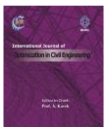
INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2025; 15(2):141-179



IMPACT IDENTIFICATION IN FRAMED STRUCTURES USING DEEP LEARNING: A CNN-BASED APPROACH OPTIMIZED BY BAYESIAN OPTIMIZATION

H. Sheikhpour[1], S. H. Mahdavi[2], S. Hamzehei-Javaran[1], and S. Shojaee [1*,†]
[1]*Department of Civil Engineering, Shahid Bahonar University of Kerman, Kerman, Iran*
[2]*Department of Civil Engineering, Higher Education Complex of Bam, Bam, Iran*

[DOI: 10.22068/ijoce.2025.15.2.629]

ABSTRACT

Accurate detection and localization of impacts in structural systems are crucial for safety and enabling effective structural health monitoring (SHM). This paper aims to identify multiple consecutive impacts in framed structures with unknown dynamic properties, using time-domain acceleration data. Traditional methods often struggle under complex conditions such as noisy environments and multiple impacts. To overcome these limitations, we propose a deep learning-based framework utilizing Convolutional Neural Networks (CNNs) to extract intricate patterns from acceleration signals. Input data are generated through high-fidelity numerical simulations based on the Finite Element Method (FEM), allowing precise control over impact characteristics and their spatial distribution. A fixed-length sliding window is employed to segment the acceleration time series, enabling the model to perform localized and near-real-time impact detection. To further improve model performance, Bayesian optimization is utilized for hyperparameter tuning, enhancing accuracy and efficiency over traditional grid search. The proposed model is numerically evaluated on two-dimensional structures: a steel pin-jointed camel-back truss and a shear frame. The results reveal that the proposed strategy achieves high accuracy in estimating the location, timing, and magnitude of impacts, even under noisy conditions. The key novelty of this research lies in combining deep learning with advanced optimization techniques to solve the impact detection problem in structures with unknown parameters. These findings establish a robust framework for advancing intelligent, data-driven SHM systems, with direct applications in real-world infrastructure. The proposed methodology demonstrates significant potential to mitigate economic costs and safety risks associated with structural failures under impact loading.

Keywords: Impact identification, Impact localization, Framed structures, Deep

[*]Corresponding author: Department of Civil Engineering, Shahid Bahonar University of Kerman, Kerman, Iran
[†]E-mail address: saeed.shojaee@uk.ac.ir (S. Shojaee)

learning, CNN, CNN-LSTM, Bayesian algorithms, Time domain.

Received: 5 April 2025; Accepted: 8 May 2025

1. INTRODUCTION

Rapid urbanization and the increasing complexity of modern infrastructure have made Structural Health Monitoring (SHM) essential for ensuring the safety, functionality, and longevity of civil engineering structures. Among the various threats to structural integrity, sudden impact events such as seismic shocks, vessel collisions with bridge piers, or dynamic loadings on cranes pose a particular challenge because they can induce localized damage that may propagate into progressive failures if not detected promptly [1]. The absence of immediate impact detection can result in significant safety risks, catastrophic damage, and considerable economic losses.

While aerospace engineering has focused on impact detection in lightweight composite structures and energy-absorbing plates and shells, civil engineering primarily deals with large-scale frame structures made of concrete and steel, which demand more complex analyses. In frame structures, impact detection is vital not only for health monitoring but also for long-term maintenance. Thus, impact loading plays a crucial role in civil engineering, especially in the design of resilient frame structures, urban infrastructure, and bridges. Continuous monitoring enables early detection of critical impacts, ensuring structural safety without overdesign [2]. Rapid and accurate impact detection is vital for assessing the safety of frame structures, particularly under multiple impacts [3, 4]. While research on composite structures is extensive, [5, 6] studies on frame structures are still limited [7]. Several real-world examples underscore the importance of impact detection in frame structures. The collapse of the Francis Scott Key Bridge in Baltimore, USA, in 2024 [8], caused by a container ship collision, illustrates how the absence of an impact detection system can lead to progressive failure. In industrial settings, cranes are vulnerable to impact-induced cracking during load movements or collisions. Urban buildings may experience damaging interactions under seismic or blast loads [9], while power transmission towers face threats from airborne debris or extreme winds. Even sports arenas are susceptible to structural impacts from dynamic crowd movements or unforeseen incidents.

Numerous studies have investigated impact detection in both the time and frequency domains. Traditional frequency domain techniques, while effective for periodic or stationary loads, often lack the temporal resolution required to distinguish transient or closely spaced impacts and depend heavily on accurate modal models and frequency response functions. Time domain analysis overcomes these limitations by enabling precise localization of an impact in both time and magnitude without reliance on modal parameters. This eliminates model errors and minimizes data processing, reducing the risk of signal corruption and loss of impact information [10, 11]. Advanced time–frequency methods, including wavelet and Hilbert–Huang transforms, further enhance the ability to isolate impact signatures under noisy conditions and to resolve multiple, sequential events [12].

Classical deconvolution approaches extract impact forces by inverting the convolution integral between measured input and output signals. This process effectively separates

[DOI: 10.22068/ijoce.2025.15.2.629]

impact signals from noise, thereby improving the signal-to-noise ratio (SNR). As a result, key impact parameters, such as amplitude, duration, and peak frequency, can be extracted, and the original impact waveform can be reconstructed [13, 14]. However, under high-amplitude loading, nonlinear behavior and noise can cause significant estimation errors, and the transfer matrix may become ill–conditioned, especially in structures with repetitive modes. Additionally, the lack of rotational sensors for angular measurements limits precision, highlighting the need for complementary methods to improve results [7].

Optimization-based methods are widely used in various engineering and scientific problems and are particularly prominent in the field of civil engineering, Kaveh [15, 16]. These methods can seamlessly integrate with signal processing and machine learning techniques to minimize discrepancies between reconstructed and measured signals. Mahdavi et al. [4] combined genetic algorithms with spectral finite element models in the time domain to improve impact localization in large scale frame structures. Doyle [17] proposed a spectral element approach for simply supported beams using genetic optimization. Yan and Zhou [18] developed an inverse identification strategy for composite structures under impact loading. Mahdavi et al. [7] proposed a wavelet-based approach for impact localization in framed structures using a combined genetic and water cycle algorithm. Despite these advances and their effectiveness in addressing ill-conditioning and noise sensitivity, optimization-based schemes remain computationally intensive and often require accurate initial estimates of structural parameters, limiting their applicability in real-time structural health monitoring (SHM) of large-frame structures [1].

Accurate estimation of structural parameters such as mass, damping, and stiffness is crucial for reliable structural analysis, design, and health monitoring, and many studies have been conducted in this area [19]. However, in this study, the structural dynamic parameters were assumed to be unknown and were randomly sampled within predefined ranges to account for their inherent uncertainties and to generate diverse training data, which ultimately complicates impact detection.

Due to the complex relationships in force identification, especially when both the magnitude and location are unknown, advanced machine learning algorithms are essential. These data-driven methods can solve the problem without initial structural parameters, using input-output relationships to develop surrogate models with lower computational costs, providing reliable estimations even under uncertainties and noise [20, 21, 22]. Consequently, these methods can simplify complex force identification processes and demonstrate effective performance in real-world scenarios where data may be limited or noisy. Recent studies have explored machine learning for impact detection, such as Liu et al. [23] proposing support vector regression for estimating unknown loads from heterogeneous responses. Sarego et al. [24] used a large volume of vibration data along with a combination of artificial neural networks (ANNs) and genetic algorithms (GAs) for accurate impact force reconstruction in composite panels. Other methods include artificial neural networks for impact load identification in submerged floating tunnels under collision [25], a wing rib structure [26], and a Gradient Boosting Decision Trees model for impact load identification and localization [27], and machine learning-based predictive models for self-compacting concrete properties [28].

Deep learning methods have become essential tools for analyzing complex data, including structural signals, through automatic modeling of the relationships between system

[DOI: 10.22068/ijoce.2025.15.2.629]

inputs and outputs with high precision and efficiency [29]. These methods excel in capturing complex, nonlinear relationships without the need for manual feature engineering or predefined system models. One widely used architecture is Convolutional Neural Networks (CNNs), which are effective in impact detection, classification, and defect recognition in structures [30]. For example, Tabian et al. [2] used CNNs for detecting and localizing impacts in composite assemblies, achieving over 95% accuracy. Yang et al. [31] introduced a deep dilated CNN (DCNN) for dynamic load identification, offering strong robustness and noise resistance by directly modeling the relationship between vibration responses and external excitations. Similarly, Yu et al. [32] employed a one-dimensional CNN for impact detection and accurate estimation of system parameters in target identification. Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) networks, are suitable for sequential data but face challenges such as high computational demands and the vanishing gradient problem. Hybrid CNN–LSTM models address these issues, combining spatial and temporal feature extraction to enhance performance in time-series data, such as rapid succession impacts [33]. For instance, Zhou et al. [34] employed RNNs with LSTM layers for impact load identification in complex structures, while Yang et al. [35] proposed an RNN method for dynamic load identification in beam structures. Moreover, Maragheh et al. [36] optimized a hybrid CNN–LSTM model to improve multi-impact detection performance.

This paper proposes a novel deep learning-based framework for time-domain detection, localization, and quantification of impact in framed structures. The core innovation of this research lies in resolving impact localization and identification in structures with unknown dynamic parameters. This framework enables precise estimation of impact time, location, and intensity, representing a significant advancement in the field of structural health monitoring. Following a review of the existing technical literature, and with the aim of addressing the identified research needs and gaps, this study will focus on the following areas:

- A two–dimensional CNN classifier for accurate near real-time identification and localization of impacts on structures in the time domain.
- A one–dimensional CNN-LSTM regression model for continuous estimation of impact intensity from fixed-length data windows.
- Bayesian optimization is employed to fine-tune the model parameters, enhancing both accuracy and generalization.
- Evaluation of the effects of noise, insufficient and optimal sensor placement, and the consideration of a sliding window with a fixed time length for data collection on model performance.
- Numerical validation of the developed methods on a two-dimensional shear structure and a two-dimensional pin-jointed camel-back truss structure.

Section 2 details the proposed methodology, including data acquisition, preprocessing procedures, and network architectures. Section 3 introduces the Bayesian optimization strategy for hyperparameter tuning. Section 4 presents the numerical validation studies on a

[DOI: 10.22068/ijoce.2025.15.2.629]

shear frame and a pin-jointed camel-back truss, demonstrating the framework's accuracy and computational efficiency. Finally, Section 5 summarizes the findings and outlines directions for future research.

2. THE PROPOSED STRATEGY FOR DATA GENERATION AND FEATURE SELECTION

2.1 The proposed algorithm for impact identification

Fundamentally, the core of the proposed strategy in this study lies in the three phases: data generation, model training and validation tests. The data were generated through finite element method (FEM) simulations, ensuring accuracy and reliability in the computational modeling process. The model training phase is performed using a CNN model for classification and a CNN-LSTM model for regression, optimized and fine-tuned through Bayesian algorithm for hyperparameter optimization. In other words, this phase includes both the use of deep learning and optimization techniques. The underlying idea is to enhance structural health monitoring by developing an efficient and accurate impact detection system.

The schematic flowchart of the proposed strategy is depicted in Figure 1. This strategy consists of three synthesis stages: Binary classification is employed for impact detection, multi-class classification for impact localization, and regression for impact identification. The following sections provide a detailed, step-by-step explanation of the methodology, which includes data generation, model training approaches, and hyperparameter optimization techniques.

2.2 Data generation overview

In this study, acceleration time-history signals are continuously recorded in real time by sensors embedded at the structure's degrees of freedom. These signals capture the structural response to random impact excitations applied at various locations. These raw acceleration data are collected from the sensors, which may contain environmental noise or sensor malfunctions. After evaluating and confirming that the noise levels are acceptable without distorting the signal's essential information, the data are passed through a preprocessing stage and then fed into deep learning models. This approach ensures that impact detection and identification on the structure are performed with high accuracy.

To enhance the impact detection accuracy and reduce computational complexity under real-time monitoring conditions, a technique so-called the Sliding Window is proposed. In this method, as illustrated in Figure 2, a fixed and short time window is defined, covering a specific duration and containing a predefined number of acceleration signal samples. This window slides continuously along the signal and at each position (i.e., within each 25 samples on the time axis), the data within it is fed into the data analysis model. The purpose of this method is to evaluate the likelihood of impact occurrence in short, consecutive time intervals without needing to process the entire time span or the full signal at once. In fact, using this window brings the monitoring conditions closer to real-time, enabling the system to assess the likelihood of impact in a lighter and faster manner by performing periodic

[DOI: 10.22068/ijoce.2025.15.2.629]

analyses within each window, instead of conducting heavy and detailed analysis for every single second of long data. This periodic analysis not only increases the system's speed but also reduces computational complexity and improves its real-time responsiveness.

For both the classification and regression models, a hold-out validation strategy was employed, where the dataset was split into three subsets: 80% for training, 10% for validation, and 10% for testing. This approach enabled proper monitoring of the models' performance and generalization, which helped to prevent overfitting.

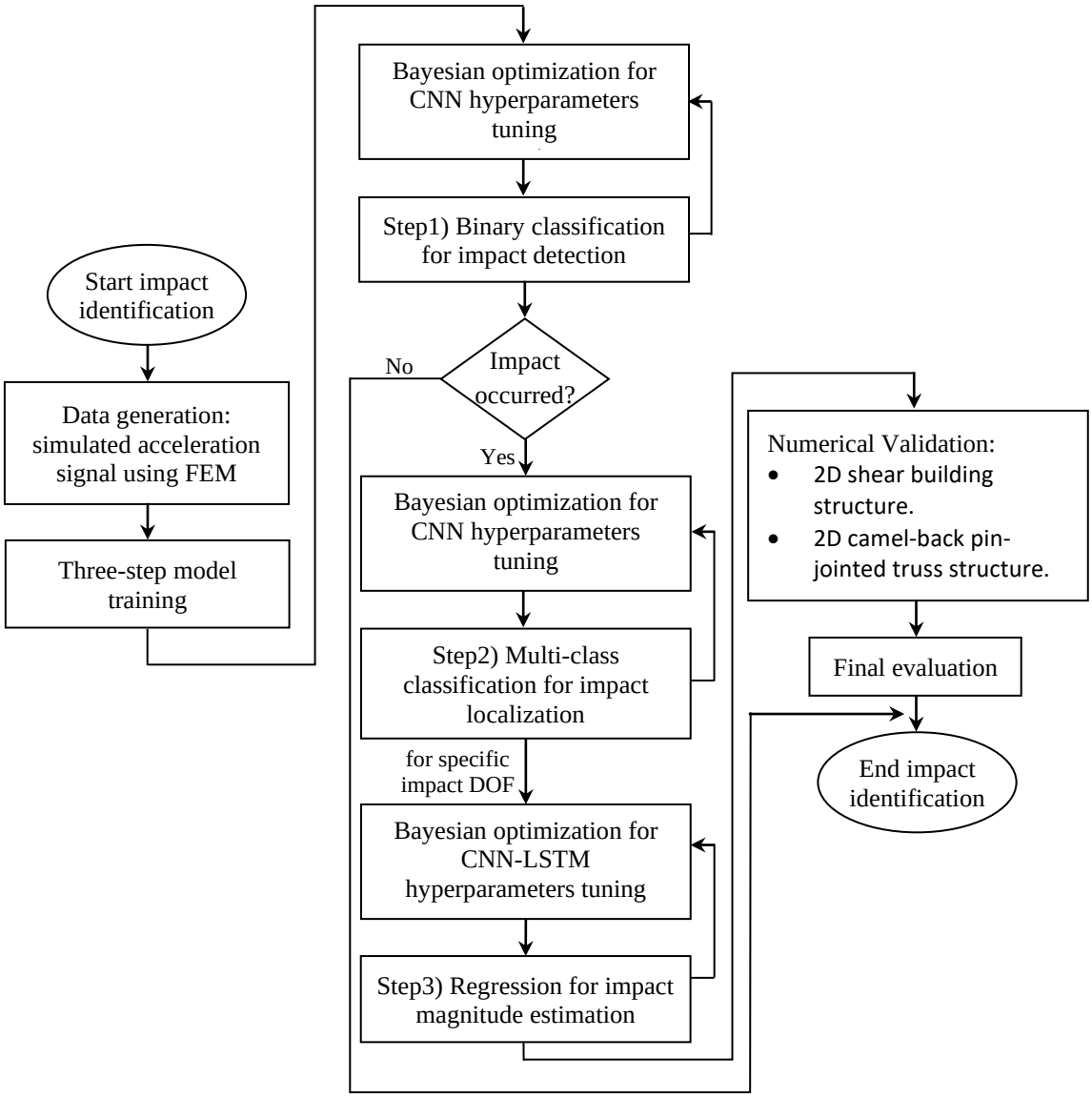


Figure 1: Flowchart of the proposed strategy for impact identification.

[DOI: 10.22068/ijoce.2025.15.2.629]

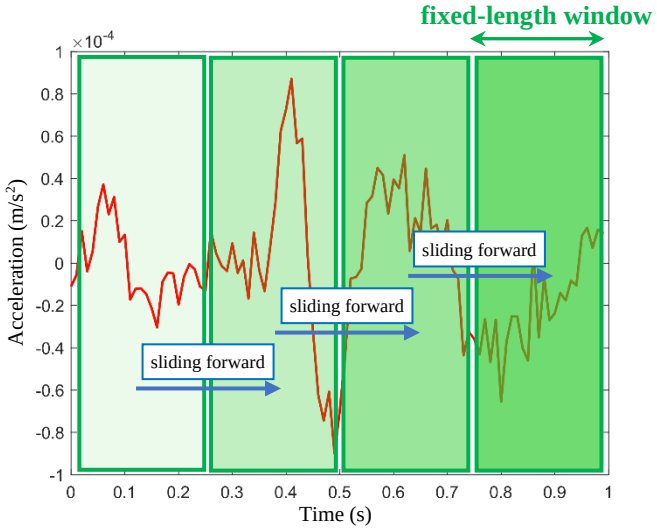


Figure 2: Sliding a fixed-length window over the acceleration signal.

2.3 The proposed scheme for Classification tasks

2.3.1 Data preprocessing

As shown in Figure 1, the first two steps involve the classification of data for both impact occurrence detection and impact localization, performed using deep neural network models. In these steps, the acceleration data are transformed into grayscale image form. This is because in classification tasks, the absolute magnitude of each signal is not critical; rather, the pattern of acceleration variations over time and measured degrees-of-freedom (DOFs) are of primary importance, enabling the detection of sudden changes such as impacts. Accordingly, data normalization, along with the reduction of extreme numerical values, is implemented to enhance the model's stability during the learning process. Figure 3 summarizes the proposed process. As mentioned earlier, the acceleration signal is first extracted, and then a sliding window consisting of specific samples (N) is applied to capture and analyze the variations within that segment. Subsequently, the corresponding grayscale image is generated, and this process is repeated in the same manner until a total of 22,000 images are available for model training. The image dimensions and pixel arrangement are determined as follows:

- The number of rows corresponds to the degrees of freedom equipped with sensors (DOF).
- The number of columns corresponds to the number of samples at each time window (N).
- The image is single-channel due to its black-and-white representation.

[DOI: 10.22068/ijoce.2025.15.2.629]

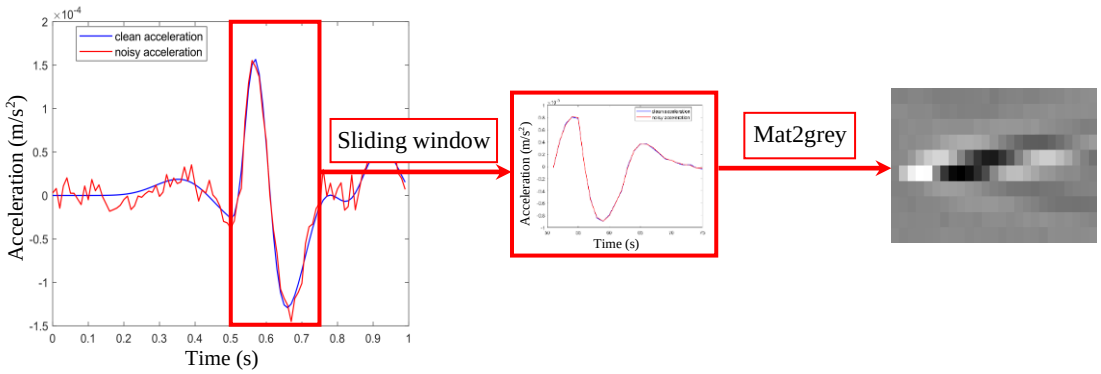


Figure 3: A summary of the acceleration image generation process.

2.3.2 Model architecture

The proposed two-dimensional convolutional neural network (2D-CNN) architecture, illustrated in Figure 4, consists of the following layers:

1. Input layer: grayscale images of dimensions DOF×N.
2. Feature Extraction Blocks:
   - Two 3×3 convolutional layers for extracting important features.
   - Batch normalization layer to enhance learning stability.
   - ReLU activation function is used to introduce non-linearity.
   - 2×2 max pooling with stride 2 to dimensionality reduction.
   - Global Feature Selection: The Global Max Pooling layer selects the maximum value from each channel to identify the most prominent features.
3. Classification Block:
   - Fully connected layers with ReLU activation.
   - Dropout layer for overfitting prevention.
   - Final dense layer with neurons corresponding to the number of classes.
4. Output Layers:
   - Softmax activation for class probability estimation.
   - Classification layer for final classify.

For model training, the Adam optimization algorithm was employed due to its ability to adaptively adjust learning rates during training [37], thereby improving both convergence speed and training stability. The optimal values of the hyperparameters were tuned according to the specific problem characteristics, as detailed in Section 4.

[DOI: 10.22068/ijoce.2025.15.2.629]

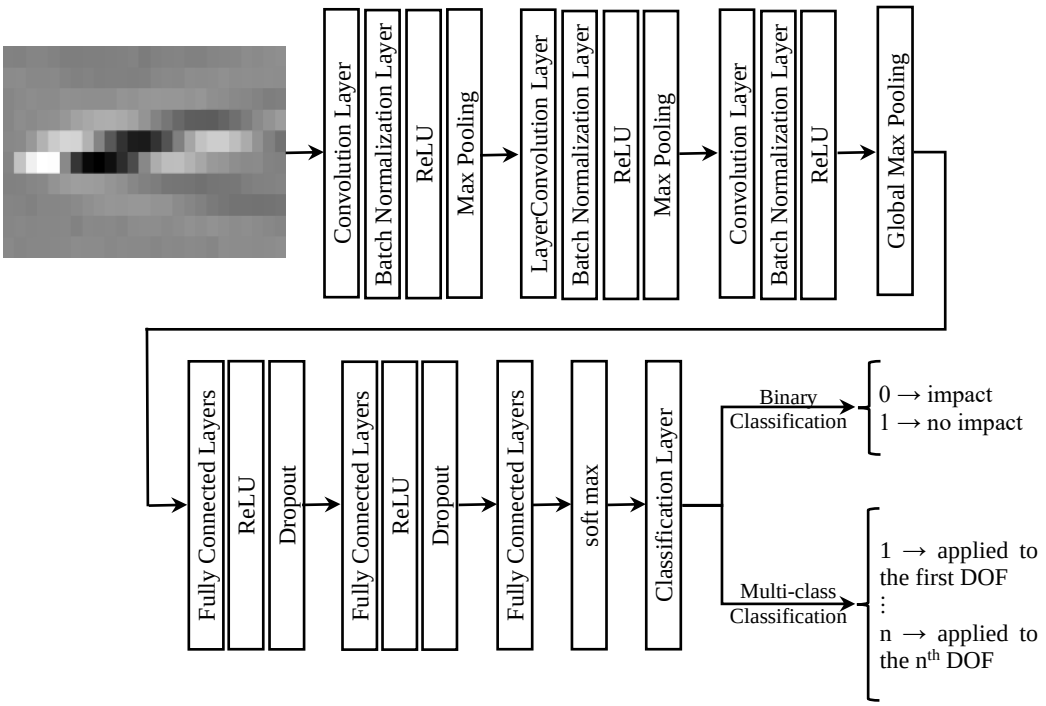


Figure 4: 2D-CNN Classification architecture.

2.4 The proposed scheme for Regression tasks

2.4.1 Data preprocessing

As mentioned earlier, in classification problems involving CNNs, the model primarily focuses on identifying patterns of acceleration changes over time and the gradients between pixels in the images, rather than the absolute intensity values of individual pixels. However, in regression, merely finding the pattern of changes is not sufficient. The model must also identify the absolute intensity of each pixel, as it provides information about its magnitude. Consequently, in the regression part, useful information for the impact magnitude cannot be extracted from images. Instead, it is more effective to directly process the raw acceleration signals by combining three acceleration signals along the longitudinal axis of time, thus constructing cellular acceleration capsules with the following dimensions:

- The number of rows (channels) corresponds to the degrees of freedom equipped with sensors (DOF).
- The number of columns corresponds to three times the number of samples in each time window (3×N).

Figure 5 illustrates a representation of this structure. This process is repeated until 15,000 samples are ultimately obtained for each degree of freedom.

As illustrated in Figure 6, the feature layer corresponding to each acceleration capsule consists of three repeated sets of triangular impact time histories. In the regression part, the

[DOI: 10.22068/ijoce.2025.15.2.629]

goal is to predict these feature layers. The triangular impact is simulated using a triangular function that increases from the start point to the peak and then decreases to the end point. As mentioned in previous sections, four important parameters are the impact start point, the peak point, the end point, and the peak magnitude (emphasizing the impact history). Each feature layer is provided to the model as a cell-structured label with dimensions of 1×(3×N).

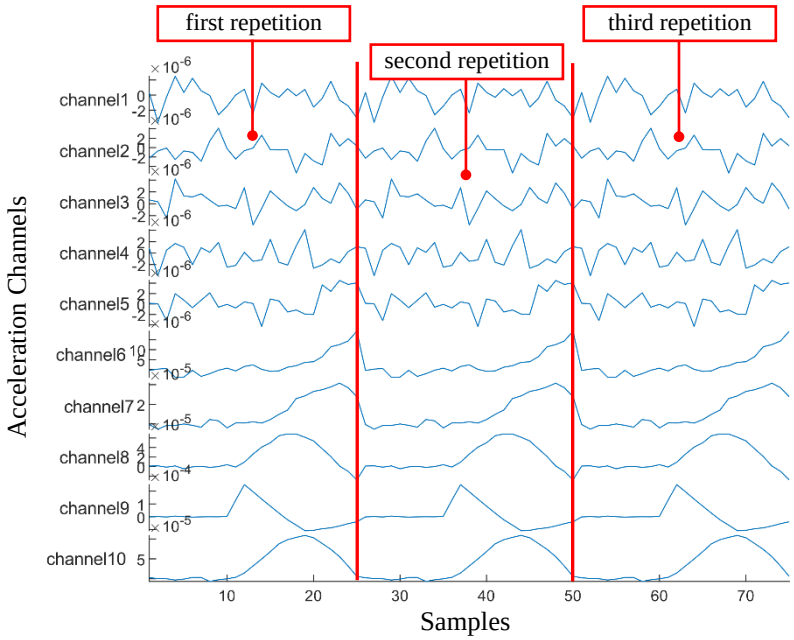


Figure 5: Cellular acceleration capsules with DOF×(3×N) dimensions.

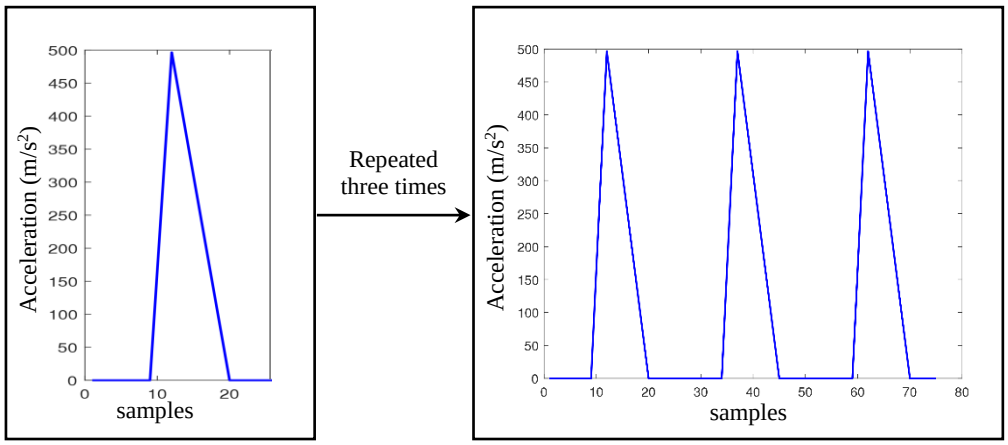


Figure 6: Three impact time histories with rectangular pulses.

[DOI: 10.22068/ijoce.2025.15.2.629]

2.4.2 Model architecture

The proposed one-dimensional convolutional neural network (1D-CNN) architecture, combined with LSTM for regression tasks and illustrated in Figure 7, is composed of the following layers:

1. Input layer: cellular acceleration array with dimensions of DOF×(3×N).
2. Feature Extraction Blocks:
   - Five one-dimensional convolutional layers with varying numbers of filters and progressively increasing filter depth.
   - Causal padding to preserve temporal order.
   - Batch normalization layer to enhance learning stability.
   - ReLU activation function to introduce non-linearity.
3. BILSTM Block:
   - Three Bidirectional Long Short-Term Memory (BILSTM) layers for capturing temporal dependencies from both past and future contexts.
   - Dropout layer for overfitting prevention.
4. Fully connected layers:
   - A fully connected layer with ReLU activation.
   - A Final fully connected layer with a single neuron corresponding to the number of responses (one in this paper).
5. Output Layer:
   - Regression layer for final prediction (impact parameters prediction).

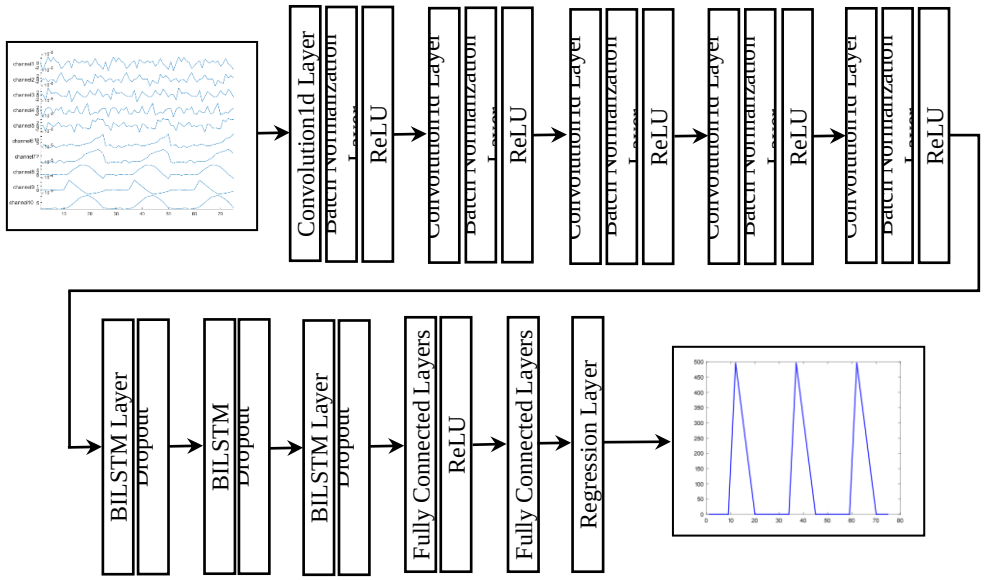


Figure 7: One-dimensional CNN-LSTM regression architecture.

[DOI: 10.22068/ijoce.2025.15.2.629]

Similar to the classification tasks, the Adam optimization algorithm was also employed here for model training and the hyperparameters were also tuned, as detailed in Section 4.

To assess the performance of the regression model, the Root Mean Square Error (RMSE) is employed as the evaluation metric, which represents the average difference between the predicted values and the actual values, as presented in Equation 1.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2} \tag{1}$$

Where y_i is the actual value, $\tilde{y}_i$ is the predicted value, and n is the number of observations. However, since RMSE shares the same unit as the target variable, it is not scale-independent, which can limit its interpretability across different datasets or problem domains. To address this, the relative Root Mean Square Error (rRMSE) is computed by scaling the RMSE against the range of observed values and expressing the result as a percentage, as shown in Equation 2. This normalization enhances the interpretability of the errors and facilitates comparisons across models or datasets.

$$rRMSE = \frac{RMSE}{Range = Max(y_i) - Min(y_i)} \times 100 \tag{2}$$

3. THE APPLICATION OF BAYESIAN OPTIMIZATION FOR HYPERPARAMETER TUNING

3.1 Brief discussion on the Bayesian algorithm

The Bayesian optimization is a powerful technique attempts to minimize a scalar objective function in a bounded domain. This algorithm is particularly useful when dealing with complex, nonlinear, and computationally expensive objective functions. To be noted that, using metaheuristic optimizers such as Particle Swarm Optimizer (PSO) for this type of problems can be highly time-consuming and computationally expensive. In Bayesian optimization problems, the objective function can be either deterministic or stochastic, indicating that it can produce varying outputs when evaluated at the same input point (x). The input vector may consist of continuous (real-valued), discrete (integer-valued), or categorical variables [38].

The origin of this approach can be traced back to the work of American applied mathematician Harold J. Kushner [39], who, while not directly proposing Bayesian optimization, utilized Wiener processes to solve unconstrained one-dimensional optimization problems. Additionally, the probability of improvement maximization criterion was employed to select the next sample. In 1978, the Lithuanian mathematician Jonas Mockus developed a new acquisition function, called expectation of improvement (EI), which is one of the core sampling strategies of Bayesian optimization [40]. Stuckman [41], extended the Kushner's method from a one-dimensional framework to higher dimensions to propose a flexible, efficient, and global applicable optimization method and to demonstrate

[DOI: 10.22068/ijoce.2025.15.2.629]

its effectiveness across both standard and novel types of test problems, especially when the function has many extrema (local minimum or maximum), the function may not be differentiable, and evaluations are computationally expensive. Perttunen and Elder also contributed to global optimization research by employing rank transformations to reduce computational costs and by extending Kushner's univariate probabilistic approach to multivariate problems [42, 43]. After Jones et al. [44] proposed the Efficient Global Optimization (EGO) method, Bayesian optimization gradually gained a special position in engineering fields and attracted widespread attention. Zhang et al. [45] proposed a novel method for mixed-variable Gaussian process using Bayesian optimization to improve accuracy in materials design. This approach demonstrates significant improvements in modeling qualitative parameters compared to existing methods. In recent years, Bayesian optimization has emerged as one of the most practical methods for solving complex and computationally expensive problems, due to its ability to find optimal solutions with limited evaluations. In 2018, Frazier [46] provides a comprehensive review of its foundational concepts, advanced extensions, and software tools. In 2021, Mathern et al. [47] applied multi-objective constrained Bayesian optimization to the design of reinforced concrete beams, outperforming genetic algorithms in convergence and solution quality. In 2022, Røstum et al. [48] demonstrated faster convergence in designing post-tensioned bridges. In 2024, Gautam [49] utilized explainable Gaussian process-assisted multi-objective optimization for sustainable soil stabilization. Recently, in 2025, Li et al. [50] present a novel experimental design framework that integrates active multi-criteria sampling with Bayesian optimization to efficiently identify optimal process parameters while significantly improving constraint estimation accuracy in advanced manufacturing.

3.2 Variable and cost function development

In this study, Bayesian optimization is used to find the optimal values of the deep learning model's hyperparameters with the fewest number of evaluations, without the need to blindly test all possible combinations. The step-by-step procedure is shown in Table 1.

Table 1: Bayesian optimization algorithm for hyperparameters tuning.

Bayesian optimization algorithm
1. **Defining the Optimization Problem** The goal of optimization is to minimize the model's error on the validation data and identify the optimal hyperparameter values for the model. • For classification, the loss value is minimized. • For regression, the RMSE (Root Mean Square Error) is minimized. 2. **Defining Optimizable Variables** Parameters to be optimized include network architecture parameters and training parameters are listed in Tables 1.1 and 1.2. For each variable, its type (e.g., integer, real) and acceptable range of variation must also be specified. **3. Defining the Objective Function** The objective function takes the optimizable variables as input, builds and trains the model using those values, and finally returns a numerical output. This output is then

[DOI: 10.22068/ijoce.2025.15.2.629]

evaluated by the Bayesian optimizer (as the error), which aims to minimize it through multiple evaluations.

4. **Running the Bayesian Optimizer**

The search process is carried out within the parameter space, and the best combination of parameters for optimizing the objective function is identified.

5. **Extracting the Best Parameters Corresponding to the Optimal Objective Value**

Using this information, the optimal combination of parameters for the model can be selected and used as input for the final training of the model.

Table 1.1: Classification hyperparameters.

Parameter	Description	Range
Number of filters	Determines how many different features are detected per layer.	2^4 - 2^{10}
filter sizes	Size of the filter window used to scan the input.	3 - 7
Dropout rate	Prevents overfitting by randomly turning off connections.	0 - 1
Number of Units Neurons	Controls the output size of the FC layer.	2^4 - 2^{10}
Number of Hidden Units	Determines the memory capacity of the LSTM cells.	2^4 - 2^{10}
Initial Learning Rate	Controls how fast the model learns.	0.0001-0.1
Mini-Batch Size	Number of training samples used in one forward-backward pass.	2^4 - 2^6

Table 1.2: Regression hyperparameters

Parameter	Description	Range
Number of filters	Determines how many different features are detected per layer.	2^4 - 2^{10}
filter sizes	Size of the filter window used to scan the input.	3 - 19
Dropout rate	Prevents overfitting by randomly turning off connections.	0 - 1
Number of Units Neurons	Controls the output size of the FC layer.	2^4 - 2^{10}
Number of Hidden Units	Determines the memory capacity of the LSTM cells.	2^4 - 2^{10}
Initial Learning Rate	Controls how fast the model learns.	0.0001-0.1
Mini-Batch Size	Number of training samples used in one forward-backward pass.	2^4 - 2^6

The schematic flowchart for the implementation of Bayesian optimization is shown in Figure 8.

[DOI: 10.22068/ijoce.2025.15.2.629]

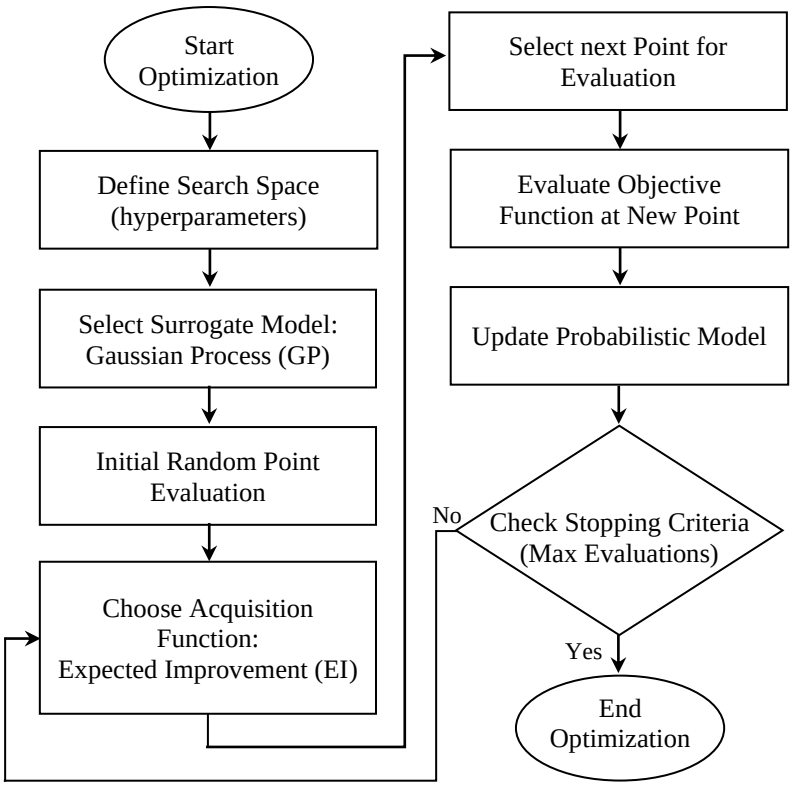


Figure 8: Bayesian optimization flowchart.

4. NUMERICAL VALIDATION STUDY

4.1 Overview of the proposed approach and data simulation

In this section, the performance of the proposed model is evaluated in the process of impact detection on two structural systems: a 2D shear frame and a 2D camel-back pin-jointed truss. The validation process aims to evaluate the model's capability across the three aforementioned steps.

In this study, numerical simulations were performed using the finite element method (FEM) to generate synthetic acceleration datasets that closely emulate real-world sensor outputs. The simulations were designed to capture the dynamic behavior of the structure by applying uncertainty factors in the mass and stiffness of the structure to better approximate real-world conditions, adding noise to replicate realistic measurement conditions, and applying diverse randomized loading scenarios to ensure a comprehensive representation of dynamic responses. Subsequently, the generated data were preprocessed into two distinct formats: grayscale images for training classification models and time-series acceleration arrays for regression model training. The following sections provide detailed explanations of these topics.

[DOI: 10.22068/ijoce.2025.15.2.629]

Step 1) Formation of mass, stiffness, and damping matrices

- Assemble the global mass and stiffness matrices using the known theoretical mass and stiffness properties of each member.
- Calculation of the damping matrix using Rayleigh damping method, expressed as a linear combination of the mass (M) and stiffness (K) matrices.

Consequently, in each simulation iteration, the mass and stiffness values are modified by applying different uncertainty factors. These variations are introduced to cover a reasonable range of structural conditions. The upper and lower bounds of these random uncertainty factors are provided in Table 2.

Table 2: The uncertainty factors applied to the mass and stiffness.

	Lower bound	Upper bound
Stiffness factors	0.8	1.2
Mass factors	0.7	6

Step 2) Application of random impact loads to the structural system

The structure is subjected to random impacts with a triangular waveform, under load scenarios within a specific range of magnitudes. The impact characteristics - including application time, location (degree of freedom of impact), and force magnitude - are all determined randomly during the simulation process. Figure 9 presents the triangular impact parameters including start sample (SS), peak sample (PS), end sample (ES), and peak magnitude (F_p).

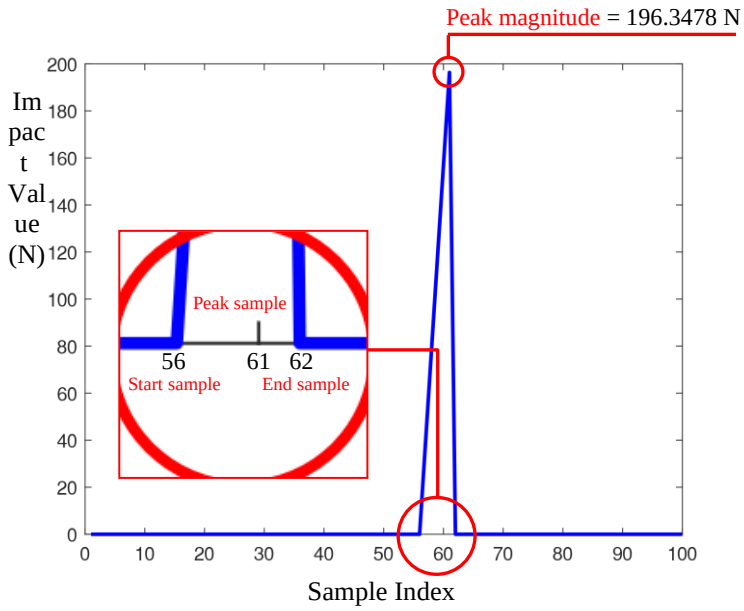


Figure 9: An example of a random triangular impact load applied to the structure over a total duration of 1 second and a sampling rate of 100 Hz.

[DOI: 10.22068/ijoce.2025.15.2.629]

Step 3) Acceleration response extraction using the Newmark-β integration method

The dynamic response of the structure to the applied impacts was calculated using the Newmark-β time integration method.

In this study, our analysis assumes linear behavior, therefore effectively utilizing the linearized constant average acceleration method and parameters were set to $\beta = 0.25$ and $\gamma = 0.5$, ensuring stability and accuracy for linear systems. As a result, the simulated accelerations are obtained in this step. Figure 10 shows the simulated acceleration before and after applying noise.

In real-world, sensor measurements inherently contain noise from various sources, including environmental conditions, measurement errors, and system imperfections. To approximate these realistic conditions, 10% Gaussian white noise was added to the acceleration outputs obtained from the Newmark integration method.

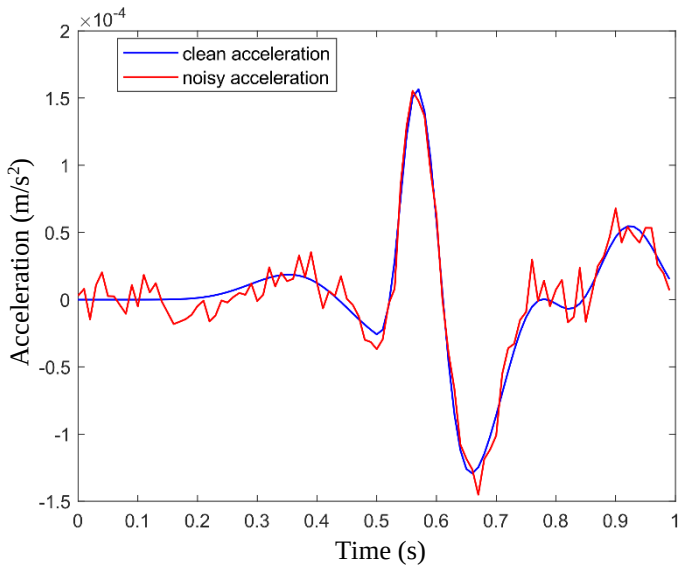
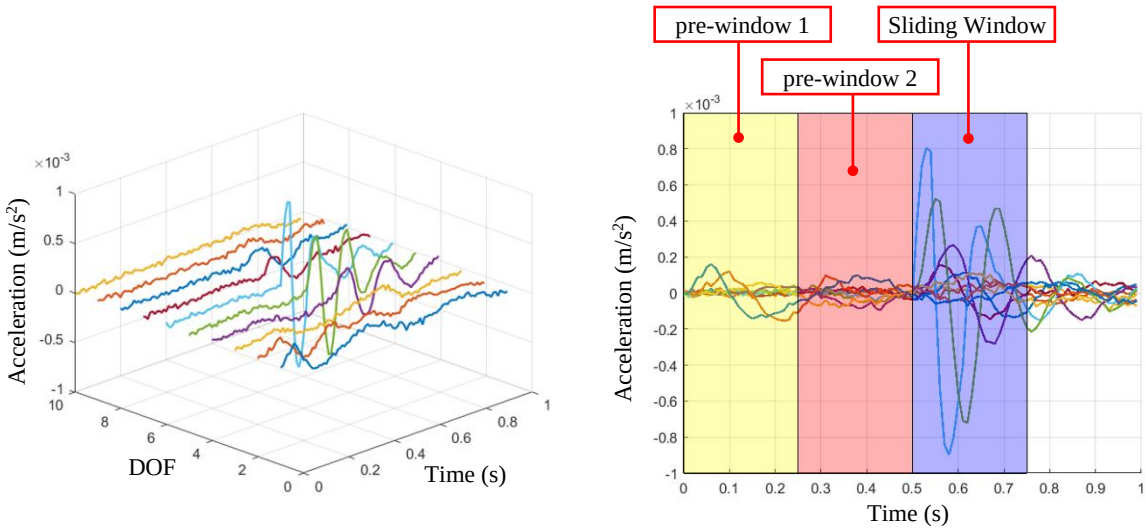


Figure 10: Simulated acceleration before and after applying 10 percent noise.

In addition, we aim to develop a dynamic real-time detection method for identifying impacts applied to the structure, capable of continuous and real-time processing. To bring the method closer to these conditions, we use a sliding window with a fixed length of 25 samples and analyze the acceleration signal through this window, similar to what is shown in Figure 11b. Figure 11a shows a three-dimensional representation of the acceleration outputs corresponding to 10 degrees of freedom (complete sensor measurement for each DOF) and Figure 11b presents a time-acceleration view, illustrating the sliding window along with the two previous windows. The window length of 25 samples corresponds to 0.25 seconds at a 100 Hz sampling rate. This time interval is short enough to statistically ensure that only one impact occurs within the window. By assuming a single impact, the processing conditions effectively approximate real-time operation. It is worth to emphasize here that, the length of the sliding window is selected based on the sampling rate and is subsequently adjusted according to the specific problem.

[DOI: 10.22068/ijoce.2025.15.2.629]



(a) (b)

Figure 11: Simulated acceleration, (a) three-dimensional representation of the acceleration, (b) time-acceleration view, illustrating the sliding window and two previous windows.

In order to develop a more accurate analytical framework and ensure sufficient comprehensiveness for this case, the dynamic effects of impacts occurring in pre-windows must be systematically evaluated. Because these impacts generate residual vibrations within the main window, and their effects are dynamically significant and cannot be neglected. The probability of an impact not occurring during the pre-windows is higher than the probability of its occurrence. Accordingly, the impact probabilities for each pre-window are set as follows:

- First pre-window: 30%
- Second pre-window: 10%

In the main window, the probability of an impact occurring is assumed to be the same for each degree of freedom.

Incomplete sensor measurement:

A significant challenge in Structural Health Monitoring (SHM), particularly for impact and damage detection applications, is the limited availability of sensors due to the cost constraints, installation challenges, and technical limitations. Consequently, only a few sensors are usually placed in critical locations with the highest probability of damage or impact. To address this limitation, researchers focus on Optimal Sensor Placement (OSP) methodologies [3], which aim to maximize the information obtained about a structure's dynamic behavior while minimizing the number of sensors required. Following data acquisition from this optimized sensor network, structural responses at unmeasured locations are subsequently estimated through numerical interpolation techniques.

[DOI: 10.22068/ijoce.2025.15.2.629]

4.2 A 10-DOFs two-dimensional shear building

4.2.1 Problem definition and data generation

The behavior of a two-dimensional shear frame structure with 10 degrees of freedom, as illustrated in Figure 12, is analyzed and simulated in this study. Various impact scenarios with different intensities (randomly ranging from 100 to 1000 Newtons) and random locations were applied to the structure, and the resulting acceleration signals were recorded. These acquired time-history signals were then utilized as input features for deep learning models to identify impact characteristics.

To accurately observe and analyze the maximum natural frequencies of the structure, it is essential to adjust the sampling rate in accordance with variations in the dynamic properties. Given that, due to uncertainties in mass and stiffness values, the maximum frequency content of the structure can vary within a range of 20 to 70 Hz, the sampling rate must be selected appropriately to prevent aliasing. Aliasing refers to the phenomenon where high-frequency signals are misrepresented as lower frequencies due to insufficient sampling. According to the Nyquist criterion, the sampling rate must be at least twice the maximum frequency present in the signal. Accordingly, a sampling rate between 40 and 140 Hz is adopted to ensure that the maximum natural frequencies of the structure are accurately captured and analyzed. Consequently, selecting a sampling rate of 200 Hz is a reliable option. However, it is important to note that using this sampling rate will significantly increase the computational data volume. Therefore, to reduce this data volume, impact simulations are performed at a sampling rate of 200 Hz, and the resulting data are then downsampled through an interpolation process. This approach ensures that the frequency content corresponding to the 200 Hz sampling rate is accurately extracted, while the data length is effectively reduced to 100 Hz.

A window length of 25 samples, corresponding to 0.25 seconds, is selected at a 100 Hz sampling rate. This time interval is short enough to statistically ensure that only one impact occurs within the window. By assuming a single impact, the processing conditions effectively approximate real-time operation.

For this shear frame structure with ten degrees of freedom, ten acceleration signals were extracted from ten embedded sensors (full sensor measurement). These signals, generated from the simulated accelerations, have dimensions of 10 by 25, corresponding to a sliding window length of 25.

4.2.2 Model training

Step 1) Impact detection - binary classification

At this step, grayscale images of the recorded acceleration signals under various impact scenarios, as shown in Figure 13, are fed into the 2D CNN binary classification model for impact detection. The acceleration data obtained from the impact simulations on the structural system are carefully processed, and various features are extracted through the model. This process enables the model to identify patterns related to impact occurrence from other data under normal conditions, classifying them into two categories: no impact (0) and impacted (1).

[DOI: 10.22068/ijoce.2025.15.2.629]

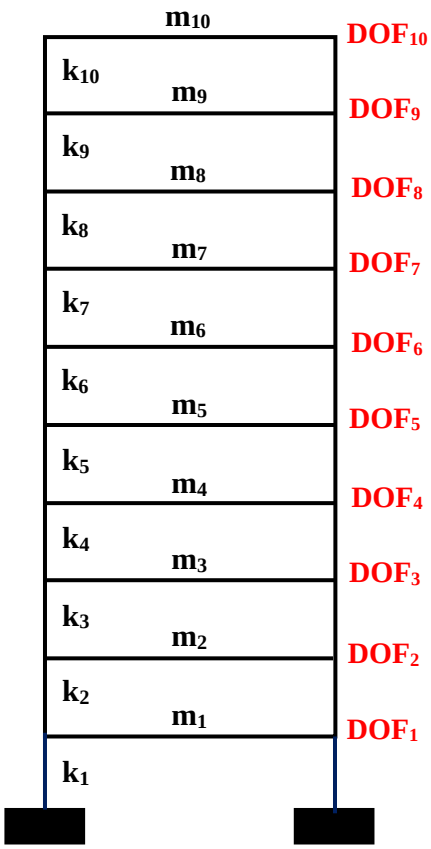


Figure 12: A 10-DOF two-dimensional shear building.

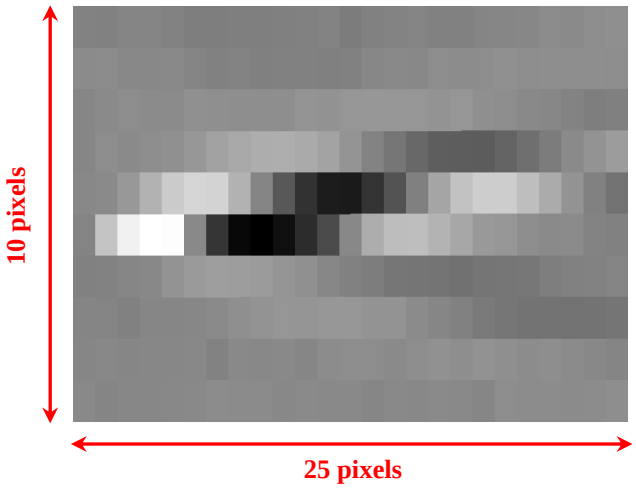


Figure 13: Grayscale image of the recorded acceleration signals under an impact scenarios for a 10-DOF 2D shear building.

[DOI: 10.22068/ijoce.2025.15.2.629]

The model's hyperparameters were initially tuned using Bayesian optimization. To ensure the reliability of the results, the Bayesian optimization process was independently executed five times with random initializations. In each run, the key parameters were tuned accordingly. Ultimately, the optimal solution was selected based on achieving the highest reproducibility in accuracy as well as its cost history is illustrated in Figure 14, and the corresponding optimal parameters are listed in Table 3. Model training utilized the Adam optimizer and kept constant for all the considered cases.

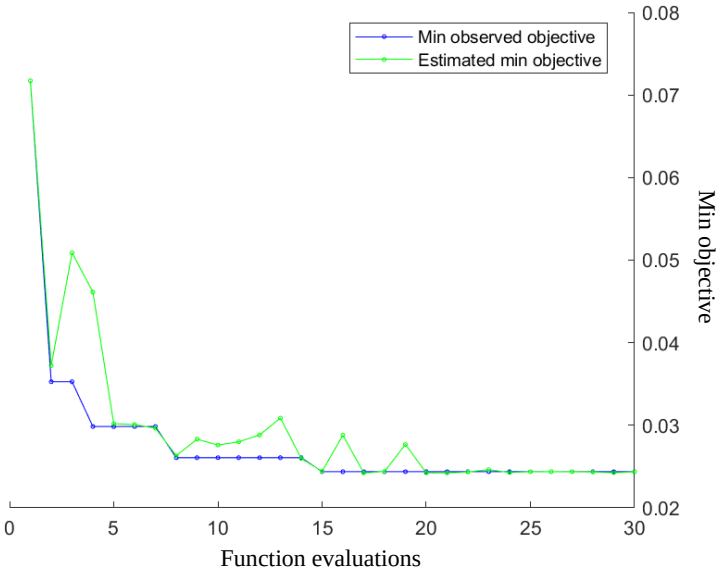


Figure 14: Bayesian optimization cost history.

Table 3: The hyper parameters of CNN binary and multi-class classification models.

Convolutional 2D Layers	Layers	Filter sizes	Number of filters
	Layer1	3 × 3	16
	Layer2	3 × 3	32
	Layer3	3 × 3	128
FC Layers	Layers	Number of units	Dropout rate
	Layer1	128	0.17502
	Layer2	64	0.17502
Learning rate		0.00667	
Mini-batch size		64	

The model training results indicate an accuracy of 99.31% on the test dataset. The training progress plot, including the variation of accuracy over learning epochs, the confusion matrix, and the Receiver Operating Characteristic (ROC) curve which is used to evaluate the model's performance and identify classification errors, are all illustrated in Figure 15 The Area Under the ROC Curve (AUC) for both labels was reported as 0.9991,

[DOI: 10.22068/ijoce.2025.15.2.629]

indicating the model's ability to discriminate between impacted and non-impacted states in binary classification. Therefore, the proposed model accurately distinguishes between impact and non-impact states, successfully progressing to the next stage.

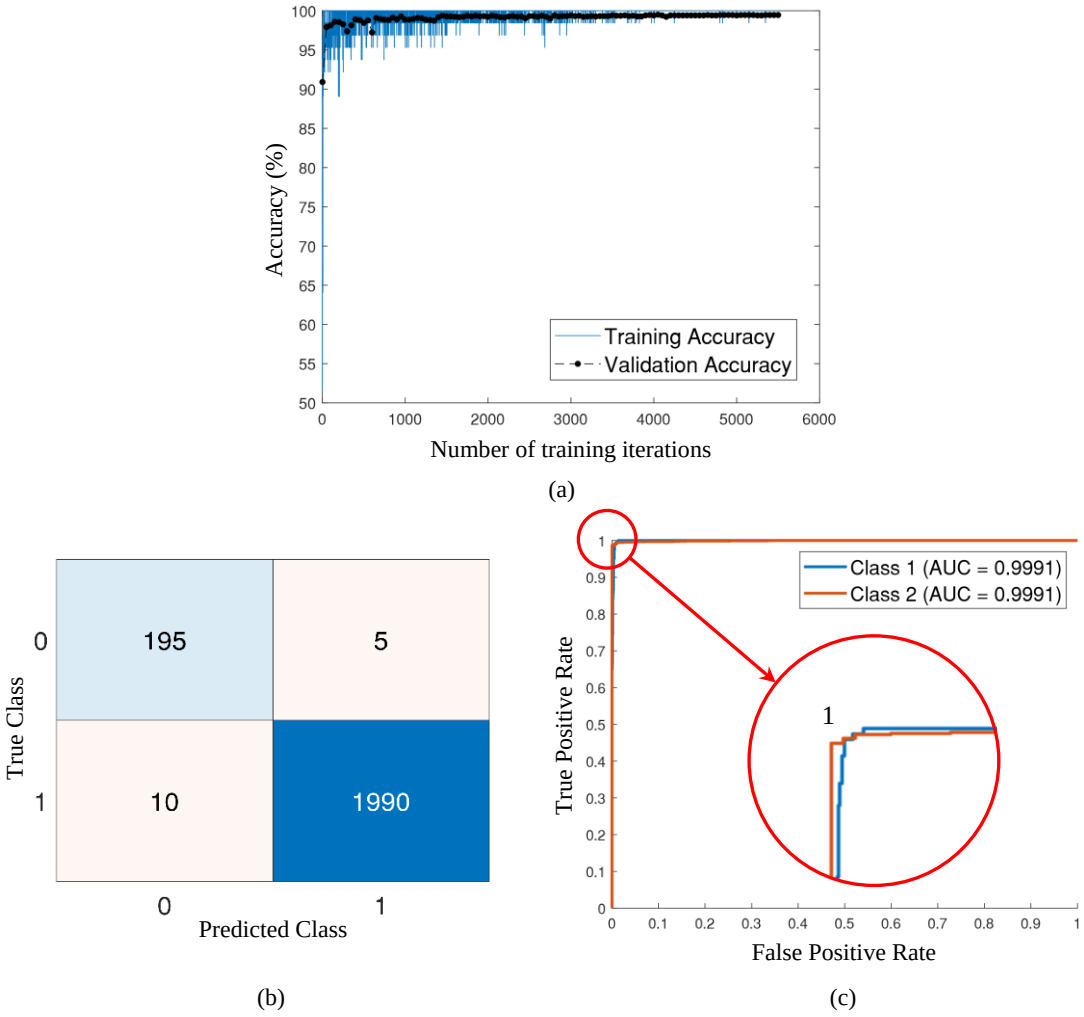


Figure 15: CNN Binary classification results. (a) The training progress accuracy, (b) the confusion matrix, (c) the ROC

Step 2) Impact localization - multi class classification

At this step, after confirming the occurrence of an impact, the exact location of the impact is identified using a 2D-CNN multi-class classification model. In fact, the model analyzes the patterns in the variations of the acceleration image, which are caused by impacts at different points on the structure. Subsequently, the accuracy of the impact localization results is evaluated by comparing the predicted locations with the corresponding actual impact positions.

[DOI: 10.22068/ijoce.2025.15.2.629]

The hyperparameters are the same as those used in the binary model. The results shown in Figure 16 demonstrate the model's strong capability in distinguishing between different impact locations, achieving an average localization accuracy of 99.6% on the test dataset. For a more precise evaluation, a confusion matrix was constructed for the different impact locations, revealing that the true positive rate (TPR) exceeded 99% for all points on the structure. Finally, the model achieved near-perfect AUC scores (close to 1) for all classes, indicating excellent class discrimination in the multi-class classification. These findings confirm the high reliability of the proposed method in distinguishing between different impact locations.

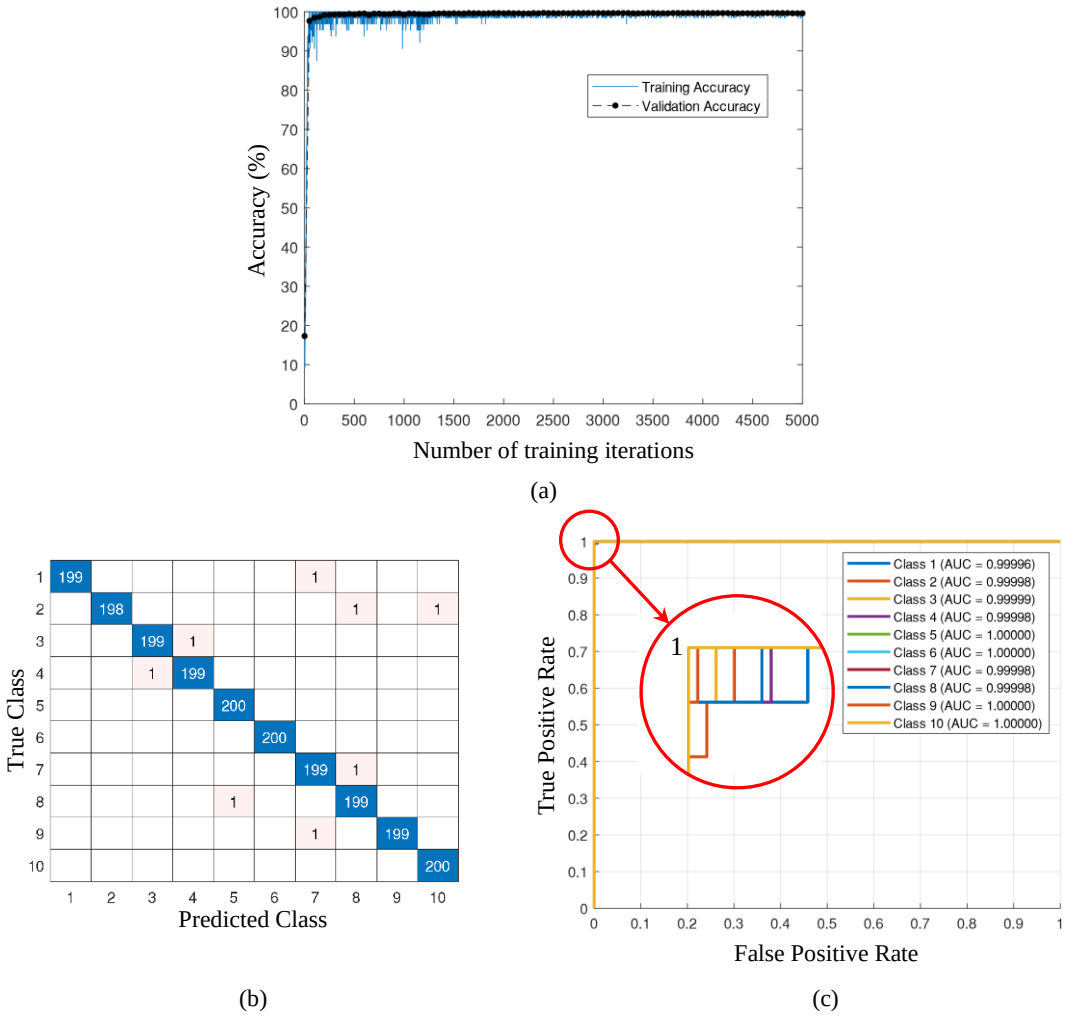


Figure 16: CNN Multi-class classification results. (a) The training progress accuracy, (b) the confusion matrix, and (c) the ROC curve.

[DOI: 10.22068/ijoce.2025.15.2.629]

Step 3) Impact magnitude measurement-regression

Finally, after confirming the occurrence of the impact and identifying its location, the exact magnitude of the force (in Newtons) is estimated using regression-based methods. As described in Section 2, acceleration signals resulting from varying impact forces were processed using a CNN-LSTM deep learning architecture with continuous output, optimized using Root Mean Squared Error (RMSE) as the loss function for regression. In this assessment, it is assumed that the impact occurred at DOF 9, and the intensity of the applied impact is aimed to be predicted. The Bayesian optimization was executed three times for this network, and the most stable outcome was selected as the final configuration. Its cost history is shown in Figure 17, and the corresponding optimized hyperparameters are listed in Table 4.

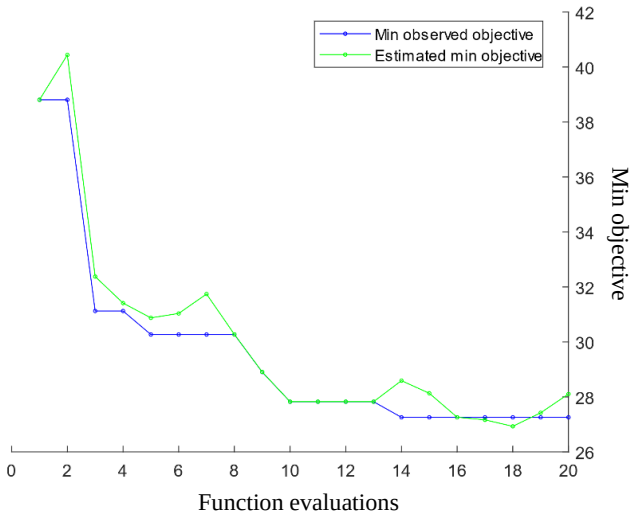


Figure 17: Bayesian optimization cost history.

Table 4: The hyperparameters of the CNN-LSTM regression model.

Convolutional 1D Layers	Layers	Filter sizes	Number of filters
	Layer1	6	32
	Layer2	7	64
	Layer3	11	128
	Layer4	15	256
	Layer5	16	512
BILSTM Layers	Layers	Number of units	Dropout rate
	Layer1	256	0.5
	Layer2	128	0.5
	Layer3	64	0.5
FC Layers	Layers		Number of units
	1		32
Learning rate		0.0017	
Mini-batch size		32	

[DOI: 10.22068/ijoce.2025.15.2.629]

The results shown in Figure 18 indicate that the model has successfully learned the relationship between the structure's dynamic response and the impact force with acceptable accuracy. The scatter plot of the actual force values versus the predicted force values also demonstrates a high correlation between the model's results and the actual data. The error distribution, illustrated by the histogram, further supports the overall reliability and accuracy of the model's predictions across the dataset. Therefore, the results of this step demonstrate that employing the proposed CNN-LSTM-based models based on acceleration signals is an effective method for estimating the magnitude of impact force applied to shear structures.

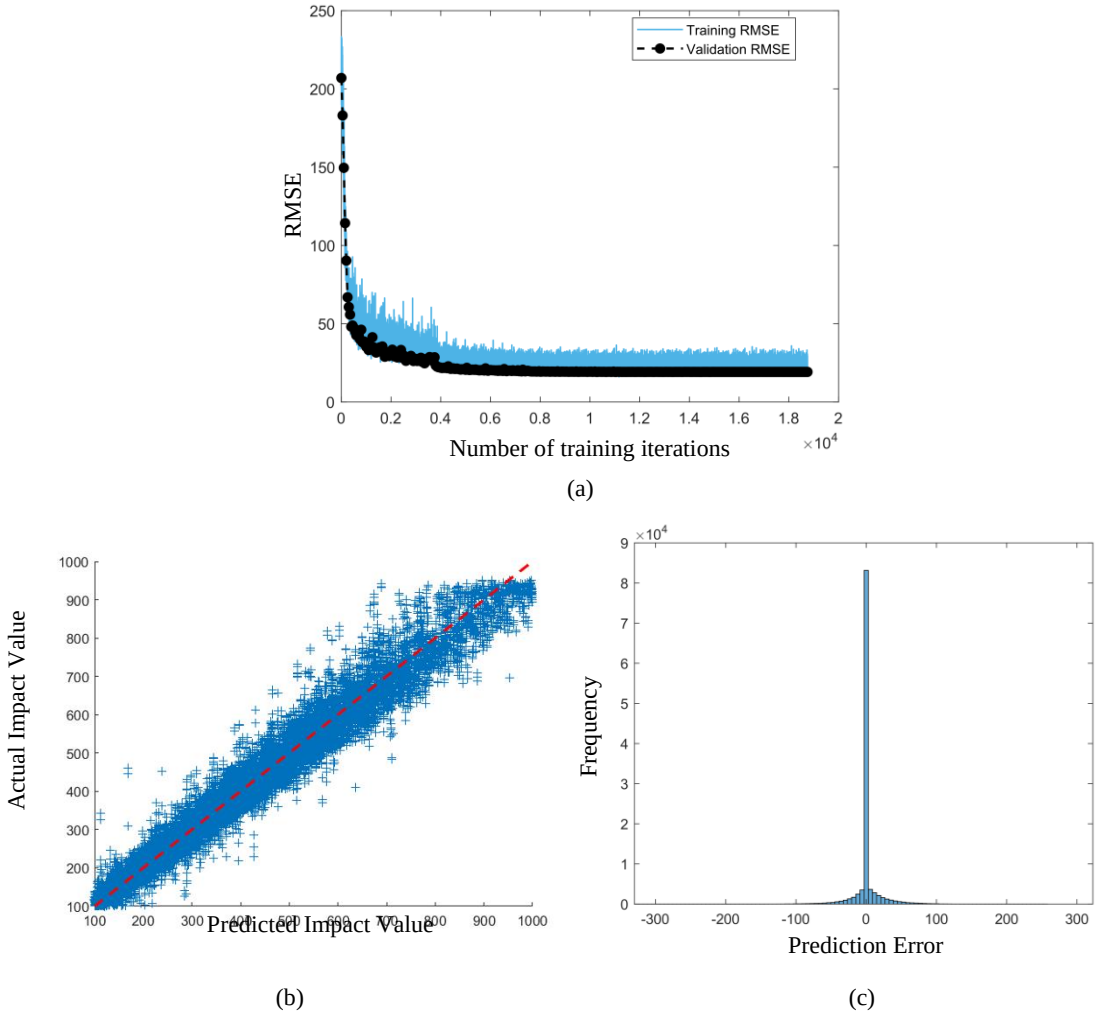


Figure 18: CNN-LSTM Regression model results. (a) The training progress RMSE, (b) the scatter plot, and (c) the histogram of RMSE values.

As illustrated in Figure 19, the prediction with the least error is selected from three different impact time history estimates to yield the most accurate final regression output for the impact intensity.

[DOI: 10.22068/ijoce.2025.15.2.629]

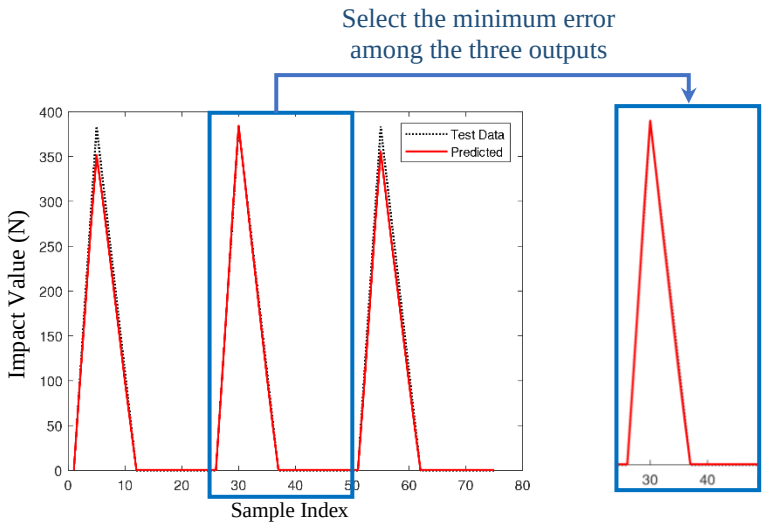


Figure 19: Select the minimum error among the three impact time history estimates.

The Root Mean Squared Error (RMSE) of the model is reported to be 16.2679 N. The relative Root Mean Square Error (rRMSE) is calculated by scaling the RMSE against the range of observed values as defined in Equations (3) and (4), which indicates that the model has an error of only 1.8%.

$$Range = Max(y) - Min(y) = 1000N - 100N = 900N \tag{3}$$

$$rRMSE = \frac{RMSE}{Range} \times 100 = \frac{16.2679N}{900N} \times 100 = 1.8075\% \tag{4}$$

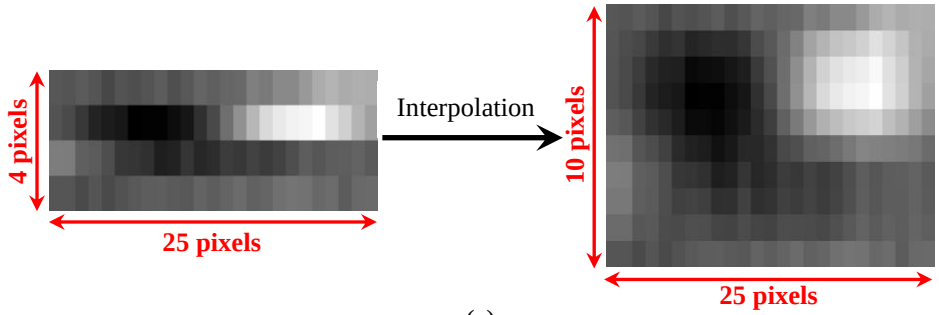
4.2.3 Incomplete sensor measurement - 4 sensors

In a shear structure with 10 degrees of freedom, due to the limited number of available sensors, only four accelerometers are placed at DOFs 1, 4, 7, and 10, corresponding to the first, last, and two intermediate positions. This limitation reduces the dimensionality of the input acceleration signal.

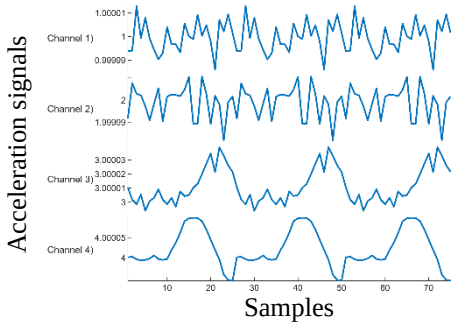
For classification, these reduced acceleration signals are then reconstructed to full dimensionality through interpolation, as demonstrated in Figure 20a. In contrast, for the regression, the reduced-dimensional signal data are directly fed into the model without further modification, as illustrated in Figure 20b, and the acceleration input was repeated three times, as previously mentioned.

The results presented in Table 5 indicate a decrease in model accuracy across the different stages of the process. Nonetheless, despite considerable uncertainties, unknown physical parameters of the structure such as mass and stiffness, and the absence of a sensor at the impact location, the model demonstrates sufficient capability in achieving the objectives of this study by successfully completing the three-stage procedure and accurately identifying various impact-related parameters.

[DOI: 10.22068/ijoce.2025.15.2.629]



(a)



(b)

Figure 20: Incomplete sensor measurement data. (a) For classification model. (b) For regression model.

The results presented in Table 5 indicate a decrease in model accuracy across the different stages of the process. Nonetheless, despite considerable uncertainties, unknown physical parameters of the structure, such as mass and stiffness, and the absence of a sensor at the impact location, the model demonstrates sufficient capability in achieving the objectives of this study by successfully completing the three-stage procedure and accurately identifying various impact-related parameters.

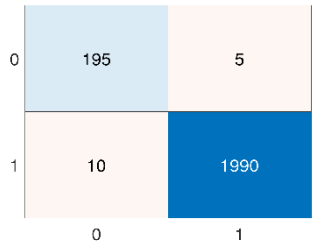
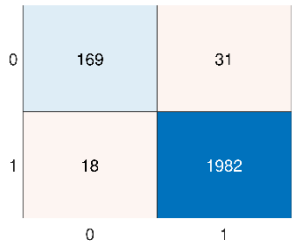
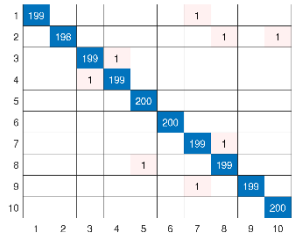
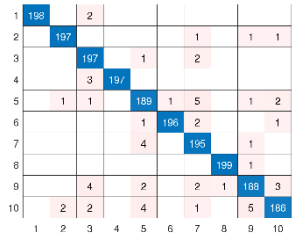
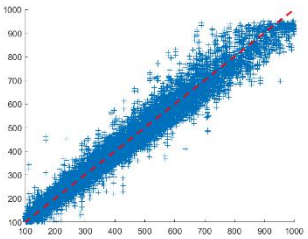
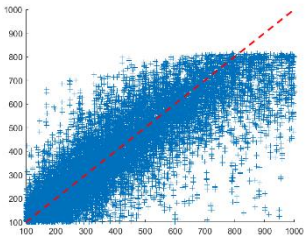
4.3 A 24-DOF two-dimensional steel pin-jointed camelback truss

4.3.1 Problem definition and data generation

The behavior of a two-dimensional camelback truss structure with 24 degrees of freedom, as illustrated in Figure 21, is analyzed and simulated in this section. The structure consists of 12 nodes, each with two degrees of freedom (horizontal and vertical translations), leading to a total of 24 degrees of freedom. Node 1 is fully restrained, and node 12 is restrained vertically. With three degrees of freedom constrained by boundary conditions, 21 active degrees of freedom remain. In a similar manner to the shear structure, various impact scenarios with different intensities (randomly ranging from 500 to 5000 Newtons) and random locations were applied to the structure. The resulting acceleration signals were recorded as time-history data and subsequently used as input features for deep learning models to detect and characterize the impacts.

[DOI: 10.22068/ijoce.2025.15.2.629]

Table 5: Model Performance under Incomplete Sensor Measurement.

Three-Step Process	Full-field	Incomplete Measurement
	99.31%	97.77%
Step1: Binary Classification (Accuracy %)		
	99.6%	97.1%
Step2: Multi-class Classification (Accuracy %)		
	1.8%	6.52%
Step3: Regression (Error %)		

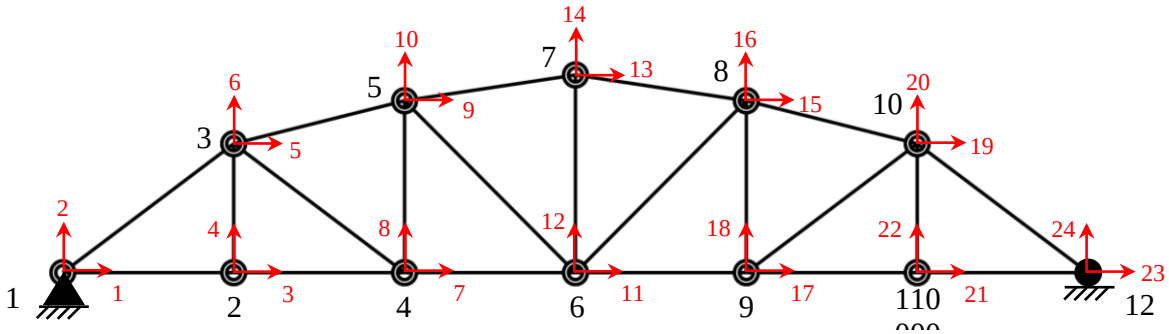


Figure 21: A 24-DOF two-dimensional steel pin-jointed camelback truss.

To accurately observe and analyze the maximum natural frequencies of truss structures, a similar approach to that adopted for shear structures has been considered. Taking into account the uncertainties in mass and stiffness properties, it is expected that the maximum

[DOI: 10.22068/ijoce.2025.15.2.629]

frequency content lies within the range of 160 to 500 Hz. To prevent aliasing, and according to the Nyquist criterion, which states that the sampling rate must be at least twice the maximum frequency of the signal, the sampling frequency should be selected between 320 and 1000 Hz. Based on this, choosing a sampling rate of 1000 Hz is regarded as a conservative and reliable decision. However, based on the experimental point of view, in truss structures, the very high-frequency components are primarily associated with local vibrations of the truss members and have a limited contribution to the global dynamic behavior of the structure. Furthermore, these high-frequency modes are often recognized as noise in dynamic analyses and therefore do not hold significant importance. Therefore, the analysis focuses on frequencies up to 200 Hz. To capture this frequency content, simulations are initially performed at a sampling rate of 400 Hz and subsequently downsampled to 100 Hz using interpolation, reducing computational load and noise sensitivity. Consequently, the sliding window length is again set to 25 samples.

For this camelback truss structure with 21 degrees of freedom, acceleration signals were collected from all 21 embedded sensors, representing a full measurement setup. These acceleration signals were organized into a 21 × 25 matrix, with each row corresponding to the time-series response of a sensor, and the window length was set to 25 time steps.

4.3.2 Model training

Step 1) Impact detection - binary classification

The grayscale images of the recorded acceleration signals under various impact scenarios, as shown in Figure 22, are input into the 2D CNN binary classification model for impact detection. The model classifies these images into two categories: no impact (0) and impact (1).

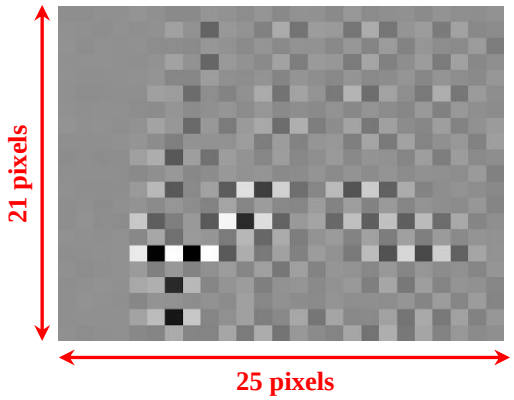


Figure 22: Grayscale image of the recorded acceleration signals under an impact scenario for a 21-DOF 2D steel pin-jointed camelback truss.

Similar to the previous structure, the model's hyperparameters were optimized using Bayesian optimization to achieve the highest accuracy. To ensure result reliability, the process was independently executed five times with random initializations. The optimal solution was selected based on maximum reproducibility of accuracy, and its cost history is shown in Figure 23, with final values presented in Table 6. Training was conducted with the

[DOI: 10.22068/ijoce.2025.15.2.629]

Adam optimizer over 20 epochs, with the learning rate reduced by 20% every 5 epochs, transitioning from an exploration to an exploitation phase.

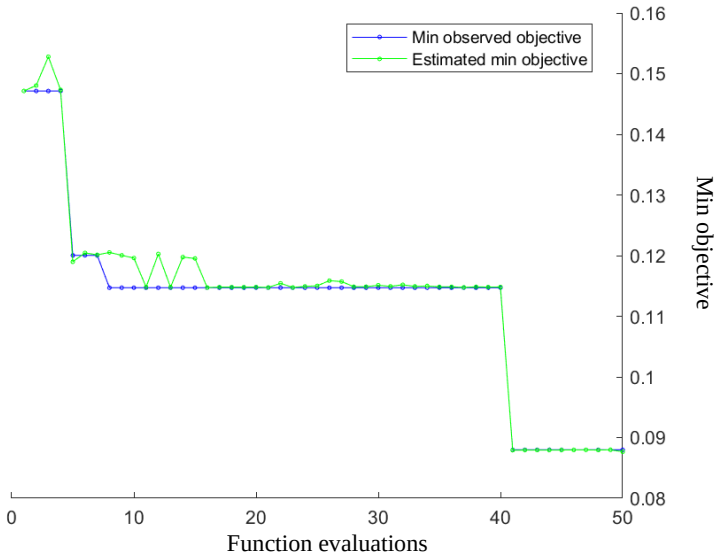


Figure 23: Bayesian optimization cost history for 2D truss classification.

Table 6: The hyperparameters of 2D truss CNN binary and multi-class classification models.

	Layers	Filter sizes	Number of filters
Convolutional 2D Layers	Layer1	5 × 5	32
	Layer2	5 × 5	64
	Layer3	5 × 5	128
	Layers	Number of units	Dropout rate
FC Layers	Layer1	512	0.4101
	Layer2	256	0.4101
Learning rate		0.0013	
Mini-batch size		32	

The model training results indicate an accuracy of 99.5% on the test dataset. The training progress plot, including the variation of accuracy over learning epochs, the confusion matrix, and the ROC curve, is illustrated in Figure 24. The Area Under the ROC Curve (AUC) for both labels was reported to be approximately 0.999, indicating the model's ability to discriminate between impacted and non-impacted states in binary classification.

Therefore, the proposed model accurately distinguishes between impact and non-impact states, successfully progressing to the next stage.

[DOI: 10.22068/ijoce.2025.15.2.629]

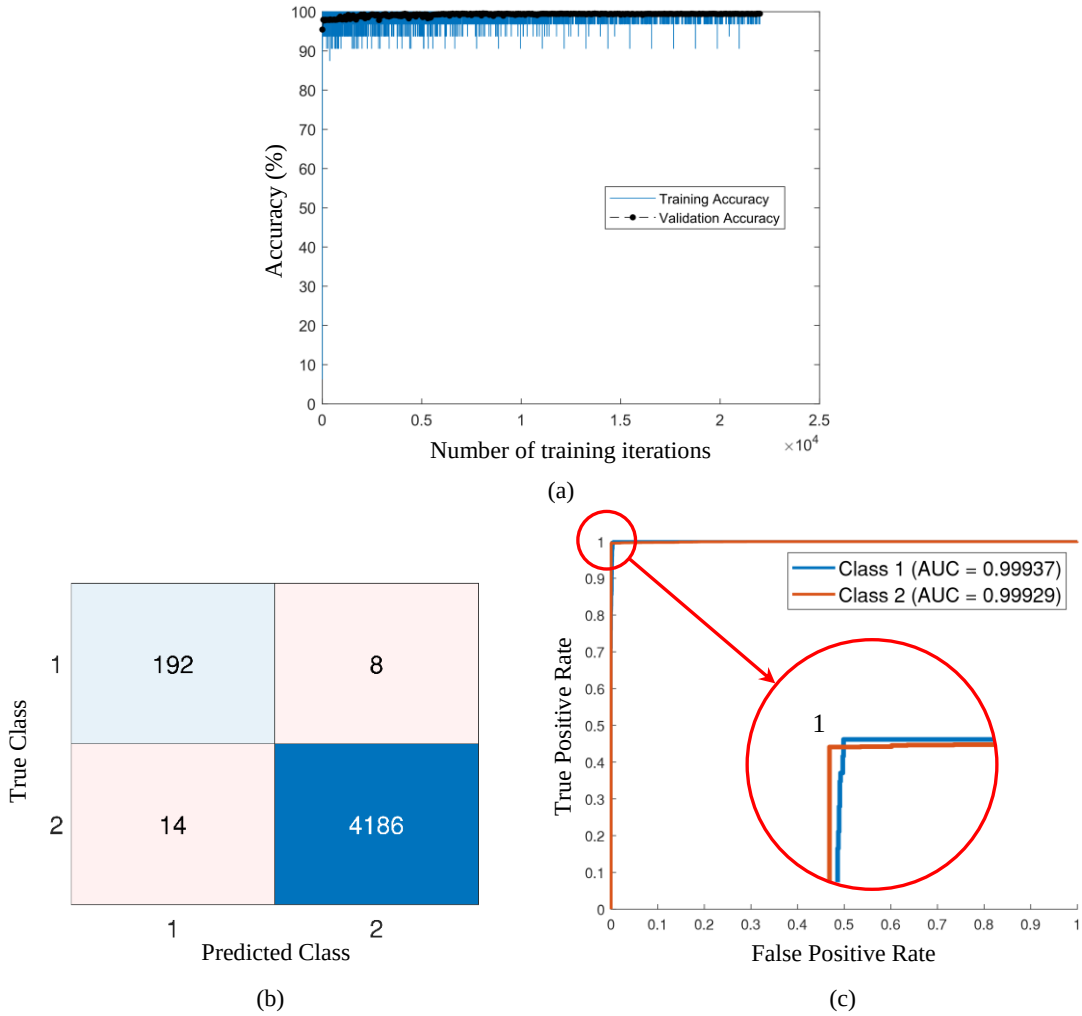


Figure 24: CNN Binary classification results. (a) The training progress accuracy, (b) the confusion matrix, and (c) the ROC curve.

Step 2) Impact localization – multi-class classification

At this step, after confirming the occurrence of an impact, the exact location of the impact is identified using a 2D-CNN multi-class classification model. The hyperparameters are the same as those used in the binary model. The results shown in Figure 25 demonstrate the model's strong capability in distinguishing between different impact locations, achieving an average localization accuracy of 98.5952% on the test dataset. For a more precise evaluation, a confusion matrix was constructed for the different impact locations, revealing that the true positive rate (TPR) exceeded 99% for all points on the structure. For a more precise evaluation, a confusion matrix was constructed for the different impact locations, revealing that the true positive rate exceeded 97% for most points on the structure. And the model achieved near-perfect AUC scores (close to 1) for all classes, indicating excellent class discrimination in the multi-class classification. These findings confirm the high reliability of the proposed method in distinguishing between different impact locations.

[DOI: 10.22068/ijoce.2025.15.2.629]

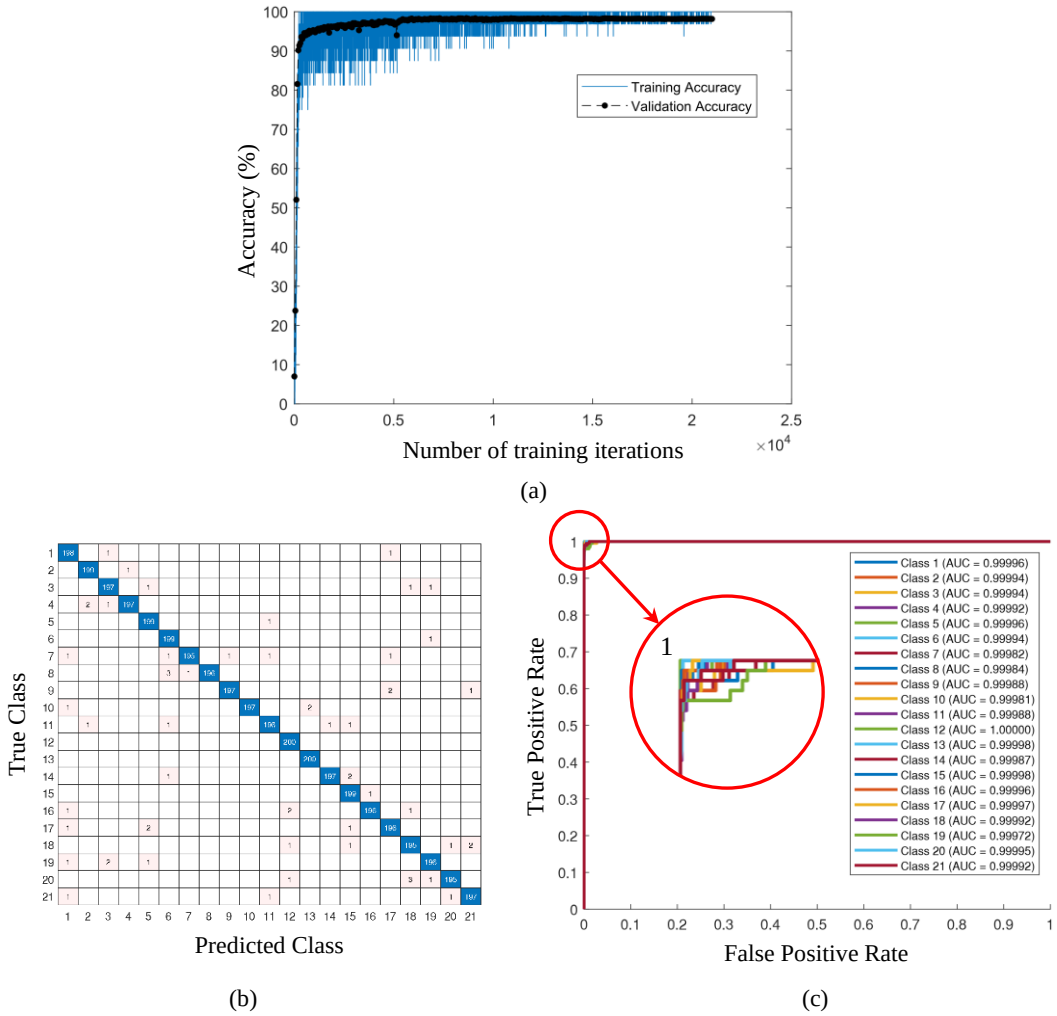


Figure 25: CNN Multi-class classification results. (a) The training progress accuracy, (b) the confusion matrix, and (c) the ROC curve.

Step 3) Impact magnitude measurement-regression

Finally, similar to the previous structure, the exact magnitude of the impact force (in Newtons) was estimated using regression-based methods. As outlined in Section 2, the acceleration signals corresponding to different impact forces, assuming the impact is applied to degree of freedom 10, were fed into a CNN-LSTM model with continuous output, trained using RMSE as the loss function. Bayesian optimization was performed three times, and the most stable result, as shown in Figure 26, was selected, with the corresponding hyperparameters listed in Table 7.

[DOI: 10.22068/ijoce.2025.15.2.629]

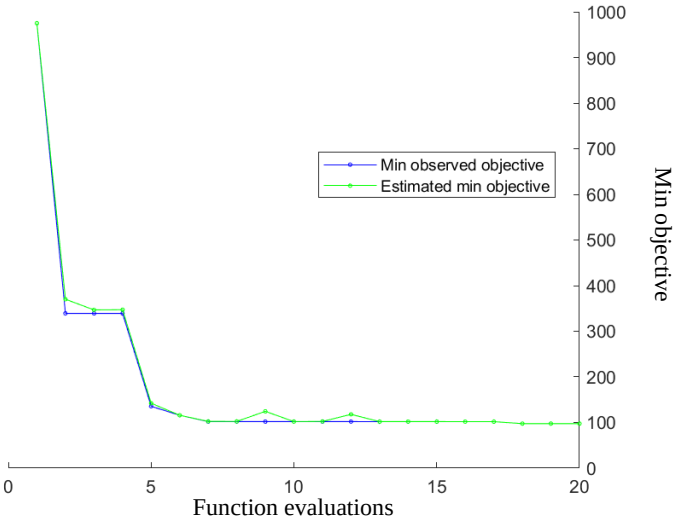


Figure 26: Bayesian optimization cost history.

Table 7: The hyperparameters of the CNN-LSTM regression model.

Convolutional 1D Layers	Layers	Filter sizes	Number of filters
	Layer1	5	16
	Layer2	8	32
	Layer3	9	64
	Layer4	15	128
	Layer5	17	256
BILSTM Layers	Layers	Number of units	Dropout rate
	Layer1	256	0.5763
	Layer2	128	0.5763
	Layer3	64	0.5763
FC Layers	Layers	Number of units	
	1	64	
Learning rate	0.0039		
Mini-batch size	32		

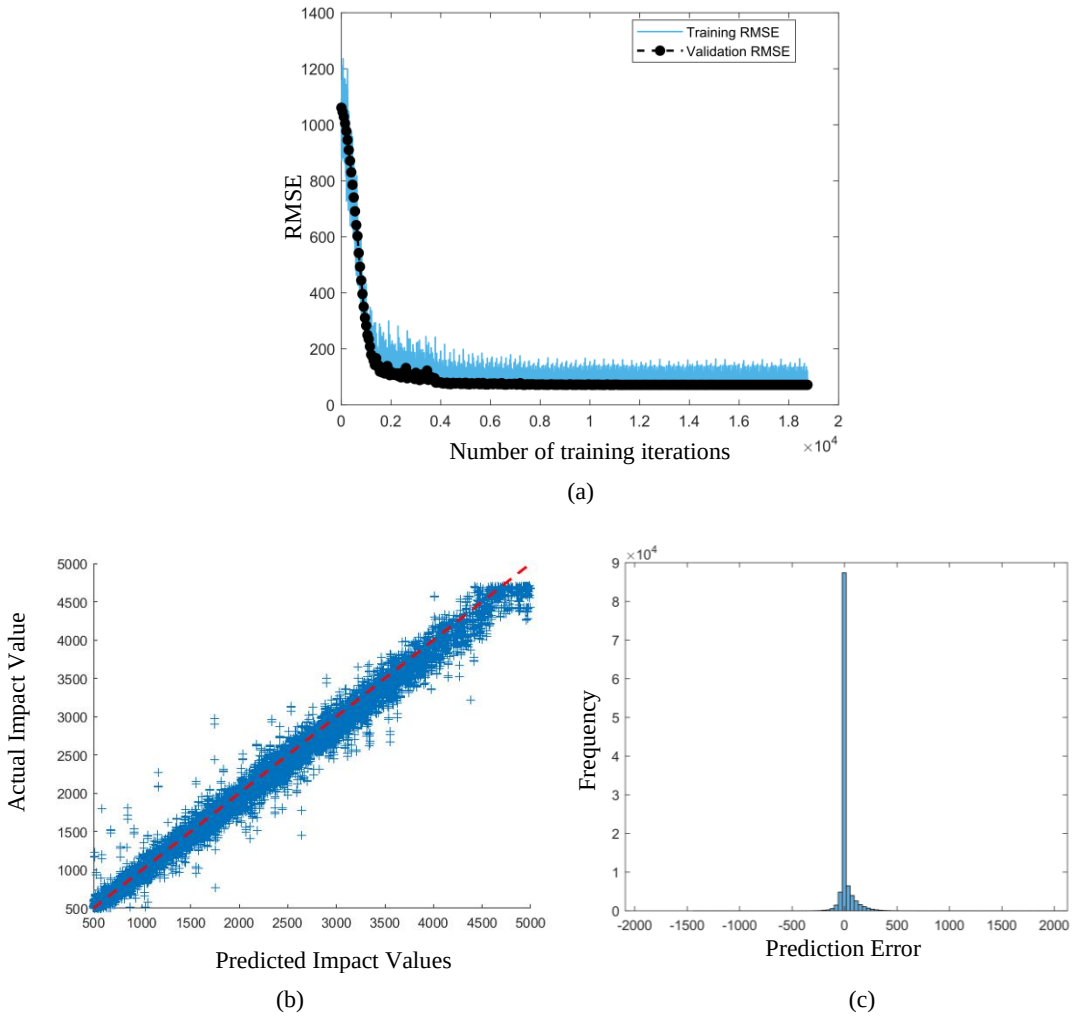
To ensure the most accurate estimation of impact intensity, the prediction with the lowest error is selected from three different impact time history estimates, similar to the approach used for the truss structure. The Root Mean Squared Error (RMSE) of the model is reported to be 56.3866 N. The relative Root Mean Square Error (rRMSE) is calculated by scaling the RMSE against the range of observed values as defined in Equations (5) and (6), which indicates that the model has an error of only 1.25%.

$$Range = Max(y) - Min(y) = 5000N - 500N = 4500N \quad (5)$$

[DOI: 10.22068/ijoce.2025.15.2.629]

$$rRMSE = \frac{RMSE}{Range} \times 100 = \frac{56.3866N}{4500N} \times 100 = 1.25\% \quad (6)$$

The results shown in Figure 27 indicate that the model has successfully learned the relationship between the structure's dynamic response and the impact force with acceptable accuracy. The scatter plot of the actual force values versus the predicted force values also demonstrates a high correlation between the model's results and the actual data. The error distribution, illustrated by the histogram, further supports the overall reliability and accuracy of the model's predictions across the dataset. Consequently, the results of this step demonstrate that employing deep learning models based on acceleration signals is an effective method for estimating the magnitude of impact force applied to 2D truss structures.



(a)

(b)

(c)

Figure 27: CNN-LSTM Regression model results. (a) The training progress RMSE, (b) the scatter plot, (c) The histogram of RMSE values.

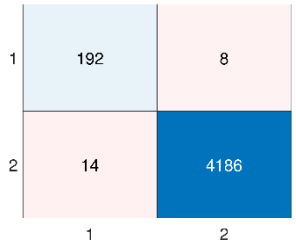
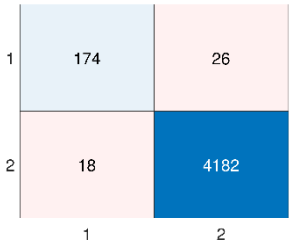
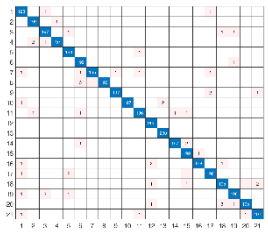
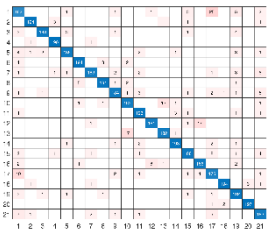
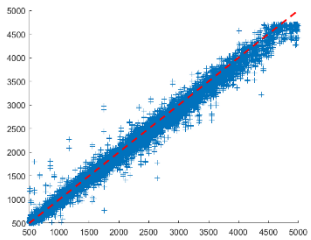
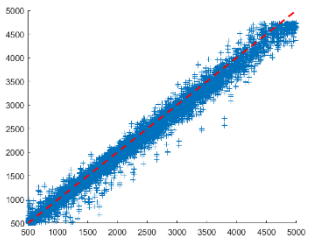
[DOI: 10.22068/ijoce.2025.15.2.629]

4.3.3 Incomplete sensor measurement - 8 sensors

In a 2D truss structure with 24 degrees of freedom, due to the limited number of available sensors, only 8 accelerometers are placed at DOFs 5, 6, 8, 13, 14, 18, 19, and 20 (as shown DOFs in Figure 21).

The results presented in Table 8 indicate a decrease in model accuracy. Nonetheless, the model demonstrates sufficient capability in meeting the objectives of the present study by successfully completing the three-step procedure and accurately identifying various impact-related parameters.

Table 8: Model performance under Incomplete sensor measurement.

Three-Step Process	Full-field	Incomplete Measurement
	99.5%	99%
Step1: Binary Classification (Accuracy %)		
	98.5952%	93.5%
Step2: Multi-class Classification (Accuracy %)		
	1.25%	1.43%
Step3: Regression (Error %)		

5. CONCLUSION

In this study, a three-stage deep learning framework was developed for structural impact analysis using acceleration signals. The first two stages employed the same CNN-based

[DOI: 10.22068/ijoce.2025.15.2.629]

classification model: Stage 1 was dedicated to detecting the occurrence of an impact, while Stage 2 aimed to localize the impact on the structure. In Stage 3, a CNN-LSTM model was utilized to perform regression on raw time-series acceleration data to estimate the impact magnitude. To optimize the performance of the models in all three stages, Bayesian optimization was implemented for hyperparameter tuning. Moreover, physical parameters such as mass and stiffness were considered uncertain and assumed to be unknown in this study, which reflects real-world structural variability. The proposed models demonstrated high classification accuracy and promising regression performance, validating the effectiveness of the approach. By integrating image-based and sequence-based representations, the framework provides a robust and reliable solution for real-time structural health monitoring.

While the proposed approach shows promising results, further research is needed to validate and enhance its effectiveness in broader contexts:

- **Nonlinear behavior modeling:** Addressing the nonlinear dynamic response of structures under high-energy impacts to improve the model's fidelity in realistic scenarios.
- **Experimental validation:** Conducting physical experiments to validate the proposed framework and assess its practical feasibility.
- **Frequency-domain analysis:** Investigating structural responses in the frequency domain to complement time-domain analysis and potentially capture additional features relevant to impact characterization.
- **Generalization across structural types:** Extending the framework to a wider range of structural forms and materials to ensure robustness under diverse physical conditions.

[DOI: 10.22068/ijoce.2025.15.2.629]

REFERENCES

1. Shahabi P, Mahdavi SH, Hamzehei-Javaran S, Shojaee S. An online time-domain strategy for multiple impact identification of framed structures using neural networks and ensemble learning algorithms. *Eng Struct*. 2025;**335**:120353.
2. Tabian I, Fu H, Sharif Khodaei Z. A convolutional neural network for impact detection and characterization of complex composite structures. *Sensors*. 2019;**19**(22):4933.
3. Mahdavi SH, Azimbeik K. A modified genetic algorithm strategy for optimal sensor exciter placement capable of time domain structural identification. *Int J Optim Civ Eng*. 2022;**12**(4):517-43.
4. Yu Z, Mahdavi SH, Xu C. Time-domain spectral element method for impact identification of frame structures using enhanced GAs. *KSCE J Civ Eng*. 2019;**23**(2):678-90.
5. Khoo SY, Ismail Z, Kong KK, Ong ZC, Noroozi S, Chong WT, Rahman AGA. Impact force identification with pseudo-inverse method on a lightweight structure for under-

determined, even-determined and over-determined cases. *Int J Impact Eng*. 2014;**63**:52-62.
6. Sanchez N, Meruane V, Ortiz-Bernardin A. A novel impact identification algorithm based on a linear approximation with maximum entropy. *Smart Mater Struct*. 2016;**25**(9):095050.
7. Mahdavi SH, Rofooei FR, Sadollah A, Xu C. A wavelet-based scheme for impact identification of framed structures using combined genetic and water cycle algorithms. *J Sound Vib*. 2019;**443**:25-46.
8. Ghasemi SH, Nowak AS, Nazari R. Analyzing reliability missteps: The collapse of the Francis Scott Key Bridge—Target reliability, redundancy, and extreme load combinations. *Struct*. 2025;**74**:108590.
9. Maison BF, Kasai K. Dynamics of pounding when two buildings collide. *Earthq Eng Struct Dyn*. 1992;**21**(9):771-86.
10. Cattarius J, Inman D. Time domain analysis for damage detection in smart structures. *Mech Syst Signal Process*. 1997;**11**(3):409-23.
11. Seyedpoor S, Ahmadi A, Pahnabi N. Structural damage detection using time domain responses and an optimization method. *Inverse Probl Sci Eng*. 2019;**27**(5):669-88.
12. Staszewski WJ, Robertson AN. Time–frequency and time–scale analyses for structural health monitoring. *Philos Trans R Soc A Math Phys Eng Sci*. 2007;**365**(1851):449-77.
13. Doyle JF. A wavelet deconvolution method for impact force identification. *Exp Mech*. 1997;**37**:403-08.
14. Lai T, Yi TH, Li HN. Parametric study on sequential deconvolution for force identification. *J Sound Vib*. 2016;**377**:76-89.
15. Kaveh A. Advances in metaheuristic algorithms for optimal design of structures. Springer International Publishing; 2021.
16. Kaveh A. Applications of metaheuristic optimization algorithms in civil engineering. Springer; 2017.
17. Doyle JF. Determining the size and location of transverse cracks in beams. *Exp Mech*. 1995;**35**:272-80.
18. Yan G, Zhou L. Impact load identification of composite structure using genetic algorithms. *J Sound Vib*. 2009;**319**(3-5):869-84.
19. Dehghani AA, Shahabi P, Mahdavi SH, Hamzehei-Javaran S, Shojaee S. A time-domain strategy for structural system identification based on incomplete structural responses using fire hawk optimization algorithm. *Iran J Sci Technol Trans Civ Eng*. 2025:1-34.
20. Kaveh A. Applications of artificial neural networks and machine learning in civil engineering. *Stud Comput Intell*. 2017;1168.
21. Aucejo M. A data-driven metamodel-based approach for point force localization. *Mech Syst Signal Process*. 2022;**171**:108881.

[DOI: 10.22068/ijoce.2025.15.2.629]

22. Seno AH, Aliabadi MF. Uncertainty quantification for impact location and force estimation in composite structures. *Struct Health Monit*. 2022;**21**(3):1061-75.
23. Liu Y, Wang L, Gu K. A support vector regression (SVR)-based method for dynamic load identification using heterogeneous responses under interval uncertainties. *Appl Soft Comput*. 2021;**110**:107599.
24. Sarego G, Cappellini L, Zaccariotto M, Galvanetto U. Impact force reconstruction in composite panels. *Procedia Struct Integr*. 2017;**5**:107-14.
25. Baek SM, Park JC, Jung HJ. Impact load identification method based on artificial neural network for submerged floating tunnel under collision. *Ocean Eng*. 2023;**286**:115641.
26. Cooper SB, DiMaio D. Static load estimation using artificial neural network: Application on a wing rib. *Adv Eng Softw*. 2018;**125**:113-25.
27. Guo C, Jiang L, Yang F, Yang Z, Zhang X. An intelligent impact load identification and localization method based on autonomic feature extraction and anomaly detection. *Eng Struct*. 2023;**291**:116378.
28. Kaveh A, Bakhshpoori T, Hamze-Ziabari SM. M5' and Mars based prediction models for properties of self-compacting concrete containing fly ash. *Period Polytech Civ Eng*. 2018;**62**(2):281-94.
29. Cha YJ, Ali R, Lewis J, Büyüköztürk O. Deep learning-based structural health monitoring. *Autom Constr*. 2024;161:105328.
30. Jia J, Li Y. Deep learning for structural health monitoring: Data, algorithms, applications, challenges, and trends. *Sensors*. 2023;**23**(21):8824.
31. Yang H, Jiang J, Chen G, Zhao J. Dynamic load identification based on deep convolution neural network. *Mech Syst Signal Process*. 2023;**185**:109757.
32. Yu X, Dan D. Identification of impact load and partial system parameters using 1D-CNN. *Open Sci Index*. 2024;**5**:3.
33. Zhou J, Cai Y, Dong L, Zhang B, Peng Z. Data-physics hybrid-driven deep learning method for impact force identification. *Mech Syst Signal Process*. 2024;**211**:111238.
34. Zhou JM, Dong L, Guan W, Yan J. Impact load identification of nonlinear structures using deep recurrent neural network. *Mech Syst Signal Process*. 2019;**133**:106292.
35. Yang H, Jiang J, Chen G, Mohamed MS, Lu F. A recurrent neural network-based method for dynamic load identification of beam structures. *Mater*. 2021;**14**(24):7846.
36. Maragheh HK, Gharehchopogh FS, Majidzadeh K, Sangar AB. A hybrid model based on convolutional neural network and long short-term memory for multi-label text classification. *Neural Process Lett*. 2024;**56**(2):42.
37. Diederik K. Adam: A method for stochastic optimization. *Int Conf Learn Representations (ICLR)*. 2014.
38. Shahriari B, Swersky K, Wang Z, Adams RP, De Freitas N. Taking the human out of the loop: A review of Bayesian optimization. *Proc IEEE*. 2015;**104**(1):148-75.

[DOI: 10.22068/ijoce.2025.15.2.629]

39. Kushner HJ. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *J Fluids Eng*. 1964;97-106.
40. Minka TP. A family of algorithms for approximate Bayesian inference. *Massachusetts Inst Technol*. 2001.
41. Stuckman BE. A global search method for optimizing nonlinear systems. *IEEE Trans Syst Man Cybern*. 1988;**18**(6):965-77.
42. Perttunen CD, Stuckman BE. The rank transformation applied to a multivariate method of global optimization. *IEEE Trans Syst Man Cybern*. 1990;**20**(5):1216-20.
43. Elder JF. Global r/sup d/ optimization when probes are expensive: the grope algorithm. *IEEE Int Conf Syst Man Cybern*. 1992;577-582.
44. Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. *J Glob Optim*. 1998;**13**:455-92.
45. Zhang Y, Apley DW, Chen W. Bayesian optimization for materials design with mixed quantitative and qualitative variables. *Sci Rep*. 2020;**10**(1):4924.
46. Frazier PI. A tutorial on Bayesian optimization. *arXiv*. 2018;1807.02811.
47. Mathern A, Steinholtz OS, Sjöberg A, Önnheim M, Ek K, Rempling R, Gustavsson E, Jirstrand M. Multi-objective constrained Bayesian optimization for structural design. *Struct Multidiscip Optim*. 2021;**63**:689-701.
48. Røstum H, Gros S, Aas-Jakobsen K. Constrained Bayesian optimization for engineering bridge design. *Struct Multidiscip Optim*. 2025;**68**(1):1-14.
49. Gupta KK, Bhowmik D. Exploring sustainable solutions for soil stabilization through explainable Gaussian process-assisted multi-objective optimization. *Mater Today Commun*. 2024;**40**:110154.
50. Li G, Wang Y, Kar S, Jin X. Bayesian optimization with active constraint learning for advanced manufacturing process design. *IISE Trans*. 2025:1-23.

[DOI: 10.22068/ijoce.2025.15.2.629]