Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 10(8): 18558–18585.
DOI: 10.3934/math.2025829
Received: 05 May 2025
Revised: 06 August 2025
Accepted: 14 August 2025
Published: 18 August 2025

Research article

Data-driven structural transition detection using vibration monitoring and LSTM networks

A. Presno Vélez, M. Z. Fernández Muñiz* and J. L. Fernández Martínez

Department of Mathematics, Group of inverse problems, optimization and machine learning, Oviedo University, C/Federico García Lorca 18, Oviedo 33007, Spain

* **Correspondence:** Email: zulima@uniovi.es.

Abstract: Structural health monitoring (SHM) is essential for ensuring the safety and durability of civil infrastructure. Traditional SHM approaches, based on manual inspections or threshold-based analyses, often fail to detect early or subtle structural changes. In this work, we propose a data-driven framework for detecting structural regime transitions using long short-term memory (LSTM) networks trained on power spectral density data. This method does not require prior knowledge of the excitation sources or structural dynamics, enabling robust and interpretable transition detection under real-world conditions. A key component of the framework is the empirical transition point, T_a, computed from prediction probabilities through persistence thresholding and entropy filtering. This allows for precise and automated detection of regime shifts, even in the absence of explicit ground truth. The model is validated on vibration data collected under ambient and train-induced excitations, achieving high accuracy in distinguishing pre- and post-retrofitting states. It demonstrates strong robustness across a range of operational and dynamic conditions. To enhance interpretability, we introduce the confidence variability index (CVI), which quantifies the temporal stability of the model's predictions and serves as an indicator of transition consistency. While the framework does not currently identify the physical causes of transitions, its sensitivity to dynamic changes makes it a valuable early-warning tool in SHM. Despite the inferential nature of transition detection due to the lack of ground truth, the approach offers a scalable, interpretable, and real-time solution for structural regime monitoring—contributing to the advancement of SHM systems through uncertainty quantification, causal inference, and intelligent infrastructure management.

Keywords: structural health monitoring (SHM); long short-term memory (LSTM) networks; regime transition detection; confidence variability index (CVI); entropy-based vibration analysis
Mathematics Subject Classification: 62M10, 68T05, 93E10, 74H15

1. Introduction

In recent years, vibration-based structural health monitoring (SHM) has become essential for assessing the integrity and safety of civil infrastructure [1]. SHM systems use sensors—such as accelerometers, strain gauges, and displacement transducers—to capture dynamic responses under ambient or operational loads. These measurements allow us to track changes in the modal parameters (e.g., natural frequencies, mode shapes, damping), which are often indicative of structural damage or degradation [2].

Optimization algorithms have further improved SHM by enhancing damage detection and optimal sensor placement [3], enabling monitoring strategies tailored to specific constraints. The widespread availability of low-cost, high-frequency accelerometers now supports large-scale, continuous vibration monitoring. However, the analysis of these noisy, and often incomplete data remains challenging due to the nonlinear dynamics and changing operational conditions [4].

To address these issues, data-driven methods—particularly deep learning models such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks—have shown strong potential. Originally introduced in [5], LSTMs are well suited for learning long-term dependencies in noisy time-series data and have been applied in hybrid convolutional neural network (CNN)-LSTM models for damage detection [6], attention-based state estimation [7], and fault detection in optimization-based pipelines [8]. Similar approaches have been explored in related domains like traffic accident prediction [9], federated learning for fault diagnosis [10–12], and railway infrastructure monitoring via CNNs and guided wave analysis [13–16].

Recent work has tackled key SHM challenges such as data incompleteness [17], signal reconstruction with LSTMs [18], and fault prediction [19, 20]. Machine learning has also been used to evaluate retrofitting effects and detect early transitions [21, 22]. However, most approaches rely on static classification, limiting their ability to capture the gradual nature of structural evolution. Moreover, the lack of interpretable early-warning indicators hinders real-world deployment. Traditional operational modal analysis (OMA) methods characterize structural dynamics under ambient loads [23] but require manual post-processing and struggle with nonlinear or nonstationary behavior.

SHM has been extensively applied to civil infrastructure, particularly in long-span and railway bridges, where early detection of structural changes is critical for preventing failures and optimizing maintenance strategies. Notable examples include continuous vibration monitoring of operational bridges [24], the integration of machine learning techniques for damage detection in complex bridge systems [25], and the deployment of sensor networks in cable-stayed bridges for long-term performance assessment [26]. These real-world applications highlight the critical role of SHM in infrastructure's resilience and underscore the need for scalable, robust, and automated methods capable of operating under varying environmental and operational conditions.

Theoretical work has helped explain the success of deep networks in high-dimensional settings [27, 28], while generalization bounds and learning guarantees have been developed using complexity measures such as Rademacher and Gaussian complexities [29], as well as Vapnik–Chervonenkis (VC) dimension-based approaches [30]. These insights underpin the robustness of deep models like LSTMs in SHM contexts.

Our core innovation is a probabilistic transition metric, T_a, which marks the earliest time when a structural change is confidently detected. This reformulation frames transition detection as a temporal inference task, offering a continuous and interpretable view of structural evolution. Our method uses LSTM networks trained on power spectral density (PSD) sequences from accelerometer data recorded before and after retrofitting. This eliminates the need for explicit modal identification and enhances robustness to environmental variability. The networks are trained using the adaptive moment estimation (Adam) optimizer [31], which facilitates efficient convergence in noisy, high-dimensional parameter spaces.

The framework is validated on real-world data from the KW51 railway bridge, focusing particularly on the retrofitting phase, which lacks ground truth. The model successfully infers regime shifts based solely on signal dynamics.

To evaluate generalization, the framework is tested under two excitation scenarios: Ambient vibration and train-induced loads. The consistent performance across conditions demonstrates the model's adaptability and deployment potential. The T_a metric is defined as the first time point when the prediction probability surpasses a threshold while meeting the persistence and low-entropy criteria. This ensures reliable detection and minimizes false positives. While other changepoint detection methods exist—such as sequential probability ratio test (SPRT) [32], cumulative sum control chart (CUSUM) [33], or Bayesian online changepoint detection [34, 35]—these typically require explicit probabilistic assumptions. In complex, label-sparse SHM environments like ours, the proposed approach offers a more pragmatic and interpretable alternative.

To better illustrate the methodological differences, Table 1 compares the proposed LSTM-based approach with two widely used changepoint detection methods, CUSUM and SPRT.

Table 1. Comparison of changepoint detection methods.

Feature	**LSTM**	**CUSUM**	**SPRT**
Approach	Deep learning	Statistical control chart	Sequential hypothesis test
Data requirements	Large labeled time series	Small baseline sample	Probabilistic model (H_0, H_1)
Type of change	Gradual, nonlinear	Small mean shifts	Abrupt changes
Computational complexity	High	Low	Moderate
Typical use case	SHM with noisy, evolving dynamics	Fast detection in univariate series	Real-time decision-making

As summarized in Table 1, traditional changepoint detection techniques offer advantages in simplicity and speed but rely on strong statistical assumptions. In contrast, LSTM-based models provide greater flexibility and adaptability for complex, label-sparse SHM scenarios, which motivates their use in this work.

2. Materials and methods

2.1. The data

The dataset employed in this study originates from the long-term vibration monitoring of the KW51 railway bridge, which underwent structural retrofitting between May and September 2019. The monitoring system included 12 PCB 393B04 piezoelectric accelerometers (1 mV/g sensitivity), 12 CAE-06-250UN-350 uniaxial strain gauges (constantan in polyimide), and 4 CAE–06–250UR–350 triaxial displacement sensors, all sampled at high resolution (24-bit Analog-to-Digital A/D conversion).

Two Welotec OWLL 8025 displacement sensors (6.3 μm resolution) were installed at the Leuven-side supports. Environmental data were collected using a Type-T thermocouple and an HM1500LF humidity sensor. Due to the quality, consistency, and direct relevance of acceleration data to the modal analysis conducted, only the accelerometer recordings were used in this study.

Acceleration data were collected continuously during this period under two distinct operational conditions

- Ambient: Free vibration data under natural excitation sources (e.g., wind, traffic), recorded without the passage of trains;
- Train-induced: Forced dynamic responses captured during train crossings.

Accelerometer data were grouped into three categories as follows:

- Before: Signals acquired prior to the commencement of retrofitting;
- While: Signals collected during the ongoing reinforcement activities;
- After: Signals recorded following the completion of all structural interventions.

From the total available dataset, 7% of the data were excluded due to sensor outages or communication failures. The complete dataset comprises $46,672$ PSD vectors distributed across structural phases and excitation conditions (see Table 2).

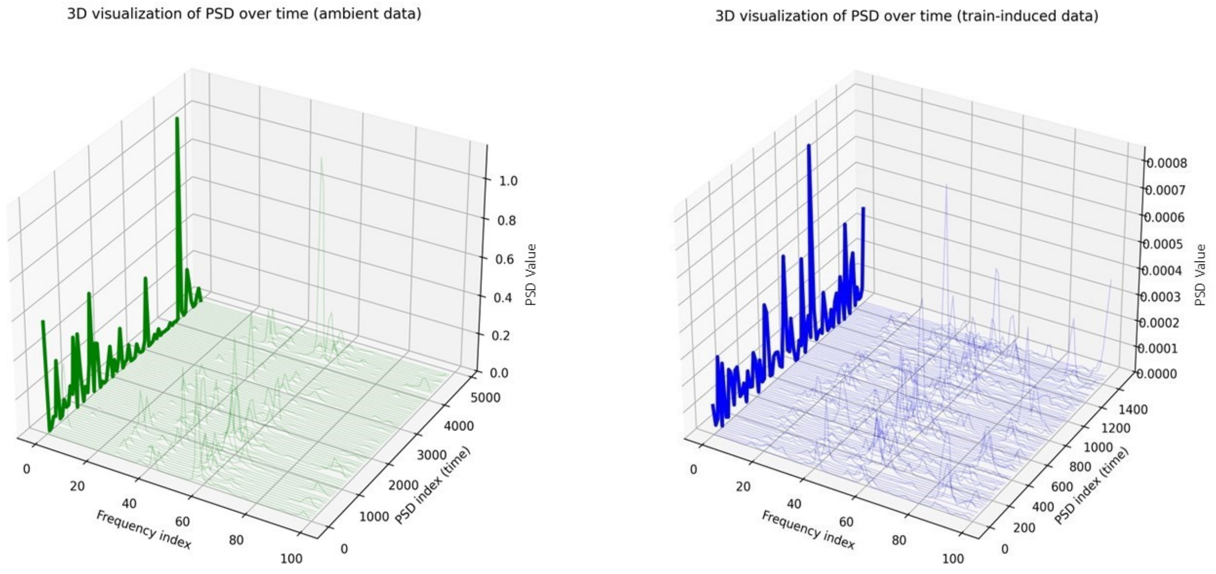
Table 2. Number of samples under different structural conditions.

	Ambient	Train-induced
Before retrofitting	14864	4232
While retrofitting	4872	1494
After retrofitting	16272	4968

During the retrofitting period—from May 15 to September 24, 2019—ambient data were collected at a relatively high frequency, averaging approximately one measurement every 39 minutes. Given the acquisition rate, this temporal aggregation corresponds to approximately 6.5 hours of structural monitoring, aligning the model's temporal resolution with the expected progression of retrofitting activities during a standard workday. This temporal match reinforces the interpretability of $T_a = 10$ as not only the minimum evaluable transition point but also as a value naturally associated with the pace of structural changes expected during a standard retrofitting workday. For ambient vibration data, there are 14864 samples corresponding to the *before* phase, 4872 to the *while* phase, and 16272 to the *after* phase. For train-induced responses, the corresponding figures are 4232, 1494, and 4968 accelerometer samples, respectively.

PSD calculations during the *while* phase were performed using all available accelerometer channels from the daily files, provided they contained valid data. This approach ensured a comprehensive representation of the system's dynamic behavior during the retrofitting process, incorporating contributions from multiple sensor locations distributed across the bridge.

Figure 1(a) illustrates the temporal evolution of PSD values under ambient excitation conditions. The frequency index is represented on the horizontal axis, the sequence of PSD observations over time on the depth axis, and the corresponding PSD values on the vertical axis. This three-dimensional (3D) representation shows the PSD of different accelerometers as a function of time and the frequency bands that are predominant in each case. To better characterize the spectral dynamics under train-induced excitation, Figure 1(b) displays a 3D visualization of the PSD over time. This visualization reveals the transient and highly localized nature of the vibrational energy during train passages. Sharp spectral peaks appear intermittently for different frequencies ranges across time. The temporal variation of these spectra is what should enable the detection of the regime transition from a structural *while* mode to a structural *after* mode. Detailed statistical visualizations of PSD variability are available in [21].



(a) Ambient phase

(b) Train-induced phase

Figure 1. Three-dimensional visualization of PSD's evolution over time for (a) ambient, and (b) train-induced vibration data. The x-axis represents the frequency bins, the y-axis corresponds to the sequence of PSD measurements over time, and the z-axis shows the corresponding PSD values. Localized peaks reveal dominant frequency bands activated by the dynamic loading for diferent accelerometers as a function of time. Each PSD is composed of 100 sample frequencies.

For LSTM training, all samples were chronologically ordered by acquisition timestamp and grouped into overlapping sequences of 10 consecutive PSDs using a one-step sliding window. Each sequence inherited the label of its final time step and served as a training instance. As a result, a typical 10-step sequence spans approximately 6-7 hours of structural response. In contrast, train-induced data were more sparsely sampled due to dependence on train passages, meaning that sequences in this case may reflect cumulative dynamic behavior over several days. This difference in temporal density is critical for interpreting the model's outputs under each excitation regime. The application of the algorithm to the *while* data uses all the spectra from the accelerometers involved in the predictive sequence, yielding the probability of a transition to the after phase.

The inclusion of both ambient and train-induced vibration signals during the retrofitting (*while* data) allows for cross-validation of the model's sensitivity to varying excitation conditions. Moreover, the chronological ordering of the samples facilitates the detection of gradual shifts in modal behavior, particularly during the *while* phase, where no explicit ground truth labels are available for the transition points.

To enhance the robustness and interpretability of detection, the methodology includes the analysis of uncertainty-related metrics. Specifically, it evaluates the confidence variability index (CVI), which measures the temporal stability of predicted probabilities over a sequence, and the model output's entropy, which directly quantifies the uncertainty of each prediction. These indicators not only help confidently identify the transition point but also capture the overall behavior of the transition during the *while* phase, providing a richer and more nuanced understanding of the dynamic process the structure undergoes as retrofitting progresses.

2.2. Long short-term memory networks

LSTM networks are an advanced type of recurrent neural network (RNN), specifically designed to model sequential data with long-range dependencies. Unlike standard neural networks that process inputs independently, RNNs incorporate memory, allowing past information to influence the current outputs. However, traditional RNNs struggle to retain information over long sequences. LSTMs networks, introduced by Hochreiter and Schmidhuber in 1997 [5], address this issue with an internal memory cell regulated by gating mechanisms. These gates control the flow of information by determining what to keep, update, or discard. This allows LSTM networks to:

- Retain important information over long sequences,
- Filter out irrelevant or outdated data, and
- Propagate essential context forward in the sequence.

This architecture makes LSTM particularly effective for tasks involving complex temporal patterns, such as language modeling, time series prediction, and speech recognition. In this work, we implement an LSTM-based RNN to classify structural states and detect dynamic transitions. This approach is well suited for SHM systems, where the data exhibit both temporal structure and long-range dependencies. Additionally, LSTMs networks are resilient to noise and missing data—key advantages in real-world applications. However, their enhanced modeling capabilities come with an increased computational cost.

Table 3 summarizes the key architectural properties of LSTM networks that make them particularly suitable for SHM applications and the evaluation of structural retrofitting. Their ability to model temporal dynamics, handle nonlinearities, and retain relevant memory over long periods is especially valuable in detecting both gradual degradation and post-retrofit improvements.

Table 3. LSTM features that are relevant to SHM and retrofitting.

Feature	**LSTM Capability**	**Relevance to SHM**
Temporal modeling	Tracks long-term patterns	Detects slow damage or improvements
Memory mechanism	Keeps relevant events, forgets noise	Focuses on key structural changes
Predictive accuracy	Learns normal behavior, flags changes	Identifies deviations after the retrofit
Nonlinear dynamics	Handles complex responses	Captures retrofit-induced effects
Pre/post modeling	Trains on separate phases	Compares conditions before/after

The internal dynamics of an LSTM cell are governed by the following set of equations:

a. **Forget gate:** Determines what information is discarded from the memory cell

$$\mathbf{f}_t = \sigma(\mathbf{W}_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f). \tag{2.1}$$

Here, $\mathbf{x}_t$ is the input at time t, $\mathbf{h}_{t-1}$ is the previous hidden state, $\mathbf{W}_f$ is the weight matrix for the forget gate, $\mathbf{b}_f$ is the bias vector, and $\sigma(\cdot)$ is the sigmoid activation function

$$\sigma(z) = \frac{1}{1 + e^{-z}}. \tag{2.2}$$

The input and hidden state are concatenated and linearly transformed before applying the nonlinearity to produce the forget vector $\mathbf{f}_t$, which scales the previous memory content.

b. **Input gate:** Controls the incorporation of new information, composed of two substeps

$$\mathbf{i}_t = \sigma(\mathbf{W}_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i), \tag{2.3}$$
$$\tilde{\mathbf{c}}_t = \tanh(\mathbf{W}_c[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c). \tag{2.4}$$

In the first equation, $\mathbf{W}_i$ and $\mathbf{b}_i$ are the weights and biases for the input gate; the gate decides which parts of the new information to store in the memory cell. In the second equation, $\mathbf{W}_c$ and $\mathbf{b}_c$ are the weights and biases for computing the candidate cell content, and $\tanh(\cdot)$ squashes the values into the range $[-1, 1]$; this candidate is modulated by the input gate to update the cell's state.

c. **Memory cell update:** Combines the previous memory and new candidate information:

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t. \tag{2.5}$$

This equation combines the previous cell state $\mathbf{c}_{t-1}$, scaled by the forget gate $\mathbf{f}_t$, with the new candidate $\tilde{\mathbf{c}}_t$, scaled by the input gate $\mathbf{i}_t$. The symbol $\odot$ denotes element-wise multiplication.

d. **Output gate:** Determines the exposed part of the memory

$$\mathbf{o}_t = \sigma(\mathbf{W}_o[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o), \tag{2.6}$$

where $\mathbf{W}_o$ and $\mathbf{b}_o$ are the weights and biases for the output gate, which determines how much of the memory content is exposed to the next hidden state.

e. **Hidden state update:**

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{2.7}$$

The updated hidden state $\mathbf{h}_t$ is obtained by applying tanh to the current cell state $\mathbf{c}_t$, and modulating it with the output gate $\mathbf{o}_t$, allowing the network to control information flow to the next time step.

f. **Classification layer:** Once the final hidden state is obtained, $\mathbf{h}_T$, it is passed through a dense layer followed by a sigmoid activation to produce an estimated probability vector for the testing samples

$$\hat{\mathbf{y}} = \sigma(\mathbf{W}_{\text{out}}\mathbf{h}_T + \mathbf{b}_{\text{out}}), \tag{2.8}$$

where $\mathbf{W}_{\text{out}}$ and $\mathbf{b}_{\text{out}}$ are the weights and biases of the output layer. The value $\hat{\mathbf{y}}$ is interpreted as the probability that the input sequence belongs to Class 1 (e.g., the after state). The model is trained using the binary cross-entropy loss function

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right], \tag{2.9}$$

where $\hat{y}_i \in [0, 1]$ is the predicted probability for the i-th sample, $y_i \in \{0, 1\}$ is the corresponding true class label, and N is the number of testing samples.

Training the LSTM model involves determining the different weight matrices and bias terms, depending on the architecture of the network. Optimization is carried out using the Adam algorithm, which adapts the learning rate for each parameter using estimates of the first and second moments of its gradients ([31]). The equations of the Adam algorithm are provided in the Appendix.

The proposed framework uses a compact LSTM-based architecture aimed at balancing temporal modeling capacity and computational efficiency. All experiments, including model training and inference, were carried out on a standard laptop (Intel® Core™ i9-13900H @ 2.60 GHz, 32 GB RAM, no dedicated GPU). The initial model training was completed in approximately 1 minute and 52 seconds. Subsequent inference operations were executed in the order of milliseconds per sequence, supporting near-real-time deployment. The design is modular and compatible with distributed or edge-based implementations, facilitating future scalability to larger sensor networks or more complex structures.

2.3. The prediction algorithm

The overall methodology adopted in this study, from data acquisition to transition detection and interpretation of the results, is summarized in the flowchart presented in Figure 2. This schematic representation provides a high-level overview of the sequential stages followed in the development of the proposed framework.

The algorithm is composed of different steps, as described below.

Data preprocessing

Each measurement corresponds to a short-duration time window of acceleration data recorded by sensors installed on the bridge. These raw time-domain signals were transformed into the frequency domain using Welch's method—a widely adopted technique in structural dynamics for producing reliable spectral estimates with reduced variance. In this process, each time window is divided into overlapping segments, windowed individually (typically using a Hanning function), Fourier transformed, and then averaged to obtain the final spectrum. As a result, each measurement is represented as a PSD vector comprising 100 frequency bins ([21, 22]). These bins encapsulate the distribution of vibrational energy across the relevant frequency range, providing a compact yet informative summary of the structure's dynamic characteristics. The frequency resolution and range were selected to encompass the fundamental and low-order modes of the bridge, which are most sensitive to structural changes. This spectral representation constitutes the input feature space for subsequent machine learning analyses, enabling the model to capture patterns associated with evolving structural conditions over time. All PSD samples were timestamped and subsequently classified into one of three structural phases: before, while, and after retrofitting.

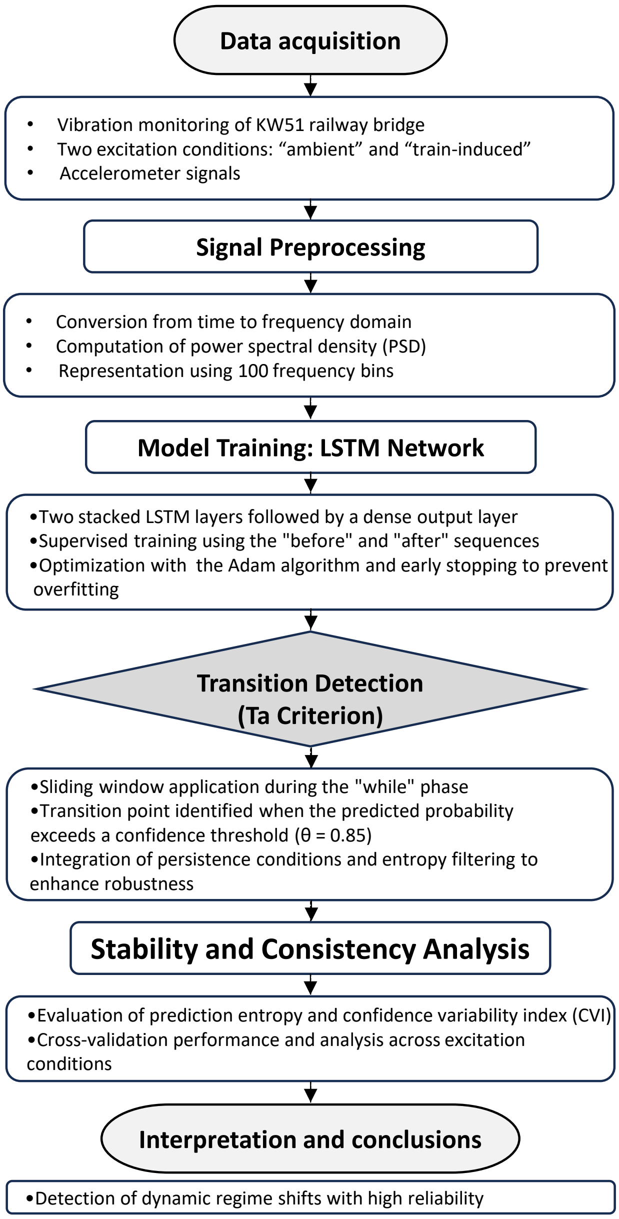

Data acquisition
• Vibration monitoring of KW51 railway bridge
• Two excitation conditions: “ambient” and “train-induced”
• Accelerometer signals
Signal Preprocessing
• Conversion from time to frequency domain
• Computation of power spectral density (PSD)
• Representation using 100 frequency bins
Model Training: LSTM Network
•Two stacked LSTM layers followed by a dense output layer
•Supervised training using the "before" and "after" sequences
•Optimization with the Adam algorithm and early stopping to prevent overfitting
Transition Detection (Ta Criterion)
•Sliding window application during the "while" phase
•Transition point identified when the predicted probability exceeds a confidence threshold (θ = 0.85)
•Integration of persistence conditions and entropy filtering to enhance robustness
Stability and Consistency Analysis
•Evaluation of prediction entropy and confidence variability index (CVI)
•Cross-validation performance and analysis across excitation conditions
Interpretation and conclusions
•Detection of dynamic regime shifts with high reliability

Figure 2. Overview of the data-driven framework for structural transition detection. The flowchart outlines the main stages: data acquisition, signal preprocessing, LSTM-based model training, transition detection based on the Ta criterion, stability analysis, and final interpretation.

Figure 3 clearly illustrates the differences between ambient and train-induced vibrations in both the time and frequency domains. Ambient signals exhibit low-amplitude random fluctuations, with distinct spectral peaks around 200–300 Hz for the *before* and *while* signals, and approximately 100 Hz for the ambient signal, likely associated with structural resonances. The train-induced signal prior to the main event is almost negligible. During the train's passage, the acceleration time series is dominated by high-amplitude transients, particularly in the *before* signal, which reflects the direct dynamic loading.

The corresponding (PSD) reveal broadband excitation with prominent peaks between 200 and 300 Hz, indicating activation of higher-order modes. After the train passes, both signals return to lower amplitudes, although the ambient signal retains slight modulations, suggesting a residual structural response. These observations confirm that train-induced vibrations significantly alter both the energy content and the frequency characteristics of the system—an essential aspect for damage detection and modal analysis in SHM.

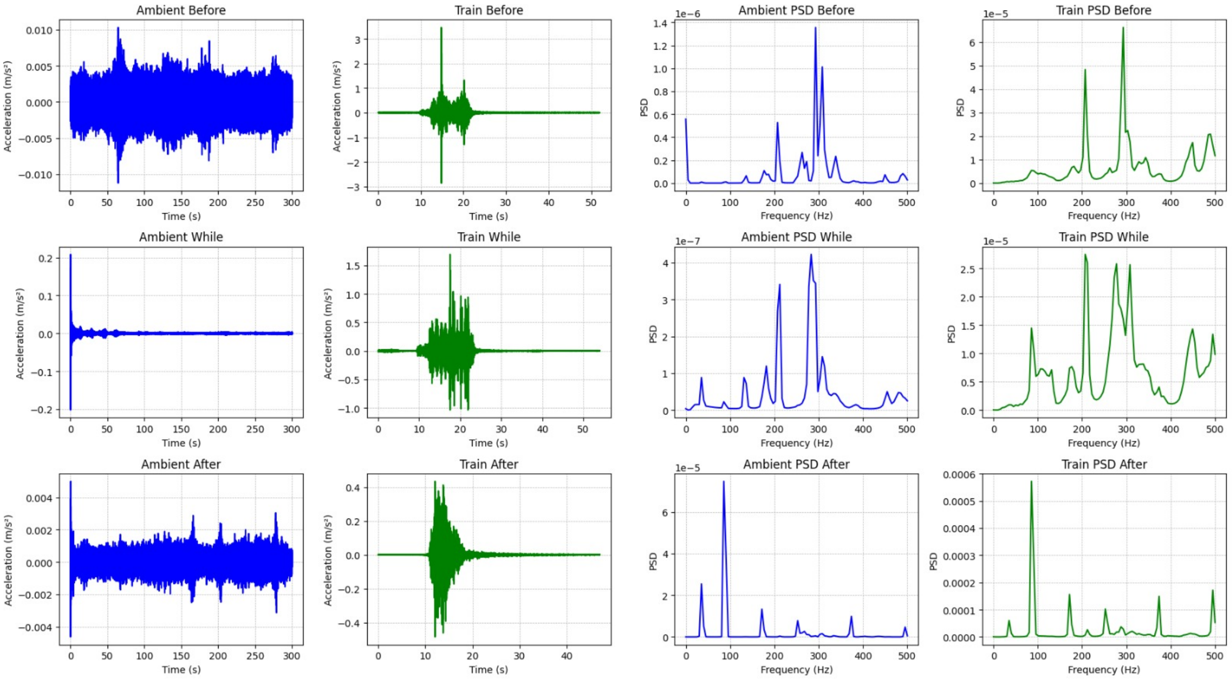


Figure 3. Time series (left and middle columns) and PSD (right columns) of vertical acceleration under *ambient* (blue) and *train-induced* (green) conditions at three different time intervals: before (top row), during (middle row), and after (bottom row) train passage. Ambient signals were acquired over 5 minutes, while train signals correspond to 50-second windows around the train event. PSDs are computed using Welch's method.

Model training on before and after retrofitting data sampled

The model training consists in a LSTM network for the binary classification problem before and after retrofitting, as described in section devoted to the description of the LSTM network.

The model input in time t, $\mathbf{X}_t$, is a concatenation of 10 consecutive PSD vectors, $\mathbf{X}_t = [\mathbf{x}_{t-9}, \ldots, \mathbf{x}_t]$, where $\mathbf{x}_i \in \mathbb{R}^{100}$, and $\mathbf{x}_t$ stands for the PSD vector corresponding to the accelerometer at time t. The model outputs a predicted probability, $\hat{y} \in [0, 1]$ for this binary classification problem.

In the present work, the network architecture has the following components, where the input is multidimensional and the output is a scalar, probabilistic value produced at each time step.

(1) Stacked LSTM layers: Two layers with 64 and 32 units, respectively. Stacking enables the capture of progressively abstract temporal features: The first layer models short-term dependencies, while the second aggregates higher-level patterns. The number of units was chosen to improve computational efficiency while maintaining accuracy, according to empirical validation.

(2) Dropout regularization: A dropout layer with a rate of 0.3 is inserted between the LSTM layers to prevent overfitting by randomly deactivating units during training.

(3) Fully connected dense layer: A dense layer with 32 units and rectified linear unit (ReLU) activation, performing nonlinear transformations over the encoded temporal representation. The ReLU activation is defined as

$$\mathrm{ReLU}(\mathbf{z}) = \max(0, \mathbf{z}), \tag{2.10}$$

where $\mathbf{z}$ is the result of the linear transformation applied to the output of the previous layer

$$\mathbf{z} = \mathbf{W}_{\text{dense}}\mathbf{h}_i + \mathbf{b}_{\text{dense}}, \tag{2.11}$$

where $\mathbf{h}_i$ is the input to the dense layer, $\mathbf{W}_{\text{dense}}$ is the weight matrix, and $\mathbf{b}_{\text{dense}}$ is the bias vector.

(4) Output layer: A single sigmoid neuron producing the predicted probability, $\sigma(\mathbf{o})$, where $\mathbf{o}$ is the linear output before the sigmoid activation

$$\mathbf{o} = \mathbf{W}_{\text{out}}\mathbf{h}_{\text{dense}} + \mathbf{b}_{\text{out}}, \tag{2.12}$$

where $\mathbf{h}_{\text{dense}}$ is the output of the previous dense layer (with ReLU activation) and $\mathbf{b}_{\text{out}}$ the bias term added to adjust the activation function's input.

Model prediction on while data

Once trained on labeled *before* and *after* data, the LSTM model is applied to sequences from the while phase—corresponding to the structural reinforcement process—using a sliding window approach. Unlike the *before* and *after* phases, the while data lack explicit labels indicating the structural state at each moment. Therefore, the model is used inferentially during this phase: For each windowed PSD sequence, it outputs a probability $\hat{y}_t$ that the sequence belongs to the after class.

A transition is declared when the predicted probability exceeds a predefined threshold θ

$$T_a = \min\{t \mid \hat{y}_t \geq \theta\}, \tag{2.13}$$

providing a data-driven estimate of the earliest structural regime change. The optimal threshold θ, discussed in the Results section, is expected to be 0.85. This is the value of the threshold that will be used to classify the structural transition of the *while* sequences.

This framework combines probabilistic inference, temporal modeling, and vibrational signal analysis into a robust tool for automatic detection of structural transitions in civil infrastructure. The use of an LSTM is motivated by the complexity of the transition dynamics, which may be nonlinear, nonmonotonic, and temporally diffuse—requiring a model capable of capturing long-term dependencies.

Post-processing: visual analysis of complementary metrics

To provide a deeper understanding of model behavior, five types of visualizations were generated for both *ambient* and *train-induced* datasets. Each plot targets a specific analytical objective, either related to the input signals' dynamics or the structure of the model's predictive responses.

(1) **PSD correlation matrix.** This plot displays the pairwise Pearson correlation coefficients across all frequency bands in the PSD. Under stationary conditions, a strongly organized matrix structure is expected. In our results, the ambient dataset exhibits a highly structured correlation pattern, indicative

of coherent vibrational modes. In contrast, the train-induced matrix is more irregular and diffuse, consistent with transient, spatially localized excitations.

(2) T_a **evolution plot.** This plot tracks the temporal evolution of the detected T_a values, allowing us to evaluate the stability and timing of transition detections. For the ambient data, T_a stabilizes early and remains consistent across time. The train-induced data, however, show greater temporal dispersion, reflecting the nonstationary nature of train passages.

(3) T_a **histogram.** This histogram summarizes the distribution of T_a values across all sequences. In the ambient case, the histogram exhibits a sharp peak at $T_a = 10$, suggesting consistent early transitions. In contrast, the distribution for the train-induced dataset is broader, indicating more variability in transition timing.

(4) **Maximum confidence per sequence.** This plot reports the maximum predicted probability for the *after* class within each sequence. Ambient sequences consistently reach high and stable confidence levels. Conversely, train-induced sequences show greater fluctuation, with lower peaks and increased variability.

(5) **Temporal evolution of output entropy.** This visualization quantifies the uncertainty of the model's classification outputs over time using entropy. Lower entropy values reflect higher confidence. Ambient sequences maintain low entropy throughout, whereas train-induced sequences show elevated entropy for longer durations, indicating sustained uncertainty in classification.

3. Results

3.1. Cross-validated classification performance in the training phase: before versus after retrofitting (binary classification)

To evaluate the model's robustness and generalization capability, a five-fold cross-validation procedure was performed using non-overlapping temporal blocks from the labeled *before* and *after* phases. This strategy ensures independence between the training and validation sets by reserving entire contiguous time segments for evaluation, thus avoiding potential data leakage. To enhance robustness, a dropout rate of 0.3 was employed during training, offering a favorable balance between generalization and convergence speed. Although higher dropout values (e.g., 0.5) were tested, they led to slower convergence and marginally decreased accuracies. The model achieved an average classification accuracy of 99.4% across folds, with individual fold accuracies ranging from 98.7% to 99.8%, as summarized in Table 4. Such consistently high performance validates the model's ability to generalize its temporal learning across independent sequences and operational periods, reinforcing its potential for deployment in real-world monitoring applications.

Table 4. Cross-validation accuracy per fold.

Fold	Accuracy (%)
1	99.78
2	99.80
3	99.51
4	98.80
5	98.69
Mean	**99.34**

3.2. Optimum value of the transition threshold θ for while data detection

To avoid arbitrary selection and provide statistical grounding for this choice, an empirical analysis of the distribution of $\hat{y}_t$ over labeled *before* and *after* sequences was conducted.

For instance, in the case of the *train-induced* data, the analysis of the empirical distribution of $\hat{y}_t$ for both classes (before and after retrofitting) shows that the predicted probabilities for the *before* class are highly concentrated near zero, with over 99% of observations falling below 0.002. This extreme concentration causes the histogram for this class to collapse into the first bin. In contrast, after predictions cluster near 1.0, producing a clear separation between the two distributions. This behavior suggests high model specificity in the *before* phase, where $\hat{y}_t$ rarely exceeds the minimal values. Consequently, the threshold $\theta = 0.85$ lies in a high-confidence region for the *after* class and is practically unreachable under *before* conditions, thus minimizing the risk of false detections.

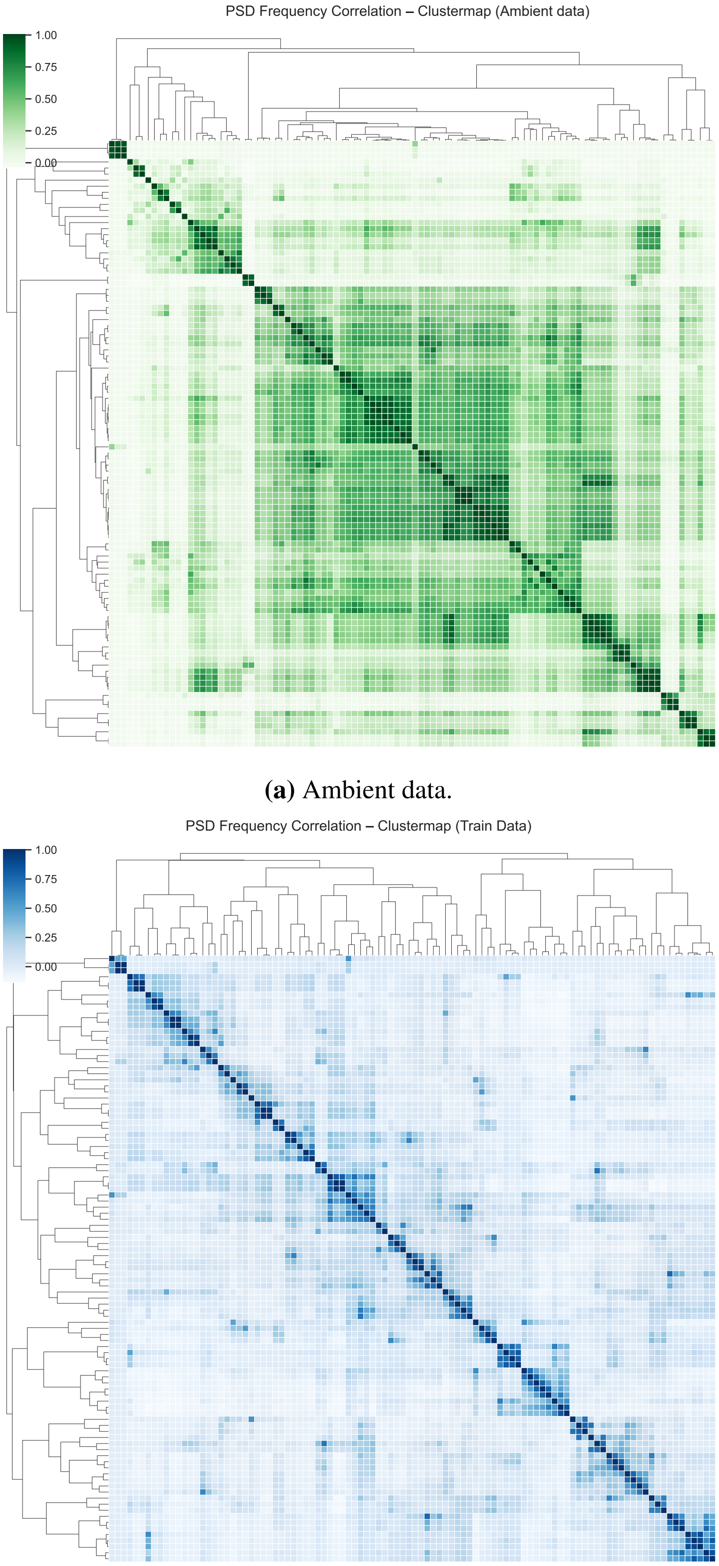
The statistical validity of this separation was confirmed through two nonparametric tests, namely The Kolmogorov–Smirnov test and the Mann–Whitney U test, both yielding p-values below 10^{-5}, indicating highly significant differences between the distributions. Together, these empirical results provide a solid foundation for adopting a high threshold such as $\theta = 0.85$, ensuring both specificity and sensitivity in the detection of structural transitions. Similar results were achieved for the ambient data.

3.3. Ambient vibration results

This subsection presents the model's behavior when applied to sequences acquired under ambient excitation. Multiple complementary visualizations were used to characterize the detection of transition points, the evolution of prediction confidence, the structure of input features, and the overall temporal uncertainty. To assess the input feature structure, the correlation matrix among PSD frequency components is shown first in Figure 4(a). Low off-diagonal correlations indicate weak interdependencies between features, confirming that the frequency bands contribute complementary information to the model. This spectral independence justifies the use of all PSD components without dimensionality reducing or decorrelation preprocessing, and it supports the model's ability to generalize transition detection across diverse signal compositions.

Figure 5 shows the evolution of the detected transition points T_a across time. A consistent detection pattern emerges, with transitions concentrated around specific steps, suggesting a structured dynamic change potentially induced by environmental effects. This observation is quantitatively supported by the histogram in Figure 6, where most transitions occur between Steps 9 and 12, with a clear mode at Step 10. Such concentration indicates the model's ability to consistently identify a dominant structural threshold.

Out of 100 sequences analyzed under ambient excitation, a transition was detected in 92 sequences. The mode of detected transition steps is $T_a = 10$, and the mean transition step is approximately 11.48, consistent with the occurrence of an early dynamic shift. The histogram of T_a values confirms this trend, showing a strong concentration between Steps 10 and 15, with a declining tail toward Step 30. These findings indicate that the model consistently detects a significant change shortly after the retrofitting process begins.



(a) Ambient data.

(b) Train-induced data.

Figure 4. While data. Correlation matrix of PSD frequency components. Low off-diagonal values suggest diverse and complementary spectral features.

Figure 7 depicts the maximum predicted probability for the *after* class across sequences taking into account that the analysis of each accelerometer, for a given day, is independent. The consistently high confidence values confirm the model's robustness in distinguishing post-transition behavior with minimal ambiguity. Data points lying below the 0.85 threshold are interpreted as exhibiting structural behavior comparable with that of the before retrofitting phase. The model quickly transitions to a high and stable level of confidence from June onward, suggesting the detection of a sustained structural change starting at that time.

Further insights into the model's uncertainty are provided by Figure 8, which shows the temporal evolution of prediction entropy. Entropy peaks briefly near the transition points but remains low elsewhere, suggesting that the model maintains high confidence during stable regimes and becomes momentarily uncertain only around regime shifts.

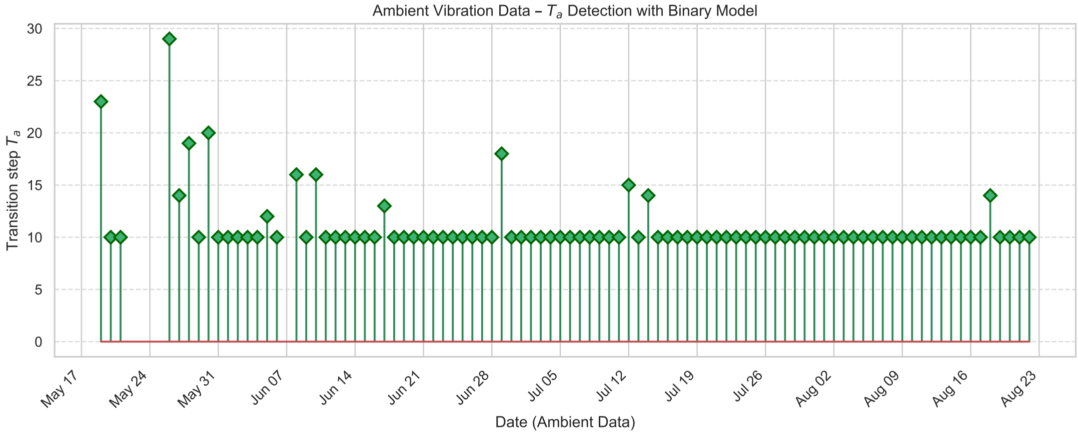


Figure 5. *Ambient* data while retrofitting. Evolution of detected transition points T_a. A consistent concentration of transitions is observed.

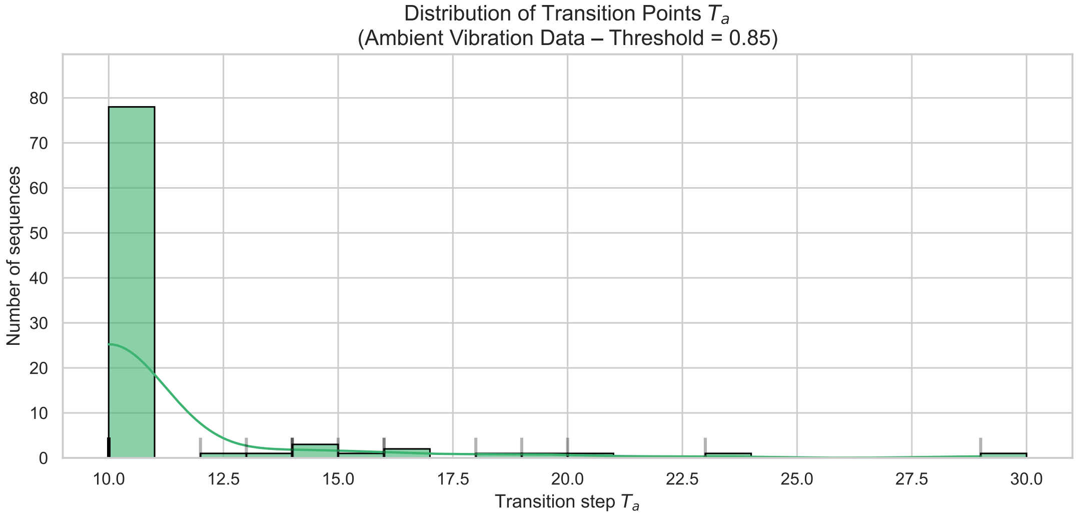


Figure 6. *Ambient* data while retrofitting. Histogram of detected T_a values for ambient sequences. Most transitions cluster between Steps 9 and 12.

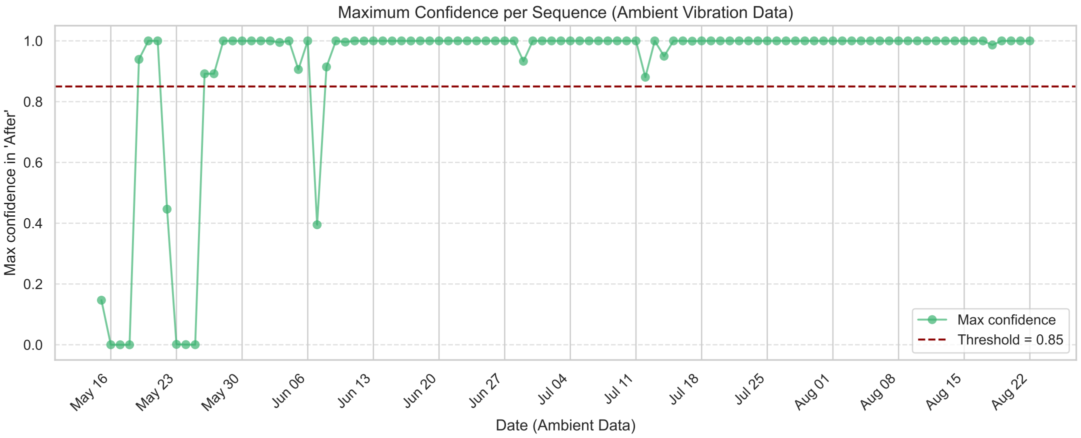


Figure 7. *Ambient* data while retrofitting. Maximum confidence assigned to the *after* class across 100 ambient sequences from May 16 to August 22. High probability values indicate reliable transition detection of the while accelerometer towards the *after* retrofitting class.

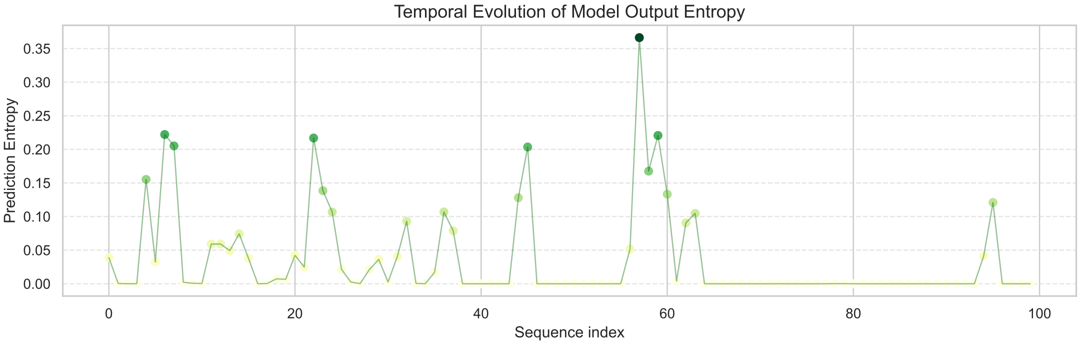


Figure 8. Ambient data while retrofitting. Temporal evolution of prediction entropy. Peaks coincide with detected transitions, while stable phases exhibit low entropy.

3.4. Train-induced vibration results

This subsection presents the model's behavior when applied to sequences acquired under train-induced excitation. As in the ambient case, we use a set of complementary visualizations to examine transition detection, prediction confidence, input features' structure, and temporal uncertainty. Compared with ambient conditions, forced excitation introduces more complex dynamic patterns, potentially leading to more heterogeneous responses.

To evaluate the structure of the input features, the PSD frequency correlation matrix is shown in Figure 4(b). The matrix reveals consistently low off-diagonal correlations, indicating weak dependencies between frequency bands. This spectral independence persists despite the added complexity from forced excitation, suggesting that each frequency component retains distinctive dynamic information. As with the *ambient* data, this validates the use of all PSD components without the need for dimensionality reduction, supporting the model's generalization capabilities across excitation regimes.

Figure 9 shows the temporal evolution of detected transition points T_a across sequences. In contrast to the ambient case, where the transitions clustered tightly, the distribution here is more scattered and irregular. This indicates that regime shifts under forced excitation are less temporally coherent, likely due to the localized and transient nature of train-induced loads.

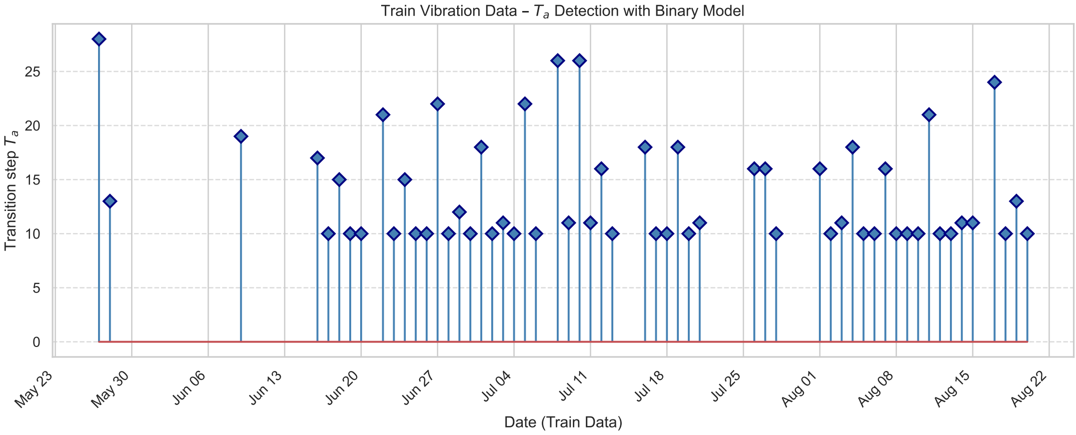


Figure 9. *Train-induced* data while retrofitting. Evolution of detected transition points T_a. Increased temporal dispersion compared with ambient excitation is observed.

The histogram in Figure 10 reinforces this interpretation, displaying a broader spread of detected transitions. Although the most frequent values still cluster between Steps 12 and 18, no single dominant mode is observed. The average transition point is approximately $T_a = 15.05$, indicating a slight delay compared with ambient excitation and highlighting the greater variability introduced by forced loading, and the overall distribution exhibited greater dispersion than in the ambient case.

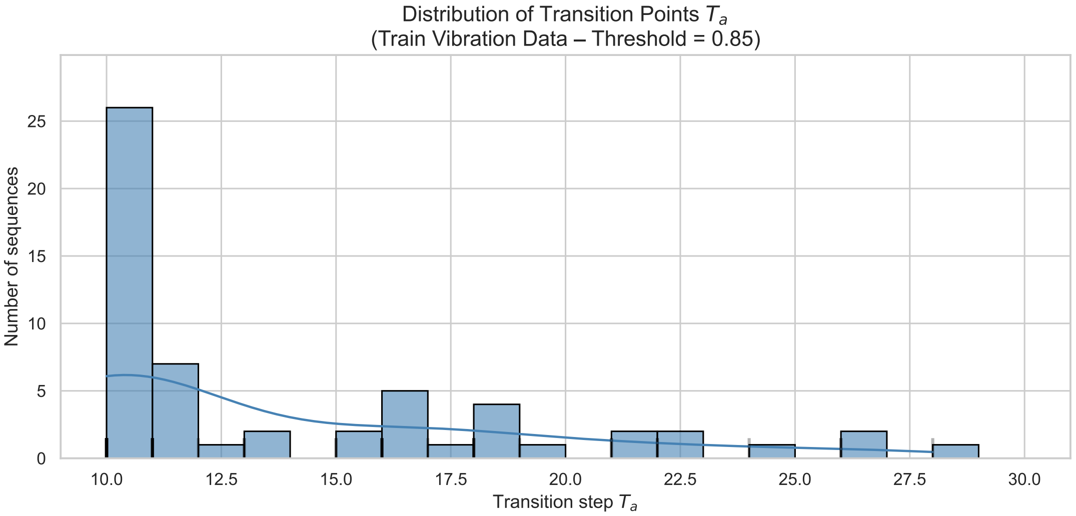


Figure 10. *Train-induced* data while retrofitting. Histogram of detected T_a values for train-induced sequences. A broader distribution and delayed mean transition are evident.

Figure 11 presents the predicted probability for the *after* class across sequences. Despite the increased variability in excitation with respect to the ambient data, the model consistently assigns high confidence values after transitions, demonstrating its robustness in capturing structural changes. Sequences with lower confidence levels remain below the threshold of 0.85 and are thus interpreted as not having undergone a full regime shift. Confidence rises and falls, reflecting more intermittent or less clearly defined transitions. It is possible to observe the maximum confidence the model assigns to each sequence when classifying it as *after* data; that is, as occurring after the structural change. This provides a way to examine how the model detects the transition over time.

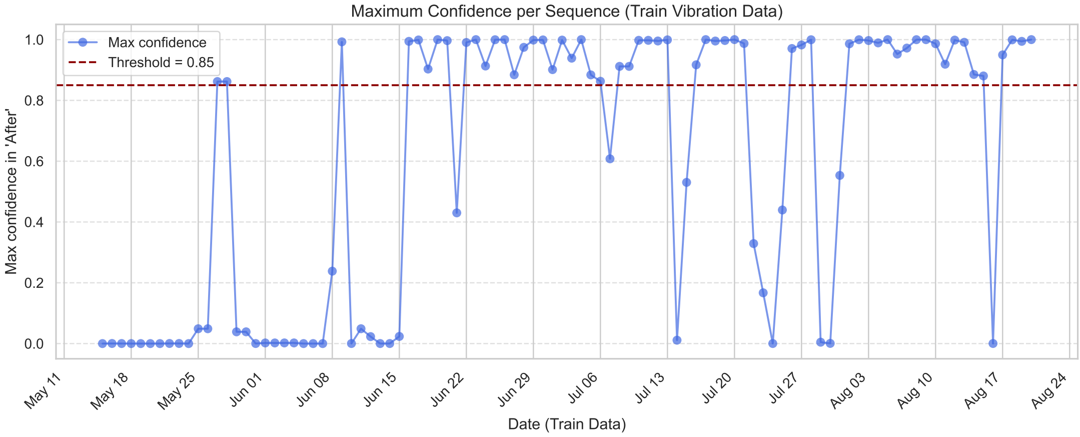


Figure 11. Train-induced data while retrofitting. Maximum prediction confidence assigned to the *after* class. High confidence levels are maintained post-transition despite forced excitation's complexity.

Temporal uncertainty is characterized in Figure 12, which shows the entropy of the model's output predictions. Local entropy spikes are observed near the detected transition points, while the baseline entropy remains low during stable periods. This pattern suggests that the model remains confident during quasi-stationary regimes and becomes uncertain only during dynamic shifts.

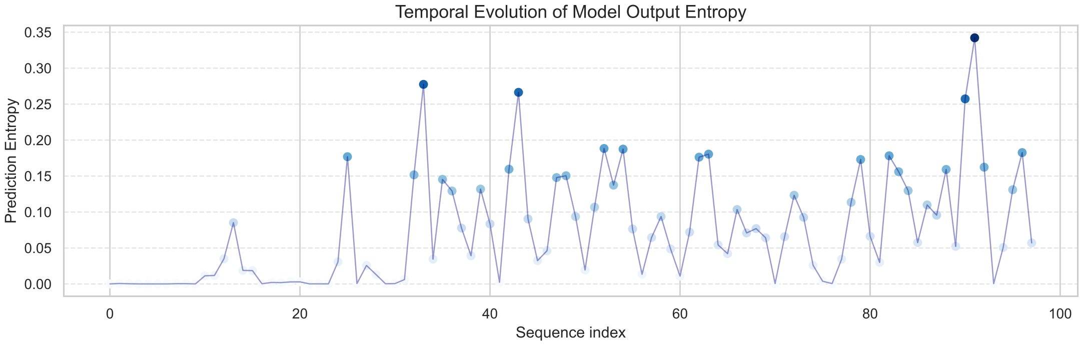


Figure 12. *Train-induced* data while retrofitting. Temporal evolution of output entropy under train-induced excitation. Localized peaks near the transitions are observed, consistent with transient uncertainty.

3.5. Comparative summary: ambient versus train-induced vibration

A comparative analysis between ambient and train-induced excitation regimes reveals distinct patterns in the transition detection behavior. Under ambient excitation, the model identifies transitions with greater temporal coherence, as evidenced by the tightly clustered T_a values around Step 10 and consistently low prediction entropy outside the transition zones. The associated confidence levels are sharply defined, and transition events manifest as abrupt shifts in the prediction heatmap, underscoring the model's ability to capture regime changes in relatively smooth, low-noise conditions.

In contrast, train-induced excitation leads to more temporally dispersed and less sharply defined transitions. The average transition point shifts to approximately Step 15.05, with a broader histogram and more gradual shifts in prediction probability. Although the PSD features remain largely decorrelated—suggesting stable input separability—the transient and spatially localized nature of train loads introduces ambiguity in both timing and confidence. Entropy spikes become more frequent, reflecting localized uncertainty near transitions, while prediction heatmaps exhibit fuzzy transitions rather than distinct boundaries. Despite these challenges, the model sustains high confidence post-transition and demonstrates reliable adaptability across both excitation contexts, highlighting its robustness and generalization capacity in dynamic monitoring scenarios.

Table 5. Comparative summary for ambient versus train-induced vibration detection.

	Ambient	Train induced
Sequences analyzed	100	100
Transitions detected	92	57
Average T_a	11.48	15.05
Mode of T_a	10	10

3.6. Prediction entropy analysis

Given the absence of explicit ground truth labels during the retrofitting phase (while), an analysis of the model's prediction entropy was conducted to assess the temporal consistency and confidence of its outputs. For each sequence, the mean prediction entropy was computed, with the entropy of a binary prediction defined as

$$H_t = -\hat{y}_t \log \hat{y}_t - (1 - \hat{y}_t) \log(1 - \hat{y}_t), \tag{3.1}$$

where H_t reaches its maximum at $\hat{y}_t = 0.5$ (maximum uncertainty) and tends toward zero as the prediction becomes more confident.

Figure 13 displays the distribution of average entropy values across sequences during the while phase for both ambient and train-induced data. Most sequences exhibit low entropy, with a mean value of 0.200 and a standard deviation of 0.160. The maximum observed entropy (0.694) indicates that even the most uncertain sequences remain far from complete ambiguity. These results confirm that the model's outputs during retrofitting are stable, structured, and meaningfully confident, rather than reflecting random or noisy fluctuations. This finding reinforces the practical utility of the framework for transition detection in partly labeled or unlabeled real-world scenarios.

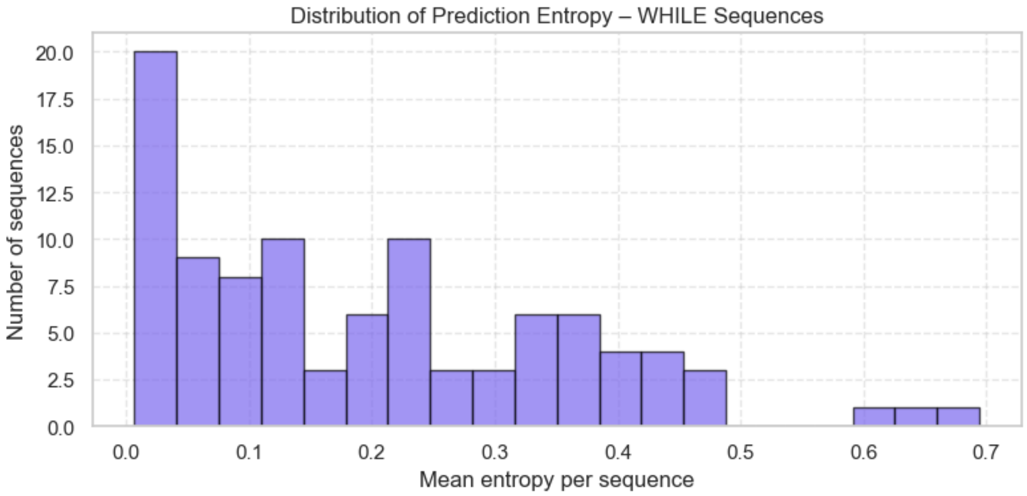


Figure 13. Distribution of average prediction entropy across sequences in the while phase. Most sequences exhibit low entropy, indicating stable and confident predictions despite the absence of ground truth labels.

3.7. The confidence variability index as a measure of temporal stability

To assess the temporal consistency of the model's predictions within each sequence, we introduce the CVI—a metric that quantifies how smoothly the predicted probabilities evolve across time steps. Specifically, the CVI is defined as the variance of the predicted class probabilities within each sequence, averaged over all sequences

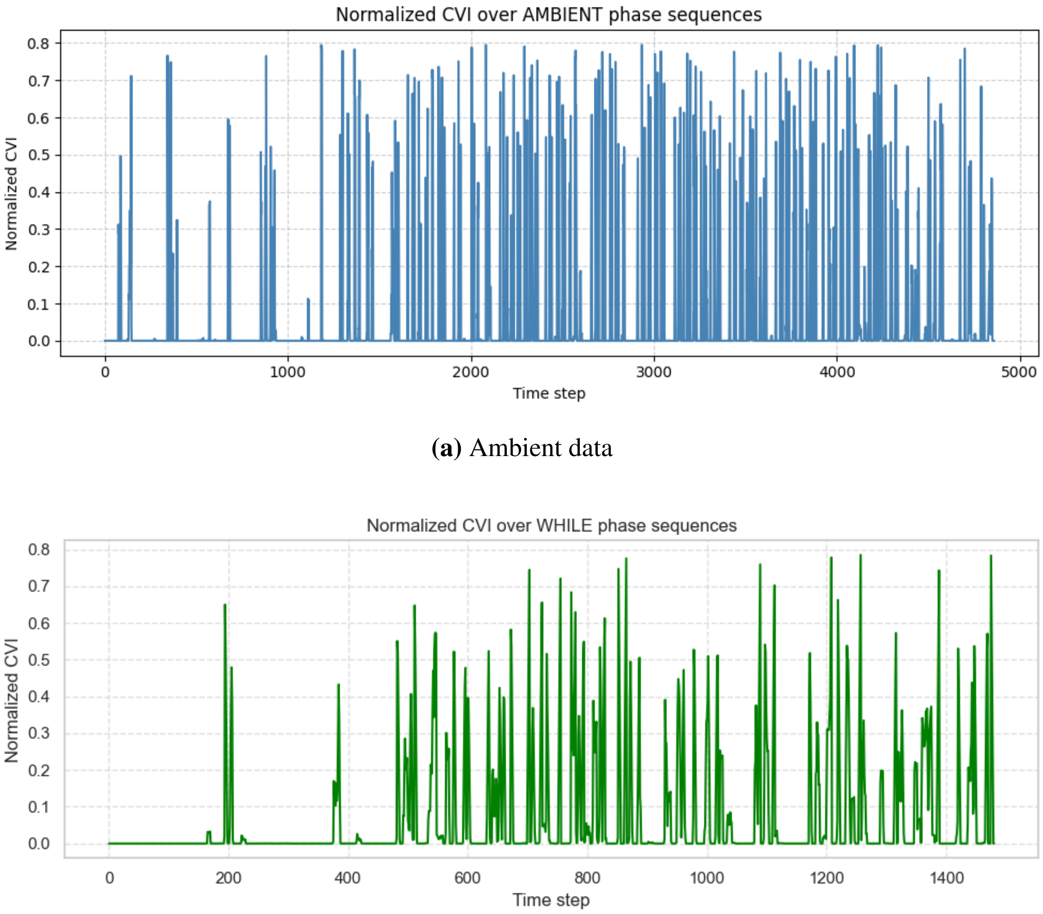
$$\mathrm{CVI} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Var}(\hat{y}_{i,1:T}), \tag{3.2}$$

where $\hat{y}_{i,1:T}$ is the sequence of predicted probabilities for instance i, spanning T time steps, and N is the number of sequences. In our case, $T = 10$, corresponding to the length of the input window used by the LSTM. Low CVI values indicate stable, temporally coherent predictions, while high values suggest internal fluctuations or uncertainty in the model's confidence.

The CVI provides valuable insight into the behavior of the model under different excitation regimes. Under ambient vibration, the model yields low CVI values (0.0204 ± 0.0519), indicating highly stable predictions. In contrast, train-induced data show significantly higher CVI values (0.0843 ± 0.0756), reflecting more fluctuating prediction patterns due to the complex, localized responses triggered by train passages.

As illustrated in Figure 14, the ambient excitation phase displays a smooth, diffuse CVI pattern, suggesting consistent prediction behavior influenced by minor environmental variations. Meanwhile, the train-induced excitation phase reveals sharp CVI peaks at specific intervals—likely corresponding to discrete retrofitting interventions or sudden dynamic changes.

The CVI complements transition detection by offering a quantitative measure of predictions' stability and a means to contextualize variability across excitation conditions. It strengthens the interpretability and robustness of the proposed framework by distinguishing between gradual environmental drift and abrupt structural changes.



(a) Ambient data

(b) Train-induced data

Figure 14. While data. Correlation matrix of PSD frequency components. Low off-diagonal values suggest diverse and complementary spectral features.

4. Discussion: interpretation, limitations, and perspectives

LSTM networks are designed to capture patterns over time and can learn dependencies across long sequences. This makes them ideal for detecting changes in signals' behavior that indicate a shift from one state (*before*) to another (*after*). To detect this transition, we treat the *before* and *after* data in the frequency domain using PSD sequences, since the while data are unlabeled. Therefore, the transition detection is framed as an inferential task.

The value of the parameter T_a, defined as the earliest time step at which the predicted probability exceeds the confidence threshold ($\theta = 0.85$), provides complementary insight into the dynamics of structural change. When $T_a = 10$—the minimum possible, as the model requires 10 consecutive PSDs to perform a prediction—this indicates an early and clear detection of a regime shift: The model confidently identifies the transition from the very first evaluable window. In contrast, higher T_a values imply that the model needs to observe more consecutive sequences before reaching sufficient confidence to confirm the change. This may result from a more gradual or ambiguous transition, or from elevated noise levels in the data. If the maximum number of evaluable windows within a given day is reached (e.g., $T_a = 37$) without exceeding the threshold, the model moves on to the next day. Therefore, a high T_a may reflect complex dynamic processes during the retrofitting phase, in which the structural transition unfolds progressively rather than abruptly. In this context, T_a functions as an inferential metric that, in real-world applications, could be compared with external information—such as retrofitting activity schedules or environmental records—to enrich the interpretation of structural behavior. When combined with other indicators such as prediction entropy or the CVI, it enables a more nuanced characterization of the transition process and the associated level of certainty in each detection.

Taken together, the metrics T_a, prediction entropy, and the CVI provide complementary perspectives on the temporal structure and certainty of the detected transition. For ambient excitation data, where transitions are detected with a mode of $T_a = 10$, both entropy and CVI values remain low, indicating that the model identifies regime shifts early and with stable, confident predictions. In contrast, under train-induced excitation—where the average T_a shifts to approximately 15.05 and the transitions are more temporally dispersed—the CVI is markedly higher (0.0843 ± 0.0756), and entropy peaks appear more frequently. This joint behavior suggests that delayed or less coherent transitions are accompanied by greater internal variability and uncertainty in the model's outputs. Although these metrics are analyzed separately, their combined interpretation reinforces the robustness of the proposed framework by offering a multi-faceted view of the transition dynamics, particularly in scenarios lacking ground truth. Such integration enables more informed interpretation and supports practical deployment in real-time SHM contexts.

Our framework can also be interpreted under the general paradigm of sequential binary classification using sliding windows, a formulation commonly adopted in time series fault detection tasks. In this approach, each input sequence of 10 PSD vectors can be viewed as a window encompassing a temporal snapshot of the structure's dynamic behavior. This setting frames the problem as a time-dependent classification task, where each window is labeled either as nondefective (*before*) or defective (*after*), on the basis of the final time step. The use of overlapping windows allows the LSTM model to learn the temporal dependencies and precursors of structural changes. This design aligns with standard practices in changepoint detection, where the goal is to learn patterns that precede or signal regime transitions. Furthermore, the sequential labeling of windows and the probabilistic interpretation of the model output enable both window-level (sequence-wise) and step-level (point-wise) detection, depending on the desired temporal granularity. In this sense, our inference of the transition point T_a can be viewed as a sequence-to-sequence classification mechanism, where the model outputs a probability of change based on the contextual temporal information encoded in each window.

While the present study focuses on SHM through vibration-based transition detection in bridges, certain parallels can be drawn with fault detection strategies developed for meshed high voltage direct current (HVdc) grids. For instance, Yousaf et al. [36–38] explored protection schemes based on deep learning and intelligent architectures to detect and localize faults in noisy high-dimensional environments. Despite the different domains, both contexts share key methodological challenges: Uncertainty during transition-like events, the need for robust real-time inference, and the absence of direct ground truth. In their work, signal preprocessing using discrete wavelet transform (DWT) is combined with neural network models (e.g., feedforward neural networks (FFNNs) or LSTMs) to identify dynamic changes, much like our use of PSD-based input features and LSTM-based temporal modeling. However, unlike fault detection in power systems, structural transition detection in SHM lacks discrete fault events and often evolves gradually, making temporal consistency metrics (e.g., entropy or the CVI) crucial for interpretability. Thus, while both approaches leverage deep learning under noisy conditions, their objectives and inference strategies remain context-specific.

Biases and assumptions in LSTM training

When training a LSTM model, there is a risk of inheriting the biases or assumptions present in the training data. These biases can affect the model's ability to generalize and yield reliable predictions. Common potential sources of bias include the following:

(1) Temporal bias: If the training data are not representative across time (for example, they are concentrated within a specific timeframe) the model may struggle to generalize to sequences from other periods. This can limit its applicability in dynamic or evolving environments. In our case, the train-induced data show very high performance in cross-validation.

(2) Class imbalance: When one class significantly outweighs another (e.g., many more instances of Class 0 than Class 1), the model may disproportionately learn to predict the dominant class, leading to poor performance on the minority class. However, this issue does not arise in our dataset, which is balanced.

(3) Feature bias: Some features may have disproportionately large influence due to their scale or frequency, causing the model to overfit to these dominant features while ignoring others that may carry important but subtler information. This issue does not arise in our case, since the model was trained on the PSDs of the accelerometers, which contain the PSDs of 100 equidistributed frequencies.

(4) Population and sampling bias: If the training data reflects only a specific subgroup or are collected in a nonrandom or biased manner, the model may not generalize to different populations or behaviors. In our case, this bias is minimized, as the dataset is well-defined physically and the sampling represents the target problem appropriately.

(5) Labeling bias: Manual labeling by human annotators often introduces subjectivity and inconsistencies, especially in cases involving ambiguous instances that may be interpreted differently by different individuals. However, this source of bias is not present in our dataset, as the before, after, and while classes are clearly defined. Nevertheless, the classification of while instances remains inferential in nature.

(6) Contextual or sequence assumptions: LSTM models inherently assume that the order of input sequences carries meaningful temporal or contextual information. If this assumption is violated, the model may learn misleading patterns based on spurious correlations. Our dataset satisfies this assumption, as the accelerometer signals and their corresponding PSDs are temporally ordered.

(7) Stationarity assumption: LSTM models implicitly assume that the statistical properties of the input data remain consistent over time—that is, the data are stationary. When the underlying data distribution shifts, the model's performance can deteriorate. Therefore, this assumption should be carefully evaluated, depending on the specific application domain. In our case, however, the prediction of the while instances employs a standard sliding window approach commonly used for detecting transitions between two distinct states (e.g., *before* vs. *after*, or healthy vs. faulty), which helps mitigate issues arising from nonstationarity.

Looking ahead, future work will explore the integration of multimodal sensing data—such as strain gauges, global navigation satellite system (GNSS) displacement measurements, and temperature sensors—to enhance the contextual understanding of structural responses under varying operational and environmental conditions. Additionally, causal inference techniques, including Granger causality and attention-based temporal models, will be investigated to identify interpretable cause–effect relationships that govern structural transitions. These methodological extensions are expected to increase both the robustness and the explanatory power of data-driven SHM frameworks, enabling deeper insight into the physical processes underlying observed vibrational patterns.

5. Conclusions

This study proposed a data-driven framework for detecting structural regime transitions based on temporal sequences of PSD vectors and LSTM networks. The approach introduces an empirical transition point, T_a, defined through probabilistic thresholding and entropy filtering, enabling temporally precise and interpretable change detection suitable for integration into SHM systems.

The model achieved high classification performance across two distinct excitation regimes—ambient and train-induced—and demonstrated its ability to identify transition periods even in the absence of explicit ground truth. These results highlight the framework's potential for real-time deployment in partially supervised SHM scenarios.

A key contribution of this work is the introduction of the CVI, which quantifies the temporal stability of the model's predictions. The CVI proved effective in distinguishing localized transition events from background fluctuations, thereby enhancing both the interpretability and reliability of the system under varying operational conditions.

Despite the promising results, two factors currently limit the scope of the proposed framework. First, the absence of ground truth during the transition phase means that the detections rely on inference, making direct validation challenging. Second, the model has been applied to a single real-world structure, so its generalization capability remains to be explored. These considerations suggest the results should be interpreted with caution and highlight the need for future studies to expand and validate the approach in more diverse contexts.

Future research could address these limitations by incorporating data from multimodal sensors and causal inference techniques. Additionally, it would be valuable to undertake a systematic comparison with traditional changepoint detection methods such as SPRT or CUSUM, including a critical assessment of their limitations in scenarios characterized by scarce ground truth and high dynamic uncertainty.

In summary, the proposed framework provides a practical, interpretable, and scalable solution for early detection of dynamic regime shifts in vibration-based SHM. While not yet diagnostic, it lays a solid foundation for next-generation monitoring systems capable of operating under real-world uncertainty and supporting long-term infrastructure resilience.

Author contributions

A. Presno Vélez, M. Z. Fernández Muñiz, and J. L. Fernández Martínez: Conceptualization, data curation, visualization, formal analysis, investigation, methodology, software, validation, supervision,

writing – original draft, writing – review and editing. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they have not used artificial intelligence (AI) tools in the creation of this article.

Conflict of interest

Prof. M. Z. Fernández Muñiz is the Guest Editor of special issue "New insights of the Application of Inverse Problems and Machine Learning in Science and Technology" for AIMS Mathematics. Prof. M. Z. Fernández Muñiz was not involved in the editorial review and the decision to publish this article. The authors declare no conflicts of interest.

References

1. W. Fan, P. Qiao, Vibration-based damage identification methods: a review and comparative study, *Struct. Health Monit.*, **10** (2010), 83–111. https://doi.org/10.1177/1475921710365419
2. S. W. Doebling, C. R. Farrar, M. B. Prime, D. W. Shevitz, *Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review*, Technical Report, Los Alamos National Laboratory Laboratory, New Mexico, 1996. https://doi.org/10.2172/249299
3. S. Hassani, U. Dackermann, A systematic review of optimization algorithms for structural health monitoring and optimal sensor placement, *Sensors*, **23** (2023), 3293. https://doi.org/10.3390/s23063293
4. T. Xin, Y. Yang, X. Zheng, J. Lin, S. Wang, P. Wang, Time series recovery using adjacent channel data based on LSTM: A case study of subway vibrations, *Appl. Sci.*, **12** (2022), 11497. https://doi.org/10.3390/app122211497
5. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
6. M. Ahmadzadeh, S. M. Zahrai, M. Bitaraf, An integrated deep neural network model combining 1D CNN and LSTM for structural health monitoring utilizing multisensor time-series data, *Struct. Health Monit.*, **24** (2025), 447–465. https://doi.org/10.1177/14759217241239041
7. V. Barzegar, S. Laflamme, C. Hu, J. Dodson, Multi-time resolution ensemble lstms for enhanced feature extraction in high-rate time series, *Sensors*, **21** (2021), 1954. https://doi.org/10.3390/s21061954
8. M. M. Eltahir, G. Aldehim, N. S. Almalki, M. M. Alnfiai, A. E. Osman, Reinforced concrete bridge damage detection using arithmetic optimization algorithm with deep feature fusion, *AIMS Math.*, **8** (2023), 29290–29306. https://doi.org/10.3934/math.20231499
9. M. G. Al-Thani, Z. Sheng, Y. Cao, Y. Yang, Traffic transformer: transformer-based framework for temporal traffic accident prediction, *AIMS Math.*, **9** (2024), 12610–12629. https://doi.org/10.3934/math.2024617

10. O. Hrizi, K. Gasmi, A. Alyami, A. Alkhalil, I. Alrashdi, A. Alqazzaz, et al., Federated and ensemble learning framework with optimized feature selection for heart disease detection, *AIMS Math.*, **10** (2025), 7290–7318. https://doi.org/10.3934/math.2025334

11. J. Hou, X. Lu, Y. Zhong, W. He, D. Zhao, F. Zhou, A comprehensive review of mechanical fault diagnosis methods based on convolutional neural network, *J. Vibroeng.*, **26** (2023), 44–65. https://doi.org/10.21595/jve.2023.23391

12. E. L. Lydia, C. Santhaiah, M. Altafahmed, K. V. Kumar, G. P. Joshi, W. Cho, An equilibrium optimizer with deep recurrent neural networks enabled intrusion detection in secure cyber-physical systems, *AIMS Math.*, **9** (2024), 11718–11734. https://doi.org/10.3934/math.2024574

13. W. Liu, Z. Tang, F. Lv, X. Chen, An efficient approach for guided wave structural monitoring of switch rails via deep convolutional neural network-based transfer learning, *Meas. Sci. Technol.*, **34** (2022), 024004. https://doi.org/10.1088/1361-6501/ac9ad3

14. W. Liu, S. Wang, Z. Yin, Z. Tang, Structural damage detection of switch rails using deep learning, *NDT E Int.*, **147** (2024), 103205. https://doi.org/10.1016/j.ndteint.2024.103205

15. W. Liu, J. Hu, F. Lv, Z. Tang, A new method for long-term temperature compensation of structural health monitoring by ultrasonic guided wave, *Measurement*, **252** (2025), 117310. https://doi.org/10.1016/j.measurement.2025.117310

16. W. Liu, Z. Tang, F. Lv, X. Chen, Multi-feature integration and machine learning for guided wave structural health monitoring: Application to switch rail foot, *Struct. Health Monit.*, **20** (2021), 2013–2034. https://doi.org/10.1177/1475921721989577

17. F. Deng, X. Tao, P. Wei, S. Wei, A robust deep learning-based damage identification approach for SHM considering missing data, *Appl. Sci.*, **13** (2023), 5421. https://doi.org/10.3390/app13095421

18. P. Ba, S. Zhu, H. Chai, C. Liu, P. Wu, L. Qi, Structural monitoring data repair based on a long short-term memory neural network, *Sci. Rep.*, **14** (2024), 9974. https://doi.org/10.1038/s41598-024-60196-2

19. X. Wang, J. Wu, C. Liu, H. Wang, Y. Du, W. Niu, Exploring LSTM based recurrent neural network for failure time series prediction, *J. Beijing Univ. Aeronaut. Astron.*, **44** (2018), 772–784. https://doi.org/10.13700/J.BH.1001-5965.2017.0285

20. A. Vaishnavi, A. Sharma, V. P. S. Naidu, Fault detection in machine bearings using deep learning, *SAE Tech. Pap.*, **26** (2024), 0473. https://doi.org/10.4271/2024-26-0473

21. A. P. Vélez, M. Z. F. Muñiz, J. L. F. Martínez, Enhancing structural health monitoring with machine learning for accurate prediction of retrofitting effects, *AIMS Math.*, **9** (2024), 30493–30514. https://doi.org/10.3934/math.20241472

22. A. P. Vélez, M. Z. F. Muñiz, J. L. F. Martínez, Advancing structural health monitoring with deep belief network-based classification, *Mathematics*, **13** (2025), 1435. https://doi.org/10.3390/math13091435

23. B. Peeters, G. De Roeck, Stochastic system identification for operational modal analysis: a review, *J. Dyn. Syst. Meas. Control*, **123** (2001), 659–667. https://doi.org/10.1115/1.1410370

24. J. M. W. Brownjohn, Structural health monitoring of civil infrastructure, *Phil. Trans. R. Soc. A*, **365** (2007), 589–622. https://doi.org/10.1098/rsta.2006.1925

25. C. R. Farrar, K. Worden, *Structural health monitoring: a machine learning perspective*, John Wiley & Sons, Ltd, 2012. https://doi.org/10.1002/9781118443118

26. H. Li, J. Ou, The state of the art in structural health monitoring of cable-stayed bridges, *J. Civil. Struct. Health Monit.*, **6** (2016), 43–67. https://doi.org/10.1007/s13349-015-0115-x

27. T. Poggio, H. Mhaskar, L. Rosasco, B. Miranda, Q. Liao, Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review, *Int. J. Autom. Comput.*, **14** (2017) 503–519. https://doi.org/10.1007/s11633-017-1054-2

28. A. Jentzen, B. Kuckuck, P. von Wurstemberger, Mathematical introduction to deep learning: methods, implementations and theory, *arXiv*, 2023. https://doi.org/10.48550/arXiv.2310.20360

29. P. L. Bartlett, S. Mendelson, Rademacher and gaussian complexities: risk bounds and structural results, *J. Mach. Learn. Res.*, **3** (2002), 463–482.

30. M. Anthony, P. L. Bartlett, *Neural network learning: theoretical foundations*, Cambridge University Press, 2009.

31. D. P. Kingma, J. L. Ba, Adam: a method for stochastic optimization, *arXiv*, 2017. https://doi.org/10.48550/arXiv.1412.6980

32. D. Siegmund, *Sequential analysis: tests and confidence intervals*, Springer, 1985. https://doi.org/10.1007/978-1-4757-1862-1

33. D. V. Hinkley, Inference about the change-point from cumulative sum tests, *Biometrika*, **58** (1971), 509–523. https://doi.org/10.1093/BIOMET/58.3.509

34. R. P. Adams, D. J. C. MacKay, Bayesian online changepoint detection, *arXiv*, 2007. https://doi.org/10.48550/arXiv.0710.3742

35. M. Altamirano, F. X. Briol, J. Knoblauch, Robust and scalable Bayesian online changepoint detection, *arXiv*, 2023. https://doi.org/10.48550/arXiv.2302.04759

36. M. Z. Yousaf, S. Khalid, M. F. Tahir, A. Tzes, A. Raza, A novel dc fault protection scheme based on intelligent network for meshed dc grids, *Int. J. Electr. Power Energy Syst.*, **154** (2023), 109423. https://doi.org/10.1016/j.ijepes.2023.109423 .

37. M. Z. Yousaf, H. Liu, A. Raza, A. Mustafa, Deep learning-based robust dc fault protection scheme for meshed HVDC grids, *CSEE J. Power Energy Syst.*, **9** (2023), 2423–2434. https://doi.org/10.17775/CSEEJPES.2021.03550

38. M. Z. Yousaf, M. F. Tahir, A. Raza, M. A. Khan, F. Badshah, Intelligent sensors for dc fault location scheme based on optimized intelligent architecture for HVdc systems, *Sensors*, **22** (2022), 9936. https://doi.org/10.3390/s22249936

Appendix

The updated equations of the Adam algorithm ([31]) are as follows.

a. First moment estimate

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\nabla\mathcal{L}_t, \tag{A.1}$$

where m_t is the exponentially weighted average of the gradient, β_1 (typically 0.9) controls how much past gradients influence the estimate and $\nabla\mathcal{L}_t$ is the gradient of the loss at step t.

b. Second moment estimate

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)(\nabla \mathcal{L}_t)^2, \tag{A.2}$$

where v_t is the exponentially weighted average of the squared gradients and β_2 (typically 0.999) smooths the estimate of the gradient variance.

c. Bias correction

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t}. \tag{A.3}$$

At the beginning, m_t and v_t are biased toward 0, so bias correction compensates for this, especially in the early steps.

d. Parameter update

$$\theta_{t+1} = \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon}, \tag{A.4}$$

where θ_t are the model parameters at time t and α is the learning rate. The update normalizes the gradient mean by the square root of its variance, making it adaptive for each parameter, while ε (typically 10^{-8}) prevents division by zero.

To avoid overfitting, early stopping halts training if the validation loss does not improve after a set number of epochs.