Damage detection coupling convolutional neural networks and numerical simulations

Kosmas Dragos and Kay Smarsly
Hamburg University of Technology, Germany
kosmas.dragos@tuhh.de

Abstract. Damage detection in structural health monitoring (SHM) typically relies on global structural dynamic parameters, which may be hardly sensitive to the onset of structural damage. As a result, data analysis in SHM has been increasingly drawing from the field of machine learning (ML) for detecting subtle patterns in structural response data, indicative of structural damage. However, ML for damage detection requires structural response data corresponding to damage, which is hardly available. This paper proposes a damage detection approach coupling convolutional neural networks with numerical simulations. Specifically, the capabilities of convolutional neural networks are utilized for classifying structural response data into different damage scenarios. Furthermore, structural response data for training the convolutional neural networks is generated through numerical simulations. The proposed approach is validated through simulations of a steel pylon, showcasing that the proposed approach is capable of correctly classifying images of structural response data corresponding to different damage scenarios.

1. Introduction

A cornerstone of structural maintenance strategies for civil infrastructure is the ability to detect the onset of structural damage, which enables stakeholders to implement timely interventions that ensure public safety (Frangopol and Liu, 2007). In this direction, structural health monitoring (SHM) has matured over the last decades into a structural maintenance strategy that complements nondestructive evaluation techniques, such as visual inspections, by providing continuous information on the structural behavior of civil infrastructure (Dragos and Smarsly, 2017). In particular, the continuous acquisition and analysis of structural response data has proven capable of detecting structural damage at early stages (Farrar and Worden, 2006).

Damage detection in SHM builds upon data analysis (Catbas, 2009). Specifically, damage detection methods have been developed that convert structural response data, collected from civil infrastructure being monitored, into "knowledge" on the structural condition, based either on analyzing statistical relationships (e.g. cross-correlations) between different sources of data ("data-driven analysis") or on matching numerical simulations to the data ("physics-based analysis") (Cross et al., 2022). Knowledge from data-driven analysis is frequently provided in the form of structural dynamic parameters, such as eigenfrequencies and mode shapes, extracted as part of vibration-based SHM without making assumptions on the physical properties of civil infrastructure (Dragos and Smarsly, 2015). By contrast, physics-based analysis relies on fine-tuning structural parameters of numerical models so that the behavior of the models matches the actual structural behavior as close as possible. Common ground between most data-driven analysis and physics-based analysis methods is that both types of methods usually yield knowledge on the global structural behavior parameters: On the one hand, data-driven analysis is restricted to structural dynamic parameters at locations measured by SHM systems and, on the other hand, physics-based analysis is typically restricted to fine-tuning global structural parameters, such as elasticity moduli. However, global structural behavior parameters may be hardly sensitive to small changes, induced by the onset of structural damage. Therefore, recent research in SHM has been exploiting the powerful computational capabilities of artificial intelligence (AI) (Smarsly et al., 2007). Specifically, machine learning (ML), representing an

AI subset, has advantageously been deployed in an attempt to associate subtle changes in structural parameters with structural damage scenarios (Rafiei and Adeli, 2018).

Early research approaches on ML-based damage detection for SHM can be traced back to the early 1990s, mostly proposing artificial neural networks for finite element model updating. Seminal works, in this context, include the multi-layer perceptrons (feed-forward neural networks), introduced by Wu et al. (1992), Elkordy et al. (1993), and Masri et al. (1996), which primarily map perturbations of structural parameters with structural dynamic parameters, such as mode shapes, eigenfrequencies, and frequency-response functions. Moreover, neural-network-based (NN-based) approaches have been proposed for associating the effective cross-section areas of truss bridges with static displacements (Pandey and Barai, 1995) and for applying damage detection at a substructure level (Yun et al., 2000). Building on the seminal works, further approaches have accounted for damping in NN-based model updating (Lu and Tu, 2004), modeling errors in finite element simulations (Lee et al., 2005), and nonlinearities (Hasançebi and Dumlupinar, 2013). In recent literature, Park et al. (2017) have proposed a multilayer perceptron for mapping boundary conditions (rotational spring constants) to bridge responses, and Deng et al. (2017) have introduced a NN-based interval model updating method, accounting for the stochasticity of structural parameters. Finally, research attention has increasingly been drawn to neural networks for classification, such as convolutional neural networks, as reported in Abdeljaber et al. (2018), Zhang et al. (2018), and Sung et al. (2021). The powerful mapping capabilities of convolutional neural networks enable associating subtle patterns in structural response data with changes in structural parameters. However, convolutional neural networks fall within the category of supervised learning, thus requiring labeled input data corresponding to damage scenarios, which is hardly available in practice.

This paper presents a damage detection approach for detecting structural damage using convolutional neural networks trained with structural response data that are generated via simulations of damage scenarios using numerical modeling. The proposed approach is a feasibility study towards transferring knowledge from simulations to practice in the form of features that can be detected by a convolutional neural network (CNN), to which time-domain structural response data is fed as input. The classification capabilities of convolutional neural networks have been investigated in previous work of the authors on sensor fault identification (Fritz et al., 2022). The purpose of this paper is to showcase how the classification capabilities can be utilized to detect features in image representations of the structural response data, without prior "manual" extractions of structural dynamic parameters that are common in several NN-based model updating methods. The proposed approach is validated via numerical simulations of a real-world steel pylon, showcasing the capability of the CNN to detect patterns in the structural response data that are associated with limited damage inflicted upon the pylon. In the remainder of the paper, first the theoretical background of the damage detection approach is illuminated, followed by the implementation and validation tests, using a numerical model of the steel pylon. The paper ends with a summary and a short discussion of the conclusions drawn, as well as with a brief outlook on follow-on research.

2. Detecting structural damage using numerical simulations and convolutional neural networks

In this section, the theoretical background of the damage detection approach is presented. First, convolutional neural networks are briefly introduced. Then, coupling of convolutional neural networks with numerical simulations towards damage detection is discussed.

The classification capabilities of convolutional neural networks have been the topic of intensive research in recent years across several disciplines (Li et al., 2022). Tasks assigned to convolutional neural networks largely involve classifying images on the basis of patterns and objects "recognized" on the images. In this context, convolutional neural networks are trained to detect patterns and objects by being presented with several images, "labeled" according to the classes considered in the classification, and by scanning the images. The basic architecture of the CNN used in this study is illustrated in Figure 1 by means of a typical CNN architecture, and it is further described in the following paragraphs.

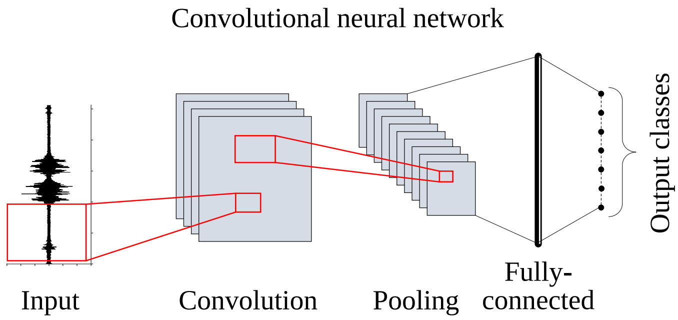

Figure 1: Layout of a typical convolutional neural network.

The basic architecture of the CNN used in this study follows a concept similar to the feedforward artificial neural network (ANN) that has been implemented by the authors in the context of SHM applications (Smarsly and Law, 2014; Dragos and Smarsly, 2016). The architecture includes an input layer, one or more hidden layers, and an output layer. Contrary to the feedforward ANN concept, the input layer of the CNN accepts images as input data, instead of structural response data. Furthermore, the output layer consists of classes, with each class characterizing different damage scenarios. As regards the hidden layers, three types of layers are included in the CNN:

- *Convolution layers* scan images using "kernel functions". For a typical convolution layer, each kernel function is essentially a matrix of weights, which is progressively "moved" across an image, in steps defined as "strides". The dimensions of the kernel function must be equal to or smaller than the image dimensions. In every stride, the convolution between the weights and a block of the image of the same size as the kernel function matrix is computed (i.e. the sliding dot product). Upon completing the scanning of the image with all kernel functions, the convolutions are used to populate the so-called "feature map" of the convolution layer. Different kernel functions are employed for scanning different features in the images. The convolutions are passed through an activation function for producing the output of the convolution layer.

- *Pooling layers* reduce the dimensionality of the outputs of convolutional layers. Specifically, a pooling layer combines the outputs from several neurons of a convolutional layer ("cluster of neurons") into a single neuron in the next convolutional layer. As such, a pooling layer is usually inserted between two convolutional layers or between a convolutional layer and a fully connected layer. Pooling is performed either by computing the average of the outputs ("average pooling") or by using the maximum output in the cluster of neurons.

- *Fully connected layers* connect neurons between successive layers. One fully connected layer is placed between the last pooling layer and the output layer to perform the classification. The neuron-by-neuron connection of the fully connected layer ensures that all the outputs of the pooling layer are considered for the classification.

For performing damage detection, the CNN are trained with images of structural response data corresponding to different damage scenarios. Training essentially involves updating the weights of the kernel functions to improve the accuracy of image classification. The images used for training are organized in a data set, from which three subsets are deducted:

(i) The *training set* typically comprises 70% of the data set. The training set is propagated through the CNN in "mini batches", the propagation of one mini-batch being referred to as "iteration". In each iteration, the classification accuracy is estimated and, depending on the error in the classification, the weights are adjusted according to the so-called "learning rate". The propagation of all mini-batches of the training set is termed "epoch".

(ii) The *validation set* typically consists of 20% of the data set. The validation set is propagated through the CNN periodically during training, after a predefined number of iterations. The purpose of the validation set is to ensure – during the training – that the CNN can "generalize" in classifying images, i.e. that the same classification accuracy is achieved between images of the training set and an independent set of images, thus avoiding overfitting.

(iii) The *testing set* typically consists of 10% of the data set. Upon completing all epochs, training is terminated. The testing set, representing an independent data set that the CNN has not "seen" during training, is then propagated through the CNN to estimate the classification accuracy.

For training the CNN, i.e. to detect damage effectively, abundant images of structural response data from different damage scenarios are generated to populate the data set. Evidently, in real-world structures, structural response data from damage scenarios is hardly available. Therefore, in this study, the data set is populated with numerical simulations using finite element analysis. In particular, for a structure equipped with an SHM system, a finite element (FE) model is created and calibrated using preliminary structural response data collected from the SHM system. The preliminary structural response data is also used to estimate the load cases, to which the structure is subjected. Thereupon, simulations are devised by performing time-history analyses of the FE model. The simulations include the load cases, previously defined, and perturbations of structural parameters of the FE model indicative of structural damage. The images of the structural responses from the simulations are organized in classes, depending on the damage scenario, to be used for training the CNN. It should be noted that the information contained in images of structural responses in the form of time-response plots, such as the image exemplarily shown in the input layer of Figure 1, may be relatively poor. As a result, the structural responses from the simulations are first subjected to continuous wavelet transform (CWT), which has proven capable of exposing rich features in structural response data (Peralta et al., 2021):

$$L_x(a,\tau) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t)\,\psi\left(\frac{t-\tau}{a}\right) dt \,. \tag{1}$$

In Equation 1, L_x is the CWT coefficient of function x, and $\psi(t)$ is the "mother" wavelet function, which is used as a basis for producing variant wavelet functions by changing the "scale" a and the shift factor τ. The steps towards performing damage detection are shown in Figure 2. The implementation and validation of the proposed approach are presented in the next section.

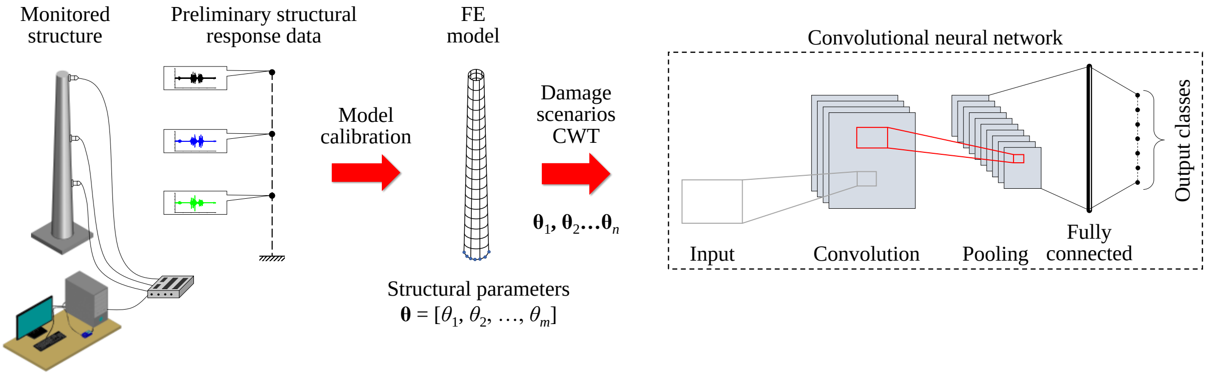

Figure 2: Damage detection steps of the proposed approach.

3. Implementation and validation

The tests devised to validate the proposed approach are presented in this section. First, the simulations conducted to generate the data set are discussed, followed by a description of the CNN implemented for the purposes of the tests. Finally, the training of the CNN is explained, and the results from propagating the testing set are presented and discussed.

The structure used for the simulations is a steel pylon, shown in Figure 3, which, in laboratory tests, is subjected to impact hammer testing. The pylon consists of two welded parts with a total height of 2,480 mm and a tapered circular hollow section (CHS) with diameters of approximately 200 mm at the base and 170 mm at the top. The bottom part has a thickness of 4 mm and a height of 130 mm, practically serving as an “reinforced” transition zone to a base plate of dimensions 400 mm × 400 mm × 20 mm (length × width × thickness). The thickness of the upper part is equal to 2.5 mm. The base plate is bolted to a strong floor at four points, ensuring adequate fixity.

Figure 3: Steel pylon used for the simulations.

The FE model of the pylon is created and analyzed using the ANSA-META-EPILYSIS suite, which includes a finite element pre-processing software package (ANSA), a solver (EPILYSIS), and a post-processing software package (META) (BETA CAE SA, 2021). The pylon is modeled with approximately 4,000 shell elements with six degrees of freedom. The material properties are computed from experimental tests on specimens extracted from the pylon and are equal to E = 230 GPa (modulus of elasticity) and ρ = 7.88 t/m^3 (material density). The bolted connections are modeled as fully fixed, using rigid body constraints. The FE model of the pylon is shown in Figure 4.

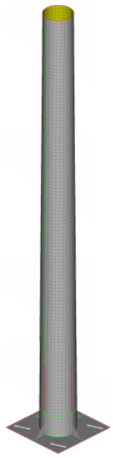

Figure 4: Finite element model of the pylon used for simulations.

The load cases considered for the simulations are indicative of the loads to which the structure is subjected, i.e., in this case, impact hammer testing. As a result of the impact hammer testing, preliminary acceleration response data is collected using an SHM system. The preliminary acceleration response data is used to fine-tune structural parameters of the FE model, ensuring that the structural dynamic parameters of the FE model match the frequency content of the preliminary data, as well as to estimate the load cases. Thereupon, three scenarios are devised, one corresponding to an initial ("undamaged") condition and two corresponding to structural damage in the x and y direction, respectively. Structural damage is simulated by removing shell elements from a thin 1-mm wide zone of finite elements at the welding between the two parts of the pylon, which represents a weak part of the pylon, as shown in Figure 5.

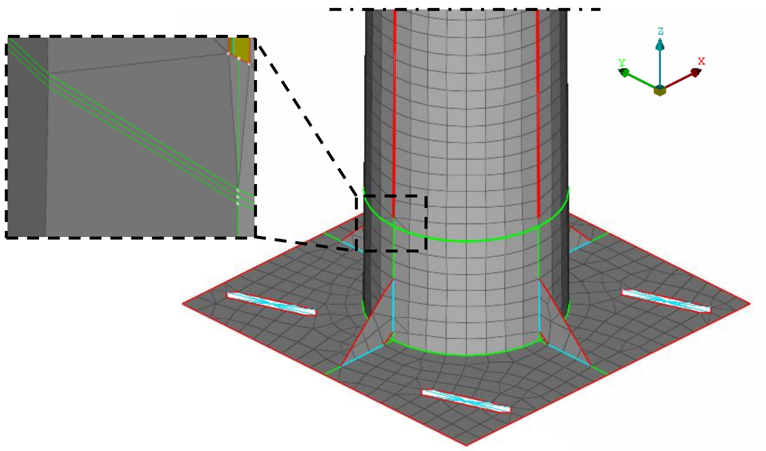

Figure 5: Simulation of damage in the FE simulations.

In each scenario, the time-history analyses of the simulations are conducted with a sampling rate of f_s = 256 Hz, and each analysis yields 120,000 points in the structural responses. The structural responses of the simulations are subjected to CWT with 1,000-point-length windows, producing 120 images for each scenario, which are used as input to the CNN. Exemplary CWT images are shown in Figure 6.

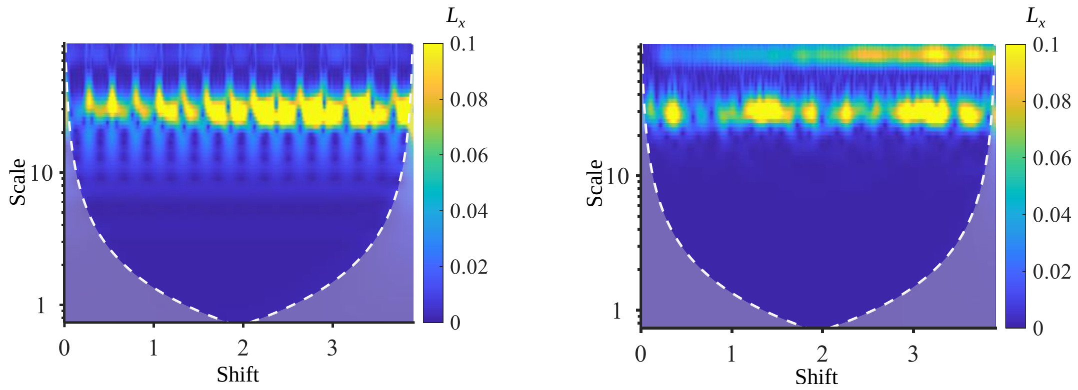

Figure 6: CWT images used for training the CNN: Damage in *x* direction (left) and in *y* direction (right).

Selecting the architecture of the convolutional neural network for the proposed approach is usually a trial-and-error process, attempting to find a trade-off between computational burden and classification accuracy. However, for the sake of simplicity of the feasibility study presented herein, the convolutional neural network employed for the purposes of the tests builds upon the pre-trained architecture of "GoogleNet", which has been adopted in several AI-ML studies (Yoo, 2015). Specifically, the outputs of the fully connected layer and of the output layer, respectively, of the GoogleNet CNN are adjusted to the three classes used in this study. The CNN is trained using the Matlab deep learning toolbox (Mathworks, 2018). For each class, 84 images are used as a training set, 24 images as a validation set, and 12 images as a testing set. The mini-batch size is set to 21 images, resulting in 4 iterations per epoch, and the total number of epochs is set to 100. The learning rate is set equal to 0.02, gradually reduced every 10 epochs by a factor of 0.5. Weight updating is performed using the steep gradient descent method.

Upon completing training, the maximum classification accuracy, reached during training, for the validation set is 97.78%, which is deemed adequate for the purposes of the study. The results from applying the testing set are shown in the form of a confusion matrix in Table 1.

Table 1: Results from propagating the testing set through the CNN.

		Predicted label		
		Undamaged	Damage in *x* dir.	Damage in *y* dir.
Actual label	Undamaged	12	0	0
	Damage in *x* dir.	1	11	0
	Damage in *y* dir.	1	0	11

As can be seen from Table 1, the CNN classifies correctly 34 out of the 36 images of the testing set, corresponding to a classification accuracy of 94.5%, which is compatible with the accuracy of the validation set. Furthermore, considering that SHM systems are employed for long-term operation, damage detection in real-world structures using the proposed approach would rely on classifying several images, thus overcoming potential problems from isolated false classifications. The next step, which will be addressed in future work, involves obtaining structural response data from the actual structure and classifying the data using the CNN.

4. Summary and conclusions

Structural health monitoring strategies attempt to detect the onset of structural damage in civil infrastructure. Damage detection typically relies on analyzing structural response data, collected from structures being monitored, either utilizing statistical relationships within the structural response data or calibrating numerical models according to the structural response data. However, the knowledge obtained from data analysis in SHM usually concerns global structural dynamic parameters, which may be hardly sensitive to the onset of structural damage. To overcome the limitations of global structural dynamic parameters, data analysis in SHM has been increasingly drawing from the field of artificial intelligence and machine learning for detecting subtle patterns in the structural response data that are indicative of structural damage. Despite the powerful computational capabilities, common AI and ML methods in SHM require vast amounts of structural response data for training, which, for the case of structural damage, is hardly available.

This paper has proposed a damage detection approach coupling convolutional neural networks with numerical simulations. Specifically, the proposed approach suggests using the capabilities of convolutional neural networks for classifying structural response data into damage scenarios. Furthermore, generating structural response data for training the convolutional neural networks is achieved through numerical simulations for each damage scenario, the results of which are subjected to continuous wavelet transform for producing images of structural responses. The damage detection approach has been implemented and validated through simulations of a steel pylon. In particular, time-history simulations of a finite element model of the pylon have been conducted for generating structural response data corresponding to three scenarios, one without damage and two with damage along two horizontal directions. The structural responses of the simulations have been used to train a CNN. The results of the tests have shown that the CNN has been trained successfully and has been capable of correctly classifying the independent set of images of the testing set with high classification accuracy. Future work will focus on using a CNN, trained with simulations, for classifying structural response data from actual structures, as well as on optimizing the architecture of the CNN.

Acknowledgments

The authors gratefully acknowledge the support offered by the German Research Foundation (DFG) through grant SM 281/20-1. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the DFG.

References

Abdeljaber, O., Avci, O.m Kiranyaz, M.S., Boashash, B., Sodano, H. and Inman, D.J. (2018). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. Neurocomputing 275, pp. 1308-1317.

BETA CAE SA (2021). A guide to the superiority of BETA CAE systems software suite. Available at: https://www.beta-cae.com/pdf/ansa_meta_superiority.pdf (accessed on 04/13/2023).

Catbas, F.N. (2009). Structural health monitoring: applications and data analysis. In: Karbhari, V.M. and Ansari, F. (eds.) "Structural health monitoring of civil infrastructure systems". Pp. 1-39. Cambridge, UK: Woodhead Publishing Ltd.

Cross, E.J., Gibson, S.J., Jones, M.R., Pitchford, D.J., Zhang, S. and Rogers, T.J. (2022). Physics-informed machine learning for structural health monitoring. In: Cury, A., Ribeiro, D., Ubertini, F. and Todd, M. (eds.) "Structural health monitoring based on data science techniques". Pp. 347-367. Cham, Switzerland: Springer Nature AG.

Deng, Z., Guo, Z. and Zhang, X. (2017). Interval model updating using perturbation method and radial basis function neural networks. Mechanical Systems and Signal Processing, 84, 699-716.

Dragos, K. & Smarsly, K. (2017). Decentralized infrastructure health monitoring using embedded computing in wireless sensor networks. In: Sextos, A. & Manolis, G. D. (eds.). Dynamic Response of Infrastructure to Environmentally Induced Loads. Pp. 183-201. Cham, Switzerland: Springer International Publishing AG.

Dragos, K. & Smarsly, K. (2015). A Comparative Review of Wireless Sensor Nodes for Structural Health Monitoring. In: Proceedings of the 7th International Conference on Structural Health Monitoring of Intelligent Infrastructure. Turin, Italy, 07/01/2015.

Dragos, K. and Smarsly, K. (2016). Distributed adaptive diagnosis of sensor faults using structural response data. Smart Materials and Structures 25(10), 105019.

Dragos, K. & Smarsly, K. (2017). Decentralized infrastructure health monitoring using embedded computing in wireless sensor networks. In: Sextos, A. & Manolis, G. D. (eds.) "Dynamic Response of Infrastructure to Environmentally Induced Loads". Pp. 183-201. Cham, Switzerland: Springer International Publishing AG.

Elkordy, M.F., Chang, K.C. and Lee, G.C. (1993). Neural networks trained by analytically simulated structural damage states. ASCE Journal of Computing in Civil Engineering 7(2), pp. 130-145.

Farrar, C.R. and Worden, K. (2006). An introduction to structural health monitoring. Philosophical Transactions of the Royal Society 365, pp. 303-315.

Frangopol, D.M. and Liu, M. (2007). Maintenance and management of civil infrastructure based on condition, safety, optimization, and life-cycle cost. Structure and Infrastructure Engineering 3(1), pp. 29-41.

Fritz, H., Peralta, J., Legatiuk, D., Steiner, M., Dragos, K. and Smarsly, K. (2022). Fault diagnosis in structural health monitoring systems using signal processing and machine learning techniques. In: Cury, A., Ribeiro, D., Ubertini, F., Todd, M. D. (eds.). Structural health monitoring based on data science techniques. Pp. 143-164. Cham, Switzerland: Springer.

Hasançebi, O. and Dumlupinar, T. (2013). Linear and nonlinear model updating of reinforced concrete T-beam bridges using artificial neural networks. Computers and Structures 119, pp. 1-11

Lee, J.J., Lee, J.W., Yi, J.H., Yun, C.B. and Jung, H.Y. (2004). Neural networks-based damage detection for bridges considering errors in baseline finite element models. Journal of Sound and Vibration 280 (3-5), pp. 555-578.

Li, Z., Liu, F., Yang, W., Peng, S. and Zhou, J. (2022). A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Transactions on Neural Networks and Learning Systems 33(12), pp. 6999-7019.

Lu, Y. and Tu, Z. (2004). A two-level neural network approach for dynamic FE model updating including damping. Journal of Sound and Vibration 275(3-5), pp. 931-952.

Masri, S.F., Nakamura, M., Chassiakos, A.G. and Caughey, T.K. (1996). Neural network approach to detection of changes in structural parameters. ASCE Journal of Engineering Mechanics 122(4), pp. 350-360.

Mathworks (2018). Deep learning with Matlab. Available at: https://www.mathworks.com/content/dam/mathworks/fact-sheet/deep-learning-with-matlab-quick-start-guide.pdf (accessed on 04/13/2023).

Pandey, P.C. and Barai, S.V. (1995). Multilayer perceptron in damage detection of bridge structures. Computers and Structures 54(4), pp. 597-608.

Park, Y.-S., Kim, S., Kim, N. and Lee, J.-J. (2017). Finite element model updating considering boundary conditions using neural networks. Engineering Structures 150, pp. 511-519.

Peralta, J., Fritz, H., Dadoulis, G., Dragos, K. and Smarsly, K. (2021). Automated decision making in structural health monitoring using explainable artificial intelligence. In: Proceedings of the 28th International Workshop on Intelligent Computing in Engineering (EG-ICE). Berlin, Germany, 06/30/2021.

Rafiei, M.H. and Adeli, H. (2018). A novel unsupervised deep learning model for global and local health condition assessment of structures. Engineering Structures 156, pp. 598-607.

Smarsly, K., Lehner, K. & Hartmann, D. (2007). Structural Health Monitoring based on Artificial Intelligence Techniques. In: Proceedings of the International Workshop on Computing in Civil Engineering (EG-ICE). Pittsburgh, PA, USA, 07/24/2007.

Smarsly, K. and Law, K. H. (2014). Decentralized Fault Detection and Isolation in Wireless Structural Health Monitoring Systems using Analytical Redundancy. Advances in Engineering Software 73(2014), pp. 1-10.

Sung, H., Chang, S. and Cho, M. (2021). Component model synthesis using model updating with neural networks. Mechanics of Advanced Materials and Structures 30(2), pp. 400-411.

Wu, X. and Garrett, J.H. (1992). Use of neural networks in detection of structural damage. Computers and Structures 42(4), pp. 649-659.

Yoo, H.-J. (2015). Deep convolutional neural networks in computer vision: a review. IEEE Transactions on Smart Processing and Computing 4(1), pp. 35-43.

Yun, C.-B. and Bahng, E.Y. (2000). Substructural identification using neural networks. Computers and Structures 77(1), pp. 41-52.

Zhang, Y., Miyamori, Y., Mikami, S. and Saito, T. (2019). Vibration-based structural state identification by a 1-dimensional convolutional neural network. Computer-aided civil and infrastructure engineering 34(9), pp. 822-839.