Robustness Test of a CNN-Based Structural State Identification Method

Kitami Institute of Technology ○ Student member Youqi Zhang
Kitami Institute of Technology Member Yasunori Miyamori
Kitami Institute of Technology Member Takehiko Saito
Kitami Institute of Technology Member Shuichi Mikami

1. Introduction

Neural networks are widely used in solving tough problems which are difficult to establish mappings directly from data to result by applying domain knowledge, since neural networks are purely data-driven methodology, with no requirement of domain knowledge. Thus, in civil engineering field, neural networks show potential to deal with the vibration-based structural damage detection (SDD) issue with no high-cost knowledge of dynamics.

It has already been validated that by applying convolutional neural networks (CNNs), which have been well trained through supervised learning algorithms by feeding one-to-one matched structural vibration data and labels with damage information into the CNNs and updating the inner parameters in sufficient iterations, structural damages such as stiffness and mass drop in a beam structure (Lin et al. 2017), loosen bolt of steel frame (Abdeljaber et al. 2017; Abdeljaber et al. 2018), and so on, can be identified with very high accuracies. The achievements show feasibility of the CNN-based SDD method. However, there is also obvious limitation of the supervised learning methods that only limited number of damaged patterns (limited locations and scales) of training data can be prepared. Considering time and labor costs, it is impossible to establish a database which contain all kinds of damaged pattern of a structure. Therefore, the result of predicting structural state by feeding a vibration data sample which is in an untrained category into a well-trained CNN is unknown. The robustness of the CNN-based SDD method need to be investigated.

In this paper, robustness tests of a CNN-based SDD method were conducted and discussed based on two databases of free damped vibration data which are acquired from a T-shape steel beam.

2. Vibration Experiments

The vibration experiments were conducted on a steel T-shape beam, as shown in Fig. 1. The steel beam is 2090 mm long, with 360 mm width flange. The thicknesses of the web and flange are 10 mm and 20 mm respectively. The height of the web is 390 mm. The beam is fixed by 8 bolts on both left and right ends (4 bolts on each side). Local structural damages were simulated by attaching a magnet in different locations to change local structural states slightly.

In total two vibration experiments were conducted on the beam to establish 2 databases. The 1st experiment is designed for validating the performance of the classification CNN model to identify different trained structural states, and the 2nd experiment is used for test the robustness of the CNN model when predicting data in untrained categories. The 1st and 2nd experiments contain 16670 and 2151 vibration tests respectively. The layout of the experiments is shown in Fig. 2. In total 9 accelerometers are installed on the surface of the flange to measure the vertical vibration, with 10000 Hz sampling frequency. The sampling period is set to 1.0 second. Thus, the shape of each datum is 9 Ch. × 10000. A magnet, weight of 1.37 kg, is attached on the web in different positions to change the local structural mass slightly. Comparing to the weight of the beam (182.11 kg), the local mass change is only 0.75% (1.37/182.11).

Fig.1. T-shape steel beam

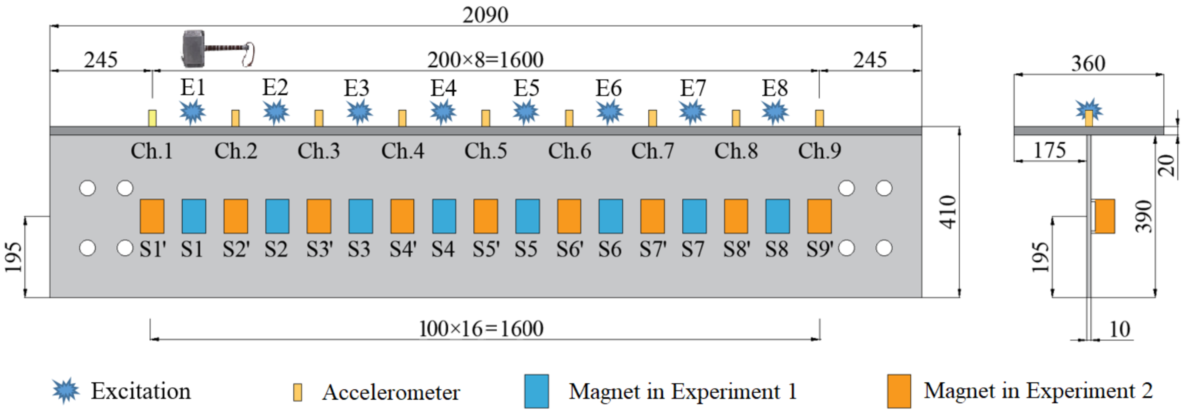

Fig. 2. Layout of the vibration experiments

Table 1. Data distribution of Database 1

Category	*Amount*	*Category*	*Amount*
State 0	1894	State 5	1861
State 1	1833	State 6	1858
State 2	1849	State 7	1831
State 3	1859	State 8	1825
State 4	1860		
In total			16670

Table 2. Data distribution of Database 2

Category	*Amount*	*Category*	*Amount*
State 1’	238	State 6’	239
State 2’	239	State 7’	239
State 3’	239	State 8’	238
State 4’	239	State 9’	240
State 5’	240		
In total			2151

Table 3. Configuration of the classification CNN models

Layer Number	*Layers*	*Output Shape*	*Parameter*	*Activation*	*Variables*
1	Input Layer	10000×9	None	None	0
2	Convolution 1-D	9991×5	Kernel number: 5; Kernel size: 10×9; Stride: 1; Padding: Valid	Linear	455
3	Batch normalization	9991×5	None	None	20
4	Max Pooling 1-D	3330×5	Kernel number: 3; Stride: 1	None	0
5	Flatten	16650	None	None	0
6	Dropout	16650	Rate: 0.25	None	0
7	Dense	40	None	ReLU	666,040
8 (Classification)	Output: Dense	18	None	Softmax	738
8’ (Regression)	Output’: Dense	2	None	Sigmoid	82

Table 4. Experimental procedures

Step	*Model*	*Training Data*	*Test data*
1	Classification	80% of Database 1	Other 20% of Database 1
2	Classification	Database 1	Database 2
3	Regression	Database 1	Database 2

In the 1st experiment, the magnet was attached in 8 different positions to create 9 structural states (S0: original state with no magnet attached, and S1 - S8: states with the magnet attached on the web between two adjacent accelerometers). In every structural state, more than 1800 vibration tests were conducted to generate the acceleration data. The detailed data distribution is show in Table 1. In the 2nd experiment, the locations of the magnet in each structural state were horizontally shifted to the locations under the accelerometers, and the corresponding structural states are named as S1’ - S9’. The detailed data distribution is shown in Table 2. There are about 240 data in each structural state. In total, 2151 tests were conducted in the 2nd experiment.

Hammer hitting impulse load is chosen as the excitation method in the experiments. After impacting the beam, free damped vibration of the beam was measured. There are 8 different excitation positions between every two adjacent accelerometers, as shown in Fig. 2. Data in every structural state contains the data excited in all 8 positions from E1 to E8. The inputs are in a range between 100 Gal to 1200 Gal to keep the variety of the database.

3. CNN Models, Trainings, and Validations

To validate that CNN is able to classify different structural states by feeding raw vibration data into the CNN, a simple classification CNN was designed as Table 3, with only one 1-D convolutional layer and one 1-D max pooling layer. The labels of the data for the classification CNN model are in one-hot key encoding, indicating different structural states. Categorical cross entropy and Adam are loss function and optimizer of the CNN respectively. As we know, classification neural networks are not able to predict data in untrained categories correctly. Thus, for the proposed classification CNN model for SDD in this paper, authors expect that the prediction of the data in untrained structural damage cases should be the closest trained structural damage categories.

To obtain a CNN which can have a better expression of the damage locations than the classification CNN model, a regression CNN model was proposed that the labels of the data and output layer of the CNN were redesigned. The new label consists 2 units. The first unit is a value between 0 and 1 for the confidence of structural damage. The second unit is a value greater than 0 and less than 1 representing relative damaged location of the beam. The far left of the beam is 0 and the far right of the beam is 1. The updated output layer has only 2 units corresponding to the new labels, and excited by the sigmoid function. Meanwhile, the loss function of the new regression CNN is changed to mean square error.

The trainings and validations of the proposed CNNs are conducted as following steps. Firstly, 80% of the Database 1 are used for training the classification CNN model, and the other 20% of the Database 1 is applied to test the performance of the CNN when predicting data in trained categories. Secondly, the classification CNN model is trained with the Database 1, and tested with the Database 2 to investigate the capacity of the classification CNN model when predicting data in untrained categories. Finally, the regression CNN model is trained by the Database 1, and tested by the Database 2, validating the performance of the regression CNN model when predicting data in untrained categories. Above 3 steps are summarized in Table 4.

4 Result

The results consist of 3 parts corresponding to the 3 steps introduced in Table 4. In the first step, 100% training accuracy and 99.8% test accuracy are obtained. There are 5 wrong

Table 5 Test result of Step 2

		Prediction											
		S0	*S1*	*S2*	*S3*	*S4*	*S5*	*S6*	*S7*	*S8*	*Overall*	*Accurate Prediction*	*Accuracy*
Label	*S1′*	5	1	0	49	1	5	0	177	0	238	1	0.4%
	S2′	0	13	35	2	90	0	91	0	8	239	48	20.1%
	S3′	143	0	0	74	2	0	0	18	2	239	74	31.0%
	S4′	0	9	0	0	227	1	0	1	1	239	227	95.0%
	S5′	164	1	0	70	1	2	0	0	2	240	3	1.3%
	S6′	34	2	0	88	19	45	51	0	0	239	96	40.2%
	S7′	11	0	1	2	0	0	0	28	197	239	28	11.7%
	S8′	237	0	0	0	0	0	0	1	0	238	1	0.4%
	S9′	30	1	0	8	0	0	0	201	0	240	0	0.0%
	Overall	624	27	36	293	340	53	142	426	210	2151	478	22.2%

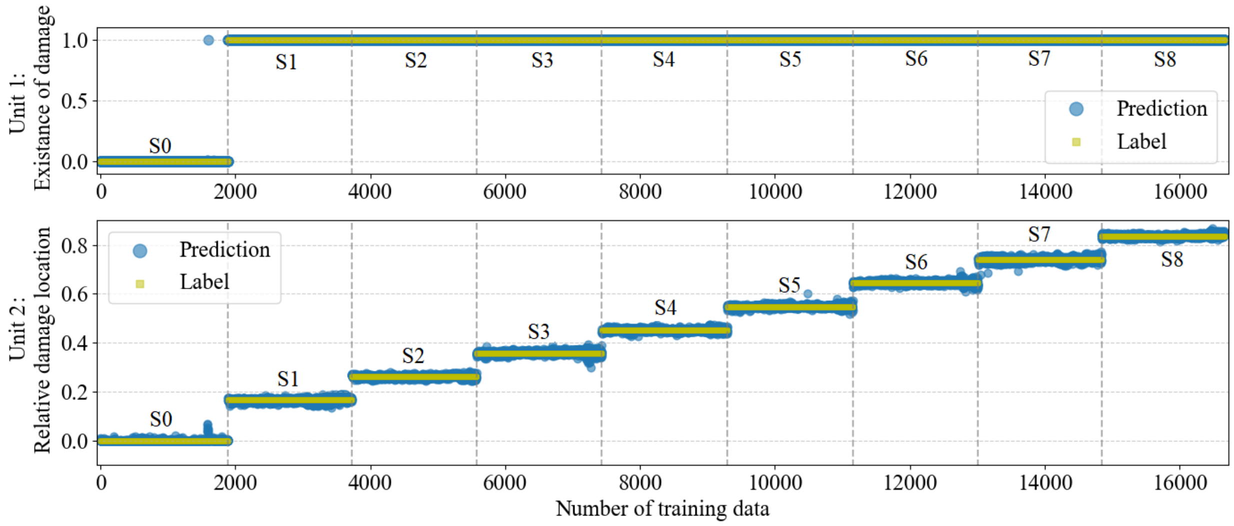

Fig. 3. Training result of Step 3

predictions of the test data. The result shows that the classification CNN model is able to define a clear boundary between data in each trained category and have a very good performance to identify the data in those trained categories.

In the second step, the classification CNN model was trained and tested with the Database 1 and Database 2 respectively. Thus, in the confusion matrix of the test result as shown in Table 5, only structural states of S1’ - S9’ are in the vertical axis, since only the data in the Database 2 is in the test set. Meanwhile, only structural states S0 - S8 are in the horizontal axis since the prediction of a classification CNN model can only in a trained category. Predictions were expected in the diagonal or the secondary diagonal in the confusion matrix (marked with underbars in Table 5) which could indicate an approximate location of damage. However, in Table 5, only data in S4’ are mainly predicted in expected positions. One third of the data in S3’ are predicted in S3, and the other two third are determined as S0 (intact). Predictions of data in S7’ have small error which are in S8, and predictions of S9’ data show small error which are mainly in S7. All other structural states (S1’, S2’, S5’, S6’, and S8’) shows very big error. In total, there are 478 accurate predications in the diagonal or the secondary diagonal in the confusion matrix. The overall accuracy of the prediction is only 22.2% (478/2151). Therefore, the classification CNN model is not feasible to predict vibration data in untrained structural state owning to the low accuracy.

In the third step, the regression CNN model was trained and tested with the Database 1 and Database 2 respectively. The training and test results are shown in Figs. 3-4. Fig. 3 shows that the regression CNN model has learned the features from the training data, since in Fig. 3 only values close to 0 or 1 are outputted in the Unit 1, and in the Unit 2 all the predicted damage locations are close to the actual locations.

Fig. 4 shows the test result of the predictions of the regression CNN model. Generally, in the Unit 1, most data were correctly predicted as damaged cases. However, a large number of data in S8’ are wrong predicted in intact state as negative errors. The mean error of the Unit 1 of the predictions is 0.15, which is calculated as the Eq. 1. For the Unit 2 of the predictions of the test data: relative location of the structural damage, data in S3’, S4’, and S7’ show very good result that the predictions are mainly in the areas of the labels, indicating that for the test data in S3’, S4’, and S7’, the regression CNN model can predict the correct damage locations even though those cases are not trained. The error of the Unit 2 of the predictions of the test data in S6’ and S5’ are about 0.2 and 0.4 which are obviously higher than test results of the data in S3’, S4’, and S7’. The big errors of the predictions are in S1’, S2’, S8’, and S9’ corresponding to the structural damages on the 2 ends of the beam, as shown in Fig. 5. The mean error of the predictions of relative damage location is 0.31 which is also computed as the Eq. 1.

$$E = \text{mean}\ (|\text{Prediction} - \text{Label}|) \qquad (1)$$

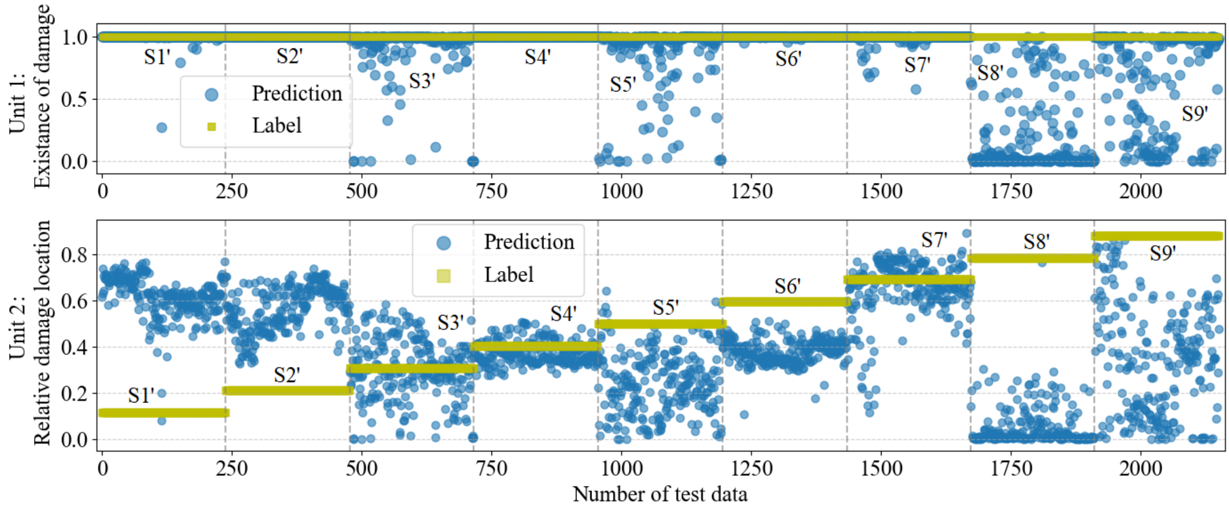

Fig. 4. Test result of Step 3

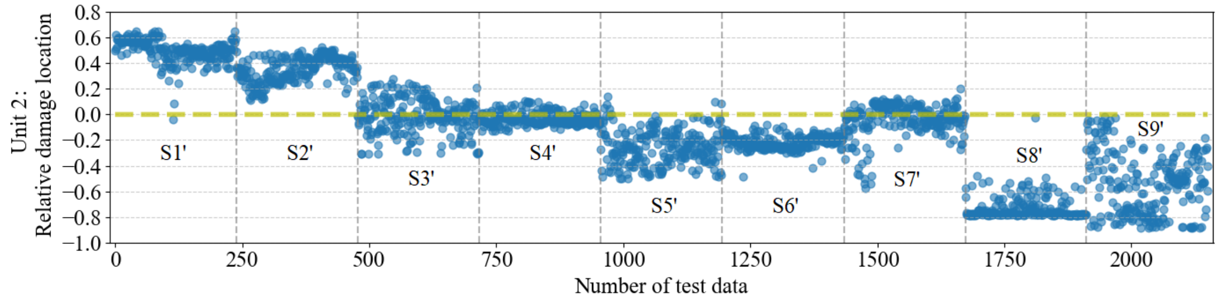

Fig. 5. Errors of the Unit 2 of the test result of Step 3

Comparing the predictions of the classification and regression CNN models, firstly, it is found that the classification model tends to predict more data as S0 (intact) state than the regression model; secondly, regression model has more correct predicted states than the classification model. Overall, the regression model shows higher robustness than the classification model when predicting data in untrained states. Moreover, on both models, an interesting phenomenon is presented that the data in S1' tends to be predicted as S7 or S7' on other side of the beam, which is unexplainable in current phase.

5. Conclusion

In this paper, the robustness of vibration-based SDD though CNN models are discussed. Based on 2 vibration experiments conducted on a steel T-shape beam, two databases are established. The performances of a classification CNN model and a regression CNN model to identify damage locations of data in untrained categories are discussed.

Classification CNN model can hardly predict the approximate structural damage locations which are in untrained categories. In total there are only 478 correct predications with 22.2% accuracy, which shows that it is unfeasible to use classification CNN model to detect the approximate structural damage locations.

The regression CNN model shows better performance than the classification CNN model, since the regression CNN model can predict the approximate damage locations when the structural damages are in the inner area of the beam. However, the structural damages in the two ends of the beam tend to be predicted with big error. Overall, regression CNN model has higher robustness than the classification CNN model when predict the data in untrained categories.

Meanwhile, the phenomenon of that the predictions of data in S1' tend to be S7 or S7' on other side of the beam, may give hints of the special relation between vibration data and structural changes which cannot be ignored.

In future works, to increase the robustness of the CNN-based SDD method, other architectures of the neural networks will be investigated from the views of both data science and dynamics.

Acknowledgement

This research was partially supported by JSPS Grant-in-Aid for Scientific Research(C) Grant Number 15K06176 and NEDO Advanced Research Program for Energy and Environmental Technology. The contributions of Prof. Tomoyuki Yamazaki, Mr. Masato Nakano, Ms. Sanae Suzuki, Ms. Yukina Hashizume for the experiments, and Mr. Huangyi Qin and Ms. Ruijing Yang for technical support are greatly acknowledged.

References

Lin, Y. Z., Nie, Z. H., & Ma, H. W. (2017), Structural damage detection with automatic feature extraction through deep learning, *Computer Aided Civil and Infrastructure Engineering*, 32(12), 1025-1046.

Abdeljaber, O., Avci, O., Kiranyaz, S., Gabbouj, M., & 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.

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