

Research on Structural Damage Identification Method Based on Convolutional Neural Network

Zongchao Liu¹, Shuai Teng^{2,3}, Shaodi Wang⁴

¹College of Finance and Commerce, Guangzhou Railway Polytechnic, Guangzhou, China

²School of Intelligent Construction and Civil Engineering, Zhongyuan University of Technology, Zhengzhou, China

³Research Center for Wind Engineering and Engineering Vibration, Guangzhou University, Guangzhou, China

⁴Earthquake Engineering Research and Test Center, Guangzhou University, Guangzhou, China

Email: liuzongchao@gtxy.edu.cn

How to cite this paper: Liu, Z.C., Teng, S. and Wang, S.D. (2025) Research on Structural Damage Identification Method Based on Convolutional Neural Network. *Open Journal of Applied Sciences*, 15, 3524-3533. <https://doi.org/10.4236/ojapps.2025.1511228>

Received: November 1, 2025

Accepted: November 14, 2025

Published: November 17, 2025

Copyright © 2025 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Structural damage identification is a critical task in civil engineering for ensuring the safety and serviceability of infrastructures. Traditional methods often rely on manually selected features from vibration data, which may lack sensitivity to incipient or localized damage. In recent years, convolutional neural networks (CNNs) have demonstrated remarkable capability in automatically extracting discriminative features from raw data, providing a powerful data-driven alternative for structural health monitoring. This paper proposes a novel structural damage identification method based on a CNN framework. Numerical simulations of the free vibration of a simply supported beam were conducted using the finite element method. The modal strain energy of the first-order element was collected as the primary damage-sensitive feature. Through batch automatic calculation, a comprehensive dataset encompassing various damage scenarios—including different locations and severity levels—was efficiently generated to serve as training, validation, and testing samples for the network. The proposed CNN model was specifically designed and implemented in MATLAB to adapt to the characteristics of the structural damage detection data. This study systematically verifies the recognition performance of the CNN from three key aspects: identification of single damage location, quantification of damage severity at a single location, and detection of multiple damage locations. The results indicate that the CNN can not only accurately identify the location of structural damage with high precision but also achieve a considerable level of accuracy in quantifying the degree of damage, demonstrating its strong potential for comprehensive damage assessment in engineering structures.

Keywords

Convolutional Neural Network, Structural Damage Recognition, Simply Supported Beams, Modal Strain Energy, Finite Element Simulation

1. Introduction

In recent years, structural damage identification has become a hot topic of concern for scholars [1]. With the development of science and technology, although some problems have been solved, new problems continue to arise as structural forms become more complex [2]. At present, there are mainly methods for structural damage identification based on frequency variation [3], modal information [4], and intelligent algorithms (neural networks [5], genetic algorithms [6], wavelet analysis [7]). The structural damage recognition method based on neural networks has achieved significant results [8]. Convolution Neural Network (CNN) is a new method for identifying structural damage that differs from traditional neural networks.

A significant challenge in traditional vibration-based damage identification is the reliance on manually selected features and their often-limited sensitivity to incipient, localized damage, especially in the presence of environmental and operational variability. CNNs offer a distinct advantage by automatically learning optimal damage-sensitive features directly from raw or pre-processed monitoring data, such as modal parameters or strain energy. This data-driven approach reduces the need for expert knowledge in feature engineering and enhances the potential for detecting subtle changes in structural behavior. This paper leverages this capability by using modal strain energy as the input, a parameter known for its high sensitivity to local stiffness reductions [9].

This article conducts numerical simulations on simply supported beams to obtain the modal strain energy of structural elements, which is used as input for the network to analyze the effectiveness of convolutional neural networks in locating structural damage, determining the degree of structural damage, and identifying multiple damage locations.

2. Basic Principles of CNNs

The CNN typically consists of multiple layers, including input layer, convolutional layer, pooling layer, activation layer, and output layer. These layers are stacked to form a deep architecture capable of learning hierarchical features from simple to complex. Convolution and pooling are two important execution commands for CNN, forming the core operations for feature extraction and data reduction.

There are two parts involved in the convolution process, as shown in **Figure 1**. The first part is called the input, which typically represents the raw data or features from a previous layer. The second part is the convolution kernel (or filter), which is a small matrix of weights designed to detect specific patterns, such as edges or

textures in data. In the convolution process, according to the defined convolution kernel weight size and step size (stride), the convolution kernel is initially located at the top left corner of the input matrix, and the weights on the convolution kernel are multiplied with the corresponding input data in a localized receptive field. This element-wise multiplication is followed by a summation of all products to produce a single value in the output matrix, known as a feature map. This value is then moved down or to the right by a certain step size, and the product summation is repeated until the kernel has scanned the entire input data, systematically covering all possible locations. This sliding-window mechanism allows the same kernel to detect a specific feature regardless of its position in the input, providing the model with translation invariance. Finally, after convolution operation, a new matrix is formed, which highlights the presence of the features learned by the kernel.

The pooling layer is another important part of CNN, which is used to reduce the dimensionality of the matrix, thereby decreasing the computational load and controlling overfitting by providing an abstracted form of representation. There are usually two different pooling methods, namely max pooling and average pooling. Maximum pooling (as shown in **Figure 2**) selects the maximum value from the region of the feature map covered by the pooling kernel, thereby retaining the most prominent features and enhancing the network's invariance to small shifts and distortions. Average pooling, on the other hand, computes the average value within the window, which can smooth out the features. Maximum pooling is generally considered superior to average pooling for tasks that require strong feature detection and positional robustness, so this article adopts maximum pooling. The size of the pooling window and the stride are key hyperparameters that determine the degree of down-sampling.

Following convolution and pooling, the non-linear connection of activation functions can enhance the learning ability of neural networks by introducing non-linearity, which is crucial for enabling the network to learn and represent complex, real-world relationships that are not linear. Without these functions, a CNN would simply be a linear model, drastically limiting its power. The commonly used non-linear activation functions include Sigmoid function, which compresses values into a range between 0 and 1 but can suffer from vanishing gradients; the ReLU function, which outputs the input directly if positive else zero, enabling faster training but sometimes leading to "dead neurons"; and the Leaky ReLU function, which addresses the dying ReLU problem by allowing a small, non-zero gradient when the unit is not active. In this paper, Leaky ReLU function is used to mitigate the potential issue of inactive neurons and to promote more robust learning throughout the network.

The output of the network can be classified or regressed, with the classification problem using the Softmax function to transform raw output scores into a probability distribution over multiple classes, and the regression problem using a linear activation or similar function to produce a continuous numerical value. This article will use different functions for different problems, selecting the appropriate

output layer activation based on the specific task of damage localization (classification) or severity estimation (regression).

The comprehensive working process of CNN can be summarized as follows: The initial convolutional layers extract low-level data features, such as basic patterns, and the subsequent pooling layers immediately reduce the data dimensionality, preserving only the most significant information. By repeating the above two processes in multiple stages, the network can progressively combine simple features into more complex ones, and high-level features of the data can be obtained, such as specific damage signatures, while the data can be significantly reduced in dimensionality for efficient processing. Then, the activation layer is applied after these operations to adapt to complex nonlinear problems, allowing the network to model intricate decision boundaries. Finally, the processed information is typically flattened and fed into fully connected layers, where the error is calculated based on the difference between the predicted and actual outputs, and backpropagation along with an optimization algorithm like gradient descent is performed to update the weights of the kernels and connections, thereby completing the iterative process of network training.

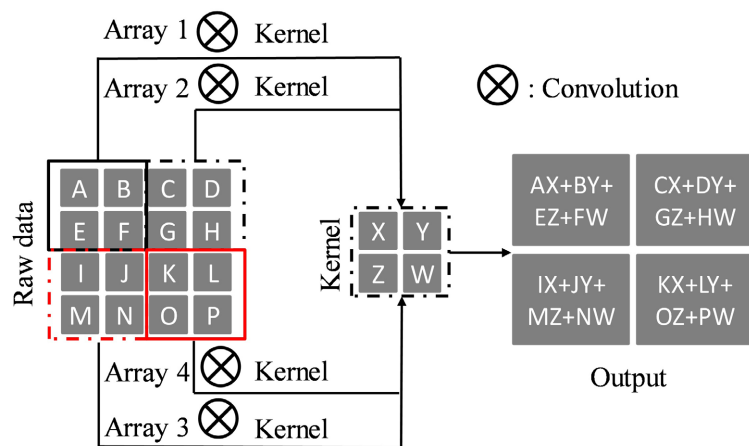


Figure 1. Convolution process.

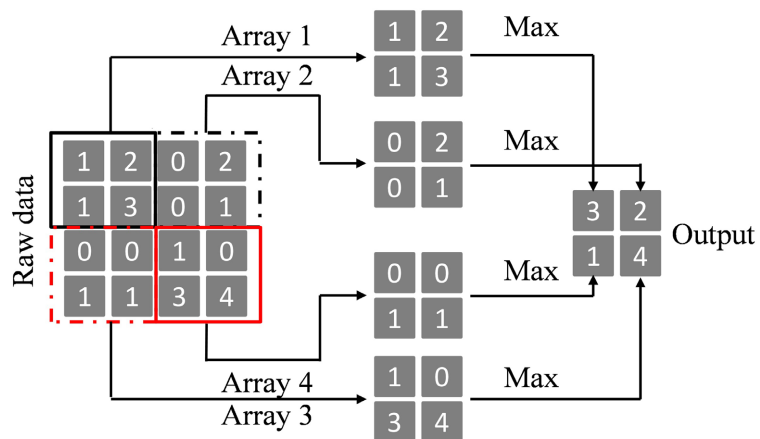


Figure 2. Pooling process.

3. Numerical Simulation

3.1. Data Acquisition

This article studies a rectangular section beam that is 9 m long, 0.3 m wide, and 0.2 m high. The structure is discretized into 36 elements, with each element numbered as shown in **Figure 3**.

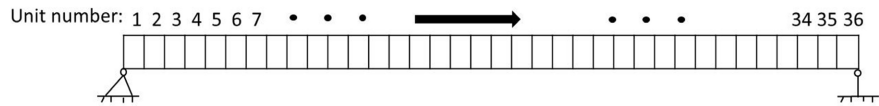


Figure 3. Simply supported beam.

Using ABAQUS solid element modeling, the material properties are as follows: Young's modulus of 211 Gpa, Poisson's ratio of 0.288, density of 7800/m³. The damage to the structure is achieved by changing the elastic modulus of the elements, simulating six degrees of damage of 15%, 30%, 45%, 60%, 75%, and 90% for each element, with corresponding elastic moduli reduced by 15%, 30%, 45%, 60%, 75%, and 90%, respectively.

By automating numerical calculations and extracting data, a significant amount of time can be saved and a large amount of data can be obtained. This article utilizes the parametric analysis function of Abaqus to simulate damage and collect first-order modal strain energy by changing the material properties of each element in the inp file. The different degrees of damage in different locations, as well as the different degrees of damage in multiple locations, can all be automatically calculated and data saved through Python programs.

This article mainly verifies in three aspects: CNN identification of single damage location in structures, CNN identification of damage degree in single location of structures, and CNN identification of multiple damage locations in structures.

For the identification of a single damage location, the dataset consists of 145 sets of data with no damage and 15%, 30%, 75%, and 90% damage to each unit (145 groups) as the training set, 37 sets of data with no structural damage and 45% damage to each unit (37 groups) as the validation set, and 36 sets of data with 60% damage to each unit (36 groups) as the test set.

For the recognition of the degree of damage to a single location, the dataset consists of 577 sets of data with no damage and 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 50%, 55%, 65%, 70%, 75%, 80%, 85%, and 90% damage to each unit (577 groups) as the training set, 37 sets of data with no damage to the structure and 45% damage to each unit (37 groups) as the validation set, and 36 sets of data with 60% damage to each unit (36 groups) as the testing set.

For the identification of multiple damage locations, the dataset consists of the following: randomly select (without duplication) 3 rods from 24 rods (1, 2, 4, 6, 7, 8, 10, 11, 13, 15, 16, 18, 19, 20, 22, 24, 25, 26, 28, 29, 31, 32, 34, and 36) for damage, with each rod having a damage degree of 50%. There are a total of 2024 cases, from which 10 sets of validation data and 10 sets of test data are selected, as shown

in **Table 1**.

Table 1. Damage location of validation and testing dataset.

Condition	Validation dataset	Testing dataset
1	29, 31, 32	28, 29, 31
2	29, 31, 34	28, 29, 32
3	29, 31, 36	28, 29, 34
4	29, 32, 34	28, 29, 36
5	29, 32, 36	28, 31, 32
6	29, 34, 36	28, 31, 34
7	31, 32, 34	28, 31, 36
8	31, 32, 36	28, 32, 34
9	31, 34, 36	28, 32, 36
10	32, 34, 36	28, 34, 36

3.2. Network Model

This article relies on the Deep Learning Toolbox resources in Matlab 2024a (MathWorks) to independently develop a convolutional neural network that can fully adapt to the input of structural damage detection data. The modal strain energy data extracted from the 36 discrete elements was formatted as a 1×36 feature vector, where the position of each value in the vector corresponds to the sequential element number along the beam length. This one-dimensional sequential input preserves the spatial order of the structural elements, allowing the CNN to learn location-sensitive features effectively.

The detailed architecture of the proposed CNN is as follows:

Input Layer: The input data is the modal strain energy vector of the 36 elements, resulting in an input size of 1×36 .

Convolutional Layers:

Conv1: One-dimensional convolution with 32 filters of size 3, stride 1, and zero-padding to maintain the sequence length.

Activation: Leaky ReLU with a slope of 0.01 for negative inputs.

Pool1: Max pooling with a pool size of 2 and stride 2.

Conv2: 64 filters of size 3, stride 1, followed by Leaky ReLU.

Pool2: Max pooling with a pool size of 2 and stride 2.

Fully Connected Layers:

The output from the last pooling layer is flattened into a feature vector.

FC1: A fully connected layer with 128 neurons, followed by Leaky ReLU.

FC2: A fully connected layer with 64 neurons, followed by Leaky ReLU.

Output Layer:

For damage localization (classification task): Softmax layer with 37 units (36 damage locations +1 no-damage case).

For damage severity (regression task): A single neuron with linear activation.

The proposed CNN model was trained in a supervised manner. The training process was configured with the following key hyperparameters: the Adam (Adaptive Moment Estimation) optimizer was used for stochastic gradient descent with an initial learning rate of 0.001. The mini-batch size was set to 16. The training was set to run for a maximum of 50 epochs. To mitigate overfitting and ensure efficient training, a validation-based early stopping criterion was implemented, which halted the training if the validation loss did not improve for 10 consecutive epochs. The model with the lowest validation loss during training was retained for final evaluation on the test set.

4. Result Analysis

Following the normalization of the numerical simulation data, the training and validation sets were fed into the independently constructed convolutional neural network. After the network training was completed, its performance was evaluated using the test set. **Figure 4** presents the results for single-damage localization. It can be observed that the network accurately identified the location of the damaged element in all test samples, achieving a 100% identification accuracy. This indicates the high reliability of the CNN in addressing the problem of single-damage localization.

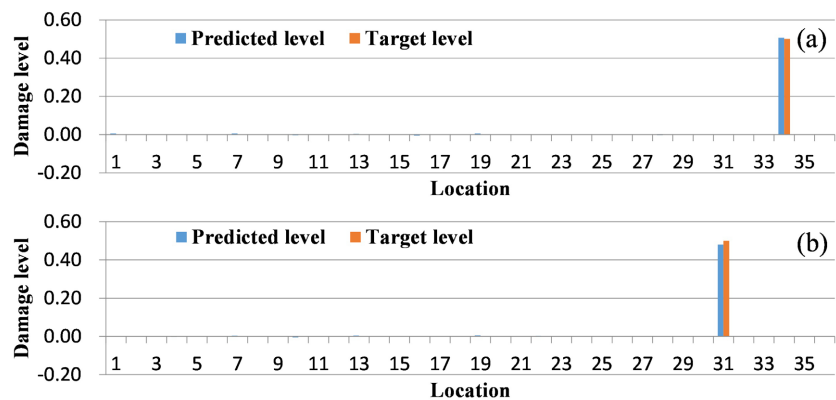


Figure 4. Single damage identification result.

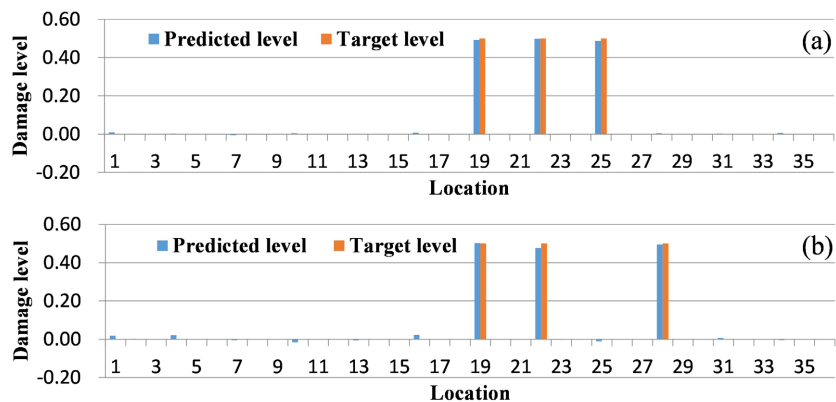


Figure 5. Multiple damage identification results.

Figure 5 demonstrates the performance of the CNN in the multiple-damage identification task. Although this problem is more complex due to coupling effects between multiple damaged elements, the CNN successfully and accurately identified the locations of all three damaged elements in every test case, with no instances of false positives or missed detections. This confirms that the constructed CNN model possesses strong feature extraction and generalization capabilities, enabling it to handle the complexities of multi-damage scenarios effectively.

Furthermore, in the task of damage severity identification, the CNN exhibited high precision across all damage levels (15% to 90%), with an average identification error of less than 5%. The error was even lower for moderate to severe damage ranges (45% to 90%), underscoring the model's strong suitability for quantifying the extent of structural damage.

In summary, the CNN demonstrated excellent performance in structural damage identification, meeting the precision requirements for engineering applications in single-damage localization, severity assessment, and multiple-damage identification.

5. Discussion and Limitations

While the proposed CNN-based method demonstrated excellent performance in the numerical studies presented herein, it is important to acknowledge certain limitations of this study that point to directions for future research.

First, the model was developed and validated using data from finite element simulations, which are inherently clean and free from the measurement noise and environmental variability encountered in real-world monitoring systems. The presence of noise could potentially degrade the model's accuracy and robustness, particularly in identifying minor damage. Future work will involve injecting various levels of noise into the training data to enhance the model's resilience and validate its performance against experimental data.

Second, the methodology was validated on a relatively simple simply-supported beam structure. The generalization of the approach to more complex structural types (e.g., frames, bridges, buildings) with more intricate dynamic characteristics remains to be thoroughly investigated. Applying the method to such structures may require adaptations in the input feature engineering or network architecture to capture their complex modal behaviors.

Addressing these limitations by testing the model's noise immunity and expanding its application to complex, real-world structures will be the critical next steps in translating this promising data-driven approach into practical engineering practice.

6. Conclusions

This paper proposed a structural damage identification method based on a Convolutional Neural Network (CNN). Using a simply supported beam as the research object, modal strain energy data obtained from finite element simulations

served as the network input. The performance of the CNN was systematically investigated for single-damage localization, damage severity quantification, and multiple-damage identification. The findings lead to the following conclusions:

The CNN demonstrated exceptional performance in the single-damage localization task, achieving 100% accuracy on the test set, which highlights its high reliability for pinpointing damage locations.

For damage severity identification, the CNN achieved high precision across various damage levels (15% - 90%), with an average error of less than 5%, demonstrating its effectiveness in quantifying the degree of structural damage.

In the challenging task of multiple-damage identification, the CNN accurately identified the locations of three damaged elements simultaneously without any misidentification, verifying its strong generalization capability in complex damage scenarios.

In conclusion, the CNN possesses significant advantages for structural damage identification, capable of simultaneously achieving accurate damage localization and effective severity assessment. Unlike traditional methods such as those based on frequency variation or wavelet analysis, which often rely on expert-driven manual feature selection and may lack sensitivity to incipient damage, the proposed CNN approach automatically learns optimal damage-sensitive features directly from raw modal data. This data-driven paradigm eliminates the need for sophisticated hand-crafted feature engineering, enhances robustness, and provides a more versatile and powerful solution for comprehensive structural assessment. This provides an efficient and reliable data-driven solution for structural health monitoring. Future work will extend this approach to more complex structural types and real-world engineering environments to further validate its practical potential.

Acknowledgements

This study was partially supported by the Talent introduction program of Guangzhou Railway Polytechnic (No. GTXYR2431), the Natural Science Foundation of Henan Province (No. 252300423474), and Guangdong Basic and Applied Basic Research Foundation (No. 2025A1515010155).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Fang, S., Li, L., Luo, Z., Fang, Z., Huang, D., Liu, F., *et al.* (2023) Novel FRP Interlocking Multi-Spiral Reinforced-Seawater Sea-Sand Concrete Square Columns with Longitudinal Hybrid FRP-steel Bars: Monotonic and Cyclic Axial Compressive Behaviours. *Composite Structures*, **305**, Article ID: 116487. <https://doi.org/10.1016/j.compstruct.2022.116487>
- [2] Mammeri, S., Barros, B., Conde-Carnero, B. and Riveiro, B. (2025) From Traditional Damage Detection Methods to Physics-Informed Machine Learning in Bridges: A Review. *Engineering Structures*, **330**, Article ID: 119862.

- <https://doi.org/10.1016/j.engstruct.2025.119862>
- [3] Teng, S., Chen, G., Liu, Z., Cheng, L. and Sun, X. (2021) Multi-Sensor and Decision-Level Fusion-Based Structural Damage Detection Using a One-Dimensional Convolutional Neural Network. *Sensors*, **21**, Article 3950. <https://doi.org/10.3390/s21123950>
- [4] Azimi, M., Eslamlou, A. and Pekcan, G. (2020) Data-Driven Structural Health Monitoring and Damage Detection through Deep Learning: State-Of-The-Art Review. *Sensors*, **20**, Article 2778. <https://doi.org/10.3390/s20102778>
- [5] Chen, S. and Li, G. (2001) Application of Artificial Neural Networks in Structural Damage Identification. *Vibration, Testing, and Diagnosis*, **2**, 116-124.
- [6] Li, H., He, C. and Yin, X. (2007) Structure Damages Detection Based on Genetic Algorithm with Experimental Validation. *Second International Conference on Innovative Computing, Information and Control (ICICIC2007)*, Kumamoto, 5-7 September 2007, 76. <https://doi.org/10.1109/icicic.2007.543>
- [7] Zhang, C.L. and Liu, H.Q. (2011) Study on Structural Damage Identification Based on Wavelet Analysis. *Advanced Materials Research*, **374**, 2588-2592. <https://doi.org/10.4028/www.scientific.net/amr.374-377.2588>
- [8] Pandey, P., Rai, A. and Mitra, M. (2022) Explainable 1-D Convolutional Neural Network for Damage Detection Using Lamb Wave. *Mechanical Systems and Signal Processing*, **164**, Article ID: 108220. <https://doi.org/10.1016/j.ymssp.2021.108220>
- [9] Stubbs, N., Kim, J.T. and Farrar, C.R. (1995) Field Verification of a Nondestructive Damage Localization and Severity Estimation Algorithm. *Journal of Structural Mechanics*, **23**, 145-164.