Vibration-based Multiclass Damage Detection and Localization Using Long Short-Term Memory Networks

Sandeep Sony[1], Sunanda Gamage[2], Ayan Sadhu[3], and Jagath Samarabandu[4]

ABSTRACT

This paper proposes a novel damage detection and localization method of civil structures using a windowed Long Short-Term Memory (LSTM) network. A sequence of windowed samples are extracted from acceleration responses in a novel data pre-processing pipeline, and an LSTM network is developed to classify the signals into multiple classes. Predicted classification of a signal by the LSTM network into one of the damage levels indicates the presence of damage. Furthermore, multiple structural responses obtained from the vibration sensors placed on a structure are provided as input to the LSTM model, and the resulting predicted class probabilities are used to identify the locations with a high probability of damage. The proposed method is validated on the experimental benchmark data of the Qatar University Grandstand Simulator (QUGS) for binary classification, as well as the Z24 bridge benchmark data for multiclass damage classification associated with different levels of pier settlement and the numbers of ruptured tendons. The results show that the proposed LSTM-based method performs on par with the one dimensional convolutional neural network (1D CNN) on the QUGS dataset and outperforms 1D CNN on the Z24 bridge dataset. The novelty of this paper lies in the use of recurrent neural network-based windowed LSTM for multiclass damage identification and localization using vibration response of the structure.

KEYWORDS

Structural health monitoring; LSTM; 1D CNN; damage detection; damage localization;

[1]Former PhD Student, Department of Civil and Environmental Engineering, Western University.

[2]PhD Student, Department of Electrical and Computer Engineering, Western University.

[3]Corresponding Author: Assistant Professor, Department of Civil and Environmental Engineering, Western University, Canada. Email: asadhu@uwo.ca, Ph: 519-661-2111 x81431.

[4]Professor, Department of Electrical and Computer Engineering, Western University.

data augmentation.

1. INTRODUCTION

Next-generation civil infrastructure is designed and constructed using emerging technologies and advanced construction materials to achieve resiliency against natural disasters and operational conditions. However, integrity and condition monitoring of existing ageing structures are still lacking in advancements. Over time, structures deteriorate and lose their load-carrying capacity due to various factors such as environmental, heavy traffic, and human-induced damages, requiring cutting-edge technology for continuous structural inspection. Recently, the American Society of Civil Engineers (ASCE) reported the current condition of infrastructure in the United States with a low grade of D+. Canadian infrastructure has been ranked along the same lines; it has been reported that it would take 1.1 trillion dollars to replace the crumbling infrastructure that is in deplorable condition. It can be inferred that, the development of efficient strategies for continuous structural monitoring is of paramount importance for the ageing structures. The conventional approach is to employ a well-trained structural inspector to investigate and inspect the structure, identify defects and implement appropriate maintenance strategies. However, manual condition assessment of the structures is subjective, error-prone and laborious, incurring a significant portion of the annualized maintenance budget. Structural Health Monitoring (SHM) provides a sensor-driven real-time inspection technology to address these challenges of manual visual inspection (Kaveh et al. 2014, Qarib and Adeli 2014, Cawley 2018, Erazo et al. 2019, Singh and Sadhu 2020).

SHM employs suitable diagnostic algorithms and assists infrastructure owners and decision-makers in maximizing the safety, serviceability, and functionality of critical structures. Vibration-based SHM has emerged as one of the promising fields for condition monitoring and damage diagnosis of civil infrastructure (Gatti 2019, Okayasu and Yamasaki 2019, Singh and Sadhu 2021) and offers a viable option for tracking time-varying damages in the structures based on the measured data. Existing damage identification techniques involve analysis in time-domain, frequency-domain, and time-frequency domain (Sirca and Adeli 2012, Sony and Sadhu 2020, Barbosh et al. 2020, Singh et al. 2021) along with a combination of artificial intelligence techniques (Ying et al. 2013, Sadhu et al. 2017, Rafiei and Adeli 2018, Perez-Ramirez et al. 2019). Artificial intelligence (AI) methods have successfully been applied to solve challenging tasks in several engineering domains and to automate and improve the classification and data mining tasks (Kaveh et al. 2021, Pouyanfar et al. 2019). Likewise, AI techniques provide promising opportunities for detection and localization of damages in civil infrastructure by analyzing various sensor measurements with minimal user intervention, thereby reducing cost and increasing accuracy and reliability of infrastructure management. Recently, infrastructure monitoring using images of cracks and damage (Sony et al. 2019, Salehi and Burgueno 2018) has garnered significant attention as a straightforward autonomous approach to monitor large-scale structures, where Convolutional Neural Networks (CNNs) has gained immense popularity.

Historically, CNN was first introduced to classify low-resolution images of handwritten characters and was named as LeNet (LeCun et al. 1998). Since then, various CNN models with different architectures were developed. A popular ImageNet CNN model, *AlexNet* (Krizhevsky et al. 2012), was developed by researchers from the University of Toronto where several layers of convolution and max-pooling were used to train the database. The Visual Geometry Group of Oxford University improved AlexNet and named *VGGNet* (Simonyan and Zisserman 2014) that showed how the depth of CNN influences the accuracy of image reconstruction. The development of new CNN architectures introduced a trend towards using more and more (i.e., *deep*) layers. Computing giant *Google* developed a deeper network, *GoogleNet* (Szegedy et al. 2014), with improved dimensionality reduction and computational efficiencies, while *ZFNet* provided a considerable improvement in classification of error rate over AlexNet.

Primarily designed for object recognition, 2D CNN algorithms were mostly explored for images in various SHM applications to detect defects and anomalies autonomously. Cha et al.

(2017) presented a vision-based methodology for detecting cracks in concrete structures. The authors used around 40,000 images of damaged and undamaged concrete surfaces collected from various concrete structures to evaluate the accuracy of damage classification using a 2D CNN architecture. Zhang et al. (2017) proposed a pixel-level CNN to detect cracks on 3D pavement surfaces. The proposed network was more efficient than the traditional CNN architecture because of the absence of pooling layers that downsized the output of previous layers. Zhao et al. (2018) investigated CNN for crack detection in bridges. For bridge damage classification, an AlexNet-based CNN was trained first with around 3800 images of various bridges. For recognition of the bridge components, a ZFNet-based faster regions-CNN was trained with 600 images. To detect cracks, a GoogleNet-based CNN was trained with 60000 cracked and uncracked images. Apart from ageing-related damage identification, image-based damage detection also showed significant promises for post-disaster reconnaissance. Liang (2019) investigated CNN bridge inspection for system level, component level, and local damage detection. The proposed network was made up of a VGG16 Transfer Learning-based neural network with Bayesian optimization for classification, a faster R-CNN for component detection, and a deep CNN for semantic damage segmentation. Recently, several researchers (Ye et al. 2019, Azimi et al. 2020, Sun et al. 2020, Sony et al. 2021) provided a systematic critical review of various deep learning techniques, especially CNN, for structural damage detection.

Similar to the images of damage/crack, researchers have also explored deep learning methods for damage detection using temporal information of sequential data, such as, acceleration measurements. For example, Guo et al. (2014) proposed sparse coding to extract features from unlabeled acceleration measurements. The damage classification was carried out using CNN, and the results were compared with the traditional machine learning methods, such as, logistic regression and decision trees. Gulgec et al. (2017) conducted a simulation study on a steel gusset plate connection by varying the size and location of the damage. The measurements were also contaminated with 1% and 2% noise to simulate real-world conditions, and CNN was used to classify damage. The proposed method achieved an error of 2% and showed robustness against environmental noise. Out of various CNN architectures, One dimensional (1D) CNN (Kiranyaz et al. 2019) has shown promising results in capturing the temporal information and undertaking damage detection using vibration data.

In the recent few years, Abdeljaber et al. (2017,2018) introduced 1D CNN for a non-parametric vibration-based damage detection. The authors trained the neural network on a vibration signal dataset obtained on a 30-joint steel truss structure, named Qatar Grandstand, by damaging each joint and keeping the other joints undamaged. The proposed model was trained individually on each joint, therefore a total of 31 measurement setups were conducted to develop the training database for binary classification. Ni et al. (2020) showed the applicability of 1D CNN with autoencoders for anomaly detection under data compression. Moreover, Zhang et al. (2019) extended the applicability of 1D CNN to detect changes in physical parameters of the structures, such as, stiffness and mass. Various structural components such as a beam and steel girder bridge were used for validating the proposed algorithm. Recently. Sharma and Sen (2020) showed the applicability of 1D CNN for damage detection in structural steel frames. Experimental validation was performed on a 2D-steel frame with different damage locations and severity of the damage. In another study, Liu et al. (2020) used transmissibility function-based 1D CNN to effectively identify damage in the ASCE SHM benchmark structure and compared the performance against time-domain and frequency-domain methods. 1D CNN primarily exhibited superior performance over artificial neural networks (ANNs) in the context of computation efficiency and noise imperative for big data. While, 1D CNN captures relevant information in a neighborhood of samples, it lacks the ability to learn the long-term dependencies of the sequential datasets, which is relevant for structural damage identification over a long period of data. To address this challenge, Long Short-Term Memory (LSTM) was recently explored for damage detection in numerical benchmark data (Lin et al. 2019), where the performance of LSTM was compared with various Machine Learning techniques.

In this paper, a windowed Long Short-Term Memory (LSTM) network is proposed for multiclass structural damage detection and localization. The proposed method captures long term patterns in an acceleration signal by feeding a sequence of windows extracted from the signal as input to the LSTM model, allowing it to make predictions on the acceleration data. This makes the LSTM architecture a valuable technique in vibration-based SHM. This study makes four novel contributions. First, it introduces a standalone LSTM-based approach for damage localization using acceleration measurements. Second, the limited dataset is augmented by windowing the acceleration measurements and a novel approach of voting on the prediction class for windowed-dataset is presented to increase the prediction accuracy. Third, a thorough hyperparameter tuning analysis is conducted followed by the random initialization of the weights for tuned parameters, comparing the results with 1D CNN. Fourth, the proposed method is demonstrated for multiclass and multi-level damage identification in a full-scale bridge.

The paper is structured as follows. A brief introduction of the structural damage identification using deep learning techniques is presented in Section 1, followed by the gap areas of the existing research and novelty of the proposed method. Next, Section 2 presents the proposed methodology based on LSTM networks along with the data pipeline and performance metrics to identify and localize damage. The results including both binary and multiclass damage localization are illustrated in Section 3. This section also highlights the importance of hyperparameter tuning and evaluation of optimal window size along with the sensitivity to random initialization of weights. The key conclusions of the proposed research are described in Section 4.

2. PROPOSED METHODOLOGY

In this paper, a novel method for damage classification and localization is proposed using an LSTM network. First, a theoretical explanation of the proposed method is provided and next, the performance metrics for evaluating the proposed method are discussed.

2.1 Preliminaries of LSTM model

In this section, a method based on a Long Short-Term Memory (LSTM) network (Hochreiter and Schmidhuber 1997) is proposed to classify the vibration measurement into damage levels. Given an input acceleration signal, x=(x_1, ... , x_T), a standard recurrent neural network computes hidden vector sequence as h=(h_1, ... , h_T) and the output vector sequence as y=(y_1, ... , y_T) by iterating Eqs. (1 - 2) from time, t =1 to T:

$$h_t = \mathcal{H}(W_{ih}x_t + W_{hh}h_{t-1} + b_h) \tag{1}$$

where W denotes weight matrices, W_{ih} is the input-hidden weight matrix, W_{hh} is the hidden-hidden weight matrix, b denotes bias vectors, b_h is hidden bias vector, and $\mathcal{H}$ is the hidden layer activation function.

$$y_t = W_{ho}h_t + b_0 \tag{2}$$

where y_t denotes output at time t, W_{ho} is the hidden-output weight matrix, and b_o is output bias vector. $\mathcal{H}$ is usually an element-wise application of a sigmoid function. Fig. 1 illustrates a typical single LSTM memory cell (Chen 2016). The transformations applied to the input x_t and hidden state from the previous time step h_{t-1} in the LSTM cell are described by Eqs. (3-8).

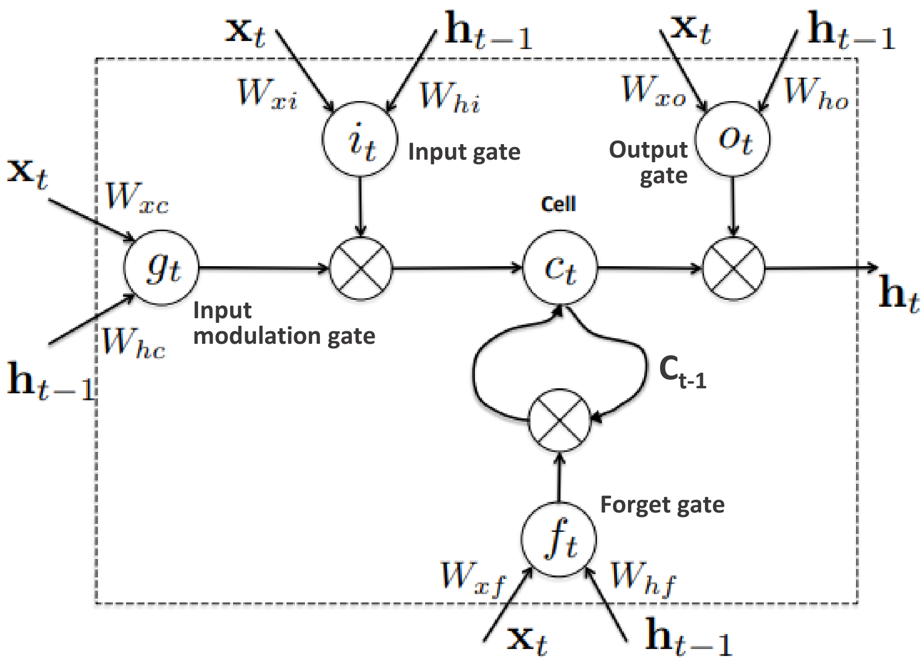

FIG. 1: A typical internal structure of an LSTM cell (Chen 2016).

$$i_t = \tanh(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{3}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{4}$$

$$g_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{5}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{6}$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes g_t \tag{7}$$

$$h_t = \tanh(c_t) \otimes o_t \tag{8}$$

where σ is the logistic sigmoid function, and i, f, g, o and c are, respectively, the input gate, forget gate, input modulation gate, output gate and cell activation vectors. All of which are the same size as the hidden vector h. The weight matrix subscripts have the same meaning, for example, W_{hi} is the hidden-input gate matrix, W_{xo} is the input-output gate matrix. The bias terms (which are added to i, f, g, and o) have been omitted in Fig. 1 for clarity. The weight matrices represent the learnable parameters of the model, while gradient decent algorithm is used to minimize prediction error on a training set.

The cell state c_t encodes the information of the sequence observed up to that time step. The input gate controls the information added to the cell state from the current time step, and the forget gate controls what information needs to be forgotten from the current cell state. For example, if the output vector of the forget gate f_t has a near-zero value in the first dimension, it indicates that the first dimension of the cell state c_t needs to be "forgotten". The forgetting occurs in the element-wise multiplication, i.e., multiplying an element by a near-zero value results in a near-zero element in the output vector. By maintaining cell state in this manner, the LSTM cell is able to capture both long and short term relationships between the input time-series values and the predicted variable (e.g., damage classification). During training, the truncated back propagation is used on truncated sequences to make the process computationally feasible. During prediction, the forward pass can be applied to a long sequences (the LSTM cell can be repeatedly applied to any number of input time steps). More details of the internal structure and training of LSTM can be found in (Chen 2016).

2.2 Damage detection using LSTM model

The proposed deep learning model for the time-series level classification is a multi-layer LSTM network architecture, as shown in Fig. 2. The pre-processed sequences of windows are given as input to the model, and the output of the final LSTM time step is considered as the prediction of a sequence (the set of classification probabilities $P(y = c_t)$ to each class c_t). During training, forward and backward passes are performed on the input sequences, and the weight updates are made to minimize the cross-entropy loss on a batch of sequences. The predicted set of classification probabilities $P_p(y_c)$ for a full acceleration measurement is obtained by summing the class probabilities of all the window sequences in a single time-series. The class with the maximum probability is the predicted damage level classification of the series. It may be noted that this is equivalent to voting on the classification probabilities of individual window sequences to arrive at the prediction of the full series. It is observed that the voting process improves the prediction accuracy and other evaluation metrics on the test data.

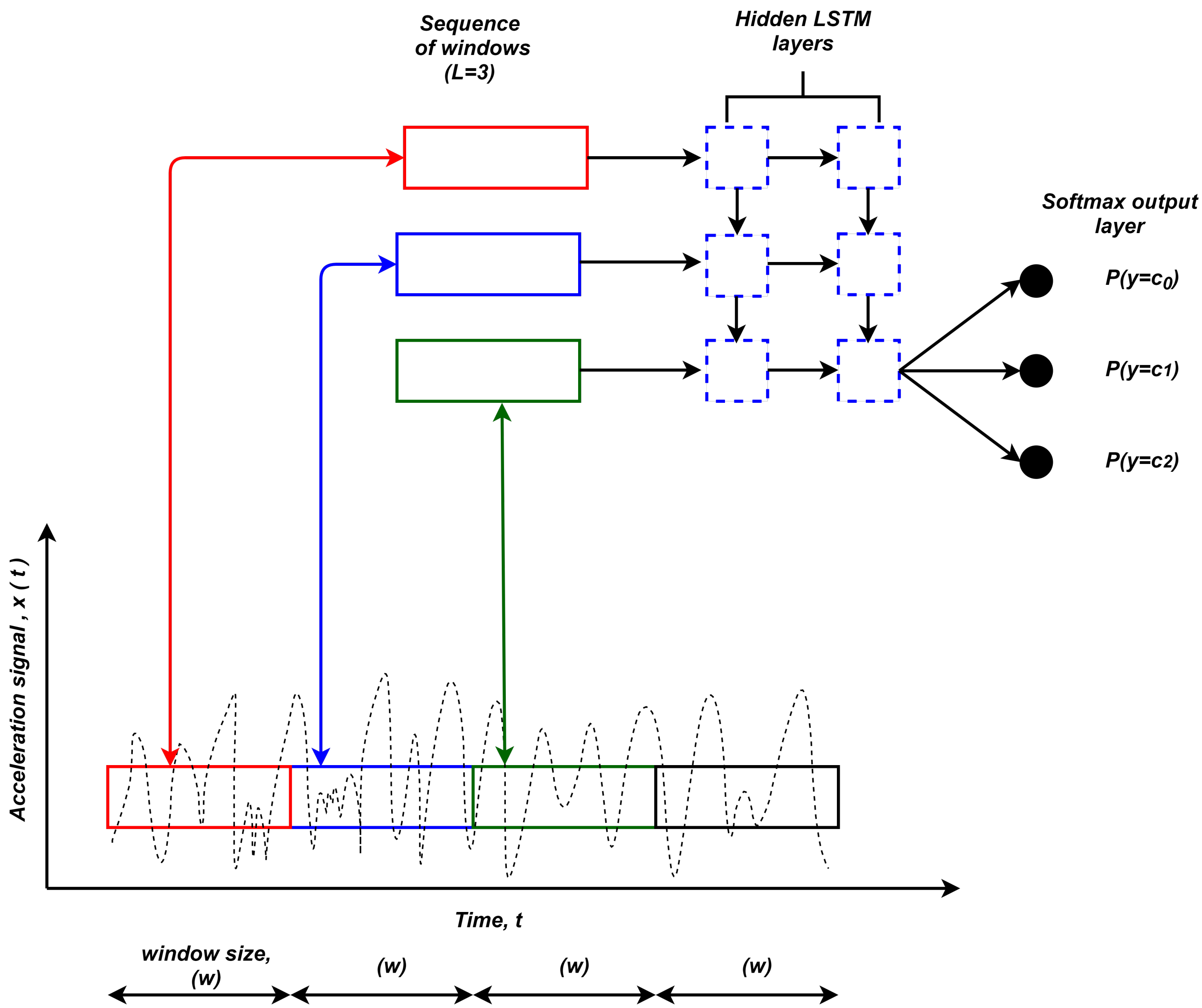

FIG. 2: Extracted sequences of windows from the vibration signals in the LSTM network.

In the proposed method, the window size w and the sequence length L (number of windows in a sequence) become hyperparameters that are tuned to improve the accuracy of the neural network. The hyperparameters of LSTM network include a number of layers and a finite number of nodes in each layer (network architecture), activation function, and batch size responsible for performing weight updates during the training. Optimal parameters are found using a random search on a hyperparameter space (Bergstra and Bengio 2012). In each iteration, the search algorithm randomly selects a configuration of values for hyperparameters from a specified set of possible values, and trains three models with those parameters on three splits of training and validation sets, resulting in a 3-fold validation. The configurations of the hyperparameters providing the highest mean validation accuracy is selected as the final tuned set of hyperparameters of the model. A training session is terminated when, either a specified maximum number of epochs is reached, or the validation loss does not decrease for a specified number of epochs (i.e., early stopping). The final network weights are taken from the epoch with the smallest validation loss.

The purpose of the proposed method is to classify acceleration responses depending on various damage levels and classes. The classification problem is presented as binary (undamaged vs. damaged) or multiclass (undamaged, and damage of more than two levels). Fig. 3 illustrates the proposed data pipeline, which consists of a series of pre-processing and post-processing steps with an LSTM network as the classification model. It should be noted that the solid black lines represent datasets, blue dotted lines represent operation on the datasets, and multiple arrows represent multiple data instances. A single acceleration time-series consists of a large number of samples (for example, a record of long vibration data). In the proposed method, the acceleration is first normalized with respect to its mean and standard deviation. This improves the convergence rate of models trained on the datasets and prevents outliers from dominating the input (Ioffe and Szegedy 2015). Second, the segment of the scaled time-series is fed into a sequence of continuous windows (window size of w), and a sequence of such windows (length L) is arranged to form one input instance to the LSTM network. Thus, the input of the network is a w-dimensional sequence of length L. Many such sequences can be extracted from a single original acceleration time-series, and each sequence is assigned to a label (damage level) of the original series. The process of extracting sequences of windows from a time-series is illustrated in Fig. 2. This technique of transforming the original series into sequences of windows effectively reduces the data dimension, and additionally, it increases the training set size (multiple sequences per time-series), which in turn allows training deep learning models with less over-fitting. Imbalanced data can cause problems in model training. To alleviate this problem, a balanced dataset is created by preparing training set by randomly selecting a number of undamaged sequences equal to the number of damaged sequences.

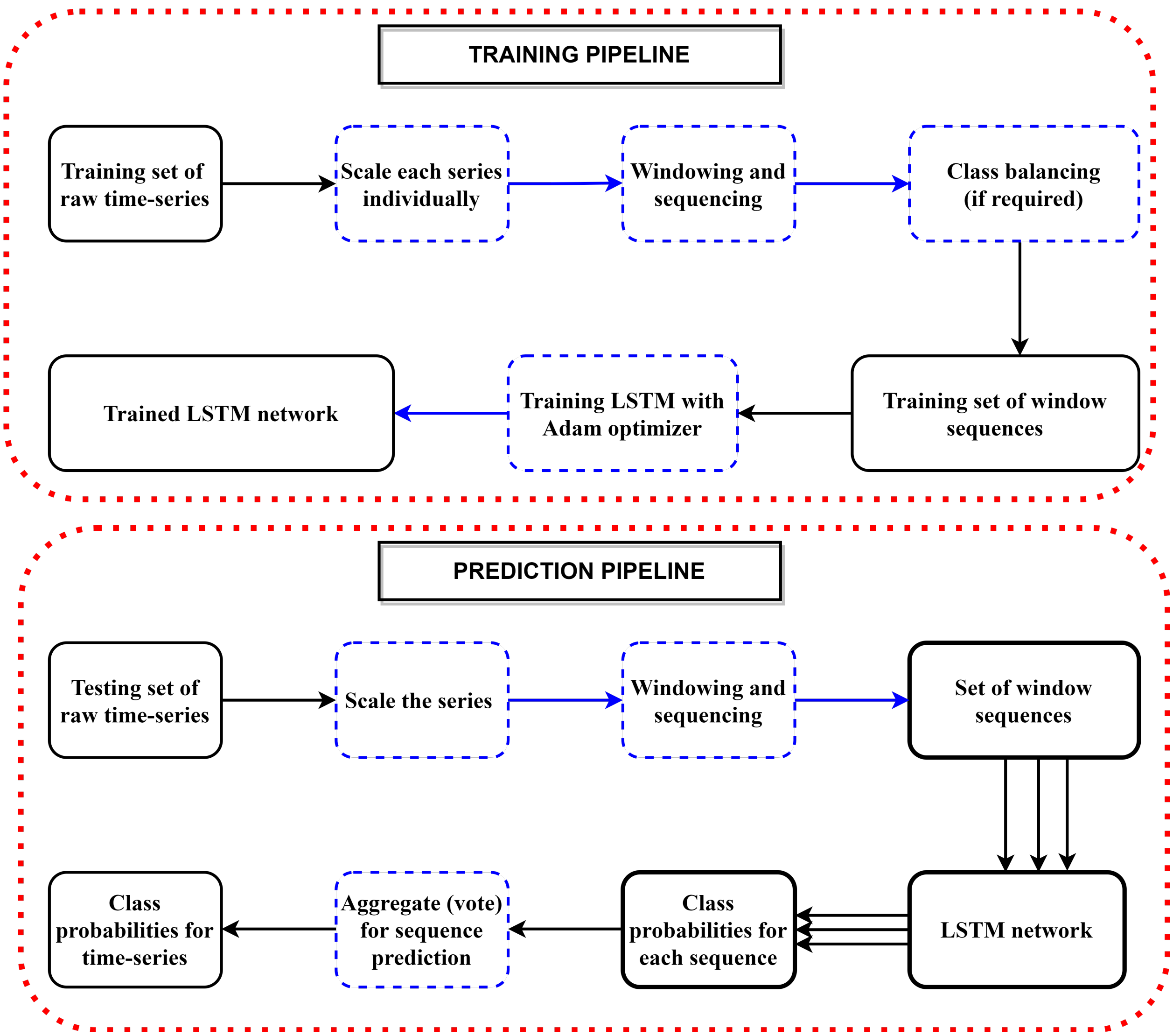

FIG. 3: Data pipelines for training and prediction of the LSTM network from the acceleration response.

2.3 Damage localization using LSTM model

The LSTM network described in the previous section produces a set of probabilities $P_p(y_c)$ for a given acceleration signal. These indicate the probability that the signal belongs to each class c. Multiple signals are obtained from accelerometers placed in the critical locations of a structure, and model prediction probabilities are computed for each signal. In this manner, a probability of damage distribution over the space of the structure can be estimated, and locations with high probability of damage can be identified. This damage probability distribution over the structure can be visualized (for example, heatmaps of probability values over a 2D structure) to aid an engineer in quickly localizing damages. Algorithm 1 summarizes the proposed approach for damage detection and localization.

Algorithm 1: Damage detection and localization

Input: A set of acceleration signals from a structure.

Output: Predicted damage class (damage detection) and probabilities of damage over the structure (damage localization)

(a) The acceleration signal data is pre-processed, as shown in the training pipeline in Fig. 3.

(b) A single LSTM model is trained for damage classification.

(c) The predicted class label (undamaged or damage level) for a signal is computed from the model (i.e., damage detection).

(d) Probabilities of classification (to damage level classes) are computed for each signal from the model.

(e) Damage probabilities for all signals form a distribution of damage probabilities over the structure. High probabilities correspond to damaged locations (damage localization).

(f) Visualize the distribution of damage probabilities for inspection and automated decision.

2.4 Performance criteria

Several metrics are used to evaluate the performance of a classification model. These metrics measure different aspects of the obtained results. A brief description of the selected metrics is provided below and explained in the context of SHM of civil infrastructure. The primary form of prediction results is given by the confusion matrix, which is a tabulation of classifications made by a model. It shows the classification distribution of a model, and helps identify properties of the model, such as when it is consistently misclassifying one class as another. The confusion matrix is obtained for both binary and multiclass classifications. Table 1 shows the confusion matrix for the case of binary classification. It should be noted that True Positive is denoted as TP, True Negative as TN, False Positive as FP, and False Negative as FN. Multiple metrics can be derived from the confusion matrix, as shown in Table 2.

TABLE 1: Confusion matrix for a binary classification problem.

	Predicted class	
	Damage	Healthy
Damage	TP	FN
Healthy	FP	TN

In this study, two key metrics are used as the performance metrics to evaluate the proposed method, namely, accuracy, and FN rate (FNR). Accuracy is the primary evaluation metric to understand the ability of the model to correctly classify the inputs. A false negative (type II error) represents a truly damaged series that is classified as undamaged by the model, which could lead to catastrophic consequences in a critical structure. Therefore, the FNR is measured and compared in the model evaluation experiments in this work. Two plots highlight the trade-off between metrics as the decision threshold of the classifier changes: the Receiver Operating Characteristic (ROC) curve shows the trade-off between FPR and true positive rate, and the precision-recall (PR) curve shows the trade-off between precision and

TABLE 2: Description of the selected performance metrics.

Metric	Formula	Remarks
ROC-AUC	Recall Vs FPR	Degree of separability between classes
Accuracy	$\frac{\text{TP+TN}}{\text{TP+FN+FP+TN}}$	Less useful for heavily imbalanced data
Precision	$\frac{\text{TP}}{\text{TP+FP}}$	Positive predicted value
Recall	$\frac{\text{TP}}{\text{TP+FN}}$	True positive rate or sensitivity
False Positive Rate (FPR)	$\frac{\text{FP}}{\text{TN+FP}}$	False alarm when there is no damage
False Negative Rate (FNR)	$\frac{\text{FN}}{\text{TP+FN}}$	No alarm for actual damage

recall. By analyzing these curves obtained from a test set of model predictions, an engineer can make an informed decision on the balance that is needed between the metrics and make a decision on threshold suitable for the task. The PR curve, in particular, is suitable for the evaluating the cases where the datasets are highly imbalanced. In this study, ROC-AUC and PR-AUC curves are plotted for visual comparison. The corresponding area-under-the-curve (AUC) of these graphs (ROC-AUC and PR-AUC) represent an aggregate measure of the model's ability in terms of the relevant metrics. The damage localization results are evaluated by inspecting the visualization of damage probabilities over the structure, and verifying that high damage probabilities have been assigned to the damaged location, while the other locations are assigned with low damage probabilities.

3. PERFORMANCE EVALUATION

The proposed method is evaluated in two studies: an experimental setup, and a full-scale bridge. For performance evaluation, accuracy, FNR, ROC-AUC and PR-AUC are used. Several parametric studies are conducted to evaluate the performance of the proposed method with the window size in the model input, and the effect of voting on individual windows to obtain the final prediction.

3.1 Experimental study

3.1.1 A brief description of the data

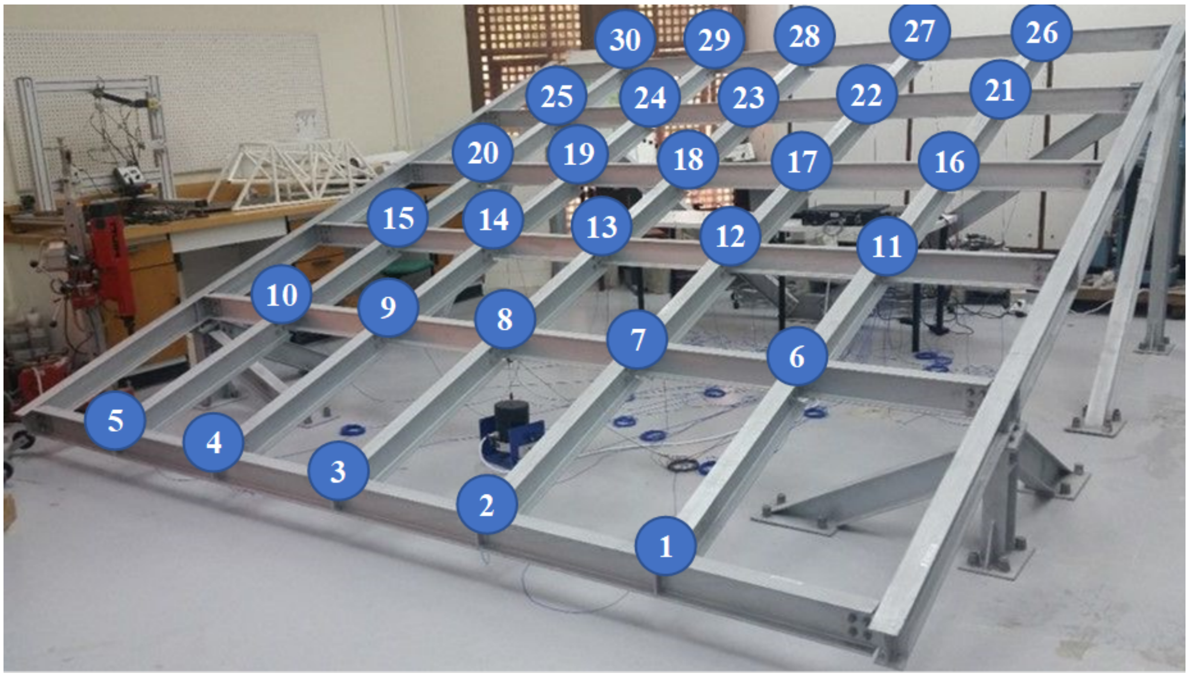

FIG. 4: QUGS testbed, where the joints are numbered from 1 to 30 (Abdeljaber et al. 2017).

Damage detection where the classification is undertaken between one of two classes, such as damaged and undamaged, is called a binary classification. In this paper, the experimental benchmark data of Qatar University Grandstand Simulator (QUGS) (Abdeljaber et al. 2017) is used to evaluate the performance of the proposed method. The QUGS was constructed to evaluate and develop effective structural damage detection techniques suitable for monitoring of modern stadia, as shown in Fig. 4. The frame was designed to carry a total of 30 spectators with area dimensions of 4.2 m × 4.2 m. The design considerations used for the experimental test structure was to guarantee its safety and compatibility with the specifications of modern grandstands. The structure was utilized in its current form (steel frame only) to generate vibration data under several structural damage cases. The grandstand was excited using a shaker and two 16-channel data acquisition systems were used to acquire the acceleration responses. A total of 31 scenarios are considered: the first scenario corresponds to the reference (undamaged) case, while in scenarios 2 to 31, damage was introduced by loosening the bolts at the joints 1 to 30, respectively. A total of 30 accelerometers were mounted at the joint of each girder and the filler beam, and a magnetic mounting plate was used to attach the sensors on the frame. The acceleration data was sampled at 1024 Hz and collected for 256 seconds under a band-limited white noise excitation from the shaker with a frequency range of 0-512 Hz.

3.1.2 Optimal hyperparameter tuning

A range of values for hyperparameters is used for tuning the LSTM model using a random search, as shown in Table 3. The window size is varied as an external parameter between 64 and 512 samples. Various values for other hyperparameters, such as, the number of hidden layers (range of 1-6), and nodes in hidden layers (range of 32-1024) are considered to achieve optimal performance of the proposed model. Consequently, the random search algorithm explores shallow, wide and deep LSTM architectures. The optimal hyperparameter configuration obtained using random search algorithm for QUGS experiment is presented in Table 4. It can be observed that the highest performing window is 64, and it requires at least three hidden layers for best accuracy. A comparison is drawn between the window-size (w) and the performance metric (P_m) to understand model performance with changing w. Three different metrics are used, namely, ROC-AUC, accuracy, and FNR as these three metrics broadly cover the efficacy and any shortcomings of the classification model. The result is shown in Fig. 5. It can be observed that w=64 and w=128 yield the best performance metric values (high accuracy and ROC-AUC with low FNR), and performance metrics consistently degrade with increasing window size. This behavior is attributed to a reduction in the data samples (sequences of windows) with increasing w, causing the LSTM network to overfit when trained on smaller datasets.

TABLE 3: Hyperparameters used for tuning in LSTM by random search algorithm.

Parameter	Values
Window size	64, 128, 160, 256, 512
No. of windows in a sequence	2, 4, 8, 16
No. of hidden layers	1 - 6
No. of hidden nodes	1024, 512, 256, 128, 64, 32
Dropout rates	0.2 and 0.5 for each hidden layer
Learning rate	0.0003, 0.001, 0.01
Batch size	64, 256, 512
Cell type	LSTM with *tanh* activation

TABLE 4: Optimal configuration of the LSTM hyperparameters for QUGS data.

Parameter	Values
Window size	64
No. of windows in a sequence	8
No. of hidden layers	3
Architecture	[64, 128, 64, 32, 1]
Dropout rates	0.2
Learning rate	0.001
Batch size	256
Training epochs	100 with early stopping

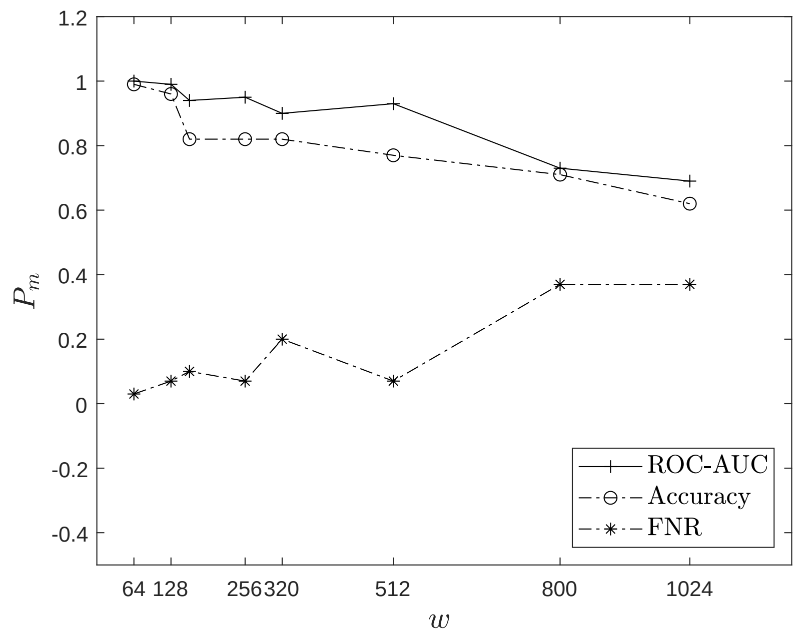

FIG. 5: Performance evaluation of the proposed LSTM model based on window size in the QUGS data.

3.1.3 Effect of majority voting and classification

To compare the proposed LSTM approach with the 1D CNN method introduced by Abdeljaber et al. (2017), a 1D CNN is trained with the same hyperparameter tuning process as outlined before. The optimal 1D CNN network has 4 hidden layers with nodes 256, 128, 64, 32 in each layer and a kernel size of 64. After acquiring the optimal tuned parameters, a parametric study is conducted to understand variance in the metrics by training the LSTM and 1D CNN models with random initialization of network weights of 5 times. The result is shown in Fig. 6. It is found that both models perform well consistently with accuracy at 1.0 and FNR at 0.0 with negligible variability.

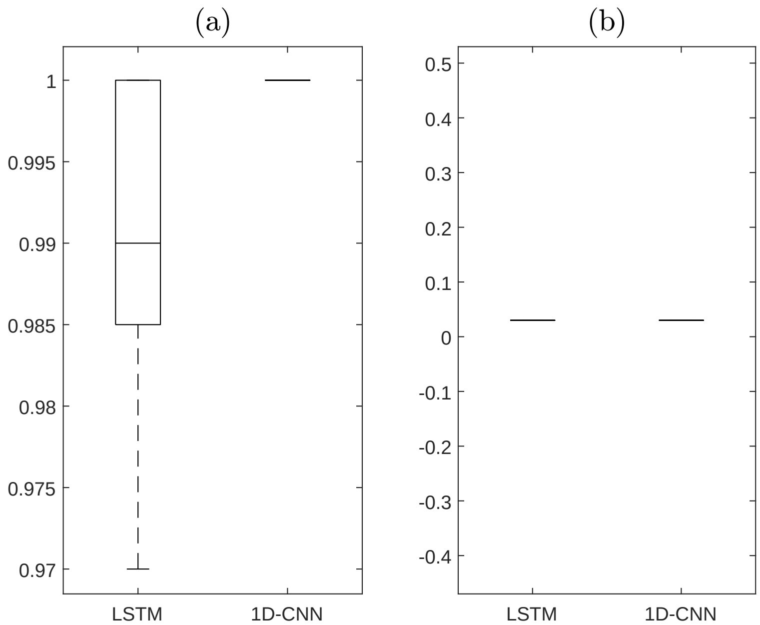

FIG. 6: Performance evaluation for random weight initialization of QUGS data: (a) Accuracy, and (b) FNR.

One possible explanation for the perfect performance behavior of the machine learning models is based on the nature of the acceleration signals. The damages in the joints of the QUGS are highly localized, and the sensors are placed at the exact locations of damage. Furthermore, the controlled experimental laboratory setup makes the acquired signals of high quality. These properties lead to acceleration signals that have distinct, discriminating patterns between the damaged versus undamaged cases. The 1D CNN's capability to learn local structure of the time signals, and the LSTM's capability to learn long-term irregular dependencies lead to excellent performance of both models.

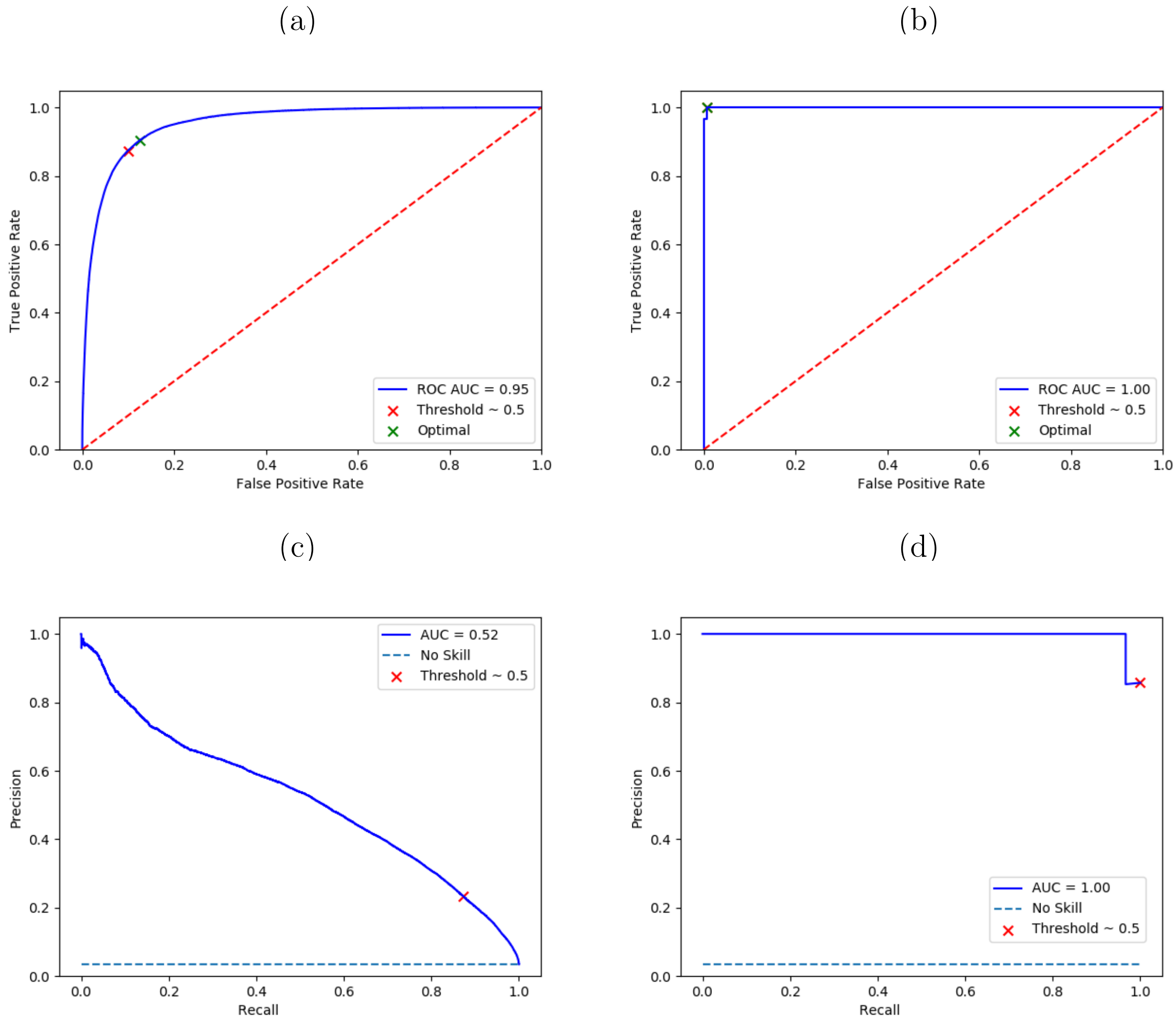

FIG. 7: Performance of the LSTM model on the QUGS data: (a) ROC in individual windows, (b) ROC in voted series, (c) PR in individual windows, (d) PR in voted series.

In the proposed method, the damage classification of a time-series is obtained by voting on the classification of individual windows that constitute the entire time-series. It is observed that this voting process increases performance metrics on time-series predictions considerably in contrast to the predictions on individual windows, as illustrated by the ROC and precision-recall (PR) curves in Fig. 7. The ROC curve is closer to the upper left corner in the voted series predictions (Fig. 7. (b)) than on the individual window predictions (Fig. 7. (a)), which represent an increase in ROC-AUC from 0.95 to 1.0. The PR curve also shifts to the upper right corner as shown in Fig. 7 (d), with an increase in PR-AUC from 0.52 for voted series predictions to 0.99 for individual window predictions. It can be concluded that voting on windows decreases the probability of error significantly to obtain the final prediction. The evaluation metrics (the mean values of five independent trials) are reported in Table 5. It can be observed that, both LSTM and 1D CNN perform well to undertake the binary classification associated with this data. However, the LSTM model performs slightly better than the 1D CNN model in terms of the F1 score and FNR.

TABLE 5: Evaluation metrics of LSTM and 1D CNN for the QUGS data.

Method	Accuracy	Precision	Recall	F1 Score	FPR	FNR	ROC-AUC
LSTM	0.99	0.99	0.98	0.99	0.016	0.017	1.0
1D CNN	0.99	0.99	0.97	0.98	0.0007	0.033	1.0

Damage localization is performed for the QUGS data using Algorithm 1. In each damage scenario, a single joint out of the 30 joints in the grandstand setup was damaged by loosening of the bolts. Acceleration signals are acquired from all joints, and the proposed localization method gives damage probabilities for each joint location. The distributions of damage probabilities for multiple damage scenarios are presented as the heatmaps in Fig. 8. Note that the blue color indicates $P(Damage) \approx 0$ and red color represents $P(Damage) \approx 1$. For example, Fig. 8 (a) shows the scenario where joint 1 of the structure is damaged as shown in Fig. 4 for an illustration of the numbered joints. The damage appears to be heavily localized at the joint. Fig. 8 (e) shows the scenario where joint 23 (location: [3,5]) is damaged, and it is clear that the damage is spread out as it is surrounded by 4 connected branches signifying the affected damage after loosening of the bolts.

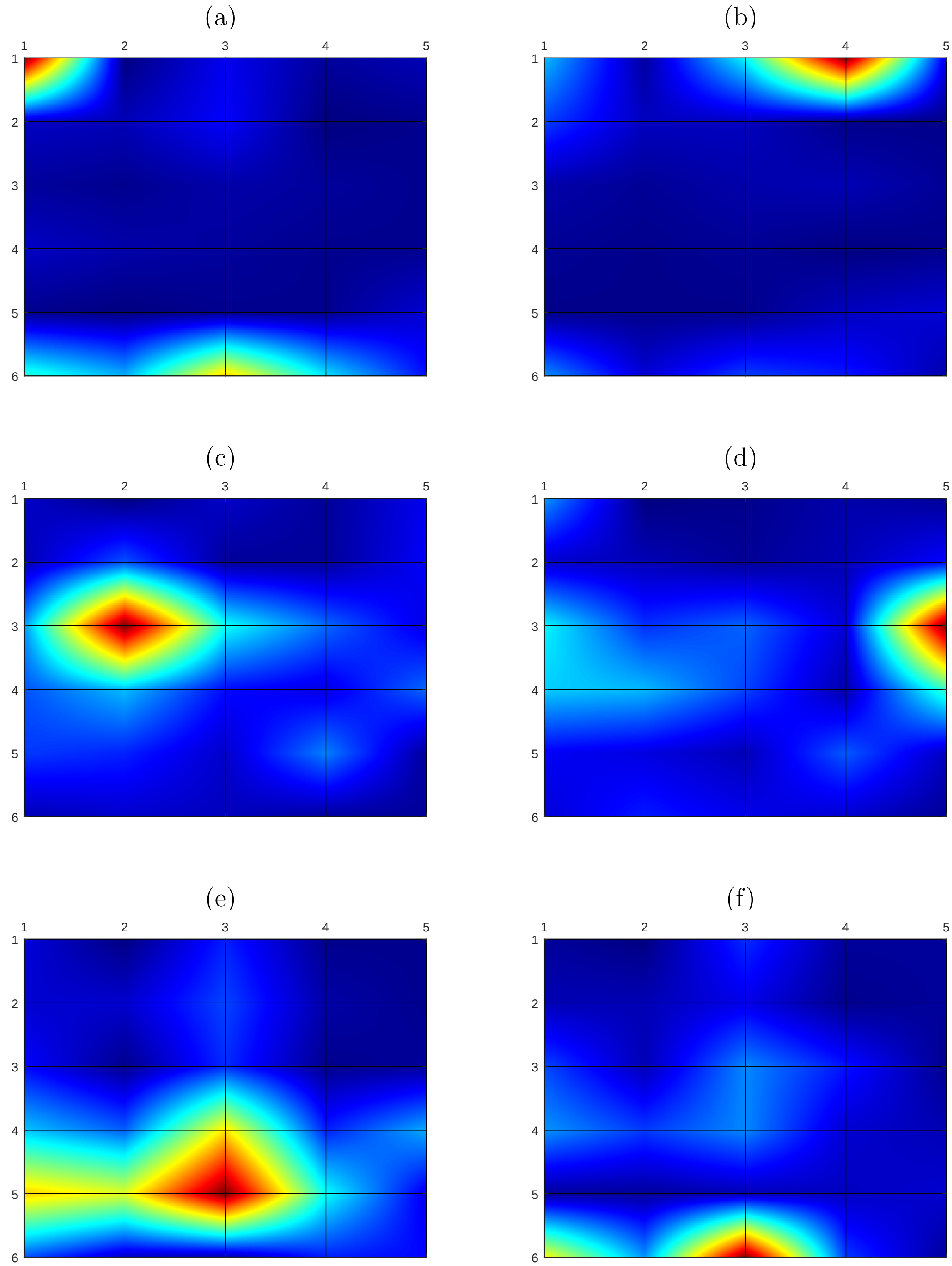

FIG. 8: Damage localization probabilities in the QUGS data for scenarios where the damaged joint is: (a) Joint 1 [1,1], (b) Joint 4 [4,1], (c) Joint 12 [2,3], (d) Joint 15 [5,3], (e) Joint 23 [3,5], and (f) Joint 28 [3,6].

3.2 Full-scale Study

3.2.1 A brief description of the data

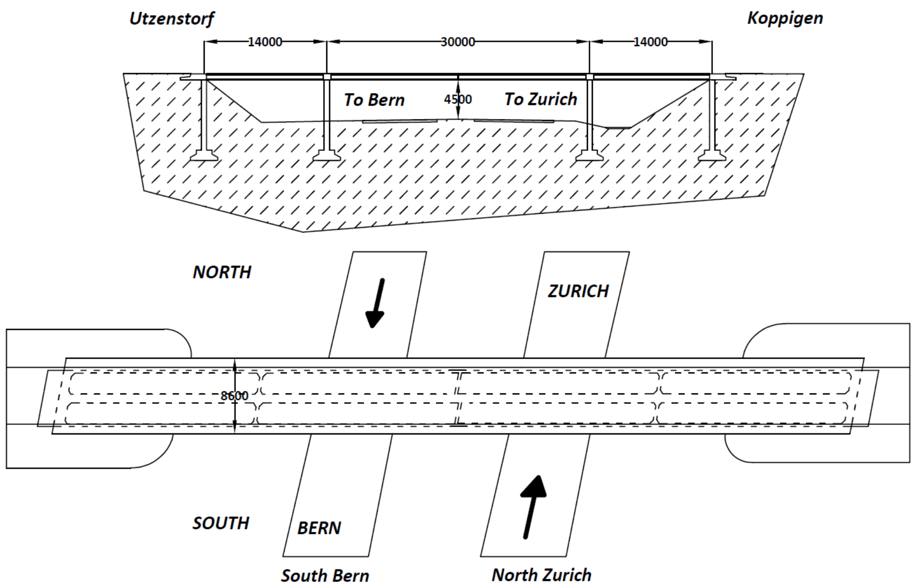

FIG. 9: Schematic of the Z24 bridge (Maeck and Roeck 2003).

Damage detection, where classification is more than two classes, is considered as a multiclass problem. Z24 bridge benchmark data (Maeck and Roeck 2003) is used to evaluate the performance of the proposed method for multiclass damage detection. In this study, two classes of damage are used, namely, pier settlement and rupture of tendons. Both damage classes further have multiple damage levels. The bridge was a classical post-tensioned concrete box-girder bridge with a main span of 30 m and two side spans of 14 m, as shown in Fig. 9. The bridge was demolished at the end of 1998 because a new railway adjacent to the highway required a new bridge with a larger side span. The SHM data was acquired using 16 accelerometers placed at different spans of the bridge, as shown in Fig. 10. The bridge was excited by two shakers, one at the mid-span of the bridge and another at a side-span. Because of the size of the bridge, response was measured in nine setups with each setup containing up to 15 sensors each. In all setups, three accelerometers and the two force sensors were common. The data was sampled at 100 Hz, and a total of 65,536 samples were acquired.

This data was made publicly available by researchers at the Katholieke Universiteit Leuven and is available at: https://bwk.kuleuven.be/bwm/z24.

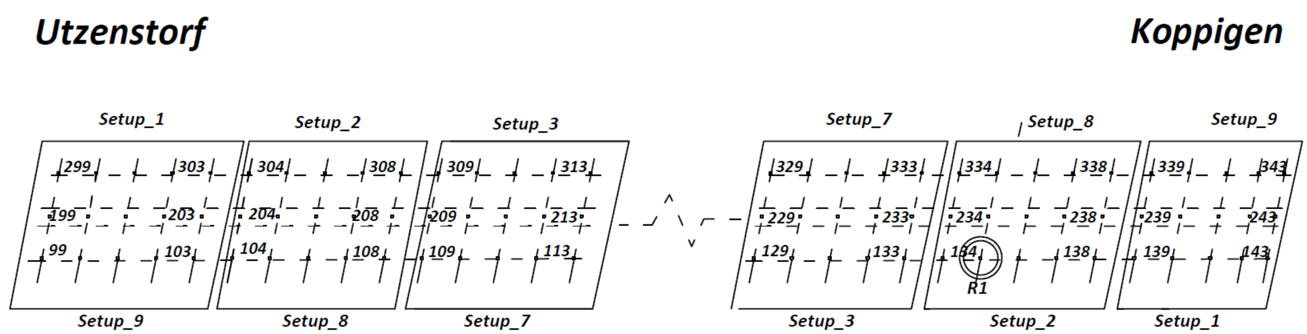

FIG. 10: Sensor placement for the data acquisition in the Z24 bridge (Maeck and Roeck 2003).

TABLE 6: Multiclass problem description for two damage scenarios along with the class label.

Problem	Damage scenario	Class label
0	Undamaged	0
1	Rupture of 2 tendons	1
	Rupture of 4 tendons	2
	Rupture of 6 tendons	3
2	Lowering of pier, 20 mm	1
	Lowering of pier by 40 mm	2
	Lowering of pier by 80 mm	3
	Lowering of pier by 95 mm	4

Several progressive damage scenarios were considered for vibration-based damage identification. Each damage scenario has multiple level of damage. All these damage scenarios are compared with the baseline undamaged state. It can be observed that each damage scenario have different classes of damage, and they were chosen to evaluate the performance of the proposed method to classify various multiclass and multilevel damage cases. For example, rupture of tendons have three levels and lowering of pier have four levels, and together they make a case of two separate damage classes. For detailed explanation of how the damages were induced to the bridge, readers are suggested to referred to (Roeck and Teughels 2004). Multiclass problem is considered based on the type and level of damage. The reference undamaged condition is considered as class-zero for all the cases and the other damages were assigned classes starting from 1 to n depending upon the level of damage, as shown in Table 6. For example, in the case of rupture of tendons, the damage level classes 1, 2 and 3 correspond to damages induced by rupture of two tendons, then four, and finally six tendons. Similarly, there are four damage level classes for lowering of the pier.

3.2.2 Optimal hyperparameter tuning

Optimal hyperparamters of the LSTM model on the Z24 bridge dataset are obtained by performing a random search. Table 7 shows the hyperparameter configuration with highest accuracy. It can be observed that LSTM performed well with w=128. An analysis is performed to understand the effect of window size w on performance P_m. The results are illustrated in Fig. 11, which shows that optimal performance is achieved at w=128, with highest ROC-AUC and accuracy, and lowest FNR. In case of Fig. 11 (a), the ROC-AUC and accuracy reduce while FNR increases, similarly, in case of Fig. 11(b) the FNR remains consistent after w=512 and other metrics are at their peak. Due to larger w, the data size reduces per damage class and it leads to overfitting of the model on the data. A similar random search of hyperparameters is conducted to find optimal parameters for the 1D CNN model. Five repetitions of training and testing sessions are performed to verify the robustness of the models against random initialization of network weights. Accuracy and FNR are computed for each session, and the results are shown in Fig. 12.

TABLE 7: Optimal configuration of the LSTM hyperparameters for the Z24 bridge benchmark data.

Parameter	Values
Window size	128
No. of windows in a sequence	16
No. of hidden layers	3
Architecture	[128, 64, 32, 5]
Learning rate	0.001
Batch size	512
Training epochs	100 with early stopping

(a) (a)

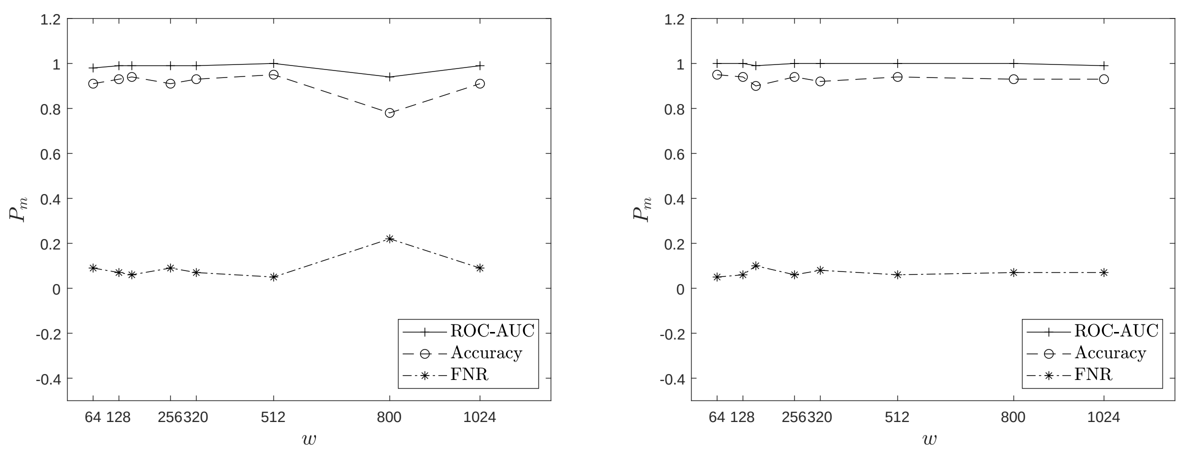

FIG. 11: Performance evaluation of the proposed LSTM model based on window size for (a) rupture of tendons, and (b) pier settlement.

3.2.3 Effect of majority voting and classification

It can be observed that for pier settlement which was measured using 4 sensors for data acquisition, and associated with 4 damage levels, the damage is considered as localized around these sensors. The LSTM performs well with highest accuracy and lowest FNR in comparison to 1D CNN as seen in Fig. 12 (a), and (b). In case of rupture of tendons, where the damage is fairly distributed throughout the bridge deck, the damaged signals are not highly distinguishable in comparison to undamaged signal and other severe damage measurements such as lowering of piers. The accuracy of LSTM drops to 0.9 and FNR increases to 0.2. However, LSTM still performs better than 1D CNN as shown in Fig. 12 (c), and (d). It can be further emphasized that the superior performance of LSTM is attributed to its capability to learn long-term irregular dependencies of complex time signals whereas 1D CNN learns prominently the local neighborhood structure of the signals.

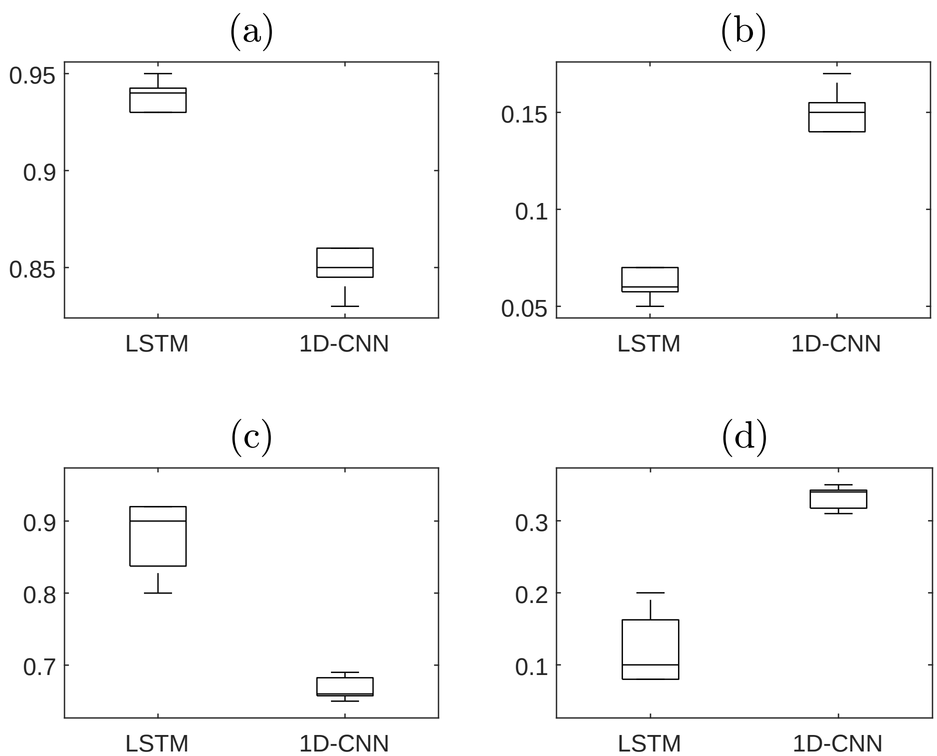

FIG. 12: Performance evaluation of LSTM for random weight initialization for various damage cases in the Z24 bridge: (a) accuracy for pier settlement, (b) FNR for pier settlement, (c) accuracy for rupture of tendons, (d) FNR for rupture of tendons.

Finally, the optimal parameters are used to evaluate the performance of proposed method on full-series versus voted-windowed samples. It is observed that voting on windowed dataset increases accuracy considerably and it is evident in ROC-AUC and PR-AUC curves, as presented in Fig. 13, and 14, respectively. It can be observed that voting on windows from non-localized signal increases the probability considerably by allocating the majority class and ignoring the non-prominent class along with increasing the data samples per class. As shown in Fig. 13, voting on individual windows has improved both ROC-AUC and PR-AUC. However, due to the localized measurement acquisition, and severity of damage in pier settlement, the difference in AUCs of various cases was comparatively similar to the QUGS damage scenario. Moreover, as observed in Fig. 14, where the damage was considerably distributed in case of rupture of tendons, voting on windows increased the ROC-AUC by 5% and PR-AUC by 13%.

Damage localization is performed using Algorithm 1 for multilevel and multiclass damage scenarios of lowering of pier and rupture of tendons. In these damage scenarios, the damage is not highly localized as in the experimental study of section 3.1. Three different structural components of the bridge is used to localize damage and understand the effect of pier settlement. An undamaged pier in Utzenstorf, bridge deck, and damaged pier in Koppigen are used to represent the predicted probability (P_p) and infer damages in three components. The Koppigen pier is used for inducing the damage by lowering it in several increments starting with 20 mm, 40 mm, 80 mm, and moving to 95 mm at the last stage. The P_p is plotted against the sensor number and a dash-dotted average of P_p of structural component is shown as a representation of combined P_p for corresponding structural component, as shown in Fig. 15. For example, Fig. 15 (a, b, c) represents P_p for undamaged pier (UDP), bridge deck (BD), and damaged pier (DP) for 20 mm lowering of piers, respectively. Similarly, Figs. 15 (d, e, f) and (g, h, i) show P_p for 40 mm and 95 mm lowering of piers, respectively.

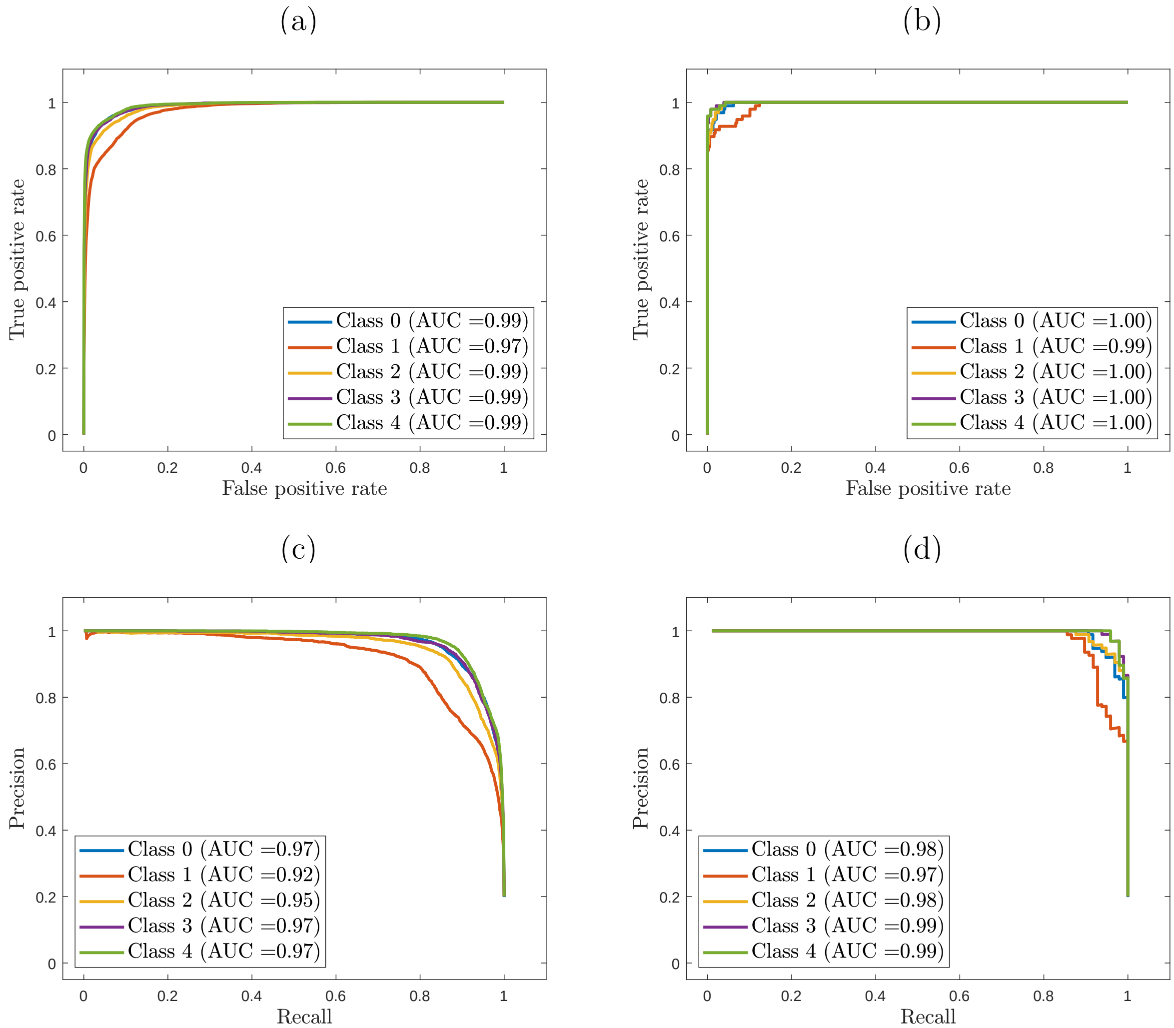

FIG. 13: Performance of the proposed LSTM model in the Z24 bridge for pier settlement: (a) ROC in individual windows, (b) ROC in voted series, (c) PR in individual windows, (d) PR in voted series.

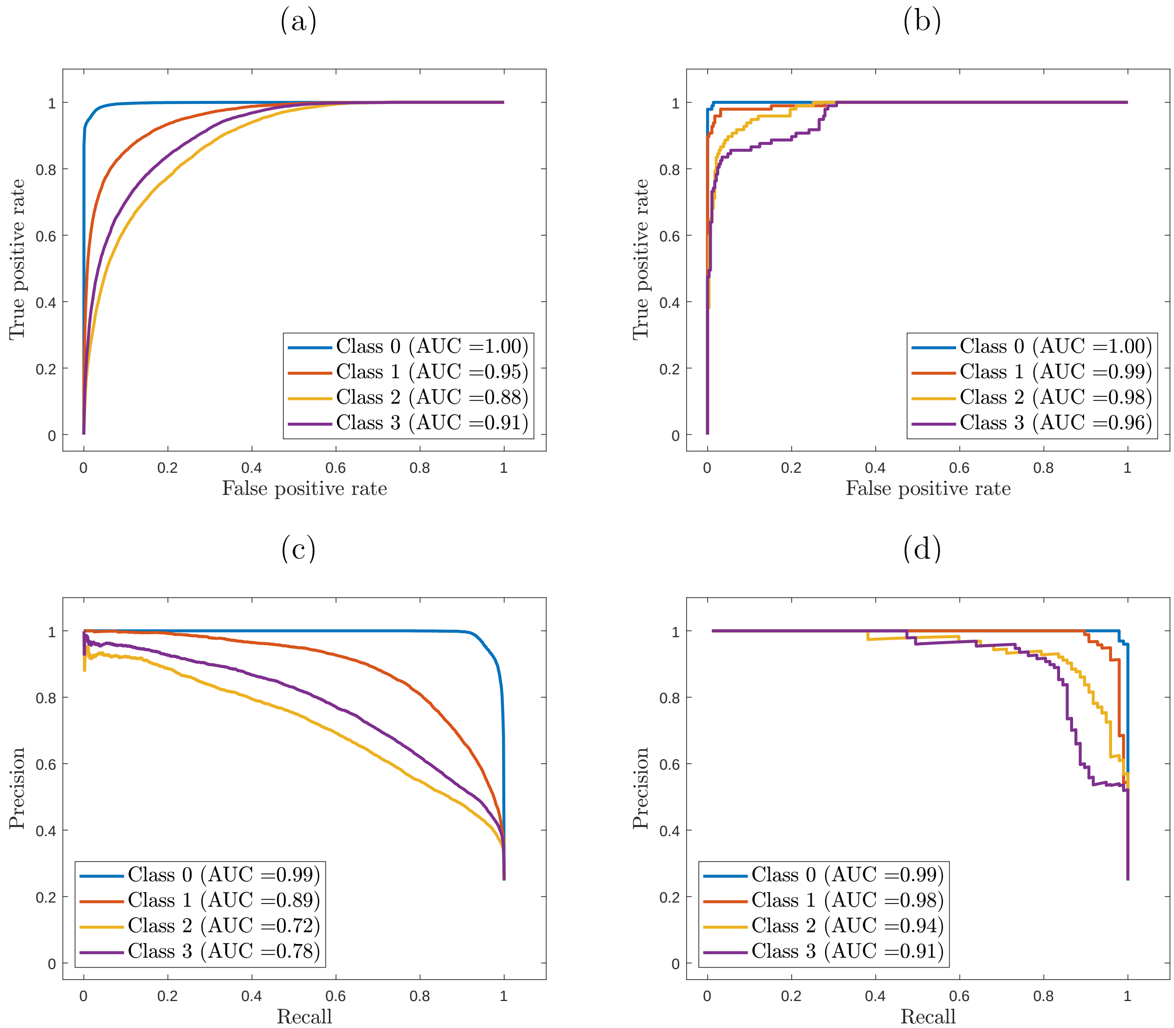

FIG. 14: Performance of the proposed LSTM model by windowing the data of the Z24 bridge in case of rupture of tendons: (a) ROC in individual windows, (b) ROC in voted series, (c) PR in individual windows, (d) PR in voted series.

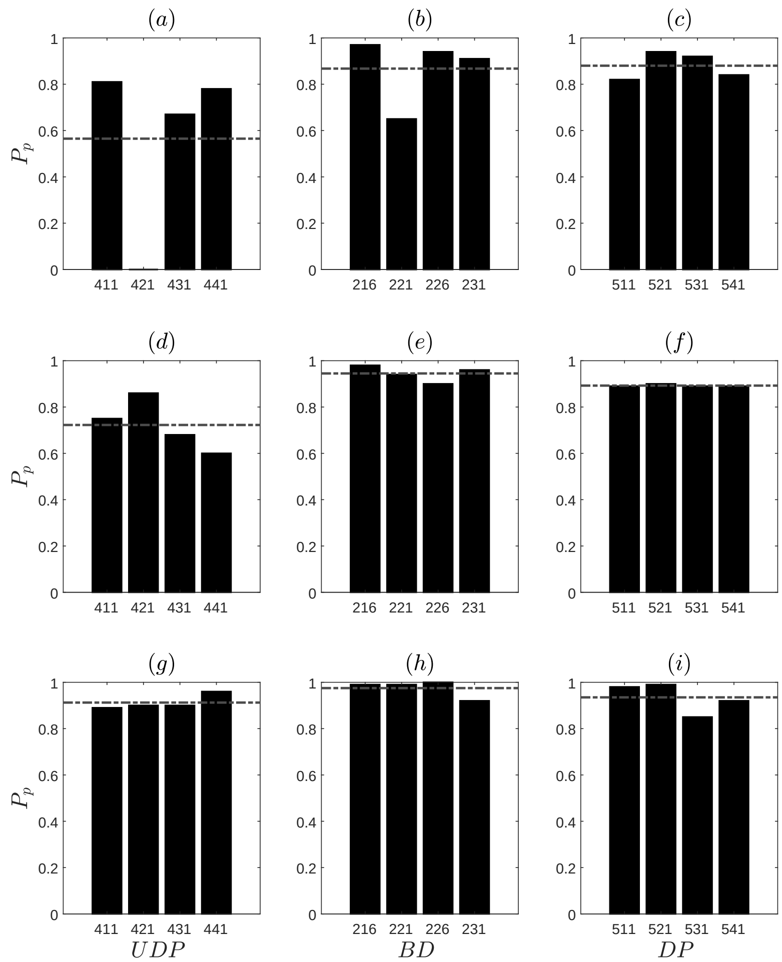

FIG. 15: Damage localization for lowering of pier for three damage levels, where, (a, b, c) are for 20 mm lowering of piers, (d, e, f) are for 40 mm lowering of piers, (g, h, i) are for 95 mm lowering of piers.

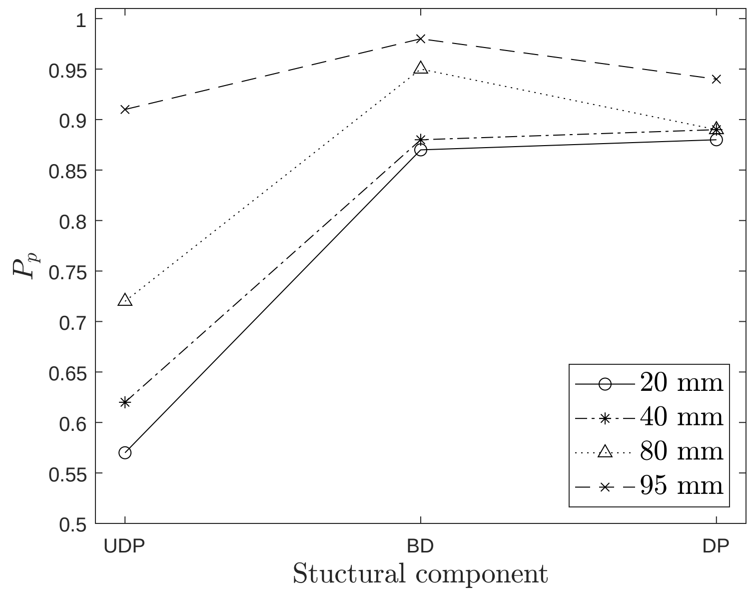

FIG. 16: Damage localization for lowering of pier, where legend shows the amount of pier-settlement.

Due to non-localization of measurement acquisition, it is difficult to infer damage location while considering each sensor separately. However, it is possible to compare average P_p of each structural component for various damage cases. The results considering average P_p for each structural component and various damage levels are shown in Fig. 16. Although there is no correlation between P_p and damage severity, however, as the severity increases, the signals become more distinguishable and the proposed LSTM model performs the classification more effectively. It can be observed from Fig. 16 that UDP shows lowest predicted probability due to its similarity to undamaged baseline signal, however, both BD and DP show higher prediction accuracy. The reason for bridge deck's highest probability of damage is attributed to the surface area and larger affect of differential pier-settlement in entire bridge system. The bridge suffers higher changes in structural responses than at damaged pier itself, as it acted as a support. Finally, Table 8 and 9 show the confusion matrices for the pier settlement damage classification problem with five classes, starting from no damage (class-0) to 95 mm damage (class 4), using LSTM and 1D CNN, respectively. It can be seen that the LSTM model accurately classifies a large percentage of signals in each class (i.e., the diagonal of the confusion matrix), demonstrating high precision by the LSTM model. In comparison, the 1D CNN fails to differentiate between the undamaged case and damage level 1, as well as, between levels 2 and 3, which can lead to significant inaccuracy in the bridge maintenance.

TABLE 8: Confusion matrix for a pier settlement damage problem using the LSTM.

		Predicted Class				
		Pred-0	Pred-1	Pred-2	Pred-3	Pred-4
True Class	True-0	92	1	0	0	3
	True-1	3	87	2	2	3
	True-2	0	4	92	0	1
	True-3	0	0	3	93	1
	True-4	0	0	0	0	97

TABLE 9: Confusion matrix for a pier settlement damage problem using the 1D CNN.

		Predicted Class				
		Pred-0	Pred-1	Pred-2	Pred-3	Pred-4
True Class	True-0	76	11	5	0	4
	True-1	19	66	7	0	5
	True-2	0	0	96	0	1
	True-3	0	0	22	70	5
	True-4	0	0	2	1	94

Similarly, for rupture of tendons, the most affected area would be the bridge deck and the damage induced due to rupture of tendons will create a non-localized and distributed damage throughout the bridge deck in comparison to bridge piers. The damage localization per sensors is avoided due to non-conclusive inference and a comparison between structural components of the bridge is provided directly in Fig. 17. It can be observed that rupture of 6 tendons prove to be worse damage level scenario in comparison to the ruptures with 2 and 4 tendons. Table 10 and 11 show the confusion matrices for the rupture of tendon damage classification problem with four classes, starting from no rupture (class-0) to rupture of 6 tendons (class 3), using LSTM and 1D CNN, respectively. The LSTM model classifies a large percentage of signals in each class correctly (e.g., the diagonal of the matrix). This results in high precision by the LSTM model. In comparison, the 1D CNN model fails to differentiate between damage levels 1, 2 and 3. A considerable percentage of signals in damage level 2 are classified into levels 1 and 3.

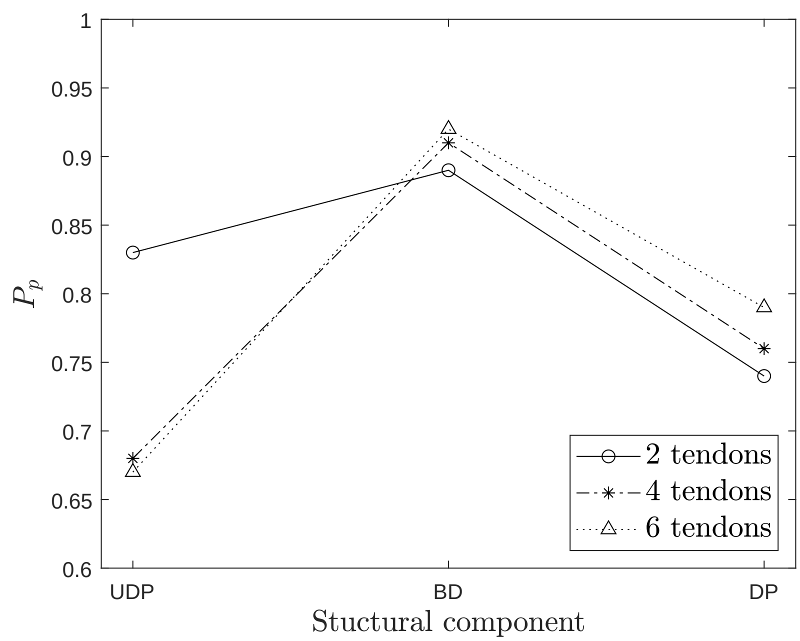

FIG. 17: Damage localization for rupture of tendons, where the legend shows the number of ruptured tendons.

TABLE 10: Confusion matrix for a rupture of tendons damage problem using the LSTM.

		Predicted Class			
		Pred-0	Pred-1	Pred-2	Pred-3
True Class	True-0	96	0	0	0
	True-1	1	95	0	1
	True-2	2	5	84	6
	True-3	7	2	6	82

The evaluation metrics (the mean values of five independent trials) are finally reported in Table 12. It can be observed that, for both types of multiclass damages, the proposed LSTM outperforms 1D CNN.

TABLE 11: Confusion matrix for a rupture of tendons damage problem using the 1D CNN.

		Predicted Class			
		Pred-0	Pred-1	Pred-2	Pred-3
True Class	True-0	93	0	0	3
	True-1	0	83	11	3
	True-2	1	19	14	63
	True-3	1	9	20	67

TABLE 12: Evaluation metrics of the LSTM and 1D CNN for the Z24 bridge.

Lowering of pier							
Method	Accuracy	Precision	Recall	F1 Score	FPR	FNR	ROC-AUC
LSTM	0.94	0.95	0.94	0.94	0.02	0.06	0.99
1D CNN	0.85	0.86	0.85	0.85	0.04	0.15	0.97
Rupture of tendons							
Method	Accuracy	Precision	Recall	F1 Score	FPR	FNR	ROC-AUC
LSTM	0.88	0.89	0.88	0.88	0.04	0.12	0.98
1D CNN	0.67	0.63	0.67	0.64	0.11	0.33	0.92

4. CONCLUSIONS

In this paper, damage localization using a windowed LSTM-based deep learning algorithm is employed for multiclass and multilevel damage detection. Various classes starting from binary to a maximum of five classes were classified into multiclass damage level. Limited dataset is augmented using windowing of the time-series measurements and the prediction accuracy is improved by a novel voting approach on windowed data. It is observed that the proposed algorithm performs well with nonlocalized and irregular sample sizes, and learns the long-term dependencies. The proposed algorithm is analyzed with sensitivity analysis on window-size as the external parameter to the model. A parametric study is also presented for random initialization of weights. The accuracy improvement of the proposed algorithm is illustrated through a comparison between a single time-series and windowed-voted time-series for ROC and precision-recall AUC. In this paper, it is demonstrated that a simple LSTM architecture is capable of classifying the time-series signals into multiclass and multidamage levels with high accuracy.

Traditional unsupervised methods rely on the appropriate selection of model orders and extraction of useful features that involved significant user discretion for a large-scale civil structure. On the other hand, the proposed deep learning model based on sequential data is independent of any feature selection process and offers robust, accurate and autonomous approaches to complex damage localization. Like any other supervised techniques, the LSTM method also requires a significant amount of training data to classify and predict the damage. With the recent advancement of remote and autonomous sensors and internet-of-things, long-term SHM technologies have shown significant promise to monitor critical infrastructure in smart cities (Talari et al. 2017). Such long-term and real-time monitoring allows the SHM researchers and practitioners to provide low-cost periodical SHM data with multiclass health conditions of the structures (Mishra et al. 2020), serving as the potential training data for the deep learning techniques, such as, CNN and LSTM.

ACKNOWLEDGEMENTS

The authors would like to acknowledge funding provided to the first author through Government of Ontario, Canada as Ontario Graduate Scholarship (OGS), and Queen Elizabeth II Graduate Scholarship in Science and Technology (QEGS-II) for PhD studies. The authors would also like to acknowledge funding provided to the second author by the Natural Sciences and Engineering Research Council of Canada (NSERC) [grant number RGPIN-2017-04755]. The authors also acknowledge NSERC and Western University for providing the financial support to conduct this research through the corresponding author's NSERC Discovery grant [grant number RGPIN-2015-6157] and NSERC Accelerator grant, respectively.

REFERENCE

1. Abdeljaber, O., Avci, O., Kiranyaz, M. S., Boashash, B., Sodano, H., and Inman, D.

J. (2018). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. Neurocomputing, 275, 1308-1317.

2. Abdeljaber, O., Avci, O., Kiranyaz, S., Gabbouj, M., and Inman, D. J. (2017). Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. Journal of Sound and Vibration, 388, 154-170.

3. Azimi, M., Eslamlou, A. D., and Pekcan, G. (2020). Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review, Sensors, 20.

4. Barbosh, M., Singh, P., and Sadhu, A. (2020). Empirical mode decomposition and its variants: a review with applications in structural health monitoring. Smart Materials and Structures, 29(9), 093001.

5. Bergstra, J., and Bengio, Y. (2012). Random search for hyper-parameter optimization. The Journal of Machine Learning Research, 13(1), 281-305.

6. Cawley, P. (2018). Structural health monitoring: Closing the gap between research and industrial deployment. Structural Health Monitoring, 17(5), 1225-1244.

7. Cha, Y-J., Choi, W., and Buyukozturk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. Computer-Aided Civil and Infrastructure Engineering, 32(5), 361-378.

8. Chen, G. (2016). A gentle tutorial of recurrent neural network with error backpropagation, preprint arXiv:1610.02583.

9. Erazo, K., Sen, D., Nagarajaiah, S., and Sun, L. (2019). Vibration-based structural health monitoring under changing environmental conditions using Kalman filtering. Mechanical Systems and Signal Processing, 117, 1-15.

10. Gatti, M. (2019). Structural health monitoring of an operational bridge: A case study. Engineering Structures, 195, 200-209.

11. Gulgec, N. S., Tak, M., and Pakzad, S. N. (2017). Structural damage detection using convolutional neural networks. Conference Proceedings of the Society for Experimental Mechanics Series, 331-337.

12. Guo, J., Xie, X., Bie, R., and Sun, L. (2014). Structural health monitoring by using a sparse coding-based deep learning algorithm with wireless sensor networks. Personal and Ubiquitous Computing, 18(8), 1977-1987.

13. Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

14. Ioffe, S., and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. preprint arXiv:1502.03167.

15. Kaveh, A., Javadi, S. M., and Maniat, M. (2014), Damage assessment via modal data with a mixed particle swarm strategy, ray optimizer, and harmony search, Asian Journal of Civil Engineering, 15(1), 95-106.

16. Kaveh, A., Eslamlou, A. D., Javadi, S. M., and Malek, N. G. (2021), Machine learning regression approaches for predicting the ultimate buckling load of variable-stiffness composite cylinders, Acta Mechanica, 232, 921-931.

17. Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., and Inman, D. J. (2019). 1D Convolutional Neural Networks and Applications: A Survey, Mechanical Systems and Signal Processing, 151, 107398

18. Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.

19. Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

20. Liang, X. (2019). Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with Bayesian optimization. Computer-Aided Civil and Infrastructure Engineering, 34(5), 415-430.

21. Lin, Z., Liu, Y., Zhou, L. (2019). Damage Detection in a Benchmark Structure Using Long Short-term Memory Networks, Chinese Automation Conference, IEEE, 22-24 Nov, Hangzhou, China.

22. Liu, T., Xu, H., Ragulskis, M., Cao, M., and Ostachowicz, W. (2020). A data-

driven damage identification framework based on transmissibility function datasets and one-dimensional convolutional neural networks: Verification on a structural health monitoring benchmark structure. Sensors, 20(4), 1-25.

23. Maeck, J., and Roeck De. G. (2003), Description of Z24 benchmark, *Mechanical Systems and Signal Processing*, 17(1), 127-131.

24. Mishra, D., Das, G., and Das, D. (2020). An IoT based building health monitoring system supported by cloud. Journal of Reliable Intelligent Environments, https://doi.org/10.1007/s40860-020-00107-0.

25. Ni, F. T., Zhang, J., and Noori, M. N. (2020). Deep learning for data anomaly detection and data compression of a long-span suspension bridge. Computer-Aided Civil and Infrastructure Engineering, 35(7), 685-700.

26. Okayasu, M., and Yamasaki, T. (2019). Structural Health Monitoring System for Remote Inspection of Material Failure. Journal of Nondestructive Evaluation, 38(2), 1-6.

27. Perez-Ramirez, C. A., Amezquita-Sanchez, J. P., Valtierra-Rodriguez, M., Adeli, H., Dominguez-Gonzalez, A., and Romero-Troncoso, R. J. (2019). Recurrent neural network model with Bayesian training and mutual information for response prediction of large buildings. Engineering Structures, 178, 603-615.

28. Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., and Iyengar, S. S. (2019). A Survey on Deep Learning. ACM Computing Surveys, 51(5), 1-36.

29. Qarib, H., and Adeli, H. (2014). Recent advances in health monitoring of civil structures. Scientia Iranica, 21(6), 1-30.

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

31. Roeck, De. G., and Teughels, A. (2004), Structural damage identification of the highway bridge Z24 by FE model updating, Journal of Sound and Vibration, 278, 589-610.

32. Sadhu, A., Narasimhan, S., and Antoni, J. (2017). A review of output-only structural mode identification literature employing blind source separation methods. Mechanical Sys-

tems and Signal Processing, 94, 415-431.

33. Salehi, H., and Burgueo, R. (2018). Emerging artificial intelligence methods in structural engineering. Engineering Structures, 171, 170-189.

34. Sharma, S., and Sen, S. (2020). One-dimensional convolutional neural network-based damage detection in structural joints. Journal of Civil Structural Health Monitoring, 0123456789.

35. Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556.

36. Singh, P., and Sadhu, A. (2020). System identification enhanced visualization tool for infrastucture monitoring and maintenance. Frontier in Built-up Environment. 6(76).

37. Singh, P., Keyvanlou, M., and Sadhu, A. (2021). An improved time-varying empirical mode decomposition for structural condition assessment using limited sensors. Engineering Structures. 232, 111882.

38. Singh, P., and Sadhu, A. (2021). Limited sensor-based bridge condition assessment using vehicle-induced nonstationary measurements. Structures. 32, 1207-1220.

39. Sirca, G. F., and Adeli, H. (2012). System identification in structural engineering. Scientia Iranica, 19(6), 1355-1364.

40. Sony, S., and Sadhu, A. (2020). Synchrosqueezing transform-based identification of time-varying structural systems using multi-sensor data. Journal of Sound and Vibration, 486, 115-576.

41. Sony, S., Dunphy, K., Sadhu, S., and Capretz, M. (2021). A Systematic Review of Convolutional Neural Network-based Structural Condition Assessment Techniques. Engineering Structures, 226, 111347.

42. Sony, S., Laventure, S., and Sadhu, A. (2019). A literature review of next-generation smart sensing technology in structural health monitoring. Structural Control and Health Monitoring, 26(3), e2321.

43. Sun, L., Shang, Z., Xia, Y., Bhowmick, S., and Nagarajaiah, S. (2020). Review of Bridge Structural Health Monitoring Aided by Big Data and Artificial Intelligence: From Condition

Assessment to Damage Detection. Journal of Structural Engineering, 146(5).

44. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A. (2014). Going Deeper with Convolutions, arxiv:1409.4842.

45. Talari, S., Shafie-khah, M., Siano, P., Loia, V., Tommasetti, A., and Catalao, J. (2017). A Review of Smart Cities Based on the Internet of Things Concept. Energies, 10(421).

46. Ye, X. W., Jin, T., and Yun, C. B. (2019). A review on deep learning-based structural health monitoring of civil infrastructures. Smart Structures and Systems, 24(5), 567-586.

47. Ying, Y., Garrett, J. H., Oppenheim, I. J., Soibelman, L., Harley, J. B., Shi, J., and Jin, Y. (2013). Toward data-driven structural health monitoring: Application of machine learning and signal processing to damage detection. Journal of Computing in Civil Engineering, 27(6), 667-680.

48. Zeiler, M.D., and Fergus, R. (2013). Visualizing and understanding convolutional networks. arXiv:1311.2901v3.

49. Zhang, A., Wang, K. C. P., Li, B., Yang, E., Dai, X., Peng, Y., and Chen, C. (2017). Automated Pixel-Level Pavement Crack Detection on 3D Asphalt Surfaces Using a Deep-Learning Network. Computer-Aided Civil and Infrastructure Engineering, 32(10), 805-819.

50. Zhang, J., Cao, X., Xie, J., and Kou, P. (2019). An Improved Long Short-Term Memory Model for Dam Displacement Prediction. Mathematical Problems in Engineering, 6792189.

51. Zhang, Y., Miyamori, Y., Mikami, S., and Saito, T. (2019). Vibration-based structural state identification by a 1-dimensional convolutional neural network. Computer-Aided Civil and Infrastructure Engineering, 34(9), 822-839.

52. Zhao, X., Li, S., Su, H., Zhou, L., and Loh, K. J. (2018). Image-based comprehensive maintenance and inspection method for bridges using deep learning. ASME 2018 Conference on Smart Materials, Adaptive Structures and Intelligent Systems, SMASIS 2018, 2, 1-7.