Vibration-based structural damage detection using 1-D convolutional neural network and transfer learning

Article *in* Structural Health Monitoring · December 2022

DOI: 10.1177/14759217221137931

5 authors, including:

Shuai Teng
Guangzhou University

Gongfa Chen
Guangdong University of Technology

David Bassir
University of Technology of Belfort-Montbéliard

Original Article

Structural Health Monitoring
1–22

DOI: 10.1177/14759217221137931
journals.sagepub.com/home/shm
SAGE

Vibration-based structural damage detection using 1-D convolutional neural network and transfer learning

Shuai Teng[1], Gongfa Chen[1], Zhaocheng Yan[1], Li Cheng[2] and David Bassir[3]

Abstract

This paper presents a novel vibration-based structural damage detection approach by using a one-dimensional convolutional neural network (1-D CNN) and transfer learning (TL). The CNN can effectively extract structural damage information from the vibration signals. However, the CNN training needs enough samples, while some damage samples (scenarios) obtained from real structures are limited, which will compromise the CNN ability to detect structural damage. As a solution, the numerical models have potential to provide sufficient CNN training samples; meanwhile, the state-of-the-art TL technique can significantly shorten the network training time and improve the accuracy. Therefore, this paper proposes a new method to detect the damage of a bridge model. The 1-D CNN is firstly trained with the samples of the single damage scenarios of the numerical bridge model. And then it is transferred to the complex scenarios of multi-damage (double or triple simultaneously), random size structures, and experimental model. The results demonstrate that: with the TL, the accuracy of damage detection is increased by about 47% at most, and the convergence speed is increased by at least 50%; in particular, the TL can inhibit over-fitting, and for the real bridge case, the accuracy also increased by 44.4%. It is demonstrated that: the TL can effectively improve the damage detection accuracy and convergence effect, and the application of this method to the random size structures also proves its generalization.

Keywords

Structural damage detection, convolutional neural network, transfer learning, vibration signals, bridge model

Introduction

Structural damage detection (SDD) is widely concerned in structural health monitoring (SHM).[1] The structural fatigue damage (caused by the harsh environment and long-term loading) is the most common damage type during the service of infrastructures.[2] Timely SDD can effectively prevent the sudden collapse of the infrastructures and protect the people's life and property. The manual inspection, image recognition, and vibration characteristic analyses are popular SDD methods. The manual inspection is labor-intensive, subjective, and the inspectors need on-site observations; for some high and/or hard-to-reach infrastructures, the inspectors face some potential risks. The image recognition technology can replace some manual operations; especially with the most advanced image acquisition devices and image recognition algorithms, the detection efficiency and accuracy have been significantly improved. However, the image recognition can only detect the structural surface defects, which is insufficient for evaluation of the overall performance of structures. Especially for the damage inside structures, the image recognition-based SDD method is inapplicable. However, the changes of structural stiffness, mass, and damping will induce the changes of structural vibration information,[3] for example, the natural frequencies,[4] mode shapes,[5] and their derivatives (modal flexibility,[6] curvature,[7] modal strain energy,[8,9] etc.); the change of these information can reflect the damage state of the

[1]School of Civil and Transportation Engineering, Guangdong University of Technology, Guangzhou, China
[2]Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong, China
[3]Centre Borelli, CMLA, ENS Cachan, CNRS, Université Paris-Saclay, Gif-sur-Yvette, France

Corresponding author:
Gongfa Chen, School of Civil and Transportation Engineering, Guangdong University of Technology, 100 Waihuan Xi Road, Panyu, Guangzhou, Guangdong 510006, China.
Email: gongfa.chen@gdut.edu.cn

structure concerned. Meanwhile, with the progress of signal processing technology (e.g., the Kalman Filter,[10] Principal Component Analysis (PCA),[11] etc.), the vibration signal-based (acceleration[12] and displacement[13]) SDD methods have attracted extensive attention and achieved encouraging results; however, in most cases, the manual diagnosis is still needed. As an alternative, the artificial neural network (ANN) is a more potential, intelligent, and automatic signal processing and detection tool for vibration-based SDD method.

The ANN technology provides a new way to detect structural damage; it enables the network to automatically learn from the experience, and the learning knowledge is applied to the detection scenarios of the corresponding structures of interest. The ANN algorithm has been widely used in vibration-based SDD. The relevant research has confirmed that using the back-propagation (BP) algorithm to train neural networks[14] has achieved encouraging results, and it has been applied to damage detection methods based on parametric (modal shapes and their derivatives) and non-parametric (vibration) signals. For example, damage detection of a truss[15] and steel frames,[16,17] and also has been validated in the real models (simple supported beam[18] and bridge model[19]). However, the above methods all need to extract a set of fixed features by some popular signal processing methods (e.g., the modal analysis, PCA, and wavelet decomposition[20–23]). Furthermore, the BP neural network is prone to overfitting, weak generalization ability, and high computational cost, which will compromise its effectiveness in SDD. As an algorithm of deep learning (DL), the convolutional neural network (CNN) has better performance than BP-trained neural networks by introducing the weight sharing and partial connection mechanism, and multiple functional layers, through which the features can be extracted directly from the original data. There are also some limitations in some scenarios (e.g., too few samples are likely to lead to poor generalization ability). As an auxiliary strategy, the transfer learning (TL), which takes the model developed for Task A as the initial point and reuses it in the process of developing the model for Task B, can effectively solve the generalization problem of the CNN.

As a DL algorithm, the CNN provides an unprecedented method for SDD, because of its excellent feature extraction ability and powerful computing performance. The superiority of the CNN has been proved in SDD from the modal information,[24] acceleration signals,[25] and surface defect images.[26] For vibration-based methods, firstly, a damage detection case of a benchmark structure (numerical and experimental[27]) based on the 2-D CNN proves its effectiveness. As an alternative, the 1-D CNN achieves breakthrough progress in electrocardiogram (ECG) detection,[28] engine detection,[29] and voltage/current detection of electronic equipment.[30] These cases illustrate the excellent performance of the 1-D CNN in signal processing. In civil engineering, the 1-D CNN is used in damage detection of a laboratory frame[31,32] and the mass change detection of a real bridge.[33] Recently, by fusing the detection results of multiple 1-D CNNs, the accuracy of damage detection is significantly improved, which has been confirmed by the numerical and experimental bridge models.[34] The relevant research shows that the 1-D CNN is easier to train, and the detection speed is 45 times that of a 2-D CNN.[35] Although the SDD method based on the CNN has achieved encouraging results, for structures in service, the structural damage samples that can be obtained are limited (the generalization ability of CNN is poor, and it is difficult to detect the corresponding structure); meanwhile, training a CNN from scratch is an arduous task, especially for the complex detection tasks, these challenges will compromise the further application of the CNN in practical engineering. The TL-based CNN method provides a new solution to further improve the training effect and detection accuracy (especially improve the generalization ability of CNN).

Related works

The TL technique can transfer the knowledge from the source domain to the target domain, and it will provide a potential and excellent SDD/SHM method. It has been widely used in the field of image recognition and achieved excellent results in structural surface defects detection.[36,37] Li et al.[38] collected crack image datasets of bridges, walls, and houses as the training samples of a pre-trained network, and then transferred it to the concrete dam crack detection task, which greatly improved the detection accuracy (limited images of concrete dam cracks). Zheng et al.[39] used a pre-trained network obtained from the COCO dataset and applied it to the detection task of rail surface defects through the TL (there are only 102 images in the rail surface defect dataset, the classical non-transfer learning (NTL) method is not available). The results show that the ideal detection effect can be obtained through the TL strategy. These cases prove that the TL can achieve better detection performance with limited training samples. In the field of SDD, it is still faced with the problem of limited training samples, especially for structures in service, it is difficult to obtain a large number of samples in the damage scenarios, so the TL will be a potential solution.

The TL also shows excellent performance in signal processing. A study realized the fault diagnosis method of rolling bearing through the TL,[40] and heartbeat murmurs detection in phonocardiogram recordings via the TL.[41] These excellent performances advocate its potential application in SDD field of bridge structures; for real structures, the damage samples of structures are limited, and it is difficult to obtain a CNN model with excellent performance. Therefore, obtaining a large number of CNN training samples through numerical simulations and transferring it to the damage detection of actual structures is a potential solution to the shortage of samples. Thus, this paper will explore the TL in vibration-based SDD, that is, using a population of bridge numerical models to train the CNN (pre-trained CNN), and then using a small amount of actual structure samples to fine-tune the pre-trained CNN model to adapt to the damage detection task of actual structures. In summary, the novel contributions are as follow: a pre-trained network is established through simple detection scenarios and then it is transferred to: (1) In complex detection scenarios (double and triple damages simultaneously), it will be proved that the TL can realize damage detection in similar scenarios; (2) In random size structures, the generality of the proposed method will be confirmed; (3) In an experimental model, the practical value of the proposed method will be confirmed; and (4) In a real bridge case, the reliability of the proposed method will be validated.

Methods

In this paper, the population of bridges (some similar numerical models) of single damage scenarios were first established, and then a CNN model with strong compatibility was obtained as a pre-trained network model. Subsequently, the pre-trained CNN was transferred to various damage scenarios of other numerical models, an experimental model, and a real bridge case (Figure 1).

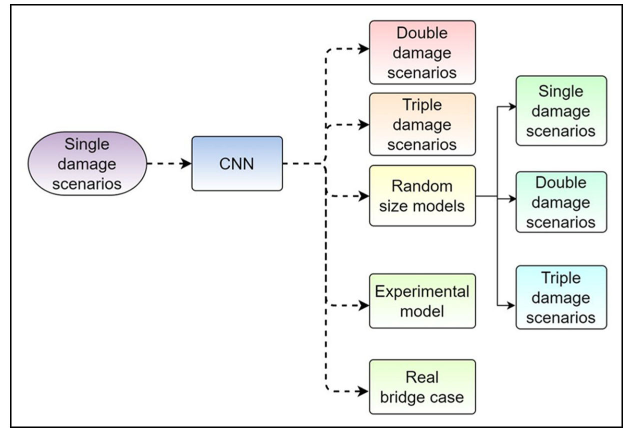

Figure 1. Implementation strategy of proposed method.

CNN and transfer learning

In this paper, a 1-D CNN was established by using the "Deep Learning Toolbox" of MATLAB (MathWorks Inc., Natick, MA, USA), including two convolution layers, one pooling layer, two activation layers (with the Leaky ReLU activation function), one fully connected layer, and one softmax layer (see Appendix A for the principle of the 1-D CNN). Detailed network parameters are shown in Table 1. According to the previous research results, the damage detection strategy based on the 1-D CNN and decision-level fusion has achieved excellent results,[34] that is, the data obtained at each acquisition point were used to train a 1-D CNN respectively, and the detection results of multiple 1-D CNNs (acquisition points) were fused to obtain the final detection result. This paper will inherit this method.

TL is a DL method, which transfers knowledge from one domain (previous domain) to another domain (novel domain).[42] Through the training of previous domain, the network model has strong ability of feature extraction for the similar data; subsequently, just with fine-tuning (re-training) it with the new data in different scenarios, the network model will quickly adapt to the new detection task (Figure 2). With this

Table 1. Parameters of the 1-D convolutional neural network (CNN).

Layer	Type	Kernel num.	Kernel size	Stride	Activation function
1	Input	None	None	None	None
2	Convolution (Conv_1)	128	3×1	1	Leaky ReLU
3	Max pooling	None	2×1	1	None
4	Convolution (Conv_2)	256	2×1	1	Leaky ReLU
5	FCN	None	None	None	None
6	Softmax	None	None	None	None
7	Classification	None	None	None	None

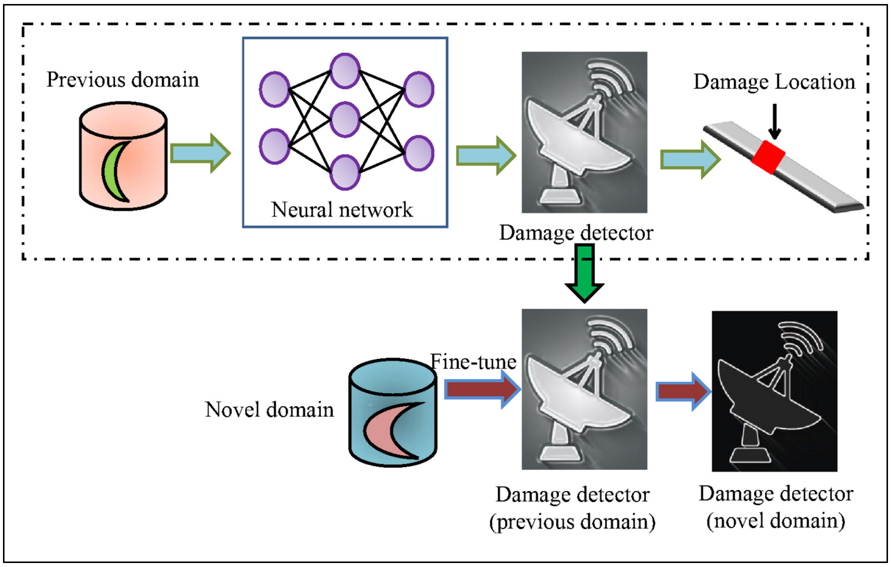

Figure 2. Damage detection method using the transfer learning (TL).

method, a lot of training time of the CNN will be saved for the novel domain, and better training effect can be achieved, especially when there are limited training samples in the novel domain.

The TL strategy implemented in this paper included the following processes:

Process 1: Training a 1-D CNN to detect single damage of the numerical simulations (population of bridge) to build a pre-trained CNN model;
Process 2: The pre-trained 1-D CNN was transferred to multi-damage detection tasks (double and triple damages simultaneously) and 10 random size structures of the numerical simulations. The specific operation is shown in the Figure 3; in Process 1, the pre-trained Network A was obtained through the classical training process using these pre-trained CNN samples. On this basis, the TL strategy was implemented, that is, the fully connected layer (fc) of the pre-trained network was replaced by a new one (new_fc). Then, the samples of the new damage scenarios were used to fine-tune the modified network to adapt to the new damage detection task;
Process 3: The pre-trained 1-D CNN was transferred to the damage detection task of the experimental model.
Process 4: The pre-trained 1-D CNN was transferred to the damage detection task of the real bridge case.

Experimental and numerical models

A real bridge model (Figure 4(a)) was used in this paper for the vibration experiment, and its length, width, and height were 2.4, 0.3, and 0.3 m, respectively. This bridge model consisted of 58 flat steel bars, which had a rectangular cross section (0.02 m × 0.002 m) and were connected by the bolts. The vibration was excited and measured by using the following equipment: an instrumented hammer, a JM3840 dynamic data acquisition instrument, seven accelerometers, and a laptop. Various damage scenarios were realized by replacing the intact flat bars with damaged ones (the flat steel bar was cut, Figure 4(b)).

A numerical model of the bridge structure (Figure 5) was established using ABAQUS (SIMULIA Inc., Providence, RI, USA). The elastic modulus, Poisson's ratio, density, and modal damping ratio were 210 GPa, 0.3, 7,800 kg/m^3, and 0.03 respectively for the bridge model. All flat steel bars were meshed with beam element (B31 type). The 58 flat steel bars were named FSB-1, FSB-2, …, FSB-58 respectively. The structural damage was induced by reducing the elastic modulus at the damage location.

Data and experimental setup

Firstly, for the damage detection of the numerical model (Figure 5), four datasets were established:

Dataset (A), based on Section "Experimental and numerical models," the following two random factors were applied to the model. With these two factors, 100 numerical bridge models with randomly created geometric dimensions and excitation forces were obtained; thus, a population of bridge models was created. This

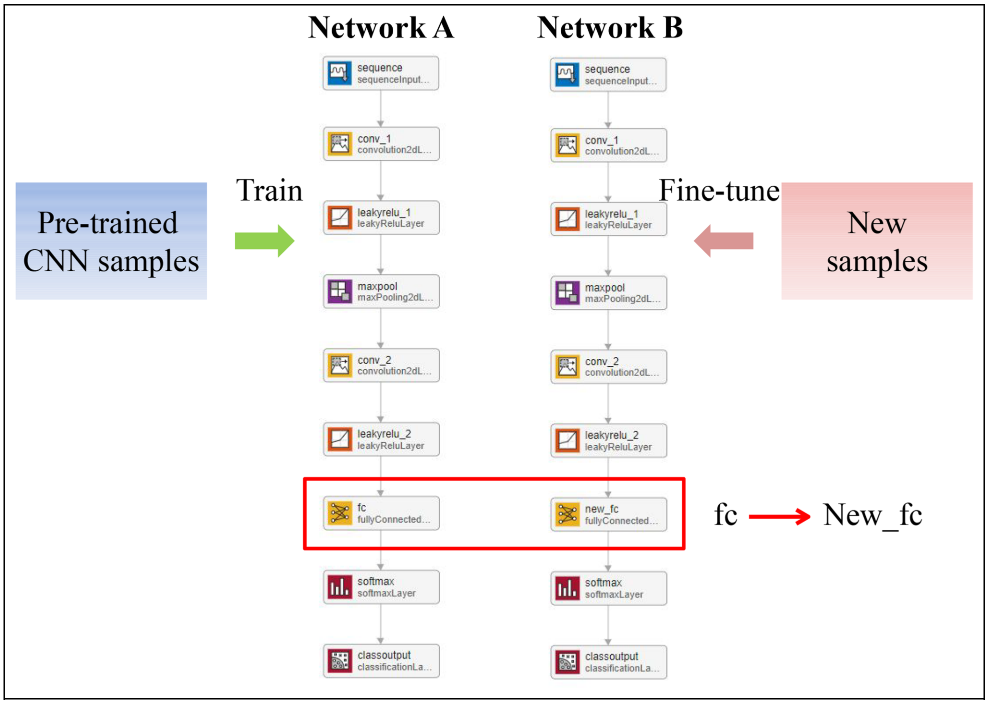

Figure 3. Specific operation of transfer learning (TL).

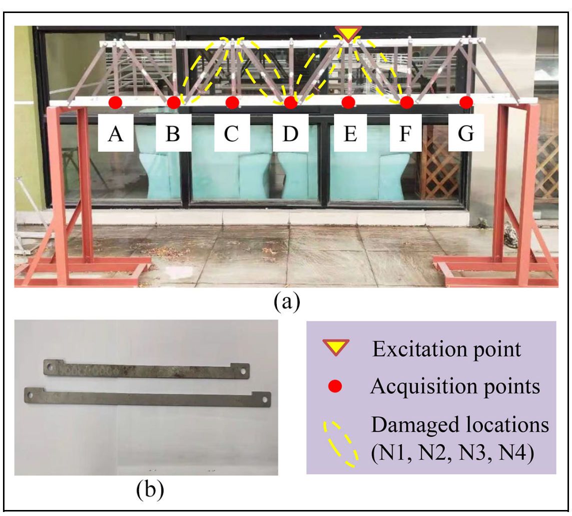

Figure 4. The bridge model with 58 flat steel bars: (a) bridge model and (b) damaged flat steel bars.

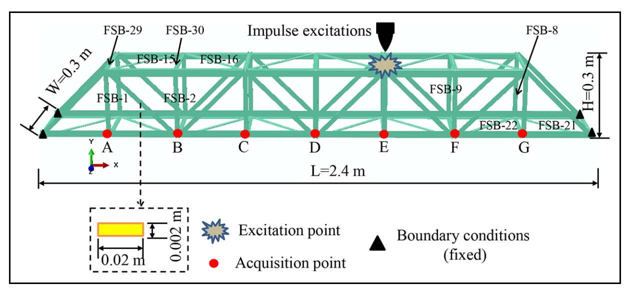

Figure 5. The numerical bridge model with 58 flat steel bars.

method can effectively improve the compatibility of a CNN and reduce the impact of environment on structural vibration.[43]

Factor 1: Geometric dimensions

Based on the bridge model, its length, width, and height were randomly modified between −50% and +50%.

Factor 2: Excitation forces (amplitude curves)

The random amplitude curves were given during the loading process of excitation forces (Figure 6), so random vibration was induced on the structure.

The scenarios with a single damage of the flat steel bars was studied, which means that a damage in one flat steel bar was considered for each damage scenario. For each structure, the structure consisted of 58 flat steel bars; thus, there were 59 structural scenarios, including 58 damage scenarios plus the intact structure.

Dataset (B), the scenarios with damages simultaneously in two flat steel bars, 2 of 13 flat steel bars (FSB-1, FSB-5, FSB-10, …, FSB-55, FSB-58) were randomly selected; there were 78 (C_{13}^{2}) damage scenarios.

Dataset (C), the scenarios with damages simultaneously in three flat steel bars, 3 of 13 flat steel bars (FSB-1,

Table 2. Random scaling factor of the models.

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
Length	1.19	1.24	0.78	1.25	1.08	0.76	0.87	1.03	1.27	1.28
Width	0.87	1.11	1.09	0.80	0.77	1.00	1.28	0.90	1.05	0.83
Height	0.80	1.18	0.89	1.02	0.80	1.06	0.86	1.09	1.11	1.15

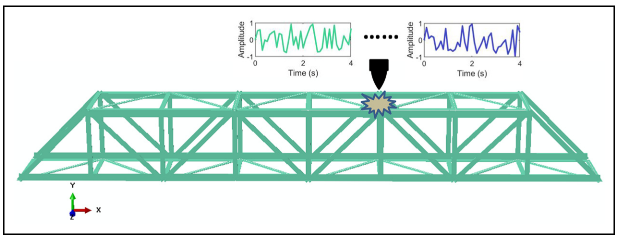

Figure 6. Random excitation force loaded into the numerical bridge model.

FSB-5, FSB-10, …, FSB-55, FSB-58) were randomly selected; there were 286 (C_{13}^3) damage scenarios.
Dataset (D), in order to validate the generality of the proposed method, this paper extends it to 10 numerical models (M1–M10) with random geometric dimensions (the random scaling factor (length, width, and height) were (±30%), shown in Table 2). For each model, its datasets with single damage, double damages, and triple damages were obtained. The sub-datasets of the 10 models were named Sub-Dataset M1, Sub-Dataset M2, Sub-Dataset M3, …, Sub-Dataset M10, respectively.

The data of the above multiple structural scenarios were obtained through the parametric analysis method (ABAQUS platform and PYTHON scripts). It was assumed that the damage level of the flat steel bars was proportional to the reduction of its elastic modulus. In this paper, the elastic modulus of the flat steel bars at the damage location was reduced by 60%. Two consecutive impulse excitations (800 and 1000 N) were applied to the structure at the excitation point, and then the acceleration signals of 400 sampling points (the sampling time of 4 s with an increment of 0.01 s) for each impulse excitation was collected. The CNN samples were created as follow:

As shown in Figure 7, the vibration signals (1 × 400 array) of the acquisition point A (Figure 1) generated by an excitation, including 400 sampling points, was divided into four equal parts through the fixed size windows, that is, 4 samples (4 1 × 100 array); therefore, according to this method, the sample number of the four datasets was defined as (Table 3):

Dataset (A): 3304 samples (4 × 7 (acquisition points) × 59 (structural scenarios) × 2 (two excitations)).
Dataset (B): 4368 samples (4 × 7 (acquisition points) × 78 (structural scenarios) × 2 (two excitations)).
Dataset (C): 14,336 samples (4 × 7 (acquisition points) × 256 (structural scenarios) × 2 (two excitations)).
Dataset (D): each sub-dataset (Sub-Dataset (M#)) included the scenarios with a single damage (3304 samples), double damages (4368 samples), and triple damages (14,336 samples).

Secondly, the intact flat steel bars were replaced by the damaged ones (the flat steel bar was cut, Figure 4(b)) in the experimental model, and the following 15 structural states were designed (Table 4) to simulate various damage scenarios. Each structural scenario was stimulated three times by a hammer, the data obtained from the first and second stimulations were used as the training samples, and the data from the third one was used as the testing samples. According to the above sample acquisition method (Figure 7), the numbers of training samples and testing samples were 840 (4 × 7 (acquisition points) × 15 (15 structural scenarios) × 2 (2 instantaneous excitations)) and 420 (4 × 7 (acquisition points) × 15 (15 structural scenarios)×1 (1 instantaneous excitation)) respectively.

Real bridge case

This case used the dataset obtained from a 15 months monitoring activity on a steel string railway bridge in Leuven, Belgium.[44] During this monitoring activity, the connecting components between the bridge deck and arch were strengthened after damage was observed. This railway bridge was also known as KW51 railway bridge (as shown in Figure 8, the geographic coordinates were 50.9004 N and 4.7066 E). The bridge was bowstring type, length: 115 m and width: 12.4 m. The bridge was located on the L36N railway line between Leuven and Brussels, across the Leuven-Mechelen canal.

The railway bridge was monitored since October 2, 2018. From May 15 to September 27, 2019, the bridge was reconstructed to solve the construction errors found during the inspection. The connecting components of the arch and bridge deck were strengthened during the reconstruction. Figure 9 shows images of these connections before and after the retrofitting, as

Table 3. Dataset summary of numerical models.

	Dataset			
	Dataset (A)	Dataset (B)	Dataset (C)	Dataset (D)
Training samples	165,200	2184	7168	110,040
Testing and validation samples	165,200	2184	7168	110,040
Total	330,400	4368	14,336	220,080

Fifty percent of these samples were used for training and the rest 50% for testing and validation.

Table 4. The damage scenarios of experimental model.

Structural state	State 1	State 2	State 3	State 4	State 5
Damaged location	Intact structure	N1	N2	N3	N4
Structural state	State 6	State 7	State 8	State 9	State 10
Damaged location	N1&N2	N1&N3	N1&N4	N2&N3	N2&N4
Structural state	State 11	State 12	State 13	State 14	State 15
Damaged location	N3&N4	N1&N2&N3	N1&N2&N4	N1&N3&N4	N2&N3&N4

N# represents the damaged location (Figure 1).

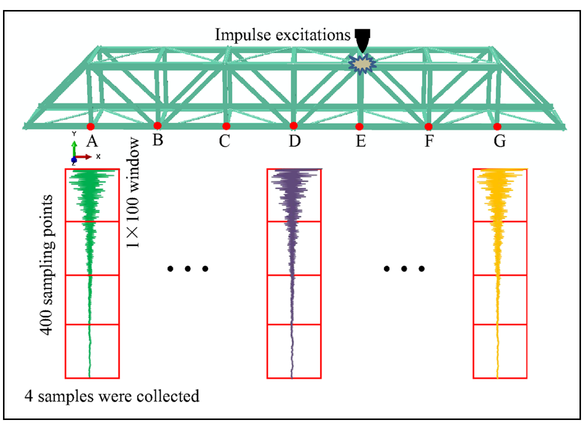

Figure 7. The sample acquisition of the convolutional neural network (CNN).

Figure 8. KW51 railway bridge.

well as images of scaffolding installed on the bridge during the retrofitting engineering. For each connecting component, a steel box was welded around the original bolted connection intersecting the arch and bridge deck (as shown in Figure 9(a) and (c), before and after retrofitting, respectively). The data used in this paper were collected before the retrofitting (October 27, 2018), during retrofitting period (July 18, 2019), and after the retrofitting (November 14, 2019). And six accelerometers were mounted on the bridge deck.

Results and discussions

For the numerical model, Dataset (A) was first used for the 1-D CNN training, and then the 1-D CNN was transferred to detection of the complex damage scenarios of Dataset (B), Dataset (C), and Dataset (D). Finally, the 1-D CNN model obtained from the numerical model was transferred to the experimental model and real bridge case. The results including two parts:

(1) The damage detection results based on the 1-D CNN and TL in the numerical model;
(2) The damage detection results based on the 1-D CNN and TL in the experimental model;
(3) The damage detection results based on the 1-D CNN and TL in the real bridge case.

Detection results of the numerical model

Firstly, ABAQUS was employed for numerical simulations of the bridge model. The vibration signals of the

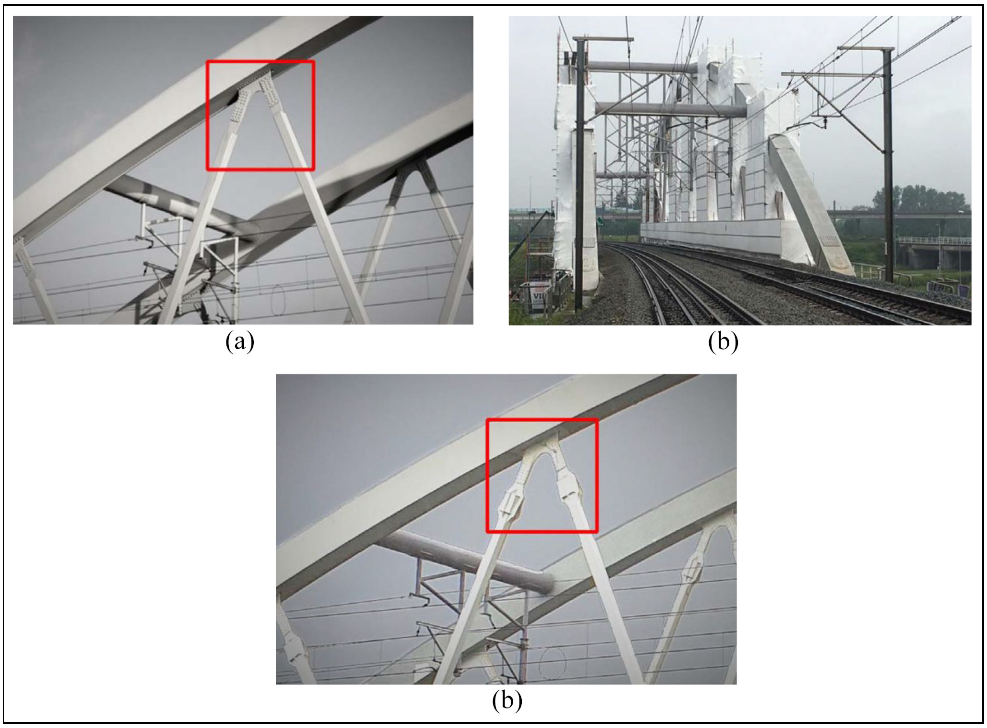

Figure 9. (a) Original connections; (b) retrofitting period; (c) after the retrofitting.

intact structure are shown in Figure 10 (where, S1, S2, S3, …, and S7 were the acceleration signals of the acquisition points A, B, C, …, and G respectively), and all vibration signals (for all structural scenarios) can be obtained through corresponding authors. Then, the training and testing samples of the 1-D CNNs were obtained by using the method described in Section "Data and experimental setup." The damage detection can be realized in the following four steps:

Step 1, detection results of the single damage scenarios: the training samples were used to train the 1-D CNNs: After 300 iterations, the accuracy (ratio of correct detection number to total number) and loss value (the cross entropy loss[45] between the predicted results and the real results) of the validation samples tend to be stable, which are showed in Figure 11. The testing results are shown in Figure 12, the average accuracy of each sensor (acquisition point) was about 90%, and it reached 100% after the decision-level fusion.

Step 2, detection results of the double damages simultaneously: The loss value of the training process (Non-Transfer Learning, that is, NTL, it means that instead of using the TL strategy, an initialized network was trained from scratch) is shown in Figure 13; there was a serious fluctuation, despite 2000 iterations, and the accuracy and loss value were still unstable . The testing results are shown in Figure 14(a), which illustrated the accuracy of each sensor and the decision-level fusion; the accuracy of the single sensors was about 80%–92%, and that of the decision-level fusion was about 96%.

Then, the TL technique was applied to the detection task of double damages, in which the training samples were used to fine-tune the pre-trained 1-D CNNs obtained in Step 1. After 300–600 iterations, the loss values of the networks tended to be stable (Figure 13). The testing results are shown in Figure 14(b), the accuracy of the single sensors was about 95%, and that of the decision-level fusion was 100%. Therefore, the detection accuracy of the double damages was improved by more than 4% by the TL technique.

Table 5 showed the number of iterations when the network converged (*Note*: the total number of iterations based on the TL method = pre-trained network + TL network, the same hereinafter). The results showed that the TL technique can significantly improve the convergence effect of the network. Especially, the convergence speed was increased by at least 52.9%.

Step 3, detection results of the triple damages simultaneously: the training samples were used to train the 1-D CNNs. The training process of the networks is shown in Figure 15, there were more serious fluctuations, although 7600 iterations were carried out, the

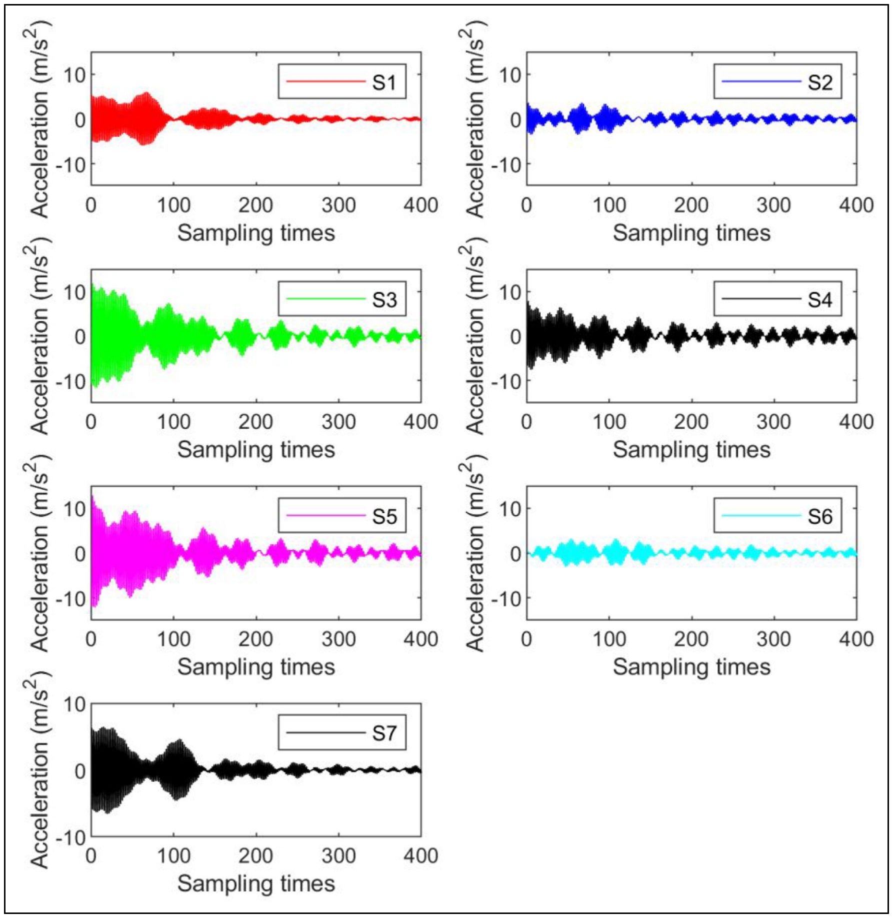

Figure 10. Acceleration signals of the numerical model.
S#: Sensor #.

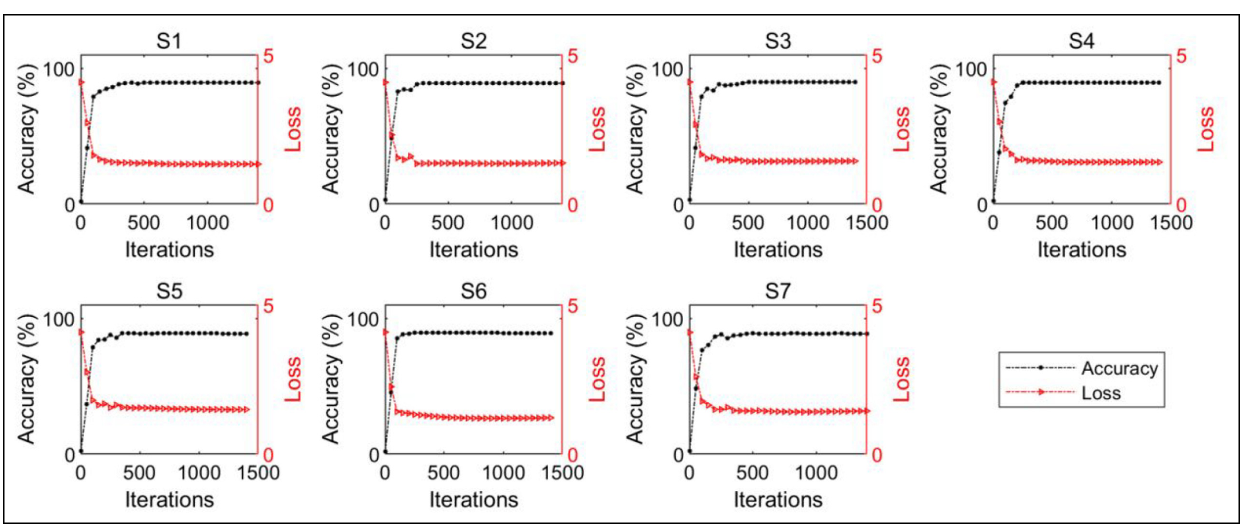

Figure 11. Validation process in network training.

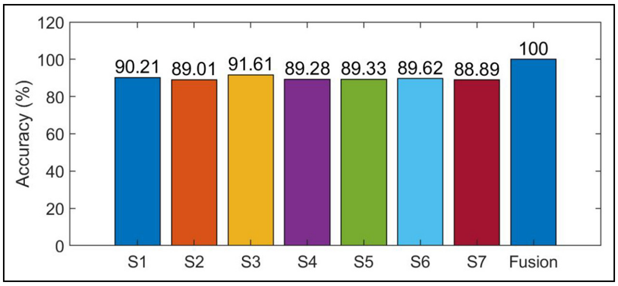

Figure 12. Testing results of the single damage scenario.

loss value were still unstable, there were still some small fluctuations. The testing results are shown in Figure 16(a), the accuracy of the single sensor was about 75%–86%, and that of the decision-level fusion was about 93%.

Then, the TL technique was applied to the detection task of triple damages, in which the training samples were also used to fine-tune the pre-trained 1-D CNNs. After 2500–4000 iterations, the loss values of the networks tended to be relatively stable (Figure 15). The testing results are shown in Figure 16(b), the accuracy of the single sensors was improved to about 98%, and that of the decision-level fusion was 100%. Therefore, the detection accuracy of the triple damages was improved by about 7% by the TL technique. Table 6 also showed that the convergence speed was increased by at least 51.8% (with TL).

Step 4, the pre-trained 1-D CNN was transferred to 10 random size models: The detection accuracy of the scenarios with single damage, double damages, and triple damages are shown in Figures A3 to A5 (in Appendix B). The results showed that the TL could significantly improve the detection accuracy of the 10 random size models with single damage, double damages, and triple damages (Tables 7–9), and the accuracy of these models were increased by 8.4%, 13.0%, and 7.9% on average. In particular, the best improvement effects (M1 in Table 7, M7 in Table 8, M1 in Table 9) of the scenarios with single damage, double damages, and triple damages were 29.3%, 43.6%, and 23.4% respectively, their validation processes are shown in Figure 17(a) to (c), respectively. Figure 17(a) showed that, although both the NTL and TL methods trended to convergence, the TL could significantly reduce the loss value of the networks; Figure 17(b) showed that the TL could reduce the occurrence of over-fitting phenomenon. The loss value of the NTL method gradually increases with the increase of iteration number, while the TL was still in a relatively stable state; Figure 17(c) showed that the TL could significantly reduce the loss value and quickly converge. The loss value of the NTL method fluctuated during the training process and did not converge until the end of training. These detection results confirm that the TL was valuable for SDD in terms of the detection accuracy, convergence effect, and suppression of over-fitting, and so on.

Detection results of the experimental model

The vibration signals of the experimental bridge model were obtained by using the accelerometers. Figure 18

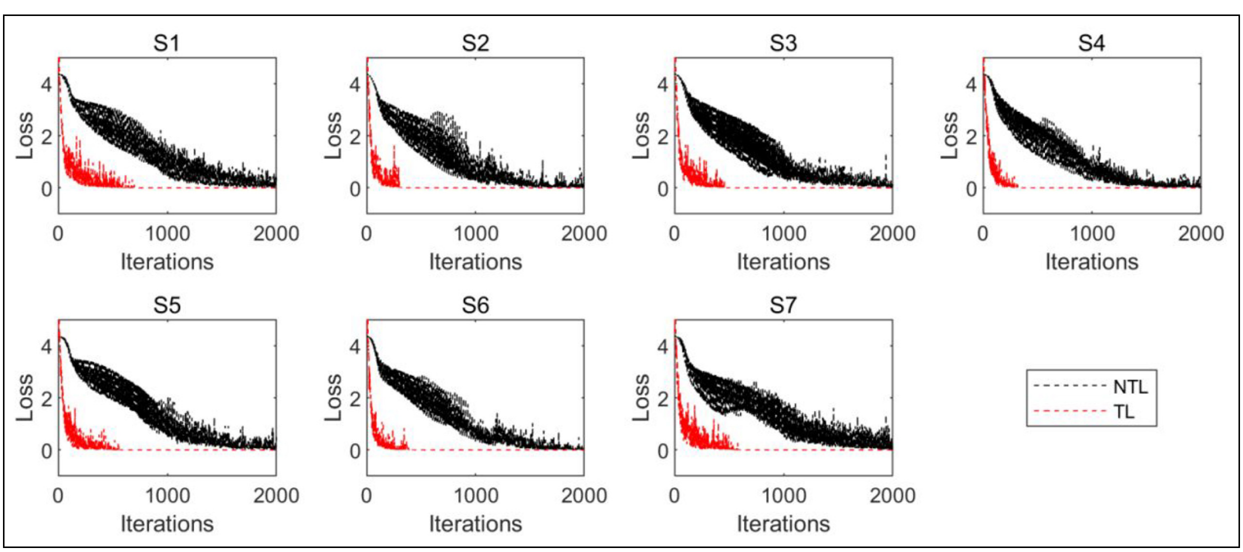

Figure 13. Training loss of the NTL and TL.
NTL: non-transfer learning; TL: transfer learning

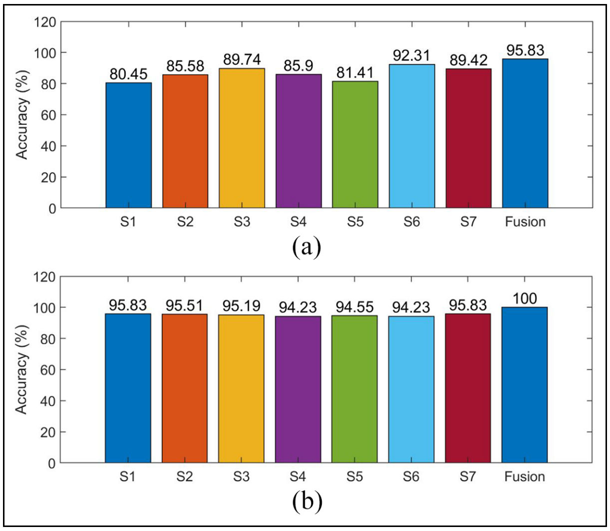

Figure 14. The testing results of the NTL and TL: (a) NTL and (b) TL.
NTL: non-transfer learning; TL: transfer learning

showed some acceleration signals of State 1, in which S1, S2, S3, …, S7 were the acceleration signals of acquisition points A, B, C, …, G, respectively.

The training samples of the experimental bridge model were used to train a 1-D CNN. The training process of the network is shown in Figure 19. There were more serious fluctuations in the training process, even after 4000 iterations, the loss value were still unstable. The testing results are shown in Figure 20(a), the accuracy of the decision-level fusion was 90%.

Then, the TL technique was used to detect damage of the experimental bridge model, that is, the 1-D CNN model was fine-tuned by using the experimental data. Figure 19 was the training process of the 1-D CNN. The loss value decreased with the increase of iterations, and finally tended to be stable, and the loss value was close to 0. The accuracy of the decision-level fusion reached 95% (Figure 20(b)). Therefore, the accuracy was improved by 5%.

The detailed testing results are shown in Table 10, which listed the comparisons of the detection accuracy and convergence speed between the NTL and TL. The results showed that the accuracy of the numerical models was improved by about 5%. Meanwhile, the TL technique could significantly improve the convergence speed of the 1-D CNN. The NTL model did not converge, while the TL model could quickly enter the stable stage, and the convergence speed could be improved by at least 75% (experimental model).

Detection results of the real bridge case

The vibration signals of the real bridge were obtained by using the accelerometers. Figure 21 showed some acceleration signals of original connections, in which Sensor 1, Sensor 2, Sensor 3, …, Sensor 6 were the acceleration signals of the six accelerometers mounted on the bridge deck.

The training samples of the real bridge were used to train a 1-D CNN. Its training process is shown in Figure 22. There were more serious fluctuations in the training process, even after 400 iterations, the accuracy was still not ideal, and the loss value decreased slowly. The testing results are shown in Figure 23(a), the accuracy of all sampling points ranged from 30% to 50%, and the accuracy of the decision-level fusion was 55.6%.

Then, the TL technique was used to detect damage of the real bridge, that is, the 1-D CNN model was fine-tuned by using the measured data. Figure 24 was the training process of the 1-D CNN. The loss value decreased quickly with the increase of iterations and finally tended to be stable, and the loss value was close to 0. The accuracy of the decision-level fusion reached 100% (Figure 23(b)). Therefore, the accuracy was improved by 44.4%.

Table 5. The convergence speed of the NTL and TL.

Model	TL/NTL	Improved	Model	TL/NTL	Improved
S1	600 + 500 2000 more	45.0% more	S6	400 + 500 2000 more	55.0% more
S2	300 + 500 2000 more	60.0% more	S7	600 + 500 2000 more	45.0% more
S3	500 + 500 2000 More	50.0% more	Average	943 2000 more	52.9% more
S4	300 + 500 2000 More	60.0% more			
S5	500 + 500 2000 More	50.0% more			

NTL: non-transfer learning; TL: transfer learning.

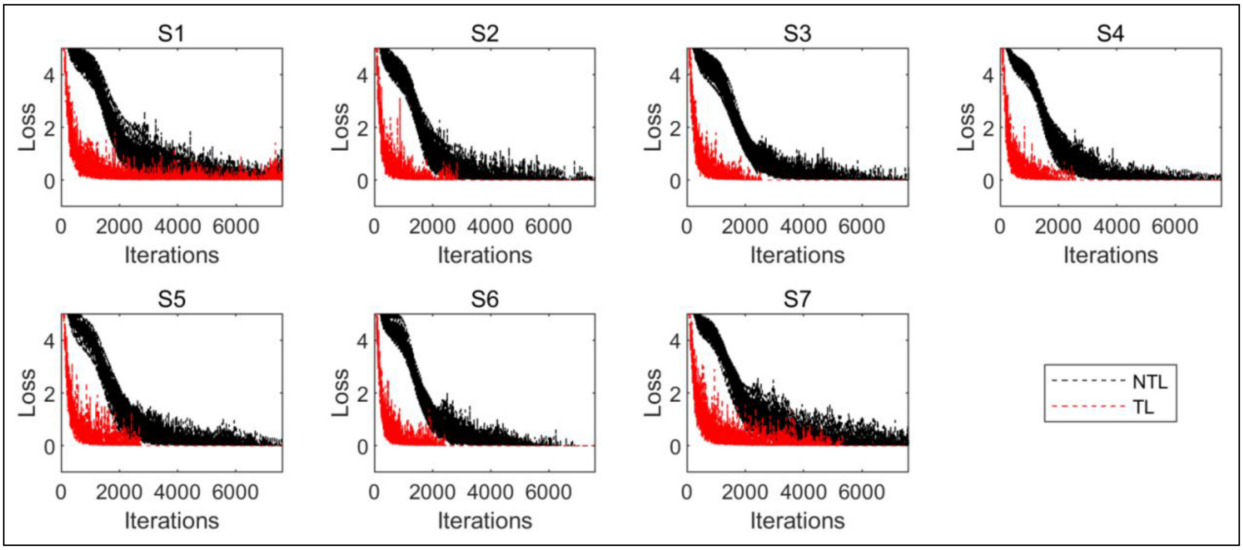

Figure 15. Training loss of the NTL and TL.
NTL: non-transfer learning; TL: transfer learning.

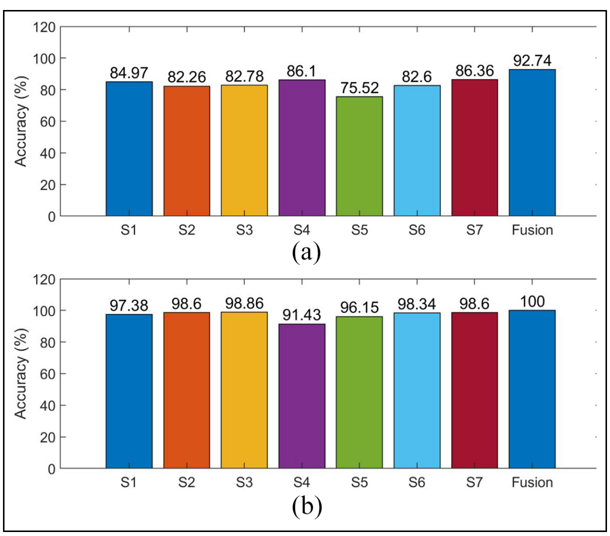

Figure 16. The testing results of the NTL and TL: (a) NTL and (b) TL.
NTL: non-transfer learning; TL: transfer learning.

Subsequently, in order to validate the superiority of the proposed method, the Gate Recurrent Unit (GRU) was also employed to study the real bridge case. The comparisons between the detection results of the proposed method and the GRU are shown in Table 11. In terms of computational complexity, compared with the GRU (384.3k learning parameters), the 1-D CNN (140k learning parameters) has fewer learning parameters and faster computational performance. In addition, the combination of 1-D CNN with the TL strategy can effectively reduce the dimensions of training samples and achieve 100% detection accuracy with only 720 training samples; while to achieve the same accuracy, the GRU combined with the TL strategy requires three times as many training samples as the CNN does.

Conclusions

In this paper, the TL-based SDD method has been utilized to detect different damage scenarios. Specifically, the scenarios with single damage in the numerical models were used to train a 1-D CNN, which was then transferred to the scenarios with multi-damage and random size of the numerical model; it was also transferred to the experimental model. The proposed method can transfer the knowledge of simple damage scenarios to complex scenarios, random size models, experimental structure, and real bridge case, and improve the performance in the case of limited samples. The pre-training network learns some damage information in the simple damage scenarios of population of bridges and obtains the weights of the network model. When transferring to other scenarios, these weights can reach the ideal state only by fine-tuning, avoiding the interference of network uncertainty factors (e.g., inducing network over-fitting when the number of samples is too small) during training from scratch. Compared with the NTL method, the TL method could significantly improve the accuracy of damage detection; which may increase about 47%, in

Table 6. The convergence speed of the NTL and TL.

Model	TL/NTL	Improved	Model	TL/NTL	Improved
S1	None None	None	S6	2400 + 500 7000	31.4%
S2	2800 + 500 7600 more	67.1% more	S7	5400 + 500 7600 more	43.4% more
S3	2600 + 500 7600 more	54.0% more	Average	3567 7500 more	51.8% more
S4	2600 + 500 7600 more	54.0% more			
S5	2600 + 500 7600 more	47.4% more			

NTL: non-transfer learning; TL: transfer learning.

Table 7. The detection accuracy of the NTL and TL (single damage).

Model	TL/NTL (%)	Improved (%)	Model	TL/NTL (%)	Improved (%)
M1	55.5 84.8	29.3	M6	96.2 99.2	3.0
M2	60.6 77.1	16.5	M7	91.1 94.1	3.0
M3	96.6 97.9	1.3	M8	90.7 93.6	2.9
M4	77.5 83.1	5.6	M9	76.7 83.5	6.8
M5	86.4 93.2	6.8	M10	44.9 53.8	8.9

NTL: non-transfer learning; TL: transfer learning.

Table 8. The detection accuracy of the NTL and TL (double damages).

Model	TL/NTL (%)	Improved (%)	Model	TL/NTL (%)	Improved (%)
M1	81.1 90.1	9.0	M6	93.3 100	6.7
M2	75.6 93.6	18	M7	33.0 76.6	43.6
M3	87.8 98.4	10.6	M8	100 100	0
M4	95.2 99.7	4.5	M9	63.1 97.1	34.0
M5	100 100	0.0	M10	94.9 98.7	3.8

NTL: non-transfer learning; TL: transfer learning.

Table 9. The detection accuracy of the NTL and TL (triple damages).

Model	TL/NTL (%)	Improved (%)	Model	TL/NTL (%)	Improved (%)
M1	72.6 96.0	23.4	M6	99.6 100	0.4
M2	79.8 93.3	13.5	M7	99.6 100	0.4
M3	100 100	0	M8	94.1 99.5	5.4
M4	79.7 88.4	8.7	M9	86.9 94.6	7.7
M5	90.3 99.0	8.7	M10	71.9 83.0	11.1

NTL: non-transfer learning; TL: transfer learning.

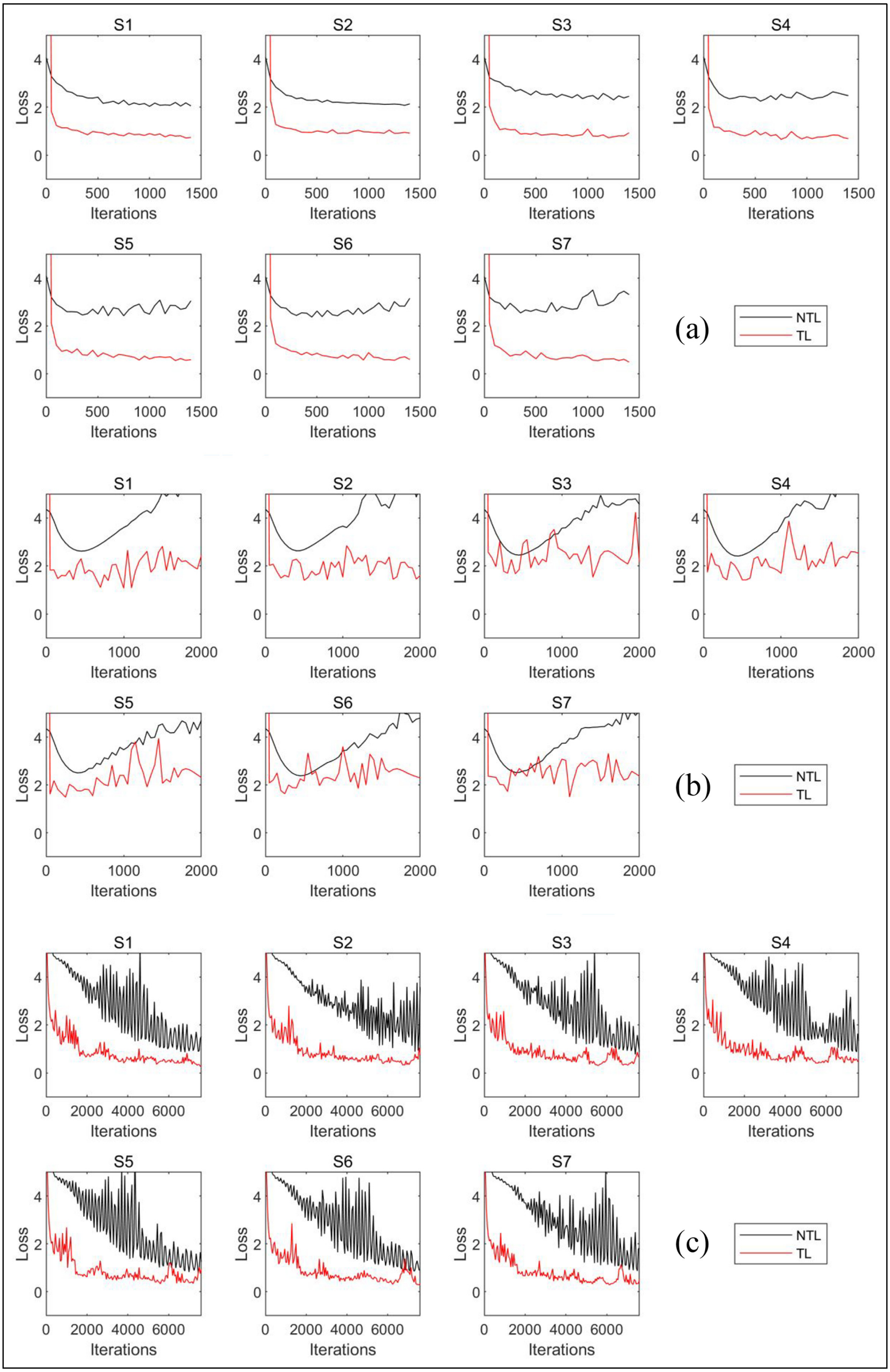

Figure 17. Validation loss of the NTL and TL: (a) M1 of single damage scenario, (b) M7 of double damage scenarios, and (c) M1 of triple damage scenarios.
NTL: non-transfer learning; TL: transfer learning.

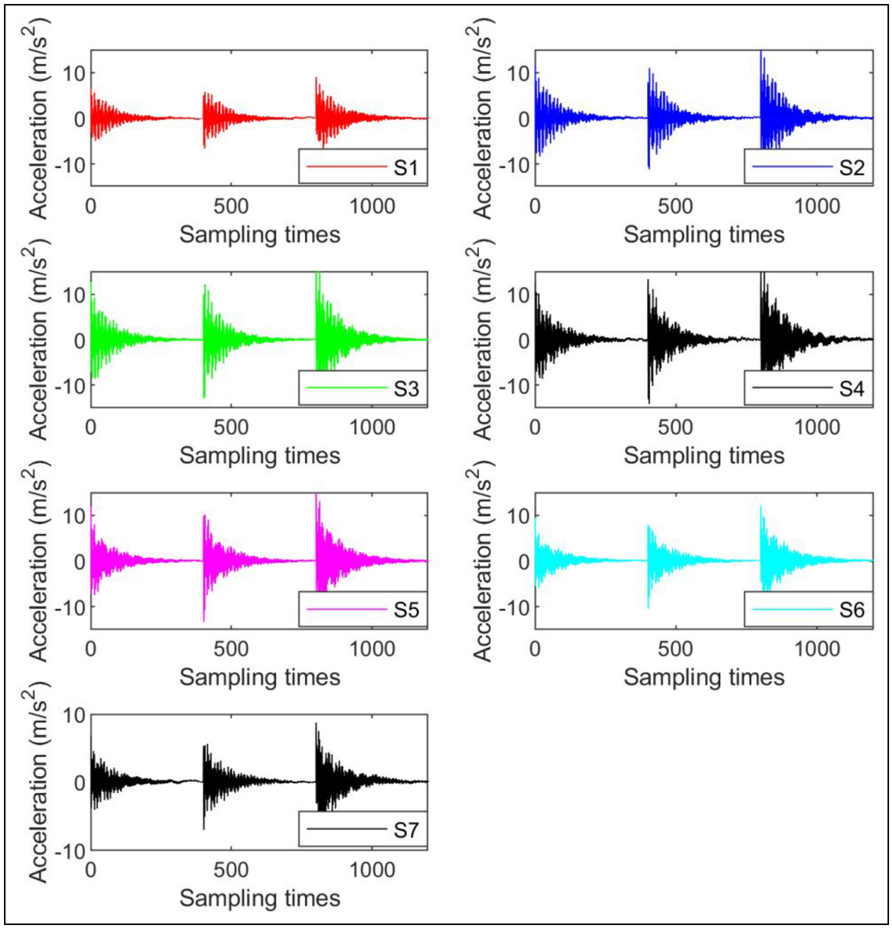

Figure 18. Acceleration signals of the experimental model.

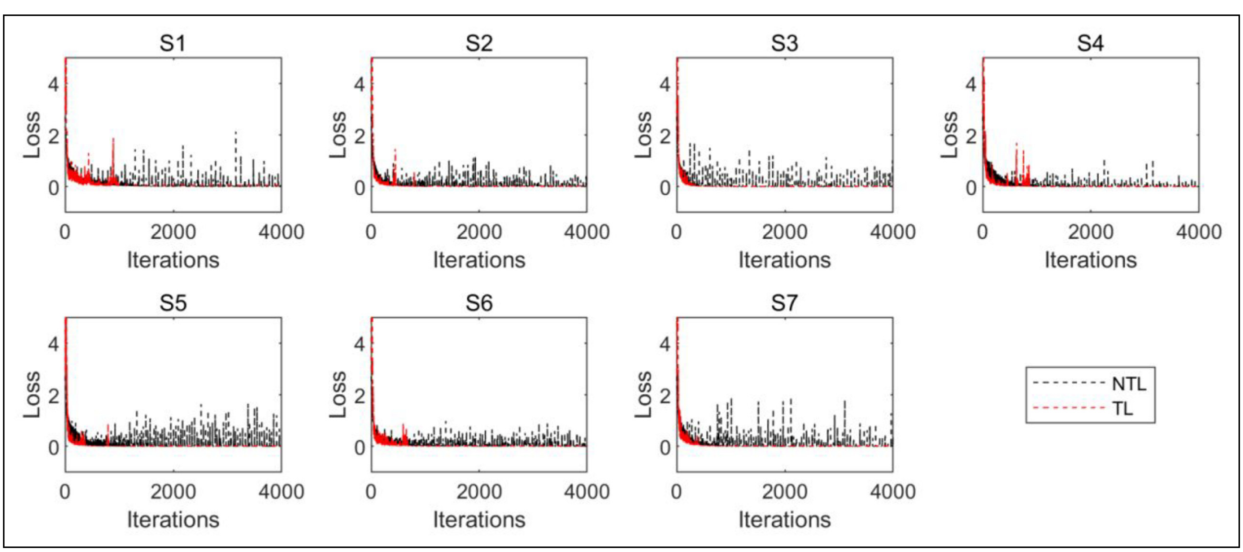

Figure 19. Validation loss of the NTL and TL.
NTL: non-transfer learning; TL: transfer learning.

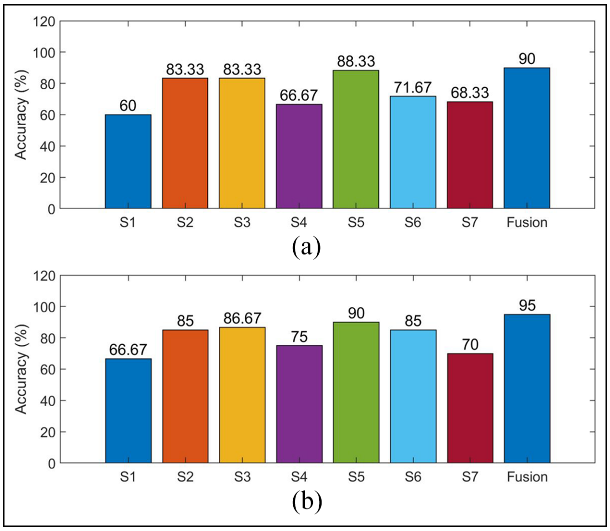

Figure 20. The testing results of the (a) NTL and (b) TL.
NTL: non-transfer learning; TL: transfer learning.

particular, the convergence speed of the 1-D CNN can be greatly improved by the TL method; it may increase by about 50%. Meanwhile, it also has outstanding performance in real bridge case (the accuracy also increased by 44.4%). These encouraging results confirm that the TL is effective and may improve the detection performance of CNNs.

Table 10. Comparisons of the experimental models between the NTL and TL.

	Accuracy (%)	Convergence speed (iterations)
NTL	90	4000 more
TL	95	About 500 + 500
Improved	5	75% more

NTL: non-transfer learning; TL: transfer learning.

Based on the above results, the following conclusions are drawn:

(1) The TL-based SSD method can effectively improve the detection accuracy of the multi-damage scenarios (numerical model, increased by 4%–7%).
(2) The TL-based SSD method can significantly improve the detection accuracy of the random size models (numerical model, increased by about 47%).
(3) The TL-based SSD method can effectively improve the detection accuracy of the experimental models (increased by about 5%).
(4) The TL-based SSD method can effectively improve the detection accuracy of the real bridge case (increased by about 44.4%).

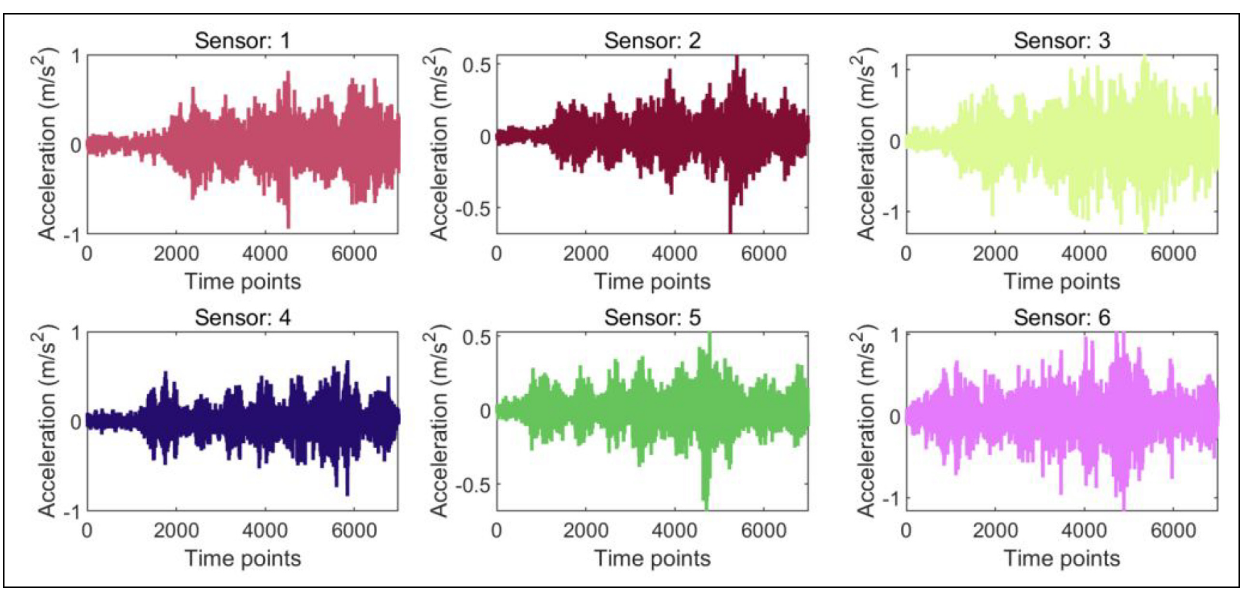

Figure 21. The acceleration signals of six accelerometers.

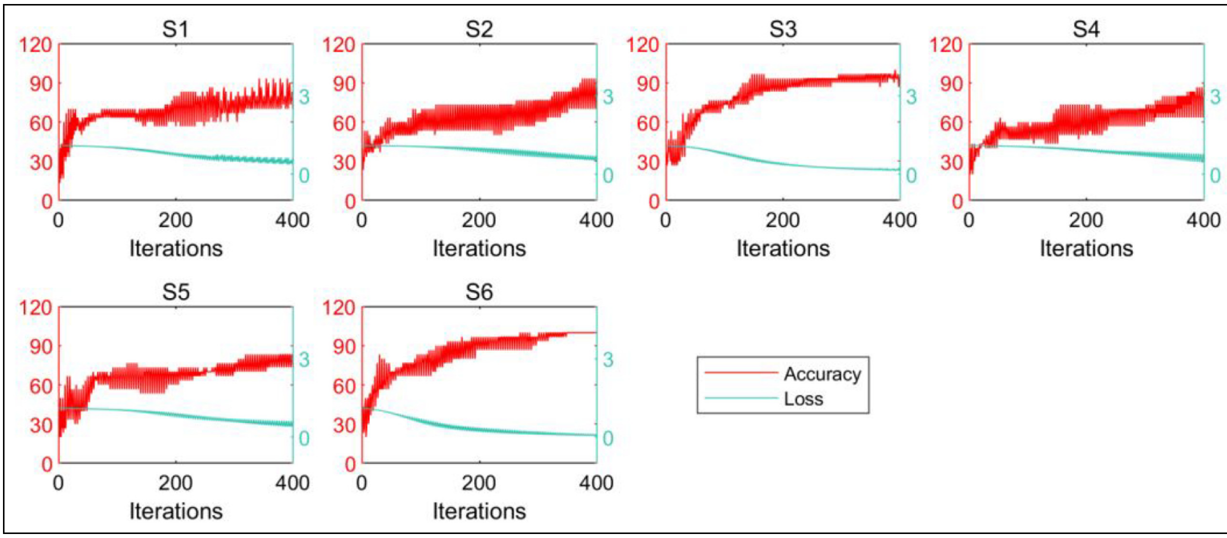

Figure 22. The training process of NTL.
NTL: non-transfer learning; TL: transfer learning.

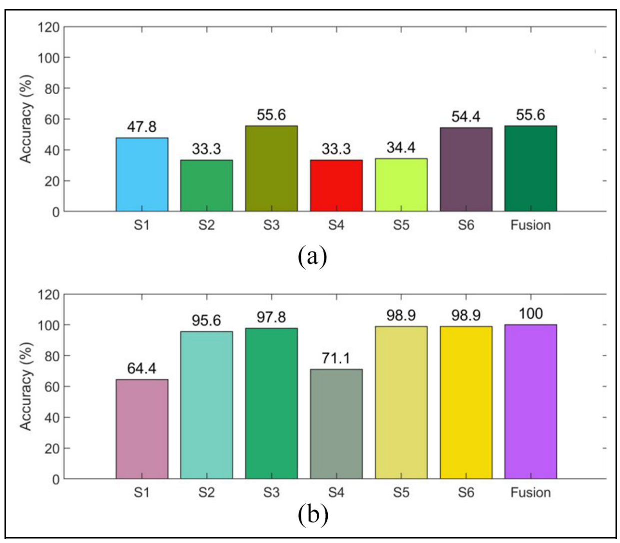

Figure 23. The testing results of the (a) NTL and (b) TL.
NTL: non-transfer learning; TL: transfer learning.

Table 11. Comparison results with other algorithms.

	Number of learning parameters	Detection accuracy and dimensionality of the training samples					
		Dimensionality	Accuracy	Dimensionality	Accuracy	Dimensionality	Accuracy
1-D CNN	104k	720	100%	None	None	None	None
GRU	384.3k	720	67.4%	1440	87.3%	2160	100%

CNN: convolutional neural network; GRU: Gate Recurrent Unit

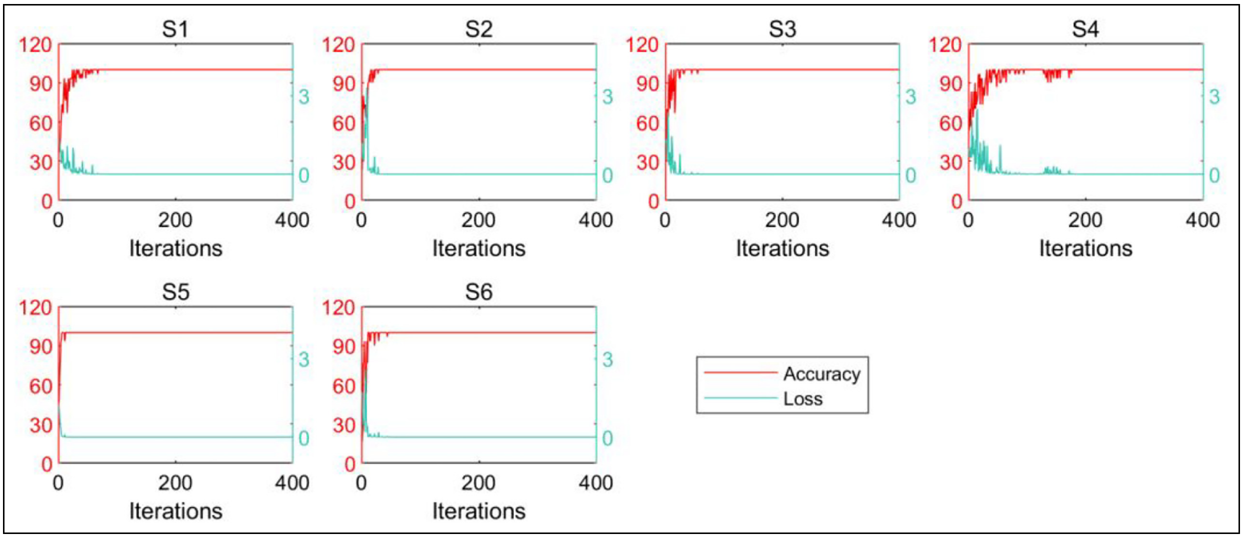

Figure 24. The training process of transfer learning (TL).

(5) Most importantly, with TL, the convergence speed of the CNNs (both numerical and experimental models) increases by more than about 50%.

Author contributions

Shuai Teng: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Gongfa Chen:** Conceptualization, Methodology, Validation, Formal analysis, Data curation, Writing – original draft, Writing – review & editing. **Zhaocheng Yan:** Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Li Cheng:** Methodology, Formal analysis, Data curation, Writing – original draft, Writing – review & editing, Visualization. **David Bassir:** Conceptualization, Resources, Supervision, Visualization.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Gongfa Chen https://orcid.org/0000-0003-1703-3362

References

1. Soleimani-Babakamali MH, Sepasdar R, Nasrollahzadeh K, et al. Toward a general unsupervised novelty detection framework in structural health monitoring. *Comput-Aided Civ Infrastruct Eng* 2022; 37: 1128–1145.
2. Liu J, Xu S, Bergés M, et al. HierMUD: hierarchical multi-task unsupervised domain adaptation between bridges for drive-by damage diagnosis. *Struct Health Monit*; 0: 14759217221081159.
3. Yan YJ, Cheng L, Wu ZY, et al. Development in vibration-based structural damage detection technique. *Mech Syst Signal Process* 2007; 21: 2198–2211.
4. Cawley P and Adams RD. A vibration technique for non-destructive testing of fibre composite structures. *J Compos Mater* 1979; 13: 161–175.
5. Pandey AK, Biswas M and Samman MM. Damage detection from changes in curvature mode shapes. *J Sound Vib* 1991; 145: 321–332.
6. Sung SH, Koo KY and Jung HJ. Modal flexibility-based damage detection of cantilever beam-type structures using baseline modification. *J Sound Vib* 2014; 333: 4123–4138.
7. Lu Q, Ren G and Zhao Y. Multiple damage location with flexibility curvature and relative frequency change for beam structures. *J Sound Vib* 2002; 253: 1101–1114.
8. Teng S, Chen G, Liu G, et al. Modal strain energy-based structural damage detection using convolutional neural networks. *Appl Sci Basel* 2019; 9: 3376.
9. Cha Y and Buyukozturk O. Structural damage detection using modal strain energy and hybrid multiobjective optimization. *Comput-Aided Civ Infrastruct Eng* 2015; 30: 347–358.
10. Li J, Zhu X, Law SS, et al. A two-step drive-by bridge damage detection using Dual Kalman Filter. *Int J Struct Stab Dyn* 2020; 20: 2042006.

11. Tibaduiza DA, Mujica LE, Rodellar J, et al. Structural damage detection using principal component analysis and damage indices. *J Intell Mater Syst Struct* 2016; 27: 233–248.
12. Ni F, Zhang J and Noori MN. Deep learning for data anomaly detection and data compression of a long-span suspension bridge. *Comput-Aided Civ Infrastruct Eng* 2020; 35: 685–700.
13. Khuc T and Catbas N. Completely contactless structural health monitoring of real-life structures using cameras and computer vision. *Struct Control Health Monit* 2017; 24: e1852.
14. Yam LH, Yan YJ and Jiang JS. Vibration-based damage detection for composite structures using wavelet transform and neural network identification. *Compos Struct* 2003; 60: 403–412.
15. Mehrjoo M, Khaji N, Moharrami H, et al. Damage detection of truss bridge joints using Artificial Neural Networks. *Expert Syst Appl* 2008; 35: 1122–1131.
16. Gonzalez MP and Zapico JL. Seismic damage identification in buildings using neural networks and modal data. *Comput Struct* 2008; 86: 416–426.
17. Lautour O and Omenzetter P. Damage classification and estimation in experimental structures using time series analysis and pattern recognition. *Mech Syst Signal Process* 2010; 24: 1556–1569.
18. Hakim S, Razak HA and Ravanfar SA. Fault diagnosis on beam-like structures from modal parameters using artificial neural networks. *Measurement* 2015; 76: 45–61.
19. Chun PJ, Yamashita H and Furukawa S. Bridge damage severity quantification using multipoint acceleration measurement and artificial neural networks. *Shock Vib* 2015; 2015: 1–11.
20. Dackermann U, Li J and Samali B. Dynamic-based damage identification using neural network ensembles and damage index method. *Adv Struct Eng* 2010; 13: 1001–1016.
21. Ghiasi R, Torkzadeh P and Noori M. A machine-learning approach for structural damage detection using least square support vector machine based on a new combinational kernel function. *Struct Health Monit* 2016; 15: 302–316.
22. Zhang S and Cheng L. Wavelet decompositions for high frequency vibrational analyses of plates. *Int J Appl Mech* 2017; 9: 1750088.
23. Zhang S and Cheng L. On the efficacy of the wavelet decomposition for high frequency vibration analyses. *J Sound Vib* 2016; 380: 213–223.
24. Zhong K, Teng S, Liu G, et al. Structural damage features extracted by convolutional neural networks from mode shapes. *Appl Sci Basel* 2020; 10: e4247.
25. Lin YZ, Nie ZH and Ma HW. Structural damage detection with automatic feature extraction through deep learning. *Comput-Aided Civ Infrastruct Eng* 2017; 32: 1–22.
26. Teng S, Liu Z, Chen G, et al. Concrete crack detection based on well-known feature extractor model and the YOLO_v2 network. *Appl Sci Basel* 2021; 11: 813.
27. Yu Y, Wang C, Gu X, et al. A novel deep learning-based method for damage identification of smart building structures. *Struct Health Monit* 2019; 18: 143–163.
28. Kiranyaz S, Ince T and Gabbouj M. Real-time patient-specific ECG classification by 1-D convolutional neural networks. *IEEE Trans Biomed Eng* 2016; 63: 664–675.
29. Abdeljaber O, Sassi S, Avci O, et al. Fault detection and severity identification of ball bearings by online condition monitoring. *IEEE Trans Ind Electron* 2019: 8136–8147.
30. Kiranyaz S, Gastli A, Ben-Brahim L, et al. Real-time fault detection and identification for MMC using 1D convolutional neural networks. *IEEE Trans Ind Electron* 2018; 66: 8760–8771.
31. Abdeljaber O, Avci O, Kiranyaz S, et al. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *J Sound Vib* 2017; 388: 154–170.
32. Avci O, Abdeljaber O, Kiranyaz S, et al. Wireless and real-time structural damage detection: a novel decentralized method for wireless sensor networks. *J Sound Vib* 2018; 424: 158–172.
33. Zhang Y, Miyamori Y, Mikami S, et al. Vibration-based structural state identification by a 1-dimensional convolutional neural network. *Comput-Aided Civ Infrastruct Eng* 2019; 34: 1–18.
34. Teng S, Chen G, Liu Z, et al. Multi-sensor and decision-level fusion-based structural damage detection using a one-dimensional convolutional neural network. *Sensors* 2021; 21: 3950.
35. Kiranyaz S, Avci O, Abdeljaber O, et al. 1D convolutional neural networks and applications: a survey. *Mech Syst Signal Process* 2021; 151: 107398.
36. Dais D, Bal İE, Smyrou E, et al. Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Autom Constr* 2021; 125: 103606.
37. Gao Y and Mosalam KM. Deep transfer learning for image-based structural damage recognition. *Comput-Aided Civ Infrastruct Eng* 2018; 33: 748–768.
38. Li Y, Bao T, Xu B, et al. A deep residual neural network framework with transfer learning for concrete dams patch-level crack classification and weakly-supervised localization. *Measurement* 2022; 188: 110641.
39. Zheng Z, Qi H, Zhuang L, et al. Automated rail surface crack analytics using deep data-driven models and transfer learning. *Sustainable Cities Soc* 2021; 70: 102898.
40. Wang B, Wang B and Ning Y. A novel transfer learning fault diagnosis method for rolling bearing based on feature correlation matching. *Meas Sci Technol* 2022; 33: 125006.
41. Almanifi ORA, Ab Nasir AF, Mohd Razman MA, et al. Heartbeat murmurs detection in phonocardiogram recordings via transfer learning. *Alexandria Eng J* 2022; 61: 10995–11002.
42. Yang Q, Shi W, Chen J, et al. Deep convolution neural network-based transfer learning method for civil infrastructure crack detection. *Autom Constr* 2020; 116: 103199.

43. Teng S, Chen X, Chen G, et al. Structural damage detection based on convolutional neural networks and population of bridges. *Measurement* 2022; 202: 111747.
44. Maes K and Lombaert G. Monitoring railway bridge KW51 before, during, and after retrofitting. *J Bridge Eng* 2021; 26: 04721001.
45. Qu Z, Mei J, Liu L, et al. Crack detection of concrete pavement with cross-entropy loss function and improved VGG16 network model. *IEEE Access* 2020; 8: 54564–54573.
46. Teng S, Chen G, Gong P, et al. Structural damage detection using convolutional neural networks combining strain energy and dynamic response. *Meccanica* 2020; 55: 945–959.

Appendix A

1-D convolution neural network

A standard CNN usually includes a series of convolution layers, pooling layers, activation layers, a fully connected layer, a softmax layer, and an output layer. The raw data (network input) is transferred through a series of layers, and finally mapped to the class to which the raw data belongs. Especially, the input of a 1-D CNN is a 1 × N or N × 1 array. As shown in Figure A1, an N × 1 array goes through a series of convolution and pooling layers, and the class (Class 1, Class 2, and so on) of the array is finally obtained in the output layer.

A convolution process (Figure A2(a)) is to multiply each element in the convolution kernel with the corresponding element in a sub-region (e.g., Green box, Red dotted box) of the raw data of the convolution layer and sum up the products to obtain an element in the feature map. Each time, the sub-region moves down one step and the process is repeated until all elements of the raw data are involved; in the end, the convolution operation will form a new array (i.e., the feature map).

The pooling operation is a down-sampling technique that greatly improves the CNN computational speed and effectively prevents over-fitting. There are usually two different pooling methods, max pooling and mean pooling. Max pooling was utilized in this paper as it is better than mean pooling.[46] Figure A2(b) demonstrates that max pooling picks up the maximum value of a sub-matrix (1 × 2) to form an element of the feature map. The activation layer, softmax layer, and fully connected layer are similar to those in the popular 2-D CNN, which have been described in the relevant reference.[46]

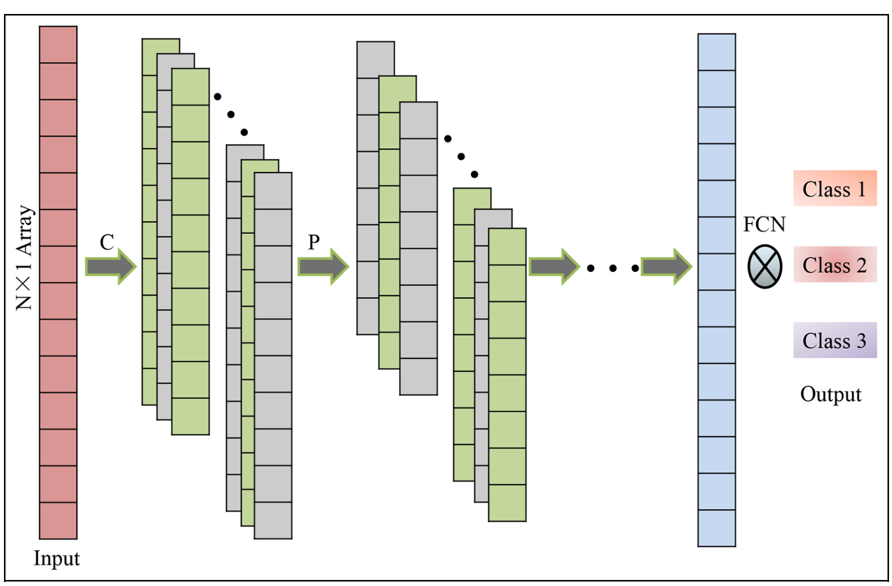

Figure A1. Architecture of the 1-D convolutional neural network (CNN).
C: convolution layer; FCN: fully connected layer; P: pooling layer.

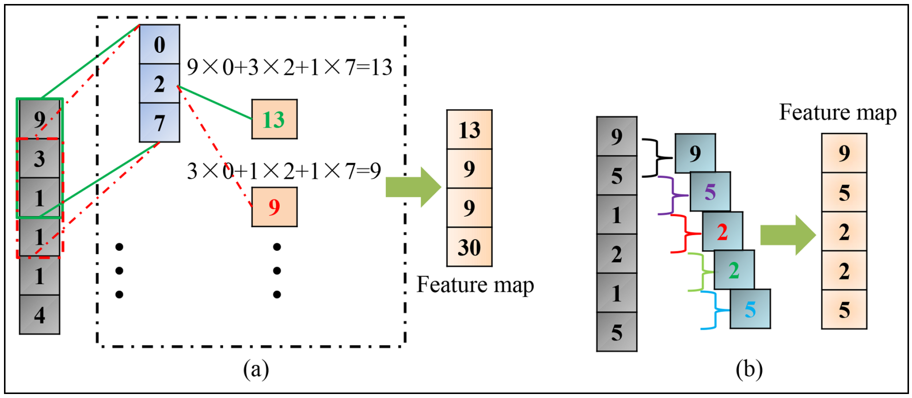

Figure A2. (a) Convolution and (b) pooling operation.

Appendix B

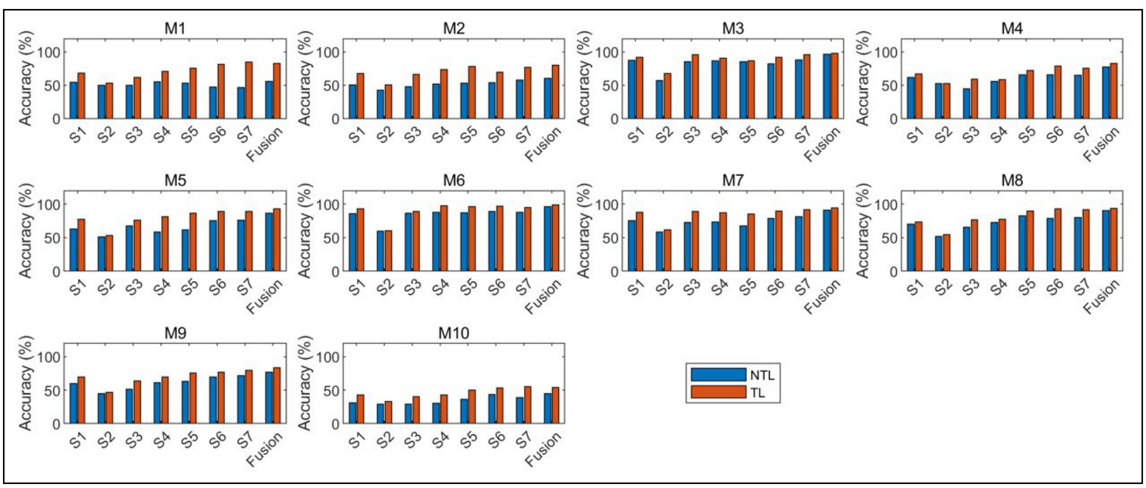

Figure A3. The testing results of the NTL and TL (single damage scenario).
NTL: non-transfer learning; TL: transfer learning.

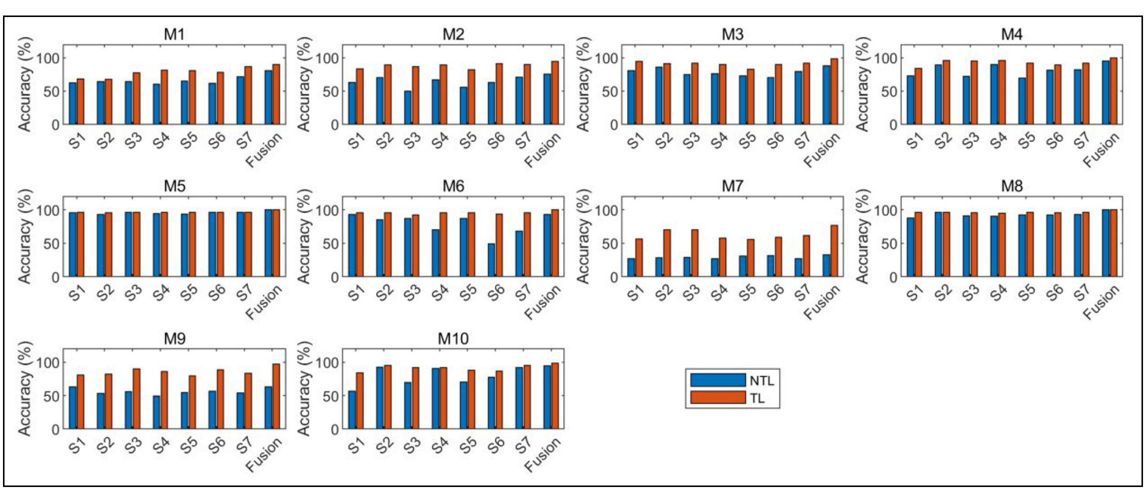

Figure A4. The testing results of the NTL and TL (double damage scenarios).
NTL: non-transfer learning; TL: transfer learning.

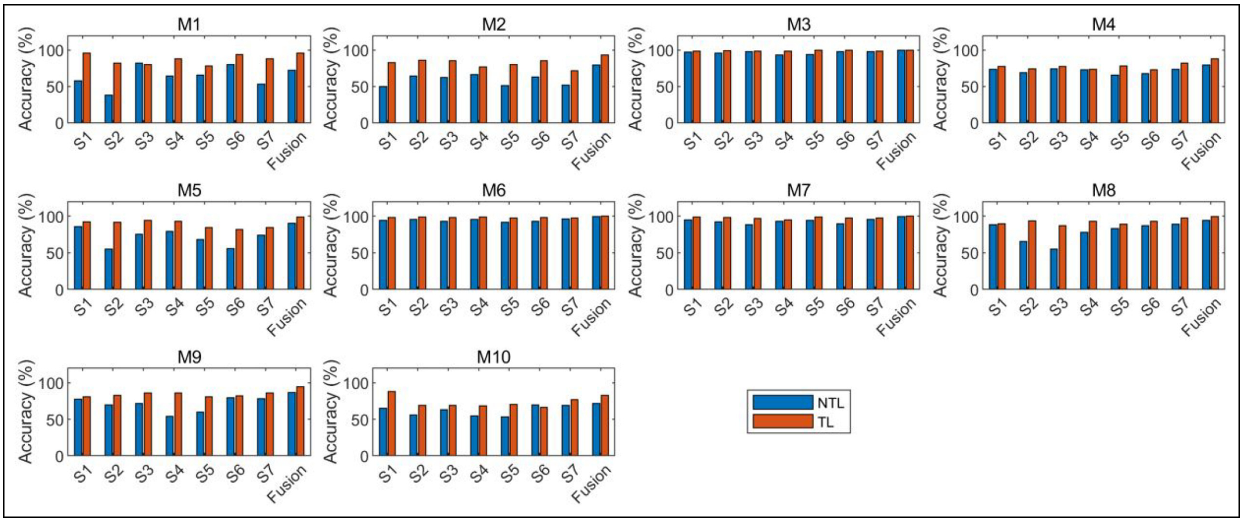

Figure A5. The testing results of the NTL and TL (triple damage scenarios).
NTL: non-transfer learning; TL: transfer learning.