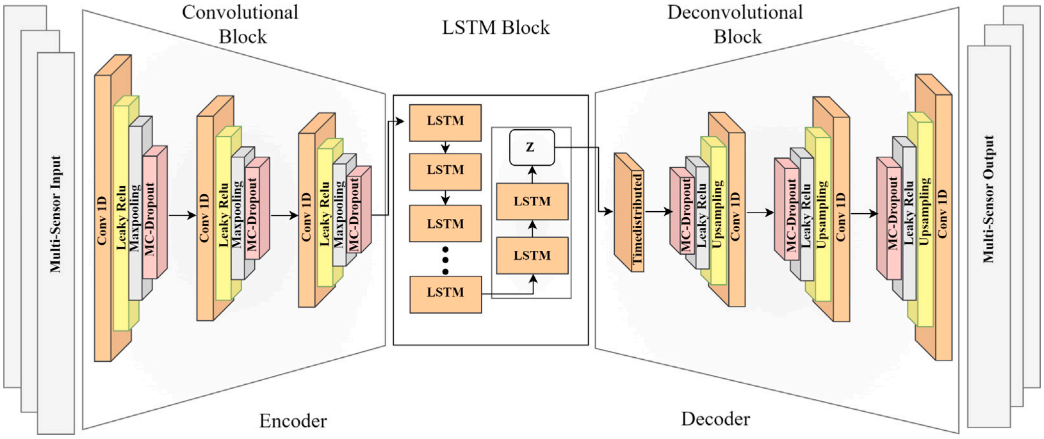


Fig. 3. Architecture of proposed Probabilistic Temporal Autoencoder (PTAE).

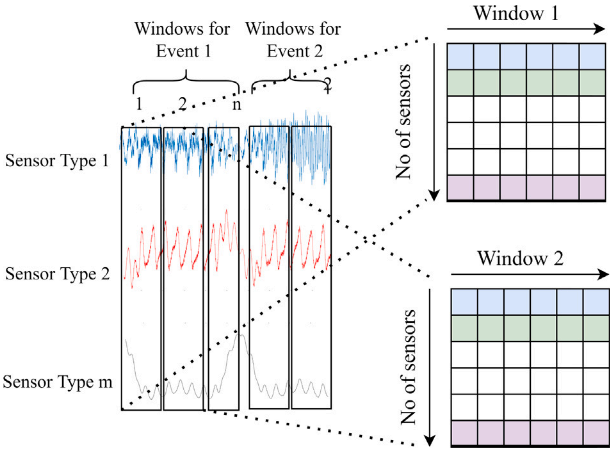


Fig. 4. Illustration of the windowing of single event.

2.5. Damage feature

To evaluate damage, the discrepancy between original and reconstructed time series signals within fixed-length windows is quantified by computing the normalised root mean square error (NRMSE) for each sensor. This calculation is performed using Eq. (4).

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_{\mu i})^2}}{\sigma_x} \tag{4}$$

where x_i is the actual value of the ith sample, and $\hat{x}_{\mu i}$ is the mean predicted value of PTAE, while σ_x is the standard deviation of the measured time series sequence of each sensor.

During the training phase, considering multiple train crossing events, the NRMSE of each window for various sensors varies between events due to fluctuating train speeds and loading conditions. Nonetheless, the NRMSE distributions for individual events can be employed to distinguish between healthy and damaged states. This can be accomplished by calculating the statistical variation between the NRMSE distributions of baseline conditions and subsequent train crossing events. To quantify the dissimilarity between two distributions, this study utilises the Kullback–Leibler (KL) divergence. The KL divergence is a technique that measures the information loss when a probability distribution p is employed to approximate a distribution q.

In this study, it is assumed that the NRMSE distributions follow a Gaussian distribution for both the baseline condition and each subsequent train crossing event. For multiple sensors, these distributions are characterised by their respective mean vectors $(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)$ and covariance matrices (Σ_1, Σ_2). The KL divergence between two multivariate Gaussian distributions can be expressed using Eq. (5).

$$D_{kl}(p \parallel q) = \frac{1}{2}\left(\mathrm{tr}(\Sigma_2^{-1}\Sigma_1) + (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \Sigma_2^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) - m + \log\frac{|\Sigma_2|}{|\Sigma_1|}\right) \tag{5}$$

where $\boldsymbol{\mu}_1$ and Σ_1 denote the mean vector and covariance matrix of the multivariate Gaussian distribution for the baseline condition, respectively. Similarly, for subsequent events, $\boldsymbol{\mu}_2$ and Σ_2 represent the mean vector and covariance matrix of the multivariate Gaussian distribution, respectively. In addition, m corresponds to the number of sensors in use, and $\mathrm{tr}(\cdot)$ signifies the trace of a matrix. From Eq. (5) it is evident that the KL divergence is exponentially related to the distance between distributions. For damage detection, the equation is mapped to a linear relationship, as shown in Eq. (6), where e is Euler number.

$$\mathrm{DF} = \ln\left[D_{kl}\left(p \parallel q\right) + e\right] - 1 \tag{6}$$

2.6. Exponentially weighted moving average

For damage assessment, it is crucial to employ a robust method capable of identifying small changes caused by structural damage. To this end, statistical process-based control charts have been used as a novelty detection method [26,27]. One such technique is the Exponentially Weighted Moving Average (EWMA), a widely used statistical method for detecting underlying patterns in time series data [28]. EWMA calculates the average of a series of data points while assigning exponentially decreasing weights to older data points. This allows EWMA to respond more quickly to recent changes in the data, making it effective for detecting small shifts in the underlying process. The EWMA process can be defined as follows:

$$Z_t = \alpha X_t + (1-\alpha)Z_{t-1} \tag{7}$$

where Z_t is the EWMA value at time t, which is typically the control variable to be monitored, X_t is the observed data point at time t, and α is the smoothing parameter, with $0 < \alpha \leq 1$. The parameter α determines the degree of weight given to the most recent data points. When α is close to 1, more weight is assigned to recent data points, whereas when α is close to 0, the weight is distributed more evenly across the entire series.

In the context of this study, the control variable is the Damage Index (DI), and the observed variable is the Damage Feature (DF) defined by Eq. (6). Therefore, we modify Eq. (7) for this study as follows:

$$DI_j = \alpha DF_j + (1-\alpha)DF_{j-1} \tag{8}$$

where j is the number of inspections or train crossing events.

For effectively assessing the structural condition, it is essential to establish a fixed threshold to monitor the Damage Index (DI_j). In the context of the EWMA control chart, two thresholds, known as the Upper Control Limit (UCL) and Lower Control Limit (LCL), are utilised to determine whether the process is in control or not. The UCL and LCL are defined as follows:

$$\mathrm{UCL} = \mu_0 + L\sigma\sqrt{\frac{\alpha}{(2-\alpha)}\left[1-(1-\alpha)^{2j}\right]} \tag{9}$$

$$\mathrm{LCL} = \mu_0 - L\sigma\sqrt{\frac{\alpha}{(2-\alpha)}\left[1-(1-\alpha)^{2j}\right]} \tag{10}$$

In these equations, μ_0 represents the average value of the EWMA statistics when the process is in control, while σ denotes the standard deviation of the Damage Feature (DF) values. The parameter L is a constant that determines the width of the control chart.

These limits allow for the identification of anomalies or shifts in the process. If the Damage Index (DI_j) lies within the UCL and LCL, the process is considered to be in control, indicating that the structural condition remains unchanged. However, if the DI_j exceeds the UCL or falls below the LCL, it signals a potential anomaly or change in the structural condition that warrants further investigation.

3. Numerical case study

3.1. Model description

This section presents the numerical model used to validate the strategies and techniques proposed in this study. The numerical model integrates the behaviour of the train, ballasted track, and bridge, as depicted in Fig. 5. A 2D representation is utilised for the entire problem, an approach that has been widely adopted by researchers for analogous studies. Each modelling component is briefly discussed next.

Firstly, the train is modelled as a sequence of individual vehicles, each characterised by a multibody system with a six-degree-of-freedom configuration. A primary suspension connects the two axles of each bogie, while a secondary suspension system supports the main body on two bogies. This vehicle model has been extensively employed in the literature and effectively captures the primary components. Rail irregularities, which serve as a significant source of excitation, contribute to the combined dynamic response of the vehicle and infrastructure [29]. In this study, Class 6 track irregularities from the Federal Railroad Administration are employed [30].

The track model comprises a combination of elements, as illustrated in Fig. 5, including rail, pad, sleeper, ballast, and sub-ballast. The rail is modelled as a beam, with the remaining components represented as lumped masses. Dashpot and spring systems interconnect these elements, exhibiting the viscoelastic behaviour of each component. The bridge is represented as an Euler–Bernoulli beam using a Finite Element Model (FEM) with beam elements, each consisting of two nodes and two degrees of freedom per node. he bridge's specifications are based on a simply supported concrete bridge, as reported in [31]. The bridge has a length of $le = 50$ m, a second moment of area $I = 51.3$ m^4, a mass per unit length $\rho = 69000$ kg/m, and a modulus of elasticity $E = 3.5 \times 10^{10}$ N/m.

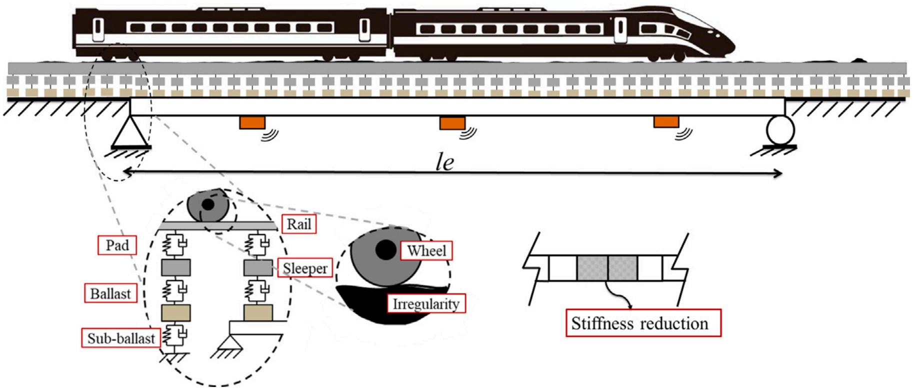


Fig. 5. Overview of the Train-Track-Bridge model.

Table 1
Mean and standard deviation values for the modelling of the effect of temperature on concrete's elastic modulus.

	Q	S	κ	τ	R
μ	1.0129	−0.0048	0.1977	3.1466	0.1977
σ	0.003	0.0001	0.0027	0.0861	0.0027

The subsystems are integrated into a unified system, with the equations of motion for the train, track, and bridge coupled together through the mass, stiffness, damping, and force matrices, These matrices are time-varying and change according to the vehicle's position. The time-varying equations of motion are solved using numerical integration with the Newmark-β scheme, yielding the dynamic response of the train as it traverses the bridge. The reader can refer to [32] for a complete description and more detail of the coupling procedure and its validation.

In this study, the vehicle configuration of the ICE3 Velaro train is considered, comprising a convoy of eight wagons. The mechanical properties and dimensions are provided in Gia et al. [33]. To establish the dynamic equilibrium of the vehicle before it enters the bridge, a track 100 m longer than the bridge is modelled. The rail is designed as a standard UIC60 rail with a sleeper spacing of 0.6 m. The Matlab implementation of the train-track model for numerical simulation is available at [29]

3.2. Modelling of damage and temperature effect

For damage assessment using the train-track model, damage is simulated by implementing a localised stiffness reduction in the beam elements. The bridge is discretised into 160 elements, resulting in elements with a length of 0.3 m. Damage is modelled at the mid-span of the bridge by reducing the stiffness of two elements, which is approximately equal to 1.2% of the total bridge length.

Another critical aspect of long-term bridge monitoring is accounting for the influence of temperature, as it directly affects the bridge's material properties and, consequently, its modal properties. In the present study, the temperature's effect is modelled using an empirical model derived from real bridge measurements, which establishes the relationship between temperature and bridge properties. One such model proposes a bi-linear equation for the bridge's elastic modulus [34], as shown in Eq. (11).

$$E_T = E_0\left[Q + S, T + R\left(1 - erf\left(\frac{T-\kappa}{\tau}\right)\right)\right] \tag{11}$$

In Eq. (11), the temperature-dependent elastic modulus is denoted by E_T, while E_0 represents its value at a specific reference temperature. The parameters Q and S establish the linear relationship, and the expression $R\left(1 - erf\left(\frac{T-\kappa}{\tau}\right)\right)$ modifies the relationship to account for sub-zero temperatures. In Eq. (11), the temperature is expressed in degrees Celsius as T, with κ and τ serving as parameters that dictate the transition around the freezing point.

In this study, the effect of temperature is assessed using a 1-year dataset of temperature records from a weather station in Trondheim, Norway. In the numerical model, temperatures for each train crossing event are randomly chosen from these records. The elastic modulus of the concrete is then adjusted accordingly with Eq. (11). To accommodate possible uncertainties, parameters within this equation are also randomly sampled, following the suggestion in [34], based on the mean and standard deviation values listed in Table 1. Fig. 6 illustrates the influence of fluctuating temperature on the first four modes of the bridge, both in its healthy condition and with damage introduced. All modes exhibit a bi-linear relationship with varying temperature.

3.3. Data generation

The evaluation of the proposed damage assessment method is carried out using numerically simulated data generated by solving the Train-Track-Bridge interaction model presented in Section 3.1. The dataset is created by assuming that the train speed varies for

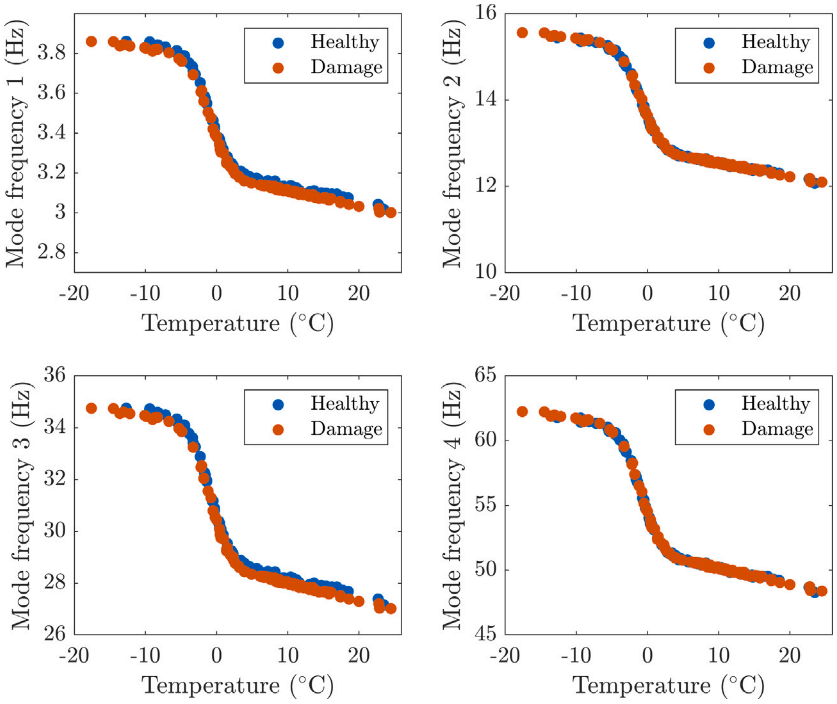


Fig. 6. Effect of temperature on modes of the bridge.

Table 2
Train model parameters variability.

	Min.	Max.	Mean	SD
Velocity (km/h)	150	170	160	3
Body mass (kg)	42 100	53 500	47 800	500

each crossing, while the vehicle's suspension properties remain constant. Small variations in the body mass of each train wagon are also considered to account for the inherent uncertainty in the payload. The varying train parameters and the statistical variability of these parameters (i.e., maximum, minimum, and standard deviation) are presented in Table 2. To generate the dataset, these parameters are randomly sampled based on the given statistical variation within a Monte Carlo simulation. For the healthy case, a batch of 200 events is created. For each event, displacement, rotation, and acceleration responses are measured at three locations on the beam, as shown in Fig. 5, with a sampling frequency of 1000 Hz. To account for a realistic scenario for each event, noise levels are randomly sampled from a Gaussian distribution $\mathcal{N}(30\text{ dB}, 1\text{ dB})$ with values in the range [30 dB, 40 dB] signal-to-noise ratio, and added to all measured responses.

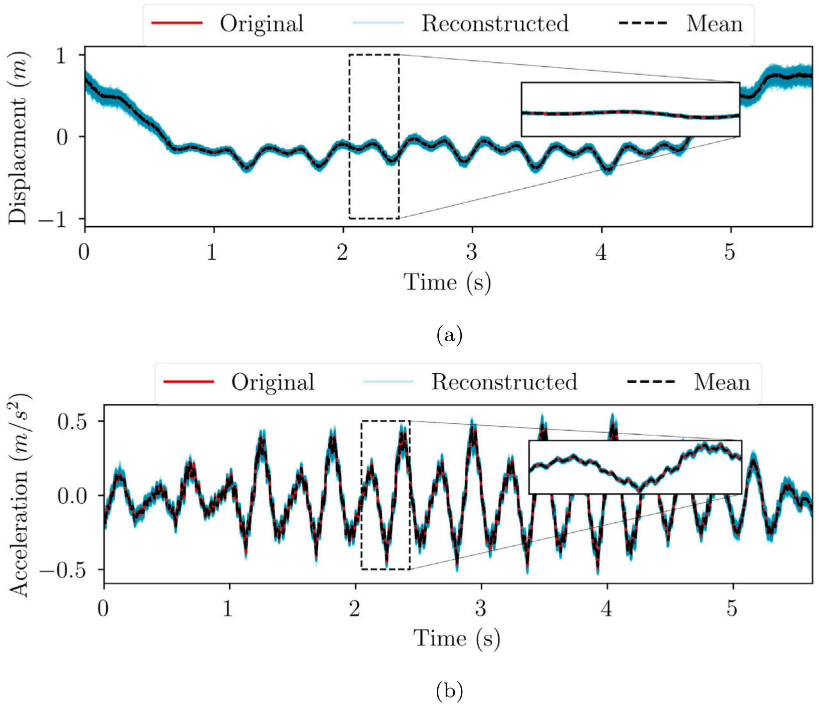
To demonstrate the performance of the proposed method, this study considers four different scenarios. Each scenario is defined based on the available sensor information:

- Scenario-1: The data contains only displacement responses from 3 locations on the bridge.
- Scenario-2: The data contains only rotation responses from 3 locations on the bridge.
- Scenario-3: The data contains only acceleration responses from 3 locations on the bridge.
- Scenario-4: The data contains all responses, including acceleration, rotation, and displacement from 3 locations on the bridge.

For PTAE input in each scenario, all measured signals are standardised to have zero mean and unit variance. Each event is then sliced into windows of length 128, as discussed in Section 2.4, resulting in a dataset shape of $(N \times l \times m)$, where N is the number of windows in a single event, l is the length of the window, and m is the number of variables or sensors. For the 200 events, considering only acceleration responses at three different bridge locations, the size of the dataset becomes $(8847 \times 128 \times 3)$.

3.4. Training and evaluation method

The PTAE architecture is built using TensorFlow modules, and the code is developed in Python 3.9. The selected network architecture is based on achieving the lowest reconstruction loss after carrying out a random search for hyperparameter optimisation. The encoder module of the PTAE comprises three convolutional blocks, followed by two LSTM layers. Each convolutional block



(a)

(b)

Fig. 7. Comparison of original and reconstructed signals of two different type of sensors; (a) Displacement signal at mid-span of the bridge, (b) Acceleration signal at mid-span of the bridge.

consists of a 1D convolutional layer, Leaky-ReLU activation function, max-pooling layer and an MC-Dropout layer. The decoder utilises the same number of convolutional blocks arranged in reverse order, with the max-pooling layer replaced by an up-sampling layer. An MC-Dropout rate of 0.5 is chosen, meaning that during each forward pass, 50% of the connections are dropped, enhancing the model's robustness and preventing overfitting. Table 3 displays the detailed architecture of the proposed PTAE, including various hyperparameters such as activation function, filters and kernel size, and dropout rate.

The training process employs adaptive moment estimation (Adam) with a batch size of 128 samples, while setting learning and decay rates at 0.001 and 0.0001 respectively. The model undergoes training for 1500 epochs. All training and numerical computations for the models are conducted on a standard PC fitted with an Intel Core i9-10900 K CPU, 64 GB RAM and an NVIDIA GTX 2080Ti graphics card.

The proposed PTAE architecture is trained separately for all four scenarios, maintaining the same network architecture as presented in Table 3, with $m = 3$ and the size of the latent space dimension $z = 1$. However, for Scenario-4, the value of m increases to 9 and the dimension of the latent space to $z = 3$. Each model is trained using only the healthy bridge data. For training, 70 train crossing events are selected from the available dataset of 200 events. To visualise the model's reconstruction performance, two random cases from the validation dataset are illustrated in Fig. 7. These cases are reconstructed using the PTAE model trained for Scenario-4. The trained model undergoes 50 simulations, with each layer's 50% connection randomly pruned, as discussed in Section 2.2. Fig. 7 displays the reconstructed output for each forward pass and compares the mean value of all forward passes to the original input signal. The zoomed-in section of the figure reveals an almost perfect match between the input signals and the mean of the reconstructed signals. The results demonstrate the PTAE-trained model's ability to replicate the bridge's healthy state.

3.5. Damage detection using PTAE

In order to demonstrate the performance of PTAE, and how it can be utilised for damage detection, four new datasets with different damage severities are simulated, where damage is modelled as a stiffness reduction as discussed in Section 3.2. Each dataset consists of 100 train crossing events. The details for different damage cases are:

- DC0: Healthy case
- DC1: Damage at mid-span with 10% stiffness reduction
- DC2: Damage at mid-span with 20% stiffness reduction
- DC3: Damage at mid-span with 30% stiffness reduction

Table 3
Architecture of proposed PTAE.

Layers	Output shape	Kernel size	Activation
Encoder			
Input	(128 × m)	–	–
Conv_1D	(128 × 512)	1 × 5	Leaky-ReLU
Max-pooling	(64 × 512)	1 × 5	–
MC-Dropout	(64 × 512)	–	–
Conv_1D	(64 × 256)	1 × 5	Leaky-ReLU
Max-pooling	(64 × 256)	1 × 5	–
MC-Dropout	(64 × 256)	–	–
Conv_1D	(64 × 128)	1 × 3	Leaky-ReLU
Max-pooling	(64 × 128)	1 × 3	–
MC-Dropout	(64 × 128)	–	–
LSTM	(64 × 128)	–	Leaky-ReLU
LSTM	(64 × z)	–	Leaky-ReLU
Decoder			
TimeDistributed	(64 × 128)	–	Leaky-ReLU
Up-sampling	(64 × 128)	1 × 5	Leaky-ReLU
Conv_1D	(64 × 128)	1 × 5	–
MC-Dropout	(64 × 128)	–	–
Up-sampling	(128 × 128)	1 × 5	Leaky-ReLU
Conv_1D	(128 × 256)	1 × 5	–
MC-Dropout	(128 × 256)	–	–
Up-sampling	(128 × 256)	1 × 5	Leaky-ReLU
Conv_1D	(128 × 512)	1 × 5	–
MC-Dropout	(128 × 512)	–	–
Conv_1D	(128 × m)	–	Linear

[m] Number of sensors. [z] Size of latent space dimension.

To demonstrate the capability of the proposed method for damage detection, for each damage case, NRMSE is computed for each window sequence to quantify the discrepancies between the original and the reconstructed sequence as discussed in Section 2.5. Similarly, NRMSE is also computed for the window sequence of the training dataset. The mean vector ($\boldsymbol{\mu}_1$) and covariance matrices (Σ_1) of multiple events of the training dataset establish a baseline condition. For subsequent inspections, the distribution of NRMSE of each single crossing event with mean vector ($\boldsymbol{\mu}_2$) and covariance matrices (Σ_2) is used to compute the variation in distribution using Eq. (6). The damage feature (DF) is computed for all four scenarios. Here it is important to mention that the damage index based on EWMA proposed in this paper is sensitive to outliers, so it is important to carefully analyse the computed damage feature and if there are outliers that need to be taken care of with the appropriate method.

3.5.1. Outlier removal using Tukey's method

Tukey's method is a robust technique employed to detect outliers in a dataset [35]. This approach is based on the interquartile range (IQR), which is the difference between the first quartile (Q_1) and the third quartile (Q_3) of a dataset. The (IQR) represents the central 50% of the data, and Tukey's method identifies data points outside this range as potential outliers. When using a fixed window size, this method can be applied to detect outliers within each window of the dataset. In this method, the lower and upper bounds for outliers are determined using $Q_1 - I \times \text{IQR}$ and $Q_3 - I \times \text{IQR}$, respectively, where I is a constant typically ranging from 1.5 to 3. A larger value of I will make the method less sensitive, identifying only the most extreme values as outliers. On the other hand, a smaller value of I increases the sensitivity of the method, flagging more points as outliers, including those that deviate moderately from the median. This could be useful when a more conservative approach is necessary, and even slight deviations are considered significant. In practice, selecting an appropriate value of I requires a careful consideration of the specific context, the data distribution, and the goals of the analysis. This method is advantageous for outlier detection because it is less sensitive to extreme values compared to methods based on the mean and standard deviation. By employing a fixed window size, this approach facilitates the detection and removal of local outliers within each window, which is particularly useful when dealing with non-stationary or heterogeneous datasets.

To compute the damage index using Eq. (8) for all scenarios, the outlier removal method is utilised. To illustrate how the outlier removal method is employed, the damage feature (DF) for Scenario-1 is computed where displacement sensors are used. Outlier analysis is performed considering a non-overlapping window size of 20. For each window, Tukey's method is applied with the value of $I = 3$ as discussed previously, and outliers are marked in red as shown in Fig. 8. The figure demonstrates that the outlier removal method effectively removes outliers with progressively varying damage features due to increasing damage severity.

3.6. Results

In the numerical case study, all four scenarios mentioned in the previous section are examined. Each scenario involves separately training PTAE and computing damage features for each trained model. The outlier removal method discussed earlier is applied

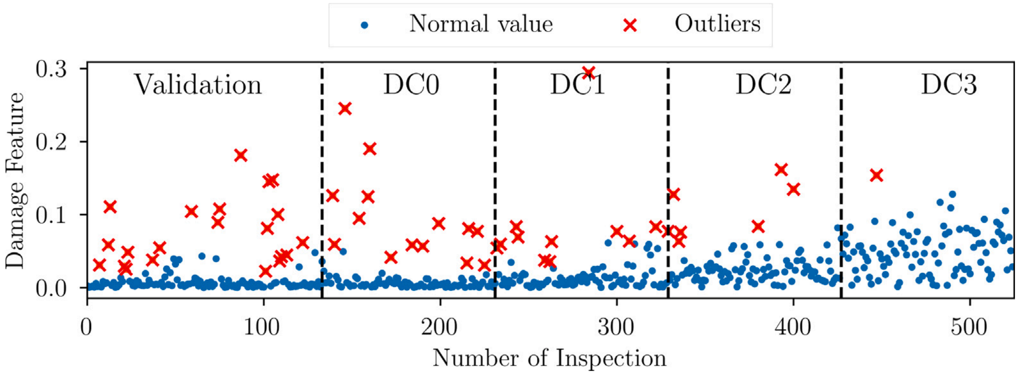


Fig. 8. Removal of outliers using Tukey's method.

before utilising EWMA to calculate the damage index. To compute the damage index using EWMA, the value of α is set to 0.02, while $L = 4$. The LCL and UCL are determined using the validation dataset according to Eqs. (10) and (9) respectively. These values will be employed to establish the threshold for differentiating between the healthy and damaged conditions of the bridge. For comparison, the threshold-based system is also examined alongside the commonly used one-class support vector machine (OC-SVM) anomaly detector. The OC-SVM is implemented with an RBF kernel function and regularisation parameters $\nu = 0.1$ and $\gamma = 10$, which define the decision boundary's shape. The OC-SVM is trained using damage features from the validation dataset, similar to the threshold selection in the EWMA-based damage index.

To demonstrate the performance of the proposed method, the results of all four scenarios for different damage cases are illustrated in Fig. 9. In the figure, the black dashed lines represent the UCL and LCL thresholds. For DC0 in all scenarios, the majority of the damage index values fall within the thresholds, confirming the bridge's unaltered healthy state. Similarly, OC-SVM accurately predicts DC0 as the healthy condition of the bridge in all scenarios. As the severity of damage increases from DC1 to DC3, the damage index proportionally rises. These results demonstrate the sensitivity of the proposed damage index to damage severity in all scenarios, unlike OC-SVM, which only functions as an anomaly detector.

The results in Fig. 9 enable also a direct comparison of the method's performance based on various measured load effects. The results indicate that Scenario-1, which employs displacement signals, outperforms the other scenarios (Fig. 9*a*). This is because displacements are more sensitive to structural damage than acceleration or other load effects. Rotation measurements offer also insights into a structure's global behaviour by tracking changes in the overall structural response. However, this load effect is generally less sensitive to minor, localised damage and may not effectively capture higher mode shapes [8]. This is evident in the results presented in Fig. 9*b*, where some events of DC1 are misclassified as healthy bridge conditions. The results (Fig. 9*d*) also highlight that integrating multi-sensor information can improve damage sensitivity across different damage cases. Thus, our proposed damage assessment strategy proves to be effective, irrespective of the sensor type utilised. It is crucial to clarify that this study does not aim to compare and determine which sensor type or combination is superior. Instead, the focus is on demonstrating that the proposed method can be effectively employed with any sensor type.

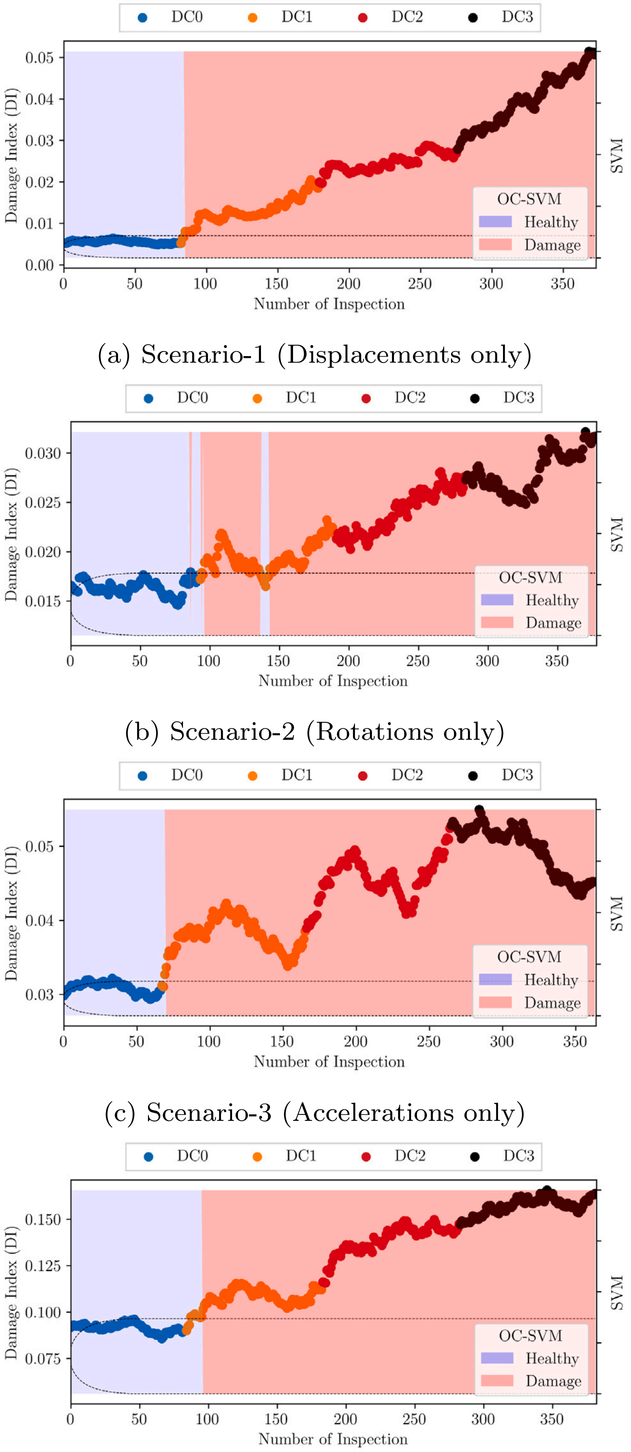
4. KW51 bridge case study

4.1. Introduction

The experimental validation of the proposed method considers the signals recorded on a steel bowstring bridge in Leuven, Belgium, known as KW51 railway bridge. This bridge spans 115 m and is 12.4 m wide, with two separate ballasted tracks, designated as track A and track B, located on the north and south sides respectively. Both tracks have curved horizontal alignment, and a speed limit of 160 km/h is enforced for passenger trains (see Fig. 10).

The monitoring system was installed on the bridge in September 2018 with a configuration as shown in Fig. 11. Various sensing devices were placed on the structure, including accelerometers on the bridge deck in both lateral and vertical directions, as well as accelerometers on the arch in the lateral direction. Additionally, strain sensors were installed on the bridge deck along the longitudinal direction. A National Instruments data acquisition system collected the measurements, which also included temperature and relative humidity measurements taken beneath the bridge deck. This study only considers the acceleration measurements from the bridge deck, because strain signals for multiple train crossing are missing. For a comprehensive description of the monitoring system and data, readers are directed to [37].

During the monitoring period, the bridge underwent retrofitting to address a construction error. This process involved reinforcing the connections between the diagonals, arches, and bridge deck. The retrofitting occurred between 15 May and 27 September. Therefore, there exist measurements from three distinct time periods: the original bridge before the retrofit (7.5 months); during the works of retrofit installation (4.5 months); and for the strengthened bridge with the operational retrofit (3.5 months).



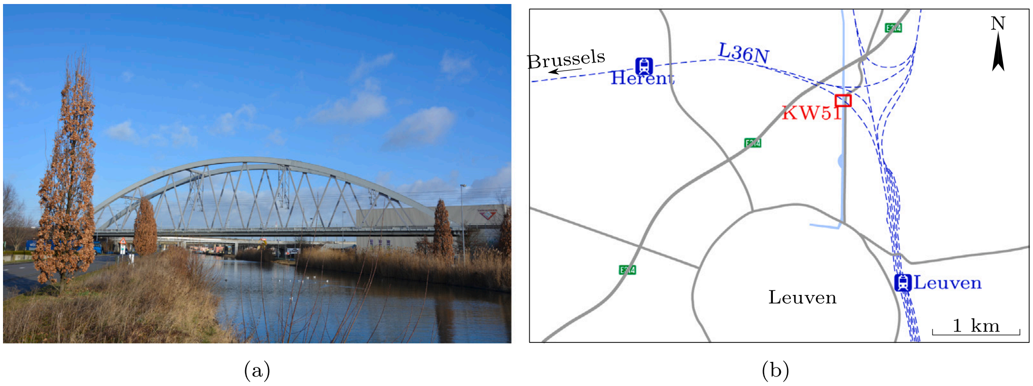
(a) Scenario-1 (Displacements only)

(b) Scenario-2 (Rotations only)

(c) Scenario-3 (Accelerations only)

(d) Scenario-4 (All responses)

Fig. 9. Evolution of damage index during progressive bridge condition change for different scenarios. Dashed line indicate the UCL and LCL threshold values.



(a) (b)

Fig. 10. (a) KW51 railway bridge in Leuven, Belgium (Image by Kristof Maes) [36]; (b) situation sketch [36].

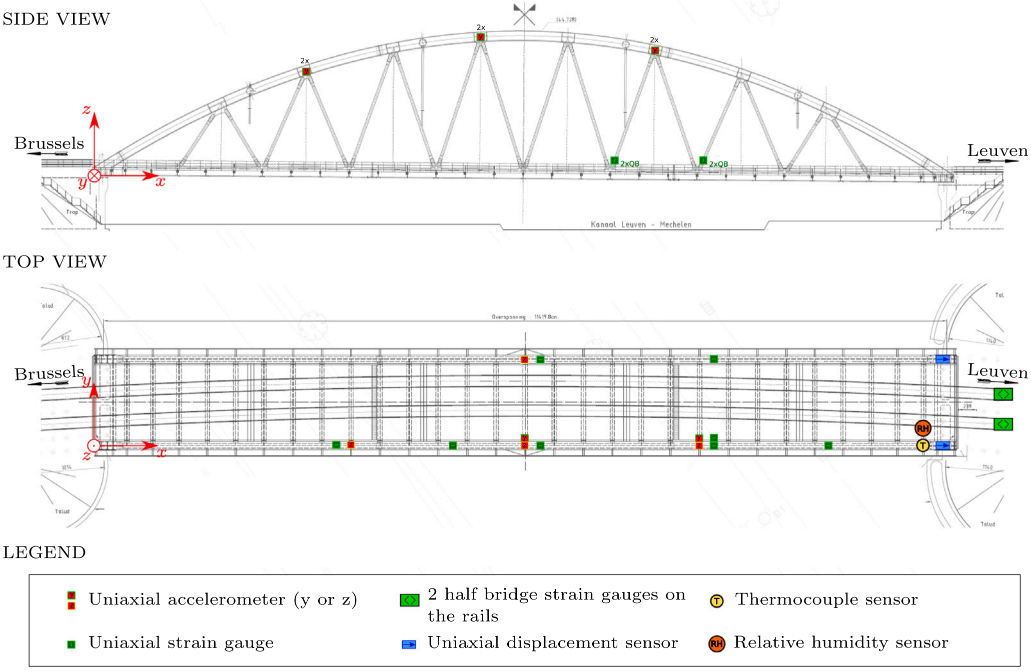


Fig. 11. Sketch of KW51 bridge with an overview of the measurement setup. Sensors installed on the diagonals and the arches are shown in the side view and sensors installed on the bridge deck are shown in the top view [36].

4.2. Data preparation and training

To validate the proposed method, only acceleration signals during train passages are considered. Each day the bridge was in operation, two train passages were recorded. For each passage, vibration data was collected from 10 s before the train entered the bridge until 30 s after the train left. To focus on data collected while the train was on the bridge, a 10-s time window was removed. For each sensor, the moving root-mean-square (RMS) value of acceleration was calculated for a 1-s interval, and when that value fell below the 0.05 m/s^2 threshold, the remaining signal was discarded. The acceleration signals had a sampling frequency of 825.8065 Hz. In this study, six acceleration sensors were used, including four in the vertical direction and two in the lateral direction, as shown in Fig. 11.

For the PTAE training, 284 train crossing events were considered before the retrofitting, and 130 train passage events were identified after the retrofitting for damage detection cases. No train passages were taken into account during the retrofitting stage, as Fig. 12a clearly shows significant changes in bridge modal frequencies. Any existing damage detection method would identify the change in structural behaviour. This period of the monitoring campaign is not representative of typical damage. On the other hand, after the bridge is fully strengthened (retrofitted) the structural response can be considered as “positive” damage. a change

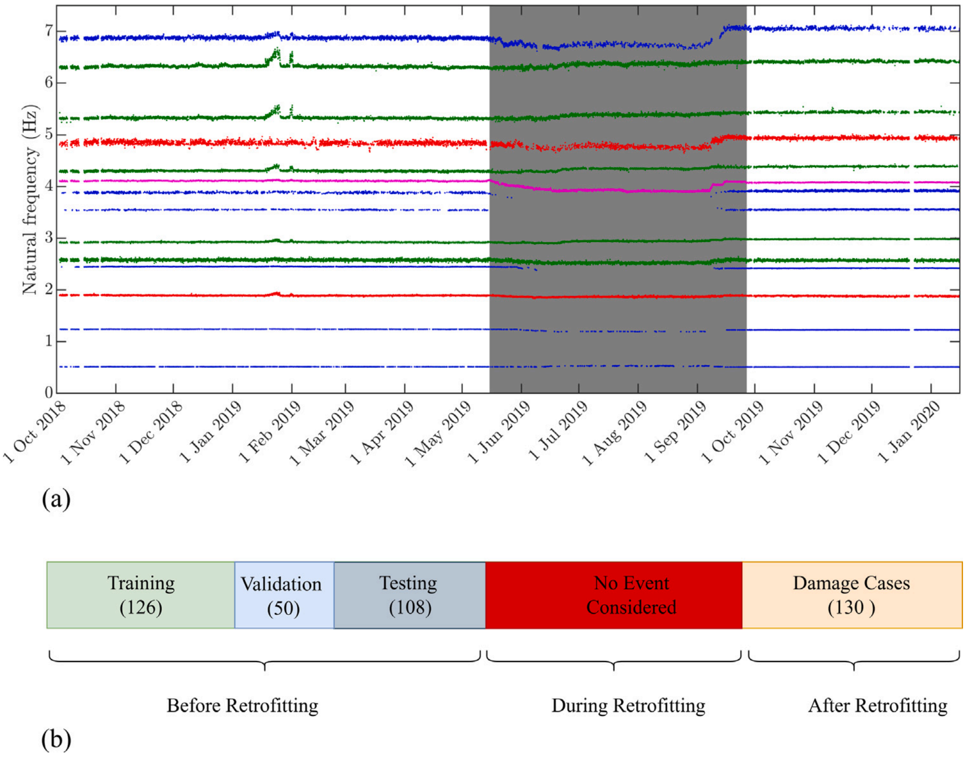


Fig. 12. (a) Identified frequencies of the KW51 bridge, with the shaded area representing the time period during which the installation of the retrofitting was carried out; The colour blue corresponds to the lateral modes of the arches, while red represents the lateral modes of the bridge deck. The colour green denotes global vertical modes, and magenta indicates global torsional modes; (b) Visualisation of dataset used for assessing bridge conditions. In brackets, number of train crossing events.

in the structure. The train crossing events before retrofitting were further divided for training, validation, and testing, as shown in Fig. 12*b*.

A pre-processing step was implemented to slice the signal into fixed window lengths of 128 and 256 samples, using the procedure discussed in Section 2.4. Two separate PTAE models were trained with the same architecture as described in Section 3.4, with $m = 6$ and $z = 1$. For the model using a window size of 256, the input shape became $(256 \times m)$, and the architecture's shape was adjusted accordingly. During training and subsequent testing, 50% random connection pruning was performed for each forward pass. The trained PTAE network was tested with 50 simulations for the same input. Fig. 13 displays the reconstructed signals and their respective mean responses of randomly chosen events from the testing dataset. The zoomed-in sections of each signal show that the mean value of multiple passes aligns well with the original input signal. These results were expected since the model was trained and tested with a dataset from the same bridge conditions (before retrofitting).

4.3. Results

This section evaluates the PTAE model's ability to detect changes in the bridge's condition (after retrofitting). The trained PTAE is employed to reconstruct the input frames for training, validation, and testing data. The NRMSE is calculated for each window frame. The distribution of computed NRMSE for the training dataset of all acceleration sensors is utilised to compute the mean ($\boldsymbol{\mu}_1$) and covariance matrices (Σ_1), representing the baseline condition. Likewise, for each subsequent event, the statistics (mean ($\boldsymbol{\mu}_2$) and covariance matrices (Σ_2)) of NRMSE are calculated and used in Eq. (6) to compute the damage feature. Before applying the EWMA-based damage index, outliers are removed from the dataset using the method discussed in Section 3.5.1. The EWMA is applied with the value of $\alpha = 0.02$ and $L = 4$ to compute the damage index using Eq. (8). The threshold is calculated using the damage index data computed for the validation events by employing the UCL (Eq. (9)) and LCL (Eq. (10)) expressions. Similar to the numerical case study, OC-SVM with the same parameters is also trained with the validation dataset and subsequently used to differentiate between the two bridge conditions.

Fig. 14 demonstrates how to track the bridge condition's evolution using train crossing events. A threshold line is established, and significant deviation from this line indicates a change in the structure's condition. Fig. 14*a* and *b* display the results of the bridge's condition assessment using two different window lengths. The effect of retrofitting is clearly depicted by the jump in the damage index at the end of the time period when no retrofitting had been applied. Likewise, OC-SVM can effectively differentiate between

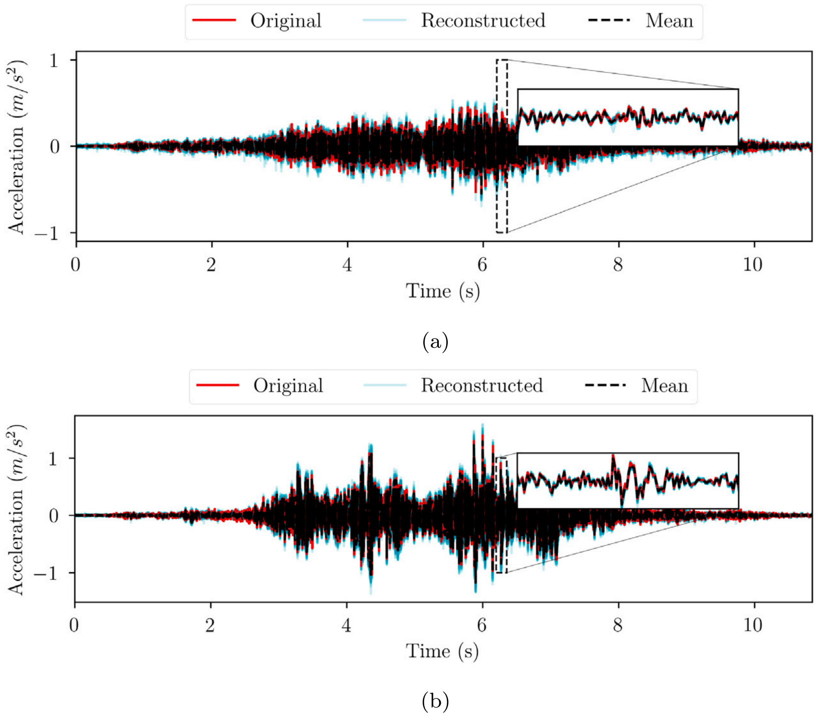


Fig. 13. Comparison between original and reconstructed signals from two sensors; (a) Vertical acceleration signal at mid-span of the bridge (Track A, North side); (b) Vertical acceleration signal at mid-span of the bridge (Track B, South side).

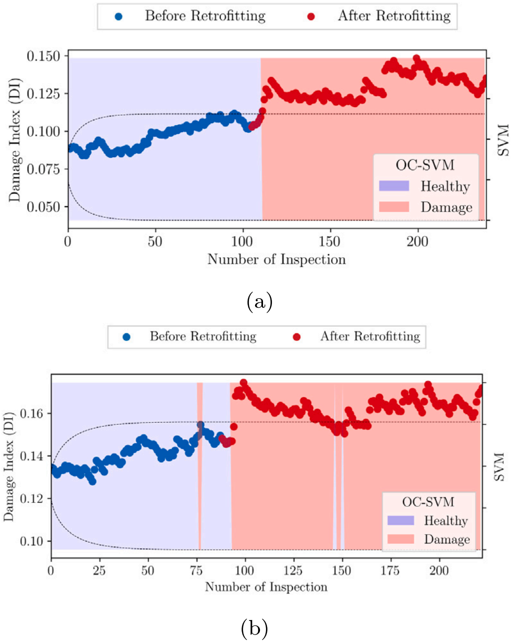


Fig. 14. Evolution of damage index over periods of the monitoring campaign. The dashed lines indicate the UCL and LCL threshold values. (a) PTAE is trained with window size 128; (b) PTAE is trained with window size 256.

the two bridge conditions, particularly when the model is trained using a length of 128, as depicted in Fig. 14*a*. In instances where the window length was 256, both the proposed method and SVM misclassified a few train crossing events; however, the overall result remains within a satisfactory performance scope.

In summary, the analysis of the results showcases the effectiveness of the proposed PTAE model-based damage index in detecting changes in the bridge's condition, particularly when the extent of damage or alteration in condition is minimal. The proposed method efficiently handles multi-sensor data from train crossing events, requiring minimal signal pre-processing and eliminating the need for individual sensor data training, a requirement in some previous studies [19,36,38]. The method presents a reliable and efficient means for assessing the bridge's condition by solely utilising train-induced responses, while employing different window lengths and monitoring the damage index. It is important to note that the proposed method exhibits considerable robustness against environmental fluctuations, even without implementing specific data pre-processing methods to eliminate the effects of varying environmental and operational conditions. This study sets the stage for further advancements in structural health monitoring and provides valuable insights for implementing bridge monitoring systems using vehicle-induced responses.

5. Conclusions

This paper presents an innovative methodology for assessing bridge conditions using a probabilistic temporal autoencoder (PTAE). By gathering multi-sensor data during train crossings and employing a PTAE comprising multiple convolutional blocks and an LSTM recurrent neural network, the framework effectively extracts features and captures temporal relationships in the data. Through the computation of the reconstruction loss and KL divergence-based damage features, the methodology enables the detection of potential damage in the bridge structure. The application of an Exponentially Weighted Moving Average (EWMA) filter and a control chart-based threshold mechanism further refines the damage assessment process, allowing for differentiation between healthy and subsequent progressive damage cases. The effectiveness of the proposed method was evaluated using numerically generated data, where a train crossed a simply supported bridge under realistic speed and loading conditions. Similarly, the methodology was validated by applying it to data obtained from the KW51 bridge to detect different bridge conditions. The main findings and advantages of this study can be summarised as follows:

- The proposed methodology accommodates multi-sensor data, such as accelerations, displacements, and rotations, making it adaptable to various monitoring scenarios and sensor configurations. This also allows for the avoidance of manual feature extraction or single sensor level model training for similar tasks.
- The probabilistic nature of the autoencoder enhances robustness for varying operational and environmental conditions and enables the quantification of uncertainty in predictions under these conditions.
- The Exponentially Weighted Moving Average (EWMA) control chart-based threshold mechanism facilitates the identification of subtle changes in the bridge's condition, allowing for early damage detection and timely maintenance interventions.
- The case study results indicate that displacement measurements are particularly robust in identifying damage. However, it was also observed that acceleration responses yielded promising outcomes in detecting structural damage. This is further validated in the real case study where the proposed methodology is able to detect damage with 100% accuracy. This underscores the practical viability of the proposed methodology in operational scenarios of structural health monitoring.
- The results of this study suggest that detecting and quantifying different types of damage is possible even without employing any pre-processing method to remove the effects of operational and environmental conditions.
- By incorporating model uncertainty and providing accurate damage assessments, the methodology supports better decision-making in maintenance planning and resource allocation.
- The proposed method can be easily integrated with existing monitoring systems and data collection platforms, enabling seamless adoption and implementation in various contexts.

In conclusion, this methodology holds significant potential for the future of bridge health monitoring and maintenance, offering a data-driven solution that can enhance the safety of infrastructure.

CRediT authorship contribution statement

Muhammad Zohaib Sarwar: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft. **Daniel Cantero:** Conceptualization, Funding acquisition, Investigation, Resources, Software, Supervision, Validation, Writing – review & editing.

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.

Data availability

Data will be made available on request.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this manuscript, the authors used ChatGPT and Grammarly in Sections 1 and 2 to improve the readability and grammar of the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Acknowledgement

The authors would like to acknowledge Kristof Maes and Geert Lombaert, members of the Structural Mechanics Section at KU Leuven, for providing the measurement data for the KW51 bridge.

References

[1] B.F. Spencer Jr., V. Hoskere, Y. Narazaki, Advances in computer vision-based civil infrastructure inspection and monitoring, Engineering 5 (2) (2019) 199–222.
[2] H. Nick, A. Aziminejad, Vibration-based damage identification in steel girder bridges using artificial neural network under noisy conditions, J. Nondestruct. Eval. 40 (2021) 1–22.
[3] Z. Lingxin, S. Junkai, Z. Baijie, A review of the research and application of deep learning-based computer vision in structural damage detection, Earthq. Eng. Eng. Vib. 21 (1) (2022) 1–21.
[4] Y. An, E. Chatzi, S.-H. Sim, S. Laflamme, B. Blachowski, J. Ou, Recent progress and future trends on damage identification methods for bridge structures, Struct. Control Health Monit. 26 (10) (2019) e2416.
[5] Y. Zhang, K.-V. Yuen, Review of artificial intelligence-based bridge damage detection, Adv. Mech. Eng. 14 (9) (2022) 16878132221122770.
[6] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, M. Gabbouj, D.J. Inman, A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications, Mech. Syst. Signal Process. 147 (2021) 107077.
[7] M.M. Alamdari, K. Kildashti, B. Samali, H.V. Goudarzi, Damage diagnosis in bridge structures using rotation influence line: Validation on a cable-stayed bridge, Eng. Struct. 185 (2019) 1–14.
[8] F. Huseynov, C. Kim, E.J. Obrien, J. Brownjohn, D. Hester, K. Chang, Bridge damage detection using rotation measurements–Experimental validation, Mech. Syst. Signal Process. 135 (2020) 106380.
[9] E.J. Obrien, J. Brownjohn, D. Hester, F. Huseynov, M. Casero, Identifying damage on a bridge using rotation-based Bridge Weigh-In-Motion, J. Civ. Struct. Health Monit. 11 (2021) 175–188.
[10] S. Quqa, L. Landi, P.P. Diotallevi, Automatic identification of dense damage-sensitive features in civil infrastructure using sparse sensor networks, Autom. Constr. 128 (2021) 103740.
[11] S. Quqa, L. Landi, P.P. Diotallevi, Instantaneous identification of densely instrumented structures using line topology sensor networks, Struct. Control Health Monit. 29 (3) (2022) e2891.
[12] H. Salehi, R. Burgueño, Emerging artificial intelligence methods in structural engineering, Eng. Struct. 171 (2018) 170–189.
[13] K.A. Eltouny, X. Liang, Large-scale structural health monitoring using composite recurrent neural networks and grid environments, Comput.-Aided Civ. Infrastruct. Eng. 38 (3) (2023) 271–287.
[14] C. Rainieri, F. Magalhaes, D. Gargaro, G. Fabbrocino, A. Cunha, Predicting the variability of natural frequencies and its causes by Second-Order Blind Identification, Struct. Health Monit. 18 (2) (2019) 486–507.
[15] Y. Bao, H. Li, Machine learning paradigm for structural health monitoring, Struct. Health Monit. 20 (4) (2021) 1353–1372.
[16] M.Z. Sarwar, D. Cantero, Vehicle assisted bridge damage assessment using probabilistic deep learning, Measurement 206 (2023) 112216.
[17] Y.-W. Wang, Y.-Q. Ni, S.-M. Wang, Structural health monitoring of railway bridges using innovative sensing technologies and machine learning algorithms: a concise review, Intell. Transp. Infrastruct. 1 (2022).
[18] I. Gonzalez, R. Karoumi, BWIM aided damage detection in bridges using machine learning, J. Civ. Struct. Health Monit. 5 (2015) 715–725.
[19] A.C. Neves, I. González, R. Karoumi, Development and validation of a data-based SHM method for railway bridges, in: Structural Health Monitoring Based on Data Science Techniques, Springer, 2022, pp. 95–116.
[20] M.R. Azim, M. Gül, Damage detection of steel girder railway bridges utilizing operational vibration response, Struct. Control Health Monit. 26 (11) (2019) e2447.
[21] A. Meixedo, J. Santos, D. Ribeiro, R. Calçada, M. Todd, Damage detection in railway bridges using traffic-induced dynamic responses, Eng. Struct. 238 (2021) 112189.
[22] Y. Gal, Z. Ghahramani, Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: International Conference on Machine Learning, PMLR, 2016, pp. 1050–1059.
[23] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, in: International Conference on Machine Learning, Pmlr, 2013, pp. 1310–1318.
[24] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[25] N.S. Madiraju, Deep Temporal Clustering: Fully Unsupervised Learning of Time-Domain Features (Ph.D. thesis), Arizona State University, 2018.
[26] O. Ibidunmoye, A.-R. Rezaie, E. Elmroth, Adaptive anomaly detection in performance metric streams, IEEE Trans. Netw. Serv. Manag. 15 (1) (2017) 217–231.
[27] Z. Shang, L. Sun, Y. Xia, W. Zhang, Vibration-based damage detection for bridges by deep convolutional denoising autoencoder, Struct. Health Monit. 20 (4) (2021) 1880–1903.
[28] A.S. Neubauer, The EWMA control chart: properties and comparison with other quality-control procedures by computer simulation, Clin. Chem. 43 (4) (1997) 594–601.
[29] D. Cantero, TTB-2D: Train–Track–Bridge interaction simulation tool for Matlab, SoftwareX 20 (2022) 101253.
[30] L. Fryba, W. Courage, Dynamics of railway bridges, Meccanica 32 (1) (1997) 95.
[31] H. Xia, N. Zhang, G. De Roeck, Dynamic analysis of high speed railway bridge under articulated trains, Comput. Struct. 81 (26–27) (2003) 2467–2478.
[32] D. Cantero, T. Arvidsson, E. OBrien, R. Karoumi, Train–track–bridge modelling and review of parameters, Struct. Infrastruct. Eng. 12 (9) (2016) 1051–1064.
[33] K.N. Gia, J.M.G. Ruigómez, F.G. Castillo, Influence of rail track properties on vehicle–track responses, in: Proceedings of the Institution of Civil Engineers-Transport, Vol. 168, Thomas Telford Ltd, 2015, pp. 499–509.
[34] I. Behmanesh, B. Moaveni, Accounting for environmental variability, modeling errors, and parameter estimation uncertainties in structural identification, J. Sound Vib. 374 (2016) 92–110.
[35] S. Seo, A Review and Comparison of Methods for Detecting Outliers in Univariate Data Sets (Ph.D. thesis), University of Pittsburgh, 2006.
[36] K. Maes, L. Van Meerbeeck, E. Reynders, G. Lombaert, Validation of vibration-based structural health monitoring on retrofitted railway bridge KW51, Mech. Syst. Signal Process. 165 (2022) 108380.
[37] K. Maes, G. Lombaert, Monitoring railway bridge KW51 before, during, and after retrofitting, J. Bridge Eng. 26 (3) (2021) 04721001.
[38] S. Kamali, S. Quqa, A. Palermo, A. Marzani, Reducing false alarms in structural health monitoring systems by exploiting time information via Binomial Distribution Classifier, Mech. Syst. Signal Process. 207 (2024) 110938.