

Article

A Comparison of Deep Learning Algorithms for Anomaly Detection in Discrete Mechanical Systems

Francesco Morgan Bono ^{*}, Luca Radicioni , Simone Cinquemani  and Gianluca Bombaci 

Mechanical Department, Politecnico di Milano, 20156 Milano, Italy; luca.radicioni@polimi.it (L.R.); simone.cinquemani@polimi.it (S.C.); gianluca.bombaci@mail.polimi.it (G.B.)

* Correspondence: francescomorgan.bono@polimi.it

Abstract: The application of intelligent systems for structural health monitoring is investigated. A change in the nominal configuration can be related to a structural defect that has to be monitored before it reaches a critical condition. Evidently, the ability to automatically detect changes in a structure is a very attractive feature. When there is no prior knowledge on the system, deep learning models could effectively detect a change and enhance the capability of determining the damage location. However, the acquisition of data related to damaged structures is not always practical. In this paper, two deep learning approaches, a physics-informed autoencoder and a simple data-driven autoencoder, are applied to a test rig consisting of a small four-storey building model. Modifications to the system are simulated by changing the stiffness of the springs. Both the machine learning algorithms outperform the traditional approach based on an experimental modal analysis. Moreover, the increased potential of the physics-informed neural networks to detect and locate damage is confirmed.

Keywords: structural health monitoring; fault detection; neural networks; convolutional autoencoder; physics-informed neural network



Citation: Bono, F.M.; Radicioni, L.; Cinquemani, S.; Bombaci, G. A Comparison of Deep Learning Algorithms for Anomaly Detection in Discrete Mechanical Systems. *Appl. Sci.* **2023**, *13*, 5683. <https://doi.org/10.3390/app13095683>

Academic Editor: Jean-Jacques Sinou

Received: 13 April 2023

Revised: 26 April 2023

Accepted: 2 May 2023

Published: 5 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, there has been a growing interest in the structural health monitoring (SHM) of civil structures such as buildings and bridges [1,2]. Indeed, structures are subjected to many environmental factors that may affect their integrity. As examples, (i) structural cracks can affect the stiffness of the structure; (ii) balancing weight losses can affect its mass; and (iii) wear or looseness in joints can affect the boundary condition of the structure's dynamics and the connections within its different sections [3]. For this reason, appropriate damage detection techniques are required. In this context, SHM includes different monitoring strategies to detect structural damage by analyzing dynamic response measurements, using feature extraction algorithms, and conducting statistical analyses [4]. As a general concept, damage can be defined as changes in the structure which affect its current or future performance. Hence, in order to detect damage, a comparison between two different states of the system is needed, where one of these two states should represent the nominal condition, often corresponding to the undamaged state of the structure. Much introduced the application of the fuzzy approach combined with finite element numerical computations for composite multilayered structures that can be applied in static, dynamic, and fatigue failure problems [5]. In this field, visual inspections are often used to locate damage, but they can be inaccurate, unreliable, and time-consuming [6]. On the contrary, vibration-based techniques have been shown to provide a more reliable way of assessing the health of a structure [7–10]. Indeed, vibrations are the strongest indicator of a structure's state when compared to other indicators [11]. In this context, due to the large amount of data, deep learning is a powerful tool, as it can identify meaningful features in large datasets using multiple processing layers [12]. In general, deep learning models for damage

detection are based on supervised learning strategies, where data from both healthy and damaged conditions of the structure are used as training sets to compute a function able to map new data fed as input. However, the acquisition of the input exciting the structure is often prohibitive, leading to a lack of robustness and a failure to guarantee the convergence of the machine learning technique [13]. In addition, data from a damaged structure are challenging to obtain. For this reason, to overcome these limitations, convolutional autoencoders (CAEs) can be used to detect damage based only on raw vibrational data of the healthy structure [14–16]. Jian et al. [17] showed that a one-dimensional CNN was useful for detecting anomalies in bridge vibration signals. Do et al. [18] demonstrated the capability of autoencoders based on a long short-term memory structure for detecting anomaly vibrations in a Vertical Carousel Storage and Retrieval System for industrial applications. Finotti et al. [19] assessed the structural condition of a viaduct by means of a sparse autoencoder that learned important data features (to characterize the vibration signals) and a support vector machine that classified the corresponding damage based on the extracted features. Jimenez-Martinez et al. improved fatigue life prediction through a combination of synthetic data and an ANN without requiring additional tests or new parameters by overcoming the limit of Miner's damage rule when taking into account different factors such as temperature, environment, sequence loads, and mean stress [20].

Moreover, through the adoption of physics-informed neural networks (PINNs), it is possible to restrict the space of admissible solutions by considering the physical laws that govern the time-dependent dynamics of the structure [13,21,22]. Yucesan et al. [23] introduced a novel approach to modeling the bearing fatigue life of wind turbines by incorporating both physics-based and data-driven components into the model. The authors proposed a recurrent neural network (RNN) that included a physics-informed layer to account for known factors affecting the bearing fatigue and a data-driven layer to describe more complex components, leading to a hybrid model that offered improved accuracy and predictive capabilities for assessing bearing fatigue in real-world applications.

The primary motivation behind PINNs in anomaly detection is to overcome the high cost of acquiring abnormal data in physical systems and the substantial amount of data required for training NNs. With PINNs, the approach combines first principles (physical laws and equations) with neural networks, thereby reducing the search space for network parameters and lowering the need for vast amounts of training data. This compression of the parameter space gives PINNs a significant advantage over traditional NN approaches, making them an attractive option for anomaly detection and location [24].

This paper presents a comparison of the results obtained from an unsupervised deep learning algorithm and a PINN for structural monitoring using only vibrational data acquired in the healthy state as the training set. Both the NNs are tested on a four-storey building using acceleration data from accelerometers placed on each floor. The aim of the paper is to demonstrate the improved capability of PINNs in detecting damage over conventional unsupervised neural networks. The paper is organized as follows: a description of the structure and the mathematical model are presented in Section 2, the experimental campaign is discussed in Section 3, the algorithm is described in Section 4, the results are presented in Section 5, and the conclusion and future research trends are discussed in the Section 6.

2. System Description

The system under study is a multi-storey building shown in Figure 1. Five aluminum plates connected by steel laminas model the storeys and the pillars of the building, respectively, and the physical data of the system are reported in Table 1 [25].

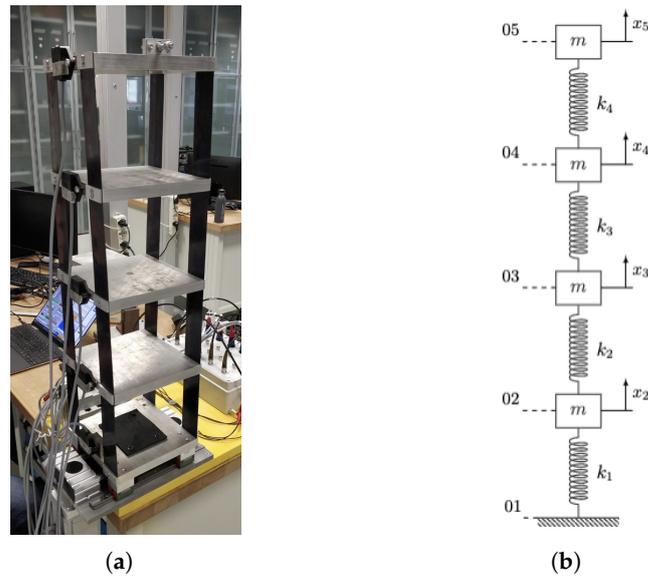


Figure 1. Real system and lumped mass model. (a) A photo of the real system. (b) Lumped mass model of the system.

Table 1. Data of the system.

Storeys	
Area	$200 \times 200 \text{ mm}^2$
Thickness	20 mm
Mass	2.26 kg
Pillars	
Area	$0.5 \times 50 \text{ mm}^2$
Length	180 mm
Thickness	Negligible
Mass	Negligible

Mathematical Model

The mass of each storey is much larger than the mass of the laminas. For this reason, the system can be modeled through a lumped mass approach, considering four degrees of freedom. According to this, the dynamic model consists of a series of four masses connected by springs, as shown in Figure 1b. In the following sections, a physics-informed neural network (PINN) will be introduced. PINNs are characterized by a custom function loss which takes into account information about the physical laws governing the system. For this reason, the equations of motion describing the dynamics of the tested building are derived as reported in Equation (1) and will be a crucial point for the training of the implemented machine learning algorithm.

$$[M] \ddot{\underline{x}} + [C] \dot{\underline{x}} + [K] \underline{x} = \underline{0} \tag{1}$$

In particular, the column vector \underline{x} represents the absolute horizontal displacement of the storeys:

$$\underline{x} = [x_1 \ x_2 \ x_3 \ x_4]^T \tag{2}$$

The mass matrix of the model is diagonal:

$$[M] = \begin{bmatrix} m & 0 & 0 & 0 \\ 0 & m & 0 & 0 \\ 0 & 0 & m & 0 \\ 0 & 0 & 0 & m \end{bmatrix} \quad (3)$$

where m takes into account the mass of the accelerometers and the cables. In particular, this additional contribution is evaluated as 0.1 kg per floor.

Even if in further studies the following assumption may be relaxed considering a displacement and degree of rotation as the boundary conditions, in this paper, a clamped-clamped beam is considered to compute the stiffness matrix [26].

In particular, being one of the extremities subjected to a transversal displacement, the stiffness value of the equivalent spring can be derived through the force method, as shown in Figure 2:

$$k_{eq} = 4 \cdot k = 4 \cdot \frac{12EJ}{L^3} \quad (4)$$

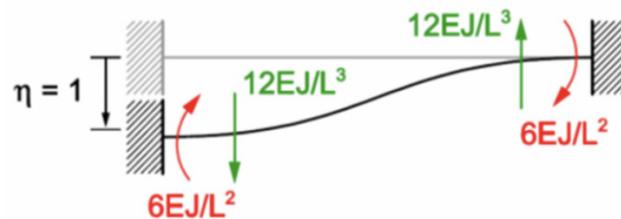


Figure 2. Force method for computing the equivalent stiffness parameter [25].

Moreover, the weight of each storey has an important effect on the transversal stiffness and, for this reason, it must be taken into account [27]. To do this, the term $T = m \cdot g/L$ is considered and the final stiffness matrix is reported as:

$$[K] = \begin{bmatrix} 2k_{eq} - 7T & -k_{eq} + 3T & 0 & 0 \\ -k_{eq} + 3T & 2k_{eq} - 5T & -k_{eq} + 2T & 0 \\ 0 & -k_{eq} + 2T & 2k_{eq} - 3T & -k_{eq} + T \\ 0 & 0 & -k_{eq} + T & k_{eq} - T \end{bmatrix} \quad (5)$$

Regarding damping, Rayleigh’s damping model has been considered. This model allows us to express the damping matrix as a linear combination of the mass matrix and the stiffness matrix [28], as follows:

$$[C] = \alpha \cdot [M] + \beta \cdot [K] \quad (6)$$

The coefficients α and β are computed through a least square minimization process. Indeed, if modal coordinates are adopted, the mass, stiffness, and damping matrices are diagonal matrices, and for each vibration mode i of a given set, it is possible to write:

$$c_i = \alpha \cdot m_i + \beta \cdot k_i \Rightarrow \zeta_i = \frac{c_i}{2m_i\omega_i} = \frac{\alpha}{2\omega_i} + \frac{\beta\omega_i}{2} \quad (7)$$

where ζ_i and ω_i are the modal damping and the natural frequency associated with the vibration mode i , respectively. Moreover, in general, ζ and ω_i are derived by analyzing the

vibrational response of the structure via a modal analysis. The following over-determined system of equations can be written:

$$\begin{bmatrix} \frac{1}{2\omega_1} & \frac{\omega_1}{2} \\ \dots & \dots \\ \frac{1}{2\omega_i} & \frac{\omega_i}{2} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \zeta_1 \\ \dots \\ \zeta_i \end{bmatrix} \tag{8}$$

which is then resolved through a least square minimization process. In this paper, during the experimental campaign, which will be further discussed in the following sections, the modal damping, ζ_i , and the natural frequency, ω_i , are derived for the four vibration modes of the structure, and the resulting coefficients for Rayleigh’s damping model are reported in Table 2.

It is important to underline the fact that the approach described in this paper can be extended to more complex continuum structures such as bridges. In this case, the model can be approximated by estimating and considering the modal parameters of the N first vibration modes with an experimental modal analysis [26] or by developing a finite element model.

Table 2. Results of the least square minimization process.

α	0.03
β	0.028

3. Experimental Campaign

To locate damage, both the conventional method and the machine learning algorithm rely on detection of the changes in the behavior of the structure. In particular, the methods studied in this paper use the differences in the vibration measurements between the building in its nominal configuration, called “healthy”, and the “damaged” configuration as an indicator of damage. For this reason, the experimental campaign conducted on the tested structure aims to acquire raw data for both “healthy” and “damaged” structures. The experimental setup for both the cases consists of:

- Four TE triaxial capacitive MEMS accelerometers, one per storey;
- A PCB piezotronics impact hammer;
- A National Instruments c-DAQ.

The structure is excited by an impact hammer and the transversal vibrations are measured. In particular, as the impact hammer is used manually, it was checked that the peak force for each acquisition ranged between 15 N and 70 N in order to excite the structure with similar impact energies. For the “healthy” case, a set of 1000 records of 70 s each was recorded with a sampling frequency equal to 128 Hz. An example of the responses, limited to the first 30 s, is shown in Figure 3.

As the vibration signals were acquired in Volts, they need to be pre-processed before being analyzed. To do this, the sensitivity of both the accelerometer and the impact hammer must be taken into account and the unit of measurement must be correctly restored (m/s² and N, respectively). Moreover, it must be kept in mind that the measured data are digitized signals and acquired over a finite window of time. For this reason, they may be affected by issues such as (i) noise, (ii) aliasing (as they are sampled), and (iii) leakage, as the signals are acquired for a finite window of time. For these reasons, an averaging procedure was conducted and two different estimators, reported in Equation (9), are considered to evaluate the inertance (acceleration/force) Frequency Response Function (FRF) for each accelerometer.

$$\begin{aligned} H_1(f) &= \frac{X(f)}{F(f)} = \frac{G_{XF}(f)}{G_{FF}(f)} \\ H_2(f) &= \frac{X(f)}{F(f)} = \frac{G_{XX}(f)}{G_{FX}(f)} \end{aligned} \tag{9}$$

where F and X are, respectively, the Fourier transforms of the input force and the output vibration and G_{FF} and G_{XX} are the estimates of the auto spectra, while G_{XF} and G_{FX} are the estimates of the cross spectra, which in general are evaluated as:

$$G_{XX}(f) = \frac{1}{N} \sum_{j=1}^N X_j^*(f) \cdot X_j(f) \tag{10}$$

$$G_{XY}(f) = \frac{1}{N} \sum_{j=1}^N X_j^*(f) \cdot Y_j(f)$$

where X and Y are two general sampled signals and N is their length. It could be interesting to evaluate the cross and power spectra among the different floor accelerations. This is represented in Figure 4, where it is possible to observe the natural frequencies of the system and the ratio between the responses of the different floors corresponding to these frequencies.

Figure 5 shows an example of FRFs for the case in which the structure was excited by an input force acting on the 5th floor.

Once the FRFs were computed for each accelerometer, the natural frequencies and the mode shapes were extrapolated by means of the Experimental Modal Analysis (EMA) [29,30]. In particular, the natural frequencies of the system are reported both for the numerical and the experimental model in Table 3. The mode shapes, normalized with respect to their maximum value, are reported in Figure 6.

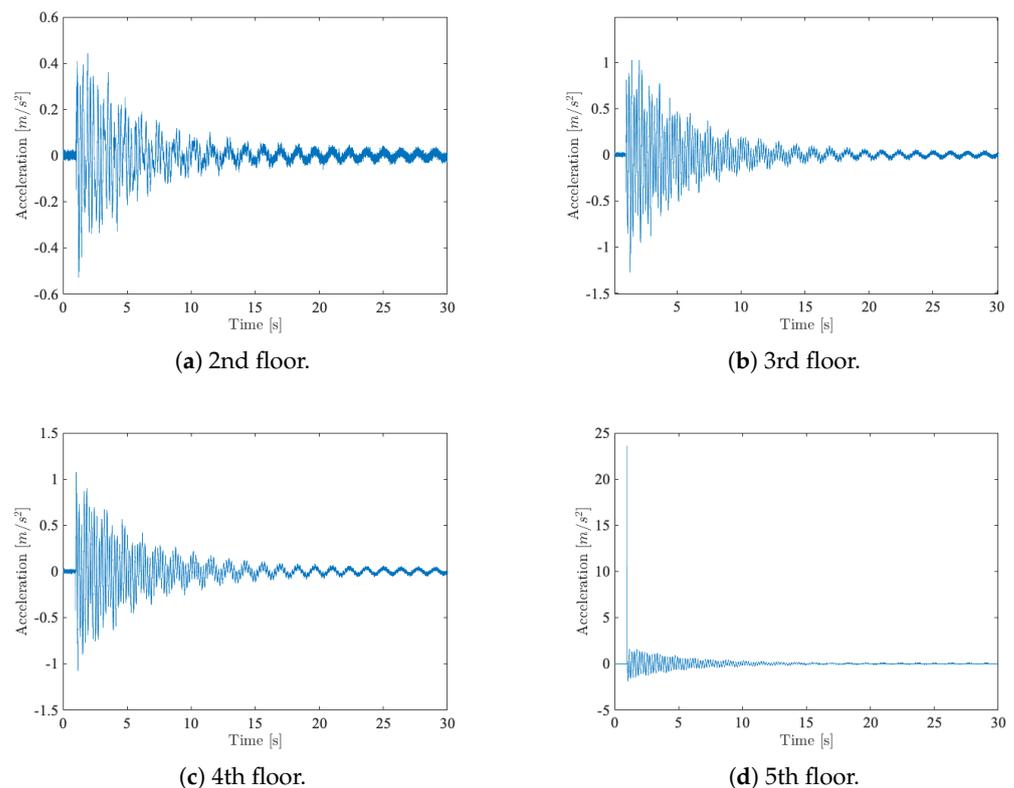


Figure 3. Acceleration of each floor after an input was applied on the 5th floor.

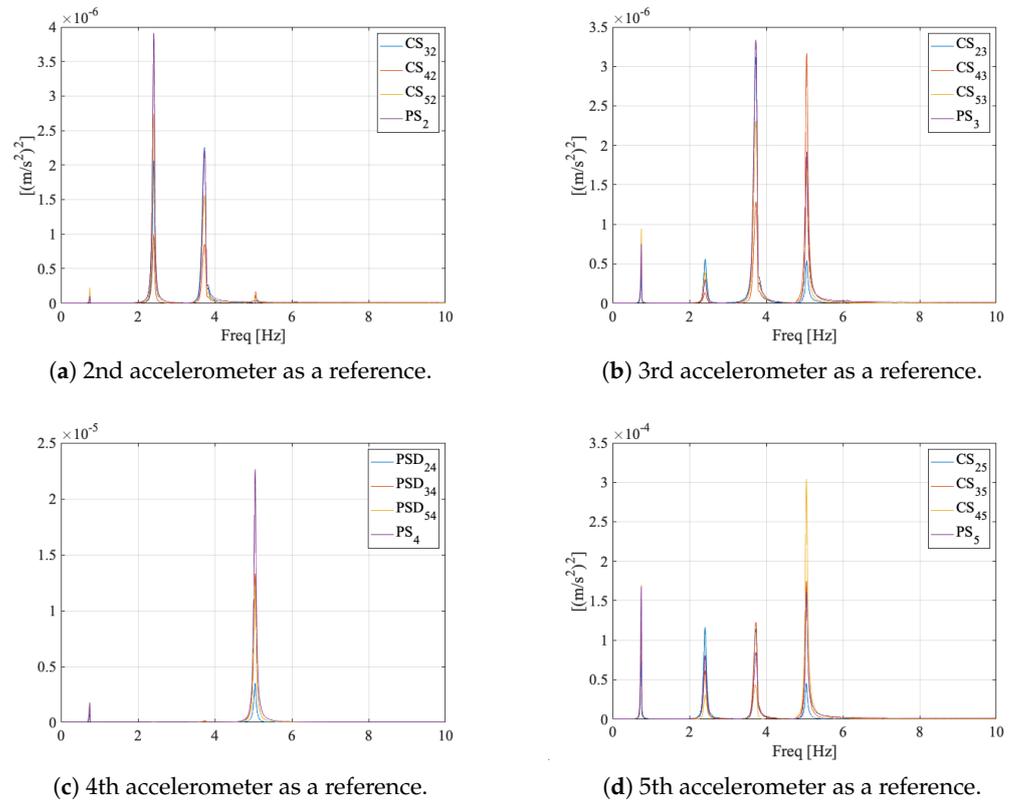


Figure 4. Cross and power spectra for each accelerometer and different references.

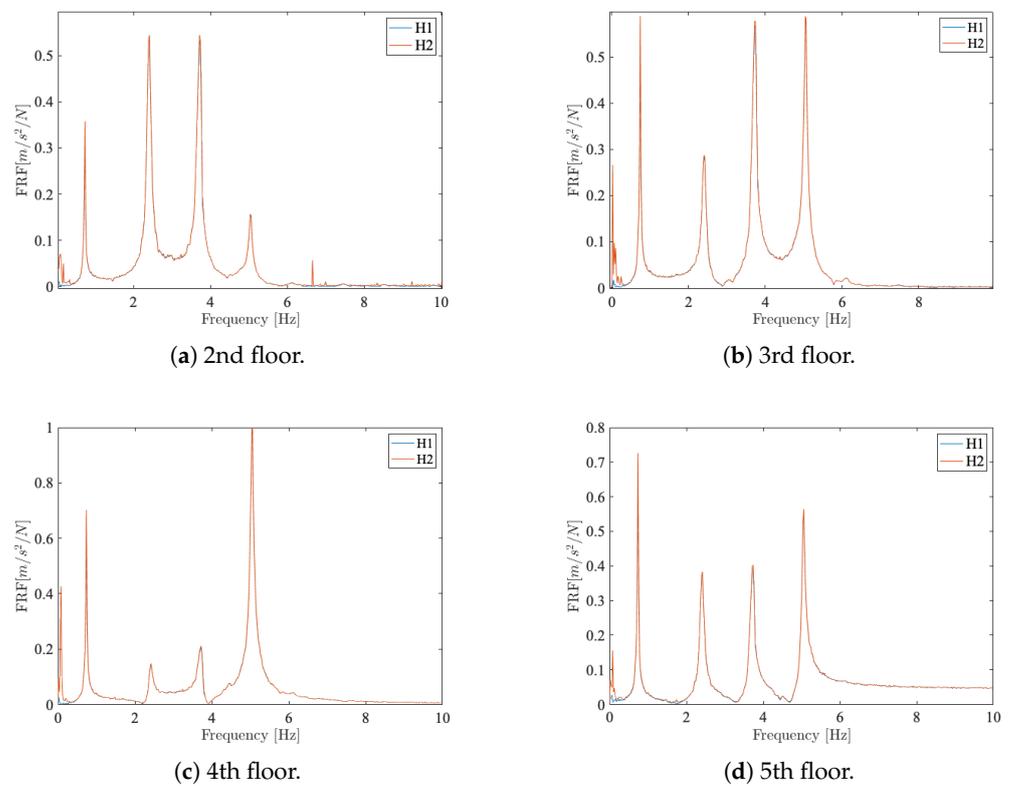
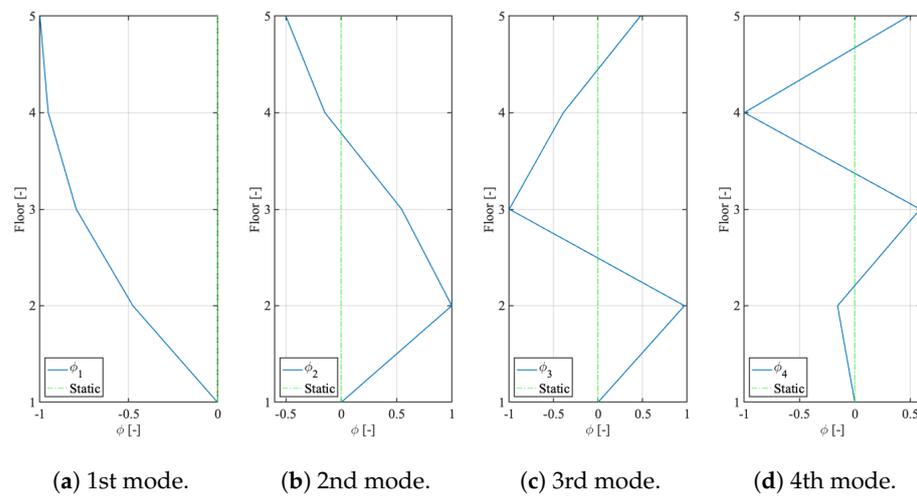


Figure 5. FRFs evaluated for each accelerometer, one for each floor, after an input was applied on the 5th floor.

Table 3. Natural frequencies for both the numerical and the experimental models.

Mode	Numerical Model (Hz)	Experimental Model (Hz)
1	0.79	0.75
2	2.51	2.41
3	3.88	3.74
4	5.01	5.04

**Figure 6.** Vibrational modes of the structure in the healthy scenario.

It is important to highlight that the natural frequencies and the mode shapes derived for the “healthy” structure and previously presented will be used as a reference for the vibration-based damage detection method, which, in turn, will be adopted as a reference to compare the performance of the two machine learning algorithms.

When considering the damaged case, it should be kept in mind that typically internal structural damage is not determined by a loss of material and hence to a related change in mass, but by a change in the geometry or material properties which affects one or many elements in the stiffness matrix [31]. For this reason, the “damaged” time histories are acquired, changing only the stiffness value of the laminas. In particular, six different sets of laminas with same sections but different lengths, as reported in Table 4, were used to decrease the stiffness of the spring connecting two subsequent floors in the range of 10–60% of the nominal value.

In total, 240 time histories of 70 s each were acquired, with a sampling frequency of 128 Hz, resulting in 10 records per combination of type of the damage (type of lamina) and damage location (four floors).

Table 4. Lengths of the set of laminas used to reproduce damage in the structure.

Damage Percentage	Length
0%	180.0 mm
−10%	186.5 mm
−20%	194.0 mm
−30%	203.0 mm
−40%	213.5 mm
−50%	227.0 mm
−60%	244.0 mm

4. Network Architecture and Training

The basic idea behind this work is to compare the ability of detecting damage between a physics-informed neural network and a purely data-driven neural network. Moreover, the results obtained through a conventional vibration-based technique, based on the analy-

sis of the changes in both the natural frequencies and the modes of vibration of the structure, are taken as a reference to understand the advantages in using machine learning algorithms compared to conventional methods.

4.1. Pre-Processing

As previously mentioned, for the healthy structure, 1000 time histories of 70 s were recorded. In particular, the transversal accelerations were measured by the four accelerometers with a sampling frequency of 128 Hz arranged in a four-column matrix. Then, the dataset must be normalized [32]; to do this, the maximum absolute value for each channel was computed and stored in a vector G , that will be useful in the following experiments. In this way, each sample signal will range between -1 and 1 . Finally, the dataset was divided into training, validation, and test subsets composed of 800, 100, and 100 records, respectively.

4.2. Training and Test

The training set was used to train the autoencoder model shown in Figure 7. The key difference between the training phase of the PINN-CAE and the DD-CAE lies in the implementation of a custom function loss in the former, which is capable of taking into account the mathematical model of the system. This custom function loss will be addressed in the following subsections. Regarding the training of the CAE, 200 epochs were considered, with MAE as a loss function. Moreover, a callback setting in the validation loss was adopted. The MAE was then evaluated separately for each accelerometer to evaluate the reconstruction error in the prediction of the test set by the previously trained model. The maxima over all the test sets were taken as thresholds, to be considered for detecting anomalies. Indeed, as already said, the MAE is expected to be greater than these thresholds for time histories representing the damaged structure.

The models were trained on a Nvidia RTX 3080 GPU and the training times were different; the time taken was 320 s and 1031 s for the DD-CAE and the PINN-CAE, respectively.

4.3. Autoencoder

Both the machine learning algorithms, that in the following will be respectively referred as PINN-CAE and DD-CAE, share the same neural network architecture (characterized by a total number of 31,240 trainable parameters), based on a convolutional autoencoder. Autoencoders are unsupervised learning algorithms which, after several transformation and data compression series, aim to reconstruct the input at the output with the least distortion. This technique is widely used to remove noise and for compressing and visualizing high-dimensional data [33]. Convolutional neural networks (CNNs), on which convolutional autoencoders (CAEs) are based, are a class of artificial neural networks (ANNs) which make use of algorithms based on convolution operations and are characterized by many advantages: (i) each neuron is no longer connected to all neurons of the previous layer but to a smaller portion, reducing the parameters and speeding up convergence and (ii) dimension reduction allows the removal of trivial features while retaining useful information [34]. In this paper, the autoencoder was built using the package TensorFlow and assembled with Keras API.

As reported in Figure 7, the model has 11 layers of the following types:

1. **Separable convolutional 1D layer:** This layer applies 1D-convolutional windows separately to every channel. Then, it mixes the channel by point-wise multiplication. Indeed, the application of convolutional layers is proven to be particularly efficient in the analysis of time series [35].
2. **MaxPooling layer:** A limitation of convolutional layers is that they greatly increase the number of parameters in the output tensors compared to the input ones; when many filters are involved, the magnitude of the tensors grows exponentially. For this reason, a pooling layer usually follows a convolutional one. Its purpose is to sub-sample the feature map by retaining only the most attractive information extracted by the convolutional layer. There

are many possible pooling functions, but in this work, the MaxPooling function is adopted, which takes only the max value out of a predefined sub-matrix.

3. **Dropout layer:** A common problem in the development of an ANN is overfitting; this occurs when a model learns from a particular random feature in the training data so that it is able to perfectly manage that set, but these learned concepts may not apply to new data, leading to poor performance. Dropout is a form of regularization, i.e., an approach that makes the network more robust in the training phase by forcing the network to learn general and recurrent patterns. During training, if a tensor passes through a dropout layer, some of its values are randomly dropped according to the dropout probability, i.e., the fraction of input's elements whose value is set to zero. During testing, no values become zero, but the output is scaled by a factor equal to the dropout probability. Sometimes, the values are adjusted by the same fraction only in training to leave the test and prediction phases untouched. Some guidelines to manage the dropout layer can be found in [36].
4. **Dense layer:** This is the simplest and most straight-forward type of layer that can be used. It is used to define the latent space. All the neurons in a dense layer are connected to all the neurons in the previous layer. Every connection is characterized by a weight, which multiplies the input value. The dense layer defines a bias b and an activation function f . If x is the input tensor, z is the output tensor, and W is the weights tensor, the mathematical equation of a dense layer is:

$$z = f(\tilde{z}) = f(Wx + b) \tag{11}$$

where \tilde{z} is called the weighted sum of the input.

5. **Transposed convolutional 1D layer:** This is a type of convolutional layer that can be used to increase the spatial resolution of an input tensor while maintaining a connectivity pattern that is compatible with some convolutional layer. It can be thought of as an operation that takes an input tensor and produces an output tensor with a larger spatial resolution. This operation is also called deconvolution.

The training of the autoencoder was performed taking into account the Mean Absolute Error (MAE) as a loss function. As can be seen in Equation (12), the MAE measures the average magnitude of the absolute differences between the predicted values (outputs of the autoencoder) y_i and the inputs x_i [37].

$$MAE = \frac{\sum_{i=1}^N |y_i - x_i|}{N} = \frac{\sum_{i=1}^N |e_i|}{N} \tag{12}$$

where N is the number of samples of the signals. Moreover, at the end of the training phase, the MAE is also taken as indicator for the detection of anomaly time histories for the tested structure. Indeed, the MAE loss represents the error of reconstruction performed by the autoencoder. Thus, it is likely to assume that the greater the error of reconstruction, the greater the damage [38].

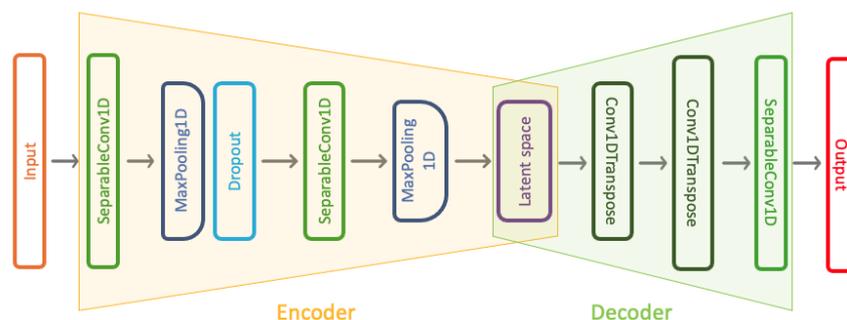


Figure 7. Autoencoder model.

4.4. Physics-Informed Neural Network

The training of deep neural networks requires the availability of large datasets, which may not always be simple to acquire for pre-existing structures and are difficult to retrieve in damage scenarios. Physics-informed neural networks are able to overcome this limitation. Indeed, such networks can be trained with additional information from the physical laws governing the dynamic behavior of the system taken into account, seamlessly integrating data and mathematical models, which may not be even totally understood and may be uncertain and highly dimensional [39,40]. For this reason, the built-in loss function is not useful for training the PINN and a custom loss function is needed. In particular, this loss function should take into account the physical laws governing the dynamic response of the system, restricting the space of admissible solutions in the training of the autoencoder. The scheme of the proposed custom loss is reported in Figure 8.

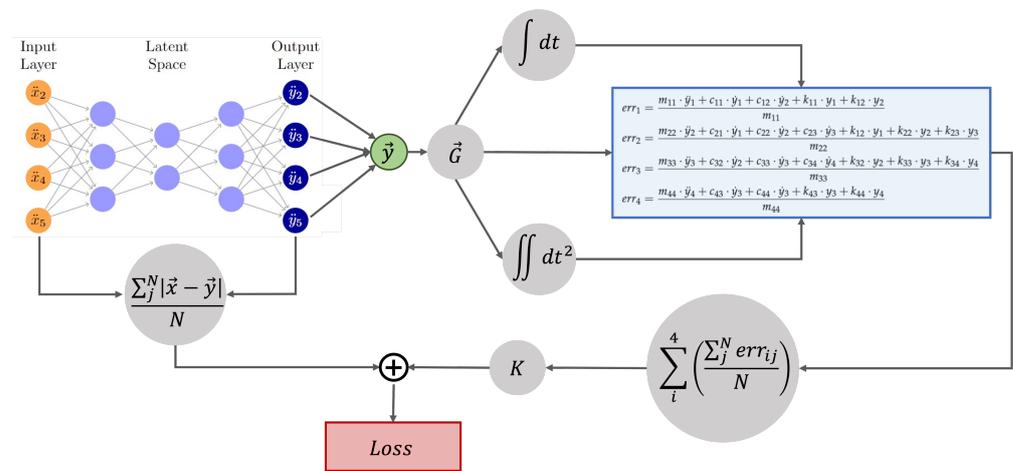


Figure 8. Custom loss scheme.

The signals outputs y_i of the autoencoder are the reconstructed time histories of the scaled vibration signals acquired for the healthy scenario, which in turn represent the response of the system to the input force applied by the hammer. However, after a certain amount of time, set equal to 10 s in this case, it is possible to assume that the transient behavior due to the forced motion of the system completely dies out. Thus, after inverse scaling through the vector G , containing the values of the scaler used during the pre-processing phase, the remaining part of each reconstructed time history should satisfy, with low error, the already presented set of ODEs (1). In particular, in order to be coherent with the measurements units, the ODEs are divided by the mass of the correspondent floor; in this way, the error functions are expressed in acceleration units. This is performed in order to ensure that the physical loss has the same measurements units as that calculated by the data-driven model. Moreover, it is likely that a greater error will result when considering the time histories of the damaged scenario. For this reason, after proper integration in the time domain [41,42], the following error functions were evaluated for each time instant:

$$\begin{aligned}
 err_1 &= \frac{m_{11} \cdot \ddot{y}_1 + c_{11} \cdot \dot{y}_1 + c_{12} \cdot \dot{y}_2 + k_{11} \cdot y_1 + k_{12} \cdot y_2}{m_{11}} \\
 err_2 &= \frac{m_{22} \cdot \ddot{y}_2 + c_{21} \cdot \dot{y}_1 + c_{22} \cdot \dot{y}_2 + c_{23} \cdot \dot{y}_3 + k_{12} \cdot y_1 + k_{22} \cdot y_2 + k_{23} \cdot y_3}{m_{22}} \\
 err_3 &= \frac{m_{33} \cdot \ddot{y}_3 + c_{32} \cdot \dot{y}_2 + c_{33} \cdot \dot{y}_3 + c_{34} \cdot \dot{y}_4 + k_{32} \cdot y_2 + k_{33} \cdot y_3 + k_{34} \cdot y_4}{m_{33}} \\
 err_4 &= \frac{m_{44} \cdot \ddot{y}_4 + c_{43} \cdot \dot{y}_3 + c_{44} \cdot \dot{y}_4 + k_{43} \cdot y_3 + k_{44} \cdot y_4}{m_{44}}
 \end{aligned} \tag{13}$$

where, as already said, y_i , \dot{y}_i , and \ddot{y}_i are the displacement, the velocity, and the acceleration signals coming from the reconstruction of the autoencoder, respectively. Then, the absolute of their mean values is used to constitute the physical portion of the custom function loss:

$$\mathcal{L}^{physic} = \frac{\sum_{i=1}^N |err_{1,i}|}{N} + \frac{\sum_{i=1}^N |err_{2,i}|}{N} + \frac{\sum_{i=1}^N |err_{3,i}|}{N} + \frac{\sum_{i=1}^N |err_{4,i}|}{N} \tag{14}$$

where N is the number of time steps for which the error functions are evaluated.

In order to demonstrate the validity of the information coming from the physical portion of the loss, the error functions, reported in Equation (13), are evaluated for both a “healthy” and a “damaged” time history, in which the defect, identified by a reduction in stiffness of 20% of the nominal value, is located in the third floor. The results, shown in Figure 9, confirm the equation of motion of the system as a valuable option for the physics portion of the custom loss. Indeed, it is clear that for a generic anomaly time history, the error of the equations of motion is larger with respect to the healthy one, especially for the degrees of freedom near to the damage location.

On the other hand, for the data-driven portion of the custom loss, again, the MAE loss function is taken into account.

At the end, the obtained custom loss function is:

$$\mathcal{L} = K \cdot \left[\frac{\sum_{i=1}^n |err_{1,i}|}{n} + \frac{\sum_{i=1}^n |err_{2,i}|}{n} + \frac{\sum_{i=1}^n |err_{3,i}|}{n} + \frac{\sum_{i=1}^n |err_{4,i}|}{n} \right] + MAE \tag{15}$$

where K is a constant to express the physical part of the custom loss in an adimensional form as the data-driven portion of the loss function.

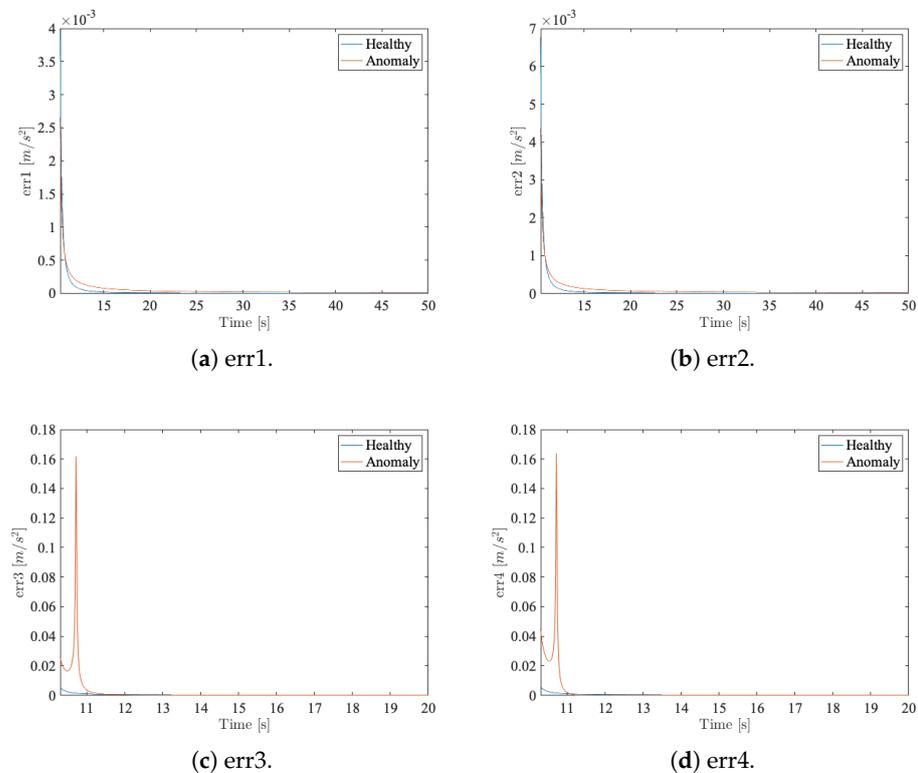


Figure 9. Error functions evaluated for two different scenarios: “healthy” and “damaged”.

5. Results

The 240 damaged records were firstly analyzed via an EMA. However, the natural frequencies and mode shapes were found to slightly change for a damage of 10%, as shown in Figure 10.

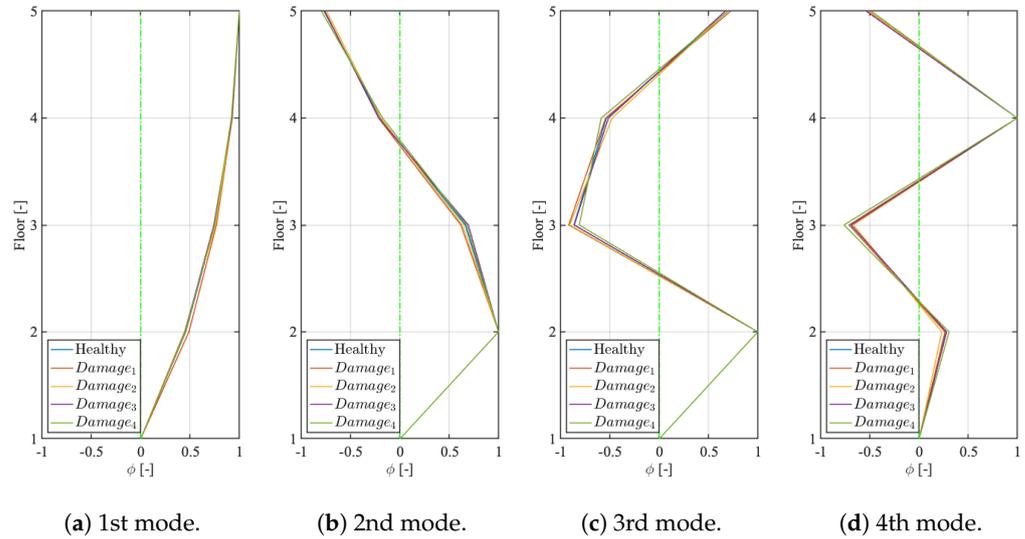


Figure 10. Vibration modes of the structure for a 10% reduction in the stiffness value and for different positions of damage.

For damage of a higher degree, reported in Figure 11, the differences are appreciable, but the detection of the damage position is not straightforward, confirming the main issue described in the literature concerning the fact that vibration signals are a strong indicator of damage, but it is difficult to retrieve its location in this way [43].

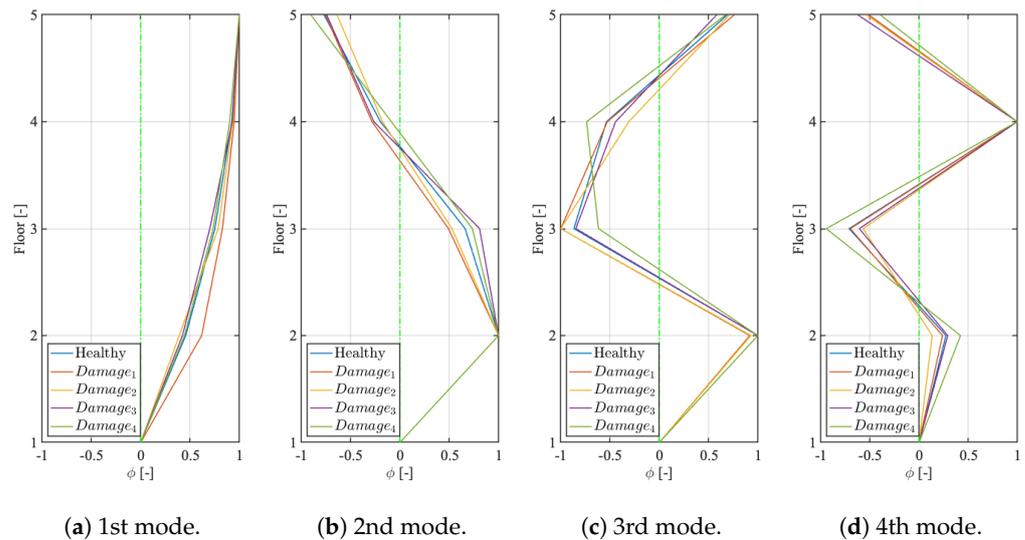


Figure 11. Vibration modes of the structure for a 50% reduction in the stiffness value and for different positions of damage.

With this in mind, the anomalies set were pre-processed with the same scaler obtained for the training set and then fed to both the PINN-CAE and the DD-CAE. The MAE errors for each of the anomalies recorded and for each channel (accelerometer) were then compared to the previously found MAE test thresholds. Each record with a loss higher than

the threshold is classified as an anomaly. Both the architectures detect all the time histories considered in the anomaly set as anomalies, confirming the expected higher precision of data-driven algorithms in detecting structural damage compared to conventional methods. In particular, the difference is extremely significant for the cases in which the extent of damage is not so high. Moreover, for each detected anomaly, the channel (corresponding to the accelerometer position, i.e., the floor) with the maximum value of the MAE loss was selected as the predicted damage position and compared with the real and known one. Then, for each damage extent in the range of 10–60% taken as a reference, the following accuracy indicator was evaluated for both the considered algorithms:

$$\mathcal{A} = \frac{n_d}{n_{tot}} \times 100 \quad (16)$$

where n_d is the number of anomalies with damage extent equal or higher than the reference value and whose position is correctly detected by the model, while n_{tot} is the total number of anomalies with a damage extent equal to or higher than the reference value. The results are reported in Table 5.

Table 5. Anomaly detection rates as a function of the damage percentage for both PINN-CAE and DD-CAE.

Damage Percentage	Accuracy \mathcal{A}	
	DD-CAE	PINN-CAE
–10%	33.19%	79.43%
–20%	40.20%	82.81%
–30%	52.24%	87.22%
–40%	65.11%	92.03%
–50%	84.61%	100%
–60%	100%	100%

It is possible to conclude that the PINN outperforms the results obtained with the purely data-driven approach, as expected.

6. Conclusions

The accuracy in detecting structural damage is investigated for two different machine learning algorithms: a physics-informed convolutional autoencoder (PINN-CAE) and a purely data-driven convolutional autoencoder (DD-CAE). In particular, raw data from experiments on a four-storey building were fed to the autoencoder, whose structure was the same in both strategies. The MAE error of reconstruction is taken as an indicator to detect anomalous records. Both the PINN-CAE and the DD-CAE outperformed conventional vibration-based methods in the capability of detecting damage and finding its location. Indeed, they are able to detect all the anomalous time histories, showing good precision in the detection of structural change locations. The physics-informed network was more precise in the detection of the damage location compared to the data-driven one, with a significant increase in the accuracy for a lower damage extent. However, as shown in Table 5, it is possible to observe that the higher the damage percentage, the higher the possibility of correctly locating the damage. With the hybrid approach, it is possible to continuously monitor the structure and to identify accumulated damage, i.e., damage that increases over time, in advance and with a better accuracy with respect to traditional NNs. This confirms the high potential of combining a data-driven architecture and information about the physical characteristics of the system under study. Future developments of this work will include changes in the mass of the system and the use of the model together with NNs for detecting anomalies in more complex structures, such as bridges or viaducts. In particular, when dealing with these types of structures, an analytical representation of the system is difficult to construct, but a numerical model based on simulation methods

(such as finite element methods) can be adopted. The extension of this work by considering a numerical model of a more complex system will be developed in the future by the authors.

Author Contributions: Conceptualization, F.M.B., L.R., S.C. and G.B.; methodology, F.M.B. and L.R.; software, F.M.B. and G.B.; validation, F.M.B., L.R. and G.B.; formal analysis, L.R.; investigation, F.M.B. and G.B.; resources, F.M.B. and L.R.; data curation, G.B.; writing—original draft preparation, F.M.B., L.R. and G.B.; writing—review and editing, F.M.B., L.R., G.B. and S.C.; visualization, F.M.B. and G.B.; supervision, S.C.; project administration, S.C.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are unavailable due to privacy.

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

Abbreviations

The following abbreviations are used in this manuscript:

SHM	Structural health monitoring
RNN	Recurrent neural network
NN	Neural network
PINN	Physics-informed neural network
CAE	Convolutional autoencoder
MAE	Mean absolute error
EMA	Experimental modal analysis
FRF	Frequency response function
PINN-CAE	Physics-informed convolutional autoencoder
DD-CAE	Data-driven convolutional autoencoder

References

1. Song, G.; Wang, C.; Wang, B. Structural health monitoring (SHM) of civil structures. *Appl. Sci.* **2017**, *7*, 789. [\[CrossRef\]](#)
2. Li, H.N.; Ren, L.; Jia, Z.G.; Yi, T.H.; Li, D.S. State-of-the-art in structural health monitoring of large and complex civil infrastructures. *J. Civ. Struct. Health Monit.* **2016**, *6*, 3–16. [\[CrossRef\]](#)
3. Montalvao, D.; Maia, N.M.M.; Ribeiro, A.M.R. A review of vibration-based structural health monitoring with special emphasis on composite materials. *Shock Vib. Dig.* **2006**, *38*, 295–324. [\[CrossRef\]](#)
4. Farrar, C.R.; Worden, K. *Structural Health Monitoring: A Machine Learning Perspective*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
5. Muc, A. Fuzzy approach in modeling static and fatigue strength of composite materials and structures. *Neurocomputing* **2020**, *393*, 156–164. [\[CrossRef\]](#)
6. Wu, X.; Ghaboussi, J.; Garrett, J., Jr. Use of neural networks in detection of structural damage. *Comput. Struct.* **1992**, *42*, 649–659. [\[CrossRef\]](#)
7. Salawu, O. Detection of structural damage through changes in frequency: A review. *Eng. Struct.* **1997**, *19*, 718–723. [\[CrossRef\]](#)
8. Goyal, D.; Pabla, B. The vibration monitoring methods and signal processing techniques for structural health monitoring: A review. *Arch. Comput. Methods Eng.* **2016**, *23*, 585–594. [\[CrossRef\]](#)
9. Fritzen, C.P. Vibration-based structural health monitoring—concepts and applications. *Key Eng. Mater.* **2005**, *293*, 3–20. [\[CrossRef\]](#)
10. Magalhães, F.; Cunha, Á.; Caetano, E. Vibration based structural health monitoring of an arch bridge: From automated OMA to damage detection. *Mech. Syst. Signal Process.* **2012**, *28*, 212–228. [\[CrossRef\]](#)
11. Deraemaeker, A.; Reynders, E.; De Roeck, G.; Kullaa, J. Vibration-based structural health monitoring using output-only measurements under changing environment. *Mech. Syst. Signal Process.* **2008**, *22*, 34–56. [\[CrossRef\]](#)
12. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#)
13. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **2019**, *378*, 686–707. [\[CrossRef\]](#)
14. Rastin, Z.; Ghodrati Amiri, G.; Darvishan, E. Unsupervised structural damage detection technique based on a deep convolutional autoencoder. *Shock Vib.* **2021**, *2021*, 6658575. [\[CrossRef\]](#)
15. Rosafalco, L.; Manzoni, A.; Corigliano, A.; Mariani, S. A time series autoencoder for load identification via dimensionality reduction of sensor recordings. *Eng. Proc.* **2021**, *2*, 34.

16. Ma, X.; Lin, Y.; Nie, Z.; Ma, H. Structural damage identification based on unsupervised feature-extraction via Variational Auto-encoder. *Measurement* **2020**, *160*, 107811. [CrossRef]
17. Jian, X.; Zhong, H.; Xia, Y.; Sun, L. Faulty data detection and classification for bridge structural health monitoring via statistical and deep-learning approach. *Struct. Control Health Monit.* **2021**, *28*, e2824. [CrossRef]
18. Do, J.S.; Kareem, A.B.; Hur, J.W. LSTM-Autoencoder for Vibration Anomaly Detection in Vertical Carousel Storage and Retrieval System (VCSRS). *Sensors* **2023**, *23*, 1009. [CrossRef]
19. 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]
20. Jimenez Martinez, M.; Alfaro Ponce, M. Effects of synthetic data applied to artificial neural networks for fatigue life prediction in nodular cast iron. *J. Braz. Soc. Mech. Sci. Eng.* **2021**, *43*, 10. [CrossRef]
21. Raissi, M.; Wang, Z.; Triantafyllou, M.S.; Karniadakis, G.E. Deep learning of vortex-induced vibrations. *J. Fluid Mech.* **2019**, *861*, 119–137. [CrossRef]
22. Cai, S.; Mao, Z.; Wang, Z.; Yin, M.; Karniadakis, G.E. Physics-informed neural networks (PINNs) for fluid mechanics: A review. *Acta Mech. Sin.* **2021**, *37*, 1727–1738. [CrossRef]
23. Yucesan, Y.; Viana, F. A Physics-informed Neural Network for Wind Turbine Main Bearing Fatigue. *Int. J. Progn. Health Manag.* **2020**, *11*, 17.
24. Huang, B.; Wang, J. Applications of Physics-Informed Neural Networks in Power Systems—A Review. *IEEE Trans. Power Syst.* **2023**, *38*, 572–588. [CrossRef]
25. Di Carlo, S.; Benedetti, L.; Di Gialleonardo, E. Teaching by active learning: A laboratory experience on fundamentals of vibrations. *Int. J. Mech. Eng. Educ.* **2022**, *50*, 869–882. [CrossRef]
26. Cheli, F.; Diana, G. *Advanced Dynamics of Mechanical Systems*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 2020.
27. Ghali, A.; Neville, A.M.; Brown, T.G. *Structural Analysis*; CRC Press: Boca Raton, FL, USA, 2017. [CrossRef]
28. Liu, M.; Gorman, D.G. Formulation of Rayleigh damping and its extensions. *Comput. Struct.* **1995**, *57*, 277–285. [CrossRef]
29. Schwarz, B.J.; Richardson, M.H. Experimental modal analysis. *CSI Reliab. Week* **1999**, *35*, 1–12.
30. Cunha, Á.; Caetano, E. Experimental modal analysis of civil engineering structures. *Sound Vib.* **2006**, *40*.
31. Hajela, P.; Soeiro, F. Structural damage detection based on static and modal analysis. *AIAA J.* **1990**, *28*, 1110–1115. [CrossRef]
32. Bono, F.; Radicioni, L.; Cinquemani, S. A novel approach for quality control of automated production lines working under highly inconsistent conditions. *Eng. Appl. Artif. Intell.* **2023**, *122*, 106149. [CrossRef]
33. Zhang, Y. A better autoencoder for image: Convolutional autoencoder. In Proceedings of the ICONIP17-DCEC, 2018, Guangzhou, China, 14–18 October 2017 Available online: http://users.cecs.anu.edu.au/Tom.Gedeon/conf/ABCs2018/paper/ABCs2018_paper_58.pdf (accessed on 23 March 2017).
34. 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]
35. Zhang, C.; Song, D.; Chen, Y.; Feng, X.; Lumezanu, C.; Cheng, W.; Ni, J.; Zong, B.; Chen, H.; Chawla, N.V. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 1409–1416.
36. Srivastava, N. Improving neural networks with dropout. *Univ. Tor.* **2013**, *182*, 7.
37. 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]
38. Bono, F.M.; Radicioni, L.; Cinquemani, S.; Benedetti, L.; Cazzulani, G.; Somaschini, C.; Belloli, M. A Deep Learning Approach to Detect Failures in Bridges Based on the Coherence of Signals. *Future Internet* **2023**, *15*, 119. [CrossRef]
39. Karniadakis, G.E.; Kevrekidis, I.G.; Lu, L.; Perdikaris, P.; Wang, S.; Yang, L. Physics-informed machine learning. *Nat. Rev. Phys.* **2021**, *3*, 422–440. [CrossRef]
40. Cuomo, S.; Di Cola, V.S.; Giampaolo, F.; Rozza, G.; Raissi, M.; Piccialli, F. Scientific machine learning through physics-informed neural networks: Where we are and what's next. *J. Sci. Comput.* **2022**, *92*, 88. [CrossRef]
41. Davis, P.J.; Rabinowitz, P. *Methods of Numerical Integration*; Courier Corporation: North Chelmsford, MA, USA, 2007.
42. Brandt, A.; Brincker, R. Integrating time signals in frequency domain—Comparison with time domain integration. *Measurement* **2014**, *58*, 511–519. [CrossRef]
43. Doebling, S.W.; Farrar, C.R.; Prime, M.B.; Shevitz, D.W. *Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibration Characteristics: A Literature Review*; U.S. Department of Energy Office of Scientific and Technical Information: Washington, WA, USA, 1996.

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.