Explainable Machine Learning Framework for Guided Waves Signal Reconstruction and Structural Health Monitoring Under Varying Operating and Environmental States

YIMING FAN and FOTIS KOPSAFTOPOULOS

ABSTRACT

Lamb Wave (LW) signal has been widely used as a technique for damage identification and localization in Structural Health Monitoring (SHM) due to its sensitivity to varying types of state changes. Analyzing and constructing the guided wave signals then become a critical step in damage detection and assessment. Researchers have discovered features in both temporal and frequency domains for signal description and reconstruction. Yet the related features are challenging to be manually developed as the stochastic acoustic signals captured by sensors can be complex and are determined by various factors such as complex boundary conditions and material properties. Recently, neural network has exhibited the capability for time series reconstruction yet lacking of interpretability. In this study, a convolutional autoencoder (CAE) network has been developed to compress the information of the collected signals along with parameters such as damage levels and external loads into time-invariant and time-variant latent spaces at the bottleneck layer which can be easier to analyze and more efficiently used for state estimation and signal reconstruction. The power of estimating signal and its corresponding conditions has been examined by combining a feed forward neural network (FFNN) with the encoder or decoder extracted from the CAE network so that states of raw signals can be predicted and signals under known states can be reconstructed. The proposed framework has been applied in two test cases to verify its capability and stability in terms of different latent space types. The experiment was conducted on an Al plate under different damage states with PZTs serving as actuators and receivers. It is shown that the state parameters can be estimated with high accuracy and the signals can be generated with low error and thus alleviate the requirements of time-consuming experiments.

INTRODUCTION

Guided-wave signals have been widely used in active-sensing Structural Health Monitoring (SHM) field for damage detection, localization and quantification. Existing approaches include advanced signal processing techniques [1, 2], advanced statistical modeling techniques [3, 4], and analytical models [5]. Yet the stochastic nature of the signals increases the model complexity and computational cost. Moreover, various operating and environmental conditions, including external loads, temperatures, etc., make it more challenging to build accurate models that take multi-factors into consideration while maintaining accuracy and robustness under various conditions.

Researchers then sought for approaches to find damage sensitive features in order to create damage detection schemes that are less computationally intensive and thus makes it possible for online monitoring process. One of the most popular techniques is to develop Damage Indices (DIs) that are sensitive to different damage types and are relatively simpler for computation [6, 7]. Yet the main drawback of such method comes from its deterministic property, meaning only point estimation is available and the

Yiming Fan, Fotis Kopsaftopoulos, Intelligent Structural System Laboratory (ISSL), Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute (RPI), Troy, NY 12180, U.S.A.. Email: fany5@rpi.edu, kopsaf@rpi.edu

uncertainty cannot be calculated. When it comes to the dynamic operating conditions where the uncertainty needs to be considered, implementation of DIs may not meet the accuracy requirements. Even in constant and controlled environments, propagations of damage and vibrations of sensor locations can lead to fault detections. Under this circumstance, Gaussian Process Regression Models (GPRM) have been developed to integrate with time series models to account for the uncertainties [8,9]. Yet these approaches still require domain expertise which can be restricted for online monitoring purpose.

On the other hand, NNs such as Convolutional Neural Network (CNN), Deep Neural Networks (DNN), and Long Short-term Memory Networks (LSTM) have become popular in SHM applications [10, 11]. Among which, CNN is known for its sparse connection property and temporal feature extraction capabilities. With the aid of convolutional layer followed by down sampling layer, CNN can keep the majority information of input tensors while saving computational space. Autoencoder (AE) which is a recently developed network, can compress and then stretch the data for reconstruction purpose. Researchers have studied the combination of these two networks, called Convolutional Autoencoder (CAE) for active sensing SHM. For example, Guo develops a CAE-based health indicator by comparing similarities between learned features for baseline samples and unknown samples [12]. Lee applies fatigue damage detector by checking whether RMSE of the dominant features from the CAE latent space has exceeded the threshold [13]. Yet these techniques can only identify whether the system is under healthy condition or not and the detailed damage level cannot be predicted. In addition, the models are only sensitive to single damage case and cannot take multiple factors into account.

In this work, a complete CAE- and FFNN- based scheme for damage quantification and signal reconstruction scheme is employed. In the train phase, CAE is trained to give accurate reconstructed signals and FFNN is trained to build accurate mappings. In the test phase, the scheme includes two branches: the first one is to estimate multivariate damage states given raw signals while the second one is to provide signals taking only states as inputs. The former branch serves as a damage quantification approach while the later one can simplify the data collection process when gathering data is time-consuming and restricted. The main contribution of this study can be summarized as follows: (i) the construction of a CAE based data-driven modular model consisting of two branches that can compress the signals into a low-dimensional representation; (ii) the accurate simultaneous prediction of damage severity and load in the active-sensing SHM framework; (iii) signal reconstruction with low RSS/SSS given only state information as inputs

EXPERIMENTS AND DATA COLLECTION

The expreiment in this study was completed on a 152.4 × 304.8 mm (6 × 12 in) 6061 Aluminum coupon (2.36 mm/0.093 in thick) where six lead zirconate titanate (PZT) piezoelectric sensors (PZT-5A) were attached onto its surface shown in Figure 1, where the two . The adhesive was cured under vacuum for stable performance for 24 hrs at room temperature. In order to implement load, the plate was installed onto a tensile testing machine (Instron, Inc). Five static loading conditions were applied consecutively: 0, 5, 10, 15 and 20 kN. To simulate different damge level conditions, 1-4 three-gram weights were attached onto the surface of the plate next to each other during each loading state. Under each condition, sensors 1-3 were actuated in a consecutive manner, using 5-peak tone bursts (5-cycle Hamming-filtered sine waves) with 90 V peak-to-peak amplitude. The signals were then collected from sensors 4-6 as receivers. 20 response signals (8 for training and 12 for testing) per structural case were collected at each sensor except for 20kN load where only 2 responses were collected (1 for training and the other one for testing), which gives a total of 3690 signals in the entire data set. Data was collected within 333.33μs with a sampling rate of 24 MHz using a ScanGenie III data acquisition system (Acellent Technologies, Inc). After the data acquisition process, each signal was down sampled from 8000 length to 800 and the sampling frequency was decreased from 24MHz to 2.4MHz to decrease the effect of noise yet the main information was kept.

METHODOLOGY

1. Latent Space Representation

Conventional signal processing methods involve time and frequency domain feature extraction. Though

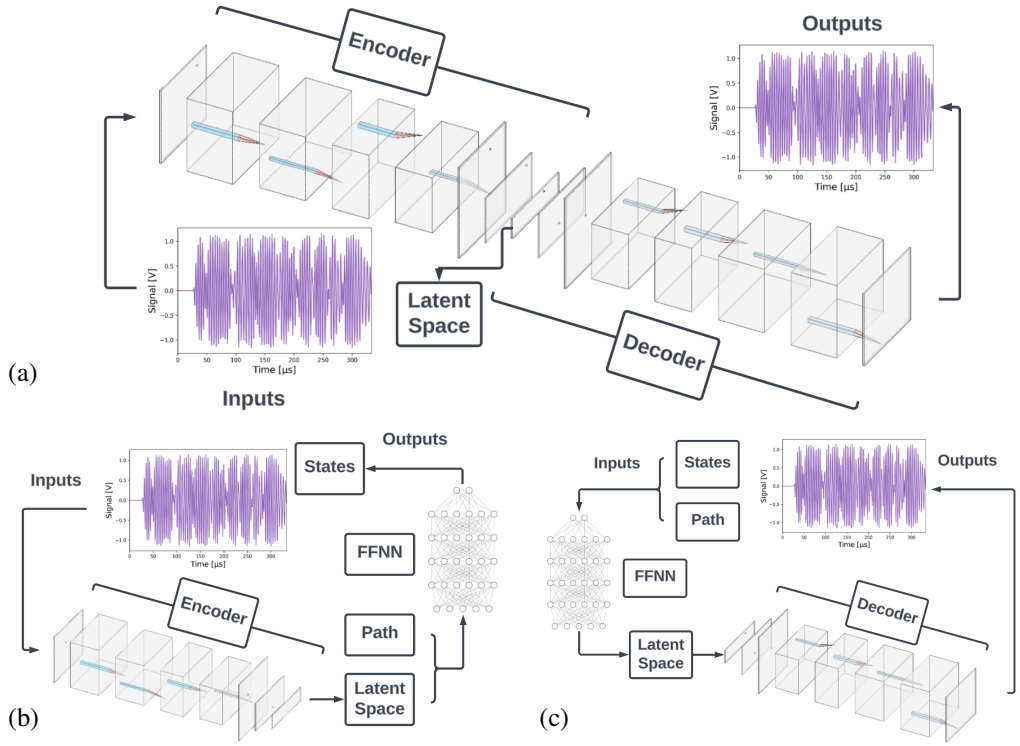

Figure 1. Flowchart of the proposed scheme including the train phase and two online phases: (a) train phase of CAE for signal reconstruction, latent space can be obtained automatically; (b) online phase one - state estimation given signals; (c) online phase two - signal reconstruction given corresponding states.

those features can be sensitive to various types of damages and potentially useful for damage detection, the extraction process always require signal processing related domain knowledge and thus can be restricted and time consuming. In this study, the procedures to process the data before putting them into matrices are relatively simple, including down sampling, demean and eliminating the cross-talk region.

The core step of the methodology is to compress the data into a latent space, where the complex dynamics of the raw stochastic signals can be simplified and thus makes it easier for further analysis. Denote the signal as $\mathbf{y} \in \mathbb{R}^N$, where N is the signal length and state vector as $\mathbf{s}_i$ where $i \in [1, K]$, K is the total number of states. Since there are two state factors, i.e., damage severity and load, in this study, $\mathbf{s}_i = [s_j, s_h]$, where $j \in [1, J]$, $h \in [1, H]$, J, H are the number of possible choices in each state factor.

Figure 1 panel a schematically outlines the procedure of the train phase of CAE along with its interior structure. By minimizing the error of the reconstructed signals, the latent space, denoting $\mathbf{z} \in \mathbb{R}^D$, where D is the latent space dimension, keeps the majority information of the time series. Two different cases have been tested by extracting various types of latent space, namely time-invariant and time-variant spaces. The main difference between these two spaces is that the former one contains the information of signals within a fixed time interval with only a few variables, while the later one has the length same as the signal and thus can reflect the details of the dynamics. Both cases can be meaningful in practical applications.

For each path, the signal matrix can be stacked as,

$$\mathbf{Y}_p = [|\mathbf{y}^1_{\mathbf{s}_1} \dots \mathbf{y}^{tr}_{\mathbf{s}_1}|\mathbf{y}^1_{\mathbf{s}_2} \dots \mathbf{y}^{tr}_{\mathbf{s}_2}| \dots |\mathbf{y}^1_{\mathbf{s}_k} \dots \mathbf{y}^{tr}_{\mathbf{s}_k}|] \tag{1}$$

where the superscript denotes multiple measurements of signals and the subscript denotes the corresponding state vector. Then the training input tensor can be expressed as,

$$\mathbf{Y}_{tot} = [\mathbf{Y}_1 \dots \mathbf{Y}_P] \tag{2}$$

where P is the total path number. In the two test phases, the tensors are stacked differently. For the former one, the matrices are stacked vertically, meaning all the signals are placed in parallel, while for the later

one, the matrices are stacked along the path dimension, making it possible to convolve along the paths. This paper proves their capability of being adopted into two real-world scenarios, which are structure state prediction and signal reconstruction under given states.

2. State Estimation Given Signals

After training the CAE model, it is trivial to use the encoder and the decoder parts separately. This means the data can be compressed using encoder and the latent space can be transformed back to signals using decoder independently. This property leads to the two branches of the test phase, i.e., state estimation and signal reconstruction. In the first branch, also called test phase one in this work, the feedforward neural network (FFNN) that has been trained to connect the latent variables to the state vector, i.e., the mapping of $\mathbf{z} \rightarrow \mathbf{s}$, is integrated as shown in Figure 1 panel b. The structural state of an unknown signal can then be predicted by passing through the encoder and FFNN sequentially. The mathematical expression can be written as:

$$\widehat{\mathbf{s}} = \varphi_1(\mathbf{z}) = \varphi_1(\alpha_{yz}(\mathbf{W}_{yz}\mathbf{y} + \mathbf{b}_{yz})) \tag{3}$$

where φ_1 is the ffnn function of test phase one, α_{yz} is the nonlinear function for the encoder, $\mathbf{W}_{yz}$ and $\mathbf{b}_{yz}$ are weights and bias for encoder, respectively. **3. Signal reconstruction given states**

The second branch as shown in Figure 1 panel c, also named test phase two, works in an opposite direction, i.e., the mapping of $\mathbf{s} \rightarrow \mathbf{z}$. The FFNN herein takes state vector as input and generate latent vector results. Then the decoder is applied to construct the signals. The mathematical expression can be written as:

$$\widehat{\mathbf{y}} = \alpha_{zy}(\mathbf{W}_{zy}\mathbf{z} + \mathbf{b}_{zy}) = \alpha_{zy}(\mathbf{W}_{zy}\varphi_2(\mathbf{s}) + \mathbf{b}_{zy}) \tag{4}$$

where φ_2 is the ffnn function of test phase two, α_{zy} is the nonlinear function for the decoder, $\mathbf{W}_{zy}$ and $\mathbf{b}_{zy}$ are weights and bias for decoder, respectively.

RESULTS

In this work, the proposed scheme has been tested on two test cases. Though the data set is identical in the two cases, the ways to arrange it are different. To be more specific, stacking signals under different paths, damage sizes and loads along with different dimensions as CAE inputs results in different matrix construction methods. The convolution inside the CAE layer is then conducted along various dimensions and finally leads to time-invariant and time-variant latent spaces. To fit the different methods, the CAE structures in the test cases are slightly different and the details are recorded in Table I. It has been observed that both methods can be embedded naturally in the state estimation and signal reconstruction processes and are meaningful in real-world applications. Under this concern, two test cases have been investigated.

Test Case I: Time-invariant Latent Space

In test case 1, all the available signals under different states are stacked vertically. Since the signal length has been down sampled from 8,000 to 800, the tensor size herein is Nx800x1, where N is the

TABLE I. DETAILED INFORMATION OF CAES IN BOTH TEST CASES.

CAE in Test Case I		CAE in Test Case II	
Characteristic	Description	Characteristic	Description
Input Layer	Size: (800,1)	Input Layer	Size: (800,3,3)
Bottleneck Layer	Size: 5	Bottleneck Layer	Size: (800,7)
1st Conv Layer	1D, Filter Number: 64	1st Conv Layer	2D, Filter Number: 128
2nd Conv Layer	1D, Filter Number: 32	2nd Conv Layer	2D, Filter Number: 64
Optimization Method	Adam	Optimization Method	Adam
Epoch	1000	Epoch	5000
Final Loss	0.248	Final Loss	0.0264

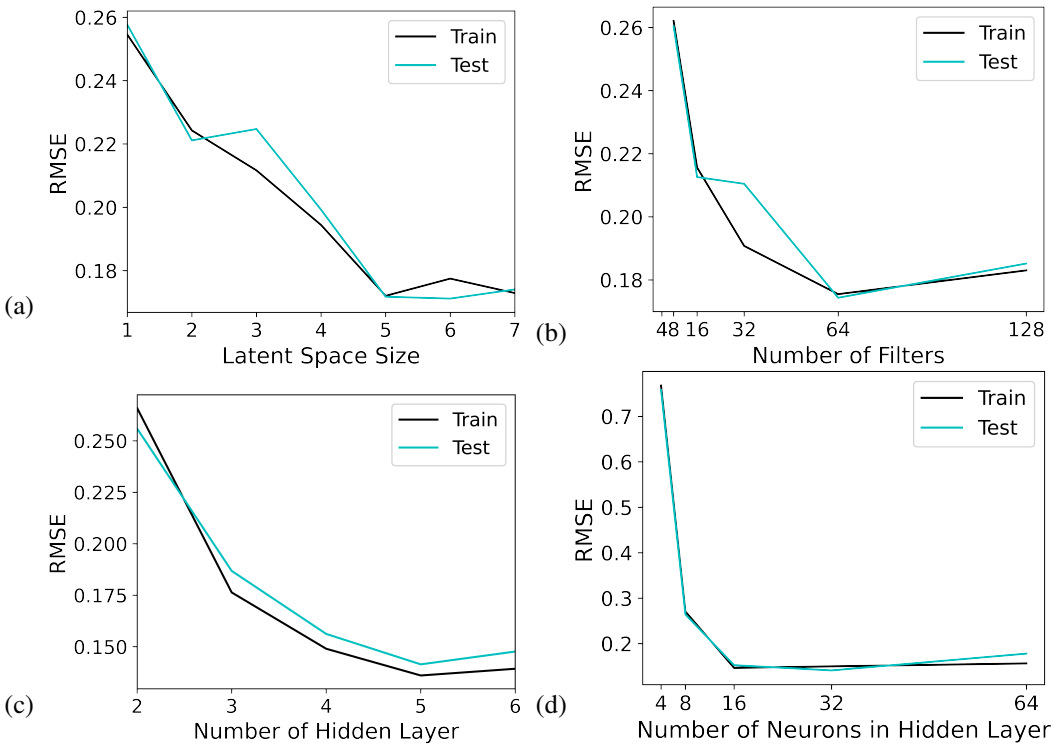

Figure 2. Model configuration: (a) RMSE with respect to CAE latent space size; (b) RMSE with respect to number of filters in the first CAE convolutional layer; (c) RMSE with respect to number of FFNN's hidden layer; (d) RMSE with respect to number of neurons in FFNN's hidden layer.

number of all realizations under all states. While training the CAE model, the MSE loss of the signals is minimized. The latent space from the original data space is compressed automatically. In this test case, each time series can be compressed into a latent space with very limited user defined size. A complete CAE model selection process has been done by recording the loss with different model hyperparameters such as filter size and latent space size. Filter number of the 2nd convolutional layer is set to be half of the 1st one to further compress the data. According to Figure 2, filter number of the first and second convolutional layers have been chosen as 64 and 32, respectively, and the latent space size is set to 5. The epoch is chosen to be 1000, batch size is 32 and the optimization function is Adam. With this setting, the signal reconstruction functionality performs well with low RMSE. The average RSS/SSS per path is recorded in Table II. In most cases, the RSS/SSS is within 2%, which means the reconstructed signals are very similar to the original ones.

The fact that reconstructed time series have negligible error leads to a conclusion that the latent space as the output of the bottleneck layer contains most of the characteristic information of the original signals. An advantage of this process is the latent space can be generated automatically while training the CAE,

TABLE II. SUMMARY OF $RSS/SSS(\%)$ OF THE RECONSTRUCTED SIGNALS FOR EACH PATH FROM CAE TRAIN PHASE AND TEST PHASE TWO, RESPECTIVELY IN TEST CASE 1.

Path	1-4	1-5	1-6	2-4	2-5	2-6	3-4	3-5	3-6
Train Phase RSS/SSS (%)	0.7966	1.4318	0.9442	1.0459	1.1832	0.9028	1.0746	1.0343	1.1480
Test Phase Two RSS/SSS (%)	0.7623	1.6238	0.9899	1.1494	1.1447	1.0584	1.0631	1.1584	1.2508

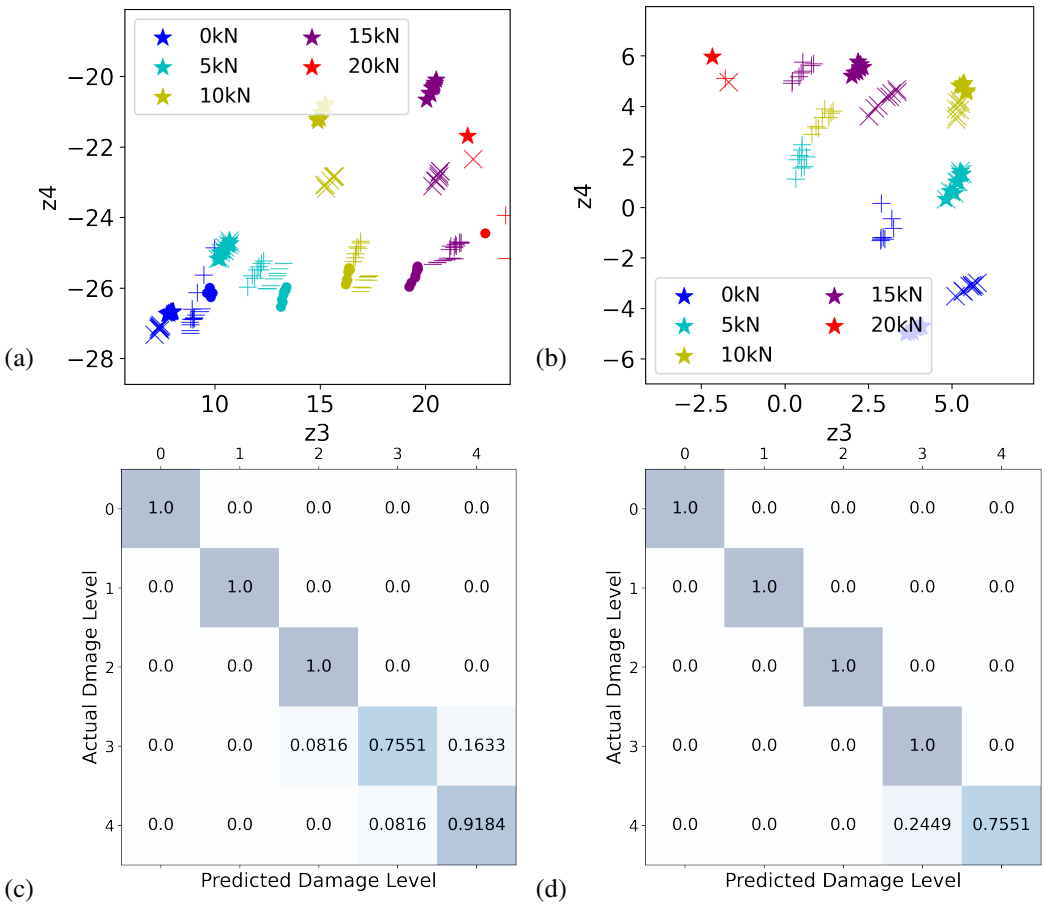

Figure 3. Latent space representation and confusion matrix of state estimation results from test phase one of test case 1: (a) latent space of all states from path 1-4, each color represents for a load condition while each marker shape represents for a damage level ('·': level 0; '−': level 1; '+': level 2; '×': level 3; '⋆': level 4); (b) latent space of all states from path 1-6; (c) confusion matrix of damage level predictions from path 1-4 in test case 1; (d) confusion matrix of damage level predictions from path 1-6 in test case 1.

so signal processing and feature extraction related domain knowledge is not required. The informative property of the latent representations gives the way to state estimation given raw signals and also signal reconstruction given only corresponding states.

In the first test phase, the goal is to predict states given a raw piece of signal. Once the CAE is trained, the encoder part can be used to compress the data. Then the latent space is fed into the trained FFNN which maps to the states. In this paper, 2 types of states, i.e., damage severity level and load, are estimated simultaneously. Figure 3 panels a and b show the latent space pair $(z3, z4)$ under all available states from path 1-4 and 1-6 with all test data, respectively. Different colors represent for different load conditions while different marker shapes represent for different damage levels. It can be observed from both plots that under most states, these representations have their own clusters, meaning the differences of signals under various states can be reflected by the latent variable values. Panels c and d show the confusion matrix of state estimation during the test phase one of test case I. The fault detection occurs when actual damage level is at 3 or 4. When looking into the details, it was observed that the majority deviations happen when the corresponding loads are at $5kN$. This observation can be explained though the overlap of markers of damage level 3 and 4 represented by '×' and '⋆', respectively, with cyan color in panel a. A similar situation is observed when comparing panels b and d where the overlap of the two markers at $5kN$ still occurs. The load estimation, on the other hand, gives 100% accuracy, which can be verified that the markers in panels a and b with different colors have their own well-separated clusters.

In the second test phase, latent space variables are estimated though the second FFNN given states as inputs. Then the results are passed into decoder to get reconstructed signals. Figure 4 panels a and b

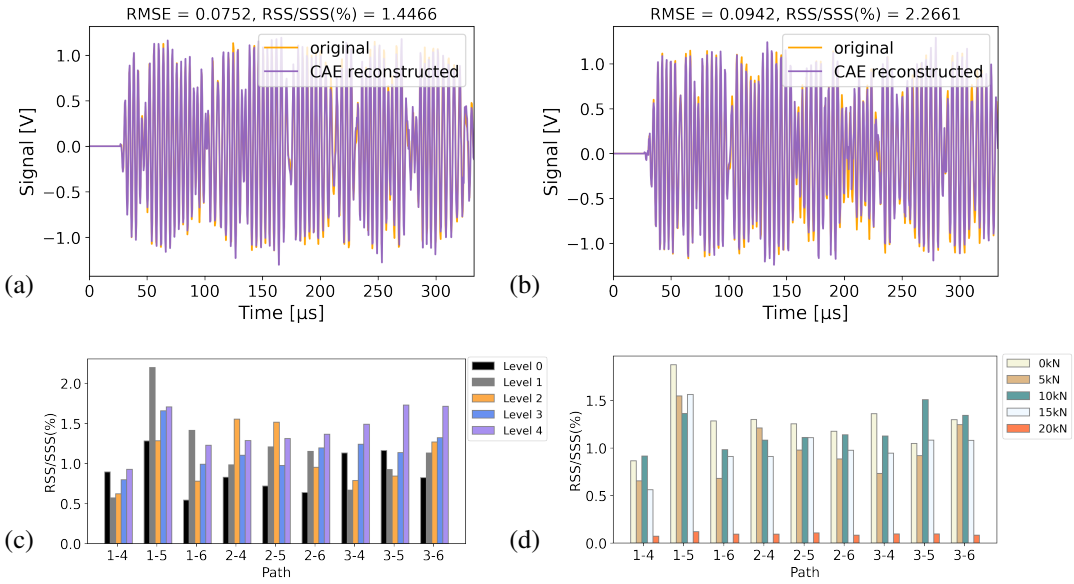

Figure 4. Results of test phase two in test case I: (a) comparison of reconstructed and original signals under damage level 0, load $0kN$ from path 1-4; (b) comparison of reconstructed and original signals under damage level 4, load $20kN$ from path 1-6; (c) average $RSS/SSS(\%)$ w.r.t each damage level per path from test case I; (b) average $RSS/SSS(\%)$ w.r.t each load per path from test case I.

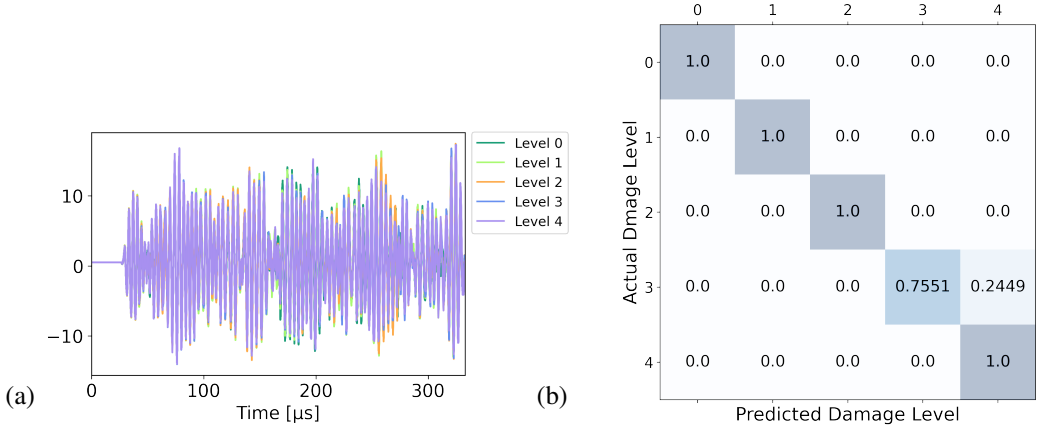

Figure 5. (a) Time-variant latent space under different damage conditions extracted from test case II; (b) Confusion matrix of damage level predictions from test phase one in test case II.

show two examples of reconstructed and original signals. The first one is under damage level 0, load $0kN$ from path 1-4 while the second is under damage level 4, load $20kN$ from path 1-6. Again, low RMSE and RSS/SSS are achieved. The overall average signal reconstruction results for each path are shown in panels c and d, with respect to damage level and load, respectively. From which, RSS/SSS$(\%)$ remains at low values, validating the performance of this test phase.

Test Case II: Time-variant Latent Space

In the second test case, time series from the same path are first stacked together. Then each matrix is stacked along the third and fourth dimension. For all the damage sizes and loads, there are 410 realizations from each path, leading to a tensor with dimension of $410\times800\times3\times3$, where the last two dimensions are derived from the 9 paths in this work. Note the second dimension is with respect to time in this case, which is the precondition of the latent space being time-variant.

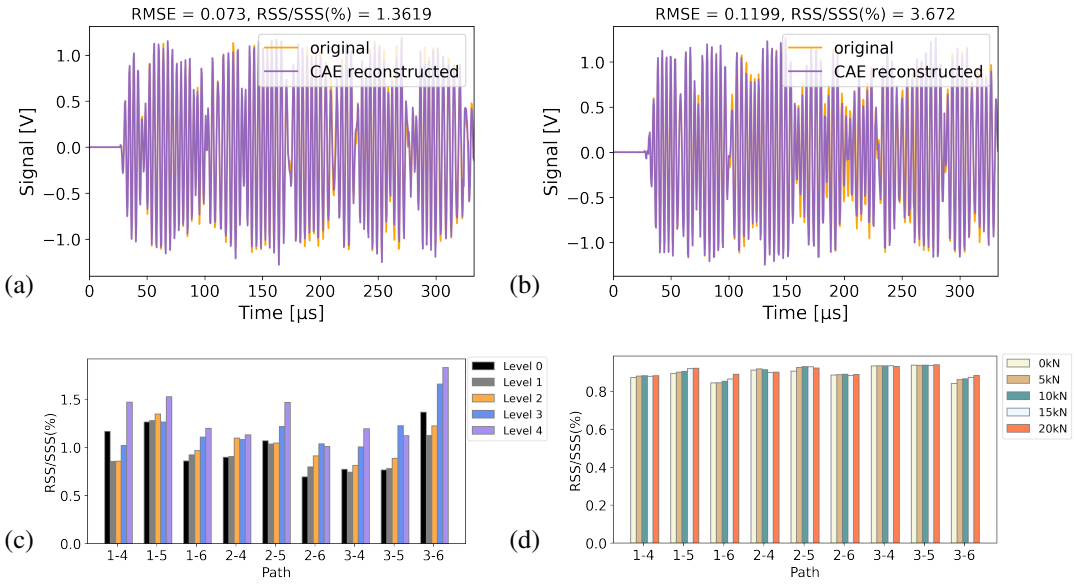

Figure 6. Results of test phase two in test case II: (a) comparison of reconstructed and original signals under damage level 0, load $0kN$ from path 1-4; (b) comparison of reconstructed and original signals under damage level 4, load $20kN$ from path 1-6; (c) average $RSS/SSS(\%)$ w.r.t each damage level per path from test case II; (b) average $RSS/SSS(\%)$ w.r.t each load per path from test case II.

Similar to test case I, the model configuration is first optimized based on the validation loss. The CAE model still has two convolutional layers in both encoder and decoder yet modifications need to be done to fit the new data set. For the best performance, the latent space size is chosen to be 7 and filter sizes are 128 and 64 in the first and second convolutional layer, respectively. In this work, the total path number is limited and the computational cost is not decreased dramatically. However, when dealing with more paths such as in numerical models using this scheme can convolve paths significantly and thus improve the efficiency.

The latent space representations under all damage levels are shown in Figure 5 panel a. It can be observed that the signals after convolution and compression is time-variant. Unlike test case I, the latent expressions herein can not be interpreted intuitively. Yet the damage level estimation results of path 1-4 from panel b still exhibit a relatively high accuracy. In test phase two, the FFNN is modified in a similar manner and the overall accuracy of the reconstructed signals is recorded in Table III. Again, the signals again can be reconstructed with acceptable accuracy.

CONCLUSION

In this study, a CAE and FFNN based active-sensing SHM scheme has been proposed for damage

TABLE III. SUMMARY OF $RSS/SSS(\%)$ OF THE RECONSTRUCTED SIGNALS FOR EACH PATH FROM CAE TRAIN PHASE AND ONLINE PHASE TWO, RESPECTIVELY IN TEST CASE 2.

Path	1-4	1-5	1-6	2-4	2-5	2-6	3-4	3-5	3-6
Train Phase RSS/SSS (%)	1.1239	1.0436	0.7663	0.8338	0.9492	0.6535	0.6577	0.6992	1.2882
Online Phase Two RSS/SSS (%)	1.0730	1.3348	1.0103	1.0221	1.1665	0.8892	0.9045	0.9558	1.4392

quantification and signal reconstruction tasks under multi-variant states. Time series collected by PZTs attached on an Al plate, after simple processing procedures, are fed directly into the CAE to obtain reconstructed signals. Time-variant and time-invariant latent space can be achieved by constructing input tensors along different directions. The latent space is then connected with multi-variable state vectors using FFNN. Then state can be estimated given raw signals and signals can be generated given only state information. In both test cases, the damage quantification accuracy is satisfied and the signals can be reconstructed with low RMSE and RSS/SSS values.

REFERENCES

1. Patra, S. and S. Banerjee. 2017. "Material State Awareness for Composites Part I: Precursor Damage Analysis Using Ultrasonic Guided Coda Wave Interferometry (CWI)," *Materials*, 10:1436.
2. Amjad, U., S. K. Yadav, and T. Kundu. 2016. "Detection and quantification of delamination in laminated plates from the phase of appropriate guided wave modes," *Optical Engineering*, 55(1):011006.
3. Peng, T., A. Saxena, K. Goebel, Y. Xiang, S. Sankarararman, and Y. Liu. 2013. "A novel Bayesian imaging method for probabilistic delamination detection of composite materials," *Smart Materials and Structures*, 22:125019–125028.
4. Yang, J., J. He, X. Guan, D. Wang, H. Chen, W. Zhang, and Y. Liu. 2016. "A probabilistic crack size quantification method using in-situ Lamb wave test and Bayesian updating," *Mechanical Systems and Signal Processing*, 78:118–133.
5. Borkowski, L. and A. Chattopadhyay. 2014. "Electromagnetomechanical elastodynamic model for Lamb wave damage quantification in composites," in *Proc. SPIE 9064, Health Monitoring of Structural and Biological Systems*, San Diego, CA, USA.
6. Soman, R., P. Malinowski, and W. Ostachowicz. 2018. "Comparative study of deterioration of composite due to moisture using strain, electro-mechanical impedence, and guided waves," in *Proc. SPIE 10600, Health Monitoring of Structural and Biological Systems XII*, Denver, CO, USA.
7. Yan, J., H. Jin, H. Sun, and X. Qing. 2019. "Active monitoring of fatigue crack in the weld zone of Bogie frames using ultrasonic guided waves," *Sensors*, 19:3372.
8. Amer, A. and F. Kopsaftopoulos. 2023. "Gaussian process regression for active sensing probabilistic structural health monitoring: experimental assessment across multiple damage and loading scenarios," *Structural Health Monitoring*, 22(2):1105–1139.
9. Fan, Y. and F. Kopsaftopoulos. 2022. "Damage State Estimation via Multi-fidelity Gaussian Process Regression Models for Active-Sensing Structure Health Monitoring," in *European Workshop on Structural Health Monitoring: EWSHM 2022-Volume 2*, Springer, pp. 267–276.
10. Drakoulas, G., T. Gortsas, C. Kokkinos, F. Kopsaftopoulos, and D. Polyzos. 2022. "A Machine Learning Framework for Reduced Order Modeling of Guided Waves Propagation," in *13th International Congress on Mechanics HSTAM*.
11. Zhang, S., C. M. Li, and W. Ye. 2021. "Damage localization in plate-like structures using time-varying feature and one-dimensional convolutional neural network," *Mechanical Systems and Signal Processing*, 147:107107.
12. Guo, L., Y. Yu, A. Duan, H. Gao, and J. Zhang. 2022. "An unsupervised feature learning based health indicator construction method for performance assessment of machines," *Mechanical Systems and Signal Processing*, 167:108573.
13. Lee, H., H. J. Lim, T. Skinner, A. Chattopadhyay, and A. Hall. 2022. "Automated fatigue damage detection and classification technique for composite structures using Lamb waves and deep autoencoder," *Mechanical Systems and Signal Processing*, 163:108148.