

# **Damage identification of spatial structures based on machine learning techniques**

**by Jiajia Hao**

Thesis submitted in fulfilment of the requirements for  
the degree of

**Doctor of Philosophy**

under the supervision of A/Prof. Xinqun Zhu and Prof.  
Jianchun Li

University of Technology Sydney  
Faculty of Engineering and Information Technology

September 2024

## **CERTIFICATE OF ORIGINAL AUTHORSHIP**

I, Jiajia Hao declare that this thesis is submitted in fulfillment of the requirements for the award of Doctor of Philosophy, in the School of Civil and Environmental Engineering, Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Production Note:  
Signature      Signature removed prior to publication.

Date: September 17, 2024

## ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my principal supervisor, Prof. Xinqun Zhu. He is knowledgeable, patient, trustworthy, and always supportive, inviting questions in an encouraging manner. He always listens carefully to everyone's presentation and provides thoughtful recommendations. He values my work, respects my time, and appreciates my progress even when there are mistakes. As I am heading to the industry soon, I look forward to potential future collaborations with him. Upon graduation, I would like to give my best wishes to him, my supervisor, my friend, and my role model.

I would also like to express my thanks to my co-supervisor Prof Jianchun Li. He is knowledgeable and energetic, and he has a lot of passion for teaching and research. He consistently provides inspiring ideas for me to pursue. Talking to him about both research and life has always been incredibly beneficial. Many thanks to Dr. Yang Yu for his generosity in sharing his knowledge and experiences with me. The support and trust from Prof. Chunwei Zhang and Dr. Hadi Kordestani have also been invaluable to me.

I also had a great time with my friends Bing Zhang, Xutong Zhang, Chenguang Li, Jiahao Xia, and Saeid Talaei as well as other friends in the research group. We have supported each other like brothers and sisters. Staying with all of you for dinner at Xinqun's home has always been one of the best moments for me.

A big thanks to my mentors Dr. Carmen Wong and Prof. Stuart Perry, and my friends Dimity Wehr and Stuart Allabush. I enjoyed hearing Carmen's big laugh, Stuart talking

about food, Dim sharing about her recent journey, and Stuart's encouragement and support. It has been so good to have these nice friends around.

Lastly, my sincere gratitude to my parents, and my sisters. They are the ones that are always by my side. If there is always a place to go, that would be my home. I could not have reached this point without their support. This thesis is dedicated to them.

## LIST OF PUBLICATIONS

1. **Hao, J., Zhu, X., Yu, Y., Zhang, C., & Li, J.** “Damage localization and quantification of a truss bridge using PCA and convolutional neural network”, *Smart Structures and Systems*, 30(6), 673-686,2022.
2. **Hao, J., Zhu, X., & Li, J.** Structural damage detection for spatial frame structures with semirigid joints using wireless measurements. (under revision by the *Journal of Vibration and Control*).
3. **Hao, J., Zhu, X., & Li, J.** Long short-term memory-based variational autoencoder (LSTM-VAE) based semi-supervised learning for damage identification using a multi-sensor system (to be submitted).
4. **Hao, J., Zhu, X., & Li, J.** CWT-based damage quantification of spatial structures using convolutional VAE (under preparation).

# TABLE OF CONTENTS

CERTIFICATE OF ORIGINAL AUTHORSHIP.....	i
ACKNOWLEDGEMENTS .....	ii
LIST OF PUBLICATIONS .....	iv
TABLE OF CONTENTS .....	v
LIST OF FIGURES.....	ix
LIST OF TABLES.....	xiii
ABSTRACT.....	xiv
CHAPTER 1 Introduction.....	1
1.1 Background .....	1
1.2 Problem definition.....	4
1.3 Research objectives .....	4
1.4 Contribution of this research .....	5
1.5 Organisation of this thesis .....	6
CHAPTER 2 Literature review .....	8
2.1 Overview .....	8
2.2 The Vibration-based damage detection (VBDD).....	8
2.2.1 Model-based methods.....	9
2.2.2 Data-driven methods.....	11
2.3 Machine learning based structural damage detection .....	14
2.3.1 Artificial neural networks (ANNs) .....	14
2.3.2 Support vector machine (SVMs) .....	16
2.3.3 Naïve bayes classifier.....	19
2.3.4 Decision trees (DTs) .....	19
2.3.5 Principal component analysis (PCA) .....	21
2.4. Deep learning for structural damage detection .....	23
2.4.1 Structural damage detection based on convolutional neural networks .....	24
2.4.2 Structural damage detection based on Autoencoders .....	29
2.4.3 Structural damage detection based on Recurrent Neural Networks (RNNs).....	33
2.5 Summary .....	36

CHAPTER 3 Structural damage detection for spatial frame structures with semirigid joints using wireless measurements .....	38
3.1 Overview .....	38
3.2 Generic element for a nonuniform beam with semi-rigid joints .....	41
3.2.1 Stiffness matrix of the nonuniform beam with rigid joints .....	41
3.2.2 Stiffness matrix of a uniform beam with semi-rigid joints .....	45
3.2.3 Mmodal strain energy-based damage index .....	48
3.3 Experimental setup and modal analysis .....	50
3.3.1 The bridge model .....	50
3.3.2 Experimental setup.....	51
3.3.3 Ref-SSI based spatial mode shapes extraction using multiple set measurements.....	53
3.4. Numerical simulation and model updating .....	61
3.4.1 Numerical model.....	61
3.4.2 Comparisons between numerical and experimental results .....	61
3.4.3 Model updating .....	63
3.5. Structural damage detection using the updated model.....	67
3.5.1 Scenarios description .....	67
3.5.2 Structural damage detection using the proposed model .....	68
3.5.3. Joint damage detection performance (semi rigid joint model) .....	70
3.6 Summary .....	72
CHAPTER 4 Structural damage detection using PCA and convolutional neural network .....	73
4.1 Overview .....	73
4.2 Methodology .....	73
4.2.1 Convolutional neural network (CNN) architecture for damage detection .....	73
4.2.2 Loss and damage evaluation indices .....	76
4.2.3 Mode shape based and modal strain energy based damage index .....	80
4.2.4 Effects by element and mode based on principal component analysis (PCA) .....	82
4.2.5 Flowchart of this chapter.....	84
4.3. Numerical simulations.....	85

4.3.1 Numerical modelling.....	85
4.3.2 Implementation details of CNN.....	86
4.3.3 Data augmentation.....	88
4.3.4 Modal strain energy based PCA.....	88
4.4. Results and discussions.....	90
4.4.1 Architecture selection.....	90
4.4.2 Effects of the loss function.....	91
4.4.3 Feature visualization.....	93
4.4.4 Effects of data augmentation.....	95
4.4.5 The <i>NMSEC-PCA</i> based CNN.....	97
4.4.6 Effects by element and mode.....	100
4.5. Summary.....	101
CHAPTER 5 Enhancing Bridge Health Monitoring with LSTM-VAE: A Semi-Supervised Learning Framework for Damage Detection.....	103
5.1 Introduction.....	103
5.2 Methodology.....	105
5.2.1 LSTM-VAE networks for enhancing structural health monitoring....	105
5.2.2 Reconstructed Error-Based Indices.....	110
5.2.3 Semi-Supervised Learning Framework.....	112
5.3. Experimental model and modal analysis.....	114
5.3.1 Experimental model and setup.....	114
5.3.2 Sensor arrangement.....	115
5.3.3 Modal analysis.....	115
5.4 Numerical study.....	116
5.4.1 The numerical bridge and model updating.....	116
5.4.2 Data compression using the LSTM-VAE for damage quantification.....	117
5.4.3 Semi-supervised learning for damage classification.....	125
5.5 Experimental study.....	128
5.5.1 Experimental study.....	128
5.5.2 Experimental verification.....	129
5.6 Summary.....	136
CHAPTER 6 CWT based Damage quantification of bridges using convolutional VAE.....	137



6.1 Overview .....	137
6.2 Methodology .....	137
6.3 Numerical study .....	139
6.3.1 Numerical model and data preparation .....	139
6.3.2 Convolutional VAE network .....	143
6.3.3 Numerical results discussion .....	145
6.4 Experimental study.....	149
6.4.1 The experimental data collection .....	149
6.4.2 Experimental verification .....	153
6.5 Summary .....	157
Chapter 7 Conclusions and recommendations .....	159
7.1 Conclusions .....	159
7.2 Recommendations for future work.....	160
References .....	164

## LIST OF FIGURES

Figure 1.1 Flowchart of structural health monitoring .....	2
Figure 3.1 Uniform and nonuniform beams .....	42
Figure 3.2 3D beam element with semi-rigid joints.....	45
Figure 3.3 Flowchart of this chapter .....	50
Figure 3.4 The 7.98m experimental truss bridge .....	51
Figure 3.5 Experimental setup .....	52
Figure 3.6 Acceleration response of accelerometer No.4 in horizontal direction .....	52
Figure 3.7 Accelerometer locations of four set measurements .....	53
Figure 3.8 Sets modeling in ARTeMIS .....	54
Figure 3.9 The stabilization diagram of four sets.....	57
Figure 3.10 Spatial mode shapes of numerical and experimental models .....	60
Figure 3.11 Finite element model of the bridge .....	61
Figure 3.12. Elastic modulus effect on frequency based on three objective functions ...	64
Figure 3.13 The real bar element and its simulation .....	67
Figure 3.14 Element number .....	68
Figure 3.15 elemental <i>NMSEC</i> of diagonal bar damage .....	69
Figure 3.16 Elemental <i>NMSEC</i> of vertical bar damage .....	70
Figure 3.17 elemental <i>NMSEC</i> of double bar damage scenarios .....	70
Figure 3.18 <i>NMSEC</i> of double joint damage scenario .....	71
Figure 4.1 The example target matrix .....	78
Figure 4.2 Flowchart of the investigations in this chapter.....	84
Figure 4.3 Finite element model of the truss bridge .....	86
Figure 4.4 The data preparation and the architecture of the CNN .....	86
Figure 4.5 Explained variance by components .....	89
Figure 4.6 Validation loss of different architectures .....	90
Figure 4.7 Comparisons of validation curves relating to different weighting factor combination .....	91

Figure 4.8 CNN feature visualization with t-SNE: .....	94
Figure 4.9 Comparisons of the CNN with and without data augmentation .....	96
Figure 4.10 Prediction results of selective samples using CNN1 and CNN2.....	97
Figure 4.11 <i>NMSEC</i> -PCA based CNN feature visualization with t-SNE: .....	98
Figure 4.12 Validation curve comparisons between mode shapes based and <i>NMSEC</i> - PCA based index .....	99
Figure 4.13 Prediction results of selective samples using mode shape based CNN and <i>NMSEC</i> -PCA based CNN .....	99
Figure 4.14 The effect by bar element .....	100
Figure 4.15 The effect by mode .....	100
Figure 5.1 The flowchart of this chapter .....	107
Figure 5.2 Variational autoencoder (VAE)(“Variational Autoencoder,” 2024).....	108
Figure 5.3 The flowchart of LSTM-VAE architecture .....	110
Figure 5.4 Sensor arrangement.....	115
Figure 5.5 Acceleration of Sensor 4 in the vertical direction.....	115
Figure 5.6 The numerical bridge model .....	116
Figure 5.7 Excitation location .....	117
Figure 5.8 The force excitation .....	118
Figure 5.9 The vertical signal from Sensor 1 of the healthy scenario .....	118
Figure 5.10 Training process .....	120
Figure 5.11 Testing ARE .....	122
Figure 5.12 Statistical ARE plot of the testing results .....	122
Figure 5.13 The histogram of the ARE .....	123
Figure 5.14 The comparison of the original and reconstructed signals .....	123
Figure 5.15 FFT Comparison of the original numerical signal and reconstructed signal (Sensor 3) .....	123
Figure 5.16 t-SNE features .....	125
Figure 5.17 Confusion Matrix using SVM for classification.....	126
Figure 5.18 Comparison of the original signal and 30% noise signal.....	127

Figure 5.19 Testing Confusion matrix using noisy signals .....	128
Figure 5.20 bar replacements and dimensions .....	129
Figure 5.21 Training curve .....	131
Figure 5.22 Testing ARE using the experimental signals from all scenarios.....	132
Figure 5.23 Statistical ARE bar plot.....	132
Figure 5.24 The reconstructed sample and original signal (Sample 20 from Scenario 1) .....	133
Figure 5.25 The reconstructed sample and original signal (sample 500 from Scenario 7) .....	133
Figure 5.26 FFT comparison .....	133
Figure 5.27 Confusion Matrix with 2d t-SNE feature.....	135
Figure 5.28 Confusion Matrix with 3d t-SNE feature.....	135
Figure 5.29 Confusion Matrix with 3d tSNE and ARE feature .....	136
Figure 6.1 The Gaussian white noise force excitation of 100 seconds .....	140
Figure 6.2 Recorded acceleration locations .....	140
Figure 6.3 The vertical acceleration response from the middle span location.....	140
Figure 6.4 CWT images across different sensors (random sample 10).....	143
Figure 6.5 The training process of VAE.....	146
Figure 6.6 Damage quantification using Average Reconstructed Error(ARE) .....	146
Figure 6.7 Box plot of SARE.....	147
Figure 6.8 The testing average reconstructed error of VAE distribution .....	147
Figure 6.9 Comparison of the original and reconstructed CWT images (Sensor 3) .....	148
Figure 6.10 Generating new CWT images using VAE.....	148
Figure 6.11 Experimental acceleration signal from healthy scenario .....	150
Figure 6.12 CWT image across different sensors (random sample 10) of the experimental bridge .....	152
Figure 6.13VAE training process .....	153
Figure 6.14 Damage quantification using Average reconstructed error(ARE) .....	155
Figure 6.15 Box plot of SARE of VAE.....	155

Figure 6.16 The testing error distribution of VAE .....	156
Figure 6.17 Comparison of the original experimental CWT image and reconstructed image .....	157
Figure 6.18 Generating experimental CWT images using VAE .....	157

## LIST OF TABLES

Table 3.1 Natural frequencies of four sets ( $Hz$ ) .....	55
Table 3.2 Natural frequencies of the experimental truss bridge.....	55
Table 3.3 MAC matrix of experimental mode shapes.....	55
Table 3.4 Details of the model and the cross section of beam4 element.....	61
Table 3.5 Natural frequency and mode shapes comparison between experimental and numerical model.....	62
Table 3.6 MAC matrix of numerical mode shapes.....	62
Table 3.7 MAC matrix of experimental and numerical mode shapes .....	63
Table 3.8 The mode matching of the experimental and numerical models before and after updating element stiffness.....	65
Table 3.9 The mode matching of the experimental and numerical models after updating member and joint stiffness .....	66
Table 3.10 Equivalent elastic modulus of bars in three directions .....	67
Table 3.11 Scenarios design .....	68
Table 4.1 Scenarios for damage detection.....	85
Table 4.2 Mode shape based CNN architecture .....	88
Table 4.3 <i>NMSEC-PCA</i> based CNN architecture.....	89
Table 4.4 Comparisons of architectures .....	91
Table 4.5 CNN testing results evaluation of the updated MSE.....	92
Table 4.6 CNN performance evaluation using noise-free and 5% noise testing samples .....	96
Table 4.7 Comparisons of the two methods .....	99
Table 5.1 The scenario design .....	119
Table 5.2 Validation and test accuracy of semi-supervised methods .....	126
Table 6.1 The convolutional VAE Architecture .....	144

## **ABSTRACT**

Spatial structures are widely used in large buildings and bridges. These structures are usually composed of thousands of members and joints. The damage of members and joints may result in changes in the loading path, reducing the load-bearing capacity and even progressive collapse of the entire structure. This study deals with the critical challenge of structural health monitoring (SHM) for large spatial bridge structures. With machine learning techniques, this work introduces a comprehensive approach including supervised, semi-supervised, and unsupervised methods to refine and enhance structural damage detection capabilities.

Firstly, a novel generic element for nonuniform beams with semi-rigid joints is developed, addressing a main error that caused by assuming semi-rigid joints as rigid and nonuniform members as uniform elements. The proposed approach is proven to improve the accuracy of numerical modelling and damage detection for spatial structures. Moreover, an updated Mean Square Error (UMSE) loss function is proposed, specifically designed for supervised deep-learning-based structural damage detection problems. It is proved to achieve significant high convergence speed over the traditional MSE. Additionally, the *NMSEC-PCA* (normalized modal strain energy change index processed by PCA) index is proposed as damage feature, resulting significant low computational costs while achieving faster convergence and high detection accuracy. Furthermore, to enable the structural health monitoring in practice, a semi-supervised learning frame is proposed for structural damage detection, together with indices used for damage quantification. The

efficacy and applicability of the proposed method are validated using the experimental bridge model, showing their practicality for real-world scenarios. Finally, Continuous Wavelet Transform (CWT) based damage quantification of bridges using a convolutional Variational Autoencoder (VAE) is implemented, demonstrating the VAE's effectiveness in accurately reconstructing images and damage quantification, bridging the gap between theoretical models and real-world structural health monitoring applications.

Overall, the thesis presents comprehensive research using supervised, semi supervised, and unsupervised methods for structural damage detection, all illustrated using numerical and experimental bridge models. The study provides machine learning based practical solutions for structural damage detection that are essential for maintaining the safety and integrity of large-scale spatial structures.

**Keywords:** Structural health monitoring; Machine learning; Deep learning; Damage detection; Spatial frame structures with semi-rigid joints; Modal updating; Variational autoencoder; Long short-term memory (LSTM) network; Unsupervised learning; Semi-supervised learning; Multiple set wireless measurements.



# CHAPTER 1 INTRODUCTION

## 1.1 BACKGROUND

During the long service period, bridge infrastructures continuously suffer from both environmental influences such as eroding effects from the ground or the sea, and operational effects including the traffic (e.g., traffic-induced vibrations and traffic mass effects), temperature, and wind. Because of these effects, structural integrity is threatened all the time. Thus, it is of vital importance to monitor the health conditions of those major bridges, especially those newly constructed and aging bridges. Moreover, urgent damage detection tasks are often needed for those bridges that suffer from large-scale natural disasters like earthquakes and hurricanes. Huge losses of human life and property would have been avoided if effective monitoring systems had been set up before the disasters.

In the past decades, as structural safety became a top priority, structural health monitoring (SHM) has been gaining increasing attention from both engineers and researchers. SHM is a process of observing and evaluating structures based on measurements from a sensing system. A typical SHM system can be seen in Figure 1.1, in which the key task is structural damage detection. The method of SHM uses signals from a sensor system to indicate the structural internal properties. Then structural change could be indicated by signal changes. Through extraction, processing, and analysis of the changed signals, the damage could be identified, and corresponding maintenance and reinforcement measures could be implemented to ensure structural safety.

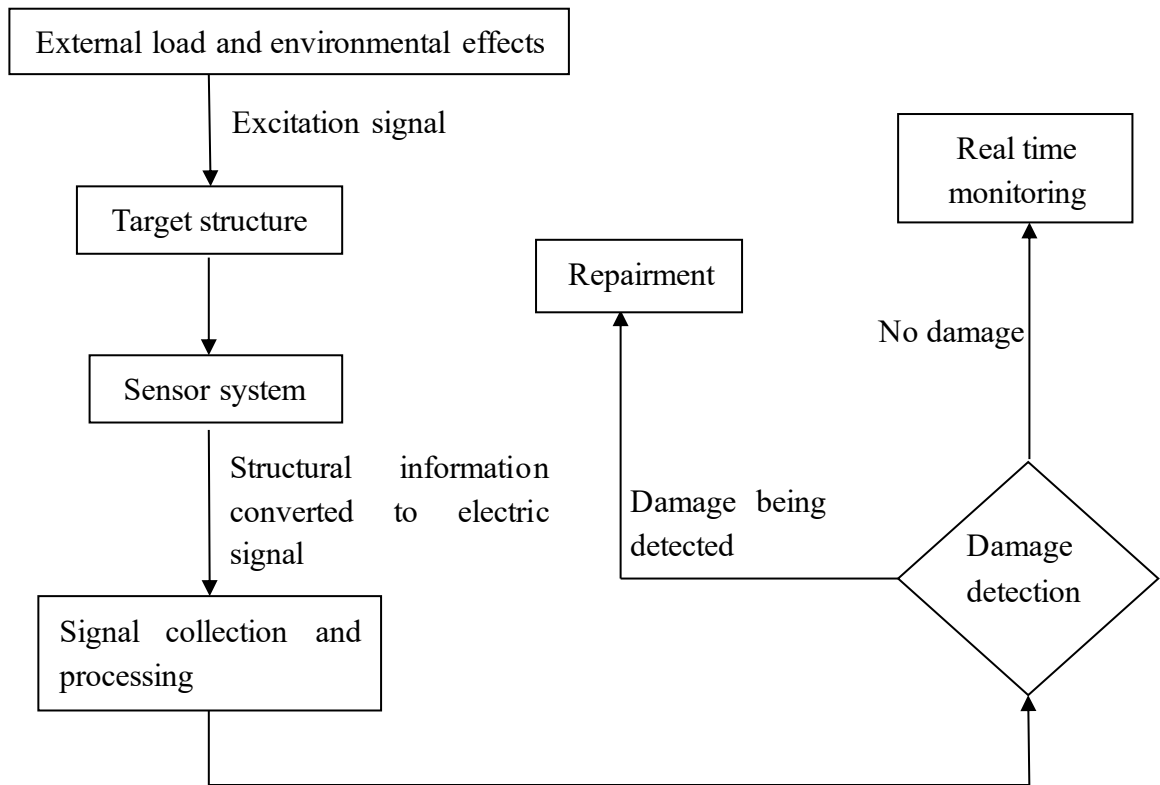


Figure 1.1 Flowchart of structural health monitoring

Damage detection has always been one main research interest in the SHM domain. Traditional damage detection methods include vision-based methods and non-destructive testing (NDT) methods such as ultrasonic, acoustic, electromagnetic, piezoelectric active sensing methods, and the electro-mechanical impedance method. The vibration-based damage detection (VBDD), aiming at detecting the existence, location, and severity of structural damage, has attracted massive research interest in detecting the damage to civil, mechanical, and aircraft structures. The basic idea of VBDD is that structural damage induces changes in mechanical properties such as stiffness and mass. These mechanical property changes could be further indicated by the variations of vibration properties (e.g., natural frequencies and mode shapes). By analysing the damage features extracted from the vibration data, structural damage can be detected, localized, and quantified.

The VBDD can be classified into the model-based and data-driven methods. The model-based methods aim to establish a computational model, e.g. finite element model, by updating model parameters with real measurements (Sun et al., 2020a). Despite the high accuracy they often achieve, they require prior knowledge and associated assumptions about the real structure which are sometimes not accessible or accurate enough. The uncertainty introduced by factors such as measurement noise, modelling errors, and environmental variations can significantly impact the reliability and accuracy of these models. Unlike the model-based method, the data-driven methods employ statistical models to directly interpret the vibration data into structural patterns without any prior knowledge.

Recently, advances in the Artificial Intelligence (AI) technique revolutionized the data-driven methods by allowing large data sets processing with high computation speed. This improvement has made the data-driven method one of the most attractive and promising approaches in the SHM domain. During the last two decades, research on machine learning-based methods has been extensively conducted to develop various damage detection methods. Machine learning modelling is a process of mapping the monitoring data into different structural patterns. Depending on the data availability of vibration data in damaged conditions, they could be divided into supervised and unsupervised learning. To be more specific, supervised learning requires labelled training data to indicate the real structural conditions while unsupervised ones do not. Nevertheless, both have been proven to be effective in structural damage detection.

This project aims to develop machine learning based methods for damage detection of spatial structures. The machine learning techniques including supervised, semi-supervised and unsupervised methods are explored and developed for the damage detection of spatial bridge structures.

## **1.2 PROBLEM DEFINITION**

Current advancements in machine learning are mainly developed for computer vision, with substantial successes in image recognition and analysis. Although the potential of these techniques could serve as advanced tools for Structural Health Monitoring (SHM), the challenge lies in adapting these methods to address the unique requirements of SHM, where the focus shifts from visual interpretation to the detection of subtle patterns for structural damage. This research aims to bridge this gap by extending machine learning's research into SHM, ensuring structures' integrity and safety with innovative, data-driven approaches.

## **1.3 RESEARCH OBJECTIVES**

This research project aims to develop damage detection methods for spatial structures using machine learning techniques, bridging the gap between research and practical application in SHM. The detailed objectives are listed as follows.

- 1) Developing a generic element for nonuniform members that consisting semi-rigid joints for structural damage detection of large-scale spatial structures.

- 2) Developing a machine learning-based approach for structural damage localization and quantification.
- 3) Conducting both numerical and experimental verification of the proposed machine learning based SHM damage detection approaches.

## **1.4 CONTRIBUTION OF THIS RESEARCH**

This study investigates the damage detection of complex spatial bridges using machine learning techniques. To bridge the gap between research and practice and adopt the advanced machine learning methods for damage detection, a couple of machine learning based methods were developed and investigated. The contributions of this study lie in

- A novel generic element for nonuniform beams with semi-rigid joints at two ends was developed to improve the accuracy of the numerical modelling and damage detection for large-scale spatial structures.
- A loss function for CNN based structural damage detection was proposed, specifically customized for the detection of structural damage, significantly reducing the computational cost, and achieving faster convergence speed and higher detection accuracy. This is for solving the challenge that the currently loss functions developed in machine learning are mostly for computer vision tasks rather than SHM.
- The research successfully illustrated a practical approach to employing multi-

channel data within a Long Short-Term Memory-Variational Autoencoder (LSTM-VAE) integrated semi-supervised framework for localising and quantifying the damage.

- CWT based Variational Autoencoder (VAE) provided a robust framework for interpreting complex signal patterns and quantifying structural damage.
- The effectiveness of supervised, semi-supervised, and unsupervised machine learning methods for damage detection in real-world scenarios were demonstrated.

## **1.5 ORGANISATION OF THIS THESIS**

This thesis is organized into seven chapters as below,

Chapter 1 introduces the research background, objectives, and contributions.

Chapter 2 reviews Structural Health Monitoring (SHM), particularly Vibration-Based Damage Detection (VBDD) techniques. It highlights the combination of machine learning (ML) within SHM to enhance damage detection accuracy and efficiency across various structures, showing a shift towards data-driven in ensuring structural safety.

Chapter 3 develops a novel generic element for nonuniform cross-section members with semi-rigid joints at both ends and proves its accuracy in numerical modelling and damage detection for spatial bridge structures.

Chapter 4 investigates a 2D Convolutional Neural Network (CNN) based method for

structural damage detection using vibration data. It proposes an innovative UMSE loss function as well as evaluation indices customized for structural damage detection.

Chapter 5 proposes a semi-supervised learning framework that combines a Long Short-Term Memory Variational Autoencoder (LSTM-VAE) network with a support vector machine to address structural damage detection for spatial bridge structures.

Chapter 6 presents an unsupervised approach using a convolutional Variational Autoencoder (VAE) for bridge damage quantification, focusing on the VAE's capability to quantify damage with the proposed Average Reconstructed Error (ARE) and reconstruct Continuous Wavelet Transform (CWT) images.

Chapter 7 presents the conclusions of this research together with recommendations for future study.

## **CHAPTER 2 LITERATURE REVIEW**

### **2.1 OVERVIEW**

The chapter begins with an exploration of Structural Health Monitoring (SHM), emphasizing the vital role of Vibration-Based Damage Detection (VBDD) in identifying structural damage through dynamic response analysis. VBDD has emerged as a critical technique for ensuring structural integrity, proving high structural damage detection accuracy and efficiency compared to traditional damage detection methods. This section introduces the current research using vibration-based methods for damage detection of various infrastructures.

Moreover, this chapter introduces machine learning (ML) and deep learning (DL) applications, presenting their significant impact on SHM. These AI techniques make full use of large datasets to find patterns and anomalies that might indicate structural damage, improving the accuracy and predictive capabilities of damage detection methods. The combination of ML and SHM presents a future where real-time, data-driven decision-making can solve structural damage detection problems, ensuring the safety of various structures.

### **2.2 THE VIBRATION-BASED DAMAGE DETECTION (VBDD)**

Vibration-Based Damage Detection (VBDD) is an advanced technique in SHM, and it uses dynamic responses to identify structural damage. In terms of algorithms, it could be



generally classified as model-based (or parametric) and data-driven (non-parametric).

The vibration parameters could be divided into frequency domain (e.g. modal parameters), time domain (e.g. time-history responses), and time-frequency domain (Hou & Xia, 2021).

The VBDD implementation involves two procedures: feature extraction and feature discrimination. The former extracts damage-sensitive features from the original measured responses while the latter discriminates features from damaged to healthy quantitatively.

Various damage features have been developed in the last three decades, including natural frequencies, mode shapes, modal strain energy, frequency response functions (FRFs), etc, each offering unique insights into structural health. The VBDD has superiority in the following two aspects, 1). they are suitable for even complex structures because they evaluate the overall structural condition rather than examining only the potential damaged local area like Non-Destructive Testing (NDT) methods; 2). Prices of the sensing system are quite acceptable.

### **2.2.1 Model-based methods**

The model-based methods, known as the physics-based method or parametric method, aim to build a computational model, e.g. finite element (FE) model, though updating model parameters through real measurement data. The aim of model updating (or termed model calibration) is to minimize the discrepancy between the real target structure and the FE model since the FE model may not present every physical aspect of the real structure. Model property matrices could be updated, including stiffness mass, and

damping matrices.

Based on the work of Chen et al. (2018), the updating parameters for framed structures could be material properties, geometric properties, and non-dimensional scalar multipliers. To be more specific, these updates could be generalized as 1) stiffness and mass at elemental level; 2) stiffness at integration point level; 3) material properties and sectional properties; 4) parameters for continuum structures; 5) updating joints and boundary conditions.

Model-based methods use parameters like mode shapes, modal strain energy, FRFs, and other dynamic or static responses to facilitate damage detection. Compared to the data-driven method, the model updating could provide a more calibrated numerical model for damage detection generalized. The model updating has been studied for the damage detection for various structures. Lee & Cho. (2016) used the updated finite element model to predict the probabilistic fatigue life for bridges. Additionally, Pu et al. (2019) used the model updating method and derived the frequency response functions for the damage detection and quantification of a laboratory-reinforced concrete beam.

One challenge of model updating is that model simplification and omission could lead to unavoidable modelling errors. Another problem when applying model updating on large structures is the high computation cost because of the significant number of parameters. This would cause ill-condition detection problems with many local minima. To solve this problem, the sub-structuring technique has been used. The sub-structuring technique

divides an FE model into substructures that each consist of elements so that they can be analysed separately. The substructuring method was originally developed in the 1960s because of computing limits. Nowadays, component mode synthesis (CMS), one of the most popular substructuring techniques, is used to divide and condense large-scale structures for damage detection. Xu et al. (2018) presented a multi-level damage identification approach using CMS. They demonstrated its effectiveness by decomposing a bridge structure into substructures and verified the approach through both numerical and experimental studies.

### **2.2.2 Data-driven methods**

The data-driven based methods use pattern recombination based on the previous measurements of the structure, rather than numerical or any other analytical model. The measured data could be natural frequencies, modal strain energy, damping, FRF, and their variants. The principle is that these modal parameters are the functions of the physical properties of structures such as mass and stiffness. Thus, structural damage could be detected by analysing these modal parameters. The widely used damage indicators in data-driven methods are reviewed here.

- *Damage indicator 1: Natural frequencies and mode shapes*

Natural frequencies, relatively simple to measure, are a major focus in damage detection. However, one main drawback of natural frequencies as damage indicators is that major damage may even lead to only small natural frequency changes. In addition, the measured

frequency may be influenced by the temperature change. By contrast, mode shapes, which are another parameter measured in data-driven methods, offer detailed information about local areas of a structure and is thus more sensitive to local damage. Despite this, the main drawbacks of mode shapes are that they can be easily affected by environmental noise, and also, mode shapes of lower modes may not easily be affected by local damage (Kim et al. 2003). Considering this, the natural frequencies and the mode shapes are sometimes combined to enhance damage detection capabilities. Capecchi et al. (2016) compared the role of modal frequencies, mode shapes, and curvatures on the damage detection using a double-hinged parabolic arch with a notch. Additionally, Umar et al. (2018) developed a new response surface damage detection method through combining the natural frequency and mode shapes and demonstrated its efficiency using a numerical beam and laboratory-test steel frame.

- *Damage indicator 2: Modal strain energy*

Modal Strain Energy (MSE) is a fundamental VBDD concept, providing information about the energy distribution of a structure under vibrational motion. It could identify damage where the energy distribution of a healthy state changes due to structural stiffness and mass properties change. The modal strain energy-related damage features include modal strain energy change (*MSEC*) and *MSEBI*(modal strain energy change). Shi et al. (2000) used elemental *MSEC* to localize and quantify the damage and verified its efficiency using a numerical model and an experimental single-bay, two-story steel portal frame. Seyedpoor (2012) used modal strain energy-based index as the damage index and

combined it with the particle swarm (PSO) to determine damage extent and proved the proposed method on a numerical model of the cantilevered beam. More recently, Khosravan et al. (2021) proposed the Improved Modal Strain Energy Decomposition Method (IMSEDM) to detect jacket-type offshore platforms damage, illustrating the continuous advancement. The modal strain energy is sometimes also combined with other modal parameters like mode shapes and natural frequencies to enhance the damage detection accuracy.

- *Damage indicator 3: Frequency response functions (FRFs)*

The use of modal parameters like natural frequencies and mode shapes can lead to inaccuracies due to measurement and modal extraction errors, as these are often determined indirectly. To avoid such problem, Wang et al. (1997) used the nonlinear perturbation equations of FRF data for detecting damage and proved the validity of the proposed method on a numerical and experimental plane 3-bay frame structure. The FRFs have the advantage of providing more structural information in the desired frequency range. Recently, the use of FRFs for damage detection has been increasing as demonstrated by Yoon et al. (2010), who use the experimental FRF to localize and quantify various damage including notches in steel beams, composite beams delamination, and dry spots of the composite hull structure. Moreover, Kim & Eun. (2017) showed that the proper orthogonal modes (POMs) from the FRFs in antiresonance frequency ranges could be used to provide explicit information about damage locations.

## **2.3 MACHINE LEARNING BASED STRUCTURAL DAMAGE DETECTION.**

Machine learning (ML) is the main branch of Artificial Intelligence (AI). It has become an essential tool in various domains, including computer vision, medical diagnostics, civil engineering, mechanical and electrical engineering, and financial markets. By processing historical data, the ML makes decisions and predictions. In the Structural health monitoring field, the ML has applications for tasks including damage detection, damage localization, predictive maintenance, condition assessment, and anomaly detection. The section reviews the foundational machine learning methods including ANNs, SVMs, and other methods, and presents their applications in structural damage detection.

### **2.3.1 Artificial neural networks (ANNs)**

ANNs are a computational system that consists of interconnected neurons to process information when there are external stimuli (Moein., 2014). It is a non-linear, multi-layered, and parallel regression system. Two basic concepts of the ANNs are the neurons and weights. The neurons are the basic units of the neural network with internal and adjustable coefficients (weights) bound to them. The learning process of ANNs could be considered as weight updating. Depending on the structure, which is constituted by the connected neurons, ANNs could be divided as follows. 1) multilayer neural networks; 2) recurrent neural networks; 3) cellular neural networks. Multilayer neural networks are the most commonly used. A typical ANNs consists of the input layer, hidden layers, and

output layer, each layer consisting of numerous neurons. The ANNs have been widely applied in many practical tasks including process monitoring, medical diagnosis, and fault diagnosis. Recent years have seen increasing research using ANNs. İzgi et al. (2012) used the ANNs to predict the short-mid-term solar power. Mohanraj et al. (2012) concluded that ANNs could achieve acceptable accuracy in energy and exergy analysis of refrigeration, air conditioning, and heat pump (RACHP) systems. In the field of atomistic simulations, Pun et al. (2019) developed the physically informed neural network (PINN) by combining a physical-based model with a neural network.

Artificial neural networks are a commonly used technique for bridge damage detection. Among various ANNs, the Back-Propagation Neural Network (BPNN) is the most used. For example, Neves et al. (2017) trained Artificial neural networks with acceleration data of healthy scenarios on a three-dimensional finite element of a fictitious railway bridge and established a threshold based on the statistical network prediction errors. By comparing the damage indices with the threshold, the damaged scenarios could be discriminated from the healthy. Shu et al. (2013) trained the Back-Propagation Neural Network (BPNN) with vibration responses for detecting the damage of a simply supported beam railway bridge. The results showed that damage localization is more difficult than damage quantification. It is also found that damage near the middle span of the bridge is more likely to be detected than damage in the support area. Also, the detection reliability is sensitive to noise, train speed, and axle loading. Malekjafarian et al. (2019) developed a two-stage machine learning method to detect bridge damage with

responses from a passing vehicle over the bridge. The ANN is trained in the first stage and in the second stage, a damage indicator is defined for detecting the distribution changes of the prediction errors. This method was implemented on a numerical vehicle-bridge interaction and managed to detect damage when there are road roughness profiles and measurement noise.

### **2.3.2 Support vector machine (SVMs)**

The support vector machine was a supervised machine learning methods that built a hyper plane to categorize new data points from a given dataset. It was firstly proposed in 1964. It was originally proposed for the classification and later was extended to solve regression problems. It is with high generalization ability, available in the high dimensional dataset, and could achieve high performance in dealing with recognition sample tasks of small size (Qin & He., 2005). One limitation of SVM is determining the kernel function. The SVM has been used in many domains such as computer vision, biotechnology, and chemistry. Recently, researchers tended to combine algorithms and techniques with SVM for various applications. For instance, Olatomiwa et al. (2015) introduced a hybrid method called SVM-FFA by coupling the SVM with the firefly algorithm (FFA) and used it to predict horizontal global solar radiation. Shamshirband et al. (2016) designed a coupled model by integrating the SVM and wavelet transform (WT) algorithm and used the model to estimate horizontal diffuse solar radiation.

The research using support vector machines mostly focuses on the damage detection of



bridge piers. Many use hybrid soft computing algorithms, e.g., SVM tuned by particle swarm optimization. Bao et al. (2013) pointed out that damage detection of bridge pier is important yet hard to detect since the piers are either under or have many affecting factors on the damage. They employed the support vector machine to detect the damage to the bridge pier and then used Grubb's test method to delete the outliers for SVM and extracted the SVM input parameters from a numerical analysis on a bridge pier. Results proved the efficiency of SVM for the detection of bridge pier damage. Gong et al. (2011) designed the SVM optimized by particle swarm optimization (PSO) for damage localization and quantification of a simply supported beam bridge with five girders. Both single and multiple damage cases were considered, and the results verified the effectiveness of the SVM using the PSO optimization algorithm. Another PSO-SVM is the work of Sreedhara et al. (2019). The authors investigated different shapes of the pier including circular, round-nosed, sharp-nosed, and rectangular piers, and built two models for comparison, these are the PSO-SVM and adaptive network-based fuzzy inference system (ANFIS) model with Gbell membership. The results showed that PSO-SVM outperformed the ANFIS model for the rectangular and sharp nose-shaped bridge pier. They concluded the PSO-SVM model can be an accurate and efficient method for pier scour depth prediction. In 2015, Soualhi et al. (2015) combined the Hilbert-Huang transform (HHT), the SVM, and the support vector regression (SVR) to monitor the ball bearing. In this method, the HHT is used for health indicator extraction from vibration signal, SVM is for detecting the degradation status, and the SVR is for estimating the remaining life. The results

proved the method could enhance the bearing degradation on detection, diagnostic and prognostic. Most recently, Huang et al. (2021) proposed a new method combining the SVM and moth-flame optimization (MFO) to detect structural damage considering temperature variations. In this method, the SVM is for determining the temperature variations and potential damage locations while MFO is for accurately locating and quantifying the damage. A numerical model proved that this combination could enhance damage detection efficiency and robustness. The efficiency of this method was proved through a simulated simply supported beam. They also implemented the method on a practical I-40 bridge and the results showed this method achieved high optimization performance and could detect damage of large complex bridges.

There is also research that employed multiple machine learning algorithms for bridge damage detection. For example, Lu et al. (2020) stated that slender bridges may experience dynamic responses and complex stress status because of the dynamic effects of wave, wind, and vehicle loads. Such cases would initiate cracks and would further trigger elemental failures. Therefore, they used different machine learning methods including the SVM, neural network, random forest, and Gaussian process (GP) to quantify the stochastic fatigue damage, thus further estimating the remaining fatigue life of these slender bridges. The results showed that all methods exhibited an acceptable ability to estimate fatigue accumulation, yet GP methods outperformed other machine learning algorithms. Moreover, Sun et al. (2018) used the dynamic fingerprints as damage features and combined the Rough Set theory and the Naïve-Bayes classifier (RSNB) for

damage qualification and validated the efficacy of the this method with a numerical continuous bridge. They also compared the performances of the RSNB with BPNN, SVM, and decision trees method and found that RSNB was superior to other methods in accuracy, efficiency, transparency, robustness, and stability.

### **2.3.3 Naïve bayes classifier**

Naïve Bayes classifier (or termed a simple Bayesian classifier) is a simple Bayesian Network (BN) and it uses a table of probabilities to calculate the likelihood of one instance belonging to a specific class (Fan & Leng Poh, 1 C.E.). The premise of the Naïve Bayes classifier is that the probability of historical events could potentially be used for estimating the future events' probability. The Naïve Bayes classifier is easy to build and implement as structure learning is not needed. It is highly resistant to overfitting because of its inherent inability to fit the training datasets accurately (Cichosz., 2015). Text classification is one domain where Naïve Bayes is most often used and successful. The Naïve Bayes method has also been proven to be effective at early gesture recognition (Escalante et al., 2016). One problem is that in reality, the independence assumption among attributes is sometimes violated. This problem could be solved in two ways, one is relaxing the independence assumption, and the other is employing a pre-processing method to select or transform the attributes.

### **2.3.4 Decision trees (DTs)**

The decision tree is a machine learning methods that divide data into branches at decision

points and it is a tree-like structure of conclusions mostly used for classification tasks (Quinlan., 1986). These methods have been studied for several decades and have numerous refinements and variations. Commonly used DT methods include random forest (RF), functional trees, classification, and regression trees. The DT requires less computation cost compared to other methods and does not need an assumption on data distribution. In the remote sensing domain, DTs could be used to extract geographical information (Wang & Li., 2008). Duan et al. used the decision trees to predict the potential zone of groundwater. Chen et al. (2018) compared the best-first decision tree, random forest, and naïve Bayes tree for landslide susceptibility mapping and found the random forest is the most promising method for landslide susceptibility mapping.

Research using decision trees for damage detection of bridge structures is limited. Among various decision tree methods, random forest trees are relatively commonly used. Lei et al. (2020) developed a vibration-based seismic method for evaluating the damaged statuses for regional bridges. In this method, they measured the structural dynamic and configuration parameters and trained a random forest algorithm. The results showed the model achieved 90% prediction accuracy. The random forest algorithm could also be combined with other machine learning algorithms for bridge damage detection. An example is the work of Jia et al. (2020). The authors combined the random forest algorithm (RF) and artificial neural network (ANN) to develop a fast assessment for seismic damage to bridges. The importance of the uncertain influencing factor of seismic damage on arch bridges and girder bridges was evaluated. Data from the Wenchuan

earthquake were taken as training and testing datasets while data from the Tangshan earthquake were used for validation. The results demonstrated that the RF-ANN method performed well at damage states assessment of two real bridges. It was with good robust ability and could save time for rescue work after the earthquake and help with bridge construction.

### **2.3.5 Principal component analysis (PCA)**

The principal component analysis, although not a machine learning method, is commonly used for dimension reduction in machine learning methods, reducing high-dimensional datasets into lower ones with orthogonal transformation yet retaining the most valuable information. PCA could collect hidden information within available data. The principal components (PCs) are the directions where the original data have the largest variances. By expressing the data with these selective PCs, the most contents of the data could be contained. PCA has been applied and studied in a wider variety of domains, including agriculture, genetics, geology, psychology, biology, chemistry ecology, and economics (Jolliffe., 2002). In omics fields, PCA is used for visualizing the omics data and confirming separation among groups (Gray., 2017). In ocean engineering, Eckert-Gallup et al. (2016) used PCA for creating an uncorrelated representation of sea state data. Recently, Cui et al. (2021) first employed PCA to obtain hidden information in the partial-nitrification process.

PCA has found significant use in Structural Health Monitoring (SHM), especially for

bridge damage detection. Its ability to condense high-dimensional vibration signals into informative, manageable components makes it an invaluable tool in this domain. Over the past two decades, PCA has been increasingly combined with machine learning techniques to enhance damage detection in bridges. This synergy often results in faster computation speeds with minimal loss in accuracy.

In 2005, Mei et al. (2019) designed a novel PCA-based framework based on data collected from sensors installed on passing-by vehicles on bridges. They collected the acceleration data from the vehicles, extracted the transformed feature from Mel-frequency cepstrum coefficients (MFCCs) and PCA, and detected the damage by comparing the transformed feature distributions. The numerical and experimental results showed that the proposed method could detect damage existence and provide valuable information on damage degrees. Sun et al. (2009) conducted PCA dimension reduction on the frequency response function to extract features and inputted those extracted features into the self-organizing map neural network. The effectiveness was demonstrated by a steel box girder model. The results showed that this method could distinguish the damaged scenario with good accuracy. Wang et al. (2020) pointed out the main problem of the previously used PCA-based method for detecting bridge damage in varying environmental temperatures is the decrease of effectiveness caused by the fact that the deleted information of PCA operation may be related to damage features. To solve this problem, they proposed a hybrid method in which the PCA deals with the non-principal components while the Gaussian mixture method (GMM) is for classifying principal components into various clusters. Then the

proposed method was conducted on each cluster. By doing this, all information on the damage feature is fully used for bridge damage detection. Finally, they implemented this method on the damage detection of a numerical and a real bridge and proved that this method managed to detect the bridge under environmental temperature variations.

PCA's role in SHM, particularly in bridge damage detection, highlights its effectiveness in distilling complex data into actionable insights. The continuous evolution of PCA-based methodologies, especially when combined with other analytical techniques, underscores its enduring relevance in engineering and data science.

## **2.4. DEEP LEARNING FOR STRUCTURAL DAMAGE DETECTION**

The deep learning (DL) technique, a sub-branch of machine learning, has gained the most attention among various machine learning algorithms and has been successfully applied to computer vision, natural language processing, medical diagnosis, online advertising, literary translation, and autopilot. Deep learning establishes deep neural networks (DNNs) with multiple layers to learn features of the data with multiple inherent patterns. DNNs characterize their automatic feature extraction and extraordinary ability to deal with big data sets. Great advantages of introducing DNNs into the SHM domain include, 1) the expert intervention of feature extraction in traditional SHM will be no longer necessary; 2) real-time monitoring is expected as big data sets from large-scale structures could be effectively and rapidly processed.

Most research modifies deep learning models and compares their performances with the

original models or other machine learning methods. Among various DNNs, Convolutional Neural Networks (CNNs) remain the state-of-the-art for their high accuracy and computational efficiency in object detection and classification. CNNs are particularly effective in processing large-scale structured sensor data and detecting localized damage patterns, which makes them well-suited for structural health monitoring tasks. However, recent developments in Graph Neural Networks (GNNs), introduced by Kipf and Welling (2016), have introduced techniques for capturing relational data in complex spatial structures, such as joints, beams, or different elements of a bridge, which CNNs may not fully capture. While CNNs are highly efficient for current structural damage detection, future research could explore the integration of GNNs to handle more complex relationships between structural components, potentially enhancing damage detection in such spatially distributed systems.

#### **2.4.1 Structural damage detection based on convolutional neural networks**

Among various DNNs, convolutional neural networks (CNNs) are considered to be state-of-art for their high accuracy and computational efficiency in object detection and classification. Considering the input data of CNNs, the current research could be divided into the vibration based CNNs and vision based CNNs. This division could also be considered as time-series based CNN and image-based CNN according to Sun et al. (2020). The former CNN is trained by structural responses such as mode shape and modal strain energy while the latter is trained by images either signal images or structure damage images, e.g. crack images. In 2017, a CNN-based approach was first used to perform



vibration-based damage detection on a steel frame (Abdeljaber et al., 2017). Since then, CNN has gained increasing attention on structural damage detection and more related research was conducted.

#### *2.4.1.1 Vibration based CNNs*

The input data of the vibration based CNNs are the modal parameters such as modal shapes, natural frequency, modal strain energy, and FRFs, as presented in Section 2.2.1. In such cases, both Conv1D and Conv2D operations could be used. An example of a Conv1D application is the work of Zhang et al. (2020). The authors designed a new deep CNN called the SHMnet and trained the SHMnet with the acceleration data of damaged scenarios from a laboratory steel frame. The results showed that the SHMnet was effective and reliable and had high anti-noise abilities. In 2018, Abdeljaber et al. (2018a) input the acceleration data into enhanced 1D CNNs for damage detection of a four-story steel structure (a benchmark frame) and verified the efficiency of proposed method on damage quantification. Li et al. (2020) proposed a deep CNNs based model to detect the damage of a scale-down bridge model and achieved 96.9% accuracy. It is also verified that the proposed deep learning model outperformed other machine learning algorithms including the random forest, support vector machine (SVM), and k-nearest neighbour.

The 1D training data could also be dealt with Conv2D operation. In such cases, the original data were usually collected with the reshape operation. An example is the work by Teng et al. (2020a) where the authors reshaped the original 1D data into 2D with

padded zeros and achieved good damage detection accuracy. They compared the results of training the CNN with a single damage parameter (modal strain energy) and multiple parameters (modal strain energy and acceleration data) from a numerical steel frame bridge and found the latter saw improvement in both classification accuracy and convergence speed. The results showed that the CNN could predict all damage states. Also, the method showed good anti-noise ability and was sensitive to small structural condition changes.

#### *2.4.1.2 Vision-based CNNs*

The vision-based CNNs are the direct adaptation of computer vision as they are trained with real structural images, e.g., crack images or bolt images. Based on the author's knowledge, the vision-based CNNs could also be classified as signal-based and damage image-based according to the input type. For the signal-based, the input images are signals e.g. acceleration signal images. For example, He et al. (2020) first used the recurrence graph as a damage feature as it could indicate the internal structures, similarity, and damage information. They input the recurrence graphs of damaged scenarios into CNN and proved their efficiency using numerical and model experiments of a beam bridge. Nguyen et al. (2020) trained the CNN using images from the Gapped Smoothing Method (GSM) to locate the damage in the numerical beam model. They also constructed a finite element model of the girder bridge to test the trained CNN. The results proved the efficiency of the proposed GSM and CNN-based method on damage detection and localization.

For the damage images based method, Li et al. (2020) used the multi-layer features obtained from a full CNN and a naïve Bayes data fusion (NB-FCN) model for automatic crack segmentation. They trained the NB-FCN with 7200 images of 10 bridges taken in different illuminates and distances and calculated the crack length and width to verify the NB-FCN method. With 7200 datasets from 20 bridges, they compared the proposed method with other crack detection machine learning methods including the crack tree method, the fusion CNN, the relatively competitive CNN, and proved the superiority of the method on training time, error rates and detection accuracy.

However, the performance of deep CNNs would be degraded because of the gradient vanishing problem as network depth increases. To overcome this difficulty, in 2016, Kaiming He et al. (2016) developed the deep ResNet by applying the residual learning in deep CNNs. They explicitly reformulate the layers as learning residual functions concerning the layer inputs. To do this, the shortcut connection was used to skip layers, which formed the building block of the residual network called the residual block. Currently, only limited research on bridge damage detection using ResNet has been found. Ahmed et al. (2020) proposed a novel automated rebar detection method for bridge inspections and maintenance by combining the deep residual networks and k-mean clustering techniques. They compared performances of different parameters including layer number and batch size of different deep residual networks and the results showed a positive correlation between these parameters and performances.

For the damage-image-based methods, commonly used CNNs are regional CNNs

(RCNNs), fast RCNNs, and faster RCNNs. The RCNN algorithm was first developed by Girshick et al. (2013). The main difference between RCNN and CNN is that RCNN proposes a bunch of potential regions called region proposals. RCNN aims to examine which regions contain the target object. The faster RCNN is the combination of fast RCNN and region proposal networks (RPNs). The Regional CNN and faster Regional CNN are considered as two-stage detection, as the first stage is region proposing. Up till now, little research has been found using the RCNN to detect the damage on bridge structures. For the civil engineering domain, the RCNN-related model is mostly used to detect detection of looseness of bolts, concrete cracks, and stay cables. This research could also be potentially used for bridge structures; thus they are reviewed in the following.

Zhang et al. (2020) employed the Faster RCNN for detecting the bolt states (tight and loosened) and the results showed that Faster RCNN could detect the bolt looseness with high accuracy. However, they found that the detection accuracy could be reduced due to the shooting angles, lighting conditions, and vibration conditions. Another example of RCNN for damage detection of bridge structures is the work by Huynh et al. (2019). They proposed a quasi-autonomous method for loosened bolt detection in which the RCNN is for detecting and cropping bolts and the Hough line transform for bolt angle estimation. They used the proposed method for detecting and cropping bolts on a real-scale girder ridge connection and the results showed the proposed method could be used for quasi-real-time monitoring for bolts, but the shooting perspective should be under 40 degrees

to guarantee the detection accuracy. To assess the crack for concrete structures, Kim used the mask and region-based convolutional neural network (Mask R-CNN). They trained the Mask R-CNN with crack images from concrete structures and found the network could successfully detect most of the cracks 0.3mm or wider cracks.

A problem of RCNN and its updated version is that the generation of region proposals can be time-consuming and inefficient because of the repetitive computation on overlapped regions (Zhang et al., 2020). Most recently, the single-stage detector called YOLO and their updated versions have been proposed for bridge damage detection. For example, Zhang et al., (2020) proposed an improved YOLOv3 for detection the of multiple damages on a concrete bridge and proved that the proposed method outperforms two-stage detectors like faster RCNN and ResNet. X. Chen et al. (2020) proposed the DT-YOLOv3 based on YOLOv3 and proved its efficiency.

In conclusion, both vibration-based and vision-based CNNs have their advantages and challenges on damage detection, yet the former is more promising because they use internal features and, hence could potentially detect damage regardless of location. For vision-based CNNs, one biggest challenges is that they rely heavily on image quality. Consequently, detection performance could be affected by lighting conditions, shooting angles, and shooting distances.

#### **2.4.2 Structural damage detection based on Autoencoders**

Autoencoders are unsupervised learning models that have gained significant traction in

structural health monitoring (SHM) for their ability to identify and quantify structural damage. Autoencoders consist of two components: the encoder and the decoder. The encoder compresses the original input data of high dimensions into a low-dimensional latent-space representation, and the decoder reconstructs the latent space representation back to its original dimension. This process of reconstruction aims to retain as much structural information as possible while reducing reconstruction error—the discrepancy between the input and its reconstructed output. An increased reconstruction error typically signals the presence of anomalies or damage within the structure, thereby serving as a critical performance indicator for the autoencoder.

#### *2.4.2.1 Autoencoders*

As one of the powerful unsupervised learning methods, they encode inputs into a lower-dimensional latent space and subsequently reconstruct the latent space into the output, making them particularly useful for identifying anomalies. The application of autoencoders in SHM has evolved from early supervised approaches for regression problems to the current focus on unsupervised methods for anomaly detection. Most research uses time series data or modal information as input and uses reconstructed error as a damage index for anomaly detection. The following reviews the progressive use of autoencoders in SHM, highlighting input methods and anomaly detection indices.

In 2018, Pathirage et al. (2018) proposed an autoencoder-based deep neural network method, AutoDNet, for detecting damage in steel frame structures. AutoDNet employs

modal information, like natural frequencies and mode shapes, as inputs. Its accuracy and efficiency in pinpointing damage locations and severities were validated through both numerical and experimental studies, AutoDNet is a significant step over traditional ANN methods, making full use of the strengths of autoencoders in a supervised way.

Bao et al. (2019) take a supervised method by converting time-series SHM data into images for anomaly detection. This method trains the stacked autoencoders to classify and detect anomalies, achieving high accuracy in identifying multiple anomaly patterns. However, the requirement for labelling for training data, especially for damaged scenario data has been a limitation.

In contrast, Hurtado et al. (2023) proposed a novel unsupervised learning framework using an adversarial autoencoder (AAE) for bridge health monitoring. This method use acceleration signals from vehicles crossing bridges to detect structural damage, using the Euclidean distance as the damage detection index. The innovation lies in its unsupervised nature, requiring no labelled data for training, and its ability to effectively identify damage severity. However, it relies on a consistent test vehicle and assumes constant vehicle speed, which poses practical limitations. Recent studies, including Mao et al. (2021) exploration of generative adversarial nets and Ni et al. (2020) development of a new SHM data compression and reconstruction method, signify the ongoing advancements in this field.

#### *2.4.2.2 Variational Auto-encoders (VAEs)*

VAE, a variant of autoencoders, is a generative model with latent space consisting of a

mixture of distributions rather than a fixed vector as the autoencoder. This is conducted by introducing randomness in the encoding process, making VAE a generative version of the autoencoders. This allows the VAE being able to generate new samples that resemble the previous sample, making the VAEs more robust as well as more suitable in situations with insufficiency of training data for damage detection.

The VAE method was introduced for structural health monitoring in 2020 by X. Ma et al. (2020). They developed a 1D convolutional variational autoencoder (CVAE) for the damage localization of beam-like structures under moving load. Featuring being base line free and data-driven, this method was validated by numerical and experiments and is suitable for practical SHM application. However, there are generalization issues that the model may not be generalizing well to detect the scenarios that were not trained on. Anaissi et al. (2023) present a multi-objective variational autoencoder (MVA) for smart infrastructure maintenance, using multiple sensor data for detecting, localizing, and assessing the damage levels in an unsupervised way. This method does not require a preset threshold for damage detection and thus is promising for detecting damage from practical real-world structures. The potential limitations may include the need for large amounts of data to achieve high accuracy, as well as not being suitable for the infrastructure it wasn't trained on. Most recent research by Pollastro et al. (2023) uses a hybrid method for structural anomaly detection, combining the variational autoencoder for feature extraction and 'a one-class support vector machine (OC-SVM)' for classification.

The current VAE based research for structural health monitoring shares the common



limitations 1) data dependency: Performance is heavily influenced by the quality and quantity of data; 2) high computational cost; 3) Interpretability: The latent representation results from VAE can be hard to interpret; 4) the hybrid way of VAE based method in a semi-supervised way still needs labelling, which might be difficult to get. Overall, improvements on the model interpretability and generalizability as well as computational efficiency are the focus.

### **2.4.3 Structural damage detection based on Recurrent Neural Networks (RNNs)**

#### *2.4.3.1 RNNs*

RNNs have been extensively explored as a Structural Health Monitoring (SHM) method across various types of structures due to their ability to maintain the memory of previous inputs. This ability is vital for processing time-series data features of SHM, where the detection of temporal patterns can indicate structural deterioration. Notably, the efficiency of RNN in SHM has been demonstrated within various structures including bridges (Mousavi & Gandomi, 2021; Sony et al., 2021), buildings (Perez-Ramirez et al., 2019), and Dams (Y. Li et al., 2022). These studies prove the capability of RNNs in capturing complex dependencies, therefore improving early detection accuracy of potential damage. This research mostly uses time series data (Fathnejat et al., 2023; Li et al., 2022) as input data, leveraging the sequential processing capabilities of RNNs to interpret temporal patterns that might indicate damage. Moreover, natural frequencies, waveform entropy, and other conventional indicators could also serve as data inputs and be proven to improve the prediction performance of related models (Zhang et al., 2019; Mousavi & Gandomi,

2021).

Emergent research also suggests that the efficacy of RNNs in damage detection can be further improved by customizing the network architecture. This customization may involve replacing standard layers with LSTM units, Gated Recurrent Units (GRU), or developing neural network variants suitable to specific structural characteristics. Such adaptations aim to optimize the network's ability to capture complex dependencies within the input data, thus providing a more reliable framework for SHM (Zhang et al., 2019).

Innovative approaches were also seen recently by combining RNNs with other machine learning approaches to enhance feature extraction from sensory data. Fathnejat et al. (2023) proposed a deep learning model that combines both 1D CNN and RNN variants. Furthermore, a Hierarchical CNN and GRU framework (HCG) was proposed by Yang et al. (2020) to model the relationship between spatial and temporal data for structural damage detection and was proven to outperform other existing models significantly.

Despite their successes, RNN applications in SHM face challenges such as being sensitive to noisy data and high computational costs. Additionally, the black-box nature of these models often leads to challenges in interpretability, a critical aspect of SHM. The integration of RNNs with emerging technologies such as digital twins and the Internet of Things (IoT) presents exciting prospects. Concurrently, there's also a growing interest in exploring unsupervised learning models to overcome the limitations of extensive labeled data dependency, making the SHM method more applicable and flexible in various

conditions.

#### *2.4.3.2 LSTM Units*

LSTM units, a typical Recurrent Neural Network (RNN), feature their capability to obtain long-term dependencies from datasets, a critical aspect for monitoring the structural integrity over time and identifying structural damages in various structures. Their internal memory allows them to process input data of varying lengths, suitable for practical applications in SHM.

The RNNs with LSTM layers have marked a significant progression in SHM, specifically in the areas of predictive maintenance and continuous real-time monitoring. A representative paper by Zhang et al. (2019) proposed a LSTM recurrent neural network method for bearing performance degradation assessment. This study demonstrates the potential of LSTM-RNNs in identifying the bearing degradation states and the remaining life prediction. Mousavi et al. (2021) combined Variational Mode Decomposition (VMD), Minimum Covariance Determinant (MCD), and RNN with Bi-directional Long-Short Term Memory (BiLSTM) cells for long-term out-only condition monitoring of civil infrastructures, aiming to detect early signs of structural changes or damage. It identifies significant deviations in the prediction errors of Mahalanobis distances, which are not considered in the RNN's initial healthy condition training stage. Damage becomes clear when there is a significant rise in the prediction errors for these distances, suggesting a change from the previously established patterns the model has learned. The bidirectional

nature of the LSTM cells offers a more robust representation of the signals, as it consists of both past and future information, which is a significant improvement compared to the traditional unidirectional LSTMs. This approach was designed for the real-time monitoring of civil infrastructures, presenting a refined method for the early detection of potential structural failures.

## **2.5 SUMMARY**

This chapter reviews the structural damage detection methods, from the fundamentals of Vibration-Based Damage Detection (VBDD) to advanced machine learning methods. Innovations in deep learning, including CNNs, VAEs, and Recurrent Neural Networks (RNNs), have advanced the methods for predictive maintenance, real-time structural monitoring and damage detection. Two major research gaps have been identified:

- The assumptions about joint rigidity and elemental uniformity result in inaccuracies in numerical modelling, which are particularly evident in modelling large structures with numerous elements and joints. When data from these less accurate models are used to power data-driven methods, the efficacy of machine-learning approaches for identifying structural damage is reduced.
- To use machine learning techniques, which were originally developed for a wide range of applications, for damage detection, there is a need for customization. This involves customization in selecting appropriate input data, implementing

algorithms, evaluating results, and interpreting data to ensure that these methods meet the unique challenges of structural damage detection effectively.

# **CHAPTER 3 STRUCTURAL DAMAGE DETECTION FOR SPATIAL FRAME STRUCTURES WITH SEMIRIGID JOINTS USING WIRELESS MEASUREMENTS**

## **3.1 OVERVIEW**

The large spatial structures are widely used for bridge and stadium constructions in the world. The spatial structure has deteriorated due to environmental erosion, aging material, uneven support settlement, fatigue, and extreme external loads such as earthquakes and typhoons (Adams, 2007; Kim & Bartkowicz, 1993). The failure of such structures could lead to huge economic costs and human lives loss. It is important to ensure their safety and detect damage at an early stage to avoid the possible collapse of structures and extend the structural service life. Structural damage detection for spatial structures has attracted the interest of researchers and engineers (Xu et al., 2022). The joint is a key component of the spatial structure. Most existing methods consider the joint as rigid, non-uniform element as uniform, and these assumption leads to a large error in the damage identified result. Due to the complex mechanism at the joint of the spatial structure, it is still a challenging task to detect its damage in operational environments.

Finite element (FE) modelling has been widely used to investigate the behaviour of spatial frame structures in practice. The idealization or simplification of joint connections and boundary conditions has a big effect on the results of the numerical modelling (Hou et al., 2021). The joints are usually idealized as fully rigid in existing models. The flexibility

always occurs at the joint as multiple structural elements are connected at the joint with fasteners such as bolts, welds, or screws. It has been proved that the structural behaviour is significantly affected by the flexibility of the joint (Jezequel., 1983). The flexibility of joints should be considered to achieve better accuracy for structural analysis (Law et al., 2001). The semi rigid joint model is used to include the joint flexibility (Paral et al., 2021). The stiffness matrix of a member with elastic restraints at the ends were obtained by modifying with a correction matrix and it was used to investigate the frame with semi-rigid connections (Monforton & Wu, 1963). A mechanical model with three springs and a non-deformable node was proposed to analyse the behaviour of the steel frames with semi-rigid joints (Ihaddoudène et al., 2009). The material and geometric nonlinearities have been considered in the numerical modelling for a single-layer spatial structure and the influence of the joint rigidity on the mechanical performance of the structure is studied (Ma et al., 2015). In practice, the members at the joint are mostly with nonuniform cross-section properties due to the need to install non-structural components like bolts. Little research has been found including the nonuniform cross section properties into the stiffness matrix of the member with semi-rigid joints.

The vibration-based method is widely applied for detecting damage of spatial structures. The change of structural dynamic properties, such as natural frequencies, mode shapes, damping, modal strain energy (MSE), frequency response function (FRF), and their variants, etc., can be used to detect structural damage (Hou & Xia, 2021). The *MSE* based damage detection is one of the most promising methods as it incorporates the system

vibration behaviour and physical properties ( Wang & Xu, 2019). Many modal strain energy-based methods have been proposed for structural damage detection in the last two decades, including the modal strain energy change (*MSEC*) (Shi et al., 1998), the cross-modal strain energy (*CrossMSE*) method (Hu et al., 2007; Zhang et al., 2022), the modal strain energy based index (*MSEBI*) (Seyedpoor, 2012), the modal strain energy equivalence index (*MSEEI*) (Guo & Li., 2014), and the improved modal strain energy decomposition method (*IMSEDM*) (Khosravan et al., 2021) etc. The performance of the above model-based methods highly depends on the accuracy of the finite element model. Most existing methods consider the joints are either pinned or rigid for simplicity in structural damage detection. Simplified assumptions may lead to false damage identification (Hou et al., 2021). In this study, the non-uniform cross-section element with semi-rigid joints at both ends will be incorporated into the finite element model and the modal strain energy based index (*NMSEC*) is adopted to detect the member and joint damage of spatial structures.

In this chapter, a novel generic element of the nonuniform cross-section member with semi-rigid joints at both ends has been developed and the finite element model of spatial structures has been established using the proposed generic element. An 8-meter spatial bridge model with bolted connections has been built in the laboratory. Wireless sensors are used to monitor the responses of the bridge under random excitation and the reference-based stochastic subspace identification (Ref-SSI) method is adopted to extract the mode shapes from multiple sets of measurements. The finite element model is validated first



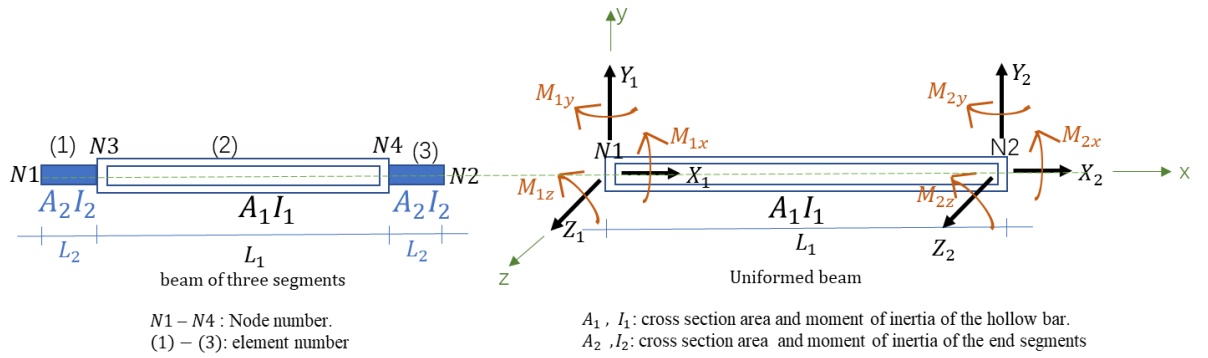
using the experimental results of the bridge without damage. Different scenarios of the bridge with the beam and joint damage are then simulated using the validated model. The modal strain energy change-based damage index is obtained and adopted for structural damage detection. The experimental results were used to further verify the damage-identified results. The results show that the proposed method is reliable and accurate in localizing both the beam and joint damage of the structure.

## **3.2 GENERIC ELEMENT FOR A NONUNIFORM BEAM WITH SEMI-RIGID JOINTS**

This section is to introduce the nonuniform beam with rigid joints first and then derive the generic element stiffness matrix of the nonuniform cross-section beam member with semi-rigid joints at the ends. Furthermore, the modal strain energy-based damage index is defined for structural damage detection.

### **3.2.1 Stiffness matrix of the nonuniform beam with rigid joints**

Beams are mostly nonuniform in practice and they consist of several segments connected by bolts. The segments have different cross-section properties. As shown in Figure 3.1(a), the nonuniform beam could split into three segments, e.g., a uniform middle beam and two end beams with small cross-sections. To derive the generic element, an equivalent uniform beam as shown in Figure 3.1(b) is considered. In this section, the stiffness matrix of the three-segment beam is obtained using the equivalent beam and joint stiffness.



(a) The three-segment beam

(b) The equivalent uniform beam

Figure 3.1 Uniform and nonuniform beams

The nonuniform beam with the total length  $L$  in Figure 3.1(a) is divided into three uniform beam segments with four nodes. The moment of inertia  $I$ , the cross-section area  $A$  and the length  $L$  of three segments are different. Here  $I_1, A_1$  and  $L_1$  are for the middle segment (noted as Segment 2), and  $I_2, A_2$  and  $L_2$  are for two side segments (noted as Segments 1 and 3). The subscripts  $y$  and  $z$  are the moment of inertia on  $Y$  and  $Z$  axes, respectively. A three-dimensional (3D) beam element has 12 degrees of freedom (DOFs), and each node has six DOFs, e.g. the displacement vector  $\{u, v, w, \phi_x, \phi_y, \phi_z\}$ . Thus, the element stiffness matrixes of three segments could be represented separately as  $\mathbf{K}^{(1)}, \mathbf{K}^{(2)}$  and  $\mathbf{K}^{(3)}$  ( $\mathbf{K}^{(1)} = \mathbf{K}^{(3)}$ ). The force vectors  $\{X_i, Y_i, Z_i, M_{ix}, M_{iy}, M_{iz}\}$  of Segments 1 and 3 can be represented using the corresponding displacement vectors  $\{u_i, v_i, w_i, \phi_{ix}, \phi_{iy}, \phi_{iz}\}$  at Nodes 1, 2, 3 and 4. Similarly, the force vectors of Segment 2 is  $\{X'_i, Y'_i, Z'_i, M'_{ix}, M'_{iy}, M'_{iz}\}$  and it can be represented as by  $\{u_i, v_i, w_i, \phi_{ix}, \phi_{iy}, \phi_{iz}\}$  at Nodes 3 and 4. Since Segments 1 and 2 share Node 3, and Segments 2 and 3 share Node 4, the following equations can be obtained by the equilibrium.

$$\begin{cases} X_i + X'_i = 0 \\ Y_i + Y'_i = 0 \\ Z_i + Z'_i = 0 \\ M_{ix} + M'_{ix} = 0 \\ M_{iy} + M'_{iy} = 0 \\ M_{iz} + M'_{iz} = 0 \end{cases} \quad i=3,4 \quad (3.1)$$

With Eq. (3.1), the nodal displacements at Nodes 3 and 4 can be represented by the functions of the nodal displacements at Nodes 1 and 2. Substitute these representations into force-displacement equations of Segments 1 and 3, and the displacement vectors of Nodes 3 and 4 are eliminated. This equivalent stiffness matrix  $\mathbf{K}^e$  of the nonuniform beam element with different cross-section properties of three segments can be obtained as,

$$\mathbf{K}^e = \begin{vmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^T & \mathbf{K}_{22} \end{vmatrix} \quad (3.2)$$

where,

$$\mathbf{K}_{11} = \begin{vmatrix} e_1 \frac{EA_1}{L} & \square & \square & \square & \square & \square \\ \square & e_2 \frac{12EI_{1z}}{L^3} & \square & \square & \square & e_2 \frac{6EI_{1z}}{L^2} \\ \square & \square & e_3 \frac{12EI_{1y}}{L^3} & \square & -e_3 \frac{6EI_{1y}}{L^2} & \square \\ \square & \square & \square & e_4 \frac{GI_{1p}}{L} & \square & \square \\ \square & \square & -e_3 \frac{6EI_{1y}}{L^2} & \square & e_5 \frac{4EI_{1y}}{L} & \square \\ \square & e_2 \frac{6EI_{1z}}{L^2} & \square & \square & \square & e_7 \frac{4EI_{1z}}{L} \end{vmatrix}$$

$$K_{12} = \begin{bmatrix} -e_1 \frac{EA_1}{L} & \square & \square & \square & \square & \square \\ \square & -e_2 \frac{12EI_{1z}}{L^3} & \square & \square & \square & e_2 \frac{6EI_{1z}}{L^2} \\ \square & \square & -e_3 \frac{12EI_{1y}}{L^3} & \square & -e_3 \frac{6EI_{1y}}{L^2} & \square \\ \square & \square & \square & -e_4 \frac{GI_{1p}}{L} & \square & \square \\ \square & \square & e_3 \frac{6EI_{1y}}{L^2} & \square & e_6 \frac{2EI_{1y}}{L} & \square \\ \square & -e_2 \frac{6EI_{1z}}{L^2} & \square & \square & \square & e_8 \frac{2EI_{1z}}{L} \end{bmatrix}$$

$$K_{22} = \begin{bmatrix} e_1 \frac{EA_1}{L} & \square & \square & \square & \square & \square \\ \square & e_2 \frac{12EI_{1z}}{L^3} & \square & \square & \square & -e_2 \frac{6EI_{1z}}{L^2} \\ \square & \square & e_3 \frac{12EI_{1y}}{L^3} & \square & e_3 \frac{6EI_{1y}}{L^2} & \square \\ \square & \square & \square & e_4 \frac{GI_{1p}}{L} & \square & \square \\ \square & \square & e_3 \frac{6EI_{1y}}{L^2} & \square & e_5 \frac{4EI_{1y}}{L} & \square \\ \square & -e_2 \frac{6EI_{1z}}{L^2} & \square & \square & \square & e_7 \frac{4EI_{1z}}{L} \end{bmatrix}$$

where  $E$ ,  $G$  are the Young's modulus and the shear modulus, respectively.  $I_{1y}, I_{1z}$  are the moment of inertia for the middle segment about the Y and Z axes, respectively.  $I_{1p} = I_{1y} + I_{1z}$  is the polar moment of inertia. The parameters  $e_1$  to  $e_8$  are as follows,

$$e_1 = \frac{A_2 L}{2A_1 L_2 + L_1 A_2}; e_2 = \frac{I_{2z} L^3}{I_{2z} L_1^3 + 6I_{1z} L_1^2 L_2 + 12I_{1z} L_1 L_2^2 + 8I_{1z} L_2^3};$$

$$e_3 = \frac{I_{2y} L^3}{I_{2y} L_1^3 + 6I_{1y} L_1^2 L_2 + 12I_{1y} L_1 L_2^2 + 8I_{1y} L_2^3}; e_4 = \frac{I_{2p} L}{2I_{1p} L_2 + I_{2p} L_1};$$

$$e_5 = \frac{I_{2y} L (16I_{1y} L_2^3 + 2I_{2y} L_1^3 + 18I_{1y} L_1 L_2^2 + 6I_{1y} L_1^2 L_2 + 6I_{2y} L_1 L_2^2 + 6I_{2y} L_1^2 L_2)}{2(12I_{1y}^2 L_1^2 L_2^2 + 24I_{1y}^2 L_1 L_2^3 + 16I_{1y}^2 L_2^4 + 8I_{1y} I_{2y} L_1^3 L_2 + 12I_{1y} I_{2y} L_1^2 L_2^2 + 8I_{1y} I_{2y} L_1 L_2^3 + I_{2y}^2 L_1^4)}$$

$$e_6 = \frac{I_{2y} L (8I_{1y} L_2^3 + I_{2y} L_1^3 + 6I_{1y} L_1 L_2^2 + 6I_{2y} L_1 L_2^2 + 6I_{2y} L_1^2 L_2)}{12I_{1y}^2 L_1^2 L_2^2 + 24I_{1y}^2 L_1 L_2^3 + 16I_{1y}^2 L_2^4 + 8I_{1y} I_{2y} L_1^3 L_2 + 12I_{1y} I_{2y} L_1^2 L_2^2 + 8I_{1y} I_{2y} L_1 L_2^3 + I_{2y}^2 L_1^4}$$

$$e_7 = \frac{I_{2z}L(16I_{1z}L_2^3 + 2I_{2z}L_1^3 + 18I_{1z}L_1L_2^2 + 6I_{1z}L_1^2L_2 + 6I_{2z}L_1L_2^2 + 6I_{2z}L_1^2L_2)}{2(12I_{1z}^2L_1^2L_2^2 + 24I_{1z}^2L_1L_2^3 + 16I_{1z}^2L_2^4 + 8I_{1z}I_{2z}L_1^3L_2 + 12I_{1z}I_{2z}L_1^2L_2^2 + 8I_{1z}I_{2z}L_1L_2^3 + I_{2z}^2L_1^4)}$$

$$e_8 = \frac{I_{2z}L(8I_{1z}L_2^3 + I_{2z}L_1^3 + 6I_{1z}L_1L_2^2 + 6I_{2z}L_1L_2^2 + 6I_{2z}L_1^2L_2)}{12I_{1z}^2L_1^2L_2^2 + 24I_{1z}^2L_1L_2^3 + 16I_{1z}^2L_2^4 + 8I_{1z}I_{2z}L_1^3L_2 + 12I_{1z}I_{2z}L_1^2L_2^2 + 8I_{1z}I_{2z}L_1L_2^3 + I_{2z}^2L_1^4}$$

### 3.2.2 Stiffness matrix of a uniform beam with semi-rigid joints

Joint connections are always simulated as pins or rigid joints while most joints behave semi-rigidly with some flexibility. A 3D generic beam element with semi-rigid joints can be modelled with zero-length springs as shown in Figure 3.2 (Hou et al., 2021). The rotational springs stiffness at the left and right ends are  $k_{\theta}^L$  and  $k_{\theta}^R$  respectively. For a 3D beam element, each end has 6 DOFs, that include the axial and shear deformations, and three rotations modelled by three rotational springs.

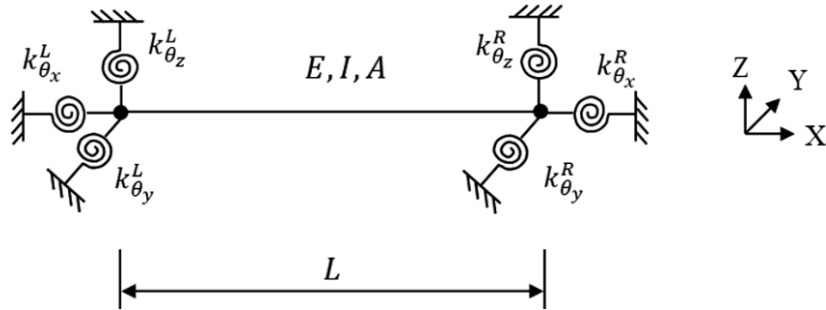


Figure 3.2 3D beam element with semi-rigid joints

The joint rotational stiffness factor is introduced to evaluate the joint contribution to bending and it is defined as the ratio of the rotational stiffness  $k_{\theta}$  over the flexural stiffness  $EI/L$ . A joint with a higher joint rotational stiffness factor behaves more rigid in that direction while the lower factor joint behaves more like a pin. As each end has three rotational springs, six rotational stiffness factors are defined as follows,

$$k_{\theta y}^L = \alpha^L \frac{EI_y}{L}; k_{\theta y}^R = \alpha^R \frac{EI_y}{L}; k_{\theta z}^L = \beta^L \frac{EI_z}{L}; k_{\theta z}^R = \beta^R \frac{EI_z}{L}; k_{\theta x}^L = \gamma^L \frac{GI_p}{L}; k_{\theta x}^R = \gamma^R \frac{GI_p}{L} \quad (3.3)$$

where  $\alpha, \beta, \gamma$  are the rotational stiffness factors at  $y, z,$  and  $x$  directions. The subscripts ‘L’ and ‘R’ represent the left and right ends respectively.  $I_y$  and  $I_z$  are the moment of inertia of  $y$  and  $z$  directions and  $I_p = I_y + I_z$  is the polar moment of inertia.  $E$  and  $G$  are the elastic modulus and shear modulus.

The above stiffness factors represent the rigidity of the semi-rigid joint. The joint damage is defined as the stiffness reduction, and it can be quantified by the factor reductions. The stiffness matrix of the beam element with semi-rigid joints can be obtained by introducing these factors into Eq. (3.2). The stiffness matrix of the nonuniform cross-section beam with semi-rigid joints can be represented as,

$$\mathbf{K}^{e'} = \begin{bmatrix} K'_{11} & K'_{12} \\ K'_{12} & K'_{22} \end{bmatrix} \quad (3.4)$$

where

$$K'_{11} = \begin{bmatrix} a_1 \frac{EA_1}{L} & \square & \square & \square & \square & \square \\ \square & a_2 \frac{12EI_{1z}}{L^3} & \square & \square & \square & a_7 \frac{6EI_{1z}}{L^2} \\ \square & \square & a_3 \frac{12EI_{1y}}{L^3} & \square & -a_8 \frac{6EI_{1y}}{L^2} & \square \\ \square & \square & \square & a_4 \frac{GI_{1p}}{L} & \square & \square \\ \square & \square & -a_8 \frac{6EI_{1y}}{L^2} & \square & a_5 \frac{4EI_{1y}}{L} & \square \\ \square & a_7 \frac{6EI_{1z}}{L^2} & \square & \square & \square & a_6 \frac{4EI_{1z}}{L} \end{bmatrix}$$

$$K'_{12} = \begin{bmatrix} -a_1 \frac{EA_1}{L} & \square & \square & \square & \square & \square \\ \square & -a_2 \frac{12EI_{1z}}{L^3} & \square & \square & \square & a_{12} \frac{6EI_{1z}}{L^2} \\ \square & \square & -a_3 \frac{12EI_{1y}}{L^3} & \square & -a_{11} \frac{6EI_{1y}}{L^2} & \square \\ \square & \square & \square & -a_4 \frac{GI_{1p}}{L} & \square & \square \\ \square & \square & a_8 \frac{6EI_{1y}}{L^2} & \square & a_9 \frac{2EI_{1y}}{L} & \square \\ \square & -a_7 \frac{6EI_{1z}}{L^2} & \square & \square & \square & a_{10} \frac{2EI_{1z}}{L} \end{bmatrix}$$

$$K'_{22} = \begin{bmatrix} a_1 \frac{EA_1}{L} & \square & \square & \square & \square & \square \\ \square & a_2 \frac{12EI_{1z}}{L^3} & \square & \square & \square & -a_{12} \frac{6EI_{1z}}{L^2} \\ \square & \square & a_3 \frac{12EI_{1y}}{L^3} & \square & a_{11} \frac{6EI_{1y}}{L^2} & \square \\ \square & \square & \square & a_4 \frac{GI_{1p}}{L} & \square & \square \\ \square & \square & a_{11} \frac{6EI_{1y}}{L^2} & \square & a_{13} \frac{4EI_{1y}}{L} & \square \\ \square & -a_{12} \frac{6EI_{1z}}{L^2} & \square & \square & \square & a_{14} \frac{4EI_{1z}}{L} \end{bmatrix}$$

where  $a_1$  to  $a_{14}$  are as follows.

$$a_1 = e_1; a_2 = e_2 \frac{\alpha^L + \alpha^R + \alpha^L \alpha^R}{4(\alpha^L + \alpha^R) + \alpha^L \alpha^R + 12}; a_3 = e_3 \frac{\beta^L + \beta^R + \beta^L \beta^R}{4(\beta^L + \beta^R) + \beta^L \beta^R + 12};$$

$$a_4 = e_4 \frac{\gamma^L \gamma^R}{\gamma^L + \gamma^R + \gamma^L \gamma^R}; a_5 = e_5 \frac{\beta^L (3 + \beta^R)}{4(\beta^L + \beta^R) + \beta^L \beta^R + 12};$$

$$a_6 = e_7 \frac{\alpha^L (3 + \alpha^R)}{4(\alpha^L + \alpha^R) + \alpha^L \alpha^R + 12}; a_7 = e_2 \frac{\alpha^L (2 + \alpha^R)}{4(\alpha^L + \alpha^R) + \alpha^L \alpha^R + 12};$$

$$a_8 = e_3 \frac{\beta^L (2 + \beta^R)}{4(\beta^L + \beta^R) + \beta^L \beta^R + 12}; a_9 = e_6 \frac{\beta^L \beta^R}{4(\beta^L + \beta^R) + \beta^L \beta^R + 12};$$

$$a_{10} = e_8 \frac{\alpha^L \alpha^R}{4(\alpha^L + \alpha^R) + \alpha^L \alpha^R + 12}; a_{11} = e_3 \frac{\beta^R (2 + \beta^L)}{4(\beta^L + \beta^R) + \beta^L \beta^R + 12};$$

$$a_{12} = e_2 \frac{\alpha^R(2 + \alpha^L)}{4(\alpha^L + \alpha^R) + \alpha^L\alpha^R + 12}; a_{13} = e_5 \frac{\beta^R(3 + \beta^L)}{4(\beta^L + \beta^R) + \beta^L\beta^R + 12};$$

$$a_{14} = e_7 \frac{\alpha^R(3 + \alpha^L)}{4(\alpha^L + \alpha^R) + \alpha^L\alpha^R + 12}$$

Eq. (3.4) represents the stiffness matrix of the generic element for a nonuniform beam with semi-rigid joints. The element can be incorporated into the finite element model of the spatial structure.

### 3.2.3 Modal strain energy-based damage index

The elemental damage is defined as the beam element stiffness reduction and the joint rotational stiffness reduction. From Eq.(3.4), the elemental stiffness matrix of a uniform beam with semi-rigid joints consists of several parameters, including  $A, I_y, I_z, I_p, \alpha^L, \alpha^R, \beta^L, \beta^R, \gamma^L, \gamma^R$ . For simplicity, in this study, the damage of the beam members is simulated by the elastic modulus reduction and the damage of the joint by the moment of inertia reduction ( $I_{YY}$ ).

Structural damage leads to stiffness reduction at one or several elements and causes changes in the mode shapes. The modal strain energy of an element is defined as the product of the elemental stiffness matrix and the mode shapes as (Shi et al., 1998),

$$MSE_{im} = \Phi_m^T K_i \Phi_m \quad (3.5)$$

where  $MSE_{im}$  is the modal strain energy of the  $m^{th}$  mode at the  $i^{th}$  element. When damage occurred, mode shapes changed, leading to changes in elemental modal strain energy. For the  $m^{th}$  mode, the modal strain energy change ratio ( $MSECR$ ) of  $i^{th}$  element was further



defined as the elemental change ratio of  $MSE$  before and after damage, as follows.

$$MSECR_i^m = \frac{|MSE_{im}^H - MSE_{im}^D|}{MSE_{im}^H} \quad (3.6)$$

where  $MSE_{im}^H$  and  $MSE_{im}^D$  is the corresponding modal strain energy of the  $m^{th}$  mode at the  $i^{th}$  element with the healthy and damaged scenarios. In this study, normalized modal strain energy ( $NMSEC$ ) based damage index is utilized by considering multiple modes, and the  $NMSEC$  of the  $i^{th}$  element is defined as,

$$NMSEC_i = \sum_{m=1}^{Nmode} w_m MSECR_i^m = \sum_{m=1}^{Nmode} w_m \frac{|MSE_{im}^H - MSE_{im}^D|}{MSE_{im}^H},$$

$$m = 1, 2, \dots, Nmode; i = 1, 2, \dots, Nelement. \quad (3.7)$$

where  $Nmode$  and  $Nelement$  are the numbers of modes and elements respectively. In this study, the number of modes is 6 and the number of elements is 160.  $w_m$  is the weighting factor of the  $m^{th}$  mode and it is defined as,

$$w_m = \left| \frac{\sum_{i=1}^{Nelement} MSE_{im}}{TMSE} \right|^{-1} \quad (3.8)$$

where  $TMSE$  is the total modal strain energy of all six modes. The weighting factor  $w_m$  is introduced as the higher mode inherently has a higher modal strain energy while it does not contribute much to the damage localization (Shi et al., 1998). Using  $w_m$ , it allows lower modes have more weight. Figure 3.3 illustrates the steps to verify the proposed model as follows.

- 1) The detailed experimental setup and procedure of a spatial bridge model are introduced first, and the tests have been carried out.

- 2) The modal parameters are extracted from multiple measurement datasets on the bridge model with different damage scenarios using Ref-SSI method.
- 3) A finite element model for the spatial bridge is built and validated using the experimental results.
- 4) The sensitivity analysis of the beam and joint damage is carried out using the updated model.

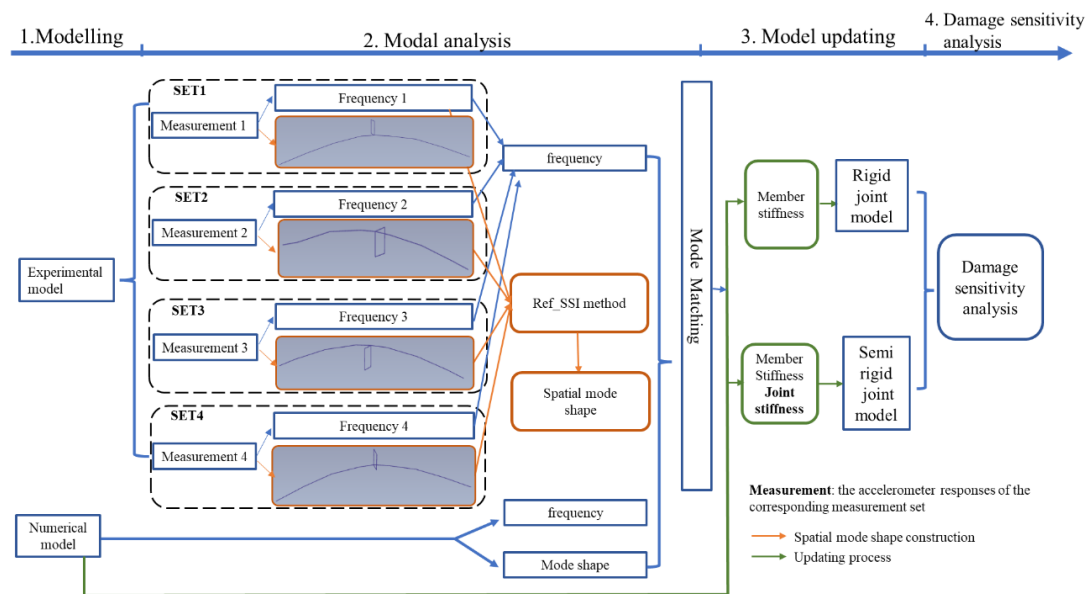


Figure 3.3 Flowchart of this chapter

### 3.3 EXPERIMENTAL SETUP AND MODAL ANALYSIS

#### 3.3.1 The bridge model

Figure 3.4 shows a 14-span spatial frame bridge with 7.98 m long, 0.6 m wide, and 0.6 m high. The model was built according to the design specification presented by the Smart Structures technology Laboratory (SSTL) of the University of Illinois at Urbana-

Champaign (Gao and Spencer, 2007). The bridge is simply supported with a pin at the left end and a roller at right end. The bridge consists of 160 tubular beam members and 56 joints. There are two kinds of beam members, e.g. the diagonal and non-diagonal members. All beam members are nonuniform and two end portions are solid segments of 10 mm diameter connected with the middle part by screws. The length of the solid segment is 100 mm. The cross section of the middle part is with an inner diameter of 12 mm and an outer diameter of 18 mm. The length of the middle part for diagonal members is 600 mm while it is 400 mm for vertical and horizontal members. The members are connected at joints by bolts.

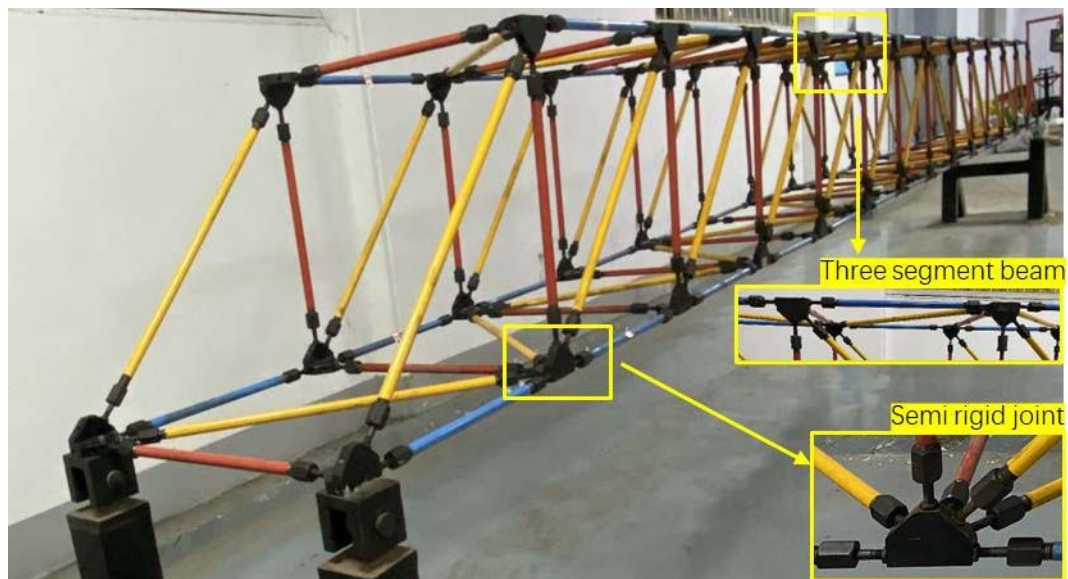


Figure 3.4 The 7.98m experimental truss bridge

### 3.3.2 Experimental setup

Figure 3.5 shows the experimental system including a power amplifier (CF6502), a shaker (CF6900-100), a signal generator (AFG1022), 13 wireless tri-axis accelerometers, and

the wireless data acquisition system. The shaker was connected to the bridge on the fifth joint from left of the lower chord through a stinger. The white noise signal with an amplitude 10 V was generated by the signal generator and amplified by the power amplifier and then the excitation force was applied on the bridge joint through the stinger. The tri-axis wireless accelerometers were installed on the upper surface of joints through strong magnetic blocks. The sampling frequency is 256Hz and the responses at horizontal (X) and vertical (Z) directions were recorded. The recording length for each test is 20 minutes. Figure 3.6 shows the horizontal response of Accelerometer No. 4 in Set 1.

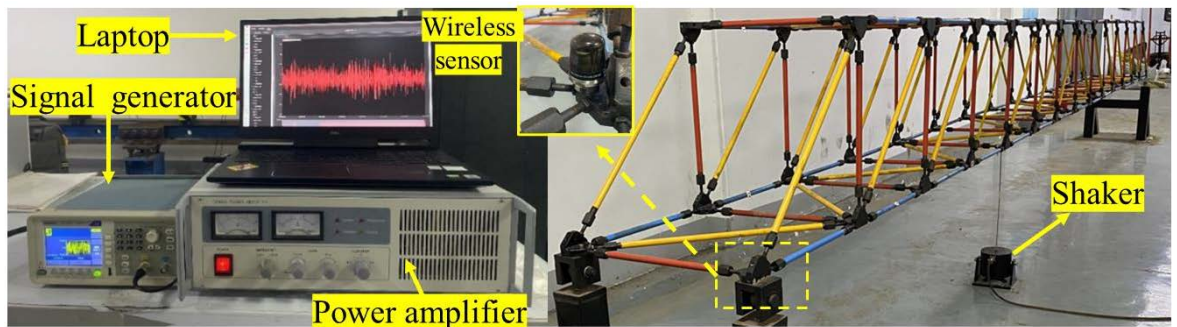


Figure 3.5 Experimental setup

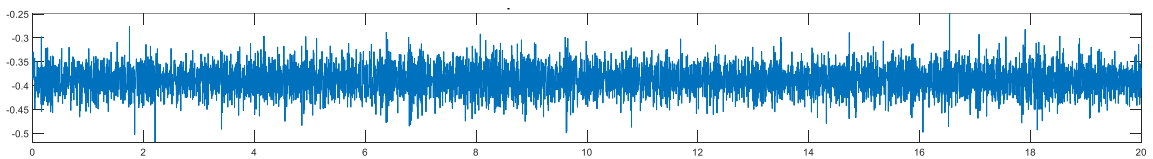


Figure 3.6 Acceleration response of accelerometer No.4 in horizontal direction

To get spatial mode shapes of the bridge, four measurement sets were recorded for four longitudinal edges respectively. For each measurement set, eight accelerometers were installed evenly along the bridge's longitudinal direction and another five accelerometers were used as reference. The locations of accelerometers for Sets 1~4 are shown in Figure 3.7. Four accelerometers (No. 5~8) in the middle span of the bridge and another

accelerometer (No. 13) were used as reference points. Upon finishing one edge measurement, eight accelerometers were moved to next edge and the reference accelerometers were stayed stationary.

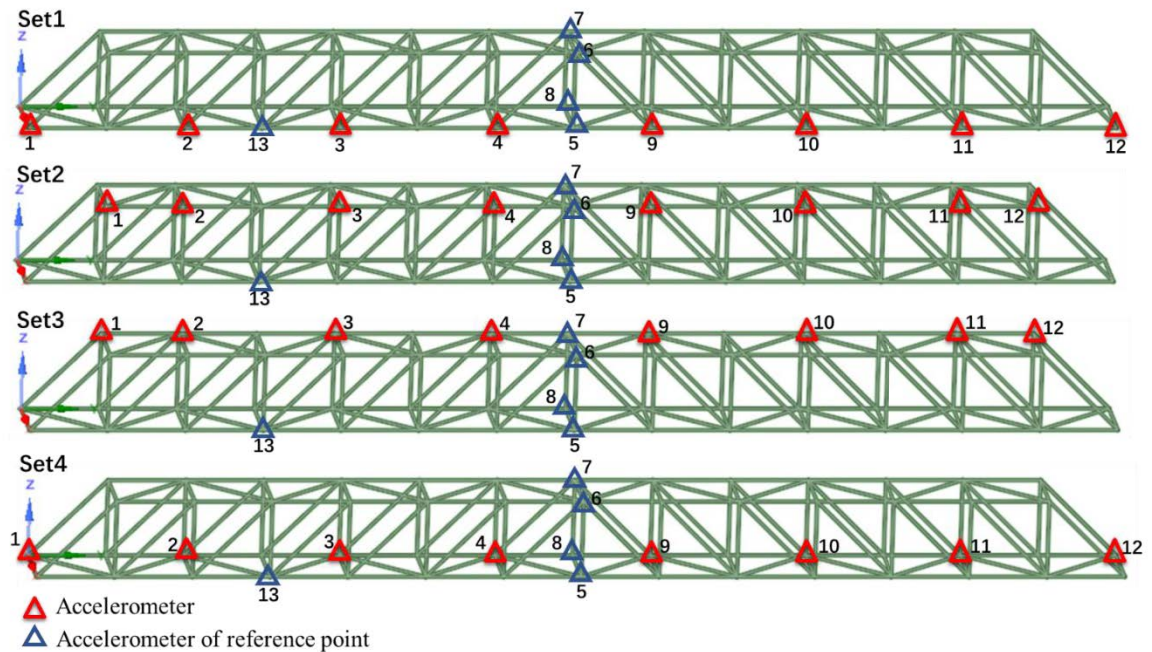


Figure 3.7 Accelerometer locations of four set measurements

### 3.3.3 Ref-SSI based spatial mode shapes extraction using multiple set measurements

Spatial mode shape was extracted from four set measurements using the reference-based stochastic subspace identification (Ref-SSI) method (Peeters & De Roeck, 2000). Firstly, the stochastic subspace identification (SSI) method was conducted on each set of measurements to obtain corresponding natural frequencies and local mode shapes of each edge. Then, with Ref-SSI method, spatial mode shapes were obtained by combining the mode shapes of four edges through rescaling.

In this study, ARTeMIS Modal 6.0 was used to extract natural frequencies and mode

shapes of each longitudinal edge from Measurement sets 1~4. Figure 3.8 shows the measurement set in ARTeMIS. For each set, there are 13 accelerometers as shown in Figure 3.7. The stabilization diagrams of four measurement sets were obtained as shown in Figure 3.9. From the figure, the natural frequencies of four sets were extracted and listed in Table 3.1. The frequencies of the bridge were obtained by averaging the power spectra of four edges. The first six modes were obtained as shown in Table 3.2, including three bending modes and three torsional modes.

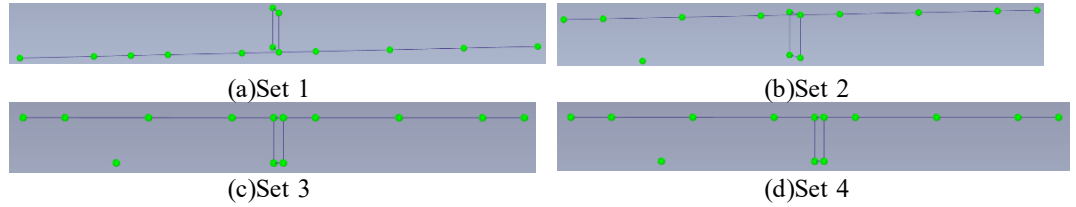


Figure 3.8 Sets modeling in ARTeMIS

With the SSI method, local mode shapes of each edge could also be extracted. The Reference based SSI method is then applied to transfer the local mode shapes of four edges into spatial mode shapes of the bridge. A rescaling procedure is conducted as (Amador & Brincker, 2021),

$$\mathbf{\Phi}_i = \frac{\Phi_{ref}}{\varphi_{i,ref}} \boldsymbol{\varphi}_i (i = 1,2,3,4) \quad (3.9)$$

where  $\boldsymbol{\varphi}_i$  and  $\mathbf{\Phi}_i$  is the local and global mode shape vector of Set  $i$ ;  $\varphi_{i,ref}$  is the local mode shape value of Set  $i$  at the reference point.  $\Phi_{ref}$  is the mode shape value at the reference point of the selected set.  $\Phi_{ref}$  could be taken from a reference point at any sets. However, there is only one  $\Phi_{ref}$  for each global mode shape. For example, for Mode 1, the  $\Phi_{ref}$  could be from Set 3, thus all sets would be rescaled according to the mode shape

value at the reference point from Set 3. Since the reference points could be any locations that remain at their original location during the whole measuring process, the No.13 accelerometer and the accelerometers (No.5, 6, 7 and 8) at the middle span could all be considered as reference points in this study.

Table 3.3 is the modal assurance criterion (MAC) matrix of these spatial mode shapes and Figure 3.10 presents the six spatial mode shapes. The diagonal values of the MAC matrix are close to one while the values at other locations are very small, and that indicates no coupling effect between these mode shapes.

Table 3.1 Natural frequencies of four sets (Hz)

Set 1	Set 2	Set 3	Set 4	Average
<b>14.63</b>	<b>14.34</b>	<b>14.61</b>	<b>14.55</b>	<b>14.6</b>
<b>23.44</b>	<b>23.18</b>	<b>23.37</b>	<b>23.40</b>	<b>23.4</b>
<b>39.50</b>	<b>39.54</b>	<b>39.55</b>	<b>39.38</b>	<b>39.5</b>
44.80	45.15	45.02	44.61	44.9
<b>45.78</b>	<b>45.70</b>	<b>45.73</b>	<b>45.52</b>	<b>45.7</b>
<b>57.96</b>	<b>57.65</b>	<b>57.83</b>	<b>57.86</b>	<b>57.8</b>
67.04	67.19	66.83	66.85	67.0
<b>72.01</b>	<b>71.69</b>	<b>72.36</b>	<b>71.74</b>	<b>71.9</b>

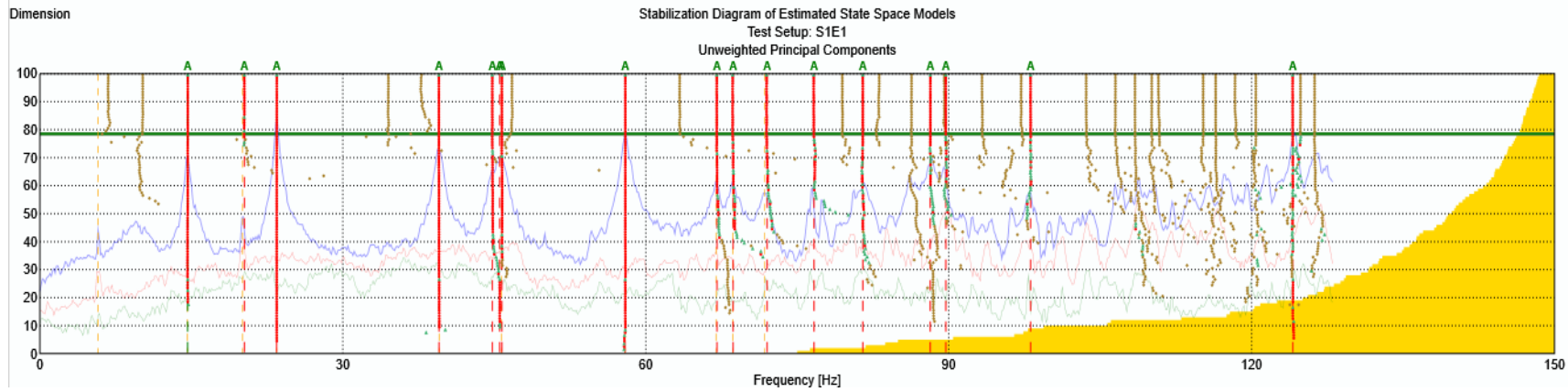
Table 3.2 Natural frequencies of the experimental truss bridge

Mode No.	1	2	3	4	5	6
Mode type	Bending	Torsional	Torsional	Bending	Torsional	Bending
Natural frequencies (Hz)	14.6	23.4	39.5	45.7	57.8	71.9

Table 3.3 MAC matrix of experimental mode shapes

	14.6 Hz	23.4 Hz	39.5 Hz	45.7 Hz	57.8 Hz	71.9 Hz
14.6Hz (Bending mode 1)	1.00	0.00	0.00	0.00	0.00	0.00
23.4Hz (Torsional mode1)	0.00	1.00	0.00	0.00	0.00	0.00
39.5Hz(Torsional mode 2)	0.00	0.00	1.00	0.00	0.00	0.00
45.7Hz (Bending mode 2)	0.00	0.00	0.00	1.00	0.00	0.00
57.8Hz(Torsional mode 3)	0.00	0.00	0.00	0.00	1.00	0.00
71.9Hz (Bending mode 3)	0.00	0.00	0.00	0.00	0.00	1.00





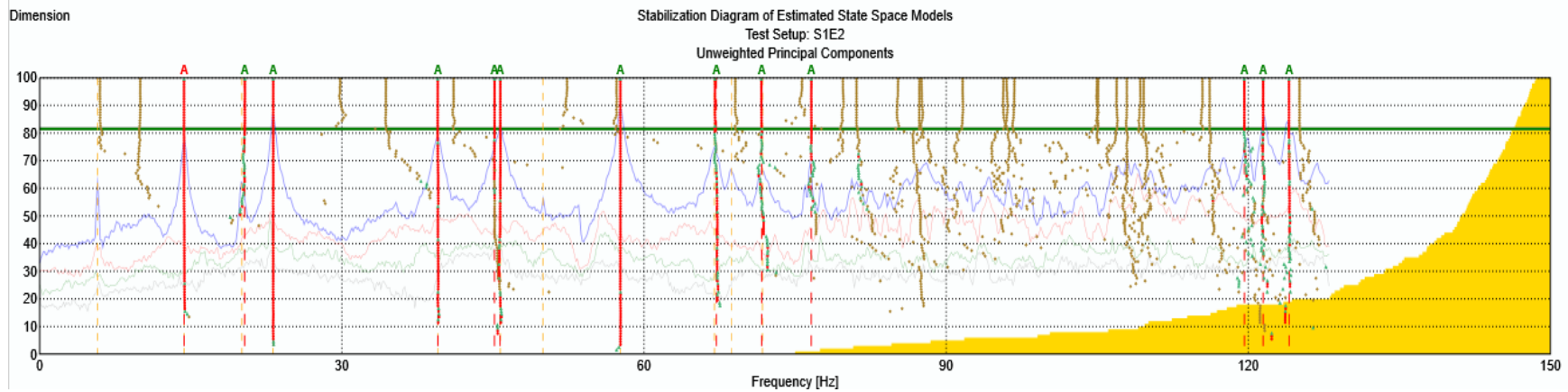
**State Space Dimensions**  
 Cursor Model = 79  
 Max. Eigenvalues = 79

**Indicators**  
 ■ Cursor Model  
 ■ Max. Eigenvalues  
 ■ Selection  
 ■ SVD Subspace  
 ● Stable Mode  
 ▲ Unstable Mode  
 ◆ Noise Mode

**Mode Markers**  
 - - - Current Estimator  
 - - - Other Estimators

**Lines**  
 — SVD Line No. 1  
 — SVD Line No. 2  
 — SVD Line No. 3

(a) Set 1



**State Space Dimensions**  
 Cursor Model = 82  
 Max. Eigenvalues = 82

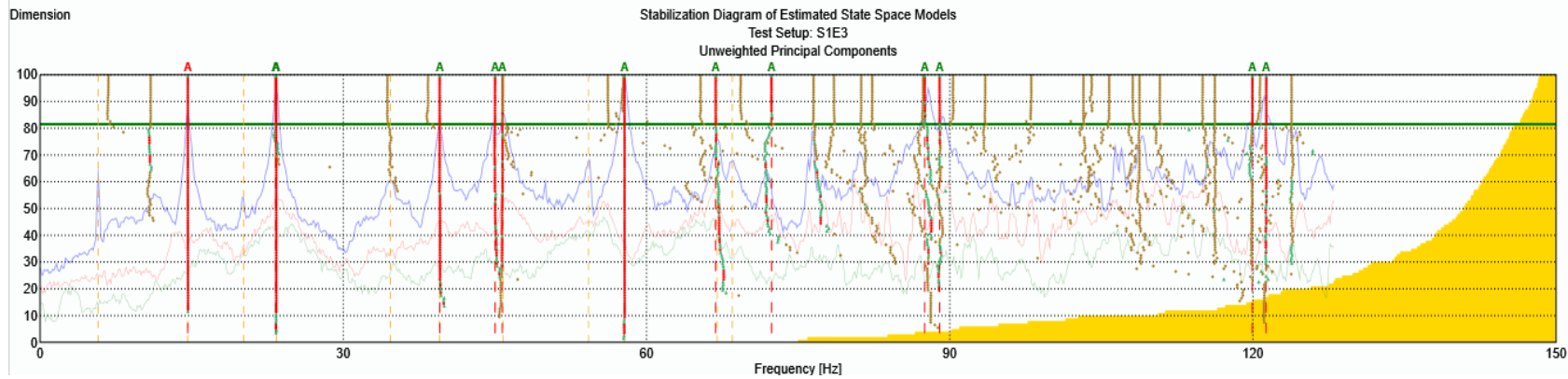
**Indicators**  
 ■ Cursor Model  
 ■ Max. Eigenvalues  
 ■ Selection  
 ■ SVD Subspace  
 ● Stable Mode  
 ▲ Unstable Mode  
 ◆ Noise Mode

**Mode Markers**  
 - - - Current Estimator  
 - - - Other Estimators

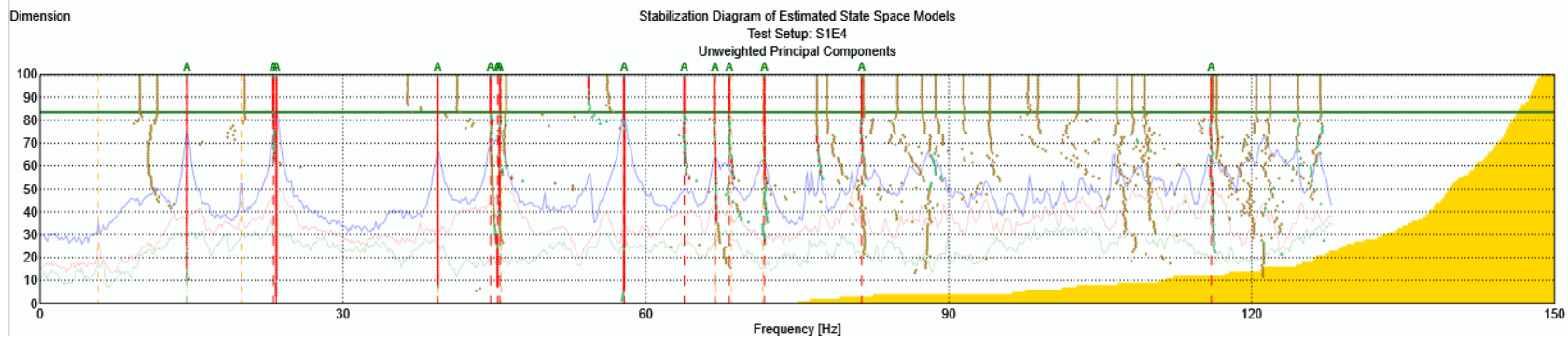
**Lines**  
 — SVD Line No. 1  
 — SVD Line No. 2  
 — SVD Line No. 3

(b) Set 2





(c) Set 3

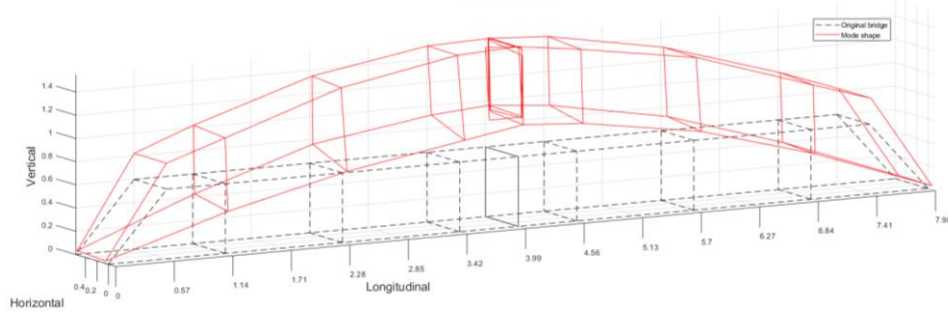


(d) Set 4

Figure 3.9 The stabilization diagram of four sets

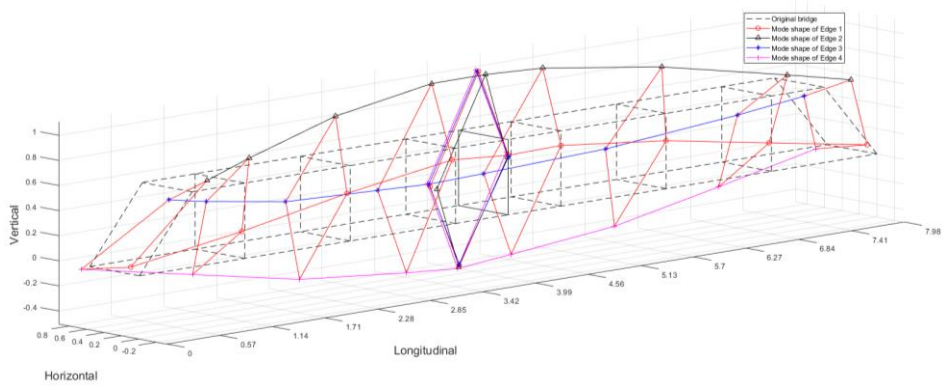
Experimental mode shape

Mode shape 14Hz



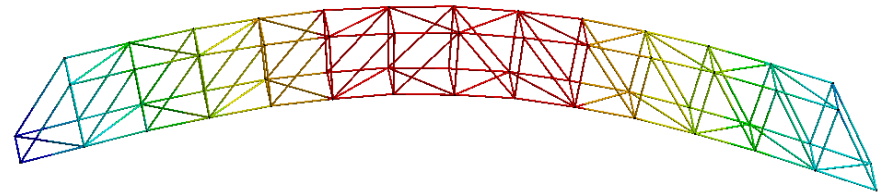
Bending mode 1. Frequency 14.0Hz

Mode shape 23Hz

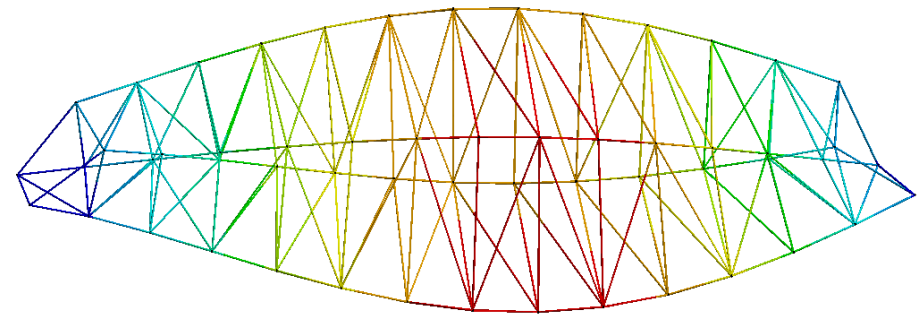


Torsional mode 1. Frequency 23.0Hz

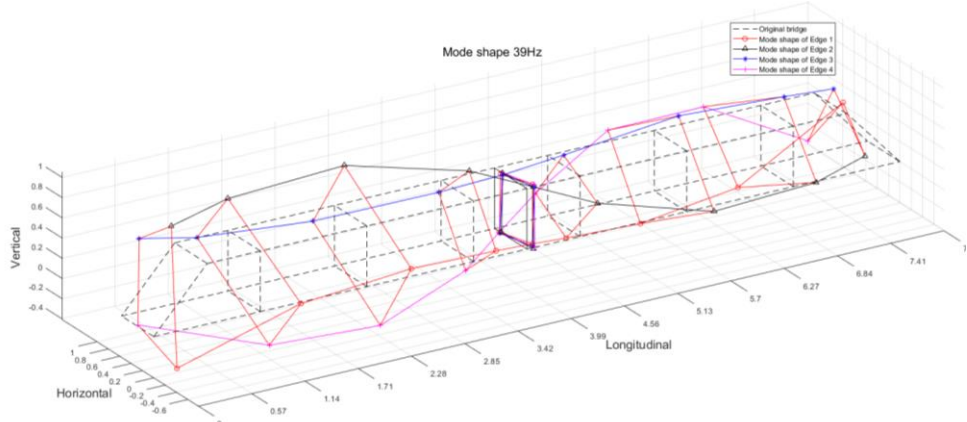
Numerical mode shape



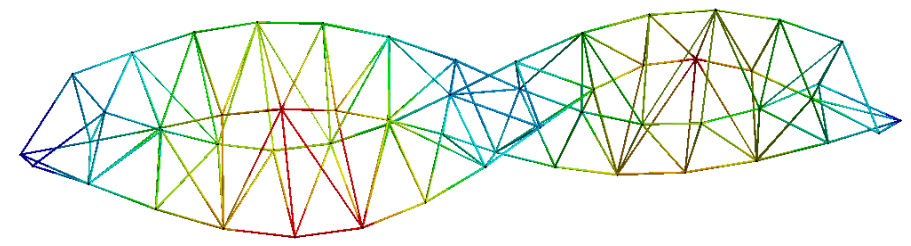
Bending mode 1. Frequency 18.6Hz



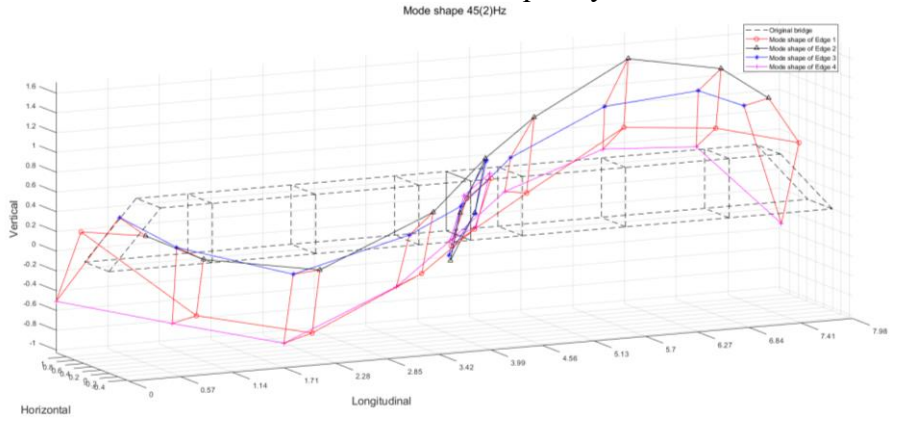
Torsional mode 1. Frequency 47.5Hz



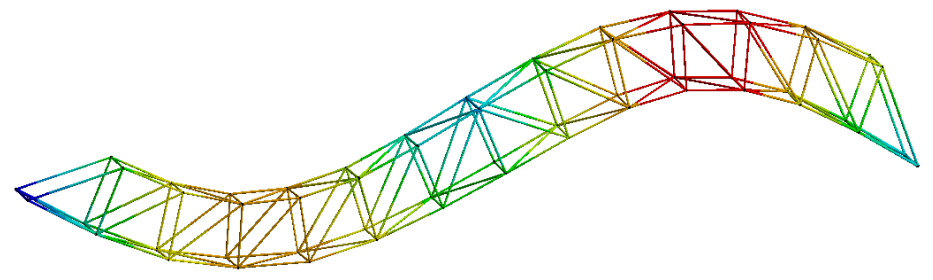
Torsional mode 2 Frequency 39.0Hz



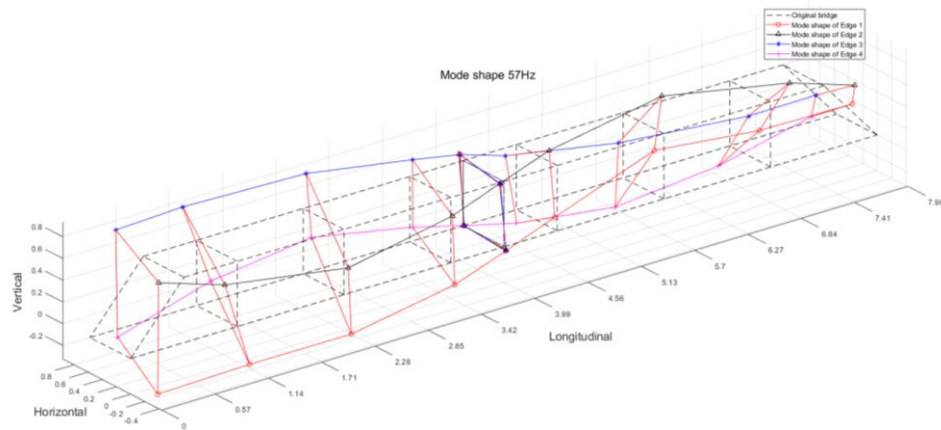
Torsional mode 2 Frequency 80.9Hz



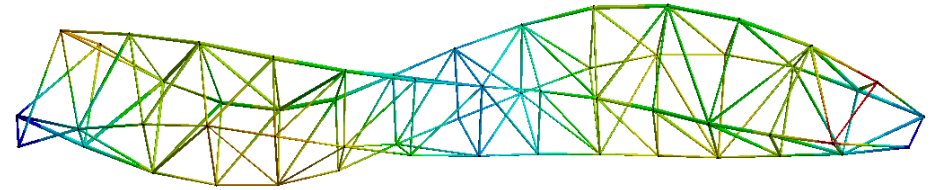
Bending mode 2. Frequency 45.7Hz



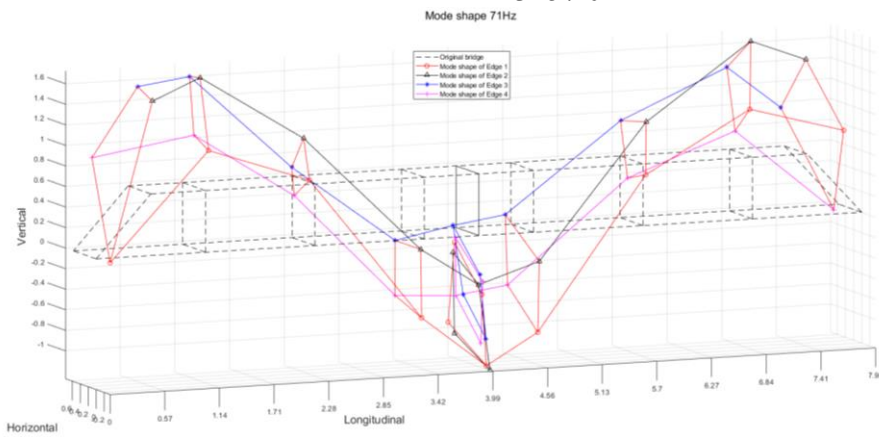
Bending mode 2. Frequency 61.0Hz



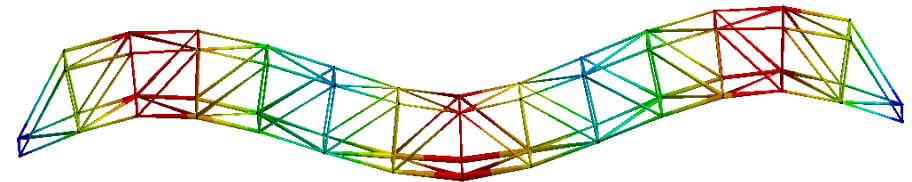
Torsional mode 3. 57.0Hz



Torsional mode 3. Frequency 94.5Hz



Bending mode 3. Frequency 71.0Hz



Bending mode 3. Frequency 112.1Hz

Figure 3.10 Spatial mode shapes of numerical and experimental models

### 3.4. NUMERICAL SIMULATION AND MODEL UPDATING

#### 3.4.1 Numerical model

Figure 3.11 is the finite element model (FE model) of the bridge established using ANSYS 19.0. The left end of the bridge is constrained for displacements in all directions (X, Y, and Z), and the right end is constrained in X, Z directions and free in Y longitudinal direction. The bridge model members are simulated by Beam4 element with uniform circular tube section. Details of the structure and Beam4 element are listed in Table 3.4. All joints are modelled as rigid joints in this section.

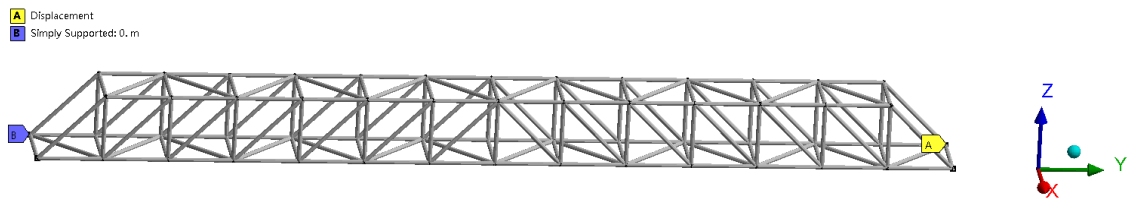


Figure 3.11 Finite element model of the bridge

Table 3.4 Details of the model and the cross section of beam4 element

Properties	Structural		Tube cross section			
	Young's modulus	Density ( $kg/m^3$ )	Outer radius	Inner radius	Area ( $m^2$ )	$I_{YY}$ and $I_{ZZ}$ ( $m^4$ )
Value	$2 \times 10^{11} Pa$	7800	0.009m	0.006m	$1.41361 \times 10^{-4}$	$4.134 \times 10^{-9}$

#### 3.4.2 Comparisons between numerical and experimental results

Figure 3.10 shows a comparison of the first six mode shapes by the numerical model together with the corresponding experimental results. Here the numerical model is with the rigid joints. From the figure, the mode shapes from the numerical model match well with the corresponding experimental results. Table 3.5 lists the natural frequencies of the numerical and experimental models and the corresponding MAC values. Table 3.6

presents the MAC values of the mode shapes from the numerical model and Table 3.7 is the MAC values between the experimental and numerical mode shapes. The MAC values at non-diagonal locations are very small in both Tables 3.6 and 3.7. The results shows that there is no coupling effect for the numerical mode shapes as well as experimental mode shapes. From Table 3.5, the torsional frequencies of the numerical model have high relative differences from 63% to 99% compared with the experimental results. There are low relative differences for bending modes, ranging from 27% to 56%. The main reasons are that 1) all joints are simulated as rigid while they are semi-rigid joints. 2) the uniform beam element is used while the real beam element composes of three segments connected by screws. These two factors will be updated in the simulation model in the following section.

Table 3.5 Natural frequency and mode shapes comparison between experimental and numerical model

Experimental Model		Numerical model				
Mode	$f$ (Hz)	Mode Order	$f$ (Hz)	Errors		MAC
				A*	R*	
Bending mode1	<b>14.6</b>	N2	<b>18.5</b>	<b>3.9</b>	<b>27%</b>	<b>0.99</b>
Torsional mode1	23.4	N5	46.2	22.8	97%	0.54
Torsional mode 2	39.5	N7	78.5	39	99%	0.59
Bending mode 2	<b>45.7</b>	N 6	<b>61.0</b>	<b>15.3</b>	<b>33%</b>	<b>0.67</b>
Torsional mode 3	57.8	N 9	94.2	36.4	63%	0.66
Bending mode 3	<b>71.9</b>	N 15	<b>112.2</b>	<b>40.3</b>	<b>56%</b>	<b>0.82</b>

$f$  Natural Frequencies; A\* Absolute error; R\* Relative error;

Table 3.6 MAC matrix of numerical mode shapes

MAC	18.6Hz	46.2Hz	78.5Hz	61.0Hz	94.2Hz	112.2Hz
18.6Hz (Bending mode 1)	<b>1.00</b>	0.00	0.01	0.00	0.00	0.03
46.2Hz (Torsional mode1)	0.00	<b>1.00</b>	0.00	0.00	0.00	0.00
78.5Hz (Torsional mode 2)	0.01	0.00	<b>1.00</b>	0.00	0.00	0.00
61.0Hz (Bending mode 2)	0.00	0.00	0.00	<b>1.00</b>	0.00	0.00
94.2Hz (Torsional mode 3)	0.00	0.00	0.00	0.00	<b>1.00</b>	0.00
112.2Hz (Bending mode 3)	0.03	0.00	0.00	0.00	0.00	<b>1.00</b>

Table 3.7 MAC matrix of experimental and numerical mode shapes

Modes and frequencies	18.6Hz	46.2Hz	78.5Hz	61.0Hz	94.2Hz	112.2Hz
14.6Hz (Bending mode 1)	<b>0.99</b>	0.00	0.00	0.01	0.00	0.03
23.4Hz (Torsional mode1)	0.00	<b>0.54</b>	0.01	0.00	0.00	0.00
39.5Hz (Torsional mode 2)	0.00	0.01	<b>0.59</b>	0.00	0.06	0.00
45.7Hz (Bending mode 2)	0.00	0.01	0.02	<b>0.67</b>	0.00	0.00
57.8Hz (Torsional mode 3)	0.00	0.00	0.11	0.00	<b>0.66</b>	0.00
71.9Hz (Bending mode 3)	0.02	0.00	0.00	0.00	0.00	<b>0.82</b>

### 3.4.3 Model updating

In this section, the finite element model updating has been conducted in two-steps. In the first step, the beam element stiffness is updated to match the bending modal frequencies of experimental and numerical models. In the second step, both the beam elemental and joint stiffness are updated to match the torsional frequencies. Finally, the updated elastic modulus value was validated using a simply supported three-segment beam.

- *Beam elemental stiffness updating*

As shown in Figure 3.4, the bridge beam components are nonuniform three segment beam elements while an equivalent uniform beam element is used in the numerical model. This effect could be considered by elastic modulus reduction. Frequency based objectives are considered, comparing natural frequency matching of the experimental and numerical models. Three objectives were investigated as defined below,

Objective 1 considers all six modes as:  $f_1 = \sum \left| \frac{f^{E_i} - f^{N_i}}{f^{E_i}} \right|_{i=1,2,3,4,5,6}$ , where  $E_i$  is the  $i^{\text{th}}$  mode of experimental bridge and the  $N_i$  is the numerical mode that matches to the  $i^{\text{th}}$  experimental mode.

Objective 2 considers three bending modes as:  $f_2 = \sum \left| \frac{f^{E_i} - f^{N_i}}{f^{E_i}} \right|_{i=1,4,6}$ , where Mode 1, 4 and 6 are the bending modes.

Objective 3 considers three torsional modes as:  $f_3 = \sum \left| \frac{f^{E_i} - f^{N_i}}{f^{E_i}} \right|_{i=2,3,5}$ , where Mode 2, 3 and 5 are the torsional modes.

Figure 3.12 is the effect of elastic modulus changes on the three objective functions. The optimum elastic modulus value is different from the objective function. The objective function  $f_2$  has the optimum  $E = 1.13 \times 10^{11} Pa$ . Table 3.8 shows the mode matching of the experimental and numerical models before and after element stiffness updating. From the results, the bending modes match well with the experimental results after the model updating using the objective function  $f_2$ . The relative differences for the first and second bending modes are 4.3% and -0.3% respectively. Moreover, the corresponding value at MAC matrix remains the same after updating.

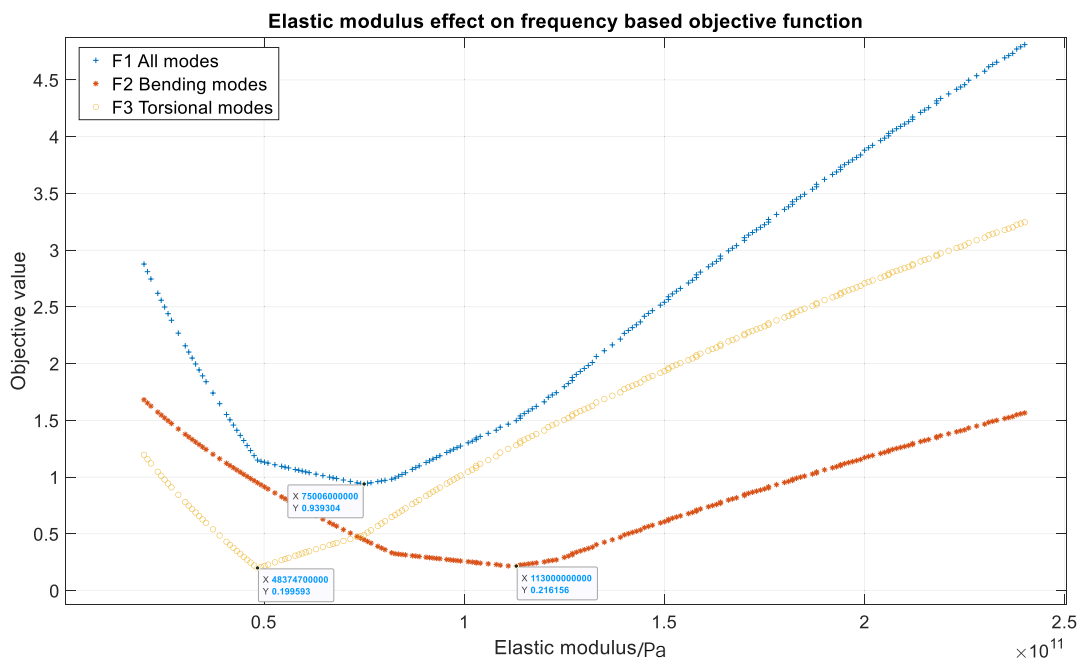


Figure 3.12. Elastic modulus effect on frequency based on three objective functions



Table 3.8 The mode matching of the experimental and numerical models before and after updating

element stiffness

Experimental Model	Numerical model							
	Before updating ( $E=2 \times 10^{11} Pa$ )				After updating ( $E=1.13 \times 10^{11} Pa$ )			
Mode	$f^*$ (Hz)	$f^*$ (Hz)	A*	R*	$f$ (Hz)	A*	R*	MAC
<b>Bending mode1</b>	<b>14.6</b>	<b>18.5</b>	<b>3.9</b>	<b>27%</b>	<b>13.98</b>	<b>0.62</b>	<b>4.3%</b>	<b>0.99</b>
Torsional mode1	23.4	46.2	22.8	97%	35.70	-12.3	-52.6%	0.54
Torsional mode 2	39.5	78.5	39	99%	60.48	-21.0	-53.1%	0.59
<b>Bending mode 2</b>	<b>45.7</b>	<b>61.0</b>	<b>15.3</b>	<b>33%</b>	<b>45.85</b>	<b>-0.15</b>	<b>-0.3%</b>	<b>0.67</b>
Torsional mode 3	57.8	94.2	36.4	63%	71.03	-13.2	-22.9%	0.66
<b>Bending mode 3</b>	<b>71.9</b>	<b>112.2</b>	<b>40.3</b>	<b>56%</b>	<b>84.25</b>	<b>-12.4</b>	<b>-17.2%</b>	<b>0.82</b>

Note:  $f^*$  is Natural Frequency; A\* is Absolute error; R\* is Relative error;

- *Joint stiffness updating*

Although bending frequencies saw better matching, torsional frequency matching remains poor, with around 50% frequency differences with the experimental mode for first two torsional modes. This is because all joints are simulated as rigid while they are semi rigid joints. To include this effect, elemental moment of inertia ( $I_{YY}$ ) is updated. It is found that the semi rigid joint consideration has significant effect on the torsional mode frequencies. By manually adjusting, it is found that when  $E^* = 0.62 \times E = 1.24 \times 10^{11} Pa$  (element stiffness updating) and  $I_{YY}^* = 0.27 \times I_{YY} = 1.12e^{-9} Pa$  (semi rigid joint updating), frequencies of all experimental and numerical modes match well, except the torsional mode 2 shows a 40% relative error. Detailed comparison results are given in Table 3.9. It is worth noting that different from element stiffness updating, the semi-rigid joint updating is conducted through manual adjusting rather than optimization methods.

- *Validation of the updated elastic modulus*

To verify rationality of the updated  $E$  as  $1.24 \times 10^{11} Pa$ , a simply supported beam with three segments was built with line pressure of  $1000N/m$ , as shown in Figure 3.13. This three-segments-beam was considered as the real bar element situation where a hollow bar is in the middle with two end beams. It is worth noting the moment of inertia of three-segments-beam differs that of the single beam, thus different bending displacement were obtained. The idea is calculating the elastic equivalent modulus through matching their maximum displacements. Table 3.10 shows the related parameters, and the equivalent elastic modulus result is  $1.4446 \times 10^{11} Pa$  which is close to the final updating results of  $E = 1.24 \times 10^{11} Pa$ .

Table 3.9 The mode matching of the experimental and numerical models after updating member and joint stiffness

Experimental Model	Numerical model								
	Before updating ( $E=2 \times 10^{11} Pa$ )					After updating ( $E^*=1.24 \times 10^{11} Pa$ and and $IYY^* = 1.12 \times 10^{-9} m^4$ )			
Mode	$f^*$ (Hz)	$f^*$ (Hz)	A*	R*	MAC	$f^*$ (Hz)	A*	R*	MAC
Bending mode 1	14.6	18.5	3.9	27%	0.99	14.6	0	0	0.99
Torsional mode 1	23.4	46.2	22.8	97%	0.54	23.7	0.3	1.2%	0.68
Torsional mode 2	39.5	78.5	39	99%	0.59	55.5	16	40%	0.55
Bending mode 2	45.7	61.0	15.3	33%	0.67	46.1	0.03	0.06%	0.67
Torsional mode 3	57.8	94.2	36.4	63%	0.66	66.6	-8.79	15%	0.71
Bending mode 3	71.9	112.2	40.3	56%	0.82	69.7	-2.17	-0.03%	0.57

Note:  $f^*$  : Natural Frequencies; A\*: Absolute error; R\* : Relative error.

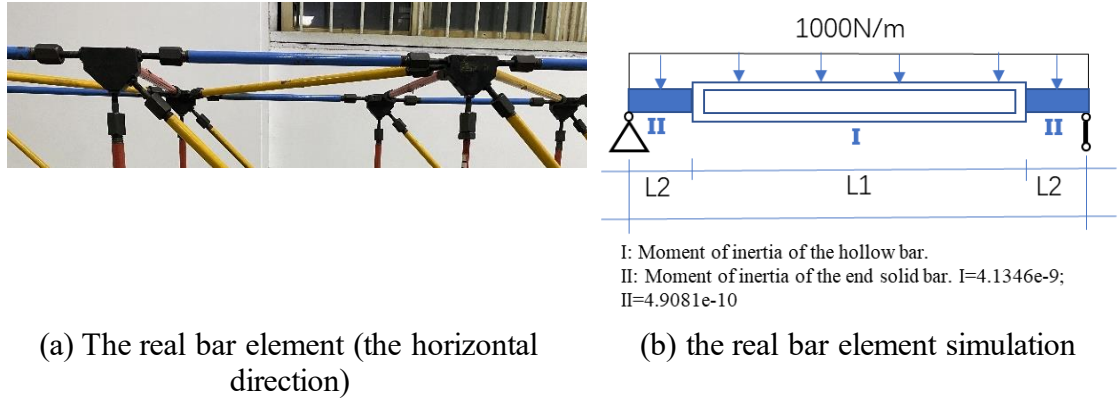


Figure 3.13 The real bar element and its simulation

Table 3.10 Equivalent elastic modulus of bars in three directions

Total length(m)	L1(m)	L2(m)	Maximum Displacement		Equivalent elastic modulus (EE)
			Single beam(m)	Three-segments-beam(m)	
0.6	0.4	0.1	0.00204	0.00283	$1.4446 \times 10^{11} Pa$

### 3.5. Structural damage detection using the updated model

As previously discussed, the model is updated in two aspects, member and joint stiffnesses. To further verify the performance of the proposed model, structural damage detection using the rigid or semi-rigid joint models were compared.

#### 3.5.1 Scenarios description

As shown in Figure 3.14, the elements were numbered along the bridge, with Element 1 at the left end, Element 160 at right end and Element 80 at the middle span. The damage detection using semi-rigid joint and rigid models were compared considering the beam element damage of three scenarios: diagonal beam damage (Element 36), vertical beam damage (Element 104), double beam damage (Elements 36 and 104), each with three damage severity levels (35%, 60% and 83%), as the scenarios design shown in Table 3.11.

The beam damage is simulated by the elastic modulus reduction.

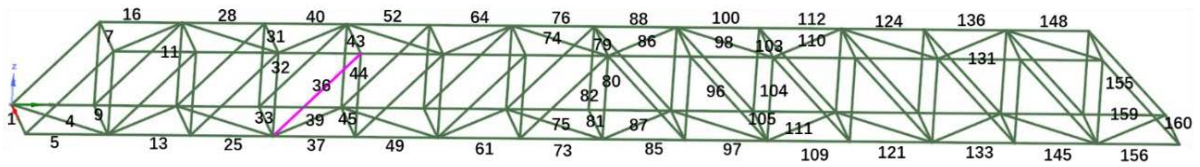
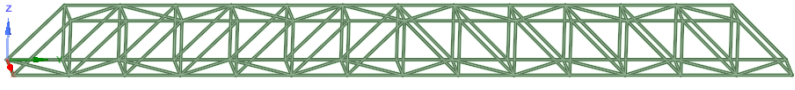
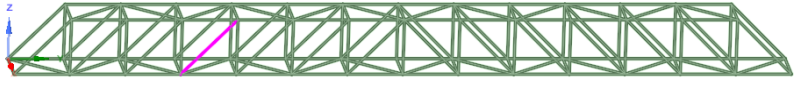
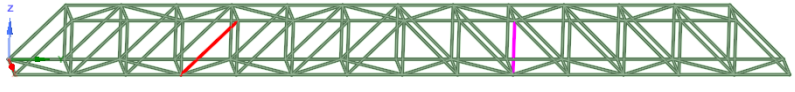
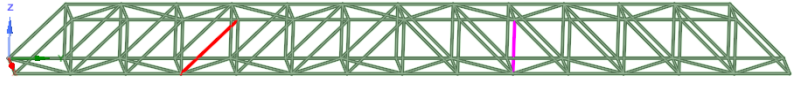


Figure 3.14 Element number

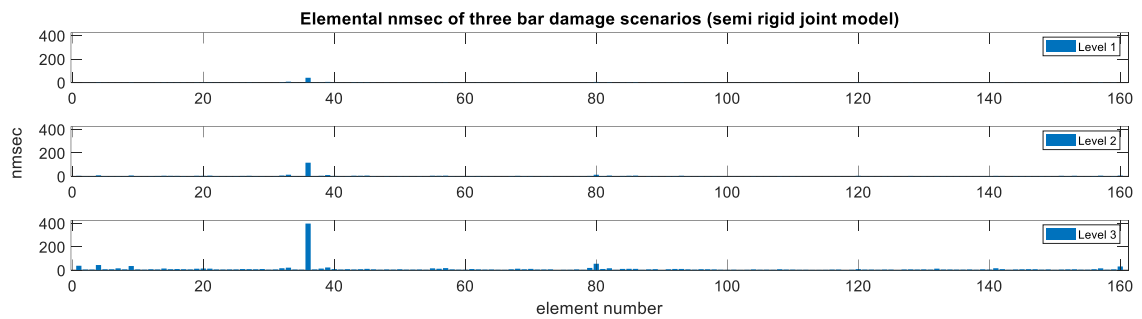
Table 3.11 Scenarios design

Damage scenarios	Damage degrees		Location
	Bar 36	Bar 104	
Healthy	-	-	
Diagonal member damage	35% 60% 83%	-	
Vertical member damage	-	35% 60% 83%	
Double damage	83%	35% 60% 83%	

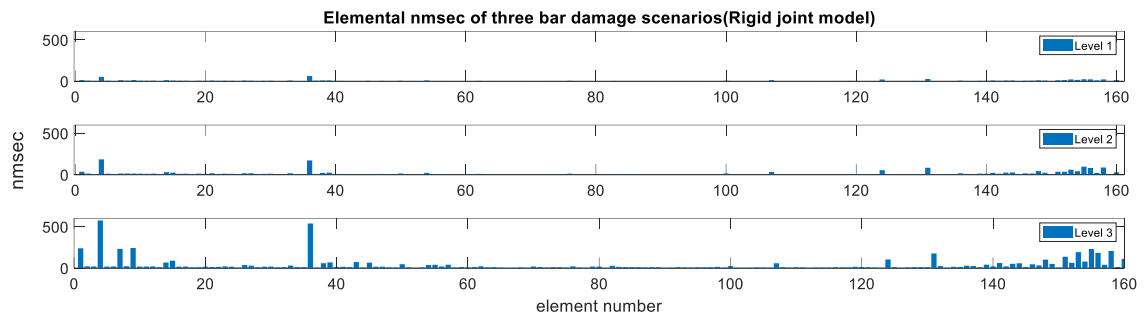
### 3.5.2 Structural damage detection using the proposed model

Elemental damage index  $NMSEC$  of three damage scenarios including, 1) damage of diagonal beam element 36; 2) vertical beam element 104; 3) double damage scenarios using the semi-rigid and rigid joint models were calculated, as shown in Figures 3.15, 3.16 and 3.17. Figure 3.15 shows the  $NMSEC$  values of the structure with the damage in a diagonal member using the semi-rigid and rigid joint models. The  $NMSEC$  values of the structure with the vertical beam damage and the double damage using the semi-rigid and rigid joint models were shown in Figures 3.16 and 3.17 respectively. From Figure 3.15(a), the damaged beam is identified clearly for diagonal element damage scenario using the

semi-rigid joint model. Also, the *NMSEC* value is increased with the damage level. Figure 3.15(b) shows the results using the rigid joint model and there are large errors at the two ends of the bridge. The similar observations are also obtained from Figures 3.16 and 3.17. As shown in Figure 3.17, the damage in the diagonal member is more clearly identified than that of the vertical member. The damage could be identified clearly using the semi-rigid joint model compared with the rigid joint model.

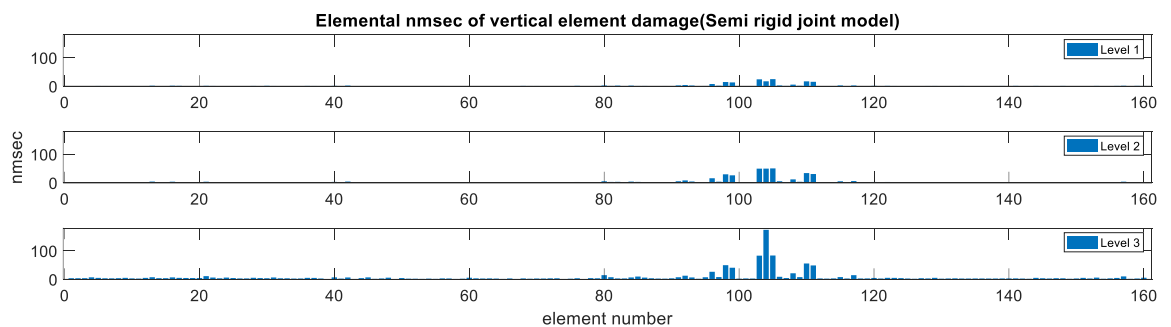


(a) Semi rigid joint model

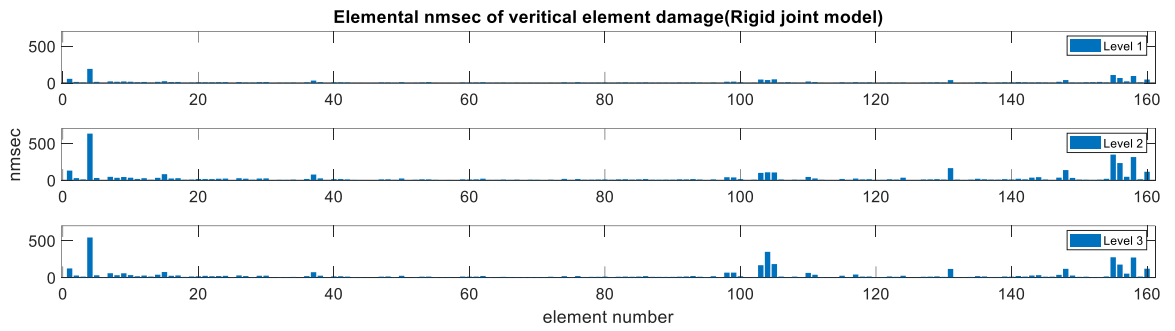


(b) Rigid joint model

Figure 3.15 elemental *NMSEC* of diagonal bar damage

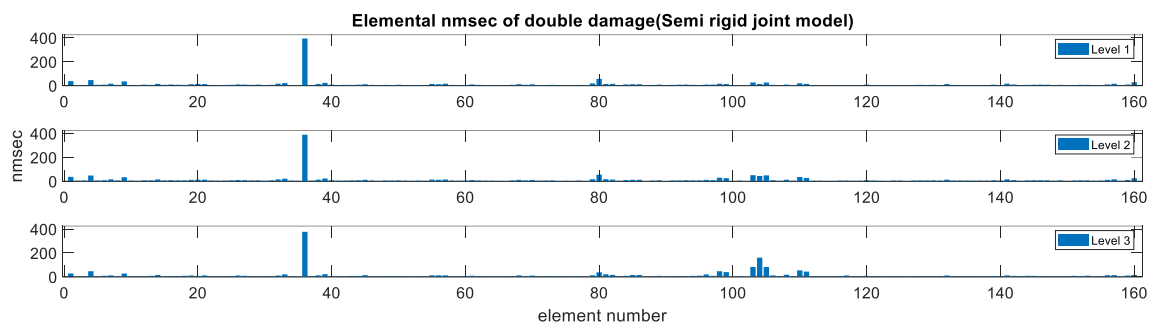


(a) Semi rigid joint model

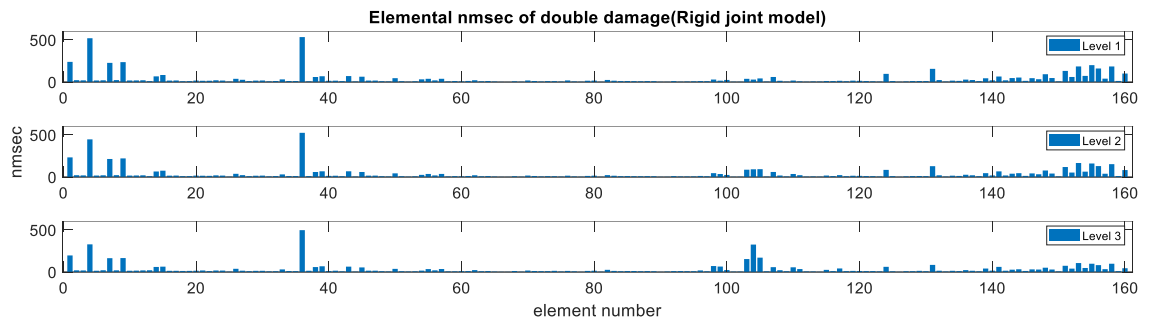


(b) Rigid joint model

Figure 3.16 Elemental *NMSEC* of vertical bar damage



(a) Semi rigid joint model

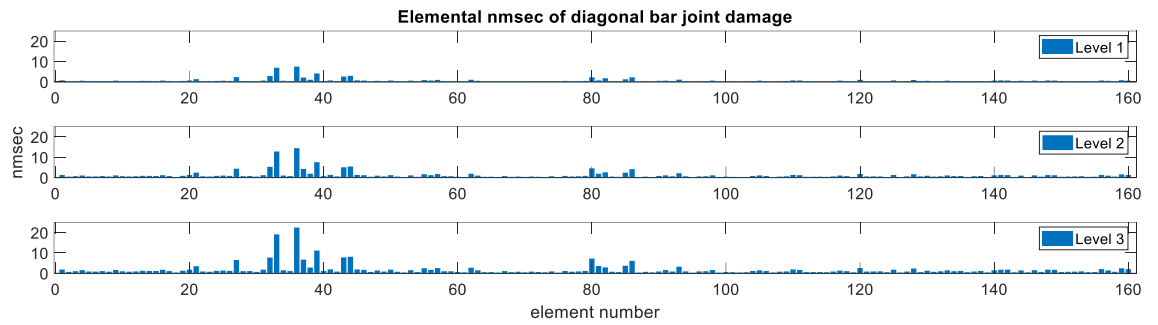


(b) Rigid joint model

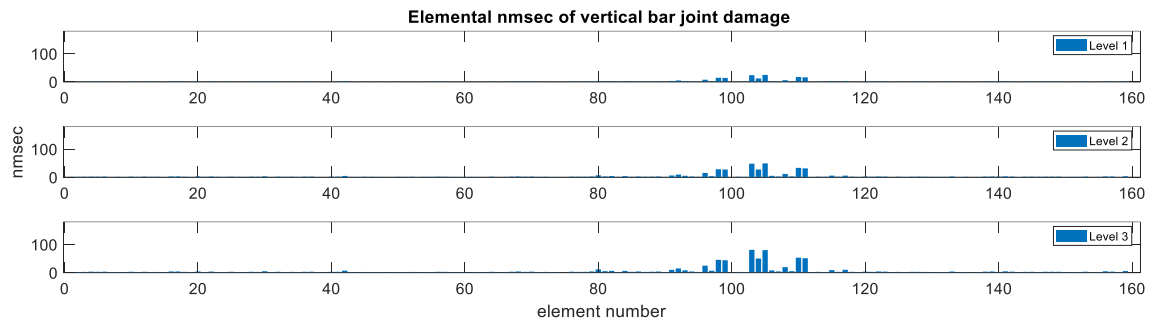
Figure 3.17 elemental *NMSEC* of double bar damage scenarios

### 3.5.3. Joint damage detection performance (semi rigid joint model)

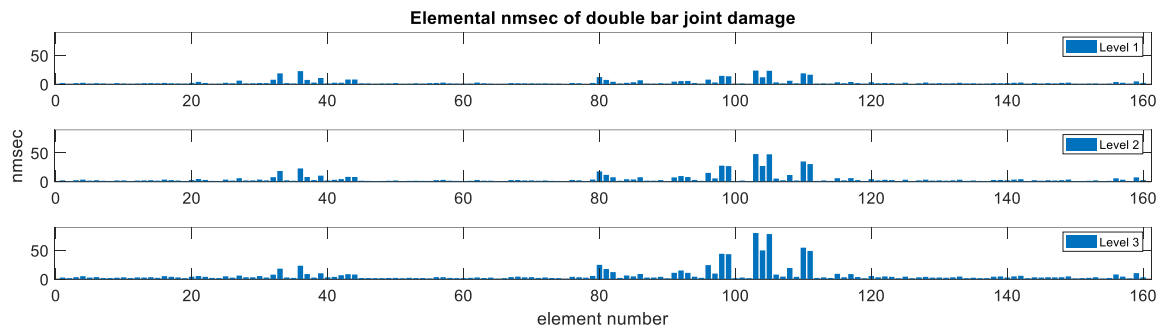
This section investigates the joint damage detection of the frame structure. All damage scenarios and damage severities are simulated as Section 3.5.1 except that the joint damage is simulated by reduction of  $I_{YY}$ . For example, 60% joint damage at location 36 means that the  $I_{YY}$  of Element 36 is  $0.4 \times (0.27 \times I_{YY})$ .



(a) joint damage of diagonal bar 36



(b) joint damage of vertical bar 104



(c) double joint damage on bar 36 and bar 104

Figure 3.18 *NMSEC* of double joint damage scenario

The *NMSEC* values for the structure with three joint damage scenarios are shown in Figure 3.18. Unlike beam damage, the joint damage inevitably influences nearby connected elements. From the figure, the joint damage could be detected for all scenarios (e.g. around 1/3 of the span in Figure 3.18 (b)). In Figure 3.18(a), Element 36 and its nearby elements (Elements 33-44) show a higher *NMSEC* value. Same observation could be obtained for the joint damage around the vertical beam (Figure3.18(b)). In the joint damage around the diagonal beam 36 (Figure 3.18(a)) and double joint damage (Figure

3.18(c)), some elements at the middle span (bar 80-86) also show slightly higher *NMSEC* values. In all scenarios, the *NMSEC* value increases with the damage level, and it could be used to indicate the severity of the joint damage.

### **3.6 SUMMARY**

A novel generic element for nonuniform beams with semi-rigid joints has been developed and the modal strain energy-based damage index has been proposed for structural damage detection. An 8-meter spatial frame bridge model has been built in laboratory. Numerical and experimental results show that the proposed method is reliable and accurate to detect the structural damage of the spatial frame structures. The following conclusions can be obtained,

- 1) Compared with the existing methods with the rigid joint model, the damage of beam members can be identified accurately using the semi-rigid joint model.
- 2) The results show that the damage in diagonal members is easier to detect than that of vertical elements. The member damage is more easily detected than joint damage. This is because joint damage usually influences nearby members, leading to most nearby elements with high damage index values.
- 3) The proposed method can improve the damage detection accuracy for the complex spatial frame structures in practice.



# **CHAPTER 4 STRUCTURAL DAMAGE DETECTION USING PCA AND CONVOLUTIONAL NEURAL NETWORK**

## **4.1 OVERVIEW**

In this chapter, a 2D CNN is used to detect the damage in a numerical bridge model based on structural vibration information. CNNs were trained using the mode shapes and normalized modal strain energy-based index (*NMSEC*) respectively. The *NMSEC* features were conducted with PCA dimension reduction before inputted into the CNN. We intend to overcome the common challenges of the vibration based CNNs like high computational cost, poor robust ability and make corresponding suggestions.

## **4.2 METHODOLOGY**

### **4.2.1 Convolutional neural network (CNN) architecture for damage detection**

The CNN consists of multiple convolutional layers, activation layers, pooling layers, fully connected layers, and output layers. The following are brief descriptions of these layers.

- 1) Input layer.

The input layer could be vectors, matrices, images, or any other kinds of data. For damage detection problems, mode shapes, acceleration signals and images of the target structures are commonly used inputs.

- 2) Convolutional layer.

The convolutional layer, a key operation of the CNN, is a process of weights learning of kernels, with various kernels sliding over the input features. This process could also be simplified as  $\text{Output} = \text{kernel} \otimes \text{Input}$ , where  $\otimes$  represents the convolution operation. The depth of kernels is the same as the input, but the width and height are smaller. Each kernel produces an output feature and then all kernel features are stacked together and considered as the input of the next layer.

### 3) Activation layer.

The convolutional layer is usually followed by a nonlinear activation layer to enable a nonlinear mapping and efficient training. Commonly used activations include Sigmoid, Tanh, and more recently, Rectified linear unit (Relu). Relu is used in this chapter since it's faster and does not have the gradient vanishing effect (Wang et al., 2020).

### 4) Pooling layers.

Pooling is a process of compressing feature dimensions (width and height) by taking the maximum or average value in each input patch. The two most used pooling are Max-pooling and Average-pooling. It is worth noting that pooling is not a learning process but only a size reduction operation. It could reduce the number of parameters to learn and improve training speed.

### 5) Fully connected (FC) layers.

The fully connected layers are the same as in artificial neural networks. At this point, the

outputs are usually flattened into 1D dimensional vectors before going through the fully connected layers.

#### 6) Output layer.

The output layer constrains the network's output through the number of units and activation function. The unit number determines the output classes while the activation constrains the output value. For example, the sigmoid activation converts arbitrary values into the  $[0,1]$  interval, and the Relu activation zeros out negative values. In this study, the CNN output layer ended with a sigmoid activation to achieve damage quantification.

#### 7) Mini-batch.

Training an entire huge dataset at one time is slow and sometimes infeasible because of memory constraints. In such cases, the datasets could be split into small batches for calculating error and updating coefficients of the model. Each fixed size of samples is called a mini-batch and the fixed number of samples in each mini-batch is called the batch size. For each iteration, a small subset of the whole dataset was processed. The mini-batch technique could improve computational efficiency as could allow reaching the global minimum quickly in the cost function.

#### 8) Dropout.

Drop out is one of the most used techniques for reducing overfitting. When applying a dropout layer, the dropping out rate is required, representing the fraction of features that

are zeros out, and this rate is usually set between 0.2-0.5. After building the CNN, repeated tests are conducted to modify the model. During the training, overfitting problem might occur. This is when the model overfit to the training data, and in such cases, the validation accuracy degrades with the training process going. Reducing the network's size and regularization techniques are effective ways to avoid overfitting (Yang et al 2019).

To achieve better performances, hyper-parameters such as the learning rate, the unit number of each layer and dropping out rate could be tuned. These hyper parameters are usually decided based on empirical results or parameter studies.

#### 4.2.2 Loss and damage evaluation indices

The loss function is also called the objective function. It measures the error between the predicted value and the true value. The commonly used MSE loss function is as follows.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (4.1)$$

where  $Y$  and  $\hat{Y}$  are the true and predicted value of the  $i^{\text{th}}$  sample among the total  $n$  samples.

Another commonly used loss function is the mean absolute error ( $MAE$ ) of the difference between the predictions and the targets, as shown in Eq. (4.2).

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (4.2)$$

More often, the  $MAE$  is used as a training metric during the training process while  $MSE$  is used for testing the CNN performances. In this study, an updated  $MSE$  loss function  $UMSE$ , a training metric  $MAED$  and two testing performance evaluation indices  $DLA$  and

*MAEDD* were specifically developed for damage detection using a popular framework Pytorch.

#### 4.2.2.1 The UMSE loss function

For the damage detection problem, the training label of CNN is a matrix of which all locations are zeros except for some locations with specified damage degrees. Hence these locations in the training matrix could be divided into two categories: one called the intact region, the other called the damaged region. Figure 4.1 illustrates the intact regions (white) and damaged regions (blue) using an example target matrix, where there are 9 samples in total and three damage scenarios, three samples for each scenario. A loss function based on this location division could be designed as follows.

$$UMSE = \alpha \times MSEI + \beta \times MSED \quad (4.3)$$

where *MSEI* and *MSED* are the mean square error of the wrong detection in the intact region and the failure detection in the damaged region respectively.  $\alpha$  and  $\beta$  are the weighting factors for *MSEI* and *MSED* respectively. In detail, they could be calculated by the following equations.

$$MSEI = \frac{1}{m} \sum_{i=1}^m (Y_i - \hat{Y}_i)^2 \quad m = 1, 2 \dots ni \quad (4.4)$$

$$MSED = \frac{1}{h} \sum_{i=1}^h (Y_i - \hat{Y}_i)^2 \quad h = 1, 2 \dots nd \quad (4.5)$$

where *ni* and *nd* are the number of intact and damaged locations in the label. It is worth noting that the sum of  $\alpha$  and  $\beta$  is one as they represent weighting factors for the intact and

damaged regions which are the two components that make up the matrix of the training label. For example, after certain damage localization accuracy is reached, an assignment of  $\beta=0.8$  allows the network to focus on the iterations on damage degrees rather than wasting computation on damage locations. Moreover, an updated loss function with  $\alpha=\beta=0.5$  is just the same as the *MSE* loss function except they differ only in scale.

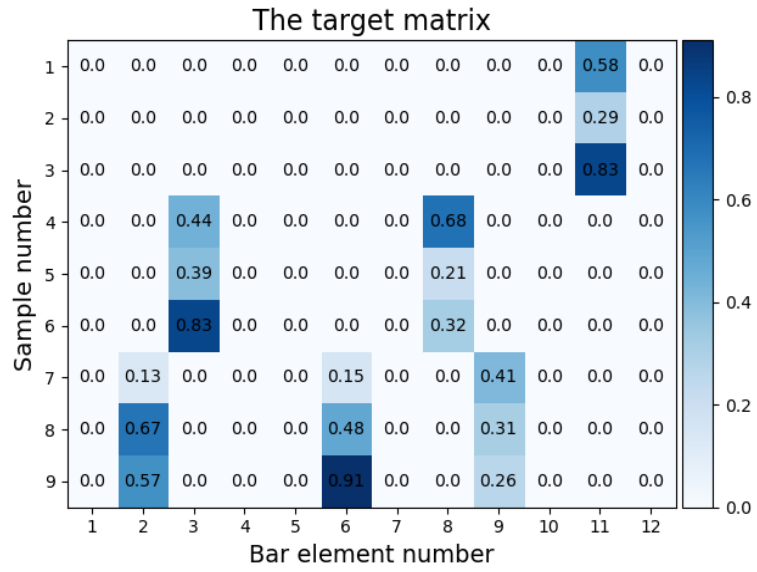


Figure 4.1 The example target matrix

#### 4.2.2.2 MAED training metric

The *MAED*, the Mean absolute error (*MAE*) of failure detection in the damaged region, was proposed and used as a training metric. Suppose there are  $m$  damage scenarios, each with  $N_m$  training samples and for each sample  $i$ , there are  $k$  damaged elements, the *MAED* is defined as follows.

$$MAD = \frac{1}{N_1 + 2N_2 \dots + kN_m} (\sum_{i=1}^{N_1} |T_{i1} - P_{i1}| + \sum_{i=1}^{N_2} \sum_{j=1}^2 |T_{ij} - P_{ij}| + \dots + \sum_{i=1}^{N_m} \sum_{j=1}^k |T_{ij} - P_{ij}|) \quad (4.6)$$

where  $T_{ij}$  and  $P_{ij}$  stand for the true and predicted damage degree of the  $j^{th}$  damaged member for the  $i^{th}$  sample; Basically,  $MAED$  is the mean absolute difference of the true and predicted matrix in the damaged region. While the  $MAE$  evaluates the prediction matrix, the  $MAED$  evaluates only the damage degrees in the damage region. Thus, the  $MAED$  is a more straightforward way of validating the model prediction accuracy on damage degrees.

#### 4.2.2.3 Damage localization and quantification indices: $DLA$ and $MAEDD$

It is of vital importance to fully evaluate the performance of the deep learning network on damage identification. The commonly used  $MSE$  and  $MAE$  indices do tell the damage detection accuracy as a whole but fail to indicate the accuracy of damage localization and quantification separately. To solve this problem, two indices, damage localization accuracy ( $DLA$ ) and  $MAE$  of damage degrees ( $MAEDD$ ) were proposed. The  $DLA$  is defined as follows:

$$DLA = \frac{R}{TN} \quad (4.7)$$

where  $R$  represents the number of samples of which the CNN outputs the correct locations of damaged members, while  $TN$  is the total number of testing samples. The definition of  $MAEDD$  is defined as follows.

$$MAEDD = \frac{1}{N_1} \left( \frac{1}{N_1} \sum_{i=1}^{N_1} |T_i - P_i| + \frac{1}{2N_2} \sum_{i=1}^{N_2} \sum_{j=1}^2 |T_{ij} - P_{ij}| + \dots + \frac{1}{kN_m} \sum_{i=1}^{N_m} \sum_{j=1}^k |T_{ij} - P_{ij}| \right) \quad (4.8)$$

Different from the testing  $MSE$  and  $MAE$ , the  $MAEDD$  evaluates only the damage degree prediction accuracy of damaged members. The smaller the  $MAEDD$ , the higher damage quantification accuracy the network achieves. To avoid the incorrectly identified members have impacts on both  $DLA$  and  $MAEDD$ , the calculation of  $MAEDD$  accounts only the correctly identified bars. With  $DLA$  and  $MAEDD$ , the overall performance of CNN on damage detection could be fully evaluated.

### 4.2.3 Mode shape based and modal strain energy-based damage index

This section introduces the commonly used features for structural damage detection, namely, mode shape and modal strain energy-based features. To get structural mode shapes, modal analysis is conducted. For free vibrations of the undamped structure, this could be represented by (Paz & Kim, 2019)

$$(K - \omega_i^2 M)\varphi_i = 0, i = 1 \dots \dots ndf \quad (4.9)$$

where  $K$  and  $M$  are the stiffness matrix and mass matrix.  $\omega_i$  is the  $i^{th}$  natural frequency and  $\varphi_i$  is the modal shape vector corresponding to the  $i^{th}$  frequency. In this study, the first six mode shapes of the designed damaged scenarios were extracted using the finite element software Ansys 19.0.

The modal strain energy of the  $e$ th structural element in mode  $i$  is defined as (Wang & Xu, 2019):

$$MSE_i^e = \frac{1}{2} \varphi_i^{eT} K^e \varphi_i^e, i = 1, \dots, ndf, e = 1, \dots, nte \quad (4.10)$$

where  $\varphi_i^e$  is the nodal displacement of the  $e$ th element in mode  $i$ , namely the mode shape.



$K^e$  is the stiffness matrix of the  $e$ th element. The total modal strain energy of the mode  $i$  could be calculated as:

$$MSE_i = \sum_{e=1}^{nte} MSE_i^e \quad i = 1, \dots, ndf, e = 1, \dots, nte \quad (4.11)$$

Using Eq.(4.11), the modal strain energy of  $e$ th element in  $i$ th mode is normalized by the modal energy summation of the  $e$ th element:

$$NMSE_i^e = \frac{MSE_i^e}{MSE_i} \quad i = 1, \dots, ndf, e = 1, \dots, nte \quad (4.12)$$

Structural damage, simulated by stiffness parameter variation of the element, leads to changes of the  $NMSE_i^e$ . Thus, an index called normalized modal strain energy change ( $NMSEC$ ) is defined based on the difference of the  $nmse_i^e$  before and after damage, as expressed in Eq. (4.13).

$$NMSEC = (NMSE_i^e)^d - (NMSE_i^e)^h, e = 1, 2, \dots, nte \quad (4.13)$$

where  $(NMSE_i^e)^d$  and  $(NMSE_i^e)^h$  denote the normalized modal strain energy in damaged and healthy scenarios.

Another commonly used modal strain energy based index for damage detection is  $MSEBI$ , defined as follows(S. Lee et al., 2021).

$$MSEBI^e = \max \left[ 0, \frac{(NMSEC_i^e)^d - (NMSEC_i^e)^h}{(NMSEC_i^e)^h} \right] \quad (4.14)$$

In this study, the original  $NMSEC$  datasets were used for CNN training rather than  $MSEBI$ .

The main reason is that the  $NMSEC$  does not have the max operation, meaning that negative values are also taken into consideration. Also, the  $NMSEC$  matrix is filled with values in the range of  $[-1,1]$  which are inherently suitable for the training since no

normalization will be needed in such a case. We firstly conducted the principal component analysis (PCA) on the original *NMSEC* data and then fed the produced datasets into the CNN.

#### 4.2.4 Effects by element and mode based on principal component analysis (PCA)

In this study, the effects by element and mode were also investigated. This started with dimension reduction of the original *NMSEC* dataset with PCA. After that, the selected *PCs* could be expressed by the linear combination of the original variables. In this study, the original *NMSEC* input is 960 dimensional (160 elements×6modes). Suppose 36 *PCs* were selected (this result could be further seen in Section 3.4), these selected *PCs* could be expressed as follows.

$$[PC_1 \quad PC_2 \quad \dots \quad PC_{36}] = [X_1 \quad X_2 \quad \dots \quad X_{960}] \begin{bmatrix} w_{11} & w_{1,2} & \dots & w_{1,36} \\ w_{2,1} & w_{22} & \dots & w_{2,36} \\ \vdots & \vdots & \vdots & \vdots \\ w_{960,1} & w_{960,2} & \dots & w_{960,36} \end{bmatrix} \quad (4.15)$$

where  $PC_j$  and  $X_i$  is the  $j^{th}$  principal component and the  $i^{th}$  original variable. The  $w_{ij}$  is the coefficient of the  $i^{th}$  original variable in the  $j^{th}$  principal component, namely the loadings.

This equation could be simplified as

$$P = XW \quad (4.16)$$

where  $P$ ,  $X$  and  $W$  denote the principal component matrix, original variable vector, and the loadings. The importance of original variables could be measured by multiplying the principal component matrix  $P$  by the explained variance ratio vector, as written in

$$PR^T = XWR^T \quad (4.17)$$

$$\begin{aligned}
PR^T &= [X_1 \quad X_2 \quad \dots \quad X_{960}] \begin{bmatrix} w_{11} & w_{1,2} & \dots & w_{1,36} \\ w_{2,1} & w_{22} & \dots & w_{2,36} \\ \vdots & \vdots & \vdots & \vdots \\ w_{960,1} & w_{960,2} & \dots & w_{960,36} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_{36} \end{bmatrix} \\
&= [X_1 \quad X_2 \quad \dots \quad X_{960}] \begin{bmatrix} w_{11}r_1 + w_{1,2}r_2 + \dots + w_{1,36}r_{36} \\ w_{2,1}r_1 + w_{22}r_2 + \dots + w_{2,36}r_{36} \\ \vdots \\ w_{960,1}r_1 + w_{960,2}r_2 + \dots + w_{960,36}r_{36} \end{bmatrix} \quad (4.18)
\end{aligned}$$

where  $R=(r_1, r_2, \dots, r_{36})$  and  $r_j$  stands for the explained variance ratio of the  $j^{th}$  principal component ( $PC_j$ ). Eq.(4.18) could also be written in the following simple form.

$$XWR^T = c_1X_1 + c_2X_1 + \dots + c_{960}X_{960} \quad (4.19)$$

where  $c_i = w_{i1}r_1 + w_{i2}r_2 + \dots + w_{i,36}r_{36}$ ,  $i = 1, 2, \dots, 960$ .  $c_i$ , the coefficient for  $X_i$ , represents the importance of  $X_i$  for the whole dataset.

To evaluate the effect of the element, for the element  $k$ , the coefficients in the six modes could be added together and then normalized with respect to the total coefficients' summation. This could be considered as the coefficient for the bar element  $k$ , as written in

$$E_k = \frac{\sum_{m=1}^6 |c_{160(m-1)+k}|}{\sum_1^{960} |c_n|} \quad m = 1, 2 \dots 6, \quad k = 1, 2, \dots, 160, n = 1, 2, \dots, 960 \quad (4.20)$$

Accordingly, the effect by mode could also be evaluated according to the following equation.

$$M_i = \frac{\sum_{k=160(i-1)}^{160i} |c_k|}{\sum_1^{960} |c_n|} \quad i = 1, 2 \dots 6, \quad k = 160(i-1), n = 1, 2, \dots, 960 \quad (4.21)$$

where  $M_i$  represents the effects of mode  $i$ . The normalized  $M_i$  were used for mode effect evaluations. The effect evaluation on bar elements and modes would potentially work as

a guidance for representative information selection for damage detection and thus would greatly reduce the computational cost.

#### 4.2.5 Flowchart of this chapter

For clarity, Figure 4.2 illustrates the procedures of damage detection on a truss bridge model using the proposed loss function and indices in this study. Firstly, the mode shapes under three damaged scenarios were extracted from a FE model of a truss bridge. These mode shapes were trained in CNN with the proposed UMSE as a loss function. After the training, *DLA* and *MAEDD* were used for performance evaluation on damage detection. Moreover, the modal strain energy was extracted and corresponding *NMSEC* were calculated and then conducted with PCA. Finally, dimension-reduced data (The PCs, namely the *NMSEC-PCA* data) were sent into the CNN for damage detection. The damage detection effects by elements and modes were analysed based on theories in Section 4.2.4.

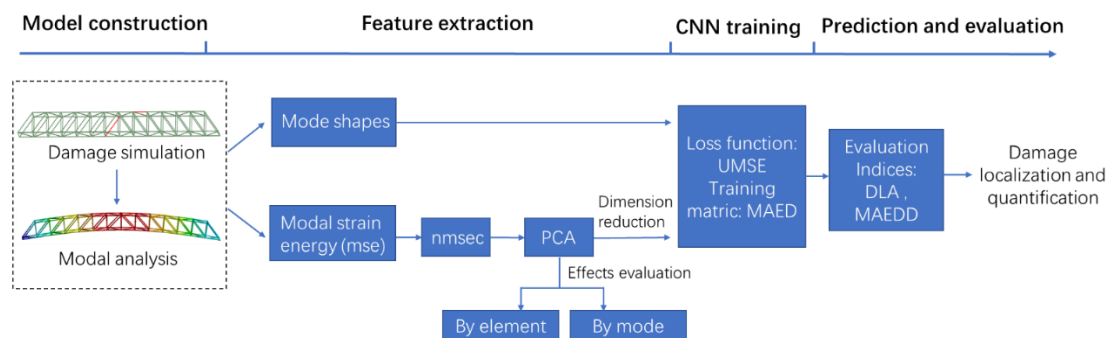


Figure 4.2 Flowchart of the investigations in this chapter

As shown in Figure 4.2, a novel method based on the UMSE loss function was proposed and performance indices damage localization accuracy (DLA) and mean absolute error of damage degrees (MAEDD) were used for damage detection problems in this chapter. The UMSE loss function achieved faster convergence speed and higher accuracy on

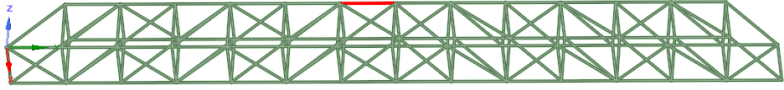
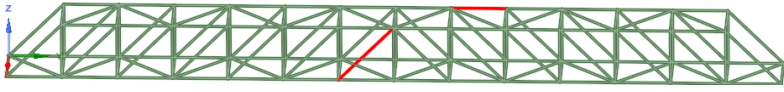
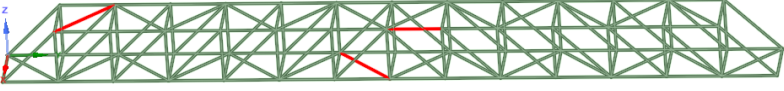
damage quantification than the commonly used MSE.

### 4.3. NUMERICAL SIMULATIONS

#### 4.3.1 Numerical modelling

Figure 4.3 shows the finite element model of a truss bridge constructed in Ansys workbench 19.0. The bridge model consists of 56 nodes and 160 beam elements. It is with a length of 8.4 m, a width of 0.6 m, and a height of 0.6 m. The left end of the model is simply supported. The right-end are constrained in X, Z directions and free in the longitudinal direction Y. The bridge model is a frame structure with members simulated by beam 188 element with circular tube section, giving 336 (56×6) degree-of-freedom (DOF). The Young's modulus and density are  $2.0 \times 10^{11}$  Pa and  $7800 \text{ kg/m}^3$ . The designed damage scenarios consist of single damage, double damage, and triple damage. Details of the damaged element number and locations are given in Table 4.1. All the damage is simulated by elastic modulus reduction of bar elements. For instance, 10% damage of a certain bar element means a 10% reduction of the elastic modulus on that bar.

Table 4.1 Scenarios for damage detection

Scenarios	Damaged elements	
	No.	Locations
Single Damage	129	
Double Damage	82,96	
Triple Damage	13,89 100	

\*Elements in red are damaged

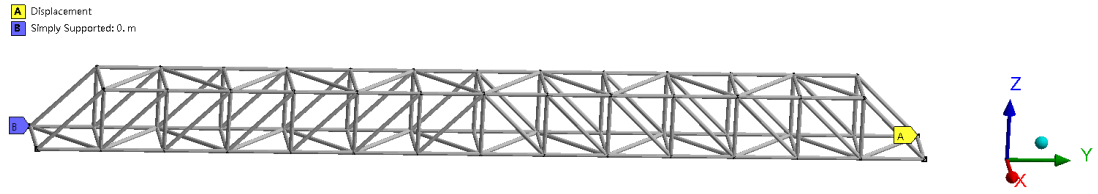


Figure 4.3 Finite element model of the truss bridge

### 4.3.2 Implementation details of CNN

Modal analysis was conducted on the numerical model of the truss bridge. The first six mode shapes of the designed damaged scenarios were extracted and taken as the original input of CNN. Figure 4.4 shows the data preparation and the architecture of CNN. As can be seen, the original data were nodal displacement data, namely the mode shapes, at six DOFs in six modes and they were flattened to one-dimensional (1D) vector of shape  $1 \times 2016$  ( $56 \text{ nodes} \times 6 \text{ DOFs} \times 6 \text{ modes}$ ). To apply 2D convolution, they were reshaped to a square matrix of  $1 \times 45 \times 45$  with 9 padded zeros. Before being fed into CNN, these mode shapes were rescaled to a range of  $[-1, 1]$  using min-max normalization.

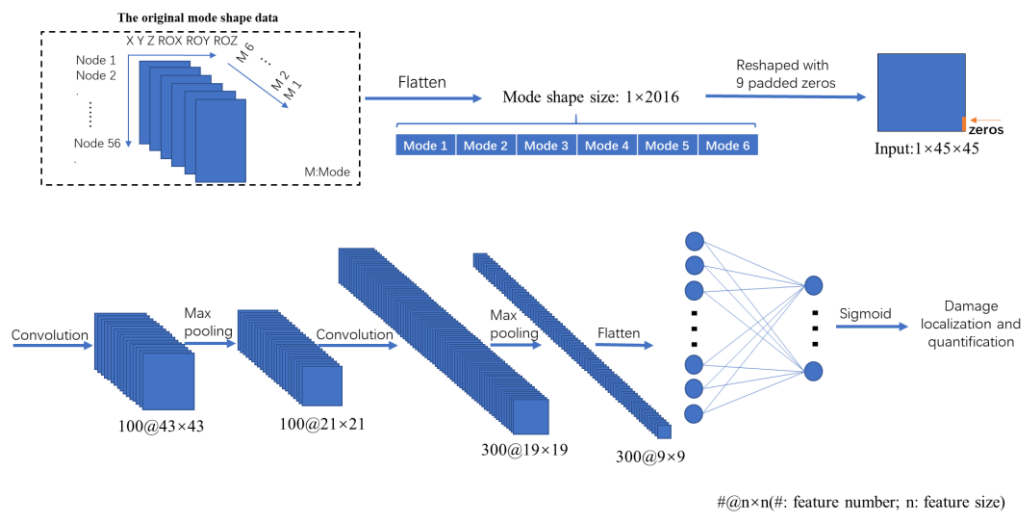


Figure 4.4 The data preparation and the architecture of the CNN

To get enough training samples, for each scenario, 2000 randomly generated damage

degrees from 0 to 1 were generated and corresponding mode shapes were extracted and used for training. The sample size of 2000 per scenario was selected based on a balance between computational feasibility and ensuring a comprehensive representation of the damage range from 0 to 1. This number of samples allows the model to adequately capture varying damage degrees while maintaining computational efficiency. In total, the dataset concluded 6000 samples from three scenarios, each with 2000 samples. The dataset division was 70% (4200 samples) for training, 15% (900 samples) for validation, and 15% (900 samples) for testing.

Table 4.2 presents the architecture of the mode shape-based CNN. It contained two convolutional layers with kernel size 3, each followed by a ReLu activation and max-pooling layer with a kernel of size  $2 \times 2$ . After the convolution, damage features were flattened into vectors and went through multiple fully connected (FC) layers. The first FC layer contained 300 neurons while the output layer contained 160 neurons as there are 160 potential damaged bar elements. Since CNN aimed to quantify damage, the output layer was a regression layer with the sigmoid activation. The proposed *UMSE* (Eq. (4.3)) was used as the loss function and rmsprop as the optimizer. The learning rate was 0.0002. The proposed *MAED* (Eq. (4.6)) worked as a training metric and *DLA* (Eq. (4.7)) and *MAEDD* (Eq. (4.8)) were used for performance evaluation. The models were implemented using PyTorch within the PyCharm IDE and trained on a G7 7700 laptop with an Intel Core i7-10750H CPU @ 2.60 GHz, 16.0 GB of RAM, and an NVIDIA GeForce RTX 2070 GPU for accelerated computations. The training time depends on the

model architecture, and for a typical CNN with two convolutional layers as described in this section, training on 4200 samples took approximately 10 minutes.

Table 4.2 Mode shape based CNN architecture

Layer	Layer type	Kernel size	No. of kernels/ neurons	Stride	Activation	Output shape
1	Input	-	-		-	45×45
2	Convolution	3×3	100	1	ReLU	43×43
3	Max pooling	2×2		2	-	21×21
4	Convolution	3×3	300	1	ReLU	19×19
5	Max pooling	2×2		2	-	9×9
6	FC	-	300	-	-	-
7	Output	-	160	-	Sigmoid	-

### 4.3.3 Data augmentation

Data augmentation is a commonly used technique for mitigating overfitting in computer vision. With data augmentation, more training samples could be generated from the existing training samples. In this study, 900 extra training samples were generated by adding 5 % Gaussian noise to the original datasets using the Eq. (4.22), 300 for each damage scenario.

$$\bar{h} = h(1 + 0.05R) \quad (4.22)$$

where  $h$  is the original data, and  $\bar{h}$  is the data with 5% noise.  $R$  follows a distribution of  $R \sim N(0,1)$ . The added noise follows a distribution of  $N(0,1)$ . By feeding the network with these extra samples, the learning ability and anti-noise ability are expected to improve.

### 4.3.4 Modal strain energy based PCA

The *NMSEC* training data was conducted with PCA for dimensionality reduction. The



explained variance by the first forty components was shown in Figure 4.5. As can be seen, the first 10 components contained approximately 90% of the variance, while around 30 components described close to 100% of the variance. In this case, 36 components, retaining 99.98% of the variance, were selected. With the PCA transform, the dimension of the train data matrix was reduced from the original 960 (160 elements×6modes) to 36. The transformed matrix was then reshaped to 6×6 and inputted into CNN. The architecture of *NMSEC-PCA* based CNN is given in Table 4.3 The proposed *UMSE* and *MAED* were used as loss function and training metric. *DLA* and *MAEDD* were used for performance evaluation.

Table 4.3 *NMSEC-PCA* based CNN architecture

Layer	Layer type	Kernel size	No. of kernels/ neurons	Stride	Activation	Output shape
1	Input	-	-	-	-	6×6
2	Convolution	2×2	100	1	ReLU	5×5
3	Max pooling	2×2	-	2	-	2×2
4	Convolution	2×2	300	1	ReLU	1×1
5	FC	-	300	-	-	-
6	Output	-	160	-	Sigmoid	-

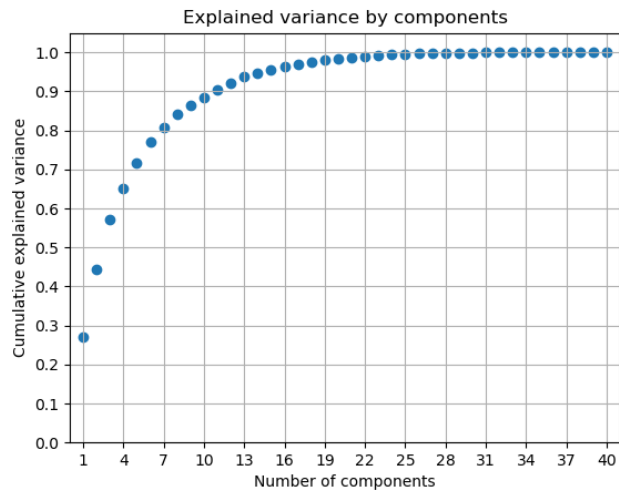


Figure 4.5 Explained variance by components

## 4.4. RESULTS AND DISCUSSIONS

### 4.4.1 Architecture selection

Network architecture selection is important to achieve high performance. In this section, three architectures with a different number of convolutional layers were compared and the commonly used *MSE* was used as a loss function. Figure 4.6 shows the validation loss comparison of these architectures. Among all, one convolutional layer CNN with 100 kernels (CNN1) converges the fastest, while the network with two convolutional layers (CNN 2) achieves the best loss performance. After training, for each CNN, 900 samples were tested, and a prediction matrix could be obtained. Table 4.4 shows the prediction results when evaluated with three statistical indices. It can be seen that CNN2 outperforms other architectures in terms of testing *MSE* ( $2.1903 \times 10^{-05}$ ), *DLA* (99.66%), and *MAEDD* (0.026). Overall, CNN with two convolutional layers achieved the best performance and was thus used for the rest of this chapter.

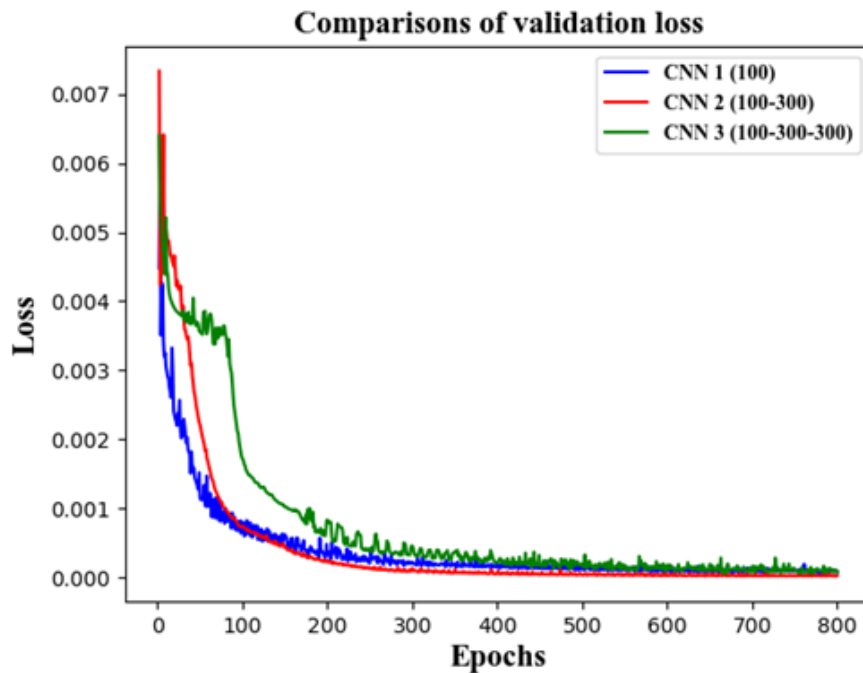


Figure 4.6 Validation loss of different architectures

Table 4.4 Comparisons of architectures

Architectures	Test MSE	DLA	MAEDD
100	$6.3165 \times 10^{-05}$	96.33%	0.038
<b>100-300</b>	<b><math>2.1903 \times 10^{-05}</math></b>	<b>99.66%</b>	<b>0.026</b>
100-300-300	$6.3046 \times 10^{-05}$	98.00%	0.037

#### 4.4.2 Effects of the loss function

To study the effects of weighting factors  $\alpha$  and  $\beta$  on CNN performance, a parameter study on the updated loss function was conducted. The validation curves of the updated MSE (*UMSE*) with selective weighting parameter combinations were compared, as presented in Figure 4.7. It can be seen that the updated MSE with  $\alpha=0$  and  $\beta=1$  (C8) converged the fastest among all, followed by  $\alpha=0.2$  and  $\beta=0.8$  (C7). Moreover, with enough epochs of 800, most *MAED* curves converged to the same level. Interestingly, for cases where  $\beta < 0.4$ , namely C1-C3, the *MAED* curves remained stable at around 0.5 at the beginning. For C1, the *MAED* even ended at around 0.5. This is because the CNN is trapped into a local optimum of a prediction matrix filled mostly with numbers close to zero. In such a case, the values in the damaged regions were also near zero, leading to a poor damage localization accuracy.

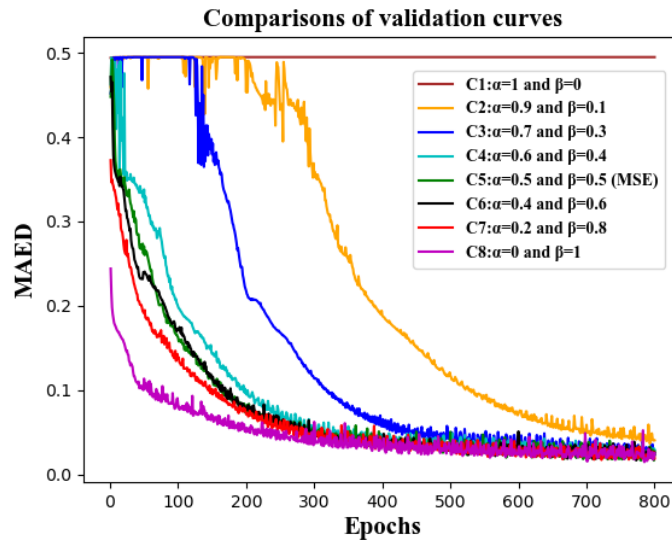


Figure 4.7 Comparisons of validation curves relating to different weighting factor combination

Testing performances of these combinations (*C1-C8*) were also compared in terms of testing *MSE*, *DLA*, and *MAEDD* as shown in Table 4.5. It can be seen that from *C2* to *C7*, as  $\beta$  grows, the *DLA* remained stable at around 99% while both testing *MSE* and *MAEDD* saw improvements, test *MSE* from 4.466 e-5 to 1.307e-5 and *MAEDD* from 0.0359 to 0.0170. Among all combinations, *C7*( $\alpha=0.2$  and  $\beta=0.8$ ) performed the best, achieving low test *MSE* (1.307e-05) and *MAEDD* (0.0170).

Table 4.5 CNN testing results evaluation of the updated MSE

Combination No.	Loss function of CNN	Testing MSE	DLA (%)	MAEDD
<i>C1</i>	$\alpha=1$ and $\beta=0$	0.0041	33.33	0.517
<i>C2</i>	$\alpha=0.9$ and $\beta=0.1$	4.466e-05	98.33	0.0359
<i>C3</i>	$\alpha=0.7$ and $\beta=0.3$	1.920e-05	99.78	0.0202
<i>C4</i>	$\alpha=0.6$ and $\beta=0.4$	2.741e-05	97.67	0.0265
<i>C5</i>	$\alpha=0.5$ and $\beta=0.5$ ( <i>MSE</i> )	2.221e-05	99.00	0.0268
<i>C6</i>	$\alpha=0.4$ and $\beta=0.6$	2.355e-05	98.67	0.0236
<i>C7</i>	$\alpha=0.2$ and $\beta=0.8$	1.307e-05	98.67	0.0170
<i>C8</i>	$\alpha=0$ and $\beta=1.0$	0.2558	0.78	0.0086

When  $\alpha=1$ , *C1* reached a random *DLA* of 33% as the prediction matrix was filled mostly with near-zero numbers. As expected, in *C8* where  $\beta = 1$ , the CNN achieved the best *MAEDD* but the worst *DLA* (0.78%). This means that it failed to locate damage in 893 out of a total 900 testing samples. The main reason is that the *UMSE* of *C8* considered only the values in the damaged regions, leading to a prediction matrix where intact regions filled mostly with values around 0.5.

Based on the above analysis,  $\beta$  relates to how close the damage degrees are predicted, namely the *MAEDD*, while  $\alpha$  relates to the damage localization accuracy (*DLA*). In other words, with high  $\beta$ , the updated *MSE* would usually achieve good damage quantification

performance, while with a high  $\alpha$ , a better damage localization accuracy could be expected.

It is worth noting that, in practice,  $\alpha$  and  $\beta$  should be rigorously assigned according to situations. Generally,  $\beta$  in the range of [0.3,0.8] is recommended to get a higher CNN performance as well as avoid local optimum. For the rest of this study, the updated *MSE* with  $\alpha = 0.2$  and  $\beta = 0.8$  was used as it achieved not only a faster convergence speed but also the best testing *MSE*, *MAEDD*, and a relatively higher *DLA*.

Merits of the proposed *UMSE* loss function are as follows: 1). It has high flexibility as the users could weigh the damage localization and damage quantification by assigning different weighting factor values. 2). The updated MSE loss function ( $\beta > 0.5$ ) avoids unnecessary iterations on the intact region but focuses on only the predictions of damage degrees in the damaged region, thus achieving a higher damage quantification accuracy. 3). The practical engineering projects usually consist of thousands of bar elements, resulting in a large CNN training label matrix. The customized *UMSE* would reduce the epoch number, and iteration time and enhance the damage detection accuracy of the network.

#### **4.4.3 Feature visualization**

To demonstrate the automatic feature extraction ability of the proposed CNN, the t-distributed Stochastic Neighbor Embedding (t-SNE) was used for feature visualization.

The t-SNE was firstly proposed by Maaten (Maaten & Hinton, 2008) for dimension

reduction of high-dimensional data for visualization. The t-SNE firstly converts the points similarities to joint probabilities and then takes the Kullback-Leibler divergences between joint probabilities of the low-dimensional space and high dimensional data as cost function (Lin et al. 2017). The results of t-SNE varies due to the different initializations.

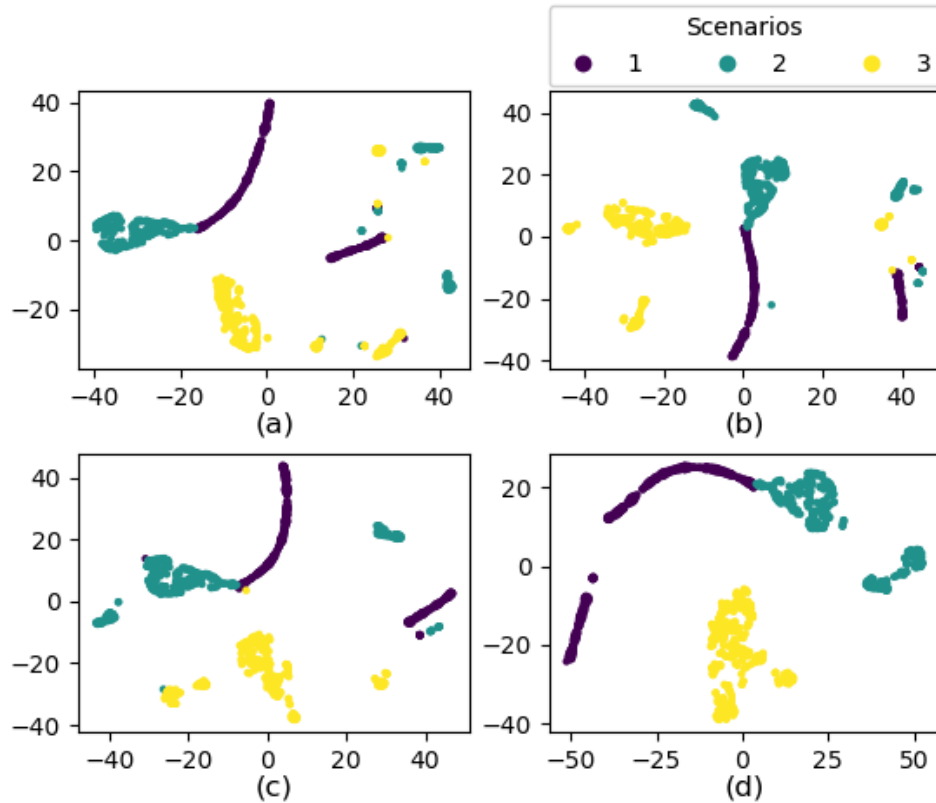


Figure 4.8 CNN feature visualization with t-SNE:

(a) the input (b) after first Conv layer (b) after the second Conv layer (d) after the FC layer

In this study, the testing datasets (900 samples, three scenarios each with 300 samples) were inputted into the CNN for visualization. It is worth noting that as the original outputs of the convolutional layers were four dimensional (samples, channel, width, height), they were reshaped to two dimensional before conducting the t-SNE transformation. Four output features were visualized as shown in the Figure 4.8, including the features of the original testing data, features after two convolutional layers and features after fully

connected layer. It can be seen that the original testing data were dispersed into several clusters (Figure 4.8 (a)) while after the two convolution layers, fewer clusters were seen. In Figure 4.8 (d), most points from the same scenarios gather in the same cluster, indicating good feature extraction ability of the proposed CNN.

#### 4.4.4 Effects of data augmentation

For comparison, two Convnets, CNN1 and CNN2, were trained, CNN 2 with data augmentation while CNN 1 without. Implementation details of data augmentation could be seen in Section 4.3.3. After the training, the original 900 testing samples were added with 3% and 5% noise and were tested respectively. Figure 4.9 shows the validation loss and metric (*MAED*) of the Convnets. It can be seen that the CNN 2 converges slightly faster in the first 200 epochs. A positive effect of data augmentation on anti-noise ability could be seen in Table 4.6 where the prediction results of noise-free 3% and 5% noise polluted samples are evaluated in terms of testing *MSE*, *DLA*, and *MAEDD*. When tested with noise-free samples, both CNN1 and CNN2 achieved high damage detection accuracy, with the testing *MSE* from  $1.30e$  to  $5-1.97e-5$ , *DLA* at around 98%, and *MAEDD* at around 0.018. However, when tested with noise polluted samples, both saw noticeably declined performances in all three indices. Despite of the declines, *CNN2* achieved higher damage localization and quantitation accuracy than *CNN 1* for both 3% and 5% noise polluted data.

To further prove the effectiveness of data augmentation, prediction results of three

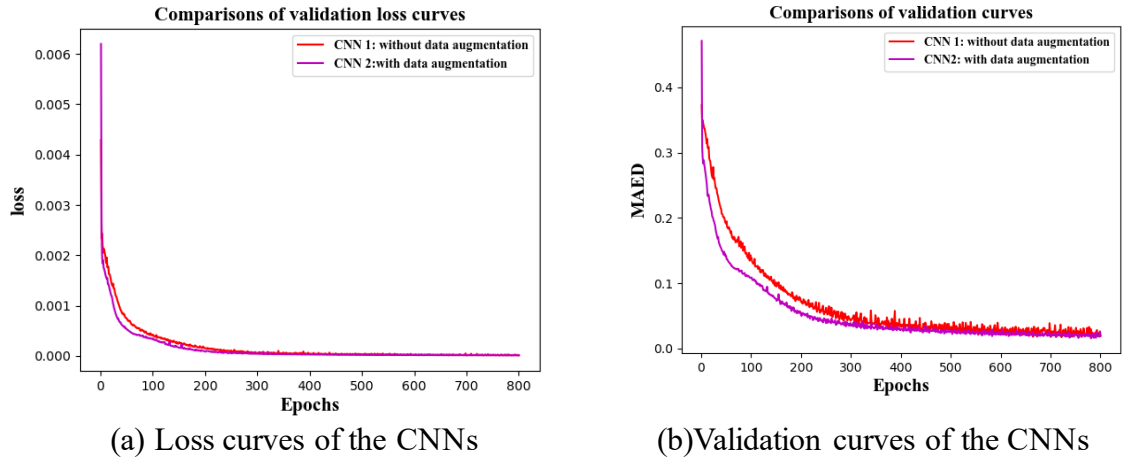


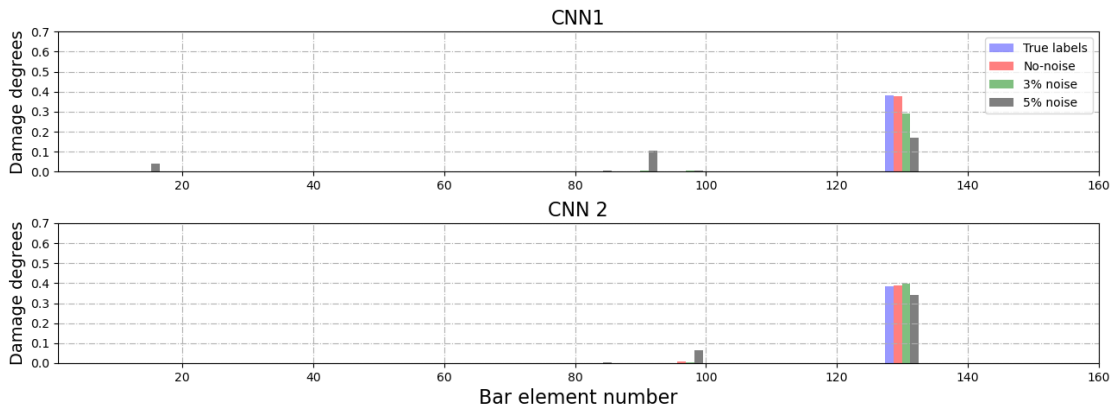
Figure 4.9 Comparisons of the CNN with and without data augmentation

randomly selected samples (sample 26, 418, and 635) were presented in Figure 4.10. It can be seen that 1) when tested with non-noise testing data, both CNN1 and CNN 2 achieved high detection accuracy on damage localization and quantification in all samples; 2) for the polluted data (3% and 5% noise), CNN2 outperformed CNN1 on damage localization and quantification. Particularly, in Figure 4.10 (b), for the 5% noise polluted sample, CNN 1 failed to detect the damaged bar 96 and showed false alarms on other elements while CNN 2 successfully identified the damaged bars with acceptable prediction accuracy on damage degrees. Overall, although Convnets trained by the original mode shape datasets could achieve high damage detection accuracy, they may exhibit slightly poor robust ability. In such cases, data augmentation is a way to mitigate this problem.

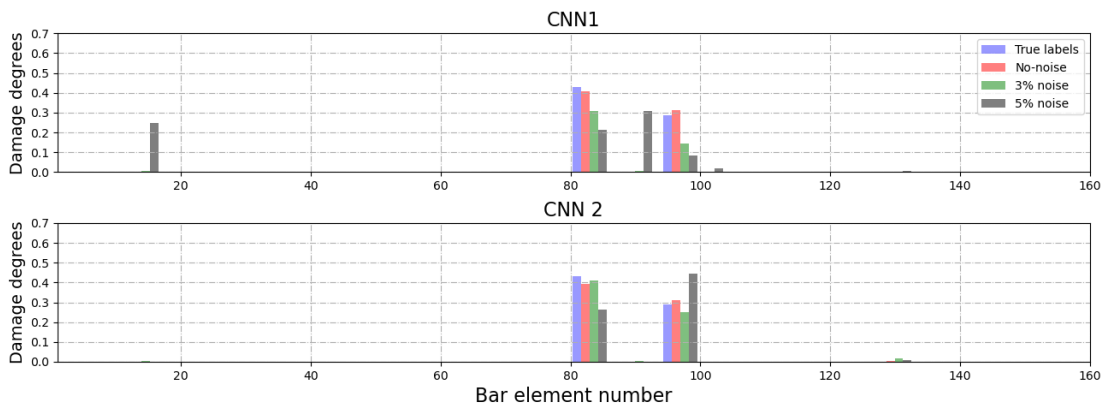
Table 4.6 CNN performance evaluation using noise-free and 5% noise testing samples

Test samples	Testing MSE		DLA		MAEDD	
	CNN 1	CNN 2	CNN 1	CNN 2	CNN1	CNN 2
Noise-free	1.3067e-5	1.9706e-05	98.67%	98.88%	0.0170	0.0186
3% noise	0.0003	0.0002	89.22%	92.33%	0.0850	0.0649
5% noise	0.0006	0.0004	81.00%	88.11%	0.134	0.0941

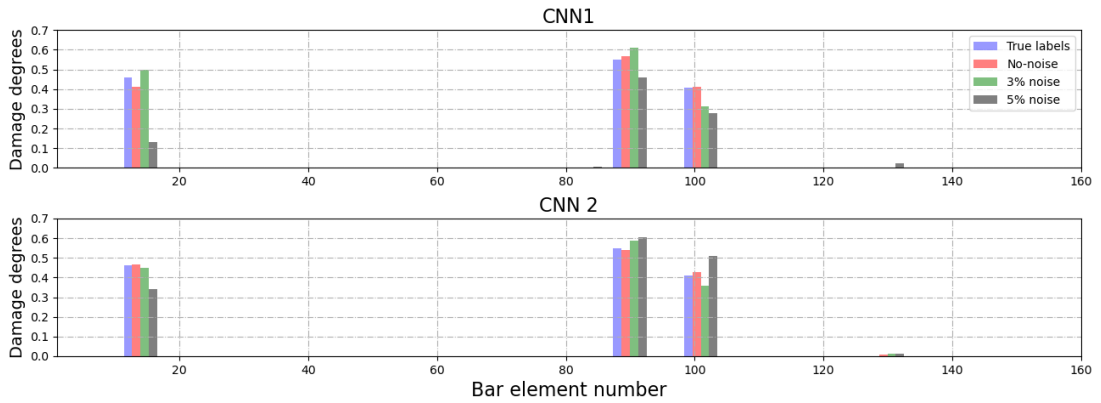




(a) Prediction results of sample 26 (damaged bar number: 129)



(b) Prediction results of sample 418 (damaged bar number: 82,96)



(c) Prediction results of sample 635 (damaged bar number: 13,89,100)

Figure 4.10 Prediction results of selective samples using CNN1 and CNN2

#### 4.4.5 The *NMSEC-PCA* based CNN

According to Section 4.3.4, the PCA dimensionality reduction operation produced a *NMSEC-PCA* based input matrix of 36 dimensional. It was then reshaped to  $1 \times 6 \times 6$  and fed into the CNN. After the training, the features of the CNN using testing samples were

visualized using t-SNE method, as presented in Figure 4.11. Figure 4.12 presents validation curve comparisons between the mode shape-based method and the *NMSEC-PCA* based method. Accordingly, Table 4.7 compares these two methods in terms of input shape, training time, and testing indices. Testing results on selected samples (sample 26,418, and 635) of these two CNNs were also compared in Figure 4.13. Overall, it can be seen that these two methods achieved almost the same convergence speed and testing performance. In terms of testing *DLA* and *MAEDD*, the *NMSEC-PCA* based CNN achieved slightly higher accuracy. However, *NMSEC-PCA* based method is significantly ten times faster than the mode shape based. This advantage would enable the *NMSEC-PCA* based method to have more potential for real application on the damage detection problem of large-scale structures.

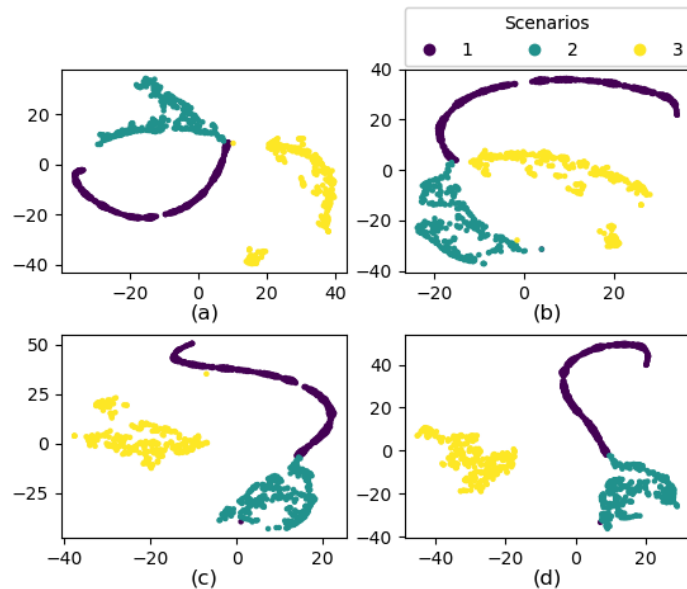


Figure 4.11 *NMSEC-PCA* based CNN feature visualization with t-SNE:  
(a) the input (b) after first Conv layer (c) after the second Conv layer (d) after the FC layer

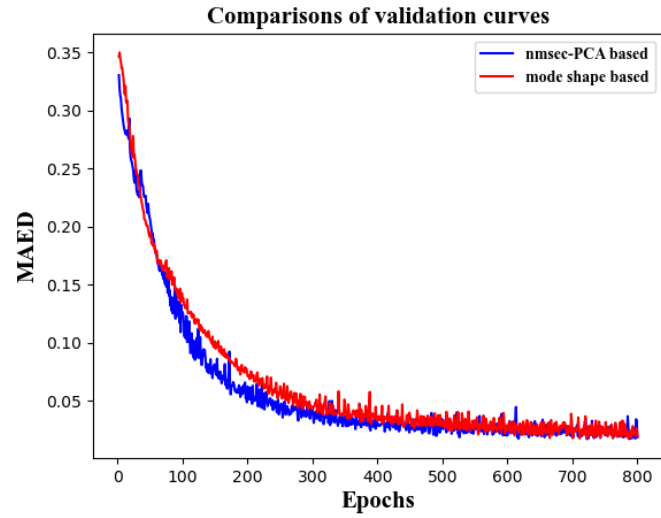


Figure 4.12 Validation curve comparisons between mode shapes based and *NMSEC*-PCA based index

Table 4.7 Comparisons of the two methods

Methods	Mode shape based	<i>NMSEC</i> -PCA based
Input shape	4200×1×45×45	4200×1×6×6
Training time	Around 10 minutes	Within a minute
Testing MSE	1.3067e-05	3.9385e-5
Testing <i>DLA</i>	98.67%	99.22%
Testing <i>MAEDD</i>	0.0170	0.0154

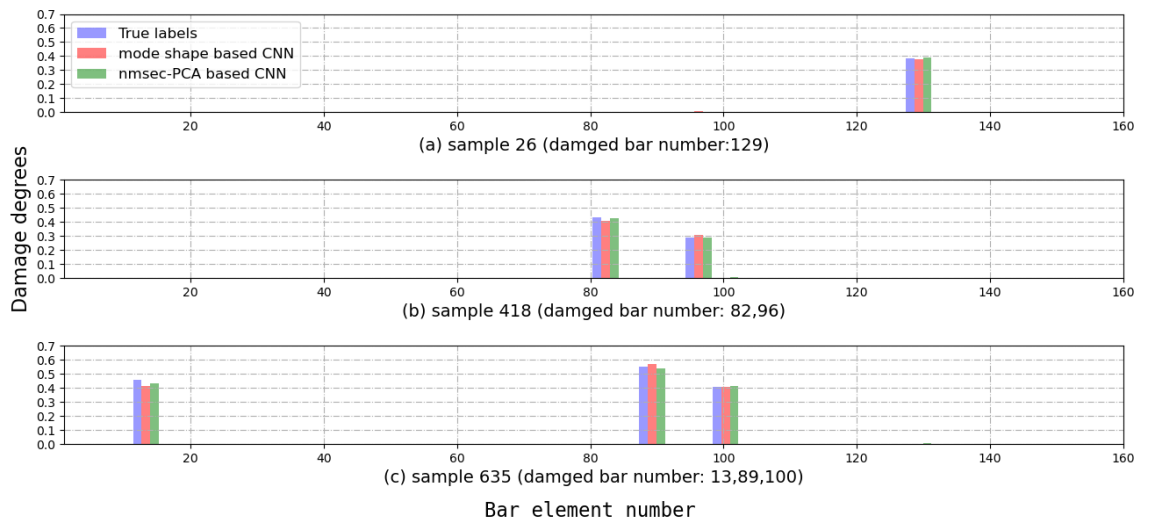


Figure 4.13 Prediction results of selective samples using mode shape based CNN and *NMSEC*-PCA based CNN

#### 4.4.6 Effects by element and mode

From the obtained PCs, effects by element and mode were calculated according to Eqs. (4.20) to (4.22) as shown in Figures 4.14 and 4.15. Bar elements 10,108 and 115 account for the first three most valuable pieces of modal strain energy information. Figure 4.15 shows that mode 4 accounts for the most information of the whole dataset while mode 6 contributes the least information, thus mode 6 could be abandoned. In the future research, this analysis could help estimate the sensitivity of elements and modes on the modal strain energy.

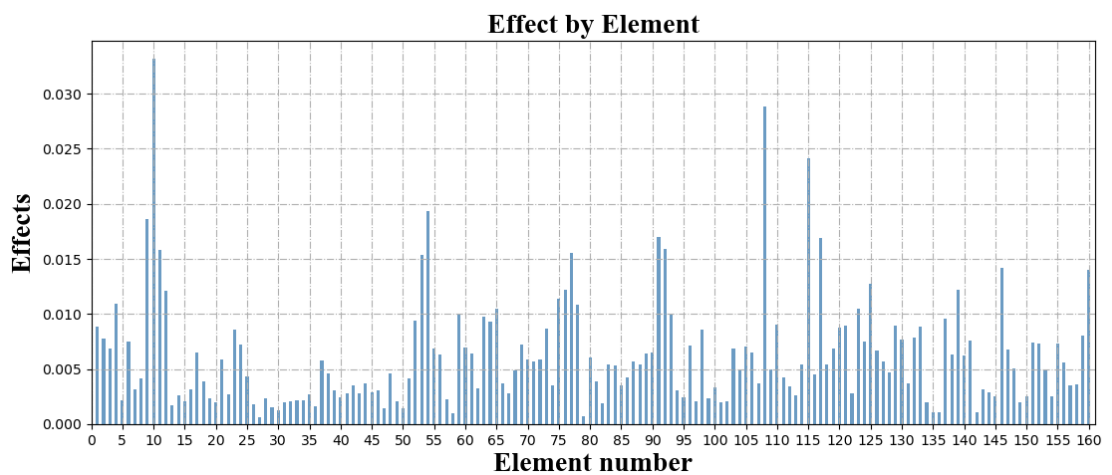


Figure 4.14 The effect by bar element

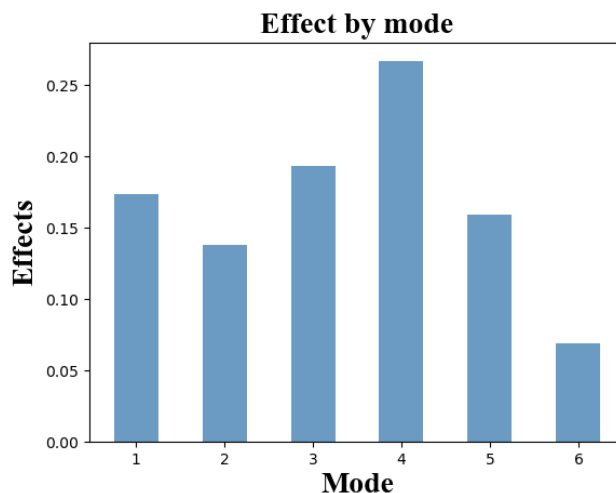


Figure 4.15 The effect by mode

## 4.5. SUMMARY

This section proposed a novel updated loss function *UMSE* and damage detection evaluation indices for damage detection problems and demonstrated the efficiency of the proposed method on a truss bridge model. PCA was conducted on the *NMSEC* for dimension reduction. The *NMSEC*-PCA based data were trained by CNN and damage detection accuracy were compared to the mode shape-based CNN. The following conclusions could be drawn.

- 1) CNN with two convolutional layers were selected as it achieved the best damage detection accuracy. The t-SNE were employed to visualize the output of each network layer to demonstrate the automatic feature extraction of the proposed CNN.
- 2) The proposed *UMSE* loss function outperforms the commonly used MSE as it could greatly reduce the computational cost, achieve faster convergence speed and higher detection accuracy, and is highly flexible for practical engineering structures applications as it could be customized according to needs. Based on this research, a combination of  $\alpha=0.2$  and  $\beta=0.8$  in the *UMSE* is suggested for damage quantification problems.
- 3) For a clear comparison, we used both testing MSE and proposed indices (*DLA* and *MAEDD*) for evaluation. A near-zero testing MSE cannot indicate an overall high detection accuracy since they are inherently near zeros in this study. In reality,

even a slight change in testing MSE could mean a great difference in the damage accuracy. An example of this could be seen in Table 4.6. When the two CNNs were tested by 5% noise polluted samples, there is only a slight difference in the testing MSE (CNN1:0.0006 and CNN2:0.0004) while notable differences were seen in DLA and MAEDD. Therefore specifically developed indices are needed.

- 4) When tested with noise polluted data, the CNN with data augmentation achieved better detection accuracy than the one without. In practice, the mode shape-based CNN may exhibit a slightly poor robust ability. In such a case, data augmentation, that is generating samples by adding noise, could be a way to mitigate the noise effect.
- 5) Compared to mode shape input, the *NMSEC*-PCA as input is computationally cheaper yet equally effective. This reduction in computation cost would allow CNN more suitable for real structural application, especially for large-scale and complex structures.
- 6) Effects by element and mode were analysed based on *NMSEC*-PCA index. The result could potentially provide guidance on selecting valuable components from the original modal strain energy training data for the future research.

# **CHAPTER 5 ENHANCING BRIDGE HEALTH MONITORING WITH LSTM-VAE: A SEMI-SUPERVISED LEARNING FRAMEWORK FOR DAMAGE DETECTION**

## **5.1 INTRODUCTION**

Structural health monitoring (SHM) system guarantees the integrity and safety of key infrastructures like bridges. Due to their importance, the early detection and repair of damage in time in these bridges are important as there would be a huge human and financial loss if structural failure happened. However, the health monitoring of the bridges, particularly those consisting of thousands of joints and members, can be a challenging task due to the complexity and the vast amounts of data recorded by numerous sensors (Avci et al., 2021).

Identifying and quantifying damage in complex bridge structures is not straightforward. Traditional methods often require a lot of manual inspection, and they can be time-consuming, subjective, experience-based, and prone to errors. Also, the structural data can be influenced by various environmental factors, making it a complicated task. Thus, it is needed to have a more robust and automated yet accurate damage detection method.

Recent research has seen a shift towards data-driven methodologies, particularly those using machine learning methods, to address these challenges. These approaches offer a systematic and objective means of processing and analysing structural data, significantly reducing the effect of external factors, and enhancing the detection accuracy of subtle

damage indicators. Among the various machine learning techniques, unsupervised learning models, especially those using Variational Autoencoders (VAE), have been promising in identifying anomalies within complex datasets. Importantly, the VAEs do not require prior knowledge of damage conditions. This is particularly beneficial in situations where the damaged states are not predefined or entirely unknown.

The variational autoencoders, with LSTM layers (LSTM-VAE), are used for time series anomaly detection. This model combines the LSTM layer's and VAE capabilities. LSTM is for capturing temporal dependencies in sequential data, and VAE is for compressing high-dimensional data into meaningful latent representations. Such a combination is particularly useful for analysing time-series data, making it a powerful tool for structural damage detection in bridges.

Despite the potential of LSTM-VAE models in SHM, their application to bridge health monitoring, particularly in quantifying damage levels without damage data, remains underexplored. To the best of our knowledge, no research has been found employing LSTM-VAE specifically for SHM, highlighting a significant gap in the field. This study aims to fill this gap by proposing a novel application of the LSTM-VAE network tailored for bridge SHM. We introduce the Average Reconstructed Error (ARE) and the Statistical ARE (SARE) as innovative indices for damage quantification, providing a direct and objective measure of the extent of structural damage. Furthermore, a semi-supervised learning framework that combines t-SNE with ARE features is proposed, achieving significant accuracy in classifying damage levels. This methodology's effectiveness is



validated through both numerical simulation and experimental results, proving its practicality and potential to significantly enhance maintenance strategies for bridges.

This chapter not only contributes to the advancement of machine learning applications in civil engineering but also addresses a critical need for more efficient, reliable, and data-driven approaches to structural health monitoring. Through this research, a significant step towards enhancing the safety and longevity of bridges is taken, thereby supporting the sustainability of our infrastructure.

## **5.2 METHODOLOGY**

### **5.2.1 LSTM-VAE networks for enhancing structural health monitoring**

Bridges, with their complex structures comprising numerous joints and members, generate huge amount of data when attached with hundreds of sensors. Current machine learning methods, mostly designed for computer vision tasks, may not be directly applicable to the specific demands of multi-dimensional bridge sensor data. These methods usually reshape information from various sensors into a single sample, which can lead to the loss of critical spatial information essential for accurate damage detection. Furthermore, many existing approaches are supervised methods that rely on known damage information for training which is often not available structures in practice.

To address these challenges, a novel semi supervised network framework specifically designed for analysing multi-channel time-series data from structural damage detection

system is introduced. This approach maintains spatial information that is important in the detection of damage. Two key indices within this framework were also developed: Average Reconstructed Error (ARE) and Statistical ARE (SARE) are indices derived from the unsupervised part of model's output. ARE offers a direct measure of potential damage, while SARE refines this measurement by averaging anomalies across multiple samples to improve reliability.

The steps of the semi-supervised method for damage detection are as follows:

- 1) Data is collected from an array of sensors that are placed along the healthy bridge structure and then pre-processed for analysis.
- 2) The pre-processed data is then trained in the LSTM-VAE model, an unsupervised learning component that has been designed for multi-channel input, effectively capturing both temporal and spatial features intrinsic to the data.
- 3) After the training, the whole datasets are sent to the trained model to compute the Average Reconstructed Error (ARE), offering an initial damage quantification by comparing the ARE of the healthy baseline with that of the unknown samples.
- 4) t-SNE is used to visualize the latent features of the datasets, improving the interpretative understanding of the datasets.
- 5) ARE and t-SNE are combined to create a comprehensive feature for damage classification.
- 6) An SVM classifier is subsequently trained using this feature, achieving the classification of the bridge's structural damage state.

This methodology is illustrated in Figure 5.1 and it takes advantage of LSTM-VAE in dealing with multi-channel time-series data, which is more suitable for the practical needs of structural engineering than standard image-based machine learning models. With the proposed indices, we establish a robust and practical framework for the two-step task of damage detection: initial damage quantification and subsequently classifying the severity of detected damages. This method contributes to a comprehensive strategy for the maintenance and safety of bridge infrastructures in practice.

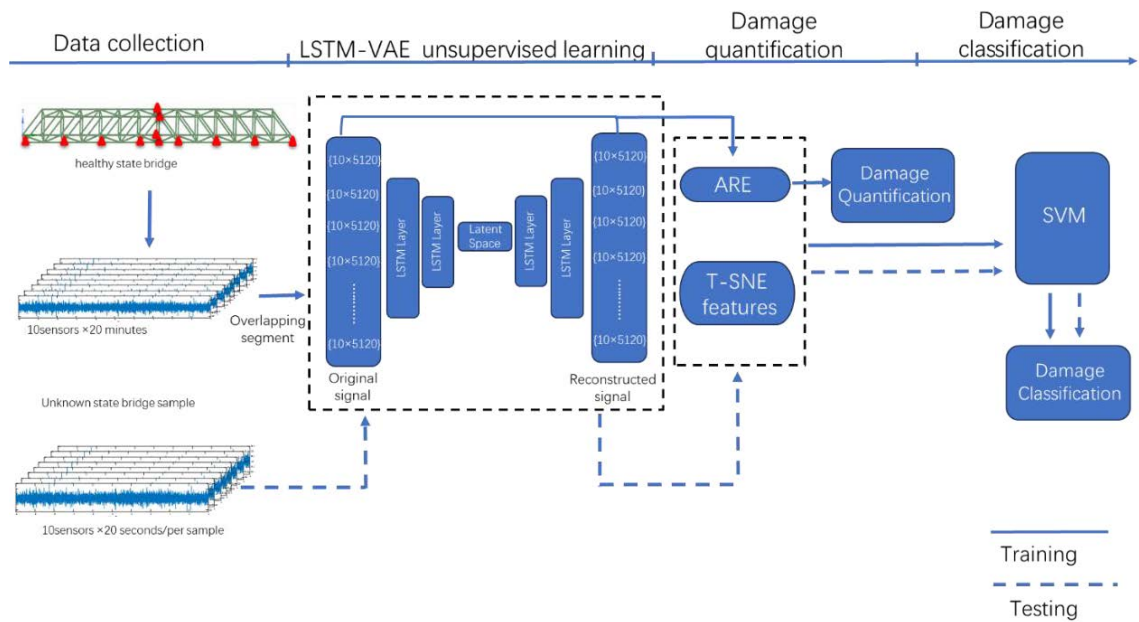


Figure 5.1 The flowchart of this chapter

### 5.2.1.1 VAE network architecture

Variational Autoencoders (VAE) is suited for data compression and generation, encoding data into a latent space with a defined probabilistic distribution. Different from the traditional autoencoders, VAEs are designed to be generative, being able to generate new data using latent space representation. The newly generated data matches the statistical

properties of the original dataset. Figure 5.2 illustrates a typical VAE that receives  $x$  as input and encodes it into the latent space. Then the decoder receives the input from the latent space and produces  $x'$  which shares the statistical property with  $x$  (“Variational Autoencoder,” 2024). This function could be useful for the aim of predictive maintenance and anomaly detection.

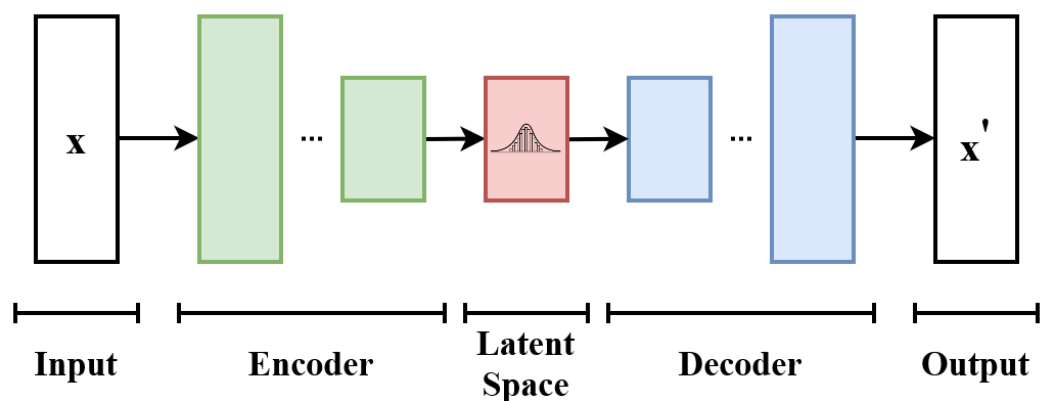


Figure 5.2 Variational autoencoder (VAE)(“Variational Autoencoder,” 2024)

### 5.2.1.2 RNNs with LSTM layer

Recurrent Neural Networks (RNNs) with LSTM layer are designed to break the limitations of traditional RNNs by dealing with long-range dependencies. The key layers to their architecture are LSTM layers which contain units called LSTM cells. LSTM layer consists of a series of memory cells that can maintain information in memory for long periods. The key components of these cells are the gates: the input, the output, and the forget gates. These gates regulate the flow of information into and out of the cell, and the decision to retain or discard data at each timestep. This gating mechanism allows LSTM layers to capture temporal dynamics and dependencies within the data, which is crucial

for accurate modelling and prediction of structural responses. This is particularly beneficial in Structural Health Monitoring (SHM) applications that involve time-series data from sensors monitoring the integrity of structures over time.

#### *5.2.1.3 LSTM-VAE*

The LSTM-VAE is an advanced machine learning model (Terbuch et al., 2022). Merging the LSTM's temporal modelling capabilities with the VAE's compression ability, the LSTM-VAE network is adept at reducing high-dimensional sensor data into a latent representation that contains the most essential features for accurate reconstruction and damage detection. This combination, consisting of the LSTM's encoding of time-series data into a latent space followed by the VAE's reconstruction of the data, ensures that the temporal integrity of the signals is maintained, improving anomaly detection in structural damage.

Informed by the advanced framework detailed in Terbuch et al.'s paper (Terbuch et al., 2022) on hybrid machine learning for anomaly detection in industrial time-series data, this chapter proposes a similar LSTM-VAE network architecture for multi-channel time-series analysis using the MATLAB programming.

An experimental study has been conducted on an eight-meter bridge model under the external excitation. 10 wireless accelerometers are installed on the bridge to capture structural acceleration responses of the bridge. The sensor data is typically organized into samples, each sample consisting of 10 channels  $\times$  5120 data points that represent the

signals from all ten sensors over a given period.

Figure 5.3 shows the architecture of the LSTM-VAE and its architecture illustrates a process of signal transformation. The original signal, structured in batches of ten channels each containing 5120 data points, is fed through a sequence of LSTM layers that capture temporal dependencies and significant features of the bridge acceleration data. This information is then encoded into a latent space, providing a representation that is decoded back into the original data space, producing the reconstructed signals. Discrepancies between the original signal and reconstructed signal were used to detect and quantify structural damage, illustrating the network's ability to find subtle patterns indicative of damage.

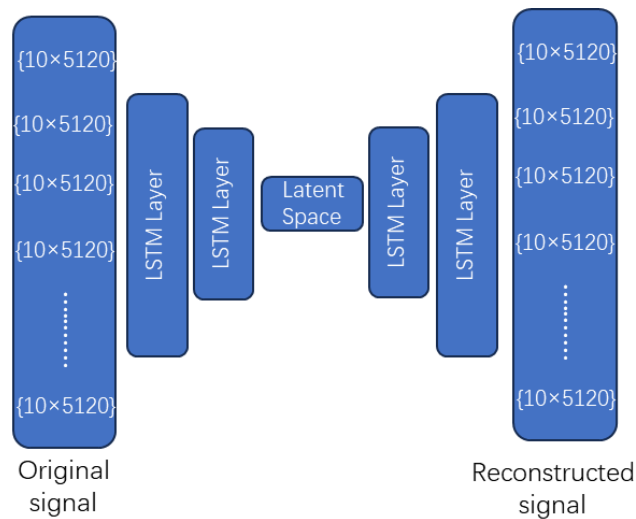


Figure 5.3 The flowchart of LSTM-VAE architecture

### 5.2.2 Reconstructed Error-Based Indices

In structural health monitoring, accurately evaluating damage through data interpretation, particularly for large-scale or complex bridges, remains a significant challenge. To

address this, we introduce two metrics: Average Reconstructed Error (ARE) and Statistical ARE (SARE), both of which are derived from the reconstruction errors of the LSTM-VAE model.

- *Metric 1: Average Reconstructed Error (ARE)*

The absolute reconstructed error is often used as an anomaly detection threshold in unsupervised learning methods. Given the multi-channel nature of the sensor data in this study, which employs unsupervised learning, there is a need for a damage quantification index that is customized for machine learning based damage detection using multi-channel sensor data. The *ARE* is thus proposed as the mean of the absolute differences between the original and the reconstructed signal across all sensor channels. This measure is intended to evaluate damage levels using the multi-channel LSTM-VAE model. The basic idea is that as damage influences structural response, an increase in *ARE* is expected with increasing damage severity, making it a potential metric of structural damage levels.

The *ARE* for a single sample  $i$  with  $N$  data points across  $H$  channels is defined as:

$$ARE = \frac{1}{NH} \sum_{h=1}^H \sum_{n=1}^N |x_{n,h}^{(i)} - \hat{x}_{n,h}^{(i)}| \quad (5.1)$$

where  $x_{n,h}^{(i)}$  is the original signal value and  $\hat{x}_{n,h}^{(i)}$  is the reconstructed signal value at the  $n^{th}$  data point for the  $h^{th}$  channel of the  $i^{th}$  sample.

- *Metric 2: Statistical ARE (SARE)*

To gain a clear measure of the damage level, particularly when multiple samples from the same scenario are available for damage detection, SARE is introduced. This index is the

average of the ARE values across all samples for a given scenario, thus reducing sample-specific noise and variability. SARE for a scenario with  $M$  samples is defined as:

$$SARE = \frac{1}{M} \sum_{i=1}^M ARE_i \quad (5.2)$$

This approach allows for a more robust and statistically significant assessment of damage levels, which is particularly useful in scenarios where individual sample errors may vary due to environmental factors or operational noise.

This methodology allows for a more robust and statistically significant assessment of damage levels, which is particularly valuable in scenarios where individual sample errors may vary due to environmental factors or noise. The Statistical ARE (SARE) is especially practical, as it provides damage measurement by averaging the ARE over multiple samples, thereby offering more stable damage evaluation results. This is crucial when monitoring structures that are subject to varying conditions or in cases where data collection may be inconsistent. With SARE, the assessment of structural health is not only more reliable but also reflects a true representation of the structure's condition over time. The proposed indices, ARE and SARE, are robust indicators customized for damage detection methods using multi-channel machine learning techniques, serving as a practical tool in health monitoring system for complex structures.

### **5.2.3 Semi-Supervised Learning Framework**

In this section, the semi-supervised learning framework is proposed by combining the LSTM-VAE and SVM. This approach combines the benefits of unsupervised feature



extraction with the classification ability of supervised learning models.

In the semi-supervised learning section, the feature includes t-SNE features and the ARE values. The ARE values, derived from the reconstruction error of the LSTM-VAE network, provide a quantitative measure of the deviation from normal structural behaviour. By combining t-SNE with ARE, we generate feature datasets that capture both the inherent structural damage information and the quantifiable damage metrics.

The generated feature is then used as input to a Support Vector Machine (SVM) classifier. SVM is chosen for its robustness and effectiveness in handling non-linear classification problems. It features in finding the optimal hyperplane that maximizes the margin between the classes in the feature space, which now represents varying levels of structural damage.

In summary, Section 5.2 presents a structured methodology that combines an LSTM-VAE network with a semi-supervised learning framework to address structural damage detection in bridges. By using the LSTM's ability to capture temporal patterns and the VAE's efficient data compression, this approach establishes a robust framework for extracting features. The application of t-SNE provides a visual interpretation of the data, while the combination of ARE with SVM classification shows the potential of machine learning in improving SHM applications. The validation of this approach, using both numerical and experimental data, proves the effectiveness of the proposed method. This research could serve as a practical tool for engineers and infrastructure managers,

improving better decision-making in the maintenance and damage detection of bridge structures.

### **5.3. EXPERIMENTAL MODEL AND MODAL ANALYSIS**

An experimental bridge model was built, and modal analysis was conducted to obtain the basic structural information. Seven scenarios were designed to conduct experimental studies in this section.

#### **5.3.1 Experimental model and setup**

A 14-span spatial truss bridge with dimensions of length  $\times$  width  $\times$  height:  $7.98\text{ m} \times 0.6\text{ m} \times 0.6\text{ m}$  was constructed, left end simply supported and the right end with a roller, as shown in the Figure 3.5. The bridge consists of 160 tube steel elements and 56 joints. The bridge consisted of two kinds of tube element members, diagonal (length  $0.6\text{ m}$ ) and non-diagonal members, all with a cross-section of an inner radius of 6mm and an outer radius of 9 mm. All members consist of three segments connected by screws, and they are connected to a L-shaped semi-rigid joint by bolts.

Experimental instruments included a signal generator, a shaker, a power amplifier, a laptop, and wireless accelerometers. White noise excitation with a 10 V amplitude was produced, amplified through a power amplifier, and transferred to a shaker to induce vibrations, and the acceleration signals of the bridge were subsequently recorded by a laptop. More details about the experimental bridge and setup could be found in Section

3.3.

### 5.3.2 Sensor arrangement

Figure 5.4 illustrates the arrangement of ten sensors on the bridge, six sensors are distributed along the lower chord, and another four are attached at the middle span. Under white noise excitation, 20-minute signals of vertical (Z) directions were recorded. All signals were normalized using the Z-score method before further processing. Figure 5.5 is twenty second acceleration signal of Sensor 4 in the vertical direction.

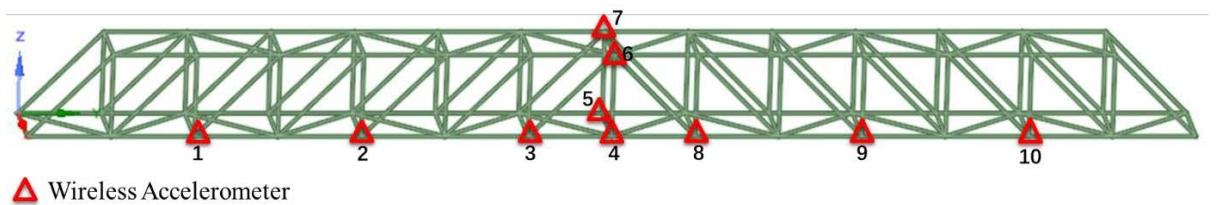


Figure 5.4 Sensor arrangement

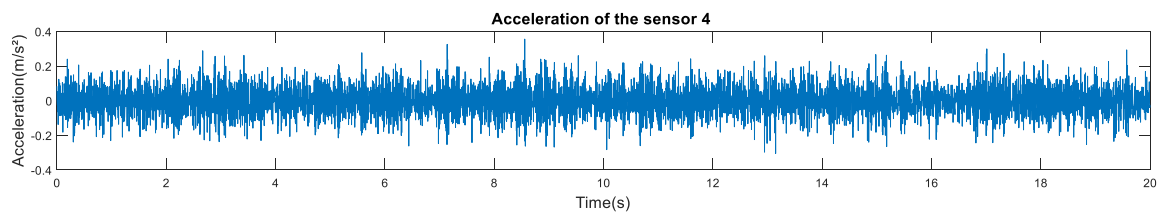


Figure 5.5 Acceleration of Sensor 4 in the vertical direction

### 5.3.3 Modal analysis

Modal analysis was conducted on the experimental bridge using ARTEMIS Modal 6.0 software. The modal analysis results are calculated based on the four sets measurement described in Section 3.3. The four sets modelling in ARTEMIS can be found in Figure 3.8(a). The stabilization diagram, provided in Figure 3.9(a) serves as a verification tool,

illustrating the stability of the estimated modes across a range of model orders, thereby confirming the accuracy of the identified natural frequencies. Table 3.2 lists the natural frequencies of the bridge's first six modes, obtained through the Stochastic Subspace Identification (SSI) method, consisting of three bending and three torsional modes.

## 5.4 NUMERICAL STUDY

### 5.4.1 The numerical bridge and model updating.

A finite element model as shown in Figure 5.6 of the bridge was constructed in Ansys Workbench 19.0, mirroring the dimensions and structural properties of the experimental bridge. The finite element model comprises 56 nodes and 160 elements, each with a tubular cross-section to simulate the steel segments used in the experiment. Details of the material properties and cross-sectional dimensions are given in Table 3.4.

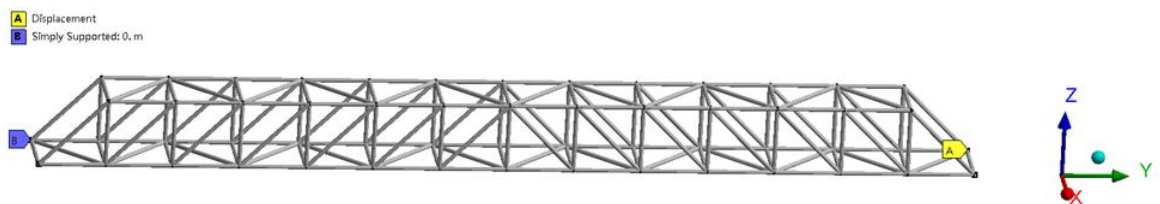


Figure 5.6 The numerical bridge model

Modal analysis was conducted on the numerical model and natural frequencies from the numerical model were compared with the experimental results. To address the natural frequency discrepancies between experimental and numerical model, model updating was performed as described in Section 3.4.3. This involved utilizing a generic element with a nonuniform cross-section member with semi-rigid joints at both ends. The updated numerical model showed markedly improved alignment with the experimental results.

The updated model serves as a fundamental model for the subsequent damage identification process.

#### **5.4.2 Data compression using the LSTM-VAE for damage quantification.**

##### *5.4.2.1 Data preparation*

A 2100-second vertical white noise excitation was applied to the fifth node from the left of the bridge as depicted in Figure 5.7. The bridge dynamic response was then recorded, sampling the vertical direction signals at 256 Hz from 12 nodes that align with the experimental sensor positions. Figure 5.8 illustrates the force excitation segment of 100 seconds and Figure 5.9 presents the corresponding raw acceleration signal from Sensor 1. An overlap ratio of 30% was applied to the total 2100-second time series to ensure thorough coverage of the dynamic behaviour. The resulting dataset for each of the seven damage scenarios, therefore, comprised 149 overlapping samples from 10 sensors, with every sample encompassing a time series of 5120 data points. Consequently, the dataset for each scenario is of size 149 samples  $\times$  10 sensors  $\times$  5120 data points. After collection, the dataset was normalized using z-score normalization, ensuring comparability and consistency for machine learning analysis.

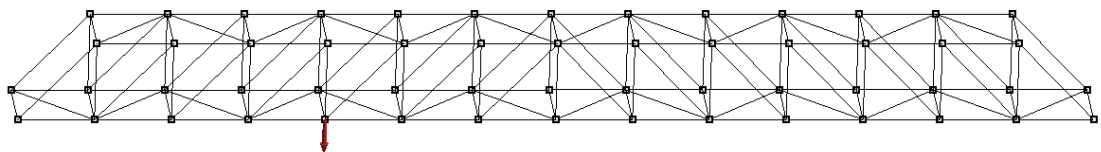


Figure 5.7 Excitation location

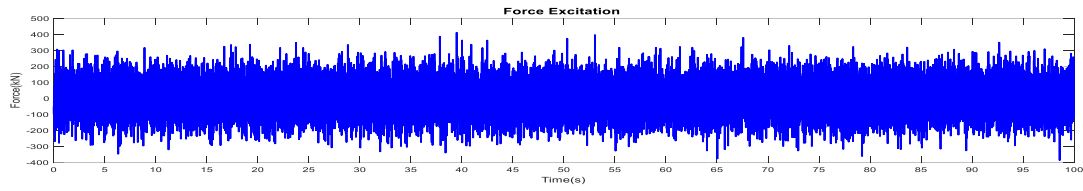


Figure 5.8 The force excitation

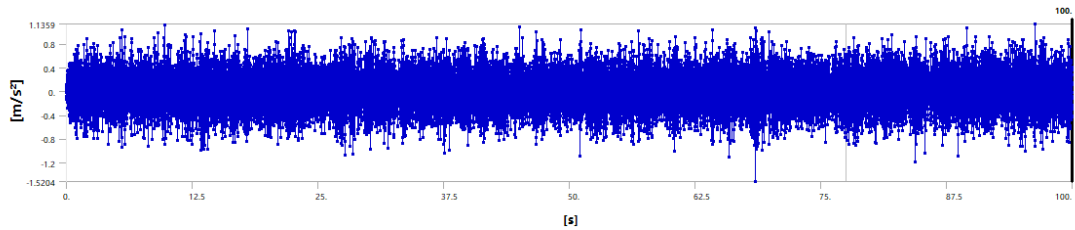
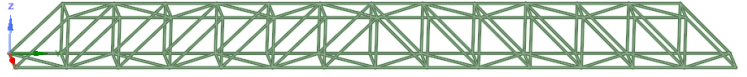
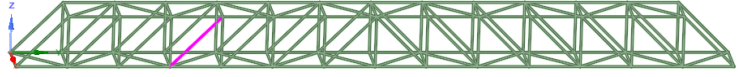
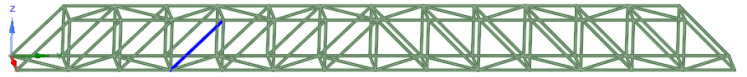
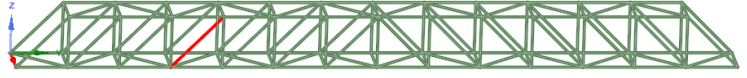
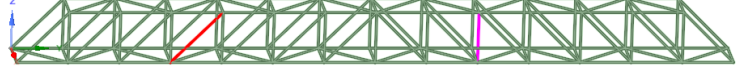
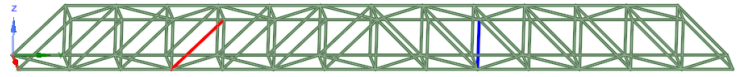
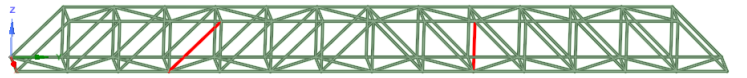


Figure 5.9 The vertical signal from Sensor 1 of the healthy scenario

Seven damage scenarios were established to reflect conditions ranging from single to double bar damage, simulated by reducing the elastic modulus of the selected elements. Table 5.1 presents the designed damage scenarios applied to the bridge model, each labelled from S1 through S7. Scenario S1 shows the bridge in a healthy state with no damage. Progressively, Scenarios S2 to S7 introduce damage at various degrees to Bar 1 and Bar 2, simulating real-world deterioration. For instance, S2 demonstrates a 35% damage degree applied on Bar 1, whereas S3 escalates this damage to 60%. The severity of damage continues to increase with S4, where Bar 1 experiences 83% damage in integrity. Scenarios S5, S6, and S7 illustrate double damages, combining 83% damage in Bar 1 with 35%, 60%, and 83% damage in Bar 2, respectively. The damage in each scenario is visually indicated on the bridge model, providing an intuitive guide to the location and level of structural damage. The damage was introduced to the bar by elastic modulus reduction. For instance, 60% damage on Bar 1 is simulated by reducing the original elastic modulus value to Young's modulus.

Table 5.1 The scenario design

Damage scenarios	Damage degrees		Location
	Bar 1	Bar 2	
S1	-	-	
S2	35%	-	
S3	60%	-	
S4	83%	-	
S5	83%	35%	
S6	83%	60%	
S7	83%	83%	

#### 5.4.2.2 LSTM-VAE Architecture

The designed LSTM-VAE architecture for this study combines a Long Short-Term Memory (LSTM) network with a Variational Autoencoder (VAE), forming a powerful tool for time-series analysis. The encoder consists of two LSTM layers with 100 and 60 neurons, which process the input sequences to capture the inherent temporal correlations and encode them into a latent space. The VAE structure effectively organizes the latent space to support generative processes, enabling the efficient reconstruction of data. The decoder, mirroring the encoder, utilizes a 100-neuron LSTM layer for this purpose.

This architecture is particularly well-suited for the high-dimensional and sequential nature of bridge monitoring data, as it compresses the time series while retaining vital structural health information. By training the LSTM-VAE with a careful selection of learning rate (0.001) over extensive training (1000 epochs), the network is trained to

discern tiny damage signatures from the sensor data. This results in a model that not only performs well in data compression but also provides an enhanced capability for damage quantification.

#### 5.4.2.3 Training Process and Testing

The LSTM-VAE model was trained on a dataset from only the healthy bridge scenario to establish a baseline for normal structural behavior. During training, the Mean Squared Error (MSE) loss function was used to monitor and optimize the network's performance. Convergence of the MSE loss, as shown in Figure 5.10, indicates the model's ability to accurately reconstruct the input time series signals.

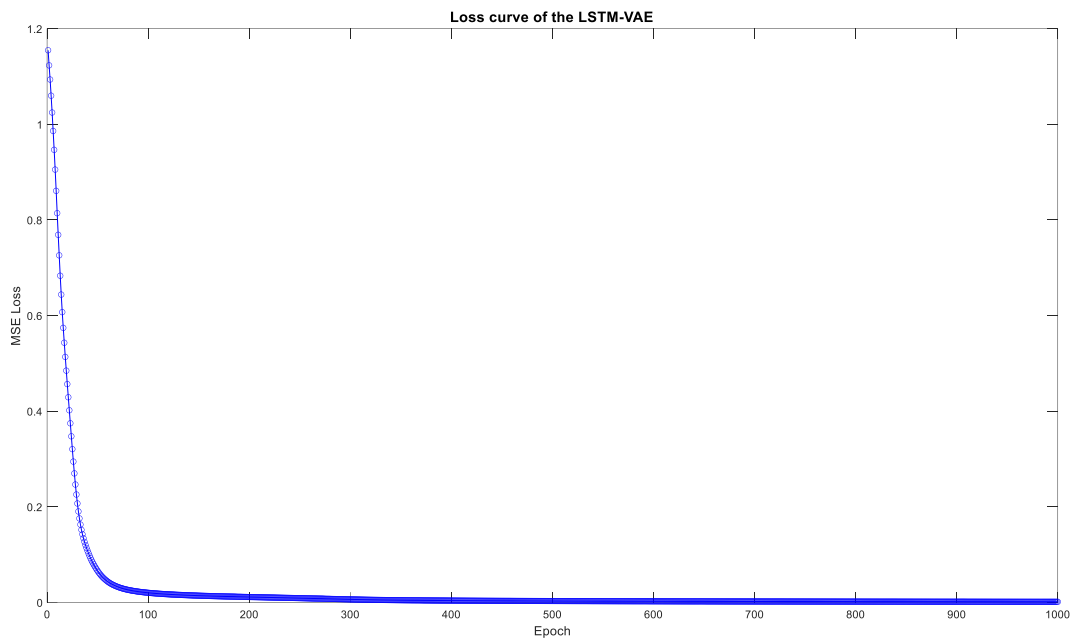


Figure 5.10 Training process

Subsequently, the model was evaluated using a testing dataset of 1043 samples, with 149 samples labeled from S1 to S7. The Average Reconstructed Error (ARE) between the original and reconstructed signals served as an indicator for damage quantification. As



depicted in Figure 5.11, the ARE increases with the level of damage, providing a quantitative measure of structural damage. This relationship was further affirmed by the Statistical ARE plots (Figure 5.12), which show a clear trend of rising ARE values corresponding to increasing damage levels.

To enhance the robustness of our findings, we analyzed the SARE using different sample sizes. The box plot using all 149 samples (Figure 5.12(a)) was compared against a reduced sample size of 20 (Figure 5.12(b)), showing that a larger dataset yields more accurate damage quantification results. Nonetheless, even with a smaller sample size, the resulting plot displayed consistent damage quantification trends.

Figure 5.13's histogram of reconstructed error offers a view of the model's error distributions. The diversity in the shape and spread of these distributions across scenarios allows for a deeper interpretation of the model's performance. Specifically, the varied distribution widths suggest differing levels of uncertainty in damage quantification, while the central error range of the distributions relate to the model's typical error magnitude in each scenario.

For further validation, a single sample from Sensor 3 was selected to demonstrate the LSTM-VAE's reconstruction accuracy. Figure 5.14 compares the original and reconstructed signals in the time domain, and Figure 5.15 shows the Fast Fourier Transform (FFT) comparison. The close matching of the peaks in the FFT plots verifies the model's ability to preserve the original essential signal characteristics.

The outcomes of this comprehensive training and testing procedure, combined with the robustness of the SARE analysis, shows the LSTM-VAE model's dependability for bridge damage identification. The consistency and reliability demonstrated by the model, coupled with the proposed ARE and SARE indicators, not only detect damage with precision but also quantify it effectively, making it a promising tool for structural damage detection.

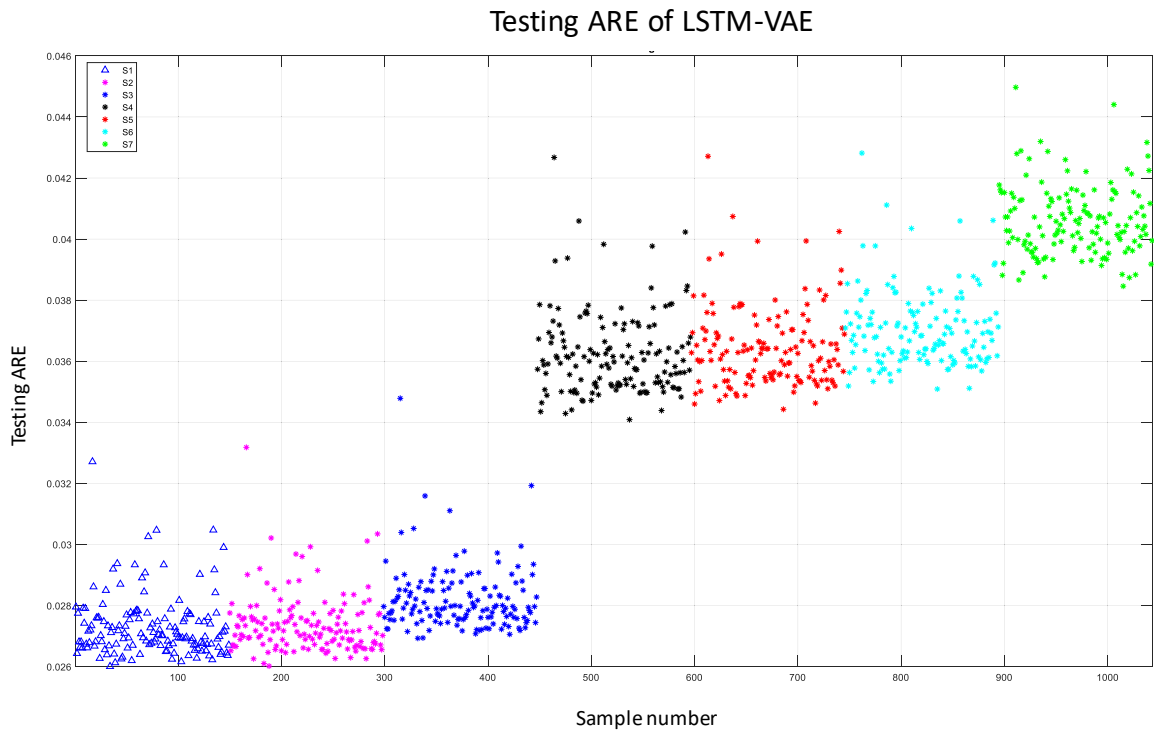
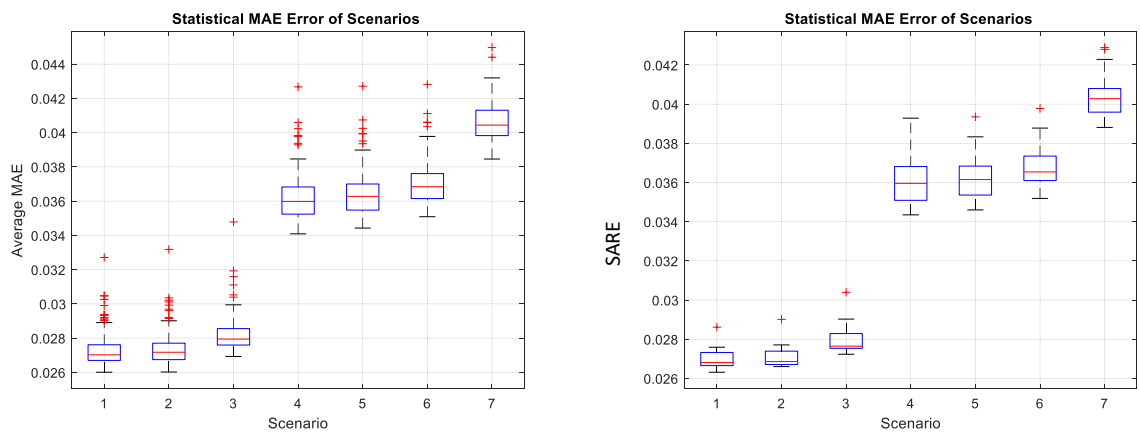


Figure 5.11 Testing ARE



(a) Statistical Box plot using all 149 samples (b) Statistical Box plot using 20 samples

Figure 5.12 Statistical ARE plot of the testing results

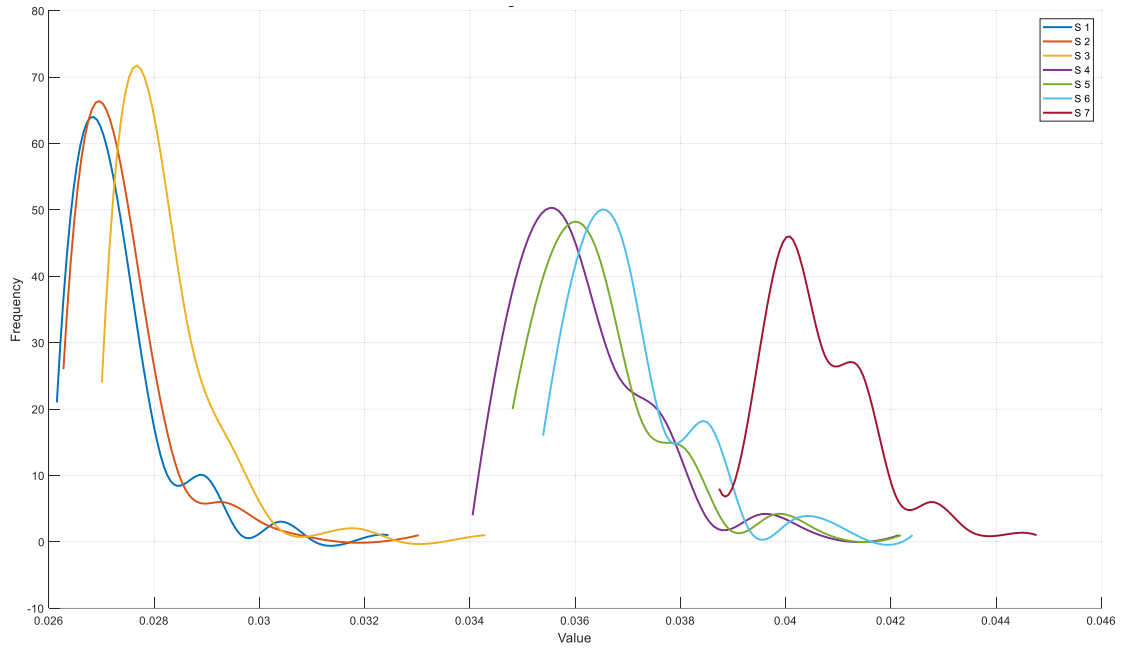


Figure 5.13 The histogram of the ARE

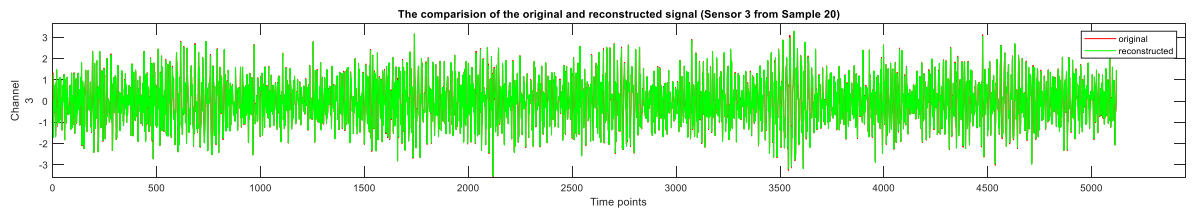


Figure 5.14 The comparison of the original and reconstructed signals

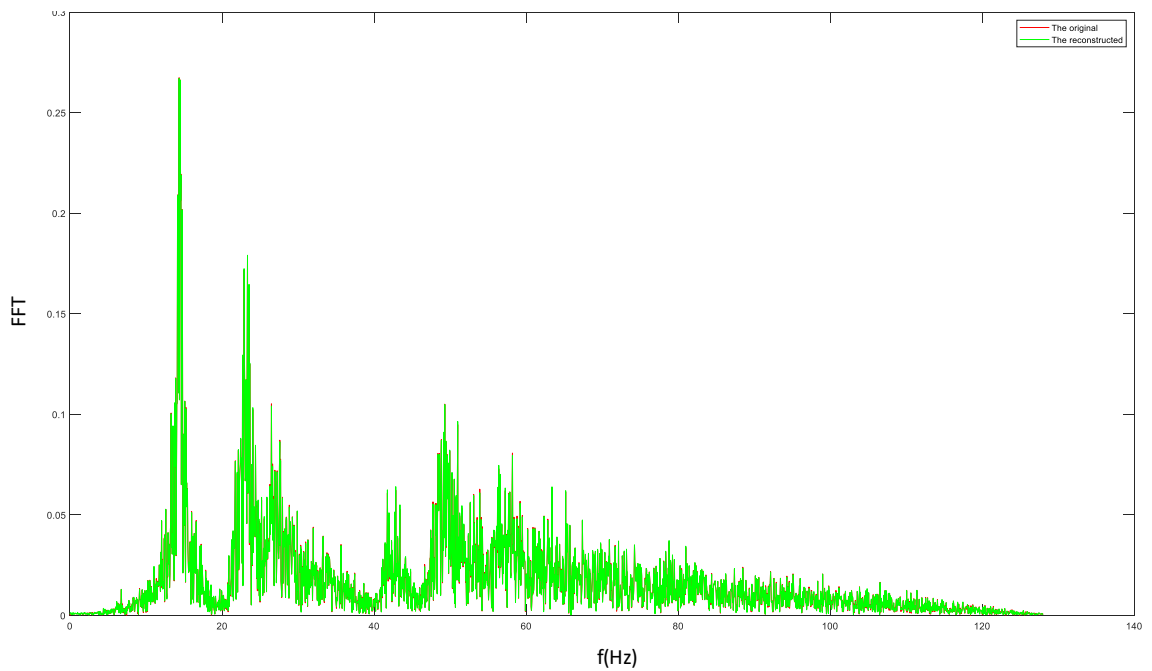


Figure 5.15 FFT Comparison of the original numerical signal and reconstructed signal (Sensor 3)

#### *5.4.2.4 Latent Representation Visualization using t-SNE*

For bridge structures with thousands of joints and elements, the interpretation and use of sensor data is critical yet challenging. In the last section, the LSTM-VAE model plays a vital role in addressing this challenge by reducing the dimension from a 10-channel signal, each channel dense with 5120 data points, to a 5-dimensional latent representation. This reduction is particularly significant in the context of bridges, enabling the processing of extensive sensor information with efficiency and retaining essential structural health information.

Once the data is compressed to latent space, the t-SNE (t-distributed Stochastic Neighbour Embedding) is employed to project the flattened latent space onto a two-dimensional space, transforming complex and abstract high-dimensional data into a visual format that is easy to interpret. By subtracting the t-SNE features of a healthy baseline from the damaged scenarios, the plot in Figure 5.16 presents the different damage scenarios (S1 to S7), allowing for a direct visual comparison of the degrees of damage.

The combined use of LSTM-VAE and t-SNE offers a practical solution for the health monitoring of large bridge infrastructures. It enables a rapid, clear assessment of structural health across a bridge's expanse, potentially averting structural failures. By providing a visual map of structural health, this method allows stakeholders to make data-driven decisions with greater confidence and precision.

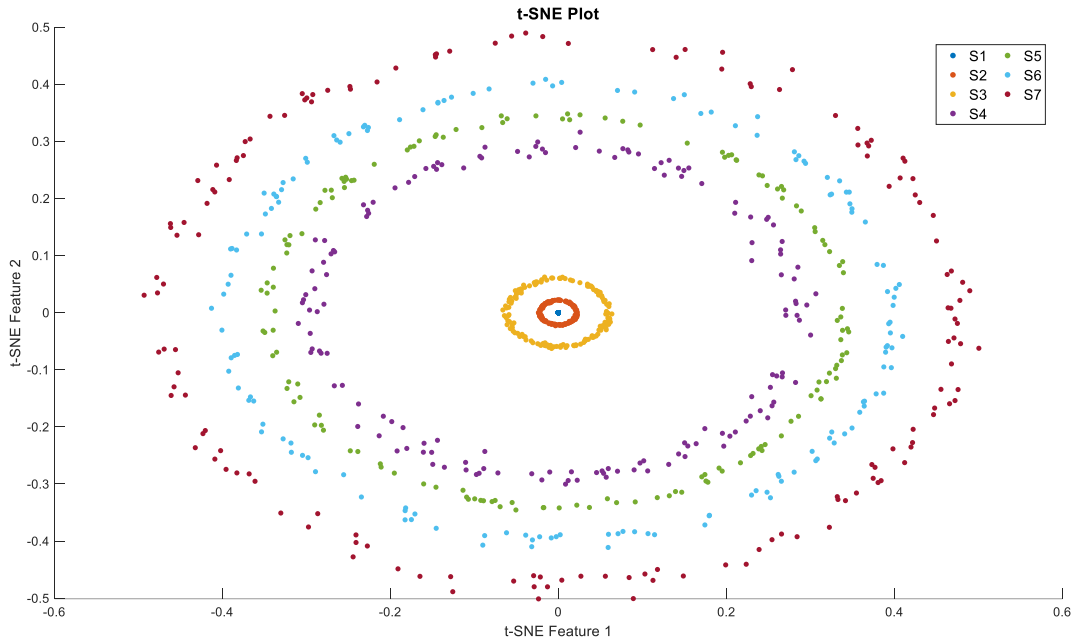


Figure 5.16 t-SNE features

### 5.4.3 Semi-supervised learning for damage classification

For the classification of bridge scenarios, two-dimensional t-SNE extracted from latent representation from Section 5.4.2 is employed as damage features. This technique proved effective for scenario classification, using the features for a clearer separation of damage states. 90% of the total 1043 samples is used for training and validation while the remaining 10% is for testing.

In evaluating various classification models, we considered Support Vector Machines (SVM), Ensemble Bagged Trees, and Fine Trees. Based on the comparison of validation and testing accuracies, as shown in Table 5.2, the SVM model achieved the best performance, achieving a validation accuracy of 99.9% and a test accuracy of 100%. This exceptional performance shows the suitability of SVM for this semi-supervised learning

task, particularly when dealing with the high-dimensional, complex datasets typical in bridge health monitoring.

The confusion matrices from the SVM validation and testing phases, depicted in Figure 17(a) Validation and (b) Testing, further verify the effectiveness of the SVM approach. The validation phase exhibited high classification with a single misclassification, while the testing phase demonstrated perfect classification across all numerical scenarios. These results highlight the robustness of the SVM model in distinguishing between varying degrees of damage, enhancing its application as a reliable tool for semi-supervised learning in structural health monitoring.

Table 5.2 Validation and test accuracy of semi-supervised methods

Classification Model	Validation accuracy	Test accuracy
SVM	100%	100%
Ensemble bagged trees	91.3%	96.2%
Fine tree	84.3%	86.5%

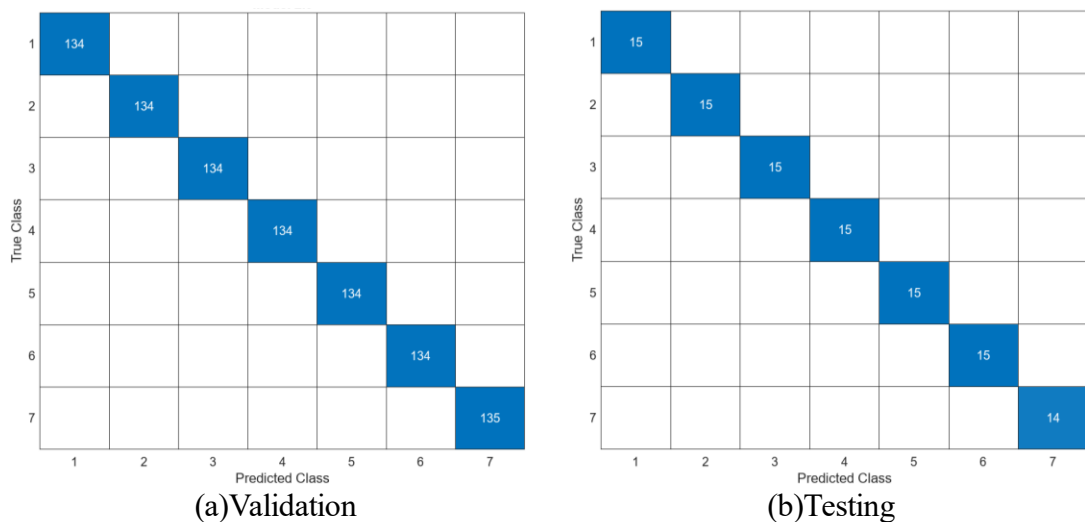


Figure 5.17 Confusion Matrix using SVM for classification

The robustness of the semi-supervised classification method was thoroughly evaluated

by introducing noise at various levels, e.g. 10%, 20%, and 30% to the whole original signal dataset. This evaluation is designed to reflect potential real-world disruptions where environmental noise may affect data integrity. The noisy datasets were firstly sent to the LSTM-VAE to extract latent representations, subsequently processed with t-SNE, and finally tested using the trained SVM classifier.

Figure 5.18 compares the original signal to one with 30% added noise. The test confusion matrices, displayed in Figure 5.19, prove the highly robust performance of the LSTM-VAE semi-supervised method. The results of high testing classification accuracies of 100% for 10% noise, 99.8% for 20% noise, and 96.1% for 30% affirm the proposed method's high precision and robustness, illustrating its ability to consistently deliver accurate classifications despite the presence of increasing levels of noise.

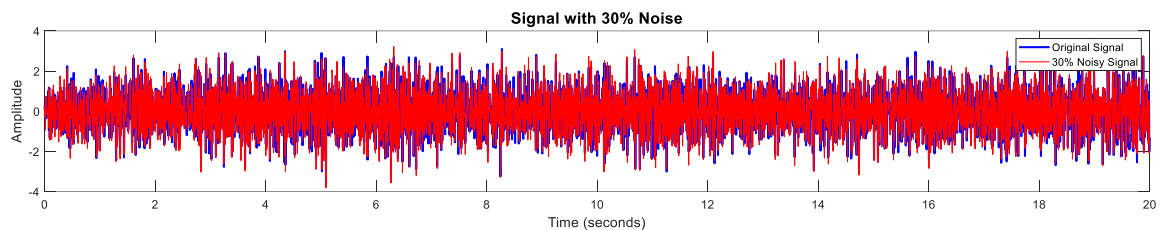


Figure 5.18 Comparison of the original signal and 30% noise signal

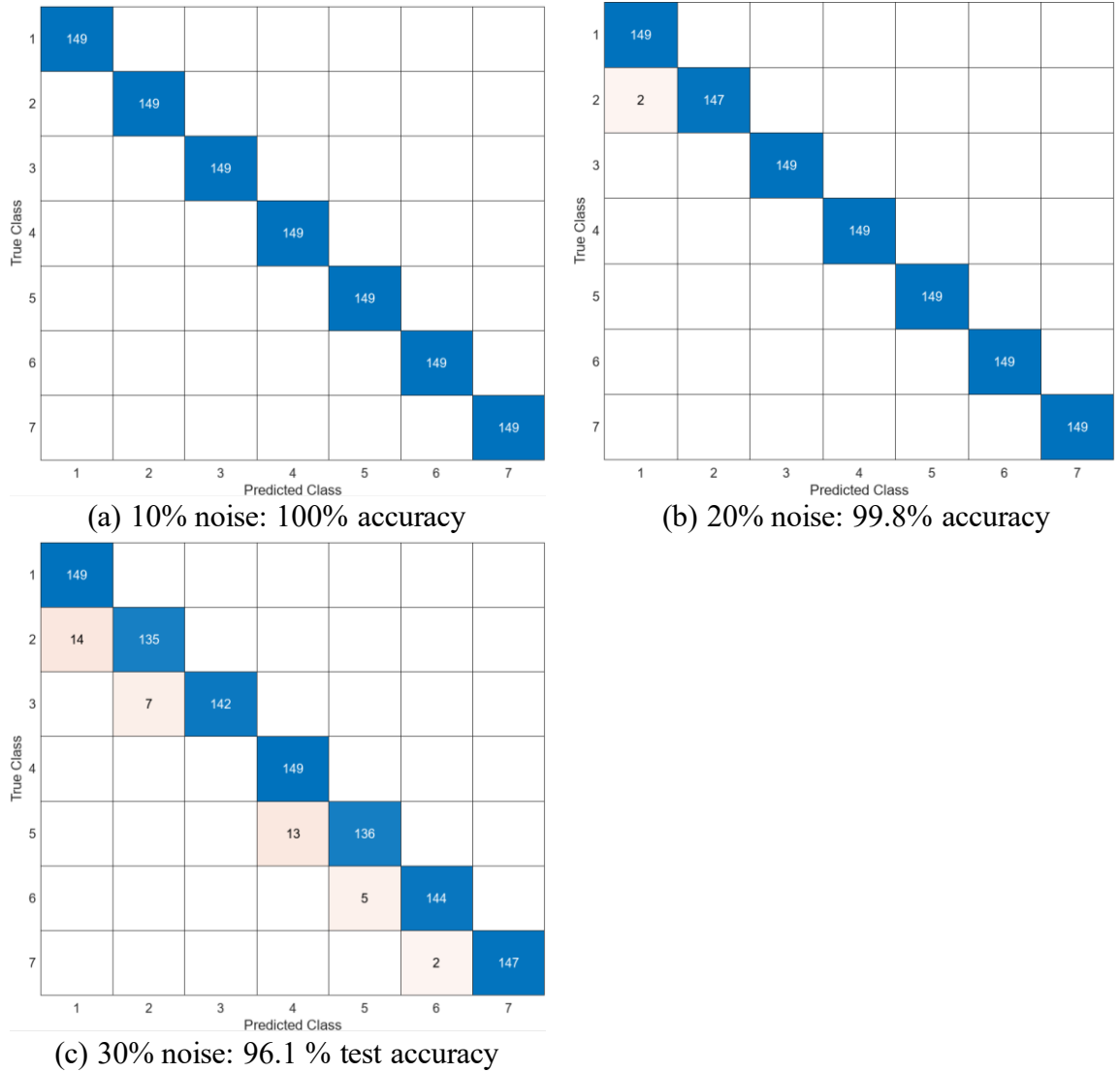


Figure 5.19 Testing Confusion matrix using noisy signals

## 5.5 EXPERIMENTAL STUDY

### 5.5.1 Experimental study

The experimental study was designed to validate the LSTM-VAE semi-supervised method's efficacy. Following the setup described in Section 5.3, we simulated damage by substituting structural elements of the experimental bridge with bars of reduced cross-sections, as depicted in Figure 5.20. The original bars were replaced by bars of smaller cross sections with smaller moments of inertia to simulate the damaged scenarios. The



experiment involved capturing acceleration signals under multiple damage scenarios, as detailed in Table 5.1. Beginning with the intact S1 scenario, we applied white noise excitation and recorded the acceleration response. Subsequently, Bar 1 was replaced with a smaller cross-section bar, representing the S2 damage scenario, as illustrated in Figure 5.20. This process was iterated for each damage case, culminating in seven scenarios, with signal acquisition and recording undertaken for each. In the experimental procedure, the bridge was subjected to white noise excitation by a shaker, and the response signals were recorded for 20 minutes for each damage scenario. These signals were then segmented into 85 samples for each scenario, employing a 30% overlap ratio to enrich the dataset and enhance the continuity across samples.



Figure 5.20 bar replacements and dimensions

## 5.5.2 Experimental verification

### 5.5.2.1 LSTM-VAE

In the verification phase, the Long Short-Term Memory Variational Autoencoder (LSTM-VAE) model was trained using a dataset consisting of healthy scenarios from both numerical simulations and experimental models, 234 samples in total—149 from the numerical model and 85 from experiments. The model's architecture and parameters were

kept consistent with those used in the preliminary numerical study. The training loss curve, depicted in Figure 5.21, shows a satisfactory reduction over epochs, indicating the model's learning progress.

After the training phase, all experimental signals —595 samples across 10 sensors, each with a time series of 5120 points—were used to test the LSTM-VAE to assess the model's ability of damage quantification and signal reconstruction. The ARE between the original and reconstructed signals was calculated to serve as for damage quantification. As illustrated in Figure 5.22, the scatter plot shows individual ARE values for each sample, revealing variances in reconstruction. Notably, Figure 5.23 provides a clear visual indication that the SARE tends to increase with the severity of damage in subsequent scenarios. This trend serves as an important confirmation of the model's sensitivity to damage, although an interesting deviation is observed in Scenario 4, where the SARE does not follow the incremental pattern seen in other scenarios. Although the severity of damage increases from S1 to S7, it is important to note that the Statistical ARE (SARE) does not show a perfectly consistent rise, as demonstrated by the values for S4 and S5. This discrepancy may be attributed to the complex nature of how damage impacts different structural components, where variations in local stiffness might influence the recorded responses. Further investigation into these factors could provide insights into refining the model's performance in quantifying damage severity more consistently.

The consistency between the original and reconstructed signals, particularly for samples from Scenarios 1 (S1) and 7 (S7), is analysed in Figures 5.24 and 5.25. These figures

demonstrate that the model can accurately reconstruct signals, with a strong correlation between the original and the reconstructed data. This capability is further validated by the Fourier Transform comparisons in Figure 5.26, which show that the reconstructed signals retain essential structural information, a critical aspect of accurate signal representation.

Moreover, the testing of LSTM-VAE on all experimental datasets of 595 samples, enabled the extraction of latent representations on samples from damaged scenarios. These representations were condensed into a two-dimensional space through the T-distributed Stochastic Neighbour Embedding (T-SNE) technique, facilitating the upcoming classification task. This dimensionality reduction is a crucial step towards using a semi-supervised method to classify the states of structural damage.

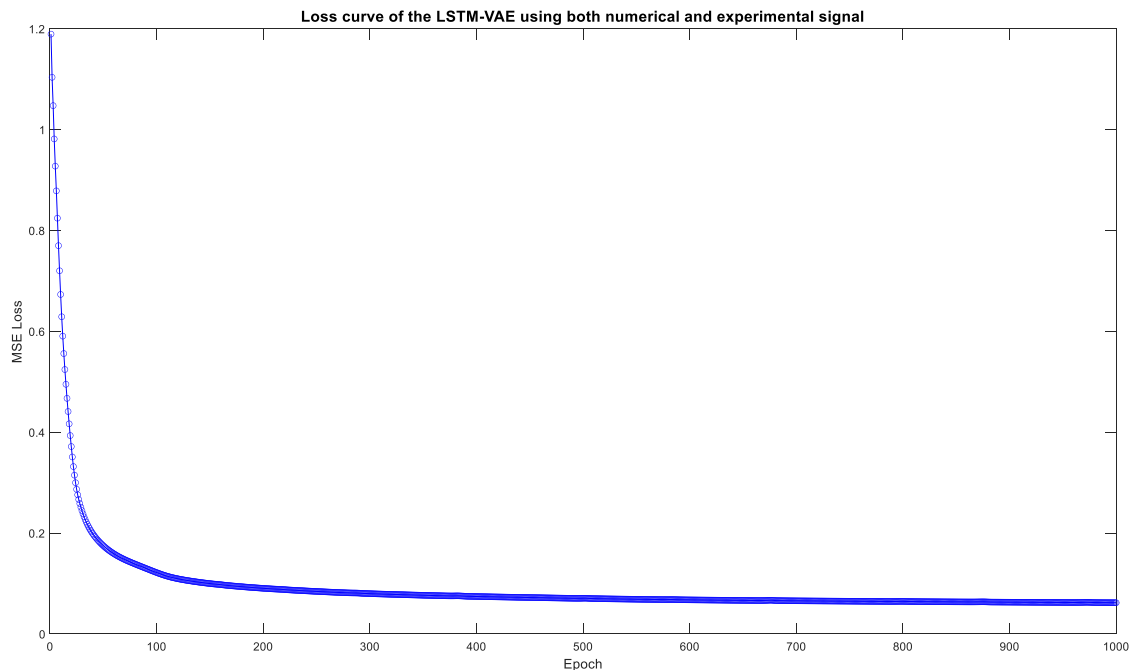


Figure 5.21 Training curve

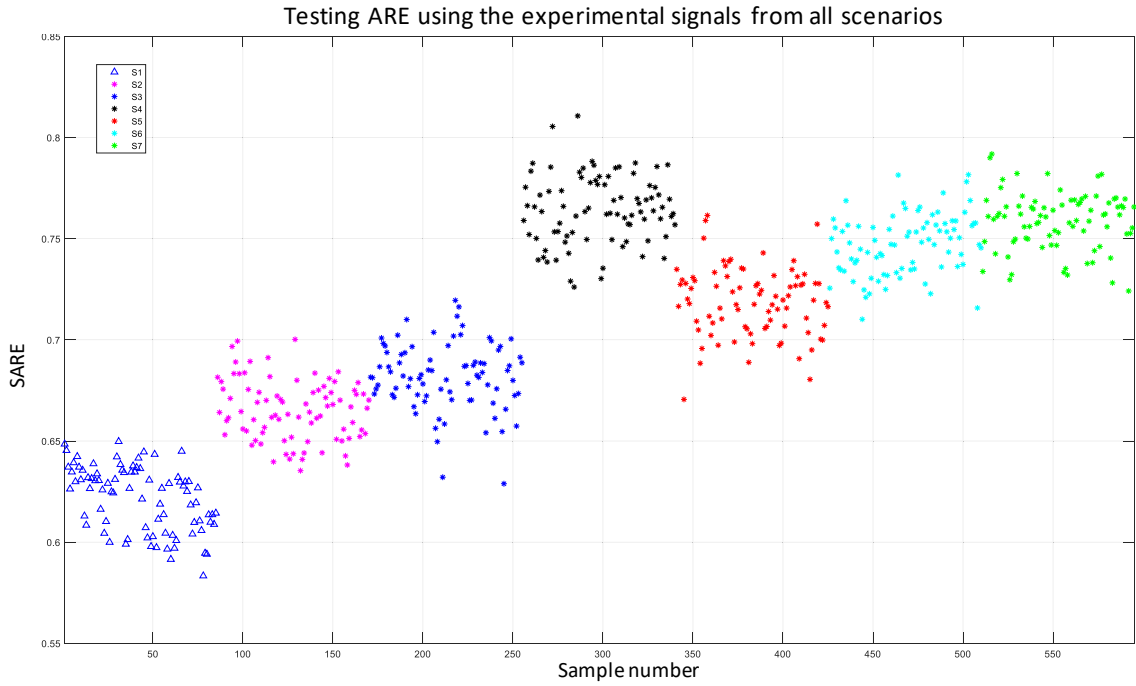


Figure 5.22 Testing ARE using the experimental signals from all scenarios.

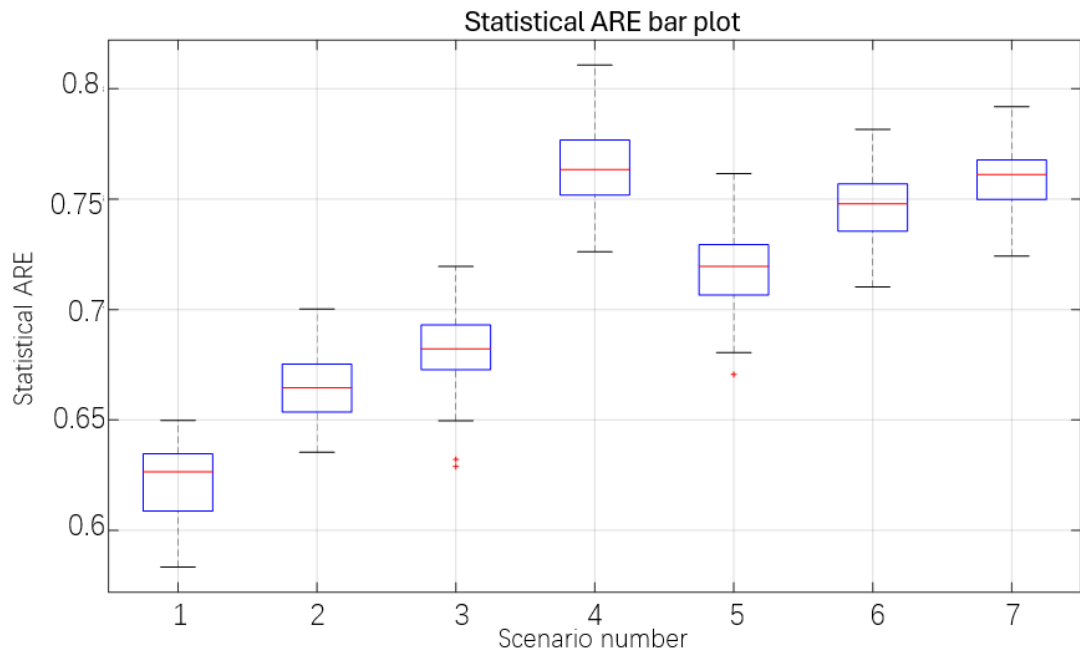


Figure 5.23 Statistical ARE bar plot



Figure 5.24 The reconstructed sample and original signal (Sample 20 from Scenario 1)

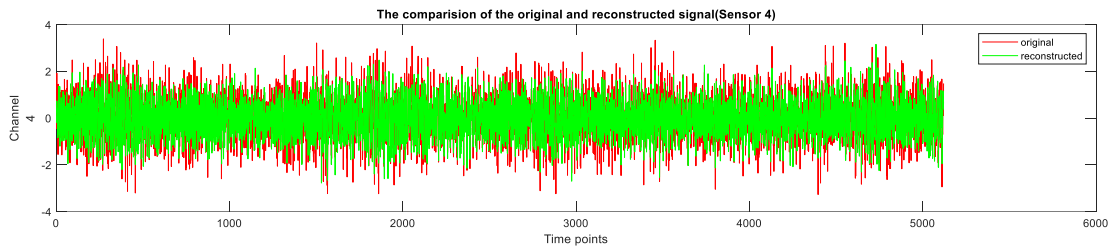


Figure 5.25 The reconstructed sample and original signal (sample 500 from Scenario 7)

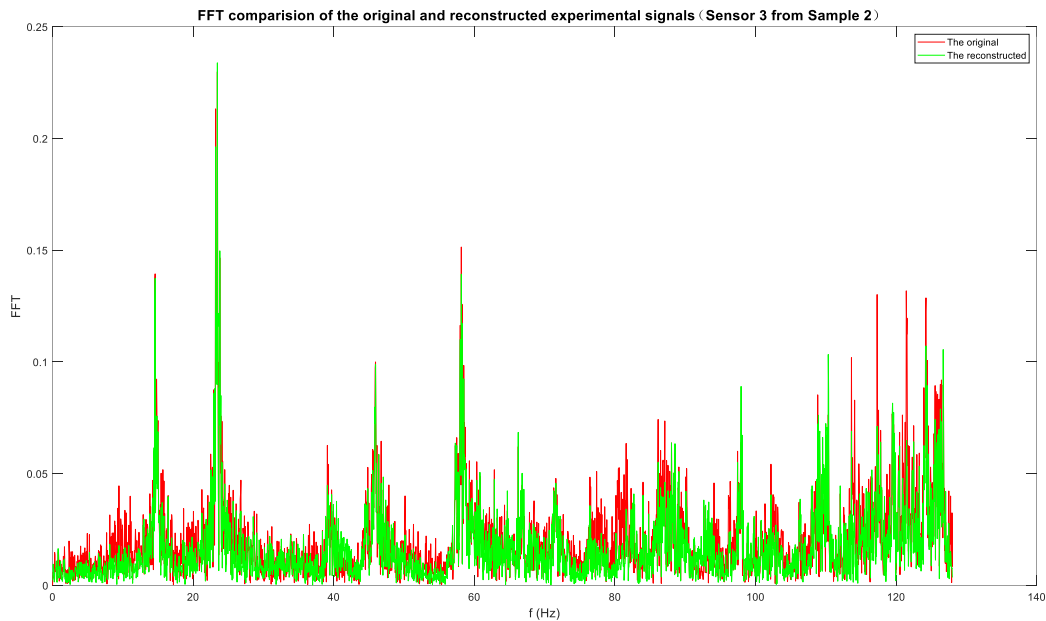


Figure 5.26 FFT comparison

### 5.5.2.2 Semi-supervised classification

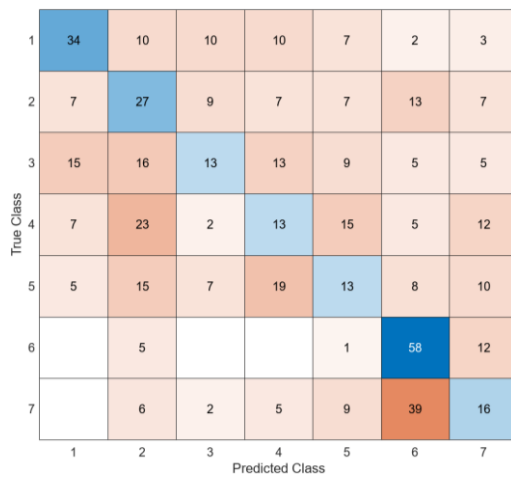
Dimensionality reduction on the latent features from all scenario samples was performed using the t-SNE method. The feature set was reduced from the 5-dimensional sensor data (each with a 5120-point time series) to a smaller 2-dimensional t-SNE space. The reduced

feature set was then classified using an SVM. 90% of the samples (535 out of a total of 595) are used for training the classifier and the remaining 10% for testing. The classification results showed an accuracy of 32.5% for validation and 32.5% for testing, as detailed in Figure 5.27. The inclusion of an additional dimension with 3D t-SNE, shown in Figure 5.28, led to better classification accuracy, with validation and testing accuracies improving to 41.0% and 35.6%, respectively. This suggests that the additional dimension provided the SVM with more information within the data, crucial for effective classification.

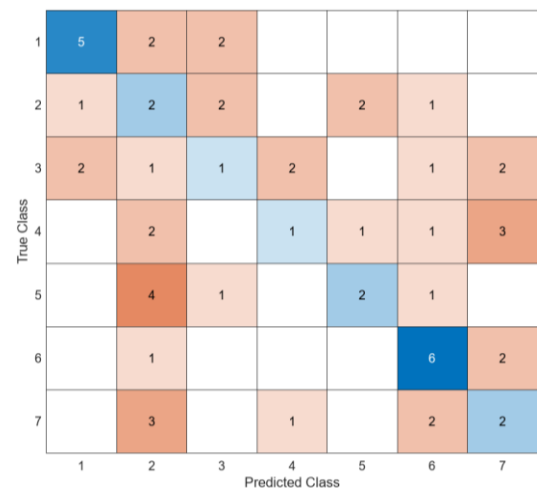
A notable improvement in classification performance was achieved by combining the 3D t-SNE features with the ARE. This combination resulted in a substantial increase in accuracy, with validation and testing accuracies rising to 68.8% and 71.2%, respectively (Figure 5.29). The effectiveness of this integrative approach confirms that combining the t-SNE feature of latent representations with error metrics like ARE enhances the model's ability to classify different scenarios.

Upon reviewing Figures 5.27, 5.28, and 5.29, it is evident that certain scenarios were more challenging to classify compared to the numerical scenarios, due to the similarity in damage patterns or the presence of noise within the signals. Specifically, the matrices indicate that while the model performs well in distinguishing certain scenarios, there is room for improvement in others, especially where the number of false positives and false negatives is higher.

The improved outcomes with the integrated feature set, particularly when incorporating the ARE, highlight the proposed method's potential in experimental scenarios. These results underscore the viability of the method for practical applications, demonstrating its capacity to effectively discern structural states in a diverse dataset, thus proving its value for real-world structural health monitoring.

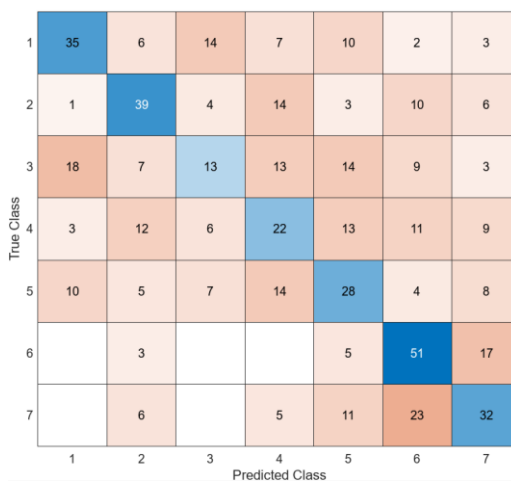


(a) Validation accuracy 32.5%

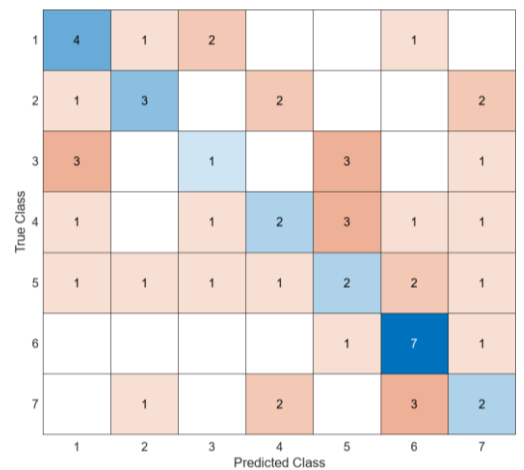


(b) Testing accuracy 32.2%

Figure 5.27 Confusion Matrix with 2d t-SNE feature

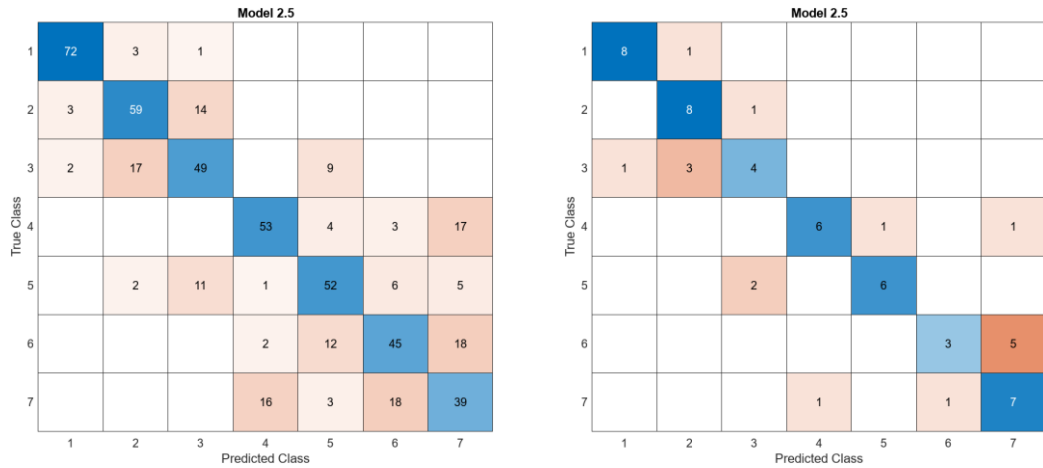


(a) Validation accuracy 41.0%



(b) Testing accuracy 35.6%

Figure 5.28 Confusion Matrix with 3d t-SNE feature



(a) Validation accuracy 68.8% (b) Testing accuracy 71.2%  
 Figure 5.29 Confusion Matrix with 3d tSNE and ARE feature

## 5.6 SUMMARY

This section introduces an innovative approach to bridge health monitoring, designed for complex structures characterized by numerous joints and elements. The novelty of this study lies firstly in the introduction of Average Reconstructed Error (ARE) as a novel index for damage quantification, providing a direct measure of structural damage quantification. Further, the study proposes a semi-supervised learning framework that combines t-SNE with ARE features, achieving a notable 71% accuracy in damage classification on experimental scenarios. The efficacy and applicability of the proposed methods are rigorously validated using an 8-meter experimental bridge model, showing their practicality for real-world scenarios. The results suggest that this approach could contribute to the enhancement of maintenance strategies for bridges, providing a step towards more data-driven management of structural health.



# **CHAPTER 6 CWT BASED DAMAGE QUANTIFICATION OF BRIDGES USING CONVOLUTIONAL VAE**

## **6.1 OVERVIEW**

This chapter introduces an unsupervised learning approach to quantify bridge damage through a convolutional Variational Autoencoder (VAE). A numerical investigation was conducted to explore the ability of the VAE to reconstruct the Continuous Wavelet Transform (CWT) based images and detect damage from the original CWT images, as indicated by the Average Reconstructed Error (ARE). The experimental study is conducted to verify the feasibility and practicability of the VAE method on bridge damage detection, using actual sensor data to detect damage in a physical bridge model. This chapter comprehensively demonstrates the unsupervised VAE method's capability for damage quantification using the numerical to experimental study.

## **6.2 METHODOLOGY**

This section outlines the methodology used for bridge damage detection using a CWT-based Convolutional Variational Autoencoder (VAE). VAE is an unsupervised machine learning model that combines principles from convolutional neural networks (CNNs) and probabilistic models, as detailed in Section 5.2.1.1. The CWT based VAE architecture employed in this section consists of an encoder and a decoder, with the purpose of learning compact representations of the input CWT data and reconstructing them for

comparison.

In the encoder phase, latent features are extracted from the input Continuous Wavelet Transform (CWT) images, which represent the time-frequency characteristics of the bridge's structural responses. These latent features are passed through a probabilistic distribution to encode the input into a latent space with defined probabilistic properties. This allows for more robust handling of structural variations in the data, enabling better generalization when detecting structural damage.

In the decoding phase, the latent representations are reconstructed into the original CWT image format, allowing for a direct comparison between the original and reconstructed images. The Average Reconstructed Error (ARE) and Statistical ARE (SARE), as presented in Eqs. (5.1) and (5.2), between these images serves as the primary indicator of structural damage. Higher ARE and SARE values suggest greater discrepancies between the original and reconstructed signals, indicative of more severe damage. This approach makes the CWT based Convolutional VAE particularly suited for identifying subtle but progressive damage states that might not be detectable using traditional methods.

The CWT process is conducted using the Morlet wavelet function in MATLAB, which is widely applied for signal analysis due to its time-frequency localization properties. The Morlet wavelet is a product of a complex exponential function and a Gaussian window, expressed as (Yan & Miyamoto, 2006)

$$\psi(t) = \pi^{-\frac{1}{4}} e^{j\omega_0 t} e^{-\frac{t^2}{2}} \quad (6.1)$$

where  $w_0$  is the central frequency of the wavelet. The CWT is implemented through the convolution of the wavelet function with the signal, represented as (Kim & Melhem, 2004)

$$CWT(s, \tau) = \int x(t) \frac{1}{\sqrt{s}} \psi^* \left( \frac{t-\tau}{s} \right) dt \quad (6.2)$$

with  $s$  as the scale factor and  $\tau$  as the time shift, and  $\psi^*$  the complex conjugate of the wavelet function. This process not only shows the time-frequency characteristics of the signal but also provides necessary analytical dimensions for damage detection.

By combining CWT for signal representation and convolutional VAE for reconstruction and anomaly detection, this methodology effectively captures both localized and global damage patterns in the bridge structure.

## 6.3 NUMERICAL STUDY

### 6.3.1 Numerical model and data preparation

A numerical model was built in Ansys workbench19.0 and it was updated considering element stiffness updating and joint stiffness updating, as proposed in Chapter 3. The parameters of this model can be found in Section 5.4.1. With this updated numerical model, a white noise excitation of 2100 seconds was applied on the fifth node from the left of the bridge of the lower chord to collect the structural acceleration responses. The white noise excitations were generated as a Gaussian distribution function with a zero mean and a standard deviation of one in MATLAB. These signals were then amplified by a factor of 100 to simulate the magnitude of forces. Due to the computation limitations,

the generated force was divided into twenty-one segment signals, each consisting of a 100-second signal. An example of 100 seconds generated force can be seen in Figure 6.1. Each of these excitation segments was applied on the numerical model one by one to record the corresponding vertical acceleration responses at 10 locations for seven scenarios as designed Table 5.1. Figure 6.2 uses the green points to indicate the positions where accelerations are recorded. An example the vertical acceleration response from the middle span is given in Figure 6.3.

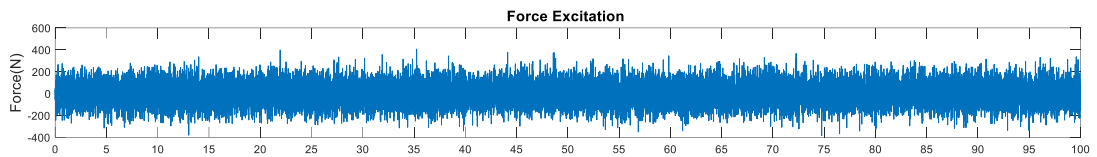


Figure 6.1 The Gaussian white noise force excitation of 100 seconds

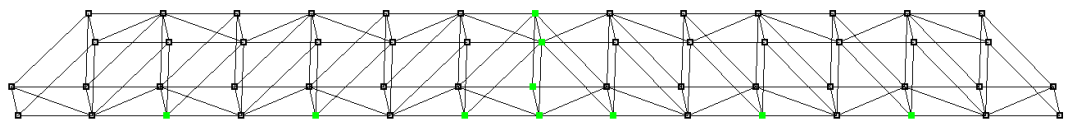


Figure 6.2 Recorded acceleration locations

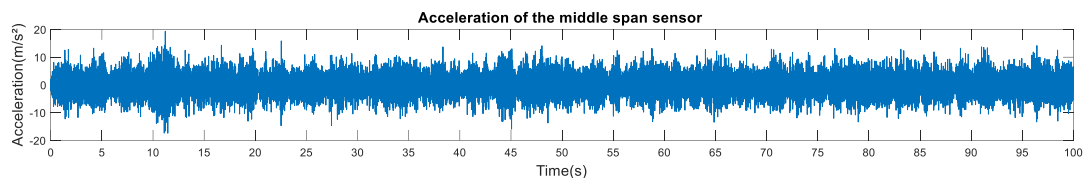


Figure 6.3 The vertical acceleration response from the middle span location

Each location's (called channel in the following) data is segmented into overlapping time windows, each lasting 20 seconds, with a sampling rate of 256 Hz, resulting in a high-resolution data structure of 5120 points per segment. This overlapping window process, with a 30% overlap ratio. This operation results in 149 samples for each scenario with seven scenarios in total.

Normalization of the signals is performed using the Z-score method, standardizing the

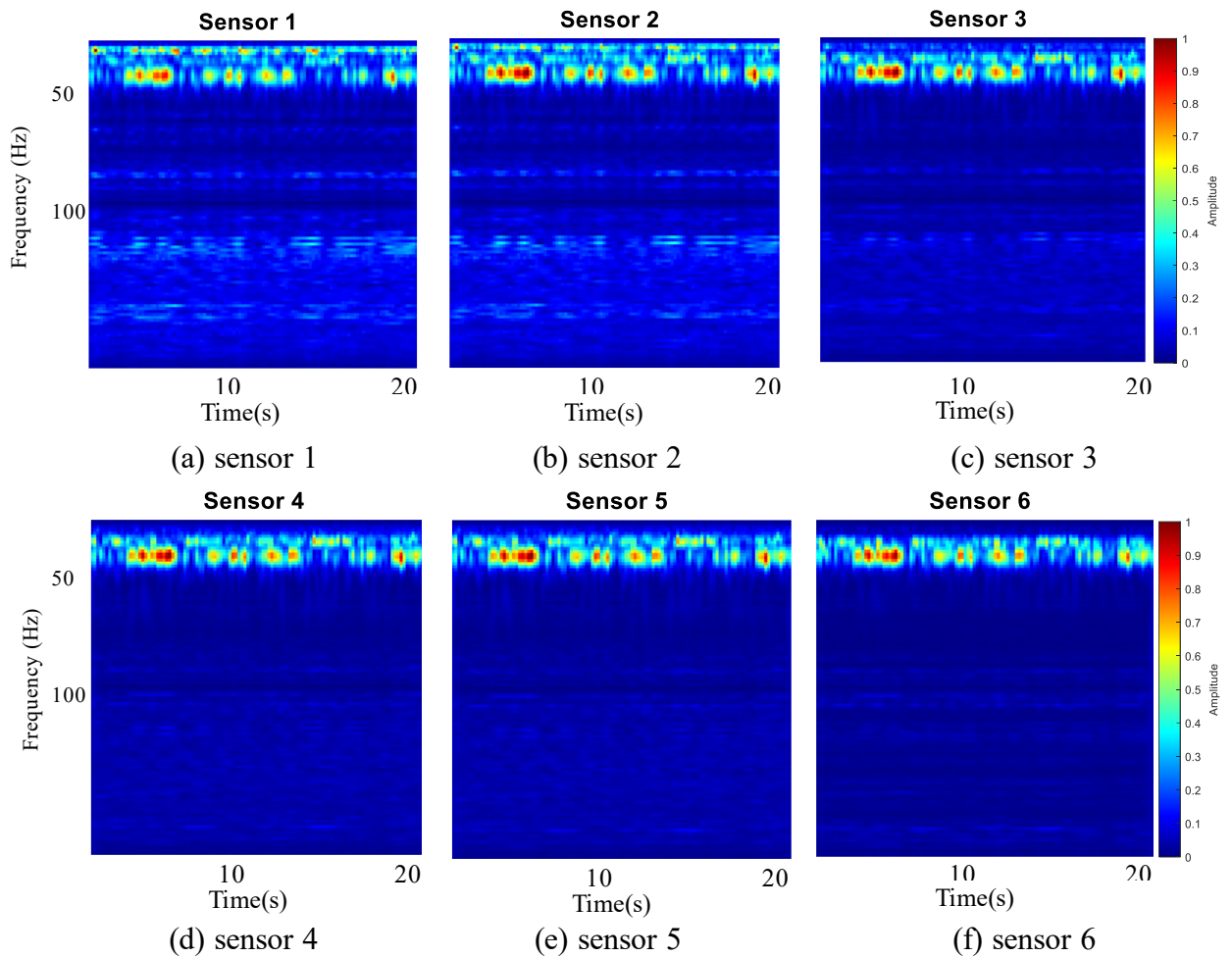
data to have zero mean and unit variance, thereby eliminating potential bias due to varying signal magnitudes across the different samples. The normalized data segments are then processed with the wavelet transform.

The processing is a multi-step procedure designed to transform the raw acceleration signal into a format suitable for a Variational Autoencoder (VAE). The following data manipulation strategy detailed in this section presents the conversion of one-dimensional acceleration data into two-dimensional time-frequency representation images using Continuous Wavelet Transform (CWT) presented as Eqs. (6.1) and 6.2 in Section 6.2. This conversion is crucial for capturing both temporal and frequency characteristics inherent in structural responses, transforming each sample of the one-dimensional signals to two-dimensional images which are then resized to a size of 112 by 112 pixels, a suitable size for the input layer of the convolutional VAE. Also, each CWT image is normalized to ensure pixel intensity values range between 0 and 1 and then stored as training and testing datasets of the VAE model.

The dataset encompasses 1043 samples, with each sample corresponding to 20-second segments from a total signal duration of 2,100 seconds, captured by ten sensors. These signals are normalized and transformed into CWT images, each with a dimension of 112×112 pixels, forming a 4D dataset across ten sensor channels. Figure 6.4 shows the CWT images representing the time-frequency characteristics of the structural response. The x-axis (originally time) and y-axis (originally frequency) have been resized to fit the 112x112 input dimensions required for the VAE model, and thus, represent relative time

and frequency scales rather than exact values. The colour indicates the magnitude of the wavelet coefficients, with brighter colours representing higher energy content in the signal.

The organized and structured approach to data processing outlined is made for structural health monitoring, ensuring that the raw structural response data is converted into a refined format, ready for the application of damage detection and analysis methods in machine learning techniques.



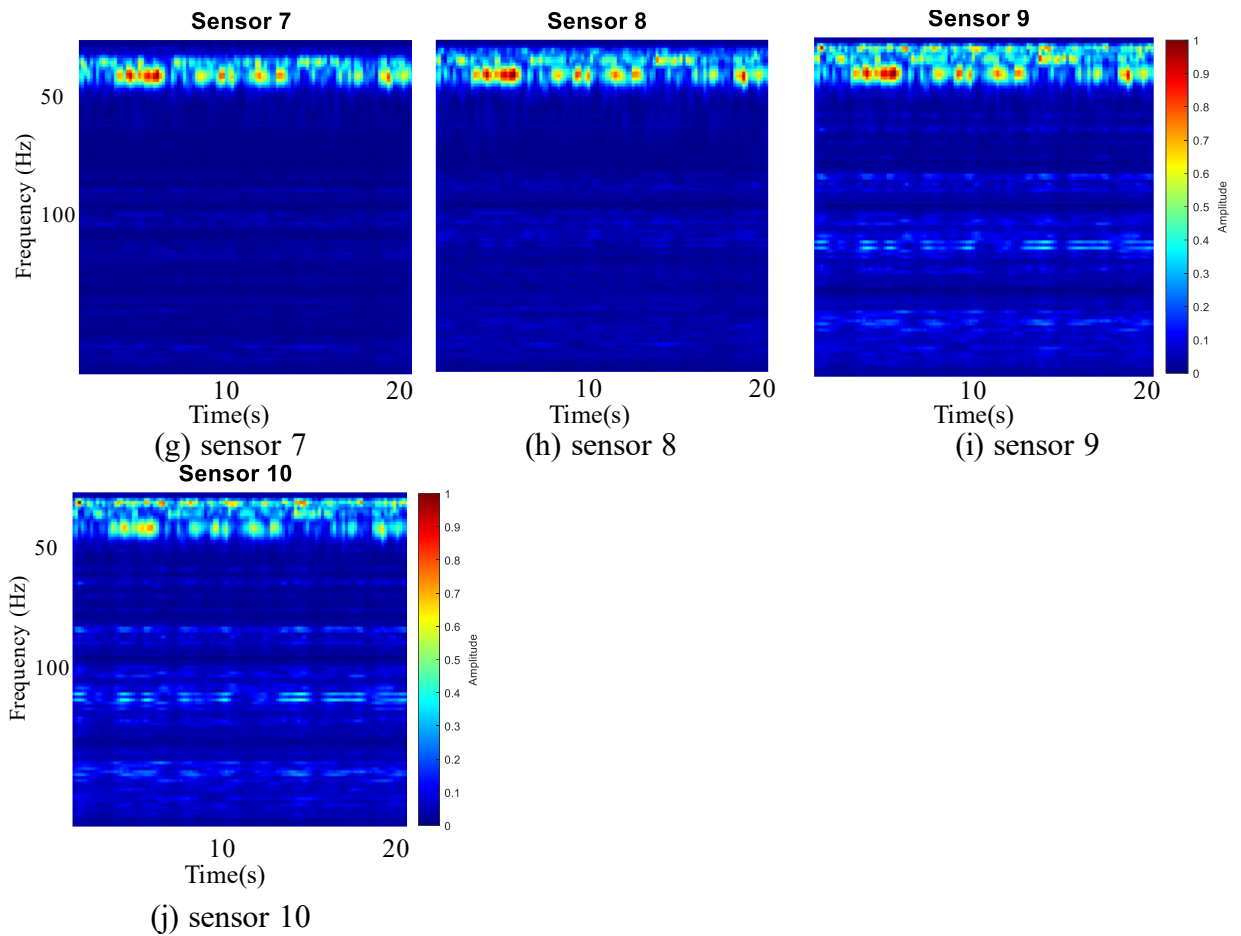


Figure 6.4 CWT images across different sensors (random sample 10)

(X-axis: Time (seconds), Y-axis: Frequency (Hz), Amplitude: colour)

### 6.3.2 Convolutional VAE network

The convolutional VAE architecture, as shown in Table 6.1, uses a non-symmetric structure with an encoder featuring convolutional layers, a fully connected layer, and ReLU activations for potent non-linear data transformation. It includes a subsequent mean and log-variance sampling stage for latent space projection. In contrast, the decoder reconstructs the CWT images using transposed convolutional layers, concluding with a sigmoid activation to match the output to the original signal distribution's size.

With the 1024 samples from seven scenarios, only those from the healthy scenario are trained. All other samples from six damaged scenarios are used for testing. During training, an Adam optimizer is employed to refine the VAE's parameters, with initial settings critical for achieving a stable and effective learning process. The training data, consisting of 149 samples from the bridge's undamaged state, is shuffled at each epoch onset to enhance model robustness. Loss value is recorded by presenting mean squared error (MSE) loss across the 1,000 epochs, ensuring the VAE's learning accuracy in reconstructing the healthy state of the bridge. By using the undamaged scenario samples for training, the VAE establishes a benchmark for the bridge's baseline state, which is critical for subsequent damage identification.

Table 6.1 The convolutional VAE Architecture

Network	Layer	Activations
Encoder	Image input	112(S) × 112(S) × 10 (C) × 1 (B)
	Convolutional 1	56 (S) × 56(S) × 32 (C) × 1 (B)
	Relu	56 (S) × 56(S) × 32 (C) × 1 (B)
	Convolutional layer 2	28 (S) × 28(S) × 64 (C) × 1 (B)
	Relu	28 (S) × 28(S) × 64 (C) × 1 (B)
	Fully connected layer	1 (S) × 1 (S) × 128 (C) × 1 (B)
	Mean and log-variance sampling	-
Decoder	Feature input	64 (C) × 1 (B)
	Projection and reshape layer	28 (S) × 28(S) × 64 (C) × 1 (B)
	Transposed Convolutional Layer 1	56 (S) × 56(S) × 64 (C) × 1 (B)
	Relu	56 (S) × 56(S) × 64 (C) × 1 (B)
	Transposed Convolutional Layer 2	112(S) × 112(S) × 32 (C) × 1 (B)
	Relu	112(S) × 112(S) × 32 (C) × 1 (B)
	Transposed Convolutional Layer 3	112(S) × 112(S) × 10 (C) × 1 (B)
	Sigmoid	112(S) × 112(S) × 10 (C) × 1 (B)



### 6.3.3 Numerical results discussion

Figure 6.5 presents the training loss curve of the VAE, showing decreasing loss over epochs, proving the model's convergence. After the training, all samples (147 samples  $\times$  7 scenarios) are sent to the trained VAE, outputting the reconstructed CWT images. Differences between undamaged and damaged states were evaluated by the Average Reconstructed Error (ARE) (Eq. 6.1) as plotted in Figure 6.6 showing higher ARE values indicative of more severe damage.

From Figure 6.6, it can be seen that the ARE values generally increase with damage severity across the seven scenarios, ranging from approximately 0.01 for Scenario 1 (the healthy case) to around 0.013 for Scenario 7 (the most damaged case). Additionally, Figure 6.7 presents the Statistical Average Reconstructed Error (SARE) values, where the median SARE for each scenario shows a clear upward trend, particularly from Scenario 4 to Scenario 7. The median SARE increases from approximately 0.011 for Scenario 1 to around 0.013 for Scenario 7, confirming the model's ability to quantify increasing damage.

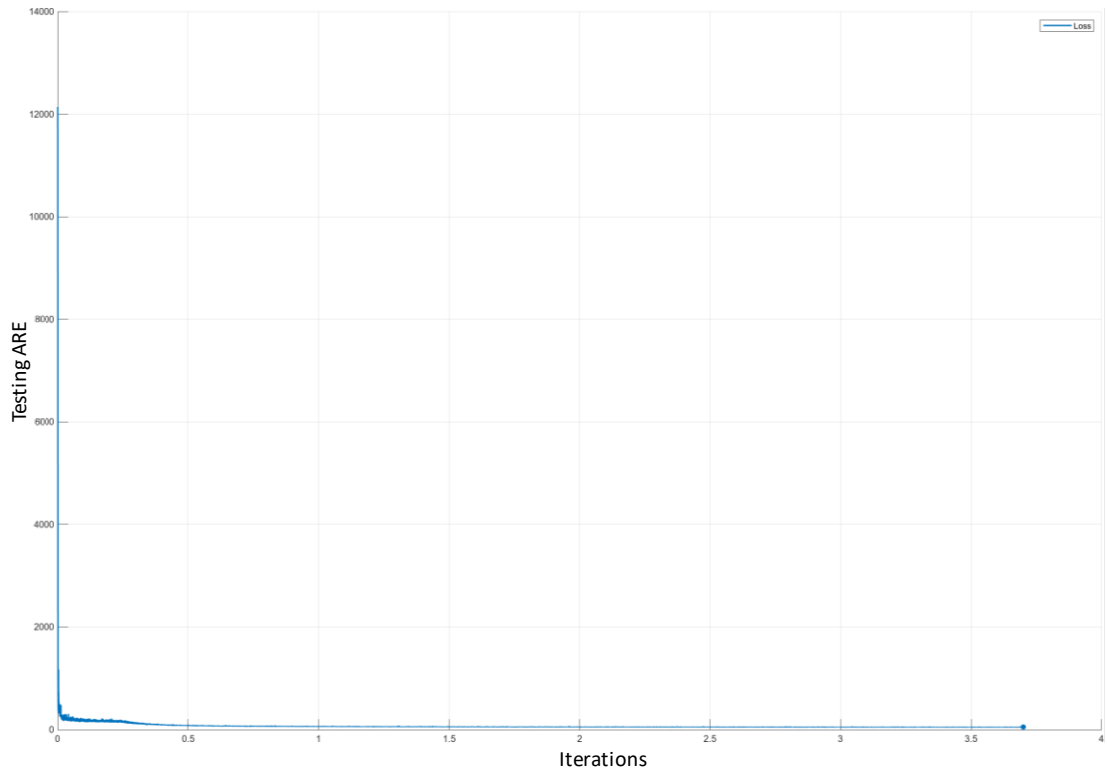


Figure 6.5 The training process of VAE

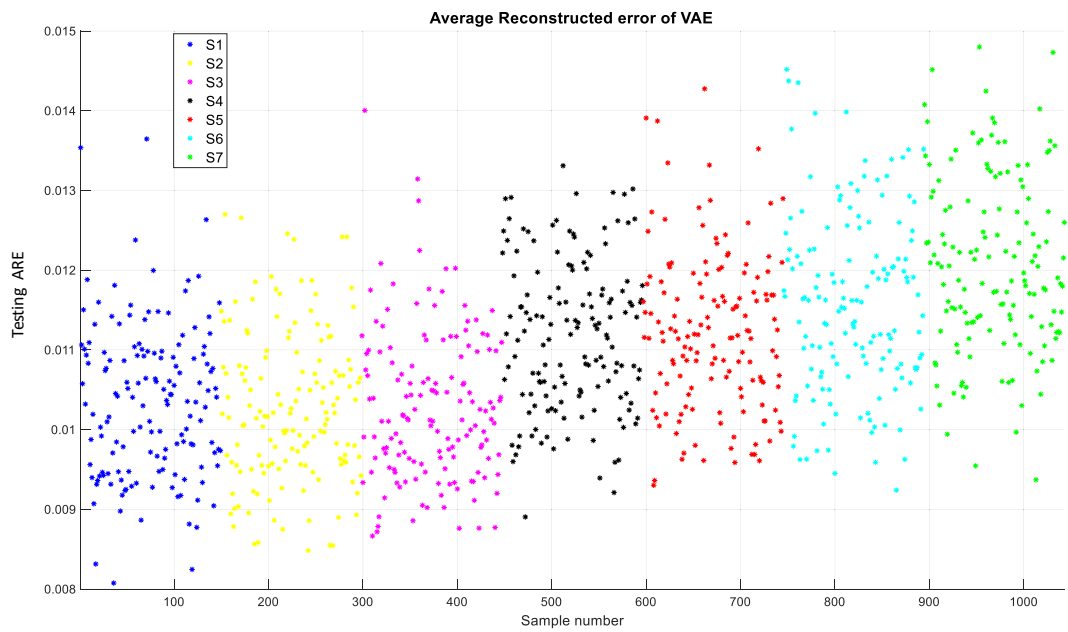


Figure 6.6 Damage quantification using Average Reconstructed Error(ARE)

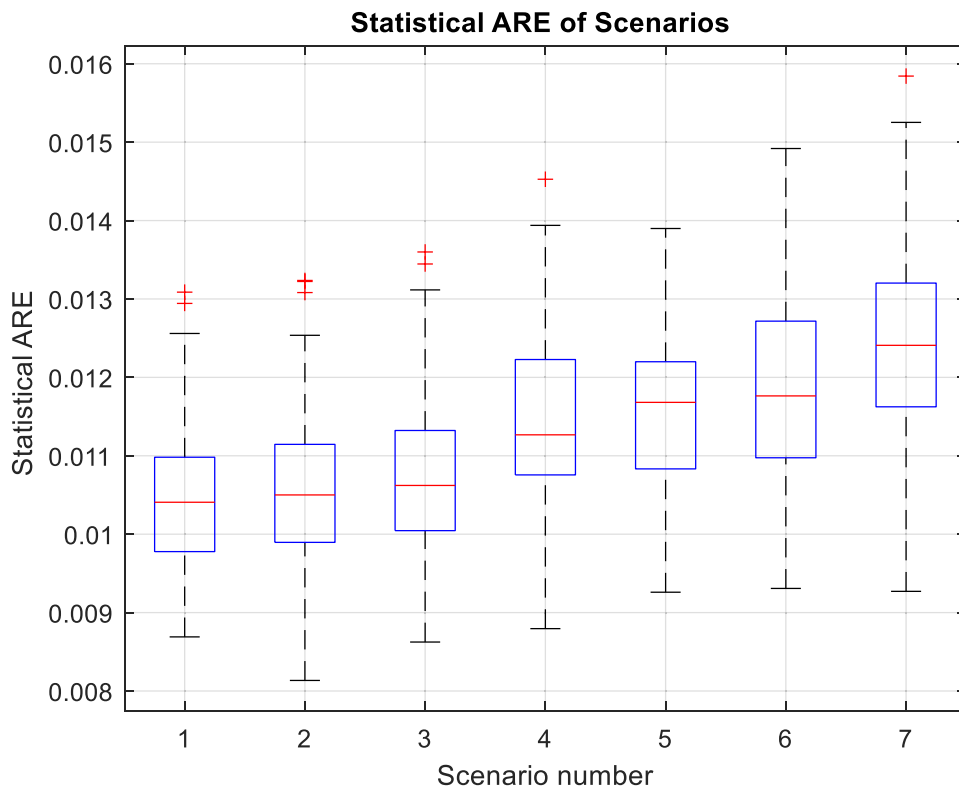


Figure 6.7 Box plot of SARE

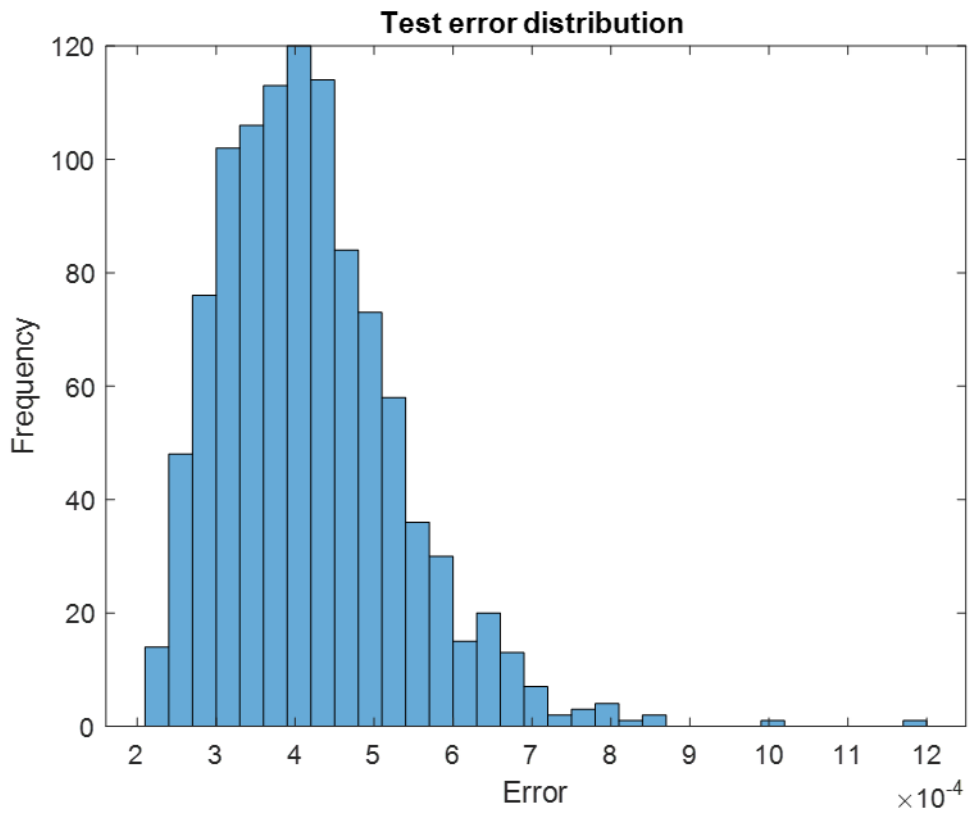
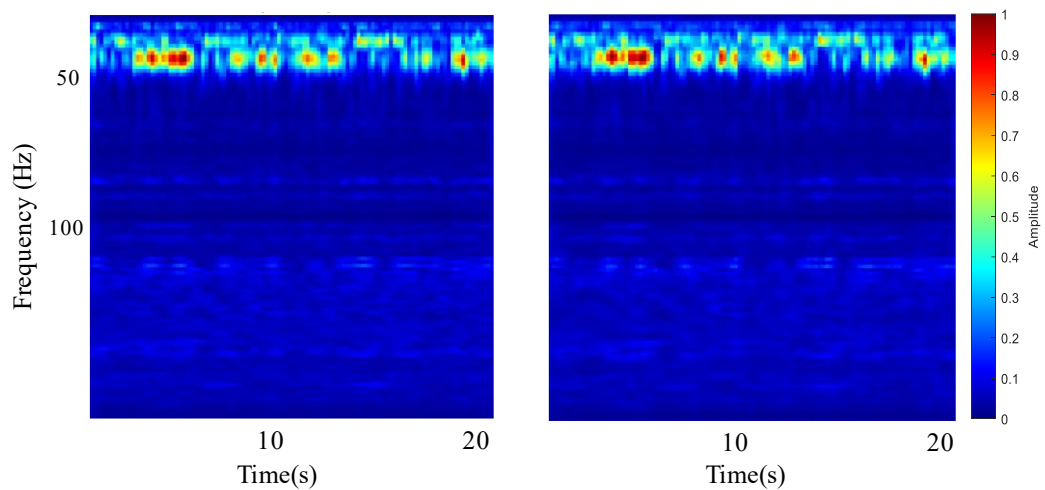


Figure 6.8 The testing average reconstructed error of VAE distribution

Figure 6.8 is the test error distribution for the VAE, with a histogram skewed towards

lower error values. This suggests that the VAE generally reconstructs the CWT images with high confidence. A tail of higher error values signifies less frequent, yet critical instances of potential structural anomalies that need further inspection.

The performance of the VAE in accurately reconstructing CWT images is visually depicted in Figure 6.9, where the original CWT image (a) is next to the reconstructed CWT image (b). The comparison showcases the VAE's high performance in replicating the detailed energy patterns of the undamaged bridge state, improving the model's utility in structural health monitoring applications.



(a)The original CWT image (b)The reconstructed CWT image  
 Figure 6.9 Comparison of the original and reconstructed CWT images (Sensor 3)

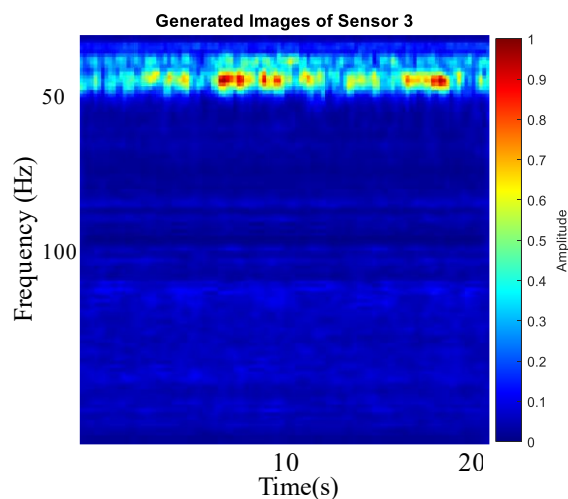


Figure 6.10 Generating new CWT images using VAE

The Variational Autoencoder (VAE) not only reconstruct but also to generate new CWT images that are representative of the learned latent space. For example, Figure 6.10 shows a generated image from Sensor 3, generated by the VAE from a random vector. The significance of this generation process lies in its contribution to data augmentation. By generating additional CWT images, the model enhances the robustness of the dataset, potentially improving the VAE's ability to generalize and aiding in the identification of damage patterns not explicitly present in the original dataset. This augmented dataset can be helpful in refining the model's predictive accuracy and in providing a more comprehensive understanding of the bridge's structural health.

In summary, this numerical analysis validates the Convolutional VAE's capability to detect and quantify bridge damage via CWT image analysis. The model's architecture could capture complex signal patterns, and it is unsupervised, focusing on training undamaged bridge samples, coupled with validation, proved its precision in quantifying damage severity.

## **6.4 EXPERIMENTAL STUDY**

### **6.4.1 The experimental data collection**

An 8-meter experimental model was built in the laboratory, consisting of 160 tube steel elements connected by 56 joints. Experimental devices include a signal generator, a shaker, a power amplifier, a laptop, and ten wireless accelerometers. More detailed information including dimensions of the bridge and tube steel elements, and devices

parameters could be found on Section 5.3.1.

White noise excitation with a 10 V amplitude was produced by the signal generator, amplified through a power amplifier, and transferred to a shaker to induce vibrations, and the acceleration signals of the experimental bridge were subsequently recorded by a laptop.

Like the numerical study in Section 6.3, seven bridge scenarios including one healthy scenario are considered, described in Table 5.1. The bridge under various damage scenarios is simulated by substituting bridge elements with replacements of reduced cross-section. Dimensions of the original element and replacements of seven scenarios are shown on Figure 5.20. For each scenario, 20 minutes responses from the bridge were recorded by wireless accelerometers following the sensor arrangement in Figure 5.4. Figure 6.11 is 20 seconds of the experimental acceleration signal recorded from the location at middle span at lower chord from healthy bridge state.

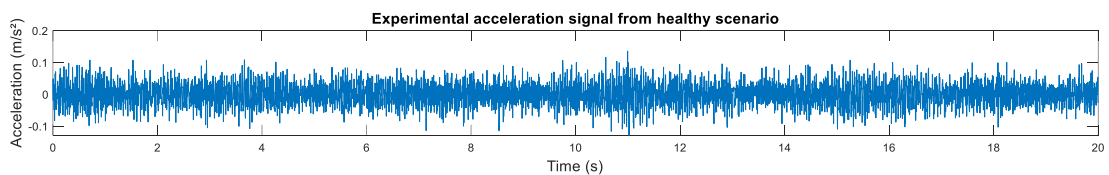


Figure 6.11 Experimental acceleration signal from healthy scenario

To effectively use the experimental data for structural health monitoring, an overlapping window technique is used. Specifically, a 30% overlap ratio is applied to the twenty-minute acceleration signals from each scenario, dividing them into eighty-five samples, each consisting of a duration of twenty seconds, the same length as the samples from the

numerical study. Across all seven scenarios, this results a total of 595 samples. These samples are uniformly normalized using the Z-score method to mitigate any potential distortion from varying signal magnitudes across the different channels. Subsequently, these samples are subjected to the same Continuous Wavelet Transform (CWT) methodology as used in the numerical study in Section 6.2.

The CWT process involves the transforming the original one-dimensional acceleration signals into two-dimensional CWT images of 112 by 112 pixels, which are then used for the unsupervised Variational Autoencoder (VAE) training. This standardized and consistent treatment of the experimental signals ensures that the subsequent VAE phase is grounded on a robust and uniform dataset, important for the accurate damage detection.

The CWT images from a randomly selected sample of ten sensors are displayed in Figure 6.12, showing the time-frequency representations captured by each sensor. Although these images might appear to be reshaped due to the nature of the CWT process, they still effectively represent the vibration patterns induced by the white noise excitation, showcasing the CWT's sensitivity in detecting structural changes.

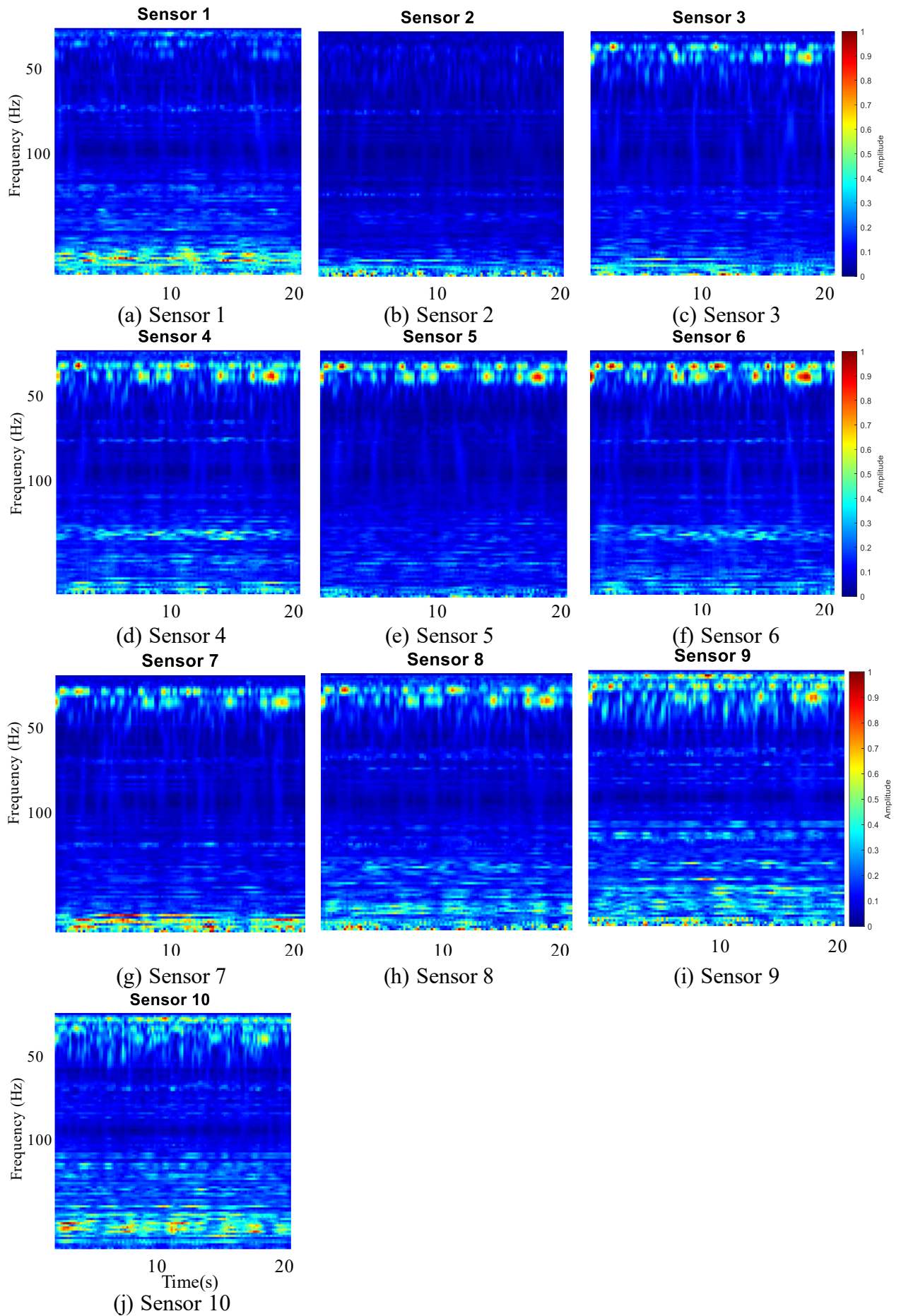


Figure 6.12 CWT image across different sensors (random sample 10) of the experimental bridge



## 6.4.2 Experimental verification

The training dataset of the VAE consists of 149 CWT images from numerical healthy bridge and 85 CWT images derived from the experimental setup in the healthy state. This combination of experimental data can be seen as an enhancement or fine-tuning of the VAE, ensuring that the model is fed to realistic baseline conditions.

The VAE training process, illustrated in Figure 6.13, shows the VAE's training curve, which quickly converges, indicating successful adaptation to the healthy state data. The VAE's ability to reconstruct the baseline state and identify deviations associated with damage was refined throughout this process.

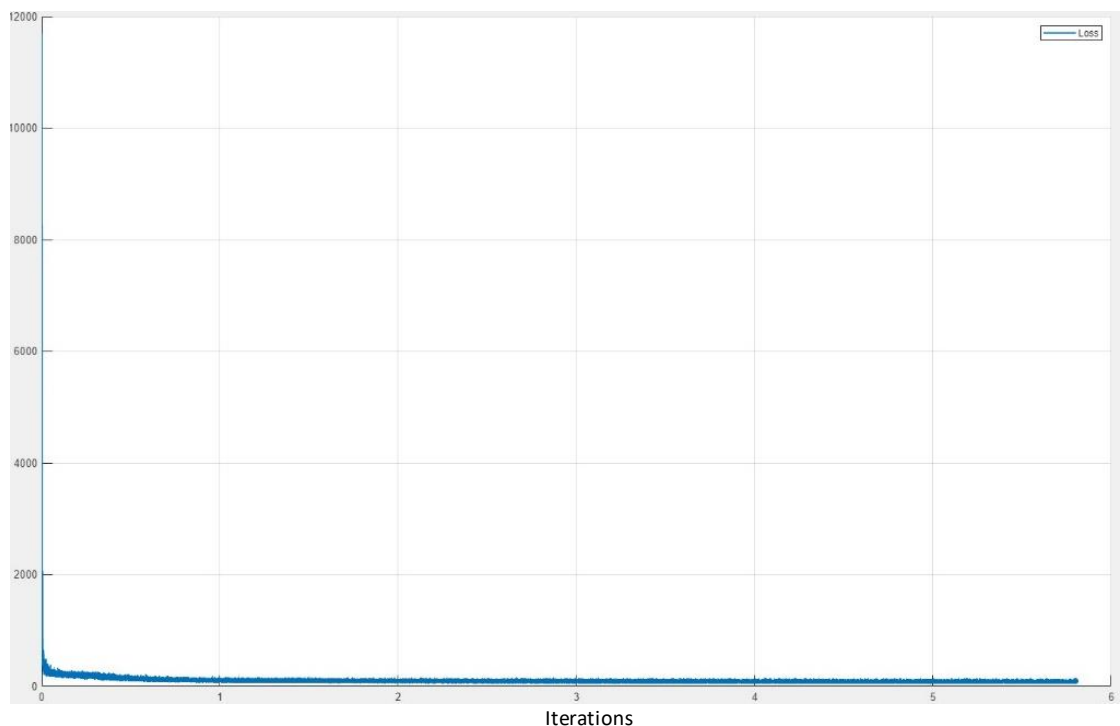


Figure 6.13 VAE training process

After the training, the whole experimental datasets are sent to VAE for testing, resulting corresponding reconstructed CWT images. ARE is calculated by comparing the original

sample with the reconstructed sample. Figure 6.14 is a scatter plot, showing ARE across all samples from seven scenarios. An overall increasing trend is seen, from S1 to S7, indicative of rising damage severity. However, between S4 and S5, this trend is not consistent, suggesting a complexity in damage progression that may require additional exploration. Further analysis of the VAE's performance for damage detection is provided in Figure 6.15, which shows a box plot of the Statistical ARE (SARE). The plot highlights the range of the errors, with scenarios of higher median Statistical ARE values indicative of more severe structural damage statistically. One possible reason for the SARE values of Scenarios 5 and 6 being smaller than Scenario 4 could be due to the non-linearity in how damage progression affects the vibration signals. As damage severity increases, the model might become less sensitive to specific types of structural changes, especially if the damage leads to more complex or distributed vibration patterns that the model cannot easily detect. This phenomenon could result in higher errors for less severe damages (like in Scenario 4) compared to more severe but more distributed damage (like in Scenarios 5 and 6). Figure 6.16 presents a histogram of the testing average reconstructed error (ARE) of the VAE, showing the distribution of error frequencies. The most prominent peak occurs in the error range just above  $5 \times 10^{-3}$ , where the highest frequency of samples, around 140, is found. This peak indicates the most common error value range that the VAE produced during testing. There are significantly fewer samples with very low errors (close to 0) or higher errors approaching  $15 \times 10^{-3}$ .

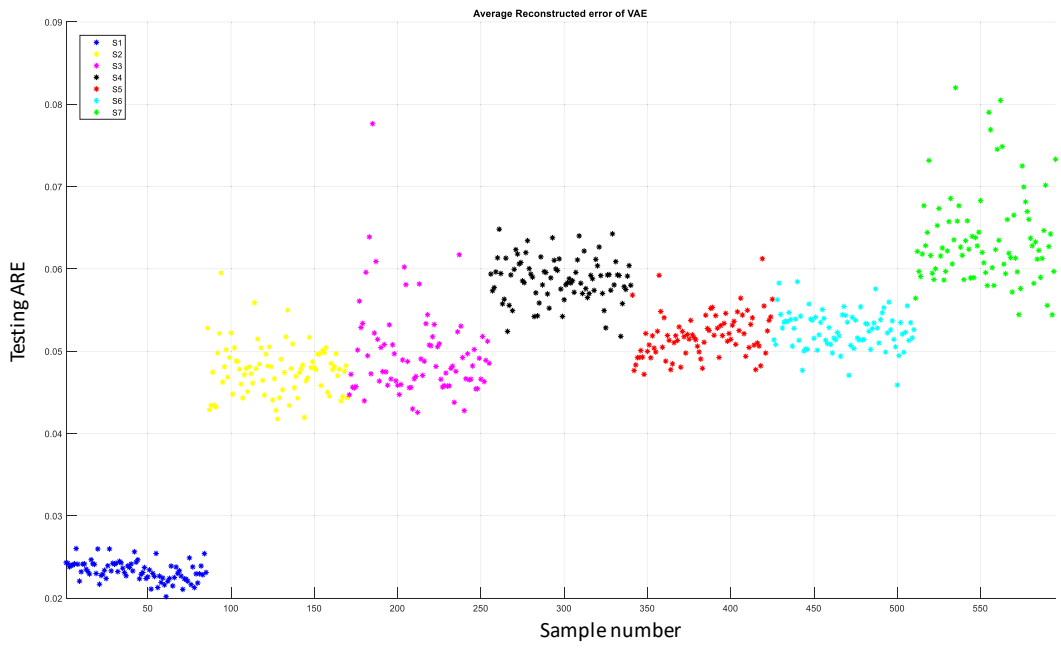


Figure 6.14 Damage quantification using Average reconstructed error(ARE)

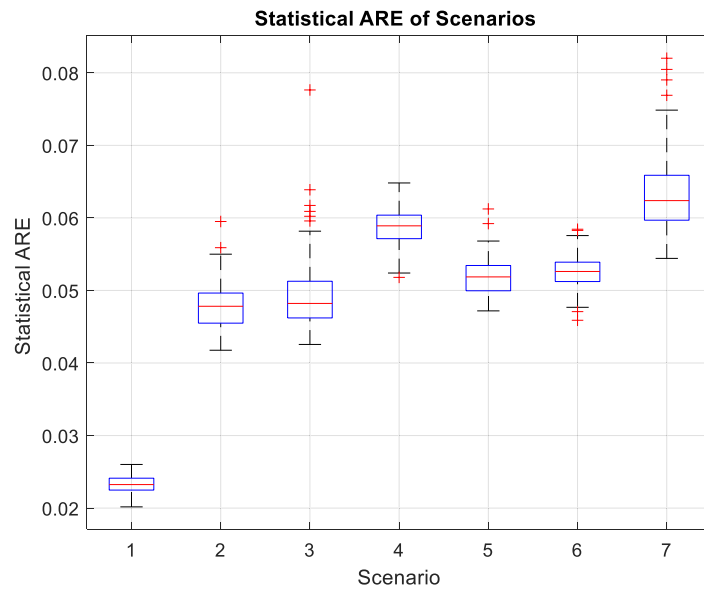


Figure 6.15 Box plot of SARE of VAE

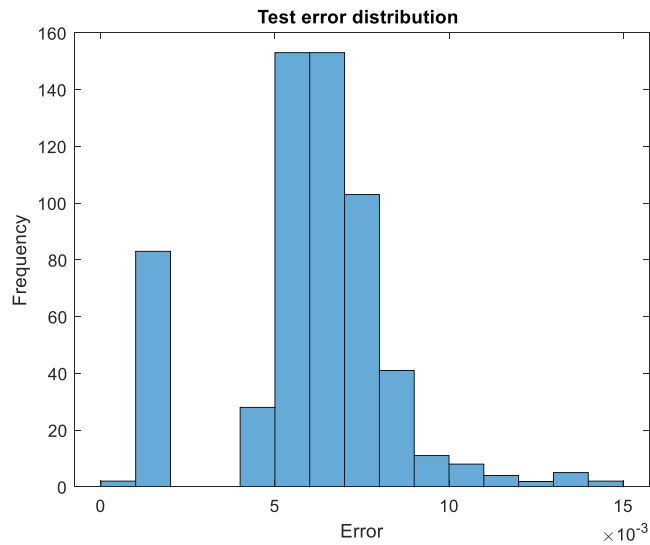


Figure 6.16 The testing error distribution of VAE

Figure 6.17 compares the selected original experimental CWT image and its corresponding reconstructed images by the VAE. This illustrates the model's ability to replicate the patterns observed in the experimental CWT data. Such precise reconstruction capability suggests that the VAE can effectively learn and imitate the underlying structural behaviour captured by the sensors.

Figure 6.18 shows the VAE's advanced generative function, where with the random input, it generates new CWT images based on the learned latent space. These generated images are predictions of how the CWT images of sensor data might appear under healthy conditions. The ability to create such data is crucial for enhancing the machine learning model's generalization capabilities, which is significant when lacking training data.

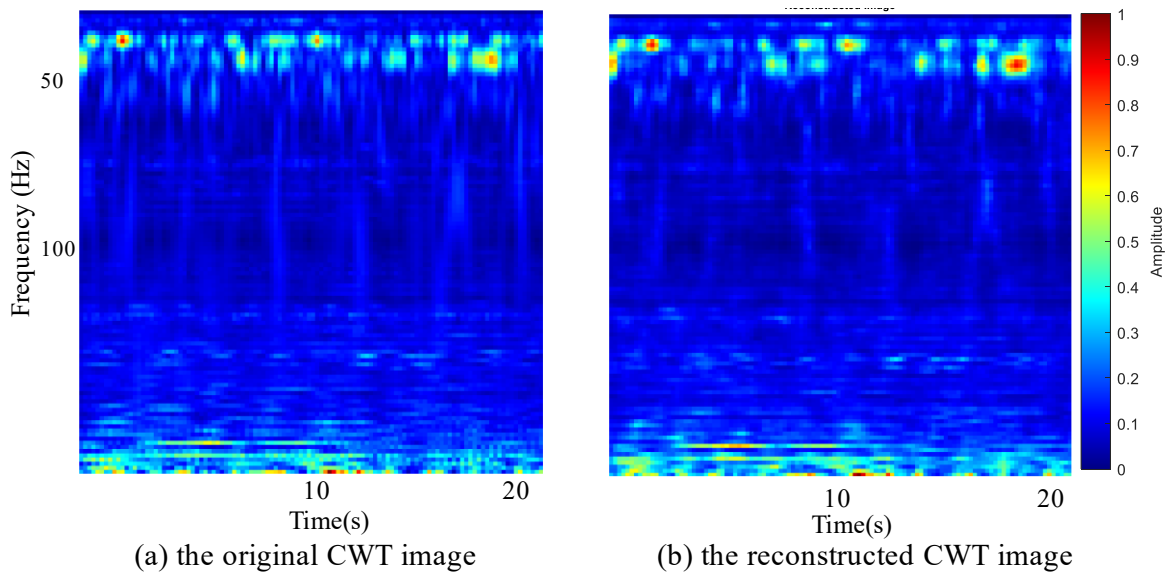


Figure 6.17 Comparison of the original experimental CWT image and reconstructed image

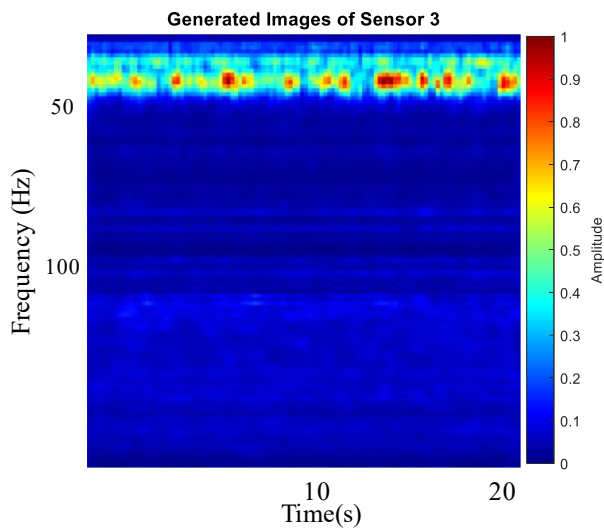


Figure 6.18 Generating experimental CWT images using VAE

## 6.5 SUMMARY

This section presents a comprehensive study of structural damage quantification using an unsupervised learning method with Continuous Wavelet Transform (CWT) based images as input. The effectiveness of the CWT using a Variational Autoencoder (VAE) was verified using numerical and experimental results, and the results show that it is a robust

framework for interpreting complex signal patterns and detecting structural damage. The results not only highlight the VAE's capability to identify damage signatures but also illustrate a clear methodology for employing this technology in the practical detection of structural damage.

# CHAPTER 7 CONCLUSIONS AND RECOMMENDATIONS

## 7.1 CONCLUSIONS

This thesis has dived deeply into the realm of damage detection for complex bridge structures, using machine learning techniques to bridge the gap between numerical simulations and practical applications. It comprehensively introduced and evaluated several machine learning methodologies - supervised, unsupervised, and semi-supervised - each contributing significantly towards enhancing processes for detecting structural damage. The following can be concluded,

- 1) The development of a novel generic element specifically designed for nonuniform beams with semi-rigid joints stands out as a main achievement. This innovation has not only improved the accuracy of numerical modelling but also enhanced the damage detection capabilities, as validated through an experimental study on an 8-meter truss bridge. This generic element's efficacy in localizing both beam and joint damages has been conclusively demonstrated, showing its potential for widespread application in Structural Health Monitoring (SHM).
- 2) The thesis introduced a specialised Convolutional Neural Network (CNN) loss function, customized for structural damage detection. This loss function significantly reduces computational costs while ensuring fast convergence and high accuracy, addressing a crucial gap in machine learning applications within

SHM.

- 3) A novel semi-supervised learning framework was proposed, using LSTM-VAE networks for damage quantification and SVM for classification. By introducing the Average Reconstructed Error (ARE) as a direct damage measurement index and combining t-SNE with ARE features, the study achieved a remarkable 71% accuracy in classifying damage levels in an experimental bridge model. This not only shows the effectiveness of the proposed semi-supervised learning framework but also highlights its practical viability for improving maintenance strategies for bridges.
- 4) The investigation also studies unsupervised learning, employing a Variational Autoencoder (VAE) enhanced with Continuous Wavelet Transform (CWT) based images inputs to quantify structural damage. This approach was rigorously validated through numerical simulations and practical applications, showing its capability to identify damage signatures effectively.

Overall, this thesis presents a comprehensive exploration of machine learning applications in damage detection for bridges, offering novel insights and methodologies that promise to significantly advance the field of structural health monitoring.

## **7.2 RECOMMENDATIONS FOR FUTURE WORK**

The recommendations provided here are intended to enhance their methodologies'



applicability for modern infrastructure monitoring. These suggestions are designed to guide future research and practice in SHM, ensuring continued progress and adaptation to new challenges.

- 1) **Data Enhancement and Model Generalization:** It's essential to enrich the dataset with a more diverse set of structural damage states, including subtle and progressive damages of more elements and combinations. This can help in improving the robustness and generalization of the models.
- 2) **Utilization of All Data Sets:** The current thesis utilizes one set of signals for damage detection, leaving three additional sets, previously used for spatial mode shape extraction, unexplored for damage information. Future work should tap into these rich data resources for a full damage detection analysis, promising substantial gains in SHM accuracy and insight.
- 3) **Advanced Element Modelling:** Further refinement and testing of the novel generic element for nonuniform beams with semi-rigid joints are recommended. This includes its application in different types of bridges and structural systems to validate its universality and effectiveness.
- 4) **Algorithm Optimization:** The developed CNN with the customized loss function for structural damage detection can be optimized further by exploring advanced deep learning techniques like transfer learning for improved performance.

- 5) Exploration of Hybrid GNN and CNN Models: Future research could explore the integration of Graph Neural Networks (GNNs) with Convolutional Neural Networks (CNNs) to improve relational data processing in spatial structures. GNNs can potentially capture the relationships between different structural elements (e.g., joints and beams) in a bridge, while CNNs are effective for analysing localized damage patterns in sensor data. Combining these two methods is expected to result in more comprehensive damage detection, allowing both localized and system-wide analysis of structural integrity.
- 6) Real-world Application and Validation: Implementing and validating the models in real-life bridge monitoring scenarios would be a crucial step. This can involve collaborations with industry partners for field-testing on operational bridges to ensure the practical applicability of the research findings.

Through these recommendations, future researchers and practitioners can continue to build based on this thesis's findings, progress the field of SHM, and enhancing the safety and longevity of bridge infrastructures. The proposed methods, especially the combination of data-driven approaches and deep learning algorithms, are particularly well-suited for complex structures with multiple joints and bar members, such as steel bridges. These structures often experience localized damage, which is challenging to detect using traditional methods. The model's ability to handle high-dimensional sensor data and capture subtle structural changes can also be extended to other infrastructure types with similar characteristics, such as trusses, towers, or offshore platforms. Moreover,

the robustness of these methods under varying environmental conditions, including fluctuating temperatures and humidity levels, could be explored to assess their applicability in a broader range of real-world settings.

## REFERENCES

- Abdeljaber, O., Avci, O., Kiranyaz, M. S., Boashash, B., Sodano, H., & Inman, D. J. (2018a). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing*, 275, 1308–1317. <https://doi.org/10.1016/j.neucom.2017.09.069>
- Abdeljaber, O., Avci, O., Kiranyaz, M. S., Boashash, B., Sodano, H., & Inman, D. J. (2018b). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing*, 275, 1308–1317. <https://doi.org/10.1016/j.neucom.2017.09.069>
- Abdeljaber, O., Avci, O., Kiranyaz, S., Gabbouj, M., & Inman, D. J. (2017). Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of Sound and Vibration*, 388, 154–170. <https://doi.org/10.1016/j.jsv.2016.10.043>
- Adams, D. E. (2007). *Health monitoring of structural materials and components: Methods with applications*. John Wiley & Sons.
- Ahmed, H., La, H. M., & Tran, K. (2020). Rebar detection and localization for bridge deck inspection and evaluation using deep residual networks. *Automation in Construction*, 120. <https://doi.org/10.1016/j.autcon.2020.103393>
- Amador, S. D., & Brincker, R. (2021). Robust multi-dataset identification with frequency domain decomposition. *Journal of Sound and Vibration*, 508, 116207-. <https://doi.org/10.1016/j.jsv.2021.116207>
- Anaissi, A., Zandavi, S. M., Suleiman, B., Naji, M., & Braytee, A. (2023). Multi-objective variational autoencoder: An application for smart infrastructure maintenance. *Applied Intelligence*, 53(10), 12047–12062. <https://doi.org/10.1007/s10489-022-04163-2>
- Aswal, N., Sen, S., & Mevel, L. (2021). Estimation of local failure in tensegrity using Interacting Particle-Ensemble Kalman Filter. *Mechanical Systems and Signal Processing*, 160. Scopus. <https://doi.org/10.1016/j.ymsp.2021.107824>
- Avci, O., Abdeljaber, O., Kiranyaz, S., Hussein, M., Gabbouj, M., & Inman, D. J. (2021). A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications. *Mechanical Systems and Signal Processing*, 147. Scopus. <https://doi.org/10.1016/j.ymsp.2020.107077>
- Bao, Y., Song, C., Wang, W., Ye, T., Wang, L., & Yu, L. (2013). Damage Detection of Bridge Structure Based on SVM. *Mathematical Problems in Engineering*, 2013, 1–7. <https://doi.org/10.1155/2013/490372>
- Bao, Y., Tang, Z., Li, H., & Zhang, Y. (2019). Computer vision and deep learning-based data anomaly detection method for structural health monitoring. *Structural Health Monitoring*, 18(2), 401–421. <https://doi.org/10.1177/1475921718757405>
- Bin Zhang, Shaohui Zhang, & Weihua Li. (2019). Bearing performance degradation assessment using long short-term memory recurrent network. *Computers in*

- Industry*, 106, 14–29. <https://doi.org/10.1016/j.compind.2018.12.016>
- Calderon Hurtado, A., Kaur, K., Makki Alamdari, M., Atroshchenko, E., Chang, K. C., & Kim, C. W. (2023). Unsupervised learning-based framework for indirect structural health monitoring using adversarial autoencoder. *Journal of Sound and Vibration*, 550, 117598-. <https://doi.org/10.1016/j.jsv.2023.117598>
- Cao, M., Radziński, M., Xu, W., & Ostachowicz, W. (2014). Identification of multiple damage in beams based on robust curvature mode shapes. *Mechanical Systems and Signal Processing*, 46(2), 468–480. <https://doi.org/10.1016/j.ymsp.2014.01.004>
- Capecchi, D., Ciambella, J., Pau, A., & Vestroni, F. (2016). Damage identification in a parabolic arch by means of natural frequencies, modal shapes and curvatures. *Meccanica*, 51(11), 2847–2859. <https://doi.org/10.1007/s11012-016-0510-3>
- Cha, Y.-J., Choi, W., & Büyüköztürk, O. (2017). Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, 32(5), 361–378. <https://doi.org/10.1111/mice.12263>
- Chen, W., Zhang, S., Li, R., & Shahabi, H. (2018). Performance evaluation of the GIS-based data mining techniques of best-first decision tree, random forest, and naïve Bayes tree for landslide susceptibility modeling. *Science of The Total Environment*, 644, 1006–1018. <https://doi.org/10.1016/j.scitotenv.2018.06.389>
- Chen, X., Ye, Y., Zhang, X., & Yu, C. (2020). *Bridge Damage Detection and Recognition Based on Deep Learning*. 1626(1). <https://doi.org/10.1088/1742-6596/1626/1/012151>
- Cichosz, P. (2015). *Naïve Bayes classifier* (pp. 118–133). John Wiley & Sons, Ltd. <https://doi.org/10.1002/9781118950951.ch4>
- Cui, F., Kim, M., Park, C., Kim, D., Mo, K., & Kim, M. (2021). Application of principal component analysis (PCA) to the assessment of parameter correlations in the partial-nitrification process using aerobic granular sludge. *Journal of Environmental Management*, 288, 112408–112408. <https://doi.org/10.1016/j.jenvman.2021.112408>
- dos Santos, F. L. M., Peeters, B., Van der Auweraer, H., Góes, L. C. S., & Desmet, W. (2016). Vibration-based damage detection for a composite helicopter main rotor blade. *Case Studies in Mechanical Systems and Signal Processing*, 3, 22–27. <https://doi.org/10.1016/j.csmssp.2016.01.001>
- Eckert-Gallup, A. C., Sallaberry, C. J., Dallman, A. R., & Neary, V. S. (2016). Application of principal component analysis (PCA) and improved joint probability distributions to the inverse first-order reliability method (I-FORM) for predicting extreme sea states. *Ocean Engineering*, 112(C), 307–319. <https://doi.org/10.1016/j.oceaneng.2015.12.018>
- Escalante, H. J., Morales, E. F., & Sucar, L. E. (2016). A naïve Bayes baseline for early gesture recognition. *Pattern Recognition Letters*, 73, 91–99. <https://doi.org/10.1016/j.patrec.2016.01.013>
- Fan, L., & Leng Poh, K. (1 C.E., January 1). *Improving the Naïve Bayes Classifier* [1]<http://www.igi.global.com/chapter/improving-naïve-bayes-classifier/10347>

- Fathnejat, H., Ahmadi-Nedushan, B., Hosseini-nejad, S., Noori, M., & Altabey, W. A. (2023). A data-driven structural damage identification approach using deep convolutional-attention-recurrent neural architecture under temperature variations. *Engineering Structures*, 276, 115311. <https://doi.org/10.1016/j.engstruct.2022.115311>
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2013). *Rich feature hierarchies for accurate object detection and semantic segmentation*. <https://arxiv.org/abs/1311.2524>
- Gong, Y. F., Bi, H. P., Sun, Y. Y., Jiao, Y. B., & Liu, H. B. (2011). Damage Identification for Simply-Supported Bridge Based on SVM Optimized by PSO (PSO-SVM). *Applied Mechanics and Materials*, 80–81, 490–494. <https://doi.org/10.4028/www.scientific.net/AMM.80-81.490>
- Gray, V. (2017). *Principal Component Analysis: Methods, Applications and Technology*. Nova Science Publishers, Incorporated. <http://ebookcentral.proquest.com/lib/uts/detail.action?docID=4801141>
- Guo, H. Y., & Li, Z. L. (2014). Structural Multi-Damage Identification Based on Modal Strain Energy Equivalence Index Method. *International Journal of Structural Stability and Dynamics*, 14(07), 1450028. <https://doi.org/10.1142/S021945541450028X>
- He, H., Zheng, J., Liao, L., & Chen, Y. (2020). Damage identification based on convolutional neural network and recurrence graph for beam bridge. *Structural Health Monitoring*, 147592172091692-. <https://doi.org/10.1177/1475921720916928>
- Hou, R., Beck, J. L., Zhou, X., & Xia, Y. (2021). Structural damage detection of space frame structures with semi-rigid connections. *Engineering Structures*, 235, 112029. <https://doi.org/10.1016/j.engstruct.2021.112029>
- Hou, S., Dong, B., Wang, H., & Wu, G. (2020). Inspection of surface defects on stay cables using a robot and transfer learning. *Automation in Construction*, 119, 103382-. <https://doi.org/10.1016/j.autcon.2020.103382>
- Hou, & Xia. (2021). Review on the new development of vibration-based damage identification for civil engineering structures: 2010–2019. *Journal of Sound and Vibration*, 491, 115741. <https://doi.org/10.1016/j.jsv.2020.115741>
- Hu, S.-L. J., Li, H., & Wang, S. (2007). Cross-model cross-mode method for model updating. *Mechanical Systems and Signal Processing*, 21(4), 1690–1703. <https://doi.org/10.1016/j.ymsp.2006.07.012>
- Huang, M., Lei, Y., Li, X., & Gu, J. (2021). Damage Identification of Bridge Structures Considering Temperature Variations-Based SVM and MFO. *Journal of Aerospace Engineering*, 34(2). [https://doi.org/10.1061/\(ASCE\)AS.1943-5525.0001225](https://doi.org/10.1061/(ASCE)AS.1943-5525.0001225)
- Huynh, T.-C., Park, J.-H., Jung, H.-J., & Kim, J.-T. (2019). Quasi-autonomous bolt-loosening detection method using vision-based deep learning and image processing. *Automation in Construction*, 105, 102844. <https://doi.org/10.1016/j.autcon.2019.102844>
- Ihaddoudène, A. N. T., Saidani, M., & Chemrouk, M. (2009). Mechanical model for the

- analysis of steel frames with semi rigid joints. *Journal of Constructional Steel Research*, 65(3), 631–640. <https://doi.org/10.1016/j.jcsr.2008.08.010>
- Islam, M. M. M., & Kim, J.-M. (2019). Vision-Based Autonomous Crack Detection of Concrete Structures Using a Fully Convolutional Encoder–Decoder Network. *Sensors*, 19(19), 4251. <https://doi.org/10.3390/s19194251>
- İzgi, E., Öztopal, A., Yerli, B., Kaymak, M. K., & Şahin, A. D. (2012). Short–mid-term solar power prediction by using artificial neural networks. *Solar Energy*, 86(2), 725–733. <https://doi.org/10.1016/j.solener.2011.11.013>
- Jezequel, L. (1983). Structural Damping by Slip in Joints. *Journal of Vibration and Acoustics*, 105(4), 497–504. <https://doi.org/10.1115/1.3269134>
- Jia, H., Lin, J., & Liu, J. (2020). Bridge Seismic Damage Assessment Model Applying Artificial Neural Networks and the Random Forest Algorithm. *Advances in Civil Engineering*, 2020, 1–13. <https://doi.org/10.1155/2020/6548682>
- Jolliffe, I. T. (2002). *Principal component analysis* (2nd ed.). Springer.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. (2016). *Deep Residual Learning for Image Recognition*. 770–778. <https://doi.org/10.1109/CVPR.2016.90>
- Khosravan, A., Asgarian, B., & Shokrgozar, H. R. (2021). Improved Modal Strain Energy Decomposition Method for damage detection of offshore platforms using data of sensors above the water level. *Ocean Engineering*, 219. <https://doi.org/10.1016/j.oceaneng.2020.108337>
- Kim, C.-W., Zhang, Y., Wang, Z., Oshima, Y., & Morita, T. (2018). Long-term bridge health monitoring and performance assessment based on a Bayesian approach. *Structure and Infrastructure Engineering*, 14(7), 883–894. <https://doi.org/10.1080/15732479.2018.1436572>
- Kim, H., & Bartkiewicz, T. (1993). *Damage Detection And Health Monitoring Of Large Space Structures*.
- Kim, H., & Melhem, H. (2004). Damage detection of structures by wavelet analysis. *Engineering Structures*, 26(3), 347–362. <https://doi.org/10.1016/j.engstruct.2003.10.008>
- Kim, J.-T., Ryu, Y.-S., Cho, H.-M., & Stubbs, N. (2003). Damage identification in beam-type structures: Frequency-based method vs mode-shape-based method. *Engineering Structures*, 25(1), 57–67. [https://doi.org/10.1016/S0141-0296\(02\)00118-9](https://doi.org/10.1016/S0141-0296(02)00118-9)
- Kim, Y.-S., & Eun, H.-C. (2017). Comparison of Damage Detection Methods Depending on FRFs within Specified Frequency Ranges. *Advances in Materials Science and Engineering*, 2017, 1–9. <https://doi.org/10.1155/2017/5821835>
- Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907. <https://doi.org/10.48550/arXiv.1609.02907>
- Kordestani, H., Xiang, Y.-Q., & Ye, X.-W. (2018). Output-Only Damage Detection of Steel Beam Using Moving Average Filter. *Shock and Vibration*, 2018, 1–13. <https://doi.org/10.1155/2018/2067680>

- Kordestani, H., Zhang, C., Masri, S. F., & Shadabfar, M. (2021). An empirical time-domain trend line-based bridge signal decomposing algorithm using Savitzky–Golay filter. *Structural Control and Health Monitoring*, 28(7), n/a. <https://doi.org/10.1002/stc.2750>
- Law, S. S., Chan, T. H. T., & Wu, D. (2001). SUPER-ELEMENT WITH SEMI-RIGID JOINTS IN MODEL UPDATING. *Journal of Sound and Vibration*, 239(1), 19–39. <https://doi.org/10.1006/jsvi.2000.3110>
- Lee, S., Park, S., Kim, T., Lieu, Q. X., & Lee, J. (2021). Damage quantification in truss structures by limited sensor-based surrogate model. *Applied Acoustics*, 172, 107547. <https://doi.org/10.1016/j.apacoust.2020.107547>
- Lee, Y.-J., & Cho, S. (2016). SHM-Based Probabilistic Fatigue Life Prediction for Bridges Based on FE Model Updating. *Sensors (Basel, Switzerland)*, 16(3), 317–317. <https://doi.org/10.3390/s16030317>
- Lei, X., Sun, L., Xia, Y., & He, T. (2020). Vibration-Based Seismic Damage States Evaluation for Regional Concrete Beam Bridges Using Random Forest Method. *Sustainability (Basel, Switzerland)*, 12(12), 5106-. <https://doi.org/10.3390/su12125106>
- Li, G., Liu, Q., Zhao, S., Qiao, W., & Ren, X. (2020). Automatic crack recognition for concrete bridges using a fully convolutional neural network and naive Bayes data fusion based on a visual detection system. *Measurement Science & Technology*, 31(7), 75403-. <https://doi.org/10.1088/1361-6501/ab79c8>
- Li, S., Zuo, X., Li, Z., & Wang, H. (2020). Applying Deep Learning to Continuous Bridge Deflection Detected by Fiber Optic Gyroscope for Damage Detection. *Sensors (Basel, Switzerland)*, 20(3), 911-. <https://doi.org/10.3390/s20030911>
- Li, Y., Bao, T., Gao, Z., Shu, X., Zhang, K., Xie, L., & Zhang, Z. (2022). A new dam structural response estimation paradigm powered by deep learning and transfer learning techniques. *Structural Health Monitoring*, 21(3), 770–787. Scopus. <https://doi.org/10.1177/14759217211009780>
- Lin, Y., & Nie, Z. (n.d.). *Structural Damage Detection with Automatic Feature-Extraction through Deep Learning*. 22.
- Lu, Q., Zhu, J., & Zhang, W. (2020). Quantification of Fatigue Damage for Structural Details in Slender Coastal Bridges Using Machine Learning-Based Methods. *Journal of Bridge Engineering*, 25(7), 4020033-. [https://doi.org/10.1061/\(ASCE\)BE.1943-5592.0001571](https://doi.org/10.1061/(ASCE)BE.1943-5592.0001571)
- Ma, Fan, Wen, P., Zhang, H., & Shen, S. (2015). Experimental and numerical studies on a single-layer cylindrical reticulated shell with semi-rigid joints. *Thin-Walled Structures*, 86, 1–9. <https://doi.org/10.1016/j.tws.2014.08.006>
- Ma, Lin, Nie, & Ma. (2020). Structural damage identification based on unsupervised feature-extraction via Variational Auto-encoder. *Measurement*, 160, 107811. <https://doi.org/10.1016/j.measurement.2020.107811>
- Maaten, L. van der, & Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86), 2579–2605.
- Malekjafarian, A., Golpayegani, F., Moloney, C., & Clarke, S. (2019). A Machine



- Learning Approach to Bridge-Damage Detection Using Responses Measured on a Passing Vehicle. *Sensors (Basel, Switzerland)*, 19(18), 4035-. <https://doi.org/10.3390/s19184035>
- Mao, J., Wang, H., & Spencer, B. F. (2021). Toward data anomaly detection for automated structural health monitoring: Exploiting generative adversarial nets and autoencoders. *Structural Health Monitoring*, 20(4), 1609–1626. <https://doi.org/10.1177/1475921720924601>
- Mei, Q., Gül, M., & Boay, M. (2019). Indirect health monitoring of bridges using Mel-frequency cepstral coefficients and principal component analysis. *Mechanical Systems and Signal Processing*, 119, 523–546. <https://doi.org/10.1016/j.ymsp.2018.10.006>
- Moein, S. (2014). *Medical diagnosis using artificial neural networks*. Medical Information Science Reference.
- Mohanraj, M., Jayaraj, S., & Muraleedharan, C. (2012). Applications of artificial neural networks for refrigeration, air-conditioning and heat pump systems—A review. *Renewable & Sustainable Energy Reviews*, 16(2), 1340–1358. <https://doi.org/10.1016/j.rser.2011.10.015>
- Monforton, G. R., & Wu, T. S. (1963). Matrix Analysis of Semi-Rigidly Connected Frames. *Journal of the Structural Division*, 89(6), 13–42. <https://doi.org/10.1061/JSDEAG.0000997>
- Mousavi, M., & Gandomi, A. H. (2021). Structural health monitoring under environmental and operational variations using MCD prediction error. *Journal of Sound and Vibration*, 512, 116370. <https://doi.org/10.1016/j.jsv.2021.116370>
- Neves, A. C., González, I., Leander, J., & Karoumi, R. (2017). Structural health monitoring of bridges: A model-free ANN-based approach to damage detection. *Journal of Civil Structural Health Monitoring*, 7(5), 689–702. <https://doi.org/10.1007/s13349-017-0252-5>
- Nguyen, D. H., Nguyen, Q. B., Bui-Tien, T., De Roeck, G., & Abdel Wahab, M. (2020). Damage detection in girder bridges using modal curvatures gapped smoothing method and Convolutional Neural Network: Application to Bo Nghi bridge. *Theoretical and Applied Fracture Mechanics*, 109, 102728-. <https://doi.org/10.1016/j.tafmec.2020.102728>
- Ni, F., Zhang, J., & Noori, M. N. (2020). Deep learning for data anomaly detection and data compression of a long-span suspension bridge. *Computer-Aided Civil and Infrastructure Engineering*, 35(7), 685–700. <https://doi.org/10.1111/mice.12528>
- Nick, H., & Aziminejad, A. (2021). Vibration-Based Damage Identification in Steel Girder Bridges Using Artificial Neural Network Under Noisy Conditions. *Journal of Nondestructive Evaluation*, 40(1). Scopus. <https://doi.org/10.1007/s10921-020-00744-8>
- Olatomiwa, L., Mekhilef, S., Shamshirband, S., Mohammadi, K., Petković, D., & Sudheer, C. (2015). A support vector machine–firefly algorithm-based model for global solar radiation prediction. *Solar Energy*, 115, 632–644. <https://doi.org/10.1016/j.solener.2015.03.015>

- Paral, A., Singha Roy, D. Kr., & Samanta, A. K. (2021). A deep learning-based approach for condition assessment of semi-rigid joint of steel frame. *Journal of Building Engineering*, *34*, 101946. <https://doi.org/10.1016/j.jobe.2020.101946>
- Pathirage, C. S. N., Li, J., Li, L., Hao, H., Liu, W., & Ni, P. (2018). Structural damage identification based on autoencoder neural networks and deep learning. *Engineering Structures*, *172*, 13–28. <https://doi.org/10.1016/j.engstruct.2018.05.109>
- Paz, M., & Kim, Y. H. (2019). *Structural Dynamics: Theory and Computation*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-94743-3>
- Peeters, B., & De Roeck, G. (2000). Reference based stochastic subspace identification in Civil Engineering. *Inverse Problems in Engineering*, *8*, 47–74. <https://doi.org/10.1080/174159700088027718>
- Pollastro, A., Testa, G., Bilotta, A., & Prevete, R. (2023). Semi-Supervised Detection of Structural Damage Using Variational Autoencoder and a One-Class Support Vector Machine. *IEEE Access*, *11*, 67098–67112. <https://doi.org/10.1109/ACCESS.2023.3291674>
- Pu, Q., Hong, Y., Chen, L., Yang, S., & Xu, X. (2019). Model updating-based damage detection of a concrete beam utilizing experimental damped frequency response functions. *Advances in Structural Engineering*, *22*(4), 935–947. <https://doi.org/10.1177/1369433218789556>
- Pun, G. P. P., Batra, R., Ramprasad, R., & Mishin, Y. (2019). Physically informed artificial neural networks for atomistic modeling of materials. *Nature Communications*, *10*(1), 2339–2339. <https://doi.org/10.1038/s41467-019-10343-5>
- Qin, J., & He, Z.-S. (2005). A SVM face recognition method based on Gabor-featured key points. *2005 International Conference on Machine Learning and Cybernetics*, *8*, 5144-5149 Vol. 8. <https://doi.org/10.1109/ICMLC.2005.1527850>
- Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning*, *1*(1), 81–106. <https://doi.org/10.1023/A:1022643204877>
- Seyedpoor, S. M. (2012). A two stage method for structural damage detection using a modal strain energy based index and particle swarm optimization. *International Journal of Non-Linear Mechanics*, *47*(1), 1–8. <https://doi.org/10.1016/j.ijnonlinmec.2011.07.011>
- Shamshirband, S., Mohammadi, K., Khorasanizadeh, H., Yee, P. L., Lee, M., Petković, D., & Zalnezhad, E. (2016). Estimating the diffuse solar radiation using a coupled support vector machine-wavelet transform model. *Renewable & Sustainable Energy Reviews*, *56*, 428–435. <https://doi.org/10.1016/j.rser.2015.11.055>
- Sharma, S., & Sen, S. (2021). Bridge Damage Detection in Presence of Varying Temperature Using Two-Step Neural Network Approach. *Journal of Bridge Engineering*, *26*(6). Scopus. [https://doi.org/10.1061/\(ASCE\)BE.1943-5592.0001708](https://doi.org/10.1061/(ASCE)BE.1943-5592.0001708)
- Shi, Z. Y., Law, S. S., & Zhang, L. M. (1998). STRUCTURAL DAMAGE LOCALIZATION FROM MODAL STRAIN ENERGY CHANGE. *Journal of Sound and Vibration*, *218*(5), 825–844. <https://doi.org/10.1006/jsvi.1998.1878>

- Shi, Z. Y., Law, S. S., & Zhang, L. M. (2000). Structural Damage Detection from Modal Strain Energy Change. *Journal of Engineering Mechanics*, 126(12), 1216–1223. [https://doi.org/10.1061/\(ASCE\)0733-9399\(2000\)126:12\(1216\)](https://doi.org/10.1061/(ASCE)0733-9399(2000)126:12(1216))
- Shu, J., Zhang, Z., Gonzalez, I., & Karoumi, R. (2013). The application of a damage detection method using Artificial Neural Network and train-induced vibrations on a simplified railway bridge model. *Engineering Structures*, 52(52), 408–421. <https://doi.org/10.1016/j.engstruct.2013.02.031>
- Sony, S., Dunphy, K., Sadhu, A., & Capretz, M. (2021). A systematic review of convolutional neural network-based structural condition assessment techniques. *Engineering Structures*, 226, 111347. <https://doi.org/10.1016/j.engstruct.2020.111347>
- Soualhi, A., Medjaher, K., & Zerhouni, N. (2015). Bearing Health Monitoring Based on Hilbert-Huang Transform, Support Vector Machine, and Regression. *IEEE Transactions on Instrumentation and Measurement*, 64(1), 52–62. <https://doi.org/10.1109/TIM.2014.2330494>
- Sreedhara, B. M., Rao, M., & Mandal, S. (2019). Application of an evolutionary technique (PSO–SVM) and ANFIS in clear-water scour depth prediction around bridge piers. *Neural Computing & Applications*, 31(11), 7335–7349. <https://doi.org/10.1007/s00521-018-3570-6>
- Sun, L., Shang, Z., Xia, Y., Bhowmick, S., & Nagarajaiah, S. (2020a). Review of Bridge Structural Health Monitoring Aided by Big Data and Artificial Intelligence: From Condition Assessment to Damage Detection. *Journal of Structural Engineering*, 146(5), 04020073. [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0002535](https://doi.org/10.1061/(ASCE)ST.1943-541X.0002535)
- Sun, L., Shang, Z., Xia, Y., Bhowmick, S., & Nagarajaiah, S. (2020b). Review of Bridge Structural Health Monitoring Aided by Big Data and Artificial Intelligence: From Condition Assessment to Damage Detection. *Journal of Structural Engineering*, 146(5), 04020073. [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0002535](https://doi.org/10.1061/(ASCE)ST.1943-541X.0002535)
- Sun, S., Liang, L., Li, M., & Li, X. (2018). Vibration-based Damage Detection in Bridges via Machine Learning. *KSCE Journal of Civil Engineering*, 22(12), 5123–5132. <https://doi.org/10.1007/s12205-018-0318-x>
- Talebpour, M. H., Goudarzi, Y., & Sharifnezhad, M. (2020). Clustering Elements of Truss Structures for Damage Identification by CBO. *Periodica Polytechnica Civil Engineering*. <https://doi.org/10.3311/PPci.16636>
- Teng, S., Chen, G., Gong, P., Liu, G., & Cui, F. (2020a). Structural damage detection using convolutional neural networks combining strain energy and dynamic response. *Meccanica*, 55(4), 945–959. <https://doi.org/10.1007/s11012-019-01052-w>
- Teng, S., Chen, G., Gong, P., Liu, G., & Cui, F. (2020b). Structural damage detection using convolutional neural networks combining strain energy and dynamic response. *Meccanica*, 55(4), 945–959. <https://doi.org/10.1007/s11012-019-01052-w>
- Terbuch, A., O’Leary, P., & Auer, P. (2022). Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement Data. *2022 IEEE International*

- Instrumentation and Measurement Technology Conference (I2MTC)*, 1–6. <https://doi.org/10.1109/I2MTC48687.2022.9806663>
- Umar, S., Bakhary, N., & Abidin, A. R. Z. (2018). Response surface methodology for damage detection using frequency and mode shape. *Measurement: Journal of the International Measurement Confederation*, 115, 258–268. <https://doi.org/10.1016/j.measurement.2017.10.047>
- Variational autoencoder. (2024). In *Wikipedia*. [https://en.wikipedia.org/w/index.php?title=Variational\\_autoencoder&oldid=1209092051](https://en.wikipedia.org/w/index.php?title=Variational_autoencoder&oldid=1209092051)
- Wang, R., Chenchao, An, S., Li, J., Li, L., Hao, H., & Liu, W. (2020). Deep residual network framework for structural health monitoring. *Structural Health Monitoring*, 147592172091837. <https://doi.org/10.1177/1475921720918378>
- Wang, S., & Xu, M. (2019). Modal Strain Energy-based Structural Damage Identification: A Review and Comparative Study. *Structural Engineering International*, 29(2), 234–248. <https://doi.org/10.1080/10168664.2018.1507607>
- Wang, X., Gao, Q., & Liu, Y. (2020). Damage Detection of Bridges under Environmental Temperature Changes Using a Hybrid Method. *Sensors (Basel, Switzerland)*, 20(14), 3999-. <https://doi.org/10.3390/s20143999>
- Wang, Y. Y., & Li, J. (2008). Feature-selection ability of the decision-tree algorithm and the impact of feature-selection/extraction on decision-tree results based on hyperspectral data. *International Journal of Remote Sensing*, 29(10), 2993–3010. <https://doi.org/10.1080/01431160701442070>
- Wang, Z., Lin, R. M., & Lim, M. K. (1997). Structural damage detection using measured FRF data. *Computer Methods in Applied Mechanics and Engineering*, 147(1–2), 187–197. [https://doi.org/10.1016/S0045-7825\(97\)00013-3](https://doi.org/10.1016/S0045-7825(97)00013-3)
- Xu, J., Hao, J., Li, H., Luo, M., Guo, W., & Li, W. (2017). Experimental Damage Identification of a Model Reticulated Shell. *Applied Sciences*, 7(4), 362. <https://doi.org/10.3390/app7040362>
- Xu, Pan, Wang, Y., Deng, D., & Han, Q. (2022). Damage identification of single-layer cylindrical latticed shells based on the model updating technique. *Journal of Civil Structural Health Monitoring*, 12(2), 289–303. <https://doi.org/10.1007/s13349-021-00543-3>
- Xu, Y.-L., Zhang, C.-D., Zhan, S., & Spencer, B. F. (2018). Multi-level damage identification of a bridge structure: A combined numerical and experimental investigation. *Engineering Structures*, 156, 53–67. <https://doi.org/10.1016/j.engstruct.2017.11.014>
- Yan, B., & Miyamoto, A. (2006). A Comparative Study of Modal Parameter Identification Based on Wavelet and Hilbert–Huang Transforms. *Computer-Aided Civil and Infrastructure Engineering*, 21(1), 9–23. <https://doi.org/10.1111/j.1467-8667.2005.00413.x>
- Yanfei Sun. (2009). *Combined Neural Network and PCA for Complicated Damage Detection of Bridge*. 2, 524–528. <https://doi.org/10.1109/ICNC.2009.580>
- Yang, J., Zhang, L., Chen, C., Li, Y., Li, R., Wang, G., Jiang, S., & Zeng, Z. (2020). A

- hierarchical deep convolutional neural network and gated recurrent unit framework for structural damage detection. *Information Sciences*, 540, 117–130. <https://doi.org/10.1016/j.ins.2020.05.090>
- Yoon, M. K., Heider, D., Gillespie Jr, J. W., Ratcliffe, C. P., & Crane, R. M. (2010). Local Damage Detection with the Global Fitting Method Using Operating Deflection Shape Data. *Journal of Nondestructive Evaluation*, 29(1), 25–37. <https://doi.org/10.1007/s10921-010-0062-8>
- Zhang, C., Chang, C., & Jamshidi, M. (2020). Concrete bridge surface damage detection using a single-stage detector. *Computer-Aided Civil and Infrastructure Engineering*, 35(4), 389–409. <https://doi.org/10.1111/mice.12500>
- Zhang, T., Biswal, S., & Wang, Y. (2020). SHMnet: Condition assessment of bolted connection with beyond human-level performance. *Structural Health Monitoring*, 19(4), 1188–1201. <https://doi.org/10.1177/1475921719881237>
- Zhang, Y., Ma, J., & Wu, X. (2022). Locating damages of space trusses by combining cross-model modal strain energy and wavelet transform. *Structures*, 44, 120–134. <https://doi.org/10.1016/j.istruc.2022.07.081>
- Zhang, Y., Sun, X., Loh, K. J., Su, W., Xue, Z., & Zhao, X. (2020). Autonomous bolt loosening detection using deep learning. *Structural Health Monitoring*, 19(1), 105–122. <https://doi.org/10.1177/1475921719837509>
- Zhou, Z., Wegner, L. D., & Sparling, B. F. (2021). Data quality indicators for vibration-based damage detection and localization. *Engineering Structures*, 230. Scopus. <https://doi.org/10.1016/j.engstruct.2020.111703>