Journal of Materials and Engineering Structures

Research Paper

Multi-level damage detection using a combination of deep neural networks

Dung Bui-Ngoc [a]***, Hieu Nguyen-Tran*** [a,*]***, Hoa Tran-Ngoc*** [b]***, Lan Nguyen-Ngoc*** [b]***, Huan-Vu*** [a]***, Thanh Bui-Tien*** [b]***, Magd Abdel Wahab*** [c]

[a] *Faculty of Information Technology, University of Transport and Communications, Hanoi, Vietnam*

[b] *Faculty of Civil Engineering, University of Transport and Communications, Hanoi, Vietnam*

[c] *Faculty of Engineering and Architecture, Ghent University, Belgium*

A R T I C L E I N F O

Article history:

Received : 15 November 2022

Revised : 19 December 2022

Accepted : 23 December 2022

Keywords:

Structural Health Monitoring

Damage detection

Time series analysis

Deep neural networks

A B S T R A C T

In recent years, bridge damage identification using a convolutional neural network (CNN) has become a hot research topic and received much attention in the field of civil engineering. Although CNN is capable of categorizing damaged and undamaged states from the measured data, the level of accuracy for damage diagnosis is still insufficient due to the tendency of CNN to ignore the temporal dependency between data points. To address this problem, this paper introduces a novel hybrid damage detection method based on the combination of CNN and Long Short-Term Memory (LSTM) to classify and quantify different levels of damage in the bridge structure. In this method, the CNN model will be used to extract the spatial damage features, which will be combined with the temporal features obtained from Long Short-Term Memory (LSTM) model to create the enhanced damage features. The combination successfully strengthened the damage detection capability of the neural network. Moreover, deep learning is also improved in this paper to process the acceleration-time data, which has a different amplitude at short intervals and the same amplitude at long intervals. The empirical result on the Vang bridge shows that our hybrid CNN-LSTM can detect structural damage with a high level of accuracy.

F. ASMA & H. HAMMOUM (Eds.) special issue, 4th International Conference on Sustainability in Civil Engineering ICSCE 2022, Hanoi, Vietnam, J. Mater. Eng. Struct. 9(4) (2022)

1 Introduction

Civil structures are subjected to damage because of long services, natural disasters, human impacts, and so on. Structural Health Monitoring (SHM) is a vital tool to predict structural behaviors, identify damages, and enhance the

* *Corresponding author. Tel.: +849 12554558.*
E-mail address: nthieu@utc.edu.vn

e-ISSN: 2170-127X, (cc) BY-SA

RESEARCH REVIEW of Sciences and Technologies

operational efficiency of the structures. In the early period, visual-based inspection methods were commonly employed. However, to obtain the damaged properties, some impacts on the structures such as cutting, and sewing were required. Furthermore, this method was impossible to detect damages located deeply in the structures. To remedy this shortcoming, vibration based SHM has been developed. With the recent development of deep learning techniques, vibration data of large structures might be analyzed in an inexpensive way using convolutional neural networks (CNNs). A convolutional neural network (CNN) utilizes local connections to extract spatial information effectively and shared weights to reduce the number of parameters significantly [1]. In 1990, Yann LeCun proposed the Backpropagation (BP) algorithm to train the first CNN which became mature in 1998 [2]. According to LeCun, there are two main layers in CNN's architecture: convolution and pooling. These two layers can be connected to other fully connected layers. From these layers, 2D (two-dimensional) featured maps can be extracted. One of the main edges of CNN is that it can learn relevant functions and share parameters from the data provided. This helps to reduce the computational cost of CNN significantly in comparison with other categories of neural networks. Krizhevsky is the first to propose a deep CNN model called AlexNet [3], which is the ancestor of the Deep Learning paradigm. AlexNet had eight layers which consist of 5 convolutional-pooling layers and 3 fully connected layers. It has been proven powerful, with an error rate of 16.4% in the ImageNet test set [4], about 10% less than the second method using the traditional ML approach. Commonly CNN takes as input a 2D matrix, but it can also use a one-dimensional (1D) matrix as input. The resulting model is called a 1D CNN, which still has all the existing advantages of CNN [5]. Many researchers have shown that 1D CNN can be useful when handling time-series data in SHM to reduce computing costs, as 1D CNN only uses table functions for forward and backward calculations. CNN's main advantage over its predecessors is that it automatically detects relevant functions without human supervision [6]. For example, Dang et al. [7] proposed using 1DCNN combined with Long Short-Term Memory (LSTM) to monitor the health of structures. Both numerical and measured models were employed to consider the effectiveness of the proposed approach. Results showed that the 1DCNN-LSTM model successfully provides high accuracy outcomes as two-dimensional CNN. Moreover, it also requires less computational time as well as CPU memory. A deep CNN was used to analyze the reduction in connection stiffness of truss members of a large-scale truss bridge [8]. To consider the effect of the imperfection of data, 10 % of noise was considered. Nguyen et al. [9] monitored the prestress-loss of reinforced concrete girders using a proposed 1DCNN based on a regression method. A supervised Deep Neural Network (DNN) was successfully applied for damage identification of a full-scale bridge in the work of [10].

In recent years, the 1D-CNN has been proven to be highly effective in obtaining temporal information to solve damage identification problems using vibration measurement data. For example, 1D CNN is proposed by Abdeljaber [11] to deal with real-time vibration-based damage detection. The authors trained 1D-CNN on a database of vibration signals taken from a bundle called Qatar Grandstand that destroys individual joints and preserves other joints intact. The model was trained separately in each binding and had an outstandingly high classification accuracy. While the proposed method is effective in dealing with single class damage case, it was not tested for a multiclass damage scenario [7]. Zhang used the computing power of 1D CNN to identify structural parameters' changes such as mass and stiffness. Data collections were conducted on three types of structures: a steel T-beam, a short and a long steel beam bridge. The proposed method was able to achieve a high level of accuracy, with an average slope accuracy of 98% [8]. In another study, Ni demonstrated the ability of 1D CNN with an automated encoder to detect data compression anomalies. The proposed algorithm was validated with 97.53% accuracy using a long-span suspension bridge [9]. Sharma and Sen further utilized 1D CNN for the detection of structural frame damage [12]. The model was validated on a two-dimensional steel frame for different damage scenarios. The method was found to be highly accurate in locating and identifying the level of damage in the generated damage scenarios. Furthermore, the resulting false-positive rate was deemed to be within acceptable ranges [10]. In addition, Liu proposed a novel method that integrates traditional TF methods with the ability of neural networks [13]. The author used the 1D CNN-based transmissibility function to effectively detect damage to the ASCE SHM reference structure. The author compared the proposed method with time series and frequency field information based on the Fast Fourier Transform (FFT), in which TF signals showed more damage-sensitive features [11]. Recently, Bao introduced a hybrid method combining FE method and 1D CNN to identify damage to a jacket-type offshore structure. However, the data were generated synthetically using a finite element model, which is not necessarily like real-time data from functional and environmental noises. Since the damage was generated by the FE model, it did not reflect correctly the real-life data [12]. In another work, Sarawgi applied 1D CNN to detect simulated hypothetical damage of an experimental pipeline structure by adjusting structural external reinforcement [14]. However, CNN's unique model for the damaged node could not be well adapted to actual multi-class damage structures due to the computational complexity. However, choosing the best network architecture and the requirement for a massive dataset continued to be difficult [15].

Recurrent neural network (RNN) is a subclass of ANN in which neurons from the previous level get feedback from the output neurons in the current layer. RNN is known to be able to process long input data. The size of the model does not increase with the number of inputs. During the computational process, data from previous points is used. The weights are shared during processing [5]. For processing temporal data, long short-term memory networks (LSTMs), a significant subset of RNNs, are frequently used. LSTM networks have emerged as one of the most cutting-edge approaches for addressing a variety of machine learning issues, including computer vision, time series prediction, and natural language processing. However, LSTM networks is still new in SHM [16]. Recently, Zhang proposed the Long Short-Term Memory (LSTM) model to predict dam displacement. The authors make use of the long-term dependency learning ability in LSTM model. Several modifications have also been brought about by external environmental factors such as water pressure, temperature, structural deterioration, and bedrock damage. The LSTM model was optimized during the investigation to demonstrate how the external environment affected the resulting displacement. The suggested technique was evaluated against different machine learning (ML) algorithms, including the support vector machine, multilayer perceptron, and multiple linear regression. The proposed algorithm was compared with various ML algorithms, such as the support vector machine, multilayer perceptron, multiple linear regression, and augmented regression tree. LSTM has been shown to outperform other methods in terms of accuracy, effectively reflecting delays and making variable selection more convenient [13]. In addition, Yang introduced CNN-LSTM model for computer vision-based modal frequency detection. Spatio-temporal information was extracted from images captured by a standard camera. The proposed technique achieved 96.6% accuracy and performed well with noisy data sets [17].

This paper proposes a novel hybrid damage detection method based on a combination of CNN and long-term memory (LSTM) to classify and quantify different levels of damage to bridge structures. In our method, the CNN model is used for a spatial damage characteristic which, in combination with the temporal characteristics obtained from the Long Short-Term Memory (LSTM) model, creates improved damage characteristics. The proposed method is field tested in the Vang Bridge to validate its effectiveness. Results show that our CNN-LSTM hybrid is capable of detecting structural damage with high accuracy.

2 Methodology: Deep learning-based damage detection

Recently, deep learning (DL) techniques have been widely developed and applied in damage detection [18-20]. Compared to traditional machine learning methods, DL has two excellent features: feature learning and scalability. The performance of the feature learning depends essentially on craft functions that require domain knowledge and preprocessing data, while the scalability can exigently extract significant functionality from raw data due to the multilevel architecture and lower levels that essentially define the functions and compile higher-level models [21].

In SHM, deep learning techniques have the potential to generate a complete system since they require no additional signal preprocessing procedures. In addition, the DL models can be used for a variety of activities, such as damage identification, localization, damage severity, etc., by optimizing some of the latest levels of architecture or using learning transfer techniques. As a result, many researchers have recently applied different neural networks to SHM, such as Convolution Neural Networks (CNN), Recurrent Neural Networks (RNN), and multi-layer perceptron (MLP).

2.1 Convolution neural network

A convolutional neural network is one kind of deep neural network model. An input layer, convolution layers, pooling layers, a fully linked layer, and an output layer make up the CNN structure. The convolution layer is one of the most important layers, which can be used to extract features from the previous layers. Convolutional kernels are learned in order to extract these features. The input will be convolved with the convolutional kernel for this task, and the output will go through an element-wise nonlinear activation function after that. Equation 1 illustrates the precise computation.

$$z_{i,j,k}^{l} = w_k^{l^T} x_{i,j}^{l} + b_k^{l} \tag{1}$$

where $z_{i,j,k}$ is the feature value at the location (I, j) in the k^{th} feature map of l^{th} layer. $X_{i,j}$ is the input patch with center (I, j) of the l^{th} layer. The z_k and b_k are the weight and bias vector of the k^{th} filter of the l^{th} layer. The advantage of CNN is the ability to extract important features from data without any human supervision. The activation function is then applied to the convolution feature. Here activation function can be sigmoid, tanh, and ReLU. The ReLU function is more commonly used and is empirically proven to be one of the standard choices. The ReLU function will assign negative values to zero and keep the value of the input when greater than 0.

$$a_{i,j,k}^{l} = \mathrm{a}(\mathrm{z}_{i,j,k}^{l}) \tag{2}$$

CNN uses the pooling layer to decrease the feature map while preserving important information from the data, thus reducing the number of parameters as well as computational complexity. It works by summarizing the features while swiping a filter across the feature map. The pooling processes may involve average or maximum pooling. Equation (3) presents the pooling for the feature map:

$$y_{i,j,k}^{l} = \mathrm{pool}(\mathrm{a}_{\mathrm{m,n},k}^{l}), \forall(\mathrm{m,n}) \in \mathrm{R_{ij}} \tag{3}$$

The abstract features are extracted following the convolution process and the pooling operation. If more layers are used, we can get higher-level feature representations. Before being sent to the fully connected layer, the feature map will be flattened to a vector with size as the product of the feature map dimensions. In the CNNs network model, fully connected layers (FC) are usually found at the end of the network and are used to optimize network goals such as prediction accuracy. The fully connected layer takes as input flattened data, each of which is connected to all neurons. The feature matrix from the preceding layer is transformed into a vector in this layer that contains the probability of the items that need to be predicted. Finally, a loss function will be utilized to determine the model's prediction error and the backpropagation technique will be used to update the weights during the training of a CNN model for the classification issue. Consider θ the weight and bias of a CNN, also called as parameters. The optimization parameters for the classification problem can be calculated by minimizing the loss function of the task. Suppose N is the number of input-output pairs $\{(x^{(i)},y^{(i)}),\ i \in [1 \cdots N]\}$, where x^i is the input data, y^i is its corresponding label. Let o^i is the output of CNN. The loss function of CNN can be calculated as:

$$\mathrm{L} = \frac{1}{N}\sum_{i=1}^{N} l\left(\theta, y^{(i)}, \mathrm{o}^{(i)}\right) \tag{4}$$

The stochastic gradient descent algorithm can be used to minimize the loss function. Optimizing the model's weights and bias to the trained dataset is the goal of model training.

2.2 Long Short-Term Memory network

Long Short-Term Memory networks were designed to capture the long-term relationship between data points. By using a feedback connection, LSTMs can process the entire data sequence without processing each time point in the sequence independently. Therefore, it can retain useful information about previous data points in the sequence. To regulate the data in the data sequence moving through the network, LSTMs employ three gates: the forget gate, the input gate, and the output gate. To control the cell state, these gates consist of a layer of sigmoid neural networks and a pointwise multiplication operation. The forget gate is the first step in information processing, and it determines what information about the cell state from the previous hidden state and new input data will be kept. Considering the previously hidden state h_{t-1} and input x_t, for each number in the cell state C_{t-1}, the network's sigmoid function will train and output a value between 0 and 1. Here the number closer to 1 means the information is relevant and the number closer to 0 means irrelevant. The output then multiplies pointwise with the previous cell state C_{t-1}, which leads to having less influence in the next step if the output is close to 0.

$$\mathrm{f}_t = \sigma\left(\mathrm{W}_f.[\mathrm{h}_{t-1},\mathrm{x}_t]+\mathrm{b}_f\right) \tag{5}$$

$$i_t = \sigma\left(\mathrm{W}_i.[\mathrm{h}_{t-1},\mathrm{x}_t]+\mathrm{b}_i\right) \tag{6}$$

To choose what fresh data will be kept in the cell state C_t, the old cell state C_{t-1} will be multiplied with f_t and then add up with $i_t * /C_t$, where i_t is the output of the sigmoid layer containing the values will update, $/C_t$ is the output of tanh layer, which has learned how to combine the previous hidden state and the input data to produce a vector of new candidate values.

$$\tilde{C}_t = \tanh\left(W_C.\left[h_{t-1},x_t\right]+b_C\right) \tag{7}$$

This vector shows the way to update each component of the long-term memory of the network given a new data point. As the result of the input gate, the C_t here contains the updated state values.

$$C_t = f_f * C_{t-1} + i_t * \tilde{C}_t \tag{8}$$

Finally, the output gate will select the new hidden state based on C_t, h_{t-1}, and the input data. To obtain the filter vector, the hidden state h_{t-1} and input data are sent through the sigmoid activated neural network. Also, the cell state is gone through the tanh function to range the value between -1 and 1. Finally, this value is multiplied by the filtered vector to obtain the desired output.

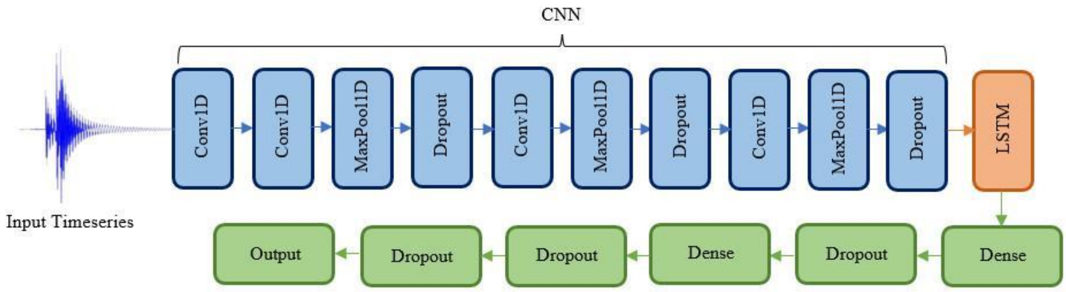

Fig. 1 – Hybrid CNN and LSTM for multiple damage detection

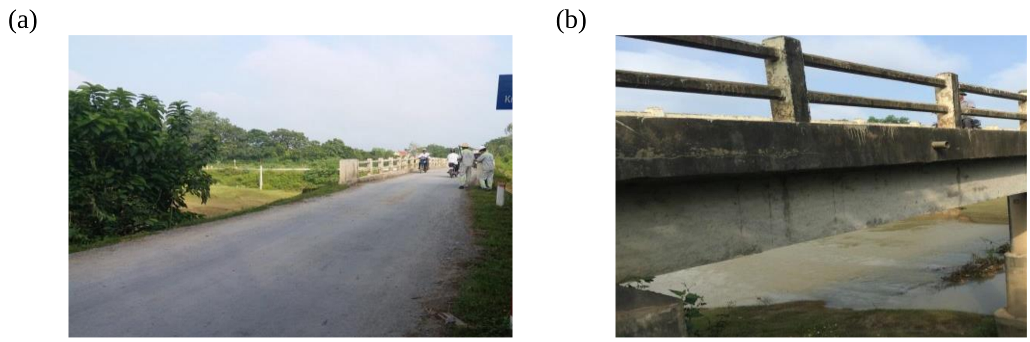

Fig. 2 – Vang bridge: (a) The top view of the bridge; (b) Downstream side of the bridge

2.3 Combine CNN+LSTM for damage detection

The proposed method includes two main steps. The method takes the raw time series signals as input. As explained in the aforementioned section, in the case of a vibration time series, CNN is unable to capture the temporal relationships between the time series's beginning and ending. This technique uses a CNN to extract the features, then an LSTM network to classify the data. The overall architecture of the method is shown in Figure 1. In our approach, the convolution neural network was fed the time series to automatically extract the features. Since our input is a 1D time series, we use a collection of 1D Convolutional layers, followed by Max Pooling layers. As in the figure, CNN uses the convolution layers to filter data and pooling layers to reduce the size of the feature map by replacing the sliding windows with a maximum value. The output of the pooling layer then goes through LSTM layers and multiple fully-connected layers. Finally, the output layer will be used for the classification task. To produce the multi-class classification results, we use the softmax function, which outputs the probability distribution for all the classes. The dropout layer is used here to avoid overfitting. The label with the maximum probability will be the damaged state prediction.

3 Experimental results

3.1 Data

To evaluate the effectiveness and accuracy of the proposed method, in this section, we conduct our experiments on a simply supported bridge. Vang bridge is situated in the middle of Vietnam at Km7+591, Yen Dinh district, Thanh Hoa Province. The bridge, which has 4 spans with a length of 15 m each, was erected and has been in use since 1995. Each span consists of four 1.15 m-long reinforced concrete T-shaped beams. The cross-section is one meter tall. Figure 2 shows some views of the bridge.

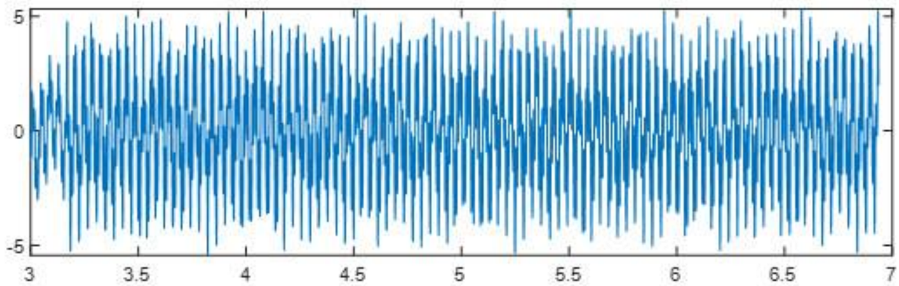

Fig. 3 – Acceleration of DOF number 10 when damage occurs at element 6 with a level of 30%.

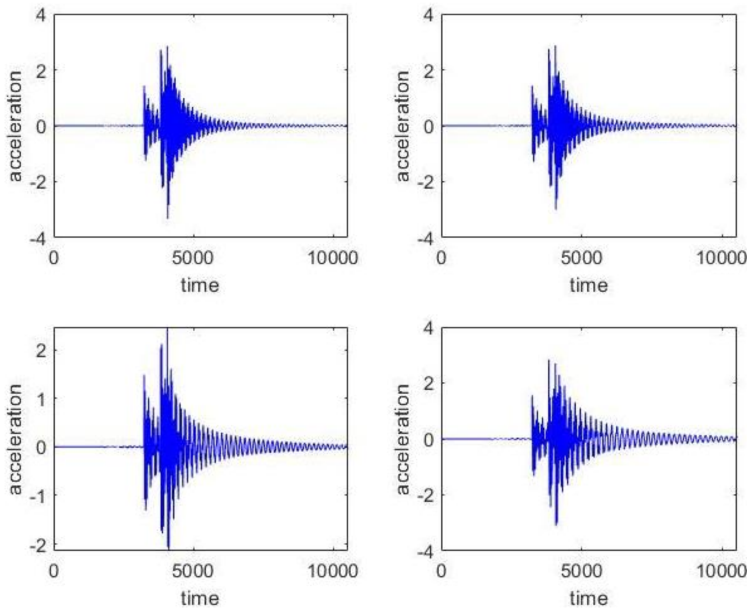

Fig. 4 – Sample data (a) undamaged, (b) damage level 1, (c) damage level 2, (d) damage level 3

To get the input data to train the network, a Finite Element Model (FEM) of the bridge is built first using MATLAB program. The model is divided into 30 beam elements, each element contains 6 degrees of freedom (DOF) at each node that consists of 3 translational displacements and rotational displacements around the x, y, and z-axis.

From the built FEM, a wide range of scenarios of structural health situations is generated. Damages are generated by reducing the stiffness of elements of the structure. Both single and multiple damages are considered. For single damage, the stiffness of each element is reduced from 0 % to 50 %, meanwhile, other elements are intact. For multiple damages, we consider damages that occur at two elements at the same time with a reduction in the stiffness from 0% to 50%. After damage scenarios are determined, data is collected. To train the network of CNN, image data of acceleration is employed. For instance, Figure 3 shows the acceleration of DOF number 10 when damage occurs at element 6 with a level of 30%.

The data has four classes, such as undamaged, damage level 1, damage level 2, and damage level 3. The data is generated from the FE model. In the model, in addition to the undamaged data, there is also damaged data at a location with a degree of < 30% will be labelled as class 1, and data with a damage level from 40 to 60% will be assigned labelled class 2, and data with more than 70% of the damage level will be labelled as class 3. A sample data description for the classes is shown below.

3.2 Results

Data are randomly distributed into a training and a test set. The ratio is 70% for training and 30% for testing. The architecture used is exactly like the one in Figure 1. Each time series consists of 10570 points.

We have evaluated our model with a set of configurations. Here we present the hyperparameters that has the best performance and can be used to reproduce our results. Our architecture has 4 convolutional layers. The number of filters for these layers are 512,256,256 and 128, respectively. The kernel size of the first layer is 5 and the kernel size of the other layers is 3. The three following dense layers have 512, 128 and 4 nodes, respectively. The optimizer used is RMSprop with a learning rate of 0.001. We train our model for a total of 500 iterations with batch size 32. Figure 5 shows the training and testing accuracy of the proposed architecture during training.

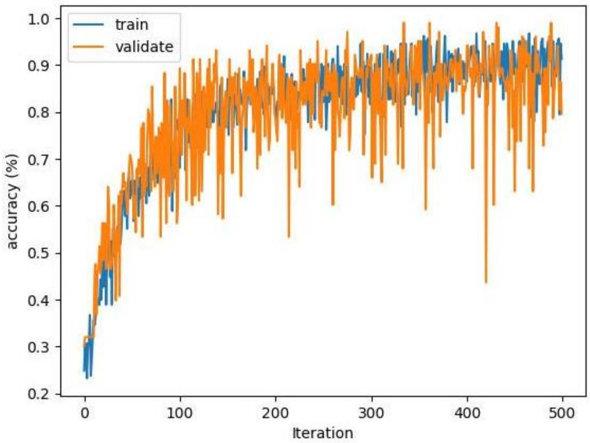

Fig. 5 – Performance of the model

As shown in figure 5, the model converges at around iteration 200. Here the training accuracy is linearly smooth while there is a higher variation for validation accuracy.

We evaluate the best validation model against the test set. The accuracy of the methods is computed using the ground truth data consisting of positive and negative detection. The confusion matrix for three methods CNN, CNN with handcrafted features and CNN with LSTM is shown in Table 1. The percentage of correctly classified samples relative to the total number of samples will be used to determine the method's accuracy.

$$Accuracy = \frac{(TN + TP)}{(FP + FN + TN + TP)} \tag{9}$$

where TP is the number true positive samples, TN the number of true negative samples. FP is the false positive number and FN the false negative number. Results are reported using the F-measure metric, which, when applied to unbalanced data, considers both false positive and false negative results.

$$F - Measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{10}$$

where:

$$Recall = \frac{TP}{FN + TP} \; ; \; Precision = \frac{TP}{TP + FP}$$

Figure 6 shows the performance of our approach when used as a simple damage detection model. For some simple use cases in real life, such binary classification model already provides sufficient prediction result. Our model shows great performance in this case, reporting an accuracy of 90.29% and an F-Measure of 0.93.

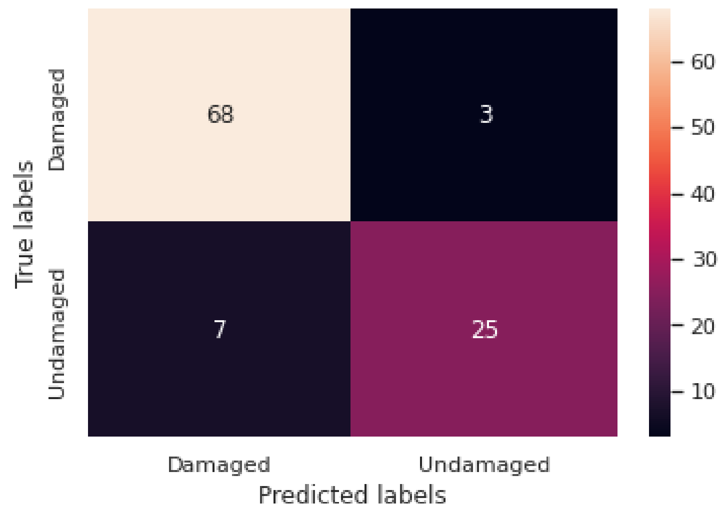

Fig. 6 – Confusion matrix of the binary classification use case

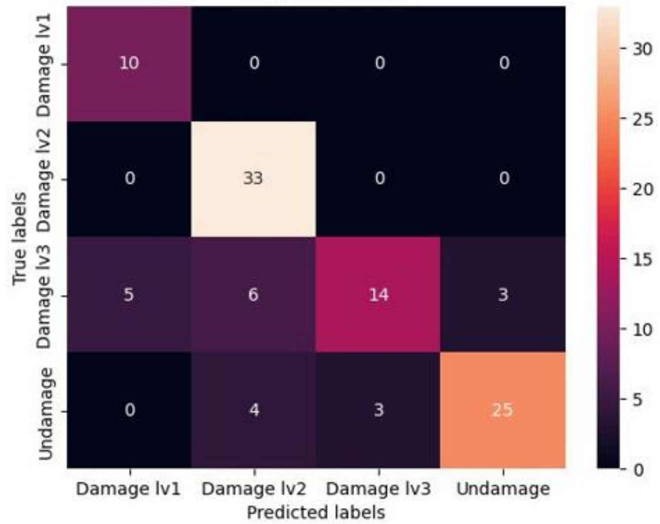

Fig. 7 – Confusion matrix of the multi-level damage detection use case

When used as a multi-level damage detection model, the testing accuracy of our method is 79.6%. The complete confusion matrix is shown in Fig. 7. The total number of test samples is 103, in which 32 are undamaged samples, 10 are damage level 1 samples, 33 are damage level 2; 28 are damage level 3, respectively. in the test phase, all damaged level 1 and level 2 samples were predicted accurately, and some undamaged and level 3 samples were misclassified. From the matrix, we get TP, TN, FP, FN which are 10, 72, 0, 21. Results are obtained for the accuracy, precision, recall, and F-measure scores, which are 0.79, 1.0, 0.32, 0.48, respectively. We observe that our model correctly identifies all Damage level 1 and Damage level 2 samples. However, Damage level 3 samples seem to be harder to identify. The overall accuracy still suggests that the transformed vibration data can be processed using CNN and LSTM in combination.

4 Conclusions

In this paper, we have proposed a damage identification method combining CNN and LSTM model for multiple damage detection. In this model, time series with multi-class labels are used as inputs for our model. Afterward, the spatial damage features from the CNN model are combined with the temporal features from the LSTM model to create enhanced damage features. The combined features are used to classify different levels of damage in the bridge. The technique can be used to address issues with damage identification, according to experimental data from a case study of the Vang bridge. The case study's final results indicate that the suggested strategy has created a high level of accuracy in the structure's damage identification. Moreover, the deep learning approach is also improved in this paper to process the acceleration-time data, which has a different amplitude at short intervals and the same amplitude at long intervals.

Acknowledgments

This research is funded by University of Transport and Communications (UTC) under grant number T2022-CN-002TĐ.

REFERENCES

[1]- A. Romero, C. Gatta, G. Camps-Valls, Unsupervised Deep Feature Extraction for Remote Sensing Image Classification. IEEE Transactions on Geoscience and Remote Sensing, 54(3) (2016) 1349-1362. doi:10.1109/TGRS.2015.2478379.

[2]- D. Bui-Ngoc, T. Bui-Tien, H. Nguyen-Tran, M. Abdel Wahab, G. De Roeck. Structural Health Monitoring Using Handcrafted Features and Convolution Neural Network. Singapore: Springer Singapore. (2021), 103-112. doi:10.1007/978-981-15-9121-1_8.

[3]- A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks. Commun. ACM, 60(6) (2017) 84–90. doi:10.1145/3065386.

[4]- J. Deng, W. Dong, R. Socher, L.J. Li, L. Kai, F.-F. Li. ImageNet: A large-scale hierarchical image database. in 2009 IEEE Conference on Computer Vision and Pattern Recognition. (2009), 248-255. doi:10.1109/CVPR.2009.5206848.

[5]- T. Bui-Tien, D. Bui-Ngoc, H. Nguyen-Tran, L. Nguyen-Ngoc, H. Tran-Ngoc, H. Tran-Viet, Damage Detection in Structural Health Monitoring using Hybrid Convolution Neural Network and Recurrent Neural Network. Frattura ed Integrità Strutturale, 16(59) (2021) 461-470. doi:10.3221/IGF-ESIS.59.30.

[6]- H. Nguyen-Tran, B.-T. Thanh, M.A. Wahab, B.-N. Dung, Damage detection in structural health monitoring using combination of deep neural networks. J. Mat. Eng. Str. «JMES», 7(4) (2020) 619-626.

[7]- H.V. Dang, H. Tran-Ngoc, T.V. Nguyen, T. Bui-Tien, G.D. Roeck, H.X. Nguyen, Data-Driven Structural Health Monitoring Using Feature Fusion and Hybrid Deep Learning. IEEE Transactions on Automation Science and Engineering, 18(4) (2021) 2087-2103. doi:10.1109/TASE.2020.3034401.

[8]- V.D. Hung, R. Mohsin, H. Tran-Ngoc, T. Bui-Tien, X.N. Huan, Connection stiffness reduction analysis in steel bridge via deep CNN and modal experimental data. Structural Engineering and Mechanics, An Int′l Journal, 77(4) (2021) 495-508. doi:10.12989/sem.2021.77.4.495.

[9]- T.-T. Nguyen, T. Tuong Vy Phan, D.-D. Ho, A. Man Singh Pradhan, T.-C. Huynh, Deep learning-based autonomous damage-sensitive feature extraction for impedance-based prestress monitoring. Eng. Struct., 259 (2022) 114172. doi:10.1016/j.engstruct.2022.114172.

[10]- A. Fernandez-Navamuel, D. Zamora-Sánchez, Á.J. Omella, D. Pardo, D. Garcia-Sanchez, F. Magalhães, Supervised Deep Learning with Finite Element simulations for damage identification in bridges. Eng. Struct., 257 (2022) 114016. doi:10.1016/j.engstruct.2022.114016.

[11]- O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D.J. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. J. Sound Vibrat., 388 (2017) 154-170. doi:10.1016/j.jsv.2016.10.043.

[12]- Y. Zhang, Y. Miyamori, S. Mikami, T. Saito, Vibration-based structural state identification by a 1-dimensional convolutional neural network. Copmut. Aided Civil Infrastruct. Eng., 34(9) (2019) 822-839. doi:10.1111/mice.12447.

[13]- T. Liu, H. Xu, M. Ragulskis, M. Cao, W. Ostachowicz, 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) (2020) 1059. doi:1–25.doi.org/10.3390/s20041059.

[14]- Y. Sarawgi, S. Somani, A. Chhabra, Dhiraj. Nonparametric Vibration Based Damage Detection Technique for Structural Health Monitoring Using 1D CNN. Singapore: Springer Singapore. (2020), 146-157.

[15]- F. Ni, J. Zhang, M.N. Noori, Deep learning for data anomaly detection and data compression of a long-span suspension bridge. Copmut. Aided Civil Infrastruct. Eng., 35(7) (2020) 685-700. doi:10.1111/mice.12528.

[16]- S. Sharma, S. Sen, One-dimensional convolutional neural network-based damage detection in structural joints. J. Civ. Str. Hea. Monit., 10(5) (2020) 1057-1072. doi:10.1007/s13349-020-00434-z.

[17]- X. Bao, T. Fan, C. Shi, G. Yang, One-dimensional convolutional neural network for damage detection of jacket-type offshore platforms. Ocean Eng., 219 (2021) 108293. doi:10.1016/j.oceaneng.2020.108293.

[18]- R. Yang, S.K. Singh, M. Tavakkoli, N. Amiri, Y. Yang, M.A. Karami, R. Rai, CNN-LSTM deep learning architecture for computer vision-based modal frequency detection. Mech.l Sys. Sig. Proc., 144 (2020) 106885. doi:10.1016/j.ymssp.2020.106885.

[19]- J. Zhang, X. Cao, J. Xie, P. Kou, An Improved Long Short-Term Memory Model for Dam Displacement Prediction. Mathematical Problems in Engineering, 2019 (2019) 6792189. doi:10.1155/2019/6792189.

[20]- R. Chalapathy, A.K. Menon, S. Chawla. Robust, Deep and Inductive Anomaly Detection. Cham: Springer International Publishing. (2017), 36-51.

[21]- J. Brownlee. What is deep learning? Retrieved from. 2019.