buildings

MDPI

Article

Structural Damage Identification Using Autoencoders: A Comparative Study

Marcos Spínola Neto [1], Rafaelle Finotti [2], Flávio Barbosa [2] and Alexandre Cury [2,*]

[1] Faculty of Engineering, University of Juiz de Fora, Juiz de Fora 36036-900, MG, Brazil; marcos.neto@engenharia.ufjf.br

[2] Graduate Program in Civil Engineering, Faculty of Engineering, University of Juiz de Fora, Juiz de Fora 36036-900, MG, Brazil; rafaelle.finotti@engenharia.ufjf.br (R.F.); flavio.barbosa@ufjf.br (F.B.)

* Correspondence: alexandre.cury@ufjf.br

Abstract: Structural health monitoring (SHM) ensures the safety and reliability of civil infrastructure. Autoencoders, as unsupervised learning models, offer promise for SHM by learning data features and reducing dimensionality. However, comprehensive studies comparing autoencoder models in SHM are scarce. This study investigates the effectiveness of four autoencoder-based methodologies, combined with Hotelling's T^2 statistical tool, to detect and quantify structural changes in three civil engineering structures. The methodologies are evaluated based on computational costs and their abilities to identify structural anomalies accurately. Signals from the structures, collected by accelerometers, feed the autoencoders for unsupervised classification. The latent layer values of the autoencoders are used as parameters in Hotelling's T^2, and results are compared between classes to assess structural changes. Average execution times of each model were calculated for computational efficiency. Despite variations, computational cost did not hinder any methodology. The study demonstrates that the best fitting model, VAE-T^2, outperforms its counterparts in identifying and quantifying structural changes. While the AE, SAE, and CAE models showed limitations in quantifying changes, they remain relevant for detecting anomalies. Continuous application and development of these techniques contribute to SHM advancements, enabling the increased safety, cost-effectiveness, and long-term durability of civil engineering structures.

Keywords: structural health monitoring; damage detection; autoencoders; sparse; variational; convolutional; Hotelling; benchmark

Citation: Spínola Neto, M.; Finotti, R.; Barbosa, F.; Cury, A. Structural Damage Identification Using Autoencoders: A Comparative Study. *Buildings* **2024**, *14*, 2014. https://doi.org/10.3390/buildings14072014

Academic Editor: Eric M. Lui

Received: 27 May 2024
Revised: 25 June 2024
Accepted: 28 June 2024
Published: 2 July 2024

1. Introduction

Among the main concerns of civil engineers throughout the construction's life cycle, the proper functioning of structural systems and the safety of users stand out. To minimize structural problems caused by wear and tear, aging, climate change, or misuse, continuous monitoring of the structure's condition is important, especially for detecting any anomalies at early stages. This is particularly relevant when dealing with sensitive structures such as large buildings, bridges, viaducts, and geotechnical constructions.

Considering that human inspections conducted visually can be susceptible to inaccuracies and errors, structural health monitoring (SHM) has emerged as a potential approach for the early identification of faults in structures [1]. This methodology primarily relies on the use of accelerometers connected to structures to record vibration data over time. With significant advancements in machine learning (ML) and Artificial Intelligence (AI), these data have become not only more accurate, but also more accessible, enabling their widespread application [2]. As a result, SHM systems offer a more reliable and cost-effective approach to structural maintenance, allowing for the remote detection of deterioration signals based on predefined thresholds, which can significantly reduce repair costs [3].

The implementation of SHM can follow supervised or unsupervised approaches, with the latter being preferable since, in real structures, only the current state of the structure is usually known, without other reference structural behaviors being available. Therefore, it is recommended to define a robust and unsupervised model for effective structural damage assessment in actual constructions.

According to Avci et al. [4], among the levels of damage identification defined by Rytter [5], damage detection is considered the most critical component of SHM. This process, defined as the systematic and automatic identification of the existence of damage, includes the subsequent localization and evaluation of its severity. This perspective is reinforced by recent articles focused on damage detection [6–10], as the localization and quantification of damage presuppose the correct identification of its occurrence.

In this context, Eltouny and Liang [11] proposed a density-based unsupervised learning strategy for detecting structural deteriorations. This method was the first to utilize cumulative intensity measurements to extract damage-sensitive features from its principal components. Additionally, the authors introduced a statistical model based on Kernel Density Maximum Entropy (KDME) and Bayesian optimization, demonstrating that this approach is useful for determining damage in post-seismic events and verifying the effectiveness of unsupervised models, where there is no prior knowledge about the structure's response to damage.

There are also approaches based on analyzing images of visible surfaces of structures through computer vision using Convolutional Neural Networks (CNNs) [12,13]. Although these approaches assist in complementing human visual inspection and reducing costs [14], they have the limitation of not allowing the detection of internal alterations, which are externally invisible, negatively impacting operations.

Among the different methodologies for identifying structural damage, classical modal data analysis is traditionally adopted [15–17]. The basic premise of these methodologies is that the structure's degradation process alters its physical properties, such as mass and stiffness, influencing its natural frequencies, mode shapes, and damping ratios. However, factors like temperature variations and load changes can also alter the structural dynamic characteristics, capable of impairing damage detection by these methods, thus requiring the development of strategies to overcome these limitations [18–22].

Alternatively, methods that directly utilize vibrational raw data are becoming more dominant [4]. These approaches can detect structural alterations by processing acquired signals from the structure over time, extracting relevant features and making classifications with relatively low computational costs [23,24]. An autoencoder is an unsupervised learning model designed to reconstruct input data and reduce dimensionality. It compresses information into a compact latent representation and then reconstructs the original data from this reduced form. Composed of an encoder and a decoder, it minimizes the difference between the input and the reconstructed output. There are specific variations, such as sparse autoencoders, which introduce regularization constraints to force the network to learn more efficient representations, variational autoencoders, which incorporate a probabilistic approach to model the data distribution, and convolutional autoencoders, which are tailored for processing image data by capturing spatial features. Despite their simplicity, autoencoders and their variations effectively identify patterns in unlabeled data and are widely used in various machine learning applications. Recently, another line of research on SHM is exploring ML-based strategies such as neural networks [25], Fuzzy Logic [26], Support Vector Machines [27], and others. Within ML strategies, those based on Deep Learning seem promising, such as autoencoders, for example. Autoencoders are a type of deep neural network algorithm of an unsupervised nature, which automatically extract features from data and incorporate probabilistic elements into their architecture [28]. Autoencoders are recognized not only for their ability to handle large volumes of data, but also for providing robust solutions, especially in nonlinear problems, such as structural anomaly detection.

As evidenced by the increasing number of works dedicated to autoencoders [29–33], it is certain that the study and application of these techniques in applications related to structural damage detection are widely open to numerous advancements. It is important to emphasize that the solution for SHM is not universal, as different techniques may be effective for some structures and not for others [4]. Furthermore, it is noteworthy that most analyses with autoencoders are directed towards machine failures, with relatively limited application in structures [34,35]. Therefore, there is ample opportunity for studies aiming at the development and dissemination of such techniques applied to SHM systems for civil engineering constructions.

In the field of structural health monitoring (SHM), the integration of autoencoders with time–frequency techniques represents a promising approach for enhancing damage assessment methodologies. Autoencoders can effectively encode the intricate temporal and spectral characteristics extracted from signals using methods like the Short-Time Fourier Transform (STFT), S-Transform, or Wavelet Transform [24,36–38]. This approach leverages the comprehensive insights provided by time–frequency analysis to capture nuanced changes in structural behavior, thereby enabling early detection and precise localization of damage. Recent studies by Dang et al. [38] and Alves et al. [24] have demonstrated the efficacy of such integrated approaches in enhancing the sensitivity and reliability of SHM systems. However, our study focuses on the frequency domain to specifically target and quantify structural alterations. This approach allows for a detailed assessment of the alterations across different structures, aligning with the primary objective of our research, which investigates the effectiveness of four autoencoder-based methodologies, combined with Hotelling's T^2 statistical tool, to detect and quantify structural changes.

After an exhaustive review of Google Scholar indexing database using the terms 'autoencoder, damage, structural health monitoring', the authors identified a total of 2070 relevant papers published over the past five years (2019–2024). Among these, 346 papers pertain to the use of sparse autoencoders (SAEs), 410 to variational autoencoders (VAEs), and 465 to convolutional autoencoders (CAEs). Google Scholar indexes a comprehensive range of academic and scientific documents, including journal articles, conference papers, theses, dissertations, technical reports, patents, and gray literature. However, none of these cataloged studies have addressed a comparative analysis among different autoencoder models for SHM applications. Thus, to fill this gap and given that SAEs, VAEs, and CAEs are among the most prevalent types of autoencoder models in the literature, the authors have decided to use these models in this paper. Furthermore, conventional autoencoders (AEs) were included to assess whether their basic architectures are sufficient or if more complex and specialized models are necessary for accurately quantifying structural changes.

Moreover, to the best of the authors' knowledge, there have not been any comparative analyses in the literature concerning the use of these models in civil engineering structures, nor have there been any studies regarding the computational processing costs required for each of these models and their effectiveness. In this context, this paper comparatively evaluates the performance of four different autoencoders (conventional autoencoder, sparse autoencoder, variational autoencoder, and convolutional autoencoder) for identifying structural alterations by considering their effectiveness and processing times. To assess the methodology, three different benchmarkable structures are considered: (i) a 2D frame tested at the Image and Signals Laboratory of the Federal University of Juiz de Fora (UFJF), subjected to five different damage scenarios [33]; (ii) a four-story steel frame located at the Earthquake Engineering Research Laboratory at the University of British Columbia (UBC) that underwent two types of alteration—removal of bracings and loosening of bolts [39–41]; (iii) the known Z24 Bridge, which connected the towns of Koppigen and Utzenstorf in Switzerland; that bridge underwent various progressive damage tests simulating real degradation situations under different temperature conditions for scientific purposes [42,43].

For all applications, a damage index was built over the Shewhart T Control Chart (T^2 statistic) [44], which was calculated using the latent layer data of the autoencoders, allowing the identification of damage in all analyzed structures.

2. Theoretical Background

The strategy used to identify structural novelties is based on a two-step methodology:

1. The autoencoder models. In this paper, these models are used to extract features capable of characterizing the monitored dynamic signals, working as a parameter reducer.
2. The Shewhart Control Chart (T^2). This graphic viewer tool of the statistical metric T^2, calculated from the reduced parameters obtained by the SAE, is used in this paper to objectively indicate the occurrence of structural changes.

A brief discussion of these two steps and the autoencoder methods are presented in the following sections. For more details, the following reference is advised [45].

2.1. Conventional Autoencoder (AE)

The conventional autoencoder (AE) is a basic model for unsupervised learning aimed at reconstructing input data and reducing dimensionality. It operates over two main aspects: compression and reconstruction. During the compression phase, information is reduced and transformed into a compact representation, called the latent layer. Then, in the reconstruction phase, the AE attempts to reconstruct the original data from these compact representations. Although less sophisticated than its variations, the AE is still widely used in various applications due to its simplicity and effectiveness in identifying patterns in unlabeled data.

An AE is a basic neural network architecture consisting of two main parts: an encoding function $\mathbf{h} = f(\mathbf{x})$ and a decoder that produces a reconstruction $\mathbf{r} = g(\mathbf{h})$. During training, the input is passed through the encoder, which maps the data to a lower dimensional latent representation, capturing the most important features of the input data. Then, the latent representation is passed through the decoder, which attempts to reconstruct the original input from this reduced representation. The AE is trained to minimize the difference between the original input and the reconstructed output, using a loss function such as the mean squared error, for instance [45]. The learning process involves minimizing a loss function, as described by Equation (1),

$$L(\mathbf{x}, g(f(\mathbf{x}))), \tag{1}$$

where L is a loss function that penalizes $g(f(\mathbf{x}))$ for being different from $\mathbf{x}$ [45].

The main objective of an autoencoder is to learn how to reconstruct its original input as accurately as possible to its output, using the most important features of the input data (latent representation), which can be utilized in various machine learning applications.

The autoencoder training process utilized Mean-Squared Error (MSE) as the loss function. MSE is preferred for its ability to assess reconstruction accuracy by averaging the squared differences between predicted values y_i generated by the autoencoder and actual input values x_i, as defined in Equation (2). This metric is chosen for its sensitivity to larger errors, which are magnified due to squaring, thus imposing a penalty proportional to the error magnitude. Minimizing the MSE during training enhances the model's capability to effectively reduce the discrepancy between the input and reconstructed output, thereby improving its ability to capture essential and meaningful data features.

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i)^2 \tag{2}$$

2.2. Sparse Autoencoder (SAE)

The SAE is a variation of the conventional autoencoder, which stands out for its ability to learn sparse representations of data, highlighting distinctive features. During training, the SAE imposes a sparsity penalty, encouraging the model to efficiently encode the most relevant features of the data. By promoting the sparsity of representations, the SAE helps to highlight fundamental aspects of the data, improving the model's ability to discriminate between significant patterns and noise.

During training, the SAE imposes a sparsity penalty $\Omega(\mathbf{h})$ on the code layer $\mathbf{h}$ in addition to the reconstruction error, as described in Equation (3):

$$L(\mathbf{x}, g(f(\mathbf{x}))) + \Omega(\mathbf{h}), \tag{3}$$

where $g(\mathbf{h})$ is the output of the decoder, $\mathbf{h} = f(\mathbf{x})$ is the output of the encoder, and $\Omega(\mathbf{h})$ is the sparsity penalty applied to the code layer. This sparsity penalty encourages the model to learn sparse representations of the data, focusing only on the most relevant features during the encoding process [45].

This is useful for highlighting important aspects of the data, allowing for a more efficient and discriminative representation. By promoting the sparsity of representations, the SAE simplifies the analyzed signals, making them more suitable for structural change detection. Its ability to identify distinctive features within high-dimensional data makes it a valuable tool in various applications, including pattern recognition and anomaly detection.

The SAE also utilizes the Mean-Squared Error (MSE) as the loss function during training. The MSE is chosen for its effective ability to minimize the squared differences between predicted and actual values, thereby enhancing reconstruction accuracy and facilitating the extraction of meaningful data features.

2.3. Variational Autoencoder (VAE)

The VAE is an advanced technique that stands out for its ability to extract meaningful features from analyzed signals, acting as a parameter reducer. By probabilistically modeling the distribution of input data, the VAE offers a robust representation of features, enabling more effective encoding. These models have been widely adopted in various applications, ranging from image generation to structural damage detection, through the learning of latent representations. Its ability to learn complex latent representations makes it a valuable choice in various applications, including structural damage detection. The VAE uses the normal distribution to model the distribution of latent data, introducing stochastic variability in data encoding. This feature is essential for dealing with the inherent uncertainty in data and improving the model's ability to capture the complexity of input data. During training, the VAE is optimized to minimize the difference, or divergence, between the original input data $(\mathbf{x})$ and the reconstructed data $(\mathbf{x'})$, while also keeping the distribution of latent data close to a normal distribution. To generate a sample from the input data, the VAE first extracts a sample $\mathbf{z}$ from the data distribution $p(\mathbf{z})$. Then, the sample passes through a differentiable generator network $g(\mathbf{z})$. Finally, $\mathbf{x}$ is drawn from a distribution $p(\mathbf{x}; g(\mathbf{z})) = p(\mathbf{x}|\mathbf{z})$. During training, the encoder network $q(\mathbf{x}|\mathbf{z})$ is used to obtain $\mathbf{z}$, and $p(\mathbf{z})$ is then a decoding network [45].

Similar to the AE and SAE, the VAE also uses the Mean-Squared Error (MSE) as its loss function. The MSE plays a critical role in minimizing squared differences between predicted and actual values, thereby improving reconstruction accuracy and facilitating the extraction of significant data features.

2.4. Convolutional Autoencoder (CAE)

The convolutional autoencoder (CAE) is an effective approach in the field of machine learning, especially for tasks related to image processing. They are capable of learning efficient representations of images through convolutional layers, capturing local and hierarchical patterns present in the input data. These models have been widely used in a variety of applications, including object recognition, structural change detection, and image segmentation and compression, providing an effective way to encode and reconstruct images with high fidelity. They can also handle one-dimensional sequential data, such as time series, using convolutional layers, with high feature extraction capability from these convolutional layers. During encoding, convolutions are employed to capture local patterns, while in decoding, transposed convolutions are used to reconstruct the original sequence. This architecture is effective for extracting and representing relevant sequential features, being useful in tasks such as time series prediction, allowing possible detection of structural changes.

A convolutional autoencoder (CAE) consists of an encoder that compresses the original data into low-dimensional features and a decoder that reconstructs the compressed features into a data similar to the original. The encoder has an input layer and four consecutive convolutional 1D and MaxPooling layers (Conv + Max), which are connected to flattening and dense layers (flat + dense). This process filters and compresses the information that the CAE needs to recognize in the image. In the decoder, symmetric to the encoder, the flattening and dense layers are connected to consecutive upsampling and convolution layers (Up + Conv) and an output layer with the same dimension as the input layer.

In a simplified manner, the 1D convolutional layer applies 1D convolution windows separately to each channel and mixes the channels through pointwise multiplication, while the MaxPooling layer is used to subsample the feature map, retaining only the most relevant information extracted by the convolutional layer. This is necessary because convolutional layers significantly increase the number of parameters in the output tensors compared to the input ones, exponentially increasing the magnitude of the tensors [46].

The convolutional autoencoder training process is executed using the Mean Absolute Error (MAE) as the loss function. The MAE is chosen due to its robustness as a statistical measure, quantifying the average magnitude of absolute discrepancies between the predicted values y_i generated by the autoencoder and the actual input values x_i, as delineated in Equation (4).

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - x_i| \tag{4}$$

This choice was motivated by the MAE's capability to directly assess reconstruction accuracy by averaging absolute differences across all N samples in the dataset [47]. In the context of structural damage detection, the MAE proves effective within the CAE for capturing local and hierarchical patterns in the data, essential for identifying significant structural changes.

Table 1 summarizes some of the key differences among these four autoencoder models.

Table 1. Comparison of features for the four different autoencoder models evaluated.

Feature	AE	SAE	VAE	CAE
Latent Distribution	Fixed	Sparse	Variable	Pooling
Regularization	None	Sparsity penalty	KL regularization	Dropout
Flexibility	Low	Moderate	High	Moderate
Advantages	Simple and fast convergence	Highlights distinctive data features	Probability distribution	Captures local patterns
Disadvantages	Limited in capturing complex data patterns	Requires careful tuning of hyperparameters	May require high computational cost	Difficult interpretability

Even though the use of autoencoder models for SHM applications has flourished over the last years, it is important to remark about some of their limitations. In fact, as with any other data-driven-based damage-detection method, these models are highly dependent on the quality of the input data. If the input data quality is poor, the models will not yield good results. In addition to this, these techniques are often referred to as 'black-box models' [48], meaning that little knowledge can be extracted from them and most of the physical meaning of the problem is lost in the way. Hence, this type of technique should not be blindly used by those who do not fully understand the structure's complexity or its underlying dynamic behavior.

Thus, according to Table 1, AEs exhibit simplicity and rapid convergence, making them a practical choice for initial experiments and quick tests. In practice, AEs are easy to implement and train, allowing for a swift assessment of model viability. However, their simplicity also limits their ability to capture complex and subtle patterns in more intricate datasets, resulting in less precise reconstructions for such data.

On the other hand, SAEs effectively highlight the most important data features, which is beneficial for tasks like anomaly detection and feature selection. Practically, their capability to focus on distinctive features is highly valuable. Nonetheless, this effectiveness necessitates careful tuning of the sparsity hyperparameters, which can be a time-consuming process requiring expertise to avoid underfitting or overfitting.

Additionally, VAEs provide a probabilistic framework that captures the data distribution, facilitating generative tasks such as data synthesis and anomaly detection. They are frequently employed when a robust and probabilistic approach to data modeling is required. However, this added complexity can lead to higher computational costs, longer training times, and increased resource consumption, posing challenges in resource-constrained environments.

Finally, CAEs are predominantly used for image data, effectively capturing spatial hierarchies and local features, which is advantageous for handling visual data. Nevertheless, the interpretability of these networks can be challenging. The convolutional layers and their filters are intricate, making it difficult to comprehend how specific features are captured and represented, thereby complicating model analysis and fine-tuning.

2.5. Shewhart T^2 Control Charts

Shewhart T^2 Control Charts are statistical graphical tools used to assess the influence of various parameters of a problem over time. They display multiple data points, composed of a specific statistical characteristic, with horizontal lines separating different classes of structural scenarios. Points outside the expected intervals indicate changes, suggesting an out-of-control situation [44]. In addition to its early-problem-detection capability, T^2 offers the advantage of being a well-established statistical tool accepted by the scientific and industrial community. Its interpretation is straightforward and intuitive, facilitating result interpretation and decision-making by users. Therefore, given its relevance and applicability in various contexts, T^2 represents a powerful tool for identifying and quantifying structural changes in analyzed signals, contributing to the improvement of the quality, efficiency, and reliability of SHM [44].

In this study, T^2 was calculated based on the parameters extracted by different autoencoder models, and its application allowed the identification and quantification of changes or damage in the analyzed signals by comparing the observed values with statistically established limits. The T^2 statistic represents the distance between a new data observation and the sample mean vector corresponding to it: the higher the value of T^2, the greater the distance of the new data from the mean is. This metric is based on the relationship between variables and the dispersion of the data (covariance matrix).

In addition to T^2, other metrics such as the Mean-Squared Error (MSE) and the Original-to-Reconstructed-Signal Ratio (ORSR) were also evaluated, but they did not present satisfactory results in classifications and were, therefore, not included in this paper.

Given a matrix $\mathbf{H}_{N\times M}$ representing a dataset over a certain time (which in this paper, are the features extracted from the latent layers of the autoencoders), the T^2 statistic can be calculated according to Equation (5):

$$T^2 = R(\overline{\mathbf{h}} - \overline{\overline{\mathbf{h}}})^T \mathbf{S}^{-1} (\overline{\mathbf{h}} - \overline{\overline{\mathbf{h}}}) \tag{5}$$

where $\overline{\mathbf{h}}$ is the mean vector of the sample of M available characteristics, obtained from a submatrix of $\mathbf{H}$ with R observations ($\mathbf{H}_{R\times M}$, $R < N$); $\overline{\overline{\mathbf{h}}}$ and $\mathbf{S}$ are the reference mean vector and the mean of the reference covariance matrices, respectively. In all conducted studies, the Upper Control Limit (UCL) was defined as immediately above 95% of the T^2 values from the training data (values greater than the UCL should be observed only 5% of the time for the structure in its original state). The Lower Control Limit (LCL) was not used and, thus, defined as 0. Hotelling's T^2 values greater than the UCL indicate changes in the data, which may suggest the presence of structural damage or other anomalies. By this definition, 5% of the T^2 values from the training data will be above the UCL, which is not a concern if they are close to it.

3. Methodology

The proposed methodology aims to evaluate the ability of the autoencoders to separate, in an unsupervised manner, vibration signals belonging to different structural dynamic behaviors. This is achieved by training the model with only a portion of the intact structural data and subsequently testing it with the remaining data. During this process, the model compares each signal with the reference class (i.e., damage state), deriving a specific difference, or reconstruction error, that gradually increases between classes, but remains relatively constant within each of them.

Thus, all data are divided into three sets: training, validation, and monitoring, as follows:

- Training phase: In this stage, a dataset extracted from the structure's undamaged state is used to train the autoencoder model. This dataset is referred to as the training set.
- Validation phase: During this phase, another dataset (validation data), also extracted from the same structural state as the training phase, is applied to the model trained in the previous stage. The goal is to verify the model's ability to classify new data. It is expected that the model will lead to statistically similar T^2 values for training and validation datasets since they belong to the same structural state.
- Monitoring phase: In this phase, datasets (monitoring data) extracted from other structural cases—different from those used in the training and validation phases—are presented to the model. It is expected that the model will yield T^2 values higher than those obtained in the previous phases because they belong to a different structural state from the one used for training.

All simulations in this study were conducted using Google Colab, a cloud-based platform that provides access to powerful computational resources. The choice of Google Colab was driven by its accessibility, ease of use, and the ability to leverage high-performance hardware without the need for local infrastructure. The machine specifications were as follows: the CPU is an Intel(R) Xeon(R) CPU @ 2.30 GHz, manufactured by Intel Corporation, located in Santa Clara, CA, USA, equipped with 13,290,464 kB (approximately 13 GB) of RAM. Python version 3.10.12 was utilized. The GPU used was a Tesla T4, manufactured by NVIDIA Corporation, located in Santa Clara, CA, USA, with driver version 535.104.05 and CUDA version 12.2, offering 15,360 MiB of memory. The available disk space on the root partition was 2.0 GB. Utilizing Google Colab facilitated efficient computation and ensured the reproducibility of the results, making it an optimal choice for the computational requirements of this study.

For all structures analyzed, the Fourier Transform (FFT) of the structural dynamic signals was used as the inputs to the autoencoder models. This choice was motivated by the fact that the compressions and reconstructions performed by the autoencoders were more accurate in this domain compared to the time domain. One advantage of processing data in the frequency domain is that the transformation using the FFT algorithm can already be understood as a parameter extractor, which facilitates the encoding step by autoencoders, as seen in studies by Resende et al. [49].

For reproducibility, a 10-fold cross-validation scheme was used for each structure to avoid variations and obtain more homogeneous and precise evaluations. The order of the data within the classes was also randomly modified at the beginning of each repetition, thereby changing the portion and sequence of intact structure data used for training and that used for validation.

After performing the FFT over the acceleration measurements, all data were standardized using **z**-score standardization, according to Equation (6):

$$\mathbf{z} = \frac{\mathbf{x} - \mu}{\sigma}, \tag{6}$$

where **z** is the standardized signal vector, **x** is the original signal vector, and μ and σ are the mean value and the standard deviation of the signal, respectively.

Table 2 summarizes some of the key differences among these four autoencoder models.

Table 2. Signals per class in each analysis of each structure.

	Analysis 1 **2D Laboratory Frame**	**Analysis 2** **3D Yellow Frame** **(Removal of Braces)**	**Analysis 3** **3D Yellow Frame** **(Loosening of Bolts)**	**Analysis 4** **Z24 Bridge**
Class 1—training	200	24	60	350
Class 1—validation	100	6	30	235
Class 2	300	30	90	585
Class 3	300	30	90	585
Class 4	300	30	-	585
Class 5	300	30	-	-
Class 6	-	22	-	-

All autoencoder models depend on the following parameters and hyperparameters: learning rate, number of epochs, batch size, original dimension, latent dimension, and the optimizer algorithm. However, for SAE, the sparsity parameter 'lambda' must also be defined; for the VAE, the intermediate dimension is used; for the CAE, since the latent layer is not one-dimensional, the number of filters and their sizes for that layer must also be defined.

The hyperparameters of the autoencoders were selected using the Optuna parameter optimizer. This is an automated and efficient hyperparameter optimization tool that employs intelligent search techniques to find the best combinations of hyperparameters for a given machine learning model through a search and selection approach based on efficient sampling [50].

In Optuna, the search space for the hyperparameters to be optimized needs to be initially defined. This includes setting the possible value ranges for each hyperparameter, such as the batch size, learning rate, number of epochs, latent dimension, intermediate dimension, and lambda sparse. These ranges were chosen based on established practices in the literature and preliminary experiments that indicated the bounds within which optimal performance is likely to be found [51,52].

Hence, a sensitive study was conducted, consisting of multiple trials to find the best combination of hyperparameters. To this end, Optuna uses sampling algorithms, such as the Tree-structured Parzen Estimator (TPE) to select sets of hyperparameters to be evaluated [50,53]. The TPE algorithm was chosen because it has been shown to perform well in high-dimensional optimization problems and to handle complex search spaces efficiently [54]. A total of 100 trials was defined for each analysis performed. In each trial, the selected set of hyperparameters was used to train and evaluate the model.

An objective function was then defined to assess the model's performance with the given set of hyperparameters. In the analyses conducted, this was performed by minimizing the difference in the T^2 values between the training Class 1 and the validation Class 1. Based on the results of previous trials, Optuna adjusts its sampling to focus on areas of the search space that are more likely to contain the best hyperparameters.

After conducting 100 trials for each analysis, Optuna identifies the top 5 hyperparameter combinations along with their respective T^2 plots that achieved the best results. From these combinations, the set of hyperparameters that resulted in the largest difference in the T^2 values between signals of different classes for each analysis, while maintaining closer values for signals of the same class, was selected. Optuna optimized parameters such as the learning rate, number of epochs, batch size, intermediate dimension, latent dimension, optimizer, and lambda sparsity according to the specific autoencoder used. These optimized hyperparameters were subsequently used for training the autoencoders.

Figure 1 provides a schematic overview of the proposed methodology. This diagram illustrates the sequential steps involved, from the initial processing of vibration signals to the final quantification of structural changes.

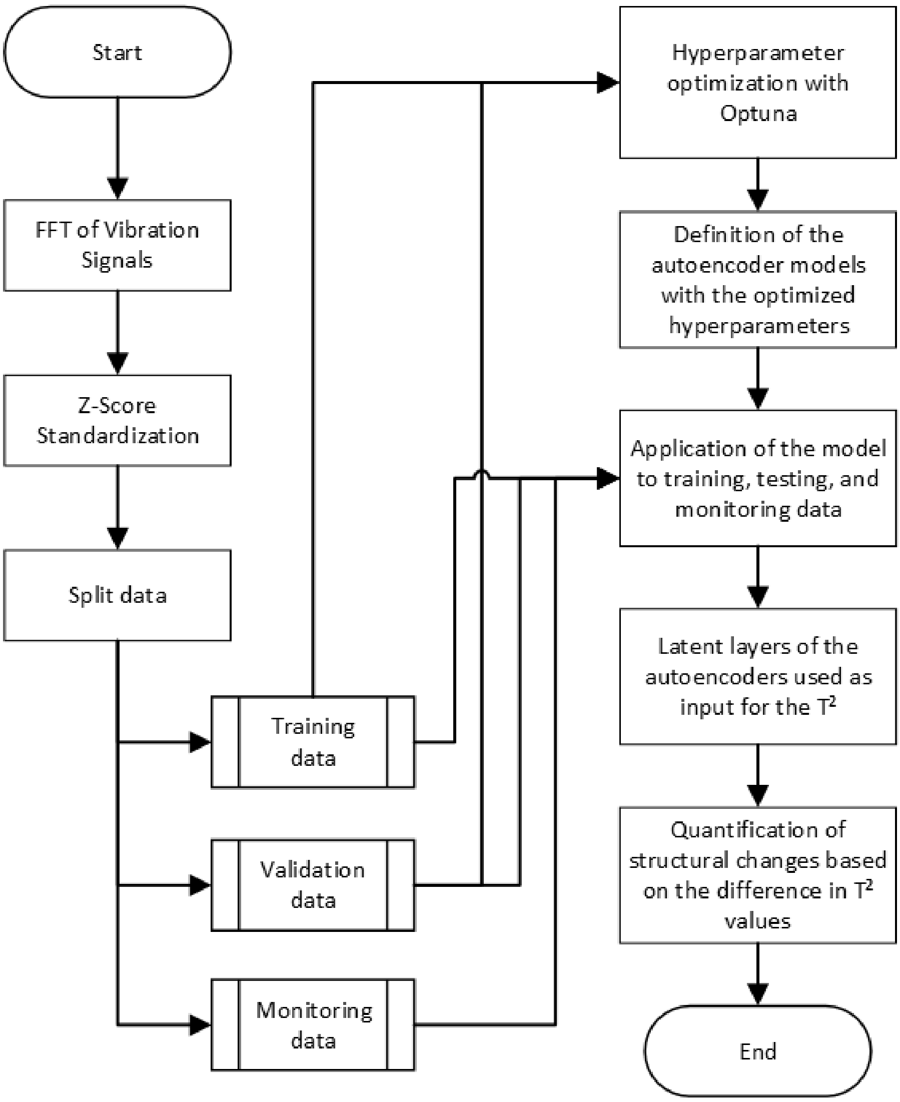

Start
FFT of Vibration Signals
Z-Score Standardization
Split data
Training data
Validation data
Monitoring data
Hyperparameter optimization with Optuna
Definition of the autoencoder models with the optimized hyperparameters
Application of the model to training, testing, and monitoring data
Latent layers of the autoencoders used as input for the T²
Quantification of structural changes based on the difference in T² values
End

Figure 1. Flowchart of the proposed method.

4. Applications

4.1. Two-Dimensional Laboratory Frame

The first structure used for structural change detection (Analysis 1) is a 2D frame shown in Figure 2, tested at the Image and Signals Laboratory of UFJF [33]. It consists of six aluminum bars with an elastic modulus E = 70 GPa, each measuring 300 mm in length, 15.875 mm in width, and 1.587 mm in height.

In Figure 3, an illustration of the experimental test is shown. Four unidirectional piezoelectric accelerometers (100 mV/g) were placed on this structure at the marked positions, measuring horizontal accelerations. An impact load was applied using a pendulum with a mass of 14 g, as shown in Figure 3. To obtain the data, the pendulum was released from rest from the position also indicated in Figure 3, being subjected to the action of gravity until its collision with the structure at the indicated point.

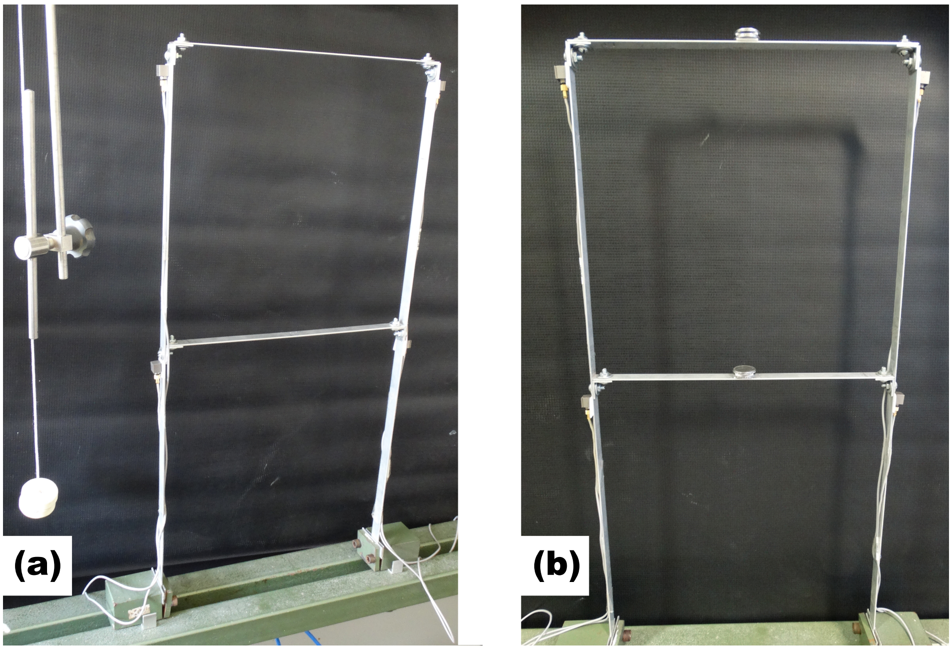


Figure 2. The 2D frame tested at the Image and Signals Laboratory of UFJF. (**a**) Front view of the structure in scenario 1; (**b**) Frontal view of the structure in scenario 4. Adapted from [33].

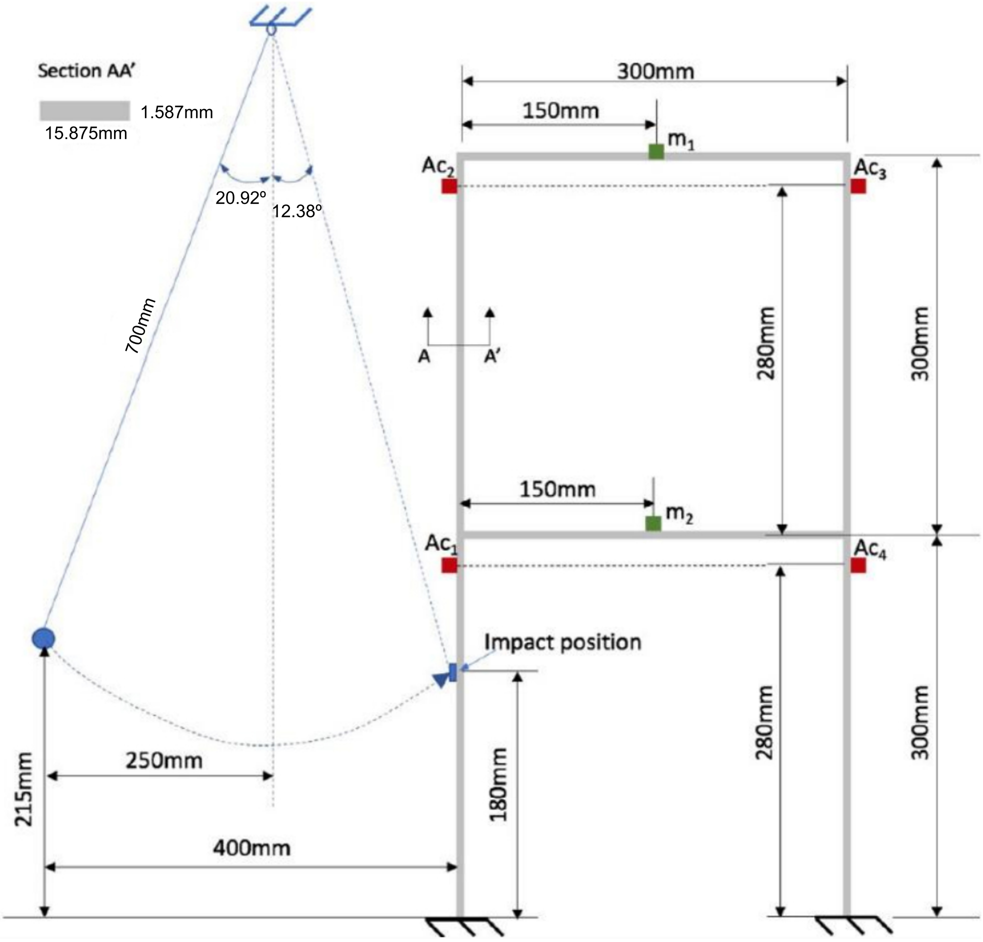


Figure 3. Illustration of the experiment conducted on the portal. Adapted from [33].

Five structural scenarios were tested, as indicated in Table 3.

Table 3. Structural scenarios evaluated in Analysis 1. Adapted from [33].

Scenario	Configuration
1	No additional mass added to the structure. m1 = m2 = 0.
2	An additional mass of 7.81 g added to the structure. m1 = 7.81 g and m2 = 0.
3	Two additional masses of 7.81 g added to the structure. m1 = 15.62 g and m2 = 0.
4	Three additional masses of 7.81 g added to the structure. m1 = 15.62 g and m2 = 7.81 g.
5	Four additional masses of 7.81 g added to the structure. m1 = 15.62 g and m2 = 15.62 g.

Each test had a duration of 8.192 s with a sampling frequency of 500 Hz, resulting in 4096 sampled points per accelerometer. Frequencies up to 40 Hz were analyzed as they are the most relevant for this structure.

The hyperparameter set for the different autoencoders for this analysis is shown in Table 4:

Table 4. Hyperparameters used in the autoencoder models for Analysis 1.

	AE	SAE	VAE	CAE
learning_rate	0.019	0.018	0.0195	0.0175
epochs	5	5	5	5
batch_size	3	3	3	3
original_dim	160	160	160	160
intermediate_dim	-	-	60	-
latent_dim	40	45	40	40 × 32
optimizer	Adam	Adam	Adam	Adam
lambda_sparse	-	0.001	-	-

The original and reconstructed FFTs by the AE, SAE, VAE, and CAE for each class are presented in Figures 4, 5, 6 and 7, respectively.

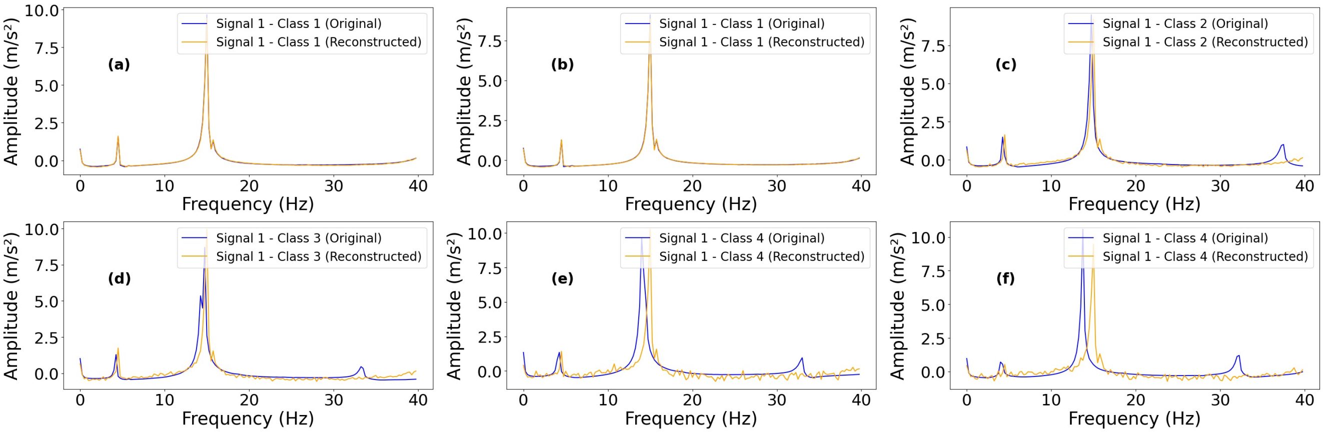


Figure 4. Original and reconstructed signals by AE from Analysis 1 (accelerometer 1). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5.

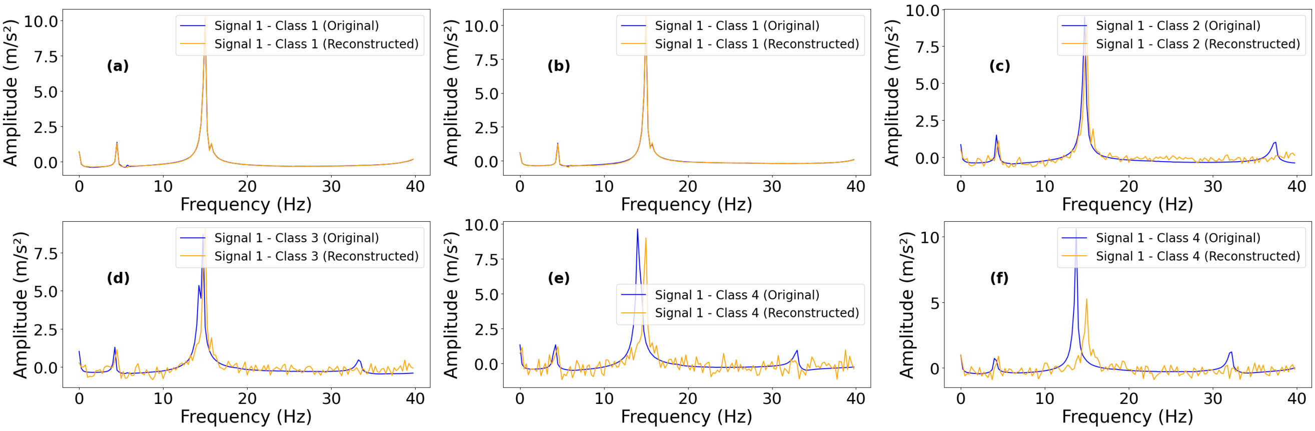


Figure 5. Original and reconstructed signals by SAE from Analysis 1 (accelerometer 1). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5.

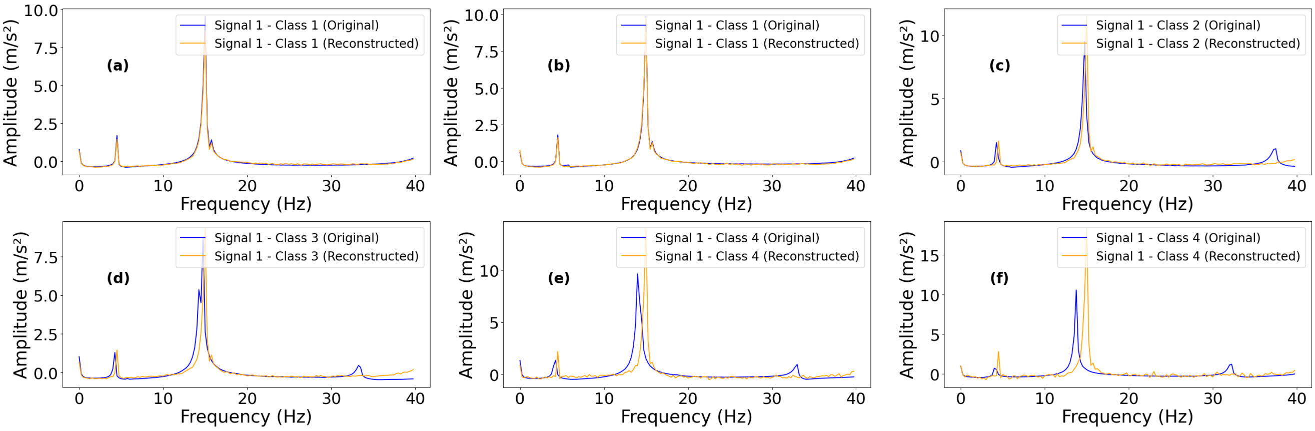


Figure 6. Original and reconstructed signals by VAE from Analysis 1 (accelerometer 1). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5.

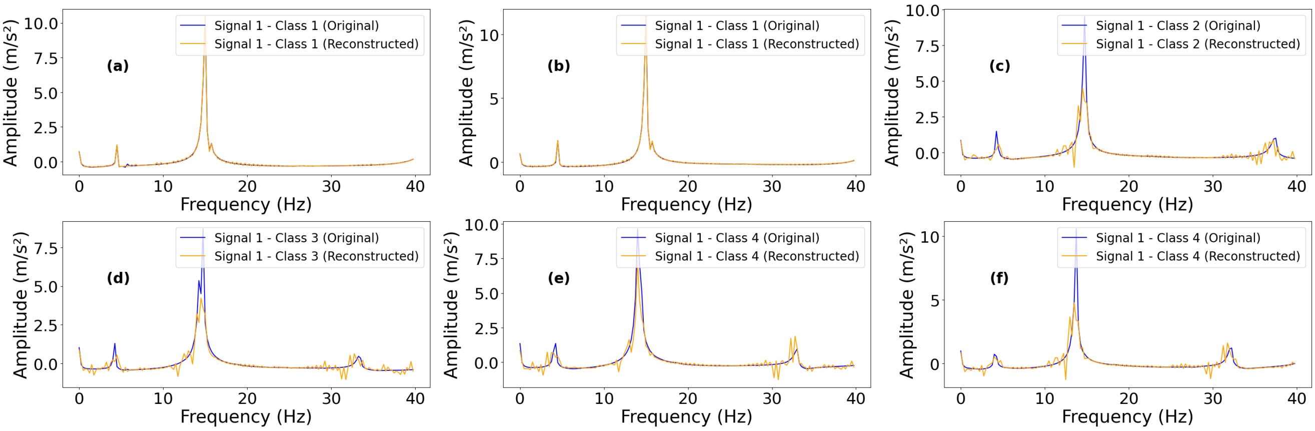


Figure 7. Original and reconstructed signals by CAE from Analysis 1 (accelerometer 1). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5.

The analysis of the signals visually reveals that the reconstructions were more similar to the original signals in Class 1 compared to the others. This is because the autoencoders were trained exclusively with the data from this class. By applying the compression and reconstruction process to the other classes using the training from Class 1, the differences between the reconstructions and the original signals increased as they diverged from the patterns of Class 1, which was expected.

The results for Analysis 1 are depicted in Figure 8, which combines the four plots generated by the Hotelling T^2 from the latent layers of the autoencoders. Each color indicates the results for each scenario, as outlined in Table 3.

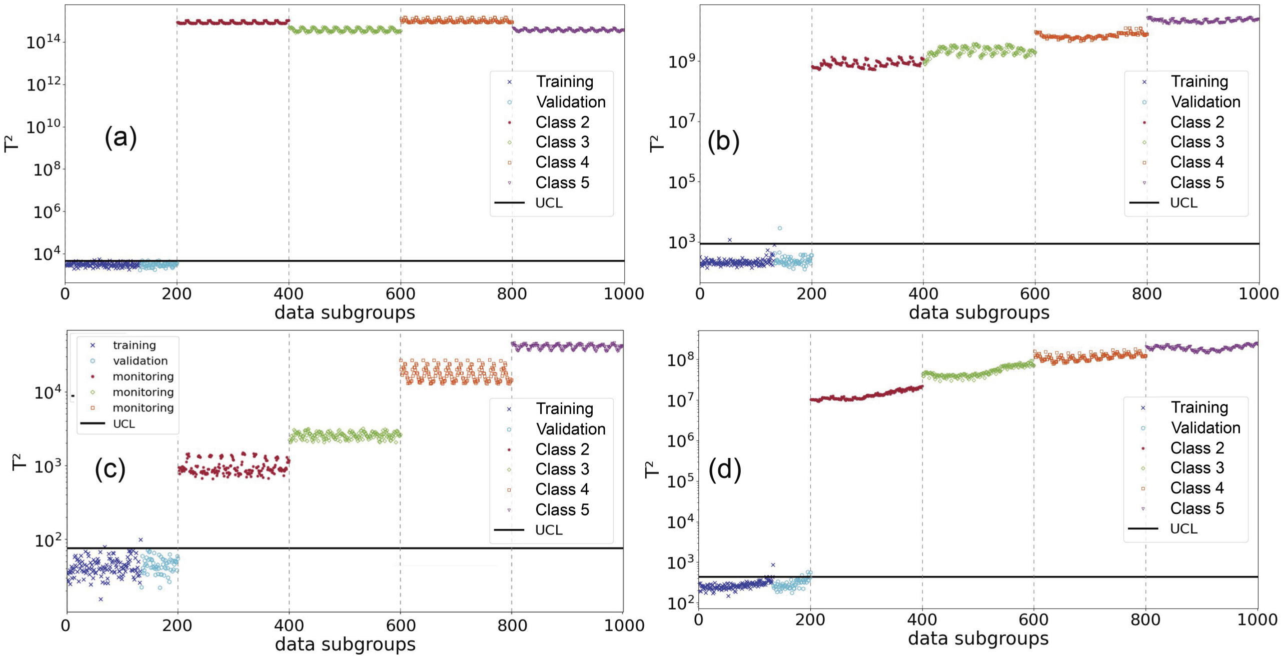


Figure 8. Shewhart T^2 Control Charts for the 2D laboratory frame (accelerometer 1). (**a**) AE; (**b**) SAE; (**c**) VAE; (**d**) CAE.

All analyzed methodologies produced similar T^2 values between the validation and training classes, demonstrating the models' ability to classify new data from this structure, as validation and training data belong to the same structural state and should, therefore, have similar T^2 values. Additionally, all methodologies were able to correctly identify the presence of structural changes, as all monitoring classes exceeded the UCL.

However, only the VAE was able to accurately quantify the structural changes, with T^2 values increasing as additional mass was added to the frame, without, in any case, points from different scenarios having values within the same range. Although the CAE results (Figure 8d) also showed an increase in the T^2 values with the addition of mass, there are still cases of different scenarios with T^2 values within the same range. Thus, the AE, SAE, and CAE methodologies (Figure 8a,b,d) presented similar T^2 values between signals from different classes, being unable to accurately quantify the levels of structural changes.

Finotti et al. [33] conducted a similar study using the same structure, employing the SAE and Principal Component Analysis (PCA) for feature extraction from structural signals in the time domain rather than the frequency domain. While their SAE approach successfully detected structural alterations, unlike our study, which focused on frequency domain analysis, they did not provide a quantitative assessment of these alterations. This underscores the advantage of using VAE models in conjunction with frequency domain data for both detection and quantification of structural changes.

4.2. Three-Dimensional Yellow Frame

Following the tests conducted in Analysis 1, similar evaluations were carried out on the well-known 'Yellow Frame' tested at the Earthquake Engineering Research Laboratory at UBC [39,40]. As depicted in Figure 9, this frame consists of four stories and was installed on a concrete slab outside the laboratory to simulate environmental conditions.

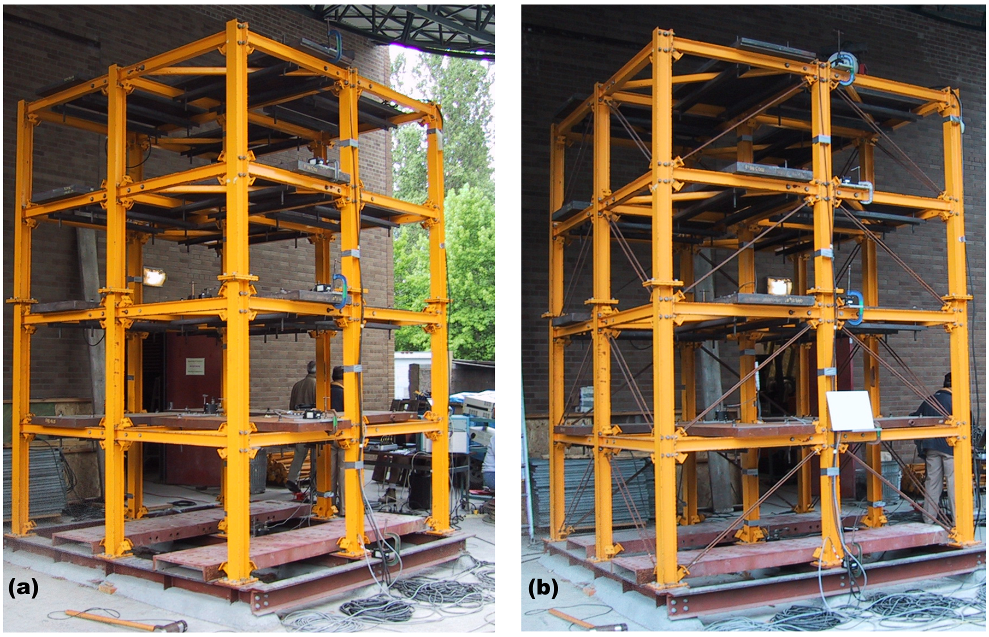


Figure 9. Yellow Frame: (**a**) with bracings and (**b**) without bracings. Source: [40].

The structure has a height of 3.6 m and a plan of 2.5 m × 2.5 m, and its members are made of hot-rolled steel, with a nominal yield stress of 300 MPa. The columns are B100 × 9 sections, and the floor beams are S75 × 11 sections. Figure 10 shows the beam–column connection of the portal components, as well as the bracing connections and an additional floor slab. Figure 11 presents the plan view and the east view of the structure.

In each span, the bracing system consists of two ½-inch-diameter steel rods, placed in parallel along the diagonal. To make the mass distribution more realistic, a floor slab was placed in each span per floor: four slabs of 1000 kg on each of the first three floors and four slabs of 750 kg on the fourth floor, as shown in Figure 12. Additionally, a floor grating was present on the second floor to be used as a work platform, adding 35 kg per span to the mass of the second floor.

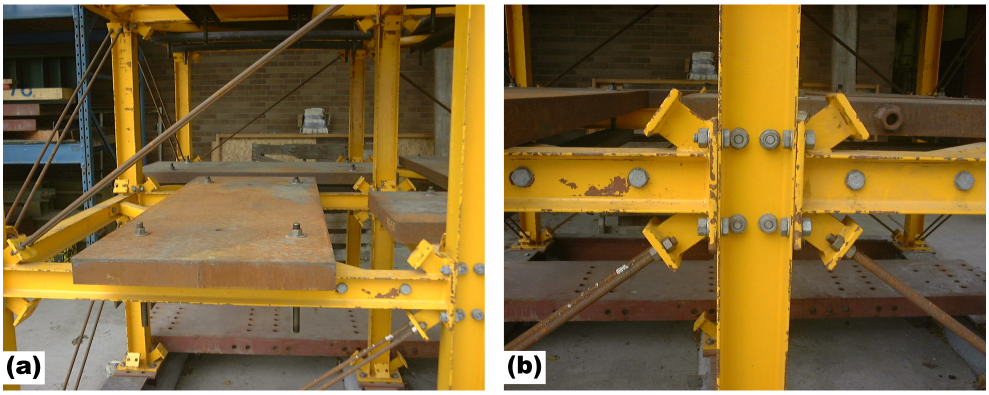


Figure 10. Zoom on (**a**) connections and (**b**) added masses. Source: [40].

Several experiments were conducted on the structure, including ambient vibration tests, impact tests, and shaker tests. To evaluate the proposed approach, only the ambient vibration tests were utilized. Hence, the excitations were due to wind, pedestrians, and nearby traffic around the structure. Further details on the experiment can be found in [39,40].

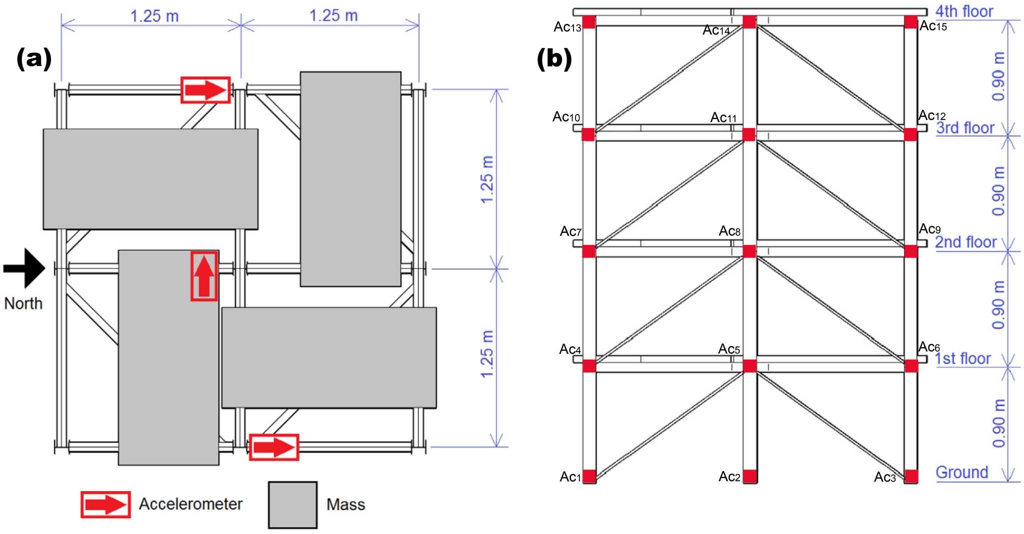


Figure 11. Tested structure: (**a**) plan view and (**b**) east view. Adapted from [40].

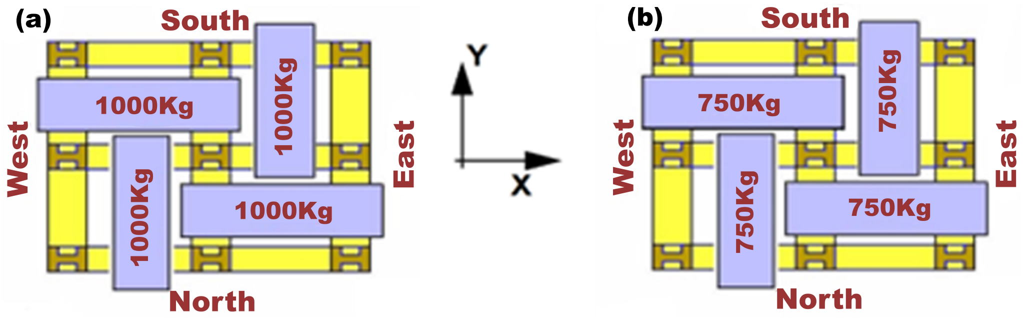


Figure 12. Added slabs in the structure: (**a**) floors 1–3 and (**b**) floor 4. Source: [40].

Fifteen accelerometers (5 V/g range) were placed throughout the structure, three on each floor, including the ground level. These transducers were positioned to measure movements in all directions and torsional modes (Figure 11a). FBA sensors were placed along the east and west frames of the structure to monitor north–south directional movement, aligning with the structure's strong axis. EPI sensors were positioned near the central column, oriented to measure the structure's east–west movement along the weak axis. Additionally, in tests involving loosened portal beams, sensors initially coupled to the loosened beam were relocated to the subsequent portal frame.

The different loading scenarios analyzed, with and without braces (Figure 9), are described in Table 5. Damage scenarios (1–6) were imposed on the braced structure by removing or gradually placing braces, as shown in Figure 13. In the acquisition order, scenarios 1, 5, 4, 3, and 2 simulate gradual damage to the bracing system. Finally, scenario 6 corresponds to the repair of multiple braces and damage to others on another face.

The unbraced structure had gradual damage simulated by loosening the bolts at the beam–column connections, as shown in Figure 14. Cases 7, 9, and 8, in that order, represent increasing damage, i.e., the number of loosened bolts.

Thus, two separate analyses were conducted for this structure: the first aimed to assess the removal of the braces (Analysis 2); the second aimed to evaluate the loosening of the bolts (Analysis 3). For both cases, the testing configurations remained the same, i.e., fifteen accelerometers with a sampling frequency of 200 Hz. Additionally, an anti-aliasing filter with a cutoff set at 50 Hz was employed.

Table 5. Structural scenarios evaluated in Analysis 2 and Analysis 3. Source: [40].

Scenario	Configuration
1	Fully braced structure.
2	Removal of all braces from the east side.
3	Removal of braces on all floors of one bay in the southeast corner.
4	Removal of braces on the 1st and 4th floors of one bay in the southeast corner.
5	Removal of braces on the 1st floor of one bay in the southeast corner.
6	Removal of braces on the 2nd floor on the north face.
7	All braces removed on all faces.
8	Unbraced structure + loosened bolts on all floors at all ends of the beam on the east face, north side.
9	Unbraced structure + loosened bolts on floors 1 and 2 at all ends of the beam on the east face, north side.

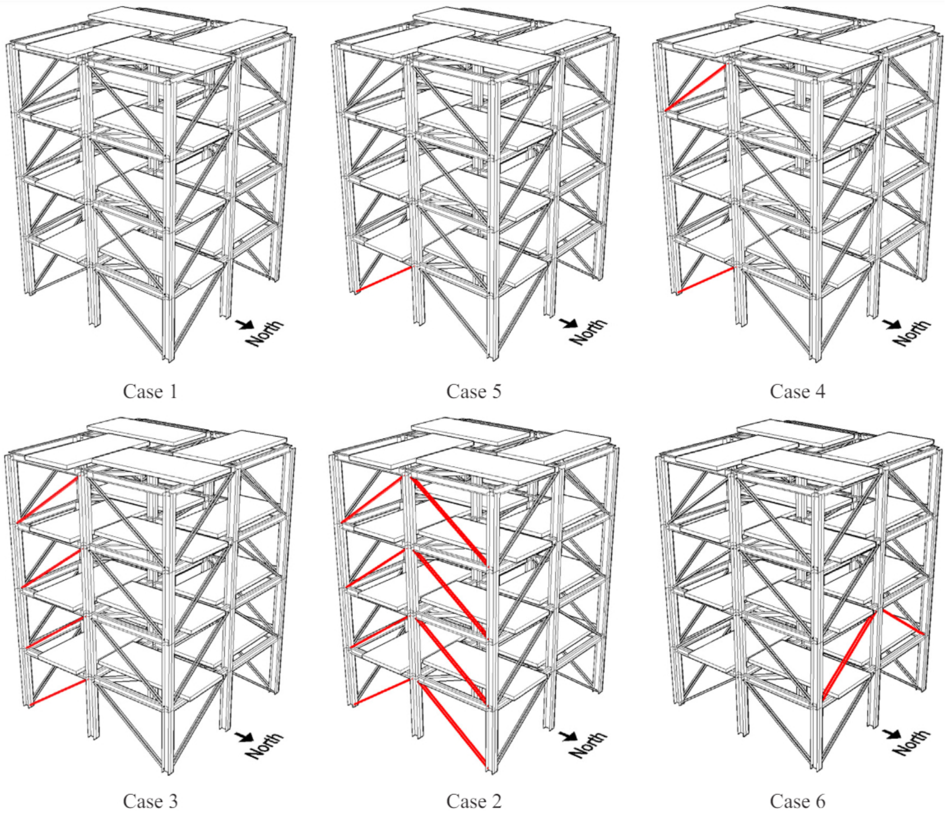
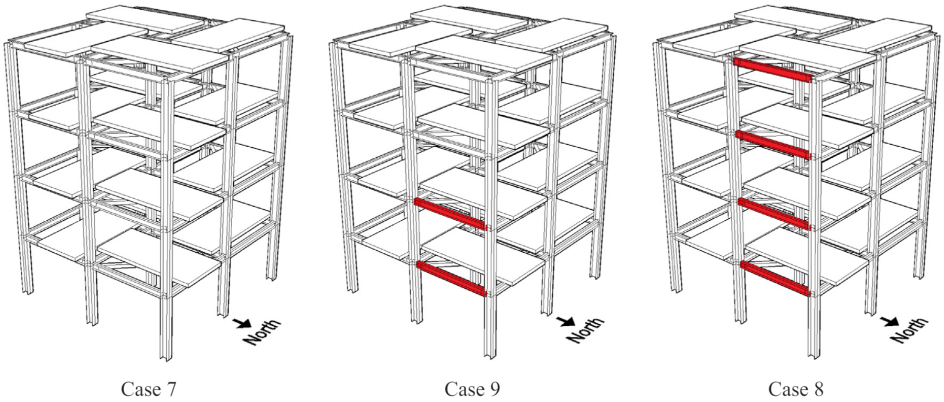


Figure 13. Braced structure: Cases 1 to 6. Red lines represent the bracing removed. Source: [40].

4.2.1. Removal of Braces

Each test lasted 5 min with a sampling frequency of 200 Hz, resulting in a single signal of 60,000 sampled points per accelerometer for each case. The signal, comprising 60,000 points, was segmented into 30 signals of 2000 points each before the analyses began, to provide a larger sample set for the models. The only exception was the signal in Case 6, which was divided into 22 segments with 2000 points because it originally had 45.568 points available.



Case 7 Case 9 Case 8

Figure 14. Unbraced structure: Cases 7 to 9. Red lines mark the floors where the bolts have been loosened. Source: [40].

The segmentation was necessary to make training and testing of the autoencoder models faster, since analyzing the entire signal would be unfeasible. To avoid biasing the results, a stratified k-fold validation was employed. Moreover, the segmentation approach was designed to improve the model's ability to detect subtle changes in the structure's health, which is crucial for accurate monitoring. In this study, a frequency range from 2 to 15 Hz was investigated. The hyperparameters used in the autoencoder models for Analysis 2 are shown in Table 6.

Table 6. Hyperparameters used in the autoencoder models for Analysis 2.

	AE	SAE	VAE	CAE
learning_rate	0.0055	0.0063	0.0071	0.0074
epochs	42	40	44	44
batch_size	16	16	12	24
original_dim	130	130	130	130
intermediate_dim	-	-	51	-
latent_dim	32	34	34	33 × 32
optimizer	Nadam	Nadam	Nadam	Nadam
lambda_sparse	-	0.01	-	-

The original and reconstructed FFTs by the AE, SAE, VAE, and CAE for each class, derived from accelerometer 4, are presented in Figures 15, 16, 17, and 18, respectively.

Just as it occurred in Analysis 1, the reconstructions were more similar for the original signals in Class 1 compared to the others. Again, this is because the autoencoders were trained exclusively with data from this class. By applying the compression and reconstruction process to the other classes using the training from Class 1, the discrepancies between the reconstructions and the original signals increased as they diverged from the patterns of Class 1.

The results obtained by the models in Analysis 2, derived from accelerometer 4, are presented in Figure 19, which compiles the four plots generated by T^2 from the latent layers of the autoencoders. Points of different colors represent the results of different scenarios. In Figure 19, classes 1 to 6 (as shown in Table 5) were reordered in the sequence: 1, 5, 4, 3, 2, and 6, due to the progression of brace removal and subsequent addition.

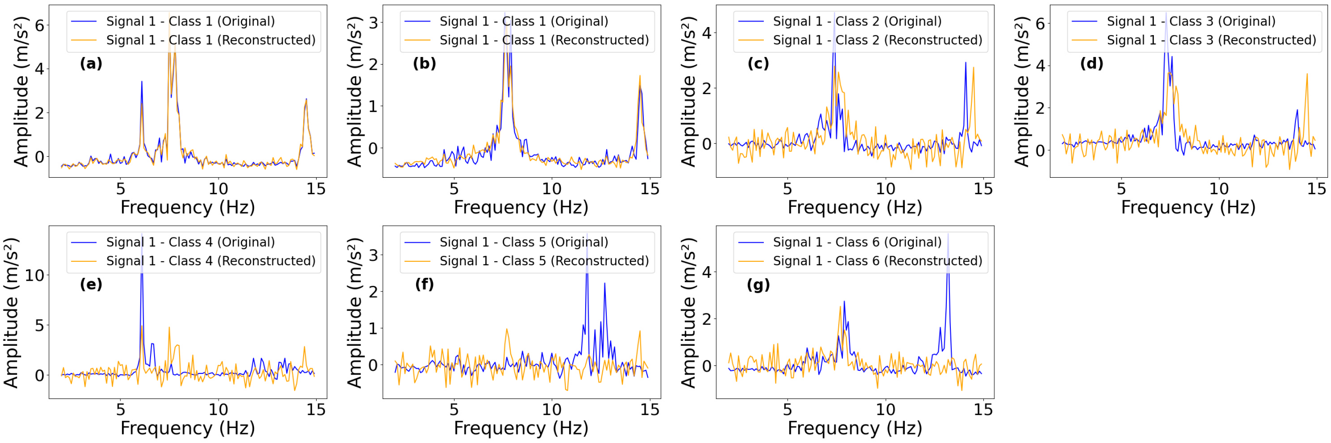


Figure 15. Original and reconstructed signals by AE from Analysis 2 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5; (**g**) Class 6.

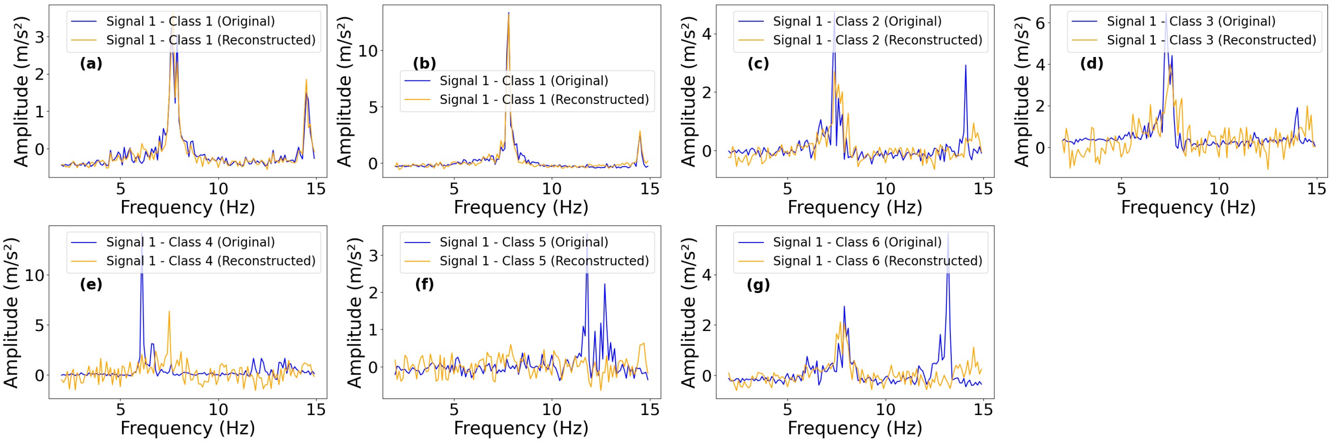


Figure 16. Original and reconstructed signals by SAE from Analysis 2 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5; (**g**) Class 6.

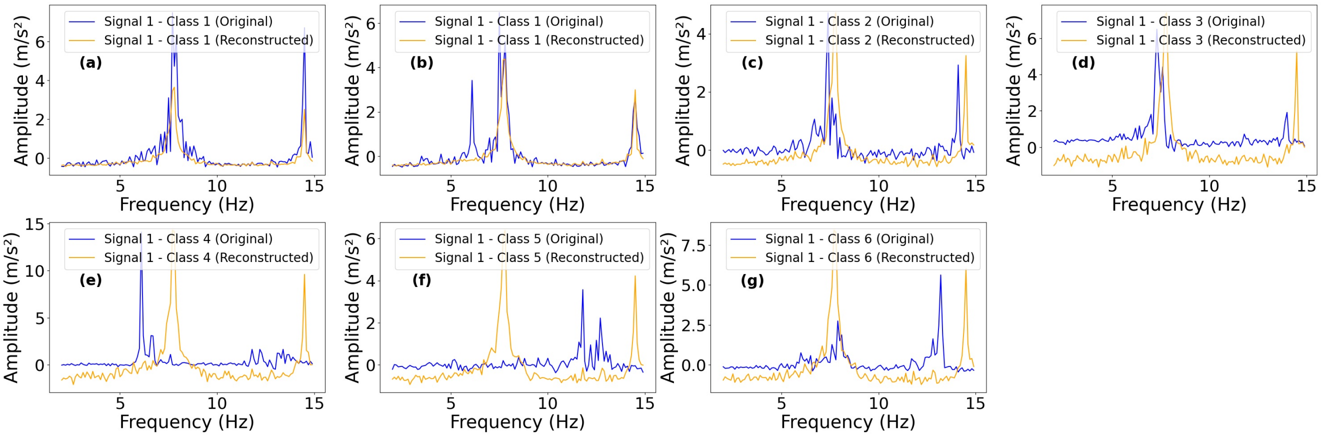


Figure 17. Original and reconstructed signals by VAE from Analysis 2 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5; (**g**) Class 6.

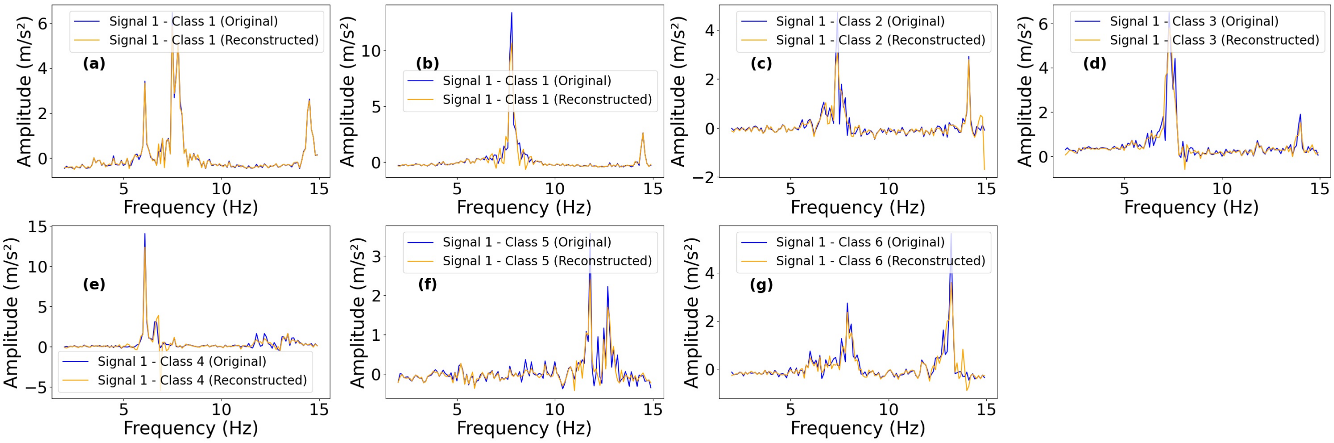


Figure 18. Original and reconstructed signals by CAE from Analysis 2 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4; (**f**) Class 5; (**g**) Class 6.

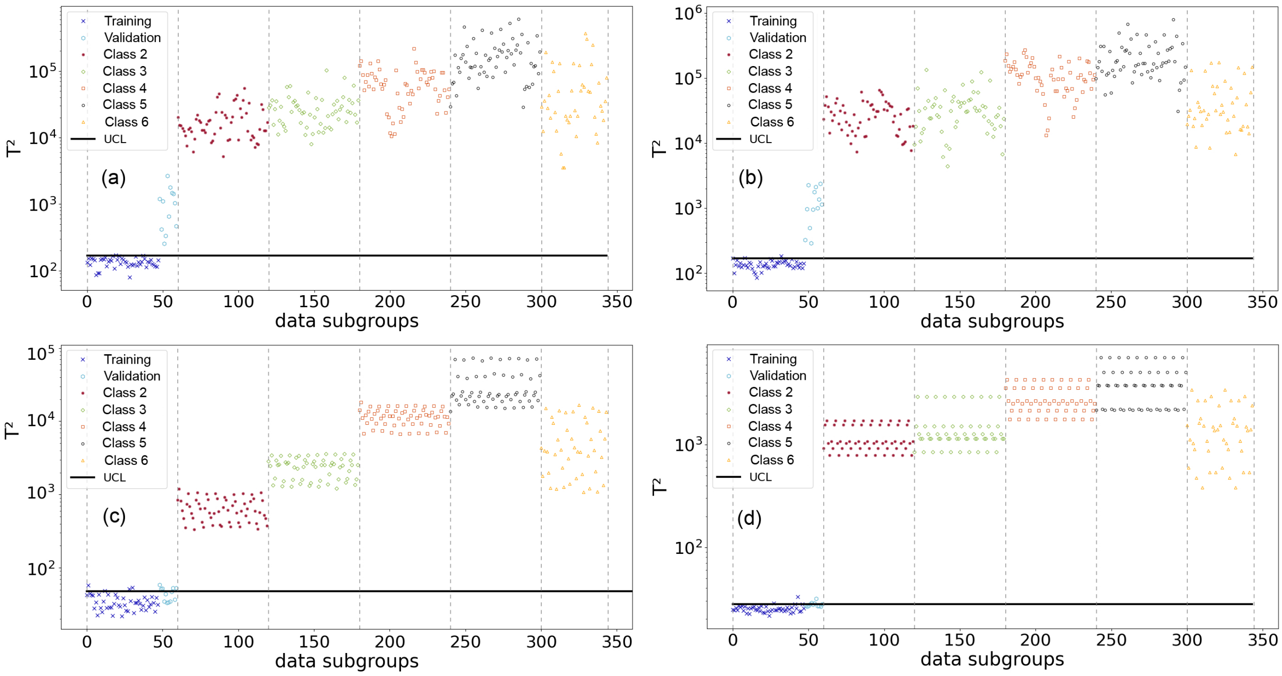


Figure 19. Shewhart T^2 Control Charts for bracing removal (accelerometer 4). (**a**) AE; (**b**) SAE; (**c**) VAE; (**d**) CAE.

In general, one observes that only the variational autoencoder (VAE) exhibited consistent T^2 values among the training and validation classes, highlighting its superiority in classifying new data. This consistency is crucial for a robust model, as it indicates its ability to regroup data from the same structural state with similar T^2 values and discriminate those from different damage scenarios (with higher T^2 values), thus avoiding false alarms of structural changes. Conversely, the discrepancies in the classifications made by the AE, SAE, and CAE (Figure 19b–d) among the training and validation sets suggest that the signals exhibit greater heterogeneity than those in Analysis 1, causing the model to incorrectly identify signals from the fully braced structure as belonging to another class. This difference underscores the effectiveness of the VAE in handling structures subject to temporal variations and behaviors that are not entirely standardized, thus providing a more robust and adaptable approach.

Despite the limitations observed for the conventional autoencoder (AE), sparse autoencoder (SAE), and convolutional autoencoder (CAE), all approaches were able to detect the presence of structural changes, as evidenced by all monitoring classes exceeding the Upper Control Limit (UCL). Furthermore, in all evaluated models, the tests belonging to Class 6 had T^2 values predominated lower than of those of Class 5, indicating a correct identification of the addition of braces (simulation of a structural repair). However, among the three investigated methodologies, the variational autoencoder (VAE) stood out again as the most effective in differentiating various levels of structural changes, demonstrated by the increase in T^2 values as braces were removed. Although there were improvements compared to the results obtained in Analysis 1, the damage quantification task was not as precise for the AE, SAE, and CAE models, with a considerable amount of data from different classes exhibiting similar T^2 values. The results obtained with data from the other accelerometers were similar to this one.

4.2.2. Loosening of Bolts

Each test resulted in a signal of 180,000 sampled points per accelerometer for each case. The signal, comprising 180,000 points, was segmented into 90 signals of 2000 points each before the analyses began, to provide a larger sample set for the models. In this study, a frequency range from 2 to 15 Hz was investigated as it is the most relevant to the structure.

The autoencoder hyperparameters for Analysis 3 are shown in Table 7.

Table 7. Hyperparameters used in the autoencoder models for Analysis 3.

	AE	SAE	VAE	CAE
learning_rate	0.0071	0.0071	0.0071	0.0071
epochs	44	75	75	44
batch_size	13	13	13	13
original_dim	130	130	130	130
intermediate_dim	-	-	51	-
latent_dim	12	12	12	16 × 32
optimizer	Adam	Adam	Adam	Adam
lambda_sparse	-	0.01	-	-

The original and reconstructed FFTs by the AE, SAE, VAE, and CAE for each class, derived from accelerometer 4,are presented in Figures 20, 21, 22, and 23, respectively.

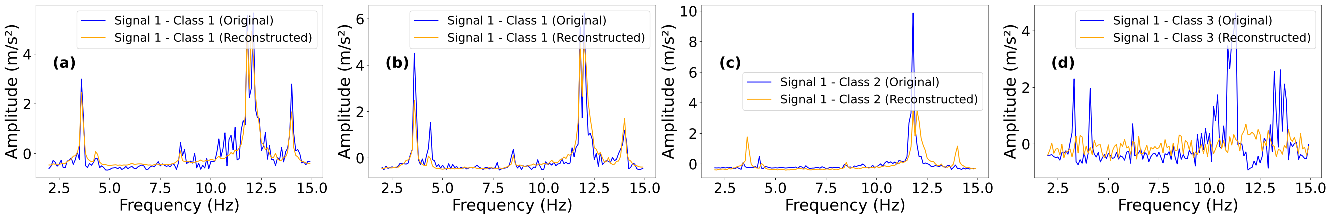


Figure 20. Original and reconstructed signals by AE from Analysis 3 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3.

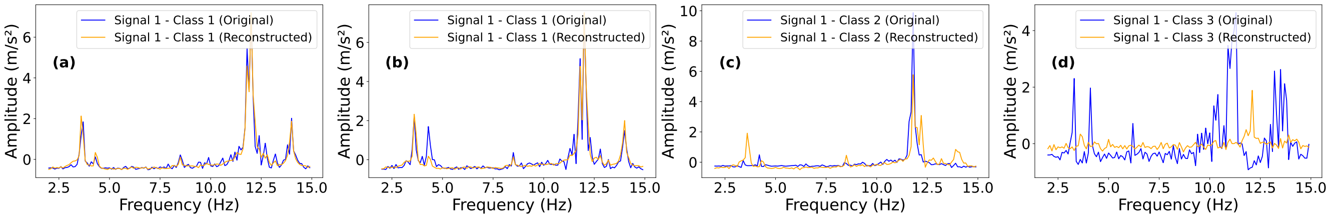


Figure 21. Original and reconstructed signals by SAE from Analysis 3 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3.

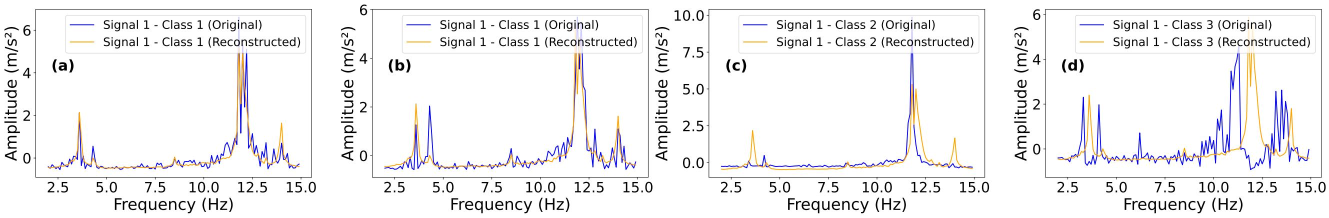


Figure 22. Original and reconstructed signals by VAE from Analysis 3 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3.

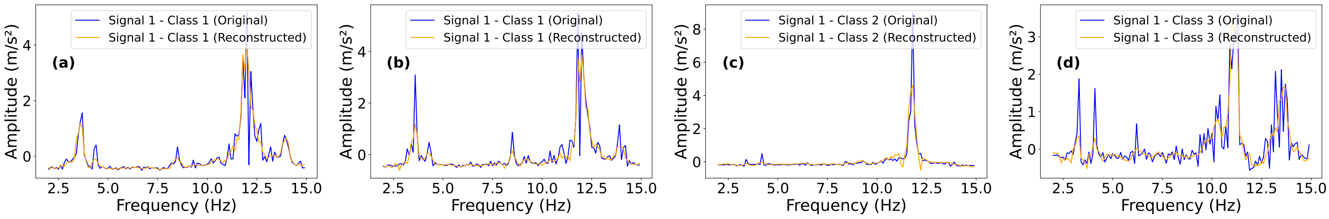


Figure 23. Original and reconstructed signals by CAE from Analysis 3 (accelerometer 4). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3.

In this case, although the reconstructions looked more similar to the original signals in all autoencoders, it is worth noting the greater effectiveness of the CAE in the reconstructions, as it successfully reconstructed the signals from all classes. Nevertheless, the reconstructions were still more similar to the original signals in Class 1 across all autoencoders, indicating that the training was effective.

The results obtained by the models in the bolt loosening analyses, derived from accelerometer 4, are depicted in Figure 24, which compiles the four plots generated by the T^2 from the latent layers of the autoencoders. In it, Classes 7 to 9 shown in Table 5 were rearranged in the sequence 7, 9, and 8, due to the progression of bolts loosening. Class 7 was used as a reference (Class 1), and Classes 9 and 8 were used for monitoring (Class 2 and Class 3), differentiated by color.

For this test, all analyzed methodologies obtained similar T^2 values between the validation and training classes in all 15 accelerometers, confirming the models' capability to classify new data, as both validation and training data belong to the same structural state. Moreover, all methodologies correctly identified the presence of structural alterations, as all monitoring classes exceeded the UCL.

However, the SAE was not able to correctly distinguish the different levels of structural alterations (quantification of alterations), presenting a considerable amount of data from different classes displaying similar T^2 values. On the other hand, the AE, VAE, and CAE models (Figure 24a,c,d) were able to correctly separate signals from different classes, being adequate approaches for the quantification of structural alterations in this analysis.

An assessment of structural alterations in this structure was also proposed by Cardoso et al. [41], employing a methodology focused solely on anomaly detection. Unlike our approach, which effectively localized and quantified structural changes across all classes, Cardoso et al.'s method did not quantify these structural changes and failed to detect damage in some classes, relying heavily on changes in vibration amplitudes for anomaly detection. This highlights the superiority of our current study, which, akin to the first analysis, demonstrated superior capabilities in both the detection and quantification of structural alterations.

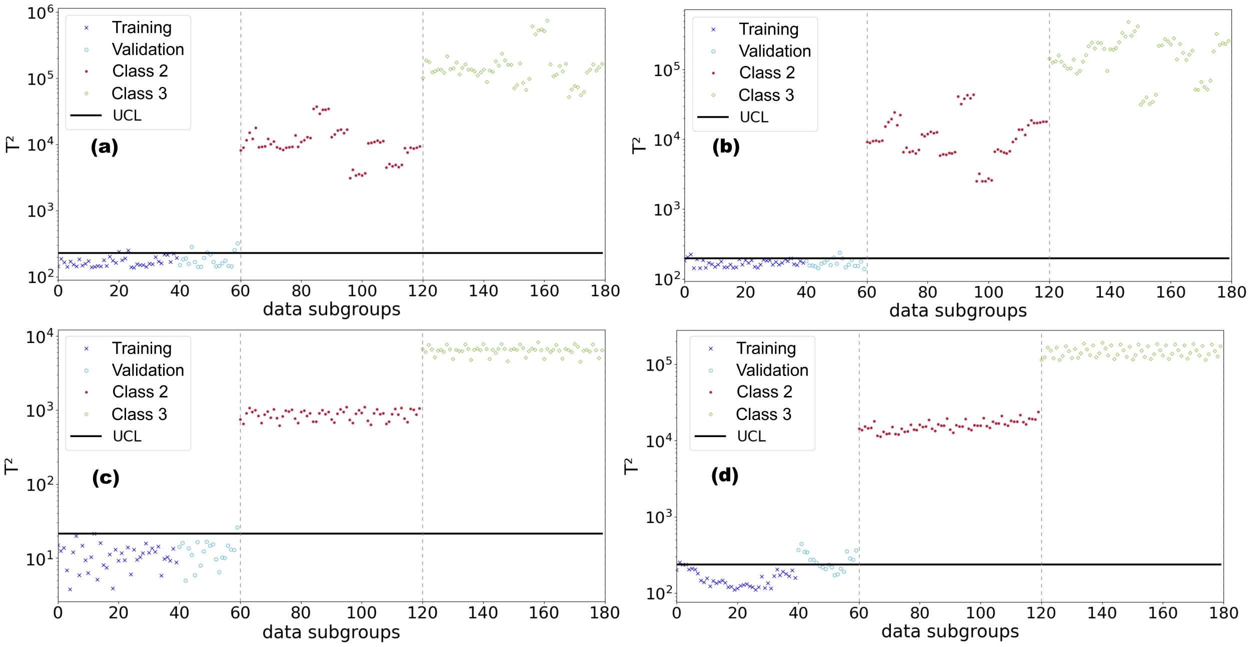


Figure 24. Shewhart T^2 Control Charts for loosening the bolts (accelerometer 4). (**a**) AE; (**b**) SAE; (**c**) VAE; (**d**) CAE.

4.3. The Z24 Bridge

After the tests and the highly promising results obtained in the analyses with ambient vibration, the models were now assessed using data from a real structure under forced vibration, the Z24 Bridge. This structure, depicted in Figure 25, connected the villages of Koppigen and Utzenstorf in Switzerland and was demolished at the end of 1998 due to the construction needs of a nearby railway, which required replacement with a new bridge with a larger lateral span.

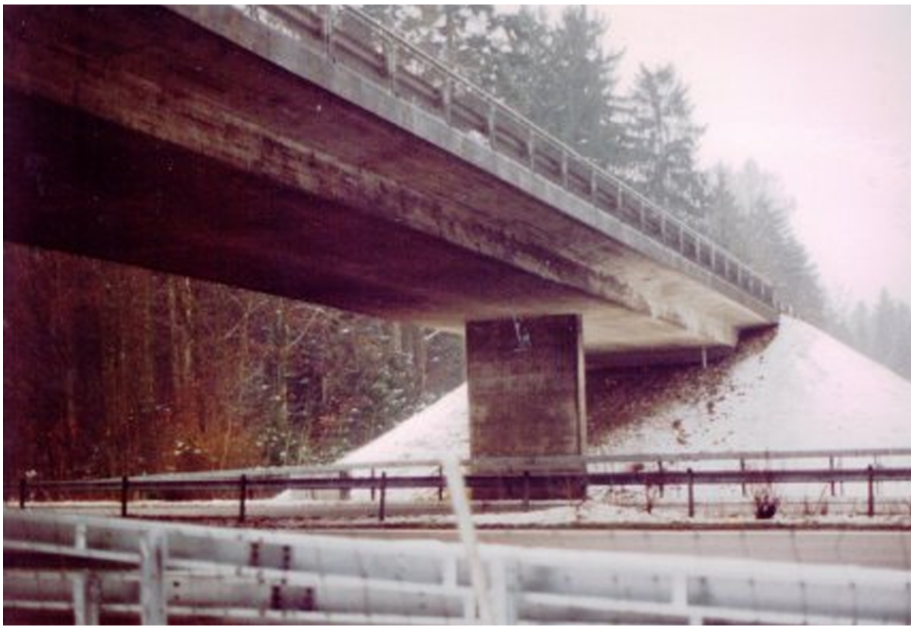


Figure 25. Section of Z24 Bridge.

This prestressed concrete structure with 16 cables consisted of a main span of 30 m, two side spans of 14 m each, and two cantilever beams of 2.7 m each, totaling a length of 63.4 m, as depicted in Figure 26. Due to the limited number of available accelerometers, vibrations were recorded separately in nine configurations, thus covering the entire span of the bridge. Accelerometers were positioned on the bridge deck along three parallel measurement lines: one located at the centerline and the other two situated along both sidelines. To simulate damage, the pier on the Koppigen side was cut, and concrete was replaced with steel filler plates and three hydraulic jacks. This installed system could lower the pier to simulate real causes of damage, such as subsurface settling and erosion.

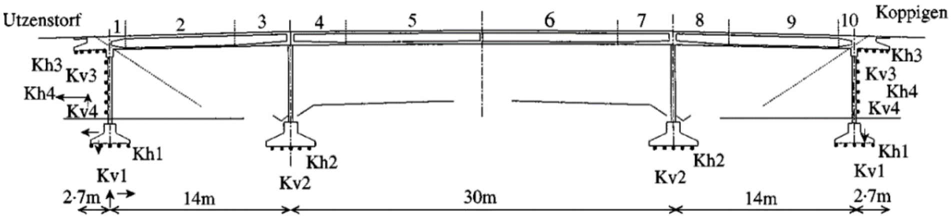


Figure 26. Sketch of Z24 Bridge. Adapted from [42].

Before its demolition, over the course of a month, the bridge underwent various progressive damage tests, including continuous short- and long-term monitoring tests, with forced and ambient vibrations, using sensors to assess variations in temperature and air humidity, rainfall presence, and wind speed and direction [42,43].

In this paper, forced vibration tests obtained from four damage scenarios under different temperature conditions were analyzed. The main goal is to evaluate and compare the robustness of autoencoders in detecting structural changes under such circumstances as well. Table 8 describes each scenario.

Table 8. Structural scenarios evaluated in Analysis 4. Source: [42].

Scenario	Configuration	Temperature
1	Undamaged structure	17 °C
2	Undamaged structure (with the addition of hydraulic jacks)	26 °C
3	Settlement of 40 mm in the indicated pier	29 °C
4	Settlement of 80 mm in the indicated pier	26 °C

For each damage scenario, nine tests were conducted using five accelerometers placed at three measurement points (denoted as R1, R2, and R3 in Figure 27).The vibrational responses from these accelerometers were sampled at 65,835 points, collected over approximately 11 min at a sampling frequency of 100 Hz. Each of the nine vibrational response signals was segmented into 65 signals with 1000 points each, resulting in a total of 585 signals of 1000 points to provide a larger sample set for the models. This study focused on a frequency range from 5 to 25 Hz.

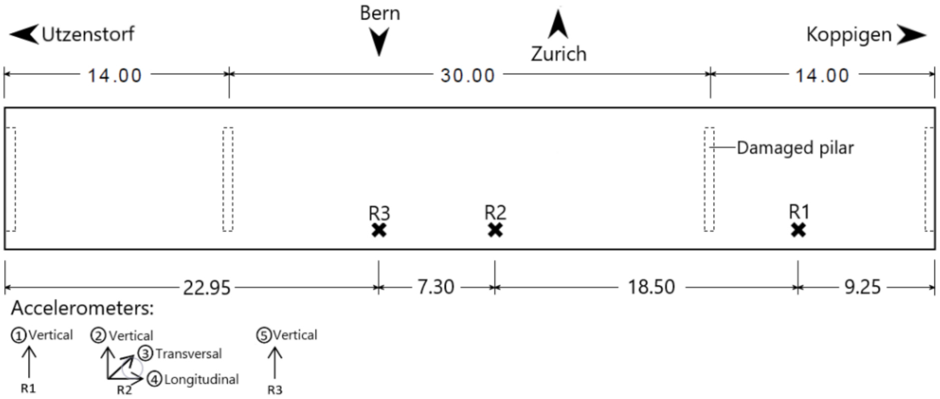


Figure 27. Top view of the bridge with the damaged pier indicated. Source: [42].

The hyperparameters used in configuring the autoencoders for Analysis 4 are detailed in Table 9.

Table 9. Hyperparameters used in the autoencoder models for Analysis 4.

	AE	SAE	VAE	CAE
learning_rate	0.00008231	0.00006624	0.00008638	0.00008658
epochs	69	64	68	62
batch_size	16	18	18	16
original_dim	200	200	200	200
intermediate_dim	-	-	140	-
latent_dim	105	110	102	100 × 32
optimizer	Adam	Adam	Adam	Adam
lambda_sparse	-	0.1	-	-

The original and reconstructed FFTs by the AE, SAE, VAE, and CAE for each class, derived from accelerometer 2, are presented in Figures 28, 29, 30, and 31, respectively.

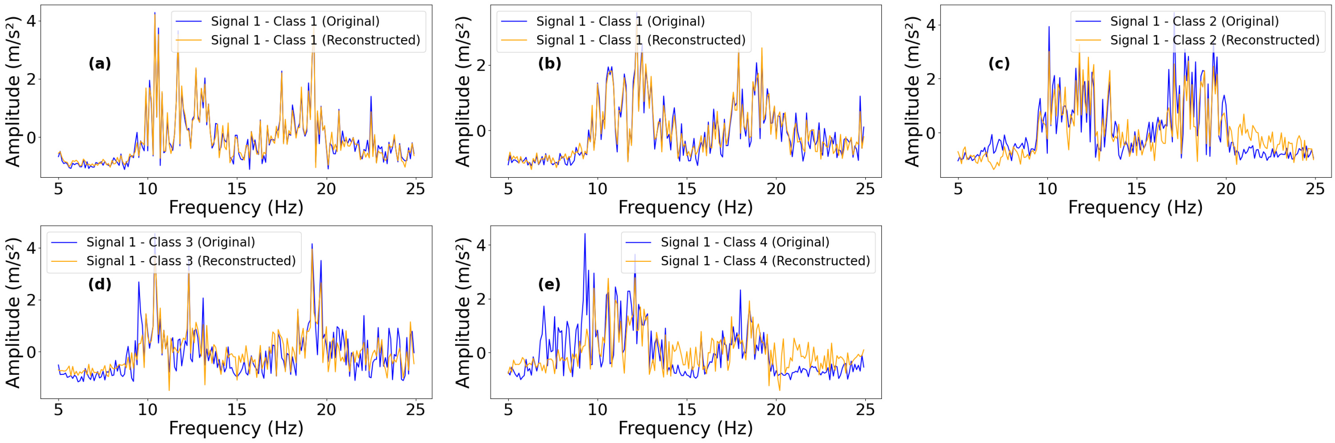


Figure 28. Original and reconstructed signals by AE from Analysis 4 (accelerometer 2). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4.

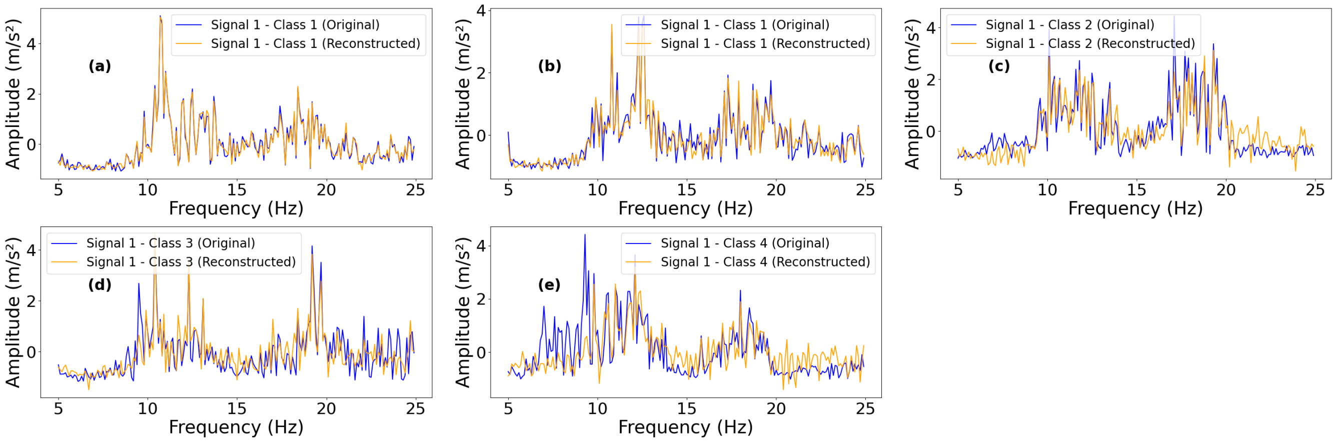


Figure 29. Original and reconstructed signals by SAE from Analysis 4 (accelerometer 2). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4.

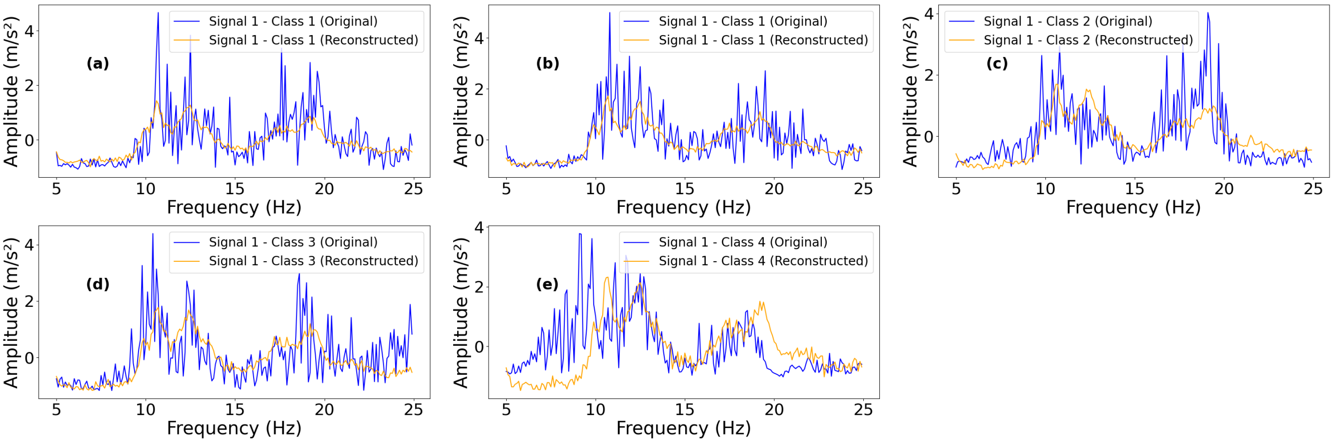


Figure 30. Original and reconstructed signals by VAE from Analysis 4 (accelerometer 2). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4.

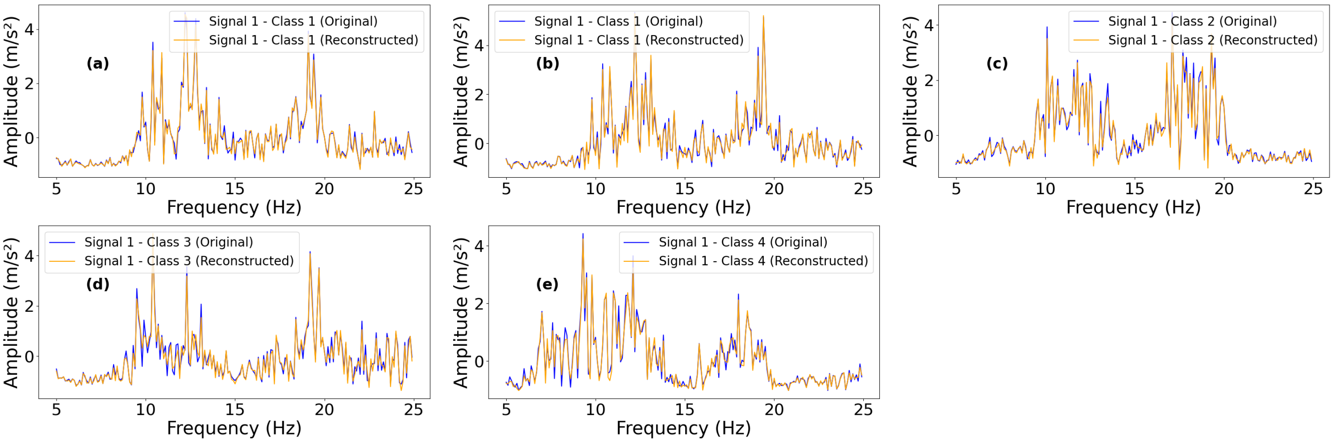


Figure 31. Original and reconstructed signals by CAE from Analysis 4 (accelerometer 2). (**a**) Class 1 train; (**b**) Class 1 validation; (**c**) Class 2; (**d**) Class 3; (**e**) Class 4.

Except for the CAE, which reconstructed signals from all classes very well, the reconstructions made by the autoencoders were more accurate for Class 1 and degraded as the differences between the signals of the analyzed class and Class 1 increased. This suggests potential for correctly identifying structural changes through comparison of the features stored in the latent layers of the autoencoders for each class.

The results obtained by the models in Analysis 4, derived from accelerometer 2, are presented in Figure 32, which compiles the four plots generated by the T^2 from the latent layers of the autoencoders. Monitoring points of different colors represent the results of distinct scenarios in the sequence presented in Table 8.

As Figure 32 shows, all methodologies yielded similar T^2 values between the validation and training classes, indicating a good ability of the models to classify new data. Furthermore, all methodologies correctly identified the presence of structural alterations, as all monitoring classes exceeded the UCL. However, the VAE was the methodology that best quantified structural changes, with T^2 values increasing as changes in relation to the original state increased.

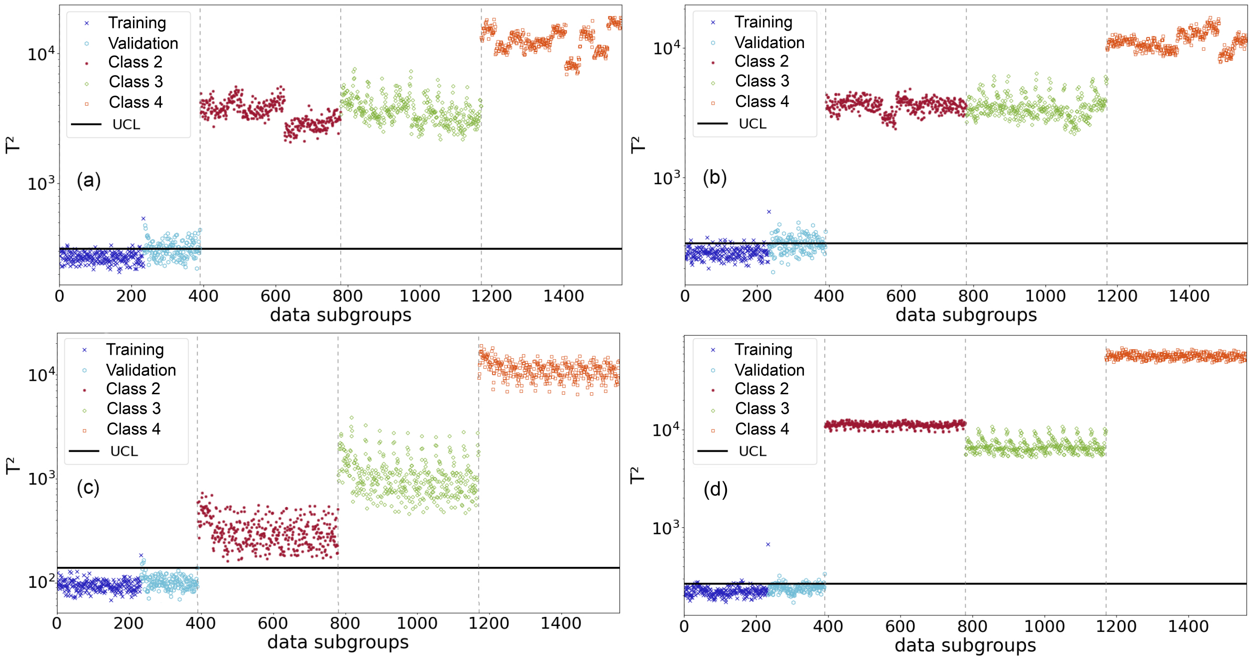


Figure 32. Shewhart T^2 Control Charts for Analysis 4 (accelerometer 2). (**a**) AE; (**b**) SAE; (**c**) VAE; (**d**) CAE.

Even with the VAE-T^2 model, some T^2 values for Class 2 were comparable to those of Class 3. This likely occurred due to temperature variations between the analyzed classes, as the natural frequencies of a concrete structure tend to be significantly higher at lower temperatures [55]. It is also important to note that variations in wind intensity and loads exerted on the structure also interfere with structural vibrations, but even with these factors, the VAE was able to quantify the structural damages satisfactorily.

As with accelerometer 2, this quantification of structural damage was also observed in accelerometers 1 and 5 (transverse). For accelerometers 3 and 4 (transverse and longitudinal), all models maintained similar T^2 values between the validation and training classes. However, the separation between the signals of Classes 2, 3, and 4 was not as evident for the set of hyperparameters used. This demonstrates the influence of the positioning and direction of the accelerometers on the models, indicating the need for specific parameter optimization for structural quantification based on the data from these accelerometers.

Finotti et al. [56] conducted a similar study on the Z24 Bridge using time domain signals and employing only the SAE for feature extraction, comparing the intact structure with the settlement of 40 mm in one pier (scenario 3 in this subsection of the present study). This previous assessment successfully identified the settlement for data from accelerometers 1 and 5, but failed to detect structural damage in data from the other accelerometers. This limitation is likely due to the use of time domain data as the input for the SAE, without performing damage quantification. In contrast, our current study represents a significant advancement by utilizing frequency domain data and quantifying structural alterations across four damage scenarios, providing a more comprehensive assessment of the structural health.

In summary, significant differences in the latent layers of the autoencoder model can indicate two possible scenarios: (1) the AE model did not fully understand the underlying problem, leading to inaccurate latent representations, or (2) the AE model effectively captured the underlying patterns, and the observed discrepancies in the latent layers indicate anomalies or structural changes. The latter scenario is the basis for most studies employing AEs for structural health monitoring (SHM).

When the AE model is trained and subsequently tested on the same dataset representing a healthy state of the structure, the differences in the latent layers, as visualized through the T^2 Control Charts, are typically small. This is because the model has learned to represent the normal conditions accurately. However, when the model is trained on healthy data and then tested on new data representing a damaged state of the structure, the differences in the latent layers are expected to be larger. This is due to the fact that the model, having been trained only on healthy data, will produce latent representations that deviate significantly from the learned healthy patterns, thereby highlighting potential damage or anomalies.

In our study, this principle was consistently observed across the various applications presented. Specifically, for Classes 2, 3, 4, and so on, when present and representing damaged structural states, larger differences in the latent layers were noted, corresponding to increasingly severe structural alterations. The progressively larger deviations in the latent layers, as visualized through the T^2 Control Charts, highlight the effectiveness of the four different autoencoders used in the study, especially the variational autoencoder (VAE). This trend is consistent with the principle that an AE model trained on healthy data will exhibit greater deviations in the latent layers when faced with data from damaged states.

The ability of the model to detect and point out these deviations is crucial for effective structural health monitoring, as it allows for timely identification and assessment of the extent of structural alterations. This analysis reinforces the applicability of AEs in SHM, where the primary objective is to detect and quantify structural changes through deviations in the latent layers. The clear correlation between the severity of structural alterations and the magnitude of deviations in the latent layers in our results demonstrates the potential of these models, particularly the VAE, as robust methods for structural health monitoring.

4.4. Time Comparisons

The procedures were rigorously executed at all stages, from data importation to analysis through the repetition of training, validation, and testing processes, followed by the T^2 damage index evaluation. The procedures were repeated five times for each accelerometer in all considered structures. This time-averaged approach was adopted to avoid any inaccurate or incorrect results, thus ensuring a robust and reliable evaluation, as well as providing a better understanding of the execution time of each model.

Table 10 displays the average execution time of each model, calculated by averaging the five repetitions performed for each accelerometer in each analysis.

Table 10. Average execution time of each methodology.

	Analysis 1 (1500 Signals)	**Analysis 2 (172 Signals)**	**Analysis 3 (270 Signals)**	**Analysis 4 (2340 Signals)**
AE	40.352 s	30.156 s	26.782 s	77.789 s
SAE	45.224 s	33.845 s	29.919 s	76.792 s
VAE	52.211 s	48.142 s	35.935 s	137.205 s
CAE	42.116 s	31.922 s	31.115 s	98.433 s

In all conducted analyses, the VAE took longer to execute all steps. This time discrepancy was more evident when there was a greater number of signals available, i.e., more samples. This characteristic is due to the complexity of VAE models that involves modeling probabilistic distributions, thus increasing the execution time compared to the AE, SAE, and CAE models.

Furthermore, the execution times of the AE and SAE models were similar, with the SAE model requiring more computational time in most tests. This occurs due to the sparsity constraint applied to the hidden layer of the SAE, which increases computational complexity compared to the AE's. Therefore, it is expected that the execution times of the SAE would be slightly higher than those of the AE in most cases. Additionally, the CAE required a longer execution time than the SAE and AE, but less than the VAE, due to the presence of convolutional layers, which require a higher computational cost than the other autoencoders.

However, it is important to note that, as all execution times were relatively low, computational cost does not hinder the use of any of these methodologies. Thus, among these four approaches, rather than prioritizing computational efficiency, the choice of the best methodology should be based primarily on the quality of the results obtained. This consideration ensures the excellence of the results, contributing to a more robust and effective approach in the practical application of autoencoder models in civil engineering structures.

5. Conclusions

In this study, the effectiveness of four distinct methodologies based on the combination of autoencoders and the Hotelling T^2 statistical tool for the detection and quantification of structural alterations in various structures and configurations was investigated. The performance of these methodologies was comparatively evaluated in terms of correctly identifying anomalies and computational efficiency.

Among the four autoencoder models, the VAE-T^2 significantly outperformed the other models in all analyzed structural states. While the VAE-T^2 demonstrated robust capability in identifying and quantifying structural alterations, the AE, SAE, and CAE models revealed inferior performances and generalization difficulties, including incorrectly classifying validation signals when the structural responses of the same class were not predominantly homogeneous. Furthermore, even though it required more time to perform the analyses, this factor was not considered a hindrance to the use of the VAE, as current technologies allow the entire methodology to be executed in a few minutes.

As with any data-driven method for structural damage detection, the effectiveness of the methods discussed in this study heavily relies on data quality. The analysis of the Z24 Bridge demonstrates the impact of sensor position and orientation on the data collected by accelerometers. Ensuring sensors are properly calibrated and maintained is essential for continuous measurement reliability and accuracy. Therefore, it is fundamental to ensure that sensors capturing structural dynamics (such as accelerations, strains, or displacements) are correctly positioned, covering critical areas of interest (such as potential damage locations or areas with significant amplitude variations). To this end, preliminary finite-element analyses can be employed to optimize sensor placement.

The AE, SAE, and CAE models consistently exhibited poorer performance compared to the VAE model, primarily because these autoencoders were unable to model the underlying structure's behavior. Additionally, we draw attention to the importance of optimizing the AE models' hyperparameters for each application to ensure that each method delivers its best—or even optimal—results. As seen in Section 4, the optimization algorithm yielded consistently different results for each model and each structure.

Future research directions could focus on developing real-time SHM systems that continuously monitor structures and provide immediate alerts upon detecting anomalies. Integration with Internet of Things (IoT) devices and cloud computing could streamline large-scale deployment and data analysis. Additionally, exploring advanced machine learning techniques such as reinforcement learning and generative adversarial networks (GANs) could contribute further to the evolution of structural health monitoring methodologies. Furthermore, incorporating time–frequency analysis techniques in future research could allow for a more comprehensive assessment of structural health by capturing both gradual and rapid changes in dynamic behavior. Future studies could also explore architectural modifications, such as adding more layers or neurons, to potentially enhance the efficacy of autoencoder models in structural health monitoring applications.

In conclusion, ongoing advancements in these methodologies not only deepen our understanding of structural behavior mechanisms, but also offer potential applications in locating structural alterations and assessing remaining structural life. This predictive capability can guide strategic and proactive decisions, allowing for the more effective implementation of preventive and corrective measures. Therefore, this research not only contributes to the advancement of structural health monitoring (SHM), but also enables new perspectives aiming to reduce risks and ensure its long-term safety, durability, and economy.

Author Contributions: Conceptualization, A.C.; Methodology, M.S.N., R.F. and A.C.; Validation, M.S.N.; Formal analysis, M.S.N. and R.F.; Investigation, F.B.; Resources, R.F.; Data curation, M.S.N.; Writing—original draft, M.S.N.; Writing—review & editing, F.B. and A.C.; Supervision, A.C.; Project administration, F.B. and A.C.; Funding acquisition, F.B. and A.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financed by CAPES (Finance Code 0001), Conselho Nacional de Desenvolvimento Científico e Tecnológico—CNPq (Brazil)—Grants CNPq/ FNDCT/ MCTI 407256/2022-9, 402533/2023-2, 303982/2022-5, and 308008/2021-9 and Fundação de Amparo à Pesquisa do Estado de Minas Gerais—FAPEMIG—Grant BPD-00080-22.

Data Availability Statement: For the two-story slender aluminum frame, the dataset is described in the work of Finotti et al. [33] and available at the SHM-UFJF repository (http://bit.ly/SHM-UFJF, accessed on 1 March 2024). For the 3D Yellow Frame, the dataset is described in the works of Dyke et al. [39], Dyke et al. [40], and Cardoso et al. [41] and available at [40]. For the Z24 bridge, the dataset is described in the works of Peeters and De Roeck [42] and Teughels and De Roeck [43] and available at https://bwk.kuleuven.be/bwm/z24, accessed on 1 March 2024). Finally, all codes used in this study are available at https://bit.ly/SHM-CODES, accessed on 1 March 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sousa, H.; Courage, W. Value of information of a pro-active SHM-based tool supported by advanced FE models and Bayesian statistics towards efficiency in bridge management. *Struct. Infrastruct. Eng.* **2021**, *18*, 554–572. [CrossRef]
2. Wang, X.; Ni, Y.Q.; Lin, K.C. Comparison of statistical counting methods in SHM-based reliability assessment of bridges. *J. Civ. Struct. Health Monit.* **2015**, *5*, 275–286. [CrossRef]
3. Carrión, F.J.; Quintana, J.A.; Crespo, S.E. Techno-economical and practical considerations for SHM systems. *J. Civ. Struct. Health Monit.* **2017**, *7*, 207–215. [CrossRef]
4. Avci, O.; Abdeljaber, O.; Kiranyaz, S.; Hussein, M.; Gabbouj, M.; Inman, D.J. A Review of Vibration-Based Damage Detection in Civil Structures: From Traditional Methods to Machine Learning and Deep Learning Applications. *Mech. Syst. Signal Process.* **2021**, *147*, 107077. [CrossRef]
5. Rytter, A. Vibration Based Inspection of Civil Engineering Structures. Ph.D. Dissertation, Department of Building Technology and Structural Enginnering, Aalborg University, Aalborg, Denmark, 1993.
6. Nunes, L.A.; Amaral, R.P.F.; Barbosa, F.S.; Cury, A.C. A hybrid learning strategy for structural damage detection. *Struct. Health Monit.* **2021**, *20*, 2143–2160. [CrossRef]
7. Finotti, R.P.; Barbosa, F.d.S.; Cury, A.A.; Pimentel, R.L. Numerical and Experimental Evaluation of Structural Changes Using Sparse Auto-Encoders and SVM Applied to Dynamic Responses. *Appl. Sci.* **2021**, *11*, 11965. . [CrossRef]
8. Finotti, R.P.; Barbosa, F.d.S.; Cury, A.A.; Pimentel, R.L. Novelty Detection Using Sparse Auto-Encoders to Characterize Structural Vibration Responses. *Arab J. Sci. Eng.* **2022**, *47*, 13049–13062. [CrossRef]
9. Liu, G.; Niu, Y.; Zhao, W.; Duan, Y.; Shu, J. Data anomaly detection for structural health monitoring using a combination network of GANomaly and CNN. *Smart Struct. Syst.* **2022**, *29*, 53–62. [CrossRef]
10. Rosso, M.M.; Aloisio, A.; Cirrincione, G.; Marano, G.C. Subspace features and statistical indicators for neural network-based damage detection. *Structures* **2023**, *56*, 104792. [CrossRef]
11. Eltouny, K.A.; Liang, X. Bayesian-optimizer unsupervised learning approach for structural damage detection. *Comput. Aided Civ. Inf. Eng.* **2021**, *36*, 1249–1269. [CrossRef]
12. Ali, L.; Alnajjar, F.; Jassmi, H.A.; Gocho, M.; Khan, W.; Serhani, M.A. Performance Evaluation of Deep CNN-Based Crack Detection and Localization Techniques for Concrete Structures. *Sensors* **2021**, *21*, 1688. [CrossRef] [PubMed]
13. Resende, L.; Finotti, R.; Barbosa, F.; Garrido, H.; Cury, A.; Domizio, M. Damage identification using convolutional neural networks from instantaneous displacement measurements via image processing. *Struct. Health Monit.* **2024**, *23*, 1627–1640. [CrossRef]
14. Dong, C.-Z.; Catbas, F.N. A review of computer vision–based structural health monitoring at local and global levels. *Struct. Health Monit.* **2021**, *20*, 692–743. [CrossRef]
15. Ditommaso, R.; Ponzo, F.C. Automatic evaluation of the fundamental frequency variations and related damping factor of reinforced concrete framed structures using the Short Time Impulse Response Function (STIRF). *Eng. Struct.* **2015**, *82*, 104–112. [CrossRef]
16. Finotti, R.P.; Barbosa, F.d.S.; Cury, A.A.; Gentile, C. A novel natural frequency-based technique to detect structural changes using computational intelligence. *Procedia Eng.* **2017**, *199*, 3314–3319. [CrossRef]
17. Ditommaso, R.; Iacovino, C.; Auletta, G.; Parolai, S.; Ponzo, F.C. Damage Detection and Localization on Real Structures Subjected to Strong Motion Earthquakes Using the Curvature Evolution Method: The Navelli (Italy) Case Study. *Appl. Sci.* **2021**, *11*, 6496. [CrossRef]

18. Anastasopoulos, D.; De Roeck, G.; Reynders, E.P. One-year operational modal analysis of a steel bridge from high resolution macrostrain monitoring: Influence of temperature vs. retrofitting. *Mech. Syst. Signal Process.* **2021**, *161*, 107951. [CrossRef]
19. Wah, W.S.L.; Chen, Y.T.; Owen, J.S. A regression-based damage detection method for structures subjected to changing environmental and operational conditions. *Eng. Struct.* **2021**, *228*, 111462. [CrossRef]
20. Wang, Z.; Yang, D.H.; Yi, T.H.; Zhang, G.H.; Han, J.G. Eliminating environmental and operational effects on structural modal frequency: A comprehensive review. *Struct. Control Health Monit.* **2022**, *29*, e3073. [CrossRef]
21. Bertagnoli, G.; Ferrara, M.; Lucà, F.; Cigada, A. Effect of Environmental Parameters on Structural Health Status Assessment Using OMA Techniques. *Appl. Sci.* **2023**, *13*, 1477. [CrossRef]
22. Ponzo, F.C.; Auletta, G.; Ielpo, P.; Ditommaso, R. DInSAR–SBAS satellite monitoring of infrastructures: How temperature affects the "Ponte della Musica" case study. *J. Civ. Struct. Health Monit.* **2024**, *14*, 745–761. [CrossRef]
23. Alves, V.; Cury, A. A fast and efficient feature extraction methodology for structural damage localization based on raw acceleration measurements. *Struct. Control Health Monit.* **2021**, *28*, e2748. [CrossRef]
24. Alves, V.; Cury, A. An automated vibration-based structural damage localization strategy using filter-type feature selection. *Mech. Syst. Signal Process.* **2023**, *190*, 110145. [CrossRef]
25. Perfetto, D.; De Luca, A.; Perfetto, M.; Lamanna, G.; Caputo, F. Damage Detection in Flat Panels by Guided Waves Based Artificial Neural Network Trained through Finite Element Method. *Materials* **2021**, *14*, 7602. [CrossRef]
26. Mojtahedi, A.; Lotfollahi Yaghin, M.A.; Hassanzadeh, Y.; Ettefagh, M.M.; Aminfar, M.H.; Aghdam, A.B. Developing a robust SHM method for offshore jacket platform using model updating and fuzzy logic system. *Appl. Ocean Res.* **2011**, *33*, 398–411. [CrossRef]
27. Qu, S.; Jiao, X.; Chen, X.; Zhao, J. Response Prediction Analysis Of RC Frame Structures Using Support Vector Machine Algorithm. *J. Appl. Sci. Eng.* **2024**, *28*, 345–353. [CrossRef]
28. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2013**, arXiv:1312.6114. [CrossRef]
29. Gao, F.; Hua, J. Damage characterization using CNN and SAE of broadband Lamb waves. *Ultrasonics* **2022**, *119*, 106592. [CrossRef]
30. Abbassi, A.; Römgens, N.; Tritschel, F.F.; Penner, N.; Rolfes, R. Evaluation of machine learning techniques for structural health monitoring using ultrasonic guided waves under varying temperature conditions. *Struct. Health Monit.* **2023**, *22*, 1308–1325. [CrossRef]
31. Abbas, N.; Umar, T.; Salih, R.; Akbar, M.; Hussain, Z.; Haibei, X. Structural Health Monitoring of Underground Metro Tunnel by Identifying Damage Using ANN Deep Learning Auto-Encoder. *Appl. Sci.* **2023**, *13*, 1332. [CrossRef]
32. Eltouny, K.; Gomaa, M.; Liang, X. Unsupervised Learning Methods for data-driven vibration-based structural health monitoring: A review. *Sensors* **2023**, *23*, 3290. [CrossRef]
33. Finotti, R.P.; Silva, C.F.; Oliveira, P.H.E.; Barbosa, F.d.S.; Cury, A.A.; Silva, R.C. Novelty detection on a laboratory benchmark slender structure using an unsupervised deep learning algorithm. *Lat. Am. J. Solids Struct.* **2023**, *20*, e512. [CrossRef]
34. Yang, Z.; Xu, B.; Luo, W.; Chen, F. Autoencoder-based representation learning and its application in intelligent fault diagnosis: A review. *Measurement* **2022**, *189*, 110460. [CrossRef]
35. Mou, M.; Zhao, X.; Liu, K.; Hui, Y. Variational Autoencoder based on distributional semantic embedding and cross-modal reconstruction for generalized zero-shot fault diagnosis of industrial processes. *Process Saf. Environ. Prot.* **2023**, *177*, 1154–1167. [CrossRef]
36. Ditommaso, R.; Mucciarelli, M.; Ponzo, F.C. Analysis of non-stationary structural systems by using a band-variable filter. *Bull. Earthq. Eng.* **2012**, *10*, 895–911. [CrossRef]
37. Bayane, I.; Leander, J.; Karoumi, R. An unsupervised machine learning approach for real-time damage detection in bridges. *Eng. Struct.* **2024**, *308*, 117971. [CrossRef]
38. Dang, H.V.; Tran-Ngoc, H.; Nguyen, T.V.; Bui-Tien, T.; De Roeck, G.; Nguyen, H.X. Data-Driven Structural Health Monitoring Using Feature Fusion and Hybrid Deep Learning. *IEEE Trans. Autom. Sci. Eng.* **2020**, *18*, 2087–2103. [CrossRef]
39. Dyke, J.; Bernal, D.; Beck, J.; Ventura, C. Experimental Phase II of the Structural Health Monitoring Benchmark Problem. 2003. Available online: https://core.ac.uk/download/pdf/216164432.pdf (accessed on 1 March 2024).
40. Mou, M.; Zhao, X.; Liu, K.; Hui, Y. NEES: Database for Structural Control and Monitoring Benchmark Problems. 2015. Available online: https://datacenterhub.org/resources/257 (accessed on 1 March 2024).
41. Cardoso, R.A.; Cury, A.; Barbosa, F. Automated real-time damage detection strategy using raw dynamic measurements. *Eng. Struct.* **2019**, *196*, 109364. [CrossRef]
42. Peeters, B.; De Roeck, G. One year monitoring of the z24-bridge: Environmental influences versus damage events. *Earthq. Eng. Struct. Dyn.* **2001**, *30*, 149–171. [CrossRef]
43. Teughels, A.; De Roeck, G. Structural damage identification of the highway bridge Z24 by FE model updating. *J. Sound Vib.* **2004**, *278*, 589–610. [CrossRef]
44. Montgomery, D. *Introduction to Statistical Quality Control*; John Wiley & Sons: Hoboken, NJ, USA, 2009.
45. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
46. Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 6999–7019. [CrossRef] [PubMed]
47. Qi, J.; Du, J.; Siniscalchi, S.M.; Ma, X.; Lee, C.H. On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression. *IEEE Signal Process. Lett.* **2020**, *27*, 1485–1489. [CrossRef]
48. Lipton, Z. The Mythos of Model Interpretability. *Commun. ACM* **2018**, *61*, 36–43. [CrossRef]

49. Resende, L.; Finotti, R.P.; Barbosa, F.S.; Cury, A.A. Structural damage detection with autoencoding neural networks. In Proceedings of the XLIII Ibero-Latin American Congress on Computational Methods in Engineering, Foz do Iguaçu, Brazil, 11–14 November 2022.
50. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19), Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631. [CrossRef]
51. Smith, L.N. A disciplined approach to neural network hyper-parameters: Part 1—Learning rate, batch size, momentum, and weight decay. *arXiv* **2018**, arXiv:1803.09820. [CrossRef]
52. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representation, San Diego, CA, USA, 7–9 May 2015. . [CrossRef]
53. Bergstra, J.; Bardenet, R.; Bengio, Y.; Kégl, B. Algorithms for hyper-parameter optimization. In Proceedings of the 24th International Conference on Neural Information Processing Systems, Granada Spain, 12–15 December 2011; pp. 2546–2554. Available online: https://dl.acm.org/doi/10.5555/2986459.2986743 (accessed on 1 March 2024).
54. Bergstra, J.; Bengio, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **2013**, *13*, 281–305. Available online: http://jmlr.org/papers/v13/bergstra12a.html (accessed on 1 March 2024).
55. Cai, Y.; Zhang, K.; Ye, Z.; Liu, C.; Lu, K.; Wang, L. Influence of Temperature on the Natural Vibration Characteristics of Simply Supported Reinforced Concrete Beam. *Sensors* **2021**, *21*, 4242. [CrossRef]
56. Finotti, R.; Gentile, C.; Barbosa, F.; Cury, A. Vibration-based anomaly detection using sparse auto-encoder and control charts. In Proceedings of the XI International Conference on Structural Dynamics, Athens, Greece, 23–26 November 2020; pp. 1335–1347. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.