UNIVERSITY OF TECHNOLOGY SYDNEY

Physics-guided Machine Learning for Condition Assessment of Building Structures in Operational Environments

by Xutong Zhang

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

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

University of Technology Sydney
Faculty of Faculty of Engineering and Information Technology

March 2025

Certificate of original authorship

I, Xutong Zhang declare that this thesis is submitted in fulfilment 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 own 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.

Signature: Xutong Zhang

Production Note:
Signature removed prior
to publication.

Data: 4th March 2025

Acknowledgement

For the past four years, I have had the opportunity to pursue a postgraduate program at the Centre for Built Infrastructure Resilience (CBIR) at the University of Technology Sydney (UTS), for which I am extremely grateful. UTS has provided me with an excellent, competitive and supportive learning environment that has been essential to my development. I would like to express my sincere gratitude to my principal PhD supervisor A/Prof. Xinqun Zhu and my co-supervisor Prof. Li Jianchun. Their guidance, assistance and encouragement throughout my PhD program have been invaluable to me. Their technical expertise, profound knowledge, amiable character and patience have greatly contributed to my academic and personal growth.

I would like to thank Mr. Yingqi Wang for his generous help with the computational coding part of my research, as well as for his encouragement and companionship regarding my entire student career and life.

I would also like to thank my good friends and my co-supervisors in my research study group, who helped me a lot in various ways and brought me a great deal of fun during my time in the study. It has been a pleasure to meet and become friends with all of you. Finally, I would like to thank my family for their constant support. I would not be able to be who I am now without you; it is your encouragement and support that have helped me to overcome all the hardships I have faced during my studies.

List of publications

Journals:

Zhang, X., Zhu, X. and Li, J. 2023 'Transfer learning-based structural damage detection using frequency response functions with limited measurements', *International Journal of Structural Stability and Dynamics*, *https://doi.org/10.1142/S0219455425500932*.

Zhang, X., Zhu, X., Wang, Y. and Li, J. 2024 'Structural Damage Detection based on Transmissibility Functions with Unsupervised Domain Adaptation', *Engineering Structure. https://doi.org/10.1016/j.engstruct.2024.119142*

Zhang, X., Wang, Y., Zhu, X., and Li, J. 2024 'Parallel Neural Ordinary Differential Equations based Damage Identification for Building Structure', *Structural Health Monitoring*, Under review.

Zhang, X., Zhu, X., Wang, Y. 2024, 'Spatio-temporal-based physics neural operators for response reconstruction and structural identification with limited sensors', manuscript in drafting.

Conferences:

Zhang, X., Zhu, X., and Li, J. 2022, 'Knowledge Transfer for Structural Damage Detection using Fine- tunning based on FRF', *International Conference on Structural Engineering Research*, 22-26 Nov 2022, Sydney, Australia.

Zhang, X., Zhu, X., and Li, J. 2023, 'Unsupervised Knowledge Transfer for Structural Damage Detection with Limited Data', *SHMII-12: 12th International Conference on Structural Health Monitoring of Intelligent Infrastructure*. 18th Oct, 2023, Hangzhou, Zhejiang, China.

Zhang, X., Zhu, X. and Li, J. 2024 'Neural Ordinary Differential Equations (NODEs) Based Damage Qualification for Building Structures', *The 1st International Conference on Engineering*

Structures.

Awards:

- 2024 UTS Faculty of Engineering and IT - FEIT HDR Excellence Awards: ‘Distinction’.
- 2024 UTS FEIT School of Civil and Environmental Engineering - Best Candidate Assessment 3(CA3) Award: ‘1st Ranking’.
- The 1st International Conference on Engineering Structures 2024, Guangzhou, China - Best Student Paper Award.
- 2023 UTS Faculty of Engineering and IT - Women in Engineering and IT Award: ‘Distinction’.

Table of Contents

Certificate of original authorship……..i

Acknowledgement……...ii

List of publications…….....iii

Table of Contents……v

List of figures…….....ix

List of tables…….....xv

Abstract……....xviii

Chapter 1. Introduction……1

1.1. Background..... 1

1.2. Research objectives and significance..... 2

1.3. Organisation of thesis 4

Chapter 2. Literature review……...7

2.1. Concepts of structural health monitoring..... 7

2.2. Vibration-based structural damage detection 9

2.2.1. Model-based methods 9

2.2.2. Data-driven Methods 10

2.3. Machine learning for structural damage detection..... 11

2.3.1. Supervised learning for structural damage detection 11

2.3.2. Unsupervised learning for structural damage detection..... 12

2.3.3. Self-supervised learning for structural damage detection 12

2.3.4. Challenges in machine learning for structural damage detection..... 13

2.4. Transfer learning based structural damage detection..... 14

2.4.1. Transfer learning (TL) 14

2.4.2. Domain adaptation (DA)..... 15

2.5. Physics informed machine learning for structural condition assessment 15

2.5.1. Physics informed neural networks (PINNs)..... 15

2.5.2. Neural ordinary differentiation equations (NODEs)..... 16

2.5.3. Physics informed Neural Operators (PINOs)..... 16

2.6. Research gaps and challenges 17

Chapter 3. Transfer learning-based structural damage identification for building structures with limited measurement data……..19

3.1. Overview 19

3.2. State of the art 19

3.3. Theoretical background 22

3.3.1. The convolutional neural networks (CNN) 22

3.4. Methodology 24

3.4.1. Problems Definitions 24

3.4.2. Fine-tuning techniques for TL 25

3.5. Data collection 26

3.5.1. Experimental tests 26

3.5.2. Numerical modelling 27

3.6. Results and discussions 29

3.6.1. CNN training 29

3.6.2. Fine-tuning based on CNN 32

3.6.3. Knowledge transfer between two models with uncertainty 34

3.6.4. Knowledge transfer from numerical to experimental models 43

3.7. Summary 52

Chapter 4. Structural damage detection based on transmissibility functions with unsupervised domain adaptation 53

4.1. Overview 53

4.2. State of the art 54

4.3. Methodology 58

4.3.1. Problem definition 58

4.3.2. The proposed method 59

4.4. Knowledge transfer between two building structures with different storeys 70

4.4.1. Numerical models 70

4.4.2. Single damage detection 73

4.5. Knowledge transfer from numerical to experimental structures 81

4.5.1. Experimental setup 81

4.5.2. Damage Detection with Imbalance Class datasets 82

4.5.3. Damage prediction with different operational excitations 85

4.6. Knowledge transfer from numerical to real structures for the Canton Tower 86

4.6.1. Real structure and its numerical model 87

4.6.2. Damage prediction with imbalanced dataset 90

4.6.3. Discussions 94

4.7. Summary 95

Chapter 5. Parallel neural ordinary differential equations based damage identification for building structures 98

5.1. Overview 98

5.2. State of the art 98

5.3. Methodology 102

5.3.1. NODEs Theory 102

5.3.2. Structural Dynamic System 102

5.3.3. Neural Ordinary Differential Equations (NODEs) 103

5.3.4. PNODEs for Structural Damage Identification 105

5.3.5. Informed priory physical knowledge 107

5.3.6. Design of the parallel neural networks 108

5.3.7. Solving the ODEs based on the parallel neural networks 109

5.3.8. Inference of the parallel NN block 110

5.3.9. Damage Index 111

5.4. Numerical study 113

5.4.1. Numerical model 113

5.4.2. Data preparation 114

5.4.3. PNODEs architectures 115

5.4.4. Results 116

5.5. An experimental study of a three-storey building structure 120

5.5.1. Experimental setup 120

5.5.2. Data preparation 121

5.5.3. Architecture of PNODEs 122

5.5.4. Results 123

5.6. Application for three-dimensional frame structures 128

5.6.1. Experimental structure information 129

5.6.2. Data Preparation 131

5.6.3. PNODEs and NODEs architecture 132

5.6.4. Results 133

5.7. Summary 138

Chapter 6. Spatial-temporal-based physics neural operators for response reconstruction and structural identification with limited sensors 140

6.1. Overview 140

6.2. State of the art 141

6.3. Theoretical Background and Methodology 147

6.3.1. Background Information 148

6.3.2. Method of PhySTN 152

6.4. Numerical Example – nonlinear 5-storey building model with an MRE isolator 160

6.4.1. Basic information 161

6.4.2. Data preparation 164

6.4.3. Construction of the spatial-feature encoder 165
6.4.4. Reconstruction of the unknown discrepancy 166
6.4.5. Reconstruction of the missing response and structure identification 167
6.5. Experimental Example - 3-storey frame structure with MRE isolators 174
6.5.1. Basic information 174
6.5.2. Data preparation 178
6.5.3. Construction of the spatial-feature encoder 179
6.5.4. Reconstruction of the unknown discrepancy 180
6.5.5. Reconstructing the missing response and structural identification 181
6.6. Summary 184
Chapter 7. Conclusions and future research 187
7.1. Conclusions 187
7.2. Recommendations for future works 188
Reference 189

List of figures

Figure 3-1 CNN structure for regression. 23

Figure 3-2. Architecture of the CNN based fine-tuning (FT) transfer learning 26

Figure 3-3. Three-storey building structure 27

Figure 3-4. 5% (2.5 mm on each side) damage to a column at the third storey 27

Figure 3-5. The lumped mass model of the three-storey building 28

Figure 3-6. FRF for experimental and numerical models (intact structure) 29

Figure 3-7. Flow chart of CNN for structural damage detection 32

Figure 3-8. CNN configurations for the pre-trained and fine-tuned models 33

Figure 3-9. FRFs of three floors with three impact locations (intact scenarios) 36

Figure 3-10.Feature visualisation through t-SNE for different damage scenarios and impact locations (Note: D0, D1, D2 and D3 represented by (×), (▯), (○) and (◊) respectively; Brown, pink and black indicates the impact on the first, second and third floors respectively.) 37

Figure 3-11. t-SNE visualization of features extracted from the data of the source domain (Note: Red is for D0, Grey, yellow and green are for D1, D2 and D3; Light blue is the double damage scenario 20%, 0-30% damage on the first and second floors; Blue is the double damage scenario 30%, 0-30% damage on the first and second floors; Light purple is the double damage scenario 20%, 0-30% damage on the second and third floors; purple is the double damage scenario 30%, 0-30% damage on the second and third floors; Light pink is the double damage scenario 20%, 0-30% damage on the first and third floors; purple is the double damage scenario 30%, 0-30% damage on the first and third floors.) 39

Figure 3-12. t-SNE visualization of features using FT for structures with modelling errors (The data for the source domain 1 and the target domain 0 are denoted by (◊) and (⊠) respectively. Red, grey, yellow and green are for D0, D1, D2 and D3 respectively.)....40
Figure 3-13. The MSE loss of the validation dataset for each epoch with the initial and FT models40
Figure 3-14. Regression analysis of testing results without or with FT 41
Figure 3-15. Box charts for predicted and true damage severities with or without FT. 42
Figure 3-16. FRFs of three floors with different damage scenarios from the experimental model (nd is intact structure; 10d is single damage scenario 10% on the second floor; 20d is single damage scenario 20% on the second floor; 20d10d is the double damage scenario 20% and 10% damage on the second and third floors)....44
Figure 3-17. Feature visualization of FT the weight through t-SNE for single damage scenarios. ((◊), (+) and (○) represent the features of the training data from the source domain, the fine-tuning data and testing data from the target domain respectively. Blue, green and red are the intact, 10% damage and 20% damage on the second floor.) 45
Figure 3-18. Damage identification results of TD1, TD2a and TD2b from without FT and with FT 46
Figure 3-19. Box charts for predicted and true damage severities with or without FT 47
Figure 3-20. Histograms of SAPE (%) on the predicted damage severities....47
Figure 3-21.Feature visualization through t-SNE for different damage scenarios. (The data for SD2, TD3 and its testing are represented by (◊), (+) and (○) respectively. Blue, brown, red and green are D0, D1/10D2, D2/20D2 and D5/2010D5.)....49

Figure 3-22. Damage identification results of TD3 and TD4 from without FT and with FT 51

Figure 3-23. Box chart for predicted and true damage severities without or with FT. 51

Figure 3-24. Histogram of SAPE (%) for the predicted damage severities 51

Figure 4-1. Overview of the proposed framework 61

Figure 4-2. Class and domain discrepancy alignments. 62

Figure 4-3. The network architecture of the proposed JMDAD 63

Figure 4-4. Step A in the training process 67

Figure 4-5. Step B in the training process 68

Figure 4-6. Step C in the training process 69

Figure 4-7. The multi-storey building structure and the numerical models 71

Figure 4-8. Damage localization accuracy with adversarial weight ($\omega = 0\ to\ 1$) 74

Figure 4-9. The results with identical damage datasets. 75

Figure 4-10. Distributions of PCA components with identical datasets 78

Figure 4-11. The classification results with limited damage datasets 79

Figure 4-12. Testing accuracy over epoch using CNN and the proposed JMDAD method 80

Figure 4-13． Feature visualization with PCA using JMDAD with imbalanced datasets 80

Figure 4-14． Experimental setup 82

Figure 4-15. Damage localization accuracy under different excitations with adversarial weight ($\omega =$ 0 to 1) 85

Figure 4-16. JMDAD method performance for single-damage damage datasets 85

Figure 4-17.JMDAD method performance with operational environmental varieties 86

Figure 4-18.Canton Tower: (a) 16-DOF numerical model, (b) real structure, and (c) the top view. . 90

Figure 4-19. Damage localization accuracy with adversarial weight (ω = 0 to 1) with WN and wind load. 93

Figure 4-20.Testing accuracy over epochs with single and unseen damage 93

Figure 4-21. Feature visualization with PCAs with single and unseen damage dataset 94

Figure 5-1. PNODEs framework 106

Figure 5-2. Phase II. Architecture of PNODEs 110

Figure 5-3. Building structure and its excitations 114

Figure 5-4. Identified results for Damage case 1M with different measurement noise. 118

Figure 5-5. The relationship between $NNi(\mathbf{h}i(t))$ and relative displacement for each floor 119

Figure 5-6. $NNi(\mathbf{h}i(t))$ versus each relative displacement for the nonlinear system. 120

Figure 5-7. Experimental set up 121

Figure 5-8. Damage models 121

Figure 5-9. Identified results for Damage case 1M under excitations of E2, E3 and WN. 123

Figure 5-10. Output of trained $NNi(\mathbf{h}i(t))$ versus each relative displacement for Damage case 1M under E2, E3 and WN 124

Figure 5-11. Identified results for Damage cases 1L, 1M and 1S. 125

Figure 5-12. Relationship between $NNi(\mathbf{h}i(t))$ and relative displacement for damage scenarios 1L, 1M and 1S under the excitation E3. 126

Figure 5-13. Identified results for Damage cases 1M, 2M and 1M2M. 127

Figure 5-14. $NNi(\mathbf{h}i(t))$ versus relative displacement for Damage cases 1M, 2M and 1M2M 128

Figure 5-15. Sensor arrangements (a) front view and (b) top view. 130

Figure 5-16. Damage scenarios: (a) removing braces on the first floor in one bay at the southeast corner and (b) removing braces on all floors on the eastern side and on the second-floor braces on the northern side. (c) the lumped mass model 130

Figure 5-17. Identified results for Damage scenario 1 of the experimental structure. 134

Figure 5-18. The relationship between $NNn(\mathbf{h}(t))$ and relative displacement for Damage scenario 1..... 135

Figure 5-19. Identified results for Damage scenario 2 of the experimental structure 137

Figure 5-20. Output of $NNi(\mathbf{h}i(t))$ versus each relative displacement for Damage scenario 2 ... 138

Figure 6-1. Flow chart of the proposed structural identification and response reconstruction and prediction framework. 154

Figure 6-2. Details of the operational mechanisms for each step in the proposed framework 157

Figure 6-3. 5-storey building model with MRE nonlinear structural system 163

Figure 6-4. Numerical example result 1: scenario of noisy data (a) reconstructed missing response; (b) Testing MSE loss (c) reconstructed discrepancy term. 170

Figure 6-5. Numerical example result 2: scenario of noisy data (a)reconstructed missing response; (b) Testing MSE loss; (c) reconstructed discrepancy term. 172

Figure 6-6. Numerical example result 3: scenario of different observation locations (a)reconstructed missing response; (b) Testing MSE loss; (c) reconstructed discrepancy term. 174

Figure 6-7. Dimensions of the 3-storey experimental structure: (a) 3-dimensional view, (b) front view, and (c) side view. 175

Figure 6-8. (a) Experimental setup and data acquisition system (b) details of the experimental equipment. 177

Figure 6-9. Experimental example result 1: scenario of different observation locations (a)reconstructed missing response of the base floor; (b) Testing MSE loss; (c) reconstructed discrepancy term. 183

Figure 6-10. Experimental example result 1: scenario of different observation locations: reconstructed missing response of the 1st, 2nd and 3rd floor. 184

List of tables

Table 3-1. Description of source and target domains. 25

Table 3-2. Natural frequencies of experimental and numerical models 28

Table 3-3. Architecture of the CNN 30

Table 3-4. Hyperparameters of CNN 31

Table 3-5.Properties of Models 1 and 2 35

Table 3-6. Natural frequencies of Models 1 and 2. 35

Table 3-7. Sample numbers of different damage scenarios for source and target domains 38

Table 3-8. Lumped mass model, single damage case study: sample numbers in each damage scenario for each domain. 44

Table 3-9. Sample numbers of each damage scenario for source and target domains 48

Table 4-1.The network structure of the proposed JMDAD 64

Table 4-3. Parameters of Models 1 and 2 70

Table 4-2. Summary of the training process 71

Table 4-4. Damage detection with identical datasets from source and target domains 73

Table 4-5. Limited damage datasets from the target domain Model 2 79

Table 4-6. Test and sample number for single damage detection (Imbalanced class). 83

Table 4-7. Natural frequencies of CT from measurements, and FE and proposed lumped mass models. 88

Table 4-8. Damage detection with unseen datasets from source and target domains 91

Table 4-9． Accuracy of JMDAD method for damage detection 93

Table 5-1. Architecture of the network and its training hyper-parameters 116

Table 5-2. Relative errors of identified results for Damage case 1M with different measurement noise. 118

Table 5-3. Damage scenarios of the experimental frame structure. 121

Table 5-4. Updated stiffness and damping matrices for the experimental model. 122

Table 5-5. Relative errors of identified results for Damage case 1M under different excitations. .. 124

Table 5-6. Relative errors of identified results for Damage case 1L, 1M and 1S. 126

Table 5-7. Relative errors of identified results for Damage case 1L, 1M and 1S. 127

Table 5-8. Input data of NODEs and PNODEs. 132

Table 5-9. Architecture of neural network in NODEs & PNODEs, and training hyper-parameters. 133

Table 5-10. Relative errors of identified results for Damage scenario 1. 135

Table 5-11. Relative errors of identified results for Damage scenario 2. 137

Table 6-1. Cases of missing measurement and limited observation of the 5-storey numerical model. 165

Table 6-2. SE network architecture for different cases and training hyper-parameters used of the numerical example. 166

Table 6-3. The architecture of the NN network and training hyper-parameters of the numerical example. 166

Table 6-4. Numerical example Result 1: scenario of noisy data - Identified model expression $fNN\mathbf{h}t$. 170

Table 6-5. The material properties for each component of the 3-storey experimental structure. 176

Table 6-6. The updated stiffness and damping matrices and the natural frequencies of the 3-storey experimental structure. 178

Table 6-7. Cases of missing measurement and limited observation of the 3-storey experimental structure. 179

Table 6-8. SE network architecture for different cases and training hyper-parameters used in the experimental example. 180

Table 6-9. The architecture of the NN network and training hyper-parameters of the experimental example. 181

Table 6-10. Experimental example result: scenario of noisy data - Identified model expression $fNN\mathbf{h}t$ 183

Abstract

Structural condition assessment is critical for ensuring the safety and functionality of building structures, yet existing methods face significant challenges, including data scarcity, noise contamination, and limited generalisation across diverse operational environments. Traditional machine learning-based approaches often rely on extensive labelled datasets and assume consistent data distributions, which are impractical in real-world scenarios. Furthermore, these methods frequently lack interpretability, limiting their adaptation to practical applications. To address these issues, advanced frameworks are required to enhance accuracy, robustness, and scalability in structural damage detection and condition assessment.

A series of physics-guided machine learning frameworks are developed in this research to overcome these above-mentioned challenges, mainly including transfer learning and physics-informed machine learning. Transfer learning methods leverage simulated frequency response function (FRF) data to pre-train deep convolutional neural networks (CNNs) and fine-tune them using limited real-world measurements, significantly improving damage localisation and severity identification. Additionally, a Joint Maximum Discrepancy and Adversarial Discriminative Domain Adaptation (JMDAD) framework is developed to eliminate the need for labelled target data. By aligning feature distributions at both domain and class levels and leveraging transmissibility functions, this approach enhances robustness against noise and environmental variations while effectively detecting damage in real structures.

Physics-informed machine learning methods further embed physical constraints into machine learning models to improve interpretability and reliability. The Parallel Neural Ordinary Differential

Equations (PNODEs) framework integrates state-space equations to provide physical constraints, enabling accurate damage quantification and enhanced model reliability. Additionally, the Temporal-Spatial Neural Operator (PhySTN) framework combines a spatial feature mapping encoder with a physics-informed time operator to enable structural parameter identification and response reconstruction from sparse sensor data, addressing challenges in data insufficiency.

The proposed frameworks are validated through extensive numerical simulations and experimental studies, including nonlinear numerical models, experimental structures, benchmark frames, and real-world applications. These methods demonstrate significant improvements in damage detection accuracy, scalability, and interpretability, offering reliable and efficient solutions for structural health monitoring. By addressing the challenges of insufficient data and enhancing the explainability of machine learning-based condition assessment, this research contributes valuable advancements to the field.

Chapter 1. Introduction

1.1. Background

Structural health monitoring (SHM) is essential for ensuring the safety and longevity of civil infrastructure. Building structures are constantly exposed to operational conditions, environmental variations, and potential damage over time, necessitating robust methods for condition assessment and damage detection. Traditional approaches, including vibration-based methods and numerical modelling, rely heavily on predefined models or extensive labelled datasets(Avci et al., 2021a) . However, these methods often struggle with uncertainties, such as modelling inaccuracies, noise contamination, and the variability of real-world environments. These challenges make it difficult to accurately identify structural damage, especially with limited data from real structures.

Recent advancements in machine learning, particularly deep learning, have introduced promising solutions for SHM by leveraging large datasets to extract complex patterns from structural responses. Despite their potential, these data-driven methods face critical limitations, including dependency on large labelled datasets, lack of interpretability, and difficulties generalising across diverse operational scenarios (Wu et al., 2024). To address these issues, this thesis investigates the integration of transfer learning and physics-informed machine learning. These advanced approaches aim to reduce reliance on extensive labelled data by transferring knowledge from numerical simulations or related domains and embedding physical principles directly into the learning process. This combination enhances model interpretability and adaptability, offering a robust framework for structural damage detection under practical operational conditions.

1.2. Research objectives and significance

The overall research objectives can be summarised:

1) to develop a physics-guided machine learning framework for structural condition assessment using **insufficient** and **unlabelled** data.

2) to explore the **interpretability** and **explainability** of machine learning models for structural condition assessment.

3) to verify the proposed methods using a **numerical** model, **experimental** structure and **real structure**.

Based on the overall research aims, the detailed research objectives of each work can be concluded as follows:

(1) Transfer Learning-Based Structural Damage Detection Using Frequency Response Functions (FRFs)

This study focuses on developing a transfer learning approach to enhance structural damage detection when data is insufficient and labelled measurements are scarce. By utilizing frequency response functions (FRFs) obtained from impact testing, the method leverages simulated numerical data to pre-train a deep convolutional neural network (CNN), which is then fine-tuned with limited real-world measurements. This work emphasises improving the accuracy of damage localisation and severity identification while aligning with the overall aim of leveraging physics-guided techniques to address data limitations. Validation through numerical simulations and experimental studies demonstrates the method's practical applicability and effectiveness.

(2) Structural Damage Detection based on Transmissibility Functions with Unsupervised

Domain Adaptation

This research develops a domain adaptation framework to address the lack of labeled target data and distribution mismatches between training and testing data. The Joint Maximum Discrepancy and Adversarial Discriminative Domain Adaptation (JMDAD) method integrates transmissibility functions with a feature generator, domain discriminator, and dual classifiers to align domain and class-level features. By extracting and aligning damage-sensitive features, the method enhances interpretability and robustness against noise and environmental variations. Validation through case studies involving numerical, experimental, and real-world structures, including the Canton Tower, underscores its effectiveness in achieving accurate structural condition assessment with unlabeled data.

(3) Parallel Neural Ordinary Differential Equations (PNODEs) for Structural Damage Identification

This study introduces a Parallel Neural Ordinary Differential Equations (PNODEs) framework to enhance the interpretability and explainability of machine learning for structural damage detection. By integrating state-space equations with parallel neural networks, the method separates and captures discrepancies in structural responses for individual components. This approach emphasises the extraction of interpretable features for localised damage detection and structural parameter identification. The framework is validated using a three-story building structure and a benchmark frame, demonstrating reliable and accurate damage localisation and quantification while advancing the understanding of machine learning behavior in structural condition assessment.

(4) Temporal-Spatial Neural Operator (PhySTN) Framework for Structural System

Identification

This work presents a Temporal-Spatial Neural Operator (PhySTN) framework designed to enable structural parameter identification and response reconstruction with sparse and limited observations. By integrating a spatial mapping encoder to relate partial observations to full structural states and a time operator combining physics knowledge with neural networks, the framework addresses challenges associated with insufficient data. The method emphasises enhancing interpretability by embedding physics into the learning process and reconstructing missing responses. Numerical and experimental validations demonstrate its capability to accurately identify nonlinear structural parameters under varying noise levels and observation scenarios, aligning with the overall research aims of this thesis.

Research significance can be drawn:

- This research develops a framework that embeds known structural physics into neural networks, **enhancing the interpretability and accuracy** of structure identification.
- The study proposes a novel method for structural condition assessment that effectively **addresses challenges** associated with **insufficient and unlabeled data in practical applications**.
- By **improving the reliability of machine learning methods**, this work ensures their applicability and robustness for condition assessment in real-world structural monitoring scenarios.

1.3. Organisation of thesis

Chapter 2 provides a comprehensive review of structural damage detection methodologies. It

begins by introducing vibration-based approaches, including model-based and data-driven methods, followed by a detailed exploration of machine learning techniques, such as supervised, unsupervised, and self-supervised learning. Challenges in applying machine learning to structural damage detection are highlighted. The chapter also examines transfer learning and domain adaptation for structural condition assessment and concludes with a discussion on physics-informed machine learning methods, specifically focusing on Physics-Informed Neural Networks (PINNs), Neural Ordinary Differential Equations (NODEs), and Physics-Informed Neural Operators (PINOs).

Chapter 3 presents a novel transfer learning-based approach for structural damage detection using frequency response functions (FRFs) obtained from limited measurements. The methodology involves pre-training a deep convolutional neural network (CNN) on simulated FRF data from a numerical model representing various damage scenarios, followed by fine-tuning with experimental data from a three-story building structure. The proposed method was validated through numerical and experimental studies, demonstrating superior performance compared to traditional CNN models, particularly in identifying damage location and severity under limited measurement conditions.

Chapter 4 introduces a novel method for structural damage detection based on joint maximum discrepancy and adversarial discriminative domain adaptation (JMDAD), addressing challenges posed by data scarcity, modelling errors, and environmental variations. The method utilises transmissibility functions of measured structural responses and combines a feature generator, two classifiers, and a discriminator to align features at both domain and class levels. Through three case studies, including applications to numerical, experimental, and real-world structures, the approach demonstrates robustness to noise and environmental variations, accurately identifying structural

damage without requiring labeled data from the target structure.

Chapter 5 presents a novel parallel neural ordinary differential equations (PNODEs) method for structural damage identification. This approach integrates state-space equations with a neural network block comprising parallel networks, each representing a structural component. By embedding physical constraints from the state-space model, the method captures and separates discrepancies in structural responses to each part, enhancing interpretability and reliability. Validated through studies on a three-story building and a benchmark frame, the method demonstrates high accuracy in localizing and quantifying structural damage.

Chapter 6 introduces a temporal-spatial neural operator (PhySTN) framework for structural parameter identification using limited observations. The framework combines a spatial feature mapping encoder to relate partial observations to the full system state with a time operator integrating physical knowledge and a neural network to capture discrepancies. This approach enables accurate parameter identification and response reconstruction under limited sensor data. Validation through a nonlinear numerical model and experimental frame structure demonstrates the framework's effectiveness in handling noise, sparse observations, and nonlinear behaviors.

Chapter 2. Literature review

This chapter concludes a review of the current challenges in the field of structural health monitoring, while the detailed background of specific issues can be found under each chapter in state of the art. In this chapter, the concept of structural health monitoring was introduced first. Then, a detailed review of vibration-based structural damage detection and machine learning-based structural damage detection is presented, highlighting key issues. Transfer learning and physics-informed machine learning structural condition assessment are followed. Lastly, the research gaps and challenges of damage detection based on machine learning methods are found through the review, setting the stage for the research investigations.

2.1. Concepts of structural health monitoring

Structural Health Monitoring (SHM) is characterized by the deployment of on-structure sensing systems to oversee structural performance and assess health conditions. The primary objective of SHM is to accurately identify and localize damage within the limits of sensor resolution, assess its severity, and detect damage at an early stage. Furthermore, predict the structure's remaining lifespan without compromising its integrity for condition assessment. As defined by Rytter (Rytter et al., 1993), the goals of SHM are categorized into four hierarchical levels:

Level 1: Detecting of damage within the structure

Level 2: Localizing of the detected damage

Level 3: Qualifing of the damage severity

Level 4: Estimating of the remaining service life

Researchers (Moughty & Casas, 2017) pointed out that vibration-based identification techniques

are capable of achieving damage detection up to Level 3. However, in the absence of a precise numerical model, these vibration-based methods face limitations in predicting the remaining service life of structures at Level 4 (Limongelli et al., 2016).

Vibration-based methods (VDD) and non-destructive testing (NDT) are the two main categories of damage detection techniques (Hou & Xia 2021). The former method can quickly identify the damage location inside the structure or the damage due to non-structural components. Rather than the NDT method, VDD identification methods are enabled to detect structural global vibration characters that have been popularly used as global methods during the past years. Consequently, the major focus of this study will be on methods for identifying damage that is based on vibration measurements.

A review of existing research on vibration-based methods was conducted by Doeling et al. (1996) and Sohn et al. (2003). Salawu (1997) explored detecting damage methods using natural frequencies. In addition, between 1996 and 2003, Carden and Fanning (2004) published a number of detailed literature evaluations. Furthermore, Fan and Qiao (2011) examined and contrasted several damage identification techniques. Several types of research (Hankim and Razak 2014; Chen et al. 2014) have also determined that The use of artificial neural networks (ANNs) for signal processing and damage detection has grown significantly in recent decades. In addition, the VDD is divided into approaches that use model-based and data-driven based methods. The key differentiation between model-based and data-driven methods for detecting structural damage lies in their methodological basis. Model-based methods rely on physical or numerical models to simulate the structure's behaviour, using deviations between predicted and observed responses to detect damage. In contrast, data-driven methods focus on identifying patterns and anomalies directly from sensor data without relying on

explicit physical models, employing techniques such as machine learning and statistical analysis to infer structural health conditions. In addition, parametric and non-parametric approaches are included in data-driven methods. By replacing model parameters with actual data, the parameter-based approaches aim to create a computational model, including a finite element model. Although these methods can achieve high-accuracy predictive results, they usually require prior knowledge and associated assumptions regarding the actual structure, which is inaccurate enough and not accessible. The non-parameter technique uses statistical models to directly transform the vibration signals into structural features without involving any prior model information, as opposed to parametric-based methods.

2.2. Vibration-based structural damage detection

2.2.1. Model-based methods

In model-based damage detection, Finite Element (FE) model updating is frequently employed as a key technique, aiming to refine computational models to match actual structural responses. This approach involves modifying model parameters, such as material properties, stiffness, or damping coefficients, to minimize discrepancies between measured and simulated dynamic responses (Mottershead & Friswell, 1993). One key advantage of FE model updating is its strong physical interpretability, allowing clear identification of structural behavior changes. Additionally, it can effectively model complex structures by incorporating detailed physical parameters (Reynders et al., 2010). However, it has notable limitations, including sensitivity to measurement noise and model uncertainties, which can lead to convergence issues or non-unique solutions (Reynders & Roeck, 2008). Moreover, the high computational cost of iterative updating processes poses challenges for

large-scale structures. Despite these drawbacks, FE model updating remains essential for damage detection due to its capacity to bridge physical modelling with real-world observations.

2.2.2. Data-driven Methods

2.2.2.1. Parametric methods

Data-driven approaches for structural damage detection frequently employ modal parametric methods, which utilize extracted modal parameters like natural frequencies, damping ratios, and mode shapes to evaluate structural health (Doebling et al., 1996; Wei Fan & Pizhong Qiao, 2011). The main advantage of these methods lies in their strong theoretical basis and the clear physical interpretation of identified parameters. They are particularly effective for detecting global damage and can be applied using relatively sparse sensor networks. However, modal parametric methods are sensitive to environmental influences and operational conditions, which can introduce variability unrelated to damage, thereby affecting reliability (B. Peeters & De Roeck, 2001). Additionally, these methods may struggle with local damage identification due to their focus on global modal characteristics. Despite these limitations, their simplicity and broad applicability make them a valuable tool in the early stages of structural health monitoring.

2.2.2.2. Nonparametric methods

Nonparametric methods in structural damage detection focus on extracting statistical features directly from raw vibration signals without relying on predefined physical models. These methods combine time-series analysis with statistical classification to identify damage-sensitive features, making them more flexible in capturing complex damage patterns. A key advantage of nonparametric approaches is their ability to detect damage features that are not easily attributed to explicit physical

changes, thus allowing for broader application in various structures. For instance, techniques like Auto-Regressive Moving Average (ARMA) models (Krishnan Nair & Kiremidjian, 2007) and Principal Component Analysis (PCA) (Goi & Kim, 2017) have shown effectiveness in isolating damage indicators and classifying damage states. However, the primary limitation of these methods lies in their sensitivity to noise and environmental variations, which can lead to false positives or reduced accuracy. Additionally, nonparametric methods often require substantial amounts of training data and sophisticated statistical algorithms to function effectively. Despite these challenges, they provide a valuable alternative to model-based methods, particularly in cases where physical models are unavailable or insufficient.

2.3. Machine learning for structural damage detection

2.3.1. Supervised learning for structural damage detection

In structural damage detection, supervised learning techniques are widely utilized for their capacity to leverage labelled datasets during model training. These methods, such as artificial neural networks (ANNs) (Yeung & Smith, 2005), support vector machines (SVMs) (Cury & Crémona, 2012) and convolutional neural networks (CNNs) are highly effective for classifying structural conditions and estimating damage severity by identifying intricate patterns between input features and labeled damaged data (Abdeljaber et al., 2018). The main advantage of supervised learning lies in its high accuracy and reliability in well-defined damage scenarios, as these models can efficiently map structural features to corresponding damage levels. However, the major limitation of supervised learning is its dependency on large, high-quality labeled datasets, which are often difficult and costly to obtain in structural health monitoring applications. Additionally, supervised models may struggle

with generalizing to new or unforeseen damage types, reducing their adaptability in diverse operational conditions. Despite these challenges, supervised learning remains a popular approach due to its effectiveness in controlled environments and when sufficient labeled data is accessible

2.3.2. Unsupervised learning for structural damage detection

Unsupervised learning methods for structural damage detection offer flexibility in dealing with unlabeled data, making them suitable for scenarios where labeled datasets are challenging to obtain. These methods typically rely on clustering techniques or anomaly detection algorithms to uncover hidden patterns and identify deviations from normal structural behavior (Silva et al., 2016). A main advantage of unsupervised learning is its ability to detect unexpected or unknown damage types, making it more adaptive to real-world conditions. Furthermore, it reduces the need for extensive human intervention in data labeling, which can be costly and labor-intensive. However, unsupervised approaches are sensitive to environmental and operational variability, which can lead to false positives or misinterpretations. In addition, their effectiveness is strongly influenced by the quality and diversity of the input data, making feature selection and preprocessing critical. Despite these challenges, unsupervised learning methods remain a promising approach for scalable and adaptive damage detection systems in structural health monitoring

2.3.3. Self-supervised learning for structural damage detection

Self-supervised learning has emerged as an effective approach for structural damage detection, particularly in situations where labeled datasets are limited or challenging to acquire (K. Zhang et al., 2020). By creating pretext tasks, such as predicting missing signals or reconstructing noisy data, this approach allows models to extract meaningful patterns from unlabeled data, thereby enhancing their

ability to identify structural anomalies. A key advantage of self-supervised learning is its capability to generalize well to new damage scenarios by learning representations that capture intrinsic relationships within the data(Akrim et al., 2023). This adaptability makes it valuable in real-world applications where structures encounter varying conditions and unforeseen damage types (Thai, 2022). However, a significant challenge lies in designing pretext tasks that are both relevant to structural behaviors and effective in highlighting damage-related features. Moreover, the effectiveness of learned representations is significantly influenced by the robustness and diversity of the training data. Self-supervised learning, despite its challenges, offers an adaptable and scalable solution, bridging the gap between supervised and unsupervised methods in structural health monitoring.

2.3.4. Challenges in machine learning for structural damage detection

Machine learning (ML) applications in structural damage detection encounter several significant challenges.

- First, data quality and availability pose significant limitations. High-quality labelled datasets are essential for supervised learning models, but obtaining them is often labor-intensive and costly, especially in structural health monitoring (Silva et al., 2016).
- Secondly, the issue of overfitting persists as a major challenge, particularly in data-driven methods, where complex models perform well on training datasets but often lack generalization to new damage conditions. Additionally, environmental and operational variability can affect the accuracy of ML models, leading to false positives or obscuring damage patterns.

- Another challenge is the interpretability of ML models, especially with deep learning approaches, which can limit the validation and acceptance of results in practical engineering contexts.
- Finally, designing self-supervised or unsupervised learning tasks that effectively capture structural behaviour remains an ongoing research area. Addressing these challenges is critical for improving the reliability and scalability of ML-based damage detection systems.

2.4. Transfer learning based structural damage detection

2.4.1. Transfer learning (TL)

Transfer Learning (TL) has attracted growing interest in structural damage detection for its capability to utilise knowledge from a source domain to enhance learning in a similar target domain. In structural health monitoring, When labeled data in the target domain are difficult or expensive to acquire, Transfer Learning (TL) becomes a particularly effective approach (Lin et al., 2022). A typical method entails training a model on numerical data from a source domain and adapting it to a target structure with sparse real-world data. This method enables the transfer of learned patterns and features, improving damage detection accuracy despite variations in structural conditions (X. Wang & Xia, 2022). The benefits of Transfer Learning (TL) include minimising the dependence on large labeled datasets and enhancing the ability to adapt across varying operational and environmental conditions. However, challenges such as domain discrepancies and feature mismatches between source and target domains can impact its effectiveness. Addressing these issues through techniques like domain adaptation or adversarial learning is crucial to enhancing the robustness of TL-based approaches in

structural health monitoring.

2.4.2. Domain adaptation (DA)

In structural condition assessment, Domain Adaptation (DA) has gained recognition as an important technique, particularly when applying models trained source domain to assess structures in a different target structure with limited labelled data (M. Wang & Deng, 2018). DA aims to bridge the gap between variations in structural responses due to environmental and operational differences, ensuring that machine learning models can generalize effectively to new conditions without requiring extensive retraining (Vadyala et al., 2022). Methods such as feature alignment and adversarial training are commonly employed in DA, enabling the model to learn invariant features that are shared across domains. This is especially beneficial in civil infrastructure monitoring, where changes in temperature, load conditions, or structural aging may create discrepancies that impact model accuracy(Z. Chen et al., 2022; Ganin et al., 2017). While DA significantly improves adaptability and reduces data requirements, challenges remain in achieving complete feature alignment and minimizing domain shift, which can introduce biases in model predictions. Addressing these limitations continues to be a focus of research to enhance DA's effectiveness in real-world structural health monitoring applications.

2.5. Physics informed machine learning for structural condition assessment

2.5.1. Physics informed neural networks (PINNs)

Physics-informed machine learning (PIML), especially through Physics-Informed Neural Networks (PINNs), are increasingly applied in structural condition assessment due to their ability to integrate physical laws, such as partial differential equations, directly into the machine learning model.

By embedding these governing equations as constraints, PINNs can model structural behaviors with improved accuracy and consistency, even in cases where data is scarce or noisy (George Em Karniadakis 1, n.d.). Integrating data-driven learning with physics-based principles minimizes the need for large labeled datasets, positioning PINNs as valuable in real-world scenarios where acquiring labelled damage data is difficult (Wu et al., 2024). Furthermore, PINNs offer enhanced interpretability compared to traditional black-box models, as their outputs adhere to known structural behaviors.

2.5.2. Neural ordinary differentiation equations (NODEs)

Neural Ordinary Differential Equations (NODEs) offer a promising approach in structural condition assessment, particularly in handling inverse problems and learning residual patterns indicative of structural anomalies (Lai et al., 2021). Unlike traditional neural networks, NODEs treat layers as continuous transformations governed by differential equations, making them highly suitable for modeling dynamic systems and capturing subtle structural changes over time (R. T. Q. Chen et al., 2018). NODEs excel in directly inferring system dynamics by learning the discrepancies between baseline models and observed data, thus identifying deviations that may signal damage or degradation. This approach enhances interpretability, as NODEs can identify the specific deviations from expected physical behavior, allowing engineers to assess structural health with greater accuracy. Furthermore, NODEs require fewer parameters than deep networks, reducing computational demands while maintaining high accuracy in representing system dynamics.

2.5.3. Physics informed Neural Operators (PINOs)

Physics-Informed Neural Operators (PINOs) extend the concept of learning mappings between function spaces, making them particularly suitable for addressing high-dimensional problems

governed by partial differential equations (PDEs). Unlike Neural Ordinary Differential Equations (NODEs), which focus on dynamic systems and model continuous transformations in time, PINOs are designed to handle spatial-temporal systems by learning operators that directly approximate solutions of PDEs (Z. Li et al., 2024). This capability allows PINOs to generalize across different geometries and boundary conditions, enabling broader applications in structural health monitoring.

The primary advantage of PINOs lies in their flexibility and efficiency in modeling complex, multi-physics problems without the need to discretize the entire domain explicitly. In comparison, NODEs are more suitable for problems with time-dependent dynamics but may struggle with spatially varying phenomena. Furthermore, PINOs require fewer labeled data points by leveraging embedded physical constraints, making them ideal for scenarios with limited measurements. However, PINOs demand higher computational resources during training due to the evaluation of PDE residuals and boundary conditions, which could lead to a limitation for real-time implementation. PINOs, despite these difficulties, demonstrate significant advancements in scalability and adaptability for structural condition assessment in various problem areas.

2.6. Research gaps and challenges

The use of machine learning (ML) in structural damage detection poses several critical challenges that impact its accuracy and reliability:

- **Data quality and accessibility**: High-quality, labelled datasets are critical for training supervised ML models. However, obtaining such data is challenging due to noise contamination, high costs, and the rarity of labelled damaged data. This scarcity particularly affects the model's ability to generalize, as limited data can lead to biased learning outcomes

and reduced detection accuracy in real-world scenarios.

- **Less damage data**: Structural damage data is inherently limited since real damage events are often irreversible. This constraint restricts the availability of comprehensive datasets, forcing researchers to rely on simulated or limited measurement data, which may not accurately capture all damage scenarios, thereby reducing the robustness of ML models in varying structural conditions.

- **Lack of physical interpretability**: Many ML models, particularly deep learning approaches, operate as "black boxes," offering limited interpretability. This lack of physical meaning poses challenges in practical applications, where engineers require models that align with known structural behaviour and can be trusted in decision-making processes.

Addressing these challenges is critical for advancing ML in structural health monitoring, enabling more reliable, interpretable, and flexible solutions for a variety of structural environments.

Chapter 3. Transfer learning-based structural damage identification for building structures with limited measurement data

3.1. Overview

Detecting structural damage is essential to maintaining the safety of civil building structures under operational conditions. Recently, deep learning-based approaches have drawn significant interest from engineers and researchers for this application. However, traditional deep learning approaches rely on challenging assumptions, namely the availability of a large labeled dataset and identical distribution between training and testing data, which are often impractical. This research presents a new transfer learning approach for structural damage detection, leveraging frequency response functions (FRFs) with limited measurement data. Impact testing was conducted on a three-storey building structure in a laboratory, measuring impact force and acceleration responses at each floor to obtain FRFs. A numerical model was developed to simulate FRF data for various damage scenarios. A custom deep convolutional neural network (CNN) was initially trained with numerically simulated data and subsequently fine-tuned using limited measurements from the actual structure. Numerical and experimental studies were conducted to validate the proposed method's effectiveness and accuracy. Results indicate that the transfer learning-based approach outperforms traditional CNN models with enough datasets. This approach is highly effective for determining the location and severity of structural damage in real-world structures, especially when measurement data are limited in practical applications.

3.2. State of the art

Civil infrastructure faces potential damage from aging, operational use, and environmental

factors, posing safety risks. Early detection of structural damage is crucial to prevent severe consequences and ensure safety. Vibration-based damage detection, exploring changes in dynamic responses, has been extensively studied. Two main categories include model-based methods, relying on accurate numerical modeling, and data-driven approaches. Model-based approaches, such as (Manoach et al. 2017), require precise numerical models but are susceptible to uncertainties like modeling errors, operational variations, and measurement noise (Meruane and Heylen 2012; Kostic and Gul 2017; Gao and Mosalam 2022). Detecting structural damage early on is crucial to mitigate risks and prevent costly failures.

The growing interest in data-driven approaches for structural damage detection is evident in recent research efforts (Ye et al. 2019; Avi et al. 2021; Cheraghzade and Roohi et al. 2022; Yu et al. 2023b; Hao et al. 2022; Sony et al. 2021). Leveraging abundant structural monitoring data, deep learning (DL) models, such as the deep convolutional neural network (CNN) models proposed by Lin et al. (2017) and Abdeljaber et al. (2018), have demonstrated capabilities in localizing and quantifying damage using raw time-domain responses and acceleration measurements. However, these traditional data-driven methods encounter practical limitations. Firstly, they assume that measurements for damage prediction share the same distribution as those used to train the DL model, a presumption often invalidated by modeling errors, measurement noise, and environmental variations. This discrepancy can render the trained model inadequate for predicting damage in the target structure. Secondly, achieving accurate predictions necessitates a large amount of labelled data representing various damage states, a costly and time-consuming endeavor. Addressing these challenges, this research introduces a transfer learning-based method for structural damage identification, designed

to improve the efficiency and practicality of data-driven approaches.

Transfer learning (TL) emerges as a solution to address data insufficiency in structural health monitoring (SHM) by leveraging numerical simulations to provide ample labelled data in the source domain. In SHM practice, obtaining labelled data from real structures in the target domain is limited (Kandel and Castelli 2020; Mishra et al. 2022). TL helps alleviate label sparsity by incorporating unlabeled data from the structure, effectively expanding the dataset (Gao et al. 2018). Previous studies, such as Yu et al. (2022, 2023a) and Azimi and Pekcan (2019), showcased the efficacy of TL in tasks like concrete crack detection and structural damage investigation using deep neural networks. Lin et al. (2022) demonstrated domain adaptation for structural damage detection, outperforming traditional CNN models. Wang and Xia (2022) addressed label space inconsistency between domains using re-weighted adversarial domain adaptation (RADA), achieving robust damage detection even with varying structural characteristics. Han et al. (2022) presented a CNN-based method for localizing and quantifying bolt loosening in large-span spatial structures, employing data augmentation and TL to enhance accuracy with limited measurements. Despite these advancements, limited studies have explored the use of labelled data from real structures to predict damage severity, and there is a gap in understanding the interpretability of features extracted by deep learning (DL) models in relation to physical performance. One main objective of this research is to improve the performance of datasets in the target domain by utilizing the characteristics of the main dataset in the source domain. Specifically, the proposed method fine-tunes the fully connected layer (FC) layers to adapt the network weights for the target domain, building on insights from Too et al. (2019) and Mishra and Passos (2021). This approach addresses the challenges of sparsely labelled data and enhances the

interpretability of DL-extracted features, contributing to the advancement of reliable and explainable structural damage detection methods.

The chapter focuses on designing an innovative method for structural damage detection, leveraging transfer learning to address challenges posed by limited measurement data. Validation is performed using a three-storey building structure, where the real structure and its numerical model represent the target and source domains, respectively. A custom CNN model is pre-trained using various damage scenarios generated from the numerical model. Subsequently, the pre-trained DL model undergoes fine-tuning using limited measurement data from the actual structure. This approach optimally utilizes labelled information from both simulated and physical models across source and target domains, creating a generalized DL model. TL for SHM involves training the network with abundant source domain data from a numerical model, adjusting weights on the fully connected layer using target domain data, and testing it with additional target domain data under corresponding damage scenarios. Numerical and experimental studies confirm the effectiveness and accuracy of the proposed method in identifying structural damage with limited measurement data from real structures.

3.3. Theoretical background

3.3.1. The convolutional neural networks (CNN)

The CNN, known for its advantages in receptive fields, spatial down-sampling, and weight sharing (Krizhevsk et al. 2012), serves as the backbone in transfer learning-based structural damage detection. Figure 1 depicts the CNN architecture for regression, consisting of Conv-ReLU-Pooling, Fully Connected (FC), and Output modules. Batch normalization (BN) is typically employed for model training, contributing to the overall effectiveness of the system.

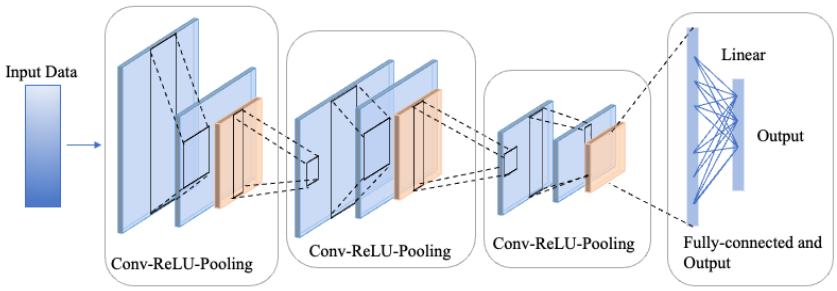


Figure 3-1 CNN structure for regression.

Conv-ReLU-Pooling module consists of a convolutional layer (Conv), an activation function layer (such as rectified linear unit - ReLU or sigmoid), and a pooling layer (Nair and Hinton 2010; Yu et al. 2022). In this block, the convolutional layer scans the input, generating dot products with kernels across the feature diagram. ReLU is a common activation function, preventing gradient vanishing. The pooling layer condenses the feature map spatially, enhancing robustness without parameters, unlike convolutional layers. Max and average pooling are common techniques, calculating the maximum or average values in receptive regions, respectively. These components collectively contribute to feature extraction and nonlinearity in the convolutional neural network architecture.

The FC and Output module in a CNN consists of FC layers following Conv-ReLU-Pooling modules. Similar to a conventional multilayer perceptron, flattened features from earlier modules are processed by FC layers for classification or regression. The linear function is commonly used for precise output label values in regression tasks.

BN is introduced to speed up the training and handle internal covariate shifts by normalising and re-scaling the input data. BN is located between the convolutional layer and the nonlinear activation

operation function (Bjorck et al. 2108).

3.4. Methodology

3.4.1. Problems Definitions

In transfer learning (TL), it is important to define two main objectives: a domain and a task

A domain $D = \{\mathcal{X}, P(X)\}$ covers the feature $\mathcal{X}$ and a marginal distribution $P(X)$, where $X = \{x_i\}_{i-1}^{N} \in \mathcal{X}$.

A task of this domain is defined as $\mathcal{T} = \{\mathcal{Y}, f(.)\}$, including $\mathcal{Y}$ is the label space and $f(.)$ predictive function (also considered as conditional distribution $P(y|x)$), learnt from the training data set $\{x_i, y_i\}_{i-1}^{N}$ where $y \in \mathcal{Y}$.

TL is given a source domain $D_S = \{\mathcal{X}_S, P(\mathcal{X}_S)\}$ and its task $\mathcal{T}_S = \{\mathcal{Y}_S, f_S(.)\}$, and the target domain $D_T = \{\mathcal{X}_T, P(\mathcal{X}_T)\}$ and the task $\mathcal{T}_T = \{\mathcal{Y}_T, f_T(.)\}$. Assuming that $D_S \neq D_T$ and $\mathcal{T}_S \neq \mathcal{T}_T$, the TL is conducted to improve the target predictive function $f_T(.)$ in $\mathcal{T}_T$ by transferring the knowledge learned from D_S and $\mathcal{T}_S$ to the target domain. TL methods are based on whether $\mathcal{X}, P(X), \mathcal{Y}$ and $P(y|x)\}$ are consistent across the source and target domains. Fine-tunning is one of the TL methods: for a given source domain D_S and its task $\mathcal{T}_S$, the target domain and its task are D_T and $\mathcal{T}_T$. It is conducted to share the part of parameters previously learned from D_S, and then reweight the residual parameters using some labelled datasets from D_T to improve the target prediction function $f_T(.)$ in $\mathcal{T}_T$. It is noted $P(\mathcal{X}_S) \neq P(\mathcal{X}_T)$ and $P(\mathcal{Y}_S|\mathcal{X}_S) \neq P(\mathcal{Y}_T|\mathcal{X}_T)$.

For the case study conducted here, the source and target domains are presented in different but similar structures, while the task is to represent the damage data to improve damage detection. The feature spaces for source and target domains are the same and their marginal distributions are different

e.g. $\mathcal{X}_S = \mathcal{X}_T, P(\mathcal{X}_S) \neq P(\mathcal{X}_T)$. The tasks are different but relative ($\mathcal{T}_S \neq \mathcal{T}_T$), in which the label spaces are the same and their prediction functions are different, e.g. $\mathcal{Y}_S = \mathcal{Y}_T, P(\mathcal{Y}_S|\mathcal{X}_S) \neq P(\mathcal{Y}_T|\mathcal{X}_T)$. Regarding the studies in this work, the damage state data for the model provides the same feature space, and the marginal distribution and the conditional distribution are various from different structures forming the sources and target domains.

3.4.2. Fine-tuning techniques for TL

The fine-tuning (FT) transfer learning method involves sharing learned parameters from a source domain with a target domain, where a pre-trained model predicts labels in the target domain using limited labelled data. The algorithm comprises two steps: training an initial CNN on well-labelled source data, extracting comprehensive features; and fine-tuning with limited target domain data, where only the fully connected and output layers are retrained. Structural damage detection involves learning labelled data from D_S, fine-tuning the model using D_{FT}, and testing it on D_{Test}. Table 3-1 provides details of source and target domains. This approach optimally leverages existing knowledge for effective structural damage prediction with minimal labelled target domain data.

Table 3-1. Description of source and target domains.

	Domain	**Task**	**Classes**	**Datasets**
Source Domain	$D_S = \{\mathcal{X}_i^S, \mathcal{Y}_i^S\}$ $i = 1, \ldots, n_s$	$\mathcal{T}_S = \{\mathcal{Y}_S, f_S(.)\}$	N_s	$D_{Train} \subset D_S$
Target domain	$D_{T1} = \{\mathcal{X}_j^T, \mathcal{Y}_j^T\}$ $D_{T2} = \{\mathcal{X}_j^T\}$ $j = 1, \ldots, n_t$	$\mathcal{T}_T = \{\mathcal{Y}_T, f_T(.)\}$	N_t $(N_t \subset N_s)$	$D_{FT} \subset D_{T1}$ $D_{Test} \subset D_{T2}$

Figure 3-2 illustrates the CNN-based TL architecture. The numerical model is validated against the experimental model, generating dynamic responses. The initial CNN is trained on datasets from numerical simulations of the three-storey building, including intact and damaged scenarios. Subsequently, the model is fine-tuned with limited measurement data, updating only the last FC and

output layers based on the new structural dynamic features. The fine-tuned model is tested on measurement data to predict real structural damage location and severities. Extracted features are visualized for damage localization and qualification.

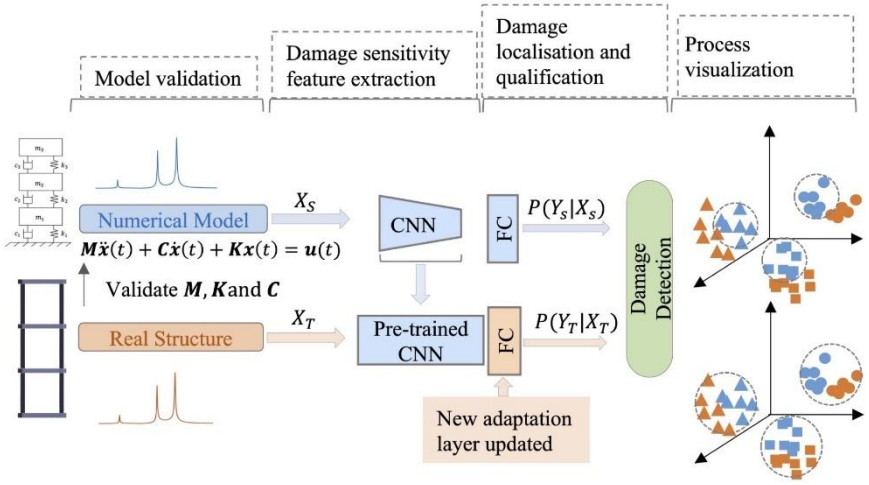


Figure 3-2. Architecture of the CNN based fine-tuning (FT) transfer learning

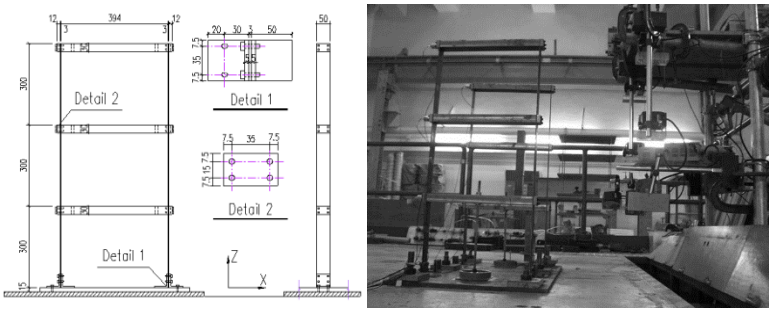
3.5. Data collection

3.5.1. Experimental tests

Figure 3-3 shows the three-storey building structure. Dynamic responses are measured using three sensors, each installed on a separate floor of the structure. The basic components of the structure were two identical columns with a 50 ×3 mm cross-section, and a 900 mm length displayed, as well as three steel beams measuring 394 (length)×50 (width)×30 (thickness) mm. The height of each storey is 300 mm, and all structural components are made of high-strength steel featuring a yield stress of 435 MPa and a modulus of elasticity of 200 GPa.

Various damage scenarios, including single and double damage, were simulated in the three-storey building structure. Symmetrical saw cuts on the column induced damage, as illustrated in

Figure 3-4. For experimental data, two symmetrical cuts simulated the damage scenario. Four structural conditions were considered: intact (D0), 10% damage on the second floor (D1), 20% damage on the second floor (D2), and double damage with 20% on the second and 10% on the third floor (D5). Impact testing with an impact hammer was conducted, measuring dynamic responses on each floor with piezoelectric accelerometers. The obtained time-domain data were transformed into frequency response functions (FRFs) with a frequency range of 30 Hz and 660 data points, creating 2D data samples representing structural vibrations. A total of 648 FRF datasets were collected through 54 repeated tests for each of the four scenarios.



(a) Configuration of the building structure (b) Experimental setup

Figure 3-3. Three-storey building structure

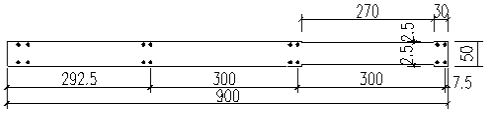


Figure 3-4. 5% (2.5 mm on each side) damage to a column at the third storey

3.5.2. Numerical modelling

A shear building model can be reduced to a lumped mass representation in a two-dimensional plane. The lumped mass model for a three-storey structure is shown in Figure 3-5. The structural system is expressed as $\boldsymbol{M}\ddot{\boldsymbol{x}}(t) + \boldsymbol{C}\dot{\boldsymbol{x}}(t) + \boldsymbol{K}\boldsymbol{x}(t) = \boldsymbol{u}(t)$. In which, Where $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{K}$ correspond to the mass, damping, and stiffness matrices of the system. $\ddot{\boldsymbol{x}}(t)$, $\dot{\boldsymbol{x}}(t)$ and $\boldsymbol{x}(t)$are the

acceleration, velocity and displacement respectively. $\boldsymbol{u}(t)$ is the input force vector. The mass of the first, second and third floors is 5.20 kg, 4.99 kg, and 4.92 kg, respectively. The lumped mass model was constructed using Matlab v.2020a.

(a)The share-type building model (b) The lumped mass model

Figure 3-5. The lumped mass model of the three-storey building

Before generating data, the validity of numerical model is confirmed by comparing it with experimental results. Table 3-2 displays natural frequencies without damage, revealing small differences 1.61%, 1.18%, and 0.19% confirming the accuracy of numerical model. Figure 3-6 illustrates FRFs for three floors from both models, highlighting that uncertainties in stiffness and mass contribute more to structural behavior changes than the numerical-experimental model discrepancies. The validation process assures the numerical model's reliability in representing the three-storey building structure accurately.

Table 3-2. Natural frequencies of experimental and numerical models

Mode	**First mode**		**Second mode**		**Third mode**	
	Freq. (Hz)	Freq. Change	Freq. (Hz)	Freq. Change	Freq. (Hz)	Freq. Change
Numerical model	5.59	-	15.27	-	20.64	-
Experimental model	5.50	1.61%	15.09	1.18%	20.68	0.19%

To train the network comprehensively, numerical simulations were conducted to simulate various damage severities and locations, allowing the network to learn diverse features of structural

damage from the source domain data. Using the validated model, seven damage scenarios are simulated, ranging from intact (D0) to 30% stiffness reduction on different floors (D0 is the intact structure, D1 is the 0-30% damage on 1st floor, D2 is the 0-30% damage on 2nd floor, D3 is the 0-30% damage on 3rd floor, D4 is the 0-30% damage on 1st and 2nd floor, D5 is the 0-30% damage on 2nd and 3rd floor, and D6 is the 0-30% damage on 1st and 3rd floors). Applying an 840 N impact force at three reference points on each floor, time-history responses are obtained, corresponding to experimental sensor locations. FRFs are then calculated with a range of 0-30 Hz. Each FRF dataset comprises 660 spectral lines with a frequency resolution of 0.045 Hz. In total, 1701 (81 damage severities×3 impact locations × 7 damage scenarios) FRF datasets are generated. The details of training, fine-tuning (validating) and testing dataset are provided in each subsection for section 5.3 and 5.4.

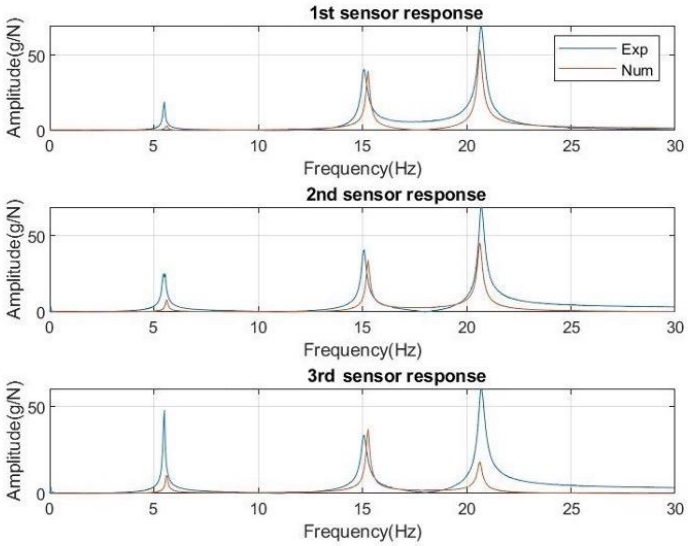


Figure 3-6. FRF for experimental and numerical models (intact structure)

3.6. Results and discussions

3.6.1. CNN training

The proposed TL-based structural damage detection relies on a CNN as its backbone, and its architecture and hyperparameters are detailed in this section. For feature identification through FT, the same CNN model is used for training and weight tuning to handle the FRF. Table 3-3 outlines the architecture of CNN, comprising convolutional and pooling layers with small kernel sizes for capturing local features. The network's capacity improves with an increasing number of layers, enhancing its ability to recognize complex patterns. Four avg-pooling layers minimize the feature map size, and to avoid premature feature loss during training, kernel numbers in convolutional layers increase with depth. The CNN concludes with a few FC layers for regression, providing predictive values with extensive capacity at a low computational cost.

Table 3-3. Architecture of the CNN

Layer	Input	Type	Kernel num.	Kernel size	Stride	Activation	Output
Layer 1	(3,660)	Conv2D	16	(1, 55)	1	ReLU	(3, 202)
		BatchNorm2D	16				
		AvgPool2D		(1, 3)			
Layer 2	(3,202)	Conv2D	16	(2, 10)	1	ReLU	(2, 96)
		BatchNorm2D	16				
		AvgPool2D		(1, 2)			
Layer 3	(2,96)	Conv2D	16	(1, 5)	1	ReLU	(2, 92)
		BatchNorm2D	32				
Layer 4	(2,92)	Conv2D	32	(2, 2)	1	ReLU	(1, 91)
		BatchNorm2D	32				
Layer 5	(1,91)	Flatten	32		None	None	(1, 3)
		Linear	None				

By processing the FRFs to compute the amplitude as input, the CNN is employed to create the

damage identification algorithm. In Section 4, it is emphasized that the calculated signal should be the number of levels × frequency length. Extracting the first 30 Hz signal to identify features, a large kernel size is realistic for the 2D vibration signal matrix. Using a 3×660 (three sensors) signal matrix as input and 1×3 as output for training the CNN model, a kernel size of 50×1 is selected. Datasets for various damage scenarios train the CNN model, each containing 243 randomly generated damage severities.

Meanwhile, all activation functions in each network layer are scaled as rectified linear units (ReLU), activating nonlinear mapping. ReLU returns positive values unchanged but converts zero or negative input values to zero. Using ReLU in convolutional layers enhances training speed, allowing for efficient gradient processing with low computational cost.

During the training, the Adam algorithm substitutes stochastic gradient descent in deep neural networks (Kingma and Ba, 2014). L^2 regularization (weight decay) is added to control layer weights, enhancing network generality. Epochs regulate the training duration, stopping when the best performance is achieved based on validation data. The optimal epoch value ensures network efficiency when the best performance is not updated.

Table 3-4. Hyperparameters of CNN

Name	Value	Description
Batch size	128	The number of training examples used in one integration
Step per epoch	600	The number of steps in a training epoch
Initial learning rate	0.001	Learning rate at which the optimization method is initially trained
Momentum	0.9	Weight of the momentum term in Adam
L^2 rate	0.003	Considering L2 regularisation (Weight decay)
Shuffle	True	

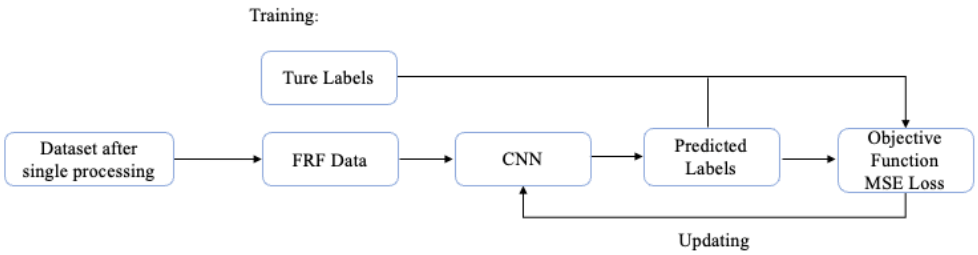


Figure 3-7. Flow chart of CNN for structural damage detection

To ensure uniformity across all damage scenarios, manual hyperparameter tuning is conducted for the specific network based on the structure in Table 3-4. A lower learning rate (0.001) stabilizes performance during convergence (Cooijmans et al., 2016). . Shuffling training data every epoch prevents bias and aids fast convergence, particularly given the orderly range of damage severities in the data. The proposed method's flowchart is outlined in Figure 3-7, depicting the systematic approach to ensure consistent and optimized training for all damage scenarios.

3.6.2. Fine-tuning based on CNN

For a deep CNN, increased network depth intensifies training duration due to heightened complexity and growing parameters. Transfer Learning (TL) offers an effective solution by reducing dependence on vast datasets. Transfer learning (TL) leverages features from a source domain and applies them to a target domain with comparable data types. While some research applies TL successfully in vision-based SHM for structural damage detection and classification tasks (Azimi and Pekcan 2019; Gao and Mosalam 2018), there is a gap in TL application for regression tasks in vibration-based SHM. In vibration-based SHM TL, pre-trained network feature layers remain fixed, optimizing regression layers with reduced computational costs.

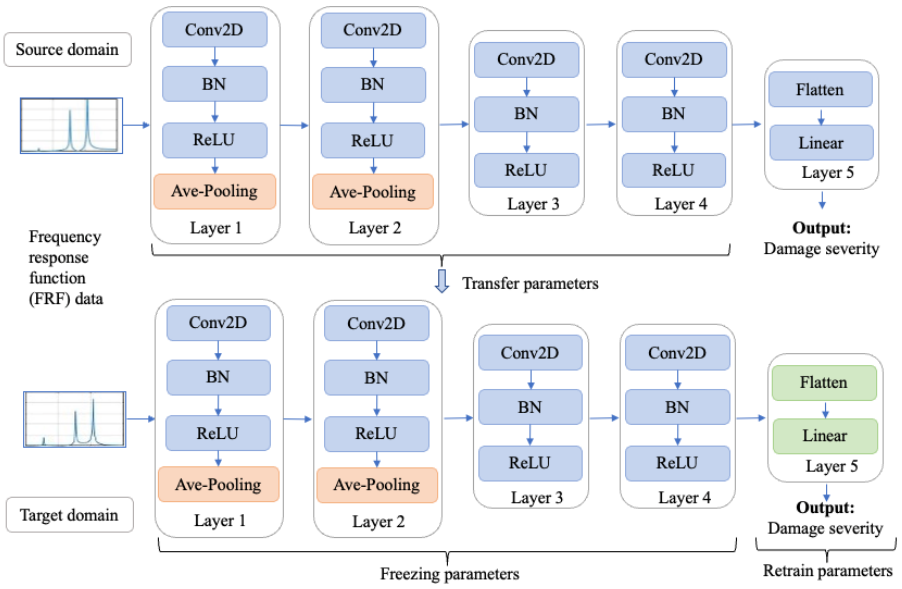


Figure 3-8. CNN configurations for the pre-trained and fine-tuned models

Figure 3-8 depicts the configurations of CNN models for both pre-trained and fine-tuned models. In typical TL, FC layers from pre-trained networks are replaced with new layers for regression tasks. In this study, FC layer structure remains consistent with the pre-trained CNN, but the weight of each neuron on the FC layer is updated based on new data. During training, the neuron weights generated from convolutional layers are fixed, and 'Flatten' output serves as training data to modify linear layer weights for predicting damage levels. In vibration-based SHM, common TL applications involve fine-tuning pre-trained DL models from image datasets. This proposed method advances upon that, replacing the dataset with FRFs from structural responses in different domains. As discussed in the pervious Section, FRFs from different damage scenarios generated in the numerical model act as the source domain for CNN training, and FRFs from the numerical or physical model with uncertainty serve as the target domain to update FC layer weights.

Loss Function

In this study, the mean square errors (*MSE*) between real and predicted values are utilised to define the loss function of CNN, illustrated in Figure 3-7. The expression of *MSE* is given as below.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \quad \text{Eq. 3-1}$$

where Y_i *and* $\hat{Y}_i$ are the true and predicted values of the i^{th} sample among the total n samples. Regularly, *MSE* is used to evaluate the fitting performance of the regression problems.

Damage evaluation indices

In this study, two regression metrics are used to indicate the performance of the prediction of FT for TL. These indices are formed from the true labelled values $\{\boldsymbol{Y}_i, i = 1,2,\dots,n\}$ and predicted values $\{\hat{\boldsymbol{Y}}_i, i = 1,2,\dots,n\}$, where n is the number of samples. Each value is a vector that has three elements to represent the damage severities of three floors in this study. Symmetric mean absolute percentage error (SMAPE) is a regression accuracy measurement based on the relative errors as below.

$$SMAPE = \frac{100\%}{n}\sum_{t=1}^{n}\frac{|\hat{Y}_i - Y_i|}{(|\hat{Y}_i| + |Y_i|)/2} \quad \text{Eq. 3-2}$$

The absolute difference between Y_i and $\hat{Y}_i$ is defined as half the sum of their absolute value of them. The value of these calculations is summed for every fitted value i and over again by the sample number n, to find the final mean value. The symmetric absolute percentage error (SAPE) is proposed to calculate the distribution of the relative error as

$$SAPE = \frac{|\hat{Y}_i - Y_i|}{(|\hat{Y}_i| + |Y_i|)/2} \quad \text{Eq. 3-3}$$

Different from SMAPE, SAPE only requires the difference between Y_i and $\hat{Y}_i$, and it requires checking the relative error of each sample for each damage scenario instead of the mean value of the entire samples.

3.6.3. Knowledge transfer between two models with uncertainty

3.6.3.1. Data generation from two models

To explore knowledge transfer and the impact of uncertainty, two numerical models, Models 1 and 2, are utilized. Model 2, validated in Section 4.2 using experimental results, serves as the source domain, while Model 1, the target domain, incorporates modelling errors. The masses are represented by three degrees of freedom ($m_1,\ m_2, m_3$), and stiffness elements are calculated from each column in bending ($k_b = 3EI/l_c^3$). Detailed properties of Models 1 and 2 are in Table 3-5, and their natural frequencies are listed in Table 3-6. Differences in natural frequencies between Models 1 and 2 (5.72%, 5.96%, and 5.47%) indicate modelling errors in Model 1, considered as the uncertainty in this study.

Table 3-5.Properties of Models 1 and 2

Model	**DOF d_t**	**Beam geometry $\{l_c, w_c, t_c\}$ (mm)**	**Mass geometry $\{m_1, m_2, m_3\}$ (kg)**	**Elastic modulus E (GPa)**	**Stiffness $\{k_1, k_2, k_3\}$ (kN/m)**
Model 2	3	{915, 50, 3}	{5.20, 4.99, 4.92}	20	{44.539,23.126,23.709}
Model 1	3	{915, 50, 3}	{5.24, 5.07,4.99}	20	{49.968,26.012,27.291}

The efficacy of proposed method is demonstrated through single and double damage detection. Stiffness reduction on columns simulates structural damage, generating datasets for different damage scenarios using two models. Acceleration responses are measured on each floor subjected to impact forces. These FRF data serve as CNN input, with CNN output indicating damage location and severity in the label space. For instance, output {0 0.2 0.1} signifies damage on the second and third floors, with severities of 20% and 10%, respectively.

Table 3-6. Natural frequencies of Models 1 and 2.

Model	**First mode**		**Second mode**		**Third mode**	
	Frequency (Hz)	Frequency difference	Frequency (Hz)	Frequency difference	Frequency (Hz)	Frequency difference
Model 2	5.59	-	15.27	-	20.64	-
Model 1	5.91	5.72%	16.18	5.96%	21.77	5.47%

3.6.3.2. Effects of impact locations

Model 2 is used to study the effect of impact location. Figure 3-9 displays FRF plots for the structure under various impact locations. Each subfigure is labeled with two digits: the first indicating the floor number, and the second representing the impact location. For instance, "23" corresponds to the second floor's FRF with an impact on the third floor. Rows represent FRFs of three floors with a single impact location, and columns show FRFs of one floor with impacts at different levels. A row with FRFs of three floors and the same impact location forms a CNN input sample. To assess the impact location effect, four damage scenarios (D0, D1, D2, D3) from Section 4.2 are simulated, generating 81 samples per floor. The total training samples for each damage scenario amount to 243 for training the CNN model.

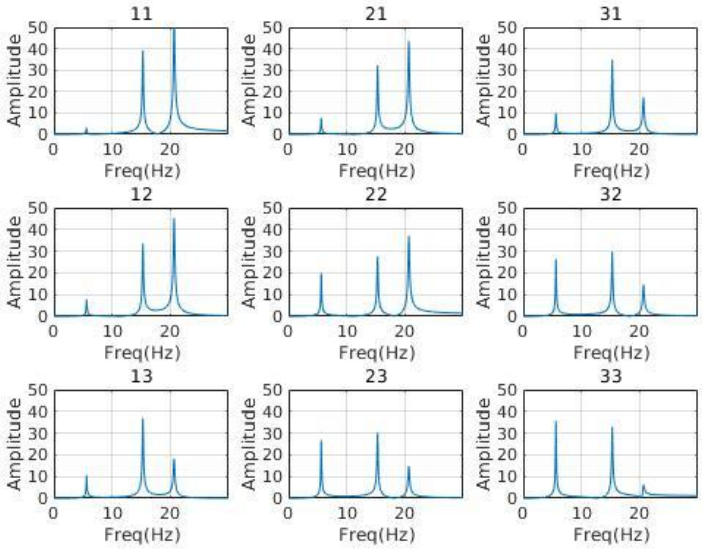


Figure 3-9. FRFs of three floors with three impact locations (intact scenarios)

The t-distribution stochastic neighbor embedding (t-SNE) method is employed for feature visualization, reducing high-dimensional data to low dimensionality (Matten and Hinton 2008). In Figure 3-10, features from FC layers depict four damage scenarios impacted by three locations (intact D0, first-floor damage D1, second-floor damage D2, and third-floor damage D3). Features are plotted

with different colors for each impact location, revealing variation with impact location. Each impact exhibits three feature clusters corresponding to damage scenarios, with features from the intact scenario concentrated at one corner. Features for each single damage scenario (0 to 30%) gradually extend from the intact scenario. In subsequent studies, the first-floor impact location is generally considered.

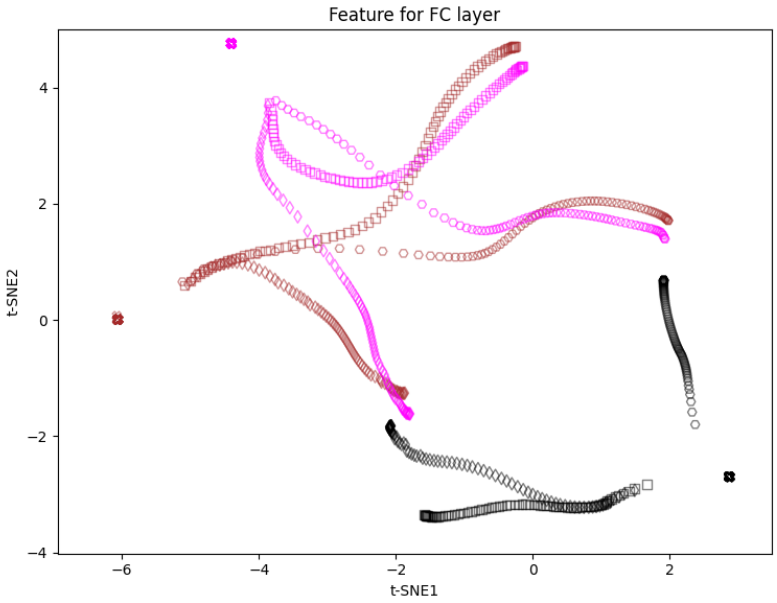


Figure 3-10.Feature visualisation through t-SNE for different damage scenarios and impact locations (Note: D0, D1, D2 and D3 represented by (×), (□), (○) and (◊) respectively; Brown, pink and black indicates the impact on the first, second and third floors respectively.)

3.6.3.3. Knowledge transfer from Model 2 to Model 1

In Section 5.3.1, Model 2 is treated as the source domain with ample labeled data, while Model 1 is the target domain with limited data. Knowledge from Model 2 is applied to Model 1 for structural damage detection. Table 3-7 displays sample numbers for training and testing in each domain, covering scenarios D0 to D6. 243 samples per scenario are generated from Model 2 as source domain 1 (SD1), totaling 1701 samples. Target domain 0 (TD0) comprises 243 samples from Model 1's intact

scenario and 81 samples from each damage scenario. The initial CNN model is trained exclusively with data from SD1. Performance is verified using testing data from TD0 (Table 3-7). The pre-trained model is then fine-tuned with limited data from TD0, and its performance is compared with the initial model using 162 samples for each scenario during testing. It is noted that the fine-tuned and testing data are from the same dataset and both are randomly selected.

Table 3-7. Sample numbers of different damage scenarios for source and target domains

		Training				**Testing**		
		Intact	Single damage 0-30%	Double damage 0-30%		Intact	Single damage 0-30%	Double damage 0-30%
Source Domain 1 (SD1)	Model 2 (CNN)	D0 243	D1 243	D4 243	Model 1 (TD0)	D0 162	D1 162	D4 162
			D2 243	D5 243			D2 162	D5 162
			D3 243	D6 243			D3 162	D6 162
Target domain 0 (TD0)	Model 1 (FT)	D0 81	D1 81	D4 81	Model 1 (TD0)	D0 162	D1 162	D4 162
			D2 81	D5 81			D2 162	D5 162
			D3 81	D6 81			D3 162	D6 162

Figure 3-11 illustrates the t-SNE feature for first-floor impact, exploring feature changes with damage location and severity, including both single and double damage scenarios. The 2D plot reveals overlapping features for double damage (D4, D5, D6) with corresponding single damage scenarios, indicating a relationship. D4 overlaps with D1 and D2, and D5 and D6 overlap with their respective single damage scenarios. Lighter blue and darker blue labels denote double damage scenarios with trends aligning with D2. Additionally, the feature trends of D1 and D4 intersect precisely at 10% and 20% damage of D1, emphasizing distinct features for double and single damage scenarios and an evident relationship between them.

It is assumed that the feature distributions of data from the source and target domains are

identical. Figure 3-12 presents a t-SNE visualization of features from these domains using the initial CNN model. The differing feature distributions highlight the discrepancies between the source and target domains, showcasing the influence of modeling errors on the mapping of damage classes in the label space. This discrepancy could lead to misclassification and poor regression performance. The study employs a large number of source domain samples for initial CNN training and a small number for fine-tuning the pre-trained model. By fine-tuning with limited target domain data, the weights in the fully connected layer are adjusted, allowing accurate classification and regression of target data while utilizing knowledge from the source domain.

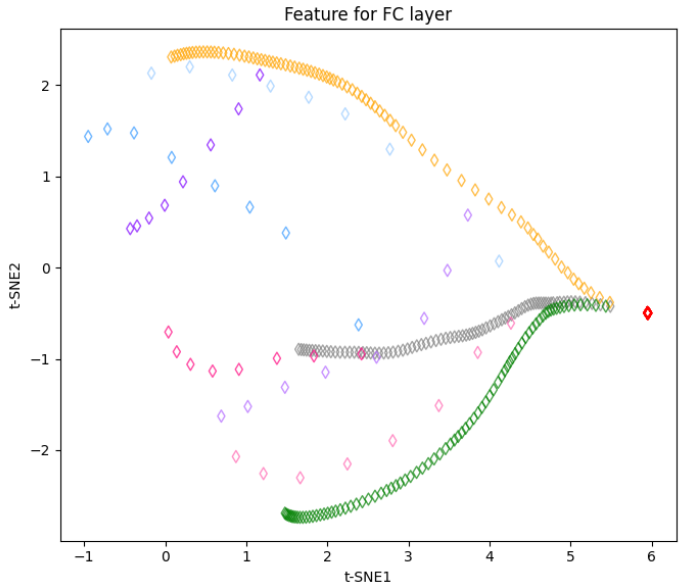


Figure 3-11. t-SNE visualization of features extracted from the data of the source domain (Note: Red is for D0, Grey, yellow and green are for D1, D2 and D3; Light blue is the double damage scenario 20%, 0-30% damage on the first and second floors; Blue is the double damage scenario 30%, 0-30% damage on the first and second floors; Light purple is the double damage scenario 20%, 0-30% damage on the second and third floors; purple is the double damage scenario 30%, 0-30% damage on the second and third floors; Light pink is the double damage scenario 20%, 0-30% damage on the first and third floors; purple is the double damage scenario 30%, 0-30% damage on the first and third floors.)

Figure 3-13 displays the MSE loss of the validation dataset for each epoch during fine-tuning on the target model data. The MSE loss of the proposed method is significantly smaller than that of the

initial model, indicating that fine-tuning leads to faster learning and more accurate results. The initial MSE value for the fine-tuned model is lower than the initial CNN model, demonstrating that target domain data efficiently readjust the weights on the FC layer, progressively reducing MSE loss to accommodate variations between Models 1 and 2. The reduced MSE loss signifies improved adaptation to structural features, suggesting more suitable weights for effective structural damage detection.

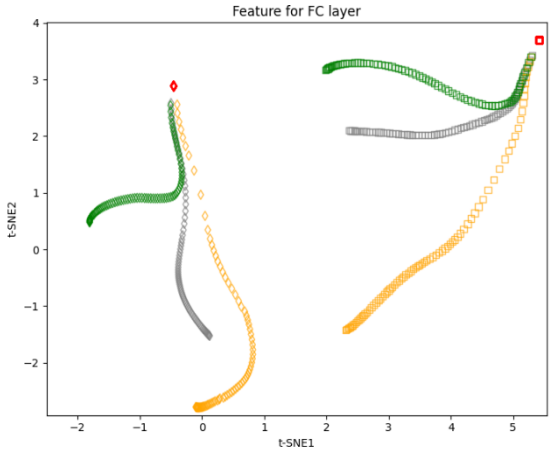


Figure 3-12. t-SNE visualization of features using FT for structures with modelling errors (The data for the source domain 1 and the target domain 0 are denoted by (◊) and (□) respectively. Red, grey, yellow and green are for D0, D1, D2 and D3 respectively.)

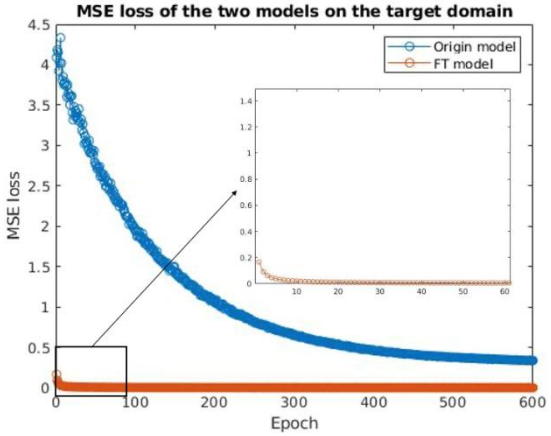


Figure 3-13. The MSE loss of the validation dataset for each epoch with the initial and FT models

Figure 3-14. indicates the model regression performance, in which the top figure shows that the initial CNN model predicted results and the bottom figure is for predicted results by the FT model.

The results show that the FT model outperforms the initial model with the performance increased from R^2=0.2075 to 0.9922. The regression model effectively captures the linear relationship between predicted and true values, indicating the proposed method's accurate prediction of damage severities even in the presence of model errors.

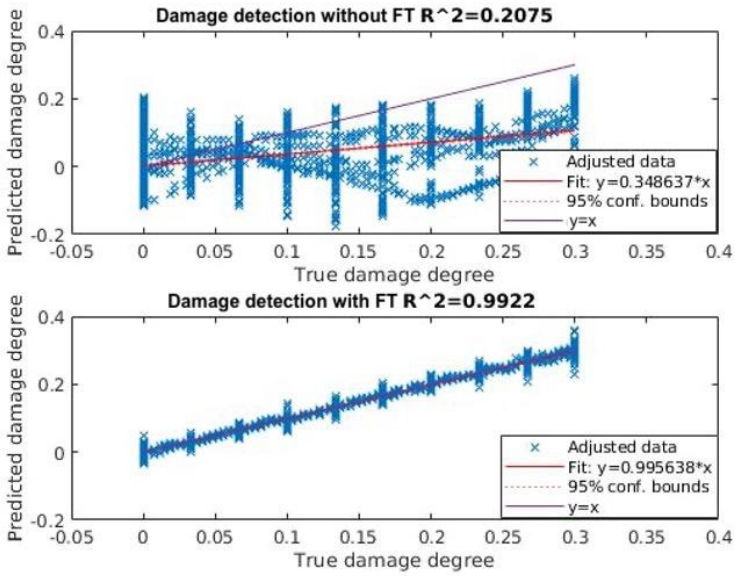
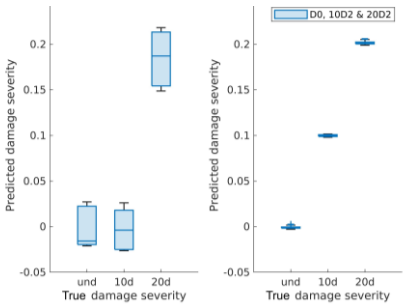
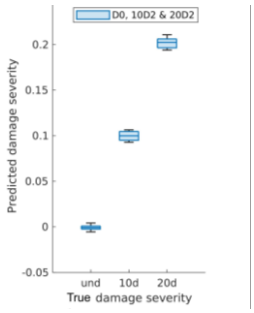


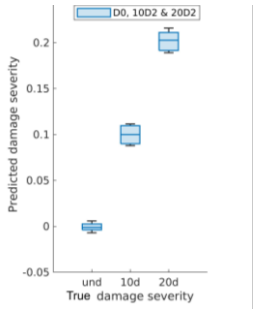
Figure 3-14. Regression analysis of testing results without or with FT



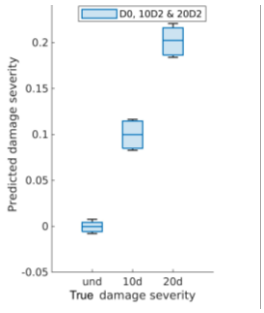
(a)Target domain 0



(b)Target domain 0 with 3% noise



Target domain 0 with 5% noise



Target domain 0 with 10% nois

Figure 3-15. Box charts for predicted and true damage severities with or without FT.

Noise effects

To further validate the robustness of the proposed method. The 3%, 5% and 10% Gaussian white noise are added the target structure responses to TD0, $X_{nosie} = \sim X + \mathcal{N}(\mu, \sigma)$, where X is the noise-free data. $\mathcal{N}(\mu, \sigma)$ is the standard normal distribution for Gaussian process. μ, σ are the mean value and standard deviation of the noise. In this study, μ is equal 0, and σ is equal to 0.03, 0.05 and 0.1 respectively. Figure 3-15 illustrates the comparison of single damage predicted results using the CNN model with and without fine-tuning. The figure displays the median of predicted data, with the box indicating the interquartile range (IQR) between the 25th and 75th quantiles. Whisker lines, extending from the box, depict the expected data variation (1.5 times the IQR). The upper and lower whisker boundaries cover 95% of data points within the predicted damage range for each severity in the experimental model. Outliers, exceeding the expected data variation, are represented by black lines. Figure 3-15 (a) plot the predicted results for TD0 comparing without FT and with FT. The damage severities can also be used to identify damage locations by examining their maximum values. The proposed method that includes FT significantly decreases the range of predicted damage severity

compared to the CNN model without FT. The medians for all three damage severities closely align with their true values, indicating high accuracy and overall performance. Figure 3-15(b) presents results for TD0 with varies noise levels, showcasing substantial improvement of proposed method in damage predictions. The comparison between the distributions in Figure 3-15(a) and 3-15(b) highlights that the outlier for TD0 with noise is larger than that for TD0 without noise, emphasizing the slightly deceased predicted accuracy with an increase level of random noise from the target domain.

3.6.4. Knowledge transfer from numerical to experimental models

This section is to study the knowledge transfer from numerical simulation to real structure. The numerical model is considered as the source domain and the real structure is the target domain. Various combinations of damage scenarios are constructed to create distinct domains, and the outcomes are utilized to evaluate the proposed method's performance.

3.6.4.1. Analysis of single damage scenarios

In this section the single damage scenarios are studied. FT technology is employed to adjust the fully connected layer weights for damage classification and regression in structural damage detection. The source domain involves four scenarios generated using Model 2: D0 (intact), D1 (0-30% damage on 1st floor), D2 (0-30% damage on 2nd floor), and D3 (0-30% damage on 3rd floor). In the target domain, limited measurement data include the intact structure D0, 10D2 (10% damage on 2nd floor), and 20D2 (20% damage on 2nd floor). Figure 3-16 displays the FRF plot for various damage scenarios from the experimental model.

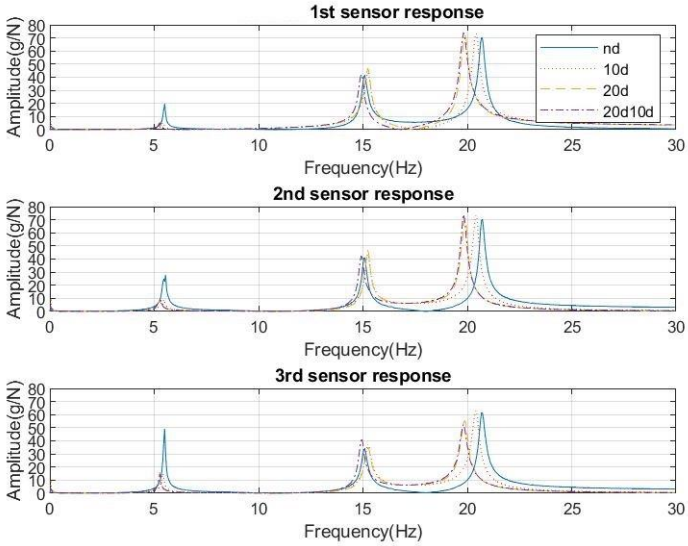


Figure 3-16. FRFs of three floors with different damage scenarios from the experimental model (nd is intact structure; 10d is single damage scenario 10% on the second floor; 20d is single damage scenario 20% on the second floor; 20d10d is the double damage scenario 20% and 10% damage on the second and third floors)

Table 3-8. Lumped mass model, single damage case study: sample numbers in each damage scenario for each domain.

		Training		**Testing**		
		Intact	Single damage	Target domain	Intact	Single damage
Source Domain 1 (SD1)	Numerical Model (CNN)	D0 243	D1 243 D2 243 D3 243	20D2 54		
Target domain 1 (TD1)	Experimental Model (FT)	D0 108	10D2 108 20D2 108	Experimental Model (TD1)	D0 54	10D2 54 20D2 54
Target domain 2a (TD2a)	Experimental Model (FT)	D0 54	10D2 54 20D2 54	Experimental Model (TD2)	D0 54	10D2 54 20D2 54
Target domain 2b (TD2b)	Experimental Model (FT)	D0 27	10D2 27	Experimental Model (TD2)	D0 54	10D2 54

In real-world scenarios, limited labelled data in the target structure pose challenges for identifying structural damage using restricted measurements. To address this, two target domains, TD1 and TD2 (a and b), are established with varying amounts of labelled data from the same database.

Details of source domain 1 (SD1), target domain 1 (TD1), and target domain 2 (TD2) are outlined in Table 3-8. TD1 involves 108 samples for each scenario, while TD2a has 54 samples and TD2b has 27 sample for each scenario. Each label is represented as a 1×3 vector, denoting damage locations and severities across three floors.

Figure 3-11 illustrates the expected feature mapping for the source domain, displaying intact and various single damage scenarios. The target domain, featuring intact (D0), 10% damage on the second floor (10D2), and 20% damage on the second floor (20D2) as per Table 3-8, is presented in Figure 3-17. The feature mapping encompasses six clusters, with three representing scenarios from the source domain and the others from the target domain. Notably, the feature spaces from the source domain are distinct from those corresponding to scenarios in the target domain. This discrepancy arises from differences between the numerical and experimental models.

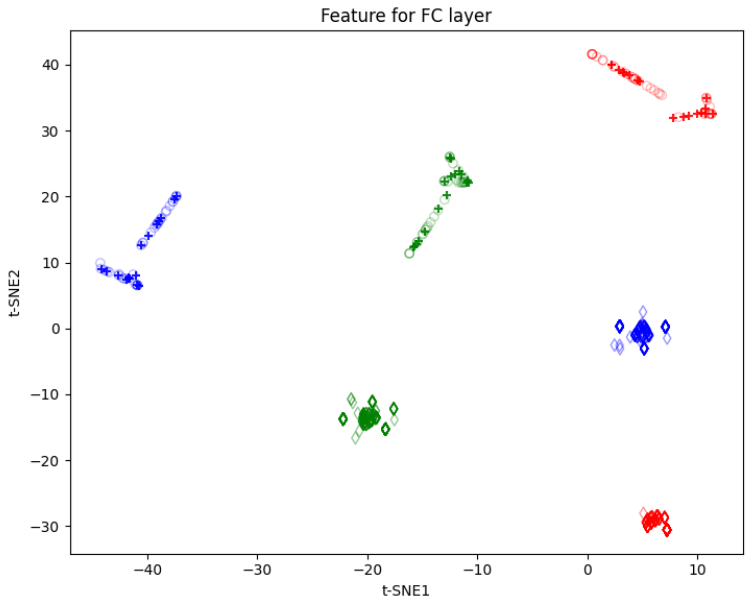
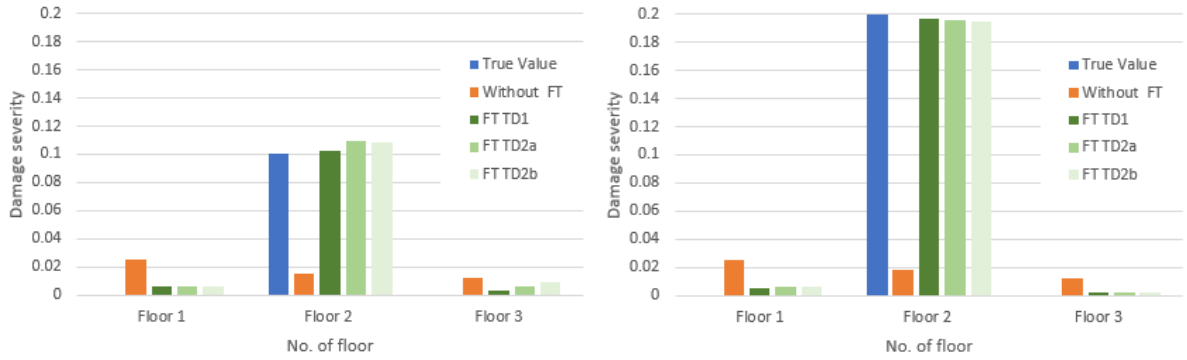


Figure 3-17. Feature visualization of FT the weight through t-SNE for single damage scenarios. ((◊), (+) and (○) represent the features of the training data from the source domain, the fine-tuning data and testing data from the target domain respectively. Blue, green and red are the intact, 10% damage and 20% damage on the second floor.)

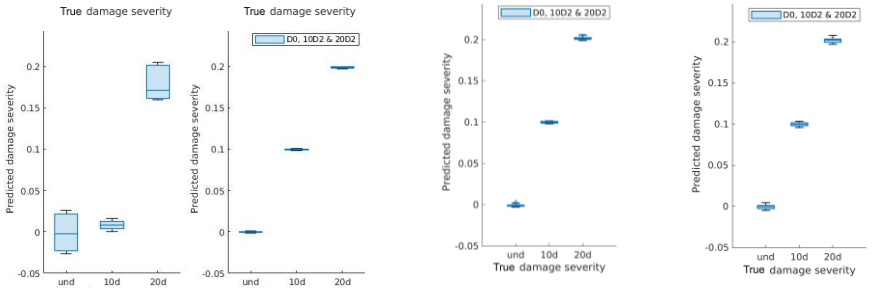
In this study, the FT-based TL is utilized for structural damage detection. FT adjusts the weights

of the pre-trained CNN based on target domain features and their respective labels. Fig 18 shows that the average damage identification results of TD3 and TD4 from without FT and with FT. The proposed method is very accurate in identifying damage compared to without FT. Figure 3-19 portrays the predicted results for TD1. The proposed method, involving FT, significantly narrows the predicted damage severity range compared to the CNN model without FT. The medians for all three damage severities closely align with their true values, indicating high accuracy and overall performance. Figure 3-19 (b) and 3- 19 (c)presents results for TD2a and TD2b, showcasing substantial improvement of proposed method in damage predictions. The comparison between the distributions in Figure 3-19 (a), 3-19 (b) and 3-19 (c) highlights that the outlier for TD2a and TD2b is larger than that for TD1, emphasizing the enhanced predicted accuracy with an increased number of FT samples from the target domain. There is also the comparison between MSE value with the number frozen layers for the testing data of TD1, TD2a and TD2b.



(a) 10% damage 2nd floor (b) 20% damage 2nd floor

Figure 3-18. Damage identification results of TD1, TD2a and TD2b from without FT and with FT

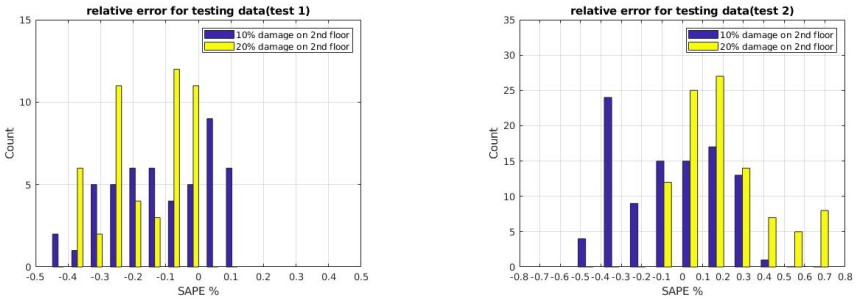


Target domain 1 Target domain 2a Target domain 2b

Figure 3-19. Box charts for predicted and true damage severities with or without FT

SAPE is introduced as relative errors to evaluate the overall predicted results.

Figure 3-20 shows the overall percentage of errors in predicted damage severities. In Figure 3-20(a), the 20% damage is much more sensitive and it is much close to zero. In Figure 3-20(b), both two damage severities are accurately predicted by the proposed method and the relative errors are between -0.5% to 0.7%. Compared with the TD1 case, the TD2 case uses half the number of experimental data for fine-tuning, and its SAPE value is slightly higher than that of TD1. This is due to the use of less data to adjust the network weight for the FC layer. Although there are half of the fine-tuning samples for TD2, the relative error still remains at a low level.



Target domain 1 (b) Target domain 2

Figure 3-20. Histograms of SAPE (%) on the predicted damage severities

3.6.4.2. Analysis of double damage scenarios

The TL is to transfer knowledge across similar structures using the pre-trained network. In this section, the proposed method is extended to the double damage scenarios. Table 3-9 lists the sample

numbers of each damage scenario for source and target domains. Those scenarios D0~D3 on the source domain and D0, 10D2 and 20D2 on the target domain are the same as that in Section 5.4.1. The double damage scenarios are introduced in this section. In the source domain, three double damage scenarios are added: D4 is the double damage scenario with 0-30% damage on the first and second floors, D5 is the double damage scenario with 0-30% damage on the second and third floors, and D6 is the double damage scenario with 0-30% damage on the first and third floors. In the target domain, the double damage scenario is 20% and 10% damage on the second and third floors (2010D5).

Table 3-9. Sample numbers of each damage scenario for source and target domains

		Training				Testing			
		Intact	Single damage		Double damage	Target domain	Intact	Single damage	Double damage
Source Domain 2 (SD2)	Numerical Model (CNN)	D0 243	D1 243 D2 243 D3 243	D4 243 D5 243 D6 243	Experimental Model (TD3)	D0 108		10D2 108 2010D5 108 20D2 108	
Target domain 3 (TD3)	Experimental Model (FT)	D0 54	10D2 54 20D2 54	2010 D5 54	Experimental Model (TD3)	D0 108	10D2 108 20D2 108		2010D5 108
Target domain 4 (TD4)	Experimental Model (FT)	D0 54	10D2 54 20D2 54	2010 D5 0	Experimental Model (TD4)	D0 0	10D2 0 20D2 0		2010D5 108

From Table 3-9, the source domain is SD2 and there are two target domains: TD3 and TD4. For SD2, there are 243 samples for each scenario. For TD3, there are 54 samples for each scenario to be used for fine-tuning and it has samples from four scenarios: D0, 10D2, 20D2 and 2010D5. The scenarios of TD3 have a large overlap with that of SD2, which can help with feature transfer for regression. TD4 has samples of intact and single damage scenarios for fine-tuning. Compared with SD2, TD3 and TD4 have the small sample size.

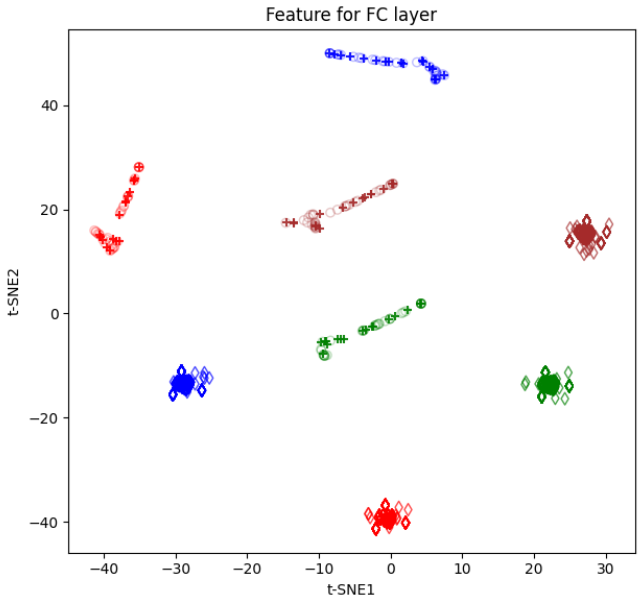


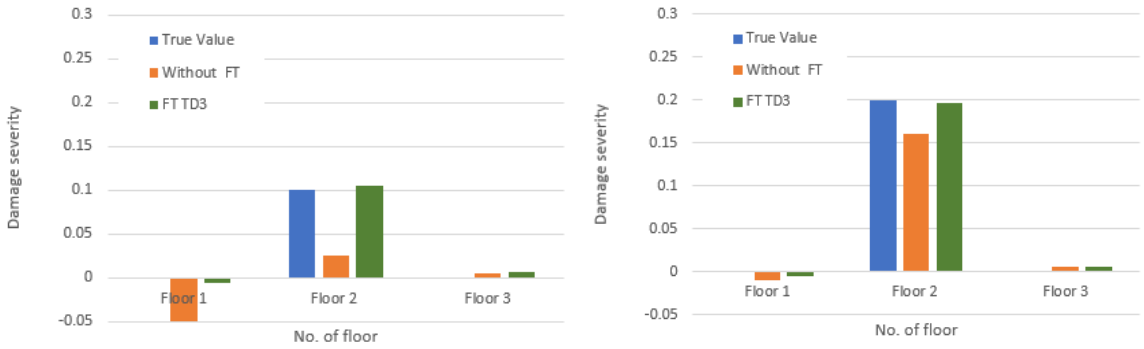
Figure 3-21.Feature visualization through t-SNE for different damage scenarios. (The data for SD2, TD3 and its testing are represented by (◊), (+) and (○) respectively. Blue, brown, red and green are D0, D1/10D2, D2/20D2 and D5/2010D5.)

Figure 3-21 shows the latent feature space for both training and testing data for different damage scenarios before the FT is conducted. The corresponding scenarios between source and target domains are compared. From Figure 3-21, there are eight clusters corresponding to four scenarios from SD2 and another four scenarios from TD3. The t-SNE feature spaces between SD2 and TD3 have a substantial discrepancy. The results show that the distributions of features from SD2 and TD3 are not aligned and there is a domain shift.

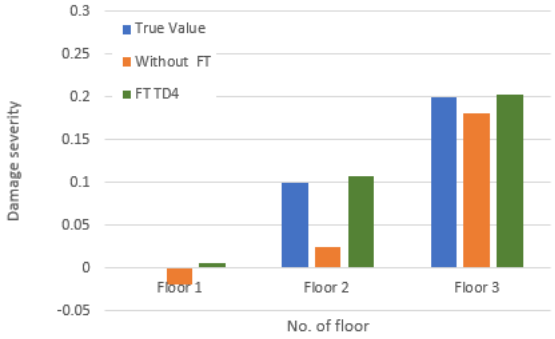
FT is conducted to readjust the weight for structural damage detection using the data from TD3 and TD4. It is noted that in TD4, there is no observation for the double damage scenario in the FT process, and the double damage data is only used for testing. Figure 3-22 shows that the average damage identification results of TD3 and TD4 from without FT and with FT. The proposed method is very accurate in identifying damage compared to without FT. It effectively identifies the location of the damage and accurately assesses the severity of the damage at the damaged location, which is

very close to the true value.

Figure 3-23 shows the predicted and true damage severities for TD3 and TD4 with or without FT. From Figure 3-23(a), the medium values of the intact and single damage scenarios without FT are -0.025, 0.025, 0.175 for D0, 10D2, 20D2, respectively, and the medium value of the double damage are -0.025 and 0.200 for 2010D5. By the proposed method, the predicted results are 0.000, 0.100, 0.200 for D0, 10D2, and 20D2, respectively, and the predicted results of the double damage are 0.100 and 0.200. The predicted results by the proposed method are much closer to the true value. The results show that the proposed method significantly improves the accuracy of the predicted damage severity. In Figure 3-23(b), the predicted damage severities for 2010D5 without FT are 0.200 and -0.025, respectively and the predicted values by the proposed method are 0.220 and -0.075. The results show that the predicted results for double damage scenarios have not been improved. This is mainly due to observations of such scenarios missing in the fine-tuning training data.

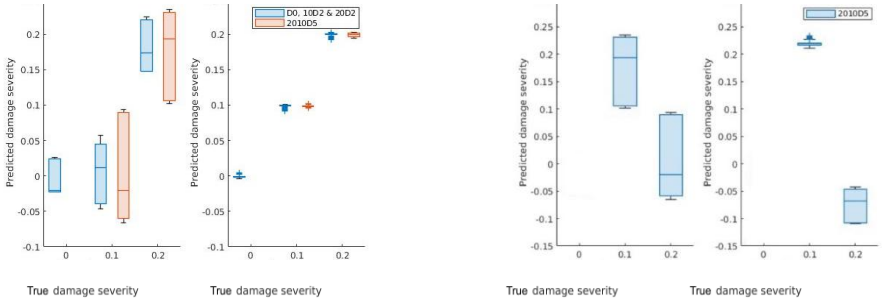


(a)10% damage 2nd floor (b) 20% damage 2nd floor



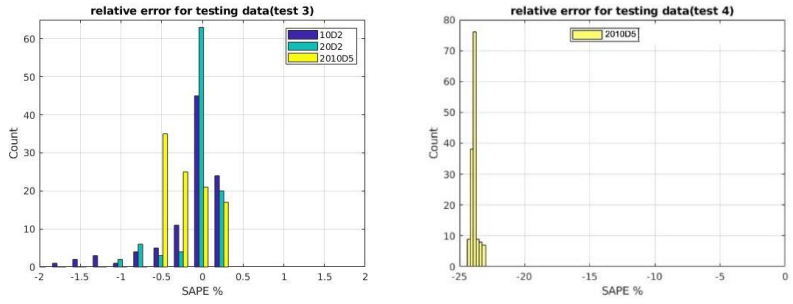
(c) 10% damage 2nd floor & 20% damage 3rd floor

Figure 3-22. Damage identification results of TD3 and TD4 from without FT and with FT



(a) Target domain 3 (b) Target domain 4

Figure 3-23. Box chart for predicted and true damage severities without or with FT.



(a) Target domain 3 (b) Target domain 4

Figure 3-24. Histogram of SAPE (%) for the predicted damage severities

Figure 3-24 shows the histogram of SAPE for predicted damage severities. Figure 3-24(a) illustrates the FT model is more sensitive to larger damage severities i.e. 20D2 and 2010D5 as they are in the ranges of -1.00% to 0.25% and -0.50 to 0.25%, respectively. In Figure 3-24(b), the distribution of the SAPE values is decentralised with the range between -25% to -20% for the damage scenario 2010D5. This reflects that the FT model on TD4 is a failure to transfer the feature of double

damage scenarios without the same scenarios for FT.

3.7. Summary

A transfer learning-based approach has been designed for structural damage detection using limited measurement data, with its performance validated through numerical and experimental results. The following conclusions can be drawn:

A TL-based method for structural damage detection is proposed, utilizing the feature extraction capabilities of CNNs. The CNN is first pre-trained on a large dataset generated through numerical simulations and later fine-tuned with limited real-world measurement data to adapt the network weights. This approach transfers labeled information from a numerical (source) structure to a real-world (target) structure. Although feature distribution differences between the structures may exist, the method helps mitigate the challenge of limited damage data for a specific structure.

By leveraging a 2D CNN, the proposed method focuses on extracting damage-relevant features from FRF data. The FT-based TL framework adjusts the FC layer weights for each specific target domain. This approach demonstrates a notable improvement in structural damage detection when compared to the initial CNN model.

The proposed method needs a small number of labelled data from the target structure to fine-tune the CNN model. The large amount of source domain datasets has been used to avoid the risk of overfitting. In practice, the required structural damage scenarios are often not available before they are identified in the real world, which makes it difficult to obtain the damage-labelled data for detecting the damage scenarios. Further study will be carried out to overcome this limitation.

Chapter 4. Structural damage detection based on transmissibility functions with unsupervised domain adaptation

4.1. Overview

Deep learning (DL) has been used for structural damage detection by training neural network models with a large amount of data. These trained models perform well when the test samples are from the data with an identical distribution of the training data. The distributions are always different due to numerical modelling errors and operational environmental varieties, and the data for damage scenarios is difficult to be obtained in practice. A novel method based on the joint maximum discrepancy and adversarial discriminative domain adaptation (JMDAD) for structural damage detection without labelled measurement data has been developed to address the above issues. To reduce the influence of external excitations in practice, the transmissibility function of measured structural responses is used for structural damage detection. The proposed network includes a feature generator, two classifiers and one discriminator. Firstly, the feature generator and domain discriminator are to extract features and merge their distributions at the domain level to overcome the issue of insufficient data from the target real structure. Secondly, the generator and two classifiers are optimised to align their distributions at the class level using the classification discrepancy between the two classifiers. As a result, the damage-sensitive features are extracted and aligned to eliminate modelling errors and operational environmental varieties between the source and target structures. Three case studies have been conducted in this chapter, e.g., one case between two building structures with different storeys, another case between the numerical and experimental structures subjected to random and earthquake excitations and the application of Canton Tower. The results show that the

proposed method is much robust to the measurement noise and operational environment and the structural damage is identified accurately without labelled measurement data from the target structure in operational environments.

4.2. State of the art

The existing civil infrastructure deteriorates due to ageing, environmental and operational loading. To avoid compromising operation or structural failure, vibration-based methods are used to detect structural damage or deterioration. Vibration based structural damage detection methods are often categorised into model-based and data-driven approaches (Avci et al. 2021). The model-based approach is based on accurate numerical modelling and the damage identified results are significantly affected by uncertainties, such as modelling errors, operational and environmental variations, and measurement noise (Hou and Xia, 2021). The data-driven approach, especially the deep learning (DL) based approach, has recently attracted the interest of researchers and engineers (Abdeljaber et al., 2017; Azimi et al., 2020; Sun et al., 2020). The DL-based methods can automatically extract structural damage features from the input response data through stacked blocks of neural network layers to identify structural damage. The performance of existing vibration-based damage detection methods to real world structures are significantly affected by the varying environmental and operational conditions (Peng et al., 2022). Structural dynamic responses are related not only to structural parameters but also to external excitations. For complex and large structures in practice, the external excitation is unknown. The influence of unknown external excitations on structural damage identification needs to be reduced. Transmissibility function (TF) is a mathematical representation of the output-to-output relationship, and it is widely used for damage detection due to its sensitivity to

the damage and robustness to external excitations. Yan et al. (2019) presented a literature review of transmissibility-based system identification for structural health monitoring. Liu et al. (2023) used TFs for data-driven identification of structural damage under unknown seismic excitations. Mei et al. (2025) a TF-based online damage detection through streaming variational inference-empowered Bayesian nonparametric clustering. To avoid measuring the input and assuming specific models for the input, TF is used as input data in this study.

Various types of DL-based methods have been proposed for structural damage detection. Lin et al. (2017) used the deep convolutional neural network (CNN) to extract the damage feature and identify the damage location from the low-level sensor data with the measurement noise. Pathirage et al. (2018) utilised a deep autoencoder for effective feature learning to build the nonlinear mapping between the input modal information and output structural stiffness parameters for structural damage identification. The effectiveness of the DL-based structural damage detection methods is mainly based on two assumptions (Zhao et al., 2021): 1) the data being predicted (test data) have the same distribution of input and its labels as that of the data used to build the DL model (training data); 2) a large amount of labelled training data are available. In practice, it is very difficult or even impossible to obtain the damage state data from the real structure. The training data are normally generated using the numerical model of the real structure and different damage scenarios are simulated using the numerical model (Chen et al. 2023; Jiao et al. 2020). Due to the uncertainty, such as modelling errors, the operational and environmental varieties and measurement noise, the generated data using the numerical model are not consistent with the measurement data from the real structure (Lin et al. 2022). The distribution of the data from the numerical model is different with that of measurements on real

structures. When the trained DL models from the numerical model is applied to the real structure, its performance is decreased significantly (Zhang et al. 2022). It is a big challenge to detect structural damage without a large amount of labelled measurement data.

Domain adaptation (DA) has been used to address the above issue for structural damage detection recently. Gardner et al. (2020; 2021) demonstrated the application of three domain adaptation techniques for population-based structural health monitoring, namely transfer component analysis, joint domain adaptation and adaptation regularisation-based transfer learning. Yano et al. (2023) also conducted a comparison for structural damage detection of the bridges using transfer component analysis (TCA), joint distribution adaptation (JDA) and maximum independent domain adaptation (MIDA) based transfer learning methods. Knowledge transfer-based DA methods highly depend on massive high-quality data to align the data distributions in the high dimension feature spaces between two domains. Xu and Noh (2021) presented a physics-informed domain adversarial network for knowledge transfer between structures with different storeys for damage detection and quantification. Lin et al. (2022) presented a maximum mean discrepancy (MMD) based domain adaptation method for structural damage localization. The MMD is used to measure the discrepancy of the feature distributions between the source and target domains. The deep domain adaptation network is to extract the damage-sensitive and domain-invariant features by reducing the discrepancy. This method requires that both domains have identical damage states (label spaces) for their datasets. Wang and Xia (2022) presented a re-weighted adversarial domain adaptation method for structural damage detection with the inconsistent label spaces between the source and target domains. A weight parameter is used to improve the importance of the shared label space in the domain adaptation

process, and it needs the manual intervention to modify classification weights. A large amount of unlabelled testing data samples are required for forming the data distribution in the target domain. Li et al. (2023) developed the ensembled damage-sensitive feature method based on domain adversarial neural network (DANN) for damage identification between simulation to real structures. As the above, the existing domain adaptation methods heavily rely on the precision classifiers trained from the source domain, and its performance is reduced significantly when the domain discrepancy is large. The same number of samples for the source and target domains are also needed in the domain adaptation process. In practice, the measurement data from real structures, especially under damaged states, are limited. When dealing with the insufficient labelled data and domain variance problem, it may be restricted for employing only class-level aligning without leveraging domain-level knowledge for categories. Also deploying only domain-level alignment may cause misclassification of the learned features. The existing methods focus on the boundary of source and target domains, and the subset class information has not been considered. The features located in the class decision boundary might be non-discriminative and misclassed (Saito et al. 2018). The information on discriminatory categories needs to be retained in the domain adaptation process. A joint adversarial domain adaption approach has been proposed to align domain and class wise distributions across the source and target domains in the domain adaptation process (Li et al. 2019). A dynamic joint domain adaptation neural network has been used to extract transferable features from cross-session motor imagery classification (Hong et al. 2021).

In this chapter, a novel joint maximum classifier discrepancy and adversarial discriminative domain adaptation (JMDAD) method for structural damage detection using TFs is proposed. The

knowledge learned with the labelled data from numerical models are utilised for structural damage identification of real structures without labelled measurement data. The proposed network mainly includes one feature generator, two classifiers and one discriminator. The global domain knowledge is captured by the domain-level distribution alignment to minimise the domain discrepancy due to the modelling errors, and the local discriminative information is explored by the class-level distribution alignment to minimise its conditional discrepancy due to uncertainty in the numerical modelling and real structure. The domain and class level distributions of structural damage features are aligned simultaneously across source and target domains in the adversarial learning process. Two label classifiers are to detect the discrepancy between the source and target samples at the class level. Three case studies, e.g. knowledge transfer from one numerical model to another one with different sizes, from the numerical to experimental structures and the application of Canton tower, are conducted to verify the performance of the proposed method. The results show that the proposed method is effective and accurate to identify structural damage with limited unlabelled datasets and the performance outperforms existing deep domain adaptation methods.

4.3. Methodology

4.3.1. Problem definition

In this study, the real structure and its numerical model are denoted as the target and source domains D_T and D_S respectively. The data for different damage scenarios can be generated using the numerical model. The source domain D_S with n_s labelled samples is defined as $D_S = \{(\boldsymbol{x}_{si}, y_{si}) | \, \boldsymbol{x}_{si} \in \boldsymbol{X}_S, y_{si} \in Y_S, i = 1,2,\dots,n_s\}$, where the input data is $\boldsymbol{X}_S = \{\boldsymbol{x}_{si}, i = 1,2,\dots n_s\}$ and the output label is $Y_S = \{y_{si}, i = 1,2,\dots n_s\}$. The measurement data from the real structure is

limited, especially the data under structural damage state. The unlabelled target domain with n_t samples can be defined as $D_T = \{(\boldsymbol{x}_{ti})|\boldsymbol{x}_{ti} \in \boldsymbol{X}_T, i = 1,2, \dots n_t\}$. There are two different types of distributions over the input and output space $X \times Y$ for the source and target domains, one is marginal distribution and another one is conditional distribution. The marginal (global) distributions for the source and target domains are defined as *P(X)* and *Q(X)* respectively, where *X* refers to a specific learning sample $X = \{\boldsymbol{x}_i | \boldsymbol{x}_i \in \boldsymbol{X}_S \, or \, \boldsymbol{X}_T, i = 1,2, \dots m\}$. The corresponding conditional (local) distributions are $P(Y|X)$ and $Q(Y|X)$ respectively. Due to the uncertainty, such as the modelling errors, the operational and environmental varieties, and the measurement noise, there is always a discrepancy between the real structure and its numerical model. Further, the distributions of the data from the source and target domains are not identical. This leads to a network pre-trained by the source domain data losing performance on the target domain data. To tackle the above problem, the unsupervised domain adaptation is used to transfer the enriched label knowledge learned from the source domain data to the unlabelled target domain data by adapting the feature representation or classifier models to reduce the distribution discrepancy between the source and target domains in this study.

4.3.2. *The proposed method*

This study is to develop a new method based on the joint maximum classifier discrepancy and adversarial discriminative domain adaptation (JMDAD) for structural damage identification. Figure 4-1 shows the framework of the proposed JMDAD method, which mainly includes three modules: data pre-processing, feature extraction and domain adaptation. The raw data includes sufficient simulated data with structural damage labels from the source domain, and the limited unlabelled

measurement data from the target domain. Firstly, the raw data is pre-processed via data filtering and data argumentation, and the transmissibility function (TF) data are obtained by the ratio between the Fourier transform of the response at each floor and that on the ground (Chesne and Deraemaeker 2013). Secondly, the 2D-CNN based feature extractor (G) with four blocks is designed to extract the damage sensitive features from the data of both the source and target domains. Lastly, the domain adaption and damage identification module include two classifiers C_1, C_2 and one discriminator D. The feature generator G is to extract the damage sensitive features from both the source and target domains. The feature generator G and domain discriminator D are to merge the marginal distributions at the domain level as the global alignment to overcome the issue of insufficient data from the real structure. The feature generator G and two classifiers C_1 and C_2 utilise the source domain data to locally align the conditional distributions of the target domain data at the class level using the classification discrepancy between these two classifiers. As a result, the damage-sensitive features are extracted and aligned to eliminate modelling errors and uncertainties between the numerical model and the real structure. The iteration of the local and global alignments enables to extract damage sensitive features for all classes to compensate the limited measurements from the real structure.

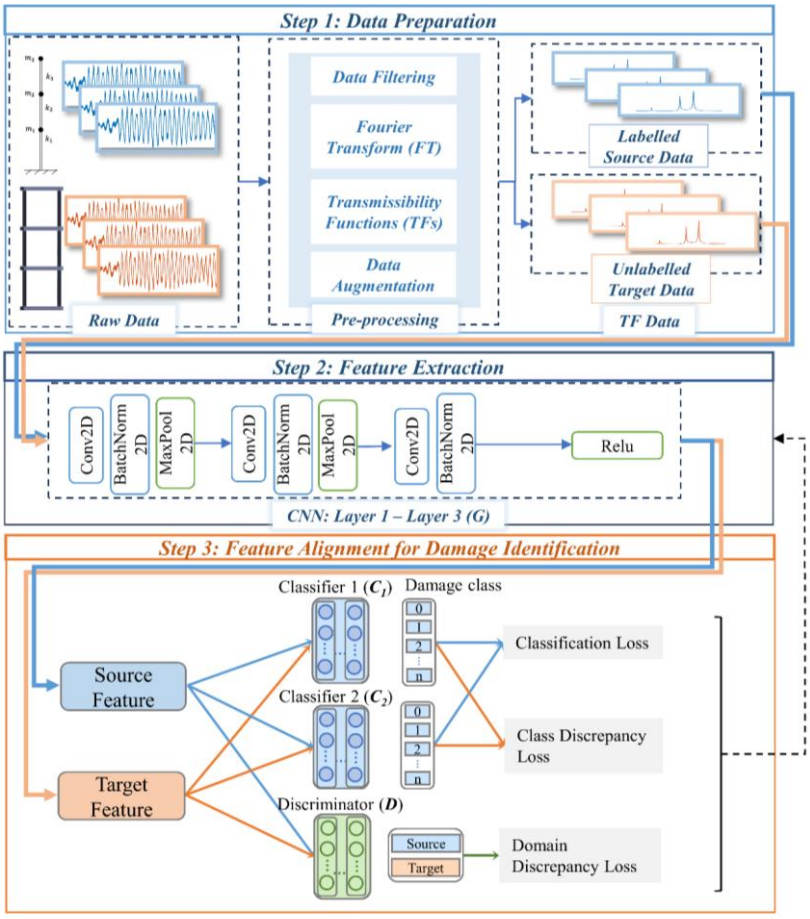
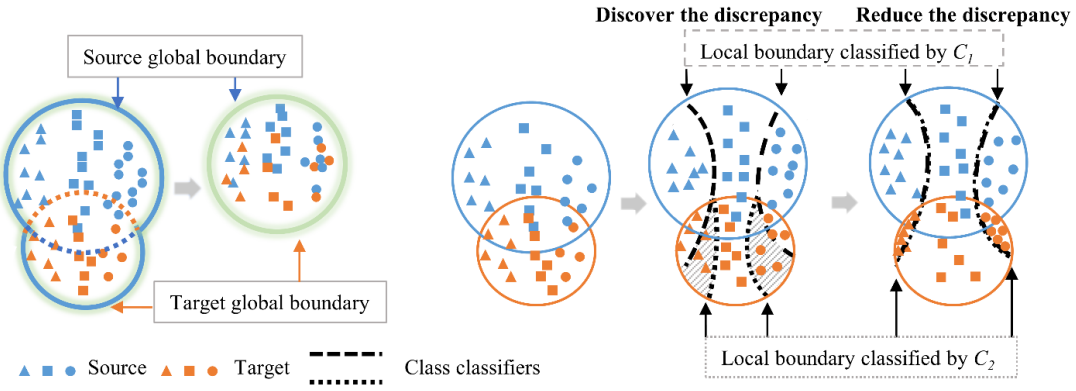


Figure 4-1. Overview of the proposed framework

In the proposed framework, the local and global alignments are integrated to reduce the discrepancy in both domain and class levels simultaneously (Ganin et al., 2016; Tzeng et al., 2017). As shown in Figure 4 2(a), the global discriminator is to align the marginal distributions (global boundary in the figure) across domains, and the classifiers trained on aligned features from the source domain data are used to improve the prediction performance in the target domain. As shown in Figure 4- 2(b), the local discriminator discovers and reduces the discrepancy of the conditional distributions for each subcategory across domains (local boundaries in the figure) by embedding the discriminative information in predictions of two classifiers which are used to assist in adversarial training to classify unlabelled target data. To summarise, the feature generator is trained to extract the target features by

minimising the discrepancy at both class and domain levels. In the adversarial training process, the generator aligns the source domain classifier boundary to generate the domain-invariance and class-separate feature from the target domain data. By applying those two adversarial networks separately, different weights for adversarial loss functions are used to adapt to various divergencies corresponding to environmental effects or modelling errors between source and target domains effectively.



(a) At domain level (b) At class level

Figure 4-2. Class and domain discrepancy alignments.

4.3.2.1. Network architecture of JMDAD

As shown in Figure 4- 3, the proposed JMDAD network architecture contains a generator G, two classifiers $C_1$and C_2, and a domain discriminator D. The network architecture and parameters are listed in Table 3-1. The feature generator G is designed as three convolutional blocks for generating the damage-sensitive feature. The first three blocks conduct three convolutional operations, including Conv2D, BachNorm2D and Maximum polling. The number of channels is set as 32, 64 and 128. With the increase in the size of the convolution block, it could compress the extracted information and get much robust sensitive feature. The first convolutional layer is designed as 2×32 with Stride 1, which allows the generator to focus on the global feature. The following layers are

decreased for extracting the local features. The small padding number allows the network to generate more feature mapping after the convolutional layer. After the features are extracted by the generator, it is then flattened at a fully connected layer block for connecting to two classifiers. Two classifiers are designed as two layers for feature dimension reduction and classification. The input of Block 1 is set as 3D with the size of $3 \times 2 \times 660$ corresponding to the size after the data pre-processing, which are corresponding to the number of data channels, and its height and width. In Table 3-1, C, H and W represent the channel numbers, height and width of each layer respectively. The final output is $m \times 1$. m is the size of outputs for damage classification.

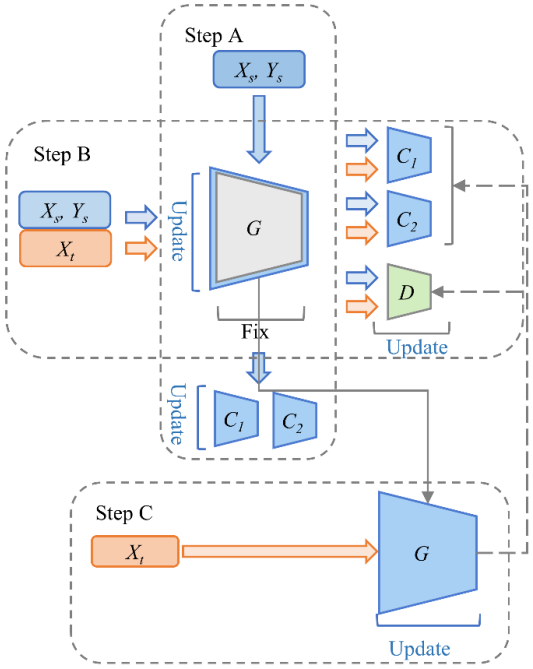


Figure 4-3. The network architecture of the proposed JMDAD

4.3.2.2. Loss Functions

In the proposed method, both marginal and conditional distributions are aligned by considering the domain and class discrepancies. The domain discrepancy exists between the marginal distributions of source and target domains, and the domain discriminator is to reduce the distance for

these distributions. The conditional distributions between two classifiers also exist between source and target domains under different conditions. The classifiers are trained to maximise their discrepancy on samples in the target domain, and the generator is to minimise it. Based on the above theory, the corresponding three loss functions in the algorithm are introduced as below.

Table 4-1.The network structure of the proposed JMDAD

Block	Input (C, H, W)	Type	Kernel num.	Kernel size	Stride	Padding	Output (C, H, W)
Block 1	(3,2,660)	Conv2D	16	(2, 32)	1	2	(32,3,317)
		BatchNorm2D	32	/			
		MaxPool2D	None	(3, 3)	2	1	
Block 2	(32,3,317)	Conv2D	16	(2, 8)	1	2	(64,3,157)
		BatchNorm2D	64	/			
		MaxPool2D		(3, 3)	2	1	
Block 3	(64,3,157)	Conv2D	16	(2, 4)	1	2	(128,6,158)
		BatchNorm2D	128				
		Relu		Activation function			
Block 4	(128,6,158)	Flatten		None			(512,1)
Classifier	(512,1)	Flatten		None			(256,1)
	(256,1)	Flatten		None			(m,1)
	(m,1)	Softmax		None			(m,1)

Classification loss

The classification loss function is to calculate the classification errors of the source domain data:

$$L_y(X_S, Y_S) = -\mathbb{E}_{(\mathbb{X}_s, y_s)\sim(X_S, Y_S)} \sum_{k=1}^{K} 1_{[k=y_s]} \log p(y|\mathbb{x}_s) \quad \text{Eq. 4-1}$$

where $1_{[k=y_s]}$ if $k = y_s$ and 0 else. k is the number of categories ($K = N_s$ for the source domain), and $p(y|\mathbb{X}_s)$ is the probability of the source domain $\mathbb{X}_s$ for each class, which is produced by the softmax functions.

Class discrepancy loss

The second loss function is the class discrepancy loss between the two classifiers. With the i^{th} input $\mathbb{X}_i$, the discrepancy between the cross-entropy outputs of two classifiers is defined as (Saito et

al. 2018):

$$d\big(p_1(y|\mathbb{X}_i), p_2(y|\mathbb{X}_i)\big) = \frac{1}{K}\sum_{k=1}^{K}\|p_{1k} - p_{2k}\| \quad \text{Eq. 4-2}$$

where $p_1(y|\mathbb{X}_i)$ and $p_2(y|\mathbb{X}_i)$ are the softmax outputs of C_1 and C_2, respectively. And the $p_{1k}$and p_{2k} represents the cross-entropy outputs for class k of two classifiers. The discrepancy loss of the target domain corresponding to the source domain is defined as the average of all samples' discrepancies (Saito et al. 2018):

$$L_{advc}(X_t) = \mathbb{E}_{\mathbb{X}_t \sim X_t}[d(p_1(y|\mathbb{X}_1), p_2(y|\mathbb{X}_2))] \quad \text{Eq. 4-3}$$

The second loss function detects the discrepancy between the source and target domains under each class, and the discrepancy of the target domain each subclass can be aligned according to the source domain.

Domain discrepancy loss

The third loss function is the domain discrepancy loss, which is classifying whether a data point belongs to the source or target domain. The domain discriminator is optimized based on the general cross-entropy loss, where the labels indicate the origin domain as (Tzeng et al. 2017):

$$L_{advd}(X_s, X_t, M_s, M_t) = -\mathbb{E}_{\mathbb{X}_s \sim X_s}\big[\log D\big(M_s(\mathbb{X}_s)\big)\big] - \mathbb{E}_{\mathbb{X}_t \sim X_t}[\log(1 - D(M_t(\mathbb{X}_t)))\ ] \quad \text{Eq. 4-4}$$

where X_s and X_t are the inputs for source and target domains, and M_s and M_t are the mappings of input samples. This loss function ensures that the adversarial discriminator treats the data from these two domains identically. The distance of the feature mapping between two entire domains can be reduced, and the generator G can generate the same feature from both domains.

Adversarial and classification weight factors

The class and domain discrepancies may have different effects on domain adaptation, which can be adjusted by the weight factor ω of their two loss functions listed in Eqs. 4-2 and 4-4. The weight is initially set as 0.5 and can be adjusted between the range [0, 1], which quantitatively evaluates the relative importance of the marginal and conditional distributions corresponding to environmental uncertainly and modelling errors.

4.3.2.3. Training process

As shown in Figure 4- 4, the network structure includes a generator and two classifiers expressed as G, C_1 and C_2 denoted as θ_f, θ_{C_1} and θ_{C_2}, respectively. Note that $C_1 \neq C_2$. Even though these two classifiers are designed with identical structures, their parameters are generated randomly at the beginning according to the labelled source domain data. The training process are mainly separated into three steps as below.

Step A

Figure 4- 4 shows Step A in the training process. The generator G and two classifiers $C_1$and C_2 are trained with the source domain data by minimising the loss function in Eq. 4-1 as,

$$\min_{G,F_1,F_2} L_y(\boldsymbol{X}_s, Y_s) \qquad \text{Eq. 4-5}$$

where $L_y(X_s, Y_s) = -\mathbb{E}_{(\mathbb{x}_s,y_s)\sim(X_S,Y_S)} \sum_{k=1}^{K} 1_{[k=y_s]} \log p(y|\mathbb{x}_s)$.

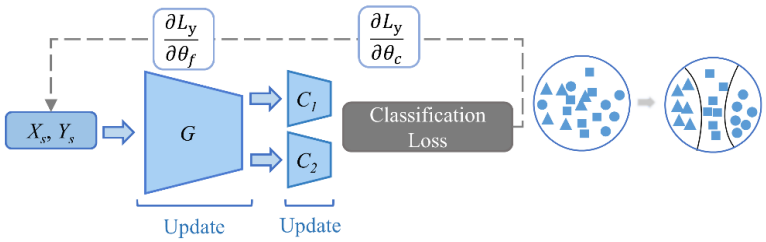


Figure 4-4. Step A in the training process

This step is to train the network using the labelled source data for structural damage detection as a classification task to identify the class of the input data X_s. This step updates the parameter weights for G (for feature extraction), C_1 and C_2 (for classification), while D is frozen. The optimisation algorithm is applied throughout the training process, with the initial learning rate of 5×10^{-5}.

Step B

In Step A, the generator G and two classifiers C_1, C_2 are trained using the labelled source domain data. These two pre-trained classifiers can classify the damage-sensitive features generated from the source data, while they still cannot identify the damage from the target domain data due to the domain discrepancy since both the features generated and classifiers fit only the source domain data.

Maximise the discrepancy between those two classifiers

In order that the target data can also be classified accurately, the source data (with labels) and target data (without labels) are input together to the feature generator. Figure 4- 5 shows Step B in the training process. In this step, the feature generator G trained in Step A is frozen and these two pre-trained classifiers are tuned by the maximum discrepancy in Eq. 4-3 under each class between two domain features. As shown in Figure 4- 5, each class of the target domain data is classified differently

by those two classifiers pre-trained in Step A. In this step, the discrepancy between two domains under each class could be caused by modelling errors.

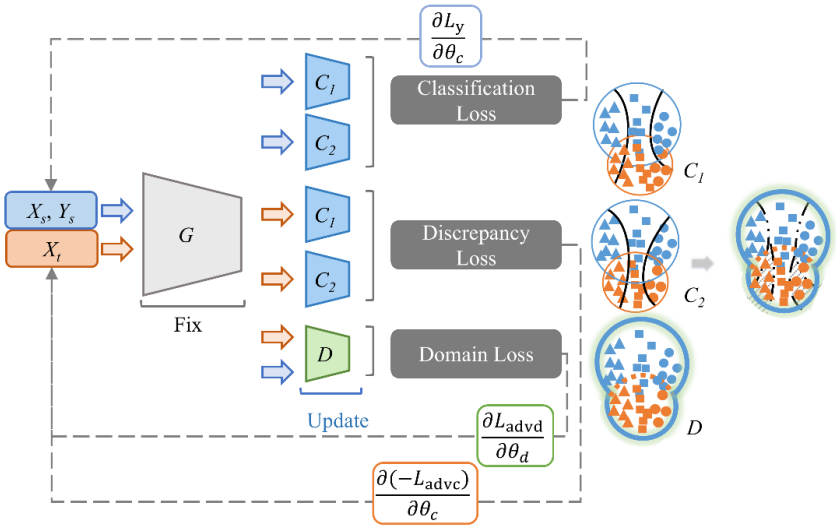


Figure 4-5. Step B in the training process

Discriminant data origin

In Step B, the discriminator D denoted as θ_d is used to detect the data whether it is from source or target domains by the discrepancy loss in Eq. 4-4. With this discriminator, the origin of the data can be identified and the same damage-sensitive features from both domains are extracted by the global distribution alignment. For structural damage detection, the discrepancy includes the environmental uncertainly or structural differences (mass, stiffness or damping) between the numerical model and real structure is adapted on the domain level.

The weights of class and domain discrepancy losses

The weighting factors of class and domain discrepancies between two domains can be adjusted to implement knowledge transfer and merge better as,

$$\min_{C_1,C_2} L_y(X_s, Y_s) - \omega L_{advc}(X_t) + \min_{D}(1-\omega) L_{advd}(X_s, X_t, M_s, M_t) \qquad \text{Eq. 4-6}$$

where ω is the weighing factor,

$$L_{advc}(X_t) = \mathbb{E}_{\mathbb{X}_t \sim X_t}[d(p_1(y|\mathbb{X}_1), p_2(y|\mathbb{X}_2))] \quad \text{Eq. 4-7}$$

$$L_{advd}(X_s, X_t, M_s, M_t) = -\mathbb{E}_{\mathbb{X}_s \sim X_s}\left[\log D\left(M_s(\mathbb{X}_s)\right)\right] - \mathbb{E}_{\mathbb{X}_t \sim X_t}[\log(1 - D(M_t(\mathbb{X}_t)))\ ] \quad \text{Eq. 4-8}$$

Step C

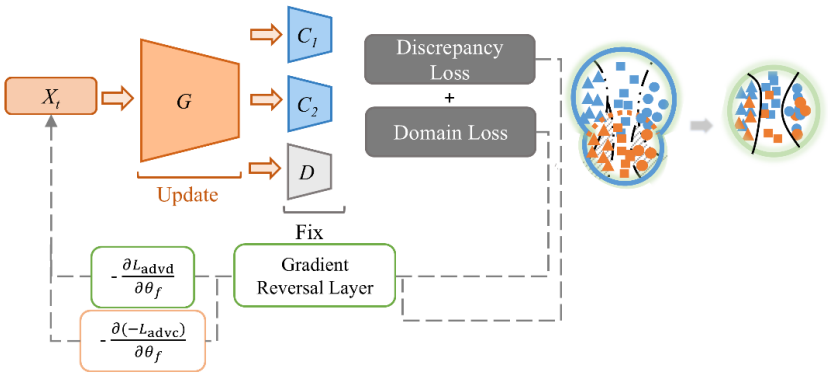


Figure 4-6. Step C in the training process

The last step in the training process is Step C, as shown in Figure 4- 6. This step is to update the feature generator G to minimize the class discrepancy loss and maximise the domain discrepancy loss. At the class level, the generator G is updated to reduce the class discrepancy loss (minimising L_{advc}) by aligning the local distributions of two domains from those two tuned classifiers in Step B (Eq. 4-7). At the domain level, the generator G is updated to align the global distributions by maximising the domain level adversarial loss (maximising L_{advd}). Thus, the generator G needs to be updated to align local and global distributions simultaneously by minimising both these discrepancy functions as,

$$\min_G(\omega L_{advc}(X_t) - (1-\omega)L_{advd}(X_s, X_t, M_s, M_t)) \quad \text{Eq. 4-9}$$

The above three steps are combined to extract the features from both domains and predict structural damage for the unlabelled target domain. For structural damage detection, the weights for class and domain levels can be optimised by the available data. The proposed training process is

summarized in Table 4-2.

4.4. Knowledge transfer between two building structures with different storeys

4.4.1. Numerical models

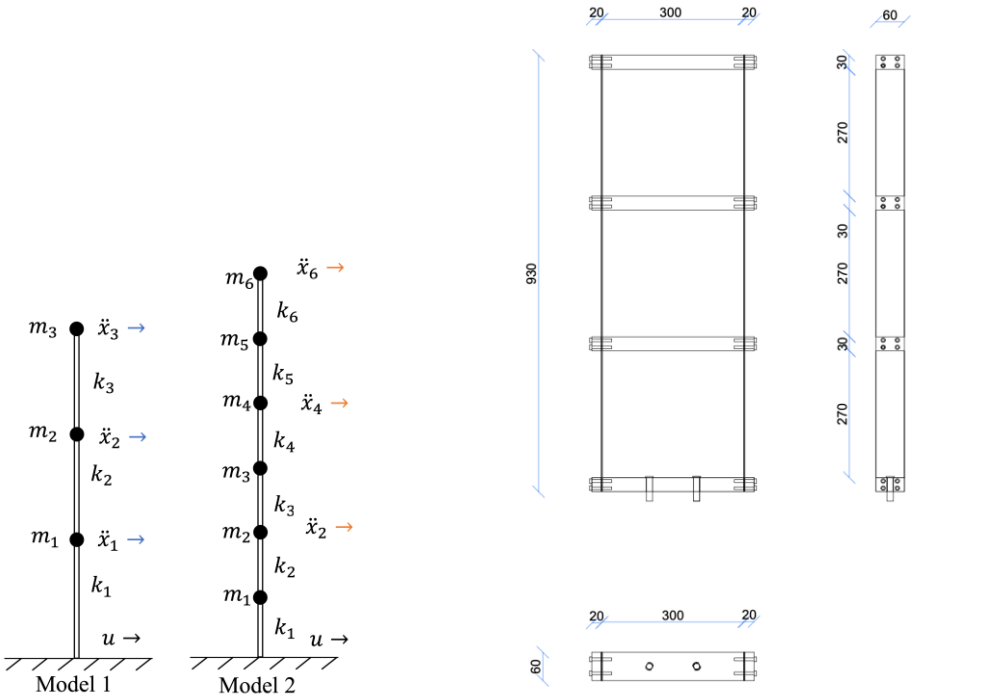
This section is transferring the knowledge from one structure to the other structure with different storeys, as shown in Figure 4- 7(a). Model 1 is a 3-story structure as the source domain with the labelled data, and Model 2 is a 6-storey structure as the target domain with limited unlabelled data. As shown in Figure 4- 7(b), a 3-storey structure mainly includes two identical columns with the height 930 mm and the cross-section 60 mm × 2 mm, and three steel floors measuring 340 mm×60 mm×30 mm. The height of each story is 300 mm. The material used to build all columns and floors is a high-strength steel with an elasticity modulus of $200GPa$ and mass density of $7.5 \times 10^3 kg/m^3$. The Rayleigh damping is adopted: $\boldsymbol{C} = \alpha \boldsymbol{K} + \beta \boldsymbol{M}$, where α =0.016 and β=0.0028 are the mass and stiffness proportional coefficient, respectively. The model parameters of these two models are listed in Table 4-3. Model 2 have the same material properties and cross-section as Model 1 but different stories.

Table 4-2. Parameters of Models 1 and 2

No. of Model	Mass $\{m_n\}$ (kg)	Stiffness $\{k_n\}$ (kN/m)	Damping $\{c_n\}$ (Ns/mm)
Model 1	{4.8,4.8,4.8}	{9.755, 9.755,9.755,}	{4.244,2.084, 2.161}
Model 2	{4.8,4.8,4.8,4.8,4.8, 4.8}	{9.755, 9.755, 9.755, 9.755,9.755, 9.755}	{4.244,4.244,2.084, 2.084, 2.161, 2.161}

Table 4-3. Summary of the training process

Algorithm: JMDAD
Note the labelled source domain $S = (\boldsymbol{x}_{si}, y_{si})_{si=1}^{n_s} \sim (D_S)^{n_s}$ and unlabeled target domain $T = (\boldsymbol{x}_{ti})_{ti=1}^{n_t} \sim (D_T)^{n_t}$, the feature generator G, two classifiers C_1 and C_2, the discriminator D, the mini-batch size and epochs. The Adam optimization algorithm is utilised the learning rate of 5×10^{-5}. **For** $\boldsymbol{p}$ **= 1 to** No. of epochs (i.e., 150): **For** $\boldsymbol{q}$ **= 1 to** No. of batch size (i.e., 256): Train G, C_1 and C_2 to classify the source data by minimizing Eq. 4-5; Train C_1 and C_2 to maximise the class-level discrepancy by minimising Eq. 4-7; and Train D to minimise the domain level discrepancy by minimising Eq. 4-8. **For** $\boldsymbol{k}$ **= 1 to** n (i.e., n =5) Train G by minimising the class level discrepancy and maximising the domain level discrepancy in Eq. 4-9. **End** **End** **End** Use the trained G, C_1 and C_2 for the classification of unlabeled target data.



(a) Models 1 and 2 (b) The three-story building structure

Figure 4-7. The multi-storey building structure and the numerical models

3.2 Data description

The building structures are subjected to ground excitations. The building structures are

simplified as the lumped mass models, as shown in Figure 4- 7(a). The equation of motion for building structures can be written by

$$\boldsymbol{M}\ddot{\boldsymbol{x}}(t) + \boldsymbol{C}\dot{\boldsymbol{x}}(t) + \boldsymbol{K}\boldsymbol{x}(t) = \boldsymbol{M}\boldsymbol{u}(t) \quad \text{Eq. 4-10}$$

where n is the degrees of freedom. $\ddot{\boldsymbol{x}}(t) \in \mathbb{R}^n$, $\dot{\boldsymbol{x}}(t) \in \mathbb{R}^n$, and $\boldsymbol{x}(t) \in \mathbb{R}^n$ are the acceleration, velocity, and displacement response vectors, respectively. $\boldsymbol{u}(t)$ is the ground excitation. $\boldsymbol{M} \in \mathbb{R}^{n\times n}, \boldsymbol{C} \in \mathbb{R}^{n\times n}, \boldsymbol{K} \in \mathbb{R}^{n\times n}$ are the mass, damping and stiffness matrices of the system respectively.

The acceleration responses of each floor at the building structure are calculated. A transmissibility function (TF) is defined as the ratio between the Fourier transform of an arbitrary acceleration response $\ddot{x}_i(t)$ and that of a reference response $\ddot{x}_r(t)$ (Yan et al., 2019).

$$T_{ir} = \frac{X_i(\omega)}{X_r(\omega)} \quad \text{Eq. 4-11}$$

where $X_i(\omega)$ (i = 1, 2, 3) and $X_r(\omega)$ donate the Fourier transform of $\ddot{x}_i(t)$ and $\ddot{x}_r(t)$, respectively. In this section, the acceleration response at the ground floor is used as the reference response. Two types of excitations are used in this study. The random excitation, e.g. white noise (WN), is to simulate the ambient excitation with the amplitude of 0.1 g and the frequency range between 1 Hz and 25 Hz. Four classics earthquake recordings, including Elcentro, Hachinohe, Kobo and Nortridge are also used to simulate earthquake excitations with the amplitude of 0.05g.

For the source domain, Model 1, e.g. the 3-storey building structure, is subjected to the ground excitation. The dynamic response at each floor and the ground floor are recorded with a sampling rate of 1000 Hz. The record length is 600s for the structure under the random excitation and it is 30 s for the structure under each earthquake excitation. 45 datasets for the structure under each type of excitations are generated. The record acceleration responses at each floor and the ground floor are

filtered to 30 Hz with the low-pass filter. Then the TF samples are obtained using Eq. 4-11. TF data for three floors are as three input channels of Block 1 in the convolutional neural network (CNN) in Table 4-1, and each channel has the size of $3 \times 2 \times 660$ as inputs to CNN. The detail information of CNN is listed in Table 4-1. For the target domain data, e.g. Model 2 for the 6-stroey building structure, the size of input data is same as that of the source domain. The data are collected from Floors 2, 4 and 6 of the 6-storey building structure, and the damage labels are unavailable during the training process.

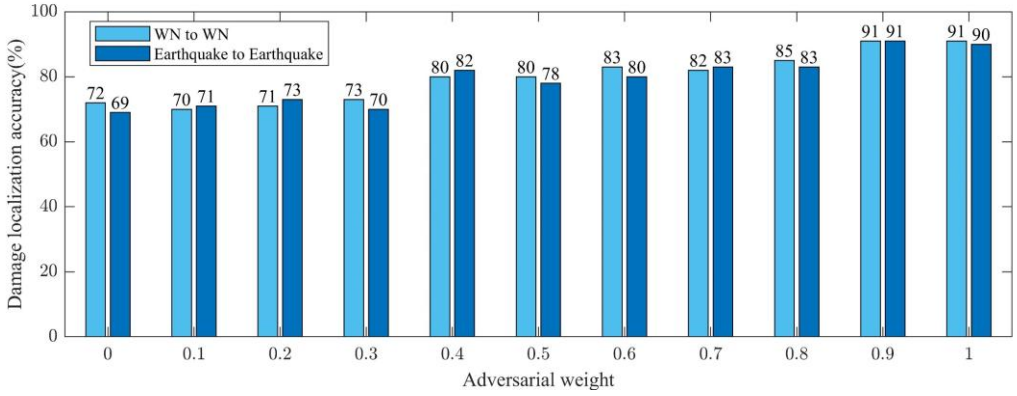
4.4.2. Single damage detection

In this study, structural damage is simulated by the sectional reduction of columns for each floor, which results in the corresponding stiffness reduction of the lumped mass model. Three damage scenarios, e.g. the damage for each floor, are simulated in this section. To verify the proposed method, three types of excitation scenarios are used e.g. the random excitation, the earthquake excitation and the mixed random and earthquake excitations. For the source domain, e.g. Model 1, the damage severities are simulated by reducing the structural stiffness between 0-30% (0%, 10%, 20% and 30%) at each floor, respectively. The damage class details are summarized in Table 4-4. A total of 180 data samples are generated, e.g. 45 samples of random excitations×4 damage scenarios, that cover all four scenarios D0, D1, D2, D3 with Labels 0, 1, 2, 3 for each type of excitation, respectively. Similarly, the unlabelled target domain data are collected from Model 2. The damage identification is conducted using the proposed method. In this study, 70% datasets are used for training and 30% of datasets are used for testing.

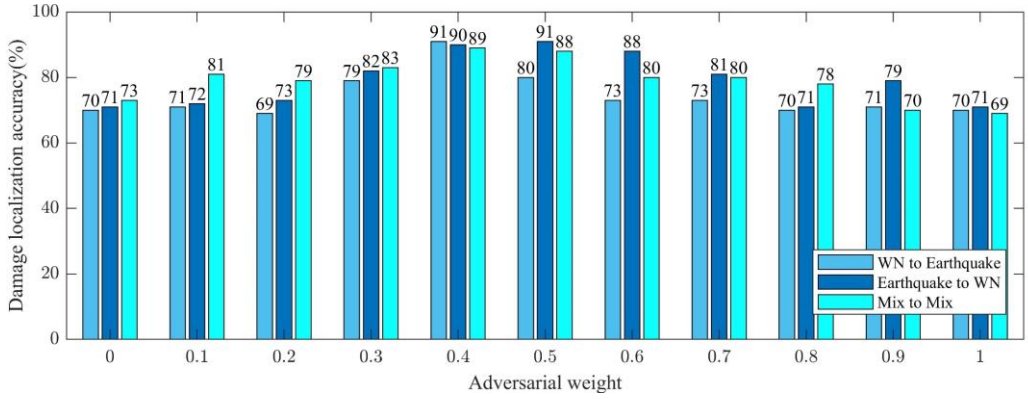
Table 4-4. Damage detection with identical datasets from source and target domains

	Model 1 **3-story model**		Model 2 **6-story model**		
Damage scenarios	Damage location	Damage severity	Damage location	Damage severity	Labels
D0	Undamaged	/	Undamaged	/	0
D1	1st floor	0-30%	1st and 2nd floor	0-30%	1
D2	2nd floor	0-30%	3rd and 4th floor	0-30%	2
D3	3rd floor	0-30%	5th and 6th floor	0-30%	3

4.4.2.1. With identical label spaces of datasets from source and target domains



(a) With the same type of excitations

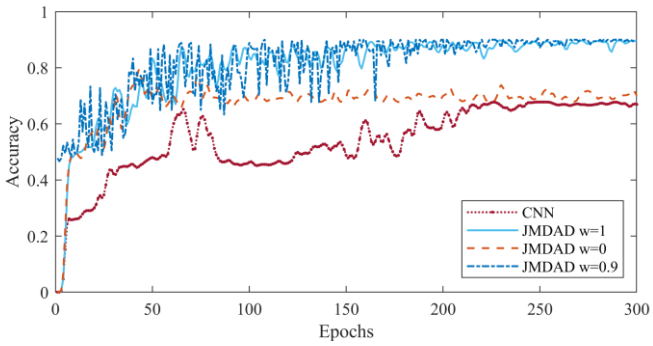


(b) With different types of excitations

Figure 4-8. Damage localization accuracy with adversarial weight ($\omega = 0\ to\ 1$)

Similar to the source domain data from Model 1, the target domain data with identical damage severities are collected from Model 2, with 180 datasets (45 excitations×(1 undamage + 3 damage scenarios)) for each type of excitations. During the training process, the target domain data are

unlabelled. Figure 4- 8(a) shows the test accuracy with different adversarial weights using the learning rate of $5e-5$ and training epochs of 300 when two structures are subjected to the same type of excitations. From Figure 4- 8(a), it reaches the highest test accuracy with 91% when the adversarial weight is 0.9. The results show that under the same type of excitations, the knowledge learned from the 3-storey building structure is transferred successfully to structural damage detection of the 6-storey building structure. Figure 4- 8(b) shows the test accuracy when two structures are subjected to different types of excitations. From Figure 4- 8(b), when the adversarial weight $\omega = 0.4$, the accuracy reaches the highest value with 89% for the mix excitation cases. As the above, the proposed method can extract the features which are invariant to the structural excitation.



%	0	1	2	3
0 (45)	91.1	8.9	0.0	0.0
1 (45)	0.0	91.1	8.9	0.0
2 (45)	0.0	0.0	91.1	8.9
3 (45)	11.1	0.0	0.0	88.9

(a) Testing accuracy over epoch (b) Confusion Matrix of Classification results

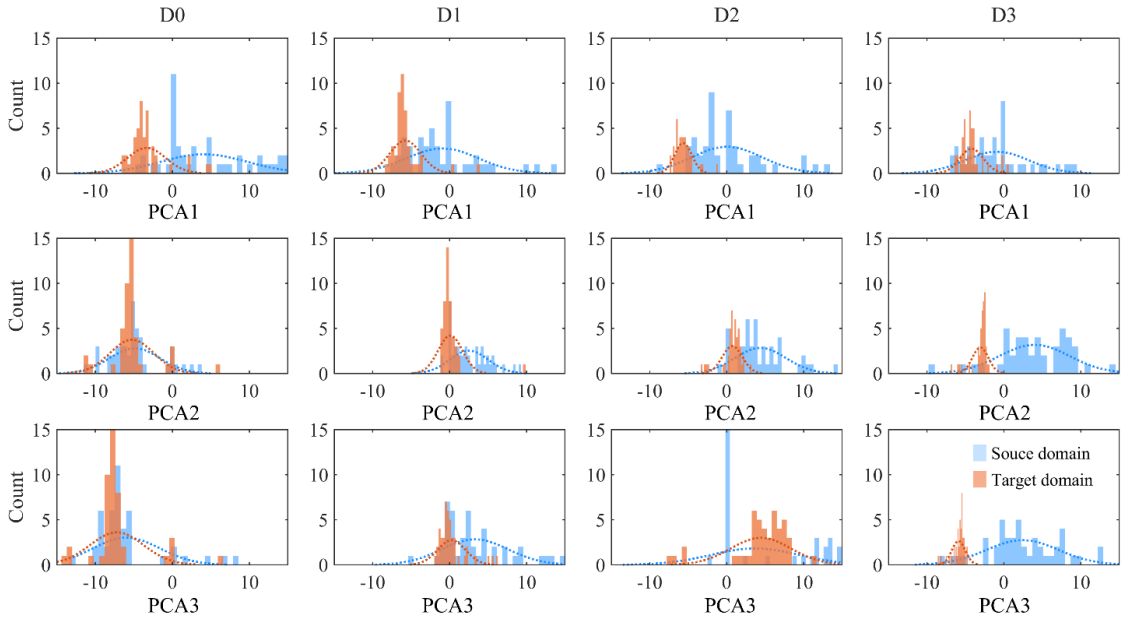
Figure 4-9. The results with identical damage datasets.

Figure 4- 9(a) shows the testing accuracy over epoch when the structures are subjected to random excitations by the proposed method and the results are compared with that by the CNN network without the domain adaption. In this figure, the red line is that by the CNN network which is trained with a learning rate of $5e-5$ using the labelled source data from Model 1. The light blue line shows the results by the proposed method with the same learning rate and the initial adversarial weight factor ω of 1 and the dash red line is for $\omega = 0$. The best performance is achieved when the weight is 0.9.

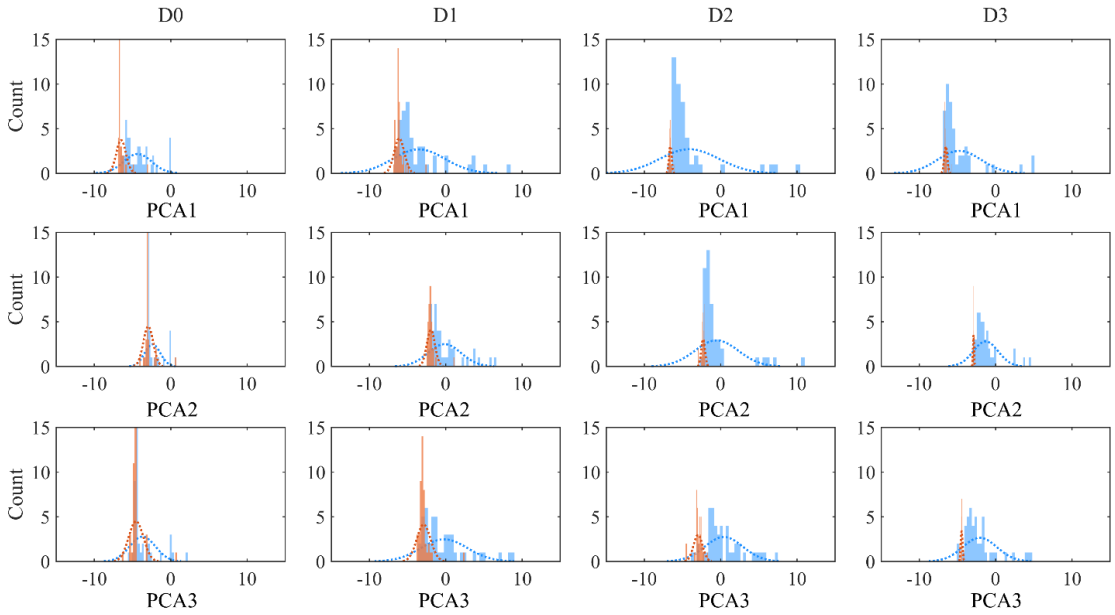
From Figure 4- 9(a), the results show that the accuracy by the proposed method is much higher that by the CNN network. Figure 4- 9(b) shows the damage classification results of the target domain by the proposed method. From Figure 4- 9(b), the overall classification accuracy by the proposed method is 91%, which is significantly improved compared 67% by the CNN network. The accuracies for Classes 0, 1, 2, 3 are 91%, 91%, 91% and 89% respectively.

To further understand the transfer learning process, the principal component analysis (PCA) is used to extract the most representative components from features of the source and target domains. The first three principal components are considered, e.g. PCA1, PCA2 and PCA3. Figure 4- 10 shows the distributions of each principal component before and after domain adaptation. The x-axis is the value of PCA, and the y-axis is the class (label) for each damage scenario. The legend (the orange and blue colour) represents the number of distributions for each class. The extreme value of the distribution for all classes constitutes the margin for the distribution of each PCA. Figure 4- 10(a) shows the distributions for each PCA component of features from source and target domains by CNN. The results show that there is a clear discrepancy between the distributions of features from source and target domain data. By the CNN network directly, the distributions of each PCA for source and target domains are not consistently mapped, and the PCA components for each class are not the same. The distribution for each class is unaligned between source and target domains. Figure 4- 10(b) shows the distribution of each PCA component after domain adaptation using the proposed method. The results show that the distribution margin of each PCA component for each class between the source and target domain is well aligned and this is corresponding to the domain discrepancy. The PCA values for both source and target domains in the x-axis are approximate the same. This global

alignment for the domain discrepancy is attributed to the discriminator D which the damage sensitivities features could be efficiently extracted and identified to overcome the effect of modelling errors. Besides, the conditional distributions for each class between source and target domains, which is corresponding to the class discrepancy, are also updated, and aligned well after the knowledge transfer and feature merger, achieved by two classifiers $C_1 and\ C_2$. The local alignment for the class discrepancy is to identify the damage-sensitive feature eliminating the effects due to modelling errors and environmental varieties.



(a) Distribution of PCA components with the CNN network



(b) Distribution of PCA components with the proposed method

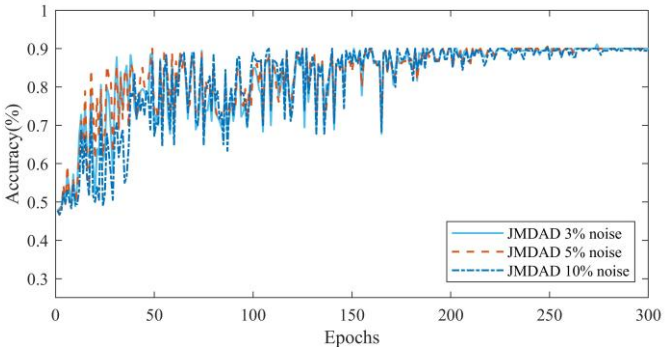
Figure 4-10. Distributions of PCA components with identical datasets

4.4.2.2. With limited noise datasets from the target domain

In practice, the damage cases are rare occurred in real structures. The damaged scenario data from the target domain are very limited and there is the measurement noise. In this study, the source domain data for all damage scenarios with damage severities between 0-30% are generated from Model 1. As listed in Table 4-5, only the data for the undamaged scenario and the 30% damage scenario of each floor are available from the target domain, which correspond to four classes with Labels 0, 1, 2 and 3 respectively. The 3%, 5% and 10% Gaussian white noise are added the target structure responses to simulate the measurement data $X_{noise} \sim X + \mathcal{N}(\mu, \sigma)$, where X is the noise-free data. $\mathcal{N}(\mu, \sigma)$ is the standard normal distribution for Gaussian process. μ, σ are the mean value and standard deviation of the noise. In this study, μ is equal 0, and σ is equal to 0.03, 0.05 and 0.1 respectively. There are a total of 180 data samples (45 random excitations×(1 undamage + 3 damage scenarios)) for each type of excitation.

Table 4-5. Limited damage datasets from the target domain Model 2

Damage scenarios /Classes	Damage location	Damage severity	Labels
D0	Undamaged	--	0
D1	1st and 2nd floor	30%	1
D2	3rd and 4th floor	30%	2
D3	5th and 6th floor	30%	3



%	0	1	2	3
0 (45)	82.2	8.9	0	8.9
1 (45)	8.9	91.1	0	0
2 (45)	8.9	0	91.1	0
3 (45)	0	0	8.9	91.1

(a) Testing accuracy over epochs (b) Confusion Matrix of Classification results

Figure 4-11. The classification results with limited damage datasets

Figure 4- 11 shows the accuracy and the classification results with limited damage datasets with different noise levels by the proposed method. The light blue, orange, and blue lines in Figure 4- 11(a) are the results with 3%, 5% and 10% measurement noise. From Figure 4- 11(a), there are some fluctuations in the beginning of the testing accuracy curve, and they approach the same accuracy after 250 epochs. The results show that the accuracy is not affected by the measurement noise. Figure 4-11(b) shows the classification results by the proposed method. From Figure 4- 11(b), the classification accuracies with measurement noise are 82.2% or over. The results show that the proposed method is much robust to the measurement noise, and those two classifiers and the discriminator work together to extract the damage sensitive feature and identify the damage accurately with limited damage datasets from the target domain.

4.4.2.3. With class imbalanced datasets from source and target domains

In the previous studies, all class datasets have the identical sizes except damage severities are limited. In practice, the data related the damage scenarios are limited and the datasets from source and target domains are unbalanced. This section is to study the structural damage detection with imbalanced datasets.

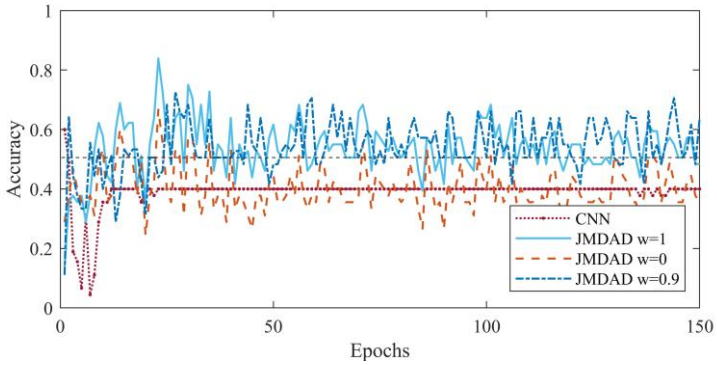


Figure 4-12. Testing accuracy over epoch using CNN and the proposed JMDAD method

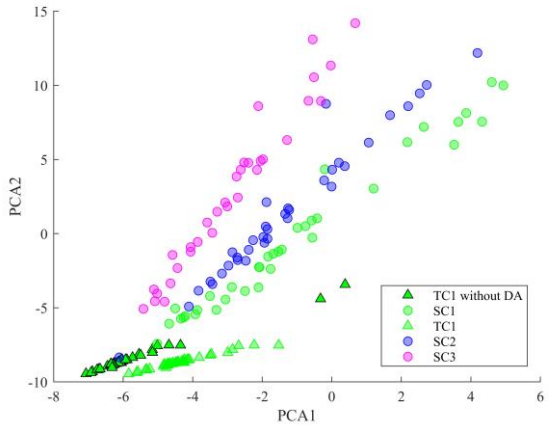


Figure 4-13． Feature visualization with PCA using JMDAD with imbalanced datasets

The source domain data are the same as that in Sections 3.3.1 and 3.3.2 for four damage classes with all the damage severities from 0 to 30%. The target domain only contains one damage scenario D1 (damage on the 1st and 2nd floors). There are 45 samples from the target domain with random excitations. Figure 4- 12 shows the test accuracy with epoch using the CNN network and the proposed method with different weights. From Figure 4- 12, the accuracies by the proposed method are much higher than that by the CNN network. When the adversarial weight factor $\omega = 0.9$, the classification

accuracy is 61.1%. Figure 4- 13 shows that the PCA visualization of features using the proposed method. From the figure, the features extracted from different classes in the target domain are clearly separated and classified by the proposed method.

4.5. Knowledge transfer from numerical to experimental structures

This section is to study the knowledge learned from the numerical model to the real structure. The numerical model is defined as the source domain and the real structure is related to the target domain. The numerical model is not calibrated using experimental results and the discrepancy between the numerical and experimental models is to simulate modelling errors. One challenge in transferring the knowledge learned from the numerical model to the real structure is the effect of the operational environment, especially the excitation. Another challenge is the limited dataset available from the real structure, in particular very limited or no data from damaged scenarios. This causes different excitations are encompassed to both numerical and experimental structures, and the target domain has much less data than the source domain for both the samples and classes. In this study, the proposed JMDAD method is used to identify the damage of the real structure under varies excitations with limited data by using the knowledge learned from the numerical model.

4.5.1. Experimental setup

To validate the proposed method for structural damage detection in the real structure, a 3-storey steel building structure was manufactured and installed on the shake table at UTS Tech Lab. The experimental setup is shown in Figure 4- 14. The structure is made of two steel columns and four mass blocks, the bottom mass block was fixed on a multi-axis shake table by two bolts. The detail dimensions of the structure are listed in Figure 4- 7(b). The four accelerometers are installed on each floor and the surface of the shaking table to monitor the acceleration responses of the ground floor.

The remote lasers are installed beside the shake table to measure the structural displacement responses. The shaking table is used to generate one-axis seismic ground motions. For each damage scenario, the WN and four earthquake excitations as described in Section 3.2 are simulated using the shaking table and dynamic responses of the building structure are measured by four accelerometers. The acceleration responses at each floor and ground of the building structure are recorded with the sampling frequency 1000 Hz. The record length for each WN test is 600 s and the record length for each earthquake excitation test is 60 s. The TF data can be obtained by Eq. 4-11. As the samples for the building structure under earthquake excitations are small, a dense sliding window with a small striding to augment the excitation response data under the earthquake excitation. The damage is introduced by inflicting two symmetrical cuts of the columns on each floor with 9 mm cuts width of the columns (30% damage), respectively. Undamaged and three damage scenarios are simulated with the corresponding stiffness reduction 30%. 45 samples (13 excitation samples×(undamaged, 1st floor damage) + 9 excitation samples×2nd floor damage+10 excitation samples×1st & 2nd floor damage) are obtained from the experimental tests for random and earthquake excitations, respectively.

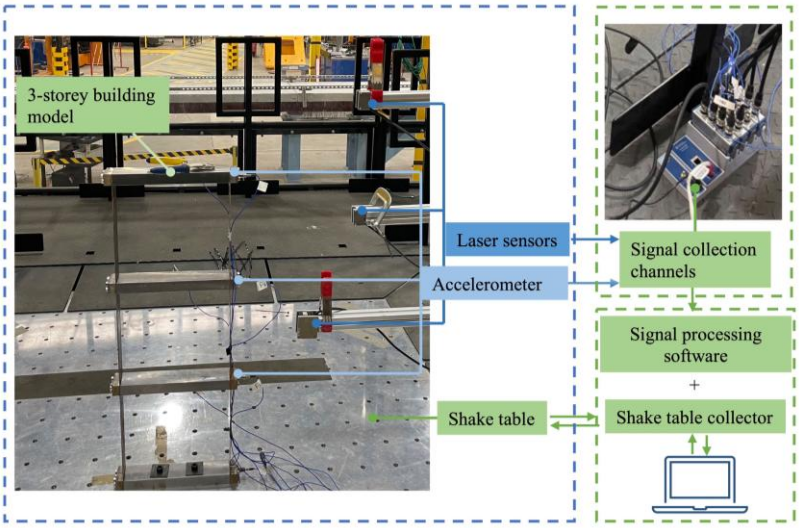


Figure 4-14. Experimental setup

4.5.2. Damage Detection with Imbalance Class datasets

As the same as Section 3.3.1, the numerical model (Model 1) is used as the source domain. Both

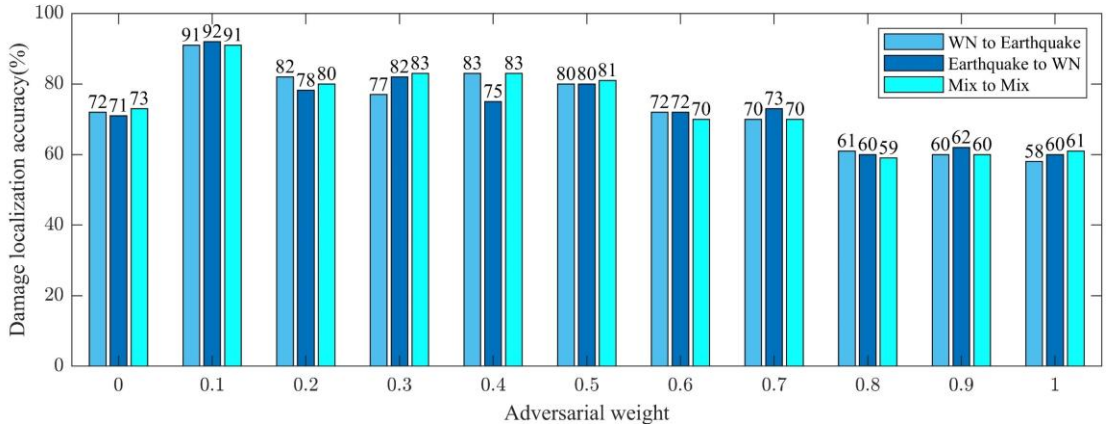
the random and earthquake excitations are applied at the bottom of structure, and 45 datasets are generated for each scenario. For the damage scenarios, a total of 45 damage severity levels are simulated on each floor using the numerical model. 180 data samples for the source domain (45 excitations×(3 damage scenarios + 1 undamaged scenario) are obtained. The damage labels are defined regarding to the damage locations, e.g. 0 (undamaged), 1 (damage on the 1st floor), 2 (damage on the 2nd floor), 3 (damage on the 3rd floor). Different with the source damage scenarios, the target domain data includes the undamaged and two single damage scenarios summarised in Table 4-6, which are from the real structure including 45 samples (15 samples for the undamaged and 1st floor damage each; 10 samples for 2nd floor damage). The labels of the target data are not available during the training process.

Table 4-6. Test and sample number for single damage detection (Imbalanced class).

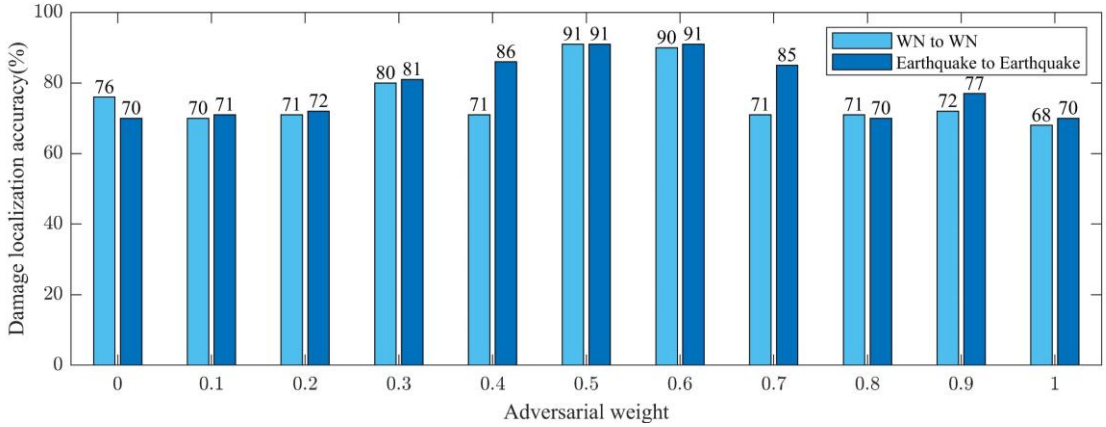
	3-story numerical model			**3-story experimental model**		
Damage scenarios	Damage location	Labels	Damage severity	Damage location	Labels	Damage severity
D0	Undamaged	0	/	Undamaged	0	/
D1	1st floor	1	0-30%	1st floor	1	30%
D2	2nd floor	2	0-30%	2nd floor	2	30%
D3	3rd floor	3	0-30%	-		-

The network is initially trained using the source domain data without domain adaptation. The batch size is 128 as the exponent of 2. The number of epochs is set equal to 150 which is applicable to the small sample dataset. The predicted test accuracy for the target domain is 48% using the pre-trained network. Three types of excitations are used to excite both numerical and experimental structures, namely, random (WN), earthquake and mixed random and earthquake excitations. Figure 4- 15(a) shows the damage localisation accuracy using the proposed JMDAD method with different weights. From Figure 4- 15(a), the case with $\omega = 0.1$ has the accuracy of 91% or above. The results

show that with different operational excitations, the proposed method can extract the damage sensitive features and identify the damage accuracy. Figure 4- 15(b) shows the identified accuracy when the numerical and real structures are subjected to the same type of excitations. From Figure 4- 15(b), when the adversarial weight $\omega = 0.5$, it has the highest accuracy of 91%. As the above, the proposed method could apply the knowledge learned from the numerical model to detect the damage accurately in real structure. Figure 4- 16(a) shows the comparisons of classification accuracies by the CNN network and the proposed JMDAD method with different adversarial weights. From Figure 4- 16(a), when $\omega = 0.1$, it achieves the highest accuracy. Figure 4- 16(b) shows the confusion matrix of classification results when $\omega = 0.1$. From the figure, the proposed method achieves the high classification accuracy with no less than 92.3%. Compared to the CNN network, the classification accuracy is increased by around 20% using the proposed method with limited unlabelled experimental data.

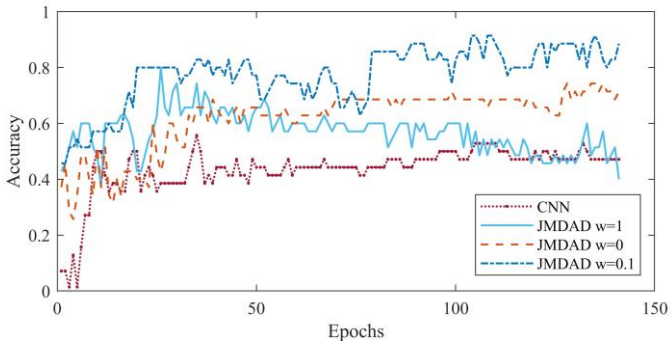


(a) Under different types of excitations



(b) Under the same type of excitations

Figure 4-15. Damage localization accuracy under different excitations with adversarial weight ($\omega = 0$ to 1)



(a) Testing accuracy and error over epochs

%	0	1	2	3
0 (13)	92.3	0.0	7.7	0.0
1 (13)	0.0	92.3	7.7	0.0
2 (9)	0.0	0.0	100.0	0.0
3 (-)	-	-	-	-

(b) Confusion Matrix of Classification results

Figure 4-16. JMDAD method performance for single-damage damage datasets

4.5.3. Damage prediction with different operational excitations

In practice, the operational excitation may be varied. The excitation has a big effect on structural damage detection. In this section, the structure under different ground excitations is studied. As the same as Section 3.3.1, the numerical model (Model 1) is used as the source domain with 45 excitations applied on the ground floor to form 180 data samples (45 excitations×(3 damage scenarios + 1 undamaged scenario) for each excitation. The real experimental structure is regard as the target domain and the experimental tests are conducted on the shaking tables. There are 35 samples (15 samples for the undamaged and 1st floor damage each; 9 samples for 2nd floor damage) for the target domain. The labels for the target domain data are not available in the training process. Both source

domain and target domain data are separate into datasets corresponding to each excitation.

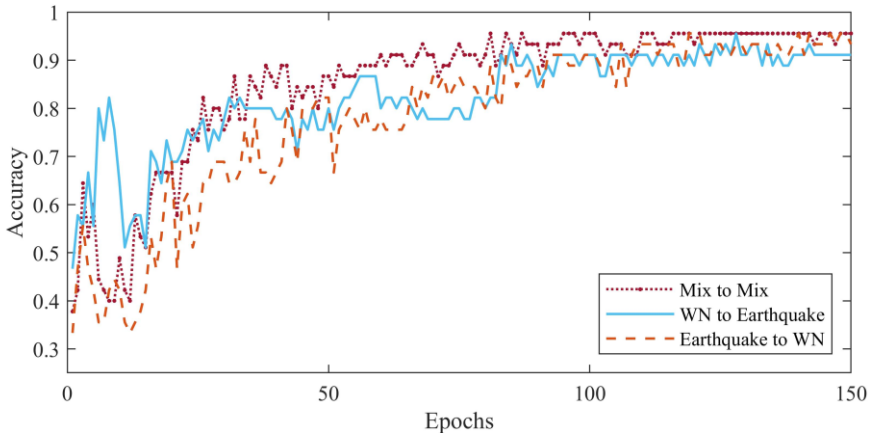


Figure 4-17.JMDAD method performance with operational environmental varieties

Three case studies have been conducted using the datasets of source and target domains from mixed to mixed excitations, random to earthquake excitations and earthquake to random excitations, respectively. Figure 4- 17 shows the accuracies of three cases using the proposed method with the adversarial weight $\omega = 0.1$. This is further confirmed a light weight is needed for operational varieties when the numerical and experimental structures have the similar scales. From Figure 4- 17, the accuracy with the mixed excitations is much higher than other two cases. Even with the case with the random excitation in the source domain and the earthquake excitation in the target domain or the opposite, the accuracy could still achieve around 90%. As a result, the proposed method could extract the feature which is sensitive to the structural damage and robust to the operational environment and accurately identify the structural damage location in civil engineering practice.

4.6. Knowledge transfer from numerical to real structures for the Canton Tower

This section presents a complicated example using the Canton Tower (CT). Firstly, the basic information about CT is introduced, including its dimensions, sensor arrangement and field measurement data. Subsequently, the field measurement is pre-processed, and the results for damage

detection are shown. Finally, the extracted damage-sensitive features are visualised. It needs note that there is only undamaged data available in field monitoring and the damage data is simulated by adding the white noise to the calculation using finite element model.

4.6.1. Real structure and its numerical model

The Canton tower, located in the city of Guangzhou, China, is a high-rise tower structure standing at a total height of 600 meters, as shown in Figure 4- 18(b). CT contains a 454 m main tower section and 146 m antenna mast (Ni et al., 2009). A long-term SHM system consisting of over 700 sensors has been installed on CT and details of the SHM system can be found in references (Ni et al., 2009; Ni et al., 2012). As shown in Figure 4- 18(b), sixteen accelerometers with the frequency range of 50Hz have been installed at eight levels along the height of the tower. Each level has two uni-anxial accelerometers to monitor the horizontal vibration along the x and y directions. Field measurement data used in this study are from the website (http://polyucee.hk/ceyxia/benchmark//). The wind direction and speed at a height of 461.1 m is also obtained by the anemometer installed in the southeast of the tower.

Based on physical properties of the tower, a reduced order finite element (FE) model has been established by Chen et al. (2011). This FE model includes 37 beam elements and 38 nodes with 185 degrees of freedom (DOFs) in total. Each node has 5 DOFs, including two horizontal translational DOFs and three rotational DOFs. To further reduce the model, a lumped mass model is proposed based on the reduced order FE model by retaining the x- and y-axis translational DOFs and removing rotational DOFs using dynamic condensation technique (Lin and Xia, 2003). The lumped mass model is built in MATLAB with only 8 segments and 9 nodes corresponding with locations of

accelerometers. Each node contains 2 DOFs, which indicate two horizontal translational DOFs along x and y directions shown in Figure 4- 18(a). As a results, the lumped mass model has 16 DOFs in total. Due to removing rotational DOFs, there are no torsional modes in the lumped mass model. The modal assurance criterion (MAC) values between mode shapes from the reduced order FE and lumped mass models are calculated to pair modes. Their MAC values are not less than 0.9900 as listed in Table 4-7. Natural frequencies of the lumped mass model are compared with existing results by the reduced order FE model and measurements as listed in Table 4-7. From the results, the maximum frequency difference compared with measurements is 12.832%, and that is due to the uncertainty, such as the modelling error, measurement noise and operational environment.

Table 4-7. Natural frequencies of CT from measurements, and FE and proposed lumped mass models.

Vibration direction	Measurements (Hz) (Chen et al. 2011)	FE model (Hz) (Chen et al., 2011)	Lumped mass model (Hz)	MAC with FE model	Difference with Measurements (%)	Difference with FE model (%)
1st y-axis	0.101	0.110	0.111	0.9998	9.606	0.638
1st x-axis	0.148	0.159	0.159	0.9995	7.534	0.095
2nd y-axis	0.358	0.347	0.370	0.9996	3.248	6.521
2nd x-axis	0.534	0.485	0.465	0.9991	12.832	4.026
3rd y-axis	0.810	0.738	0.740	0.9900	8.655	0.257
3rd x-axis	0.980	0.902	0.926	0.9936	5.506	2.665
1st torsion	0.535	0.461	-		-	-
2nd torsion	1.271	1.122	-		-	-

The measurement of wind data at 19:00, 10 January 2010, is analysed to obtain wind characteristics. The wind velocity is first decomposed into X and Y directions as v_x and v_y in a plane shown in Figure 4- 18(c), as follows (Hua et al., 2020):

$$v_x(t) = v(t)\sin(\theta(t) + 18^o)$$
$$v_y(t) = v(t)\cos(\theta(t) + 18^o) \quad \text{Eq. 4-12}$$

where v and θ are the measured instantaneous wind speed and wind direction at the time series t. The distribution of wind speed with tower height needs to be calculated. According to the Chinese design load code GB 50009-2001 (2002), the wind speed profile is an approximate exponential relation with the height of the structure as,

$$\begin{aligned} \upsilon_{xn} &= \upsilon_{x,Z_n=461.1}\left(\frac{Z_n}{461.1}\right)^{\alpha} \\ \upsilon_{yn} &= \upsilon_{y,Z_n=461.1}\left(\frac{Z_n}{461.1}\right)^{\alpha} \end{aligned} \quad \text{Eq. 4-13}$$

where $\upsilon_{x,Z_n=461.1}, \upsilon_{y,Z_n=461.1}$ are the wind velocities in X and Y directions obtained from the anemometer at $Z_n = 461.1m$. Z_n represents the height from the ground of the tower in Figure 4-18(b). α is the mean wind speed exponential index. To obtain the wind force on the actual tower, a series of wind tunnel tests on the tower model have been conducted (Zhou et al., 2009). Based on the experimental results, the equivalent wind loads at varying heights along the tower F_{xn} and F_{yn} can be calculated as,

$$\begin{aligned} F_{xn}(t) &= \frac{1}{2}\rho\upsilon_{xn}^2 H_n D_n \mu_{F_x} \\ F_{yn}(t) &= \frac{1}{2}\rho\upsilon_{yn}^2 H_n D_n \mu_{F_y} \end{aligned} \quad \text{Eq. 4-14}$$

where $\upsilon_{xn}, \upsilon_{yn}$ represent the calculated wind speeds in X and Y directions at the height Z_n by Eq. 4-13; ρ is the air density; H_n is the height of each segment of the tower and D_n is the characteristic dimension uniformly adopted as 18m. μ_{F_x}, μ_{F_y} are the aerodynamic force coefficients. Eq. 4-14 is used as the input to each DOF of the lumped mass model in the numerical simulation next section.

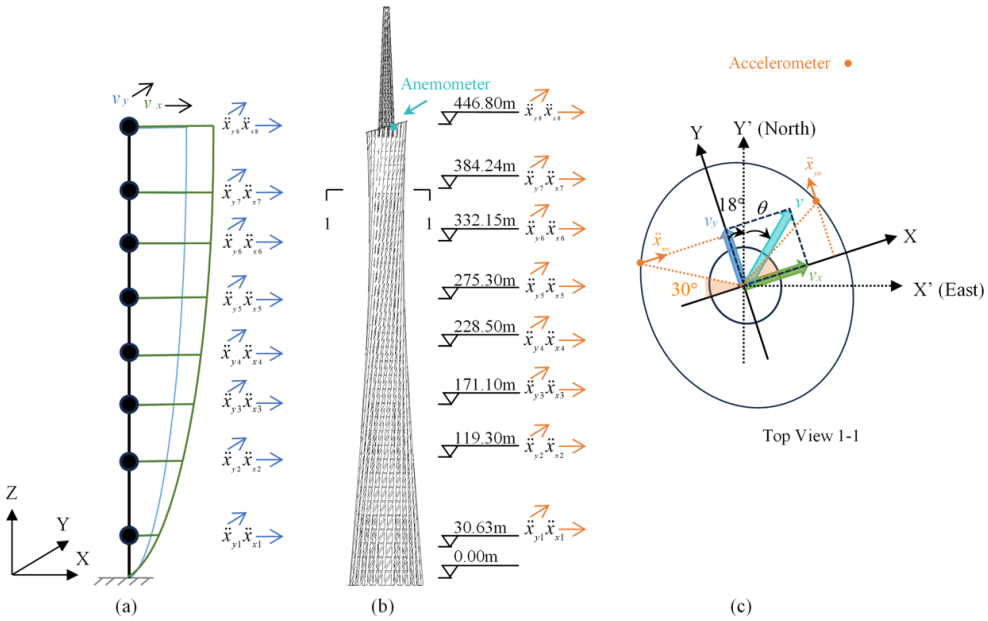


Figure 4-18.Canton Tower: (a) 16-DOF numerical model, (b) real structure, and (c) the top view.

4.6.2. Damage prediction with imbalanced dataset

In this section, the numerical model is regard as the source domain and the real structure is as the target domain. Similar to previous case studies in Sections 3 and 4, the labelled source domain data contains different damage locations and severities are generated using the lump mass model. Considering that the actual situation is that the damage to the structure is minor structural damage rather than severe damage. The labelled source domain data are designed with light damage severity. Single-element damage of 0-20% severity is simulated form the source domain data. The damage severities are randomly introduced to each element, under the wind loads simulating a total of 270 samples (Undamaged + 8 damage locations)× 30 random damage severities) listed in Table 4-8. In the unlabelled target domain, the undamaged data is collected from field measurement since the tower condition is known as health. The field damaged data are simulated using the reduced-order FE model (Chen et al., 2011) added 3% WN noise. Considering the operational condition in the real structure,

the damage in the middle and top of the tower are simulated. In a single damage dataset, the damage occurs at heights 171-228m and 382-446m, with 15 % severity to be tested as unlabelled data. Keeping a similar sample number of each class with source domain data, a total of 90 samples (Undamaged + 2 damage locations) × 30 collected wind excitations) from the target domain are obtained. Also, for an unseen damaged dataset, the damage accidentally appears in the top section of the tower in heights 350-355m (TEMPE-RATE - refuge floor damage) and 428-433m (ARCTIC - observation floor damage) with 25% severity, a total of 90 samples (Undamaged + 2 damage locations) × 30 collected wind excitations) in the target domain dataset. It is noted that the labelled source domain data for training does not cover the damage severity range of the unseen datasets.

Table 4-8. Damage detection with unseen datasets from source and target domains

16-DOF numerical model				Measurement					
				Single damage			Unseen damage		
Damage scenarios	Damage location	Labels	Damage severity	Damage location	Labels	Damage severity	Damage location	Labels	Damage severity
D0	Undamaged	0	0-20%	Undamaged	0	/	Undamaged	0	/
D1	1st element	1	0-20%						
D2	2nd element	2	0-20%						
D3	3rd element	3	0-20%	171-228m	3	15%			
D4	4th element	4	0-20%						
D5	5th element	5	0-20%						
D6	6th element	6	0-20%						
D7	7th element	7	0-20%				350-355m	7	25%
D8	8th element	8	0-20%	382-446m	8	15%	428-433m	8	25%

For both labelled source and unlabelled target datasets, the recording period is 1350 seconds for the dynamic response under the wind excitation with a sampling rate of 50Hz, and it is split into 90 samples, with each sample 15 s. The recording acceleration responses at each floor are filtered by a low pass filter with a cut-off frequency of 10 Hz. In addition to the wind load, the WN with the same sampling rate and length is also applied to the lumped mass model to simulate different operating environments for studying the effect of adversarial weights. Then the TF samples are calculated using the ratio between the bottom seven sensors' responses over the top sensor mentioned in Section3.2,

input to CNN with the size of $7 \times 2 \times 660$. The output size is 8, which corresponds to eight classes of damage scenarios.

The proposed JMDAD method is trained based on the labelled source domain datasets under the WN excitation and wind load, respectively. The single damage scenarios under wind load from field measurements are utilized to evaluate the accuracy of the proposed method with different adversarial weights. The outcome of single damage detection is illustrated in Figure 4- 19. In this figure, the accuracy of two cases using the datasets of source and target domains with different types of excitations and the same type of load excitation. Under the different types of loads, the highest accuracy appears as 90% when the adversarial weight ω=0.1. While under the same type of excitations, e.g. wind load, the best performance is 91% when the ω set to 0.5 or 0.6. The results of the tests in the real structure show that the adjustment of the adversarial weight is consistent with the results of previous numerical simulations and experimental case studies. When there are model errors (e.g., differences in degrees of freedom or size) between the model that produces data with labelled source domains and the model that produces unlabelled target domains, the adversarial weight needs to be set to be greater than 0.6 to obtain better prediction accuracy. If there are other uncertainties such as the impact of the operating environment, it suggests that the adversarial weight needs to be reduced to improve the accuracy of damage identification.

The prediction accuracy with epochs during the training process is shown in Figure 4- 20 and the results for each scenario of a single damage dataset and unseen damage dataset are illustrated in Table 4-9. The adversarial weight is kept at 0.1, considering the different types of loads between numerical and real structures. For the single damage detection, the results show that the proposed

method is successfully to detect the damage with a correct classification of 96% for undamaged case and with over 85% accuracy for the damage at 171-228m and 382-446m heights. For the unseen damage dataset, it has a good performance with the 94% accuracy for undamaged case, and the identification accuracies for damage at heights of 350-355m and 428-433m are 84% and 85% respectively.

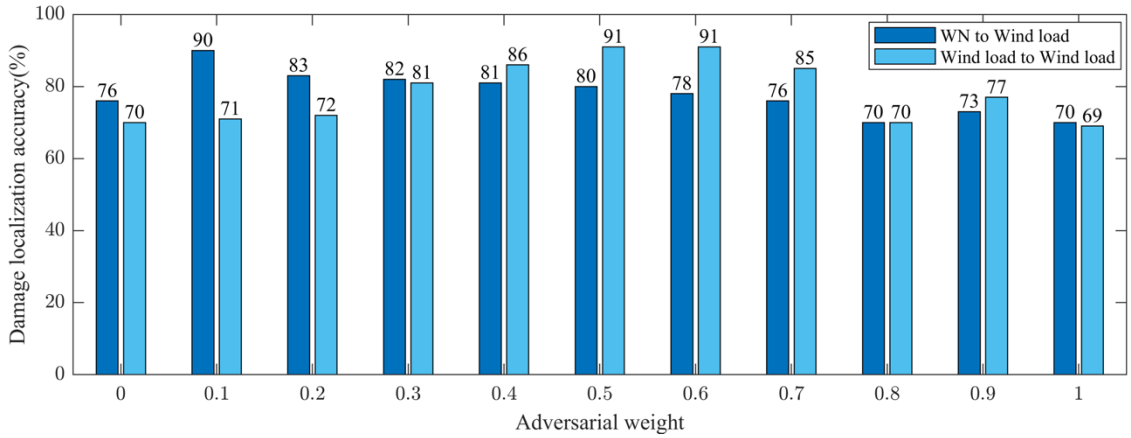


Figure 4-19. Damage localization accuracy with adversarial weight (ω = 0 to 1) with WN and wind load.

Table 4-9. Accuracy of JMDAD method for damage detection

Single damage		Unseen damage	
Damage location	Accuracy (%)	Damage location	Accuracy (%)
Undamaged	96	Undamaged	94
171-228m	85	350-355m	84
382-446m	86	428-433m	85
Overall	90	Overall	88

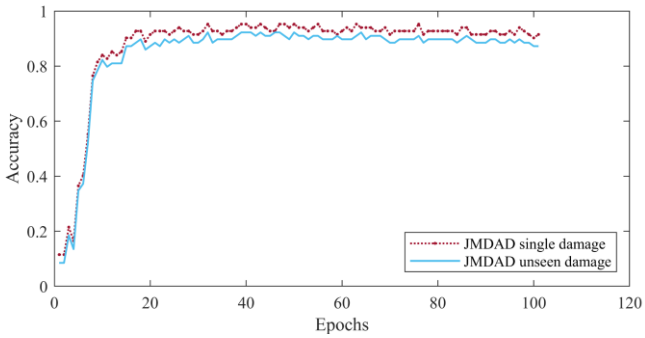
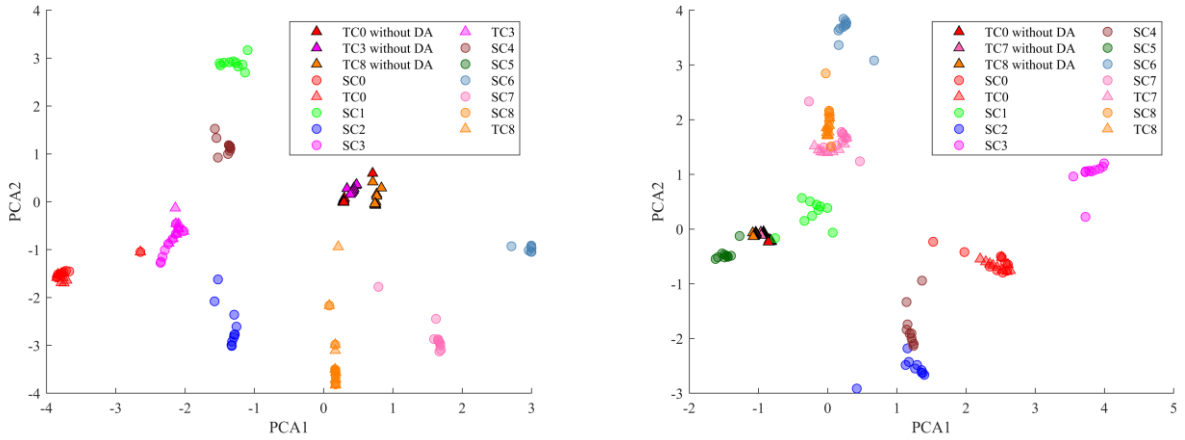


Figure 4-20.Testing accuracy over epochs with single and unseen damage

Figure 4- 21 shows that the PCA feature vitalization using the proposed method with different

unlabelled datasets. In Figure 4- 21(a), TC0, TC3 and TC8 without DA represent features of Classes D0, D3 and D8 in Table 4-8 by the CNN network, respectively. TC0, TC3 and TC8 represent the features of these three classes by the proposed method, respectively. SC0 to SC8 represent the features of Classes D0 to D8 in the source domain, respectively. Similar to the PCA feature distribution of the single damage dataset, Figure 4- 21(b) shows the feature visualization of labelled and unlabelled datasets of the unseen damaged datasets. From the figure, compared to the results obtained by the CNN network, the features in the target domain using the proposed method are better separated between each class and aligned well with the corresponding features in the source domain.



(a) Single damage dataset (b) Unseen damage dataset

Figure 4-21. Feature visualization with PCAs with single and unseen damage dataset

4.6.3. Discussions

The proposed method mainly integrates the domain and class levels of domain adaptation with an optimising weight to reduce the uncertainty due to modelling errors and varieties of operating environments when extracting features in practice. The following observations could be obtained from numerical and experimental results,

The adversarial weight is mainly to consider the influence of the class and domain discrepancies. When the source and target structures have significant different scales, the order of models will be

different, which induces a significant discrepancy in damage classifications. The weight should be chosen a large value to reduce the class discrepancy. When the source and target structures have similar scales and the operational environments are different, the domain discrepancy will be dominated, and the weight needs to take a small value for knowledge transfer.

The proposed method is to overcome this issue with limited unlabelled measurement data. In above sections, experimental examples and applications to real structures have demonstrated that the method can perform effective damage detection in the presence of environmental uncertainties and model discrepancies. Over-fitting issues usually indicate that the trained DL model only works with training data, while from the case study of the experiment and real case, applying the test data different from the training data also works for the DL model with good accuracy. The results use the test data differently from the training data, considering the environment and modelling errors to validate the method. Considering that the labelled dataset and the unknown test dataset have different damage schemas and damage severities, the proposed method has good generalisation and utilisation value for damage detection in practical engineering.

The numerical and experimental results have demonstrated the performance of the proposed method for different excitations. The results show that JMDAD has a good ability in extracting damage sensitive features and damage detection. This is mainly due to two aspects in the proposed method, 1) the transmissibility function is used to eliminate the effect of operational excitations; 2) different loading scenarios are also considered in the domain discrepancy loss.

4.7. Summary

A JMDAD based method has been developed to transfer the knowledge learned with labelled

data from the numerical model to real structure for structural damage detection with limited unlabelled measurement data. The proposed method is composed of a CNN generator and two classifiers with a fully connected layer and one discriminator. The TF data are used as the input of the proposed method. Three case studies have been conducted to verify the performance of the proposed method: the knowledge transfer between two building structures with different storeys, that between the numerical and experimental structures subjected to different excitations and the application for the Canton tower. Numerical and experimental results show that the proposed JMDAD method is reliable and accurate to identify the damage of the real structure with limited unlabelled measurement data. The following points can be concluded,

1) The proposed method integrates the maximum classifier discrepancy domain adaptation with the adversarial discriminative domain adaptation through optimising the adversarial weight. The optimisation of the adversarial weight could provide an accurate and fast assessment for damage identification of the target structure.

2) The results show that the proposed method is much robust to the measurement noise and the operational excitation. It achieves 91 % for structural damage localization accuracy between two structures with different storeys under different operational excitations.

3) The knowledge learned from the numerical model has been successfully applied for structural damage localization of the experimental structure and real structure with 90% accuracy using imbalance data and unseen data with different excitations respectively.

4) The training process can be visualised through the distribution of PCA components. The difference of the PCA values is related to the domain discrepancy and the margin of the distribution for PCA components of each class is corresponding to the class discrepancy.

While domain adaptation improves damage detection, the limited real measurement data may not fully capture all operational conditions. To address this, Chapters 5 and 6 introduce physics-informed models that integrate governing equations, enhancing generalizability and interpretability.

Chapter 5. Parallel neural ordinary differential equations based damage identification for building structures

5.1. Overview

Data-driven-based methods for structural damage identification need to extract the high dimensional features from structural responses. Most existing data-driven-based methods are based on black-box models without interpretability and explainability. In this study, a new parallel neural ordinary differential equations (PNODEs) based method is developed for structural damage identification. The proposed method integrates the state-space equation based physical model with a neural network block, which includes a bunch of parallel neural networks. Each neural network could correspond to a part of the structure. The state-space equation for the structural dynamic system provides the physical constraints of prior knowledge to this neural network block. The neural network block captures the entire discrepancy from the input data compared with the prior knowledge and separates this discrepancy into each neural network corresponding to each structural part. The separated discrepancy of each neural network includes the high dimensional features of each part for structural damage identification. The proposed approach explores the closed-form expression for a group of neural networks to identify the structural parameters, which increases the interpretability and enhances the reliability of the neural network model. A three-story building structure and a three-dimensional IASC-ASCE benchmark frame are used to verify the performance of the proposed method. The results show that the proposed method could be reliable and accurate to localise and quantify the damage.

5.2. State of the art

Civil infrastructure is deteriorated due to aging, environmental attacks, operational loading and extreme events. Structural damage identification allows for identifying structural damage location and severity and the cost-effective maintenance could be conducted timely for structural safety. Vibration-based methods have been used for structural damage detection (Fan and Qiao, 2011). These methods could mainly be categorised as model-based and data-driven approaches (Hou and Xia, 2021). Model-based approach relies on a detailed finite element model (FEM) that accurately reflects the actual structure. A dynamic response sensitivity-based method has been presented for structural damage identification in time domain (Lu and Law, 2007; Baybordi and Esfandiari, 2022). Hou et al. (2018) have presented a sparse regularisation technique for structural damage detection. Bayesian model updating has been used for structural damage identification (Huang et al. 2017; Liu et al. 2023) have explored. Despite great progress made in structural damage localization and quantification, significant challenges still remain, such as the discrepancy between the numerical model and real structure, operational environmental varieties, and the considerable computational efforts required for simulations, especially for the application of large and complex structures.

The data-driven approach for structural damage detection, especially machine learning based methods, has recently attracted the interest of researchers and engineers (Avci et al., 2021). Pathirage et al. (2018) proposed an autoencoder neural network-based approach using the relation between vibration characteristics and structure damages for steel frame damage qualification. Yu et al. (2019) proposed a deep convolutional neural network to extract high-level features from raw response data for structural damage identification of smart building structures. As the residual neural network could avoid the gradient vanishing problem by utlilising skip connections, the deep residual network

framework has been used for structural health monitoring (Wang et al., 2020). These data-driven method based on ML could automatically extract the damage-sensitive feature from the monitoring data based on classification or regression tasks to identify structural damage. Since these methods purely depend on reliable data and neural networks to build a 'black box' and rapidly learn the complicated relationship between the input and the output of various tasks based on their ability to approximate the arbitrary structure functionality. However, they usually lack a physical explanation of their intrinsic mechanisms (Zhang et al., 2020). Furthermore, in practical application, these methods face challenges due to lack of complete high-quality training data, especially the scarcity of structural damage data and measurement noise.

Physical informed neural network (PINN) integrates the physical knowledge with the ML process to solve the forward and inverse scientific problems (Karniadakis et al., 2021). Bao et al. (2021) used system response to reconstruct the mode shape for a cable stayed bridge based on a deep learning model with a complex cost function that constructs the relationship between structural response and modal information to constrain the training process. Lei et al. (2024) integrated a physical loss function related to modal sensitivity into a deep-learning network for structural damage identification. As the above, the hybrid model not only takes advantage of the fact that neural networks can quickly and accurately fit relationships between input and output data to achieve the task for system identification but also allows the overall the training process to follow informed physical rules to enhance credibility (Wu et al., 2024). The integration of physical knowledge into neural network architectures enhances the interpretability and physical consistency of the ML-based data-driven approach. Lai et al. (2021) used the Neural ordinary differential equations (NODEs) to

identify the parameters of dynamic systems. Instead of obtaining large amounts of health and damage data for pre-training the NN, the measurement time series data is directly compared with the health condition data generated by the ordinary differential equations (ODEs), which can represent structures in the form of state space equations and have been widely used in simulation for both linear and non-linear modelling tasks (Karpatne et al., 2017; Raissi and Karniadakis, 2018). Most recently, Meng et al. (2020) and Shukla et al. (2021) embedded the physic knowledge into a group of parallel neural networks for increasing computational efficiency and dealing with complex tasks. Within one physical constraint, each neural network in this group can be given separate inputs and then trained simultaneously.

In this study, a novel structural damage identification method based on parallel neural ODEs (PNODEs) has been proposed for complex frame structures. The proposed PNODEs framework mainly includes two parts: a physical information term represented the prior knowledge, and a discrepancy term processed by parallel neural networks. Each neural network was specifically designed to consider responses from a subsection of the structure. Each neural network was parallel-solved and spliced together at the same time for solving ODEs. The process is that the neural network is used to solve ODEs, learning the discrepancy between prior physical knowledge and the actual system. The health state of the structure can be informed as the prior knowledge, and parallel neural networks learn the discrepancy through each neural network corresponding to each subsection of the structure. Therefore, the priori knowledge and discrepancy terms are then summed to form the whole model, where the discrepancy term can be considered as the damaged part of the structure.

The chapter is organised as follows: the theoretical background of NODEs is introduced first,

and then the structural damage identification based on PNODEs is explained in Section 2. Numerical experimental studies of a building structure with linear or nonlinear damage were conducted to verify the proposed method in Sections 3 and 4. An application for a three-dimensional IASC-ASCE benchmark frame is used to further verify the performance of the proposed method in Section 5. The conclusions are drawn in Section 6.

5.3. Methodology

In this section, the neural ordinary differential equations (NODEs) theory is introduced briefly. Then the key phases and mechanisms of the proposed parallel neural ordinary differential equations (PNODEs) framework are presented, including the data incorporation, prior physical knowledge, the parallel neural network (NN) block and the damage index.

5.3.1. NODEs Theory

5.3.2. Structural Dynamic System

As a priori physical knowledge in this study, the ordinary differential equation (ODE) of multiple degrees of freedom (DOFs) systems is expressed as,

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) + \mathbf{g}(\mathbf{x}(t), \dot{\mathbf{x}}(t)) = \mathbf{u}(t) \quad \text{Eq. 5-1}$$

where $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are the system mass, damping and stiffness matrices, respectively. $\ddot{\mathbf{x}}(t)$, $\dot{\mathbf{x}}(t)$ and $\mathbf{x}(t)$ are the acceleration, velocity, and displacement responses, respectively. $\mathbf{u}(t)$ is the excitation force, and $\mathbf{g}(\mathbf{x}(t), \dot{\mathbf{x}}(t))$ is the terms associated with state variables to represent two situations: a) the difference between the numerical model and real structure due to modelling errors and operational environmental varieties; b) the structural damage.

Eq. 5-1 can be written as state space form as

$$\frac{d\mathbf{h}(t)}{dt} = \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\big(\mathbf{h}(t)\big) \quad \text{Eq. 5-2}$$

where $\mathbf{h}(t) = \begin{bmatrix}\mathbf{x}(t)\\ \dot{\mathbf{x}}(t)\end{bmatrix} \in \mathbb{R}^n, \mathbf{A}\mathbf{h}(t) = \begin{bmatrix}\mathbf{0} & \mathbf{I}\\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C}\end{bmatrix}\mathbf{h}(t)$, $\mathbf{G}(\mathbf{h}(t)) = \begin{bmatrix}\mathbf{0}\\ -\mathbf{M}^{-1}\end{bmatrix}\mathbf{g}(\mathbf{h}(t))$, $\mathbf{B}\mathbf{u}(t) = \begin{bmatrix}\mathbf{0}\\ -\mathbf{M}^{-1}\end{bmatrix}\mathbf{u}(t)$.

5.3.3. Neural Ordinary Differential Equations (NODEs)

NODEs have garnered significant attention in recent years for a close connection between neural networks and differential equations. NODEs could be a continuous equivalent expression of residual networks (ResNets). The transformation of the hidden state from layer *t* to layer *(t+1)* in ResNets is carried out by a differentiable function $f_t(\cdot)$ as below (He et al. 2016),

$$\mathbf{h}_{t+1} = \mathbf{h}_t + f_t(\mathbf{h}_t) \quad \text{Eq. 5-3}$$

where $\mathbf{h}_t \in R_d$ is the hidden state of the layer. The difference $\mathbf{h}_{t+1} - \mathbf{h}_t$ is the discretisation of the derivation of $\mathbf{h'}_t$ when $\Delta t = 1$, following Euler discretization (Wanner & Hairer 1996; Butcher 2016).

Eq. 5-3 can be rewritten as

$$\frac{\mathbf{h}_{t+1} - \mathbf{h}_t}{\Delta t} = f(\mathbf{h}_t) \quad \text{Eq. 5-4}$$

When $\Delta t \to 0$, Eq. 5-4 becomes to

$$\lim_{\Delta t \to 0} \frac{\mathbf{h}_{t+1} - \mathbf{h}_t}{\Delta t} = \frac{d\mathbf{h}(t)}{dt} = \lim_{\Delta t \to 0} \frac{f_t(\mathbf{h}_t)}{\Delta t} = \tilde{f}(\mathbf{h}(t), t, \theta) \quad \text{Eq. 5-5}$$

The hidden units of the neural networks are parameterised as the ODE form. The data point h_0 can be mapped into the set of features to time step *t* by solving the Initial Value Problems (IVPs) as,

$$\begin{cases} \frac{d\mathbf{h}(t)}{dt} = f\big(\mathbf{h}(t)\big) = \tilde{f}(\mathbf{h}(t), t, \theta), \\ \mathbf{h}(t_0) = \mathbf{h}_0 \end{cases} \quad \text{Eq. 5-6}$$

In the actual operation of the structural dynamic problems, there is normally forced excitation instead of the pure Initial Value Problems (IVPs) and informs the priory physical knowledge into

NODEs. By adding the excitation force **u**(t), Eq. 5-6 can be obtained as,

$$\begin{cases} \frac{d\mathbf{h}(t)}{dt} = f\big(\mathbf{h}(t)\big) = \tilde{f}(\mathbf{h}(t), t, \mathbf{u}(t)\ \theta), \\ \mathbf{h}(t_0) = \mathbf{h}_0 \end{cases} \quad \text{Eq. 5-7}$$

Solving the continuous dynamic system using a neural network $\tilde{f}(\cdot)$ is an important step in establishing NODE (Dupont et al. 2018; Chen et al. 2019). A neural network $\tilde{f}(\cdot)$ is the solution of the function $f(\cdot)$ which represents the dynamics of the system state. When solving the ODE from the initial condition $\mathbf{h}(t_0) = \mathbf{h}_0$, which initiative the states of the ODE at a given time *t* depend on the initial conditions h_0. Thus, given the **u**(t) as the input to the $\tilde{f}(\cdot)$, the output layer h_T is the solution of Eq. 5-7 at the final time *T.*

In NODEs, an initial state $\mathbf{h}(t_0)$ can be mapped to the final state $\mathbf{h}(t_T)$ from data to feature by mathematically solving an ODE as $\mathcal{L}(ODESolve(\mathbf{h}(t_0), \tilde{f},\ t_0, t_T, \mathbf{u}(t), \theta))$. This defines a loss function $L(\cdot)$ in a forward process, to evaluate the difference between the predicted state $\mathbf{h}(t_T)$ and the true state $\mathbf{h}_{\text{true}}(t_T)$ as,

$$\mathcal{L}(\mathbf{h}(t_T)) = \mathcal{L}(\mathbf{h}(t_0)\ + \int_{t_0}^{t_T} \tilde{f}(\mathbf{h}(t), t, \theta)\, dt) = \mathcal{L}(ODESolve(\mathbf{h}(t_0), \tilde{f},\ t_0, t_T, \mathbf{u}(t), \theta)) \quad \text{Eq. 5-8}$$

where θ is the trainable weight of the neural network $\tilde{f}(\cdot)$, which can be optimised by minimize the loss function $L(\cdot)$. According to Rung-Kutta methods, the solutions of $ODESolve$ are approximated by establishing NODE solvers (Dupont et al. 2018; Chen et al. 2019).

$$\mathbf{h}(t_T) = ODESolve(\mathbf{h}(t_0), \tilde{f},\ t_0, t_T, \mathbf{u}(t),\ \theta) \quad \text{Eq. 5-9}$$

where $\mathbf{h}(t_T)$ is the generated results from the neural ODEs based on Eq. 5-7.

In solving the NODE, the measured data is only required to be given to the network, the derivative function $\frac{d\mathbf{h}(t)}{dt}$ of the measurement is unnecessary. This training process can be treated as solving ODEs in Eq 5-7. As a result, the solutions of ordinary differential equations stored in $\tilde{f}(\cdot)$

are finally obtained by training the neural network.

Loss function:

During the training process of NODEs, the parameters of the neural network θ are updated by minimising the MSE loss functions:

$$\text{MSE} = \frac{1}{m}\sum_{i=1}^{m}|\mathbf{h}_{\text{true}}(t_i) - \mathbf{h}(t_i)| \quad \text{Eq. 5-10}$$

where $\mathbf{h}_{\text{true}}(t_i)$ is the true solution, $\mathbf{h}(t_i)$ is the generated results at time t_i, and m is the number of the samples of $\mathbf{h}(t_i)$.

5.3.4. PNODEs for Structural Damage Identification

As shown in Figure 5- 1, the proposed PNODEs framework is introduced in this section. The framework mainly includes three phases, e.g. data measurement, solving PNODEs and inference of trained neural network. In Phase I, the limited measured displacement and velocity responses are formed the state vectors for input of Phase II. The prior physical knowledge is integrated into the parallel neural networks with supervision of limited measurement data. Each neural network is corresponding to a floor of the building structure as a substructure. Phase III is to apply the trained PNODEs for response prediction and structural damage identification. The proposed PNODE framework can be summarised as follows: (i) By implicitly placing a prior structured dynamical system in a state space form, the general high-performance representation of NNs is used to represent the structural damage by discovering discrepancies between the measured data and a priori system. (ii) Building a set of parallel neural networks. Based on the characteristics of the structural dynamics governing equations, each neural network is provided with only the measured responses associated with the part of the structure it represents, which allows the discrepancy information to be stored

separately in each network corresponding to each part of the structure. The great favour of this parallel neural network is that it avoids the whole discrepancy information being stacked in one whole neural network, further clearly identifying the source of the discrepancy (iii) Inference with each of the trained NN as discrepancy, plotting and fitting the inputs and outputs of a neural network to discover the latent restore force. Finally, the fitted parameter of the restore force indicating the stiffness change of the relevant section is utilized to localise and quantify damage.

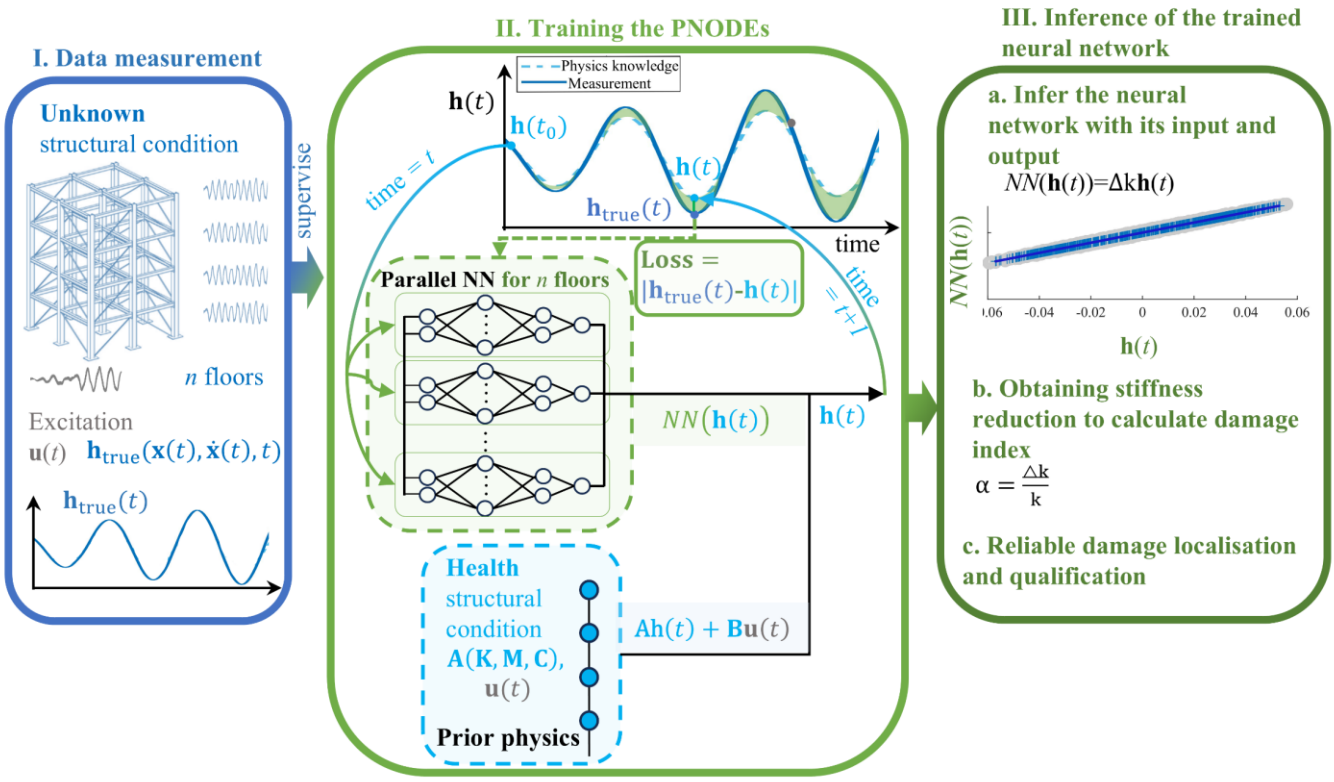


Figure 5-1. PNODEs framework

The main contribution of this work can be summarised as (i) Based on PNODES, discrepancy information between the prior physics and real structure is stored using a set of parallel neural networks with prior knowledge of the structural dynamics model under health conditions (based on ODEs) and the training of network does not require large amounts of measurement data. (ii) Damage localisation is achieved by discovering and separating the discrepancies of each part of the structure,

respectively, using each neural network in a set of parallel networks. (iii) damage quantification is performed by observing structural restoring force from each neural network and the neural network is further interpreted to enhance interpretability.

5.3.5. Informed priory physical knowledge

In Phase II, the simplified numerical model is embedded into the parallel neural network as the prior physical knowledge. As shown in Figure 5-1, this phase consists of a prior physics block (blue block) and a parallel NN block (green block). In the prior physics block, the prior structural physics knowledge can be simplified as multiple DOFs in Eq. 5-1, and it can be expressed in the form of state-space equations.

$$\frac{d\mathbf{h}_{\text{phy}}(t)}{dt} = \mathbf{A}\mathbf{h}_{\text{phy}}(t) + \mathbf{B}\mathbf{u}(t) \qquad \text{Eq. 5-11}$$

where **A** and **B** matrices in this physical model consist of the **K**, **M** and **C** matrices of their corresponding physical parameters. In this study, the prior structural physics represents the structure in a healthy state generating $\frac{d\mathbf{h}_{\text{phy}}(t)}{dt}$. Once the structure is damaged, the structural damage information compared with the health state is, therefore, modelled as parallel NNs as the discrepancy term. Then, the parallel NN block was utilized to learn the discrepancy (green shadows) between measurement (blue line) and prior structural physics knowledge (blue dash line) in the latent space through time-series data. Finally, combining the process of solving ODEs with neural networks mentioned in Section 2.1.2., and informs the physical model, Eq. 5-2 can be rewritten as the NODE form,

$$\frac{d\mathbf{h}(t)}{dt} = \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{u}(t) + f_{NN}(\mathbf{h}(t), t, \theta) \qquad \text{Eq. 5-12}$$

where $f_{NN}(\cdot)$ is the approximated functions learned by a group of parallel NNs.

5.3.6. Design of the parallel neural networks

After the displacement and velocity response are measured from the real structure, these responses are formed as a state vector $\mathbf{h}_{\text{true}}(t)$ for supervising PNODEs as shown in Phase II. In order for each neural network to represent only one part of the structure, separating the discrepancy terms from each part of the structure from the whole into each neural network, the inputs and outputs of each neural network were specifically designed based on the responses from subsections of the structure. For a building structure with *n* floors as shown in Figure 5- 1, each floor could be a subsection. The equation of motion for the i^{th} floor can be expressed as below (Nayeri et al., 2008; Zhan et al., 2014),

$$u_i(t) = -m_i\ddot{x}_i(t) + k_i x_{i-1}(t) - (k_i + k_{i+1})x_i(t) + k_{i+1}x_{i+1}(t) + c_i\dot{x}_{i-1}(t) - (c_i + c_{i+1})\dot{x}_i(t) + c_{i+1}\dot{x}_{i+1}(t) \quad \text{Eq. 5-13}$$

where k_i, m_i, c_i are the stiffness mass and damping coefficients of the *nth* floor; x_i, $\dot{x}_i$ and $\ddot{x}_i$ are the displacement, velocity and acceleration of the *i*th floor; u_i is the external force applied in the *i*th DOF. From Eq. 5-13, it can be found that the *i*th floor of the physical parameters changes are relevant to the state of the *(i-1)*th, *i*th and (i+1)*th* floor. Therefore, the input of each neural network $\mathbf{h}_i(t) = [x_{i-1}(t); x_i(t); x_{i+1}(t); \dot{x}_{i-1}(t); \dot{x}_i(t); \dot{x}_{i+1}(t)]$(when *i*=1, the input items of the neural network $x_{i-1}(t)$ and $\dot{x}_{i-1}(t)$ are set as 0; when *i*=*n*, the input items of the neural network $x_{i+1}(t)$ and $\dot{x}_{i+1}(t)$ are set as 0).

A parallel neural network is designed for PNODEs. There are *n* neural networks in the framework and each network represents a subsection, e.g. a floor of the structure**.** For each time step *t*, the PNODEs generated a state vector $\mathbf{h}(t) = [x_1(t); x_2(t); \dots; x_n(t); \dot{x}_1(t); \dot{x}_2(t); \dots; \dot{x}_n(t)]$. $\mathbf{h}(t)$ is regrouped as $\mathbf{h}_i(t)$ (*i* = 1, 2, …, *n*). $\mathbf{h}_i(t)$ is corresponding to the i^{th} neural network, e.g.

the i^{th} floor of the structure.

5.3.7. Solving the ODEs based on the parallel neural networks

Details of Phase II, architecture of PNODEs is shown in Figure 5-2. The parallel NN block cooperating with the prior physics block is to generate $\mathbf{h}(t)$. With the cooperation of the prior physics block, the parallel NN block is constrained by structural dynamic rules. For the time step *t*, the parallel neural networks block, $f_{NN}(\mathbf{h}(t), t, \theta)$ in Eq. 5-12 has been designed as a group of parallel neural networks. Eq. 5-12 can be rewritten as,

$$\frac{d\mathbf{h}(t)}{dt} = \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{u}(t) + \begin{bmatrix} \mathbf{0}_{n\times 1} \\ f_{NN1}(\mathbf{h}_1(t), t, \theta) \\ f_{NN2}(\mathbf{h}_2(t), t, \theta) \\ \cdots \\ f_{NNn}(\mathbf{h}_n(t), t, \theta) \end{bmatrix} \quad \text{Eq. 5-14}$$

where the $f_{NNi}(\mathbf{h}_i(t), t, \theta)(i = 1, 2, \ldots, n)$ is the *i*th neural network, and the input of the neural network is $\mathbf{h}_i(t)$**.** $\mathrm{G}(\mathbf{h}(t))$ can be obtained by combining $\{f_{NNi}(\mathbf{h}_i(t), t, \theta), i = 1,2, \cdots n\}$ together which is a derivative of discrepancy item relative to each section.

In the parallel NN block, the initial condition $\mathbf{h}(t_0)$ when $t = t_0$ from the real structure of is obtained and regrouped as $\mathbf{h}_i(t_0)$ given to the parallel NN block to generate discrepancy item $\mathrm{G}\big(\mathbf{h}(t)\big)$. In the prior physics block, at the same time step, the derivative of the physical structure's state $\frac{d\mathbf{h}_{\mathrm{phy}}(t)}{dt}$ in Eq. 5-11 is calculated using the excitation force **u**(t). Then, the output of the PNODEs block $\frac{d\mathbf{h}(t)}{dt}$ is determined by the sum of $\mathrm{G}\big(\mathbf{h}(t)\big)$ and $\frac{d\mathbf{h}_{\mathrm{phy}}(t)}{dt}$, and the acceleration vector of the structure $[\ddot{x}_1(t);\ \ddot{x}_2(t); \ldots;\ \ddot{x}_n(t)]$ is corresponding to last *n* values of the output. Finally, by solving the ODE, the state of the system $\mathbf{h}(t)$ in the next time can be obtained. At the same time step, the MSE loss in Eq. 5-10 is computed by the difference between the measured response $\mathbf{h}_{\mathrm{true}}(t)$ and the generated response $\mathbf{h}(t)$ from PNODEs to update parameters of the parallel NN block. After

the above process, the PNODEs complete the solution at one time step, and $\mathbf{h}(t)$ is regrouped again and input to the PNODEs block to generate solution along time series.

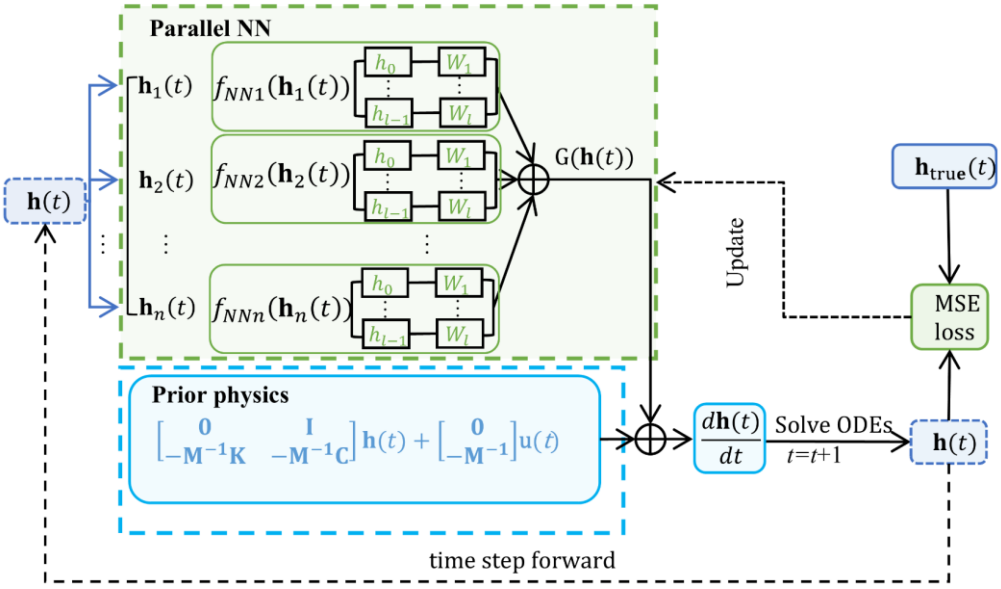


Figure 5-2. Phase II. Architecture of PNODEs

5.3.8. Inference of the parallel NN block

After the parallel NN block is trained, the structural physical parameters are fitted and determined from each NN for structural damage identification. The *i*th neural network $f_{NNi}(\mathbf{h}_i(t), t, \theta)$ is parameterised by weight θ, which can be represented by the chain structure with $L+1$ layers.

$$f_{NNi}(\mathbf{h}_i): \begin{cases} \mathbf{h}_i{}^1 = \sigma^1(\mathbf{W}^1\mathbf{h}_i{}^0 + b^1) \\ \vdots \\ \mathbf{h}_i{}^l = \sigma^l(\mathbf{W}^l\mathbf{h}_i{}^{l-1} + b^l) \\ \vdots \\ \mathbf{h}_i{}^L = \sigma^L(\mathbf{W}^L\mathbf{h}_i{}^{L-1} + b^L) \end{cases} \quad \text{Eq. 5-15}$$

where $\mathbf{h}_i^l$ is the intermediate output hidden in the $l+1$ layer of the *i*th network, $\mathbf{h}_i^0$ and $\mathbf{h}_i^L$ are assigned as input and output vector of the ith network, respectively. $\mathbf{W}^l$ is the weight vector associated with linear transfer from the $(l-1)$ layer to the l layer. b^l is the bias in the l layer .

σ^l is the activation function at the l layer.

To reconstruct the structural parameters from the parallel NN block linking to the prior physics block, the identification of variation in local stiffness value can be achieved by conducting the process below: (a) The discrepancy between the real dynamic and prior structural knowledge of the parallel NN block $\begin{bmatrix} f_{NN1}(\mathbf{h}_1(t)) \\ f_{NN2}(\mathbf{h}_2(t)) \\ \cdots \\ f_{NNn}(\mathbf{h}_n(t)) \end{bmatrix}$ can be computed from each network $\sigma(\mathbf{W}\mathbf{h}_i + b)$ from Eq. 5-15, then the parallel NN block can also be expressed as $\mathbf{W}(\mathbf{h}) = \begin{bmatrix} \sigma(\mathbf{W}_1\mathbf{h}_1 + b) \\ \sigma(\mathbf{W}_2\mathbf{h}_2 + b) \\ \cdots \\ \sigma(\mathbf{W}_n\mathbf{h}_n + b) \end{bmatrix}$; (b) The discrepancy term expression is fitted by establishing a regression problem, the neural network is then expressed by a matrix of the coefficients as shown below

$$\mathbf{W}(\mathbf{h}) = \begin{bmatrix} \mathbf{h}_1\boldsymbol{\Theta}_1 \\ \mathbf{h}_2(t)\boldsymbol{\Theta}_2 \\ \cdots \\ \mathbf{h}_n(t)\boldsymbol{\Theta}_n \end{bmatrix} \quad \text{Eq. 5-16}$$

where $\mathbf{h}_i(t)\boldsymbol{\Theta}_i$ is a polynomial function corresponding to the *i*th floor of the structure, followed by the prior physical for the structure. Then the closure approximation expression is consistently related to the known dynamical system as follows,

$$\frac{d\mathbf{h}(t)}{dt} = \mathbf{A}\mathbf{h}(t)) + \mathbf{B}\mathbf{u}(t) + \mathbf{h}(t)\boldsymbol{\Theta} \quad \text{Eq. 5-17}$$

5.3.9. Damage Index

After the structural parameters are determined, the damage index is introduced for structural damage localisation and qualification. Considering the structure under the health condition, the prior physics in Eq (12) is expressed as,

$$f_{\text{health}}(\cdot) = \frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}_{\text{health}} & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix}\mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix}\mathbf{u}(t) \quad \text{Eq. 5-18}$$

where $\mathbf{K}_{\text{health}}$ is a stiffness matrix for the structure of the health condition. Structural damage can be represented as changes of physical properties (i.e stiffness), which are treated as indicators of the structural damage location and severity (Radzienski et al. 2011; Maia 2003; Yan et al. 2019; Avci et al. 2020). Therefore, the damaged structure from Equation (18) is expressed as,

$$f_{\text{damage}}(\cdot) = \frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K}_{\text{health}} - \triangle\,\mathbf{K}) & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t)$$

$$= \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K}_{\text{health}}) & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}(-\triangle\,\mathbf{K}) & \mathbf{0} \end{bmatrix} \mathbf{h}(t) \qquad \text{Eq. 5-19}$$

where $\triangle\,\mathbf{K}$ is a matrix representing of change in stiffness of the damaged structure. Eq. 5-15 represents the health structure system that has been put into the prior physics block as the referenced physical knowledge. Therefore, the expression of the model discrepancy between health condition and damage condition $f_{\text{health}}(\cdot)$-$f_{\text{damage}}(\cdot)$ is expressed as

$$\boldsymbol{f}_{\triangle\boldsymbol{K}}(\cdot) = \frac{d\mathbf{h}_{\triangle\boldsymbol{K}}}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}(\triangle\,\mathbf{K}) & \mathbf{0} \end{bmatrix} \mathbf{h}(t) \qquad \text{Eq. 5-20}$$

Compared with Eq. 5-17, Eq. 5-19 can be rewritten as,

$$\frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K}_{\text{health}}) & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t) + \mathbf{h}(t)\boldsymbol{\Theta} \qquad \text{Eq. 5-21}$$

The discrepancy term $f_{\triangle\boldsymbol{K}}(\cdot)$ of the structural system is stored in the Parallel NN block $\mathbf{h}(t)\boldsymbol{\Theta}$ (it is obtained by the reconstruction task introduced in Section 2.2.5 process (a) and (b)). Therefore, the $\mathbf{h}(t)\boldsymbol{\Theta}$ represents the model discrepancy from Eq. $5-16$ and it could be expressed as,

$$\begin{bmatrix} \mathbf{0}_{n\times 1} \\ \mathbf{h}_1(t)\boldsymbol{\Theta}_1 \\ \mathbf{h}_2(t)\boldsymbol{\Theta}_2 \\ \cdots \\ \mathbf{h}_{n-1}(t)\boldsymbol{\Theta}_{n-1} \\ \mathbf{h}_n(t)\boldsymbol{\Theta}_n \end{bmatrix} = \frac{d\mathbf{h}_{\triangle\boldsymbol{K}}}{dt} = \begin{bmatrix} \mathbf{0}_{n\times 1} \\ -(\frac{\triangle k_1}{m_1} + \frac{\triangle k_2}{m_1})x_1(t) + \frac{\triangle k_2}{m_1}x_2(t) \\ \frac{\triangle k_2}{m_2}x_1(t) - \left(\frac{\triangle k_2}{m_2} + \frac{\triangle k_3}{m_2}\right)x_2(t) + \frac{\triangle k_3}{m_2}x_3(t) \\ \cdots \\ \frac{\triangle k_{n-2}}{m_{n-1}}x_{n-2}(t) - \left(\frac{\triangle k_{n-1}}{m_{n-1}} + \frac{\triangle k_n}{m_{n-1}}\right)x_{n-1}(t) + \frac{\triangle k_n}{m_{n-1}}x_n(t) \\ \frac{\triangle k_n}{m_n}x_{n-1}(t) - \frac{\triangle k_n}{m_n}x_n(t) \end{bmatrix} \qquad \text{Eq. 5-22}$$

To reconstruct the stiffness parameter change of the damaged structural system to explore the damage location and severity, each network $\mathbf{h}_i(t)\mathbf{\Theta}_i$ should be interpreted. Thus, considering the *i*th intermediate non-zero equation, multiplying m_i, the structural restoring force can be obtained as $m_i \frac{d\mathbf{h}_{i\triangle \boldsymbol{k}}}{dt} = -\triangle \mathrm{k}_i(x_i(t) - x_{i-1}(t)) + \triangle \mathrm{k}_{i+1}(x_{i+1}(t) - x_i(t))$. Therefore, each $\triangle \mathrm{k}_i$ can be estimated by solving the corresponding equation group with respect to $x_i(t) - x_{i-1}(t)$ on right hand side of Eq. 5-22. Then, $\triangle \mathrm{k}_i$ can be obtained by the element of $\mathbf{\Theta}_i$ times m_i. Finally, the damages index is defined as

$$\alpha_i = \frac{\triangle \mathrm{k}_i}{\mathrm{k}_i} \qquad \text{Eq. 5-23}$$

where k_i is the stiffness value of the i*th* floor in the health condition and $\triangle \mathrm{k}_i$ is the fitted stiffness change. The damages index is later used to indicate the damage.

5.4. Numerical study

In this section, a three-story building structure is used to verify the performance of the proposed method for structural damage identification. The physical properties of the building structure are introduced first. Then, the data preparation process is introduced, followed by the PNODEs architecture, including the prior physical knowledge and the neural network parameters. Subsequently, the proposed method is used for structural damage identification.

5.4.1. Numerical model

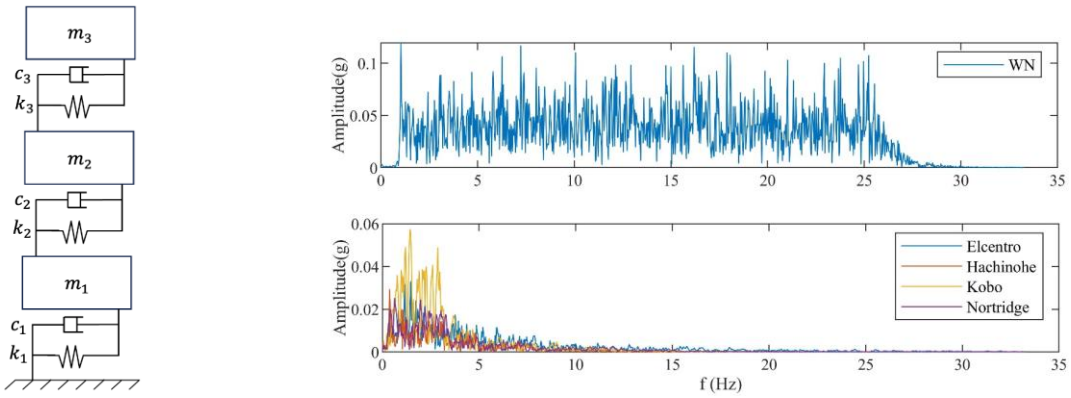
A three-storey building structure is used as an example in this section. The stiffness and mass of each floor are k_i=9.755$\times 10^3$ N/m and $m_i = 4.85$ kg respectively. The Rayleigh damping is introduced as $c = \alpha m + \beta k$, in which $\alpha = 0.016$ and $\beta = 2.136e-4$. The model has two identical columns with a 50 mm $\times$ 3 mm cross-section, a 900 mm length, and three steel floors

measuring 394 mm × 50 mm × 30 mm. The story height is 300 mm. The material used to build all columns and floors is high-strength steel with a yield stress of 435 MPa and an elasticity modulus of 200 GPa. The three-storey building structure is modelled as lumped mass model with three degrees of freedom (DOFs), as shown in Figure 5- 3(a). The three natural frequencies of the model are 3.195 Hz, 8.951 Hz and 12.935 Hz, respectively. The equations of the motion for the model are obtained as (Zhan et al., 2014),

$$\begin{cases} u_g(t) - m_1\ddot{x}_1(t) = (k_1 + k_2)x_1(t) - k_2x_2(t) + (c_1 + c_2)\dot{x}_1(t) - c_2\dot{x}_2(t) \\ -m_2\ddot{x}_2(t) = -k_2x_1(t) + (k_2 + k_3)x_2(t) - k_3x_3(t) - c_2\dot{x}_1(t) + (c_2 + c_3)\dot{x}_2(t) - c_3\dot{x}_3(t) \\ -m_3\ddot{x}_3(t) = -k_3x_2(t) + k_3x_3(t) - c_3\dot{x}_2(t) + c_3\dot{x}_3(t) \end{cases}$$

Eq. 5-24

where x_i $\dot{x}_i$ and $\ddot{x}_i$ represent the displacement, velocity and acceleration responses of the ith floor. u_g is the ground excitation. The state vector $\mathbf{h}(t) \in \mathbb{R}^{6\times1}$ is expressed as $\mathbf{h}(t) = [x_1(t); x_2(t); x_3(t); \dot{x}_1(t); \dot{x}_2(t); \dot{x}_3(t)]$.



(a) Lumped mass model (b) Spectra of WN and earthquake excitations

Figure 5-3. Building structure and its excitations

5.4.2. Data preparation

The ground excitation **u**(t) is simulated by four earthquakes, e. i., El Centro (E1), Hachinohe (E2), Kobe (E3) and Northridge (E4), and white noise (WN) as shown in Figure 5- 3(b). The sampling

rate is 100 Hz sampling rate. As mentioned above, the highest frequency of the three-story model is 12.935 Hz and a sampling rate of 100 Hz is sufficient to obtain the features of all the modes of the model from the data. In terms of frequency band of each excitation, that of the earthquake recordings are between $0-5\ Hz$ due to their natural characteristic, and that of WN is set between $1-25\ Hz$ as shown in Figure 5- 3(b). Then, the state vector $\mathbf{h}(t)$ of the structural system are formed by the displacement and velocity responses. $\mathbf{h}(t)$ is regrouped as $\mathbf{h}_1(t) = [0; x_1(t); x_2(t); 0; \dot{x}_1(t); \dot{x}_2(t)]$, $\mathbf{h}_2(t) = [x_1(t);\ x_2(t); x_3(t); \dot{x}_1(t);\ \dot{x}_2(t); \dot{x}_3(t)]$ and $\mathbf{h}_3(t) = [x_2(t);\ x_3(t); 0; \dot{x}_2(t);\ \dot{x}_3(t); 0]$ to supervise the PNODEs in Phase II. For the length of training data, a 32 s state vector $\mathbf{h}_i(t)$ containing 3200 data points is used.

5.4.3. PNODEs architectures

In the Prior physics block, the structure under health condition is given with the known ground excitation $\mathbf{u}(t)$. It can be expressed as

$$\frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M^{-1}K} & -\mathbf{M^{-1}C} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M^{-1}} \end{bmatrix} \mathbf{u}(t) = \boldsymbol{f}_{\text{health}}(\cdot) + \mathbf{Bu}(t) \qquad \text{Eq. 5-25}$$

where $\mathbf{0}_{3\times1}$ is a zero-matrix vector $\mathbb{R}^{\mathbf{3}}$; The mass matrix $\mathbf{M}$, stiffness matrix $\mathbf{K}$ and damping matrix $\mathbf{C}$ are given.

$$\mathbf{K} = 10^3 \begin{bmatrix} 19.509 & 9.755 & 0 \\ -9.755 & 19.509 & -9.755 \\ 0 & -9.755 & 9.755 \end{bmatrix} (N/m); \mathbf{M} = \begin{bmatrix} 4.8 & 0 & 0 \\ 0 & 4.8 & 0 \\ 0 & 0 & 4.8 \end{bmatrix} (\text{kg});$$

$$\mathbf{C} = \begin{bmatrix} 4.244 & 4.244 & 0 \\ -2.084 & 4.244 & -2.084 \\ 0 & -2.084 & 2.161 \end{bmatrix} (Ns/mm)$$

In the Parallel NN block, the discrepancy $NN(\mathbf{h}(t))$ from Eq. 5-14 of the structural system is

learned by three parallel neural networks $\begin{bmatrix} \mathbf{0}_{3\times1} \\ NN_1(\mathbf{h}_1(t)) \\ NN_2(\mathbf{h}_2(t)) \\ NN_3(\mathbf{h}_3(t)) \end{bmatrix}$ ($NN_i: \mathbb{R}^6 \longrightarrow \mathbb{R}^1, i = 1, 2\ and\ 3$) representing three floors of the structure. The input for each network is the state vector $\mathbf{h}_i(t) = [x_{i-1}(t);\ x_i(t); x_{i+1}(t); \dot{x}_{i-1}(t);\ \dot{x}_i(t); \dot{x}_{i+1}(t)]$, and the output of PNODEs is the acceleration vector $[\ddot{x}_1(t);\ \ddot{x}_2(t); \ddot{x}_3(t)]$. To accurately capture the discrepancy term, the $NN_i(\cdot)$ is designed with two layers, a hyperbolic tangent function $tanh(\cdot)$ is applied as an activation function $\sigma(\cdot)$. Based on the Eq. 5-10, the $NN_i(\mathbf{h}(t))$ can be represented as

$$\begin{aligned} NN_i(\mathbf{h}_i(t)) &= \mathbf{W}^2[\sigma^1(\mathbf{W}^1\mathbf{h}^0 + \boldsymbol{b}^1)] + \boldsymbol{b}^2 \\ &= \boldsymbol{w}^{[2]T}\left[\sigma^1\left(\boldsymbol{w}^{[1]T}\mathbf{h}^0 + b^{[1]}\right)\right] + b^{[2]} \end{aligned} \quad \text{Eq. 5-26}$$

where $\boldsymbol{w}$ and $\boldsymbol{b}$ is the matrix representation for weight vector and bias respectively; The number of neurons assumed in the hidden layer is equal to 10, thus dimensions of matrices are $\boldsymbol{w}^{[1]} \in \mathbb{R}^{6\times10}$; $\boldsymbol{b}^{[1]} \in \mathbb{R}^{10}$; $\boldsymbol{w}^{[2]} \in \mathbb{R}^{10\times1}$; $\boldsymbol{b}^{[2]} \in \mathbb{R}^{1}$. The architecture of NN_i and training hyper-parameters are listed in Table 5-1.

Table 5-1. Architecture of the network and its training hyper-parameters

Architecture of each neural network NN_i			Training hyper-parameters	
Layer type	Size	Number of neurons	Name	Value
Input	6		Batch size	200
Layer 1		10	Epoch	2e5
Layer 2		10	Learning rate	1e-5
Output	1	1		

5.4.4. Results

5.4.4.1. Structural damage identification with measurement noise

In practice, the measurement generally contains some noise. Thus, this section will discuss the performance of the proposed method for damage identification with measurement noise. Firstly, the

damage is introduced on the first floor by a stiffness reduction of 10%. The state space equations of the 3DOF structure system with the first-floor 10% damaged are:

$$\frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M^{-1}K} & -\mathbf{M^{-1}C} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\frac{0.1k_1}{m_1} x_1(t) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M^{-1}} \end{bmatrix} \mathbf{u}(t) = \boldsymbol{f}_{\text{damage}}(\cdot) + \mathbf{B}\mathbf{u}(t)$$

Eq. 5-27

Then, the 3% and 5% Gaussian white noise are added to the responses to simulate the measurement data X_{noise} for the training data as $X_{noise} = X + \mathcal{N}(\mu, \sigma)$, where X is the noise-free data. $\mathcal{N}(\mu, \sigma)$ is the standard normal distribution of Gaussian noise. μ, σ are the mean value and standard deviation of the noise. In this case, μ is equal to 0; σ is equal to 0.05 and 0.1 for 5% and 10% noise respectively.

Figure 5-4 shows the identification results of the damage index calculated by $\triangle \mathrm{k}_n$ from each $NN_i(\mathbf{h}_i(t))$, and their relative errors summarised in Table 5-2. From Figure 5- 4, the identified result is very close to the true value when there is no measurement noise. When the measurement noise is 5% or 10%, the damaged floor can be clearly identified. For damage severity, with the noise level increases, the value of the identified damage index gradually far away from the true value. It can be observed that with an increased noise level, the $NN_i(\mathbf{h}_i(t))$ is still enabled to capture the damage location accurately, while the damage severities can be precisely identified with the lower noise level.

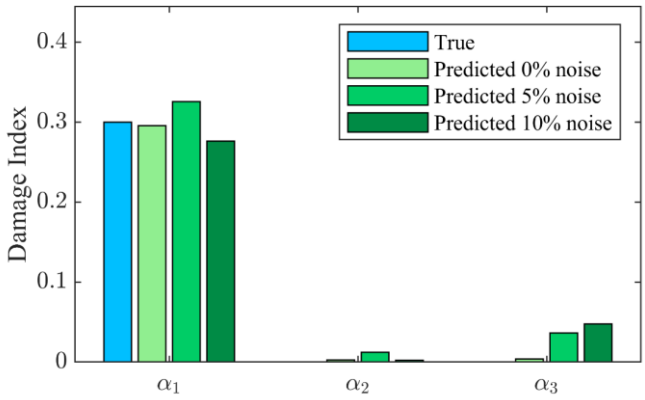


Figure 5-4. Identified results for Damage case 1M with different measurement noise.

Table 5-2. Relative errors of identified results for Damage case 1M with different measurement noise.

Damage Index	True	Identified					
		0% noise		5% noise		10% noise	
		PNODEs	Errors(%)	PNODEs	Errors(%)	PNODEs	Errors(%)
α_1	0.300	0.296	1.400	0.326	9	0.276	7.900
α_2	-	0.003	-	0.012	-	0.002	-
α_3	-	0.004	-	0.037	-	0.048	-

The output of $NN_i(\mathbf{h}_i(t))$ is the acceleration of the *i*th floor, and the restoring force can be obtained by multiplying by its mass. The force versus the relative displacement $(x_n(t) - x_{n-1}(t))$ for each floor is shown in Figure 5- 5 and it is compared with the true force from Eq. 5-26. Using the linear fitting, the stiffness change is determined, by the slope as $\triangle k_1 = 973.7N/m$. It is close to the true value of the stiffness change as $0.1\mathrm{k}_1 = 975.46N/m$. The result shows that the proposed method can efficiently quantify the structural damage given the a priori knowledge of the system under the health structure, and the PNODES can be sufficiently trained to accurately represent model discrepancies caused by the damage.

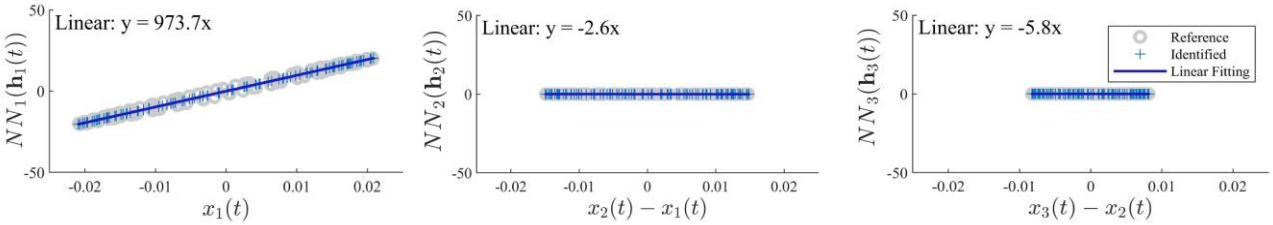


Figure 5-5. The relationship between $NN_i(\mathbf{h}_i(t))$ and relative displacement for each floor

5.4.4.2. Nonlinear structural system identification

Nonlinearities usually exist in civil structures, and it is usually caused by material, geometrical or boundary conditions (i. e. sliding surface, vibratory impact caused by semi-rigid connection, or elastic bodies) (Worden et al., 2008). In this study, the proposed method is to identify the nonlinear damage scenario in the structure under seismic excitations. The damage scenario is introduced by adding an additional nonlinear stiffness term $\frac{k_{nonlinear}}{m_1}{x_1}^3$, the state space equation of the nonlinear system are as,

$$\frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M^{-1}K} & -\mathbf{M^{-1}C} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_{nonlinear}}{m_1} {x_1}^3(t) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M^{-1}} \end{bmatrix} \mathbf{u}(t) \qquad \text{Eq. 5-28}$$

where $k_{nonlinear} = 2800 \times 10^3\ N/m$.

The relationship between the restoring force versus the relative displacement of each floor is shown in Figure 5-6. From the figure, the restoring forces for the second and third floors are close to zero and their stiffness changes are equal to zero. The restoring force for the first floor is nonlinear and the nonlinear stiffness can be determined by the polynomial fitting as $2908.2 \times 10^3\ N/m$ and the relative error between them is 3.864%. The result shows that the proposed method captures the nonlinear stiffness of structural system.

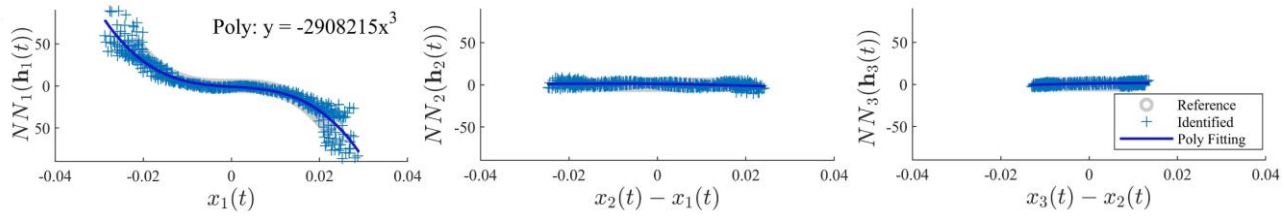


Figure 5-6. $NN_i(\mathbf{h}_i(t))$ versus each relative displacement for the nonlinear system.

5.5. An experimental study of a three-storey building structure

In this section, a 3-storey steel building model is established in the laboratory to further verify the proposed method. First, the experiment setup is introduced. Next, the experimental study of the building structure with different damage scenarios subjected different excitations are conducted on a shaking table. Finally, the feasibility of the proposed method in localizing damage is demonstrated.

5.5.1. Experimental setup

A 3-storey steel building frame was built and installed on the shake table at UTS Tech Lab. The experimental setup is shown in Figure 5- 7. The building structure is made of two spring steel strips and four mass blocks, the bottom mass block was fixed on a multi-axis shake table by two high-strength bolts. The accelerometers are installed on each floor to collect the acceleration responses and the lasers are installed beside the shake table to measure the structural displacement response. The shake table provides one-axis seismic ground motions made by the shake table controller. The first three natural frequencies of the experimental structure are 3.063 Hz, 8.620 Hz and 12.520 Hz, respectively.

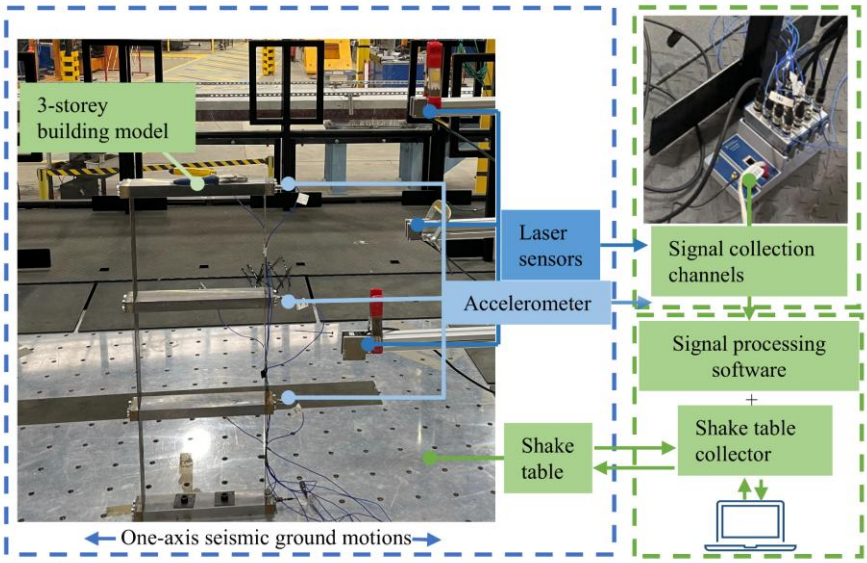


Figure 5-7. Experimental set up

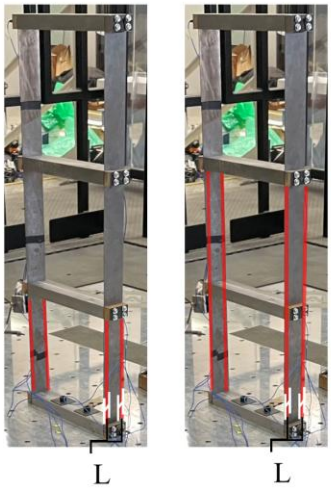


Figure 5-8. Damage models

Different damage scenarios are simulated through the saw cut on two columns as listed in Table 5-3. The steel columns on both sides of the model were cut symmetrically, and the location of the damage is shown in the red colour in Figure 5- 8. For each damage scenario, given the excitations, the displacement responses of the structure under different excitations are measured by laser sensors.

Table 5-3. Damage scenarios of the experimental frame structure.

Name	Single damage case	Name	Double damage case
1L	1st floor 10% damage; $L = 4.5mm$	1M2M	1st & 2nd floor 30% damage: $L = 9mm$
1M	1st floor 30% damage; $L = 9mm$		
1S	1st floor 50% damage; $L = 12.5mm$		

5.5.2. Data preparation

The ground excitation is the same as used in the numerical study, including four earthquakes (El, E2, E3 and E4) and WN, which were provided by the shake table to excite the experimental structure. The frequency band of the four earthquakes are between $0 - 5\ Hz$, and that of WN are set between $1 - 25\ Hz$ as shown in Figure 5- 3(b). The magnitude of all ground motions is scaled to $0.1g$ keep the structure remained in a linear behaviour. Lastly, the time series response **h**(t) is recombined into

$\mathbf{h}_i$ (i = 1, 2 and 3) for the input of the PNODE model.

5.5.3. Architecture of PNODEs

In the Prior physics block**,** the prior physics knowledge under the health condition corresponding to the undamaged experimental structure is simplified to a 3-DOFs numerical model. However, there are some differences due to the modelling error between the experimental structure and numerical model even both under health condition, these differences may be incorrectly identified as structural damage and captured by the parallel neural network, leading to inaccurate damage identification results. Thus, the numerical model is updated using the responses of the experimental structure under the health condition. After the updating process, the stiffness and damping matrix of the experimental structure of health condition $\mathbf{K}_{\text{Exp}}$ and $\mathbf{C}_{\text{Exp}}$ are obtained as prior physical knowledge summarised in Table 5-4 below.

Table 5-4. Updated stiffness and damping matrices for the experimental model.

Updated stiffness matrix	Updated damping matrix
$\mathbf{K}_{\text{Exp}} = \begin{bmatrix} 18.00 & -9.35 & -0.04 \\ -9.32 & 18.24 & -9.07 \\ -0.29 & -8.84 & 8.97 \end{bmatrix}$	$\mathbf{C}_{\text{Exp}} = \begin{bmatrix} 5.45 & 3.99 & -0.25 \\ -2.34 & 5.14 & -2.64 \\ -0.34 & -2.69 & 2.73 \end{bmatrix}$

In the parallel NN block**,** the state space equations are established by structural state vectors $\mathbf{h}(t) \in \mathbb{R}^{6\times1}$ as below,

$$\frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K}_{\text{Exp}}) & -\mathbf{M}^{-1}\mathbf{C}_{\text{Exp}} \end{bmatrix}\mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix}\mathbf{u}(t) + \begin{bmatrix} \mathbf{0}_{3\times1} \\ NN(\mathbf{h}(t)) \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K}_{\text{Exp}}) & -\mathbf{M}^{-1}\mathbf{C}_{\text{Exp}} \end{bmatrix}\mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix}\mathbf{u}(t) + \begin{bmatrix} \mathbf{0}_{3\times1} \\ NN_1(\mathbf{h}_1(t)) \\ NN_2(\mathbf{h}_2(t)) \\ NN_3(\mathbf{h}_3(t)) \end{bmatrix} \quad \text{Eq. 5-29}$$

where $NN_i(\mathbf{h}_i(t))$ (i = 1, 2, 3) represents the discrepancy between the undamaged and damaged structures. The architecture of each network is designed as same as the numerical study with 6 inputs

and 1 output introduced in Section 3.4.

5.5.4. Results

5.5.4.1. Effect of different excitations

Excitations may variously appear in practice. The performance of the proposed approach for damage identification under various excitations is further investigated in this section. Under different ground excitations E2, E3 and WN, the responses $\mathbf{h}(t)$ of the experimental structure with 30% damage (1M) of the first floor are recorded, respectively. Figure 5- 9 shows the identified damage indices compared with the true value, and the relative errors are shown in Table 5-5. They shows that the proposed PNODEs can accurately identify the damage severity of the first floor underground excitation E2 and E3, with results of 0.325 and 0.327 also with the relative errors of 9% and 8.333% , respectively. When the experimental model is under WN, it also can correctly localise the damaged location and identify the damage severity with a result of 0.343 and relative error of 14.333%. Besides, there is an error in the identified results for the second and third floors with the maximum damage index of 0.002.

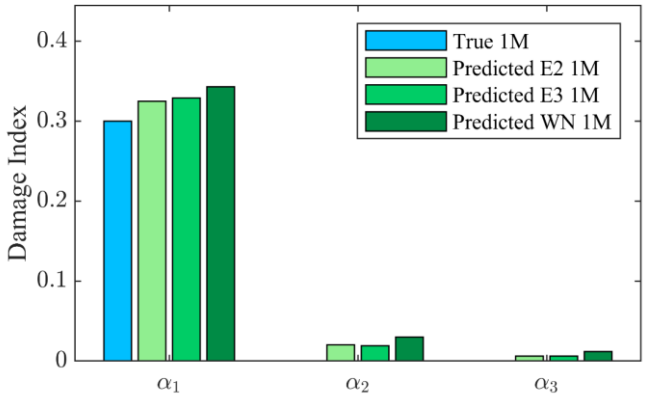


Figure 5-9. Identified results for Damage case 1M under excitations of E2, E3 and WN.

Table 5-5. Relative errors of identified results for Damage case 1M under different excitations.

Damage Index	True	Identified					
		E2		E3		WN	
		PNODEs	Errors (%)	PNODEs	Errors(%)	PNODEs	Errors(%)
α_1	0.300	0.325	8.333	0.327	9.000	0.343	14.333
α_2	-	0.002	-	0.019	-	0.030	-
α_3	-	0.006	-	0.006	-	0.012	-

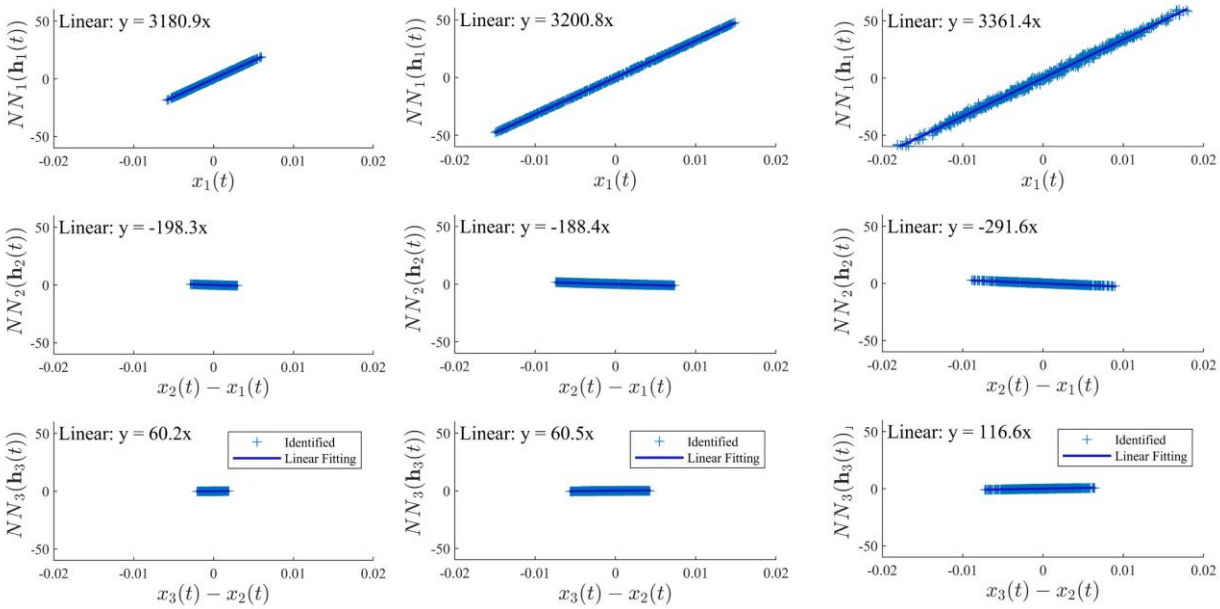


Figure 5-10. Output of trained $NN_i(\mathbf{h}_i(t))$ versus each relative displacement for Damage case 1M under E2, E3 and WN

The restoring force is extracted from $NN_n(\mathbf{h}_n(t))$ and plotted in Figure 5- 10. The linear fitting is used to determine the stiffness reduction from the restoring force. From Figure 5- 10, the identified stiffness reductions under E2, E3 and WN excitations are $\triangle k_1 = 3280.9\frac{N}{m}$, $3200.8\frac{N}{m}$ and $3361.4\frac{N}{m}$, respectively. The identified results are close to the true stiffness changes $0.3k_1 = 2793.8\ N/m$. The results show that the proposed method is robust to the environmental excitation.

5.5.4.2. Single damage identification

To explore the performance of the proposed method for structural damage identification, single damage with different severities are studied in this section. With the earthquake excitation (E1) is

applied, dynamic responses of the structure with 10%(1L), 30%(1M) and 50%(1S) damage on the first floor were used to test. Figure 5- 11 shows the identified results compared with the true value and Table 5-6 summarised the relative errors. Upon damage scenario of 1L, the damage index of the identified result 0.126 is slightly higher than the true value 0.1. Besides, the identified damage index of 1M and 1S are close to the true values. There are errors in identified results on the second and third floors with the maximum identified index of 0.03.

The restoring force is extracted from $NN_n(\mathbf{h}_n(t))$ and plotted in Figure 5- 12. The change of stiffness is obtained through the linear curve fitting. The identified stiffness changes are $\triangle k_1 = 1549.9\frac{N}{m}$, $3180.9\frac{N}{m}$ and $5359.1\frac{N}{m}$, respectively. There is a slight difference with the real stiffness changes $0.1k_1 = 931.3\ N/m$, $0.3k_1 = 2793.8\ N/m$ and $0.5k_1 = 4656.3\ N/m$ and this may be due to the modelling errors between the numerical model and real structure. The results show that the stiffness change can be well separated and accurately extracted for structural damage identification by the proposed method.

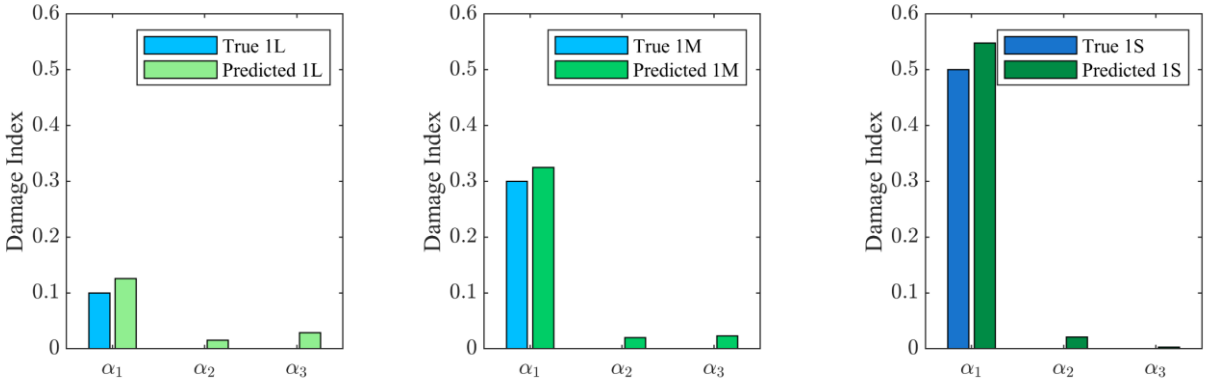


Figure 5-11. Identified results for Damage cases 1L, 1M and 1S.

Table 5-6. Relative errors of identified results for Damage case 1L, 1M and 1S.

Damage Index	1L			1M			1S		
	True	Identified		True	Identified		True	Identified	
		PNODEs	Errors (%)		PNODEs	Errors (%)		PNODEs	Errors (%)
α_1	0.100	0.126	25.800	0.300	0.325	8	0.500	0.548	9.600
α_2	-	0.015	-	-	0.020	-	-	0.021	-
α_3	-	0.029	-	-	0.023	-	-	0.003	-

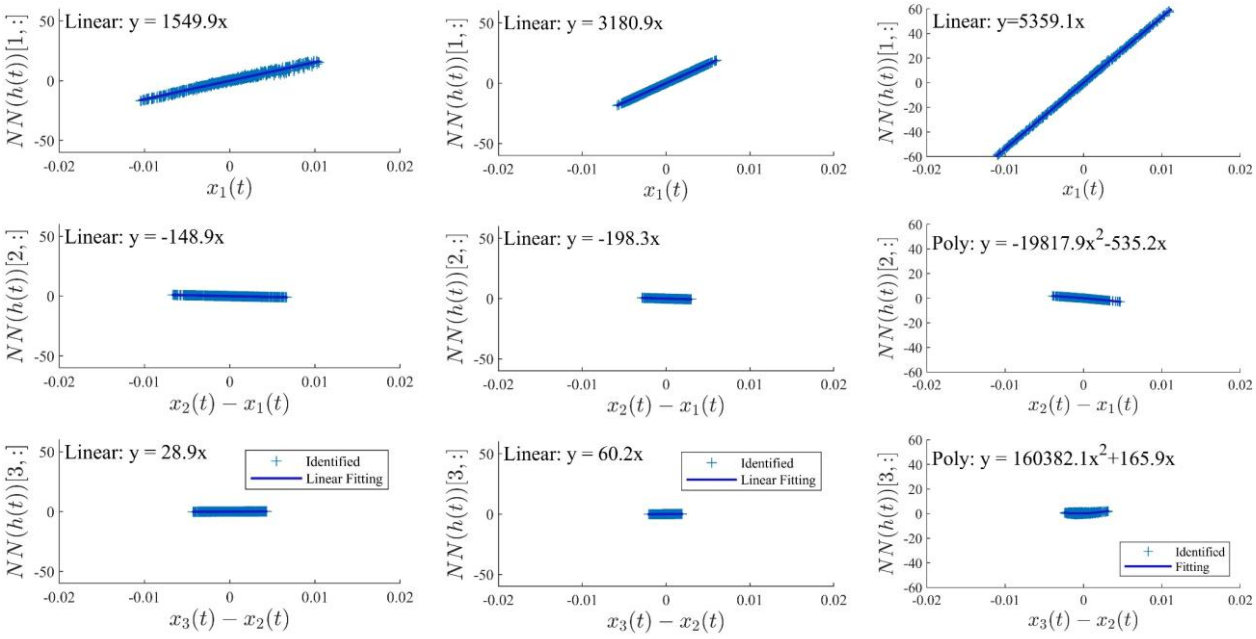


Figure 5-12. Relationship between $NN_i(\mathbf{h}_i(t))$ and relative displacement for damage scenarios 1L, 1M and 1S under the excitation E3.

5.5.4.3. Multiple damage identification

The section investigates the multiple damage identification using the proposed approach. The earthquake excitation E1 is applied on the shake table with two single damage cases 1M and 2M and one double damage case 1M2M. The damage case 1M is the damage severity 30% on the first floor and the damage case 2M is 30% damage on the second floor. The double damage case 1M2M is with

30% damage on both the first and second floors. Similar to Section 4.4.2, the identified results are shown in Figure 5- 13. From Figure 5- 13, the identified results are 0.325 and 0.281 for Cases 1M and 2M respectively, and 0.325 and 0.311 for Case 1M2M. Theie relative errors are calculated in Table 5-7. The results show that the proposed method could identify both single and double damage accurately. The restoring force is extracted from $NN_i(\mathbf{h}_i(t))$ and plotted in Figure 5- 14 and the stiffness changes are determined by the linear curve fitting. The predicted stiffness reductions are $\triangle k_1 = 3180.9\frac{N}{m}$, $\triangle k_2 = 2756.6\frac{N}{m}$ for Cases M1 and M2 respectively. For the double damage case, the identified results are $\triangle k_1 = 2267.8\frac{N}{m}$, $\triangle k_2 = 1601.4\frac{N}{m}$. The true stiffness changes are $0.3k_1 = 0.3\ and\ k_2 = 2793.8\ N/m$. The results show that the proposed method could predict the stiffness changes for both single and double damage cases. For the multi damage case, the proposed method could accurately identify the damage location, and further improvement is needed for the damage severity.

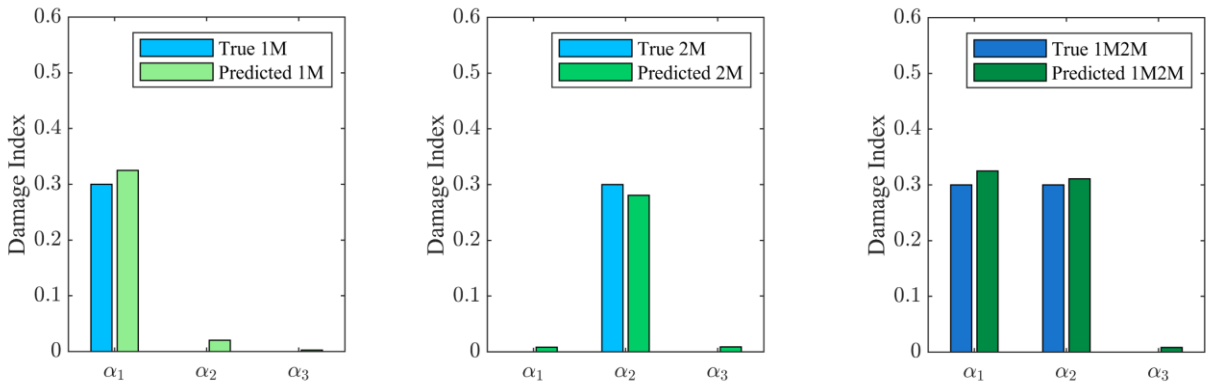


Figure 5-13. Identified results for Damage cases 1M, 2M and 1M2M.

Table 5-7. Relative errors of identified results for Damage case 1L, 1M and 1S.

Damage Index	1M			2M			1M2M		
	True	Identified		True	Identified		True	Identified	
		PNODEs	Errors (%)		PNODEs	Errors (%)		PNODEs	Errors (%)
α_1	0.300	0.325	8.333	-	0.008	-	0.300	0.325	8.333
α_2	-	0.020	-	0.300	0.281	6.333	0.300	0.311	3.667
α_3	-	0.002	-	-	0.009	-	-	0.008	-

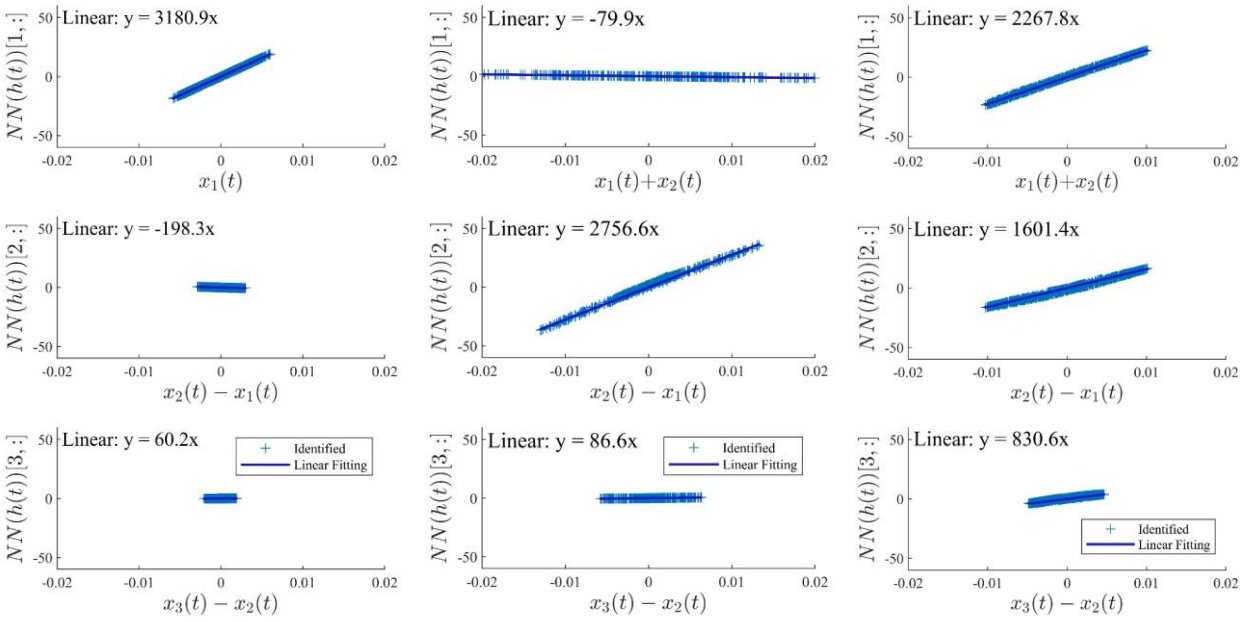


Figure 5-14. $NN_i(\mathbf{h}_i(t))$ versus relative displacement for Damage cases 1M, 2M and 1M2M

5.6. Application for three-dimensional frame structures

In this section, the IASC-ASCE benchmark frame is used to further verify the performance of the proposed method and the results are compared that by the NODEs method directly. The physical properties of the benchmark Frame is introduced first including sensor locations and damage scenarios. Then, the data preparation process is presented, following by the NODEs and PNODEs architecture and the prior physical knowledge and the neural network parameters. Lastly, the results of damage localisation and qualification by the proposed method and NODES are compared. Finally,

the practicality of using parallel neural network in PNODE for damage quantification is summerised.

5.6.1. Experimental structure information

The IASC-ASCE benchmark frame, the American Society of Civil Engineers (ASCE) benchmark structure (experimental Phase II) (Dyke et al., 2003), is shown in Figure 5- 15. The plane size of the structure is 2.5 meters square, and the height of each floor is 0.9 m. The sections of beams and columns are S75×11 and B100×9, respectively. Within each bay, the bracing system consists of two parallel, diagonally arranged threaded steel rods, each with a diameter of 12.7 mm. The braces are constructed from 300W grade hot-rolled steel. To achieve a realistically distributed mass, a single floor slab is installed within each bay at each floor. In detail, there are four slabs, each weighing 1000 kg, The slabs located at the first, second, and third floors are 1000 kg. The slab on the fourth floor is 750 kg. The details are illustrated in Figure 5- 15(b). The locations of accelerometers are depicted in Figure 5- 15(a). For each floor, three accelerometers are installed, e.g. two accelerometers on the y-axis ($\ddot{x}_{yna}$ and $\ddot{x}_{ynb}$) and one on the x-axis. Two sets of accelerometers $\ddot{x}_{yna}$ and $\ddot{x}_{ynb}$ in y-direction are divided into Groups A and B, respectively.

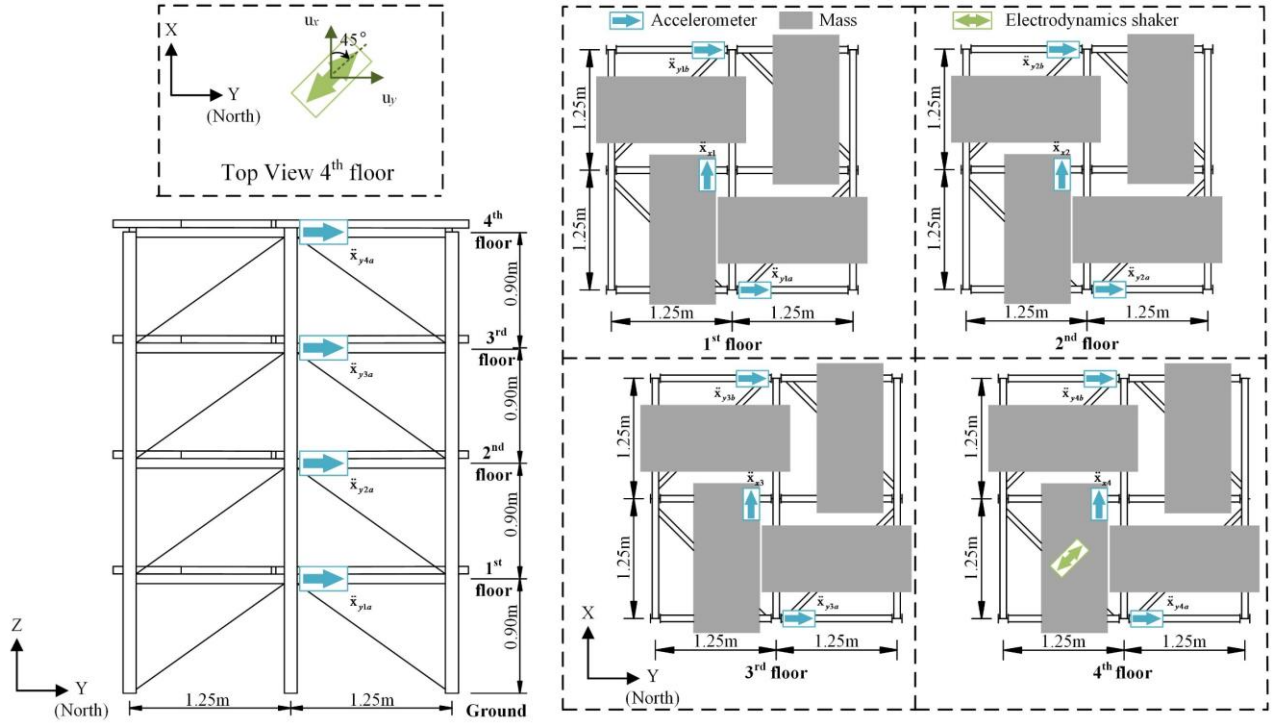


Figure 5-15. Sensor arrangements (a) front view and (b) top view.

As shown in Figs. 16(a) and 16(b), two damage scenarios have been simulated: the first damage scenario involves removing braces on the first floor in one bay at the southeast corner, and the second damage scenario involves removing braces on all floors on the eastern side and on the second-floor braces on the northern side.

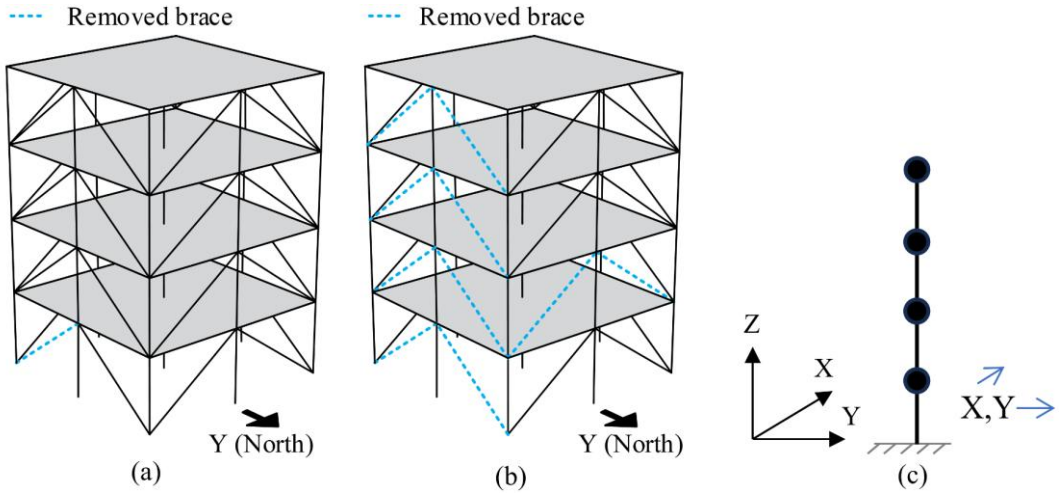


Figure 5-16. Damage scenarios: (a) removing braces on the first floor in one bay at the southeast corner and (b) removing braces on all floors on the eastern side and on the second-floor braces on the northern side. (c) the lumped mass model

The experimental data from this benchmark dataset comprises acceleration measurements gathered from the structure under various conditions: ambient conditions, excitation induced by a shaker, and excitation resulting from a force hammer. In addition, the natural frequency of the benchmark frame under the health condition are 3.5 Hz, 11.0 Hz, 17.5 Hz, 24.0 Hz, 29.5 Hz, 34.0 Hz, 37.0 Hz and 39.5 Hz from the measurement data. The detailed descriptions and experimental data for this experiment are available at https://datacenterhub.org/resources/257.

5.6.2. Data Preparation

In this work, the displacement and velocity data are necessary for input to the neural network, which is deviated from the measured acceleration data of the structure under the environment excitation. Two sets of displacement and velocity responses are obtained from $\ddot{x}_{yna}$ and $\ddot{x}_{ynb}$ along y direction, named as Groups A and B. The data from Groups A and B were fed into the neural network separately for damage identification, and an average of damage indexes from both data were used for damage qualification. The data was sampled at a frequency of 200 Hz, and a low pass filter with a cutoff frequency of 100 Hz was employed to ensure that the filtered data contains all the mode information of the experimental structure as mentioned above. Both the undamaged scenario and damage scenario 1 comprise 300 s of data each, whereas damage scenario 2 contains only 220 s of data.

The numerical model is updated using the measurement data from the undamaged structure. Then, the first 50 s of the state response $\mathbf{h}(t)$ from the two damage scenarios were used to supervised the PNODEs, and the same length of environment excitation $\mathbf{u}(t)$ is given to the ODEs for generating the response under the health condition. To identify the damage in this complex frame, the

neural network is designed as four individual neural networks, representing four floors of the frames. In each time step, $\mathbf{h}(t)$ is regrouped as $\mathbf{h}_i$ input to each neural network; where $\mathbf{h}_i = [x_{x(i-1)}; x_{y(i-1)}; x_{xi};\ x_{yi}; x_{x(i+1)};\ x_{y(i+1)};\ \dot{x}_{x(i-1)};\ \dot{x}_{y(i-1)};\ \dot{x}_{xi};\ \dot{x}_{yi};\ \dot{x}_{x(i+1)};\ \dot{x}_{y(i+1)}]$ (i = 1, 2, 3 and 4) contains the displacement and velocity responses relevant to that part of the structure it represents. Besides, the NODEs contained only one neural network is also used for damage identification using the input $\mathbf{h}(t)$. Table 5-8 shows the input data for each neural network in NODE and PNODEs. The results from the proposed method and NODEs are compared.

Table 5-8. Input data of NODEs and PNODEs.

NODEs		PNODEs	
Neural network	Input data	Neural network	Input data
$NN(\cdot)$	$\mathbf{h}(t)$	$NN_1(\cdot)$	$\mathbf{h}_1(t)$
		$NN_2(\cdot)$	$\mathbf{h}_2(t)$
		$NN_3(\cdot)$	$\mathbf{h}_3(t)$
		$NN_4(\cdot)$	$\mathbf{h}_4(t)$

5.6.3. PNODEs and NODEs architecture

In the Prior physics block, the prior physics knowledge under the health condition corresponding to the undamaged frame structure is simplified as a lumped mass model using 4 segments and 5 nodes with locations of accelerometers. Each node contains 2 DOFs, which indicate horizontal and translational DOFs along x and y directions shown in Figure 5- 16(c). As a result, each segment has 4 DOFs and 8 DOFs in total for the entire structure.

In the Parallel NN block**,** similar to Section 3.4, the discrepancy term $f_{\triangle \boldsymbol{K}}(\cdot)$ of the structural system is learned by the *n* neural network (NN_i: $\mathbb{R}^{12} \longrightarrow \mathbb{R}^2$). The input for the each network is the state vector $\mathbf{h}_i(t)$, and the output of the PNODE is the acceleration vector $[\ddot{x}_{x1}(t);\ \ddot{x}_{y1}(t); \ddot{x}_{x2}(t);\ \ddot{x}_{y2}(t);\ \ddot{x}_{x3}(t);\ \ddot{x}_{y3}(t); \ddot{x}_{x4}(t);\ \ddot{x}_{y4}(t)\ ]$. Thus, each NN_n represents the discrepancy of each floor of the frame including x and y directions. To further reconstruct the

structural parameters for structural damage identification, the $NN_n(\cdot)$ is designed with additional one layer being three layers for learning the discrepancy part related to experimental structure. The learning epoch is increased to 3e5 due to the complexity of the 3D frame. The results from the proposed method are compared with that by NODEs.

For the NODEs contained one whole neural network, the neural network $NN(\cdot)$ $(\mathbb{R}^{16} \longrightarrow \mathbb{R}^{8})$ is designed with 16 displacement and velocity responses $\mathbf{h}(t)$ of the entire structure as input and 8 acceleration vectors as outputs with three layers, in which the numbers of neuron in each layer are doubled comparing with the previous cases due to the increase of input and output sizes. Both the NODEs and PNODEs are solved using the same hyper-parameters for comparison. The details for each neural network architecture of in NODEs & PNODEs and training hyper-parameters are shown in Table 5-9.

Table 5-9. Architecture of neural network in NODEs & PNODEs, and training hyper-parameters.

Architecture of neural network $NN(\cdot)$ In NODE			Architecture of each neural network $NN_i(\cdot)$ in PNODEs			Training hyper-parameters	
Layer type	Size	Number of neurons	Layer type	Size	Number of neurons	Name	Value
Input	16		Input	16		Batch size	200
Layer 1		40	Layer 1		20	Epoch	3e5
Layer 2		40	Layer 2		20	Learning rate	1e-5
Layer 3		40	Layer 3		20		
Output	8		Output	2			

5.6.4. Results

5.6.4.1. Single damage identification

The true values of the changes in the stiffness of the frame structure were obtained from the results of the previous experiments (Johnson, 2003). The NODE and PNODE based methods were used and their results were compared with the true stiffness loss, as shown in Figure 5- 17. The relative

errors between the true and identified damage index are summarised in Table 5-10. From Figure 5-17, the identified result by the proposed PNODEs is much closer to the true value than that by NODEs. The results show that the damage location can be better separated from the discrepancy information using a parallel neural network. There are some errors in the identified results on the 1st floor & 2nd floor along the y direction and 4th floor along the x direction by NODEs. For the damage severity, the loss of stiffness of the 1st floor along the x direction is 0.051 by NODEs with 54.956% error, while the true value is 0.113. By the proposed PNODEs, there are identified damage index of 0.025 and 0.013 in the x-direction on Floors 2 and 4 respectively. The identified result in the x-direction on the 1st floor by the proposed method is 0.109 with only 3.363% errors, which is much closer to the true value compared with that by NODEs.

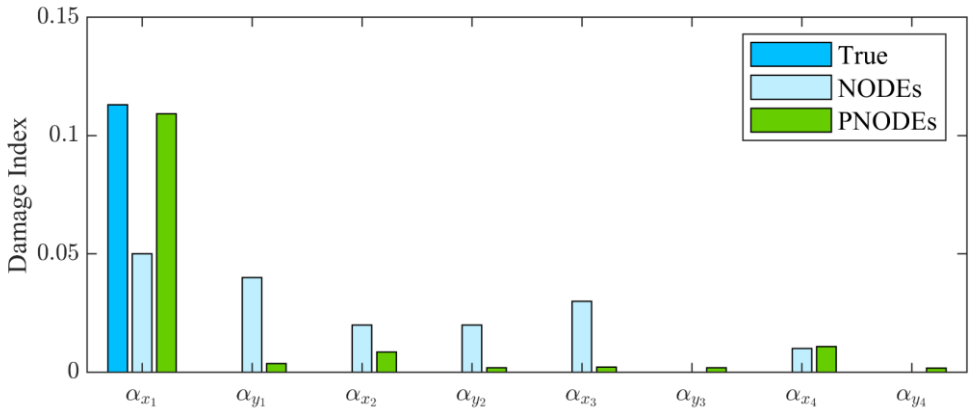


Figure 5-17. Identified results for Damage scenario 1 of the experimental structure.

Table 5-10. Relative errors of identified results for Damage scenario 1.

Damage Index	True	Identified			
		NODEs	Errors (%)	PNODEs	Errors (%)
α_{x1}	0.113	0.051	54.956	0.109	3.363
α_{y1}	-	0.047	-	0.004	-
α_{x2}	-	0.025	-	0.009	-
α_{y2}	-	0.024	-	0.002	-
α_{x3}	-	0.033	-	0.002	-
α_{y3}	-	0.001	-	0.002	-
α_{x4}	-	0.013	-	0.011	-
α_{y4}	-	0.004	-	0.002	-

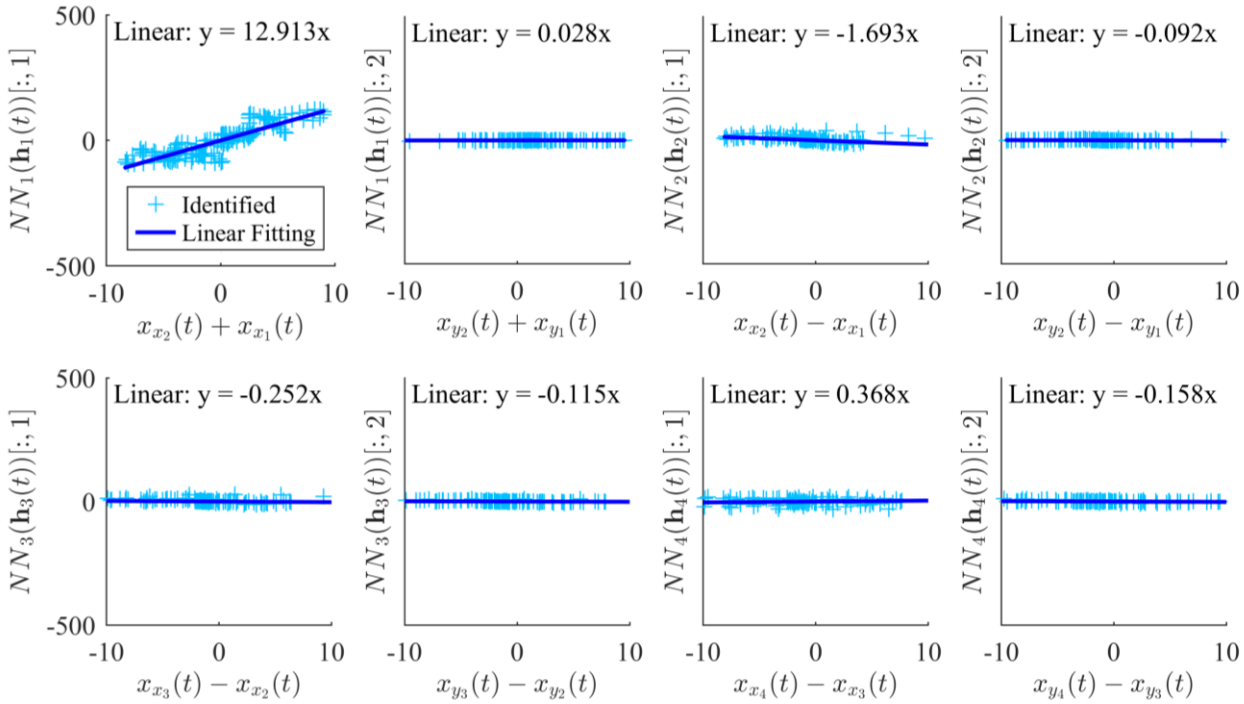


Figure 5-18. The relationship between $NN_n(\mathbf{h}(t))$ and relative displacement for Damage scenario 1.

To gain insight into the $NN_n(\cdot)$in PNODEs, the restoring forces in both directions for each floor are shown in Figure 5- 18. Since each $NN_n(\cdot)$ represents the discrepancy term between the damage and health states for each floor of the structure, a linear fitting between the displacement and restoring force relationship is used to determine the stiffness change Δk_n of each floor using Eq. 5-21 for structural damage identification. From the first plot in Figure 5- 18, there is approximate linear relationship between the relative displacement and restoring force and the slope is used to determine

the stiffness change in the x-direction of the first layer, resulting in Δk_{1x} =12.913kN/m. Overall, the proposed PNODEs improved the damage identification results compared with the NODEs for a signal damage in a 3D frame structure.

5.6.4.2. Multiple damage identification

In practice, it may be much complex with multiple damage appearing in different locations. Introducing multiple damage in different locations of the frame structure makes it even more challenging. In this section, the same amount of damage data from Damage case 2 is used for structural damage identification by the proposed PNODEs and NODEs. Figure 5- 19 shows identified results by these two methods and relative errors are summerised in Table 5-11. From Fig 19, the identified results with a maximum error of 8.840% by the proposed PNODEs method with a maximum error of 30.946% are much closer to the true values than those by NODEs. For both the damage localisation and quantification, the proposed PNODEs method outperforms the NODEs method. The NODEs method incorrectly determines the presence of damage in the Y-direction in the first and third floors of the structure, as the damage indexes representing the stiffness change are significant at these two locations. In other damage locations that can be correctly identified by NODEs, the severity of the damage cannot be accurately quantified, particularly in the y-direction of the second floor which has a very high damage severity. On the other hand, the PNODEs method localises the damage location correctly, except for the y-direction of the first floor. The identified damage severities by the PNODEs method are close to the true values. The results show that the PNODEs method can accurately quantify the damage severities of multiple damage. Overall, the proposed PNODEs method significantly increases the damage identification accuracy for the multiple

damage case compared with the NODEs method.

To further explain each network, the input and output of each $NN_n(\cdot)$ in PNODEs have also been plotted in Figure 5-20. Each plot in the figure represents the discrepancy term relative to each part of the frame structure, where the y-axis is the restoring force of the $NN_n(\cdot)$ output and the x-axis is the measured relative displacement. Similar to the single damage identification, the stiffness change Δk_n in the x and y directions with respect to each floor can be determined by linear curve fitting. From the figure, the discrepancy between the health and damage states in each part of the frame structure is captured separately by its relative neural network.

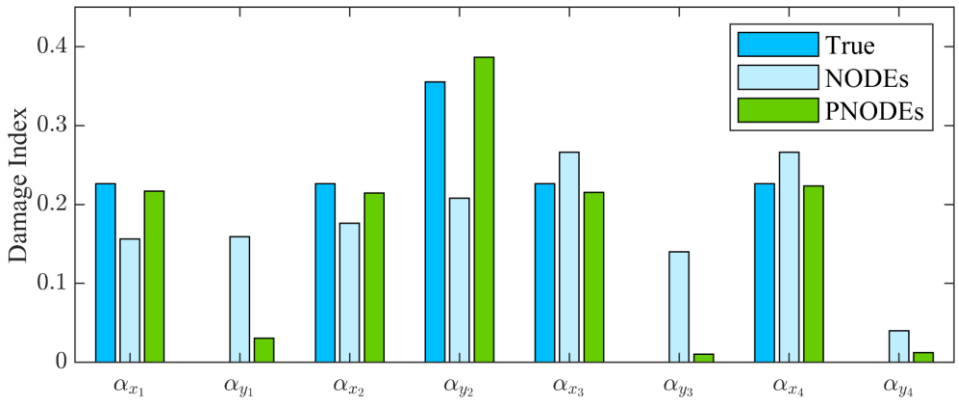


Figure 5-19. Identified results for Damage scenario 2 of the experimental structure

Table 5-11. Relative errors of identified results for Damage scenario 2.

Damage Index	True	Identified			
		NODEs	Errors (%)	PNODEs	Errors (%)
α_{x1}	0.226	0.156	30.946	0.217	4.070
α_{y1}	-	0.159	-	0.030	-
α_{x2}	0.226	0.176	22.104	0.214	5.261
α_{y2}	0.355	0.208	41.413	0.387	8.840
α_{x3}	0.226	0.266	17.683	0.215	4.863
α_{y3}	-	0.140	-	0.010	-
α_{x4}	0.226	0.266	17.595	0.224	1.194
α_{y4}	-	0.040	-	0.012	-

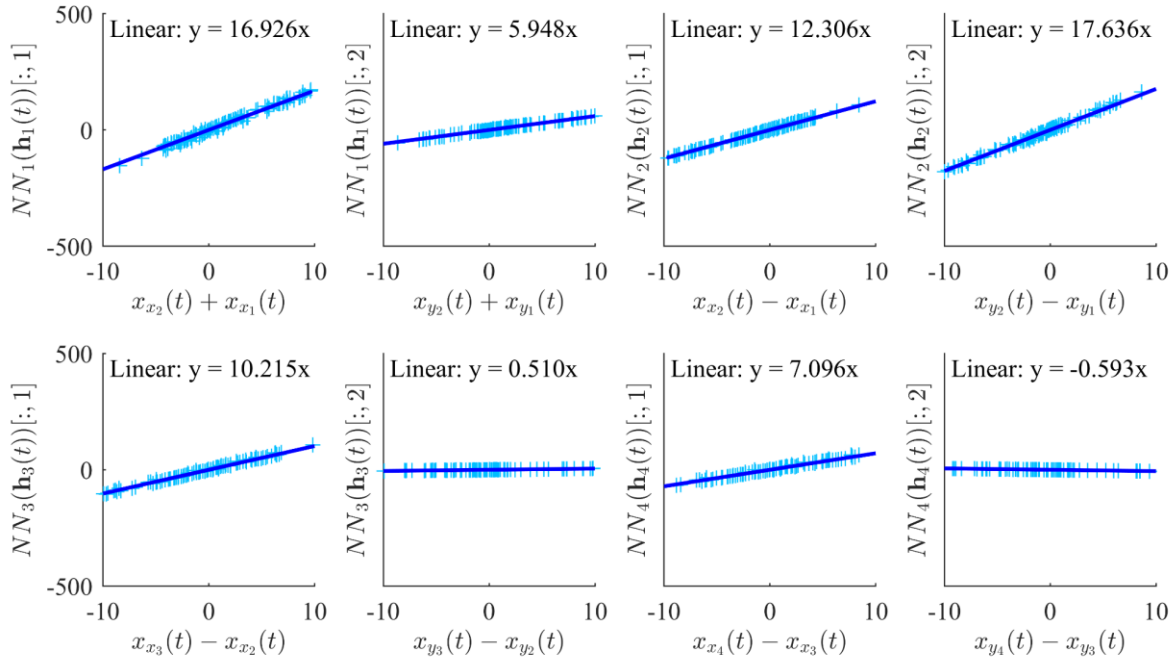


Figure 5-20. Output of $NN_i(\mathbf{h}_i(t))$ versus each relative displacement for Damage scenario 2

5.7. Summary

A new PNODEs method based on parallel neural networks has been developed for structural damage localisation and quantification. The proposed method is made by a priori physical model of structural dynamical system embedded with a group of parallel neural networks. This set of parallel neural networks efficiently separates the discrepancy terms that represent damage information for the entire structure. The damage is then accurately quantified based on the information in each network. In addition, the interpretability of the proposed method can be achieved by coupling the implementation of an identification scheme for model parameters that can extract expressions from derived each neural network terms that are consistent with each part of a structural model, thereby improving the interpretability of how the model works, overcome the issue of unexplained approximations provided by traditional black-box models. This enhances the accuracy and reliability of this damage identification method. The three-storey building structure and the three-dimensional

IASC-ASCE benchmark frame have been used to verify the effectiveness of the proposed method for damage localisation and quantification. Several findings can be concluded:

The PNODEs method requires only a small amount of damage state measurement data to solve parallel neural networks without the need for health state measurement data. The discrepancy term between the health and damaged structure is represented by these neural networks and are used to perform damage identification to obtain accurate identification results.

Existing studies on NODEs for structural damage identification can only store the entire discrepancy information into a single neural network but cannot find the source of the discrepancy caused to achieve precise localisation and quantification. The proposed method is based on the structural dynamic representation, where each neural network is fed by its associated dynamic response and separates the entire discrepancy term based on each part of the structure to overcome the problem of the confusion caused by the entire discrepancy being stacked together. Each neural network in parallel networks is utilised in PNODEs to represent a part of the structure and can find the discrepancy in each part of the structure for damage localisation.

In the purposed PNODEs, damage quantification is converted into a problem of reconstructing structural parameters using a neural network. Each neural network in PNODEs is solved to derive the restoring force for each section of the structure separately to reconstruct its stiffness change. This further makes the neural network transparent to improve interpretability.

The proposed PNODEs method requires the response information of all relevant DOFs. Due to the limitation of sensors in practice, only part of the DOFs data can be measured. Future work will investigate to utilise the limited number of measurements for structural damage identification.

Chapter 6. Spatial-temporal-based physics neural operators for response reconstruction and structural identification with limited sensors

6.1. Overview

In recent years, deep learning-based data-driven techniques have a wide range of applications in structural system identification, which can reconstruct structural parameters based on a large amount of data. However, in most practical engineering cases, large amounts of high-quality data are usually hard to obtain. Particularly when the limited sensors are placed only in some of the degrees of freedom of the structure, limited observation is obtained, which causes the identification of structural parameters to become difficult. To address this challenge, this chapter proposes a novel temporal-spatial neural operator (PhySTN) framework that allows limited observations to be used directly for structure identification. This framework consists of a spatial feature mapping encoder and a time operator. The spatial encoder describes a spatial representation of the relationship between the full state and the limited observation. The time operator consists of simplified physical knowledge and a neural network, where the neural network is used to describe the complex discrepancy terms between physical knowledge and the real structure. The temporal operator, assisted by the spatial mapping encoder, allows a limited number of observations to be used directly to reconstruct the discrepancy terms. With limited observation training, a neural network constrained by the laws of physics can be used for structural parameter identification, and the time operator can also be used to reconstruct the missing response. The performance of the proposed framework is evaluated through a nonlinear numerical model and an experimental frame structure with magnetorheological elastomer isolator validation based on cases with different noise levels, different numbers of observations, and different

locations. The results show that the purposed framework can accurately identify the nonlinear structural parameter and reconstruct missing response.

6.2. State of the art

System identification (SI) is a process of inverse physical modelling of a real structural system and determining modal parameters (e.g natural frequency and modal shape) and physical properties (e.g. stiffness or damping ratio) utilising the measured dynamic response from real structures, which represents the dynamic characteristics of the real model to increase the reliability in condition assessment and residual life prediction and widely approaches in Structural Health Monitoring (SHM)(Kerschen et al., 2006; Alvin et al., 2003; Huang et al., 2017; Ebrahimian et al., 2018).

Generally, SI includes model-based and data-driven methods. The model-based methods focus on updating the model parameters to identify the system condition changes based on measurement data with its physical interpretability. Some methods have been carried out, including the Bayesian updating algorithm and Kalman filter methods(Eftekhar Azam et al., 2015; Hou et al., 2019). The effectiveness of these methods depends heavily on the 'high-fidelity' Finite Element (FE) model with a large number of parameters for estimating the model parameters (Ding et al., 2020; Marin et al., 2015; Pedram et al., 2017). In reality, sensors are usually installed in a limited number of locations rather than in positions that correspond to each degree of freedom (DOF) of the structural model; this causes spare/incomplete measurement. Thus, identifying model parameters with sufficient accuracy requires a number of high-quality and complete measurement data from the field, which is unrealisable and expensive in practice sensing constraints (Hou & Xia, 2021, pp. 2010–2019; Y. Lu et al., 2024). Besides, the optimisation of a large number of parameters is a major challenge and

computationally heavy when optimising large-scale complex structures (Ebrahimian et al., 2018; Eftekhar Azam et al., 2015; Pedram et al., 2017).

In contrast, traditional data-driven methods are implemented directly from the system response without a physical model, a series of research based on the time domain (H. Chen & Ni, 2018; Cremona & Brandon, 1992; Cunha & Caetano, 2006), frequency domain(Ding et al., 2020; Gul & Catbas, 2008; Sadeghzadeh & Khatibi, 2017), and time-frequency(Ghahari et al., 2017; Y. Yang & Nagarajaiah, 2014), including Ibrahim time domain method (ITD) (Brincker et al., 2019) and Frequency Response Functions (Limongelli, 2010). To conduct modal analysis to extract modal information and to identify physical parameters for the task of system identification. Nevertheless, these conventional data-driven methods can obtain accurate results in the linear structure, which are challenging to apply to the nonlinear system. (Avci et al., 2021b; M. Peeters et al., 2011). But non-linearity commonly exists in practical civil structures, such as cracking under the influence of wind loads, deformations, or instability of structural boundary conditions under seismic loading, etc, which is hard to express by the simplified and approximated assumptions(Correa, 2014; Huang et al., 2017; Shiki et al., 2017; Worden et al., 2008).To handle nonlinear structures, state space equations-based methods are conducted for the nonlinear system identification based on partially known modelling assumptions and some candidate equations (Lai & Nagarajaiah, 2019; Paduart et al., 2010), but due to the multivariate nature of the representation of the real complex structures, there is a heavy reliance on a priori hypothesis functions of the physical model. Deep learning (DL)-based data driven methods have been intensively used in SHM, with their capability to approximate complex relationships (excitation and response of a real structural system) accurately and efficiently, while they always

require full measurements from all degrees of freedom (DOF) in order to construct comprehensive and accurate relationships about the structure. Moreover, such DL-based methods always the black box and lack reliability in practice (Cha et al., 2024; Y. Lu et al., 2023);

Combining model-based and data-driven approaches, hybrid models for system identification are proposed to overcome the extensive computational cost and increase the reliability of the model (Ghorbani et al., 2020). As a branch of the hybrid method, Physics-informed deep learning (PIDL) is conducted by adding the governing equations of the underlying physical principle (e.g., state space equations or modal information) to constrain deep learning process to create more accurate results and enhance the explainability(Raissi et al., 2019). Zhang et al. and Yin et al. (Yin et al., 2023; R. Zhang et al., 2020) informed the physical loss function into the deep neural networks to improve the accuracy for response prediction and damage identification of real structure with nonlinearity. However, the built network model still needs a certain amount of full measurement data to be trained to produce impressive results. Nevertheless, repeating the experimental or field test to obtain a sufficient number of data is also infeasible. To address the challenge of training data, another PIDL method that directly embeds the physics model into the neurons or layers is proposed. (L. Lu et al., 2021). Combining a low-fidelity physical model (simplified numerical model) with the neural operator, the method can precisely represent the ‘high-fidelity’ structural model (FE mode or real structure). In detail, the low fidelity model is expressed by the linear governing equations as the structural main components with fewer parameters, and neural networks are responsible for fitting the complex discrepancy component.

Due to the low fidelity linear structure model informed, which has already deployed the vital

parts of the true structural dynamic behaviours(Yaghoubi et al., 2018), therefore, the neural network with fewer trainable variables is also more capable of capturing the part of the discrepancy between low fidelity model with the real structure (L. Lu et al., 2022). Recently, Yu et al. (Yu et al., 2020) proposed a physical-guided ML learning method for structural dynamic simulation; they efficiently map the known physical information into recurrent neural networks and approximate the unknown (residual) information into the general multilayer perceptron (MLP) to improve the performance of response prediction. Similar to the process of solving differential equations, Sadeghi Eshkevari et al. (Sadeghi Eshkevari et al., 2021) proposed a time series Physics-based neural architecture (DynNet) method based on recurrent neural cells for a complete set response prediction of multiple degrees of freedom (MDOFs) nonlinear system. Even though these works perform well for the system response prediction, including (e.g. displacement and restoring force, etc.), the process of method validation is limited to the stage of numerical simulation. Kridelis et al. (Krikelis et al., 2024) proposed neural network hysteresis operators for hysteretic system identification. It takes the similarity of the classical activation function to dynamical behaviours of the hysteresis, compacting and exacting the hysteresis components into recurrent ANNs, which allows flexibly to identify complex model structures combining with other hysteresis or linear elements. Li et al. (H.-W. Li et al., 2024) modelled the forced vibration system based on a continuous-time state-space neural network (CSNN) purposing on system state prediction, which contains two independent neural networks with turned-off bias allowing to capture the linear and nonlinear components automatically to improve the stability of the built model. In all of the above PIDL studies, the training of the hybrid model was supervised using the full state of the structure of all DOF data. Because they do not consider the spatial representation

relation, i.e a mapping relation between the observation and the full state, these studies can only be applied in the relatively ideal situation where the observations are in perfect alignment with the complete state. That is, these studies lack consideration and validation of the situation that only observations from a limited number of DOFs are available. However, it is common in practice that, due to sensing constraints or an excessive number of DOFs in large civil structures, sensors are often installed in a limited number of positions rather than corresponding to each DOF of the structural model, resulting in limited observations(Park & Noh, 2017). Therefore, in order to perform system identification in most cases, it is necessary to consider the spatial relationship between the full state and the limited observations, to make full use of the full state from the low-fidelity model in the hybrid model to address the constraints of a limited number of observations in real measurements.

Taking into account the advantages of the above PIDL hybrid model-based structural system identification methods, and considering the research gap of such methods using limited observations for structural damage identification, this chapter embeds the structural physical model into a neural network, with the spatial feature representation relationship between the limited observation and the full state, to propose a spatio-temporal-based-physics neural operator (PhySTN) framework. Inspired by the previous research (Kontolati et al., 2024), the proposed PhySTN framework is made by a **spatial-feature encoder** and a **time operator (NODEs).** The **spatial-feature encoder** is constructed to represent spatial feature representation relationships utilizing the full state from a healthy numerical model and limited observations from a healthy real structure. This spatial feature representation relation remains unchanged at the same location of observation, even if the structural condition changes. Using this spatial relationship, the full state of the time operator made by prior knowledge

and a neural network (NN) can be projected onto the limited observations; the limited observations is utilized to train the time operator, which leads to unknown discrepancy term between the healthy numerical model and unknown real structure is stored in NN, and then fit their parameters to perform system identification. The proposed method can be summarised as **(i)** Based on an encoder network, a spatial feature representation relationship is constructed by using a full state (input) from low-fidelity models in the health condition to encode limited observation (output) of the actual model in the healthy state. **(ii)** The simplified numerical model under health conditions and the NN discrepancy term make up the **time operator** representing the true structure, whose generated full state is encoded as a limited observation by the **spatial-feature encoder**; under the unknown condition of the real structure, the time operator is trained using the limited observation to reconstruct the discrepancy term between the health and the unknown conditions. **(iii)** The time operator represents the real structure and outputs the full state, and the missing response of the corresponding missing sensor can be reconstructed from this full state. Further, the inputs and outputs of the NNs, the relationship of which can be used to describe the discrepancy terms of the structure between different conditions, are extracted and their corresponding parameters are fitted for structural system parameter identification; this process also demonstrates the consistency of the neural network in terms of physical knowledge. Thus, the novelty and main contribution of this work can be summarized as follows: **(i)** Using a spatial feature representation relationship of structure, an observation matrix mapping full state to limited observation is constructed by an encoder network learning latent pattern from the full state to observation. Through this spatial representation relation, the full state is compressed and reduced in dimension to a limited observation in the actual structure; **(ii)** Using the pre-trained observation

matrix based on the spatial feature relationship, the time operator consisting of NN and health numerical model is constructed to represent real structure with only limited observation. It addressing the limitation that limited observation cannot be used directly for structural parameter identification. In addition, the missing response outside of the observations is enable to be reconstructed from the outputs of the time operator. **(iii)** The proposed framework PhySTN can be used to reconstruct and predict the response in time and space. The encoder networks treated as observation matrix and NN seen as restoring force models in structural dynamics improve the reliability of the proposed framework and increase the interpretability of the neural network.

The chapter is organised as follows: The theoretical background, including the theoretical background of structural dynamic systems and Neural ODEs, and the proposed framework are introduced in Section 2. A numerical study on a nonlinear 5-storey model is used in section 3 to validate the performance of the proposed method. A nonlinear experimental model with an adaptive magnetorheological elastomer (MRE) base isolator is conducted to validate the proposed method in the laboratory in section 4. Finally, in section 5, a summary and conclusion are provided.

6.3. Theoretical Background and Methodology

The proposed Spatial-time-based-physics neural operators (**PhySTN**) are established using encoder and time operator based on the theory of structural dynamic system. The proposed PhySTN framework is established based on neural networks with physical interpretability related to structural dynamics knowledge. This framework, using limited observation from the measurement, achieves structural system identification and missing response reconstruction. The background information of the structural dynamic system, observation matrix and NODEs are introduced first. Subsequently, the

key stages and running mechanism of the proposed method are explained in detail, including the construction of spatial feature encoder, training the time operator and missing structural response reconstruction and structural system identification strategies.

6.3.1. Background Information

6.3.1.1. Structural Dynamic Systems

Considering the dynamic system characterized by its degrees of freedoms (DOFs),

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) + \mathbf{g}(\mathbf{x}(t), \dot{\mathbf{x}}(t)) = \mathbf{S_u}\mathbf{u}(t) \quad \text{Eq. 6-1}$$

where **M, K** and **C** denote the matrices representing the mass, damping, and stiffness of the system, respectively. The variables$\ddot{\mathbf{x}}(t), \dot{\mathbf{x}}(t)$ and $\mathbf{x}(t)$signify the acceleration, velocity, and displacement, correspondingly. In this model, $\mathbf{u}(t)$ represents the known ground excitation force, and $\mathbf{g}(\mathbf{x}(t), \dot{\mathbf{x}}(t))$denotes terms associated with state variables, serving to represent two scenarios: (a) all potential discrepancies between the numerical model and the actual structure, which include boundary conditions, modelling errors, and environmental uncertainties of the dynamic system; and (b) the structural differences between the healthy structure and the damaged structure.

In this study, the potential variances and structural damages are defined as discrepancies within the structural dynamic system.

The typical format of dynamic systems is represented by a state space equation, as shown below:

$$\frac{d\mathbf{h}(t)}{dt} = \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\mathbf{h}(t) \quad \text{Eq. 6-2}$$

where $\mathbf{h}(t) = \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} \in \mathbb{R}^{2n}$, $\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M^{-1}K} & -\mathbf{M^{-1}C} \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{M^{-1}} \end{bmatrix}, \mathbf{G}(\mathbf{h}(t)) = \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{g}(\mathbf{h}(t))$and *n* is the numbers of DOFs.

6.3.1.2. Observation matrix

Considering the observable measurement response, the measurement equation regarding to observation vector **q**(t) and the system full state vector **h**(t) can be written as:

$$\mathbf{q}(t) = \mathbf{C}\mathbf{h}(t) \quad \text{Eq. 6-3}$$

where **C** is the observation matrix. Recombining Eqs. (2) and (3), the state space equation of the dynamic system relevant to the observation can be expressed as

$$\mathbf{C}^{-1}\frac{d\mathbf{q}(t)}{dt} = \mathbf{A}\mathbf{C}^{-1}\mathbf{q}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\mathbf{C}^{-1}\mathbf{q}(t) \quad \text{Eq. 6-4}$$

In the case of observation, a state is obtained, which can be expressed with the dynamic system using the observation matrix

$$\frac{d\mathbf{q}(t)}{dt} = \mathbf{C}\mathbf{A}\mathbf{C}^{-1}\mathbf{q}(t) + \mathbf{C}\mathbf{B}\mathbf{u}(t) + \mathbf{C}\mathbf{G}\mathbf{C}^{-1}\mathbf{q}(t)$$
$$= \mathbf{C}\mathbf{A}\mathbf{h}(t) + \mathbf{C}\mathbf{B}\mathbf{u}(t) + \mathbf{C}\mathbf{G}\mathbf{h}(t) = \mathbf{C}[\mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\mathbf{h}(t)] \quad \text{Eq. 6-5}$$

6.3.1.3. Neural ODEs

Neural Ordinary Differential Equations (Neural ODEs) have garnered significant attention in recent years due to their ability to establish a close connection between neural networks and differential equations. As a result of this integration, various variant forms of Neural ODEs have emerged. Notably, Neural ODEs can be viewed as the continuous equivalent of residual networks (ResNets). In ResNets, the transformation of the hidden state from one layer, *t*, to the subsequent layer, *t*+1, is facilitated by a differentiable functions $f_t(\cdot)$ (R. T. Q. Chen et al., 2018)illustrated below :

$$\mathbf{h}_{t+1} = \mathbf{h}_t + f_t(\mathbf{h}_t) \quad \text{Eq. 6-6}$$

Where $\mathbf{h}_t \in R_d$ denotes the hidden state at layer t and represents differentiable functions that preserve the dimension of $\mathbf{h}_t$. The difference $\mathbf{h}_{t+1} - \mathbf{h}_t$represents the discretization of the derivative when $\Delta t = 1$, according to Euler's method of discretization (Butcher, 2016; Hairer & Wanner, 1996):

$$\frac{\mathbf{h}_{t+1} - \mathbf{h}_t}{\Delta t} = f_t(\mathbf{h}_t) \quad \text{Eq. 6-7}$$

When $\Delta t \to 1$, Eq. 6-7 becomes to

$$\lim_{\Delta t \to 0} \frac{\mathbf{h}_{t+1} - \mathbf{h}_t}{\Delta t} = \frac{d\mathbf{h}(t)}{dt} = \lim_{\Delta t \to 0} \frac{f_t(\mathbf{h}_t)}{\Delta t} = \tilde{f}(\mathbf{h}(t), t, \theta) \quad \text{Eq. 6-8}$$

So far, the hidden units of the neural networks have been parameterized in the form of ODEs. The initial data point $\mathbf{h}_0$ can be transformed into a set of features at time step *t* by solving the Initial Value Problems (IVPs), as illustrated below

$$\left\{ \begin{array}{c} \frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t)) = \tilde{f}(\mathbf{h}(t), t, \theta) \\ \mathbf{h}(\mathrm{t}_0) = \mathbf{h}_0 \end{array} \right\} \quad \text{Eq. 6-9}$$

In the practical application of structural dynamics problems, the scenarios often involve forced excitation rather than pure IVPs. This necessitates the integration of a priori physical knowledge into the NODEs. Consequently, the function $\tilde{f}(\cdot)$ in Eq. 6-9 is modified to include an excitation force term, $\mathbf{u}(t)$ as demonstrated below

$$\left\{ \begin{array}{c} \frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t)) = \tilde{f}(\mathbf{h}(t), t, \mathbf{u}(t), \theta) \\ \mathbf{h}(\mathrm{t}_0) = \mathbf{h}_0 \end{array} \right\} \quad \text{Eq. 6-10}$$

Solving the continuous dynamic system using a neural network, denoted as $\tilde{f}(\cdot)$, is a critical step in the development of NODEs (Dupont et al. (2018) and Chen et al. (2019)). The neural network $\tilde{f}(\cdot)$serves as the solution mechanism for the function $f(\cdot)$ which encapsulates the dynamics of the system state over time. This model tracks the hidden state at each time step *t*. When solving the ODE from the initial condition $\mathbf{h}(t_0) = \mathbf{h}_0$, which initiative the states of the ODE at a given time *t* depend on the initial conditions $\mathbf{h}_0$. Consequently, with $\mathbf{u}$(t) provided as the input to $\tilde{f}(\cdot)$the output layer $\mathbf{h}(t_T) = \mathbf{h}_\mathrm{T}$represents the solution of Eq. 6-6 at the final time *T*.

In neural ODEs, an initial state $\mathbf{h}(t_0)$ is mapped to a final state $\mathbf{h}(t_T)$ through a transformation from data to feature by mathematically solving an ordinary differential equation. This process is defined as ODESolve $\left(\mathbf{h}(t_0), \tilde{f}, t_0, t_T, \mathbf{u}(t), \theta\right)$. Here, $L(\cdot)$ represents the loss function

in the forward process, which evaluates the discrepancy between the predicted state $\mathbf{h}(t)$ and the true state as depicted below

$$\mathcal{L}\big(\mathbf{h}(t_T)\big) = \mathcal{L}\left(\mathbf{h}(t_0) + \int_{t_0}^{t_T} \tilde{f}(\mathbf{h}(t), t, \theta)dt\right) = \mathcal{L}\left(\mathrm{ODESolve}\big(\mathbf{h}(t_0), \tilde{f}, t_0, \mathrm{t_T}, \mathbf{u}(t), \theta\big)\right) \quad \text{Eq. 6-11}$$

where θ represents the trainable weights of the neural network $\tilde{f}(\cdot)$ which are optimized by minimizing the loss function $L(\cdot)$. The solutions to ODESolve are approximated using the Runge-Kutta methods, as described in the establishment of neural ODE solvers by Dupont et al. (Dupont et al., 2019)and Chen et al. (Z. Chen et al., 2019)

$$\mathbf{h}(t_T) = \mathrm{ODESolve}\big(\mathbf{h}(t_0), \tilde{f}, t_0, t_T, \mathbf{u}(t), \theta\big) \quad \text{Eq. 6-12}$$

where $\mathbf{h}(t_T)$ is the generated results from the neural ODEs based on Eq.6-10. During the training process of NODEs, the parameters of the neural network are updated by minimising the MSE loss functions.

The neural network $\tilde{f}(\cdot)$can be represented by the chain structure with l+1 layers:

$$\tilde{f}(\mathbf{h}) \begin{cases} \mathbf{h}^1 = \sigma^1(\mathbf{W}^1\mathbf{h}^0 + b^1) \\ \mathbf{h}^2 = \sigma^2(\mathbf{W}^2\mathbf{h}^1 + b^2) \\ \vdots \\ \mathbf{h}^L = \sigma^{\mathrm{l}}\big(\mathbf{W}^1\mathbf{h}^{1-1} + \mathrm{b}^{\mathrm{l}}\big) \end{cases} \quad \text{Eq. 6-13}$$

where $\mathbf{h}^i$ is the intermediate output hidden in the layer, $\mathbf{h}^0$ and $\mathbf{h}^l$ are assigned as input and output vector of the network, respectively. **W** is the weight matrix associated with transfer from i-1 in the layer i, b^i is the bias in layer . σ is the activation function at layer i.

6.3.1.4. Encoder network

The encoder can be used to map latent feature representation relationships from inputs to outputs, accurately fitting the complex relationships between them (S. Li et al., 2024). Assuming a vector **h** with input to the encoder at time variable t, the deterministic mapping $E(\mathbf{h}(t))$ converts the input vector intooutput $\mathbf{q}(t)$. The typical form of the encoder can be written as an affine transformation relationship as

$$\mathbf{q}(t) = E(\mathbf{h}(t)) = \sigma(\mathbf{W}\mathbf{h}(t) + b) \quad \text{Eq. 6-14}$$

where $\mathbf{W}$ is the weight matrix associated with transfer from between $\mathbf{q}(t)$ and $\mathbf{h}(t)$; σ is an activation function, and b is the bias in the layer.

6.3.2. Method of PhySTN

As shown in Figure 6-1, the proposed framework for missing response reconstruction and system identification involves three steps. It can be found in Figure 6- 1, (i**) Constructing the spatial feature representation relationship.** The simplified numerical simulation model representing the corresponding real structure has a total of *n* degrees of freedom. In the actual measurement, there are only *m* (*m*<*n*) observations with respect to this degree of freedom. Under a healthy condition, numerical simulation data are generated, and observation data are collected from real model, being input and output to train spatial-feature encoder representing **spatial feature representation relationship**, respectively. (ii) **Reconstructing the unknown discrepancy.** The time operator contains a simplified numerical model under the health condition and a neural network as a discrepancy term. Using the already constructed **spatial feature representation relationship**, from the initial state, full-state data in the next time step generated by the time operator can be encoded to the predicted limited observation; This allows the true limited observation to be directly used with the predicted limited observation in the loss function to supervise the neural network (NN). In another word, using limited observation data, the time operator consisting of a healthy numerical model and the NN gradually approximates the true structure. As a result, limited observations are used by NN to reconstruct the discrepancy between the numerical model under health conditions and the real structure under an unknown condition. (iii) **Reconstructing the missing structural response and**

identifying the structural system parameters. Missing structural response outside of limited observations under the unknown condition can be reconstructed by finding the corresponding sensor response from the output of the time operator; The displacement and internal restoring force of the structural discrepancy can be obtained from the inputs and outputs of the NN, respectively. The structural parameters of the system can be identified by fitting polynomial equations corresponding to the displacement and restoring force.

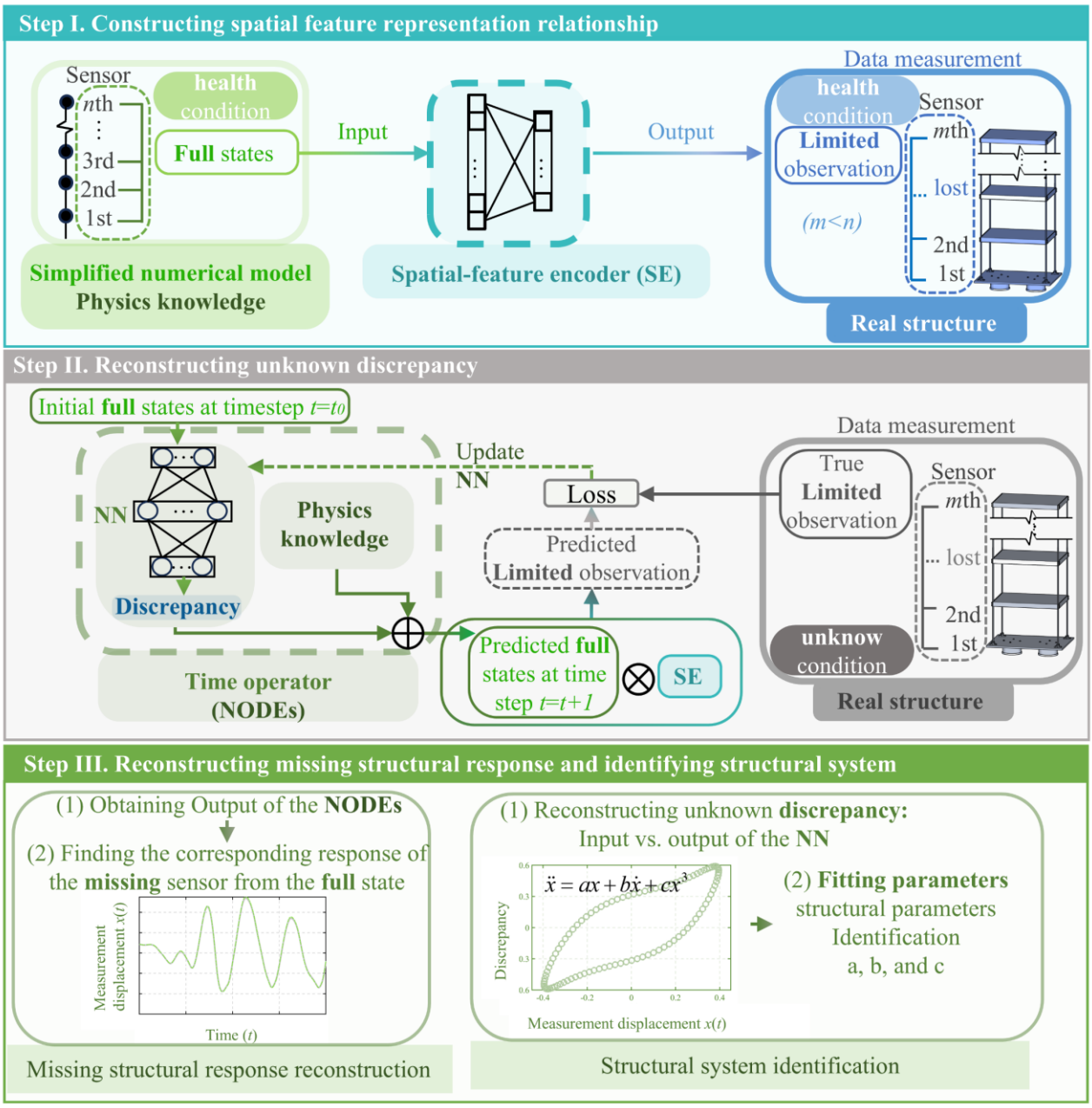


Figure 6-1. Flow chart of the proposed structural identification and response reconstruction and prediction framework.

6.3.2.1. Construction of the spatial-feature encode

As presented in section 2.1.3, solving partial differential equations for structural systems using NODEs requires the full state of the response to be involved in the computational process. In this work, a spatial feature encoder based on an encoder network is constructed in Figure 6- 2, which maps full states to finite observations for the purpose of directly using limited observations for

structural identification treated as an observation matrix. This spatial mapping relation needs to be constructed between full states and limited observations under the same conditions of structure to ensure that it only maps features that are relevant to the space of structures. Besides, this spatial relation will remain unchanged, even if the conditions of the structure change since it depends only on the spatial location of the observation. The health condition in real structure is usually known. Thus, encoder networks are trained on the same health condition by using full states from simplified numerical models and limited observations from real structures. This encoder network is introduced in the section **6.3.1.4** is designed as a fully connected layer, which can be represented as

$$\mathbf{q_u}(t) = \mathrm{SE}(\mathbf{h}_\mathrm{u}(t)) = \mathrm{Identity}(\mathbf{W}\mathbf{h}_\mathrm{u}(\mathrm{t}) + \mathrm{b}) \qquad \text{Eq. 6-15}$$

where $\mathbf{h}_\mathrm{u}$ is the full state from the numerical model and $\mathbf{q_u}(t)$is the limited observation from the real structure, $\mathbf{W} \in \mathbb{R}^{2n\times 2m}$, $\mathbf{b} \in \mathbb{R}^{2m\times 1}$, and Identity $f(x) = x$ is selected as the identity activation function due to the use of neural networks. As can be seen from Eq. 6-15, the proposed encoder uses linear activation functions and bias, and their addition enables a high-dimensional and latent pattern between the full state to a limited observation. Single-layer networks with a small number of trainable parameters allow for fast network training for parameter fitting (Cer et al., 2018). A single-layer network is also sufficient for both encoding accuracy and computational complexity from our experiment. In addition, the $\mathrm{MSE}_{\mathrm{q_u}}$ loss is computed between the predicted $\widehat{\mathbf{q}}_\mathrm{u}(t)$and true $\mathbf{q}_\mathrm{u}(t)$ observation as shown below:

$$\mathrm{MSE}_{\mathbf{q}_\mathrm{u}} = \frac{1}{n}\sum_{i=1}^{n}|\mathbf{q}_\mathrm{u}(t_i) - \widehat{\mathbf{q}}_\mathrm{u}(t_i)| \qquad \text{Eq. 6-16}$$

As an example of a numerical model with *n* degrees of freedom, it is a simplified lumped mass model representing the corresponding real structure, which can be expressed in the form of an

ordinary differential equation (ODE) state space equation,

$$\frac{d\mathbf{h}_{\mathrm{u}}(t)}{dt} = \mathbf{A}\mathbf{h}_{\mathrm{u}}(t) + \mathbf{B}\mathbf{u}(t) \qquad \text{Eq. 6-17}$$

where matrices **A** and **B** in this lumped mass model consist of the **K**, **M** and **C** matrices of their corresponding physical parameters under the health condition; The measured excitations **u**(t) is then given as input to the numerical model, generating structural full state vectors $\mathbf{h}_{\mathrm{u}}(t) = [x_1(t); x_2(t); \dots; x_n(t); \dot{x}_1(t); \dot{x}_2(t); \dots; \dot{x}_n(t)]$ of dimension $2n$. Taking a real structure arranged m ($m < n$) observation points with the excitation **u**(t) under the health condition, displacement and velocity responses of the real structure are collected to form a limited observation state vector $\mathbf{q}_{\mathrm{u}}(t) = [x_1(t); x_2(t); \dots; x_m(t); \dot{x}_1(t); \dot{x}_2(t); \dots; \dot{x}_m(t)]$of dimension $2m$. The inputs and outputs of the **SE** are $\mathbf{h}_{\mathrm{u}}(t)$ from numerical simulations and $\mathbf{q}_{\mathrm{u}}(t)$from real structures, respectively.

In the case where the location of the measurement sensor is constant, when the state of the structure is unknown, using the established spatial characterization relationship allows the encoding of the full state to the limited observation and maintains the high dimension feature of the structure condition.

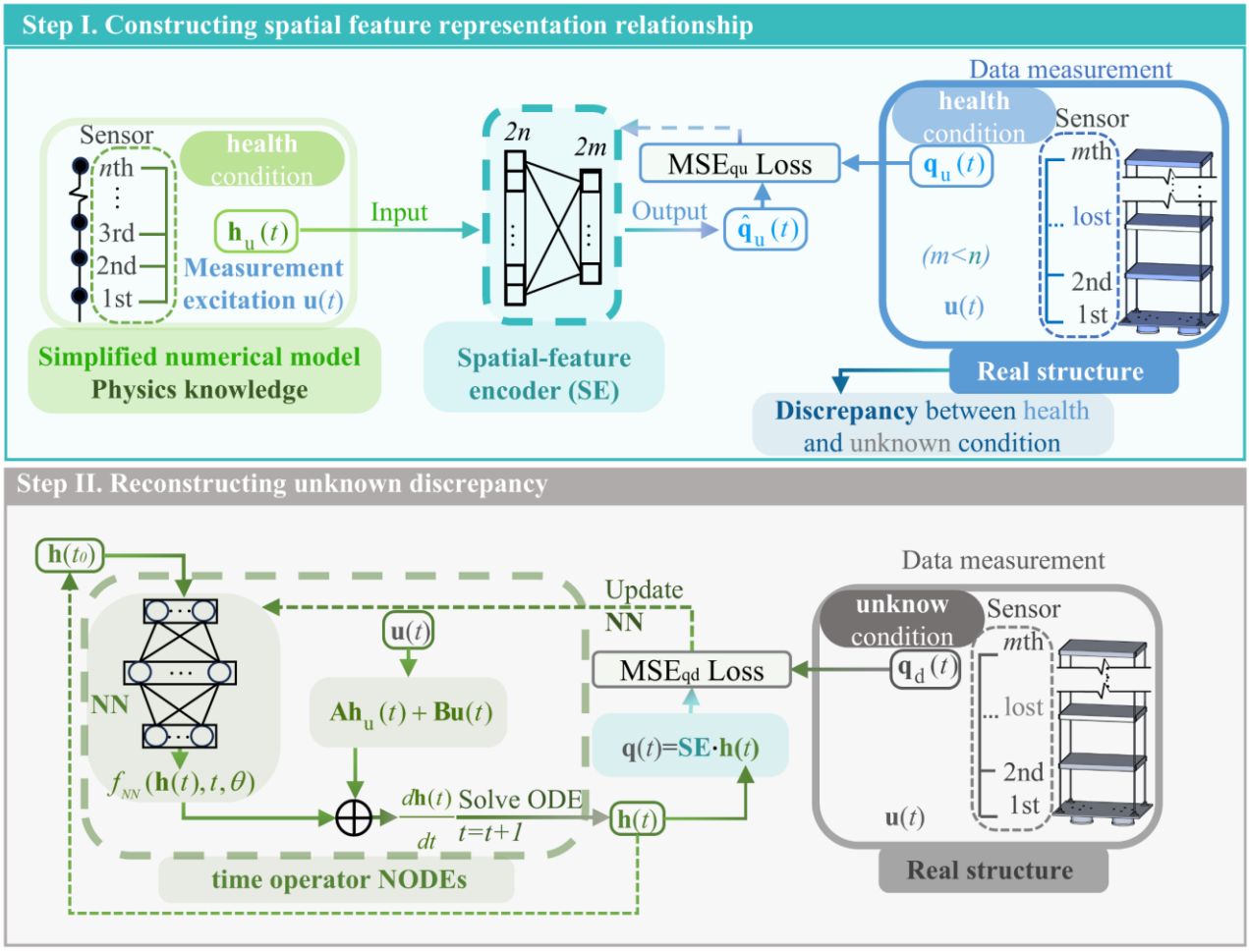


Figure 6-2. Details of the operational mechanisms for each step in the proposed framework

6.3.2.2. Reconstruction of the unknown discrepancy

The time operator combines the spatial-feature encoder constructed in **Step I**. Based on the limited number of observations, the discrepancy between the numerical model in the healthy condition and the real structure in the unknown condition can be reconstructed. The time operator contains a NN and a simplified numerical model the same as in **step I**, where the simplified numerical model represents the model with known health conditions, and the reconstructed discrepancy terms are stored in NN.

In the initial time step t_0, the initial state as the dimension of $2n$ vector $\mathbf{h}(t_0)$of the structure is input to the NN and output of NN is a discrepancy item as the dimension of n vector between the

healthy numerical model and unknown condition real structure; in the same time step, the excitations $\mathbf{u}(t)$ from the measurement is given to the numerical model to generate the derivative of a $2n$ dimension vector full state under the health state from Eq. 6-17. Then, a dimension of the n vector $\mathbf{0} \in \mathbb{R}^{n\times 1}$ is concatenated with output of NN to form $\mathrm{G_{NN}}(\mathbf{h}(t))$expressed in Eq. 6-18, in order to get a $2n$ dimensions vector to sum with $\frac{d\mathbf{h}_{\mathrm{u}}(t)}{dt}$. As a result, the derivative of the predicted full state $\frac{d\mathbf{h}(t)}{dt}$ under the unknown conditions is obtained, which is written as:

$$\begin{aligned}\frac{d\mathbf{h}(t)}{dt} &= \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathbf{u}(t) + \mathrm{G_{NN}}(\mathbf{h}(t)) \\ &= \begin{bmatrix} 0 & I \\ -\mathbf{M}^{-1}\mathbf{K}_{\mathrm{u}} & -\mathbf{M}^{-1}\mathbf{C}_{\mathrm{u}} \end{bmatrix}\mathbf{h}(t) + \begin{bmatrix} 0 \\ -\mathbf{M}^{-1} \end{bmatrix}\mathbf{u}(t) + \begin{bmatrix} 0 \\ f_{NN}(\mathbf{h}(t), t, \theta) \end{bmatrix}\end{aligned} \quad \text{Eq. 6-18}$$

where $f_{NN}(\mathbf{h}(t), t, \theta)$ is the approximated functions learned by NN with is parameterised by weight θ; $\mathbf{0} \in \mathbb{R}^{n\times n}$is a zero matrix; $\mathbf{I} \in \mathbb{R}^{n\times n}$ is an eye matrix; $\mathbf{K}_{\mathrm{u}} \in \mathbb{R}^{n\times n}$, $\mathbf{M} \in \mathbb{R}^{n\times n}$and $\mathbf{C}_{\mathrm{u}} \in \mathbb{R}^{n\times n}$ are the physical properties of the structure under the health condition. The $f_{NN}(\mathbf{h}(t), t, \theta)$ can be represented by the formula shown in Eq. 6-13, and nonlinear activation function σ of Rectified Linear Unit (ReLU), $\mathrm{ReLU}(u) = \begin{cases} u, & u > 0 \\ 0, & u < 0 \end{cases}$ is selected at each layer. Due to its efficient computing speed and introduction of nonlinearities, it helps neural networks to learn complex patterns better.

Then, the time operator is then solved by utilising the approach presented in Section 2.1.3., which leads to the predicted full state $\mathbf{h}(t)$ in the next time step t=t+1 obtained. Next, the predicted full state $\mathbf{h}(t)$ is encoded into its corresponding predicted limited observation $\mathbf{q}(t)$ through the encoder at the same time step, shown in Figure 6-2 and expressed in Eq. 6-19. At the same time the loss function $\mathrm{MSE_{q_d}} = \frac{1}{n}\sum_{i=1}^{n}|\mathbf{q}_{\mathrm{d}}(t_i) - \mathbf{q}(t_i)|$ is computed between solved $\mathbf{q}(t)$ and the true limited observation $\mathbf{q}_{\mathrm{d}}(t)$ from the real structure under the unknown condition to update the parameters of NN.

$$\mathbf{q}(t) = \mathrm{SE}[\mathbf{h}(t)] \qquad \text{Eq. 6-19}$$

Through the above process and the derived Eq. 6-18, it can be seen that the structural discrepancy terms can be reconstructed and stored in the $f_{NN}(\mathbf{h}(t), t, \theta)$ based on limited observations. The $f_{NN}(\mathbf{h}(t), t, \theta)$ working in collaboration with physical knowledge allows the NN to follow the established rules of structural dynamics.

6.3.2.3. Reconstruction of the missing response and structural identification

In the proposed framework, missing responses can be reconstructed utilising time operators that represent the real structure. Unknown structural parameters can be converted into a regression problem by finding expressions about the inputs and outputs of the NN.

The time operator has been trained using limited observation from real structure data under unknown conditions. The output of the time operator representing the real structure is the predicted full states $\mathbf{h}(t)$ shown in **step ii** of Figure 6-1. The corresponding missing sensors, including displacement and velocity in the real structure, can be found from the predicted full state $\mathbf{h}(t)$ for response reconstruction.

The NN can be expressed as the equations representing the relationship between displacement and restoring force following the process below:

(a) Considering the structure under healthy conditions, the prior physics as described in Eq. 6-17 is expressed as follows:

$$f_{\text{health}}(\cdot) = \frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}_{\text{u}} & -\mathbf{M}^{-1}\mathbf{C}_{\text{u}} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t) \qquad \text{Eq. 6-20}$$

where $\mathbf{K}_{\text{u}}$, $\mathbf{M}$ and $\mathbf{C}_{\text{u}}$ are the physical properties of the structure under the health condition. Structure changes condition, can be expressed in terms of changes in system stiffness or damping

(Ren & De Roeck, 2002). Thus, the structure under the known condition can be rewritten as

$$f_{\text{unknown}}(\cdot) = \frac{d\mathbf{h}(t)}{dt} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K}_{\text{u}} - \Delta\mathbf{K}) & -\mathbf{M}^{-1}(\mathbf{C}_{\text{u}} - \Delta\mathbf{C}) \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t)$$
$$= \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}_{\text{u}} & -\mathbf{M}^{-1}\mathbf{C}_{\text{u}} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}(-\Delta\mathbf{K}) & -\mathbf{M}^{-1}(-\Delta\mathbf{C}) \end{bmatrix} \mathbf{h}(t)$$

Eq. 6-21

where the matrix $\Delta\mathbf{K} \in \mathbb{R}^{n\times n}$ and $\Delta\mathbf{C} \in \mathbb{R}^{n\times n}$ representing the changes in stiffness and damping matrix of the structure under unknown conditions. Eq. 6-20 represents the physical knowledge of structure under health condition. Therefore, the expression of the model discrepancy between the health and unknown conditions $f_{\text{health}}(\cdot) - f_{\text{unknown}}(\cdot)$ is expressed as

$$\mathrm{G}_{\Delta}(\cdot) = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}(-\Delta\mathbf{K}) & -\mathbf{M}^{-1}(-\Delta\mathbf{C}) \end{bmatrix} \mathbf{h}(t) \qquad \text{Eq. 6-22}$$

$\mathrm{G}_{\Delta}(\cdot)$is the discrepancy item in Eq. 6-18 stored in the $f_{NN}(\mathbf{h}(t), t, \theta)$ and expressed as a function $\mathrm{G}_{\text{NN}}(\mathbf{h}(t))$. It can be also written as

$$\mathrm{G}_{\text{NN}}(\mathbf{h}(t)) = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}(-\Delta\mathbf{K}) & -\mathbf{M}^{-1}(-\Delta\mathbf{C}) \end{bmatrix} \mathbf{h}(t) = \begin{bmatrix} \mathbf{0}_{n\times 1} \\ \mathbf{h}(t)\boldsymbol{\Theta} \end{bmatrix} \qquad \text{Eq. 6-23}$$

where $\mathbf{h}(t)\boldsymbol{\Theta}$ is the discrepancy term captured by NN; $\mathbf{h}(t)$contains the displacement of the real structure; $\boldsymbol{\Theta}$is a polynomial function that can be obtained by establishing a regression problem that fits the inputs and outputs of the NN using the Matlab function polyfit (The MathWorks Inc. (2022). MATLAB version: 9.13.0 (R2022b), Natick, Massachusetts: The MathWorks Inc. https://au.mathworks.com/help/matlab/math/polynomial-curve-fitting.html). This expression can be written in the forms regarding K and C for structural parameter identification.

6.4. Numerical Example – nonlinear 5-storey building model with an MRE isolator

In this section, a 5-storey building model with a magnetorheological elastomer (MRE) isolator

will be presented to validate the proposed framework based on the identification of nonlinear structural parameters with limited observations and missing response reconstruction. The basic information about the building model and the MRE isolator setup will be presented first. This will be followed by a description of the data preparation process. Then, the results of nonlinear parameter identification and missing response reconstruction are shown and discussed, including three scenarios of noise contamination, different observation periods and observation locations. Finally, a comparative assessment between the predicted and real results of the proposed method is presented.

6.4.1. Basic information

The numerical model is made by a 5-storey building adaptive MRE base isolators, the system is presented in Figure 6- 3. Due to the seismic protection purposes of the MRE base devices, they are deployed under the first floor, the entire building model can be expressed with $n + 1$ DOFs ($n = 5$ in this example). Because of two main features of viscoelasticity and strain stiffening of MRE base isolator, the force-displacement loops can be regarded as a synthesis of two distinct curve types: one representing the conventional viscoelastic properties of the rubber, and the other illustrating the strain stiffening characteristic induced by the applied magnetic field plotted in Figure 6-3. Therefore, the nonlinear behaviours of the system can be expressed by the Kelvin model(Christensen, 2003). This model comprises a linear spring and a viscous dashpot, while a nonlinear spring is employed to describe the strain stiffening phenomenon. The nonlinear MRE restore force of the system can be expressed as

$$\begin{gathered} \mathrm{F_{MRE}}(I,t) = k_b(I)x_b(t) + c_b(I)\dot{x}_b(t) + \alpha(I)x_b^3(t) + F_0 \\ \alpha(I) = -0.1232 \cdot I^2 + 0.7366 \cdot I \\ k_b(I) = 8.062 \cdot I \\ c_b(I) = 0.0994 \cdot I^2 + 0.7598 \cdot I \end{gathered} \quad \text{Eq. 6-24}$$

where I is the constant current of is applied on the MRE isolator, equal to 1 A in this numerical example; $x_b$and $\dot{x}_b$imply the displacement and velocity of the base; both c_b and k_b are the damping and stiffness parameters of the model; α is the coefficient for the power law element, while F_0 denotes the initial shear force generated by the initial displacement of the device. The motion of the equation of each floor can be written as:

$$\begin{cases} (k_0+k_1)x_b - k_1x_1 + (c_0+c_1)\dot{x}_b - c_1\dot{x}_1 + l_c m_b \ddot{x}_b = \mathrm{F_{MRE}} - l_c m_b u_g \\ -k_1x_b + (k_1+k_2)x_1 - k_2x_2 - c_1\dot{x}_b + (c_1+c_2)\dot{x}_1 - c_2\dot{x}_2 + m_1\ddot{x}_1 = -m_1u_g \\ -k_2x_1 + (k_2+k_3)x_2 - k_3x_3 - c_2x_b + (c_2+c_3)x_2 - c_3x_3 + m_2x_2 = -m_2u_g \\ -k_3x_2 + (k_3+k_4)x_3 - k_4x_4 - c_3x_3 + (c_2+c_3)x_3 - c_4x_4 + m_3x_3 = -m_3u_g \\ -k_4x_3 + (k_4+k_5)x_4 - k_5x_5 - c_4x_4 + (c_3+c_4)x_4 - c_5x_5 + m_4x_4 = -m_4u_g \\ -k_5x_4 + k_5x_5 - c_5\dot{x}_4 + c_5\dot{x}_5 + m_5\ddot{x}_5 = -m_5u_g \end{cases} \quad \text{Eq. 6-25}$$

where l_c is the number of the MRE base isolators; $x_i, \dot{x}_i$ and $\ddot{x}_i$ indicate the displacement, velocity and acceleration of the i th DOF (i = b, 1, 2, 3, 4, 5), respectively; the stiffness coefficient is designed as k_b=8.062, k_0= 6.544 , k_1= 1146, k_2=3124, $k_3 = k_4$=3156 and k_5=2978 (N/m); m_b= 150, m_1=214 and$m_2 = m_3 = m_4 = m_5$=207 (kg); c_b= 0.8592, c_0= 0.3254; c_1=0.0584, c_2=0.1117, c_3=0.1128 c_4=0.11; c_5=123.3 (N·s/m).

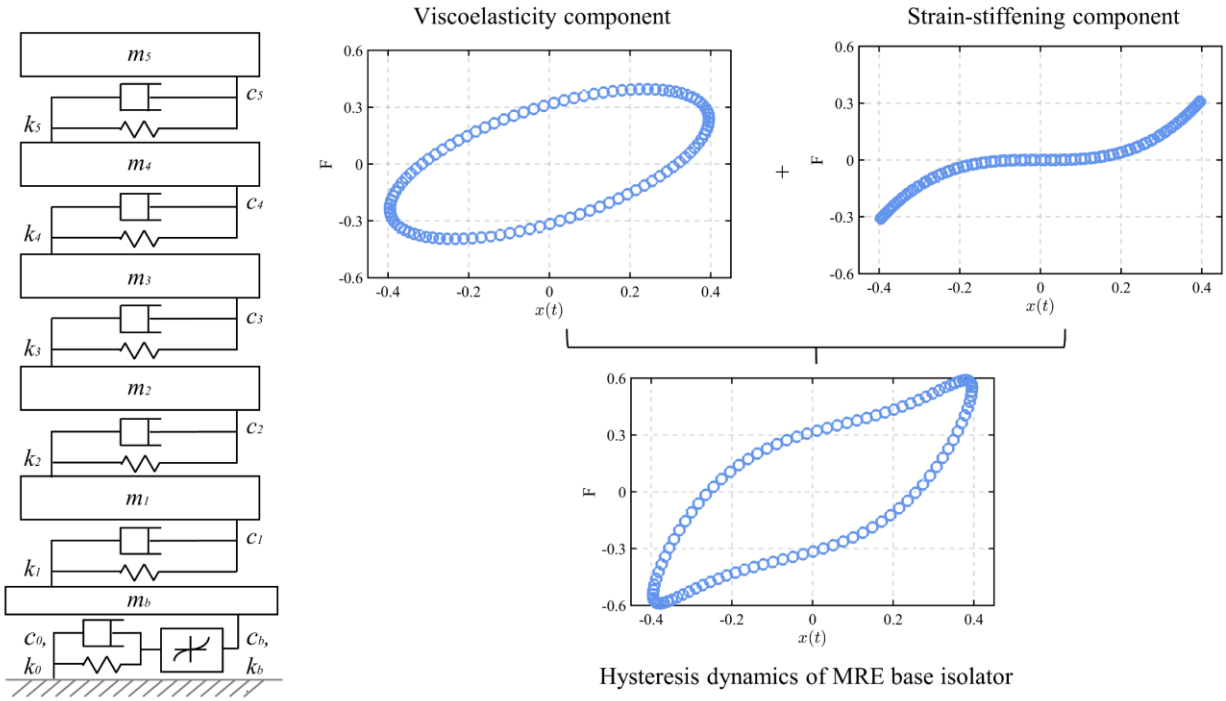


Figure 6-3. 5-storey building model with MRE nonlinear structural system

The nonlinear component of the system generated by the force of MRE base isolator is influenced by the applied current, displacement and velocity of the based floor, which can be expressed by the stiffness and damping changes $\mathbf{K}_{\mathrm{MRE}} \in \mathbb{R}^{6\times6}$ and $\mathbf{C}_{\mathrm{MRE}} \in \mathbb{R}^{6\times6}$ respectively(Y. Li et al., 2013; J. Yang et al., 2013).

The state equation of the 6-DOF numerical model can summarised as

$$\begin{aligned}\frac{\mathrm{d}\mathbf{h}(t)}{\mathrm{d}t} &= \begin{bmatrix}\mathbf{0} & \mathbf{I}\\ -\mathbf{M}^{-1}(\mathbf{K}_{I=0}+\mathbf{K}_{\mathrm{MRE}}) & -\mathbf{M}^{-1}(\mathbf{C}_{I=0}+\mathbf{C}_{\mathrm{MRE}})\end{bmatrix}\mathbf{h}(t)+\begin{bmatrix}\mathbf{0}\\ -\mathbf{M}^{-1}\end{bmatrix}\mathbf{u}(t)\\ &= \begin{bmatrix}\mathbf{0} & \mathbf{I}\\ -\mathbf{M}^{-1}\mathbf{K}_{I=0} & -\mathbf{M}^{-1}\mathbf{C}_{I=0}\end{bmatrix}\mathbf{h}(t)+\begin{bmatrix}\mathbf{0}\\ -\mathbf{M}^{-1}\end{bmatrix}\mathbf{u}(t)+\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ -\mathbf{M}^{-1}\mathbf{K}_{\mathrm{MRE}} & -\mathbf{M}^{-1}\mathbf{C}_{\mathrm{MRE}}\end{bmatrix}\mathbf{h}(t)\\ &= \begin{bmatrix}0 & I\\ -\mathbf{M}^{-1}\mathbf{K}_{I=0} & -\mathbf{M}^{-1}\mathbf{C}_{I=0}\end{bmatrix}\mathbf{h}(t)+\begin{bmatrix}\mathbf{0}\\ -\mathbf{M}^{-1}\end{bmatrix}\mathbf{u}(t)\\ &+[0;0;0;0;0;0;k_b x_b(t)+c_b\dot{x}_b(t)+\alpha x_b^3(t);0;0;0;0;0]\end{aligned} \quad \text{Eq. 6-26}$$

where $\mathbf{K}_{I=0} \in \mathbb{R}^{6\times6}$ and $\mathbf{C}_{I=0} \in \mathbb{R}^{6\times6}$ are the stiffness and damping matrix of the linear system, when The MRE isolator is powered off at current I = 0 A is a known condition, and the isolator is at its softest state leading without the MRE force; M is the mass matrix of the model; $\mathbf{0} \in \mathbb{R}^{6\times6}$ is a zero

matrix and $\mathbf{I} \in \mathbb{R}^{6\times6}$ is the identity matrix; In addition, the natural frequencies for the numerical model under this condition are 0.676 Hz, 10.068 Hz, 16.679 Hz, 24.072 Hz, 31.792 Hz and 37.306 Hz, respectively.

6.4.2. Data preparation

The purpose of the proposed framework is to identify the unknown nonlinear behaviours as the discrepancy between known and unknown conditions using limited observation. Thus, a linear 5-storey model with the MRE isolator power off under the known condition is considered as the known numerical model under the health condition, which was used in **steps i** and **ii** in the proposed framework.

In **step i**, it is required to obtain the data of full state $\mathbf{h}_{\mathrm{u}}(t)$ generated from the numerical model under the health condition and the limited observation $\mathbf{q}_{\mathrm{u}}(t)$obtained by deleting the lost sensors from the $\mathbf{h}_{\mathrm{u}}(t)$, to construct the spatial feature representation relationship. The ground motions from four earthquakes, El Centro (E1), Hachinohe (E2), Kobe (E3), and Northridge (E4), were applied to the numerical model as input, with a sampling rate of 100 Hz for $\mathbf{u}(t)$. According to Nyquist's theorem (H. Nyquist, 1928) , the data must be sampled at more than twice the highest frequency present in the structure. As mentioned earlier, the highest frequency of the 5-story numerical model is 37.306 Hz. Therefore, the 100 Hz sampling rate is sufficient to capture the features of all modes within the model. Regarding the frequency range of each excitation, the earthquake recordings are mainly within 0-5 Hz due to their natural characteristics. The full state vector $\mathbf{h}_{\mathrm{u}}(t) \in \mathbb{R}^{12\times1}$ expressed as $[x_b; x_1; x_2; x_3; x_4; x_5; \dot{x}_b; \dot{x}_1; \dot{x}_2; \dot{x}_3; \dot{x}_4; \dot{x}_5]$ are generated from the 5-storey numerical model under the known condition shown in Eq. 6-27; the limited observations $\mathbf{q}_{\mathrm{u}}(t)$ associated with the number

and location of their observation points are obtained, at the same sampling rate as the input excitation. Table 6-1 summarises the different limited observations for the 4 different cases used in this numerical example, including the different numbers and locations of observations.

$$\frac{\mathrm{d}\mathbf{h}_{I=0}(t)}{\mathrm{d}t} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}_{I=0} & -\mathbf{M}^{-1}\mathbf{C}_{I=0} \end{bmatrix} \mathbf{h}(t) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1} \end{bmatrix} \mathbf{u}(t) \qquad \text{Eq. 6-27}$$

In **step ii** of the proposed framework, the limited observation $\mathbf{q}_{\mathrm{d}}(t)$ under the unknown condition is required for reconstructing the discrepancy term between different conditions. Thus, $\mathbf{q}_{\mathrm{d}}(t)$ with the same numbers and observation location as $\mathbf{q}_{\mathrm{u}}(t)$ is collected to train the time operator. The dimensions for $\mathbf{q}_{\mathrm{d}}(t)$ different cases are summarised in Table 6-1.

Table 6-1. Cases of missing measurement and limited observation of the 5-storey numerical model.

Cases description	limited observation $\mathbf{q}_{\mathrm{u}}(t)$or $\mathbf{q}_{\mathrm{d}}(t)$
Case 1: Missing measurement $x_3, \dot{x}_3$	$[x_b; x_1; x_2; x_4; x_5; \dot{x}_b; \dot{x}_1; \dot{x}_2; \dot{x}_4; \dot{x}_5]$
Case 2: Missing measurement $x_b, \dot{x}_b$	$[x_1; x_2; x_3; x_4; x_5; \dot{x}_1; \dot{x}_2; \dot{x}_3; \dot{x}_4; \dot{x}_5]$
Case 3: Missing measurement $x_b, x_3, \dot{x}_b, \dot{x}_3$	$[x_1; x_2; x_4; x_5; \dot{x}_1; \dot{x}_2; \dot{x}_4; \dot{x}_5]$
Case 4: Missing measurement $x_b, x_2, x_3, \dot{x}_b, \dot{x}_2, \dot{x}_3$	$[x_1; x_4; x_5; \dot{x}_1; \dot{x}_4; \dot{x}_5]$

6.4.3. Construction of the spatial-feature encoder

The proposed framework constructs spatial feature representation relations to map full state to limited observation, which allows limited observations to be used directly to identify structural parameters in unknown states. Under the known condition of the structure, the full state $\mathbf{h}_{\mathrm{u}}(t)$ and limited observation $\mathbf{h}_{\mathrm{u}}(t)$ are the input and output of the spatial feature encoder, respectively. Thus, the architecture of the encoder network needs to be matched to the dimension of $\mathbf{h}_{\mathrm{u}}(t) \in \mathbb{R}^{12\times 1}$ and $\mathbf{q}_{\mathrm{u}}(t) \in \mathbb{R}^{n\times 1}$($n$=10, 8 and 6), which varies in different cases. Table 6-2 summarises the encoder network architecture for different cases. In addition, the corresponding training hyper-parameters are summarised in Table 6-2. It is noted that the relatively bigger learning rate 0.001 and fewer epochs 1000 were selected to train the encoder due to the simple structure of the encoder design being easy

to train. Finally, vectors containing 500 data points of length 5 s $\mathbf{h}_{\mathrm{u}}$ and $\mathbf{q}_{\mathrm{u}}(t)$are used to train the encoder.

Table 6-2. SE network architecture for different cases and training hyper-parameters used of the numerical example.

	Number of nodes				Training hyper-parameters	
Layer	Case1	Case 2	Case 3	Case 4	Name	Value
Input	12	12	12	12	Batch size	200
Output	10	10	8	6	Epoch	1000
					Learning rate	0.001

6.4.4. Reconstruction of the unknown discrepancy

In the step of reconstructing the discrepancy, time operators are trained to reconstruct the discrepancy terms of structure between different conditions, utilising the limited observations under the unknown condition of the structure and spatial feature representation relationships constructed in the known condition. The condition of the structure when the MRE base isolator is powered on (constant current I = 1A) is considered as an unknown condition and reconstructing this unknown discrepancy using the proposed method is a task. The time operator is made by an NN and healthy numerical model to form a NODE, in which the architecture of NN is only dependent on the dimension of the full state; Because it represents a discrepancy in the entire structure; and the healthy numerical model is the same as the physical knowledge used in **step i** expressed in Eq. 6-26. Table 6-3 summarises the architecture of NN and training hyper-parameters. It is important to note that due to the complexity of the NN structure, it requires large training epochs, lower learning rates, and more data to fit more parameters. The limited observation under the unknown structural condition $\mathbf{q}_{\mathrm{d}}(t)$ contained 1,000 data points of 10 seconds is used to train the time operator for reconstructing the discrepancy.

Table 6-3. The architecture of the NN network and training hyper-parameters of the numerical

example.

The architecture of NN		Training hyper-parameters	
Layer type	Number of nodes	Name	Value
Input	12	Batch size	200
Layer 1	24	Epoch	10000
Layer 2	24	Learning rate	0.00005
Output	6		

6.4.5. Reconstruction of the missing response and structure identification

Time operators trained by limited observations under unknown conditions can output the full state $\mathbf{h}(t)$ of the structure under unknown conditions and thus perform response reconstruction. In addition, as the structure is in an unknown condition with MRE isolator power on, this differs from the structure under the health condition, causing this discrepancy term to be stored in the NN, mentioned in section 2.2.4. This implies that the output of the NN should fulfil the following relationships:

$$\mathrm{G_{NN}}(\mathbf{h}(t)) = [0; 0; 0; 0; 0; 0; k_b x_b(t) + c_b \dot{x}_b(t) + \alpha x_b^3(t); 0; 0; 0; 0; 0] \quad \text{Eq. 6-28}$$

where $\mathrm{G_{NN}}(\mathbf{h}(t))$ as the discrepancy term contains the output of NN; $x_b(t)$ and $\dot{x}_b(t)$ is the displacement and velocity for the base floor of the structure; the unknown structural parameters k_b, c_b and α can be reconstructed by fitting the polynomial equations between the output of NN and $x_b(t)$ or $\dot{x}_b(t)$.

6.4.5.1. Result 1: scenario of noisy data

In this numerical example, a limited observation under different noise levels in **Case 1** is considered to validate the performance of the proposed framework, and then its corresponding results are discussed. Since observations usually contain noise, Gaussian white noise is only added to the limited observations $\mathbf{q}_{\mathrm{u}}(t)$ that are not in the full state $\mathbf{h}_{\mathrm{u}}(t)$. For the limited observation of noise pollution, $\mathbf{q}_{\mathrm{u\ noise}}(t) = \mathbf{q}_{\mathrm{u}}(t) + \mathcal{N}(\mu, \sigma)$ and $\mathbf{q}_{\mathrm{d\ noise}}(t) = \mathbf{q}_{\mathrm{d}}(t) + \mathcal{N}(\mu, \sigma)$ are under health condition and unknown condition, used to train the spatial feature encoder in **step i** and the time

operator in **step ii** respectively; where μ and σ are the mean value and standard deviation of the noise, where $\mu$is equal to 0; σ is equal to 0.03, 0.05 and 0.1 for 3%, 5% and 10% noise respectively.

After the completion of training in each step using limited observation of noise-polluted data, the missing responses including displacement $x_3(t)$, velocity $\dot{x}_3(t)$and acceleration $\ddot{x}_3(t)$ of **case 1** are reconstructed by time operators, respectively, shown in Figure 6-4 (a). It can be seen that the proposed framework is able to reconstruct the missed response under the training of a limited number of observations adding different white noises. The reconstructed signals almost overlap with the real responses when the noise pollution is less than 5%, while at 10% noise the reconstructed signals show less deviation from the real response; Figure 6- 4(b) shows a violin plot for the Mean-square (MSE) error, where contains a white box (interquartile (IQR) range: cover the difference between the Q1 and Q3 (25^{th} and 75^{th} percentiles of the data)), a black line in the mid of the white box (the mean value of the data), and two symmetrical black lines outside of the white box (reach lower and upper wisher: Q1-1.5IQR and Q3+1.5IQR, cover 99.65% of the data). The MSE loss between the reconstructed data and the real response and its accuracy relative to the training model decreases as the noise level increases.

For the structural identification, the nonlinear discrepancy term presented in Eq. 6-28 has been stored by the neural network and is plotted in Figure 6-4 (c); From this Fig 6-4 (c); . , the base floor displacement $x_b(t)$ and the first output of the NN are plotted at different noise levels and compared with the reference values. It can be clearly seen that after training with noise interference with 0%, 3% and 5% levels, the NN accurately captures the nonlinear MRE term of the model in Case 1, it has the largest offset at 10 % noise. Then, the polynomial regression was used to explore expressions

related to these three variables for structural parameter identification, and identified nonlinear expressions were summarized in Table 6-4, in which the maximum error of these parameters for each noise level is 1.5%, 5.7%, 10.9 and 67.8%, respectively. The maximum error occurs when the noise level is 10% since the time operator will not be able to resist this high level of noise interference to accurately capture the nonlinear discrepancy term. With only a limited number of observations, the nonlinear unknown discrepancy terms in the structure can still be accurately captured by the NN under the noise below 10%, and missing response can be reconstructed from the time operator, due to the spatial feature representation relation constructed based on the encoder. It can be drawn that when the noise level is below 10%, the proposed method with limited measurement observation can still use spatial feature relations to train a neural network with physical knowledge to infer missing signals and identify structure parameters. Spatial relations combine this a priori structural knowledge to maximise the use of physical principles, increase the reliability of the framework, and increase the interpretability of the neural network.

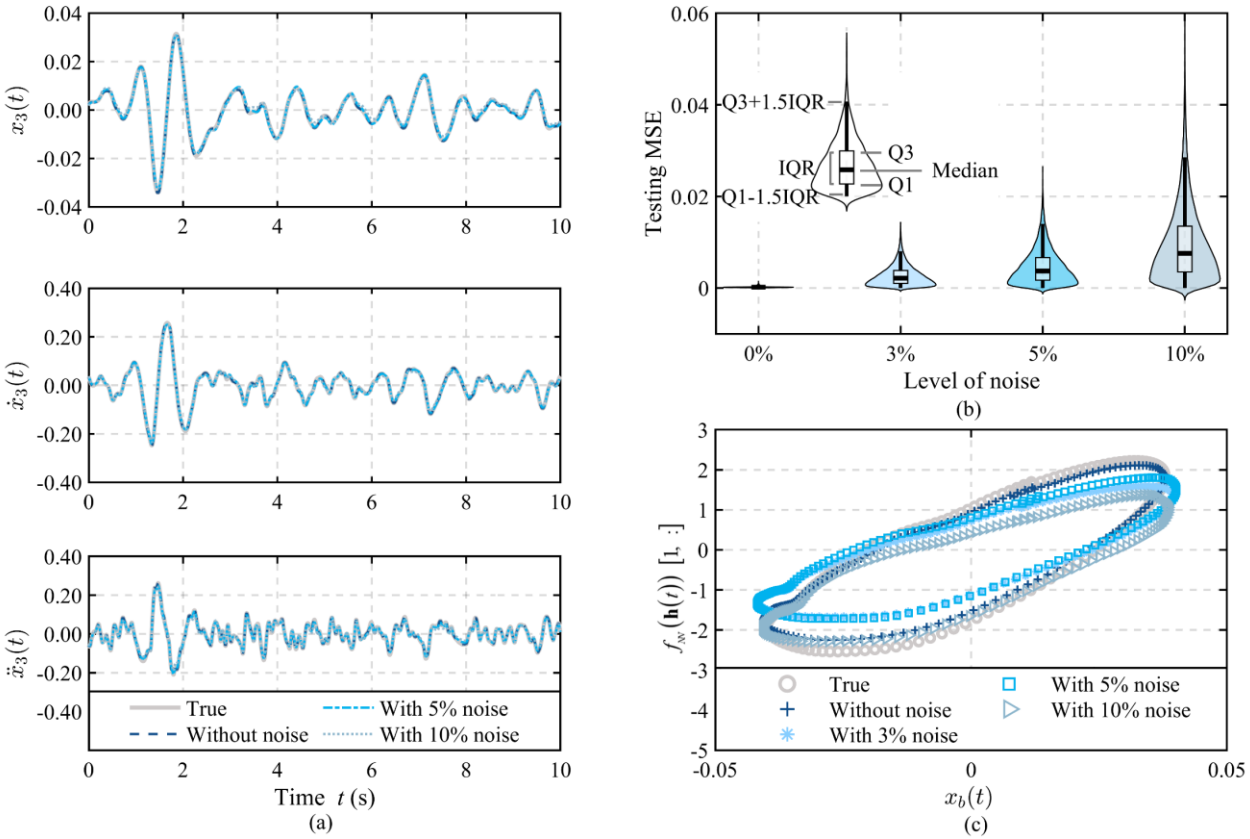


Figure 6-4. Numerical example result 1: scenario of noisy data (a) reconstructed missing response; (b) Testing MSE loss (c) reconstructed discrepancy term.

Table 6-4. Numerical example Result 1: scenario of noisy data - Identified model expression $f_{NN}(\mathbf{h}(t))$.

Case 1	$f_{NN}(\mathbf{h}(t))$ [1, :]	parameters in the discrepancy model $k_b x_b(t) + c_b \dot{x}_b(t) + \alpha x_b^3(t)$	Identified error (%)		
			k_b	c_b	α
True	$\ddot{x}_b$	$322.48x_b + 26.4\dot{x}_b + 28.97{x_b}^3$	-	-	-
Without noise	$\ddot{x}_b$	$323.1x_b + 26.8\dot{x}_b + 28.6{x_b}^3$	0.2	1.5	1.3
3 % noise	$\ddot{x}_b$	$321.2x_b + 24.9\dot{x}_b + 27.9{x_b}^3$	0.4	5.7	3.7
5 % noise	$\ddot{x}_b$	$299.9x_b + 23.6\dot{x}_b + 25.8x_b^3$	7.0	10.6	10.9
10% noise	$\ddot{x}_b$	$103.8x_b + 30.4\dot{x}_b + 10.1x_b^3$	67.8	15.2	65.1

6.4.5.2. Results 2: scenario of different observation period (numbers of points)

There is a further discussion of the performance of the proposed method with different observation periods. In practice, the observation of the structure under the health condition $\mathbf{q}_u(t)$ can be easily obtained, while that under unknown conditions $\mathbf{q}_d(t)$ is usually limited. Thus, the various data lengths of observation used in training are only considered in **step ii,** when using the

limited observation under the unknown condition $\mathbf{q}_{\mathrm{d}}(t)$to construct the discrepancy term. In case 1, the time operator in step ii is trained using time durations of 10 s (number of points=1000) of the limited observation under the unknown condition. Shorter training time data with 1s and 5s (number of points =100 and 500) of the limited observation is also used for comparison of the performance for the missing responses reconstructed by the proposed method.

Figure 6- 5(a) shows the results of reconstructing the missing response of 3rd floor displacement $x_3(t)$, velocity $\dot{x}_3(t)$ and acceleration $\ddot{x}_3(t)$ using different observation period training time operators and comparing them with the true response. In addition, the MSE loss is shown in Figure 6-5 (b). From Figure 6-5 (a) and (b), it can be observed that the performance of the time operator is enhanced with the increase in the training data length. When the training data over 5s reaches 10s, there is less increase in performance that can be found since the accuracy of the reconstructed responses has already reached a stable level.

For the nonlinear MRE force identification, predicted data beyond the training data were used for identification due to the short observation times of 1s and 5s. Figure 6-5(c) shows the first output of the NN vs. base floor displacement out of the training range, compared to the ground truth. It can be seen that when the training data points of the observation are sufficiently long and reach 5 s, the NN successfully captures the main characteristics of the MRE model's nonlinearity.

Some conclusions can be obtained from the results: first, the proposed method has the potential to achieve reconstruction and identification with a much smaller number of training samples compared with the whole observation period, which proves the robustness and efficiency in practice; second, unlike the conventional data-driven learning method that requires a large number of training

data, the proposed approach reveals that by the fusion of NN and prior physics, the cost of the learning can significantly drop but it can still maintain a stable even better performance.

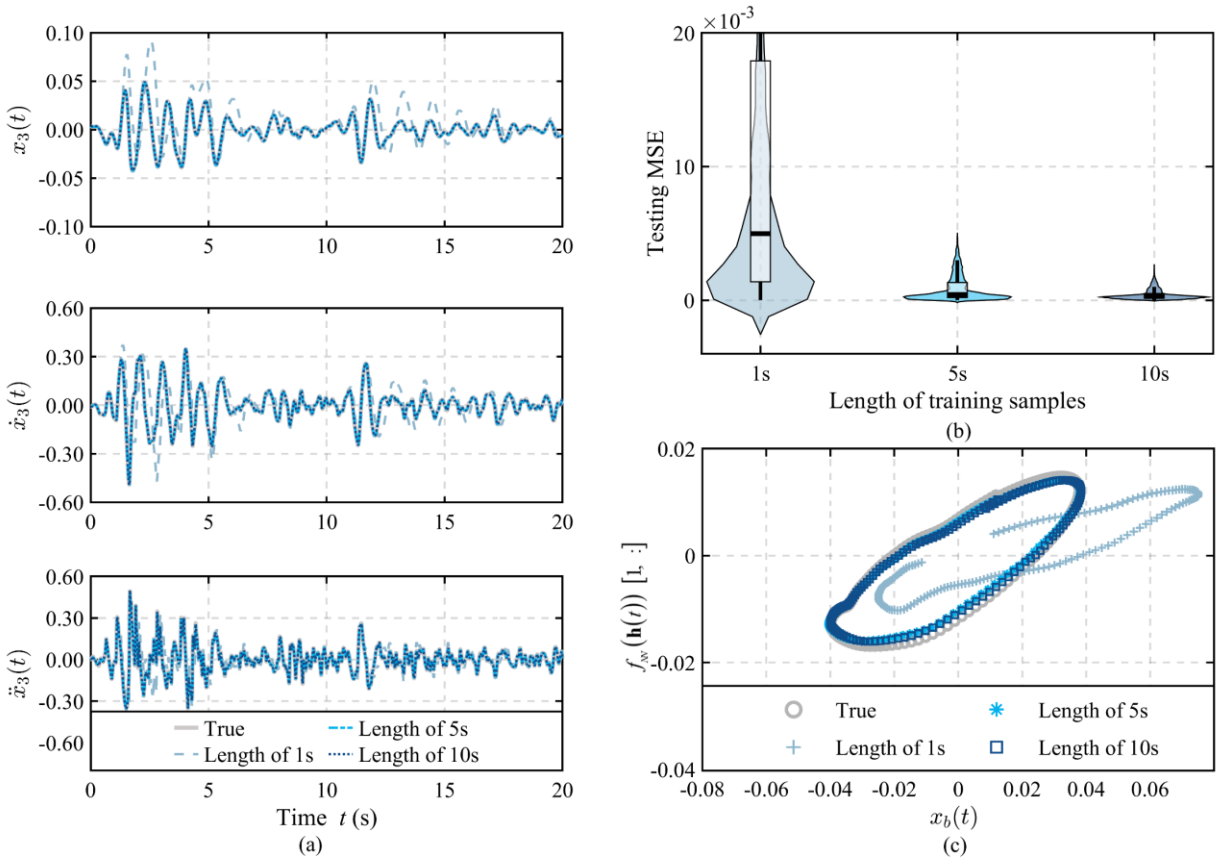


Figure 6-5. Numerical example result 2: scenario of noisy data (a)reconstructed missing response; (b) Testing MSE loss; (c) reconstructed discrepancy term.

6.4.5.3. Results 3: scenario of different observation locations

In real situations, different types of limited-observation states of the structure can occur, such as the states of some stories are unseen. The position and number of observation locations are changed to obtain several different limited observations in order to verify the performance of the proposed method for response reconstruction and structural parameter identification. It is worth noting that reconstructing the response of floors with MRE isolators is important and a principal indicator to evaluate the performance of identifying system parameters. Thus, this scenario focuses on reconstructing the response of the base floor in this section. In Case 2, the limited observations are

the responses of the other floors except the ground layer. Based on Case 2, the missing response of the 3rd floor is considered to form Case 3. Besides, the missing response from the 2nd and 3rd floors simultaneously is considered to form Case 4, summarised in Table 6-1.

As can be seen in Figure 6-6 (a), the missing ground response $x_b, \dot{x}_b$ and $\ddot{x}_b(t)$ is reconstructed by the time operator trained with a limited observation of 10 seconds. From the reconstructed response, the time operator accurately reconstructs the missed ground response, even using limited observations that miss the ground, 2nd and 3rd floor responses. Figure 6-6 (b) shows the MSE loss between the reconstructed and real signals for cases 3, 4, and 5. As the number of observations decreases, the MSE loss for cases 3 and 4 becomes slightly larger than that for case 2; in general, most of the values are in the small range of 0-0.0015, indicating that the timer operator has been well trained. Figure 6-6(c) shows the first output of the, NN versus the bottom displacement in the training range compared to the ground truth. It can be seen that the NN captures the main features of the MRE model in terms of nonlinearity for all three cases with limited observations. The proposed framework can reconstruct the missing base floor response, based on data from numerical simulations in which the number of model observations is only half of the full state.

The proposed framework can reconstruct the missed response, based on data from numerical simulations in which the number of model observations is only half of the full state. This demonstrates that the spatial feature representation relationship can be well bridge the knowledge of the physical structure and a limited observations, enabling the limited observations to be directly utilised for structural parameter identification and then for missing response reconstruction.

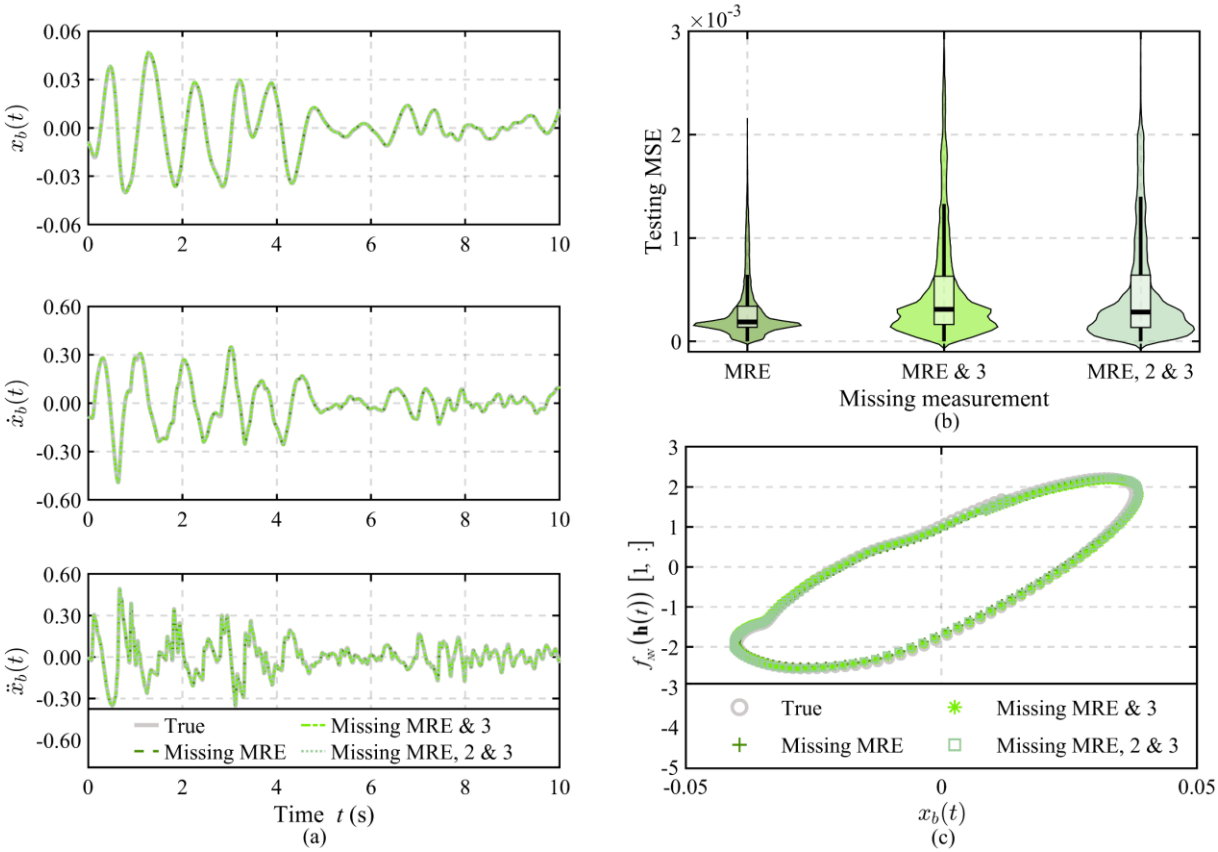


Figure 6-6. Numerical example result 3: scenario of different observation locations (a)reconstructed missing response; (b) Testing MSE loss; (c) reconstructed discrepancy term.

6.5. Experimental Example - 3-storey frame structure with MRE isolators

In this section, the efficiency of the proposed framework is validated by a 3-storey frame structure with two identical MRE base isolators. The basic information of the experiment is introduced first, including the structure's physical parameters and data acquisition system. Then, the training process of the proposed framework is introduced sequentially. Lastly, the results and discussion of structural identification and response reconstruction under different cases of various observation locations are presented.

6.5.1. Basic information

As shown in Figure 6-7, a 3-storey frame structure is designed with a total height of 1200 mm and each floor height is 400 mm, and more details can be found in (Gu, n.d.; Gu et al., 2017). Four

steel strips have a cross-section of 344 mm to provide a weak stiffness in the weak axis and strong stiffness along the strong axis. Each floor is made by a mass plate with dimension of $600 \times 500 \times 40$ mm, each of which are connected with the four strips using clamp and countersunk head screws, respectively. A bottom plate is connected to the bottom of the four steel strips with patches and screws for fixing the frame model. Two identical MRE isolators are placed under the frame model with hight of 112 mm and connected with the bottom plate. The MRE base isolator in the experimental model is based on the Kelvin model, which has been expressed in Eq. 6-24 and introduced in section 3.1. The material properties of the frame structure can be summarized as in Table 6-6.

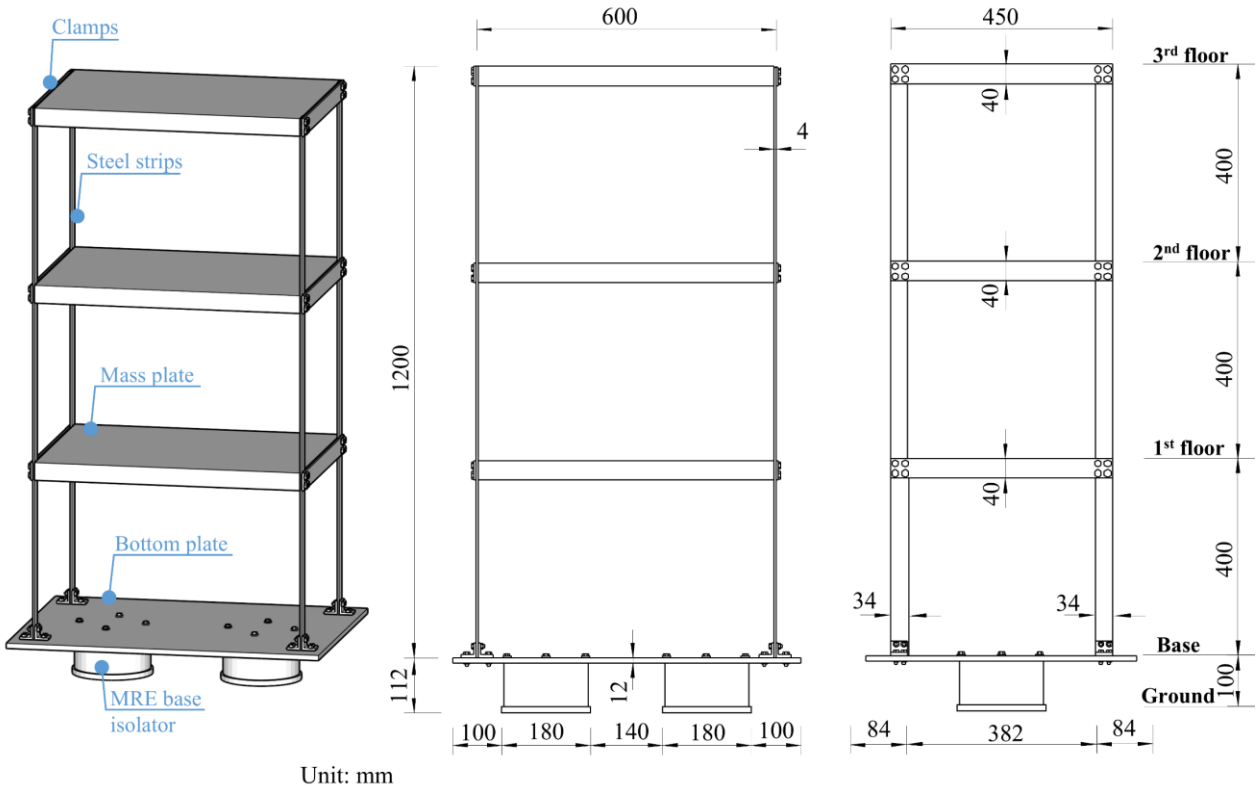


Figure 6-7. Dimensions of the 3-storey experimental structure: (a) 3-dimensional view, (b) front view, and (c) side view.

Table 6-5. The material properties for each component of the 3-storey experimental structure.

Name	Number	Material	Density (kg/m^3)	Weight (kg)
Mass plate	3	Aluminum	2.70×10^3	14.58
Strip	2	Steel	7.85×10^3	1.28
Clamp	6	Steel	7.85×10^3	0.57
Bottom plate	1	Steel	7.85×10^3	61
MRE base isolator	2	-	-	55

The experimental set up data acquisition system of the frame structure is shown in Figure 6-8. Four laser sensors (OADM 20I4460/S14C) are installed in the reference wall for measuring the displacement of each floor (x_b, x_1, x_2, x_3) for this structure, and one extract accelerometer is placed on the side of the shake table to obtain the ground excitation acceleration $\mathbf{u}(t)$. Sequentially, the built-in digital controller of the data acquisition system calculates the required current and the corresponding pulse-width modulation (PWM) signals, the PWN current driver outputs the corresponding current signals to the wire connector to control the parameters c_b and k_b of the MRE base isolators. In this experimental example, I = 0 and 1A are only considered as constant currents which are directly given to the data acquisition system without calculation;

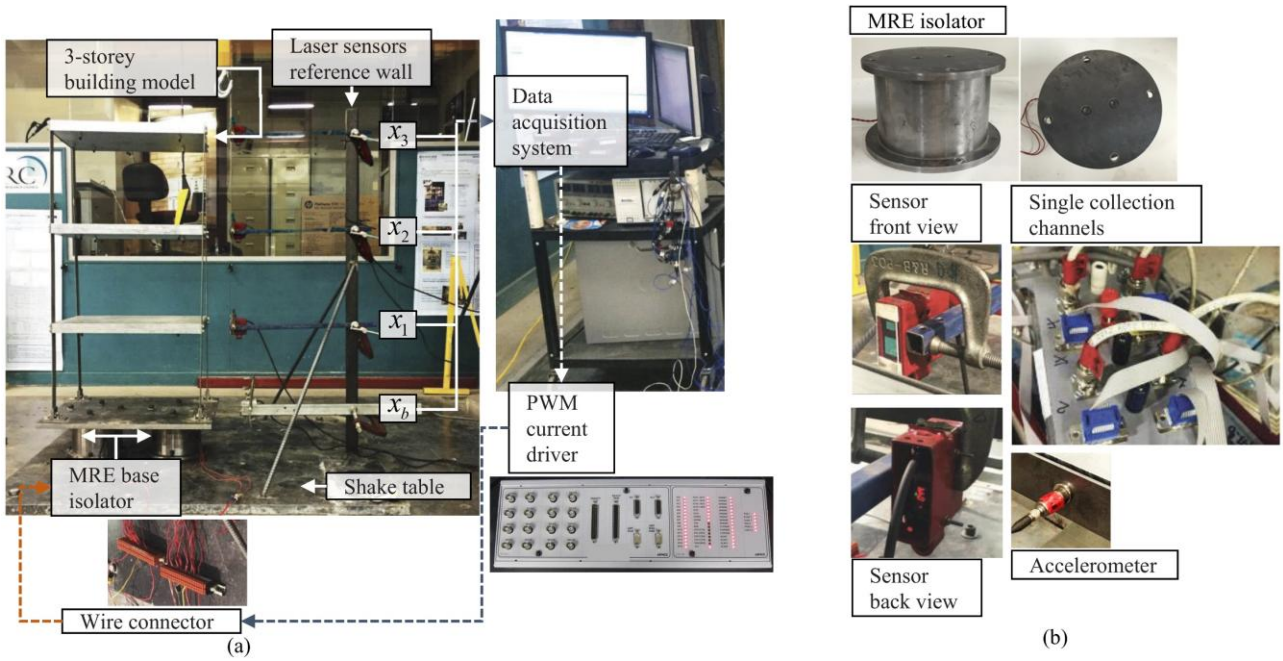


Figure 6-8. (a) Experimental setup and data acquisition system (b) details of the experimental equipment.

Similar to the numerical example, the current on the MRE isolator I=0 A and the isolator is at its softest state, the known condition of the experimental structure can be also expressed as a lumped mass model in Eq. 6-27, while $\mathbf{K}_{I=0} \in \mathbb{R}^{4\times4}$ and $\mathbf{C}_{I=0} \in \mathbb{R}^{4\times4}$ are replaced by corresponding physical parameters from the 4-DOF model. To obtain the accurate $\mathbf{K}_{I=0}$ and $\mathbf{C}_{I=0}$ of the experimental structure under the known condition, impulse hammer testing has been conducted for modal identification of the frame structure under the condition I=0 A. After this, the MATLAB toolbox (Damage Identification and Modal Analysis for Dummies) is used to obtain the updated modal parameters. Thus, the stiffness and damping matrixes updated from the modal analysis are shown in Table 6-7. There is also the comparison of the natural frequency of the structure for the designed, updated and measured results. It can be found that after the model updating, compared with the designed natural frequency, the that of the experimental structure is closer to the measurement.

Table 6-6. The updated stiffness and damping matrices and the natural frequencies of the 3-storey experimental structure.

Updated matrix		Natural frequency (Hz)			
		Number of mode	Designed value	Updated value	Measurement
Stiffness matrix	$\mathbf{K}_{I=0} = \begin{bmatrix} 32147 & -25603 & 0 & 0 \\ -25603 & 44825 & -19222 & -905 \\ 0 & -18407 & 39401 & -19631 \\ 0 & -238 & -20874 & 21052 \end{bmatrix}$ (N/m)	1st	0.961	1.015	1.166
		2nd	3.039	3.086	3.231
Damping matrix	$\mathbf{C}_{I=0} = \begin{bmatrix} 334.044 & -8.644 & 0 & 0 \\ -8.644 & 13.162 & -4.518 & -0.403 \\ 0 & -4.518 & 12.028 & -5.613 \\ 0 & -0.403 & -5.613 & 8.944 \end{bmatrix}$ (N·s/m)	3rd	5.604	5.803	5.878
		4th	7.308	7.449	7.510

6.5.2. Data preparation

To identify the discrepancy term of the frame structure between the known and the unknown conditions based on limited observation, a 3-storey model of MRE isolators under the powered off (I = 0A) condition as the known condition in Eq. 6-27 is considered the known physical knowledge used in **steps i** and **ii** of the proposed frameworks.

In **step i**, the full state $\mathbf{h}_{\mathrm{u}}(t)$ and limited observation $\mathbf{q}_{\mathrm{u}}(t)$ of the structure under known condition are required to construct the spatial feature relationship. The earthquake excitations $\mathbf{u}(t)$, including E1, E2, E3 and E4 mentioned in section 3.1 as input, with the sampling rate of 100 Hz, which satisfies the requirement of exceeding the maximum frequency of the frame structure twice ensure that the response contains the main information about its structure. They are given to the 3-storey numerical model under the known condition (I = 0) to generate the full state $\mathbf{h}_{\mathrm{u}}(t) = \left[x_b; x_1; x_2; x_3; \dot{x}_b; \dot{x}_1; \dot{x}_2; \dot{x}_3\right]$; The same excitations are given to the shake table to excite the experimental structure under the known condition (applying consistent current I = 0). The

displacements of the observations are obtained from the sensors at the same sampling rate, while the velocities are obtained from their derivation. A limited observation is obtained by removing the response of the associated missing sensors in this observation. Therefore, the limited observations of the experimental structure at different observation positions under the known condition are summarised in Table 6-7.

In **step ii,** limited observation $\mathbf{q}_{\mathrm{d}}(t)$ from the experimental structure under unknown conditions is required to reconstruct the discrepancy with the health condition. Three different unknown conditions are set by applying the current of the MRE isolator, *I*=1A, the observations are collected from the sensors, and the limited observations $\mathbf{q}_{\mathrm{d}}(t)$ under different conditions corresponding to each case are formed as summarised in Table 6-7.

Table 6-7. Cases of missing measurement and limited observation of the 3-storey experimental structure.

Cases description	limited observation $\mathbf{q}_{\mathrm{u}}(t)$or $\mathbf{q}_{\mathrm{d}}(t)$
Case a: Missing measurement $x_b, x_1, \dot{x}_b, \dot{x}_1$	$[x_2; x_3; \dot{x}_2; \dot{x}_3]$
Case b: Missing measurement $x_b, x_2, \dot{x}_b, \dot{x}_2$	$[x_1; x_3; \dot{x}_1; \dot{x}_3]$
Case c: Missing measurement $x_b, x_3, \dot{x}_b, \dot{x}_3$	$[x_1; x_2; \dot{x}_1; \dot{x}_2]$

6.5.3. Construction of the spatial-feature encoder

The proposed framework constructs a spatial feature representation relation using an encoder network that maps the full state to a limited observation, and therefore, the limited observation can be directly used to identify the structural parameters in unknown conditions. In this practical application, the powered off (current *I*=0A) of the structure is a known condition, and limited observations $\mathbf{q}_{\mathrm{u}}(t)$of the real structure can be collected from the measurement. Under the same condition, the full state $\mathbf{h}_{\mathrm{u}}(t)$is generated from the numerical model. Then, the spatial feature encoder is trained using the structure's full state $\mathbf{h}_{\mathrm{u}}(t)$ and limited observations $\mathbf{q}_{\mathrm{u}}(t)$as input and

output along the time series, respectively. Therefore, the encoder network consists of a fully connected layer and its structure needs to match the dimensions of $\mathbf{h}_{\mathrm{u}}(t) \in \mathbb{R}^{8\times1}$and $\mathbf{q}_{\mathrm{u}}(t) \in \mathbb{R}^{4\times1}$. Table 6-8 summarises the structure of the encoder network and the corresponding training hyperparameters for this experimental example. Finally, $\mathbf{h}_{\mathrm{u}}(t)$and $\mathbf{q}_{\mathrm{u}}(t)$under the excitation E1 the datasets containing 500 data points of length 5s are used to train the spatial encoder for three different cases respectively.

Table 6-8. SE network architecture for different cases and training hyper-parameters used in the experimental example.

Layer	Number of nodes	Training hyper-parameters	
Input	8	Name	Value
Output	4	Batch size	200
		Epoch	1000
		Learning rate	0.001

6.5.4. Reconstruction of the unknown discrepancy

In the reconstructing discrepancy step, the limited observations under the unknown condition $\mathbf{q}_{\mathrm{u}}(t)$ of the real structure and the spatial feature representation relations constructed in known conditions are employed to train the timing operator to reconstruct the structural discrepancy between the unknown and known condition. The experimental structure when the MRE base isolator is powered on (constant current I = 1A) is considered to be in an unknown condition, and the proposed approach is required to reconstruct this unknown discrepancy. The time operator is a NODE consisting of NN and a numerical model under the known condition; where NN represents the discrepancy of the whole structure, the architecture of which depends on the dimensions of the full state, and the numerical model under the known condition is the same as the physical knowledge used in **step i**, as shown in Eq.6-27. Table 6-9 summarises the structure and training hyperparameters of the NN used for the experimental examples. The$\mathbf{q}_{\mathrm{d}}(t)$under the excitation E1 contains 1,000 data

points over 10 s, which are used to train the time operator to reconstruct the discrepancy.

Table 6-9. The architecture of the NN network and training hyper-parameters of the experimental example.

The architecture of NN		Training hyper-parameters	
Layer type	Number of nodes	Name	Value
Input	8	Batch size	200
Layer 1	16	Epoch	10000
Layer 2	16	Learning rate	0.00005
Output	4		

6.5.5. Reconstructing the missing response and structural identification

In the same manner as the numerical example, the experimental structure is treated as the unknown condition when the MRE isolation unit is powered on. Thus, the discrepancy term between the known and unknown conditions is stored in the NN, leading to the input and output of the NN fulfilling the relationship expressed in Eq. 6-28.

6.5.5.1. Result: scenario of different observation locations

The location of the observations where sensors can be placed may vary depending on the real situation. The performance of the proposed method for response reconstruction and structural parameter identification is evaluated by changing the observation locations. Under the structure's unknown conditions, the reconstruction of the response of a floor equipped with MRE isolation devices is the main reference for evaluating the system's parameter identification performance. Thus, in several cases in this experimental example summarised in Table 6-7, the number of limited observations is half of the full state and the missing responses always contain the based floor; in cases a, b and c, the response is missed at the base floor & 1st floor, base floor & the 2nd floor, and at the base floor & the 3rd floor, respectively.

As can be seen in Figure 6- 9(a), the time operator reconstructs the missing ground response of

10 s under the excitation E1 using a limited number of observations for training. Figure 6- 9(b) shows the MSE loss between the reconstructed signal and the true signal in cases a, b, and c. From Figure 6- 9(b), it can be found that with the change of observation location, most of the MSE values in cases a, b and c are within 0-0.0001, 0-0.0002 and 0-0.00025, respectively. Moreover, when using the limited observation case a and b, the time operator accurately reconstructs the missed ground response compared to the true response. However, in the case of limited observations using case c, the reconstructed ground response has a larger error than the true response, the analysis of the reason can be found below. Figure 6- 9(c) shows the first output of the NN with the reconstructed bottom displacement compared to the real results, and the identified nonlinear expressions are summarized in Table 6-10. In this table, the errors between the identified and true parameters show that the maximum error for each case is 8.389%, 32.540%, 54.906%, respectively. It can be found that the NN accurately capture the nonlinear characteristics of the MRE model powered on in case b with less error using limited observations.

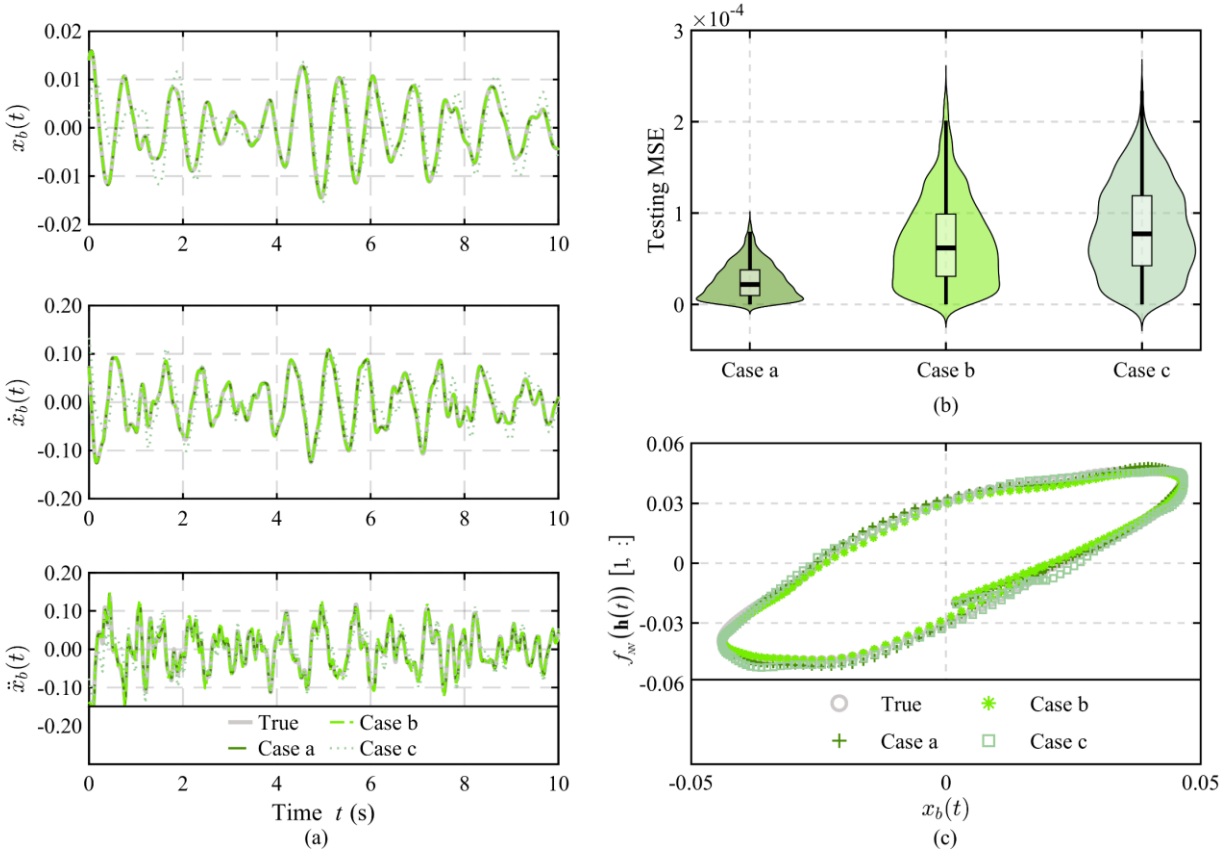


Figure 6-9. Experimental example result 1: scenario of different observation locations (a)reconstructed missing response of the base floor; (b) Testing MSE loss; (c) reconstructed discrepancy term.

Table 6-10. Experimental example result: scenario of noisy data - Identified model expression$f_{NN}(\mathbf{h}(t))$.

Case	$f_{NN}(\mathbf{h}(t))[1,:]$	parameters in the discrepancy model $k_b x_b(t) + c_b \dot{x}_b(t) + \alpha x_b^3(t)$	Identified error (%)		
			k_b	c_b	α
True	$\ddot{x}_b$	$8.062x_b + 0.660\dot{x}_b + 0.613{x_b}^3$	-	-	-
Case a	$\ddot{x}_b$	$7.985x_b + 0.605\dot{x}_b + 0.645{x_b}^3$	0.955	8.389	5.152
Case b	$\ddot{x}_b$	$7.145x_b + 0.459\dot{x}_b + 0.813{x_b}^3$	11.374	30.497	32.540
Case c	$\ddot{x}_b$	$7.002x_b + 1.023\dot{x}_b + 0.456x_b^3$	13.148	54.906	25.660

Besides, to test the extensibility of the time operator, the missing responses of the 1st, 2nd and 3rd floor in cases a, b and c were reconstructed under the excitation out of the training range, separately. The unseen earthquake excitations E2, E3 and E4 were given to pre-trained time operators of three cases, respectively, and the results are presented in Figure 6-10. Similar to the performance of each case in Figure 6- 9 (a), it can be seen that the reconstructed displacements in case c have some

shifting occurrences and oscillations compared to the true displacement, while that in case a and b is very close to the true displacement. Variation of timer operator performance due to different observations under multiple cases may be due to the fact that when the number of missed signals reaches half of the number of full states, and the positions between them are overly far, the spatial mapping relationship constructed is not accurate enough to precisely project the full state to a limited observation. More observation locations may be required to construct a more accurate spatial relation. In summary, missing responses of the experimental frame structure under unknown conditions are reconstructed, and structure parameters are identified based on limited observations.

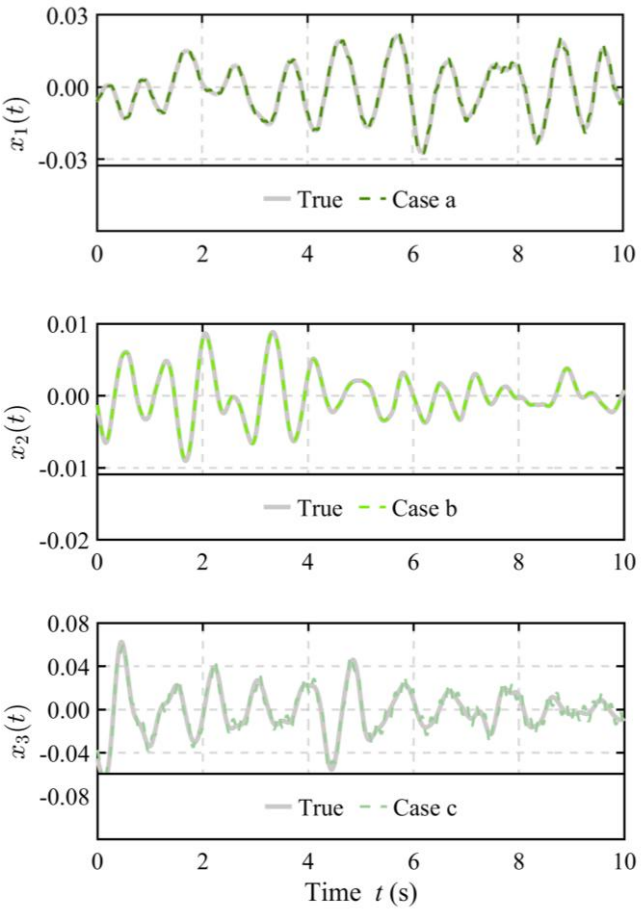


Figure 6-10. Experimental example result 1: scenario of different observation locations: reconstructed missing response of the 1st, 2nd and 3rd floor.

6.6. Summary

This chapter presents a novel PhySTN architecture used for the reconstruction of missing response and identification parameters in the nonlinear structure under the unknown condition using limited observations. The PhySTN model consists of spatial encoders and NODEs that act as spatial feature representation relationships and time operators, respectively. The spatial feature representation relationship addresses the limitation that most current methods are unable to identify structural parameters using limited observations, and it enables the limited observation directly used to reconstruct the discrepancy term represented by NN. Due to the knowledge of the structural physics contained in the time operator exploited to the fullest extent, enabling the neural networks into which this knowledge is embedded to represent nonlinear difference terms under structurally unknown conditions only based on limited observation.

A numerical study using a nonlinear 5-storey building model and an experimental study using a 3-storey framework structure are carried out to validate the performance of the proposed framework. The results of numerical and experimental example show that the proposed PhySTN can reconstruct the missing response and identify the structural parameters based on limited observations. Several conclusions can be made from the results:

(i) By considering the spatial feature representation between the full state and the limited observations, the limited observations from the measurement can be directly used for response reconstruction and structural parameter identification. (ii) Time operator combined with spatial feature representation relations, using a limited number of observations to reconstruct the discrepancy terms between different states of the structure; even if the limited observations are half of the full state, the missing responses can be reconstructed accurately, then, accurately identifying the unknown

structural parameters. (iii) In the proposed approach, using limited observational data, the integration of structural physics knowledge with the neural network allows the parameters of the structural discrepancy terms to be obtained by a regression task that fits the inputs and outputs of the neural network, which provides the neural network with interpretability and improves its reliability.

Chapter 7. Conclusions and future research

7.1. Conclusions

This thesis presents a comprehensive exploration of advanced methods for structural condition assessment under practical constraints, addressing key challenges in the field.

1. Chapter 3 proposes a transfer learning-based method leveraging frequency response functions (FRFs) for damage detection. By pre-training a convolutional neural network on simulated data and fine-tuning it with limited real measurements, this approach significantly improves damage localization and severity quantification.

2. Chapter 4 introduces a joint maximum discrepancy and adversarial discriminative domain adaptation (JMDAD) method, enabling accurate damage detection without labeled target data. The approach effectively bridges domain gaps caused by modeling errors and environmental variations.

3. Chapter 5 develops a parallel neural ordinary differential equations (PNODEs) framework, integrating physical constraints to enhance the interpretability and reliability of damage detection models. This method demonstrates robust performance in both numerical and experimental validations.

4. Chapter 6 presents a temporal-spatial neural operator (PhySTN) framework, combining physical knowledge and limited observations to reconstruct structural parameters and responses. This innovative method effectively addresses challenges in sparse sensor data scenarios.

Overall, this thesis advances structural condition assessment by integrating machine learning,

transfer learning, and physics-informed methodologies. The proposed frameworks enhance accuracy, interpretability, and robustness, offering practical solutions for real-world structural health monitoring challenges.

7.2. Recommendations for future works

Building upon the findings of this thesis, several recommendations for future research can be outlined:

1. Data Challenges: Future efforts should focus on improving the quality and accessibility of structural health monitoring (SHM) data, especially in obtaining labeled damage datasets for diverse operational scenarios. Exploring synthetic data generation and advanced simulation techniques can also address data scarcity challenges.

2. Scalability and Real-time Applications: While the proposed methods have demonstrated effectiveness on mid-rise structures and benchmark frames, their applicability to more complex or large-scale structures may require further adaptations. Future work could explore extending these frameworks to high-rise buildings, irregular structures, or infrastructure systems, incorporating additional physical constraints and domain adaptation techniques to enhance generalization.

3. Interpretable Models: Enhancing the interpretability and explainability of machine learning methods will bridge the gap between academic research and practical engineering applications, fostering broader adoption in the SHM industry.

These future directions aim to address remaining gaps and ensure the continued advancement of SHM methodologies.

Reference

Abdeljaber, O., Avci, O., Kiranyaz, M.S., Boashash, B., Sodano, H. and Inman, D.J. (2018) '1-D CNNs for Structural Damage Detection: Verification on a Structural Health Monitoring Benchmark Data', Neurocomputing, 275, pp. 1308–1317. https://doi.org/10.1016/j.neucom.2017.09.069.

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

Akrim, A., Gogu, C., Vingerhoeds, R. and Salaün, M. (2023) 'Self-Supervised Learning for data scarcity in a fatigue damage prognostic problem', Engineering Applications of Artificial Intelligence, 120, p. 105837. doi: 10.1016/j.engappai.2023.105837.

Alvin, K.F., Robertson, A.N., Reich, G.W. and Park, K.C. (2003) 'Structural system identification: From reality to models', Computers & Structures, 81(12), pp. 1149–1176. doi: 10.1016/S0045-7949(03)00034-8.

Avci, O., Abdeljaber, O., Kiranyaz, S., Hussein, M., Gabbouj, M. and 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, p. 107077.

Azam, S.E., Chatzi, E. and Papadimitriou, C. (2015) 'A dual Kalman filter approach for state estimation via output-only acceleration measurements', Mechanical Systems and Signal Processing, 60–61, pp. 866–886. https://doi.org/10.1016/j.ymssp.2015.02.001.

Azimi, M. and Pekcan, G. (2019) 'Structural Health Monitoring Using Extremely Compressed Data through Deep Learning', Computer-Aided Civil and Infrastructure Engineering. https://doi.org/10.1111/mice.12517.

Azimi, M., Eslamlou, A.D. and Pekcan, G. (2020) 'Data-Driven Structural Health Monitoring and Damage Detection through Deep Learning: State-of-the-Art Review', Sensors, 20(10), p. 2778.

Baybordi, S. and Esfandiari, A. (2022) 'A novel sensitivity-based finite element model updating and damage detection using time domain response', Journal of Sound and Vibration, 537, p. 117187.

Brincker, R., Olsen, P., Amador, S., Juul, M., Malekjafarian, A. and Ashory, M. (2019) 'Modal participation in multiple input Ibrahim time domain identification', Mathematics and Mechanics of Solids, 24(1), pp. 168–180. doi: 10.1177/1081286517733034.

Butcher, J. (2016) Numerical Methods for Ordinary Differential Equations. p. 513. doi: 10.1002/9781119121534.

Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.H., Strope, B. and Kurzweil, R. (2018) 'Universal Sentence Encoder (1773 citation(s); No. arXiv:1803.11175)', arXiv. Available at: http://arxiv.org/abs/1803.11175.

Cha, Y.J., Ali, R., Lewis, J. and Büyüköztürk, O. (2024) 'Deep learning-based structural health monitoring', Automation in Construction, 161, p. 105328. doi: 10.1016/j.autcon.2024.105328

Chen, H. and Ni, Y. (2018) Structural Health Monitoring of Large Civil Engineering Structures. 1st edn. Wiley. doi: 10.1002/9781119166641.

Chen, R.T.Q., Rubanova, Y., Bettencourt, J. and Duvenaud, D.K. (2018) 'Neural Ordinary Differential Equations', in Advances in Neural Information Processing Systems, 31. Available at: https://proceedings.neurips.cc/paper/2018/hash/69386f6bb1dfed68692a24c8686939b9-Abstract.html.

Chen, W.H., Lu, Z.R., Lin, W., Chen, S.H., Ni, Y.Q., Xia, Y. and Liao, W.Y. (2011) 'Theoretical and experimental modal analysis of the Guangzhou New TV Tower', Engineering Structures, 33(12), pp. 3628–3646.

Chen, Z., Wang, C., Wu, J., Deng, C. and Wang, Y. (2023) 'Deep convolutional transfer learning-based structural damage detection with domain adaptation', Applied Intelligence, 53, pp. 5085–5099. doi: 10.1007/s10489-022-03713-y.

Chen, Z., Zhang, J., Arjovsky, M. and Bottou, L. (2019) 'Symplectic Recurrent Neural Networks', arXiv. Available at: https://arxiv.org/abs/1909.13334v2.

Chesné, S. and Deraemaeker, A. (2013) 'Damage localization using transmissibility functions: A critical review', Mechanical Systems and Signal Processing, 38(2), pp. 569–584.

Christensen, R.M. (2003) Theory of viscoelasticity. 2nd edn. Dover Publications.

Cooijmans, T., Ballas, N., Laurent, C., Gülçehre, Ç. and Courville, A. (2016) 'Recurrent Batch Normalization', in International Conference on Learning Representations.

Correa, J.C.A.J. (2014) Parameter identification and monitoring of mechanical systems under nonlinear vibration. Elsevier Science. Available at: https://books.google.com.au/books?id=m9FzAwAAQBAJ.

Cremona, C.F. and Brandon, J.A. (1992) 'On recursive forms of damped complex exponential methods', Mechanical Systems and Signal Processing, 6(3), pp. 261–274. doi: 10.1016/0888-3270(92)90029-I.

Cunha, A. and Caetano, E. (2006) 'Experimental Modal Analysis of Civil Engineering Structures', Sound and Vibration, 40.

Cury, A. and Crémona, C. (2012) 'Pattern recognition of structural behaviors based on learning algorithms and symbolic data concepts', Structural Control and Health Monitoring, 19(2), pp. 161–186. doi: 10.1002/stc.412.

Ding, Z., Li, J. and Hao, H. (2020) 'Non-probabilistic method to consider uncertainties in structural damage identification based on Hybrid Jaya and Tree Seeds Algorithm', Engineering Structures, 220, p. 110925. doi: 10.1016/j.engstruct.2020.110925.

Doebling, S.W., Farrar, C., Prime, M. and Shevitz, D.W. (1996) Damage Identification and Health Monitoring of Structural and Mechanical Systems From Changes in Their Vibration Characteristics: A Literature Review. Technical Report No. LA-13070-MS, 30. Available at: https://doi.org/10.2172/249299.

Dupont, E., Doucet, A. and Teh, Y.W. (2019) 'Augmented Neural ODEs', arXiv. Available at: https://arxiv.org/abs/1904.01681v3.

Dyke, S.J., Bernal, D., Beck, J. and Ventura, C. (2003) 'Experimental phase II of the structural health monitoring benchmark problem', in Proceedings of the 16th ASCE Engineering Conference, pp. 1–7.

Ebrahimian, H., Astroza, R., Conte, J.P. and Papadimitriou, C. (2018) 'Bayesian optimal estimation for output-only nonlinear system and damage identification of civil structures', Structural Control and Health Monitoring, 25(4), p. e2128. doi: 10.1002/stc.2128.

Edna, T., Li, Y., Sam, N. and Liu, Y. (2018) 'A Comparative Study of Fine-Tuning Deep Learning Models for Plant Disease Identification', Computers and Electronics in Agriculture, 161. https://doi.org/10.1016/j.compag.2018.03.032.

Fan, W. and Qiao, P.Z. (2011) 'Vibration-Based Damage Identification Methods: A Review and Comparative Study', Structural Health Monitoring, 10(1), pp. 83–111. https://doi.org/10.1177/1475921710365419.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V. (2016) 'Domain-adversarial training of neural networks', The Journal of Machine Learning Research, 17(1), pp. 2096–2030.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V. (2017) 'Domain-Adversarial Training of Neural Networks', in Csurka, G. (ed.) Domain Adaptation in Computer Vision Applications. Cham: Springer International Publishing, pp. 189–209. https://doi.org/10.1007/978-3-319-58347-1_10.

Gao, Y. and Mosalam, K.M. (2018) 'Deep Transfer Learning for Image-Based Structural Damage Recognition', Computer-Aided Civil and Infrastructure Engineering, 33(9), pp. 748–768. https://doi.org/10.1111/mice.12363.

Gardner, P., Bull, L.A., Gosliga, J., Dervilis, N. and Worden, K. (2021) 'Foundations of population-based SHM, Part III: Heterogeneous populations – Mapping and transfer', Mechanical Systems and Signal Processing, 149, p. 107142.

Gardner, P., Liu, X. and Worden, K. (2020) 'On the application of domain adaptation in structural health monitoring', Mechanical Systems and Signal Processing, 138, p. 106550.

GB 50009-2001 (2002) Load Code for the Design of Building Structures.

George Em Karniadakis 1,2 ✉ (n.d.) Physics-informed machine learning.

Ghahari, S.F., Abazarsa, F. and Taciroglu, E. (2017) 'Blind modal identification of non-classically damped structures under non-stationary excitations: ID of non-Classically Damped Systems Under non-Stationary Excitations', Structural Control and Health Monitoring, 24(6), p. e1925. https://doi.org/10.1002/stc.1925.

Ghorbani, E., Buyukozturk, O. and Cha, Y.-J. (2020) 'Hybrid output-only structural system identification using random decrement and Kalman filter', Mechanical Systems and Signal Processing, 144, p. 106977. https://doi.org/10.1016/j.ymssp.2020.106977.

Goi, Y. and Kim, C.-W. (2017) 'Damage detection of a truss bridge utilizing a damage indicator from multivariate autoregressive model', Journal of Civil Structural Health Monitoring, 7(2), pp. 153–162. https://doi.org/10.1007/s13349-017-0222-y.

Gu, X., Yu, Y., Li, J. and Li, Y. (2017) 'Semi-active control of magnetorheological elastomer base isolation system utilising learning-based inverse model', Journal of Sound and Vibration, 406, pp. 346–362. https://doi.org/10.1016/j.jsv.2017.06.023.

Gul, M. and Catbas, F.N. (2008) 'Ambient vibration data analysis for structural identification and global condition assessment', Journal of Engineering Mechanics, 134(8), pp. 650–662. https://doi.org/10.1061/(ASCE)0733-9399(2008)134:8(650).

Hairer, E. and Wanner, G. (1996) 'Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems', Springer Verlag Series in Comput. Math., 14. https://doi.org/10.1007/978-3-662-09947-6.

Han, Q., Pan, Y., Yang, D. and Xu, Y. (2022) 'CNN-Based Bolt Loosening Identification Framework for Prefabricated Large-Span Spatial Structures', Journal of Civil Structural Health Monitoring, 12(3), pp. 517–536. https://doi.org/10.1007/s13349-022-00561-9.

Hao, J., Zhu, X., Yu, Y., Zhang, C. and Li, J. (2022) 'Damage localization and quantification of a truss bridge using PCA and convolutional neural network', Smart Structures and Systems, 30(6), pp. 678–686.

Hong, X., Zheng, Q., Liu, L., Chen, P., Ma, K., Gao, Z. and Zheng, Y. (2021) 'Dynamic Joint Domain Adaptation Network for Motor Imagery Classification', IEEE Transactions on Neural Systems and Rehabilitation Engineering, 29, pp. 556–565.

Hou, R. and Xia, Y. (2021) 'Review on the new development of vibration-based damage identification for civil engineering structures: 2010–2019', Journal of Sound and Vibration, 491, p. 115741. https://doi.org/10/ghr3vh.

Hou, R., Xia, Y., Bao, Y. and Zhou, X. (2018) 'Selection of regularization parameter for l1-regularized damage detection', Journal of Sound and Vibration, 423, pp. 141–160.

Hou, R., Xia, Y., Zhou, X. and Huang, Y. (2019) 'Sparse Bayesian learning for structural damage detection using expectation–maximization technique', Structural Control and Health Monitoring, 26(5), p. e2343. https://doi.org/10.1002/stc.2343.

Hua, X.G., Xu, K., Wang, Y.W., Wen, Q. and Chen, Z.Q. (2020) 'Wind-induced responses and dynamic characteristics of a super-tall building under a typhoon event', Smart Structures and Systems, 25(1), pp. 81–96.

Huang, Y., Beck, J.L. and Li, H. (2017) 'Bayesian system identification based on hierarchical sparse Bayesian learning and Gibbs sampling with application to structural damage assessment', Computer Methods in Applied Mechanics and Engineering, 318, pp. 382–411. https://doi.org/10.1016/j.cma.2017.01.030.

Jiao, J., Lin, J., Zhao, M. and Liang, K. (2020) 'Double-level adversarial domain adaptation network for intelligent fault diagnosis', Knowledge-Based Systems, 205, p. 106236.

Jin, Z., Teng, S., Zhang, J., Chen, G. and Cui, F. (2022) 'Structural Damage Recognition Based on Filtered Feature Selection and Convolutional Neural Network', International Journal of Structural Stability and Dynamics, 22(12), p. 2250134.

Johnson, E.A. (2003) 'Phase I IASC-ASCE structural health monitoring benchmark problem using simulated data', Journal of Engineering Mechanics ASCE, 130(1), pp. 3–15.

Kandel, I. and Castelli, M. (2020) 'Transfer Learning with Convolutional Neural Networks for Diabetic Retinopathy Image Classification. A Review', Applied Sciences, 10(6), p. 2021. https://doi.org/10.3390/app10062021.

Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S.F. and Yang, L. (2021) 'Physics-informed machine learning', Nature Review Physics, 3, pp. 422–440.

Karpatne, A., Atluri, G., Faghmous, J., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N. and Kumar, V. (2017) 'Theory-Guided Data Science: A New Paradigm for Scientific

Discovery from Data', IEEE Transactions on Knowledge and Data Engineering, 29(10), pp. 2318–2331.

Kerschen, G., Worden, K., Vakakis, A.F. and Golinval, J.-C. (2006) 'Past, present and future of nonlinear system identification in structural dynamics', Mechanical Systems and Signal Processing, 20(3), pp. 505–592. https://doi.org/10.1016/j.ymssp.2005.04.008.

Kingma, D. and Ba, J. (2014) 'Adam: A Method for Stochastic Optimization', arXiv. Available at: https://arxiv.org/abs/1412.6980.

Kontolati, K., Goswami, S., Em Karniadakis, G. and Shields, M.D. (2024) 'Learning nonlinear operators in latent spaces for real-time predictions of complex dynamics in physical systems', Nature Communications, 15(1), p. 5101. https://doi.org/10.1038/s41467-024-49411-w.

Kostic, B. and Gül, M. (2017) 'Vibration-based damage detection of bridges under varying temperature effects using time-series analysis and artificial neural network', Journal of Bridge Engineering, 22(10), p. 04017065.

Krikelis, K., Pei, J.-S., Van Berkel, K. and Schoukens, M. (2024) 'Identification of structured nonlinear state–space models for hysteretic systems using neural network hysteresis operators', Measurement, 224, p. 113966. https://doi.org/10.1016/j.measurement.2023.113966.

Krishnan Nair, K. and Kiremidjian, A.S. (2007) 'Time Series Based Structural Damage Detection Algorithm Using Gaussian Mixtures Modeling', Journal of Dynamic Systems, Measurement, and Control, 129(3), pp. 285–293. https://doi.org/10.1115/1.2718241.

Lai, Z. and Nagarajaiah, S. (2019) 'Sparse structural system identification method for nonlinear dynamic systems with hysteresis/inelastic behavior', Mechanical Systems and Signal Processing, 117, pp. 813–842. https://doi.org/10/gnnzpr.

Lei, Y., Li, J. and Hao, H. (2024) 'Physics-guided deep learning based on modal sensitivity for structural damage identification with unseen damage patterns', Engineering Structures, 316, p. 118510.

Li, H., Wang, T., Yang, J.P. and Wang, G. (2023) 'Deep Learning Models for Time-History Prediction of Vehicle-Induced Bridge Responses: A Comparative Study', International Journal of Structural Stability and Dynamics, 23(1), p. 2350004.

Li, H.-W., Ni, Y.-Q., Wang, Y.-W., Chen, Z.-W., Rui, E.-Z. and Xu, Z.-D. (2024) 'Modeling of Forced-Vibration Systems Using Continuous-Time State-Space Neural Network', Engineering Structures, 302, p. 117329. https://doi.org/10.1016/j.engstruct.2023.117329.

Li, S., Cao, Y., Gdoutos, E.E., Tao, M., Faisal Alkayem, N., Avci, O. and Cao, M. (2024) 'Intelligent framework for unsupervised damage detection in bridges using deep convolutional autoencoder with wavelet transmissibility pattern spectra', Mechanical Systems and Signal Processing, 220, p. 111653. https://doi.org/10.1016/j.ymssp.2024.111653.

Li, S., Liu, C.H., Xie, B., Su, L., Ding, Z. and Huang, G. (2019) 'Joint Adversarial Domain Adaptation', in Proceedings of the 27th ACM International Conference on Multimedia, pp. 729–737.

Li, Y., Li, J., Tian, T. and Li, W. (2013) 'A Highly Adjustable Magnetorheological Elastomer Base Isolator for Applications of Real-Time Adaptive Control', Smart Materials and Structures, 22(9), p. 095020. https://doi.org/10.1088/0964-1726/22/9/095020.

Li, Z., Zheng, H., Kovachki, N., Jin, D., Chen, H., Liu, B., Azizzadenesheli, K. and Anandkumar, A. (2024) 'Physics-Informed Neural Operator for Learning Partial Differential Equations', ACM / IMS Journal of Data Science, 1(3), pp. 1–27. https://doi.org/10.1145/3648506.

Li, Z.-D., He, W.-Y., Ren, W.-X., Li, Y.-L., Li, Y.-F. and Cheng, H.-C. (2023) 'Damage detection of bridges subjected to moving load based on domain-adversarial neural network considering measurement and model error', Engineering Structures, 293, p. 116601.

Limongelli, M.P. (2010) 'Frequency response function interpolation for damage detection under changing environment', Mechanical Systems and Signal Processing, 24(8), pp. 2898–2913. https://doi.org/10.1016/j.ymssp.2010.03.004.

Limongelli, M.P., Siegert, D., Merliot, E., Waeytens, J., Bourquin, F., Vidal, R., Le Corvec, V., Gueguen, I. and Cottineau, L.M. (2016) 'Damage detection in a post tensioned concrete beam – Experimental investigation', Engineering Structures, 128, pp. 15–25. https://doi.org/10.1016/j.engstruct.2016.09.017.

Lin, R. and Xia, Y. (2003) 'A new eigensolution of structures via dynamic condensation', Journal of Sound and Vibration, 266, pp. 93–106.

Lin, Y., Nie, Z. and Ma, H. (2017) 'Structural Damage Detection with Automatic Feature-Extraction through Deep Learning', Computer-Aided Civil and Infrastructure Engineering, 32(12), pp. 1025–1046. https://doi.org/10.1111/mice.12313.

Lin, Y., Nie, Z. and Ma, H. (2022) 'Dynamics-based cross-domain structural damage detection through deep transfer learning', Computer-Aided Civil and Infrastructure Engineering, 37(1), pp. 24–54. https://doi.org/10.1111/mice.12692.

Lin, Y.Z., Nie, Z.H. and Ma, H.W. (2017) 'Structural damage detection with automatic feature-extraction through deep learning', Computer-Aided Civil and Infrastructure Engineering, 32(12), pp. 1025–1046.

Liu, D., Tang, Z., Bao, Y. and Li, H. (2021) 'Machine-learning-based methods for output-only structural modal identification', Structural Control and Health Monitoring, 28(12), p. e2843.

Liu, L.J., Zhang, X. and Lei, Y. (2023) 'Data-Driven Identification of Structural Damage under Unknown Seismic Excitations Using the Energy Integrals of Strain Signals Transformed from Transmissibility Functions', Journal of Sound and Vibration, 546, p. 117490.

Liu, W., Lai, Z., Bacsa, K. and Chatzi, E. (2024) 'Neural Extended Kalman Filters for Learning and Predicting Dynamics of Structural Systems', Structural Health Monitoring, 23(2), pp. 1037–1052.

Liu, W., Sicard, B. and Gadsden, S.A. (2024) 'Physics-Informed Machine Learning: A Comprehensive Review on Applications in Anomaly Detection and Condition Monitoring', Expert Systems with Applications, 255, p. 124678.

Liu, Y., Li, L. and Chang, Z. (2023) 'Efficient Bayesian Model Updating for Dynamic Systems', Reliability Engineering & System Safety, 236, p. 109294.

Ljung, L. (2010) 'Perspectives on System Identification', Annual Reviews in Control, 34(1), pp. 1–12.

Lu, L., Jin, P. and Karniadakis, G.E. (2021) 'DeepONet: Learning Nonlinear Operators for Identifying Differential Equations Based on the Universal Approximation Theorem of Operators', Nature Machine Intelligence, 3(3), pp. 218–229. https://doi.org/10.1038/s42256-021-00302-5.

Lu, L., Pestourie, R., Johnson, S.G. and Romano, G. (2022) 'Multifidelity Deep Neural Operators for Efficient Learning of Partial Differential Equations with Application to Fast Inverse Design of Nanoscale Heat Transport', Physical Review Research, 4(2), p. 023210. https://doi.org/10.1103/PhysRevResearch.4.023210.

Lu, Y., Tang, L., Chen, C., Zhou, L., Liu, Z., Liu, Y., Jiang, Z. and Yang, B. (2023) 'Reconstruction of Structural Long-Term Acceleration Response Based on BiLSTM Networks', Engineering Structures, 285, p. 116000. https://doi.org/10.1016/j.engstruct.2023.116000.

Lu, Y., Tang, L., Chen, C., Zhou, L., Liu, Z., Liu, Y., Jiang, Z. and Yang, B. (2024) 'Unsupervised Structural Damage Assessment Method Based on Response Correlations', Engineering Structures, 302, p. 117413. https://doi.org/10/gtw37z.

Manoach, E., Warminski, J., Kloda, L. and Teter, A. (2017) 'Numerical and experimental studies on vibration based methods for detection of damage in composite beams', Composite Structures, 170, pp. 26–39.

Marin, L., Döhler, M., Bernal, D. and Mevel, L. (2015) 'Robust Statistical Damage Localization with Stochastic Load Vectors', Structural Control and Health Monitoring, 22(3), pp. 557–573. https://doi.org/10.1002/stc.1686.

Mei, L.-F., Yan, W.-J., Yuen, K.-V. and Beer, M. (2025) 'Streaming Variational Inference-Empowered Bayesian Nonparametric Clustering for Online Structural Damage Detection with Transmissibility Function', Mechanical Systems and Signal Processing, 222, p. 111767.

Meng, X., Li, Z., Zhang, D. and Karniadakis, G.E. (2020) 'PPINN: Parareal Physics-Informed Neural Network for Time-Dependent PDEs', Computer Methods in Applied Mechanics and Engineering, 370, p. 113250.

Meruane, V. and Heylen, W. (2012) 'Structural Damage Assessment under Varying Temperature Conditions', Structural Health Monitoring, 11, pp. 345–357.

Mishra, P. and Passos, D. (2021) 'Realizing Transfer Learning for Updating Deep Learning Models of Spectral Data to Be Used in a New Scenario', Chemometrics and Intelligent Laboratory Systems, p. 104283. https://doi.org/10.1016/j.chemolab.2021.104283.

Mishra, P. and Passos, D. (2021) 'Realizing Transfer Learning for Updating Deep Learning Models of Spectral Data to Be Used in a New Scenario', Chemometrics and Intelligent Laboratory Systems, p. 104283. https://doi.org/10.1016/j.chemolab.2021.104283.

Mottershead, J.E. and Friswell, M.I. (1993) 'Model Updating in Structural Dynamics: A Survey', Journal of Sound and Vibration, 167(2), pp. 347–375. https://doi.org/10.1006/jsvi.1993.1340.

Moughty, J.J. and Casas, J.R. (2017) 'A State of the Art Review of Modal-Based Damage Detection in Bridges: Development, Challenges, and Solutions', Applied Sciences, 7(5), p. 510. https://doi.org/10.3390/app7050510.

Nair, V. and Hinton, G.E. (2010) 'Rectified Linear Units Improve Restricted Boltzmann Machines', in International Conference on Machine Learning, pp. 807–814.

Nayeri, R.D., Masri, S.F., Ghanem, R.G. and Nigbor, R.L. (2008) 'A Novel Approach for the Structural Identification and Monitoring of a Full-Scale 17-Story Building Based on Ambient Vibration Measurements', Smart Materials and Structures, 17(2), p. 025006.

Ni, Y.Q., Xia, Y., Liao, W.Y. and Ko, J.M. (2009) 'Technology Innovation in Developing the Structural Health Monitoring System for Guangzhou New TV Tower', Structural Control and Health Monitoring, 16(1), pp. 73–98.

Ni, Y.Q., Xia, Y., Lin, W., Chen, W.H. and Ko, J.M. (2012) 'SHM Benchmark for High-Rise Structures: A Reduced-Order Finite Element Model and Field Measurement Data', Smart Structures and Systems, 10(4–5), pp. 411–426.

Nyquist, H. (1928) 'Certain topics in telegraph transmission theory', Transactions of the American Institute of Electrical Engineers, 47(2), pp. 617–644. https://doi.org/10.1109/T-AIEE.1928.5055024.

Paduart, J., Lauwers, L., Swevers, J., Smolders, K., Schoukens, J. and Pintelon, R. (2010) 'Identification of Nonlinear Systems Using Polynomial Nonlinear State Space Models', Automatica, 46(4), pp. 647–656. https://doi.org/10.1016/j.automatica.2010.01.001.

Park, S.-K. and Noh, H.Y. (2017) 'Updating Structural Parameters with Spatially Incomplete Measurements Using Subspace System Identification', Journal of Engineering Mechanics, 143(7), p. 04017040. https://doi.org/10.1061/(ASCE)EM.1943-7889.0001226.

Pathirage, C.S.N., Li, J., Li, L., Hao, H., Liu, W. and Ni, P. (2018) 'Structural Damage Identification Based on Autoencoder Neural Networks and Deep Learning', Engineering Structures, 172, pp. 13–28.

Pedram, M., Esfandiari, A. and Khedmati, M.R. (2017) 'Damage Detection by a FE Model Updating Method Using Power Spectral Density: Numerical and Experimental Investigation', Journal of Sound and Vibration, 397, pp. 51–76. https://doi.org/10.1016/j.jsv.2017.02.052.

Peeters, B. and De Roeck, G. (2001) 'One-Year Monitoring of the Z24-Bridge: Environmental Effects Versus Damage Events', Earthquake Engineering & Structural Dynamics, 30(2), pp. 149–171. https://doi.org/10.1002/1096-9845(200102)30:2<149::AID-EQE1>3.0.CO;2-Z.

Peeters, M., Kerschen, G. and Golinval, J.C. (2011) 'Dynamic Testing of Nonlinear Vibrating Structures Using Nonlinear Normal Modes', Journal of Sound and Vibration, 330(3), pp. 486–509. https://doi.org/10.1016/j.jsv.2010.08.028.

Peng, Z., Li, J. and Hao, H. (2022) 'Structural Damage Detection via Phase Space Based Manifold Learning under Changing Environmental and Operational Conditions', Engineering Structures, 263, p. 114420.

Raissi, M. and Karniadakis, G.E. (2018) 'Hidden Physics Models: Machine Learning of Nonlinear Partial Differential Equations', Journal of Computational Physics, 357, pp. 125–141.

Raissi, M., Perdikaris, P. and Karniadakis, G.E. (2019) 'Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations', Journal of Computational Physics, 378, pp. 686–707. https://doi.org/10/gfzbvx.

Ren, W.-X. and De Roeck, G. (2002) 'Structural Damage Identification Using Modal Data. I: Simulation Verification', Journal of Structural Engineering, 128(1), pp. 87–95. https://doi.org/10.1061/(ASCE)0733-9445(2002)128:1(87).

Reynders, E. and De Roeck, G. (2008) 'Reference-Based Combined Deterministic–Stochastic Subspace Identification for Experimental and Operational Modal Analysis', Mechanical Systems and Signal Processing, 22(3), pp. 617–637. https://doi.org/10.1016/j.ymssp.2007.09.004.

Reynders, E., Teughels, A. and De Roeck, G. (2010) 'Finite Element Model Updating and Structural Damage Identification Using OMAX Data', Mechanical Systems and Signal Processing, 24(5), pp. 1306–1323. https://doi.org/10.1016/j.ymssp.2010.03.014.

Rytter, A., Brincker, R. and Hansen, L.P. (1993) 'Vibration Based Inspection of Civil Engineering Structures', [Technical Report], Aalborg University, Denmark.

Sadeghi Eshkevari, S., Takáč, M., Pakzad, S.N. and Jahani, M. (2021) 'DynNet: Physics-Based Neural Architecture Design for Nonlinear Structural Response Modeling and Prediction', Engineering Structures, 229, p. 111582. https://doi.org/10.1016/j.engstruct.2020.111582.

Sadeghzadeh, S. and Khatibi, M.M. (2017) 'Modal Identification of Single Layer Graphene Nano Sheets from Ambient Responses Using Frequency Domain Decomposition', European Journal of Mechanics - A/Solids, 65, pp. 70–78. https://doi.org/10.1016/j.euromechsol.2017.03.009.

Saito, K., Watanabe, Y., Ushiku, Y. and Harada, T. (2018) 'Maximum Classifier Discrepancy for Unsupervised Domain Adaptation', in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3723–3732.

Shiki, S.B., Da Silva, S. and Todd, M.D. (2017) 'On the Application of Discrete-Time Volterra Series for the Damage Detection Problem in Initially Nonlinear Systems', Structural Health Monitoring, 16(1), pp. 62–78. https://doi.org/10.1177/1475921716662142.

Shukla, K., Jagtap, A.D. and Karniadakis, G.E. (2021) 'Parallel Physics-Informed Neural Networks via Domain Decomposition', Journal of Computational Physics, 447, p. 110683.

Silva, M., Santos, A., Figueiredo, E., Santos, R., Sales, C. and Costa, J.C.W.A. (2016) 'A Novel Unsupervised Approach Based on a Genetic Algorithm for Structural Damage Detection in Bridges', Engineering Applications of Artificial Intelligence, 52, pp. 168–180. https://doi.org/10.1016/j.engappai.2016.03.002.

Sony, S., Dunphy, K., Sadhu, A. and Capretz, M. (2021) 'A Systematic Review of Convolutional Neural Network-Based Structural Condition Assessment Techniques', Engineering Structures, 226, p. 111347. https://doi.org/10.1016/j.engstruct.2020.111347.

Sun, L., Shang, Z., Xia, Y., Bhowmick, S. and Nagarajaiah, S. (2020) 'Review of Bridge Structural Health Monitoring Aided by Big Data and Artificial Intelligence: From Condition Assessment to Damage Detection', Journal of Structural Engineering ASCE, 146(5), p. 0002535.

Thai, H.-T. (2022) 'Machine Learning for Structural Engineering: A State-of-the-Art Review', Structures, 38, pp. 448–491. https://doi.org/10.1016/j.istruc.2022.02.003.

Tzeng, E., Hoffman, J., Saenko, K. and Darrell, T. (2017) 'Adversarial Discriminative Domain Adaptation', in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7167–7176.

Wang, M. and Deng, W. (2018) 'Deep Visual Domain Adaptation: A Survey', Neurocomputing, 312, pp. 135–153. https://doi.org/10.1016/j.neucom.2018.05.083.

Wang, R.H., Chencho, An, S.J., Li, J., Li, L., Hao, H. and Liu, W.Q. (2020) 'Deep Residual Network Framework for Structural Health Monitoring', Structural Health Monitoring, 20(4), pp. 1443–1461.

Wang, X. and Xia, Y. (2022) 'Knowledge Transfer for Structural Damage Detection through Re-Weighted Adversarial Domain Adaptation', Mechanical Systems and Signal Processing, 172, p. 108991. https://doi.org/10.1016/j.ymssp.2022.108991.

Wei Fan and Qiao, P.Z. (2011) 'Vibration-Based Damage Identification Methods: A Review and Comparative Study', Structural Health Monitoring, 10(1), pp. 83–111. https://doi.org/10.1177/1475921710365419.

Worden, K., Farrar, C.R., Haywood, J. and Todd, M. (2008) 'A Review of Nonlinear Dynamics Applications to Structural Health Monitoring', Structural Control and Health Monitoring, 15(4), pp. 540–567. https://doi.org/10.1002/stc.215.

Xia, Y., Wang, X. (2022) 'Knowledge Transfer for Structural Damage Detection through Re-Weighted Adversarial Domain Adaptation', Mechanical Systems and Signal Processing, 172, p. 108991. (If duplicated, merge with Wang, X. and Xia, Y. (2022) entry.)

Xie, X. Yin, W. Yan, Y. Liu, Y. Zhou and L. Li (2023) 'Structural Damage Detection Based on Improved Sensitivity Function of Modal Flexibility and Iterative Reweighted l1 Regularization', International Journal of Structural Stability and Dynamics. https://doi.org/10.1142/S0219455424501116.

Xu, S. and Noh, H.Y. (2021) 'PhyMDAN: Physics-Informed Knowledge Transfer between Buildings for Seismic Damage Diagnosis through Adversarial Learning', Mechanical Systems and Signal Processing, 151, p. 107374.

Yaghoubi, V., Rahrovani, S., Nahvi, H. and Marelli, S. (2018) 'Reduced Order Surrogate Modeling Technique for Linear Dynamic Systems', Mechanical Systems and Signal Processing, 111, pp. 172–193. https://doi.org/10.1016/j.ymssp.2018.02.020.

Yan, W.J., Zhao, M.Y., Sun, Q. and Ren, W.X. (2019) 'Transmissibility-Based System Identification for Structural Health Monitoring: Fundamentals, Approaches and Applications', Mechanical Systems and Signal Processing, 117, pp. 453–482.

Yang, J., Du, H., Li, W., Li, Y., Li, J., Sun, S. and Deng, H.X. (2013) 'Experimental Study and Modeling of a Novel Magnetorheological Elastomer Isolator', Smart Materials and Structures, 22(11), p. 117001. https://doi.org/10.1088/0964-1726/22/11/117001.

Yang, Y. and Nagarajaiah, S. (2014) 'Blind Identification of Damage in Time-Varying Systems Using Independent Component Analysis with Wavelet Transform', Mechanical Systems and Signal Processing, 47(1–2), pp. 3–20. https://doi.org/10.1016/j.ymssp.2012.08.029.

Yano, M.O., Figueiredo, E., Da Silva, S. and Cury, A. (2023) 'Foundations and Applicability of Transfer Learning for Structural Health Monitoring of Bridges', Mechanical Systems and Signal Processing, 204, p. 110766.

Ye, X.W., Jin, T. and Yun, C. (2019) 'A Review on Deep Learning-Based Structural Health Monitoring of Civil Infrastructures', Smart Structures and Systems, 24(5), pp. 567–585. https://doi.org/10.12989/sss.2019.24.5.567.

Yeung, W.T. and Smith, J.W. (2005) 'Damage Detection in Bridges Using Neural Networks for Pattern Recognition of Vibration Signatures', Engineering Structures, 27(5), pp. 685–698. https://doi.org/10.1016/j.engstruct.2004.12.006.

Yin, X., Huang, Z. and Liu, Y. (2023) 'Bridge Damage Identification under the Moving Vehicle Loads Based on the Method of Physics-Guided Deep Neural Networks', Mechanical Systems and Signal Processing, 190, p. 110123. https://doi.org/10.1016/j.ymssp.2023.110123.

Yu, Y., Hoshyar, A.N., Samali, B., Zhang, G., Rashidi, M. and Mohammadi, M. (2023b) 'Corrosion and Coating Defect Assessment of Coal Handling and Preparation Plants (CHPP) Using an

Ensemble of Deep Convolutional Neural Networks and Decision-Level Data Fusion', Neural Computing and Applications, 35(25), pp. 18697–18718. https://doi.org/10.1007/s00521-023-08699-3.

Yu, Y., Li, J., Li, J., Xia, Y., Ding, Z. and Samali, B. (2023) 'Automated Damage Diagnosis of Concrete Jack Arch Beam Using Optimized Deep Stacked Autoencoders and Multi-Sensor Fusion', Developments in the Built Environment, 14, p. 100128. https://doi.org/10.1016/j.dibe.2023.100128.

Yu, Y., Samali, B., Rashidi, M., Mohammadi, M., Nguyen, T.N. and Zhang, G. (2022) 'Vision-Based Concrete Crack Detection Using a Hybrid Framework Considering Noise Effect', Journal of Building Engineering, 61, p. 105246. https://doi.org/10.1016/j.jobe.2022.105246.

Yu, Y., Wang, C.Y., Gu, X.Y. and Li, J.C. (2019) 'A Novel Deep Learning-Based Method for Damage Identification of Smart Building Structures', Structural Health Monitoring, 18(1), pp. 143–163.

Yu, Y., Yao, H. and Liu, Y. (2020) 'Structural Dynamics Simulation Using a Novel Physics-Guided Machine Learning Method', Engineering Applications of Artificial Intelligence, 96, p. 103947. https://doi.org/10.1016/j.engappai.2020.103947.

Zhan, C., Li, D. and Li, H. (2014) 'A Local Damage Detection Approach Based on Restoring Force Method', Journal of Sound and Vibration, 333(20), pp. 4942–4959. https://doi.org/10.1016/j.jsv.2014.04.043.

Zhang, K., Zhang, Y. and Cheng, H.D. (2020) 'Self-Supervised Structure Learning for Crack Detection Based on Cycle-Consistent Generative Adversarial Networks', Journal of Computing in Civil Engineering, 34(3), p. 04020004. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000883.

Zhang, R., Liu, Y. and Sun, H. (2020) 'Physics-Guided Convolutional Neural Network (PhyCNN) for Data-Driven Seismic Response Modeling', Engineering Structures, 215, p. 110704. https://doi.org/10.1016/j.engstruct.2020.110704.

Zhang, Z., Sun, C. and Guo, B. (2022) 'Transfer-Learning Guided Bayesian Model Updating for Damage Identification Considering Modelling Uncertainty', Mechanical Systems and Signal Processing, 166, p. 108426.

Zhao, Z.B., Zhang, Q.Y., Yu, X.L., Sun, C., Wang, S.B., Yan, R.Q. and Chen, X.F. (2021) 'Applications of Unsupervised Deep Transfer Learning to Intelligent Fault Diagnosis: A Survey and Comparative Study', IEEE Transactions on Instrumentation and Measurements, 70, p. 3525828.

Zhou, X., Gu, M., Zhu, L., Huang, P. and Pan, H. (2009) 'Wind Tunnel Force Balance Test and Wind-Induced Responses of the Guangzhou New TV Tower Structure. II: Analysis of Wind-Induced Responses', China Civil Engineering Journal, 42(7), pp. 14–20.