Wiley
Structural Control and Health Monitoring
Volume 2025, Article ID 8419570, 27 pages
https://doi.org/10.1155/stc/8419570

WILEY

Research Article

An Enhanced Generative Adversarial Imputation Network With Unsupervised Learning for Random Missing Data Imputation of All Sensors

Xin Xie,[1] **Ying Lei**,[1,2] **Chunyan Xiang**,[1] **Yixian Li**,[3] **and Lijun Liu**[1]

[1]*Department of Civil Engineering, Xiamen University, Xiamen 361005, China*
[2]*Xiamen Key Laboratory of Integrated Application of Intelligent Technology for Architectural Heritage Protection, Xiamen University, Xiamen 361005, China*
[3]*Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China*

Correspondence should be addressed to Lijun Liu; liulj214@xmu.edu.cn

Received 31 March 2025; Revised 16 July 2025; Accepted 19 July 2025

Academic Editor: Lucia Faravelli

Structural health monitoring (SHM) data are crucial for structural state assessment. However, long-term monitoring data are inevitably subject to data missing in actual SHM, which seriously hinders the reliability of the SHM system. So far, many deep learning-based supervised data imputation methods have been proposed, which require complete sensor data for training. Although there are studies on unsupervised data imputation, some complete sensor data are still required. Especially, there is a lack of study on the challenging problem of unsupervised data imputation with incomplete data of all sensors, which may occur in actual SHM. Therefore, an enhanced generative adversarial imputation network with unsupervised learning is proposed in this paper for such a challenging task. First, within the generative adversarial imputation network framework, convolutional neural networks (CNNs) with an encoder–decoder architecture are established to extract significant high-level local features. Furthermore, a self-attention mechanism is embedded into the generative network to globally capture remote dependencies between data. Finally, the skip connections are incorporated to enhance the parameter utilization and imputation performance of the network. The random missing data imputation with incomplete data of the field monitoring acceleration data from the Dowling Hall footbridge is used to validate the proposed method. The results show that good data imputation in both the time and frequency domains can be achieved by the proposed method in the case of random data missing in all sensors.

Keywords: generative adversarial networks; missing data imputation; structural health monitoring; unsupervised learning

1. Introduction

The large amount of sensor data accumulated in structural health monitoring (SHM) systems provides solid data support for in-depth exploration of the fundamental scientific issues of engineering structures and structural state assessment [1, 2]. Therefore, the completeness and effectiveness of structural monitoring data are crucial. However, due to harsh environments in the application of sensors and their improper maintenance during operations, sensor data missing is a common problem in SHM systems. Then, the accuracy, effectiveness, and reliability of SHM and evaluation processes are affected by data missing [3]. For example, it was shown that the effect of 0.38% missing data on power spectral density is similar to that of 5% observed noise, which has a negative impact on structural health diagnosis [4]. Therefore, to ensure accurate and reliable structural status assessment, it is necessary to effectively impute missing sensor data in SHM systems.

So far, various data imputation methodologies have been developed, and they can be generally classified into three categories: statistical imputation techniques, machine

learning (ML) approaches, and deep learning strategies [5]. Statistical imputation techniques usually rely on parameterized statistical models, such as linear interpolation, autoregressive moving average model, or expectation maximization algorithms [6, 7], to impute missing data by fitting model parameters. Chen et al. [8] adopted a method based on copula functions to capture the interdependencies within structural strain monitoring data, thereby successfully imputing missing data. However, the effectiveness of statistical imputation methods is still subject to the limitations in the imputation of random and extensive long-term missing data.

With the rapid advancement of ML technology, ML has been adopted to process and analyze large amounts of sensor data [9]. In recent years, an increasing number of ML-based methods have been proposed to impute missing data. Bao et al. [10] proposed a recovery method for wireless sensor networks in bridge structures based on compressive sensing techniques and further studied the recovery of missing data under fast-moving wireless sensing technology [11]. These methods are based on data sparsity in a specific feature space. Huang et al. [12] used Bayesian compressed sensing technology to enhance the recovery performance of approximately sparse signals.

Alternatively, deep learning strategies can deal with complex data correlations, efficiently process large-scale data, and automatically extract high-level features, which are effective for solving the problem of missing data imputation [9, 13]. In recent years, data imputation methods based on supervised learning have been developed rapidly. Fan et al. [14] proposed a fully convolutional neural network (CNN) based on U-Net to recover randomly lost acceleration data, and Jiang et al. [5] proposed a deep fully CNN with encoder–decoder architecture combining skip connections, which has excellent recovery accuracy and robustness in missing data imputation. However, such supervised learning data imputation methods require complete data from all sensors for training networks.

Generative adversarial imputation networks (GAINs) have been specially designed for missing data imputation. Some unsupervised learning methods based on GAIN have made progress in imputing missing data in other fields [15–18]. In the field of SHM, Jiang et al. [19] studied continuous missing data imputation by generative adversarial networks (GANs)-based unsupervised learning for long-term bridge health monitoring, Hou et al. [20] proposed a data imputation framework based on GAIN and data augmentation techniques, which imputed continuous missing data using the data of a few normal sensors, and Gao et al. [21] proposed a slim GAIN (SGAIN) to recover the missing deflection data of partial sensors in bridge SHM systems. Subsequently, they also [22] presented a Wasserstein GAIN (WGAIN) based on gradient penalty to impute missing acceleration data of some sensors based on partially complete sensor data and the remaining missing sensor data. However, all these unsupervised learning data imputation methods still need some complete sensor data. However, all sensing data may be incomplete, so these methods are not suitable for this challenging situation.

Based on the above literature review, existing deep learning-based imputation methods still require some complete sensor data, which ensures that missing sampling points can still be easily imputed relying on the complete data from other sensors. In particular, if all sensor data are randomly missing at the same time, the problem of random data missing from all sensors undoubtedly further increases the difficulty of data imputation. Currently, there is still a lack of study in this area in existing data imputation methods, but data imputation under the condition that all sensors are incomplete is a challenging problem.

Therefore, an enhanced GAIN is proposed in this paper for unsupervised data imputation with incomplete data of all sensors. Since there are only incomplete data from all sensors to train the imputation network, which increases the difficulty of imputation random missing data, three main improvements are proposed to enhance the performances of existing GAIN: (1) encoder–decoder architectures composed of CNNs are integrated into the generator and discriminator to enhance the learning and expression capabilities of this network; (2) the self-attention mechanism [23] is embedded into the generator, which focuses more attention on important features to enhance the ability of the network to extract global features; (3) the skip connection technique [24] is incorporated into the generator and discriminator to alleviate gradient vanishing and improve the reuse rate of features. Consequently, the local and global features can be effectively extracted by the proposed method, and the imputation problem of random missing sensor data can be better realized. The effectiveness of the proposed enhanced GAIN is validated by the random missing data imputation with incomplete data of the field monitoring acceleration data from the Dowling Hall footbridge.

The remainder of this paper is organized as follows: In Section 2, the conventional GAIN method for data imputation is briefly introduced. In Section 3, the proposed enhanced GAIN is presented, in which in Section 3.1, CNN with encoder–decoder architectures is integrated into the generator and discriminator; in Section 3.2, self-attention mechanism is embedded into the generator; in Section 3.3, the skip connection technique is incorporated into the generator and discriminator. In Section 4, imputation with different missing rates of all sensor data and high missing rates of some sensors are considered to validate the performance of the proposed method using field monitoring acceleration data of Dowling Hall footbridge. In Section 5, some conclusions of this study are summarized.

2. Brief Introduction of the GAINs for Data Imputation

The GAIN is an improvement based on the conventional GAN. GAN is composed of a generator and discriminator, which is a powerful generative model proposed under the framework of deep learning [25]. The training of GAN is essentially a process of mutual game, and the performance of the two modules is gradually improved through adversarial training. Based on the architecture of GAN, Yoon et al. [26]

introduced the GAIN in 2018 to impute missing data. The procedure of GAIN is depicted in Figure 1.

The masking matrix and prompting mechanism are incorporated into the GAIN. The masking matrix contains information about the missing data locations, and the prompting mechanism provides information about the missing data locations for the discriminator; thereby, the true data distribution can be better learned by the generator. The generator can learn the true distribution from the input data and output new data that conforms to this distribution. The confrontation mechanism between the generator and the discriminator promotes the network to learn the optimal distribution. In GAIN, the generator aims to generate data as accurately as possible, while the discriminator aims to accurately distinguish between the true values in the input data and the values generated by the generator. Through continuous iterative training, the generator can ultimately generate data that are close to the true values at the missing locations. The main structure of GAIN similarly consists of two models: a generator and a discriminator, both of which are composed of fully connected layers. The schematic of the fully connected layers is illustrated in Figure 2.

The incomplete data matrix ($\widetilde{\mathbf{X}}$), mask matrix ($\mathbf{M}$ with values of 0 or 1, indicating the positions of missing data, i.e., 1 represents the parts of the data that are present and 0 denotes the missing part), and random noise matrix ($\mathbf{Z}$ follows a uniform distribution of −0.01 to 0.01 [27], and the values can be filled in where the original data is missing) are input into the generator to produce an imputed matrix $\overline{\mathbf{X}}$. Then, the incomplete matrix and imputed matrix are combined through the mask matrix to obtain the final output matrix ($\widehat{\mathbf{X}}$). In the final output matrix ($\widehat{\mathbf{X}}$), the not missing parts are still the true data, and the missing parts are replaced by the data output by the generator.

During adversarial training, the authenticity of the data generated by the generator is judged by the discriminator. So, the output of the generator $\widehat{\mathbf{X}}$ is input into the discriminator. Then, the probability of each position being true or fake can be output by the discriminator. In this process, to ensure that the discriminator forces the generator to learn the expected data distribution, a hint matrix $\mathbf{H}$ is used to provide partial information to the discriminator. The hint matrix is derived from the mask matrix to ensure that the generator learns and generates data samples based on the true distribution of data. In the hint matrix, 1 indicates the observed true data, 0 indicates missing data, and the data information that needs to be judged by the network is represented by 0.5 (indicating that it may be missing data or true data).

Ultimately, the discriminator is trained to maximize the probability of correctly predicting the true mask matrix $\mathbf{M}$, and the generator is trained to minimize the probability that the discriminator correctly predicts $\mathbf{M}$. The objective function for the predictive process of this network model can be expressed as follows:

$$\min_G \max_D E_{\widehat{\mathbf{X}},\mathbf{M},\mathbf{H}}\left[\mathbf{M}^T \log D(\widehat{\mathbf{X}},\mathbf{H}) + (1-\mathbf{M})^T \log(1 - D(\widehat{\mathbf{X}},\mathbf{H}))\right], \tag{1}$$

where G and D denote the operation of generator and discriminator, respectively; E denotes the mathematical expectation; $(*)^T$ represents the transpose operation, which is used to adjust the dimensions of a matrix for subsequent calculations; $D(\widehat{\mathbf{X}},\mathbf{H})$ is the output when generating data $\widehat{\mathbf{X}}$ and hint matrix $\mathbf{H}$ to the discriminator network; $\min_G \max_D$ indicates the minimization and maximization of the loss function of generator and discriminator, respectively.

3. The Proposed Enhanced GAIN With Unsupervised Learning for Random Missing Data Imputation With Incomplete Data of All Sensors

In the existing research, the GAIN focuses on incomplete data of some sensor data, but all sensor data in the practical engineering are missing at random. Therefore, an enhanced GAIN is proposed based on the GAIN framework to impute all sensor data in the case of random missing. Data construction with incomplete data of all sensors can be defined as follows.

Since data missing in real engineering usually occurs randomly, the missing data of each sensor are considered an independent event in this paper. For a given complete dataset $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_i, \ldots, \mathbf{X}_n\}$ with n sensors of the same type, where $\mathbf{X}_i$ represents the time series data of the i-th sensor, and $\mathbf{X}_i = \{x_{1i}, x_{2i}, \ldots, x_{ti}\}^T$, where x_{ti} represents the sampling point of the i-th sensor at the t-th moment and $\mathbf{X}$ is a matrix of size $t \times n$. The mask matrix $\mathbf{M}$ has the same dimension of $\mathbf{X}$, which represents the location information of the random missing data in the dataset, and any element m_{ti} in the mask matrix can be represented as follows:

$$m_{ti} = \begin{cases} 1, & x_{ti} \text{ is not missing,} \\ 0, & x_{ti} \text{ is missing.} \end{cases} \tag{2}$$

The incomplete dataset $\widetilde{\mathbf{X}}$ can be obtained directly in practical engineering, and any value $\widetilde{x}_{ti}$ in the incomplete dataset can be defined as follows:

$$\widetilde{x}_{ti} = \begin{cases} x_{ti}, & m_{ti} = 1, \\ \text{nan}, & m_{ti} = 0, \end{cases} \tag{3}$$

where nan means that data is missing at that location.

Moreover, the incomplete dataset can be generated by modeling the location of the missing points through a mask matrix. The incomplete matrix $\widetilde{\mathbf{X}}$ can be represented as follows:

$$\widetilde{\mathbf{X}} = \mathbf{X} \odot \mathbf{M}, \tag{4}$$

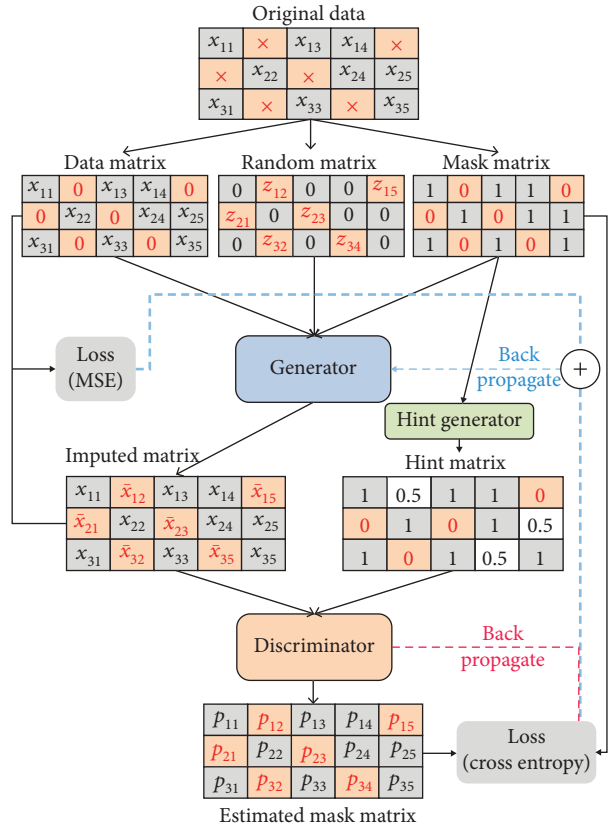


FIGURE 1: A framework for generating adversarial imputation networks [26].

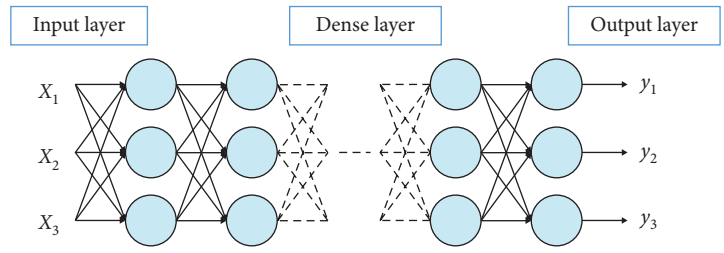


FIGURE 2: Fully connected layers in GAIN.

where ⊙ represents the Hadamard product, i.e., the product of the corresponding elements in the matrix. The schematic diagrams of $\tilde{\mathbf{X}}$ and $\mathbf{M}$ are shown in Figures 3 and 4.

Random noises are only added to the missing positions of the incomplete dataset to form the random noise matrix $\mathbf{Z}$. As mentioned above, $\tilde{\mathbf{X}}$, $\mathbf{M}$, and $\mathbf{Z}$ are input into the generator, and then, the output of the generator can be represented as follows:

$$\overline{\mathbf{X}} = G(\tilde{\mathbf{X}}, \mathbf{M}, (1 - \mathbf{M}) \odot \mathbf{Z}), \tag{5}$$

where $\overline{\mathbf{X}}$ is the output matrix after imputed by the generator. It is worth noting that although the not missing part of the network's input is the true value, the network will output a value for each position of the data, and there is still a certain error between the value and the true value. Therefore, the final complete matrix $\hat{\mathbf{X}}$ obtained by the generator should be composed of the data that is not missing in the original data and the data generated by the generator at the missing position, which can be expressed as follows:

$$\hat{\mathbf{X}} = \mathbf{M} \odot \tilde{\mathbf{X}} + (1 - \mathbf{M}) \odot \overline{\mathbf{X}}. \tag{6}$$

Then, the complete matrix $\hat{\mathbf{X}}$ and the hint matrix $\mathbf{H}$ are input into the discriminator to predict the mask matrix $\hat{\mathbf{M}}$ correctly. This process can be expressed as follows:

$$\hat{\mathbf{M}} = D(\hat{\mathbf{X}}, \mathbf{H}), \tag{7}$$

where $\hat{\mathbf{M}}$ denotes the output of the discriminator. The random noise matrix and the hint matrix are shown in Figures 5 and 6.

It can be seen that compared with only some sensor data missing, the incomplete matrix, mask matrix, random noise matrix, and hint matrix of random missing all sensor data have greater sparsity, which further increases the difficulty of

$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	nan	nan	$X_{1,6}$
nan	nan	nan	$X_{2,4}$	$X_{2,5}$	$X_{2,6}$
nan	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$	nan	$X_{3,6}$
$X_{4,1}$	nan	nan	nan	$X_{4,5}$	$X_{4,6}$
$X_{5,1}$	$X_{5,2}$	$X_{5,3}$	$X_{5,4}$	$X_{5,5}$	nan
$X_{6,1}$	nan	$X_{6,3}$	nan	nan	$X_{6,6}$

Figure 3: Incomplete data matrix.

1	1	1	0	0	1
0	0	0	1	1	1
0	1	1	1	0	1
1	0	0	0	1	1
1	1	1	1	1	0
1	0	1	0	0	1

Figure 4: Mask matrix.

0	0	0	Z	Z	0
Z	Z	Z	0	0	0
Z	0	0	0	Z	0
0	Z	Z	Z	0	0
0	0	0	0	0	Z
0	Z	0	Z	Z	0

Figure 5: Random noise matrix.

imputation random missing data. Based on the framework of GAIN, an enhanced GAIN is proposed for unsupervised learning of random missing data imputation with incomplete data of all sensors. The enhanced GAIN consists of two network modules: the generator and the discriminator, both of which are composed of CNNs with encoder–decoder architecture. Moreover, the self-attention mechanism is embedded into the generator to extract important features, and the skip connection technique is incorporated into both the generator and discriminator to alleviate the problem of gradient vanishing. The details of each part of the proposed network are illustrated in the following sections.

3.1. CNN With Encoder–Decoder Architectures Integrated Into the Generator and Discriminator. It is proposed that CNNs with an encoder–decoder architecture (consisting of convolutional layers, batch normalization layers, pooling layers, and up-sampling layers) are integrated into the generator and discriminator to impute random missing data of all sensors. The encoder (down-sampling phase) is established to extract and compress the high-level features of the input. The decoder (up-sampling phase) is conducted to decompress these compressed high-level features by the encoder and restore them to an output with the same size dimension as the input. Moreover, CNN can extract local optimal features from data in both the time and frequency domains and has a strong ability to learn features from large-scale datasets. Thus, the learning and expression ability of the network can be largely enhanced for random missing data imputation with incomplete data of all sensors. The network architecture of the generator and discriminator is shown in Figures 7 and 8, respectively.

The CNN architecture illustrated in Figures 7 and 8 adopts an encoder–decoder structure with skip connections and a self-attention module at the bottleneck. A total of seven convolutional layers are used throughout the network. In the encoder stage, the convolutional layers are utilized to extract robust high-level local features in the input data to evaluate more complex spatiotemporal information. Each convolutional layer is implemented using a 2D convolution with a kernel size of 3×3, stride of 1, and padding of 1. Bias terms are included in all convolutional layers to enhance learning flexibility. At the same time, a batch normalization layer is added after the convolution layer to normalize feature distributions across mini-batches and improve training stability. Furthermore, to make the network learn more complex nonlinear relationships, an activation function is added after the convolution and the batch normalization layers. Since the rectified linear unit (ReLU) activation function has a good effect in alleviating the gradient vanishing or explosion problem during the training process, the ReLU function is selected for the activation function of the hidden layer in the network. In the last layer of the network, a sigmoid function is chosen to map the output range between 0 and 1. The formulas for the ReLU and sigmoid activation functions can be represented as follows:

$$\text{ReLU: } f(x) = \max(0, x) = \begin{cases} x, & x \geq 0, \\ 0, & x < 0, \end{cases} \tag{8}$$

$$\text{Sigmoid: } f(x) = \frac{1}{1 + \exp^{-x}}. \tag{9}$$

Moreover, in the encoder stage, max pooling layers with a kernel size of 2×2 and a stride of 2 are utilized to reduce the dimension of the data features, thereby reducing the number of parameters and mitigating potential overfitting issues. Correspondingly, in the decoder stage, the up-sampling block that first upsamples the input feature map using nearest-neighbor interpolation with a scale factor of 2 and the convolution layers are used to recover the feature size and the dimension of the channel.

3.2. Self-Attention Mechanism Embedded Into the Generator. The self-attention mechanism has shown good performance in some time series-related tasks [28, 29]. For its computation process, first input data are multiplied by three weight matrices $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$, and three matrices are obtained,

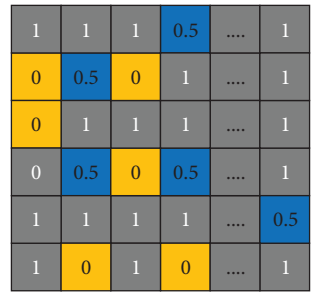


Figure 6: Hint matrix.

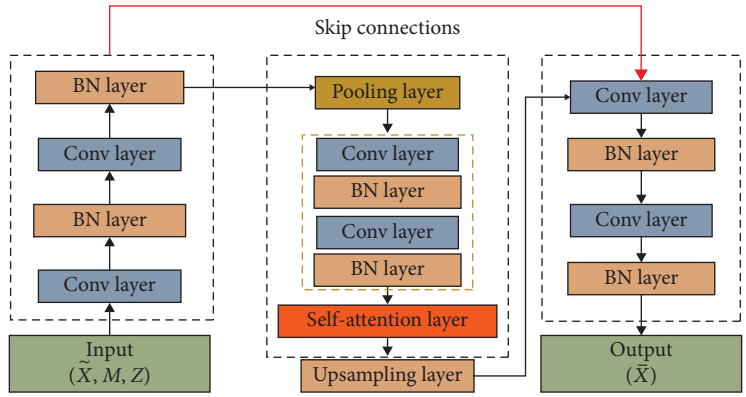


Figure 7: The proposed generator architecture.

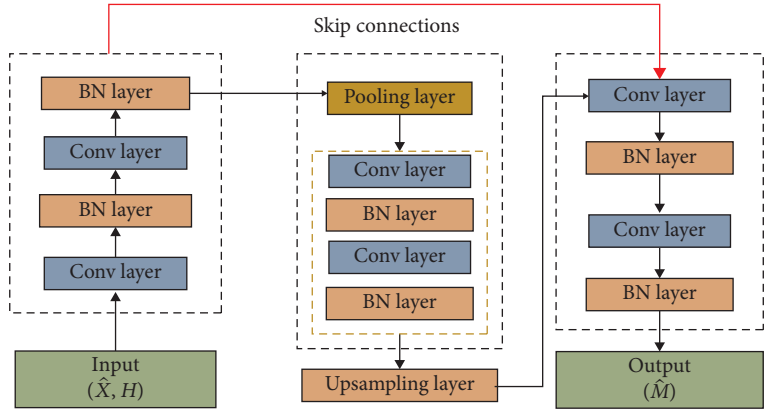


Figure 8: The proposed discriminator architecture.

i.e., retrieval value (Query, **Q**), key value (Key, **K**), and elemental value (Value, **V**). The relationships between elements within the sample data are then calculated using two of these matrices (**Q** and **K**), to get attention weight values at different positions. Finally, a weighted sum is performed on the third matrix (**V**) using these attention weights to produce the output. The computational diagram of the self-attention mechanism is shown in Figure 9.

The self-attention mechanism effectively can capture correlations among multiple time series within the input. During computation, higher weights are assigned to the more influential features of the input data, giving greater attention to critical features and thereby enhancing the performance of networks [30]. Furthermore, the sequential processing constraints inherent in recurrent neural network models are eliminated by the self-attention mechanism, allowing global characteristics to be extracted from time-series data, which demonstrates great potential for feature extraction in such data.

In practical engineering applications, SHM systems accumulate substantial time-series data over extended periods, with each sensor capturing vast amounts of data. However, to extract the features between distant samples, enough layers or a large enough convolutional kernels are needed in CNN, which often leads to excessive network parameters and reduces the efficiency of network training. Unlike convolutional kernels, whose receptive fields are restricted by kernel size, the self-attention mechanism

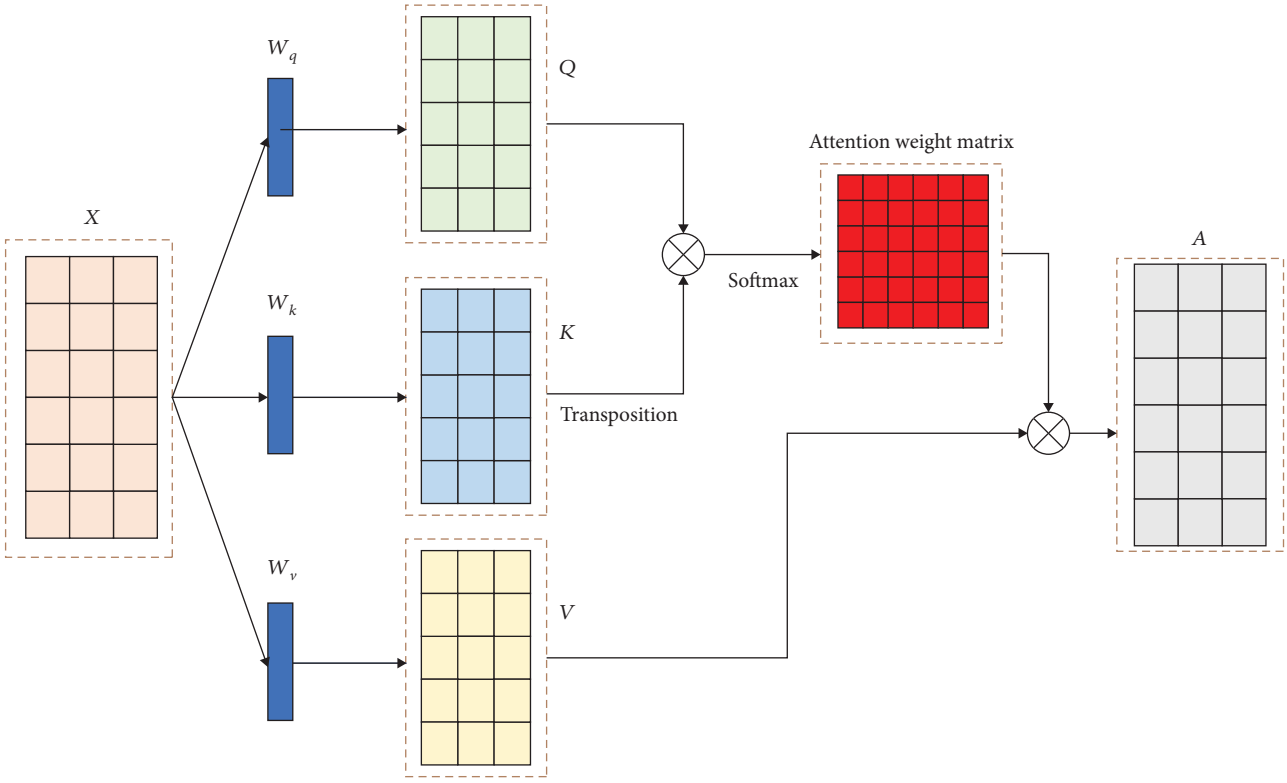


Figure 9: Schematic diagram of self-attention mechanism calculation.

provides the network with a global receptive field [31]. This mechanism enables the efficient and accurate learning of global features and long-range dependencies between sample points, facilitating the extraction of valuable information for the imputation of missing data. Therefore, a self-attention layer is incorporated between the encoder and decoder of the generator to globally capture long-range dependencies between input data, as shown in Figure 7. At this stage, the input feature map has the shape (batch size × channels × height × width). Three parallel 1×1 convolutions generate the Query, Key, and Value feature maps, where the Query and Key are reduced to one-eighth of the original number of channels for efficiency, and the Value retains the full channel dimension. The spatial dimensions are flattened to form sequences, enabling Query–Key interactions to compute attention scores that capture spatial dependencies. These scores weight the Value features to produce attention-enhanced representations, which are reshaped and fused with the input via a residual connection. This makes up for the limitations of CNN, makes the network focus on the more important parts of the global features, enhances the ability of the network to capture the global features, and then improves the overall performance of the network for random missing data imputation with incomplete data of all sensors.

3.3. Skip Connections Incorporated Into the Generator and Discriminator. Skip connections are a widely used technique in deep learning networks, initially introduced in residual networks to mitigate the issues of vanishing and exploding gradients in deep neural network training [32]. The core idea is to establish direct connections between different layers in the network, allowing the features from earlier layers to be passed directly to later layers. This approach prevents the degradation of information as it propagates through multiple layers. The introduction of skip connections effectively preserves shallow feature information while facilitating the network's ability to learn residuals, thereby improving both the training efficiency and performance of the network. Additionally, skip connections promote more efficient gradient flow, ensuring greater stability during training, especially when dealing with complex tasks and large-scale data.

Some problems such as gradient vanishing may occur in network training, which leads to the failure of network training. For GAINs or other encoder–decoder-based models, skip connections can directly transfer feature information between the encoder and decoder, thereby enhancing the network's ability to reconstruct input data and improving the retention and utilization of feature representations [24, 33]. Therefore, the skip connection technique is incorporated between the encoder and decoder stages, as shown in Figures 7 and 8. Through these connections, the low-level features in the encoder stage can be transferred to the decoder stage. The down-sampling and up-sampling processes can be bypassed to transfer the underlying details lost during the convolution process to deeper layers. At the same time, the skip connection also allows the gradient to be directly back-propagated to the shallow layer, further enhancing the

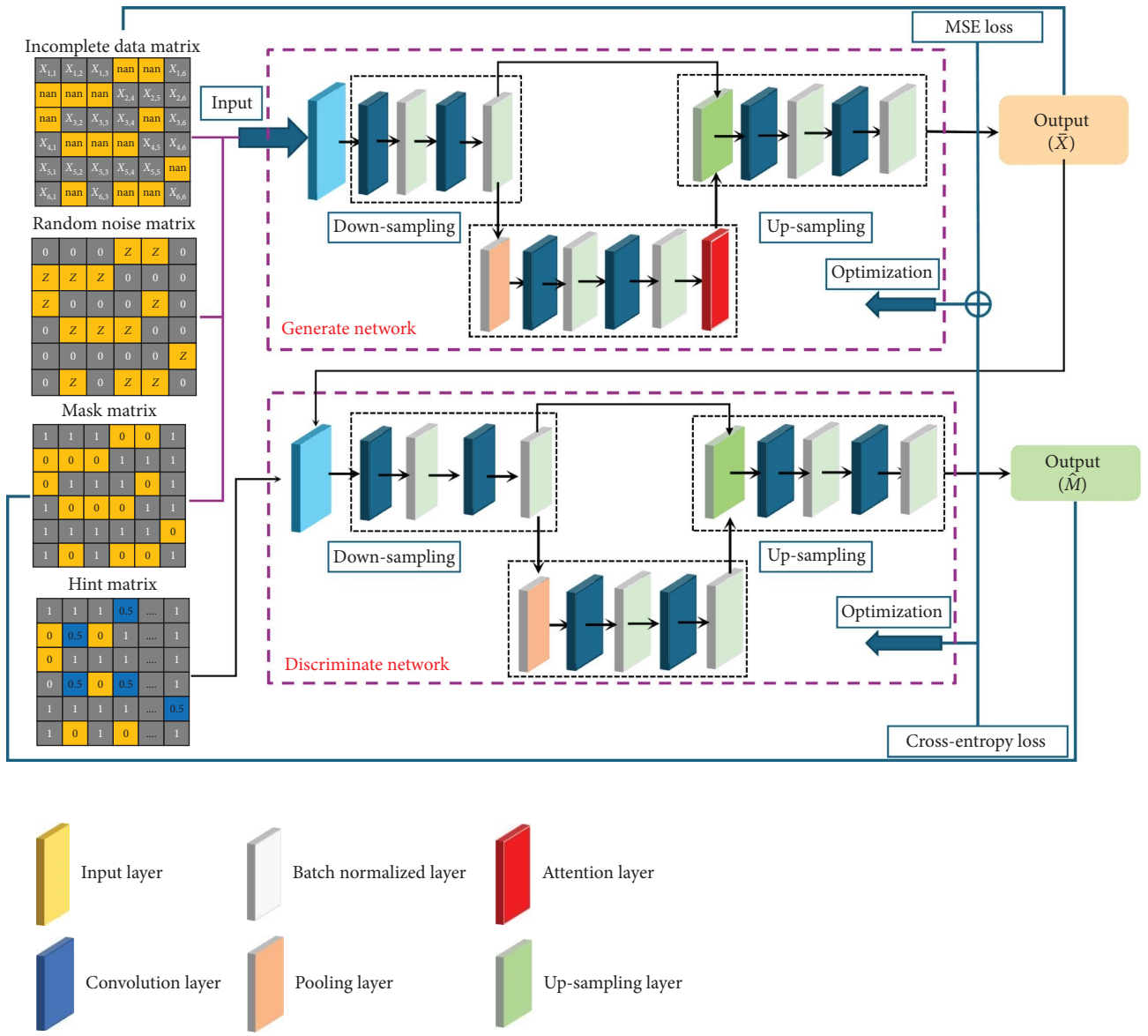


FIGURE 10: The proposed enhanced GAIN architecture and technical route.

FIGURE 11: Dowling Hall footbridge [34].

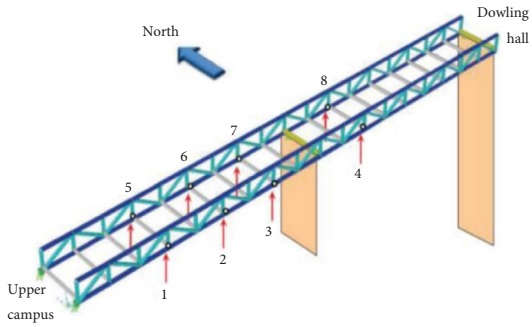


FIGURE 12: Layout of accelerometers [34].

TABLE 1: Hyperparameters employed in this study.

Hyperparameter	Value
Optimizer	Adam
β_1 (Adam parameter)	0.9
β_2 (Adam parameter)	0.99
Learning rate	0.01
Number of epochs	1000
Weight factor of reconstruction loss	100
Hint rate (h)	0.9

information flow and improving the training efficiency of the network. In this study, skip connections are incorporated to improve the GAIN, which can effectively enhance the overall performance of the network for random missing data imputation with incomplete data of all sensors, thus further enhancing model performance and estimation accuracy.

3.4. Loss Function of the Enhanced GAIN. The loss function is mainly used to measure the error between the output result and the target, which is an important part of the network. In the proposed method, the loss function of the generator is composed of a binary cross-entropy loss function and a mean square error (MSE) loss function. The binary cross-entropy loss is used to calculate the discriminant loss of the discriminant network and to optimize the discriminant network and the generation network. The MSE loss function is to ensure that the output of the generated network in the unmissed part is close to the original data, to better generate true data. The loss function for the generator can be illustrated as follows:

$$\begin{aligned}\min_G L(\alpha_G) = &-E_{\widehat{\mathbf{X}},\mathbf{M},\mathbf{H}}[(1-\mathbf{M})\odot D(\widehat{\mathbf{X}})] \\ &+ \alpha E\left[(\mathbf{M}\odot\widehat{\mathbf{X}} - \mathbf{M}\odot\overline{\mathbf{X}})^2\right],\end{aligned} \tag{10}$$

where α_G represents the network parameters of the generator; $\odot$ indicates the Hadamard product; α is the weight parameter. The term $\min_G L(\alpha_G)$ indicates the minimization of the generator's loss function.

The discriminator is trained to output values close to 1 for real data and values approaching 0 for the pseudo data (data imputed by the generator). The loss function of the discriminator can be expressed as follows:

$$\begin{aligned}\max_D L(\alpha_D) = &E_{\widehat{\mathbf{X}},\mathbf{M},\mathbf{H}}[\mathbf{M}\odot D(\widehat{\mathbf{X}},\mathbf{H}) \\ &+ (1-\mathbf{M})\odot(1-D(\widehat{\mathbf{X}},\mathbf{H}))],\end{aligned} \tag{11}$$

where α_D represents the network parameters of the discriminator. The term $\max_D L(\alpha_D)$ indicates the maximization of the discriminator's loss function.

3.5. Construction of the Enhanced GAIN for Data Imputation. In summary, in the enhanced GAIN, an incomplete acceleration sensor data matrix with a certain random missing $\widetilde{\mathbf{X}}$, a corresponding mask matrix $\mathbf{M}$, and a random noise matrix $\mathbf{Z}$ is input into the generator, to impute complete acceleration data $\overline{\mathbf{X}}$. For the discriminator, the imputed complete data $\widehat{\mathbf{X}}$ and hint matrix $\mathbf{H}$ are used to correctly predict the mask matrix $\widehat{\mathbf{M}}$. The network architecture and technical route of the proposed enhanced GAIN are shown in Figure 10.

4. Validation of the Enhanced GAIN for Random Missing Data Imputation of the Field-Measured Datasets of Dowling Hall Footbridge

4.1. Measured Datasets of Dowling Hall Footbridge. The Dowling Hall footbridge [34], as shown in Figure 11, is a pedestrian bridge located on the campus of Tufts University in Medford, Massachusetts, USA. The span and width of the bridge are 44 and 3.7 m, respectively. A continuous wireless SHM system is installed on the bridge to monitor the vibration response and environmental conditions continuously. From January to May 2010, the bridge was continuously monitored for 17 weeks. Continuous sampling was performed during the first 5 min of each hour during this process. A total of 8 PCB 393B04 single-axis accelerometers were installed at the bottom of the bridge, with a sampling frequency of 2048 Hz, and the accelerometer layout can be shown in Figure 12.

During postprocessing, the data were down-sampled to 128 Hz and band-pass filtered between 2–55 Hz. Since the literature [34] indicates low signal-to-noise ratios for Sensors 4 and 8, the data from these sensors were discarded. Instead, six accelerometers from the same span, namely, 1, 2, 3, 5, 6, and and 7, are selected to verify the performance of the proposed method. The data of a week contain 128 5 min sampling fragments, and each 5-min fragment contains 38,400 sampling points ($5\times60\times128 = 38{,}400$), but the actual data are only 38,144 samples. A five-minute segment from the first week is selected for training and verifying the effectiveness of the proposed method.

4.2. Preprocessing of Measured Datasets. First of all, according to the relevant literature [35], the frequency range of interest is between 4 and 14 Hz. Thus, to further reduce the redundant information contained in the acceleration data, the data from the six acceleration sensors are processed

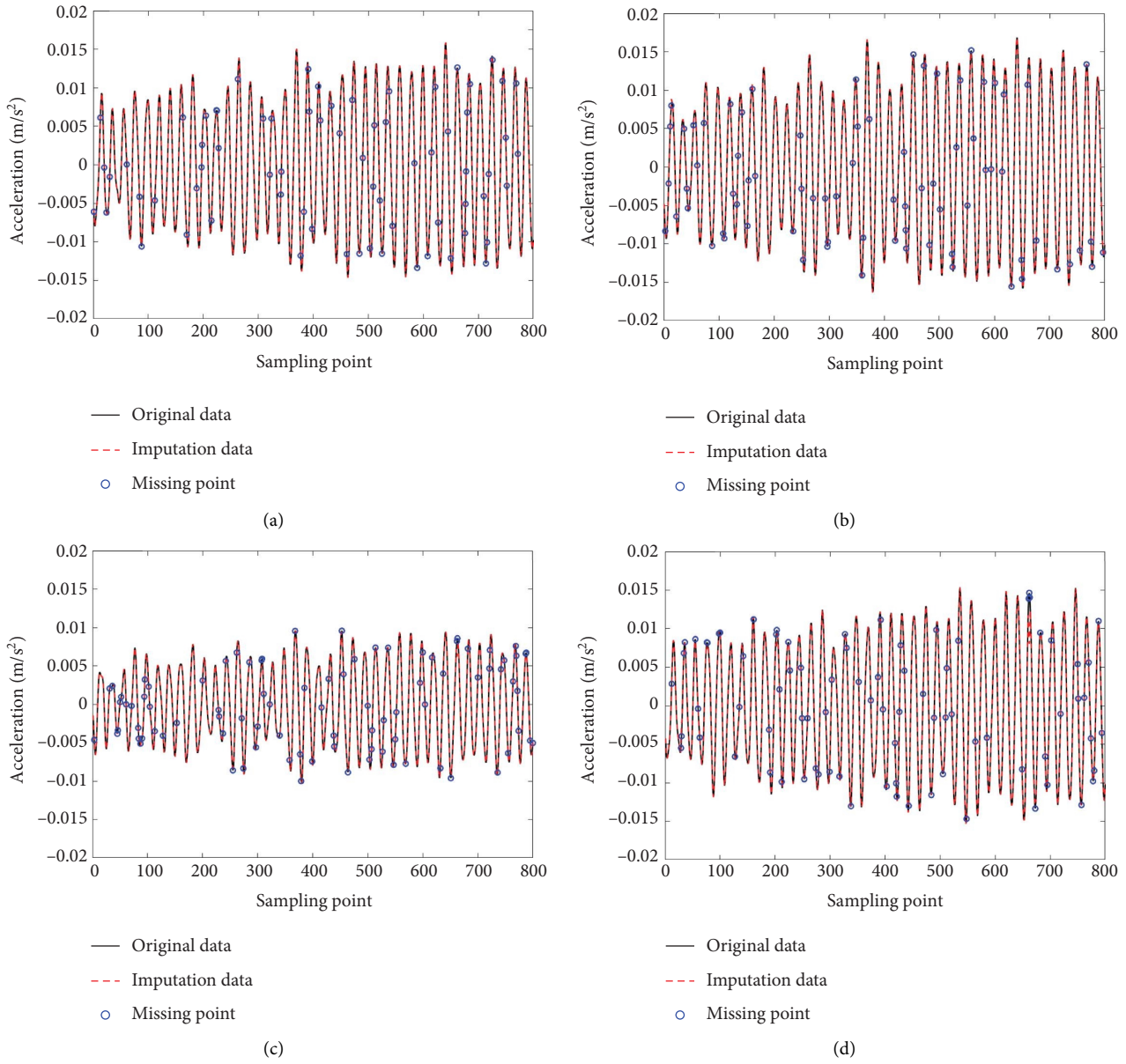


Figure 13: Continued.

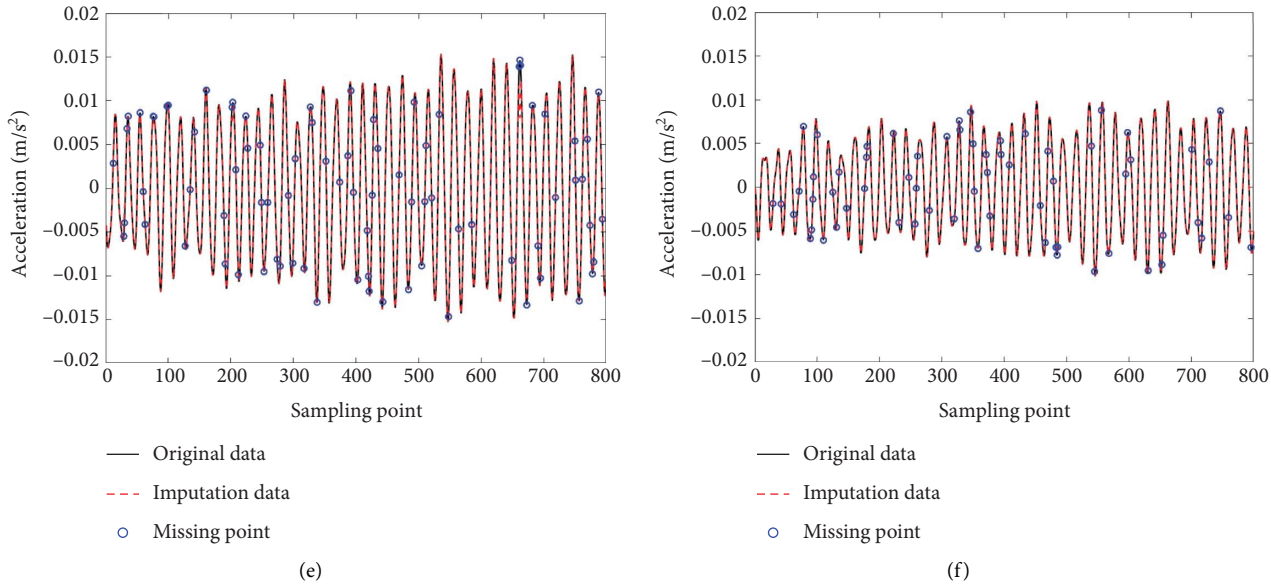


FIGURE 13: Time-domain comparisons of data imputation (10% missing rate in all sensors). (a) Sensor 1. (b) Sensor 2. (c) Sensor 3. (d) Sensor 5. (e) Sensor 6. (f) Sensor 7.

using a low-pass filter with a cutoff frequency of 16 Hz. Furthermore, to accelerate the training speed and enhance the accuracy of the network, the dataset is first normalized by the min-max normalization method, scaling the data from all acceleration sensors to a range between 0 and 1. The formula can be expressed as follows:

$$(x_{ti})_{\text{norm}} = \frac{x_{ti} - (\mathbf{X}_i)_{\min}}{(\mathbf{X}_i)_{\max} - (\mathbf{X}_i)_{\min}}, \tag{12}$$

where $(x_{ti})_{\text{norm}}$ represents the normalized sensor data of i-th sensor at t moment; $(\mathbf{X}_i)_{\max}$ and $(\mathbf{X}_i)_{\min}$ stand for the maximum and minimum values of each sensor data, respectively.

Then, the mask matrix is used to assign missing values on the complete dataset randomly, and an incomplete dataset corresponding to the missing rate can be obtained. Specifically, a certain proportion of sampling points is discarded in each sensor randomly and independently to simulate data missing in actual engineering. Since the data missing rate in practical engineering is relatively low and the high missing rate is relatively rare, three data missing cases with missing rates of 10%, 20%, and 30%, respectively, are considered in this paper.

Finally, to augment the dataset, the training data are segmented by a sliding window. The window size is 128, with a 50% overlap rate, resulting in a stride of 64. After the preprocessing of measured datasets, only the missing incomplete dataset is used to train the network, and the true value of the missing position is only used to calculate the error of the final imputation result.

4.3. Hyperparameter Selection and Network Training. The processed incomplete dataset, corresponding mask matrix, and random noise matrix are input into the enhanced GAIN. The spatio-temporal relationships between sensors are implicitly captured by the network. After undergoing adversarial unsupervised training, the complete dataset can be predicted by the neural network. In this dataset, the positions that are not missing are filled with the observed true data, while the missing positions are filled by the network output.

During training, all hyperparameters are selected through a combination of empirical initialization and loss-curve-based adjustment. Initially, hyperparameters such as learning rate, batch size, and noise standard deviation are set based on values commonly used in related studies. Subsequently, multiple trial runs are conducted while monitoring the training and validation loss curves. The final set of hyperparameters is determined based on the configuration that yielded the best performance on the validation set without overfitting. Adaptive moment estimation (Adam) optimizer has the advantages of high computational efficiency and fast convergence speed [35]. Therefore, the Adam optimizer is utilized in the generator and discriminator to implement gradient descent parameter updates, with the optimizer's parameters $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The learning rate is 0.01, and it performs 1000 epochs. The hyperparameter α is 100 in the generator loss function [22]. The hint rate h is 0.9, which represents the proportion of true information about the mask matrix in the hint matrix [17]. In addition, PyTorch is applied to construct this deep learning framework. Table 1 summarizes the hyperparameters employed in this study.

4.4. Evaluation Metrics. To measure the imputation effect of the proposed method, the L_1 norm relative error (RL_1) and L_2 norm relative error (RL_2) of the original data and the imputation data are selected as the evaluation criteria. The RL_1 aims to measure the overall performance, and the RL_2 is more sensitive to the outliers. The specific formulas of the i-th sensor are as follows:

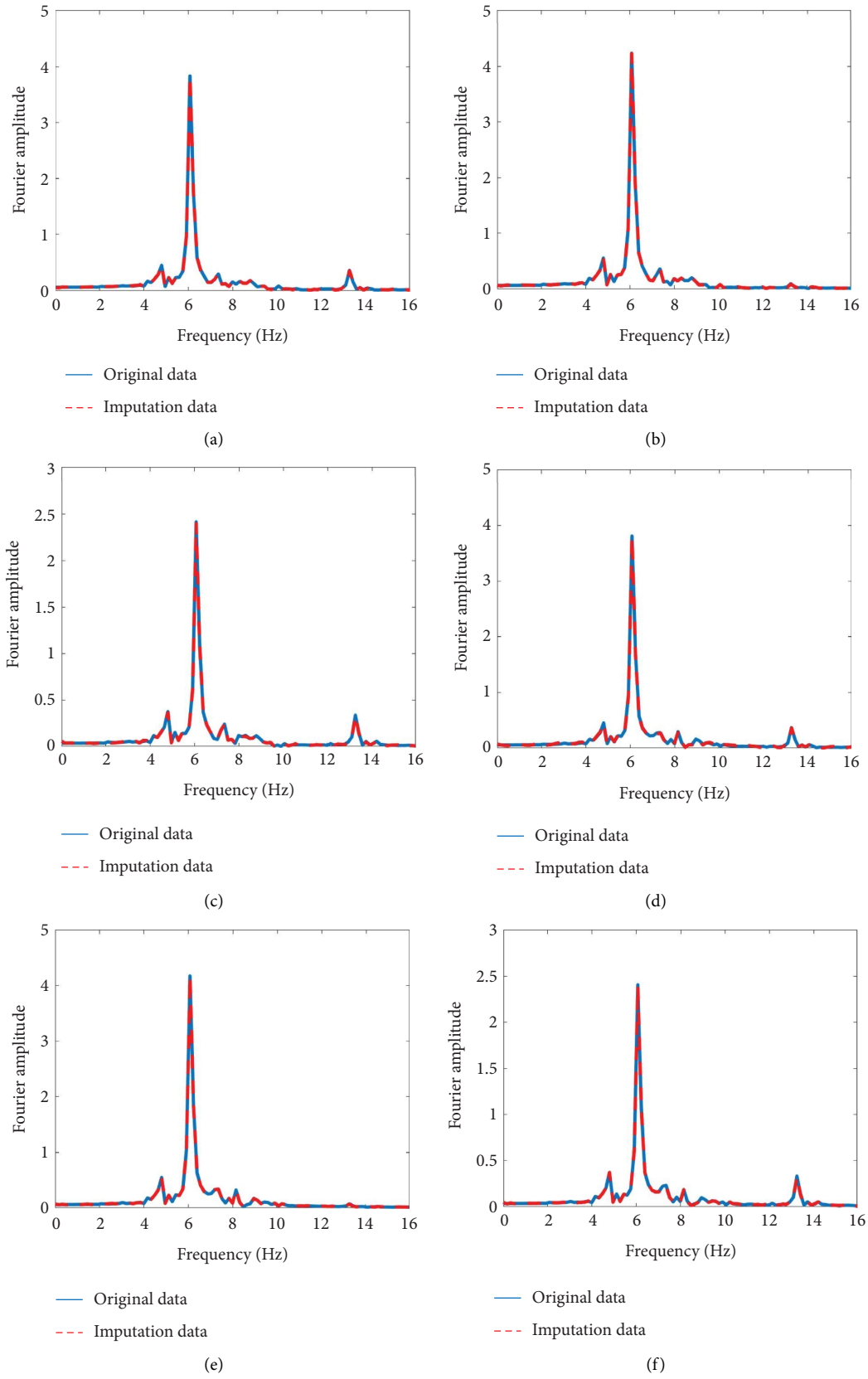
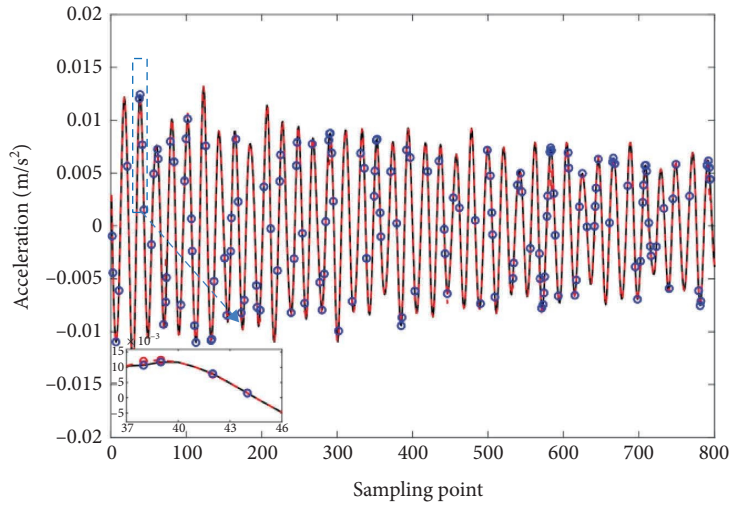
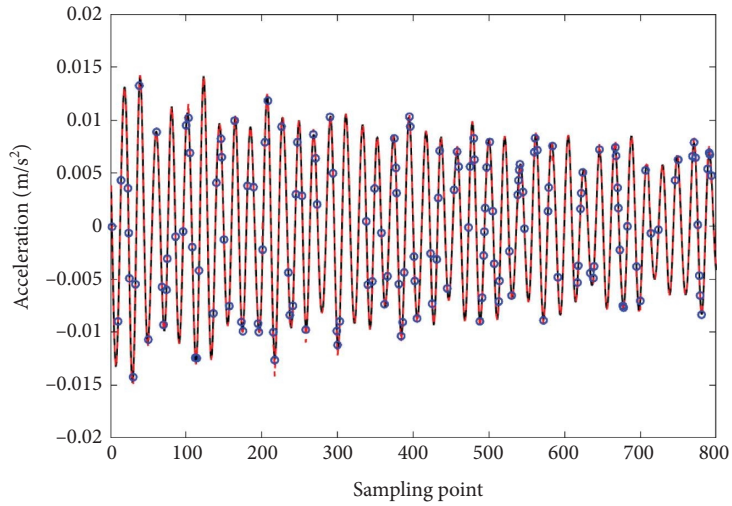


FIGURE 14: Frequency-domain comparisons of data imputation (10% missing in all sensors). (a) Sensor 1. (b) Sensor 2. (c) Sensor 3. (d) Sensor 5. (e) Sensor 6. (f) Sensor 7.

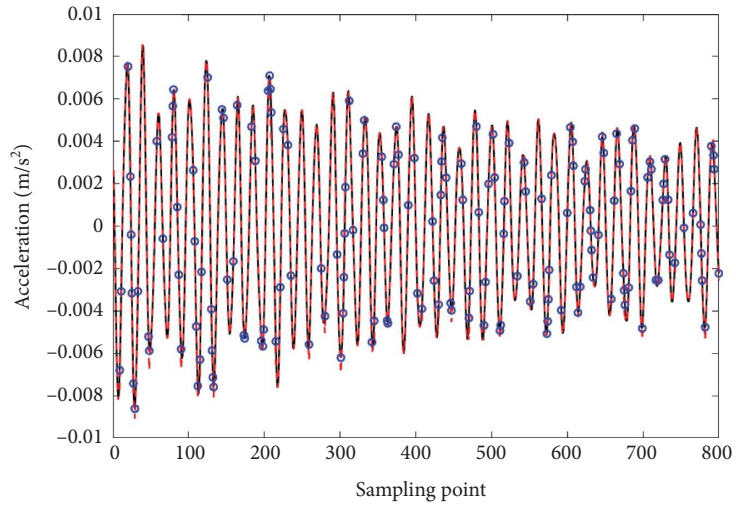


(a)

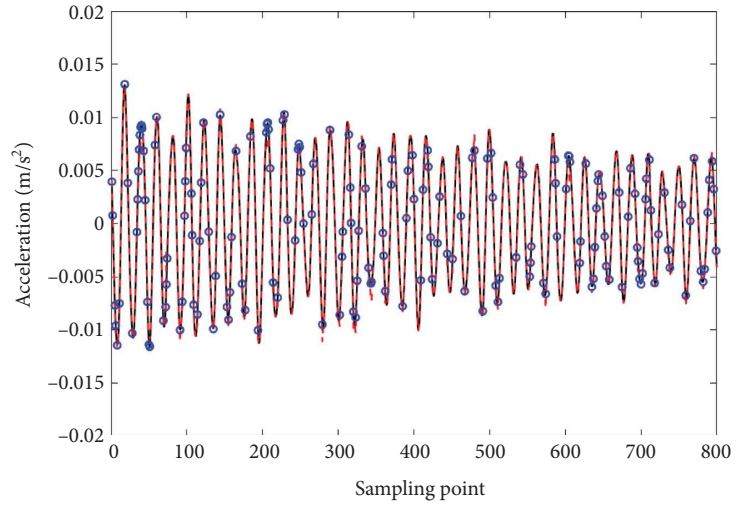


(b)

Figure 15: Continued.



(c)



(d)

Figure 15: Continued.

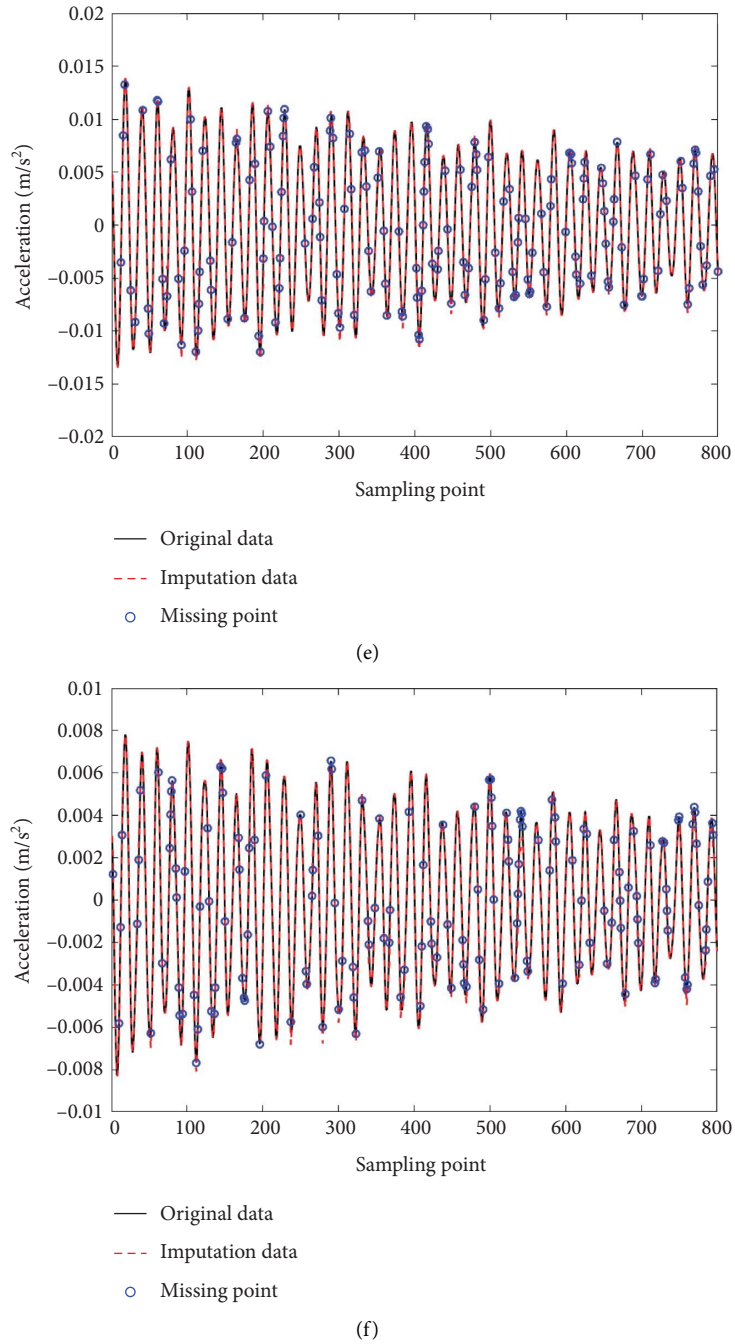


Figure 15: Time-domain comparisons of data imputation (20% missing rate in all sensors). (a) Sensor 1. (b) Sensor 2. (c) Sensor 3. (d) Sensor 5. (e) Sensor 6. (f) Sensor 7.

$$(RL_1)_i = \frac{\sum_{j=1}^{m}\left|\left(x_{ij} - \widehat{x}_{ij}\right)\right|}{\sum_{j=1}^{m}\left|\widehat{x}_{ij}\right|},$$
$$(RL_2)_i = \sqrt{\frac{\sum_{j=1}^{m}\left(x_{ij} - \widehat{x}_{ij}\right)^2}{\sum_{j=1}^{m}\widehat{x}_{ij}^2}}, \tag{13}$$

where x_{ij} represents the original data of the i–th sensor at the missing position j, and $\widehat{x}_{ij}$ denotes the imputation data of the i–th sensor at the missing position j. Moreover, only the errors of missing points are calculated to accurately value the performance of the proposed method; that is, m represents the number of missing points in this paper.

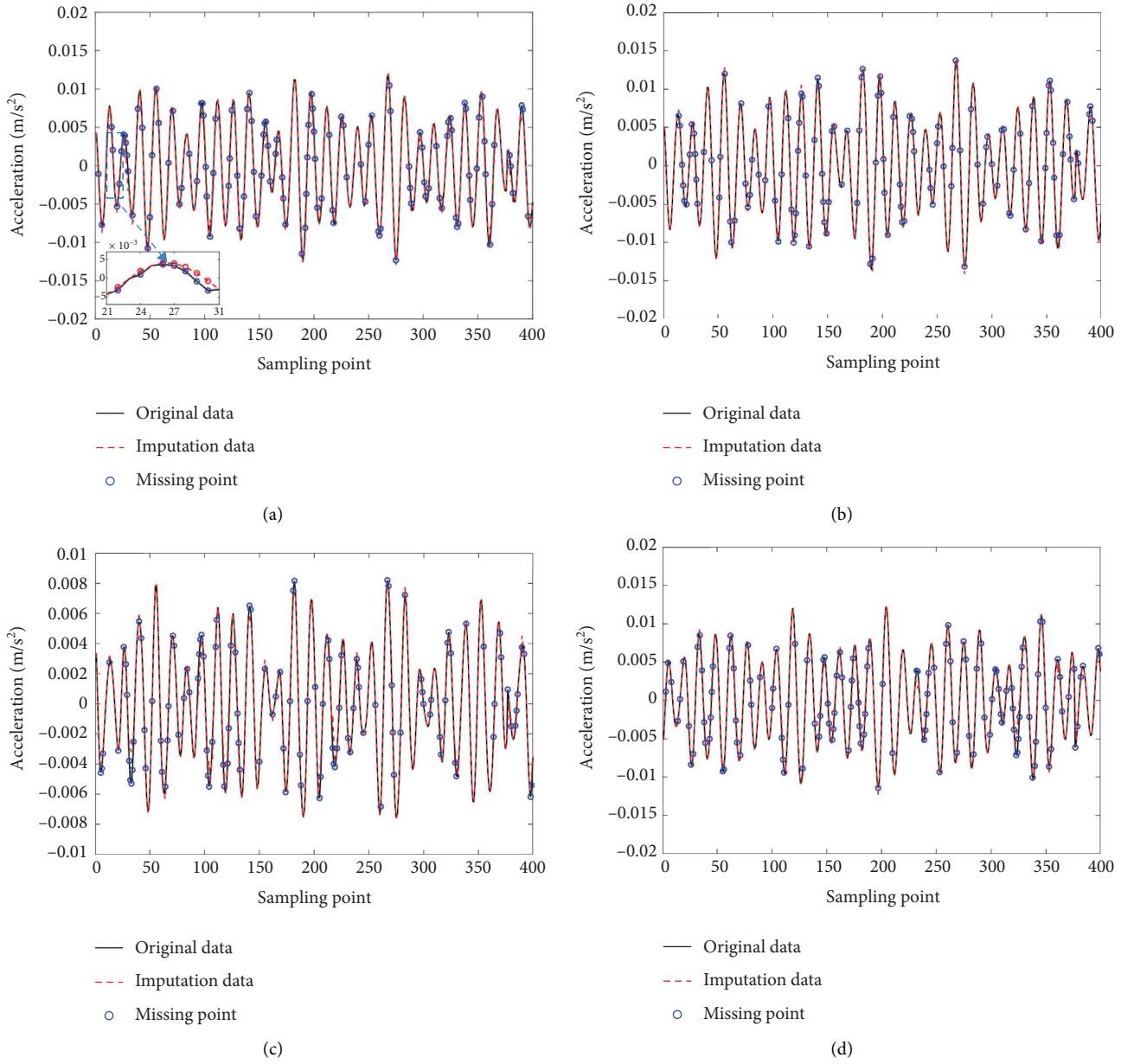


Figure 16: Continued.

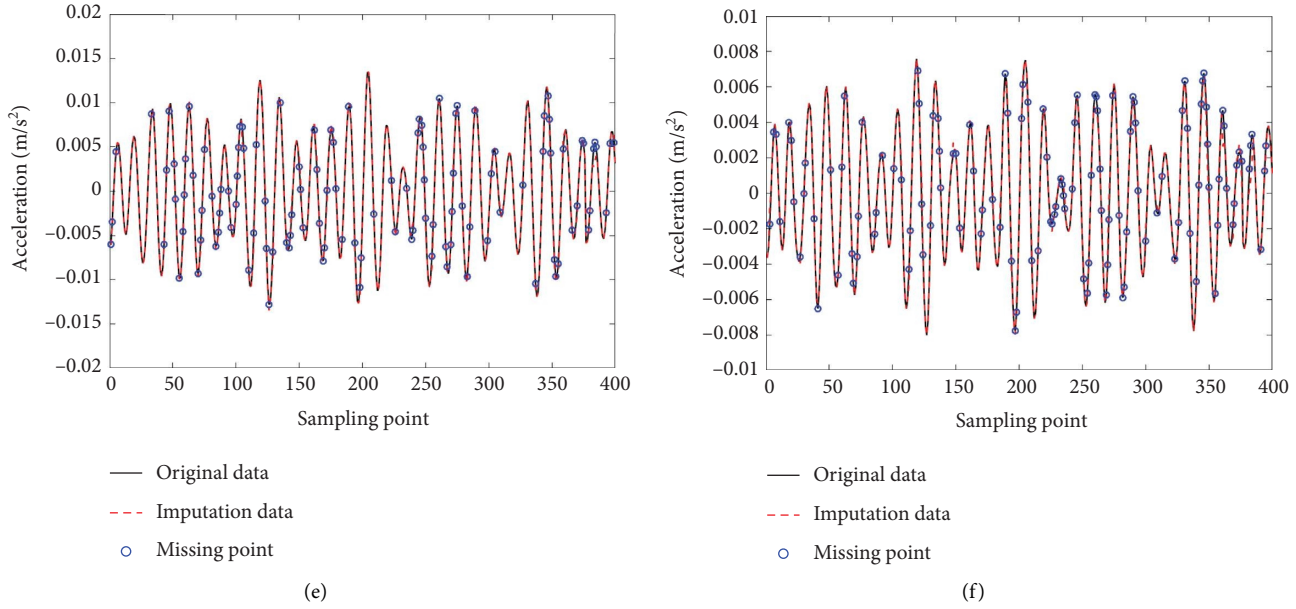


FIGURE 16: Time-domain comparisons of data imputation (30% missing rate in all sensors). (a) Sensor 1. (b) Sensor 2. (c) Sensor 3. (d) Sensor 5. (e) Sensor 6. (f) Sensor 7.

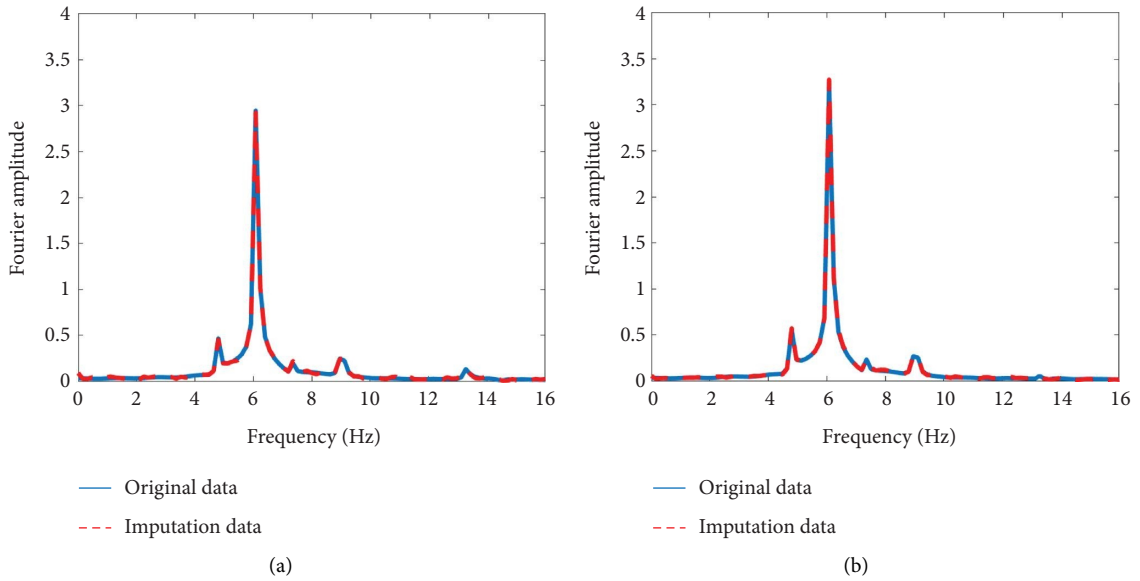


FIGURE 17: Continued.

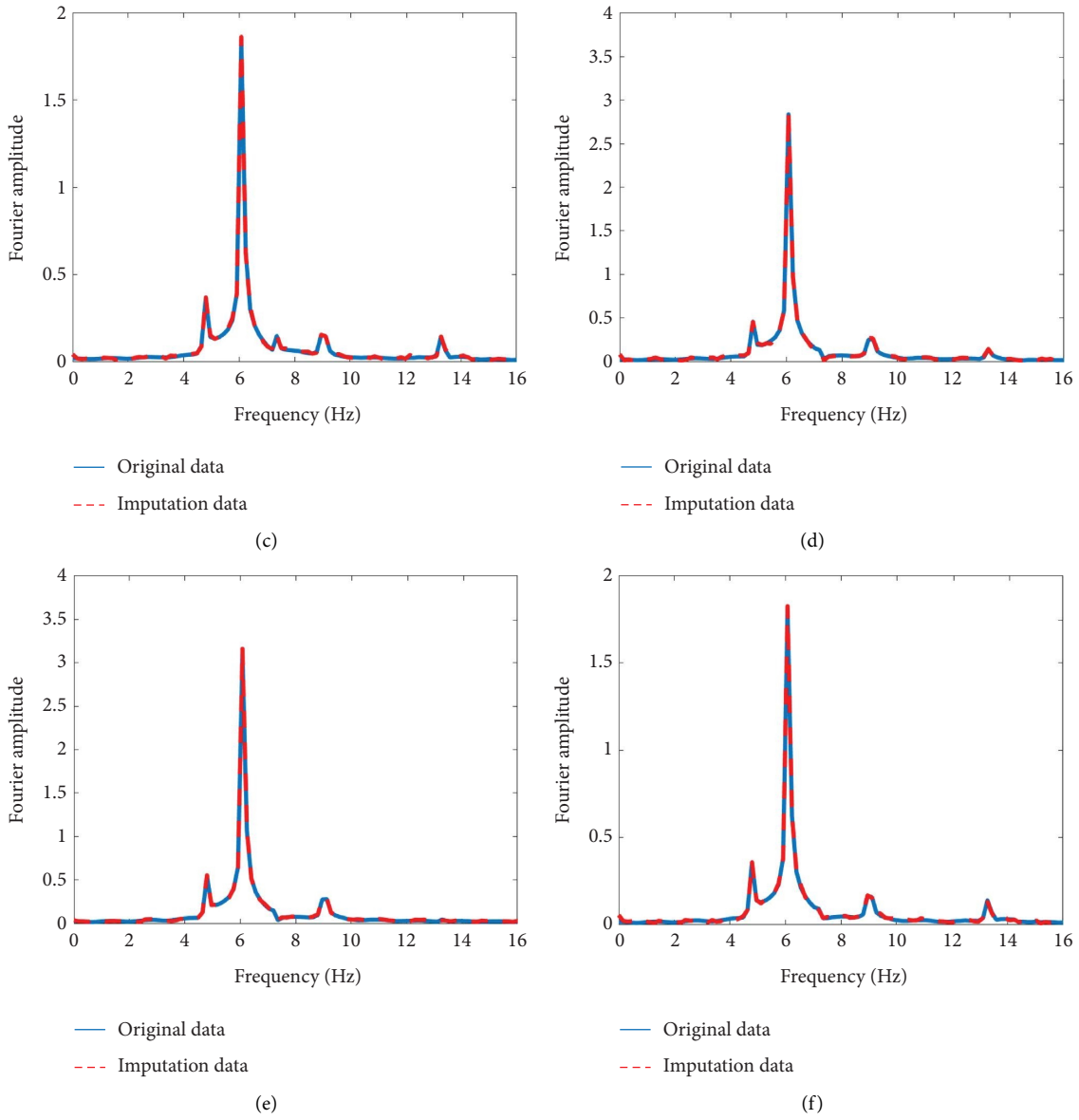


FIGURE 17: Frequency-domain comparisons of data imputation (20% missing in all sensors). (a) Sensor 1. (b) Sensor 2. (c) Sensor 3. (d) Sensor 5. (e) Sensor 6. (f) Sensor 7.

4.5. Results and Discussions of the Proposed Enhanced GAIN for Random Missing Data Imputation With Incomplete Data of All Sensors

4.5.1. Imputation Result and Discussion With Incomplete Data of All Sensors. To more clearly demonstrate the results of the acceleration data imputation, only a subset of the sampling points is displayed between the imputation results and the actual data. When the missing rate is 10%, the time-domain imputation results for 800 randomly selected sampling points from the six acceleration sensors are illustrated in Figure 13. The blue dots represent the positions of the missing samples, which follow the pattern of random missing.

It can be seen from Figure 13 that the imputation data at the missing positions are very close to the actual data in the time domain. This demonstrates that the proposed method can impute the random missing data well in the time domain if all sensors have the same missing rate of 10%, and only the incomplete datasets are used for training.

Furthermore, the imputation results are compared in the frequency domain in Figure 14 to demonstrate the performance of the proposed method.

It can be observed that although there is random missing data in all sensors, the frequency domain curves of the imputation data match very well with the original data. Thus, in both the time and frequency domains, very good performance can be obtained from the proposed approach.

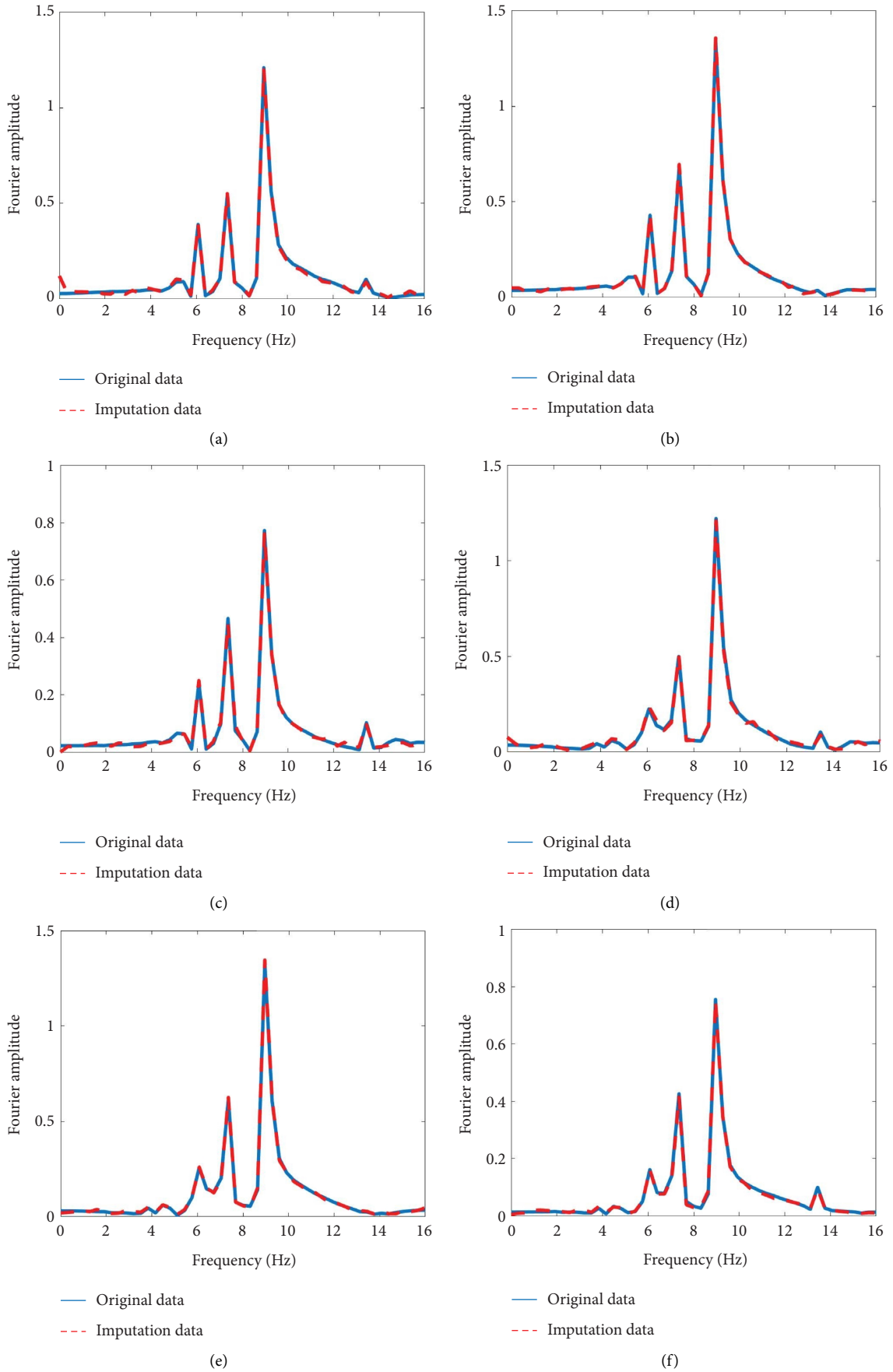


FIGURE 18: Frequency-domain comparisons of data imputation (30% missing in all sensors). (a) Sensor 1. (b) Sensor 2. (c) Sensor 3. (d) Sensor 5. (e) Sensor 6. (f) Sensor 7.

Table 2: Evaluation metrics of all sensors at different missing rates.

Missing rate (%)	Evaluation metrics (%)	Average errors	Sensor 1	Sensor 2	Sensor 3	Sensor 5	Sensor 6	Sensor 7
10	RL_1	10.55	10.3	7.6	14.1	11.3	7.2	13.1
	RL_2	12.88	12.8	9.07	16.9	14.4	9.0	15.1
20	RL_1	13.97	13.6	10.2	15.1	16.5	11.5	16.9
	RL_2	16.18	17.5	11.9	17.7	18.7	12.3	19
30	RL_1	16.47	24.1	13.6	17.7	16.9	10.8	15.7
	RL_2	18.67	25.1	14.7	20.2	20.7	12.4	18.9

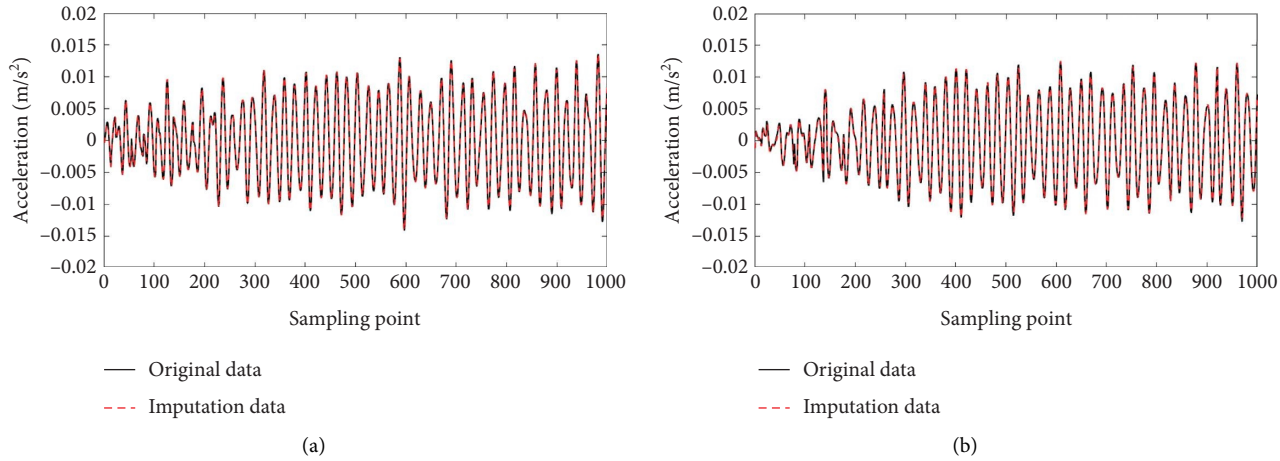


FIGURE 19: Time-domain comparisons of data imputation with 50% missing in sensors 2 and 6. (a) Sensor 2. (b) Sensor 6.

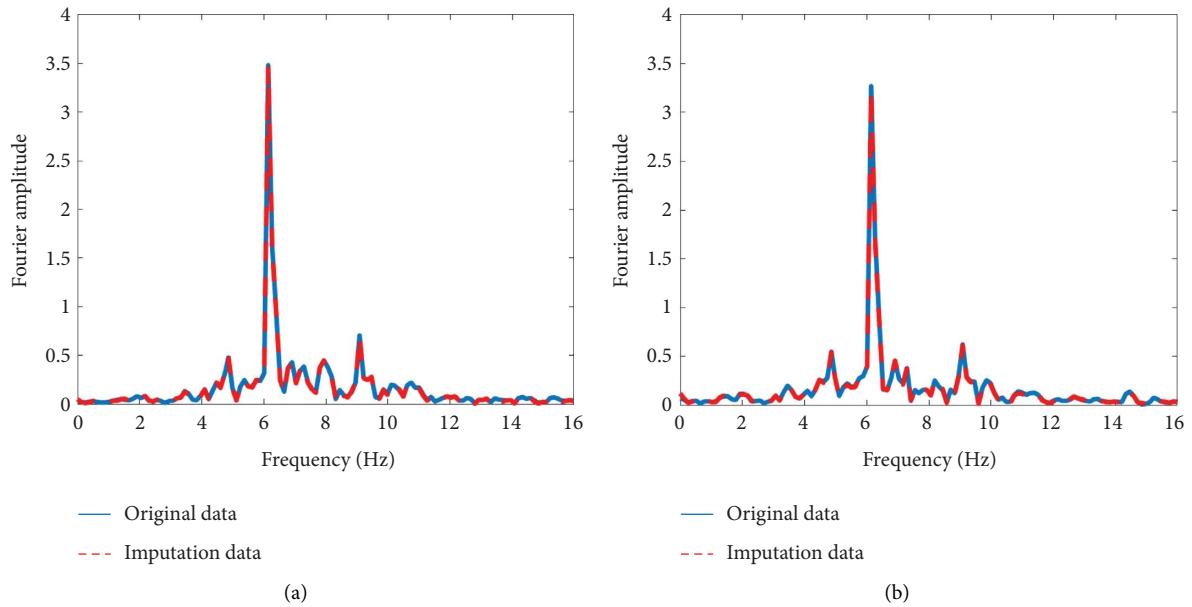


FIGURE 20: Frequency-domain comparisons of data imputation with 50% missing in sensors 2 and 6. (a) Sensor 2. (b) Sensor 6.

TABLE 3: Evaluation metrics of sensors 2 and 6 at 50% missing rate.

Missing rate (%)	Evaluation metrics (%)	Average errors	Sensor 2	Sensor 6
50	RL_1	5.48	5.01	5.96
	RL_2	6.77	6.25	7.29

In addition, to further verify the impact of the missing rate on the imputation results, datasets with missing rates of 20% and 30% are used to test the effectiveness of network imputation, respectively. The time-domain comparison of 800 randomly selected sample points with a 20% missing rate and 400 sample points with a 30% missing rate is shown in Figures 15 and 16, respectively. From Figure 15, it can be concluded that when the 20% missing rate is considered, the missing samples are well imputed, and the imputation data fit well with the original data in the time domain. Furthermore, from Figure 16, it is shown that even if the 30% missing rate is considered in all sensors, satisfactory imputation results can also be achieved. In particular, the detailed figures in Figures 15(a) and 16(a) clearly show that good imputation results at both missing rates can be obtained by the proposed method.

The corresponding frequency domain comparisons are shown in Figures 17 and 18, respectively. It can be observed that the proposed method shows good imputation results in the frequency domain when the missing rate is 20%. In the case of the 30% missing rate, satisfactory imputation results can also be obtained in the frequency domain. Consequently,

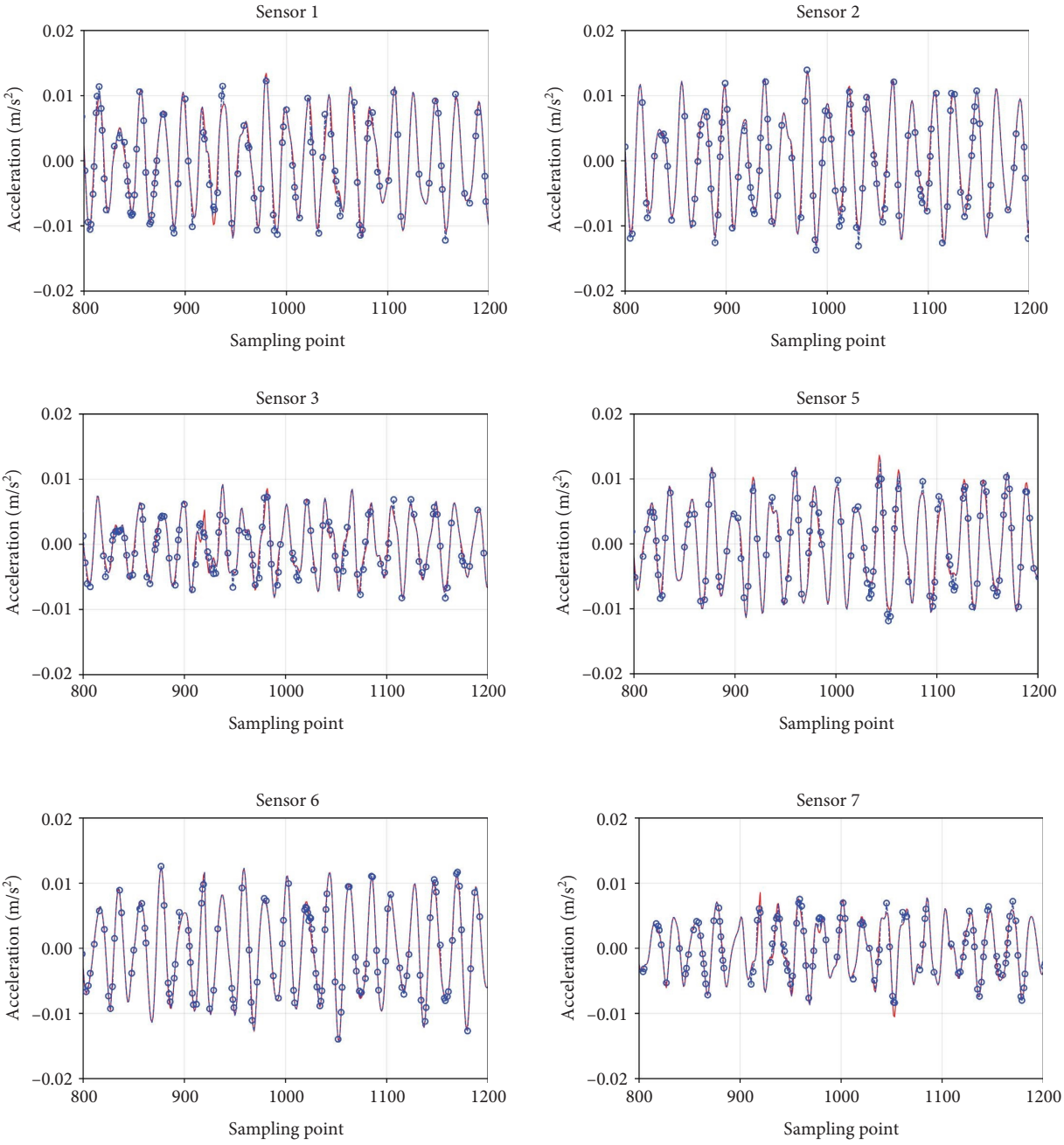


FIGURE 21: Data imputation (30% missing rate in all sensors) with removed skip connections.

under the cases of 20% and 30% missing rates, the imputation data not only reflect the shape of the original data in the time domain but also accurately reproduce the characteristics of the data in the frequency domain.

The imputation errors of the acceleration across the above missing rate cases are presented in Table 2. The results indicate that good imputation results can be achieved for the acceleration of all three scenarios. As the missing rate increases, the imputation errors are observed to gradually rise due to the reduced availability of nonmissing data for imputation in each sensor. However, the imputation errors for all sensors are acceptable.

Meanwhile, as can be seen from Table 1, the accuracy of imputation results of different sensors is not the same, which is due to the different correlations between sensors. Sensors 2 and 6 are located at the center of each span among the three sensors and have high correlations with adjacent sensors, so the imputation result errors are small. In contrast, the

Table 4: Relative L2 norm error for all sensors under a 30% missing rate using the proposed enhanced GAIN, with and without skip connection design.

Imputation model	Average errors (%)	Sensor 1 (%)	Sensor 2 (%)	Sensor 3 (%)	Sensor 5 (%)	Sensor 6 (%)	Sensor 7 (%)
Enhanced GAIN (with skip connections)	18.7	25.1	14.7	20.2	20.7	12.4	18.9
Enhanced GAIN (without skip connections)	26.0	28.2	22.2	29.3	27.4	21.1	27.9

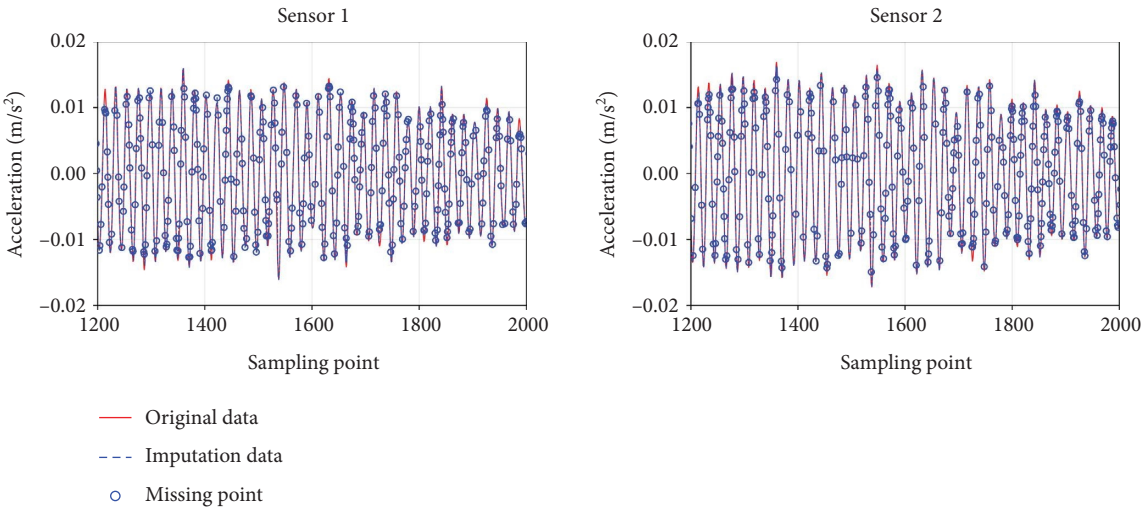


Figure 22: Data imputation with 40% missing in sensors 1 and 2 by the proposed method.

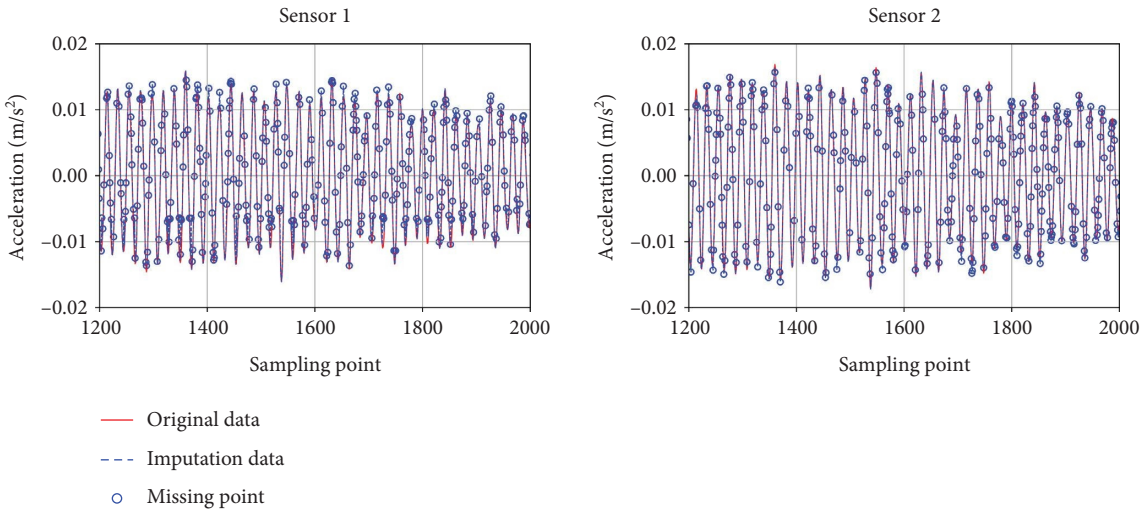


Figure 23: Data imputation with 40% missing in sensors 1 and 2 by the WGAIN-GP.

sensors located at the edge positions have lower accuracy of imputation results due to fewer sensors with high correlation.

To further verify the effectiveness of the proposed method for all sensors with higher data missing rates, a 40% data missing rate is considered in all sensors. However, the imputation data cannot be well fitted with the true data in the time domain and frequency domain, and the imputation errors of all sensors are quite large, which is unacceptable. Due to the limited space, the specific results are not presented.

4.5.2. Imputation Results and Discussions With High Missing Rates of Only Some Sensors. Furthermore, the data imputation performance by the proposed method is verified under the condition of a high missing rate only in some sensors while the other sensors are completed, as investigated in current researcher. In this case, a 50% missing rate is considered in the acceleration data of sensor 2 and sensor 6 randomly. Then, the incomplete dataset considering the high missing rate of some sensors is input into the network for training and testing. The comparison of

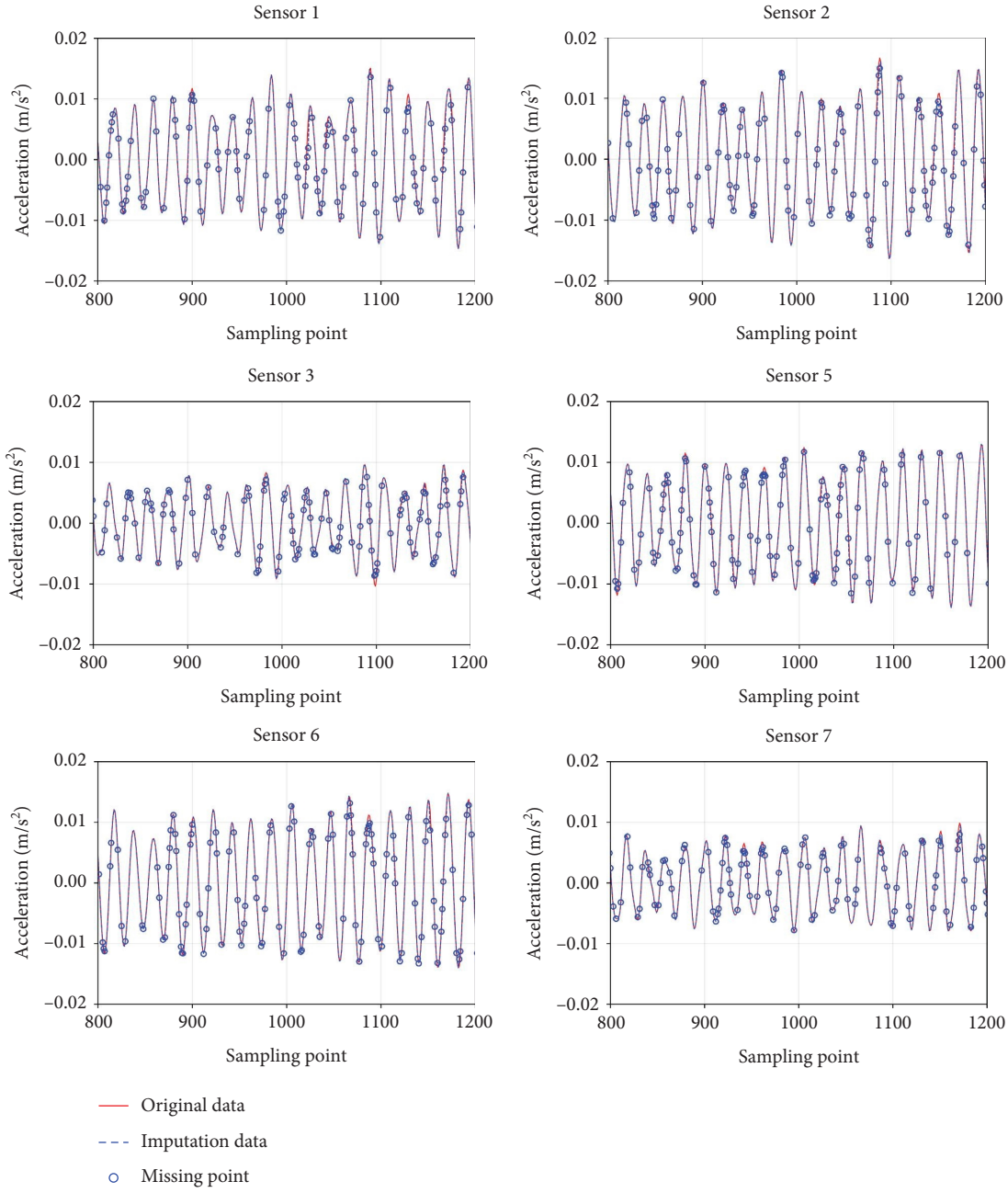


Figure 24: Time-domain comparisons of data imputation with 30% missing in all sensors using the proposed method.

imputation results with the true data in the time domain can be shown in Figure 19 (limited to space, only 1000 sampling points are randomly selected). Moreover, the frequency domain comparisons are presented in Figure 20. The comparison results illustrated that the proposed method shows excellent data imputation performance for some sensors with high missing rates in both the time domain and frequency domain.

The overall imputation errors for sensor 2 and sensor 6 are illustrated in Table 3. It can be seen that the errors of the imputation results are within 10%, indicating that the proposed method is also applicable to the case where some sensors have high missing rates. When there is some complete sensor data, the spatial correlation of other data points can be used to impute the random missing data, so it is easier to impute than when there are no complete data. In summary, these results demonstrate that when some sensors have high data missing rates, the proposed method can obtain good imputation results in both the time domain and frequency domain.

4.5.3. Imputation Results and Discussions With Incomplete Data of All Sensors Without Skip Connections. To further validate the design of the enhanced GAIN imputation model, the skip connections were removed. The

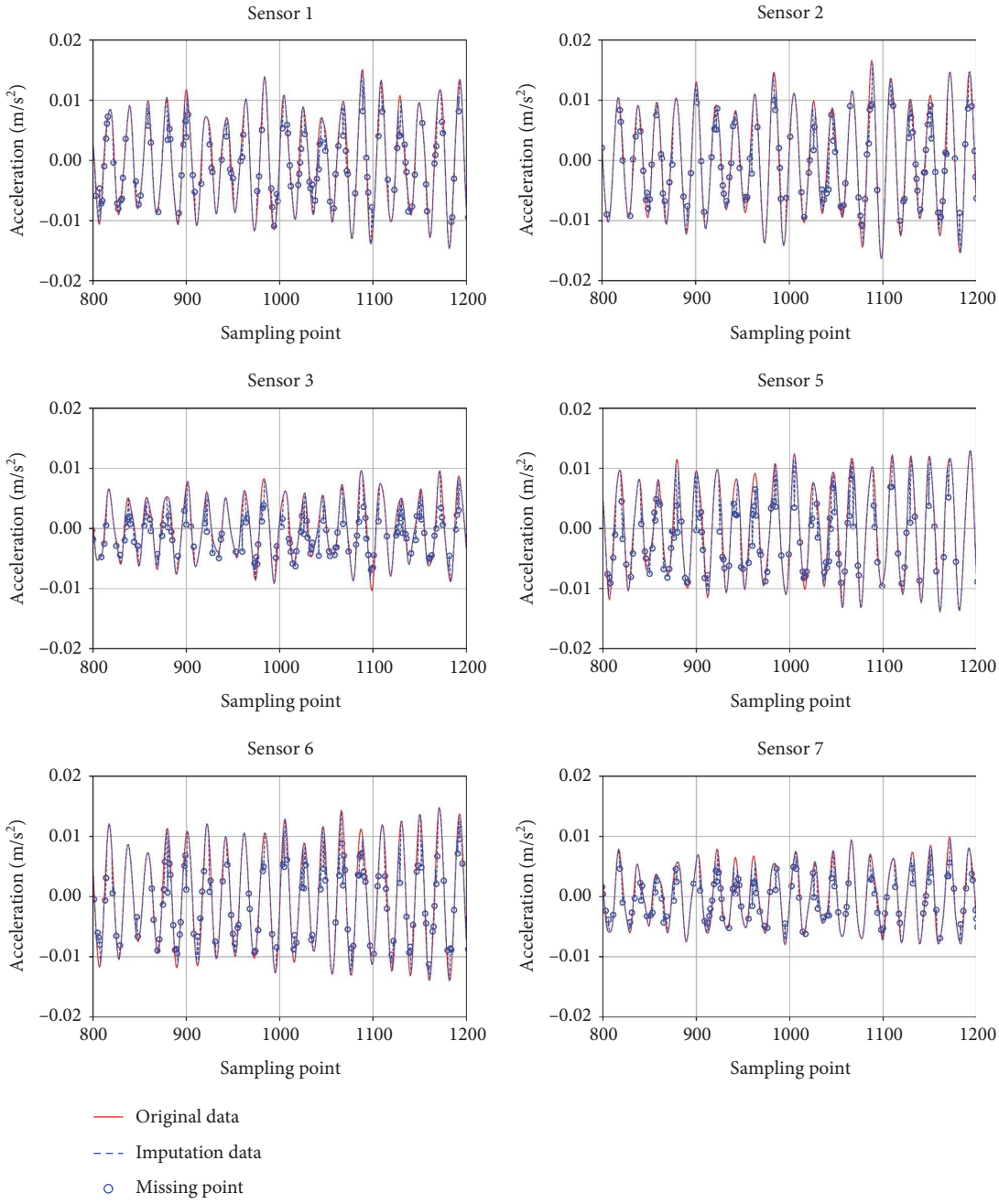


Figure 25: Time-domain comparisons of data imputation with 30% missing in all sensors using the WGAIN-GP.

corresponding imputation results are presented in Figure 21. The average relative error, measured by the L2 norm, increased to 26.0%, which is notably higher than the 18.67% reported in Table 4. This result confirms that the original design, which incorporates skip connections, effectively enhances the model's imputation accuracy.

4.6. Comparative Study and Discussions. In this section, the proposed method is compared with a recent GAIN variant, WGAIN [22]. For a fair comparison, both models are evaluated on the same dataset under identical data loss conditions. Two scenarios are considered: (1) Among the six sensors, two sensors experience a random missing rate of 40%, while the remaining four sensors remain intact. This setting is used to assess the reliability of both imputation methods under partial sensor failure. (2) All six sensors simultaneously exhibit a random missing rate of 30%, allowing for a comparative analysis of the proposed method's effectiveness when missing data occur across all sensors. As shown in Figures 22 and 23, both imputation models perform well under Scenario 1, where only two sensors are partially missing. The average L2 norm relative errors are 13.8% and 18.36% for the proposed method and the WGAIN model, respectively.

Table 5: Relative L2 norm error of all sensors with 30% missing rates using two different imputation models.

Imputation model	Average errors (%)	Sensor 1 (%)	Sensor 2 (%)	Sensor 3 (%)	Sensor 5 (%)	Sensor 6 (%)	Sensor 7 (%)
Enhanced GAIN	18.7	25.1	14.7	20.2	20.7	12.4	18.9
WGAIN-GP	55.3	46.4	42.3	69.1	63.3	49.1	61.6

Under the condition where all sensor channels exhibit 30% random missing data, the imputation results obtained by the proposed enhanced GAIN and the WGAIN-GP models are shown in Figures 24 and 25, respectively. It can be observed that the proposed enhanced GAIN, which incorporates a CNN module into the generator, significantly improves imputation accuracy when data loss occurs across all sensors. The relative L2 norm errors for all sensors are summarized in Table 5. The proposed model achieves a relative L2 norm error of 18.7%, significantly outperforming WGAIN-GP, which exhibits an average error of 55.3%. These results demonstrate that, compared to previous GAIN variants, the proposed method offers substantial improvement in handling scenarios involving widespread sensor data loss.

5. Conclusions

For the problem of data imputation when all sensor data are randomly missing in practical engineering, current GAIN-based imputation methods cannot be adopted to treat this challenging problem. In this paper, an enhanced GAIN is proposed for unsupervised random missing data imputation with incomplete data of all sensors. The accuracy and efficiency are verified by field-measured acceleration data from the Dowling Hall footbridge. Some conclusions can be summarized as follows:

1. In the proposed enhanced GAIN, encoder–decoder architectures consisting of CNN are integrated into the generator and discriminator to enhance the learning and expression capabilities of the network. At the same time, a self-attention mechanism is embedded into the generator to extract global features by focusing more attention on important features. Moreover, skip connection technology is incorporated into the generator and discriminator to improve the feature utilization rate and alleviate the problem of gradient vanishing. These improvements greatly increase the feature extraction ability of the proposed enhanced GAIN, so even when all sensor data are randomly missing, true data can be imputed by the enhanced GAIN.
2. In the proposed method, even if there is a certain loss in all sensors, only the incomplete datasets are used to train the network. After the unsupervised training, the data in the missing position can be imputed. Therefore, the problem of data imputation when all sensor data are randomly missing can be effectively solved.
3. For imputation with incomplete data of all sensors, the validation results show that the imputation data of all sensors are very close to the original data at 10% missing rate, it has good imputation results at 20% missing rate, and satisfactory imputation results can also be obtained for 30% missing rate in all sensors; however, the imputation errors are large at 40% missing rate.

Data Availability Statement

All data used in the study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding

This study was supported by the National Natural Science Foundation of China, Grant No. 52178304.

Acknowledgments

The research in this paper was supported by the National Natural Science Foundation of China through the Grant No. 52178304.

References

[1] Y. Q. Bao, Z. C. Chen, S. Y. Wei, Y. Xu, Z. Y. Tang, and H. Li, "The State of the Art of Data Science and Engineering in Structural Health Monitoring," *Engineering* 5, no. 2 (2019): 234–242, https://doi.org/10.1016/j.eng.2018.11.027.

[2] C. W. Zhang, A. A. Mousavi, S. F. Masri, G. Gholipour, K. Yan, and X. L. Li, "Vibration Feature Extraction Using Signal Processing Techniques for Structural Health Monitoring: A Review," *Mechanical Systems and Signal Processing* 177 (2022): 109175, https://doi.org/10.1016/j.ymssp.2022.109175.

[3] R. T. Wu and M. R. Jahanshahi, "Data Fusion Approaches for Structural Health Monitoring and System Identification: Past, Present, and Future," *Structural Health Monitoring* 19, no. 2 (2020): 552–586, https://doi.org/10.1177/1475921719859016.

[4] T. Nagayama, S. H. Sim, Y. Miyamori, and B. F. Spencer Jr, "Issues in Structural Health Monitoring Employing Smart Sensors," *Smart Structures and Systems* 3 (2007): 299–320, https://doi.org/10.12989/sss.2007.3.3.299.

[5] K. J. Jiang, Q. Han, and X. L. Du, "Lost Data Neural Semantic Recovery Framework for Structural Health Monitoring Based on Deep Learning," *Computer-Aided Civil and Infrastructure Engineering* 37 (2022): 1160–1187, https://doi.org/10.1111/mice.12779.

[6] R. W. Harris, "Spectra from Data With Missing Values," *Mechanical Systems and Signal Processing* 1 (1987): 97–104, https://doi.org/10.1016/0888-3270(87)90010-0.

[7] T. J. Matarazzo and S. N. Pakzad, "Structural Modal Identification Using Data Sets With Missing Observations," in *Proceedings of SPIE Smart Structures and Materials +*

Nondestructive Evaluation and Health Monitoring (2013), https://doi.org/10.1117/12.2009876.

[8] Z. C. Chen, H. Li, and Y. Q. Bao, "Analyzing and Modeling Inter-Sensor Relationships for Strain Monitoring Data and Missing Data Imputation: A Copula and Functional Data-Analytic Approach," *Structural Control and Health Monitoring* 18 (2019): 1168–1188, https://doi.org/10.1002/stc.2397.

[9] Y. Q. Bao and H. Li, "Machine Learning Paradigm for Structural Health Monitoring," *Structural Control and Health Monitoring* 20 (2021): 1353–1372, https://doi.org/10.1002/stc.2796.

[10] Y. Q. Bao, H. Li, X. D. Sun, Y. Yu, and J. P. Ou, "Compressive Sampling-Based Data Loss Recovery for Wireless Sensor Networks Used in Civil Structural Health Monitoring," *Structural Control and Health Monitoring* 12 (2013): 78–95, https://doi.org/10.1002/stc.483.

[11] Y. Q. Bao, Y. Yu, H. Li, et al., "Compressive Sensing-Based Lost Data Recovery of Fast-Moving Wireless Sensing for Structural Health Monitoring," *Structural Control and Health Monitoring* 22 (2015): 433–448, https://doi.org/10.1002/stc.1708.

[12] Y. Huang, J. L. Beck, S. Wu, and H. Li, "Bayesian Compressive Sensing for Approximately Sparse Signals and Application to Structural Health Monitoring Signals for Data Loss Recovery," *Probabilistic Engineering Mechanics* 46 (2016): 62–79, https://doi.org/10.1016/j.probengmech.2016.06.003.

[13] X. W. Ye, T. Jin, and C. B. Yun, "A Review on Deep Learning-based Structural Health Monitoring of Civil Infrastructures," *Smart Structures and Systems* 24, no. 5 (2019): 567–585, https://doi.org/10.12989/sss.2019.24.5.567.

[14] G. Fan, J. Li, and H. Hao, "Lost Data Recovery for Structural Health Monitoring Based on Convolutional Neural Networks," *Structural Control and Health Monitoring* 26 (2019): e2433, https://doi.org/10.1002/stc.2433.

[15] W. Wang, Y. M. Chai, and Y. Li, "GAGIN: Generative Adversarial Guider Imputation Network for Missing Data," *Neural Computing & Applications* 34, no. 10 (2022): 7597–7610, https://doi.org/10.1007/s00521-021-06177-3.

[16] R. Shahbazian and S. Greco, "Generative Adversarial Networks Assist Missing Data Imputation: A Comprehensive Survey and Evaluation," *IEEE Access* 11 (2023): 88908–88928, https://doi.org/10.1109/ACCESS.2023.3316063.

[17] Y. Q. Zhang, R. T. Zhang, and B. T. Zhao, "A Systematic Review of Generative Adversarial Imputation Network in Missing Data Imputation," *Neural Computing & Applications* 35, no. 27 (2023): 19685–19705, https://doi.org/10.1007/s00521-023-08836-y.

[18] E. Adeli, J. Zhang, and A. A. Taflanidis, "Convolutional Generative Adversarial Imputation Networks for Spatio-Temporal Missing Data in Storm Surge Simulations," https://doi.org/10.48550/arXiv.2111.02823.

[19] H. C. Jiang, C. F. Wan, K. Yang, Y. L. Ding, and S. T. Xue, "Continuous Missing Data Imputation With Incomplete Dataset by Generative Adversarial Networks-Based Unsupervised Learning for Long-Term Bridge Health Monitoring," *Structural Control and Health Monitoring* 21 (2022): 1093–1109, https://doi.org/10.1002/stc.2777.

[20] J. L. Hou, H. C. Jiang, C. F. Wan, et al., "Deep Learning and Data Augmentation-Based Data Imputation for Structural Health Monitoring System in Multi-Sensor Damaged State," *Measurement* 196 (2022): 111206, https://doi.org/10.1016/j.measurement.2022.111206.

[21] S. Gao, W. L. Zhao, C. F. Wan, H. C. Jiang, Y. L. Ding, and S. T. Xue, "Missing Data Imputation Framework for Bridge Structural Health Monitoring Based on Slim Generative Adversarial Networks," *Measurement* 204 (2022): 112095, https://doi.org/10.1016/j.measurement.2022.112095.

[22] S. Gao, C. F. Wan, Z. W. Zhou, J. L. Hou, L. Y. Xie, and S. T. Xue, "Enhanced Data Imputation Framework for Bridge Health Monitoring Using Wasserstein Generative Adversarial Networks With Gradient Penalty," *Structures* 57 (2023): 105277, https://doi.org/10.1016/j.istruc.2023.105277.

[23] G. Fan, Z. He, and J. Li, "Structural Dynamic Response Reconstruction Using Self-Attention Enhanced Generative Adversarial Networks," *Engineering Structures* 276 (2023): 115334, https://doi.org/10.1016/j.engstruct.2022.115334.

[24] G. Fan, J. Li, H. Hao, and Y. Xin, "Data-Driven Structural Dynamic Response Reconstruction Using Segment-Based Generative Adversarial Networks," *Engineering Structures* 234 (2021): 111970, https://doi.org/10.1016/j.engstruct.2021.111970.

[25] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., "Generative Adversarial Networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems* (Montreal, Canada: MIT Press, 2014), 2672–2680, https://doi.org/10.5555/2969033.2969125.

[26] J. Yoon, J. Jordan, and M. Schaar, "GAIN: Missing Data Imputation Using Generative Adversarial Nets," in *Proceedings of the 35th International Conference on Machine Learning* (Stockholm, Sweden, 2018), 5689–5698, https://doi.org/10.48550/arXiv.1806.02920.

[27] D. T. Neves, J. Alves, M. G. Naik, A. J. Proença, and F. Prasser, "From Missing Data Imputation to Data Generation," *Journal of Computer Science* 61 (2022): 101640, https://doi.org/10.1016/j.jocs.2022.101640.

[28] J. W. Liu, J. N. Yan, L. Z. Wang, et al., "Remote Sensing Time Series Classification Based on Self-Attention Mechanism and Time Sequence Enhancement," *Remote Sensing* 13, no. 9 (2021): 1804, https://doi.org/10.3390/rs13091804.

[29] X. R. Dai, G. P. Liu, and W. S. Hu, "An Online-Learning-Enabled Self-Attention-Based Model for Ultra-Short-Term Wind Power Forecasting," *Energy* 272 (2023): 127173, https://doi.org/10.1016/j.energy.2023.127173.

[30] P. J. Zhao, W. J. Liao, Y. L. Huang, and X. Z. Lu, "Intelligent Design of Shear Wall Layout Based on Attention-Enhanced Generative Adversarial Network," *Engineering Structures* 274 (2022): 115170, https://doi.org/10.1016/j.engstruct.2022.115170.

[31] Y. T. Li, T. F. Bao, Z. X. Gao, et al., "A New Dam Structural Response Estimation Paradigm Powered by Deep Learning and Transfer Learning Techniques," *Structural Health Monitoring* 21 (2022): 770–787, https://doi.org/10.1177/14759217211050231.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), 770–778, https://doi.org/10.1109/CVPR.2016.90.

[33] H. Yin, J. Wan, S. J. Zhang, and Z. Y. Xu, "ADSCN: Adaptive Dense Skip Connection Network for Railway Infrastructure Displacement Monitoring Images Super-Resolution," *Multimedia Tools and Applications* 80, no. 4 (2021): 6105–6120, https://doi.org/10.1007/s11042-020-10009-1.

[34] P. Moser and B. Moaveni, "Design and Deployment of a Continuous Monitoring System for the Dowling Hall Footbridge," *Experimental Techniques* 37 (2013): 15–26, https://doi.org/10.1111/j.1747-1567.2012.00831.x.

[35] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *The 3rd International Conference for Learning Representations* (San Diego, 2015), https://doi.org/10.48550/arXiv.1412.6980.