

Edge/Cloud hybrid architecture for time-domain Structural Health Monitoring transfer learning

Ivan ARAKISTAIN¹, Stergios MITOULIS², Konstantinos BANITSAS³, Sotirios ARGYROUDIS³, Jose Carlos JIMENEZ¹, Eric LOPEZ-VILLARRAGUT¹, David GARCIA-SANCHEZ¹

¹TECNALIA Basque Research and Technology Alliance “BRTA”, Derio, Spain,
ivan.arakistain@tecnalia.com

²The Bartlett School of Sustainable Construction, University College London, London, UK.

³Brunel, University of London, London, UK.

Abstract. Current Structural Health Monitoring (SHM) systems remain constrained by their reliance on handcrafted feature extraction and centralized cloud processing, limiting their real-time performance, scalability, and deployment on resource-constrained infrastructure. This study seeks to overcome these limitations by introducing a hybrid edge/cloud SHM framework that facilitates low-power, real-time anomaly detection directly on embedded devices. We introduce a temporal transfer learning approach based on the WaveNet and Recurrent Neural Network architectures that operates on raw time-series data and eliminates the need for statistical or frequency-domain feature engineering. The models were trained and validated using real-world datasets from the Z24 (Switzerland) and S101 (Austria) bridges and deployed on edge hardware (Raspberry Pi 5 and LattePanda Mu, Intel® N100) for on-device inferences.

The framework implements a hierarchical inference strategy in which edge devices perform continuous, low-latency anomaly detection, whereas cloud resources are selectively engaged for the high-precision analysis of critical events. This design reduces bandwidth requirements and enhances system autonomy without compromising accuracy. The results demonstrate that the proposed approach achieves a high anomaly detection performance in time-domain transfer learning while maintaining an efficient edge deployment. This study provides a validated pathway toward scalable, real-time, and feature-free SHM systems for deployment in operational bridge networks, supporting continuous monitoring, early damage detection, and maintenance decision-making in the future.

Keywords: SHM, edge computing, hybrid architecture, transfer learning, anomaly detection.

1. Introduction

Transfer learning (TL) has become one of the most active research directions in Structural Health Monitoring (SHM) [1] because it directly addresses one of the field’s biggest challenges: the scarcity of labelled damage data and the difficulty in collecting comparable datasets across structures, environments, and sensing conditions [2]. In addition, hybrid edge and cloud processing systems are gaining popularity to obtain the best of both approaches. Edge computing enables faster, more efficient, and reliable analyses of sensor data at the source. In contrast, in cloud systems, it is possible



to perform multiple tasks, including predictive and regression tasks, with generative or agentic AI, whether creating or fine-tuning a model [3].

Over the last four years, numerous attempts have been made to address centralized cloud processing. In a study by Okey et al. [4], which focused on cybersecurity, the authors examined a hybrid architecture in which information on network attacks against IoT devices was collected and stored in a database accessible by a data center. This study uses pretrained convolutional neural network (CNN) models and trains them on intrusion datasets to create an efficient and lightweight ensemble model capable of detecting network attacks in Cloud IoT environments.

Nan Huang et al. [5] used ensemble learning and reinforcement learning for SHM. Their research focused on detecting specific deviations in acceleration time-series data derived from a long-span bridge in China. The architecture proposed in this study is based on the strategic fusion of several advanced techniques.

In 2023, Kim et al. [6] presented an approach for SHM using one-dimensional CNNs for anomaly detection in bridge structures. The effectiveness of this methodology was tested using simulated and experimental datasets obtained from the reference SHM system. The data used consisted of raw acceleration signals captured by sensors installed on bridge structures that underwent progressive damage due to environmental vibrations. The main task of the experiment was anomaly detection using CNNs to automatically extract and learn significant features from raw acceleration signals, thereby eliminating the need for manual feature extraction.

In many cases, obtaining labelled or simulated data is impossible. As such, the study by Bayane and Leander [7] investigated and proposed an unsupervised machine learning methodology for real-time bridge damage detection. The primary motivation for their work was to address the limitations of previous supervised learning approaches, which, although demonstrated exemplary performance, often relied on bridge behaviour databases generated by finite element simulations (FEM) or laboratory tests.

In 2024, Dabbous et al. [8] drew relevant conclusions. The authors focused on the application of two CNN architectures that have proven efficient for feature extraction in time series: Waveform modelling neural Network (WaveNet) and the Minimally Random Convolutional Kernel Transformer (MiniRocket). To do this, they used a dataset extracted from the Z24 bridge using 291 sensors distributed along the bridge, enabling them to identify 17 scenarios, 15 of which were progressive damage scenarios, spanning a time series of 65,536 samples at a frequency of 100 Hz. These models were subsequently implemented and tested on two edge devices: Raspberry Pi 4 Model B and Jetson AGX Xavier.

Despite the promise of these studies in feature extraction and anomaly detection, a common shortcoming is their limitation to a single structure. Our research, therefore, addresses the question: Can similar results be replicated using time-domain transfer learning for a different structure? With this objective in mind, this study focused on applying transfer learning to the S101 bridge dataset.

2. Methods

2.1 Proposed architecture

The proposed architecture of the dual Edge/Cloud inference hybrid system is composed of wireless accelerometer sensor nodes or data loggers, a Single Board Computer (SBC) running a lightweight Edge inference model, and a cloud infrastructure running a higher-capacity inference model, as illustrated in Figure 1. To demonstrate the applicability of this hybrid Edge/Cloud architecture and evaluate its transfer-learning capability under real structural conditions, the framework was

implemented and tested using two benchmark bridge datasets, namely Z24 and S101, which are described in the following sections.

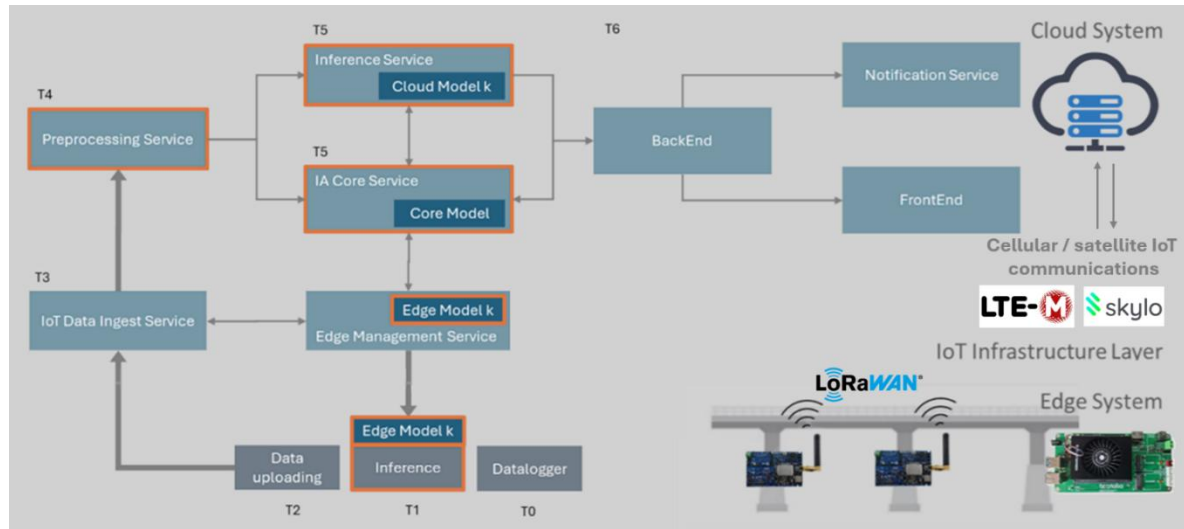


Fig. 1. General architecture of the dual-inference hybrid system.

2.2 Datasets from Z24 and S101 bridges

The Z24 bridge, located in the canton of Bern, Switzerland, near Solothurn, was built in 1963 and stood over the A1 motorway connecting Bern and Zürich. It was a prestressed concrete bridge with two box cells, featuring a main span of 30 m, two side spans of 14 m each, a width of 8.6 m, and a height of 1.01 m. The structure, as depicted in Figure 2, comprised two concrete pillars integrated into the beam. In 1998, the bridge was demolished to allow the motorway to be widened.

Mode 1: 3.860 Hz

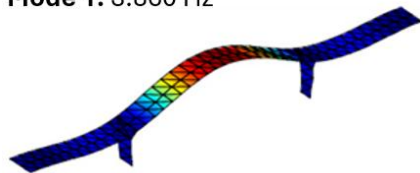


Fig. 2. 3D representation and modal analysis of Z24 bridge.

The Z24 bridge dataset originated from an extensive monitoring campaign conducted before and during the demolition. The main objective of this monitoring was to assess the feasibility of vibration-based SHM for large-scale civil structures and to understand the correlation between the measured vibrations and the overall condition of the bridge. The dataset is considered a "reference case study" in the scientific community and is widely used for SHM research because of its comprehensive and labelled nature.

The artificially induced damage scenarios on the Z24 bridge during the PDT varied and included:

- Progressive lowering of the Koppigen pier (20, 40, 80 and 95 mm).
- Elevation of the Koppigen pier.
- Concrete spalling on the underside (12 m², 24 m²).
- 1-metre landslide on the abutment.
- Failure of the concrete ball joint.

- Failure of 2 or 4 anchor heads.
- Breakage of 2, 4, or 6 of the 16 tendons.

For data collection, the PDT used 291 uniaxial accelerometer sensors (FBA 11 and 23 types) distributed across the bridge structure. The bridge was segmented into nine sections to maximize the sensor coverage and sensitivity at critical points. Each sensor provided a time series of 65,536 steps at 100 Hz. In addition to the vibration data, environmental parameters were collected concurrently with the SHM accelerometers. The Z24 bridge dataset is widely available and frequently used to validate damage detection methods in SHM.



Fig. 3. S101 bridge and modal analysis.

The dataset for the S101 bridge, depicted in Figure 3, originates from the work of Moughty et al. [9]. The S101 bridge, located near Vienna (Austria), was a three-span posttensioned overpass built in the 1960s. Its main span measured 32 m and had two side spans of 12 m each. The progressive damage test was conducted over three days (10-13 December) in 2008, just before the bridge was demolished.

2.3 Data preparation and model training

For both structures, the Z-axis acceleration data was selected. To remove irrelevant high-frequency noise, a low-pass filter was applied, discarding frequencies above 100 Hz. This basic signal conditioning is distinct from traditional handcrafted feature engineering, as the model operates directly on the filtered raw time series. Anomalies were defined based on the documented progressive damage scenarios (e.g., pier lowering, tendon breakage) for Z24, and similar documented damage states for S101, which served as ground truth for classification. Next, a fragmentation strategy was applied to adjust the sample length, for which different approaches were tested such as

- Overlap with repetition filling.
- Filling with sequence repetition (Best performance).

The sequential header with a bidirectional GRU architecture begins by adapting the base model's output via a Reshape operation, which converts the flat vector of 16 neurons into a sequence with one timestep and 16 features. This transformation enables the use of sequential layers, such as Conv1D and GRU, even when the original input is not a time series.

Once reshaped, a Conv1D layer with 64 filters and ReLU activation was applied. This layer projects and enriches the vector's characteristics, producing 64 enhanced feature representations that serve as inputs for the recurrent layers. The architecture then incorporates two bidirectional GRU layers: the first uses 32 units with *return_sequences=True* to preserve the entire sequence. In contrast, the second uses 16 units with *return_sequences=False* to compress the temporal information into a compact feature vector. This configuration captures dependencies in both the forward and backward directions, producing a comprehensive global representation.

Following the sequential summarization, a dense layer with 32 neurons and ReLU activation refined the learned representation, and a 30% Dropout layer improved generalization by reducing overfitting. Finally, a dense output layer with two neurons and Softmax activation converts the resulting feature vector into binary classification probabilities, enabling the system to adapt the base model's learned knowledge to new structural domains or varying operational conditions.

2.4 Model evaluation

The original model was evaluated using four types of fragmentation and the signal repetition strategy yielded the best results (table 1).

Table 1. Metrics of the model with different fragmentation mechanisms.

Strategy	Accuracy	Precision	Recall	F1-Score
Overlap	0.435374	0.795920	0.435274	0.398074
Repeat (chunks)	0.778912	0.793715	0.778912	0.778403

Several observations emerged from the model evaluation. Although bridge vibration anomalies were characterized over 10-minute time ranges, fragmentation into two halves yielded the best results. Additionally, Batch Normalization and Dropout were found to improve accuracy. While an eight-neuron dense layer showed good results, it proved unstable; 32 neurons offered better performance but at a higher computational cost. Finally, it was noted that model performance decreased as bridge degradation increased. Figure 4 shows an F1-Score comparison by configuration and architecture.

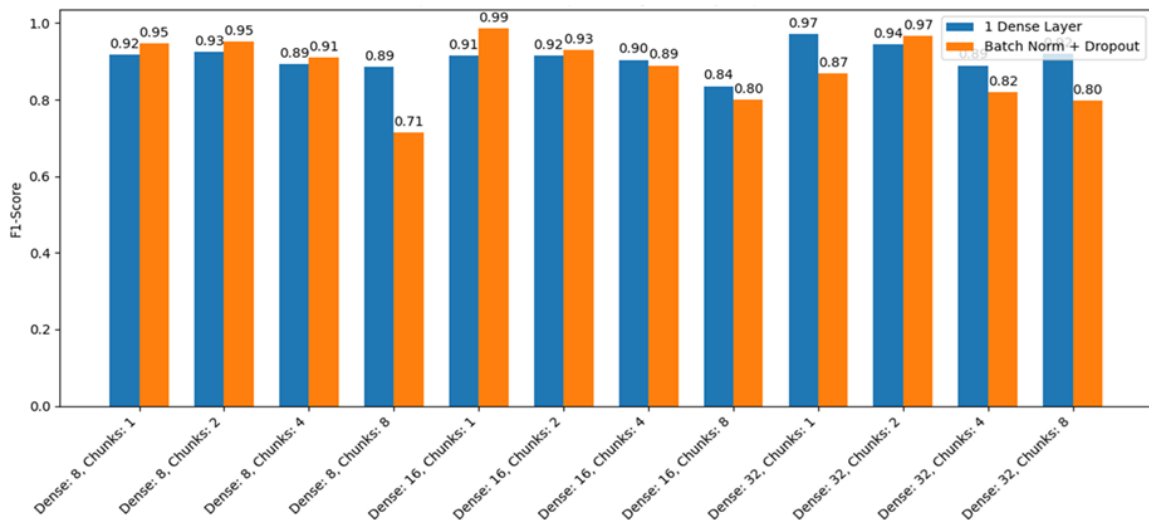


Fig. 4. F1-Score comparison by configuration and architecture.

The architectures of the new layers that act as headers for transfer learning on the binary base model are discussed below.

Sequential header with bidirectional GRU

This header begins by adapting the base model's output using a Reshape layer, followed by a Conv1D projection to enrich the features. Next, two bidirectional GRU layers are used to capture temporal relationships in both directions. After sequential summarization, the representation is refined with a dense layer, and dropout is applied to improve generalization. This header allows the knowledge of the original model to be adapted to new structural domains and different operational conditions.

This architecture uses only dense layers to refine the representation extracted from a base model. It begins with a larger dense layer with ReLU activation, followed by dropout to reduce overfitting. Next, another dense layer with half the number of neurons is added, accompanied by a second dropout. This pattern facilitates progressive feature compression, which is particularly advantageous when direct feature adaptation is required without the introduction of temporal dependencies. It ends with a dense layer with two neurons and Softmax activation for binary classification.

Attention-based header (Transformer)

This architecture incorporates a multi-head attention block (*MultiHeadAttention*) to capture the internal relationships between features, even if there is no explicit temporal sequence. After adapting the shape of the input with Reshape, self-attention is applied, and the residual connection is added. Layer normalisation stabilizes the output before using a global pooling operation. It is completed with a dense layer and dropout, followed by a Softmax output. This header is especially useful when one wants to extract cross-dependencies between multiple dimensions of the input vector.

Temporal Convolutional Header (Temporal Conv)

This architecture uses cascading 1D convolutions to transform the input vector after an initial Reshape. First, a 64-filter convolution is applied, followed by a 32-filter convolution, both with a kernel size of 1. This allows the feature space to be projected without mixing temporal positions. Next, a *GlobalAveragePooling1D* collapses the temporal dimension, producing a compact vector that is then passed through a dense layer with ReLU and Dropout. This architecture is suitable for tasks where the input is not a real sequence, but one may want to exploit the local relationships between vector dimensions.

Sequential header with bidirectional LSTM

In this architecture, a bidirectional LSTM is the main component, with the input first transformed into a sequence via Reshape. The bidirectional LSTM captures both forward and backward temporal relationships. Then, a dense layer of 32 neurons transforms the sequential output into a more abstract representation, followed by a dropout layer for regularization. It is an alternative to the GRU layer for tasks that assume longer or more complex temporal dependencies because the LSTM can retain memory over longer time frames.

3. Results

3.1 Edge inferencing hardware

A custom Edge Computing printed circuit board was developed in a 6-layer PCB with impedance control as shown in Figure 5. In addition to the host processor, the board features a triple-input power supply with a dedicated input for a solar-powered 12VDC battery source.



Fig. 5. A custom made (Tecnalia) Edge Computing board with LattePanda MU processor board

3.2 Model benchmarking

The three best models achieved based on WaveNet reached high performance metrics, as shown in Figure 6, with F1-Scores above 0.85 in all cases and peaks close to 0.95. A comparison of the execution time of the same dataset was also performed using the same three models on both edge devices (Raspberry Pi 5 and LattePanda MUN100), considering that the processing speed of the LattePanda is almost 100% faster than that of the Raspberry Pi 5, with 0.066 ms per sample for the LattePanda, compared to 0.13 ms per sample for the Raspberry Pi in one of the models.



Fig. 6. Performance benchmarking of three models

However, Recurrent networks proved to provide the best performance in comparison to the Wavenet architecture. A benchmarking test was conducted, revealing that the accuracy of the Bidirectional Long Short-Term Memory (BiLSTM) slightly surpasses that of the Bidirectional Gated Recurrent Unit (BiGRU). The BiGRU model achieved an accuracy of 0.767, whereas the BiLSTM model attained a higher accuracy of 0.789.

While different network densities (8, 16, and 32 neurons) were analyzed, BiGRU with a Dense 16-neuron-based model 3 stood out as the model with the best F1-Score (86%), and its performance on LattePanda was >110% better than that of RBpi5. There was a slight difference (<4%) in the performances of the TF and TF-Lite models. Finally, a frontend graphical interface tool was developed, where users could compare the different pretrained models, as shown in Figure 7.

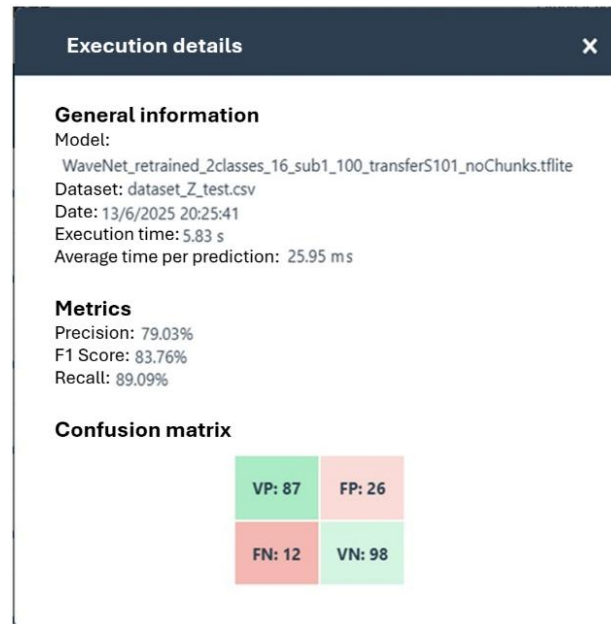


Fig. 7. Execution details as depicted in the frontend of the developed tool

4. Discussion

The results of this study demonstrate that knowledge transfer between real bridges in the time domain is both feasible and effective. Lightweight inference models were successfully executed on edge devices, achieving promising performances and inference times suitable for real-time applications. Additionally, this study validated the potential of a hybrid architecture that combines local and cloud-based inference within an integrated IoT framework. By leveraging pretrained neural networks, the need for complex preprocessing steps was eliminated, simplifying the deployment pipeline. These findings further confirm that transfer learning in the temporal domain can be successfully applied to SHM tasks on real bridges. Overall, the hybrid edge/cloud architecture provides a compelling approach for improving the accuracy, efficiency, and scalability of SHM.

This study establishes a solid foundation for future SHM systems that enhance predictive maintenance and structural safety. Future work should focus on validating the proposed framework across a broader range of bridges, sensing configurations, and environmental conditions to improve its generalizability. In practical terms, the hybrid Edge/Cloud architecture can support real-time anomaly screening and predictive maintenance in operational SHM systems, and the same concept may also be adapted to other infrastructure assets such as tunnels, buildings, wind turbines, and port structures.

Acknowledgements

This contribution was made in cooperation with Tecnia, University College London (UCL), and Brunel University of London in the framework of the ReCharged project, supported by the European Union HORIZON-MSCA-2021-SE-01, grant agreement No: 101086413, and the UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee (EP/Y003586/1, EP/X037665/1). The authors would also like to acknowledge the Horizon Europe (HE) funding program of the FUTURAL (GA 101083958) and LIAISON (GA 101103698) projects and NEXTPCB for sponsoring PCB manufacturing.

References

- [1] Yano, M. O., Figueiredo, E., da Silva, S., & Cury, A. (2023). Foundations and applicability of transfer learning for structural health monitoring of Bridges. *Mechanical Systems and Signal Processing*, 204, 110766. <https://doi.org/10.1016/j.ymssp.2023.110766>
- [2] Ferreira, L., Yano, M. O., Souza, L., Moldovan, I., da Silva, S., Lopes, R., Cimini, C. A., Costa, J. C. W. A., & Figueiredo, E. (2025). Transfer learning and bayesian calibration addressing data scarcity and uncertainty for structural health monitoring of Twin Concrete Bridges. *Mechanical Systems and Signal Processing*, 235, 112845. <https://doi.org/10.1016/j.ymssp.2025.112845>
- [3] Sharma, A., & Bhowmik, B. (2026). Autonomous agentic AI with policy adaptation for physics-informed spectral learning in Structural Health Monitoring. *Advanced Engineering Informatics*, 70, 104224. <https://doi.org/10.1016/j.aei.2025.104224>
- [4] Okey, O. D., Melgarejo, D. C., Saadi, M., Rosa, R. L., Kleinschmidt, J. H., & Rodriguez, D. Z. (2023). Transfer learning approach to ids on cloud IOT devices using optimized CNN. *IEEE Access*, 11, 1023–1038. <https://doi.org/10.1109/access.2022.3233775>
- [5] Huang, N. (2024). Anomaly detection in structural health monitoring with Ensemble Learning and Reinforcement Learning. *International Journal of Advanced Computer Science and Applications*, 15(1). <https://doi.org/10.14569/ijacsa.2024.0150112>
- [6] Kim, S.-Y., & Mukhiddinov, M. (2023). Data Anomaly Detection for structural health monitoring based on a convolutional neural network. *Sensors*, 23(20), 8525. <https://doi.org/10.3390/s23208525>
- [7] Bayane, I., Leander, J., & Karoumi, R. (2024). An unsupervised machine learning approach for real-time damage detection in bridges. *Engineering Structures*, 308, 117971. <https://doi.org/10.1016/j.engstruct.2024.117971>
- [8] Dabbous, A., Berta, R., Fresta, M., Ballout, H., Lazzaroni, L., & Bellotti, F. (2024). Bringing intelligence to the edge for Structural Health Monitoring: The case study of the Z24 bridge. *IEEE Open Journal of the Industrial Electronics Society*, 5, 781–794. <https://doi.org/10.1109/ojies.2024.3434341>
- [9] Moughty, J. J., & Casas, J. R. (2017). Performance assessment of vibration parameters as damage indicators for bridge structures under ambient excitation. *Procedia Engineering*, 199, 1970–1975. <https://doi.org/10.1016/j.proeng.2017.09.306>