Generative Adversarial Network for Damage Identification in Civil Structures

Zahra Rastin, Gholamreza Ghodrati Amiri, and Ehsan Darvishan

1Natural Disasters Prevention Research Center, School of Civil Engineering, Iran University of Science & Technology, Tehran, Iran
2Department of Civil Engineering, Raudheen Branch, Islamic Azad University, Raudheen, Iran

Correspondence should be addressed to Gholamreza Ghodrati Amiri; ghodrati@iust.ac.ir

Received 19 May 2021; Revised 15 August 2021; Accepted 23 August 2021; Published 6 September 2021

Academic Editor: Carlo Rainieri

Copyright © 2021 Zahra Rastin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, many efforts have been made to develop efficient deep-learning-based structural health monitoring (SHM) methods. Most of the proposed methods employ supervised algorithms that require data from different damaged states of a structure in order to monitor its health conditions. As such data are not usually available for real civil structures, using supervised algorithms for the health monitoring of these structures might be impracticable. This paper presents a novel two-stage technique based on generative adversarial networks (GANs) for unsupervised SHM and damage identification. In the first stage, a deep convolutional GAN (DCGAN) is used to detect and quantify structural damages; the detected damages are then localized in the second stage using a conditional GAN (CGAN). Raw acceleration signals from a monitored structure are used for this purpose, and the networks are trained by only the intact state data of the structure. The proposed method is validated through applications on the numerical model of a bridge health monitoring (BHM) benchmark structure, an experimental steel structure located at Qatar University, and the full-scale Tianjin Yonghe Bridge.

1. Introduction

Civil structures need to have their health state monitored regularly. This is necessary in order to detect damages in the early stage and ensure the safety of these structures by repairing the detected damages on time. SHM provides a practical means of automatic assessment of structural health conditions and thus has attracted so much attention in recent years. The main objectives of the studies conducted in this field are damage detection, localization, and quantification in civil, aerospace, and mechanical structures.

SHM is conducted by processing the data acquired from a network of sensors installed on the structure [1]. With great advances in computational power and sensing technologies, leading to the age of big data, new opportunities have been provided for SHM. Machine learning is a promising technology of artificial intelligence that is so popular for big data processing and has recently emerged in the field of SHM [2, 3]. Machine learning tools such as feedforward artificial neural network (ANN) [4–6], support vector machine (SVM) [7–9], and genetic algorithm (GA) [10–12] have been widely used in SHM and damage detection. Among machine learning techniques, deep-learning-based algorithms are gaining more and more popularity due to their ability to extract damage-sensitive features automatically.

Vision-based deep learning approaches have been utilized in health monitoring of structures for purposes such as crack detection [13–16], fatigue detection [17], concrete spalling detection [18], and corrosion assessment [19, 20]. Although these techniques have been proven to be effective in detecting visible damages, they cannot be used to detect damages hidden in internal parts of structures. Therefore, most of the SHM studies are recently focused on vibration-based methods. These methods use the vibration response of a structure to perform damage detection tasks. The idea behind vibration-based methods is that damage-induced changes in a structure alter its vibration response [21].

Vibration-based SHM and damage detection through deep learning can be performed using supervised or
unsupervised learning algorithms. A supervised learning algorithm receives a training data set together with the corresponding answers (labels), and its parameters are then updated accordingly [22]. Among supervised deep-learning-based techniques, convolutional neural networks (CNNs) have recently gained popularity in many engineering applications including SHM because of their remarkable capabilities in feature extraction tasks using raw signals as input. Abdeljaber et al. [23] employed one-dimensional (1D) CNNs to detect and localize damages in a steel frame. The damages were simulated by loosening some bolts at some specific joints. A 1D CNN was considered for each joint that was trained using raw acceleration signals from the healthy state and different damaged states of the structure measured at that joint, to decide whether the joint is healthy or not. The proposed method needed a large amount of data from various damaged states of the structure, so Abdeljaber et al. [24] modified it in another paper, suggesting a new approach for global SHM of the IASC-ASCE benchmark structure, that needed only raw acceleration signals collected from the healthy state and the fully damaged state of the structure to train 1D CNNs. Yu et al. [25] proposed a CNN-based method to quantify and localize damages in a laboratory benchmark structure. Nineteen scenarios, including healthy, single damaged, and multiple damaged ones, were utilized to train and validate the proposed network. Cofre-Martel et al. [26] proposed CNNs for structural damage localization and quantification utilizing images from transmissibility functions. The proposed method was validated on a mass-spring system and a structural beam where training samples were collected from calibrated finite element models. In a study by Guo et al. [27], a CNN-based model was proposed to extract damage features from mode shape data. Different scenarios such as multiple damage, noise interference, and measurement data missing were considered in designing and training the network. Won et al. [28] used the natural excitation technique to normalize structural responses. Normalized acceleration signals were then fed to a CNN to localize damaged structural elements.

The supervised learning algorithms need to be provided with labeled data from damaged states of a structure as training samples, in order to perform structural damage detection. As these data are not usually accessible for large civil infrastructures, these algorithms may be impractical for real-world applications. To solve this issue, unsupervised deep-learning-based techniques have lately been developed for SHM and damage detection purposes. In a study by Rafiee and Adeli [29], raw acceleration signals from the intact state and an unknown state of a 1:20 scaled high-rise building were first preprocessed using synchrosqueezed wavelet transform and fast Fourier transform. The preprocessed signals were then fed to a deep restricted Boltzmann machine for feature extraction. These features were employed to quantify and localize damages in the unknown state. Entezami et al. [30] presented new statistical pattern recognition methods for damage detection and localization. First, a residual-based feature extraction approach was used to extract the residuals of an autoregressive model as damage-sensitive features. Then, a new distance method called Kullback-Leibler divergence was employed to measure the discrepancy between the features extracted from healthy and damaged states. Silva et al. [31] proposed a deep nonlinear principal component analysis to eliminate the influence of changing environmental and operational conditions from monitored data and improve the performance of the damage identification process. The efficiency of the proposed technique was proven by applying it on a progressively damaged prestressed concrete bridge and a suspension bridge. Sarmadi and Karamodin [32] developed a new unsupervised anomaly detection technique based on adaptive Mahalanobis-squared distance and one-class k-nearest neighbor algorithm for SHM and damage detection under changing environmental conditions. Wang and Cha [33] trained a deep autoencoder using acceleration data from a healthy structure and used the trained network to extract damage-sensitive features from acceleration signals. A one-class SVM was then employed to recognize damage in the monitored structure. Silva et al. [34] proposed the use of stacked autoencoders, employing preprocessed vibration data, for feature extraction and unsupervised damage detection. The performance of the proposed approach was evaluated using data sets from the Z-24 Bridge.

To use the advantages of CNNs in an unsupervised manner, this paper proposes to use them in a new two-stage GAN-based technique for structural damage identification. GANs are another type of deep-learning-based unsupervised model that have been broadly used for image processing purposes [35–37]. They have recently been used in the field of SHM to generate new data [38–40] or identify anomalies in data for damage detection [41]. However, to the best of the authors’ knowledge, the technique presented in this paper is the first application of GANs for unsupervised structural damage identification.

In the first stage of the proposed method, a DCGAN is used to detect and quantify structural damages; the detected damages are then localized in the second stage using a CGAN. Unlike the other unsupervised SHM techniques presented in the literature, the proposed method aims to detect, quantify, and locate structural damages using the raw vibration response of a monitored structure. In each stage, only the intact state data is required to train the network, so the method can be applied for damage detection in real-life civil structures. The effectiveness of the proposed approach is confirmed through applications on the numerical model of a BHМ benchmark structure, an experimental steel structure located at Qatar University, and the full-scale Tianjin Yonghe Bridge.

The rest of the paper is organized as follows: GANs are briefly described in Section 2; Section 3 explains the proposed approach for structural health monitoring; applications on the three abovementioned structures together with obtained results are presented in Section 4; and at the end, conclusions are drawn in Section 5.

2. Generative Adversarial Networks

2.1. Overview. A GAN is a type of deep-learning-based generative model designed by Goodfellow et al. [42] to generate new samples of a training data set. It consists of two
neural networks, a generator \((G)\) and a discriminator \((D)\), that compete with each other and are trained at the same time in an unsupervised manner (Figure 1). The generator tries to produce samples from a latent space that are similar to the training data while the discriminator aims to distinguish between the original samples and the generated ones. During this process, samples from the training data set are presented to the discriminator; this network learns the features of the training data well and becomes more qualified to discriminate between real and fake (generated) data. Meanwhile, the generator learns to map from the latent space to the distribution of the training data; aiming to fool the discriminator by producing samples that are as close to real samples as possible. This problem is posed as the following two-player min-max game between the generator and the discriminator [42]:

\[
\min_G \max_D E_{x \sim p(x)} \left[ \log(D(x)) \right] + E_{z \sim p(z)} \left[ \log(1 - D(G(z))) \right],
\]

where \(X\) is the real data with the distribution \(p(X)\) and \(Z\) is a latent variable with the distribution \(p(Z)\). GANs are typically used for image data, and CNNs are usually employed as generator and discriminator models as a result of their remarkable ability in feature extraction and computer vision applications in recent years. Such a network is called a deep convolutional GAN (DCGAN). DCGANs are used in this study with 1D acceleration data to identify and quantify damage in civil structures, as will be discussed in Section 3.1.

2.2. Conditional GANs. Although GANs are capable of generating random new samples of a training data set, the type of samples generated cannot be controlled. These samples can be of any class. CGANs that incorporate labels in the training process were proposed by Mirza and Osindero [43] to address this problem. The labels are used in both generator and discriminator networks; the generator learns to produce data of different classes; and the discriminator learns to distinguish between real and fake data of these classes. In this paper, CGANs are trained to utilize data from the undamaged state of civil structures and used to localize damage in these structures, in the manner discussed in Section 3.2.

3. Methodology

As mentioned before, DCGANs and CGANs are utilized for damage identification in this paper. The applicability of both networks to damage detection, localization, and quantification tasks has been investigated. It was observed that DCGANs perform well in damage detection and quantification while using CGANs is a good solution to the damage localization problem; therefore, a two-stage method is proposed assigning each task to the appropriate network. CNNs are employed as generator and discriminator parts of the networks. Unlike CNNs that require data from damaged structures as training samples, the networks utilized in this paper are trained by only the intact state data and therefore are applicable for real-life civil structures. The two stages are explained in the following subsections.

3.1. Stage 1. As previously stated, DCGANs are used here to detect and quantify damages in civil structures. To reach this purpose, data from the intact and damaged states of a structure of interest, acquired from the accelerometers installed on it, are first preprocessed. The preprocessed healthy state data are then used to train the DCGAN before its discriminator part can be employed for damage detection and quantification. These steps are further explained in this section, and the whole process is depicted in Figure 2.

3.1.1. Preprocessing. Raw data from all considered accelerometers are concatenated in this step to form an \(n\)-column matrix, \(n\) being the number of sensors. This matrix is then divided into smaller \(128 \times n\) matrices, which are used as input data to the network. Matrices from the intact state data are shuffled and used as training samples.

3.1.2. Training of the DCGAN. The preprocessed healthy state data is used here to train the DCGAN. The number of healthy state matrices used for this purpose is determined through trial and error. The generator part of the DCGANs employed in this study is composed of three 1D convolution layers with the same padding. A Leaky Rectified Linear Unit (ReLU) activation layer and a 1D upsampling layer with a window size of 2 are placed after the first two convolution layers. The third convolution layer uses the hyperbolic tangent (Tanh) activation function. The discriminator has three 1D convolution layers with stride 2 and the same padding. A leaky ReLU activation layer is used after each convolution layer. The output of these layers is then flattened and inputted to a dense layer with one neuron and a Sigmoid activation function (Figure 3). In both the generator and the discriminator, the number of filters and the size of them vary for different types of structures investigated in this study. Binary cross-entropy is chosen as a loss function, which is minimized using Adam optimizer.

3.1.3. Damage Detection and Quantification. The discriminator part of the network trained in the previous step is used here for damage detection and quantification. To reach this goal, a set of \(m\) small \((128 \times n)\) matrices from an unknown state is fed to the discriminator. For each matrix, the output of the discriminator, denoted by \(O_i\) (\(i = 1, 2, \ldots, m\)) here, will be the probability that the matrix is from the type of matrices used in the training phase or, in simple terms, is from the intact state of the structure. The probability that the unknown state is actually the healthy state of the structure (PH) is then calculated by
averaging the output values corresponding to all of the input matrices of the unknown state as follows:

\[
\text{PH} = \frac{\sum_{i=1}^{m} O_i}{m}
\]  

This probability is also calculated using the healthy state data and scaled to 1 by a scale factor equal to 1 divided by the obtained PH for these data. The same factor is used to scale the PH obtained for the unknown state. The scaled PH is denoted by PHS here. The probability of damage in the structure in the unknown state is simply given by
This probability increases as the damage propagates through the structure and thus can be used as an index for damage quantification.

### 3.2. Stage 2. CGANs are used in this study to localize damages detected in the previous stage. Similar to stage 1, damage localization is conducted through three main steps that are discussed in this section. Figure 4 shows the process of damage localization.

#### 3.2.1. Preprocessing. Acceleration data from all considered sensors is first normalized between −1 and 1. Next, data from each sensor that is in the form of a long vector are divided into a number of vectors of length 128. These vectors can be used as inputs to the network. For each sensor, vectors from the healthy state data are shuffled prior to being utilized as training samples.

#### 3.2.2. Training of the CGAN. As mentioned earlier, CGANs use labels in the training phase and learn to generate samples matching specific classes. To localize damage, the number of classes is considered equal to the number of sensors (one class for each sensor). Data from sensor 1 is assigned label 0; data from sensor 2 is assigned label 1; and so on. The details of the generator and the discriminator of the CGANs are exactly the same as those of the DCGANs, as presented in Section 3.1.2 (Figure 3). Binary cross-entropy is chosen as a loss function, and Adam optimizer is used to minimize the loss. The preprocessed healthy state data from each sensor along with the corresponding labels are used to train the CGANs. The number of healthy state vectors from each sensor needed in the training phase for a reliable network is obtained by trial and error. After training, the network learns the features of the healthy state data of the sensors well, and for each accelerometer, the discriminator part can be employed to check how different the data of a damaged state are from the data of the intact state of the structure.

#### 3.2.3. Damage Localization. For each sensor, a set of 128 × 1 vectors from a damaged state together with the corresponding label are input to the discriminator. For each vector, the output will be the probability that the vector is from a healthy state. The probability that the data from the considered sensor is of the type of data from the healthy state is calculated by averaging all output values. The data from the sensors that are closer to the damage location will have a lower probability of being healthy, so the appropriate location of damage can be identified accordingly.

### 4. Case Studies and Results

The performance of the proposed method is validated at this stage. To cover all types of data sets, validation is carried out using data from a numerical model, an experimental structure, and a full-scale bridge. All calculations are performed in Python 3.6.9. For each structure, the steps of the proposed method together with the obtained results are presented in this section.

#### 4.1. A Bridge Health Monitoring Benchmark Model. The numerical model of a bridge developed by the University of Central Florida is considered here to validate the proposed methodology. A scheme of this structure is depicted in Figure 5. The bridge consists of two girders of length 5.49 m, seven transverse beams of length 1.83 m, and six W12 × 26 columns of height 1.07 m, which are fixed to the ground. All of the beams (in both longitudinal and transverse directions) have the same cross-section of S3 × 5.7. The numerical model of this structure was developed for researchers to test the reliability of their SHM techniques [44]. In this study, six accelerometers were considered to obtain the response of the bridge under dynamic loading using this numerical model. The acceleration response of this structure was recorded for the healthy state and under four damage patterns. The damage patterns are presented in Table 1 and the sensor placement together with the damaged node is shown in Figure 6.

#### 4.1.1. Stage 1. To obtain the data needed for the damage detection task, the acceleration response of the structure under a dynamic load of 10 kN is recorded with a sampling frequency of 200 Hz. At each state, raw acceleration data from the sensors are connected to create a 6-column matrix, which is then divided into 128 × 6 matrices that can be used as input data. The healthy state data are shuffled prior to being utilized as training samples.

The details of the DCGAN used for this structure are as described in Section 3.1.2. The number of filters in the three convolution layers of the generator are 64, 32, and 6, respectively; and the convolution layers of the discriminator have 6, 32, and 64 filters, respectively. The size of the filter in these layers is 4. Five hundred 128 × 6 matrices from the intact state data are used to train the network.

The discriminator part of the trained network is employed to detect and quantify damages in this structure. Five hundred intact state matrices that are not used to train the network and 500 matrices from each damaged state are input to the discriminator separately. For each state, the probability of being healthy is gained by averaging the output values of the network.

The next step is to scale the probability calculated for the healthy state data to 1. The same factor needed to do so is used to scale the probabilities obtained for the other states as well. The probability of damage in each state is then calculated by subtracting the scaled value from 1. The final results are presented in Table 2. According to this table, the probability of damage in pattern 1, in which the major moment of a beam-to-girder connection is released, is obtained at 0.188. Removing the connection plates at this node in pattern 2 increases this probability to 0.368. Similarly, a reduction of 10% on the spring constant in pattern 3 leads to a probability of damage equal to 0.285, and reducing the spring constant by 20% in pattern 4 increases the probability to 0.358. The results show that the
The proposed method can effectively detect and quantify damages in this numerical model, as the obtained results are consistent with propagation of the damage in the model, and a reasonable probability of damage is obtained in all four damage patterns.

4.1.2. Stage 2. Raw acceleration data recorded by the sensors in stage 1 is first normalized between −1 and 1. Data from each sensor is then divided into smaller vectors of length 128 that are used as input samples. The vectors from the intact state data are utilized to train the CGAN after being shuffled.

The number of healthy state vectors from each sensor needed in the training phase for a reliable network is gained equal to 500 by trial and error. Six classes are considered for these vectors (one class for each sensor). The details of the generator and the discriminator are presented in Section 3.1.2. The convolution layers of the generator have 32, 16, and 1 filters, respectively. The discriminator has three...
convolution layers with 1, 16, and 32 filters, respectively. The filter size in these layers is equal to 8.

For each damaged state, a number of 100 vectors from each sensor along with the related label are fed to the discriminator part of the trained CGAN. Here, the sensors are put into three groups, as shown in Figure 5: group 1 for sensors 1 and 2, group 2 for sensors 3 and 4, and group 3 for sensors 5 and 6. The probability of being healthy for the data gathered by the sensors of each group is obtained by averaging the output values of the discriminator related to the vectors from those sensors. For each damaged state, this probability for the data from each group is presented in Table 3. It can be seen from the table that in patterns (1) and (2), data from the sensors of the first and second groups have a lower probability of being healthy, which suggests that these groups of sensors are closer to the location of the damage. As expected, in patterns (3) and (4), data from the second group in which both sensors are close to the location of damage have the lowest probability of being healthy. Therefore, the appropriate location of damage can successfully be estimated according to these probabilities. It proves the effectiveness of the proposed technique in damage localization.

4.2. Qatar University Grandstand Simulator. This simulator was constructed at Qatar University structures laboratory to provide a common test bed for practitioners to test their vibration-based structural damage detection approaches. The data obtained from the main steel frame of this structure, depicted in Figure 7, was utilized in this study. This frame is composed of eight girders of length 4.6 m, five filter beams in the cantilever part that are about 1 m long, and twenty other filter beams of length 77 cm that are supported by four columns. Thirty accelerometers were installed on the girders at beam-to-girder connections, and damage was simulated by loosening the bolts at these joints. Data from the healthy state and thirty damaged states of this structure were recorded by these sensors [23]. Here, damage pattern 1 corresponds to loosening the bolts at joint 1; pattern 2 corresponds to loosening the bolts at joint 2; and so on. The node numbers are shown in Figure 7.

Table 2: Probability of damage for different damage patterns of the BHM benchmark structure.

| Damage pattern | h   | 1    | 2    | 3    | 4    |
|----------------|-----|------|------|------|------|
| Probability of damage | 0   | 0.188| 0.368| 0.285| 0.358|

Table 3: Probability of being healthy in each group for different damaged states of the BHM benchmark structure.

| Group | Damage pattern | 1    | 2    | 3    | 4    |
|-------|----------------|------|------|------|------|
| 1     | 0.281          | 0.352| 0.334| 0.196|

4.2.1. Stage 1. Acceleration signals obtained from the intact structure and under ten damage patterns, collected under a shaker excitation at a sampling frequency of 1204 Hz, are utilized for damage detection. Here, the sensors installed on joints number 1 to 5 are considered for this purpose. For each state, data from these sensors are connected to create a five-column matrix, which is then divided into $128 \times 5$ matrices.

Preprocessed intact state matrices are used to train the DCGAN. The convolution layers of the generator have 64, 32, and 5 filters, respectively. The discriminator has three convolution layers with 5, 16, and 32 filters. The size of the filter in all these layers is 4. One thousand intact state matrices are used to train the network.

One hundred intact state matrices that are not used to train the network and 100 matrices from each damaged state are fed to the discriminator separately. For each state, the output values of the discriminator are averaged, and the result is scaled by the factor needed to scale the result.
obtained for the healthy state data to 1. The scaled result is subtracted from 1 to gain the probability of damage. The final results are presented in Table 4. It is obvious from the table that the presence of damage and its severity can be identified according to the obtained probabilities. A reasonable probability of damage is assigned to each pattern that proves the efficiency of the proposed method for damage detection and quantification in this experimental structure.

4.2.2. Stage 2. Data from the five considered accelerometers are normalized between \(-1 \) and \(1\); data from each sensor are then divided into \(128 \times 1\) vectors. These vectors form the input data set for the network.

The three convolution layers of the generator part of the CGAN have 64, 32, and 1 filters, respectively. The discriminator has three convolution layers with 1, 32, and 50 filters, respectively. The size of the filter in these layers is 8. The rest of the details of the CGAN used for this structure are as described in Section 3.1.2. One thousand labeled healthy state vectors from each sensor are used to train this network.

To localize the damage in a specific damaged state, 100 vectors from the data recorded by each sensor in that state, along with the labels, are input to the discriminator. The sensors are put into two groups (Figure 6): group 1 for sensors 1 and 2 and group 2 for sensors 3, 4, and 5.

The average of the output values for the sensors of each group, which is the probability that the data from the considered group is of healthy type, is reported in Table 5. As can be seen from the table, in each state, data from the closer group of sensors to the location of damage has a lower probability of being of healthy type, so the appropriate location of damage can be estimated according to the results. Therefore, although very slight damage is inflicted on the structure in each case, the proposed method performs well in the damage localization task, which is proof of its reliability and effectiveness.

4.3. Tianjin Yonghe Bridge. The Tianjin Yonghe Bridge is a cable-stayed bridge located in mainland China. It consists of a 260 m long main span, two 25.15 + 99.85 m long side spans, and two 60.5 m tall towers. The width of the structure is 11 m. The bridge was opened to traffic in 1987. After 19 years of operation, serious damages were observed in the structure, so it was repaired and rehabilitated. During this period, an SHM system, including 14 accelerometers on the deck and 1 accelerometer on one of the towers, was installed on the bridge. The placement of the sensors on the structure is shown in Figure 8 [46, 47].

A while after the bridge was reopened to traffic in 2007, some damages were detected during inspections. The side spans were seriously cracked, and the piers were partially damaged, as well. The damaged area and the damaged piers are denoted in Figure 9. Meanwhile, the acceleration response of the bridge from the healthy to the highly damaged state was recorded by the installed SHM system [48]. Considering that the acceleration responses have been recorded under ambient conditions during a long period of time, the effects of environmental factors, such as variations in wind velocity and temperature, are also included in the recorded data.

Here, the data recorded by the seven sensors installed on the downstream side of the deck are employed for damage detection. The data acquired on January 17, 2008, is treated as the intact state data; from then on, the level of damage gradually increases in the structure. Here, the data recorded on March 19, March 30, April 9, May 18, June 16, and July 31 are used as damaged state data to validate the proposed methodology.

4.3.1. Stage 1. Data from the seven sensors recorded on the abovementioned days are first preprocessed. For each sensor, data is recorded for 24 hours at a sampling frequency of 100 Hz, so a total of \(864 \times 10^4\) data points are available. A total of \(360 \times 10^4\) of these data points are utilized to form a
3,600,000 × 7 matrix, which is then divided into 128 × 7 matrices that are shuffled and used as input data. Two thousand intact state matrices are employed to train the DCGAN. The number of filters in the three convolution layers of the generator is equal to 64, 32, and 7, respectively. Also, the discriminator has three convolution layers with 7, 32, and 32 filters, respectively. The length of the filters is 16 in all these layers.

Five hundred intact state matrices that are not used to train the DCGAN and 500 matrices from each damaged state are fed to the discriminator part of the trained DCGAN. For each state, the output values are averaged and scaled. The probability of damage in each state is obtained by subtracting the scaled value from 1. The final results are shown in Table 6. As expected, the results show a gradual increase in the probability of damage from 0.114 on March 19 to 0.656 on July 31, suggesting that damage is propagating through the structure. According to the results, the network can successfully estimate the severity of damage in all cases that proves the applicability of the proposed method for quantification of damage in this real structure.

4.3.2. Stage 2. Data from the considered sensors is first normalized between −1 and 1; data from each sensor is then divided into smaller vectors of length 128. These vectors are used as inputs to the network.

For each sensor, 1000 vectors from the intact state data, along with the corresponding label, are utilized to train the DCGDN. The generator has three convolution layers with
In stage 1. To the best of the authors’ knowledge, the part of the CGAN is used to localize the damages identified (one class for each sensor). After training, the discriminator CGAN is considered equal to the number of accelerometers only the intact state data. The number of classes in this probability. In the second stage, a CGAN is trained using damage in an unknown state can be determined according to this probability. In the second stage, a CGAN is trained using only the intact state data. The number of classes in the CGAN is considered equal to the number of accelerometers (one class for each sensor). After training, the discriminator part of the CGAN is used to localize the damages identified in stage 1. To the best of the authors’ knowledge, the technique presented in this paper is the first application of GANs for unsupervised structural damage identification. Acceleration data sets from a numerical model, an experimental structure, and a full-scale bridge were utilized to evaluate the efficiency of the proposed method. The results indicate that GANs can successfully be employed for damage quantification and localization in all the considered damage scenarios for the three investigated structures. Since the proposed method needs only the intact state data in the training phase, it provides a practical way for damage detection in a structure, without any information from its damaged states, and thus is suitable for application on real-life civil structures.

### Data Availability

The data are not available due to ethical/legal/commercial restrictions.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### References

[1] Y. Sun, Y. Yuan, Q. Wang, L. Wang, E. Li, and L. Qiao, “Research on the signal reconstruction of the phased array structural health monitoring based using the basis pursuit algorithm,” *Computers, Materials & Continua*, vol. 58, no. 2, pp. 409–420, 2019.

[2] J. Wang, Y. Yang, T. Wang, S. R. Simon, and J. Zhang, “Big data service architecture: a survey,” *Journal of Internet Technology*, vol. 21, no. 2, pp. 393–405, 2020.

[3] Z. Jingyu, Z. Siqi, W. Jin, Y. Xiaofeng, and A. Osama, “A storage optimization scheme for blockchain transaction databases,” *Computer Systems Science and Engineering*, vol. 36, no. 3, pp. 521–535, 2021.

[4] V. Lopes, G. Park, H. H. Cudney, and D. J. Inman, “Impedance-based structural health monitoring with artificial neural networks,” *Journal of Intelligent Material Systems and Structures*, vol. 11, no. 3, pp. 206–214, 2000.

[5] C.-T. Ng, “Application of Bayesian-designed artificial neural networks in phase II structural health monitoring benchmark studies,” *Australian Journal of Structural Engineering*, vol. 15, no. 1, pp. 27–36, 2014.

[6] O. Abdeljaber and O. Avci, “Nonparametric structural damage detection algorithm for ambient vibration response: utilizing artificial neural networks and self-organizing maps,” *Journal of Architectural Engineering*, vol. 22, no. 2, Article ID 04016004, 2016.

[7] H. Hagiwara and A. Mita, “Structural health monitoring system using support vector machine,” in *Proceedings of the International Conference on Advances in Building Technology*, pp. 481–488, Elsevier, Hong Kong, China, December 2002.

[8] J. W. Chong, Y. Kim, and K. H. Chon, “Nonlinear multiclass support vector machine–based health monitoring system for buildings employing magnetorheological dampers,” *Journal of...*
of Intelligent Material Systems and Structures, vol. 25, no. 12, pp. 1456–1468, 2014.

[9] M. Kohiyama, K. Oka, and T. Yamashita, "Detection method of unlearned pattern using support vector machine in damage classification based on deep neural network," Structural Control and Health Monitoring, vol. 27, no. 8, p. e2552, 2020.

[10] H. Y. Guo, L. Zhang, L. Zhang, and J. X. Zhou, "Optimal placement of sensors for structural health monitoring using improved genetic algorithms," Smart Materials and Structures, vol. 13, no. 3, pp. 528–534, 2004.

[11] M. Silva, A. Santos, E. Figueiredo, R. Santos, C. Sales, and J. C. W. A. Costa, "A novel unsupervised approach based on a genetic algorithm for structural damage detection in bridges," Engineering Applications of Artificial Intelligence, vol. 52, pp. 168–180, 2016.

[12] S. Tiachacht, A. Bouazzouni, S. Khatir, M. Abdel Wahab, A. Behtani, and R. Capozucca, "Damage assessment in structures using combination of a modified Cornwell indicator and genetic algorithm," Engineering Structures, vol. 177, pp. 421–430, 2018.

[13] F.-C. Chen and M. R. Jahanshahi, "NB-CNN: deep learning-based crack detection using convolutional neural network and Naive Bayes data fusion," IEEE Transactions on Industrial Electronics, vol. 65, no. 5, pp. 4392–4400, 2017.

[14] B. Kim and S. Cho, "Automated vision-based detection of cracks on concrete surfaces using a deep learning technique," Sensors, vol. 18, no. 10, p. 3452, 2018.

[15] B. Kim, N. Yuvaraj, K. R. Sri Preethaa, and R. Arun Pandian, "Surface crack detection using deep learning with shallow CNN architecture for enhanced computation," Neural Computing & Applications, vol. 33, no. 15, pp. 9289–9305, 2021.

[16] Q. Chen, X. Gan, W. Huang, J. Feng, and H. Shim, "Road damage detection and classification using Mask R-CNN with DenseNet backbone," CMC-Computers Materials & Continua, vol. 65, no. 3, pp. 2201–2215, 2020.

[17] Y. Xu, Y. Bao, J. Chen, W. Zuo, and H. Li, "Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images," Structural Health Monitoring, vol. 18, no. 3, pp. 653–674, 2019.

[18] T. Dawood, Z. Zhu, and T. Zayed, "Machine vision-based model for spalling detection and quantification in subway networks," Automation in Construction, vol. 81, pp. 149–160, 2017.

[19] D. J. Atha and M. R. Jahanshahi, "Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection," Structural Health Monitoring, vol. 17, no. 5, pp. 1110–1128, 2018.

[20] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkiani, and O. Büyükköztüirk, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," Computer-Aided Civil and Infrastructure Engineering, vol. 33, no. 9, pp. 731–747, 2018.

[21] J. P. Amezquita-Sanchez and H. Adeli, "Signal processing techniques for vibration-based health monitoring of smart structures," Archives of Computational Methods in Engineering, vol. 23, no. 1, pp. 1–15, 2016.

[22] R.-T. Wu and M. R. Jahanshahi, "Data fusion approaches for structural health monitoring and system identification: past, present, and future," Structural Health Monitoring, vol. 19, no. 2, pp. 552–586, 2018.

[23] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," Journal of Sound and Vibration, vol. 388, pp. 154–170, 2017.

[24] O. Abdeljaber, O. Avci, M. S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman, "1-D CNNs for structural damage detection: verification on a structural health monitoring benchmark data," Neurocomputing, vol. 275, pp. 1308–1317, 2017.

[25] Y. Yu, C. Wang, X. Gu, and J. Li, "A novel deep learning-based method for damage identification of smart building structures," Structural Health Monitoring, vol. 18, no. 1, pp. 143–163, 2019.

[26] S. Cofre-Martel, P. Kobrich, E. L. Droguett, and V. Meruane, "Deep convolutional neural network-based structural damage localization and quantification using transmissibility data," Shock and Vibration, vol. 2019, pp. 1–27, 2019.

[27] T. Guo, L. Wu, C. Wang, and Z. Xu, "Damage detection in a novel deep-learning framework: a robust method for feature extraction," Structural Health Monitoring, vol. 19, no. 2, pp. 424–442, 2020.

[28] J. Won, J.-W. Park, S. Jang, K. Jin, and Y. Kim, "Automated structural damage identification using data normalization and 1-dimensional convolutional neural network," Applied Sciences, vol. 11, no. 6, p. 2610, 2021.

[29] M. H. Rafieih and H. Adeli, "A novel unsupervised deep learning model for global and local health condition assessment of structures," Engineering Structures, vol. 156, pp. 598–607, 2018.

[30] A. Entezami, H. Shariatmadar, and A. Karamodin, "Data-driven damage diagnosis under environmental and operational variability by novel statistical pattern recognition methods," Structural Health Monitoring, vol. 18, no. 5-6, pp. 1416–1443, 2019.

[31] M. Silva, A. Santos, R. Santos, E. Figueiredo, C. Sales, and J. C. W. A. Costa, "Deep principal component analysis: an enhanced approach for structural damage identification," Structural Health Monitoring, vol. 18, no. 5-6, pp. 1444–1463, 2019.

[32] H. Sarmadi and A. Karamodin, "A novel anomaly detection method based on adaptive Mahalanobis-squared distance and one-class kNN rule for structural health monitoring under environmental effects," Mechanical Systems and Signal Processing, vol. 140, Article ID 106495, 2020.

[33] Z. Wang and Y.-J. Cha, "Unsupervised deep learning approach using a deep auto-encoder with an one-class support vector machine to detect structural damage," Structural Health Monitoring, vol. 20, no. 1, pp. 406–425, 2020.

[34] M. F. Silva, A. Santos, R. Santos, E. Figueiredo, and J. C. W. A. Costa, "Damage-sensitive feature extraction with stacked autoencoders for unsupervised damage detection," Structural Control and Health Monitoring, vol. 28, no. 5, p. e2714, 2021.

[35] F. Zhang, H. Zhao, W. Ying, Q. Liu, A. N. J. Raj, and B. Fu, "Human face sketch to RGB image with edge optimization and generative adversarial networks," Intelligent Automation & Soft Computing, vol. 26, no. 4, pp. 1391–1401, 2020.

[36] K. Fu, J. Peng, H. Zhang, X. Wang, and F. Jiang, "Image super-resolution based on generative adversarial networks: a brief review," Computers, Materials & Continua, vol. 64, no. 3, pp. 1977–1997, 2020.

[37] M. Zhao, X. Liu, X. Yao, and K. He, "Better visual image super-resolution with Laplacian pyramid of generative adversarial networks," CMC-Computers Materials & Continua, vol. 64, no. 3, pp. 1601–1614, 2020.
[38] K. Lee and D. H. Shin, “Generative model of acceleration data for deep learning-based damage detection for bridges using generative adversarial network,” *Journal of KIBIM*, vol. 9, no. 1, pp. 42–51, 2019.

[39] X. Lei, L. Sun, and Y. Xia, “Lost data reconstruction for structural health monitoring using deep convolutional generative adversarial networks,” *Structural Health Monitoring*, vol. 10, 2020.

[40] H. Maeda, T. Kashiyama, Y. Sekimoto, T. Seto, and H. Omata, “Generative adversarial network for road damage detection,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 1, pp. 47–60, 2021.

[41] J. Mao, H. Wang, and B. F. Spencer, “Toward data anomaly detection for automated structural health monitoring: exploiting generative adversarial nets and autoencoders,” *Structural Health Monitoring*, vol. 58, 2020.

[42] I. J. Goodfellow, “Generative adversarial networks,” 2014, https://arxiv.org/abs/1406.2661.

[43] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” 2014, https://arxiv.org/abs/1411.1784.

[44] J. L. Burkett, “Benchmark studies for structural health monitoring using analytical and experimental models,” Dissertation, University of Central Florida, Orlando, FL, USA, 2005.

[45] O. Avci, M. S. Kiranyaz, and O. Abdeljaber, “Structural damage detection,” 2019.

[46] H. Li, S. Li, J. Ou, and H. Li, “Modal identification of bridges under varying environmental conditions: temperature and wind effects,” *Structural Control and Health Monitoring*, vol. 17, no. 5, pp. 495–512, 2010.

[47] S. Li, H. Li, Y. Liu, C. Lan, W. Zhou, and J. Ou, “SMC structural health monitoring benchmark problem using monitored data from an actual cable-stayed bridge,” *Structural Control and Health Monitoring*, vol. 21, no. 2, pp. 156–172, 2014.

[48] H. B. Bisheh, G. G. Amiri, M. Nekooei, and E. Darvishan, “Damage detection of a cable-stayed bridge using feature extraction and selection methods,” *Structure and Infrastructure Engineering*, vol. 15, no. 9, pp. 1165–1177, 2019.

[49] M. R. Kaloop and J. W. Hu, “Stayed-cable bridge damage detection and localization based on accelerometer health monitoring measurements,” *Shock and Vibration*, vol. 2015, pp. 1–11, 2015.