Journal of
Instrumentation



# Damage diagnostic method by artificial intelligence analysis of shaking table data of a typical Italian building prototype

To cite this article: D. Palumbo *et al* 2025 *JINST* **20** C06063

Published by IOP Publishing for Sissa Medialab

Received: *February 27, 2025*
Accepted: *May 5, 2025*
Published: *June 26, 2025*

7th International Conference Frontiers in Diagnostic Technologies
INFN research center of Frascati, Italy
21–23 October 2024

# Damage diagnostic method by artificial intelligence analysis of shaking table data of a typical Italian building prototype

**D. Palumbo**,[a] **C. Ormando**,[a] **A. Colucci**,[a] **S. De Santis**,[b] **G. de Felice**,[b] **D. Liberatore**[c] **and I. Roselli**[a,*]

[a] *Casaccia Research Center, ENEA,*
*Via Anguillarese 301, 00123 S. Maria di Galeria (Rome), Italy*
[b] *Department of Civil Engineering, Computer Science and Aeronautical Technologies, Roma Tre University,*
*Via Vito Volterra 62, 00146 Rome, Italy*
[c] *Department of Structural and Geotechnical Engineering, Sapienza University of Rome,*
*Via Eudossiana 18, 00184 Rome, Italy*

*E-mail:* ivan.roselli@enea.it

Abstract: A damage diagnostic method based on the use of artificial intelligence (AI) was pointed out for the analysis of displacement data recorded in shaking table tests of a prototype representing a typical Central Italy historic masonry building. The displacements were measured by the use of a 3D motion capture system capable of tracking the motion of passive optical markers located at several positions on the tested building. The state of damage was initially estimated by calculating a widely accepted Damage Index (DI) based on the first mode frequency decay of the building calculated by conventional Frequency Response Function (FRF) for modal analysis of the markers displacements data. Then, machine learning was used to develop a method to estimate the damage status of the tested prototype. More specifically, the proposed AI procedure exploits the Convolutional Variational Auto-Encoder (CVAE), which is an important generative model in unsupervised deep learning. CVAE was applied to the recorded displacement data to reconstruct the original data. The hidden features in the encoder latent spaces of the CVAE were used as a basis to assess the structural damage and compared to real values of DI. The proposed method by machine learning showed very promising potentialities of assessing the structural damage in typical historic masonry buildings.

Keywords: Data processing methods; Detection of defects; Artificial intelligence; Deep learning; Historic buildings; Damage assessment

*Corresponding author.



https://doi.org/10.1088/1748-0221/20/06/C06063

2025 JINST 20 C06063

## Contents



## 1 Introduction

The damage assessment of structures after earthquake events is a very relevant issue. Consequently, experimental studies of representative structural prototypes subjected to seismic test on shaking table is of essential relevance to understand the structural behavior and the vulnerability of real structures [1]. In shaking table tests the damage is commonly assessed by analyzing the data recorded by several measurement systems (e.g. accelerometers, displacement sensors, optical markers etc.). In particular, data acquired during dynamic identification tests are generally processed by consolidated algorithms with the purpose of extracting parameters that can be related to the state of damage [2]. The recent advances in data processing through Artificial Intelligence, specifically machine learning methods like deep learning, permit to explore the possibility to guess the damage assessment given by conventional methods. Deep learning, widely used in many fields, is also used for monitoring the integrity of infrastructures [3–5]. Most recent applications use convolutional variational autoencoders (CVAE) and one-class support vector machine to identify and classify damage scenarios [6, 7]. This work explores the potentiality of the innovative use of features, such as the features constructed in the latent space in CVAE without any knowledge of the system model, on a set, even a limited one, of examples to learn to reconstruct the original data while the hidden features, in the encoder latent space, could be used as a basis to predict the damage indices.

## 2 Methods and data pre-processing

This article proposes a method based on machine learning to reconstruct the data of a structure based on the displacement data by optical markers of a 3D motion capture system. The optical markers were located at most critical measurement positions of a prototype representing a typical Central Italy historic rubble masonry building. The markers data were analyzed to provide an estimate of the modal parameters of the structure. In particular, a consolidated EMA method based on conventional Frequency Response Function (FRF) was implemented. It is based on well-established H transmissibility algorithms in multi-input multi-output (MIMO) approach, where markers signals positioned at the prototype base were used as input signals. Then a damage index (DI) based on



2025 JINST 20 C06063

the first mode frequency decay of the building was calculated by conventional Frequency Response Function (FRF) for modal analysis of the markers displacements data.

The overall machine learning method consists of several processing steps (see figure 1). However, the main step of the proposed method comprises the use of CVAE as a feature extractor of hidden features in the encoder latent space.

Initially, the data is processed to build the necessary features by selecting a certain number of measurement points based on their data standard deviation. The ones chosen were those with a standard deviation between 0.2 and 0.3. This step allows avoiding corrupted signals and selecting good quality measurements. In the present case, 18 time history signals were considered.

Also, a temporal slice was then performed on the data, without overlap, with the constraint that the number of points is a power of 2. This choice was made because neural networks with sizes in powers of 2, as well as, similarly, batch sizes in powers of 2 can improve computational efficiency during training, by better exploiting the GPU architecture or more modern GPUs using Tensor Cores [8].

Data normalization is a fundamental and necessary step in machine learning algorithms and in this work we normalize the raw data time series using a standard scaler and normalizing each sensor. We trained the scaler on the training data and then normalized training, test and validation. The scaler standardizes features by removing the mean and scaling to unit variance. The standard score of a sample $x$ is calculated as follows:

$$z = (x - u)/s \tag{2.1}$$

where $u$ is the mean of the training samples and $s$ is the standard deviation of the training samples.

### 2.1 Auto encoder

An auto encoder is a machine learning model, specifically a neural network, that is trained by unsupervised learning to reproduce its own input [9, 10].

The simplest example of an autoencoder is composed of two parts, input and output. It is usually designed to compress data.

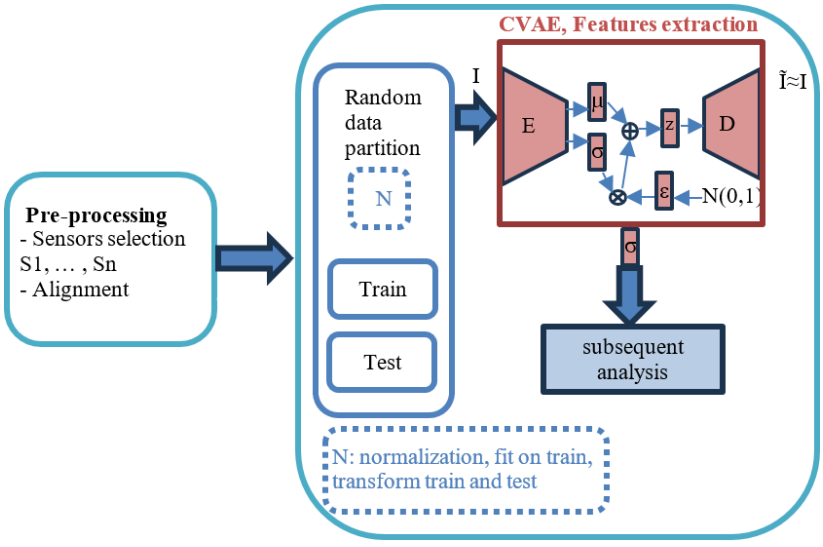


**Figure 1.** Workflow of the proposed AI procedure.

An autoencoder works by first projecting the input data into an internal or feature space z through the encoder. The feature space is the so-called latent encoding space. The decoder function takes the latent representation of the data and projects the data back to the input space. However, a simple single-layer autoencoder is not sufficient to extract representative features from raw data. There is a need for a deep autoencoder [3, 6, 11].

VAEs add an additional constraint by forcing the latent representation to follow a specific distribution, such as Gaussian, with the result that the latent variable z become a probabilistic latent space with stable statistical properties. The decoder inputs are then sampled from the distribution of the given latent variable z with the encoder constructed to output the mean $\mu$ and the variance $\sigma$ of the latent variable z. The VAE encoder thus maps a single input point, say an image, into a distribution within the latent space instead of to a single point.

However, VAEs are not always able to reconstruct the input signals, tending instead to the mean value. So we will use Convolutional VAEs (CVAEs) for their better capabilities with respect to implementing the encoder/decoder using fully-connected layers and for their ability to capture the spatial and temporal relationships present in the data [6].

As for the loss function of the CVAE, it is composed of two parts, one part that has the objective of minimizing the difference between the input and output of the CVAE and the other to verify how much the latent space distribution deviates from a specific distribution. To capture the difference between the input data and the reconstructed data in this paper we used the Mean Squared Error (MSE), although other functions are possible (e.g. the binary cross-entropy function) is calculated as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{I}^{\mathrm{i}} - \bar{\mathrm{I}}^{i})^2 \tag{2.2}$$

where I and $\bar{\mathrm{I}}$ are the input and reconstructed input of the CVAE and n is the dimensionality of the data. While to calculate how much the latent variable z approaches a specific distribution, in this case a standard normal distribution, the Kullback-Leibler (KL) divergence $\mathrm{D_{KL}}$ is used, which measures the divergence between two distributions and acts as a regularizer term [10, 12]:

$$\mathrm{D}_K = \int p\,(x) \log\left(\frac{p(x)}{q(x)}\right) dx \tag{2.3}$$

where $p(x)$ and $q(x)$ are two distributions.

So the loss function $\mathrm{L_{CVAE}}$ will be dependent on MSE and $\mathrm{D_{KL}}$ and defined as follows [13]:

$$\mathrm{L_{CVAE}} = \mathrm{k} \times \mathrm{MSE} + \mathrm{E}(\mathrm{D_{KL}}) \tag{2.4}$$

where k is a scaling factor and E the expected value.

### 2.2 Training parameters

The CVAE training is performed by partitioning the data between train set and test set in a ratio of 70%-30%, respectively. The train set is used to train the CVAE to reproduce its inputs using an unsupervised procedure. The normalization is performed by fitting on the training set and transforming the training and test set. To avoid dependency on the performed partition, the procedure was repeated 20 times. Two typical approaches are used to assess the performance of machine learning methods: a hold-out method and a cross-validation method. Hold-out is suitable for large dataset, when the number



2025 JINST 20 C06063

of samples is more than a few hundreds. On the contrary, when samples are less than a few hundreds the cross-validation method performs better [12, 14, 15]. In our case, we have more than a thousand examples for the training set, so we used the hold-out method shuffling the data at each repetition.

During the CVAE training a part of the data, 20% of the training set is used as a validation set to check the validity of the learning on 1,800 epochs, batch size 128 and with early stopping with 100 epochs of patience to stop the training in case there is no reduction of the validation loss value. Adam was the optimizer used during the training process.

The data dimensionality, the encoder and the decoder network architecture of the CVAE are summarized in table 1.

**Table 1.** Data dimensionality.

| Data set | Shape |
|---|---|
| Total samples | (2105) |
| Train | (1178, 128,18) |
| Validation | (295, 128,18) |
| Test | (632, 128,18) |

The encoder network architecture, after the input with a shape of (128, 18), is based on a double sequence of a block composed of a convolutional Conv1d layer plus a Batch_normalization layer followed by a Max_pooling1d layer at the end. Then, we used a quadruple sequence of a Conv1d layers plus a Batch_normalization layer. In Conv1d we used a different number of filters for each layer, ranging from 16 to 32, with kernel size equal to 2, and the Relu as activation function.

Successively, we implemented a Flatten layer with an output shape of (1024), which is the input of the resampling process performed by mapping the data into two dense layers of size 64, the mean and variance of the latent space.

In the decoder network architecture, the latent representation is the input of a dense layer followed by the Batch_normalization and Reshape layers. Sequences of Conv1d and Batch_normalization layers mirror the encoder block, followed by an Up_sampling1d layer to expand the data.

Finally, the Conv1d output layer of the decoder is implemented with a number of filters equal to the number of sensors, eventually providing an output with shape of (128, 18).

## 3 Experimental validation

The experimental validation of the developed procedure was carried out by analyzing the displacement data by optical markers of a 3D motion capture (3DVision) system [16]. The markers data were initially analyzed to primarily provide the first modal frequency of a prototype building subjected to shaking table tests at ENEA Casaccia Research Center in Rome. More in details, dynamic identification shaking table tests by white noise vibration (WHN) at 0.05 g were analyzed before and after seismic tests at gradually increasing intensity from 0.05 g up to the specimen failure occurred at 0.50 g of Peak Table Acceleration (PTA). The specimen was a typical Italian building prototype [10] made up of historic rubble masonry with poor limestone mortar and a light wooden roof loaded with additional mass of 18 kN (figure 2a).

The 3DVision cameras tracked the displacement of markers with accuracy of 0.03 mm at 200 Hz. The displacement time histories recorded by each marker were processed and analysed to extract the modal frequencies of the specimen. An elliptic low-pass filter with a cut-off frequency of 12 Hz

was then applied in order to clean the recordings from vibration noise at frequencies higher than the range of interest.

The damage index DI based on first modal frequency decay (figure 2b) was computed. DI value is equal to 0 for the undamaged structure and grows with the damage of the structure up to the ultimate collapse at DI equal to 1.

Contemporarily, the same markers data were processed by machine learning CVAE in the attempt to reconstruct to input data for all the different DI values computed by EMA technique.

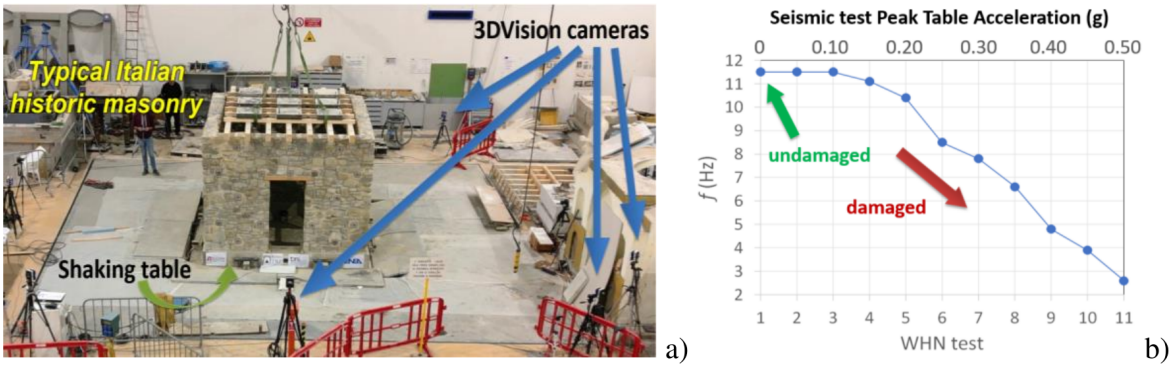


**Figure 2.** Shaking table test setup (a). First modal frequency $f$ extracted by conventional EMA analysis of the shaking table identification (WHN) tests (b).

## 4 Results

The application of the proposed AI-procedure provided a reproduction of the training and the test sets with a RMSE of 0.42 and 0.45 over the performed 20 repetitions, respectively. The relatively small errors obtained indicate that the latent space is a quite good representative of the original input data time histories.

In figure 3 the first 0.64 s of the AI-reconstructed time history of one marker can be compared with the original input data in two cases: test and train cases with values of damage index DI equal to 0.675 and 0.883, respectively.

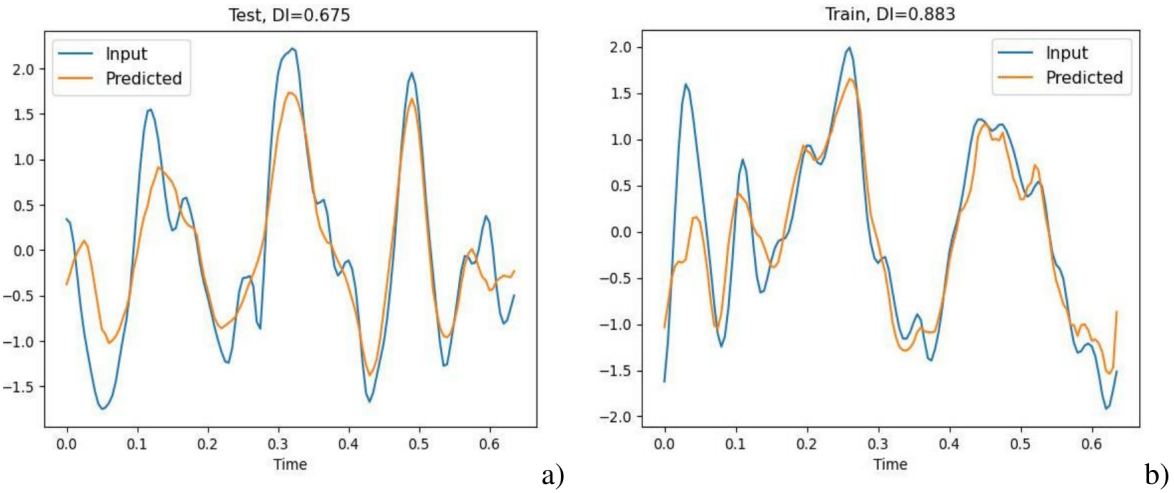


**Figure 3.** Example of AI reconstruction (predicted) of the recorded vibration time history (input) for test set at damage index DI = 0.675 (a) and for train set at DI = 0.883 (b).



2025 JINST 20 C06063

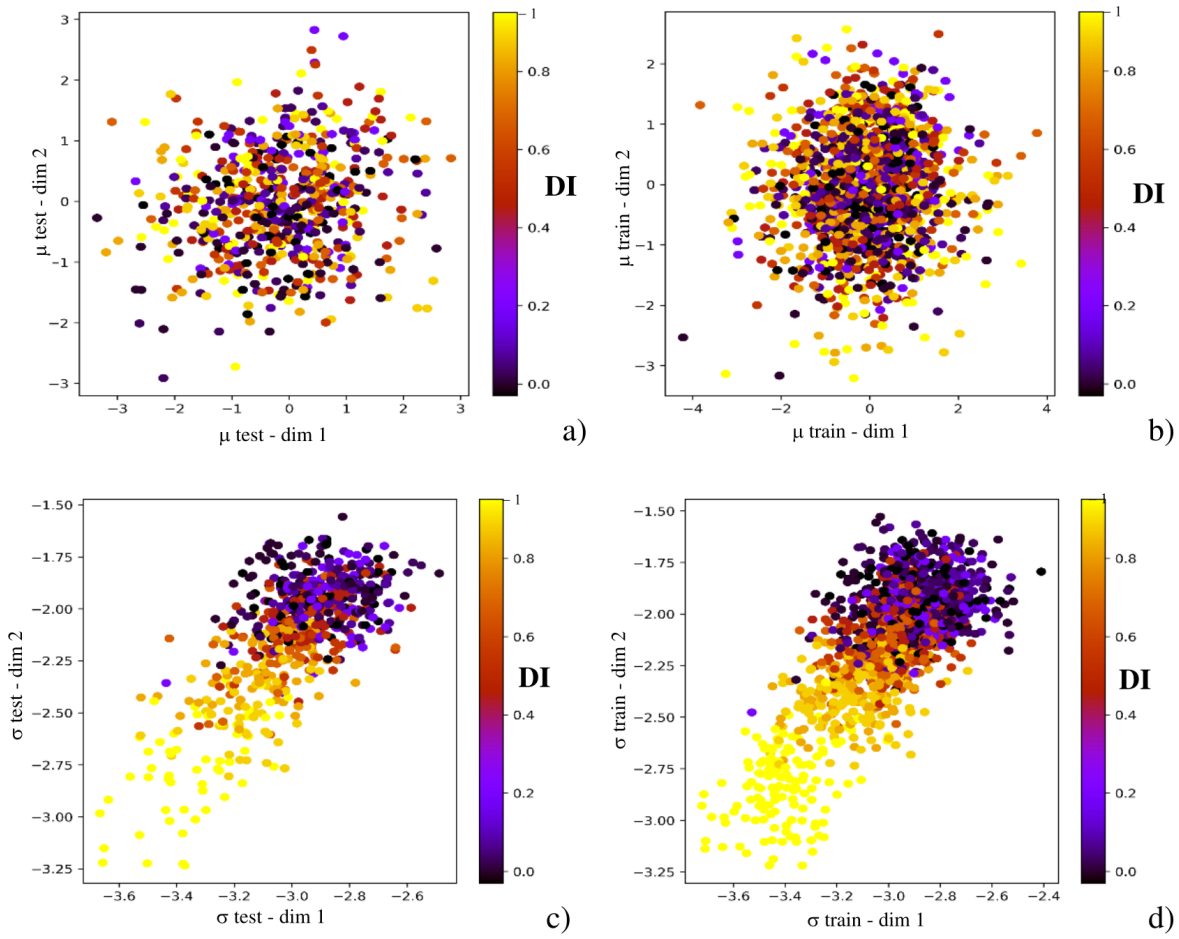


**Figure 4.** Latent spaces of 2 dimensions of mean value μ for test (a) and train (b) sets, and of variance σ for test (c) and train (d) sets.

Figure 4 shows an example of latent spaces obtained for 2 dimensions considering the mean μ and the variance σ in test and train sets. In the above example, the points representing different values of DI in the latent spaces of the mean μ (figure 4a-b) are scattered with little discriminatory or predictive capability. While the points in the latent spaces of the variance σ (figure 4c-d) show notably different positions depending on DI values, which demonstrate a high discriminatory or predictive potentiality of this parameter.

Using a simple desktop system with a CPU with base speed of 4 GHz (with 1 physical processor, 4 cores, 8 logical processors) and 32.0 GB of RAM, the training phase takes 3 seconds for 1 epoch, 90 minutes to process all samples. However, the prediction phase on one sample takes about 15 milliseconds, allowing the use of this procedure for real-time monitoring of structures

## 5 Conclusions

A method based on AI machine-learning procedure was proposed to reconstruct the vibration data of a structure specimen representative of a typical Central Italy historic masonry building.

Furthermore, the present work also explored the potentiality of an innovative use of features, such as the features constructed in the latent space in CVAE without any knowledge of the system model,

on a set of given examples to learn to reconstruct the original data, which in our case was represented by the recorded displacement time histories. While the latent features in the encoder latent space were utilized as a basis to estimate the structural damage in term of DI index values.

The experimental validation of the developed procedure was carried out by analyzing the displacement data by optical markers located at several positions of the building specimen.

The proposed AI procedure demonstrated that the use of CVAE is very effective in reconstructing the input vibration data time histories. In particular, small errors were achieved in the reproduction of the training and the test datasets.

Moreover, the latent space representation of the variance $\sigma$ revealed good potentiality to assess the damage in the structure. This can be used to implement a system based on regression techniques, or similar.

In conclusion, the results obtained in terms of accuracy and latent spaces of train and test data in the validation of the proposed machine-learning method showed very promising potentialities in assessing the structural damage in typical historic masonry buildings.

## 6 Future work

A possible next step of the present study is to use representative latent features as input to a regression system to evaluate the prediction capability of the AI system.

Moreover, several dimensionalities of the latent space could be explored. In fact, in the present work the AI procedure was implemented with 64 dimensions and very good results were achieved. Nonetheless, results could be theoretically improved by using higher dimensionalities, which will be investigated in detail in a future work.

In addition, in the present work we limited the amount of analyzed data to the best quality signals (18 best signal-to-noise markers) for simplicity reasons and to verify that the proposed methodology is able to provide good results without using too many measurement positions. However, shaking table specimens are typically monitored with about a hundred markers and in future work we will consider analyzing all of them.

Also, we plan to investigate the effect of several strategies for data augmentation like noise addition to the signals or/and using time segmentation with overlapping time windows.

Finally, for the application to real monitoring of existing buildings, some challenges in scaling the method to full-scale structures and different sensor configurations will be considered in future studies. In particular, application to ambient vibration tests with typical on-the-field accelerometric sensors with different configurations needs to be explored, as well as the effect of changing environmental conditions and excitations (i.e. vibrations intensity and frequency).

## References


[1] C. Calderini et al., *Shaking table tests of an arch-pillars system and design of strengthening by the use of tie-rods*, *Bull. Earthquake Eng.* **13** (2014) 279.

[2] I. Roselli et al., *Relative displacements of 3D optical markers for deformations and crack monitoring of a masonry structure under shaking table tests*, *Int. J. Comput. Methods Exp. Meas.* **7** (2019) 350.

[3] Y.-J. Cha, R. Ali, J. Lewis and O. Büyüköztürk, *Deep learning-based structural health monitoring*, *Autom. Constr.* **161** (2024) 105328.




2025 JINST 20 C06063

[4] A. Malekloo, E. Ozer, M. AlHamaydeh and M. Girolami, *Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights*, *Struct. Health Monit.* **21** (2021) 1906.

[5] E.J. Cross et al., *Physics-informed machine learning for structural health monitoring*, in *Structural Health Monitoring Based on Data Science Techniques. Structural Integrity* A. Cury et al. eds., Springer Cham (2022), p. 347–367, DOI:10.1007/978-3-030-81716-9_17.

[6] A. De Angelis et al., *Dynamic identification methods and artificial intelligence algorithms for damage detection of masonry infills*, *J. Civil Struct. Health Monit.* **14** (2024) 1383.

[7] A. Pollastro, G. Testa, A. Bilotta and R. Prevete, *Semi-Supervised Detection of Structural Damage Using Variational Autoencoder and a One-Class Support Vector Machine*, *IEEE Access* **11** (2023) 67098.

[8] I. Kandel and M. Castelli, *The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset*, *ICT Express* **6** (2020) 312.

[9] J. An et al., *Variational autoencoder based anomaly detection using reconstruction probability*, *Special lecture on IE* **2** (2015) 1.

[10] D. Palumbo et al., *Machine learning analysis of 3D motion markers data of a rubble masonry building prototype under dynamic identification shaking table tests*, in the proceedings of the *International Conference of Metrology for Archaeology and cultural heritage (MetroArchaeo2024)*, Valletta, Malta, 7–9 October 2024.

[11] G.E. Hinton and R.R. Salakhutdinov, *Reducing the Dimensionality of Data with Neural Networks*, *Science* **313** (2006) 1127647.

[12] D. Higgins et al., *Beta-VAE: Learning basic visual concepts with a constrained variational framework*, in the proceedings of the *5th International Conference on Learning Representations*, Toulon, France, 24–26 April 2017, https://openreview.net/forum?id=Sy2fzU9gl.

[13] X. Ma, Y. Lin, Z. Nie and H. Ma, *Structural damage identification based on unsupervised feature-extraction via Variational Auto-encoder*, *Measurement* **160** (2020) 107811.

[14] D.M. Hawkins, S.C. Basak and D. Mills, *Assessing Model Fit by Cross-Validation*, *J. Chem. Inf. Comput. Sci.* **43** (2003) 579.

[15] S. Yadav and S. Shukla, *Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification*, in the proceedings of the *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, Bhimavaram, India, 27–28 February 2016, p. 78–83 [DOI:10.1109/iacc.2016.25].

[16] I. Roselli, M. Mongelli, A. Tatì and G. de Canio, *Analysis of 3D Motion Data from Shaking Table Tests on a Scaled Model of Hagia Irene, Istanbul*, *Key Eng. Mater.* **624** (2014) 66.

2025 JINST 20 C06063