Sequential damage detection approaches for beams using time-modal features and artificial neural networks

Jae-Hyung Park[a], Jeong-Tae Kim[a],*, Dong-Soo Hong[a], Duc-Duy Ho[a], Jin-Hak Yi[b]

[a]*Department of Ocean Engineering, Pukyong National University, Nam-gu, Busan 608-737, Korea*
[b]*Korea Ocean Research & Development Institute, Korea*

Abstract

In this study, sequential approaches for damage detection in beams using time-modal features and artificial neural networks are proposed. The scheme of the sequential approaches mainly consists of two phases: time-domain damage monitoring and modal-domain damage estimation. In the first phase, an acceleration-based neural networks (ABNN) algorithm is designed to monitor the occurrence of damage in a structure by using cross-covariance functions of acceleration signals measured from two different sensors. By using the acceleration feature, the ABNN is trained for potential damage scenarios and loading patterns which are unknown. In the second phase, a modal feature-based neural networks (MBNN) algorithm is designed to estimate the location and severity of damage in the structure by using mode shapes and modal strain energies. By using the modal feature, the MBNN is trained for potential damage scenarios. The feasibility and the practicality of the proposed methodology are evaluated from numerical tests on simply supported beams and also from laboratory tests on free–free beams.
© 2008 Elsevier Ltd. All rights reserved.

1. Introduction

Vibration-based damage assessment is an important topic related to structural health monitoring (SHM). Many researchers have focused on developing reliable techniques that utilize vibration characteristics to estimate changes in structural characteristics [4–6,9]. In general, vibration-based damage detection in structures can be realized by a series of implementations such as signal acquisition, data analysis in time and frequency domains, pattern recognition and system identification process. In order to fulfill the existing vibration-based approaches which are either signal-based or model-based, at least three significant amounts of works are needed (1) to measure acceleration signals, (2) to extract modal parameters such as natural frequencies and mode shapes from the signals, and (3) to post-process modal information suitable for certain

damage detection methods which include damage index methods, genetic algorithm (GA)-based methods, and artificial neural networks (ANN)-based methods [8,10,12,14,21].

Recently, many researchers have made efforts to develop ANN-based methods for identifying the location and the extent of damage in structures [23,24,27]. Others also have attempted to implement the ANN techniques to health monitoring of bridges by using changes in modal properties [1,14,21]. Despite these combined research efforts, there are still outstanding needs: e.g., to detect damage in structures at its threshold stage and to estimate its location and severity. A possible strategy to deal with those needs is to sequentially detect damage occurrence in real time, first, and then estimate its location and severity in detail, next.

However, several problems should be resolved to realize the sequential damage detection approaches. One of the main obstacles is that most of signal process and modal analysis need time-consuming off-line works. Amount of these works depend on the number of sensors involved and the amount of signals recorded. Therefore, it is needed to develop an algorithm that can monitor the occurrence and the extent by using time-domain signals, without any frequency-domain data process, measured from a limited number of sensors [7,11,25]. Although the real-time monitoring is feasible to alarm damage occurrence, there still remains to estimate damage locations and severities in detail. An issue related to the detailed damage assessment is that the accuracy of damage detection is highly dependent on the sensitivity of vibration features used for the operation. Therefore, it is needed to develop an algorithm that utilizes very sensitive modal features to estimate the location and severity of damage.

In this study, we propose sequential damage detection approaches using time-modal features and ANN for structures. First, we describe theoretical backgrounds of the sequential damage detection methods. The scheme of the sequential approaches mainly consists of two phases: time-domain damage alarming and modal-domain damage estimation. In the first phase, an acceleration-based neural networks (ABNN) algorithm is designed to monitor the occurrence of damage in a beam structure by using cross-covariance functions of acceleration signals measured from two different sensors. By using the acceleration feature, the ABNN for damage alarming is trained for potential loading patterns and damage scenarios which are unknown. In the second phase, a modal feature-based neural networks (MBNN) algorithm is designed to estimate the location and severity of damage in the structure by using mode shapes and modal strain energies, which can be extracted from off-line modal analysis tasks. By using the modal feature, the MBNN for damage estimation is trained for potential damage scenarios. Next, the feasibility and the practicality of the proposed methodology are evaluated from numerical model tests on simply supported beams and laboratory model tests on free–free beams for which acceleration signals were measured before and after several damage cases.

2. Sequential approaches for damage detection

A multi-stage diagnosis strategy aims at successive detection of the occurrence, location and severity of damage [13,15,22]. It has many advantages such as efficiency in computational time and better estimation accuracy. Moreover, it is suitable for on-line health monitoring scheme for large structures. To make this multi-stage strategy feasible, sequential approaches for damage detection are designed as schematized in Fig. 1. The scheme of the sequential approaches mainly consists of two phases: acceleration-based damage alarming for real-time damage occurrence and location (*phase I*) and modal feature-based damage estimation for offline damage estimation (*phase II*).

In *phase I*, an ABNN algorithm continuously operates in four steps:

(1) To get accelerations from vibration tests.
(2) To extract time-domain features from accelerations.
(3) To input time-domain features to pre-trained, acceleration feature-based neural networks.
(4) To monitor the occurrence of damage by the ABNN using real-time acceleration features.

Once damage is alarmed, then in *second phase*, a MBNN algorithm operates offline in four steps:

(1) To analyze experimental modal parameters.
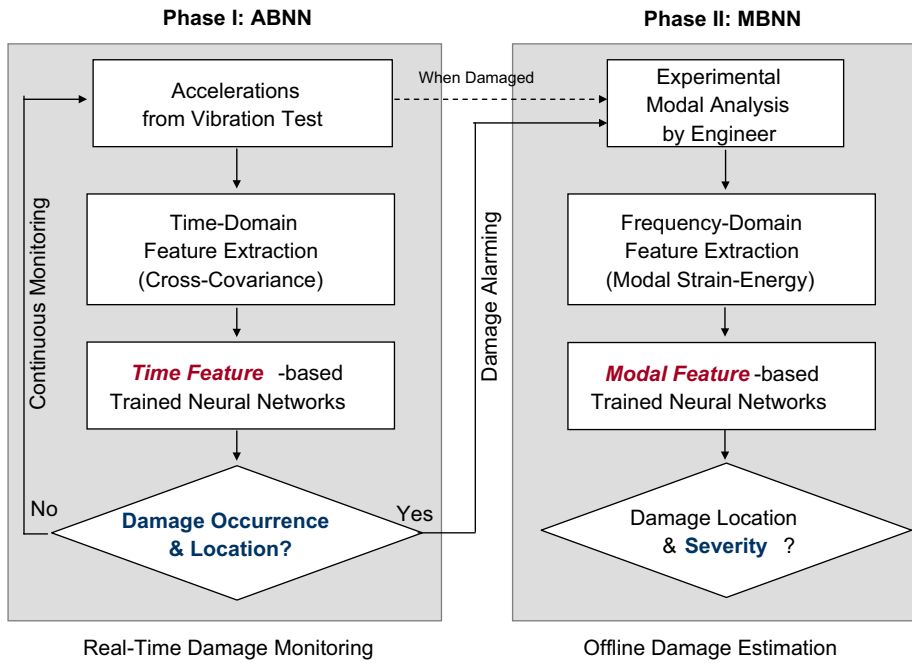(2) To extract frequency-domain features from modal parameters.

Phase I: ABNN　　　　　　　　　　　**Phase II: MBNN**

Fig. 1. Sequential damage detection approaches using artificial neural networks.

(3) To input modal features to pre-trained MBNN.
(4) To estimate the location and severity of damage by MBNN using experimental modal parameters.

2.1. ABNN algorithm

Suppose that we are given an arbitrary structure with NE elements and N nodes. By assuming that the structure behaves linearly, the acceleration response vector at a certain time t for a multi-dof system can be given by

$$\{\ddot{X}\} = [M]^{-1}(\{F\} - [C]\{\dot{X}\} - [K]\{X\}) \tag{1}$$

in which $[M]$, $[C]$ and $[K]$ are, respectively, mass, damping and stiffness matrices of the system; $\{F\}$ is unknown (or unmeasured) excitation force vector; $\{X\}$, $\{\dot{X}\}$ and $\{\ddot{X}\}$ are, respectively, displacement, velocity and acceleration response vectors.

As described in Eq. (1), the acceleration responses at any locations (i.e., $\ddot{X}_1, \ldots, \ddot{X}_i, \ddot{X}_j, \ldots, \ddot{X}_N$) change due to the perturbation of the structural parameters in NE elements. With the known force vector $\{F\}$, the patterns of the dynamic responses can be recognized as the consequence of the changes in physical parameters of the structure [2]. Consequently, the acceleration responses obtained before and after damage can be used as the input for the ANN-based damage detection. However, this approach is limited only when the external forces applying to the real structure is known and identical to the ones that are used for training neural networks. Therefore, the use of the acceleration as the direct input will cause the problem.

As an alternative input, instead, a normalized cross-covariance function of two accelerations measured at ith and jth locations (which are a pair of stationary random processes $\ddot{X}_i$ and $\ddot{X}_j$) is used as follows:

$$\rho_{\ddot{X}_i \ddot{X}_j}(\tau) = \frac{R_{\ddot{X}_i \ddot{X}_j}(\tau) - \mu_{\ddot{X}_i} \mu_{\ddot{X}_j}}{\sigma_{\ddot{X}_i} \sigma_{\ddot{X}_j}} \tag{2}$$

where $\mu_{\ddot{X}_i}$ and $\mu_{\ddot{X}_j}$ are the means; $\sigma_{\ddot{X}_i}^2$ and $\sigma_{\ddot{X}_j}^2$ are the variances; $R_{\ddot{X}_i \ddot{X}_j}(\tau)$ is the cross-correlation function between $\ddot{X}_i$ and $\ddot{X}_j$. The function $\rho_{\ddot{X}_i \ddot{X}_j}(\tau)$ measures the linear dependency between $\ddot{X}_i$ and $\ddot{X}_j$ for a

displacement of τ in $\ddot{X}_j$ relative to $\ddot{X}_i$. Using the cross-covariance function, it is expected to be less sensitive to the patterns of external forces by distinguishing transient free vibration motions from force-induced responses.

The neural networks consist of an input layer, a hidden layer, and an output layer, as shown in Fig. 2. The input/output relationship of the networks can be nonlinear, as well as linear, and its characteristics are determined by the synaptic weights assigned to the connections between the neurons in two adjacent layers. A systematic way of updating the weights to achieve a desired input/output relationship based on a set of training patterns is referred to as training or learning algorithm. The conventional back-propagation algorithm is employed. To train the neural networks, also, the sigmoid function was used as the activation function. For training parameters, we set learning rate 0.3, momentum constant 0.9, error tolerance 0.0001, and epoch number 200. Also, the mean-squared performance function was used to measure the network's performance according to the mean of squared errors. The input layer contains the measured acceleration features (i.e., the cross-covariance function described in Eq. (2)), and the output from the neural networks are element's physical properties. By assuming the mass and damping properties are not changed before and after damage, the output layer consists of the element-level stiffness indices to be identified as [14]

$$S_j = k_{j,d}/k_{j,u} \tag{3}$$

in which k is element stiffness. Subscript j denotes an element number such as $j = 1, 2, \ldots, \mathrm{NE}$ and subscripts u and d denote intact and damaged state, respectively. The element-level damage severity of the jth element is described as

$$\alpha_j = 1 - S_j \tag{4}$$

An ABNN algorithm for damage alarming is newly designed as schematized in Fig. 3. It consists of two parts: (a) training neural networks by using acceleration data and (b) alarming damage occurrence and location. In the first part, a set of ABNN are trained in the following four steps. Firstly, a baseline finite element (FE) model with NE elements is selected for the target structure. Secondly, N numbers of excitation patterns are selected on the basis of potential loading scenarios of the structure, for which each excitation pattern is characterized by intensity and duration of impulse. Thirdly, M numbers of damage patterns are decided on the basis of the potential damaging scenarios, for which each damage pattern is characterized by element-level stiffness losses. Finally, for each of the N excitation patterns, a set of neural networks are trained for the M damaging patterns. The cross-covariance values are computed from two acceleration signals
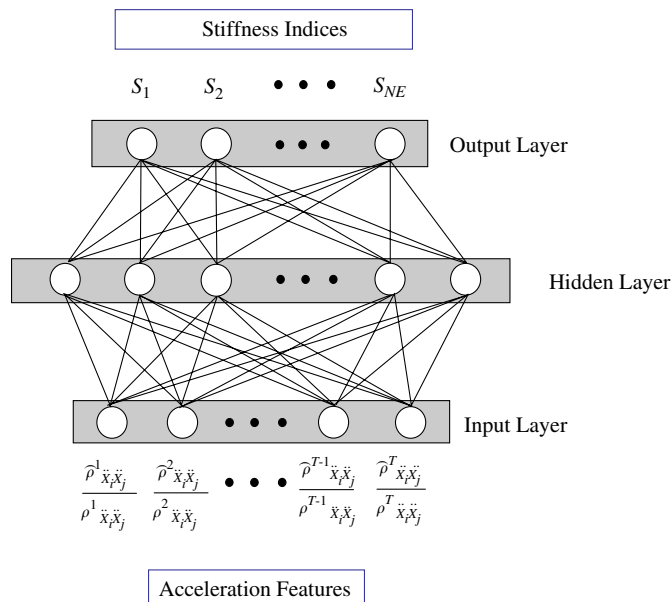

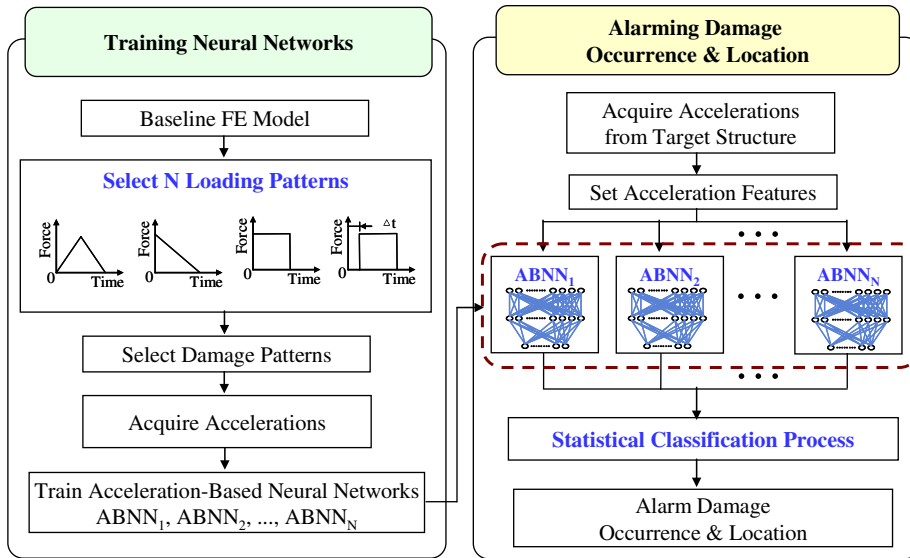Fig. 2. Architecture of back-propagation neural networks.

Fig. 3. Acceleration-based neural networks (ABNN) algorithm for damage alarming.

measured before and after damage. The ratios of the cross-covariance values between before and after damage $(\widehat{\rho}_{\ddot{X}_i\ddot{X}_j}(\tau)/\rho_{\ddot{X}_i\ddot{X}_j}(\tau))$, where the cap denotes the damaged state) are the inputs to the neural networks. Neural networks are repeatedly trained until the N sets of ABNN, the so-called $ABNN_1, \ldots, ABNN_N$, are generated for the N excitation patterns. Each set of neural networks is corresponding to a specific excitation pattern.

Once pre-trained acceleration feature-based neural networks are provided from offline numerical works (i.e., the first part of ABNN in Fig. 3), the second part of ABNN in Fig. 3 will be performed continuously to utilize the trained neural networks for real-time damage monitoring by using measured acceleration features. In the second part, damage is alarmed as the following procedures. Firstly, accelerations are measured at two different locations of the real target structure before and after damage occurred. Secondly, the ratios of the cross-covariance values of the two accelerations between intact and damaged states are computed and furthermore input into the pre-trained neural networks (i.e., $ABNN_1, \ldots, ABNN_N$). Thirdly, element stiffness indices (i.e., S_j of Eq. (3)) and severity indices (i.e., α_j of Eq. (4)) are estimated for NE output elements from the soft computing process. Alarming damage locations is repeatedly performed for each of the pre-trained neural networks. Stiffness indices and severity indices are estimated as S_{ji} and α_{ji} ($i = 1, 2, \ldots, N$, $j = 1, 2, \ldots, NE$), respectively. Finally, the occurrence and location of damage are decided in a statistical classification process. To account for all indices obtained from the N sets of neural networks we form a single damage indicator (DI) for the jth element as follows [10]:

$$\mathrm{DI}_j = \left(\sum_{i=1}^{N} \alpha_{ji}^2\right)^{1/2} \tag{5}$$

where $0 \leqslant \mathrm{DI}_j \leqslant \infty$ and the damage is located at element j if DI_j approaches the local maximum point. Next, the elements are assigned to a damage class via a statistical-pattern-recognition technique, from which the locations of damage are selected on the basis of a rejection of hypothesis. First, the collection of values DI_j ($j = 1, 2, 3, \ldots, NE$) set is treated as a sample population of damage indices and we further assume that the variable distributed normally. Then the damage indices DI_j is normalized according to the standard rule

$$Z_j = (\mathrm{DI}_j - \mu_{\mathrm{DI}})/\sigma_{\mathrm{DI}} \tag{6}$$

in which μ_{DI} and σ_{DI} represent, respectively, the mean and standard deviation of the collection of DI_j values. The null hypothesis (i.e., H_0) is taken to be that the structure is undamaged at the jth element and the alternate hypothesis (i.e., H_1) is taken to be that the structure is damaged at the jth element. Damage is assigned to a particular location by the following decision rule: (1) choose H_1 if $Z_j \geqslant z_0$; and (2) choose H_0 if $Z_j < z_0$, where

z_0 is number which depends upon the confidence level of the localization test and is selected empirically. If z_0 is set to 1.5 and confidence value of an element is larger than 1.5, for example, it means that damage in the element is predicted with probability larger than 93.3 percent.

2.2. MBNN algorithm

Neural networks algorithms using modal features require modal identification process to extract natural frequency, mode shape, and damping coefficient. Although it is difficult to be utilized for on-line health monitoring, modal parameters are used very much in the areas of system identification, damage detection, and performance evaluation of structures. Among the modal parameters, mode shapes are relatively less influenced by environmental conditions (e.g. temperature, wind, etc.) than natural frequencies and damping coefficients [14]. From the previous studies, modal shape curvatures or modal strain energies are relatively more sensitive to damages than mode shapes [10,16]. In this study, therefore, modal strain energies are utilized for damage estimation in structures.

Modal strain energies are computed by numerically differentiating the identified mode shape vectors. First, the mode shapes are identified from the acceleration data. Next, the coarse mode shapes are interpolated into the finer ones using a cubic-spline function. The interpolation process is needed to detect damage in finer resolution since the number of sensors used for mode shapes is limited. For the same structural system previously described in Eq. (1), the ith modal strain energy allocated in the jth element (between x_k and x_{k+1}) is given by [9]

$$U_{ij} = \frac{1}{2} k_j \int_{x_k}^{x_{k+1}} [\phi_i''(x)]^2 \, \mathrm{d}x \tag{7}$$

where k_j is element's stiffness and $\phi_i''(x)$ is ith modal curvature function about the beam's longitudinal axis. Dividing L.H.S. of Eq. (7) by the unknown jth element stiffness (k_j) gives the modal strain energy in terms of the measurable modal curvature function as follows:

$$\theta_{ij} = 2U_{ij}/k_j = \int_{x_k}^{x_{k+1}} [\phi_i''(x)]^2 \, \mathrm{d}x \tag{8}$$

where θ_{ij} is a function of relative modal strain energy for ith mode and jth element. As described in Eq. (8), the change in the relative modal strain energy of a location can be recognized as the consequence of the change in the stiffness at the same location.

A MBNN algorithm for damage estimation is newly designed as schematized in Fig. 4. It consists of two parts: (a) training neural networks by using modal strain energy and (b) estimation of damage location
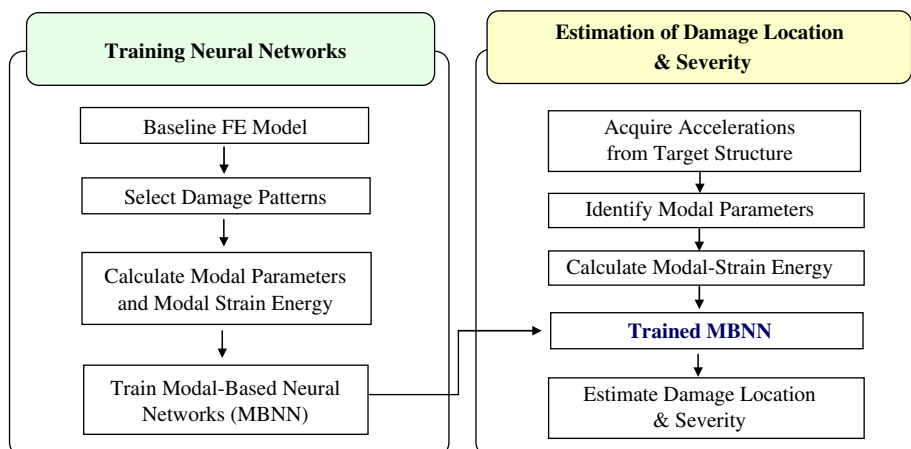

Fig. 4. Modal-based neural networks (MBNN) algorithm for damage estimation.

and severity. Its overall process is similar to the ABNN algorithm, except loading patterns for training multiple neural networks and statistical classification process for deciding the occurrence and location of damage. In the first part, modal strain energy-based neural networks are trained in the following four steps. Firstly, a baseline FE model with NE elements is selected for the target structure. Secondly, M numbers of damage patterns are selected based on the potential damaging scenarios. Thirdly, modal parameters are computed from numerical modal analysis tasks and corresponding modal strain energies are calculated for the baseline model. Finally, neural networks are trained for the M damaging patterns. The ratios of the modal strain energies between intact and damaged state are used as the inputs into the neural networks.

In the second part, the location and severity of damage are estimated in the following procedures. Firstly, accelerations are measured at several locations of the real target structure before and after damage occurred. Secondly, modal parameters such as natural frequencies and mode shapes are identified from the accelerations. Thirdly, modal strain energies corresponding to the identified modes are computed for the target structure. Finally, the locations and severities of damage are estimated from the pre-trained neural networks, MBNN. Damage localization indices are decided in a statistical classification process as described in Eqs. (5) and (6). Also, damage severity indices are estimated as described in Eq. (4).

3. Numerical verification

3.1. Description of numerical test

A simply supported beam model that consists 8 elements with equal size ($L^{EL} = 0.28$ m) was selected as shown in Fig. 5. Dynamic responses of the beam model were obtained before and after damage. The geometrical and material properties of the model are as follows: length $L = 2.24$ m, moment of inertia $I = 1.0856 \times 10^{-6}$ m, elastic modulus $E = 210$ GPa, Poisson's ratio $v = 0.3$ and mass density $\rho = 7850$ kg/m^3. Impact loads were applied at node 7 and acceleration responses were acquired from two locations (i.e., nodes 2 and 3 in Fig. 5(b)) in the z-direction. Acceleration responses were analyzed by using commercial FEA software MIDAS/CIVIL. Sampling frequency was set to 10 kHz and total 16,384 discrete data were gathered from each test.
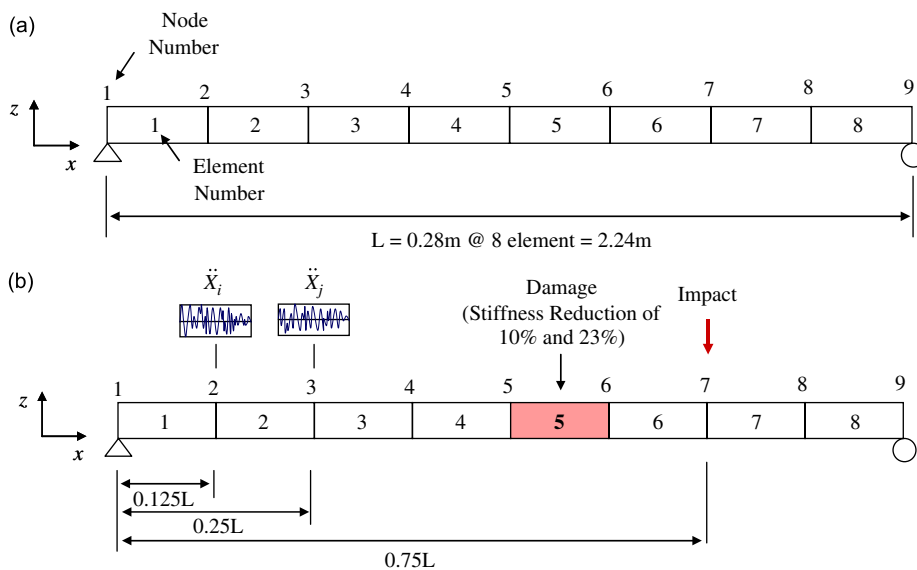
Fig. 5. Geometry and numerical test lay-out on simply supported beam: (a) simply support beam model for numerical tests; (b) numerical test lay-out.

3.2. Damage alarming by ABNN algorithm

In this proposed method, model uncertainty in neural networks may exist due to the difference between the excitation pattern selected for training neural networks and the excitation force applied for acquiring acceleration data to alarm the location of damage. For example, Fig. 6 shows three impact loads that were experimentally measured from impact hammer tests in the lab. These impact excitation patterns show different in their shapes, durations and intensities. Although the impact forces can be measured, it may be difficult that the impact excitation patterns be equalized with the excitation patterns used for training neural network. Therefore, it is needed to evaluate the feasibility of the acceleration-based ANN method under the effect of the model uncertainty due to the variability of impulse excitation patterns used for training neural networks.

3.2.1. Training ABNN for test beam

In order to examine the variability of impact loading parameters (i.e., pulse-shape, duration and intensity), eight loading patterns were selected and classified into two sets. The first four excitations, excitation 1–4 (set 1 presented in Table 1), are triangular pulses with different durations (0–0.005, 0–0.01, and 0–0.02 s) and intensities (0.5 or 1.0 percent of beam weight). The next four excitations, excitations 5–8 (set 2 presented in Table 2), are different in pulse shapes (triangular, right-triangular, rectangular, or delayed-rectangular) while duration 0–0.01 s and intensity 1.0 percent are fixed for all pulses. Each loading pattern generates an independent set of neural networks. Excitations 1–4 (set 1) generate their corresponding four sets of neural networks, $ABNN_1-ABNN_4$. Also, excitations 5–8 (set 2) generate the other four sets $ABNN_5-ABNN_8$.

Total 81 damage scenarios (including an intact case) were selected to train neural networks for each loading pattern. For each element as a single damage location, the element stiffness loss (as described in Eq. (4)) was simulated between 5 and 50 percent with a step size of 5 percent for each of the eight elements. The number of nodes for input layer was selected as 50 from trial and error by considering computational time and efficiency of computer memory. As shown in Figs. 8 and 9, the first 50 units of cross-covariance ratios contain certain features of vibration responses. A good guess for the number of hidden neurons would be to put an average of the number of input and output neurons. Another possibility would be to make the hidden layer of the same
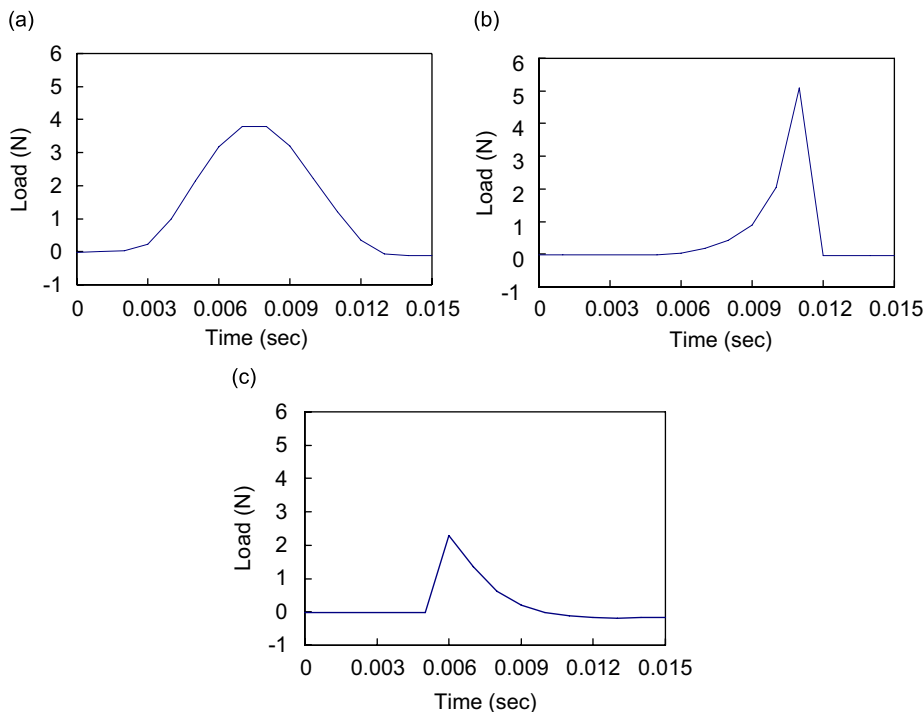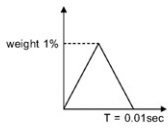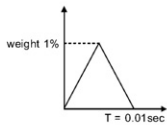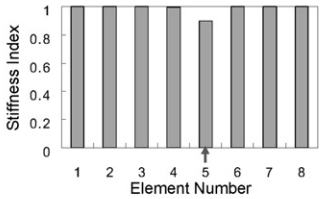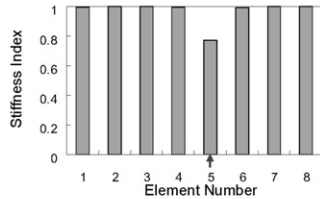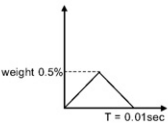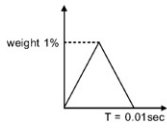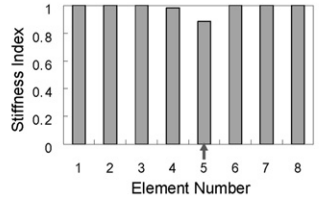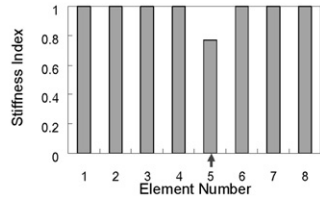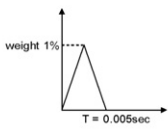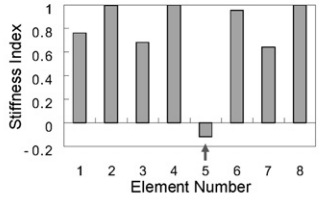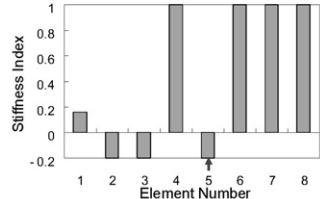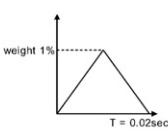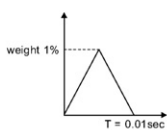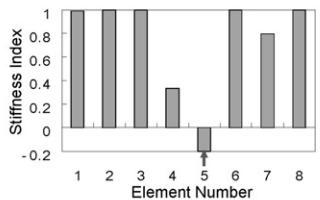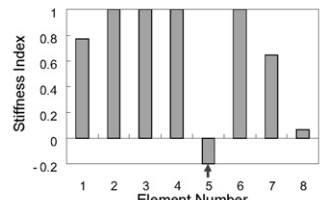

Fig. 6. Experimental excitation patterns measured from impact hammer test: (a) impact 1; (b) impact 2; (c) impact 3.

Table 1
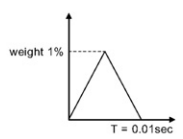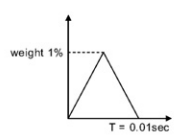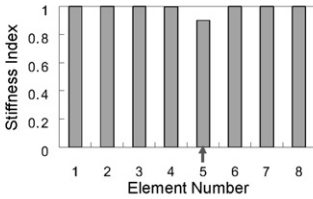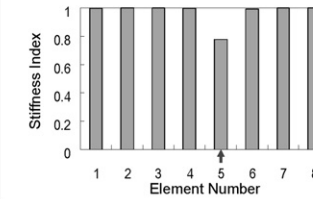Stiffness indices estimated by excitation patterns: set 1 (ABNN$_1$–ABNN$_4$).
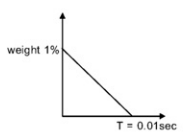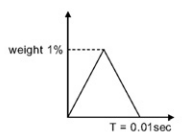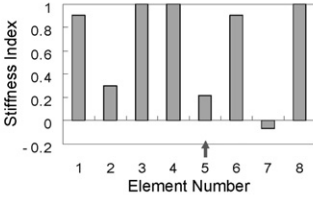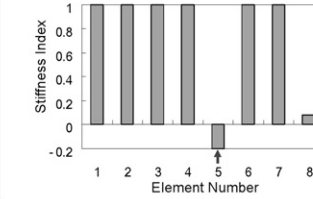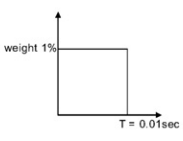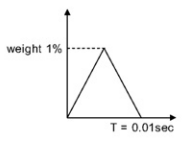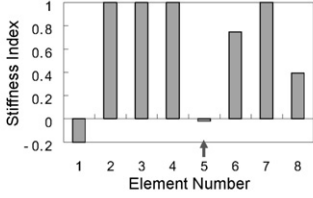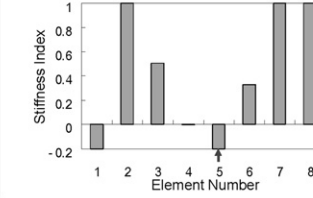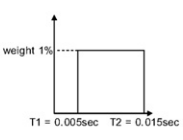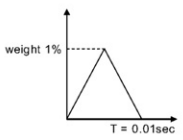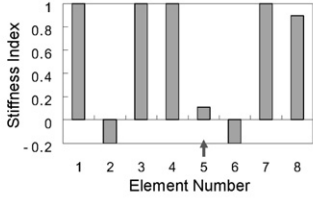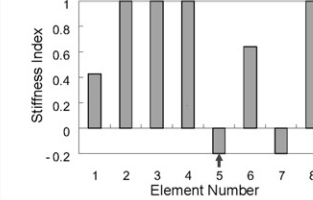
size as either the input or the output layer [1]. In this study, we made the hidden layer of the same size as the input layer (i.e., 50 nodes). For the number of output neurons, we selected eight units of element stiffness indices which were allocated to eight elements of the beam model. The 50 units of acceleration features of the input layer are the first 50 values of cross-covariance ratios between the intact and a damaged state. Either the intact or a damaged state, cross-covariance signals were obtained from two nodes in Fig. 5 (i.e., acceleration signals $\ddot{X}_i$ and $\ddot{X}_j$). Fig. 7 shows two acceleration signals and their cross-covariance function obtained for the case of excitation 1. Figs. 8 and 9 show the first 50 values of cross-covariance ratios (between intact and damaged states of total 81 damage scenarios) obtained for the cases of excitations 1–4 and excitations 5–8, respectively.

3.2.2. Alarming damage occurrence and location in test beam

Damages were introduced by stiffness reduction of 10 percent and 23 percent at element 5 (i.e., $\alpha_5 = 0.1, 0.23$), as shown in Fig. 5. Two acceleration signals were obtained from nodes 2 ($x/L = 0.125$) and 3 ($x/L = 0.25$), which are equivalent locations used for training neural networks. For alarming damage, triangular pulse with duration of 0–0.01 s and pulse intensity of 1.0 percent of beam weight, which is the same as excitation 1, was applied to the beam model to acquire acceleration data. As described previously, initial 50 cross-covariance ratios between intact and damaged state were computed from two acceleration signals.

Table 2
Stiffness indices estimated by excitation patterns: set 2 (ABNN$_5$–ABNN$_8$).

Firstly, we estimated the stiffness indices of the beam model. Table 1 shows the stiffness indices estimated from set 1 (ABNN$_1$–ABNN$_4$). Also, Table 2 shows the stiffness indices estimated from set 2 (ABNN$_5$–ABNN$_8$). As presented in Table 1, the results of stiffness estimation were accurate in cases 1 and 2 but inaccurate in cases 3 and 4. In case 1, the stiffness indices were correctly identified by indicating 10 percent stiffness-loss and 23 percent stiffness-loss at element 5. As shown in Fig. 8(a), the cross-covariance ratios for both training neural networks (by excitation 1) and alarming damage (by excitation 1) were identical as their excitation patterns were the same. In case 2, the stiffness indices were correctly estimated at element 5. As shown in Figs. 8(a) and (b), the cross-covariance ratios for both training ABNN (by excitation 2) and alarming damage (by excitation 1) were identical although the pulse intensities were different. In case 3, the stiffness indices were estimated with relatively low accuracy (but indicative to the damaged location by the minimum stiffness index). As shown in Figs. 8(a) and (c), the cross-covariance ratios for both training ABNN (by excitation 3) and alarming damage (by excitation 1) were not identical but different functions when the two excitation patterns were different in the pulse durations. Similar problem occurs in case 4 (i.e., excitation 4 for training and excitation 1 for damage monitoring).

For cases 3 and 4, modeling errors due to the difference in pulse duration cause ill-posedness problems and have relatively large impact on the accuracy of damage detection. Due to the ill-posedness, the ABNN output the outbound values with negative. To identify the event of the ill-posedness and the certain elements affected
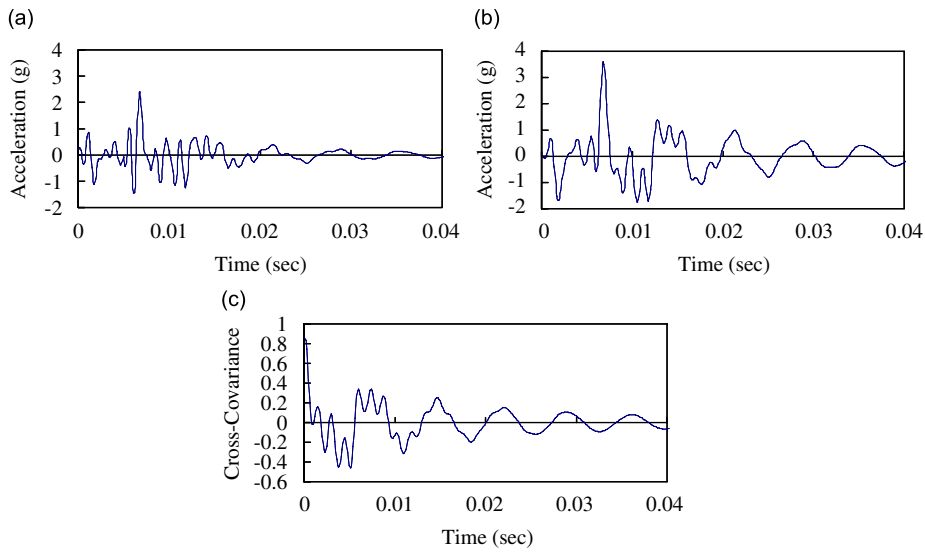
Fig. 7. Two acceleration signals and their cross-covariance function for beam model: (a) acceleration ($\ddot{X}_i$); (b) acceleration ($\ddot{X}_j$); (c) cross-covariance ($\rho_{\ddot{X}_i\ddot{X}_j}$).
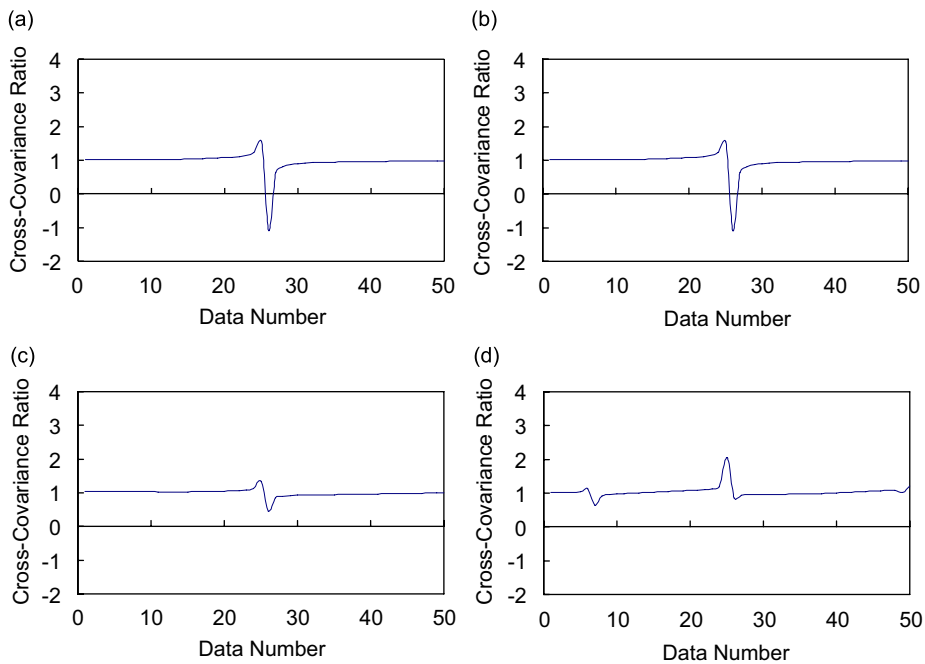
Fig. 8. Cross-covariance ratios between intact and damage state (10 percent stiffness-loss at element 5) for training ABNN: excitations 1–4: (a) excitation 1; (b) excitation 2; (c) excitation 3; (d) excitation 4.

by the problem, the lower bound of output was set to −0.2 instead of 0. Although the negative values were not physically defined, they were indicative to the locations of damage as given in Table 1. Note that the stiffness indices consistently indicate the damaged locations in Tables 1 and 2.

Secondly, we utilized the statistical damage classification process, which is described in the previous section, to detect damage under the uncertainty related to the completeness of neural networks. On assuming the stiffness indices distributed normally, normalized damage indices were generated from Eq. (6). The confidence
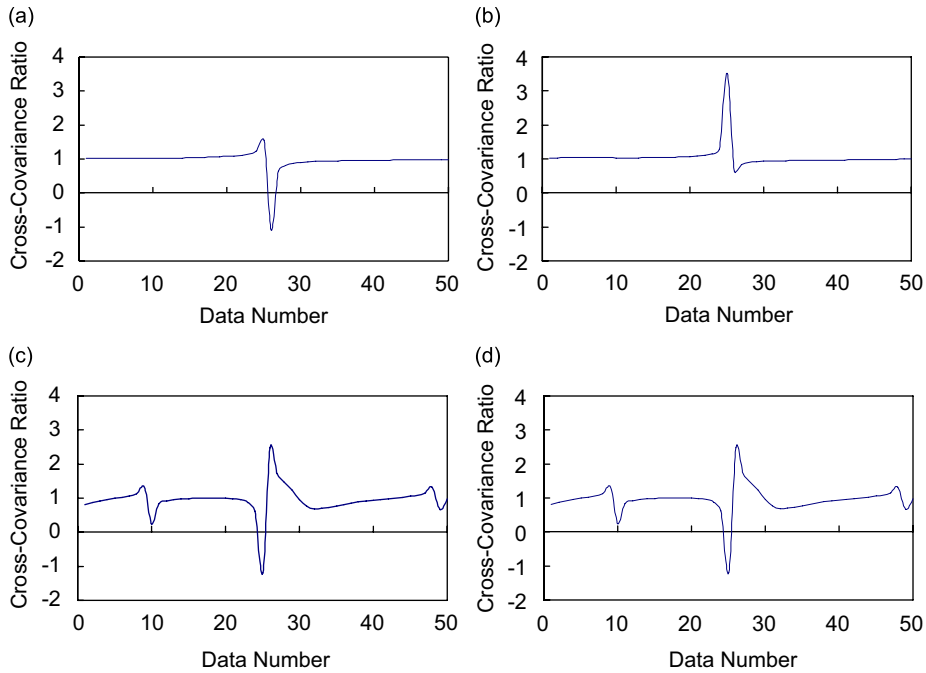
Fig. 9. Cross-covariance ratios between intact and damage state (10 percent stiffness-loss at element 5) for training ABNN: excitation 5–8: (a) excitation 5; (b) excitation 6; (c) excitation 7; (d) excitation 8.
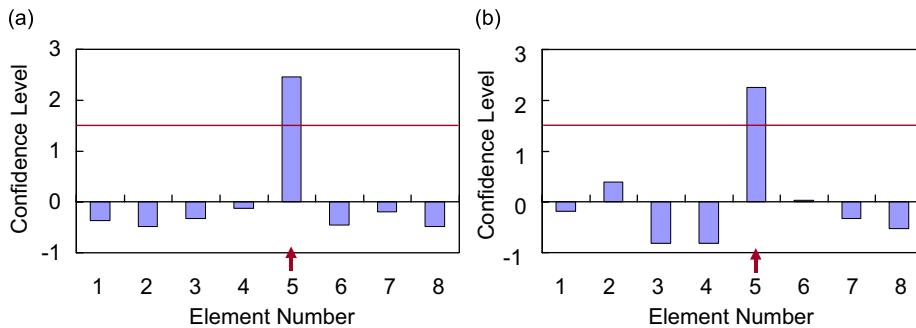
Fig. 10. Damage localization results for beam model (10 percent stiffness-loss at element 5): (a) result by Set 1 ($ABNN_1 - ABNN_4$); (b) result by Set 2 ($ABNN_5 - ABNN_8$).

level for the localization corresponded to $z_0 = 1.5$. This criterion corresponds to a one-tailed test at a confidence level of 93.3 percent. The damage localization results for sets 1 and 2 are shown in Figs. 10 and 11, respectively. Both cases of 10 and 23 percent stiffness-loss for sets 1 and 2, element 5 were predicted where damage was inflicted. In Figs. 10 and 11, the negative confidence level means that the occurrence of damage (i.e., stiffness loss) has no probability at the indicated location.

3.3. Damage estimation by MBNN algorithm

3.3.1. Training MBNN for test beam

We performed the detailed damage estimation in the same beam model as shown in Fig. 5. We first generated MBNN and next utilized them to estimate the location and severity of damage in the test structure. To train MBNN, we performed numerical modal analyses from which natural frequencies and mode shapes
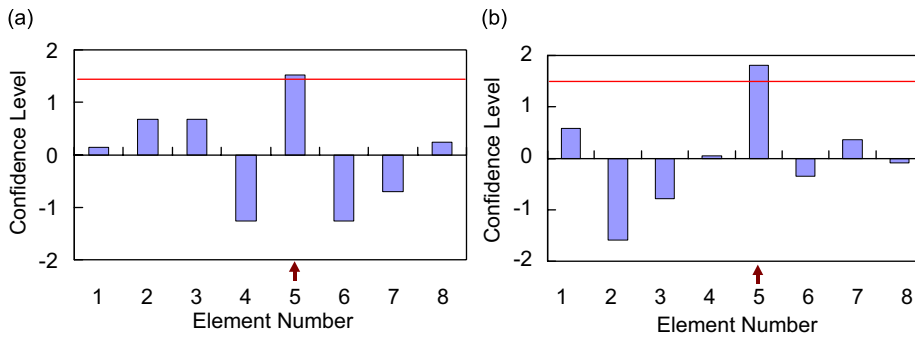
Fig. 11. Damage localization results for beam model (23 percent stiffness-loss at element 5): (a) result by Set 1 ($ABNN_1 - ABNN_4$); (b) result by Set 2 ($ABNN_5 - ABNN_8$).

Fig. 12. Numerical mode shapes and modal strain energies of the simply supported beam: (a) mode shapes; (b) modal strain energies.

were identified for the first four bending modes by using commercial FEA software MIDAS/CIVIL [17]. We considered single damage cases for which each element was selected as a damage-inflicted location and the stiffness loss (as described in Eq. (4)) was simulated between 10 and 50 percent with a step size of 10 percent. That makes 41 single-damage patterns for the eight beam elements.
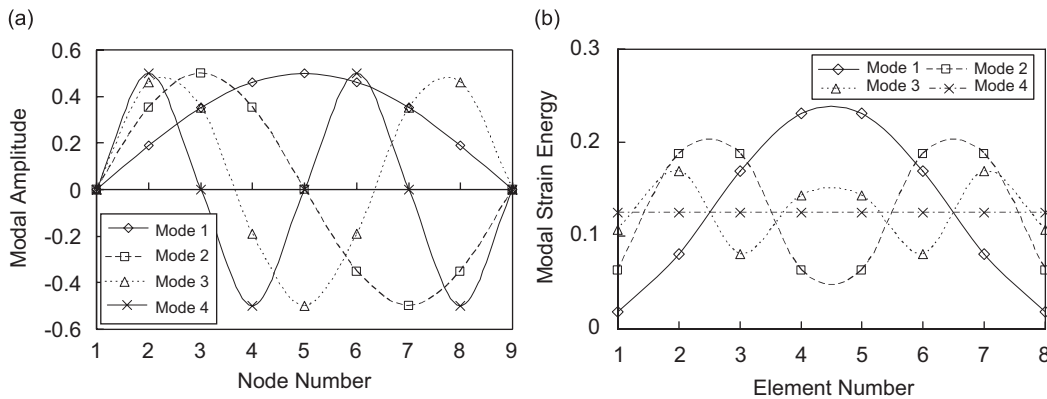
The MBNN consisted of three layers (as illustrated in Fig. 2): (1) an input layer with 32 units of modal features, (2) a hidden layer with 32 units, and (3) an output layer with eight units of element stiffness indices, each of which was allocated to each beam element. The 32 units of the input modal features are corresponding to the ratios of the modal strain energies, which were obtained for the four modes and eight elements, between undamaged and damaged states. Fig. 12 shows the four mode shapes of the baseline model and the corresponding modal strain energies calculated from Eq. (8).

3.3.2. Estimating damage location and severity in test beam

Damage inflicted in the test structure was estimated by the trained MBNN. As described previously, damages were introduced into element 5 by stiffness reduction of 10 and 23 percent (i.e., $\alpha_5 = 0.1, 0.23$). Changes in modal strain energies for the four modes and the eight elements, which were numerically obtained from pre- and post-damage states, were input into the pre-trained neural networks, MBNN. For the first mode, Fig. 13 shows the changes in modal strain energies and their ratios between pre- and post-damage states. Then stiffness indices of the eight elements were estimated as the output. Figs. 14(a) and 15(a) show the element stiffness indices predicted for each damage case. By implementing the element stiffness indices into Eqs. (5) and (6), the confidence levels of damage localization were computed for the damage case as shown in Figs. 14(b) and 15(b). By setting $z_0 = 2.0$ which gives 97.2 percent confidence level, damage locations were correctly predicted in element 5. At the predicted location, the severities of damages could be identified from the estimated stiffness indices shown in Figs. 14(a) and 15(a). They were correctly estimated as 10 and 23
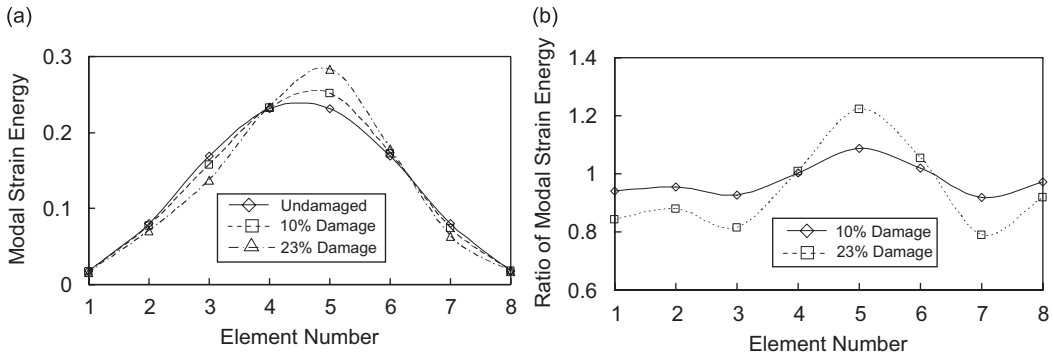
Fig. 13. Modal strain energies of undamaged and damage cases: (a) changes in modal strain energy; (b) ratios of modal strain energy.
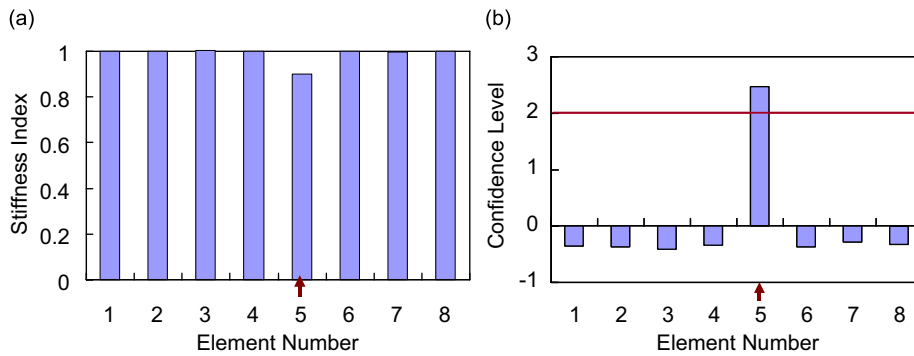
Fig. 14. Damage detection results by MBNN algorithm (10 percent stiffness-loss at element 5): (a) predicted stiffness indices; (b) damage localization result.
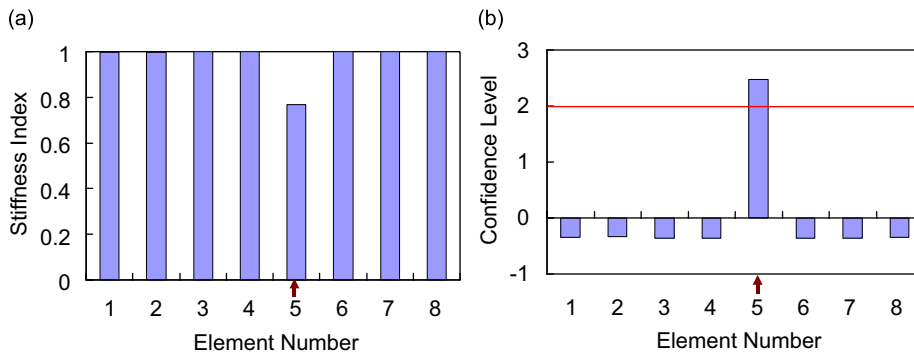
Fig. 15. Damage detection results by MBNN algorithm (23 percent stiffness-loss at element 5): (a) predicted stiffness indices; (b) damage localization result.

percent ($\alpha_5 = 0.1$, 0.23). Note that the damage localization was performed in element scale by which we can define an element as a damage location.

4. Experimental verification

4.1. Test structure and experimental setup

Dynamic tests were performed on a target structure to evaluate the feasibility and the practicality of the present damage detection approaches. As shown in Fig. 16, a free–free, aluminum beam was selected and
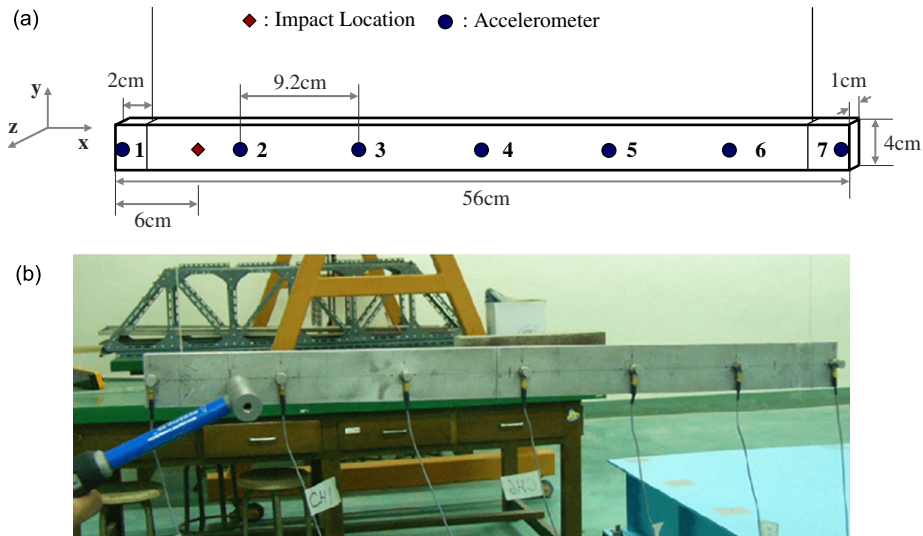
Fig. 16. Experimental setup on free–free beam: (a) free–free beam; (b) experimental setup.

dynamic responses of the structure were measured for several damage cases. The geometrical properties of the test structure are as follows: the length $L = 56$ cm and the rectangular cross-section $t \times H = 1$ cm $\times 4$ cm. The material properties of the test structure are as follows: the elastic modulus $E = 70$ GPa, Poisson's ratio $\upsilon = 0.33$ and mass density 2700 kg/m^3.

Experimental layouts with locations and arrangements of accelerometers are also shown in Fig. 16. Seven accelerometers were selected to measure the motion of the structure in the z-direction (i.e., the vertical direction to sense out-of-plane motions) and equally distanced (i.e., 9.2 cm) along the longitudinal direction. The impulse load was applied to a location 6 cm distanced from the left edge. The impulse was made by hand-hammering but not controlled. Each accelerometer (Dytran 3101BG miniature sensor) was mounted along the center line. A data acquisition system which includes 16 channels National Instruments PXI-4472 DAQ [18], PXI-8186 controller [19] and LabVIEW [20] was set up to measure signals from the accelerometers. Sampling frequency was set to 8.0 kHz and total 8450 discrete data were acquired from each measure.

Fig. 17 shows two acceleration signals measured from sensor 3 ($x/L = 0.32$) and sensor 4 ($x/L = 0.5$) and their cross-covariance function, respectively. To update the baseline model, modal characteristics of the test beam were extracted from measured acceleration-response signals by using frequency-domain decomposition (FDD) technique [3,26]. Fig. 18 shows the singular values (and the corresponding resonant frequencies) calculated by singular value decomposition of power spectral density (PSD) matrix. The figure identified the natural frequencies of the initial four modes of the intact state (as also listed in Table 3).

Damage was inflicted into the beam by sawing cuts at two different locations, as shown in Fig. 19. Three different scenarios of damage were considered as follows: (1) damage case 1—a single damage at $x/L = 0.464$ (26 cm distanced from the left edge) with severity $a/t = 0.25$, $a = 2.5$ mm (crack depth), (2) damage case 2—a single damage at $x/L = 0.464$ with $a/t = 0.5$, and (3) damage case 3—two damages at $x/L = 0.464$ and $x/L = 0.939$ (3.4 cm distance from the right edge) with $a/t = 0.5$ and 0.25, respectively. For these three damage cases, impulse-response tests were performed to measure acceleration signals from seven sensors. Mode shapes and natural frequencies were extracted from the acceleration signals. Table 3 summarizes the first four natural frequencies of the undamaged and three damaged case.

4.2. Damage alarming in test structure by ABNN algorithm

4.2.1. Training ABNN for test structure

As shown in Fig. 20, we selected the baseline model that consists of 12 beam elements with equal size ($L^{EL} = 4.6$ cm) and with uniform bending rigidity (EI $= 233.3$ N m^2). The baseline model was made by using commercial FEA software MIDAS/CIVIL. The representation of the baseline model was guaranteed by

(a)

(b)

(c)
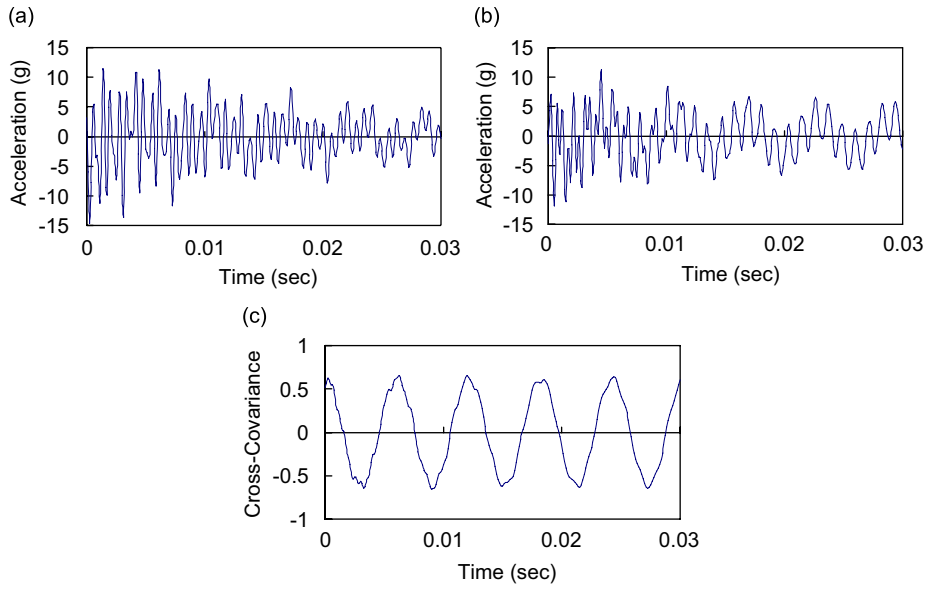
Fig. 17. Measured accelerations and their cross-covariance function: (a) acceleration 1 from sensor 3; (b) acceleration 2 from sensor 4; (c) cross-covariance function.
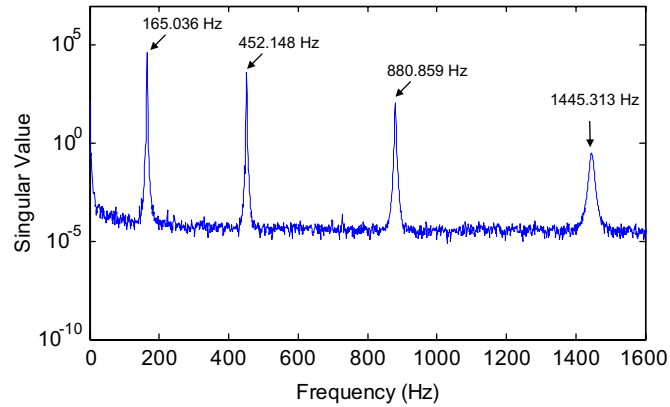
Fig. 18. Singular values of PSD matrix of acceleration signals.

Table 3
Natural frequencies (Hz) of free–free beams.

Modes	Experimental results				Baseline model	
	Undamaged	Damage 1	Damage 2	Damage 3	Undamaged	Error (%)
1	165.039	163.085	158.203	158.203	165.218	0.11
2	452.148	451.171	451.171	450.195	453.630	0.33
3	880.859	873.046	856.445	885.469	888.845	0.91
4	1445.313	1442.383	1436.523	1432.617	1463.382	1.25
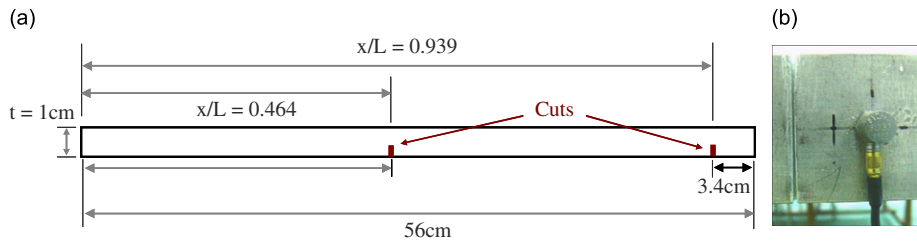
Fig. 19. Schematic of damage inflicted in test structure: (a) damage locations; (b) saw cut.
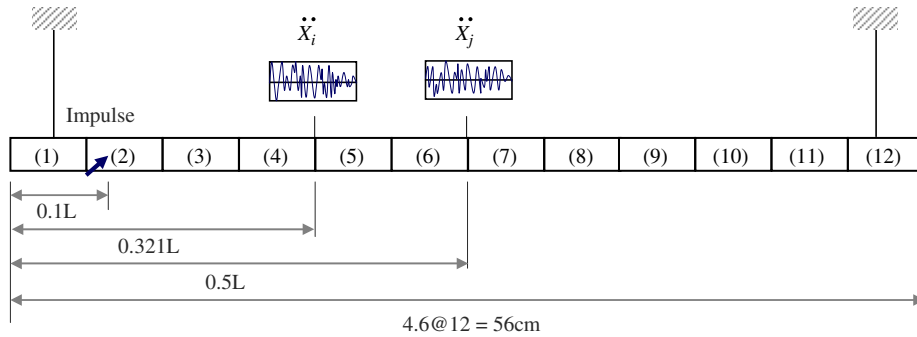
Fig. 20. Schematic of baseline model.
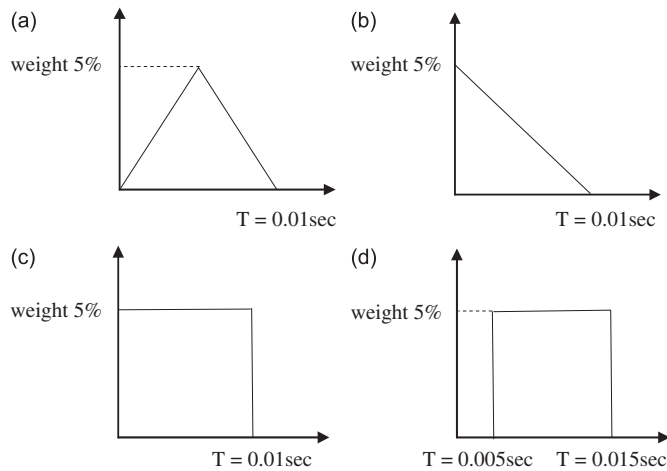
Fig. 21. Assumed excitation patterns for unknown external loads: (a) excitation 1; (b) excitation 2; (c) excitation 3; (d) excitation 4.

identifying its modal parameters with respect to experimental results. As listed in Table 3, natural frequencies of the baseline model show good matches, with very small errors, to the corresponding experimental results.

To train ABNN, we performed acceleration signal acquisitions by numerical impulse-response analyses of the baseline model. First, we selected a set of excitation types to simulate unknown impulse loadings. Due to the lack of prior knowledge on the real impulse measurements, we should rely on a set of potential excitation patterns. As shown in Fig. 21, four excitation types were selected as follows: (1) excitation 1—triangular pulse with 0–0.01 s duration, (2) excitation 2—right triangular pulse with 0–0.01 s duration, (3) excitation 3—rectangular pulse with 0–0.01 s duration, and (4) excitation 4—rectangular pulse with 0.005–0.015 s duration. Pulse intensity was set to 5 percent of self-weight of the beam model.

Next, the selected excitation impulses were applied to the location $0.1L$ and accelerations were obtained from the locations $0.3214L$ (i.e., node 5) and $0.5L$ (i.e., node 7). Sampling frequency was set to 8 kHz and total

Table 4
Damage scenarios for dual damage locations to train neural networks.

Elements	α (%)	Elements	α (%)	Elements	α (%)	Elements	α (%)	Elements	α (%)	Elements	α (%)
1, 2	10, 10	2, 3	20, 30	3, 5	30, 30	4, 8	50, 10	5, 12	7, 12	7, 12	10, 30
1, 3	20, 20	2, 4	30, 50	3, 6	40, 40	4, 9	10, 40	6, 7	8, 9	8, 9	30, 30
1, 4	30, 10	2, 5	40, 20	3, 7	50, 10	4, 10	20, 40	6, 8	8, 10	8, 10	40, 30
1, 5	40, 10	2, 6	50, 20	3, 8	10, 20	4, 11	30, 40	6, 9	8, 11	8, 11	50, 40
1, 6	50, 50	2, 7	10, 20	3, 9	20, 30	4, 12	40, 40	6, 10	8, 12	8, 12	30, 10
1, 7	10, 30	2, 8	20, 30	3, 10	30, 30	5, 6	50, 40	6, 11	9, 10	9, 10	40, 40
1, 8	20, 40	2, 9	30, 20	3, 11	40, 40	5, 7	10, 50	6, 12	9, 11	9, 11	50, 30
1, 9	30, 10	2, 10	40, 10	3, 12	50, 30	5, 8	20, 50	7, 8	9, 12	9, 12	10, 20
1, 10	40, 20	2, 11	50, 10	4, 5	20, 40	5, 9	30, 50	7, 9	10, 11	10, 11	50, 20
1, 11	50, 50	2, 12	20, 50	4, 6	30, 10	5, 10	40, 50	7, 10	10, 12	10, 12	40, 30
1, 12	10, 20	3, 4	10, 20	4, 7	40, 40	5, 11	50, 20	7, 11	11, 12	11, 12	10, 50

8450 discrete acceleration data were numerically obtained from each measure. Total 127 damage scenarios were selected to train neural networks. For each element as a single damage location, the element stiffness loss (as described in Eq. (4)) was simulated between 10 and 50 percent with a step size of 10 percent for each of 12 elements (i.e., 61 single-damage cases). For two elements as dual-damage locations, the element stiffness losses were only simulated for a limited combination of 10 percent for both elements (i.e., 66 dual-damage cases). To clarify the details of the damage scenarios, 66 dual-damage cases were listed in Table 4.

Four sets of ABNN ($ABNN_1, \ldots, ABNN_4$) were generated from the four excitation patterns. Each ABNN (e.g., $ABNN_1$) consisted of three layers (as illustrated in Fig. 2): (1) an input layer with 50 units of acceleration features, (2) a hidden layer with 50 units, and (3) an output layer with 12 units of element stiffness indices which were allocated to 12 elements of the beam model. In the input layer, the 50 units of acceleration features are corresponding to the first 50 values of cross-covariance ratios between intact and damaged state. Note that the two acceleration signals and their cross-covariance signal obtained from the baseline model are similar as shown in Fig. 7.

4.2.2. Alarming damage occurrence and location in test structure

As described previously, three damage cases were tested for the test structure. Acceleration signals measured from sensors 3 ($x/L = 0.32$) and 4 ($x/L = 0.5$), as shown in Fig. 16, before and after damage were utilized for the damage alarming exercises. For each damage case, cross-covariance ratios were computed from accelerations measured before and after damage. Initial 50 cross-covariance ratios were input into the four sets of trained neural networks, $ABNN_1, \ldots, ABNN_4$. From each ABNN, stiffness indices were estimated for the 12 elements of the test structure. The stiffness indices were predicted for the three damage cases. For example, Fig. 22 shows the element stiffness indices predicted for damage case 1, in which four sets of stiffness indices were estimated from $ABNN_1, \ldots, ABNN_4$, respectively.

Finally, the location of damage was identified from statistically classifying the results of stiffness indices. On assuming the stiffness indices distributed normally, normalized damage indices were generated in accordance with Eq. (6) (note that the analysis indicates that the damage indices approximately fit into normal distribution by excluding the damaged elements which are the special causes). The confidence level for the localization was set to $z_0 = 1.5$ and this criterion corresponds to a one-tailed test at a confidence level of 93.3 percent.

The damage localization results for all three damage cases are shown in Fig. 23. Elements 6 and 12 correspond to the real cut locations $x/L = 0.464$ and 0.939 in the test beam, respectively. In damage cases 1 and 2, element 6 was predicted, which shows the correct location. In damage cases 3, elements 1 and 6 were predicted, in which the first one was false-alarmed and the second one was correct. Also, by setting the confidence level $z_0 = 1.3$ which gives 90.3 percent confidence level, element 11 could be also predicted, which shows about 8 percent localization error. It is believed that these localization errors were caused by both acceleration-signal noise and modeling error. It is also noted that we had a limited number of training patterns for the dual-damages case, as described previously.
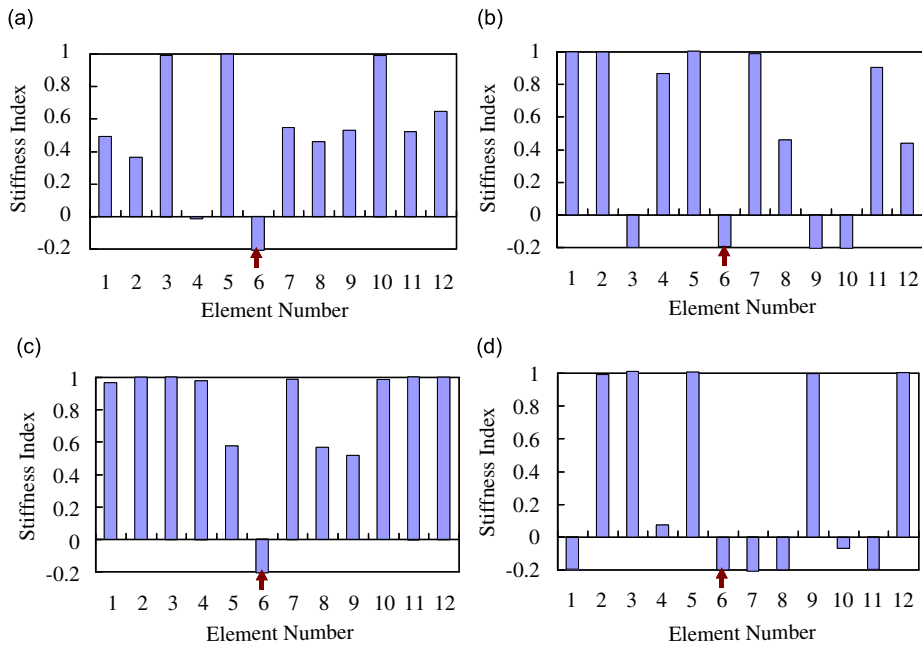
Fig. 22. Stiffness indices predicted by ABNN algorithm: damage case 1: (a) stiffness indices by ABNN1; (b) stiffness indices by ABNN2; (c) stiffness indices by ABNN3; (b) stiffness indices by ABNN4.
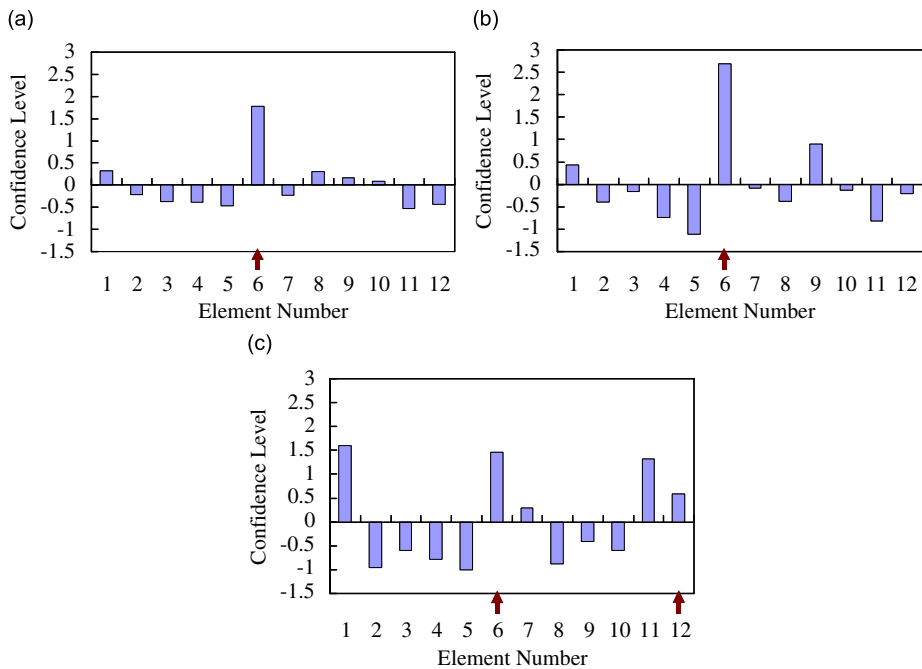
Fig. 23. Damage alarming results by ABNN algorithm: (a) damage case 1; (b) damage case 2; (c) damage case 3.

4.3. Damage estimation in test structure by MBNN algorithm

4.3.1. Training MBNN for test structure

For damage estimation practices, we used the same baseline free–free beam model as schematized in Fig. 20. We first generated MBNN and next utilized them to estimate the location and severity of damage in the test

structure. To train MBNN, we performed numerical modal analyses from which natural frequencies and mode shapes were identified for a series of damage scenarios of the baseline model. We considered both single-damage and dual-damage cases. For single-damage cases, each element was selected as a damage-inflicted location and the stiffness loss (as described in Eq. (4)) was simulated between 10 and 50 percent with a step size of 10 percent. That makes 61 single-damage patterns for the 12 beam elements. For dual-damage cases, all possible pairs out of the 12 elements were considered. Stiffness loss of each pair was simulated for a series of combinations between 10 and 50 percent with a step size of 10 percent. That makes 1650 dual-damage patterns and total 1711 damage patterns were tested to train the MBNN.

The MBNN consisted of three layers (as illustrated in Fig. 2): (1) an input layer with 48 units of modal features, (2) a hidden layer with 48 units, and (3) an output layer with 12 units of element stiffness indices, each of which was allocated to each beam element. The 48 units of the input modal features are corresponding to the ratios of the modal strain energies, which were measured for the four modes and 12 elements, between undamaged and damaged states. Fig. 24 shows the four mode shapes of the baseline model and the corresponding modal strain energies calculated from Eq. (8). Note that the numerical mode shapes of the baseline model show good matches with the experimental mode shapes.

4.3.2. Estimating damage location and severity in test structure

Damage inflicted in the test structure was estimated by the trained MBNN. As described previously, three damage cases were considered for the test structure. As listed in Table 3, pre- and post-damage natural frequencies were extracted from acceleration signals measured from the test structure. As shown in Fig. 25, experimental mode shapes of the four modes were measured from the seven sensors (as shown in Fig. 16) on the test structure. Their corresponding modal strain energies of the four modes and the 12 beam elements were acquired by implementing the experimental mode shapes into Eq. (8).

For each damage case, the changes in modal strain energies of the four modes and the 12 elements were input into the pre-trained neural networks, MBNN. Fig. 26 shows the changes in modal strain energies and
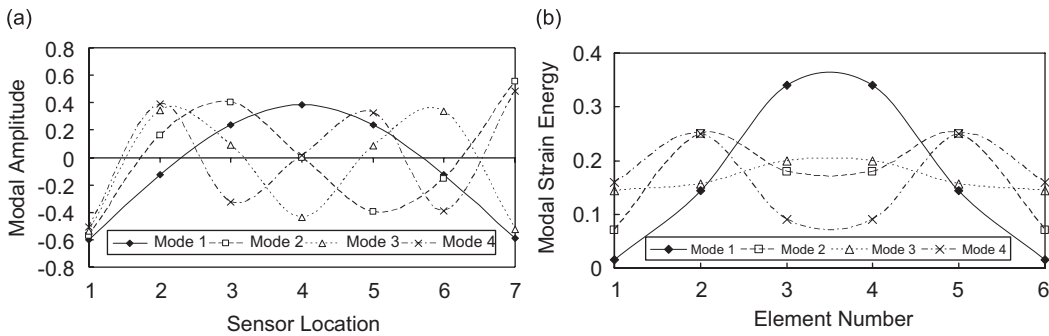

Fig. 24. Numerical mode shapes and modal strain energies of baseline model: (a) mode shapes; (b) modal strain energies.
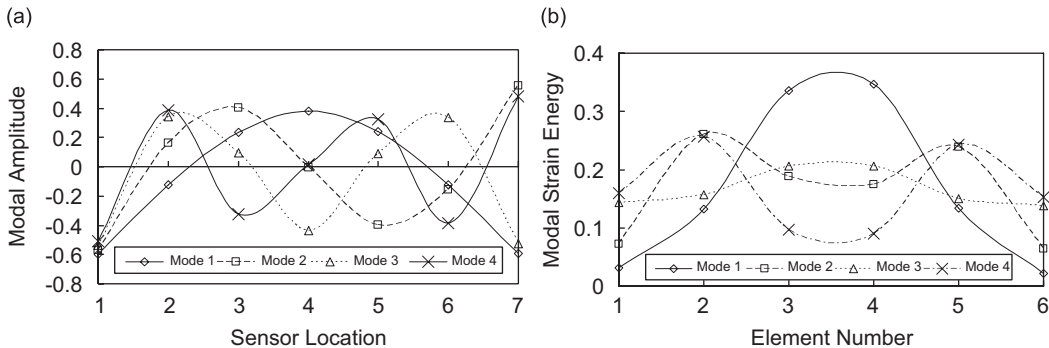

Fig. 25. Experimental mode shapes and modal strain energies of test structure: (a) mode shapes; (b) modal strain energies.
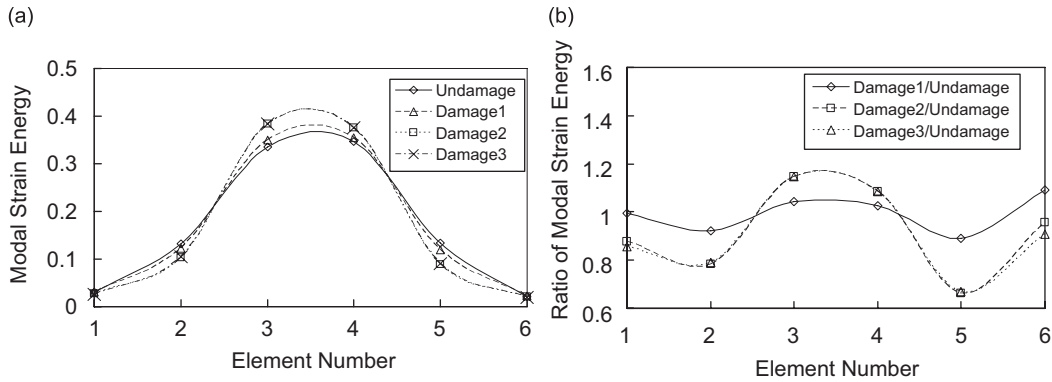
Fig. 26. Experimental modal strain energies of undamaged and three damage cases: (a) changes in modal strain energies; (b) ratios of modal strain energies.
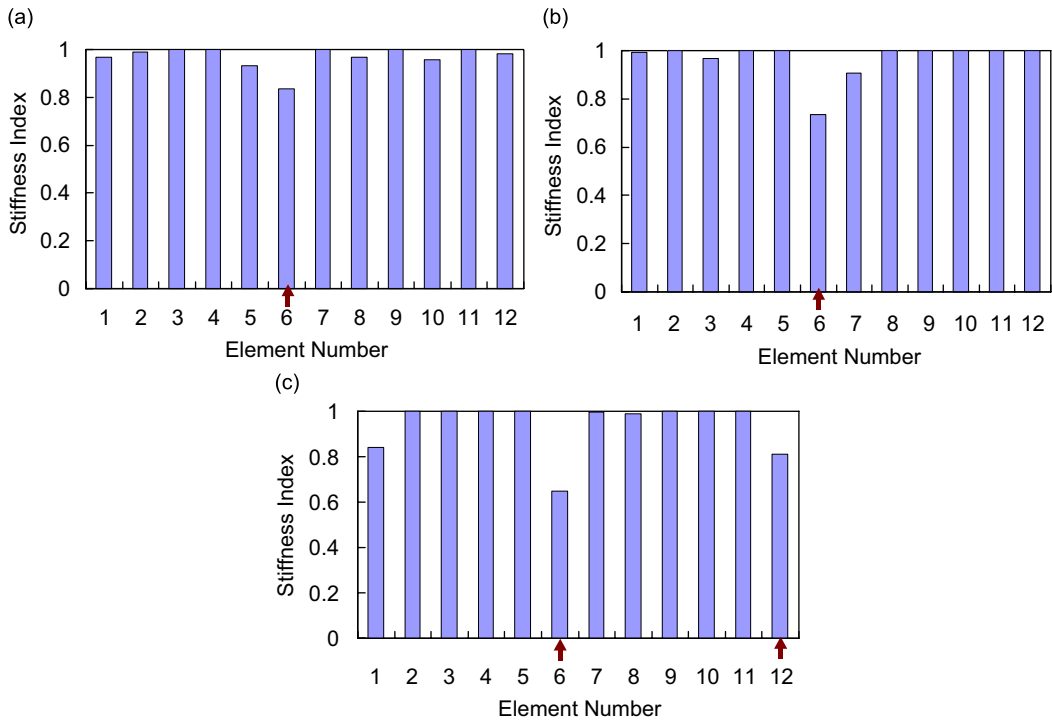
Fig. 27. Stiffness indices predicted by MBNN algorithm: (a) damage case 1; (b) damage case 2; (c) damage case 3.

their ratios between intact and the three damage states for the first mode. Then stiffness indices of the 12 elements were estimated as the output. Fig. 27 shows the element stiffness indices predicted for the three damage cases. By implementing the element stiffness indices into Eqs. (5) and (6), the confidence levels of damage localization were computed for the three damage cases as shown in Fig. 28. By setting $z_0 = 3.0$ which gives 99.8 percent confidence level, damage locations were predicted as listed in Table 5. In damage cases 1 and 2, element 6 was correctly predicted, which is identical to the real cut location, $x/L = 0.464$. In damage case 3, elements 1, 6 and 12 were predicted, in which the first one is false-alarmed and two others are the correct cuts, $x/L = 0.464$ and 0.939.

At the predicted locations, the severity of damage could be identified from the estimated stiffness indices shown in Fig. 27. The identified severities of damage at the predicted locations were summarized in Table 4.
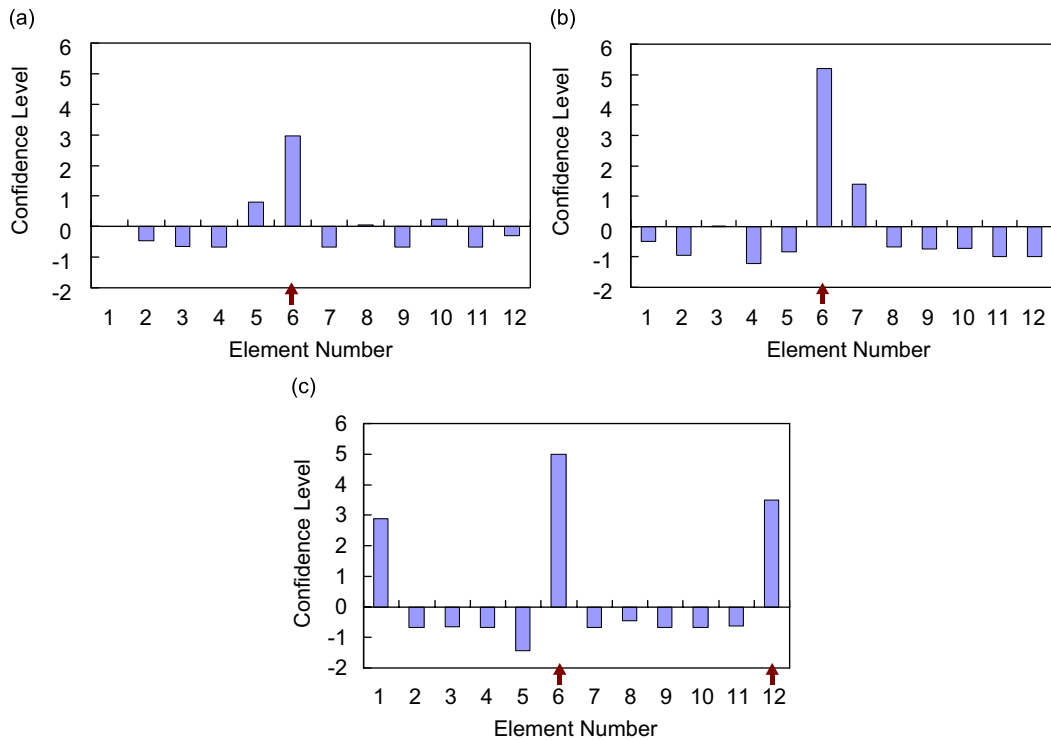
Fig. 28. Damage localization results by MBNN algorithm: (a) damage case 1; (b) damage case 2; (c) damage case 3.

Table 5
Damage prediction results by MBNN algorithm.

Damage case	Inflicted damage 2 mm thick cut(s)		Inflicted damage equivalent stiffness-loss		Predicted element and stiffness-loss	
	Location (x/L)	Size (a/t)	Element	Severity	Element	Severity
1	0.464	0.25	6	0.14–0.21	6	0.16
2	0.464	0.50	6	0.26–0.38	6	0.27
3	0.464	0.25	6	0.26–0.38	6	0.35
	0.939	0.50	12	0.14–0.21	12	0.19

Note that the damage localization was performed in element scale by which we can define an element as a damage location. However, the cut-induced stiffness reduction of the test structure is not identical to the element stiffness reduction of the model since cut-width is not the same as element-length. As described previously, damage was introduced to the test beam up to two levels ($a/t = 0.25$ and $a/t = 0.5$) by inflicting 2 mm thick cuts. By considering the impact of a thin cut on the stiffness-loss of its vicinity and furthermore by taking into account of 20–30 percent loss in element-depth, the equivalent stiffness-loss of a 4.6 cm-wide element was approximated as listed in Table 4. From the results, it is observed that the predicted severities of damage show relatively good agreement with the inflicted severities of damage. However, these results were based on the assumption of the equivalent stiffness-losses corresponding to the thin cuts.

5. Summary and conclusions

In this study, sequential damage detection approaches using time-modal features and ANN for beam-type structures were proposed. First, theoretical backgrounds of the sequential damage detection methods were

described. The sequential method consists of two phases: time-domain feature-based damage alarming and modal feature-based damage estimation. In the first phase, an acceleration-based neural networks (ABNN) algorithm was designed to monitor in real-time the occurrence and location of damage by using cross-covariance functions of two acceleration signals measured from two different sensors. In the second phase, a modal feature-based neural networks (MBNN) algorithm was designed to estimate in detail the location and severity of damage by using mode shapes and modal strain energies, which could be extracted from off-line modal analysis tasks.

The feasibility of the proposed methodology was verified from numerical tests on simply supported beams. Especially for the ABNN algorithm, the effect of model uncertainty due to the difference in impulse excitation properties was evaluated for several loading patterns. Also, the proposed approaches were experimentally evaluated from laboratory tests on a free–free, aluminum beam for which its actual loading histories were unknown. In the first phase, by implementing the ABNN algorithm, the damage occurrence and its locations could be correctly alarmed in the test structure by using cross-covariance signals out of two acceleration signals measured from two different sensors. In the second phase, by implementing the MBNN algorithm, the damage locations and their severities could be accurately estimated in the test beam by using the first four mode shapes and their corresponding modal strain energies measured from seven sensors.

In this study, the proposed sequential approach was evaluated from well-controlled test conditions. Future studies should be focused on rigorously evaluating the sequential approaches by considering optimal sensor locations, modeling uncertainties, and ambient test conditions.

Acknowledgments

References

[1] S.V. Barai, P.C. Pandey, Vibration signature analysis using artificial neural networks, *Journal of Computing in Civil Engineering, ASCE* 9 (4) (1995) 259–265.
[2] J.S. Bendat, A.G. Piersol, *Random Data Analysis and Measurement Procedures*, Wiley, Singapore, 1991.
[3] R. Brinker, L. Zhang, P. Andersen, Modal identification of output-only systems using frequency domain decomposition, *Smart Materials and Structures* 10 (2001) 441–445.
[4] F.N. Catbas, A.M. Aktan, Condition and damage assessment: issues and some promising indices, *ASCE Journal of Structural Engineering* 128 (8) (2002) 1026–1036.
[5] Y. Chen, M.Q. Feng, Condition assessment of bridge sub-structure by vibration monitoring, C.B. Yun, B.F. Spencer, (Eds.), *Second International Workshop on Advanced Smart Materials and Smart Structures Technology*, 21–24 July 2005, Gyeongju, Korea, pp. 651–676.
[6] S.W. Doebling, C.R. Farrar, M.B. Prime, A summary review of vibration-based damage identification methods, *Shock and Vibration Digest* 30 (2) (1998) 91–105.
[7] Y. Gao, B.F. Spencer, Damage localization under ambient vibration using changes in flexibility, *Journal of Earthquake Engineering and Earthquake Vibration* 1 (1) (2002) 136–144.
[8] H. Hao, Y. Xia, Vibration-based damage detection of structures by genetic algorithm, *Journal of Computing in Civil Engineering* 16 (3) (2002) 222–229.
[9] J.T. Kim, N. Stubbs, Model uncertainty impact and damage-detection accuracy in plate-girder, *ASCE Journal of Structural Engineering* 121 (10) (1995) 1409–1417.
[10] J.T. Kim, Y.S. Ryu, H.M. Cho, N. Stubbs, Damage identification in beam-type structures: frequency-based method vs mode-shape-based method, *Engineering Structures* 25 (2003) 57–67.
[11] J.T. Kim, J.H. Park, H.S. Do, Y.H. Lee, Real-time damage monitoring using output-only acceleration data, *Asia-Pacific Workshop on Structural Health Monitoring*, 2–4 December 2006, Yokohama, Japan.
[12] J.T. Kim, J.H. Park, H.S. Yoon, J.H. Yi, Vibration-based damage detection in beams using genetic algorithms, *Smart Structures and Systems* 3 (3) (2007) 263–280.
[13] J.M. Ko, Z.G. Sun, Y.Q. Ni, Multistage identification scheme for detecting damage in cable-stayed Kap Shui Mun Bridge, *Engineering Structures* 24 (2002) 857–868.
[14] J.J. Lee, J.W. Lee, J.H. Yi, C.B. Yun, J.Y. Jung, Neural networks-based damage detection for bridges considering errors in baseline finite element models, *Journal of Sound and Vibration* 280 (3) (2005) 555–578.

[15] J.J. Lee, C.B. Yun, Two-step approaches for effective bridge health monitoring, *Structural Engineering and Mechanics* 23 (1) (2006) 75–95.

[16] N.M.M. Maia, J.M.M. Silva, E.A.M. Almas, Damage detection in structures: from mode shape to frequency response function methods, *Mechanical System and Signal Processing* 17 (3) (2003) 489–498.

[17] MIDAS Information Technology, MIDAS/Civil 2006 Trial Version ⟨http://eng.midasuser.com/⟩.

[18] National Instruments, Data Sheet of NI 4472 Series ⟨http://www.ni.com/pdf/products/us/3sv414-416.pdf⟩.

[19] National Instruments, NI PXI-8186/8187 User Manual ⟨http://www.ni.com/pdf/manuals/370747c.pdf⟩.

[20] National Instruments, Labview 7.1 User Manual ⟨http://www.ni.com/labview/⟩.

[21] Y.Q. Ni, B.S. Wang, J.M. Ko, Constructing input vectors to neural networks for structural damage identification, *Smart Materials and Structures* 11 (2002) 825–833.

[22] W.L. Qu, W. Chen, Y.Q. Xiao, A two-step approach for joint damage diagnosis of framed structures using artificial neural networks, *Structural Engineering and Mechanics* 16 (5) (2003) 581–595.

[23] Z.P. Szewczyk, P. Hajela, Damage detection in structures based on feature-sensitive neural networks, *ASCE Journal of Computing in Civil Engineering* 8 (2) (1994) 163–178.

[24] X. Wu, J. Ghaboussi, J.H. Garret Jr., Use of neural networks in detection of structural damage, *Computers and Structures* 42 (4) (2001) 649–659.

[25] J.N. Yang, S. Pan, H. Huang, An adaptive extended Kalman filter for structural damage identifications II: unknown inputs, *Structural Control and Health Monitoring* 14 (3) (2007) 497–521.

[26] J.H. Yi, C.B. Yun, Comparative study on modal identification methods using output-only information, *Structural Engineering and Mechanics* 17 (3–4) (2004) 445–466.

[27] C.B. Yun, E.Y. Bhang, Joint damage assessment of framed structures using neural networks technique, *Engineering Structures* 23 (5) (2001) 425–435.