**School of Civil and Mechanical Engineering**

# Efficient Supervised Machine Learning Techniques for Structural Health Monitoring

**Chencho**

**0000000166679384**

**This thesis is presented for the Degree of**
**Doctor of Philosophy**
**of**
**Curtin University**

**June 2022**

## Declaration

To the best of my knowledge and belief, this thesis contains no material previously published by any other person except where due acknowledgment has been made.

This thesis contains no material accepted for the award of any other degree or diploma in any university.

The research presented does not contain any ethical issues or implications and is carried out as per the university's code of conduct of research.

02 June 2022

## Abstract

Structural health monitoring (SHM) techniques have been widely developed and applied for the condition monitoring and safety evaluation of aerospace, civil and mechanical infrastructure. Recently, vibration-based damage identification and quantification methods have been explored to assess and evaluate structural health conditions. Many ways have been developed based on vibration data using various techniques to understand the condition of ageing structures. For civil engineering structures, SHM systems take the measurements by sensing technology and evaluation is carried out by data interpretation algorithms. Numerous studies have used modal information, such as mode shapes and natural frequencies for vibration-based damage identification. Accurate extraction of modal information, i.e., mode shapes, requires several sensors to measure responses at many locations on the structures. Vibration tests and measurements are time-consuming and may not be practical for large-scale structures. Time domain responses measured with few sensors can be used for damage identification and quantification. Acceleration responses are sensitive to structure condition changes and can easily be measured.

Furthermore, impulse response functions (IRFs), represented as time domain responses under the input of an impulse excitation, can be obtained from the measured time domain responses. IRF is an ideal inherent vibration property for structural damage identification and quantification. It has the advantage over directly using the measured responses because it is an inherent system property.

Supervised machine learning and deep learning algorithms can generate knowledge of the structure with the data collected from the physical sensors and numerical simulations. Both conventional machine learning and deep learning models have been extensively used for damage identification and quantification. Ensemble or committee-based methods improve machine learning performance using the results from multiple models. It can achieve better prediction results than a single model with reduced variance and bias. Deep learning models have been used for the SHM of civil engineering structures when the sensor data size is big. Many of such studies use data generated from numerical simulations, mainly finite element modelling (FEM), for training the proposed machine or deep learning algorithm considering uncertainties in the measurements. The models are further validated and tested with vibration

measurement data from experimental studies using structures fabricated in a laboratory or real in-field structures. The model can also be trained and validated with the field measurement data when sufficient data size or data augmentation techniques are adopted. Furthermore, dimensional reduction techniques are often used to reduce the computational cost of machine learning models before feeding the measured data to the network models.

In this study, acceleration responses measured from a small number of sensors on the structures and machine learning algorithms, the Random Forest (RF) together with principal component analysis (PCA) for dimensionality reduction are explored for damage identification and quantification. RF is a tree-based ensemble machine learning algorithm and is developed as a regressor to predict multiple output variables in this study. Damage severity is defined in terms of reduction in elemental stiffness parameters. Numerical studies are carried out on a simply supported beam. The acceleration responses are measured with different levels of damage. The input variable to RF is the processed acceleration responses, and the target variable is the corresponding stiffness reduction. This study is further verified by an experimental study using a steel frame structure with 70 elements. The performance of the proposed method is investigated by considering four scenarios. The first scenario assumes acceleration responses without measurement noise and uncertainty in stiffness parameters. For the second scenario, noisy acceleration responses are used for identification. The third scenario considers uncertainties in stiffness parameters but without noise effect in vibration measurements. The last scenario is the most challenging, considering both the noisy measures and uncertainties in stiffness parameters. The proposed approach provides good damage identification and quantification results with less computational time.

This study is extended further by using IRFs extracted from acceleration responses as the input to machine learning algorithms. Numerical studies are conducted on a simply supported beam, and experimental studies are performed on a steel frame structure. The IRFs are obtained for identification, and another extended tree-based ensemble machine learning algorithm, the extremely randomised tree (ERT), is adopted. The performance of RF and ERT are evaluated for the four scenarios mentioned above. Moving averaging (MA) and PCA are performed to reduce random input variations

and dimensionality reduction. The damage identification using RF and IRFs has demonstrated a slight improvement in the performance compared to using acceleration response directly. The results also show that using ERT outperforms RF.

A deep learning model based on Long Short-Term Memory (LSTM) autoencoder is used to improve further the damage identification and quantification for the same problem. Deep learning models have various ways to tune and obtain better results. LSTM networks have been used widely for sequence prediction problems, including sequences that may differ in length. The same IRF datasets generated from a simply supported beam for ERT are used for damage identification based on the LSTM autoencoder. For the proposed LSTM autoencoder model, a noise layer is included on the input layer, which helps generalise different datasets, especially the input with the noise. The datasets used in the first and third scenarios are combined and used to train, validate, and test the proposed LSTM autoencoder model. Further, the model is also tested with datasets from Scenarios 2 and 4. The proposed model has produced good damage identification and quantification results and is more robust to noise than RF and ERT. Furthermore, the performance is measured by reducing the sensor numbers from five to three. The results are close to those obtained using five sensors.

For all the studies mentioned, the damage has been defined in terms of stiffness reduction in structural elements. Civil infrastructure such as bridges can be exposed to external conditions, such as wind, temperature, vehicle loads and other types of damages. Therefore, it may experience different levels of damage for each damage type. The proposed model mentioned earlier may not be suitable for damage identification and quantification when structures have different damage types and levels. Therefore, each type needs to quantify the damage level of each kind. The Z24 bridge in Switzerland has measured vibration responses during the short-term progressive damage tests under different types and damage levels for each damage type. A regressor model or sequence-to-sequence prediction discussed earlier may become too complex or not be suitable to predict damage type and quantify the damage for each damage type. A classification model using a One-dimensional convolutional network (1D-CNN) is proposed for this task. 1D-CNN has the advantage over other convolutional networks in terms of computational complexities since it is a relatively simple architecture. The time domain measurements of the Z24 bridge during the

progressive damage tests are used for damage identification with the proposed 1D-CNN. The performance demonstrated by experimental studies shows that the proposed method provides better results than the existing methods, using a small number of sensor measurements.

Following are some of the contributions of the thesis:

1. Development and application of traditional ensemble-based machine learning model, random forest with PCA, using time-domain responses from a small number of sensors for structural elemental damage identification and quantification. In this study, using random forest together with PCA can make structural element damage quantification and identification comparable to the existing deep learning model and significantly less training time. Further, it uses raw acceleration responses from the structure, which requires a small number of sensors, unlike modal information.
2. The structural elemental damage identification and quantification are improved by computing IRFs from acceleration response together and using it with moving averaging and PCA and then using it as the input to RF and ERT. The use of IRFs has an advantage over the acceleration response because it is an inherent system property. The use of moving averaging with a suitable window further helped to get an improved result.
3. Further, damage identification and quantification are improved by developing a robust LSTM Autoencoder using the same dataset. Unlike the previous two studies and existing deep learning models on structural elemental damage identification and quantification using both acceleration responses and modal information, the model is made more robust to noise in the measurement using a noise layer before the LSTM autoencoder network. In this approach, the model need not be trained, validated and tested separately for the damage scenarios considered, unlike in the first two proposed methods and existing studies. The proposed model provided good results even with a reduced number of sensors.
4. A novel 1D-CNN for damage classification is developed for a large-scale bridge structure, namely, the Z24 bridge, by using acceleration responses measured with a small number of sensors. The number of sensor measurements used in the analysis is less than that in a current study to demonstrate the

superiority of the proposed method. The data pre-processing technique is straightforward, involving autocorrelation, normalisation and splitting into smaller lengths.

**Acknowledgement**

I cannot begin to express my most profound appreciation and gratitude to my supervisors, Associate Professor Jun Li and Professor Hong Hao from the School of Civil and Mechanical Engineering and Professor Ling Li from the School of Electrical, Computing and Mathematic Sciences, for their supervision and assistance during my PhD study. Their immense knowledge, constructive criticism, patience, and continuous and unwavering support have made it possible to complete my PhD on time. I would not have imagined having better supervisors than them for my research.

I am deeply indebted to the Australian Government for giving Research Training Program (RTP) Scholarship to support my research. I must also thank my previous employer, the College of Science and Technology, under the Royal University of Bhutan (RUB), for their permission to leave the job and for allowing me to take up an RTP scholarship. I should thank KU LEUVEN for allowing me to use their dataset for a part of my research.

Further, I would also like to extend my sincere thanks to faculty members and researchers of the Centre for Infrastructural Monitoring and Protection (CIMP) for providing valuable feedback and comments on my research during the meetings. I must also thank the other staff from the School of Civil and Mechanical Engineering, Curtin University, for their prompt support in arranging the resources and services I needed for my research.

Lastly, I cannot forget to thank my family, relatives, and friends. I would like to thank my wife for being someone who has always been understanding and supported me and given more care to our two sons from the start of my PhD study until I submitted this thesis. Without her support, I would have struggled to manage time for my research and the family, ultimately affecting the completion of my studies on time.

## Acknowledgment (Country)

We acknowledge that Curtin University works across hundreds of traditional lands and custodial groups in Australia and with First Nations people around the globe. We wish to pay our most profound respects to their ancestors and members of their communities, past, present, and emerging leaders. Our passion and commitment to work with all Australians and peoples from across the world, including our First Nations peoples, are at the core of our work, reflective of our institutions' values and commitment to our role as leaders in the Reconciliation space in Australia.

## Copyright Statement

I have obtained permission from the copyright owners to use any third-party copyright material reproduced in the thesis and to use any of my published work (journal and conference articles) in which the copyright is held by another party (publisher, co-author).

## List of Publication

Journal Paper

1. Chencho, Li Jun, Hao Hong, Wang Ruhua, Li Ling. (2020). Development and application of random forest technique for element level structural damage quantification. *Structural Control and Health Monitoring*, 28(3), e2678. https://doi.org/10.1002/stc.2678
2. Chencho, Li Jun, Hao Hong, Wang Ruhua, Li Ling. (2022). Structural damage quantification using ensemble-based extremely randomised trees and impulse response functions. *Structural Control and Health Monitoring*, e3033. https://doi.org/10.1002/stc.3033
3. Chencho, Li Jun, Hao Hong, Li Ling. (Under review). Structural Damage Quantification using Long Short-Term Memory (LSTM) Auto-encoder and Impulse Response Functions.
4. Chencho, Li Jun, Hao Hong, Li Ling. (Under review). Structural Damage Classification of Large-scale Bridges using Convolutional Neural Networks and Time Domain Responses.

Conference Paper

1. Chencho, Li Jun, Hao Hong. "Long short-term memory auto-encoder based damage quantification using impulse response functions." In ACAM10: 10th Australian Congress on Applied Mechanics, pp. 346-354. Engineers Australia, 2021. https://search.informit.org/doi/10.3316/informit.323369548622379

Publications not included in the thesis.

1. Wang, R., Chencho, An, S., Li, J., Li, L., Hao, H., & Liu, W. Q. (2020). Deep residual network framework for structural health monitoring. *Structural Health Monitoring*, 20(4), 1443-1461. https://doi.org/10.1177%2F1475921720918378
2. Wang, R., Li, J., Chencho, An, S., Hao, H., Liu, W.-Q., & Li, L. (2021). Densely connected convolutional networks for vibration-based structural damage identification. *Engineering Structures*, 245, 112871.

https://doi.org/10.1177%2F1475921720918378

## Table of Contents

## List of Figures

## List of Tables

# CHAPTER 1

## 1 Introduction

This thesis presents the development and application of efficient ensemble-based machine learning and deep learning algorithms to identify and quantify damage in the civil engineering infrastructure using time domain responses measured from a small number of sensors.

### 1.1 Background

Structures have a finite life span. Processes such as corrosion, overloads, wears, erosion etc., would deteriorate structures with time until they cannot be used (Esfandiari et al., 2020; Gopalakrishnan et al., 2011). The deficiencies in these structures may lead to catastrophic consequences associated with substantial economic losses, casualties and disruptions in daily activities if proper monitoring is not carried out (Zhu et al., 2010). Structural health monitoring (SHM) techniques have been widely developed in the aerospace, civil and mechanical engineering communities for monitoring structural conditions (Farrar & Worden, 2007). Traditional visual inspection has been widely used to monitor the condition of structures in the early days. The US Federal Highway Administration (FHWA) report (Moore et al., 2001) stated the following limitations of visual inspection:

*i.* *Accessibility*

Structures may have no clear access for inspectors to conduct condition assessments.

*ii.* *Interpretability*

There are possibilities of obtaining inappropriate and inadequate condition assessment results since this could depend on inspectors' subject experiences; and

*iii.* *Timing*

Visual inspection is static, which might be ineffective in making maintenance and restoration decisions.

SHM studies have been carried out to understand better the ageing structure condition, which helps reduce risk, increase durability, and reduce maintenance expenses. Wired-based SHM collects data at different points on the structure, which are transmitted to the monitoring unit. The data is then processed, and decisions are made

about the structure (Farrar & Worden, 2007). SHM using wireless sensor (Hu, 2013) aims to assess the structural integrity using wireless transmission and signal processing approaches. The advancement of wireless sensing technology significantly reduces costs and provides flexible and powerful functionalities. The structure's performance can be evaluated for different loads, and damage can be identified in the real-time (Chae et al., 2012). With the increasing complexity and heterogeneity of data, efficient and effective data analysis has become an essential issue for decision-making concerning the diagnosis of the structural condition and the prognosis of structural damage (Smarsly et al., 2016).

The information obtained from sensor data is used to assess structural conditions such as the life cycle management (Hartmann et al., 2011) and life prediction (Smarsly et al., 2013). There exist two ways of assessing structural conditions, physics-based and data-driven. Physics-based approaches are more computationally intensive than data-driven approaches. In these approaches, the extraction of damage information from the sensor data relies primarily on the physical laws that govern the behaviour of the structure (Yuan et al., 2020). The application of physics-based techniques to the SHM has been limited due to the difficulties in modelling complex real-world structures, environmental variation, boundary conditions, operation, and uncertainty in material properties. The sensing technologies advancement has helped monitor many parameters on large and complex real-world structures, motivating the application of data-driven techniques in SHM. Machine learning algorithms, including deep learning models, have been extensively used to generate the knowledge of the structure from the data measured using sensors (Alvandi & Cremona, 2006; Doebling et al., 1998; Smarsly et al., 2016; Wu, 1992) and numerical simulations. It has become useful when (Smarsly et al., 2016)

i. the structures are complex to model,
ii. the computational efforts are to be reduced,
iii. there are a huge number of data, and
iv. the structure is isolated and unsafe

Recently, vibrational-based SHM techniques (Doebling et al., 1998; Fan & Qiao, 2011; Rafiei & Adeli, 2019) have been explored. The variation in physical properties, such as mass, stiffness and damping, can be derived from the measured vibration characteristics (Doebling et al., 1998). Vibration characteristics can be used along with

machine learning models for damage identification and quantification (Wang et al., 2021). Studies (Figueiredo et al., 2019; Pathirage et al., 2018; Pathirage et al., 2019; Rafiei & Adeli, 2019; Wang et al., 2018) have been conducted for the damage identification and quantification of civil engineering structures using vibration responses and machine learning algorithms. Damage identification and quantification are the essential components of the SHM (Wang et al., 2021). Recent studies by (Pathirage et al., 2018; Pathirage et al., 2019; Wang et al., 2020; Wang et al., 2018) have used mode shapes and natural frequencies to identify and quantify elemental structural damage.

Most traditional machine learning models are fast. However, it is generally not efficient when it must process large amounts of data in its raw form (Yuan et al., 2020). Ensemble or committee-based methods improve machine learning performance using the results from multiple models, defined as the base learners. Better prediction results can be achieved than a single model with reduced variance, bias, and training time. In many cases, the base learners use the same type of algorithms, while some methods use different algorithms. The base learners must be reasonably accurate and diverse enough to build an efficient ensemble model (Sun, 2018). Deep learning models are computationally expensive compared to traditional machine learning models. Still, it can be made more efficient if proper feature selection can be performed on the input data before feeding it into the learning pipeline.

## 1.2 Problem statement

Structural damage identification and quantification using modal information require more sensors to measure the responses, usually from the whole structure, to obtain mode shapes and related parameters accurately to ensure that enough modal information is available. This may not be practical for large-scale structures. Moreover, the cost incurred from the centralised data acquisition system, e.g., long cables, a large number of sensors, and an in-field server, is generally high (Cao & Liu, 2016). It also takes a long time to install the monitoring system when the measurements must be taken from the whole structure. Time domain responses such as acceleration responses are sensitive to structure condition changes and can be easily measured. Measurements from a small number of sensors can be used directly or along with signal processing techniques with a machine learning model for damage

identification and quantification. However, the acceleration response is dependent on other conditions such as loading. It can contain noise, making it hard for the algorithm to map the input pattern with the target of damage assessment. Traditional machine learning models are fast and give good results when the data size is small. However, they are no longer effective when the data size is big. Deep learning models generally work better for large datasets. They can automatically extract features from the raw input (Yuan et al., 2020), but it takes longer to get trained. The research is carried out to develop suitable and effective methods for damage identification and quantification using acceleration responses with both traditional machine learning and deep learning models.

## 1.3 Research Objectives

The proposed research uses time domain responses, such as acceleration, to develop robust machine learning models for damage identification and quantification for different scenarios. The following research objectives are explicitly set:

1. Explore using a smaller number of sensor measurements to monitor the health of the civil infrastructure.
2. Propose machine learning models that can provide good damage identification and quantification with reduced computational time and a smaller number of sensor measurements.
3. Study the effect of external conditions such as noise and system modelling error for damage identification and quantification.
4. Explore performing signal processing techniques like computing impulse response functions (IRFs) and autocorrelation on acceleration response and use it as an input to machine learning models and measure the performance.

## 1.4 Significance

Vibration-based damage identification and quantification methods have been explored recently using modal information and machine learning models. The proposed methods in this thesis use acceleration responses measured from a smaller number of sensors for damage identification and quantification. Further, the study explores using impulse response functions (IRF) extracted from acceleration responses together with machine learning algorithms for damage identification and quantification. IRF has the advantage of being an inherited system property.

The existing studies on structural elemental damage identification and quantification using deep learning models like autoencoders and deep residual networks are computationally expensive. An ensemble-based technique, random forest, used in many fields due to its high accuracy and stability, is developed together with PCA for dimensionality reduction. The proposed method using processed raw acceleration response gave good damage identification results with significantly less computation time than the existing deep learning models. Further, damage identification and computation time are improved using another ensemble-based technique, an extremely randomised tree (ERT) using IRF and PCA. The existing deep learning and proposed two methods in Chapter 3 and Chapter 4 are trained separately for the damage scenarios considered. Using the same IRF dataset in Chapter 4, a robust noise LSTM-autoencoder that is trained, validated, and tested on combined datasets without noise measurement data is proposed. The dataset for damage scenarios with noise measurements is used as another testing dataset for the trained model. The proposed method does not need to be trained and tested for the damage scenarios with noise measurements. It uses a noise layer before the LSTM-autoencoder helps generalise test datasets. The computation time is reduced using PCA as the dimensionality tool.

Further, civil engineering structures are exposed to different external conditions. The vibration measurements from the Z24 bridge in Switzerland during the short-term progressive damage tests were obtained under different damage types and levels for each damage type. The proposed methods in Chapters 3, 4 and 5 may become too complex or may not be suitable for predicting damage type and quantifying the damage. The current study using Z24 bridge data presented good damage classification, but there is the possibility to achieve better results using the same dataset. Moreover, it has used different data augmentation methods and measurements from five sensors. A classification model using a One-dimensional convolutional network (1D-CNN) using three sensor measurements is proposed for this task. 1D-CNN has the advantage over other convolutional networks in terms of computational complexities since it is a relatively simple architecture. The proposed D-CNN model uses two 1D-CNN with different kernel lengths, resulting in better damage classification than the current study.

## 1.5 Thesis outline

Chapter 1

It briefly provides an introduction and background to the thesis stating SHM and earlier techniques, followed by problem statements, research objectives, and the significance of the study.

Chapter 2

The literature review is presented in this chapter. It states the importance of SHM, and vibration based SHM. Further, a study on machine learning and deep learning models' application in SHM is discussed.

Chapter 3

A supervised ensemble-based traditional machine learning multi-output regression model, Random Forest (RF) with PCA using acceleration responses, is proposed. The input to RF is the principal components selected, and the output is the stiffness parameter reduction in the elements. A numerical study is carried out on a simply supported beam, and performance is measured for different damage scenarios; an experimental study is done on a seven-story steel frame structure.

Chapter 4

The RF model proposed in Chapter 3 is extended using impulse response functions (IRFs) computed from acceleration responses and compared with another ensemble-based traditional machine learning algorithm constructed from the decision tree, ERT. The IRFs is calculated from the acceleration response generated from the simply supported beam and the seven-storey steel frame structure. PCA is again employed on the processed IRFs for dimensionality reduction. The input to RF and ERT is the principal components selected, and the output is the stiffness parameter reduction in the elements. ERT output performs the RF. Further, the performance of ERT with IRF is measured using a smaller number of sensors.

Chapter 5

The proposed models were trained and tested separately for all the damage scenarios considered in the first two methods and existing deep learning models for the damage identification in structural elements. A deep learning model based on Long Short-Term Memory (LSTM) is proposed, which gets trained and validated on the dataset of

combined two scenarios without noise in the measurement and tested with datasets with noise. The proposed approach uses the IRF dataset obtained in Chapter 4. It has the advantage of generalising data over RF and ERT. PCA is employed on the processed IRF and used as an input to the proposed LSTM autoencoder to reduce the computational time. The performance is measured for scenarios considered in Chapters 3 and chapter 4.

Chapter 6

Chapter 3-5 proposed models based on machine learning for structural damage identification and quantification, defined in terms of stiffness parameter reduction. A real large-scale structure can undergo many damage types and be exposed to many external conditions that may impact the acceleration responses. The regressor models proposed in earlier chapters can become complex due to the need to identify different damage scenarios and then quantify the damage level for other damages when the data available are small. A 1D-Convolutional Neural Network (1D-CNN) is proposed for damage classification using a smaller number of sensors and provides higher damage classification accuracy than the current study.

Chapter 7

The last chapter of the thesis concludes with the findings of the studies carried out for damage identification and quantification. Further, it presents future works that can be explored to improve the performance considering more uncertainties.

# CHAPTER 2

## 2 Literature review

This chapter provides a literature review on the development and applications of machine learning techniques for structural health monitoring (SHM) and discusses the limitation of model-based SHM. In the vibration-based SHM section, the traditional vibration-based damage identification methods are reviewed, followed by discussions on artificial neural network applications. Both machine learning techniques and deep learning models using vibration responses are presented. The discussions on structural element damage identification and quantification are explained in detail to understand the problems in existing studies. Further, studies on using ensemble learning methods for SHM have been presented. It also discusses using the impulse response functions (IRF) extracted from time domain responses for SHM. Studies on principal component analysis (PCA), which is used as a dimensionality reduction technique for SHM, are also reviewed.

### 2.1 Structural health monitoring

SHM techniques have been developed to perform condition evaluation and monitoring of civil and mechanical engineering structures. It assists structural engineers in ensuring structural integrity and safety, reduces capital maintenance expenditure and increases the service lifespan of engineering structures (Karbhari & Ansari, 2009). SHM in civil engineering uses various types of sensors to collect response and loading information from the structures and interpret these measurements to monitor the health condition and performance of the structure (Ye et al., 2019). The increasing complexity and heterogeneity of the sensor data make it difficult for decision-makers to diagnose structural conditions and prognosis of structural damage (Smarsly et al., 2013). Many techniques have been explored to monitor the structure's health condition. The methods can be model-based methods or data-driven methods. Model-based approaches use finite element models and strategies for model updating for damage identification and quantification. The parameters of the FEM model are updated concerning the measured responses for the damage identification (Kim & Kawatani, 2998; Weber & Paultre, 2010). In the study (Nicknam et al., 2011), a damage identification method is proposed for damage detection using fundamental mode shape transformation and curvelet transform. The proposed study was

investigated with 5% and 10% noise. Amiri et al. (2013) studied two damage identification and quantification in shear frames. In the first method, residual modal force is adopted for damage computation. In the second method, the damage computation is carried out based on the static displacements under unique static force using modal data. More studies (Amiri et al., 2015; Li et al., 2018; Roy, 2017; Rucevskis et al., 2016; Yong & Hao, 2003) utilise modal information like natural frequencies and mode shapes for damage detection.

Further, model-updating techniques have been explored for more efficient structural damage detection. Haidarpour and Tee proposed a model updating approach for damage identification and characterisation in structural and mechanical systems (Haidarpour & Tee, 2020) by examining the change in the responses. A hybrid experimental/computation model-based framework (Giagopoulos et al., 2018) was proposed for fatigue damage and remaining lifetime estimation in a linear steel substructure of a lignite grinder assembly at a power plant. Operational vibration measurements collected from the limited number of sensors are used to update the numerical FEM model. Sipple and Sanayei presented a new method for FEM updating using frequency response and numerical sensitivities to solve the inverse problems (Sipple & Sanayei, 2013). However, model-based methods have difficulty modelling a complex real-world structure due to uncertainty in material properties, environmental conditions, construction flaws and boundary conditions (Yuan et al., 2020).

Owing to the limitation of physics-based techniques, machine learning algorithms are used to generate knowledge of the structure with the data collected from the sensors and numerical simulations. It has been applied for the safety assessment and health monitoring of civil engineering infrastructure through its capability of deriving relationships in the datasets (Chencho et al., 2020). Thus, machine learning algorithms have become powerful when the structure is complex to model. There is a need to reduce computational effort owing to generating a massive amount of data (Smarsly et al., 2016).

## 2.2 Vibration-based SHM

Vibration-based methods (Doebling et al., 1998; Figueiredo et al., 2019; Rafiei & Adeli, 2019; Wang et al., 2021; Wu & Jahanshahi, 2018) have been extensively developed for SHM to identify and quantify damage based on vibration responses.

These methods have been widely used in the health monitoring of machines and their components by measuring vibration responses such as displacement, strain, and acceleration (Heng et al., 2009; Jardine et al., 2006; Kan et al., 2015). Vibration-based methods have also been developed for SHM of civil engineering infrastructure that ultimately overcomes the issues with the traditional structural condition monitoring and safety evaluation of the monitored structures (Avci et al., 2021b; Mansouri et al., 2015; Shadan et al., 2016). Magalhaes et al. (2012) installed force balance accelerometers within the deck box girders to measure both vertical and lateral accelerations to examine the aging and degradation of a bridge. The natural frequency changes over time were monitored with the surrounding temperature to observe possible changes in the bridge's condition. The natural frequency variation over time with the RMS value of the vertical acceleration and the average day evolution of the modal damping ratios and natural frequency changes during working days were also obtained. A method was proposed to minimise the effects of environmental and operational factors on the natural frequencies, which helped to identify the structural anomalies in the subsequent stage (Magalhaes et al., 2012).

Artificial neural networks (ANN) were used along with the feature extraction methods for damage identification and localisation. A study (Mehrjoo et al., 2008) conducted a numerical study on a simple warren truss and Louisville bridge truss for damage detection of joints using modal characteristics extracted from acceleration responses. Damage was introduced by reducing the truss member’s stiffness, and blending modes were used for training the model. The first four modes of truss bridge and five modes of Louisville bridge were used for training the model. The results indicated good damage identification performance. However, the complete training process took 75,000 epochs, which is quite large. An application of Bayesian model class selection to select an optimal ANN model class was proposed for detecting damage in a four-story, 22-bay steel frame with 120 degrees-of-freedom (Ng, 2014) using modal characteristics of the model under several structural damage states. A combined method (Betti et al., 2015) consisting of ANN and genetic algorithm for structural damage identification of a three-story steel structure was developed. Structural damage was introduced by partial cuts of flanges on a column. Acceleration measurements under ambient conditions from undamaged and damaged levels were recorded to obtain natural frequencies and mode shapes through a neural network. The inputs to

the neural network are four-frequency dependent indexes and output an estimate of the likelihood of finding a structural eigenfrequency. The minimisation of modal characteristics between the experimental and numerical studies was carried out using two fitness functions, and the damage identification was tested using the genetic algorithm. Other studies (Cury & Cremona, 2012; Goh et al., 2013; Lee & Lam, 2011; Xhou et al., 2014) have also used modal information with machine learning models for damage identification and localisation.

Machine learning algorithms have also been used for damage identification and localisation with time domain measurements, mostly the structures' acceleration responses. It does not require extracting the parameters like the modal information from the measurements recorded from the sensors. Autoregressive model (AR) and Residual Error (RE) were used to extract damage-sensitive features for the support vector machine (SVM) in the study by Gui et al. (2017) to detect damage in frame structure and using time series measurement. The hyperparameters of SVM were tuned with three different optimisation techniques. The numerical results demonstrated the improvement in the performance over the conventional methods in terms of sensitivity, accuracy, and effectiveness.

A numerical study by (Dackermann et al., 2016) using principal component analysis (PCA) with ANN and ensembles of ANN was conducted for the damage identification and localisation of pin-pin supported steel beam. Modal parameters were extracted from the time domain responses, and the modal strain energy-based damage index was derived. PCA was applied to the damage index, and significant principal components were selected as an input to the ANN for damage identification and localisation. PCA was utilised to reduce the effect of noise, and performance measurement was measured with different levels of white noise. A three-stage ANN method for damage assessment of building structures was proposed in the study. The first stage determined the damaged floor, and a specific damaged element was identified in the second. The damage severity was identified in the third stage. The study was carried out on a numerical model of a 10-storey frame structure with the frequency response function obtained from the acceleration measurements. The result indicated that the PCA-based damage index was suitable for structural damage detection. PCA was used with noise filtering for feature extraction. Some researchers (Abdeljaber & Avci, 2016) proposed a nonparametric structural damage detection method using self-organising maps to

extract damage indices from ambient acceleration measurements. The study was conducted to identify and locate damage due to stiffness reduction and change in boundary conditions of a numerical model of a hot-rolled steel grid structure. Machine learning models are fast to train and do not require a high system configuration for training.

Deep learning models have been extensively used for the damage identification and quantification of civil engineering structures. It is a subset of machine learning methods. It can learn from the data in an unsupervised manner and extract the optimal input representation from the raw data without the user's intervention. Thus, deep learning models can learn how to correlate features to the desired output, and extract features (Avci et al., 2021b). An ensemble classification method for structural damage assessment was proposed using FRF generated from a numerical model (Fallahian et al., 2017). Deep neural network and couple sparse coding are two classifiers used for the damage classification, and the decision is made based on the majority voting. The input to the classifiers is the features extracted by using PCA. A numerical study was carried out on a truss bridge with 25 elements and validated with data obtained from the I-40 bridge. Good damage classification results were provided, considering temperature variations.

Auto-encoders perform hierarchical non-linear mapping to learn features representing datasets (Vincent et al., 2010) and have also been used for dimensionality reduction. In studies (Fallahian et al., 2017; Fallahian et al., 2018; Pathirage et al., 2019), autoencoders have been used for damage identification and quantification. Recent studies (Pathirage et al., 2018; Wang et al., 2020; Wang et al., 2018) on structural elemental damage identification and localisation using auto-encoders have the natural frequencies and mode shapes as the input, and the stiffness reduction in the structural elements as the output. The variation in physical properties, such as mass, stiffness, and damping, can be derived from the measured vibration characteristics of (Doebling et al., 1998), including natural frequencies and mode shapes.

A numerical study by (Pathirage et al., 2018) investigated an application of a deep autoencoder model for damage identification and quantification using modal information, natural frequencies and modes shapes. The input to the proposed model were natural frequencies and mode shapes of the first seven modes, and the elemental

stiffness parameter is taken as the output. For training and testing the proposed model, an updated finite element model (FEM) of seven storied frame structures is used for the data generation. The performance was measured for stiffness prediction for single and multiple element damage cases. In another numerical study, using the same dataset, a novel parallel auto-encoder framework (Wang et al., 2018) was proposed using the same seven-storey steel structure to predict structural elemental stiffness reduction. The sparse dimensionality reduction is performed parallelly for natural frequencies and mode shapes. The reduced dimension of natural frequency and modes shapes are combined and used as an input to predict stiffness reduction in the structural elements. The performance was measured for single and multiple element damage cases. However, no uncertainties in the modelling system and noise measurements are considered in those studies.

The same study was further extended using autoencoders (Pathirage et al., 2018) with noise measurement and uncertainty effect in the modelling system with experimental verification. There were two components, one to reduce the dimensionality of the input and another component to map features with reduced dimensionality and elemental stiffness parameter. The inputs to the model are natural frequencies and mode shapes, and labelled outputs are elemental stiffness parameters. In the study, four scenarios were considered to measure the performance of their proposed model. The four scenarios were defined based on the noise measurement and uncertainty in the system modelling. Scenario 1 has no noise measurement and uncertainty effect. Scenario 2 considers measurement with 1% random noise in frequencies and 5% in mode shapes. Scenario 3 takes a 1% uncertainty effect in elemental stiffness parameters. Scenario 4 considers both noise measurement and uncertainty effect. For every scenario, damage identification and quantification were predicted for single element and multiple element damage cases. The model was trained, tested, and validated with natural frequencies and mode shapes generated from the FEM model of a frame structure. It was demonstrated that their proposed model gave good damage prediction results. However, it is observed that the performance degrades when noise measurement and uncertainty effects are considered.

Pathirage et al. (2019) proposed a deep learning-based sparse autoencoder framework to improve the performance of the same problem. A sparse dimensionality reduction was performed, followed by relation learning between the reduced dimension input

and output. It compresses the dimensionality of the features, preserving helpful information but also robust to the effect of noise and uncertainties (Pathirage et al., 2019). With sparse dimensionality reduction as the additional component, the performance was measured for four scenarios considered in their previous study for both single and multiple element damage cases. The sparse autoencoder-based deep learning model has improved damage identification and quantification. Further, a deep residual network framework (Wang et al., 2020) for structural damage quantification was proposed using the natural frequencies and mode shapes. The proposed framework comprises a residual neural network for feature extraction and a fully connected layer as a regressor. The same dataset of the seven-storey frame structure used in studies (Pathirage et al., 2018; Pathirage et al., 2019) was used in the study. The performance was measured and compared with the sparse autoencoder network by Pathirage et al. (2019), considering all four scenarios, and it outperformed in all the scenarios.

Several sensor measurements were required at several locations of the structure to obtain accurate modal information, such as the mode shapes (Chencho et al., 2020; Wang et al., 2021). This can be expensive because of the costs incurred on long cables, sensors, and data acquisition systems. Time domain responses can overcome the issue by measuring modal information for damage identification and localisation. Abdelijaber (2017) proposed a real-time damage detection and localisation method using adaptive one-dimensional (1D) CNN and raw signal measurements with accelerometers. Wang et al. (2021) used only a few sensors and time-domain responses and achieved excellent damage identification and quantification results using densely connected neural networks. Chencho et al. (2020) conducted a similar study using principal component analysis (PCA) and an ensemble-based machine learning algorithm, namely, the random forest. The damage identification and quantification results were close to those from the deep learning models at a relatively low computational cost. PCA has been used to extract information from high-dimensional data that can be represented using a few principal components (Sarbu & Pop, 2005). PCA is faster than auto-encoder when used for dimensionality reduction.

Convolutional neural networks (CNNs) are supervised, deep learning models. They are widely used in SHM owing to a strong capability for an efficient and robust feature learning (Wang et al., 2020). CNN's were first introduced for digit recognition tasks and have also been widely used in SHM (Abdeljaber et al., 2017; Azimi & Pekcan,

2020; Seventekidis et al., 2020). Standard two-dimensional convolutional neural networks (2D-CNN) models are used for classification problems based on images. These models can learn representative features from two-dimensional input variables. The kernel slides along two data dimensions; hence, it is called a 2D-CNN. Similarly, when the kernel slides along one dimension, this convolutional neural network is called a one-dimensional convolutional neural network (1D-CNN). The only difference among these networks is the input dimensions and how the filter or kernel slides across the data.

2D-CNN has been used for damage identification and localisation using vibration responses. Khodabandehlou et al. (2018) carried out vibration-based SHM using a 2D-CNN to classify four damage classes using the vibration measurements from a shaking table testing of a reinforced concrete highway bridge. The 48 measurements from shake table experiments were used for training and testing the CNN model. A deep convolutional neural network based on 2D-CNN was proposed by Yu et al. (2018). The study was carried out on a five-level benchmark building model for damage identification. Fast Fourier Transform (FFT) was performed on the vibration signal measurements to convert the time domain signal into the frequency domain. The selected frequency bands from different sensors were converted to a two-dimensional (2D) feature matrix for their proposed model. However, 2D-CNN is best used when the data are 2D, and this can increase the computational cost. Further, the deep 2D-CNNs hardware requirement is more expensive due to the need for special parallelised hardware setups (Avci et al., 2021a), and recent studies have explored 1D-CNN for the structural damage identification and localisation using vibration responses.

1D-CNN networks have the advantage over the other CNN work in terms of computational complexities and shallow architecture; this makes the network easy to train and implement. Further, 1D-CNN does not require high system configuration, making it suitable for real-time and low-cost applications (Mitiche et al., 2020). In the study (Goodfellow et al., 2016), it was demonstrated that the 1D-CNN outperformed other machine learning algorithms like support vector machine and random forest. In the study (Abdeljaber et al., 2017), a structural damage detection system using 1D-CNN was designed to fuse feature extraction and classification blocks into a single learning body. Zhang et al. (2019) conducted a study using 1D-CNN and a wireless sensor network for structural health monitoring of bridge structures.

The damage detection in a steel frame using only two sets of measured acceleration signals regardless of the size of the structure using 1D-CNN was proposed (Abdeljaber et al., 2018). The proposed study provided good damage detection but required one 1D-CNN for each sensor measurement for training. A hybrid model consisting of 1D-CNN and Long Short-Term memory network (Hung et al., 2020) was proposed for damage identification using a series of experimentally measured vibration data from a frame structure constructed in the laboratory. Data augmentation was performed to increase the size of the experimental data. A similar study (Hung et al., 2021) using 1D-CNN and LSTM was carried out and tested on many structures, including the Z24 bridge. Large-scale civil infrastructures like bridges are exposed to external conditions such as wind, temperature, vehicle loads (Hung et al., 2021) and different types of damage. Z24 bridge progressive damage test (PDT) dataset might contain much noise, which is inevitable in a real structure due to external conditions. This can make it hard for the machine learning models to map input patterns with the damaged label or the type. The existing hybrid model using data fusion techniques (Hung et al., 2021) on Z24 data has shown damage classification accuracy of 90.1%. In their study, different methods were adopted initially to extract the features from the raw data and then fused to feed as the input to the hybrid model. Further, data augmentation techniques were performed to have sufficient data to train and test their proposed hybrid model. Five sensor measurements were used in their study.

Li et al. (2015) proposed a structural damage identification model that used a sensitivity-based method and the impulse response functions (IRFs) extracted from the acceleration responses. Lin et al. (2019) proposed a method to estimate IRFs from structural responses recorded from multiple unknown excitations and utilised IRFs for structural damage identification and quantification. An equivalent single excitation problem was obtained for various general excitations, and IRFs were estimated using wavelet and regularisation methods. The performance was measured with and without noise for a simply supported plane truss numerical model. Smith and Hernandez (2017) explored impulse response sensitivity and least absolute shrinkage and selection operator (LASSO) regularisation for sparse damage detection in terms of stiffness reduction of a numerical model of a non-uniform shear beam with noise measurement limited model parameters and limited sensor locations. The results demonstrated that sparse damage could be detected using changes in identified impulse

responses. A study (Todorovska & Trifunac, 2008) carried out on damage detection of a full-scale building damaged by the earthquake. The changes in wave travel times of seismic waves propagation through structural members were measured to detect the changes in stiffness. The recorded horizontal responses before, during, and after the earthquake were used to compute the impulse response function (IRFs), and wave travel times were obtained from the IRFs (Todorovska & Trifunac, 2008). A study was conducted to identify damage within wall structures using the Hilbert marginal energy ratio spectrums of IRFs and virtual IRFs (Xu, 2021). Hilbert damage feature vector was created based on selected feature bands that were more sensitive to damage within the retaining walls. IRFs can be an ideal inherent vibration property for structural damage identification. IRFs are one of the dynamic characteristics of structures, representing the time domain response of a structure under the input of an impulse excitation. It will vary owing to the change in physical properties of the structure, and it is related only to the excitation location (Lin et al., 2019). IRF has the advantage of using measured time domain responses directly as input to machine learning algorithms (Li et al., 2015).

## 2.3 Ensemble learning

Ensemble learning-based techniques have been developed to improve the performance of machine learning algorithms. It uses several similar or different models, resulting in better predictions by decreasing the variance and bias. Ensembles of artificial neural networks (ANNs) have been used and improved further for damage detection, localisation, and quantification of an aeronautical structure with strain field modification (Francesco et al., 2017; Sbarufatti, 2017; Sbarufatti et al., 2013). Ensemble-based models can be either regression or classification or predictive models. Knowing the difference between them is essential and choosing the best that suits the problem. The predicted results can fall into one or more classes with classification models, even when the input does not belong to any classes. This is not the case with the regression problem. The regression problems have numerical or continuous labelled output.

Ensemble methods based on traditional machine learning models like decision trees are developed for structural monitoring. Decision trees are non-parametric supervised machine learning algorithms used for regression and classification problems (Chencho

et al., 2020). However, a single decision tree usually suffers from overfitting, resulting in high variance and a low bias. Breiman (1996) proposed Random Forest (RF) to reduce the variance and increase the bias. RF is a homogenous ensemble-based supervised algorithm built from several decision trees as a base learner. The variance problem is reduced by introducing randomness, taking ensembles of decision trees, using bootstrapped samples, and splitting nodes at the best split (Breiman, 1996; Chencho et al., 2020; Lawson et al., 2017; Mariniello et al., 2020).

RF was used as a classifier to identify damage in a shear frame structure (Zhou et al., 2013). Further, a study (Zhou et al., 2014) was carried out using the RF as a feature extractor to eliminate the least important features. Structural damage detection and localisation using decision tree ensemble and modal information have been discussed by Mariniello et al. (2020). Mode shapes and natural frequencies are input to the decision tree ensemble model. The RF output is the average of the outputs from all decision trees in the case of regression problems. For classification problems, the majority vote of the classes is taken as the final output. In all the above studies, RF is used as a classifier. Kundu et al. (2020) proposed a method using RF as the regressor to predict the remaining useful life of spur gears under natural pitting progression by correlation coefficient parameter based on residual vibration signals.

Although RF requires growing several decision trees using bootstrap data, it is still popular owing to the low computational cost of growing the decision trees. RF provides some degree of randomisation by taking a random subset of features and bootstrapped samples, unlike a decision tree. It gives a better result than a decision tree in terms of accuracy. In a current study (Chencho et al., 2020), the development and application of RF for elemental level structural damage quantification were presented to overcome the limitations of existing structural elemental damage quantification. The principal component analysis was performed on the input for dimensionality reduction. The study used acceleration responses measured from a smaller number of sensors and a machine learning algorithm. A numerical study was carried out on a simply supported beam with 10 elements, and experimental validation was carried out on a seven-storey steel frame structure with 70 elements. The input to RF was the processed acceleration measurement, and the output is the reduction and stiffness parameter. The RF developed gave good structural elemental damage identification. The

computational cost was far less than the deep learning models for structural elemental damage quantification.

Another ensemble technique, Extremely Randomised Trees (ERT), is a multi-output regressor. Both RF and ERT are homogenous ensemble-based algorithms. ERT uses multiple decision trees and takes the average of all the outputs. However, compared with RF, there are two essential differences. RF selects the best feature with the high information gain or minimum error for the best split. To find the best feature, it has to thoroughly evaluate all the features in the set of randomly selected features (Lawson et al., 2017). For a given problem, ERT randomly selects a feature from a set of randomly chosen features to split at the node (Geurts et al., 2006; Lawson et al., 2017), providing a higher level of randomisation. The other difference is that ERT does not bootstrap the initial dataset. It uses whole learning samples to grow the trees. The performance of ERT in terms of accuracy can be slightly better than RF. However, the computational time for ERT will be less than RF because it does not have to check all the features for the best split.

### 2.4 Dimensionality reduction technique: Principal component analysis

Dimensionality reduction can help machine learning models to avoid the curse of dimensionality, which does not occur with low dimensional data. The high-dimensional data can be projected into a lower dimension without losing much information. There are many feature projection techniques like principal component analysis (PCA), t-distributed stochastic neighbourhood embedding (T-SNE), linear discriminant analysis (LDA), autoencoders etc. Autoencoders are unsupervised artificial neural networks that can map higher dimensional data into a linear or non-linear lower dimensional data (Lin et al., 2021; Pathirage et al., 2019; Zabalza et al., 2016). However, its neural network architecture is slower than the other techniques. An analysis of dimensionality reduction techniques on big data was conducted to investigate the performance of traditional machine learning models with PCA and LDA (Reddy et al., 2020). It has shown that the performance of machine learning models with PCA outperforms models with LDA. For the study on using machine learning techniques for detecting and classifying structural damage, Nick et al. (2015) investigated the supervised and unsupervised machine learning algorithms to classify

acoustic signals and associated crack growth. The results were tested with and without PCA for dimensionality reduction.

PCA tries to retain information on the global structure of data, and it is widely used across different fields for data compression and extracting information from high-dimensional data. It has been used in the area of SHM for performing the dimensionality reduction of the input dataset and for removing the uncertainties, e.g., measurement noise and environmental effects (Nie et al., 2020), whilst retaining most of the information of the original data (Brownlee, 2019a; Jolliffe & Cadima, 2016; Reddy et al., 2020). For the datasets measured for a long duration, the number of sampling points becomes significantly large, which increases the computational time for obtaining the covariance matrix and principal components. The principal components are obtained by performing Eigen decomposition of covariance matrix from the input data. The eigenvectors can be arranged in the descending order of the eigenvalues. The number of components can be selected based on certain criteria on the singular values or energy of the covariance (Brownlee, 2019a; Nie et al., 2020). Posenta et al. (2008) proposed using a moving window of a constant size, and the covariance matrix was carried out inside the window. Nie et al. (2020) utilised fixed moving principal component analysis with a sliding fixed window to detect the time instant of damage occurrence for real-time SHM of bridge structures.

## 2.5 Discussions

Structural damage identification methods using auto-encoders and deep residual networks using modal information data are discussed. Many sensors are required to extract accurate modal information. The measurements should be taken from several locations on the structure. Obtaining the measurement for modal information can be expensive. Installation and maintenance of these systems are time-consuming and have problems with accessibility when there are regions on the structure which could not be reached. Further, the deep learning models are computationally more expensive than the traditional machine learning models.

The ensemble technique can improve the performance. However, using ensembles of deep learning models can further increase the computation time because the results are mostly the average of all the models. Traditional ensemble machine learning models are computationally less expensive and have demonstrated good results for

classification and regression problems. Performing dimensionality reduction using PCA can reduce the computational time of both deep learning and traditional machine learning models. Besides, it helps remove the noise and the uncertainty effects in the datasets. The existing studies (Chencho et al., 2020; Wang et al., 2021; Zhang et al., 2019) demonstrated good damage identification and quantification results using a smaller number of sensors and time domain responses. Using acceleration responses for structural elemental damage identification and quantification does not require more sensors. Moreover, it is sensitive to structural conditions. Using IRFs extracted from the acceleration response as the input to the machine learning model may give better damage identification results.

It was also observed in the studies (Chencho et al., 2020; Pathirage et al., 2018; Pathirage et al., 2019; Wang et al., 2020; Wang et al., 2021; Wang et al., 2018), that the input data are modal information or acceleration responses generated or measured for the reduction in the stiffness parameter considering the problem as a regression problem. The damage identification was carried out for single element and multiple element damage cases for four scenarios considering noise measurement and uncertainty in system modelling. There were four different sets of datasets, one for each scenario. Their proposed models' performance for each scenario was measured using the specific dataset for training, testing and validation. However, it would be more appropriate to come up with a model which can be trained with a dataset and be able to generalise for all the dataset without the need to train, test and validate each scenario. Moreover, input data were either modal information or time domain responses measured for a uniform step size of stiffness reduction. This makes it suitable to develop machine learning models as regression models. A civil infrastructure may experience different types of damage and different levels for each damage type. The data available may not be sufficient to develop a regression model. For example, the Z24 bridge in Switzerland considers the other kind of damage and levels for each damage type. The time domain measurements were made for different levels for each damage type. A regression model may not be suitable or become too complex when the problem consists of more damage types, and there is the need to quantify each damage type. Therefore, an appropriate classification model may be suitable for this problem.

## 2.6 Summary

Table 2.1. lists the advantages and disadvantages of some of the methods discussed. Based on the above discussions, the limitations of existing methods on structural elemental damage identification and quantification may be mitigated with ensemble-based traditional machine learning techniques and an LSTM-based model, which can generalise the dataset with noise measurement and system modelling uncertainty. For training and testing the proposed models, numerical simulation of updated FEM models can be used for data generation and validated with experimental studies.

Further, the 1D-CNN model can be used for damage classification or identification when time domain responses are available for large-scale civil engineering structures. The use of five sensor measurements in the current study for Z24 using a hybrid deep learning model has indicated that a smaller number of sensor measurements of time domain responses can be used for damage classification and quantification in the structure. The accuracy can still be improved to get good damage classification using smaller number of sensors. The research methodologies are developed based on the gaps mentioned, and numerical and experimental studies are conducted in the following chapters.

Table 2.1 Advantages and disadvantages of some machine learning methods for damage identification and localization

| Reference | Models | Inputs | Advantage | Disadvantage |
|---|---|---|---|---|
| Dackermann et al. (2016) | PCA with ANN ensemble | Modal Information | Good damage identification and localization. The effect noise is reduced using PCA. | Requires more sensors to obtain modal information. |
| Fallahian et al. (2017) | Deep neural network and couple sparse coding | Frequency Response Function (FRF) | Good damage identification considering noise and temperature variations. | Long training time. |
| Pathiriage et al. (2018) | Autoencoder | Modal Information | Comprises of two components for dimensionality reduction and relationship learning. Good damage prediction in structural elements. | Require more sensors, long training time, Difficult to obtain measurement from complex structure. |
| Wang et al. (2018) | Parallel Autoencoder | Modal Information | Processes natural frequency and mode shapes separately and provides good damage prediction in structural elements. | Require more sensors, long training time, Difficult to obtain measurement from complex structure. |
| Pathariage et al. (2019) | Sparse Autoencoder | Modal Information | Enhances the capability of dimensionality reduction and relationship learning. Good damage prediction in structural elements. More robust to noise and uncertainty considered in their study | Require more sensors, long training time, Difficult to obtain measurement from complex structure. requires separate training for all damage scenarios considered. Requires separate training for all damage scenarios considered. |
| Wang et al. (2020) | Residual Network | Modal Information | Avoids the problem of vanishing gradient by using skip connection. Good damage prediction in structural elements | Require more sensors, long training time, Difficult to obtain measurement from complex structure. Requires separate training for all damage scenarios considered. |

| | | | | |
|---|---|---|---|---|
| Abdelijaber (2017) | Adaptive 1D-CNN | Acceleration responses | Real time damage detection and localization from raw acceleration response and has an inherent adaptive design to fuse both feature extraction and damage localization. | Takes longer training time because it uses 1D-CNN for each acceleration response |
| Wang et al. (2021) | Densely Connected Network | Acceleration responses | Strengthen feature propagation through the network and reduces the parameters making easy for training. Does not require a lot of sensor measurement. | Large training time and requires separate training for all damage scenarios considered. |
| Yu et al. (2018) | 2D-CNN | Frequency bands obtained from time domain signals | Capable of automatically extracting high-level or low-level features. Good damage classification | Requires 2D data which increases computational cost. More expensive due to the need of special hardware setups. |
| Abdelijaber et al. (2018) | 1D-CNN | Acceleration responses | Requires only two sets of measurements for damage identification regardless of size of the structure and gives good damage detection result. | Takes longer training time because it uses 1D-CNN for each acceleration response. |
| Hung at al. (2021) | Hybrid LSTM and 1D-CNN | Acceleration responses | Uses only five sensor measurement from a large-scale bridge, Z24 and provide good damage classification. | Used data augmentation methods to increase data size and uses features extracted by using different methods. Longer training time. |
| Mariniello et al. (2020) | Decision tree ensemble | Modal Information | Low computational cost and provides some degree of randomization giving better results. | Uses modal information which requires more sensors to take measurement. Difficult to obtain measurement from a complex structure. |

# CHAPTER 3

# 3 Damage Identification and Quantification Using Acceleration Response and Random Forest

## 3.1 Introduction

This chapter presents element-level structural damage quantification using an ensemble-based machine learning technique, the random forest (RF), with acceleration responses from the structure. A random forest is a machine learning algorithm with several decision trees to perform a task. The proposed approach develops a random forest as a regressor to predict multiple output variables, which are structure's elemental stiffness reductions. The acceleration responses for single-element and multiple-element damage cases are generated and further processed to feed as input to the random forest. The acceleration responses from the sensor nodes are concatenated, and principal component analysis (PCA) is applied to reduce the uncorrelated input dimension. To measure the performance of the proposed method, a simply supported beam is used as an example in numerical studies. Experimental studies on a steel frame structure with 70 elements are also conducted to investigate the performance of the proposed approach for structural damage quantification.

## 3.2 Random Forest Technique

Random forest is a supervised machine learning algorithm that uses a homogenous ensemble technique. It uses several decision trees as the base learners. A decision tree is a non-parametric supervised learning method. It can be used to predict the value of output variables by learning simple rules inferred from the features. It can be used for both regression and classification problems. A decision tree-based classifier can classify both binary and multiclass problems. It takes an input array of size *[N_samples, M_features]* and an output array values (class labels) with size *[N_samples]* for a binary class and size of *[N_samples, n_outputs]* for the multiclass problem, where *'N'* is the number of samples, *'M'* is the total number of features, and *'n'* is the number of the target output variable. The output array is expected to have floating-point values instead of integer values when used as a regressor.

A decision tree-based regression model breaks down the training dataset of size *[N_samples, M_features]* recursively into smaller subsets until each subset contains a

constant value with a minimum error (Hastie et al., 2009) or until it reaches the maximum depth. There are different types of tree algorithms. Some typical algorithms include Iterative Dichotometer 3 (ID3), classification algorithms (C4.5, C5.0), and Classification and Regression Trees (CART). They differ based on their applicability to numerical and categorical features. CART supports numerical target values, and a CART-based decision tree is used in this study.

### 3.2.1 Decision Tree as a regressor

A regression problem can have an output or multiple outputs. The model uses input variables (predictors) to predict output. For a decision tree, the leaf node will contain a constant value. Assuming that there are '*n*' outputs, the decision tree will have to predict multiple outputs from the input variables. In this case, the tree stores '*n*' outputs instead of a constant value and uses the splitting criteria to compute the average reduction across all the '*n*' outputs. In this study, the model predicts the stiffness reduction of all the structure elements when it receives a sample dataset of measured acceleration responses from the sensors. Therefore, the problem is a multi-output regression problem.

A decision tree recursively partitions the space such that the samples with the same labels are grouped. The decision node '*d*' is considered to have a data size ***'N'***. For each split consisting of a feature '*f*' and threshold '$t_f$', ***'N'*** is split into Left and $\boldsymbol{N_{right}}$ subsets. ***x*** and ***y*** are, respectively, the input sample and labelled output

$$\boldsymbol{N_{left}}=\boldsymbol{(x,y)} \text{ for '}f\text{' when less than or equal to threshold '}t_f\text{'} \quad (3.1)$$

$$\boldsymbol{N_{right}}=\boldsymbol{N}\backslash\boldsymbol{N_{left}} \quad (3.2)$$

After every split, the impurity at the decision node '*d*' is computed using the impurity function. For a proposed regression model, the impurity at node '*d*' is computed using Mean Square Error (MSE). The mean value is calculated as follows

$$\bar{y}_d = \frac{1}{N_d} \sum_{i \epsilon N_d} y_i \quad (3.3)$$

where '*i*' is the sample at the decision node, and '$N_d$' is the number of samples at the decision node 'd'. Then MSE is obtained as

$$\text{MSE} = \frac{1}{N_d} \sum_{i \epsilon N_d} (y_i - \bar{y}d)^2 \quad (3.4)$$

Subsets $\boldsymbol{N_{left}}$ and $\boldsymbol{N_{right}}$ are recursive until the maximum allowable depth is reached, where $N_d < min_{samples}$ or $N_d$=1. For '*n*' target variables, the average MSE is calculated across all '*n*' target variables.

Figure 3.1 shows an example of a decision tree for a multi-output regression problem considering ten samples. At the decision node, a feature is selected as the best splitting feature, and outputs are constant values since the problem considered is a multi-output regression problem. Mean squared error (MSE) is mainly used as the splitting criteria, the average reduction across all '*n*' outputs. For the given an example, the minimum number of samples required to split at the decision node is four. However, the decision tree may fit data too well and cause overfitting resulting in a high variance. A single random forest model that can simultaneously predict all 'n' outputs is built to alleviate the overfitting problem.

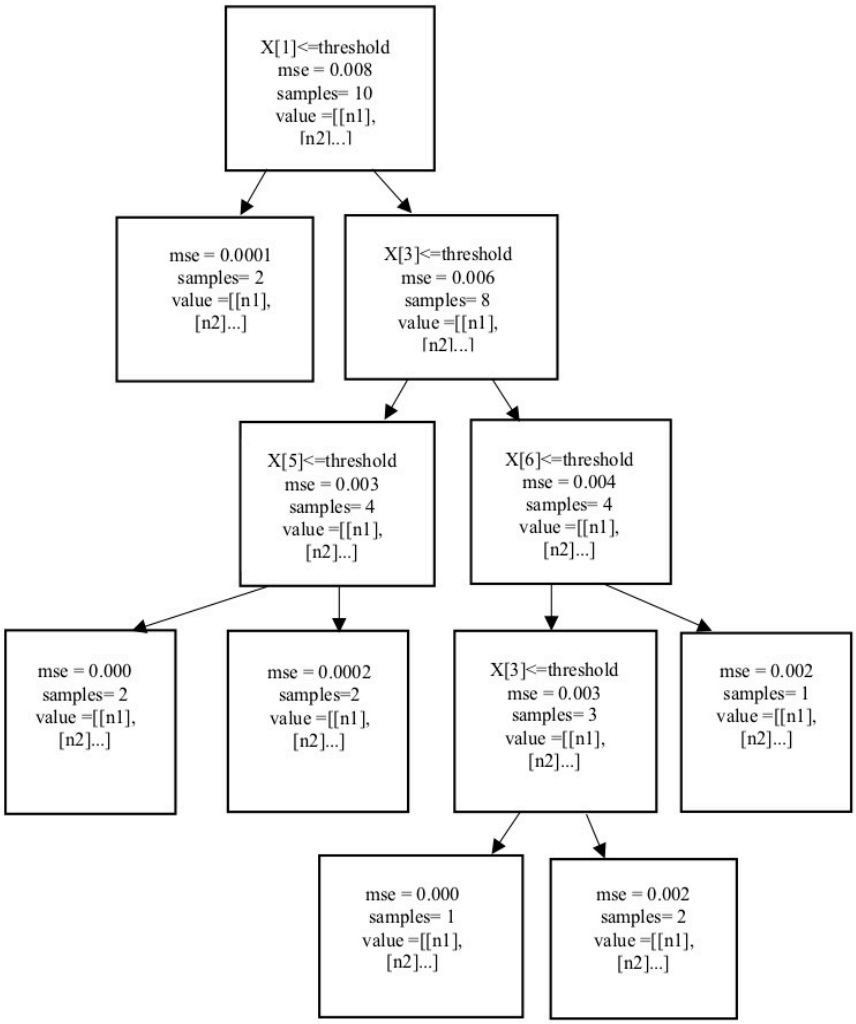



Figure 3.1 A schematic example of a decision tree model

### 3.2.2 The random forest as a regressor

A random forest consists of several decision trees. Each decision tree in a random forest is trained on a different dataset taken from the actual dataset with a replacement. This method is called bagging or bootstrapping. Bagging combines the predictions of different models while their predictions are not correlated. Each tree randomly selects a set of features from the total feature in the training dataset, making them uncorrelated. The number of decision trees and features selected are the hyperparameters that need to be tuned to obtain optimal performance. All the decision trees produce constant values when testing the model with new data. The mean of all the outputs is the final output, which improves to limit overfitting. This process is shown in Figure 3.2. The result is more accurate than the decision trees (Drakos, 2019).

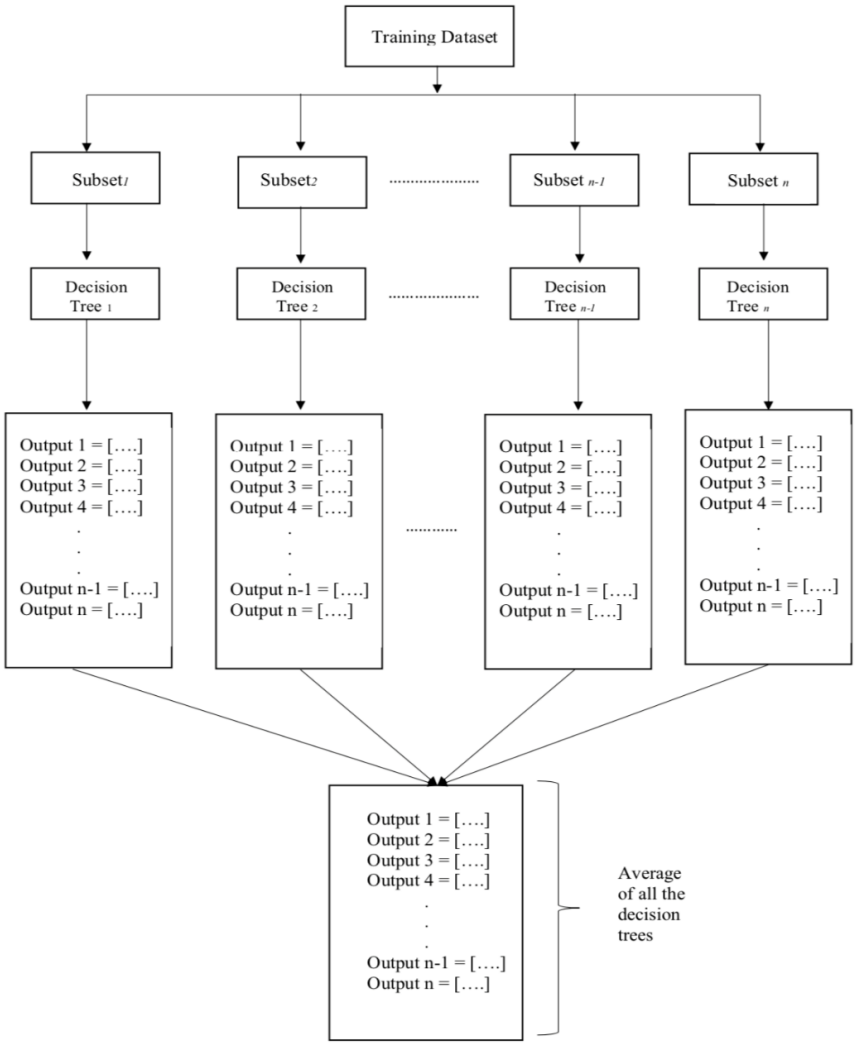



Figure 3.2 Random Forest as Multi-Output Regressor

Considering that the sample size of training data is '***N***', '***M***' variable and '***f***' variables are randomly selected at each decision node (Aldrich & Auret, 2010). The following steps are conducted:

1. Draw a bootstrap sample of size ***N*** from the training data for the number of trees considered.
2. Build each tree using the bootstrapped data by recursively selecting *f* variables and choosing a suitable variable to split the data into leaf nodes at the decision node. Repeat the process until the maximum depth size is reached.
3. Test the model with testing data, feed the testing data to all the trees built, and take the average of all the outputs as a result, i.e., the regression output can be expressed as

$$f(x) = \frac{1}{T}\sum_{n=1}^{T} DT_n(x) \tag{3.5}$$

where '*x'* is the test sample, *T* is the total number of trees and *DT* are the trees. In this study, acceleration responses from the selected four nodes are concatenated. Each response has 100 sample points making 400 points included in the concatenated response. This increases the dimension of the input to the random forest. PCA is applied to the concatenated response to reduce the input dimension. The number of principal components is selected to preserve 99% of the total variance of the data. The selected principal components are the inputs to the random forest regressor, and the output variables are the percentage reductions in stiffness parameters of structural elements.

### 3.3 Numerical Studies

This section presents numerical studies on a beam model to investigate the feasibility and accuracy of using the proposed approach based on the developed random forest technique as a regressor for elemental level structural damage quantification. A simply supported beam is used in this study as an example. The beam has ten elements of two meters each. It has a beam width of 0.6 meters and a height of 1 meter. Young's Modulus of $3.3\times10^4$ MPa is initially assumed for all the elements. The mass density and moment of inertia of the beam are 2500 kg/m$^3$ and 0.05 m$^4$, respectively. The first five natural frequencies of the undamaged beam model are 4.12 Hz, 16.48 Hz, 37.10 Hz, 66.04 Hz, and 103.425 Hz, respectively. Each node has two degrees of freedom, including one vertical translational displacement and one rotational displacement. In

the simulations below, only the acceleration response in the vertical deflection at randomly chosen four nodes are considered the measured responses.

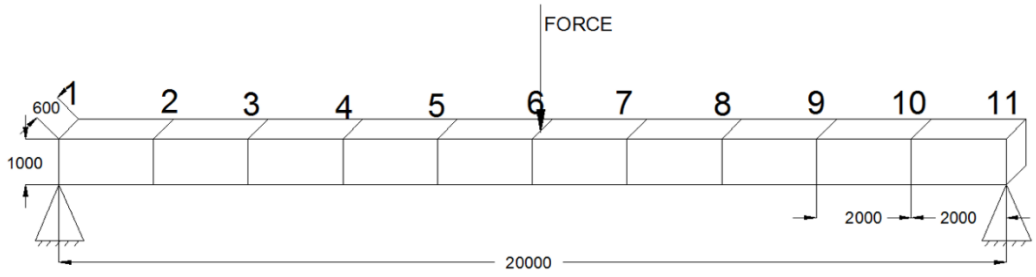



Figure 3.3 A simply supported beam model

### 3.3.1 Data Generation

The finite element model of the beam model is built for data generation. The beam is subjected to an impact force at node 6 in the vertical direction, as shown in Figure 3.3. The duration of the applied force is set as 0.1 seconds. Acceleration responses from selected nodes (2, 5, 8 and 9) are measured when an impact force is applied. Data are generated for both the damaged and undamaged cases for a second with a sampling rate of 100 Hz. For the damage cases, specific elements' stiffness parameters (i.e., Young's modulus) are reduced to reduce percentage stiffness.

The stiffness reduction of an element is taken as 30% if the stiffness parameter considered is 70% of the undamaged value for that element. The maximum stiffness reduction of 30% is used for simulating the damage in the elements for data generation. This has been used in existing studies for validating the accuracy of deep learning-based methods (Pathirage et al., 2018; Wang et al., 2018). Data are generated with 0% - 30% stiffness reduction of each element in steps on 1% increment. The dataset has '*n*' labelled outputs, where '*n*' is the number of elements, and the output represents the stiffness reduction.

For the multiple damage cases, damages in two and three elements are considered. The elements and stiffness parameters are randomly selected. Four samples are generated using different amplitudes for every damage case to consider the loading effect. Ideally, it is expected to have elements with the same stiffness parameters if all the elements are made from the same material. However, having the same stiffness parameter is impossible owing to several factors, such as inhomogeneity and different manufacturing quality. This will introduce some differences in the stiffness parameters

of structural elements, which are considered as uncertainty or modelling error. This study considers four cases; the first two cases are assumed with and without noise in the acceleration response and have no structural parameter uncertainty. The other two are with and without noise but considering uncertainty. These four cases are defined as:

*Case 1*: No uncertainties are considered in structural modelling, and no measurement noise is considered in acceleration responses.

*Case 2*: No uncertainties are considered in structural modelling, but measurement noise is considered in accelerations. White noise is added to the acceleration responses with a Signal-to-Noise Ratio (SNR) of 30dB and 20dB, respectively. No noise is added to the labelled output.

*Case 3*: Uncertainties are considered in structural modelling, but no measurement noise in acceleration responses. Uncertainties within the range of ± (1% – 3%) are included randomly in the stiffness parameters of structural elements to simulate the discrepancies in the finite element modelling.

*Case 4*: Uncertainties are considered in structural modelling and measurement noise in accelerations. Similar to Case 2, white noise is added to the acceleration responses in the datasets of Case 3 with an SNR of 30dB and 20dB, respectively.

Thirty-nine thousand five samples, including undamaged, single and multiple element damage cases, are generated from the beam model for the first two cases. One hundred thirty-five thousand five hundred samples are generated for the third and fourth cases. A sample is defined here as the concatenation of acceleration data measured from four nodes. Each sample has a dimension of 400 variables since response data from each node has 1s data with 100 sampling points per second. Each sample is generated with a randomly selected force from a force vector matrix with a varying magnitude following a normal distribution. The force is used only for data generation. It is not used as the input to the random forest. It is important to note that the impact force must be applied to the same node that is used for data generation to train and test the model for monitoring the structure to avoid the variations in structural responses associated with the loadings applied at different locations of the structure.

### 3.3.2 Data Pre-processing

The recorded data at the four nodes are concatenated and taken as input. PCA is performed on input variables to reduce the dimensionality since some of them can be irrelevant (Brownlee, 2019a; Esfandiari et al., 2020). It transforms the dataset into compressed form while retaining the essence of the original dataset (Pal, 2017), resulting in fewer uncorrelated variables (Malekzadeh & Catbas, 2016). Figure 3.4 shows the correlation between the 400 input variables, and Figure 3.5 shows the correlation of selected principal components. This demonstrates that the proposed approach is robust and provides good prediction results even with noisy measurements since the principal components of acceleration responses are used as the input to the developed algorithm.

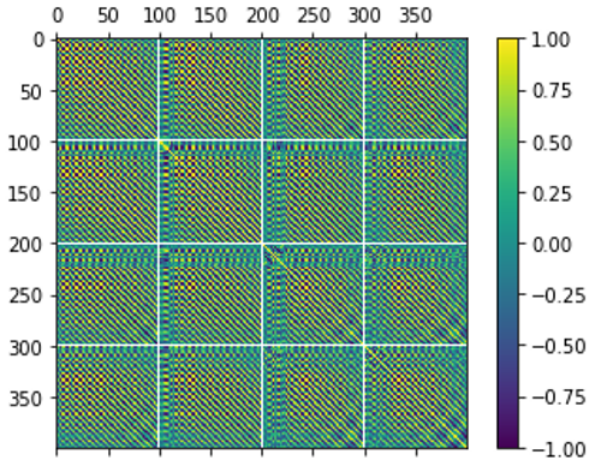



Figure 3.4 Correlation between the input variables before performing PCA

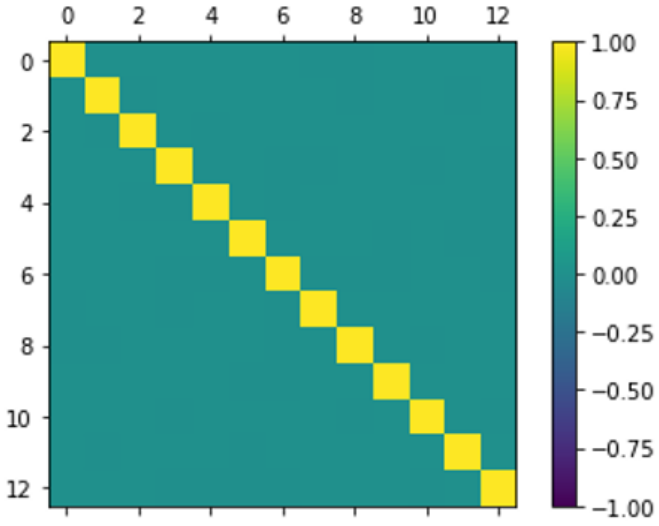



Figure 3.5 Correlation chart for the 13 principal components

Thirteen principal components are considered for the study since more than 99% of the total variance of the data is preserved, as seen in Figure 3.6. The thirteen principal components are used as input to the random forest. 85% of the dataset is used for training, and the remaining for testing the trained model. Figure 3.7 shows the variances of those 13 selected principal components for three sets of datasets, i.e., without noise, with 30dB and 20dB noises, respectively. It is observed that the variances of those components from the datasets are close to each other, except for the first component from the data with 20dB noise.

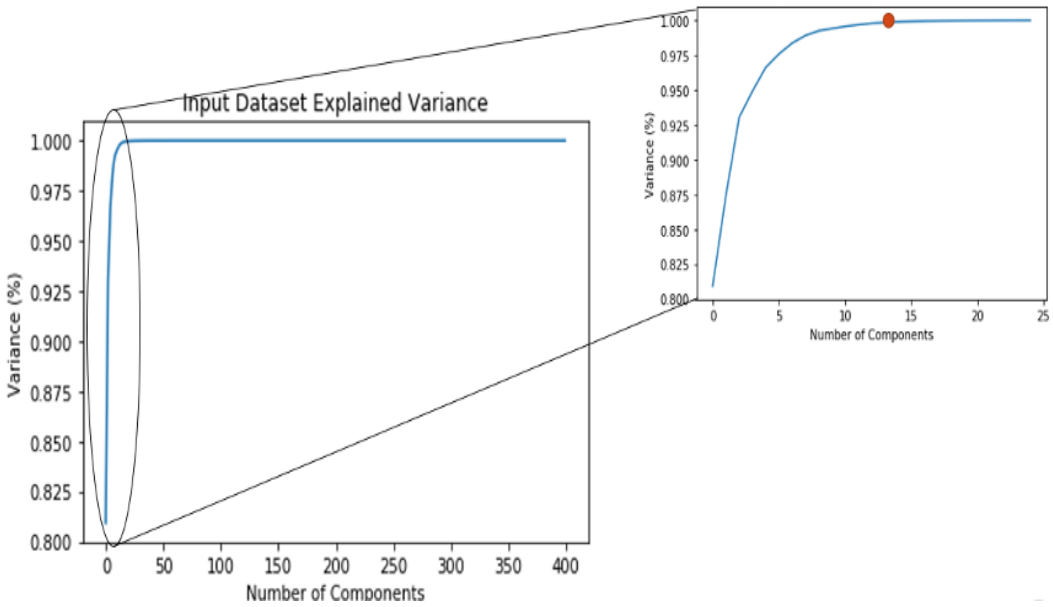



Figure 3.6 Variance plot of principal components

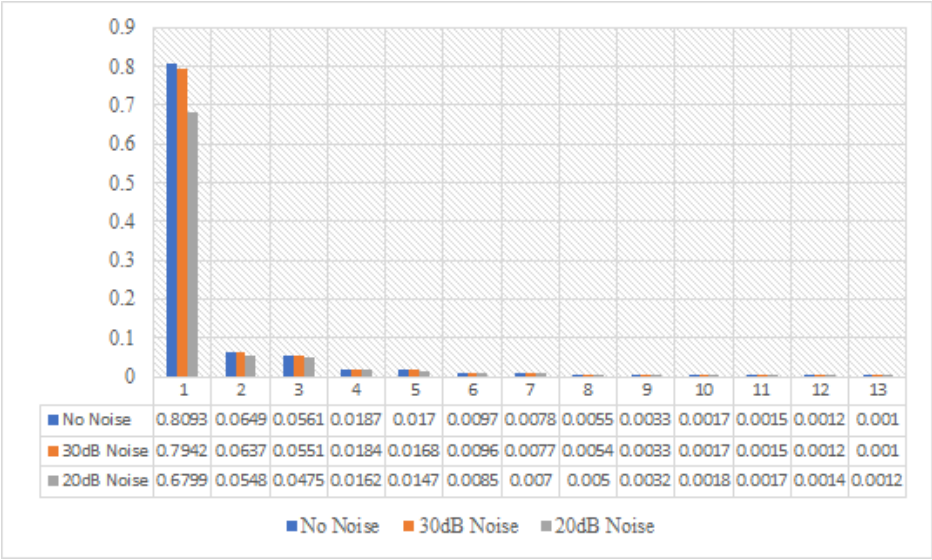



Figure 3.7 Variances of selected principal components of noise-free and noisy measurement

### 3.3.3 Results and discussions

Element level stiffness reductions are predicted using the proposed approach. MSE, R-squared and training time are used as performance metrics. R-squared provides the fitness of a set of predictions to the actual stiffness reductions (Brownlee, 2019c). The numbers of decision trees, the minimum number of samples required to split, and the minimum number of samples at a leaf node are tuned using the random search approach by Scikit-Learn tools (Koehrsen, 2018). In the random search approach, a framework of hyper-parameter values is set up, and arbitrary blends are selected to train the model. The performance of the proposed method is measured for both the beam model's major and minor stiffness reductions. Numerical computations are done on a system with Intel® Core (TM) i7-0750H, 16GB RAM and Nvidia RTX2070. The training time is also provided to demonstrate the efficiency of the developed approach.

The performance of the proposed approach is evaluated for the first two cases together. The splitting occurs when the number of samples in each leaf has ten or more samples, not allowing each leaf to have a single sample. This helps to limit overfitting. Table 3.1 shows MSE, R-squared and training time for the first two cases. It can be observed from Table 3.1 that the training is efficient. R-squared values are above 0.9, even when significant measurement noise is considered.

Table 3.1 MSE and R-squared for Cases 1 and 2

| Performance Metrics | Case 1 | Case 2 | |
|---|---|---|---|
| | | 30dB | 20dB |
| MSE | $3.8 \times 10^{-05}$ | $3.5 \times 10^{-04}$ | $5.5 \times 10^{-04}$ |
| R-Squared | 0.98 | 0.97 | 0.90 |
| Training Time (Seconds) | 13.77 | 13.99 | 13.84 |
| Sample size | | 39505 | |

The model performance is evaluated when uncertainties in the stiffness parameter are considered with and without measurement noise (i.e., Cases 3 and 4), as listed in Table 3.2. Very efficient training is achieved in less than 100 seconds, and the minimum regression value is 0.86. These results indicate that proper training is completed.

Table 3.2: MSE and R-values for Case 3 and 4.

| Performance Metrics | Case 3 | Case 4 | |
|---|---|---|---|
| | | 30dB | 20dB |
| MSE | $8.6 \times 10^{-05}$ | $3.8 \times 10^{-04}$ | $7.8 \times 10^{-04}$ |
| R-Squared | 0.98 | 0.93 | 0.86 |
| Training Time (Seconds) | 84.61 | 89.73 | 89.43 |
| Sample size | | 135500 | |

### *3.3.3.1 Single Element Damage Case*

The prediction results are measured for stiffness reduction less than and greater than 10%, defined in the study as minor and major damage cases, respectively. Figures 3.8 and 3.9 show the prediction results of two randomly selected samples from the testing dataset. The actual stiffness reduction is 24% at the 9th element for the major damage case in Figure 3.8 and 5% for the minor damage case at the 6th element in Figure 3.9. The trained random forest model provides excellent identification results with no noise in the input variable. Few false positives are observed in undamaged elements for the cases with measurement noise, but the magnitudes are less than 1% even for the minor damage case. The predicted stiffness reductions for major and minor damage cases are very close to the actual values.

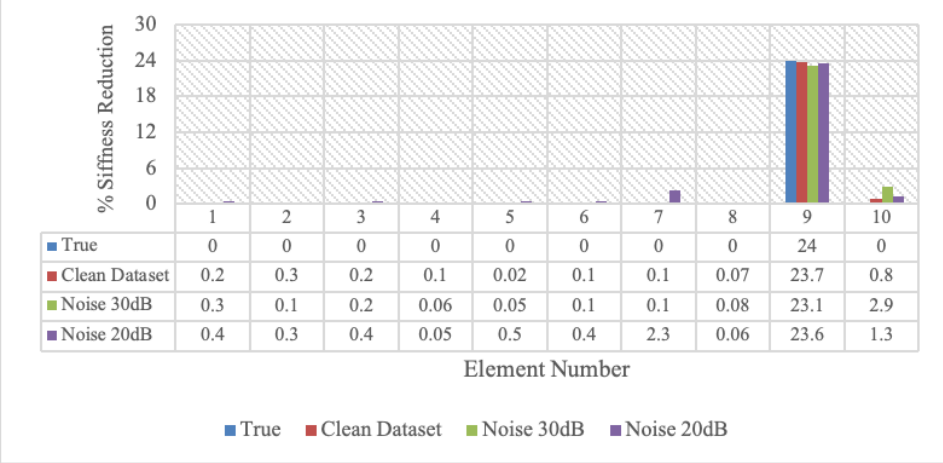



Figure 3.8 Damage quantification results of the major damage case

The accuracy level decreases with the increasing noise level, and false positives are observed in undamaged elements. However, with noise added at SNR of 20dB, the proposed method can still provide results close to actual damage for major damage

cases, although small false identification results are obtained. For the minor damage case, the location of the damage is identified accurately, but the severity is underestimated.

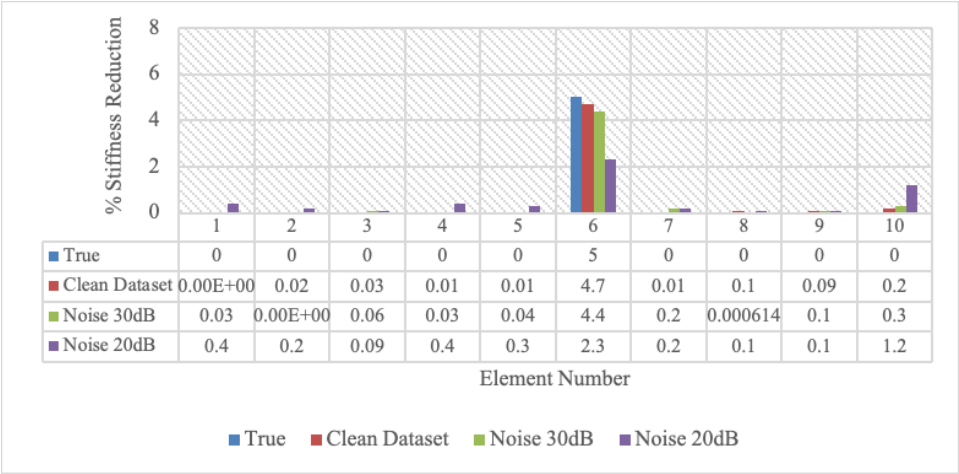



Figure 3.9 Damage quantification results of the minor damage case

Figures 3.10 and 3.11 show the quantification results for major and minor damage cases of 16% and 4% stiffness reductions in elements. No. 7 and 5, respectively, when both the uncertainty and measurement noise are considered. Robust results are obtained when uncertainties are considered only. For the case with 30dB noise, the identified results are promising. The accuracy degrades with the increasing noise level. For the case with a 20dB noise level, the identification results for the major damage case are not affected. However, only the damage location is accurately identified for the minor damage case. Its extent is underestimated due to the significant noise effect.

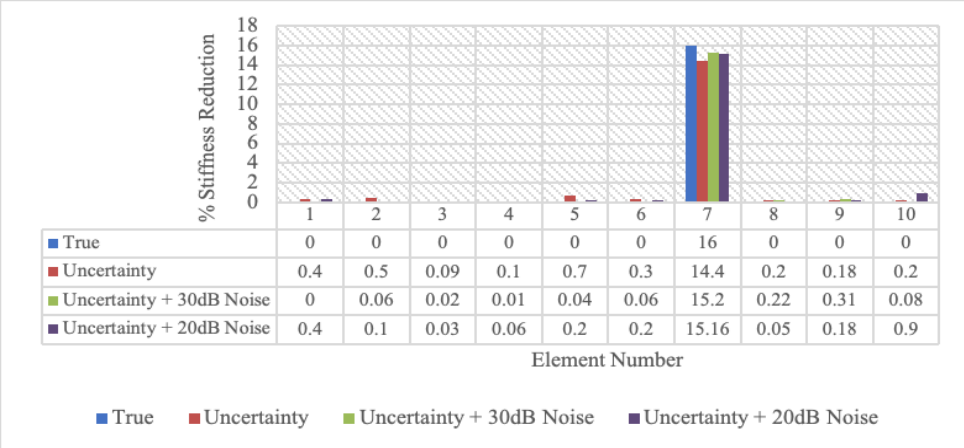



Figure 3.10 Damage quantification results for the major damage case with both measurement noise and uncertainty (modelling errors)

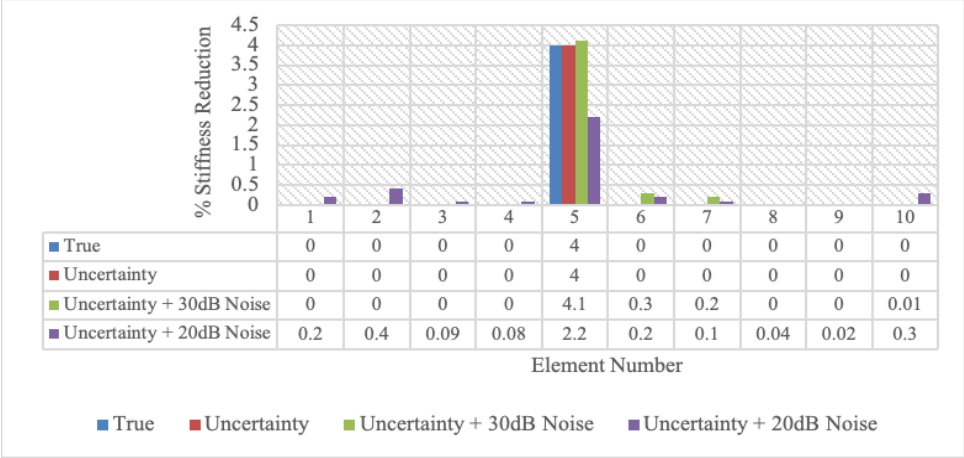



Figure 3. 11 Damage quantification results for the minor damage case with both measurement noise and uncertainty (modelling errors)

### *3.3.3.2 Multiple Damage Case*

For the multiple element damage case, the results obtained for both two-element and three-element damage cases are good when there is major damage in the elements. Figure 3.12 shows the damage identification result for the cases of two elements with major damages. The actual damages are 24% and 26% stiffness reductions in elements No. 5 and 9, respectively. The damage identification results for all the cases are good, with some insignificant false positive results observed in undamaged elements with the increase in noise level. Similar damage identification results are observed for the cases of three elements with major damages.

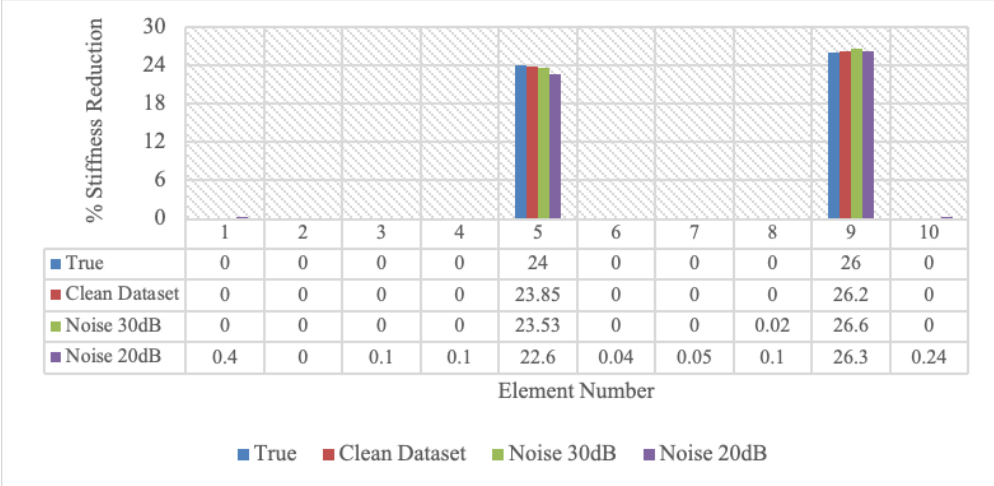
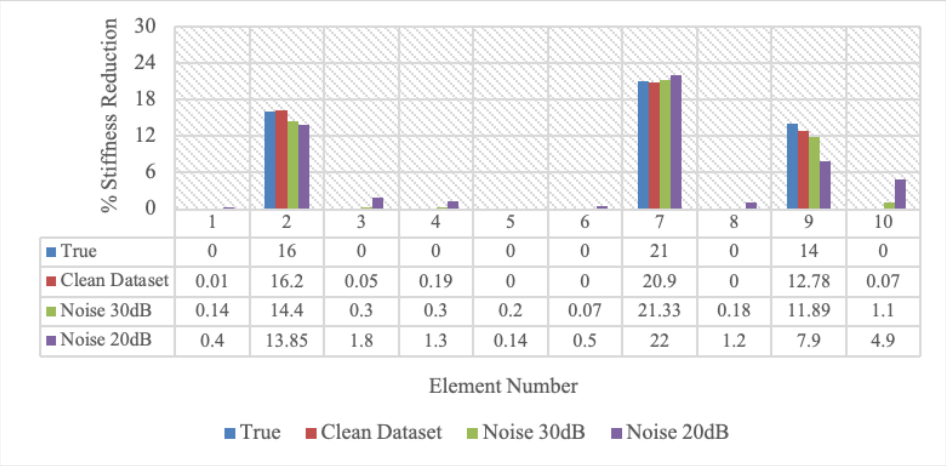



Figure 3.12 Two-element damage quantification results for the major damage case with measurement noise

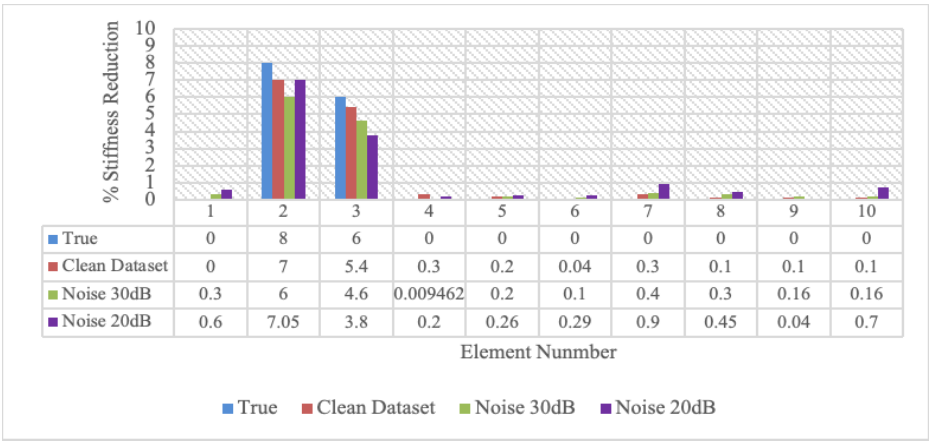
When the number of damaged elements increases to three, as shown in Figure 3.13, more false positive results are observed in undamaged elements and increase with the noise level. The stiffness reductions are 16.2%, 20.9% and 12.78% in elements No. 2, 7 and 9, respectively. The damage identification with 20 dB noise is not as good as the clean dataset and 30 dB noise case.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 0 | 16 | 0 | 0 | 0 | 0 | 21 | 0 | 14 | 0 |
| Clean Dataset | 0.01 | 16.2 | 0.05 | 0.19 | 0 | 0 | 20.9 | 0 | 12.78 | 0.07 |
| Noise 30dB | 0.14 | 14.4 | 0.3 | 0.3 | 0.2 | 0.07 | 21.33 | 0.18 | 11.89 | 1.1 |
| Noise 20dB | 0.4 | 13.85 | 1.8 | 1.3 | 0.14 | 0.5 | 22 | 1.2 | 7.9 | 4.9 |

Figure 3.13 Three-element damage quantification results for the major damage case with measurement noise

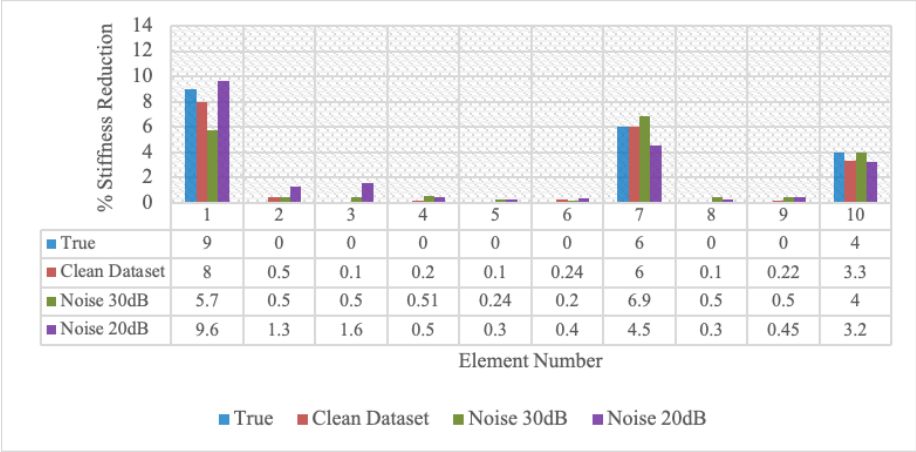
However, when the damage level in elements is less than 5%, the results are not as good as those with major damages. Figure 3.14 shows the results for the cases of two elements with minor damages. The stiffness reductions are 8% and 6% in elements No. 2 and 3, respectively. In all the cases, there are false positives in undamaged elements. With the increasing noise level, a higher level of false positive is observed.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 0 | 8 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Clean Dataset | 0 | 7 | 5.4 | 0.3 | 0.2 | 0.04 | 0.3 | 0.1 | 0.1 | 0.1 |
| Noise 30dB | 0.3 | 6 | 4.6 | 0.009462 | 0.2 | 0.1 | 0.4 | 0.3 | 0.16 | 0.16 |
| Noise 20dB | 0.6 | 7.05 | 3.8 | 0.2 | 0.26 | 0.29 | 0.9 | 0.45 | 0.04 | 0.7 |

Figure 3.14 Two-element damage quantification results for the minor damage case with measurement noise

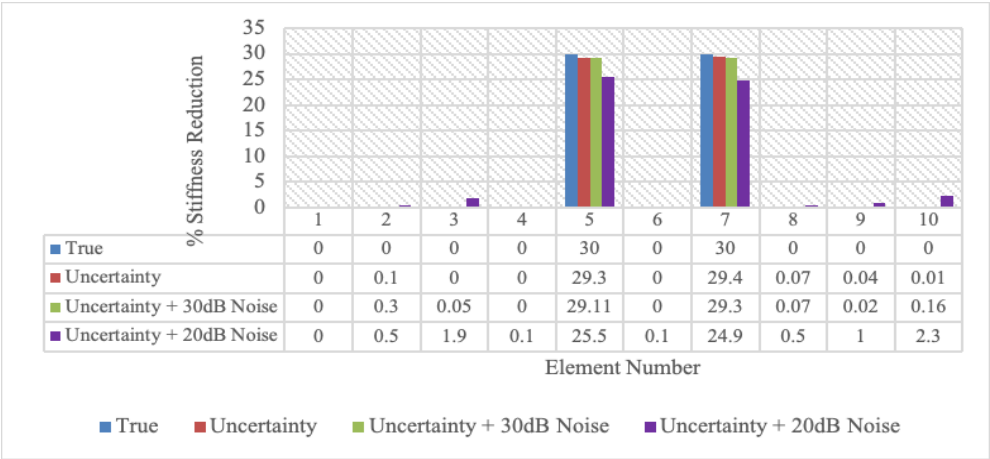
The problems seem more severe with the increasing number of damaged elements in the structure, as indicated in Figure 3.15. The three-element minor damage case has more false positives in the undamaged elements than the two-element minor damage case. Generally, good identification results are obtained in these cases.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 9 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 4 |
| Clean Dataset | 8 | 0.5 | 0.1 | 0.2 | 0.1 | 0.24 | 6 | 0.1 | 0.22 | 3.3 |
| Noise 30dB | 5.7 | 0.5 | 0.5 | 0.51 | 0.24 | 0.2 | 6.9 | 0.5 | 0.5 | 4 |
| Noise 20dB | 9.6 | 1.3 | 1.6 | 0.5 | 0.3 | 0.4 | 4.5 | 0.3 | 0.45 | 3.2 |

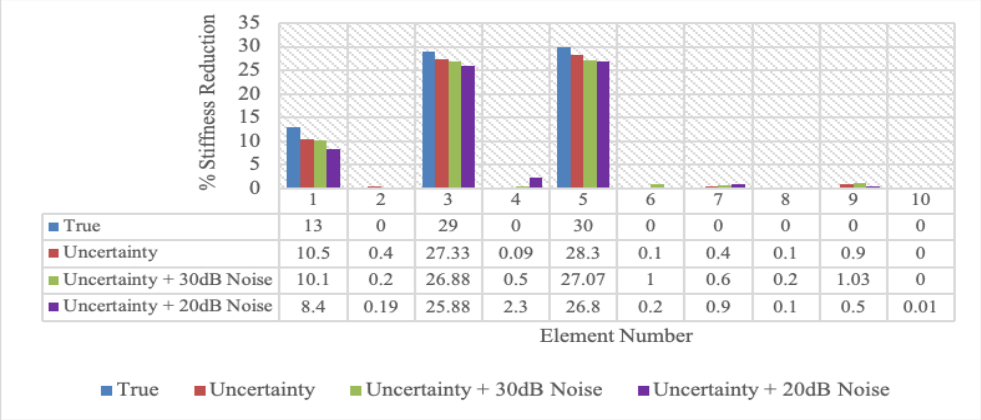
Figure 3.15 Three-element damage quantification results for the minor damage case with measurement noise

Figures 3.16-3.19 show the damage identification results for multiple element damage cases when uncertainty and noise are included. The resulting pattern obtained is similar to when only measurement noise is considered. The performance of the proposed approach degrades with the inclusion of uncertainty and noise. The increasing level of noise further degrades the performance. With uncertainties and measurement noise, damage locations and severities are identified with reasonable accuracy.



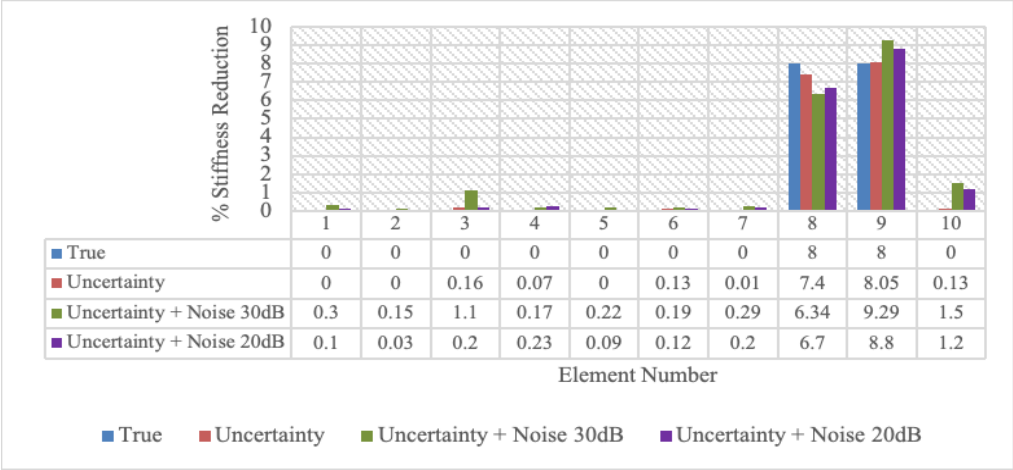

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 0 | 0 | 0 | 0 | 30 | 0 | 30 | 0 | 0 | 0 |
| Uncertainty | 0 | 0.1 | 0 | 0 | 29.3 | 0 | 29.4 | 0.07 | 0.04 | 0.01 |
| Uncertainty + 30dB Noise | 0 | 0.3 | 0.05 | 0 | 29.11 | 0 | 29.3 | 0.07 | 0.02 | 0.16 |
| Uncertainty + 20dB Noise | 0 | 0.5 | 1.9 | 0.1 | 25.5 | 0.1 | 24.9 | 0.5 | 1 | 2.3 |

Figure 3.16 Two element damage quantification results for the major damage case with both measurement noise and modelling errors



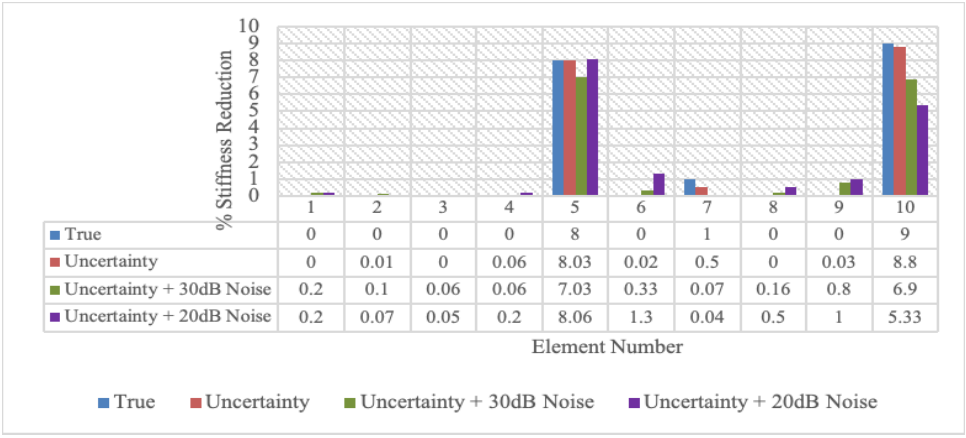

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 13 | 0 | 29 | 0 | 30 | 0 | 0 | 0 | 0 | 0 |
| Uncertainty | 10.5 | 0.4 | 27.33 | 0.09 | 28.3 | 0.1 | 0.4 | 0.1 | 0.9 | 0 |
| Uncertainty + 30dB Noise | 10.1 | 0.2 | 26.88 | 0.5 | 27.07 | 1 | 0.6 | 0.2 | 1.03 | 0 |
| Uncertainty + 20dB Noise | 8.4 | 0.19 | 25.88 | 2.3 | 26.8 | 0.2 | 0.9 | 0.1 | 0.5 | 0.01 |

Figure 3.17 Three-element damage quantification results for the major damage case with both measurement noise and modelling error




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8 | 8 | 0 |
| Uncertainty | 0 | 0 | 0.16 | 0.07 | 0 | 0.13 | 0.01 | 7.4 | 8.05 | 0.13 |
| Uncertainty + Noise 30dB | 0.3 | 0.15 | 1.1 | 0.17 | 0.22 | 0.19 | 0.29 | 6.34 | 9.29 | 1.5 |
| Uncertainty + Noise 20dB | 0.1 | 0.03 | 0.2 | 0.23 | 0.09 | 0.12 | 0.2 | 6.7 | 8.8 | 1.2 |

Figure 3.18 Two element to-element damage quantification results for the minor damage case with both measurement noise and modelling error




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| True | 0 | 0 | 0 | 0 | 8 | 0 | 1 | 0 | 0 | 9 |
| Uncertainty | 0 | 0.01 | 0 | 0.06 | 8.03 | 0.02 | 0.5 | 0 | 0.03 | 8.8 |
| Uncertainty + 30dB Noise | 0.2 | 0.1 | 0.06 | 0.06 | 7.03 | 0.33 | 0.07 | 0.16 | 0.8 | 6.9 |
| Uncertainty + 20dB Noise | 0.2 | 0.07 | 0.05 | 0.2 | 8.06 | 1.3 | 0.04 | 0.5 | 1 | 5.33 |

Figure 3.19 Three-element damage quantification results for the minor damage case with both measurement noise and modelling error

## 3.4 Experimental Validations

A seven-storey steel frame structure has been fabricated in the laboratory. The frame is 2.1 m tall, with each story being 0.3 m tall. The beam measures 0.5 m in length. The column and beam elements' cross-sections are measured at 49.98 mm × 4.85 mm and 49.89 mm × 8.92 mm, respectively. The column and beam elements' measured mass densities are 7850 kg/m$^3$ and 7734 kg/m$^3$, respectively. Initial Young's modulus for each member is assumed to be 210 GPa. At the top and bottom of the beam section, welds are used continuously to link the beam and column parts. To replicate the mass from the floor of a building structure, two pairs of mass blocks with a weight of about 4 kg each are installed at the quarter and three-quarters lengths of the beam in each storey. The detailed structure dimensions, experimental setup, sensors used, and placements for single and two-element damage scenarios can be found in a previous study (Li et al., 2012).

The finite element model of the entire frame structure is depicted in Figure 3.20. It has 70 planar frame elements and 65 nodes. The weights of the steel blocks are added as the concentrated masses at the relevant nodes of the finite element model. The system has 195 degrees of freedom (DOFs) in total, with two translational and a rotational displacement per node. Nodes 1 and 65 act as translational and rotational restrictions and are initially represented by a large stiffness of $3\times10^9$ N/m and $3\times10^9$ Nm/rad stiffness. Initial finite element model updating is conducted to reduce the difference between the analytical finite element model and the experimental model in the lab. A two-stage technique is used to carry out the model update process. The elastic modulus of each element and stiffness values of the restraints at the two supports are chosen as the parameters to be updated in the first stage of the model. Experimental modal analysis is used from the measured acceleration responses to obtain natural frequencies and mode shapes of the frame structure. The first seven natural frequencies and mode shape values are determined. The first-order modal sensitivity-based updating method minimises the discrepancy between the experimentally measured frequencies and mode shapes and those estimated from the analytical finite element model. The updating procedure uses the measured natural frequencies and mode shapes to update the 70 elastic modulus values and support stiffness values.

The second stage of model updating uses the dynamic response sensitivity method to refine the updated model based on the updated result. This stage of model updating is to as nearly as possible align the computed dynamic responses from the finite element model with the observed ones using measured available impact force. In the second stage of analysis, the elastic modulus of each frame element is updated. The updated finite element model minimises inconsistencies between the built finite element model and the experimental model. The detailed model updating process can be found in Li et al. (2012). The data generated from the updated finite element model are used for training and validating the random forest model. The measured data during the laboratory tests with introduced damage are used to test the accuracy and performance of the proposed approach for element-level structural damage quantification. For the single element damage case, the stiffness of the $12^{th}$ element is reduced by 12.5%. For the multiple damage case, 12.5% stiffness reductions are introduced into elements No. 6 and 12.

### 3.4.1 Data Generation and pre-processing

In this study, two damage cases are considered, i.e., single element damage and two-element damage cases. For each damage case, eight nodes on the structure are selected to measure the acceleration responses. Figure 3.20 shows the finite element model of the frame structure and sensor placement configurations for the single-element damage case and two-element damage case, respectively. The selected nodes are 7*x*, 9*x*, 11*x*, 17*x*, 47*x*, 50*x*, 53*x* and 56*x* for the single element damage case and 5*x*, 11*x*, 14*x*, 19*x*, 50*x*, 53*x*, 56*x* and 59*x* for the two-element damage case, where 9*x* denotes the acceleration response at node 9 along *the x* direction.

An impact force is randomly selected and applied at node 44*x* for every damage case considered when generating the training data, accounting for the possible variations in applying impact force in forced vibration tests. Figure 3.21 shows 1000 force vectors for the initial 0.025 seconds. Acceleration responses are measured from selected nodes at a sampling rate of 1000 Hz.

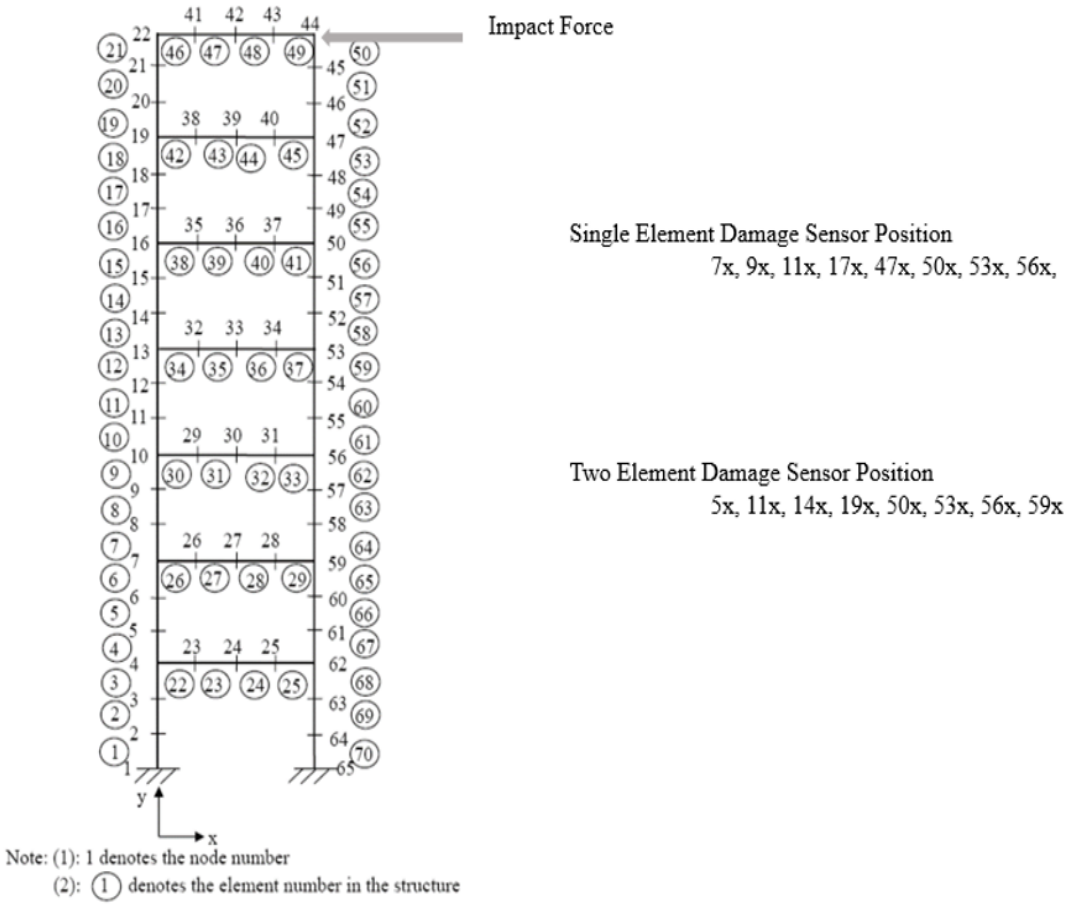



Figure 3. 20 The finite element model and sensor placement

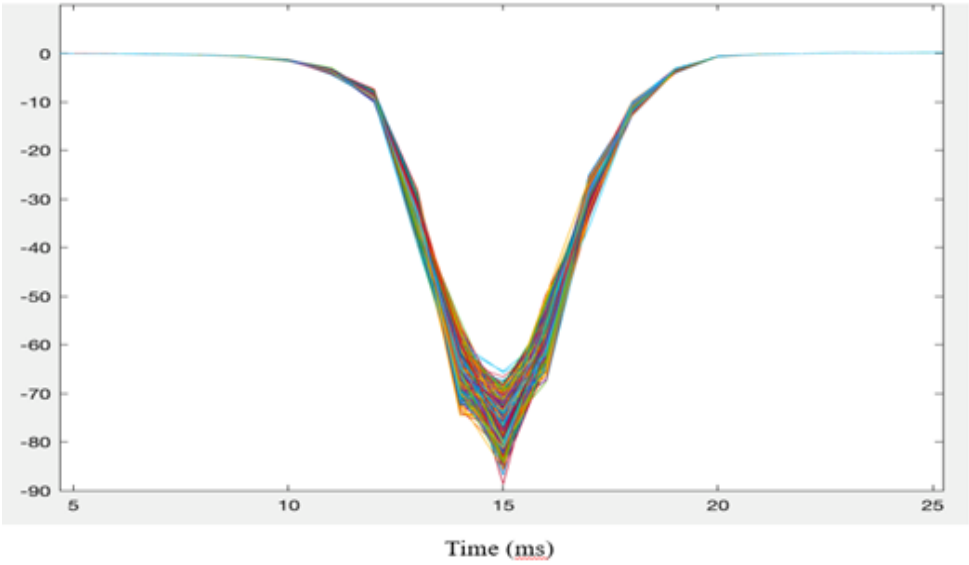



Figure 3.21 Applied random impact forces

For the single element damage case, samples are generated taking stiffness reduction in each element from 0% to 30% in an interval of 1%. Each sample is a concatenation

of acceleration responses from the selected nodes. Initially, 8400 samples of damage cases and four samples of undamaged cases are generated. Each sample is generated using a different impact force, i.e., with different amplitude and frequency to consider the loading effect. The duration of the applied impact force is changed by 1-2% of the measured test force to simulate the variance in the frequency of the applied force. For the second case, the stiffness is reduced in any two elements from element 1 to element 15 with a maximum stiffness reduction of 15%.

The data pre-processing technique remains the same as that in the numerical study. Acceleration responses measured for a damage case from the selected nodes are concatenated. For a damage case, the concatenated acceleration response has 4000 input dimensions since each response of 0.5 seconds has 500 sample points. The input dimension of 4000 is reduced to a lower dimension by applying PCA to analyse input. The number of principal components is selected to retain more than 99% variance. The sensor locations for single-element damage and two-element damage cases are different. The proposed approach is trained separately for the single-element and two-element cases for damage identification. Therefore, the PCA components for cases with two damages are different (29 for single damage case, 13 for two-element damage case). However, it is selected based on the principle of 99% variance in data. The selected principal components are used as input to the random forest. The model is trained with 80% of the datasets and validated with 20% of datasets generated from the updated finite element model. Acceleration data measured from the experimental tests of the damaged structure are finally used as the testing data to investigate whether the proposed approach can provide good identification results.

In the experimental study, the sensor positions are altered for single and multiple damage cases because the sensor setups are arbitrarily defined throughout the test. It should be emphasised that this positioning is not based on examining the optimal sensor placement. The positioning of sensors would affect the performance of the damage identification process. However, it is outside the scope of this study to analyse the optimal sensor placement for damage identification using the suggested methodology.

### 3.4.2 Results and discussions

The performance of the proposed approach is assessed using the first 0.5 seconds of acceleration responses as input. It is evaluated with different samples for the single element damage case. Hyperparameter tuning is performed using a random search approach. The numbers of decision trees, the minimum number of samples required to split, and the minimum number of samples at a leaf node are tuned. The optimal hyperparameters are obtained separately for the single and multiple damage cases. Deep learning methods (Pathirage et al., 2018; Wang et al., 2020) using mode shapes and frequencies demonstrated good identification results with relatively small data size. The training time of such methods requires 1-1.5 hours for each model training. However, the proposed method takes much less training time even though the data size is larger, as evidenced in Tables 3.3.

#### *3.4.2.1 Single Element Damage*

Figure 3.22 shows the damage identification results for the initial data size of 8804 samples. The identified stiffness reduction is 9.52 % at the $12^{th}$ element. Some false positives are observed in the undamaged elements. MSE and R-squared values on testing data are $6.98 \times 10^{-5}$ and 0.84%, respectively. The sample size is increased to investigate whether the performance can be improved. Figure 3.23 shows the identification results with 16804 samples. MSE and R-squared values are $5.46 \times 10^{-6}$ and 0.88%. The identified stiffness reduction at the element No. 12 is 9.54%. Some false positives of less than 2% are also observed in the undamaged elements. With the increase in the data size, MSE and R-squared values indicate that the results are improved. Furthermore, the sample size is increased to 23104. MSE and R-square values are $4.99 \times 10^{-5}$ and 89.2%, respectively. The identified stiffness reduction is 12.28% and is very close to the true stiffness reduction, as shown in Figure 3.24. Few and minimal false positives, less than 1%, are observed in undamaged elements. Table 3.3 summarises the obtained MSE and R-squared values and the training time for different training data sizes. These results indicate that the accuracy will be significantly improved when the data size is increased, and excellent identification results can be obtained. It is noted that only 45 seconds are required for the case with 23104 samples in the training datasets, which is significantly less than that of deep learning-based models as indicated in (Pathirage et al., 2018; Wang et al., 2020).

Table 3.3 Summary of Single Element Damage Cas

| Performance Metrics | Sample Size | | |
|---|---|---|---|
| | 8404 | 16805 | 23104 |
| MSE | 6.98 x 10 $^{-05}$ | 5.4 x 10$^{-04}$ | 4.99x 10$^{-04}$ |
| R-Squared | 0.84 | 0.88 | 0.89 |
| Training Time (Seconds) | 12.9 | 28.44 | 43.29 |

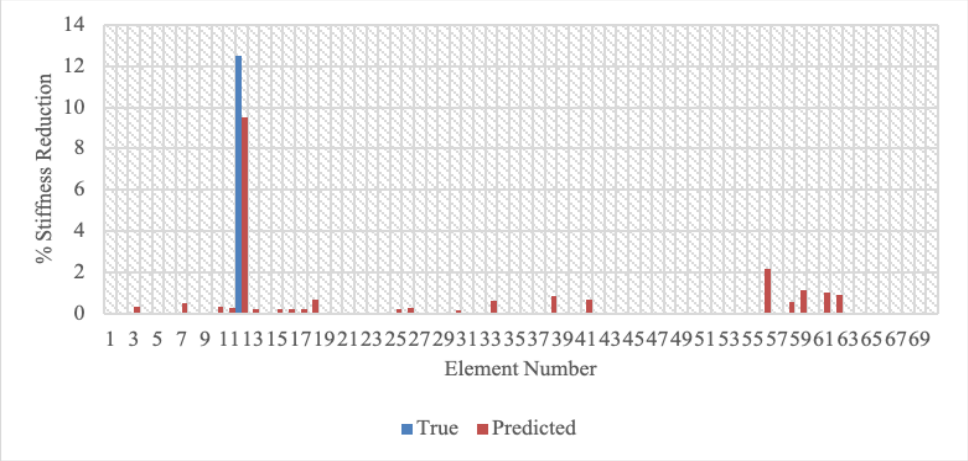



Figure 3.22 Damage identification results for the single damage case in the experimental test using 8404 samples

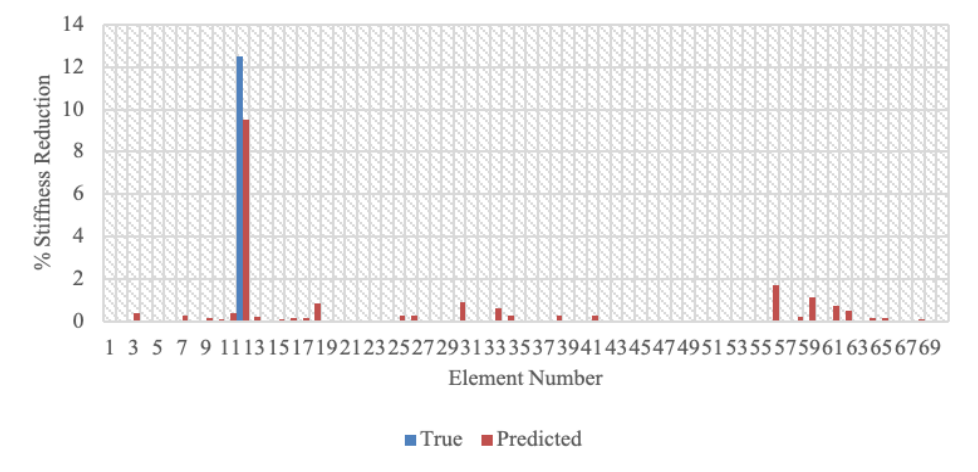



Figure 3.23 Damage identification results for the single damage case in the experimental test using 16804 samples

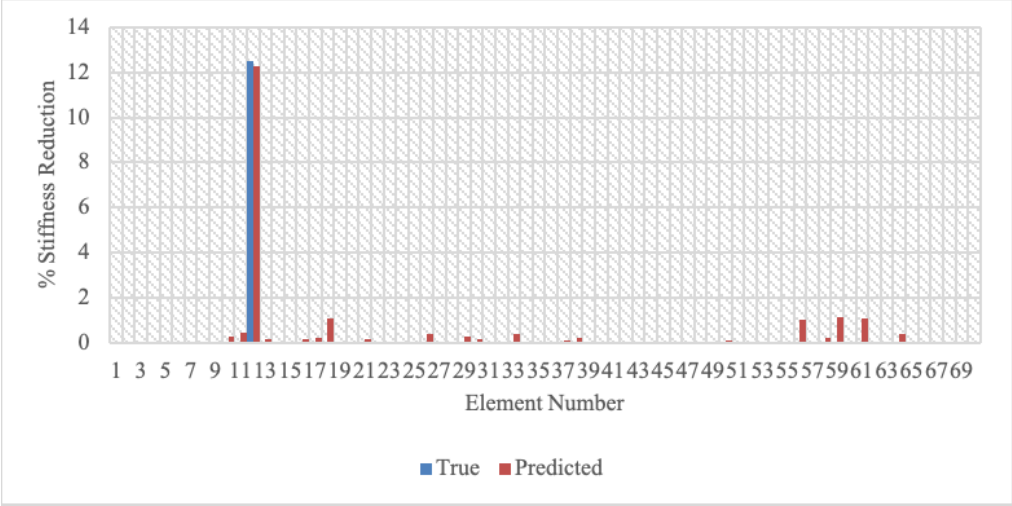



Figure 3.24 Damage identification results for the single damage case in the experimental test using 23104 samples

### *3.4.2.2 Two-Element Damage*

For the two-element damage case, 15440 samples are generated. The model is trained with 80% of the dataset and validated with the remaining dataset. The model is tested with a sample of measured data for the multiple damage case in elements No. 6 and 12 when the stiffness reduction in both elements is 12.5%. The identified stiffness reduction in the element No. 6 is 8.35%, and in the element, No. 12 is 11.83%, as shown in Figure 3.25. Some false positives are observed in the undamaged elements.

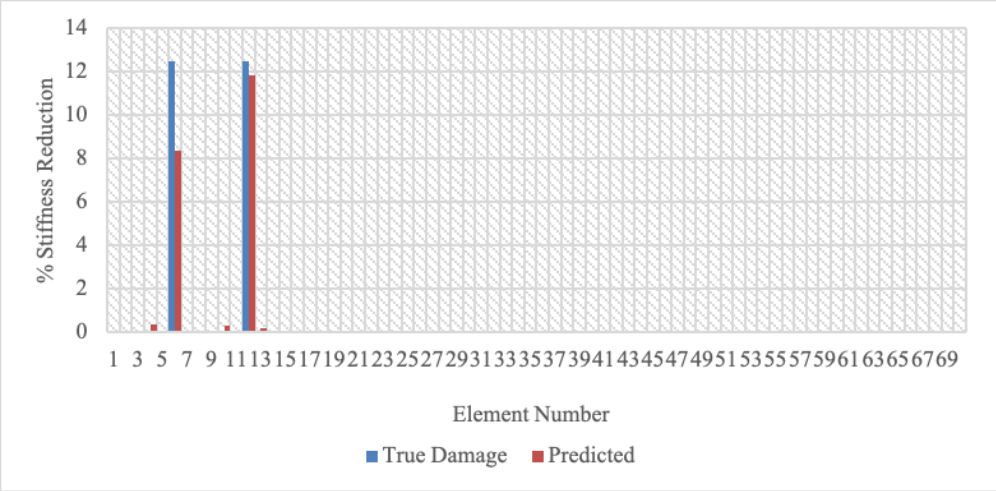



Figure 3.25 Damage identification results for two-element damage case in the experimental test

Generally, experimental data obtained good identification results for the multiple damage case. The results of the experimental study demonstrate that the proposed

approach using the random forest with PCA can locate the damage with satisfactory quantification of the damage level in the structure. MSE for the testing data is 5.04 x $10^{-05}$, and R-squared is 95.3%. The training time is 98.99 seconds.

## 3.5 Conclusion

This chapter presented an element-level structural damage quantification approach using the random forest technique with time domain acceleration responses. The ensemble machine learning technique provided better predictive performance. The proposed method based on the random forest technique can achieve a good correlation between structural damage and acceleration responses, hence good damage localisation and quantification. Measured acceleration responses from selected nodes are concatenated, and PCA is applied to have fewer uncorrelated input dimensions. It helped further to reduce the computational cost. Compared to the deep learning techniques for SHM, the proposed approach provides good damage identification with substantially less training time, as demonstrated in the numerical and experimental studies. Therefore, it is observed that the traditional ensemble-based machine learning models can be used for structural elemental damage identification and quantification. Computing responses like IRFs, which can be an ideal inherit vibration property and other fast decision tree-based ensemble machine learning models, may improve the damage prediction result of an existing problem. This is explored in the following study.

# CHAPTER 4

# 4 Damage Identification and Quantification Using Impulse Response Functions with Extremely Randomised Tree

## 4.1 Introduction

This chapter presents the development and application of another decision tree-based ensemble technique, extremely randomised tree (ERT) and using IRFs. The performance is measured and compared with RF as a multi-output regression model. Impulse response functions (IRFs) as structural vibration properties are extracted from measured acceleration response. In Chapter 3, the proposed method using acceleration response and RF is presented to overcome the limitations of existing structural damage quantification using modal information. In this method, IRFs are extracted from the acceleration responses and are processed to use as the input to ERT and RF. Moving averaging with a suitable window size is performed to reduce the effect of noise, and PCA is performed further for dimensionality reduction. The damage level is defined in terms of elemental stiffness reduction. Both numerical and experimental studies are conducted to investigate the capability of using the proposed approach for structural damage identification and quantification. Numerical studies are carried out on a simply supported beam and experimental validation on a steel frame structure in the laboratory.

## 4.2 Extremely randomised trees as a multi-output regressor

The proposed study in this chapter quantifies the damage in the structural elements in terms of stiffness reductions considering the same scenarios and single and multiple element damage cases in the previous chapter. The input dataset to the ERT is a 2d - array of shape *[N_samples, M_features]* where *N_sample* is the number of observations or instances for *N* damage cases and *M_features* is the number of principal components selected. Each observation is a vector with length *M_features*. The output dataset is also a 2d-array of shape *[N_samples, n_outputs]* and are the stiffness reductions in the structural elements. For each input observation, there will be an output vector of shape *[1, n_outputs]* where *n* is the number of elements in the structure and gives the values of stiffness reduction. For every observation, the model must predict stiffness reduction in *n* elements of the structure; therefore, the problem

is a multi-output problem. The output values are continuous numerical values, making it suitable for ERT to work as a regression model. Also, the output variables are likely correlated, and a single multioutput regressor is proposed resulting in a lower training time and reducing the overfitting (Chencho et al., 2020). Tree algorithm, Classification and Regression Trees (CART) are used in this study to build the ERT as it supports numerical target values. ERT and RF are both grown with multiple decision trees. A decision tree-based multioutput regressor recursively partitions the samples until the same output values are grouped, or the maximum allowable depth is reached.

Considering that here are *N*_samples with *M*_features and *n*_outputs at the root node of the decision tree. The initial dataset is the root node of the decision tree. A decision tree grows recursively, splitting the nodes into the child nodes. The splitting procedure is stated below in steps and is illustrated in Figure 4.1.

Step1: Select the suitable metric for the regression model for the best split. Variance measurement using mean square error (MSE) is selected for the proposed study. Calculate variance at the root node using equation 4.1.

$$V_{MSE} = \frac{1}{N\_samples} \sum_{y \in N\_sample} (y - y_d)^2 \tag{4.1}$$

$$y_d = \frac{1}{N\_samples} \sum_{y \in N\_sample} y \tag{4.2}$$

where $y$ is the output and $y_d$ is the predicted mean value.

Step 2: Select a random feature from a random subset of *M_features* and split the root node into child nodes. Calculate the variance of each child node and take the weighted average of each node for the calculation of a split variance.

Step 3: For RF, calculate variances for each feature in *M_features* and take the weighted average of each split. Select the feature which maximises variance reduction after the split.

Step 4: Repeat steps 1-4 until the child nodes contain an observation or reach the maximum defined depth.

The tree size can be controlled by selecting values of parameters, such as the maximum depth of the tree, the minimum number of samples required to split at a node, and the minimum number of samples needed for a leaf node. To illustrate how a decision tree is built, Figure 4.1 shows an example of building a decision tree for a multi-output

regression problem. For the given an example, the minimum number of samples at a leaf is set to three.

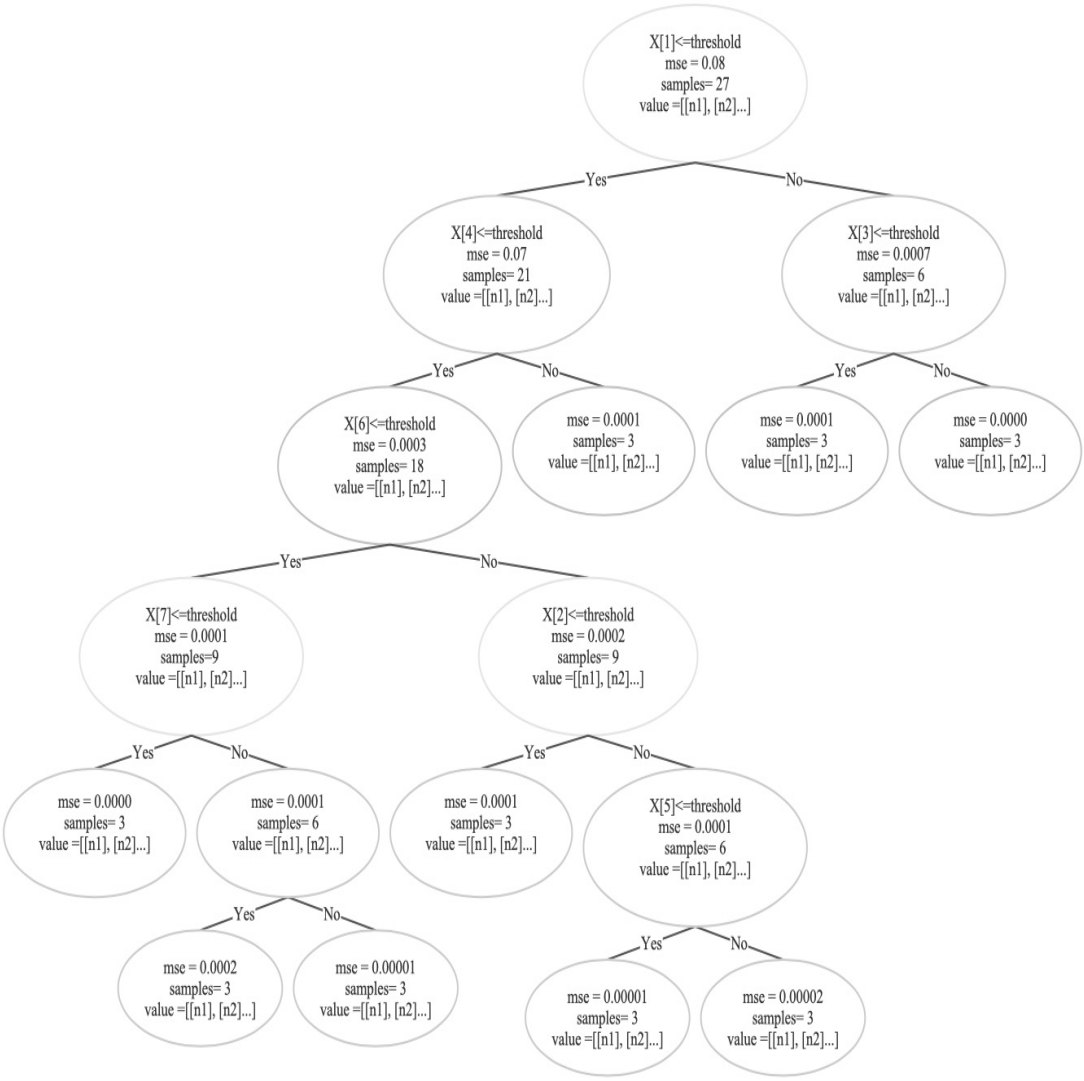



Figure 4.1 An example of a decision tree for a regression problem

Ensembles of decision trees are used to build RF and ERT. Each decision tree in RF is built on different bootstrapped datasets (Breiman, 1996; Chencho et al., 2020; Geurts et al., 2006), while ERT uses initial datasets to grow each tree. The mean of all the outputs of decision trees is the final output. The number of decision trees in both RF and ERT is one of the hyperparameters that can be tuned to obtain good results. Figures 4.2 and 4.3 show the processes of growing RF and ERT, respectively. Their algorithms are shown in Algorithms 1 and 2, respectively.

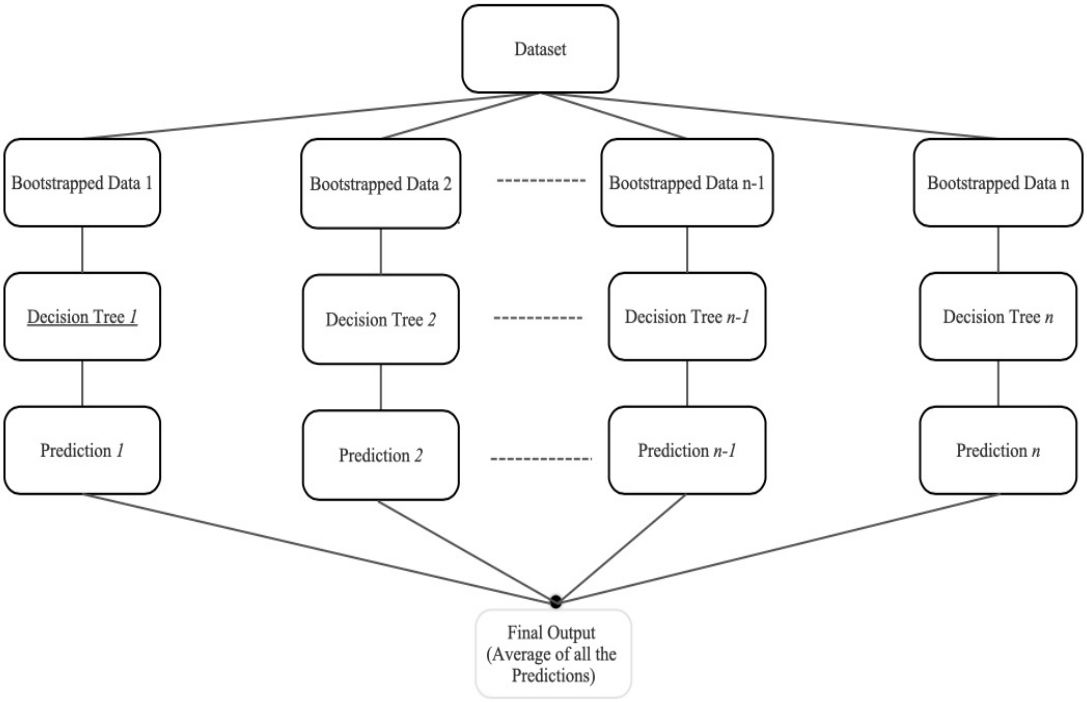



Figure 4.2 A schematic of a random forest

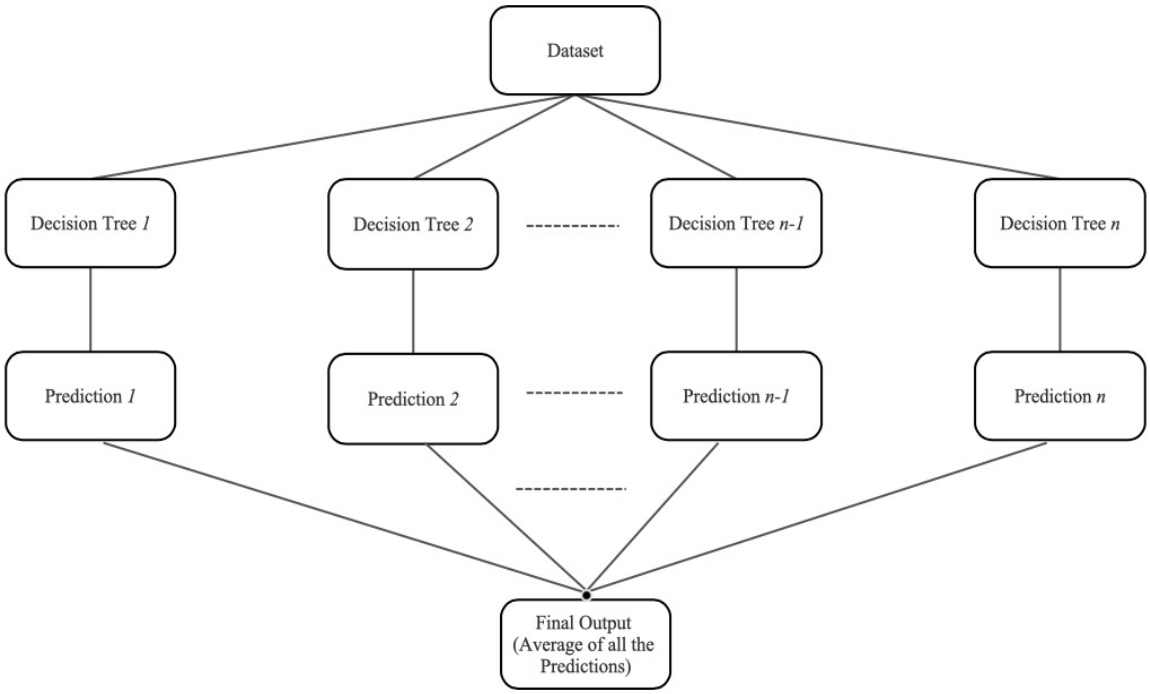



Figure 4.3. A schematic example of an ERT mode

**Algorithm 1: Random Forest**

S: Training Dataset

M: Features

T: Total number of trees

$f \subseteq M$

$S_i$ : bootstrapped data for $i = 1 \ldots T$

$H_i$: Prediction of the ith tree

H: RF Output

**Train_RF (S, F)**

for *i = 1,2… T,*

draw $S_i$ from *S*:

at each node:

randomly select *f*

split on best feature in *f*

if $|S_i| < n_{min}$, stop split

**Test_RF (Test Data, F)**

for all *i=1, 2,…,T*

Input: Test_Data

Output: $H_i$

**RF output H**

$$H = \frac{1}{T}\sum_{i=1}^{T} H_i$$

**Algorithm 2: Extremely Random Forest**

S: Training Dataset

M: Features

T: Total number of trees

$f \subseteq M$

$Hi$: Prediction of the ith tree

H: ERT Output

**Train_ERT (S, F)**

for $i = 1,2 \ldots T,$

at each node:

randomly select $f$

split on a random feature in $f$

if $|S| < n_{min}$, stop split

**Test_ERT (Test Data, F)**

for all $i=1,2,.,T$

Input: Test_Data

Output: $H_i$

**ERT output H**

$$H = \frac{1}{T}\sum_{i=1}^{T} H_i$$

## 4.3 Impulse response function

Impulse response function (IRF) is one of the dynamic properties of structures and can be extracted from time domain responses. IRF is an inherent system property, therefore it is more advantageous than using the measured time domain response. (Li et al., 2015). In Refs. (Li & Law, 2008; Robertson et al., 1998), the estimation of IRFs using discrete wavelet transform has been discussed. It can be analytically derived from the general equation of motion (Law & Li, 2007; Li et al., 2015) and is summarised here briefly. For a structure under a unit excitation, the structural equilibrium equation can be written as

$$[M]\{\ddot{x}(t)\} + [C]\{\dot{x}(t)\} + [K\{x(t)\}] = D\delta(t) \tag{4.3}$$

where $\delta(t)$ is the Dirac delta function. IRF is represented as a free vibration state with specific initial conditions. Considering that the initial condition is in static equilibrium, the unit IRF is computed from the equation of motion using the Newmark $-\beta$ method as

$$\begin{cases} [M]\ddot{h}(t) + [C]\dot{h}(t) + [K]h(t) = 0 \\ \quad h(0) = 0, \dot{h}(0) = M^{-1}D \end{cases} \tag{4.4}$$

where $h(t)$, $\dot{h}(t)$and $\ddot{h}(t)$are the unit IRFs for displacement, velocity, and acceleration responses in the time domain, respectively. For a structural system with zero initial conditions under the general external excitation $f(t)$, the acceleration response $\ddot{x}_l(k)$ at location *l* at time instant *k* can be expressed as

$$\ddot{x}_l(k) = \int_0^k \ddot{h}_l(\tau) f(t-\tau) d\tau \tag{4.5}$$

where $\ddot{h}_l(t)$ is the temporal unit IRF at location $l$. The discrete form of Equation (4.5) can be written as

$$\ddot{x}_l(k) = \sum_{i=0}^{k} \ddot{h}_l(i) f(k-i) \tag{4.6}$$

The matrix multiplication for the entire time domain response at location $l$ can be written as

$$X = H \cdot F \tag{4.7}$$

where *X* is the output response vector, *H* is the IRF vector, and *F* is the input force matrix.

Therefore, IRF can be extracted by solving Equation (4.7)

$$H = X \cdot (F^T \cdot F)^{-1} \cdot F^T \tag{4.8}$$

where $F^T$ is the transpose of the force matrix. *X*, *H* and *F* are expressed as

$$X = \{\ddot{x}_l(0), \ddot{x}_l(1), .., \ddot{x}_l(m-1), \ddot{x}_l(m)\}^T \tag{4.9}$$

$$H = \left\{\ddot{h}_l(0), \ddot{h}_l(1), .., \ddot{h}_l(m-1), \ddot{h}_l(m)\right\}^T \tag{4.10}$$

$$F = \begin{bmatrix} f(0) & f(1) & \cdots & f(m-1) & f(m) \\ 0 & f(0) & \cdots & f(m-2) & f(m-1) \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & f(0) \end{bmatrix} \tag{4.11}$$

in which $m$ is the number of sampled data in the acceleration response. For Equation (4.8), it should be noted that the pseudo-inverse is used to extract the impulse response function. Normally the condition number for the excitation force matrix *F* is not a large value since the columns are independent. However, when the matrix *F* is badly ill-conditioned, the truncated Singular Value Decomposition (TSVD) can be employed to eliminate those minimal singular values and the corresponding vectors to have a better and more stable solution for the pseudo-inverse. A more detailed explanation of the solution for obtaining IRF can be found in (Juang, 1994).

The IRF is extracted from the acceleration responses generated from the finite element model for numerical and experimental studies using the impact force and Equation (4.8). The detailed acceleration response generation is given in the following sections of numerical and experimental studies. The error in identifying IRF is reduced by using an ensemble of 50 and averaging the results, as explained in the study by Li et al. (2015). IRFs extracted from the acceleration responses from different locations are concatenated and moving average with a selected window size is performed to remove high-frequency random fluctuations in the data. The window size gives the number of observations used to calculate the moving average value. For a window width of seven, the trailing moving average is calculated as:

$$ma(t) = \frac{1}{7}(ob(t-6), ob(t-5), ob(t-4), ob(t-3), ob(t-2), ob(t-1), ob(t)) \tag{4.12}$$

The new series comprises the average of IRFs observations in the concatenated IRFs. PCA is then performed on the new series for dimensionality reduction. The number of principal components is selected to retain most information in the new series. The selected principal components are used as the input to ERT.

## 4.4 Numerical Studies

A simply supported beam with material properties of concreted structure is used for numerical studies to validate the accuracy of the proposed approach. The beam is 20 metres long, 0.6 metres in width and 1 metre1-metre height. The mass density is 2500

kg/m$^3$. Ten elements are used in the finite element model of the beam structure, as shown in Figure 4.4. Young's Modulus of 3.3 x $10^4$ MPa is considered for modelling the initial beam structure. There are eleven nodes, and each node has a vertical, translational and rotational degree of freedom. The first five natural frequencies are 4.12 Hz, 16.48 Hz, 37.10 Hz, 66.04 Hz and 103.43 Hz, respectively. This study uses vertical acceleration responses from randomly chosen five nodes, as shown in Figure 4.4. The acceleration responses are measured when an impact force is applied at node 3.

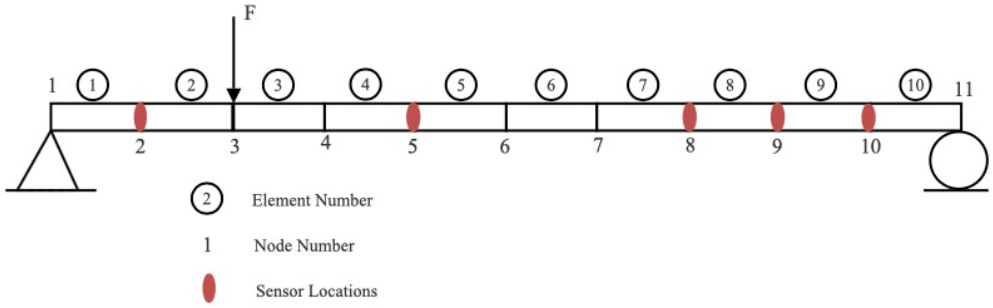



Figure 4.4. Simply supported beam and sensor locations

### 4.4.1 Data Generation and Pre-processing

To study the feasibility of the proposed method, datasets are generated from finite element model analysis. The acceleration responses are measured at a sampling rate of 100 Hz for one second when an impact force with a duration of 0.1 seconds is applied at node 3, as shown in Figure 4.4. The measurement duration of one second from a randomly selected sensor position is used for the study. Datasets are generated considering undamaged and damaged cases. Single element and multiple element damage cases are simulated. The stiffness parameters (Young's Modulus) are reduced to introduce damage in an element or various elements. The damage is taken as the percentage of stiffness reduction in specific elements. For the single element case, data are generated, taking a percentage of stiffness reduction from 0% to 20% in steps of 0.5 %. For multiple element damage cases, the maximum stiffness reduction is 15% in steps of 1.5% for two-element damage cases and 4.5% for three-element damage cases. For all the undamaged and damaged cases, there are *'n'* outputs, where '*n*' is the number of elements in the structure. For all the undamaged and damaged cases, acceleration responses from each of the selected nodes are measured using random

impact forces generated with a Gaussian distribution with a mean value of 8000 N and a standard deviation of 50.

The material of structural elements is the same. Still, it is not possible to have the exact Young's Modulus for all the elements in the structure owning to factors like inhomogeneity and manufacturing quality. There is inevitable uncertainty in the stiffness parameters of the structure. This is considered an error or uncertainty in system modelling in this study. Besides, the measured responses may contain noise. Therefore, to take care of noise measurement and uncertainty in the system modelling, acceleration responses are recorded with and without noise measurement for two cases: one considering system modelling error and the other without system modelling error. IRFs are extracted from the four cases of acceleration responses using Equation (4.8), and ensembles of 50 are averaged to reduce errors in IRFs. Figures 4.5, 4.6, and 4.7 show the acceleration responses of the undamaged beam measured at node 2, its extracted IRF and corresponding impact force, respectively.

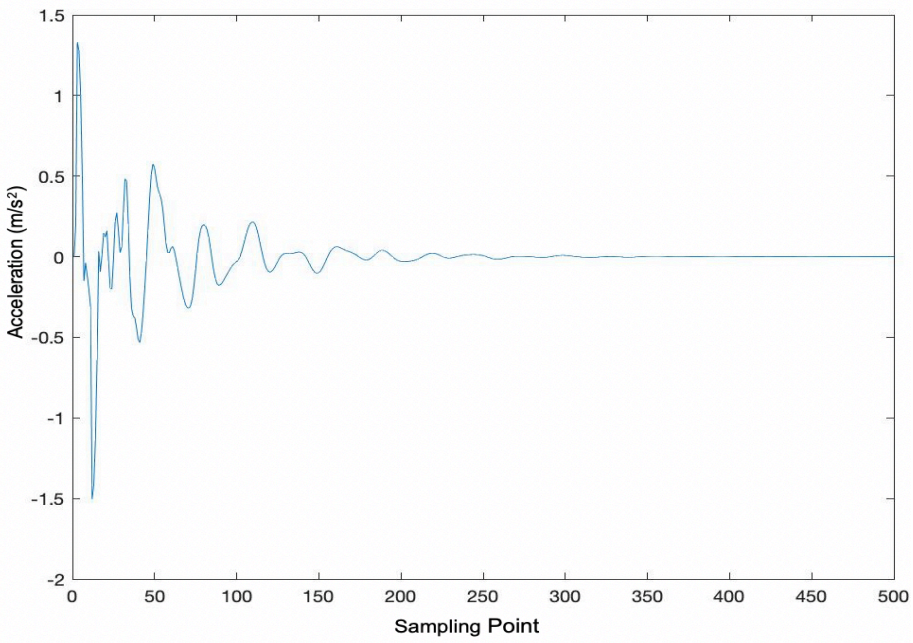



Figure 4.5 Acceleration response at Node 2

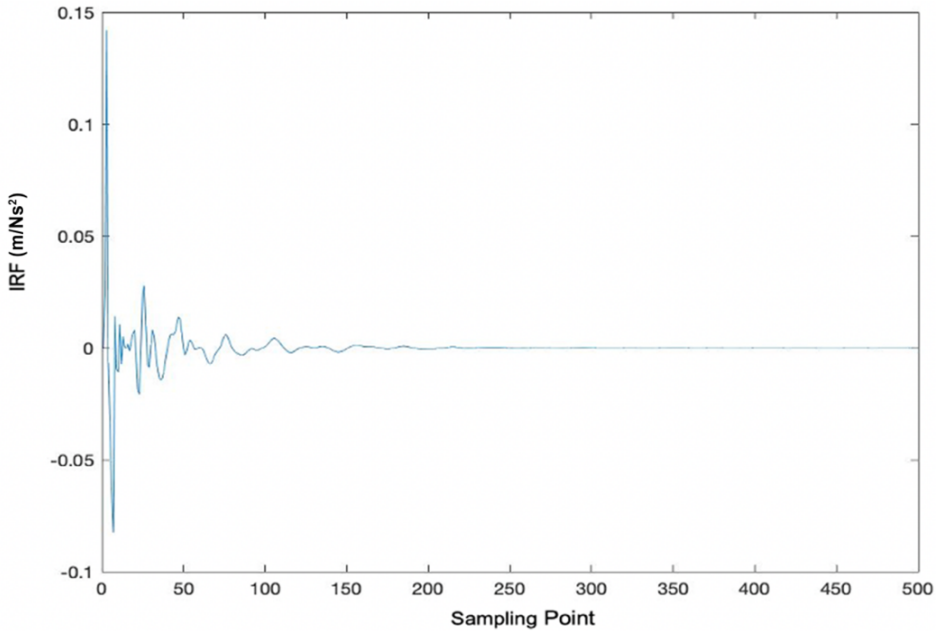



Figure 4.6 IRF at node 2

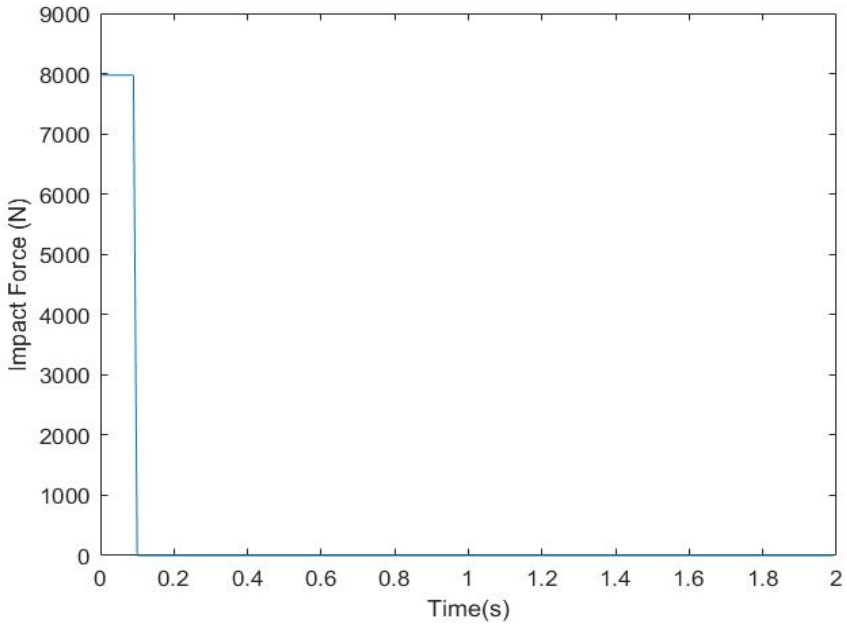



Figure 4.7 Impact Force

The error in IRFs for a particular damage case is severe when the acceleration responses contain noise. The effect of using 50 ensembles and the result of IRF taking averaging is shown in Figure 4.8. The IRF extracted from the clean acceleration response is shown in yellow colour. 5% white noise is added to the acceleration response, and IRF is extracted from it without averaging. It can be observed that the ensemble of 50 and averaging effectively reduces the error in the estimation of IRF.

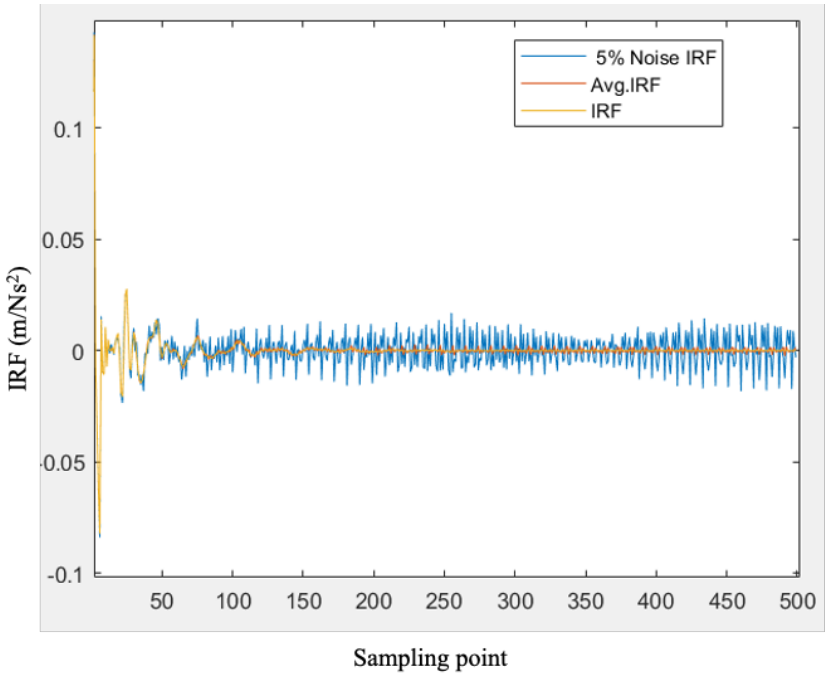



Figure 4.8 IRF obtained using ensemble averaging

This study considers four scenarios to evaluate the performance of the proposed approach for elemental damage quantification. These scenarios are defined as:

*Scenario 1:* IRFs are generated from the acceleration responses with no noise measurement and system modelling error.

*Scenario 2*: IRFs generated from the acceleration responses with noise measurement and no system modelling error. White noise is added to acceleration responses before extraction of IRFs, and then IRFs are extracted. Different levels of noise are considered. 5% and 10% noise levels are added to the input data to check the method's robustness to noise and uncertainties for structural condition monitoring. The same noise levels were also assumed in existing studies to investigate the noise effect on the performance of the damage identification methods.

*Scenario 3*: IRFs generated from the acceleration responses with no noise measurement and system modelling error. Uncertainty within the range of ± (1-3%) is included randomly in the stiffness parameters for the finite element modelling, and acceleration responses are then generated for obtaining IRF.

*Scenario 4*: Both uncertainty and measurement noise are considered. Uncertainty within the range of ± (1-3%) is included randomly in the stiffness

parameters for the finite element modelling and generation of acceleration responses. Then, 5% and 10% white noises are added to the acceleration responses, and IRFs are extracted.

For a damage case, averaged IRFs with 50 ensembles are extracted from acceleration responses measured at selected nodes. Each averaged IRF has 100 sampling points since measured acceleration response for a second is considered for the study. These IRFs are concatenated and labelled as the input for the damage case, and the element stiffness parameters are used as labelled output in the datasets. A total of 33902 samples are generated for undamaged, single element, two-element, and three-element damage cases. Trailing moving average defined by Equation (4.12) is applied to smooth the concatenated IRFs by removing random fluctuations. Figure 4.9 shows the original IRF extracted for a case in Scenario 1. The moving average is used, and a much smoother IRF result is obtained. The window size must be defined for the trailing moving average to get optimal performance. For the current study, a window size of seven is selected. The performance of the proposed method is evaluated for different window sizes, and the window size which gives optimal performance is determined. A larger window size can provide either the same or degraded performance.

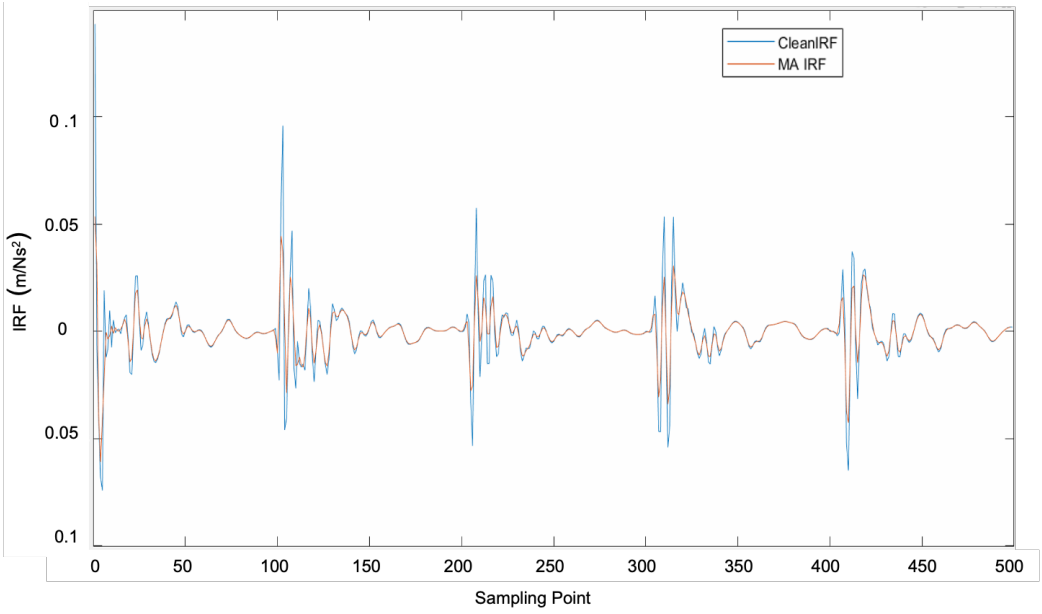



Figure 4.9 Concatenated IRF with and without performing moving averaging

Moreover, it can also help reduce the effect of noise. In Figure 4.10, the blue response is the averaged IRF extracted from the acceleration response with 10% noise. By applying the moving averaging, the effect of noise is reduced, and the obtained IRF

from noisy responses is very close to that obtained from responses without noise, indicating that the accuracy in getting IRF can be improved.

The concatenated IRF after performing moving averaging has 500 data points, bringing the proposed model's significant training demand. Therefore, PCA is applied to process the concatenated data to reduce the input data dimensionality. PCA is essential to remove irrelevant variables resulting in fewer uncorrelated variables (Brownlee, 2019a; Esfandiari et al., 2020; Malekzadeh & Catbas, 2016). The number of principal components is selected by preserving a minimum variance of 98.5% of the data. Figures 4.11(a), 4.11(b) and 4.11(c) show the total variance for 19 principal components when PCA was performed on the new series obtained after completing moving averaging on the clean dataset, 5% noise and 10% noise, respectively. It is noted that 98.71% of the variance can be preserved with 19 principal components with 10% noise added on acceleration response.

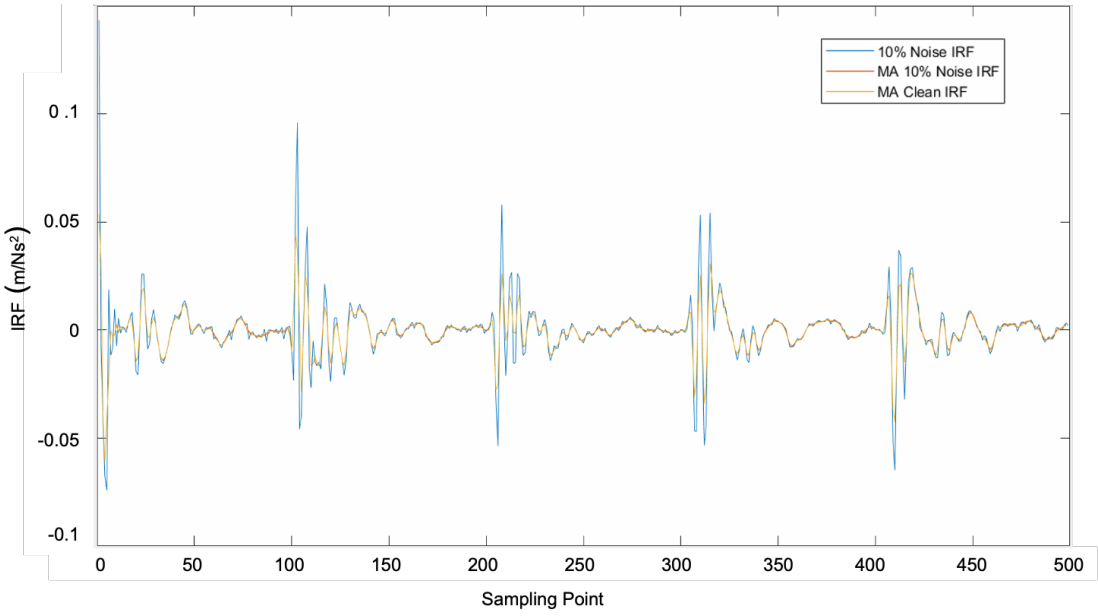
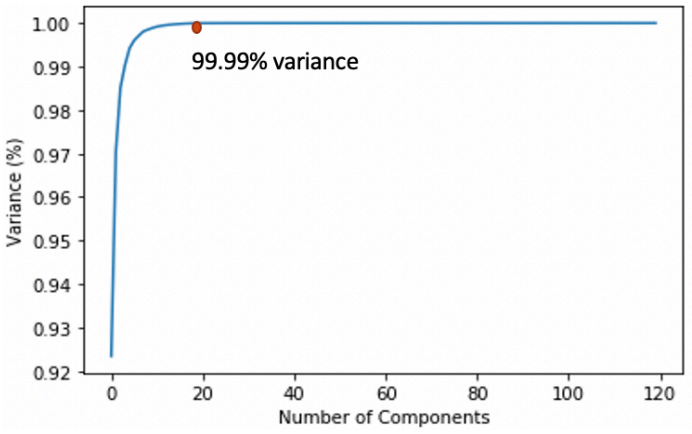
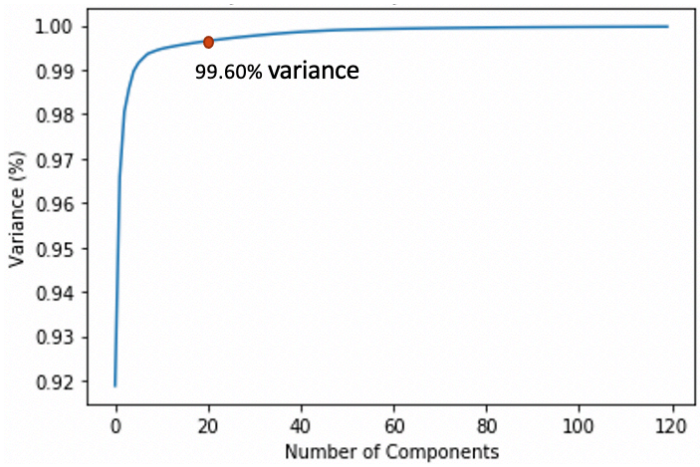


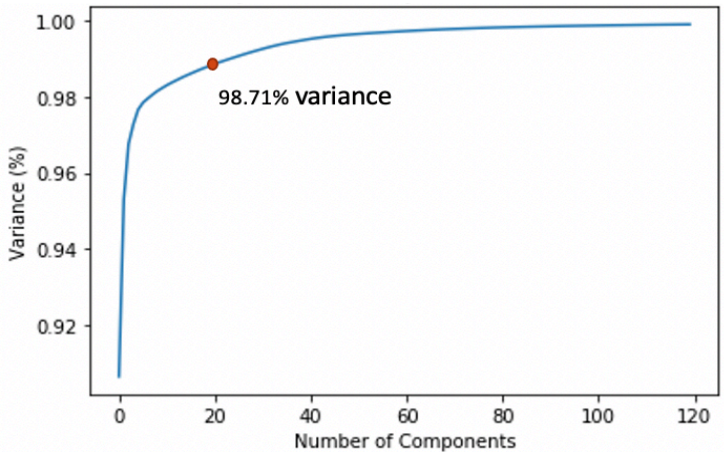

Figure 4.10 IRF extracted from noisy acceleration response with and without moving averaging.




(a)




(b)




(c)

Figure 4.11 Variance plot

### 4.4.2 Results and Discussions

These 19 selected principal components are input to the ERT, and the percentage of stiffness reduction (damage level) across the elements is used as target variables. The number of variables in the output depends on the number of elements in the structure. The structural beam model has ten elements, and the number of variables in the output will be ten. The first target variable indicates the percentage of stiffness reduction in the first element. Likewise, the $2^{nd}$, $3^{rd}$, and $10^{th}$ target variables denote the damage severities in elements No. 2, 3, and 10, respectively. The proposed approach is trained with 85% of the datasets and tested with the remaining 15%.

A system with Intel ® Core ™ i7-0750H, 16GB RAM and Nvidia RTX2070 is used for the study. Random search (Koehrsen, 2018) method by Scikit-Learn library is used for tunning the hyper-parameters. The number of decision trees, the minimum number of samples required to split the internal nodes and the minimum number of pieces in a leaf node are taken as the hyperparameters. For the proposed study, 120 decision trees with four samples are required at the internal node to split further, and two samples in the leaf node are selected. The performance of the proposed method is evaluated using mean squared error (MSE), R-squared and training time.

The MSE and R-squared are calculated between the predicted values and true damage severities. The training time is measured for the proposed learning method from the training dataset. The MSE, R-squared and training time are calculated for all four scenarios. Results obtained using the proposed approach based on ERT and IRFs are compared with those from RF with the same hyper-parameters. The results demonstrate that the proposed method can better identify damage locations and severities than RF. The results also indicate that the training time taken by ERT is less. Considering both uncertainty and measurement noise, the worst scenario among these four scenarios, the R-square value, MSE and training time obtained are 0.89, $1.39\times10^{-4}$ and 15.3 seconds, respectively, for ERT and MSE of $1.79 \times 10^{-4}$ and R-squared of 0.86 for RF. The detailed damage identification results for all four scenarios for single and multiple element damage cases are shown below.

The performance of the proposed method is evaluated for the four scenarios on the testing data. Table 4.1 shows the performance measurement of Scenario 1. It is

observed that both RF and ERT can identify the damage quite well when no noise and system modelling error is considered. The MSE and R-squared values are close, with ERT performing slightly better than RF, but ERT and RF significantly differ in training time. ERT takes less time to train or learn from the training dataset. For this study, ERT takes approximately 3.5 times less than RF. ERT gets trained faster as it saves time when splitting at the internode node. It randomly selects a feature from the subset of features to split rather than computing for all the features in the subset to find the feature for the best split.

Table 4.1 Performance measurement for Scenario

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE. | $3.6 \times 10^{-6}$ | $1.6 \times 10^{-5}$ |
| R-Squared | 0.998 | 0.99 |
| Training Time (Seconds) | 14.5 | 56.22 |

To investigate the performance with the noise measurement, different levels of white noise are added to the acceleration response without considering system modelling error. IRFs are extracted, processed, and used to quantify the elemental damage. The performance degrades with the increase in the noise level. For 5% noise, MSE and R-squared values are close to Scenario 1. However, with the rise in the noise level, MSE increases, and the R-squared value decreases by more than 2%, as seen in Table 4.2. As in Scenario 1, ERT performs slightly better than RF. The training time is not much affected for all the scenarios since the data size is the same for all the scenarios.

Table 4.2 Performance measurement for Scenario 2

| | 5% Noise | | 10% Noise | |
|---|---|---|---|---|
| Performance Metrics | ERT | RF | ERT | RF |
| MSE | $1.2\times10^{-5}$ | $3.3\times10^{-5}$ | $9.2\times10^{-5}$ | $1.4\times10^{-4}$ |
| R-Squared | 0.993 | 0.981 | 0.954 | 0.93 |
| Training Time (seconds) | 14.8 | 56.2 | 14.9 | 54.73 |

Further, the performance is measured by system modelling error without considering measurement noise. From Table 4.3, the results demonstrate that the performance is affected when uncertainty in stiffness parameter is considered. RF performance

degrades more than the ERT. The R-squared value of ERT is 0.98, 1.8% less than that in Scenario 1, but R-square for RF is 4% less than that in Scenario 1.

Table 4.3 Performance measurement for Scenario 3

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE | $6.5 \times 10^{-05}$ | $9.1 \times 10^{-05}$ |
| R-Squared | 0.98 | 0.948 |
| Training Time (Seconds) | 14.9 | 55.75 |

Finally, the performance of the proposed method is evaluated for the worst scenario in the study. Scenario 4 considers both noise measurement and system modelling error. As in Scenario 2, 5% and 10% noise levels are taken to investigate the effect of noise level when $\pm\,(1\% - 3\%)$ uncertainty in the stiffness parameter. Both noise and uncertainty in the stiffness parameter affect the performance of the proposed approach, as observed in Scenarios 2 and 3. The performance further degrades when both noise and uncertainty are considered. The R-square decreases by 4% to 5% for both ERT and RF when 5% measurement noise and uncertainty in stiffness parameters are considered. The effect is severe when the noise level is increased. R-square values for ERT and RF in Scenario 4 reduce by more than 10% of those obtained in Scenario 1 when a noise level of 10% is considered. Table 4.4 shows the results for Scenario 4. In all four scenarios, the performance of ERT is better than RF.

Table 4.4 Performance measurement for scenario 4

| Performance Metrics | 5% Noise | | 10% Noise | |
|---|---|---|---|---|
| | ERT | RF | ERT | RF |
| MSE | $7.5 \times 10^{-05}$ | $1.04 \times 10^{-04}$ | $1.39 \times 10^{-04}$ | $1.79 \times 10^{-04}$ |
| R-Squared | 0.956 | 0.94 | 0.89 | 0.869 |
| Training Time (Seconds) | 15.26 | 56 | 15.3 | 55.60 |

The prediction result for a single-element damage case and a two-element damage case are presented below for all four scenarios. The damage cases are taken from predictions made for the testing data by ERT. Figure 4.12 shows the damage identification for a case with a 17.5% stiffness reduction in element No. 8. ERT can give good predictions close to the actual damage. When the measurement noise and

uncertainty are considered, minor false positives are observed in the undamaged elements.

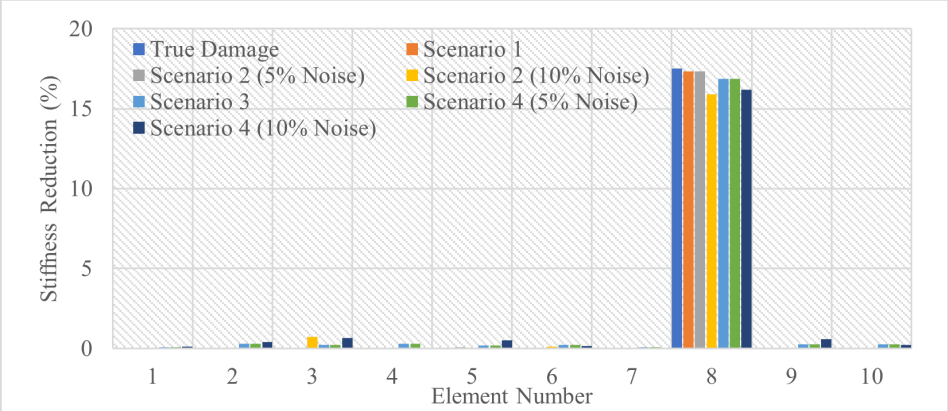



Figure 4.12 Damage identification results for a single element damage case

Another single element damage case with less than 10% stiffness reduction is selected to investigate whether the proposed method can identify minor damage. The actual damage in the element No. 1 is a 7% stiffness reduction. As shown in Figure 4.13, the prediction results show that the damage identification is close to the true damage for all scenarios. However, more false positives are observed in the undamaged elements compared to the previous single element damage case. More false positive values are observed in Scenarios 3 and 4 when both measurement noise and uncertainty in the stiffness parameter are considered.

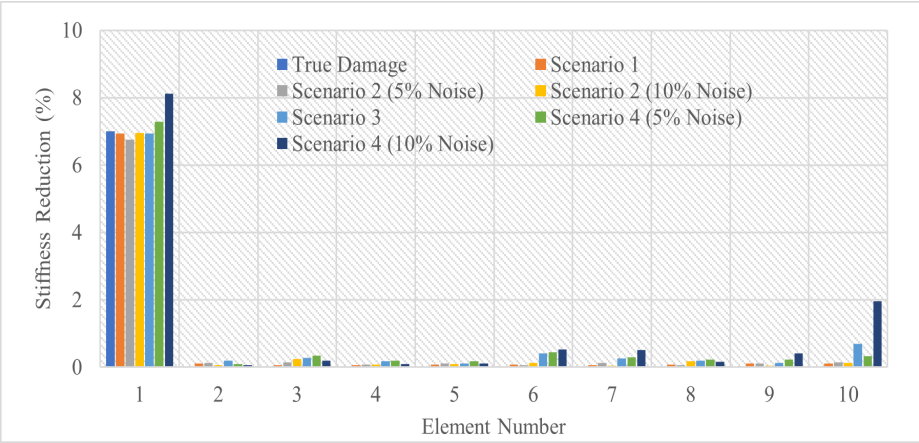



Figure 4.13 Damage identification results for a single element minor damage case

For multiple element damage cases, the prediction is shown for a case with damages in elements No. 3 and 8. The stiffness reductions are 6% and 12% in elements No. 3

and 8, respectively. The identification results corresponding to all four scenarios are presented in Figure 4.14.

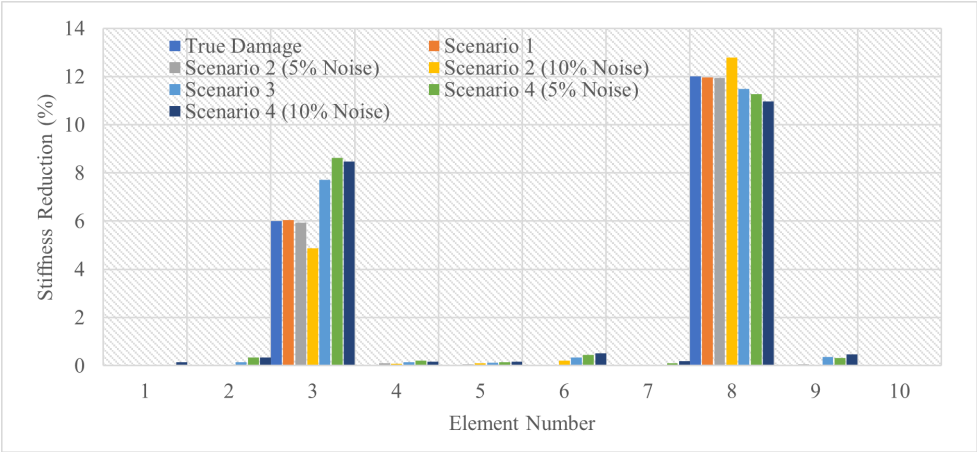



Figure 4.14 Damage identification results for a two-element damage case

Like the single element damage case, the damage locations are identified accurately, and damage quantification results are close to actual damage severities when uncertainty and noise are not considered. The performance is affected when a severe noise is considered and degrades further when uncertainty is also included. The prediction is not as close as the actual damage case, and slightly more false positive damage identifications are observed in the undamaged elements. However, the proposed method can still provide good damage identification and quantification for all four scenarios and damage cases, as demonstrated in the above numerical studies. For Scenario 4, when uncertainty and measurement noise are considered, the damage identification results are reasonably good. The damage quantification results demonstrate that damage larger than 2% stiffness reduction can be well identified under the effect of uncertainties and noise. In general, better results are obtained when damage levels are higher.

### 4.4.3 Performance Measurement with Reduced Sensors

The performance of the proposed method is further assessed by reducing the number of sensors from five to three. The sensors at node 2, node 8, and node 9 in Figure 4.4 are randomly selected. The variance plot for 10% noise in the acceleration dataset is shown in Figure 4.15 after performing PCA on concatenated IRF of the three sensors.

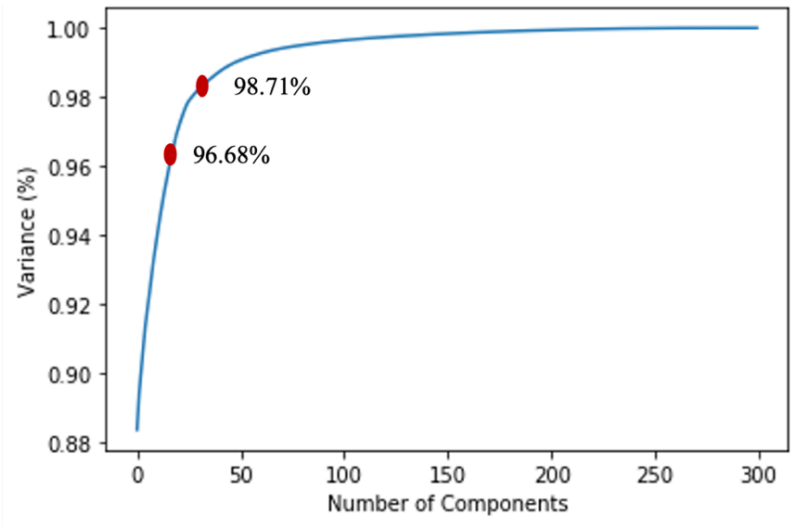



Figure 4.15 Variance plot

With 19 principal components, the variance of 98.71% is preserved when five sensors are used. However, when only three sensors are used, the variance is 96.68%. Table 4.5 - 4.8 shows the results for four scenarios using three sensors. Like the performance of using five sensors, the performance degrades with the inclusion of noise measurement and uncertainty.

Table 4.5 Performance measurement for Scenario 1

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE. | $2.4\times10^{-6}$ | $1.63\times 10^{-5}$ |
| R-Squared | 0.998 | 0.99 |
| Training Time (Seconds) | 14.28 | 56 |

Table 4.6 Performance measurement for Scenario 2

| Performance Metrics | 5% Noise | | 10% Noise | |
|---|---|---|---|---|
| | ERT | RF | ERT | RF |
| MSE | $1.2\times10^{-5}$ | $3.3\times10^{-5}$ | $1.34\times10^{-4}$ | $1.78\times10^{-4}$ |
| R-Squared | 0.95 | 0.936 | 0.62 | 0. 61 |
| Training Time (Seconds) | 14.41 | 53 | 1.16 | 54.9 |

Table 4.7 Performance measurement for Scenario 3

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE. | $6.71\times10^{-5}$ | $1.0\times 10^{-4}$ |
| R-Squared | 0.962 | 0.943 |
| Training Time (Seconds) | 14.5 | 57.2 |

Table 4.8 Performance measurement for Scenario 4

| Performance Metrics | 5% Noise | | 10% Noise | |
|---|---|---|---|---|
| | ERT | RF | ERT | RF |
| MSE | $1.0\times10^{-4}$ | $6.7\times10^{-4}$ | $1.3\times10^{-4}$ | $1.7\times10^{-4}$ |
| R-Squared | 0.943 | 0.88 | 0.623 | 0. 61 |
| Training Time (Seconds) | 14.41 | 53 | 15.16 | 54.9 |

It is observed that when there is no noise measurement in acceleration response, using three sensors and 19 principal components obtained from IRF can give almost the same R-squared and MSE as using five sensor measurements. However, the performance difference is significant when the noise is present in the acceleration response. This could be due to the variance preserved with 19 principal components when five and three sensors are used. The study investigated further the performance of the proposed method using three sensors and taking principal components which can retain 98.71% variance. For three sensors, 40 principal components can preserve 98.71% variance. The performance is measured using 40 principal components, and the results are shown in Tables 4.9 - 4.12.

Table 4.9 Performance measurement for Scenario 1

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE. | $1.5\times10^{-6}$ | $1.06\times 10^{-5}$ |
| R-Squared | 0.999 | 0.994 |
| Training Time (Seconds) | 24.60 | 109.8 |

Table 4.10 Performance measurement for Scenario 2

| Performance Metrics | 5% Noise | | 10% Noise | |
|---|---|---|---|---|
| | ERT | RF | ERT | RF |
| MSE | $5.6\times10^{-5}$ | $9.5\times10^{-5}$ | $1.6\times10^{-5}$ | $2.9\times10^{-4}$ |
| R-Squared | 0.969 | 0.949 | 0.87 | 0. 84 |
| Training Time (Seconds) | 25.63 | 113.02 | 26.10 | 109.23 |

Table 4.11 Performance measurement for Scenario 3

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE. | $6.6\times10^{-5}$ | $9.9\times 10^{-5}$ |
| R-Squared | 0.962 | 0.94 |
| Training Time (Seconds) | 25.8 | 113.8 |

Table 4.12 Performance measurement for Scenario 4

| Performance Metrics | 5% Noise | | 10% Noise | |
|---|---|---|---|---|
| | ERT | RF | ERT | RF |
| MSE | $1.0\times10^{-4}$ | $1.4\times10^{-4}$ | $2.7\times10^{-4}$ | $3.2\times10^{-4}$ |
| R-Squared | 0.936 | 0.918 | 0.82 | 0. 80 |
| Training Time (Seconds) | 26.75 | 116.34 | 26.97 | 111.68 |

It is observed that the performance of the proposed method has improved, especially in scenarios 2 and 4, by using principal components that can retain 98.71% variance, i.e., 40 principal components. However, using more principal components increases the training time. Figure 4.16 and 4.17 shows the damage identification result for single-element and two-element damage cases using 40 principal components, respectively. Some false positive damages are observed in undamaged elements, like in the case with five sensor measurements.

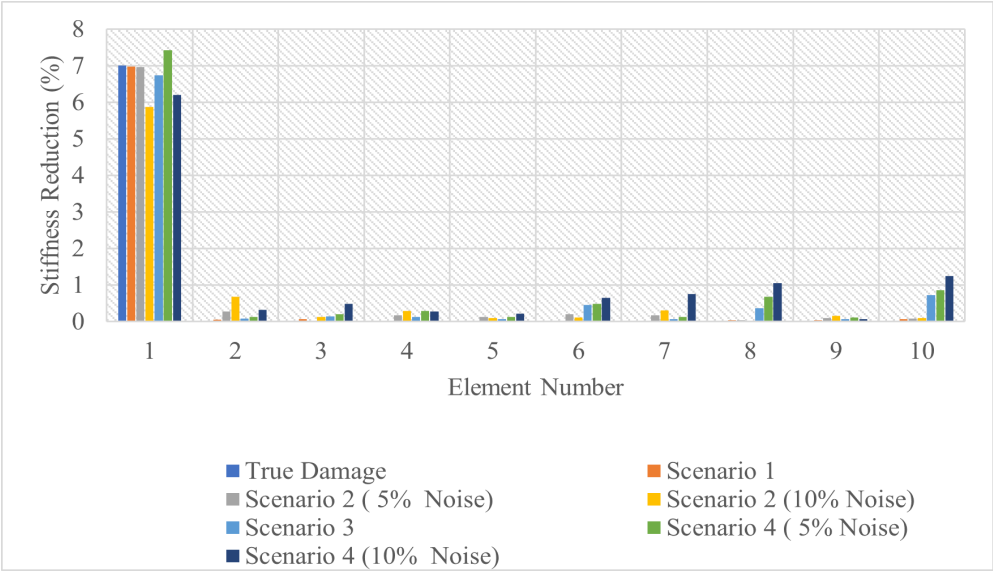



Figure 4.16 Single element damage identification

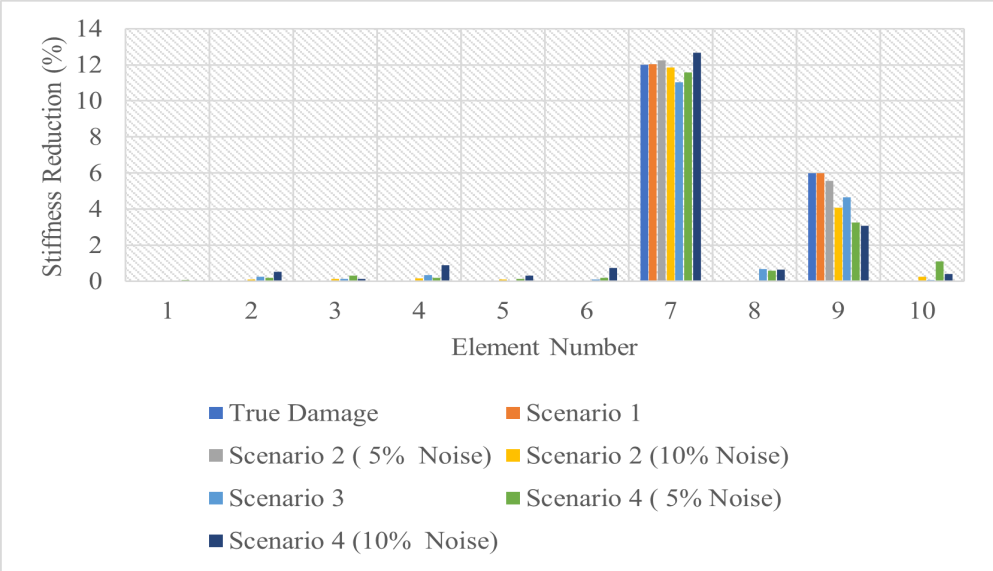



Figure 4.17 Two-element damage identification

## 4.5 Experimental Validation

Experimental studies are conducted on a steel frame structure in the laboratory (Li et al., 2012) to validate the effectiveness of the proposed approach. The detailed descriptions of the structure can be found in Refs. (Chencho et al., 2020; Li et al., 2012). The steel frame structure and the developed finite element model are shown in Figures 4.18 and 4.19, respectively. The number of sensors is reduced to six in the present study. The finite element model of the steel frame has 70 elements. The target output will have 70 labelled output variables, which are the percentage of stiffness reduction in the elements. Training datasets are extracted from the acceleration

responses measured from the updated finite element model. 15% of the total dataset is used as the validation dataset. During the laboratory testing, two samples of acceleration responses for the single-element damage and two-element damage cases are measured. The measured data are used as testing data in this study.

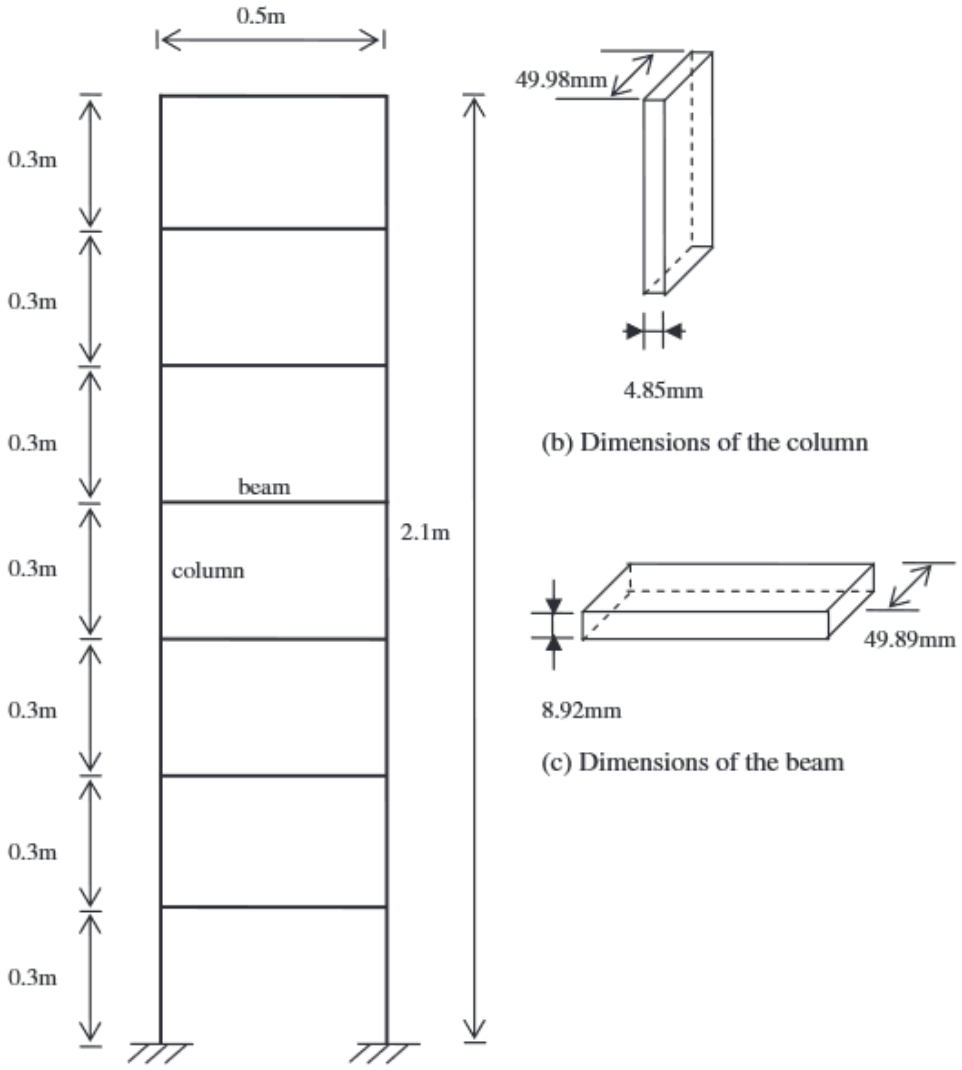



Figure 4.18 Frame Structure

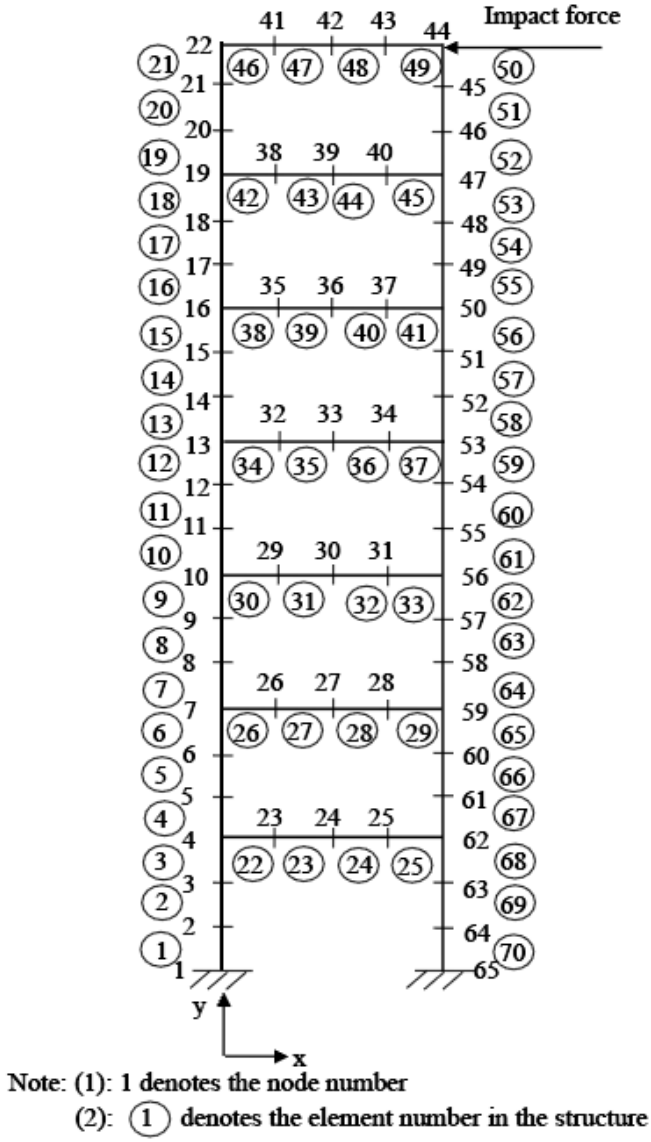



Figure 4.19 FEM of the frame structure

### 4.5.1 Data Generation and pre-processing

Datasets consisting of IRFs extracted from the acceleration responses obtained from the undamaged, single element and two-element damage cases are used to train and validate the proposed method. Acceleration responses are obtained when an impact force is applied at node 44 in the horizontal direction. Four samples of acceleration responses are collected for every damage case using random impact forces with a 1-2% variance of the measured force. Measurement is taken for one second at the sampling rate of 1000 Hz. Acceleration responses in the *x*-direction at nodes 7, 11, 17, 47, 53 and 56 are used for single-element damage cases, and at nodes 4, 11, 19, 50, 53 and 56 for the two-element damage case. IRFs are extracted using Equation (4.8), and the error is reduced by averaging the ensembles. A total of 8404 samples are

generated for the single element damage case. The stiffness reduction in each element is taken from 0% to 30% in steps of 1% for the single element damage case. The damage is introduced in any two elements with a maximum stiffness reduction of 20% for the two-element damage case. The damage is introduced in any two elements with a maximum stiffness reduction of 20% for the two-element damage case 20,883 samples are generated for the two-element damage case. The training of ERT models for single-element damage cases and two-element damage cases is carried out separately since the sensor locations used in the experimental tests under these two damage cases are different.

The data pre-processing process remains the same as discussed in the numerical study. IRFs generated using Equation (4.8) from the acceleration responses measured at six sensors are concatenated. Moving averaging is performed for smoothing the concatenated series. Further, PCA is performed for dimensionality reduction. The number of principal components is selected, taking more than 98.5% variance. For the single-element damage case, 29 principal components are selected, and 19 principal components are selected for the two-element damage case. 85% of the dataset is used for training, and the remaining is used for validation.

### 4.5.2 Results and Discussions

Experimental tests are conducted for single-element damage cases and two-element damage cases. The proposed method is tested with two sample datasets from the laboratory's real experimental data for the single-element and two-element damage cases, respectively. It should be noted that no ensembles are used in this test since only a limited number of experimental tests are conducted. For single element damage cases, the stiffness reduction in element No. 12 is 12.5%. For the two-element damage case, 12.5% stiffness reduction is introduced in elements 6 and 12. The performance of using ERT and RF is compared here to demonstrate the superiority of the proposed approach for a relatively complex structure with more elements and parameters to be identified. Both ERT and RF are grown using 120 decision trees and setting a minimum requirement of six samples at the internal node for further split and three pieces in each leaf.

For both single-element and two-element damage cases, ERT performs better than RF, as seen in Tables 4.13 and 4.14. The results demonstrate that the proposed approach

provides good damage identifications for a structure with many elements. Less training time is required by using ERT.

Table 4.13 Performance evaluation for single element damage case

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE | $1.07 \times 10^{-6}$ | $1.11 \times 10^{-6}$ |
| R-Squared | 0.999 | 0.997 |
| Training Time (Seconds) | 13.42 | 65.73 |

Table 4.14 Performance evaluation for two-element damage case

| Performance Metrics | ERT | RF |
|---|---|---|
| MSE | $2.8 \times 10^{-5}$ | $3.89 \times 10^{-5}$ |
| R-Squared | 0.985 | 0.975 |
| Training Time (Seconds) | 22.70 | 101.59 |

Figure 4.20 shows the damage identification results for the single element damage case from a sample experimental testing data. The identified stiffness reduction is 12.3% against the true stiffness reduction of 12.5% at element No.12 with ERT and 9.5% with RF. Some false positives are observed in other elements, which are less than 2%. Figure 4.21 shows the damage identification results of the two-element damage case. The identified stiffness reductions are respectively 13.4% and 11.68% in elements No. 6 and 12 with ERT, compared with 12.51% and 5.58% by using RF. The identified damage locations are very accurate, and the predicted severities are very close to the true stiffness reductions by ERT. For this multiple damage case, better identification results are obtained using ERT, compared with those from RF, as shown in Figure 4.21. The identification results in these experimental studies indicate that ERT outperforms the RF for identifying the damage more accurately and efficiently.

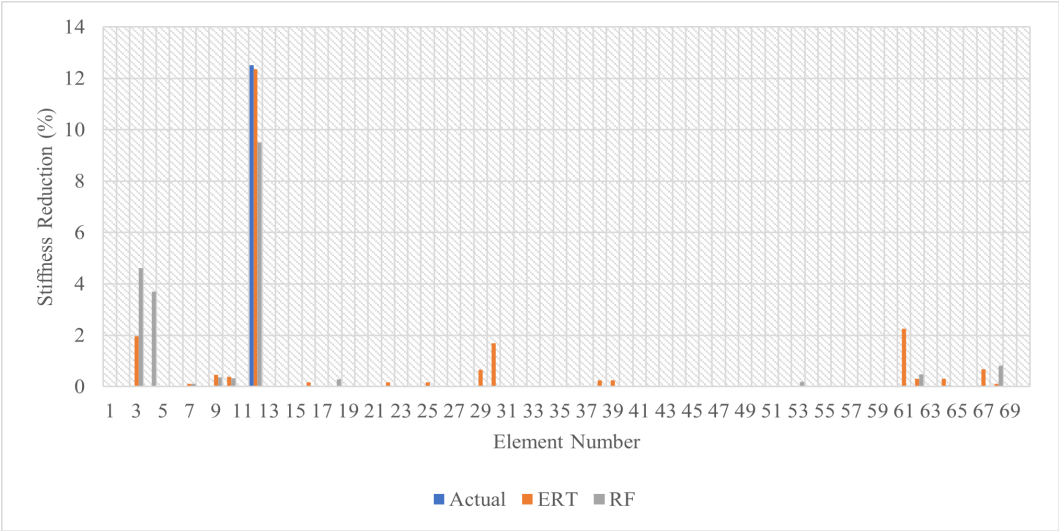



Figure 4.20 Single element damage identification results of the experimental frame structure

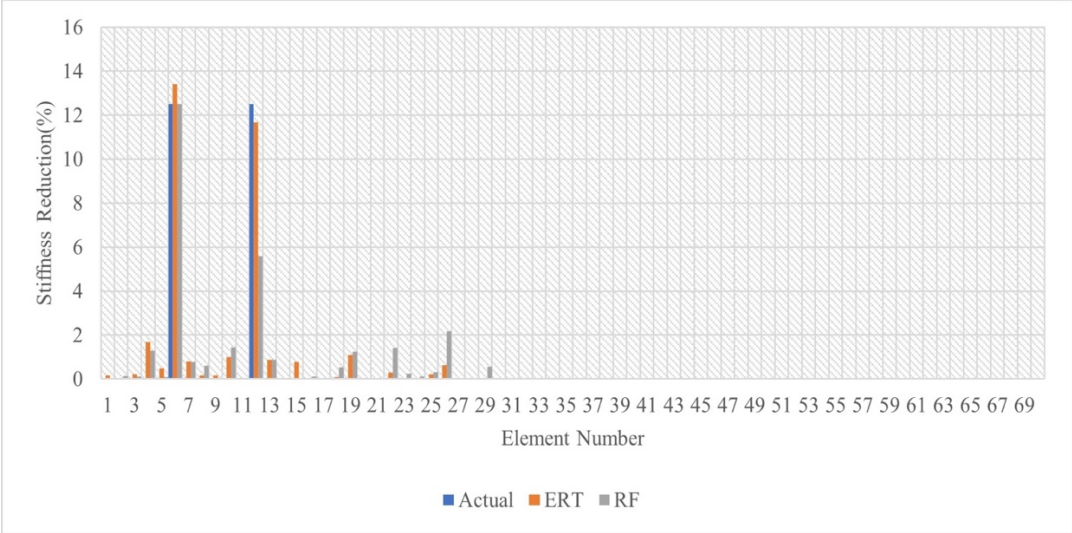



Figure 4.21 Multiple damage identification results of the experimental frame structure

## 4.6 Conclusion

This study presents a novel structural damage identification approach using an ensemble technique based on ERT and IRFs. The performance is compared with RF. IRFs extracted from acceleration responses are processed and used as the input to ERT and RF. The results in numerical and experimental studies demonstrated that the proposed approach based on ERT outperforms RF, as indicated by MSE, R-squared and training time. ERT's training time is significantly less than RF using the same structure and measurement data. It is also demonstrated that the proposed approach

performs well for structures with a relatively greater number of elements and unknown parameters to be identified.

However, it is known that the existing deep learning models and two studies carried out in Chapter 3 and Chapter 4 of this thesis needs the model to be trained and tested separately for each of the four scenarios considered. This may have problems with generalising the data ,i.e., if the model is trained with scenario 3 dataset and tested with scenario 4 dataset, overfitting may occur. Exploring and developing a machine learning model which can be trained with a set of datasets and tested with other scenario datasets to get good damage identification and quantification is a motivation behind the study carried out in the next chapter.

# CHAPTER 5

# 5 Damage Identification and Quantification Using Impulse Response Functions with Long Short-Term Memory Auto-encoder

## 5.1 Introduction

This chapter presents an approach for structural elemental damage identification and quantification using Long Short-Term Memory (LSTM) autoencoder and impulse response functions (IRFs). The same datasets generated from a simply supported beam in chapter 4 are used with the proposed model. IRF extracted from the acceleration responses are concatenated. Moving averaging with a suitable window size reduces random variations in the concatenated responses. Further, principal component analysis (PCA) is carried out on standardised datasets for dimensionality reduction. The selected principal components are then fed to the LSTM autoencoder for damage identification. The encoder of the LSTM autoencoder encodes the input sequence to a variable sequence of a fixed length vector, and the decoder maps the vector representation to a target variable. A noise layer is added as an input layer to the LSTM autoencoder, which can provide regularisation to the model. The proposed model consists of two phases, one for the reconstruction of principal components selected to extract the features and the other for the structural element damage identification. The reconstruction network is tuned to get good reconstruction results. After getting a good result, the encoder network of the reconstruction network is kept the same, and the decoder is further tuned to get a good damage identification considering the stiffness reduction in the elements as the target variable.

Numerical studies are carried out on a simply supported beam, and data are generated using finite element model (FEM) analysis. This study provides 1). A deep neural network using LSTM auto-encoder for structural damage identification; 2) IRFs extracted from the time-domain acceleration responses using a small number of sensors are explored.

## 5.2 LSTM Auto-encoder

LSTM has been widely used in sequence prediction problems, such as sequence classification and sequence-to-sequence prediction, which may differ based on the input and output sequences. LSTM is a type of recurrent neural network (RNN); it has

recurrent connections that use the state of neuron activation from the previous time step to determine the output (Brownlee, 2018). RNNs are designed for sequence prediction, for example, multi-layer perceptron by adding loops. RNNs face the challenge of getting trained effectively owing to vanishing gradient problems; LSTM is designed to overcome this challenge. Moreover, LSTM can learn long-term dependencies better than an RNN (Lecun et al., 2015). LSTM use gates that process the information contained in the incoming data sequences, stored in the network, and leaving the network. The gates include a forget gate, an input gate, and an output gate, each of which can be considered a neural network (Xu & Yonedal, 2021; Zhang et al., 2020). The output from an LSTM cell depends on the long-term memory (referred to as the cell state), output from the previous time step, and data of the input sequence of the current time step. Figure 5.1 shows the LSTM cell.

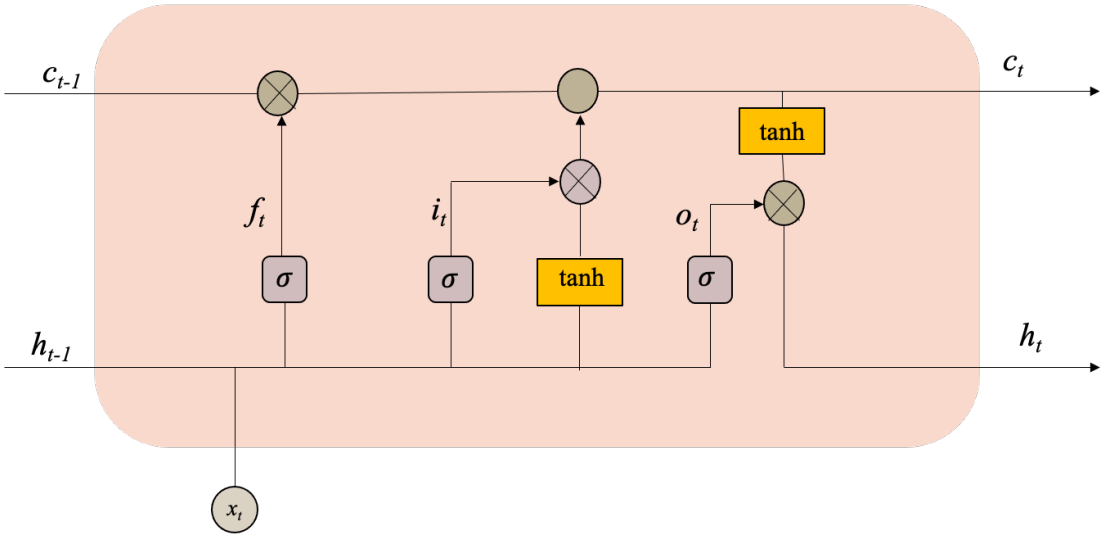



Figure 5.1 LSTM Cell

The forget gate determines which information of the cell state is useful based on the previous hidden state and the current input data. The previous hidden state and the current input data are fed to the forget gate, which generates a vector in the interval between 0 and 1 using sigmoid activation. A value of 0 indicates that the input component is irrelevant, whereas 1 show that the input component is relevant. In the following equations, $W$ and $W_h$ are the input and hidden state weights for the forget gate $f$, input gate $i$, and output gate $o$.

$$f_t = \sigma\left(x_t \times W_f + h_{t-1} \times W_{hf}\right) \tag{5.1}$$

where $x_t$ denotes data at time $t$, $h_{t-1}$ denotes previous hidden state, and $f_t$ is the forget gate at $t$.

The amount of information sent is based on this value. More information is sent when the $f_t$ value is closer to 1. The output from this part of the forget gate is multiplied pointwise by the previous cell state. The output of pointwise multiplication is given by Equation (5.2) or equation (5.3) when $f_t$ is 0 or 1, respectively. When $f_t$ is 1, all the previous cell states are sent.

$$C_{t-1} * f_t = 0 \; for \; f_t = 0 \tag{5.2}$$

$$C_{t-1} * f_t = C_{t-1} \; for \; f_t = 1 \tag{5.3}$$

where $C_{t-1}$ denotes the previous cell state.

The next stage is the input gate. The input gate uses the current input and the previous hidden states to quantify the amount of new information to be retained along with the previous cell state. The equation for the input gate can be expressed as:

$$i_t = \sigma(x_t \times W_i + h_{t-1} \times W_{hi}) \tag{5.4}$$

where $i_t$ is input gate at time instant $t$.

Using a sigmoid function, the input gate outputs a vector ranging from 0–1. The new information is also a function of the previous hidden state and current input data. It uses the *tanh* activation function to output the new information ranging from -1 and 1. The new information is determined by:

$$I_N = tanh(x_t \times W_c + h_{t-1} \times W_{hc}) \tag{5.5}$$

The new information $I_N$ generated using the input gate can help to reduce the information in the cell state from the previous, if necessary. A negative value using *'tanh'* helps to achieve this. However, the necessity of retaining all the new input data is unknown. The output from the input gate is multiplied pointwise by the new input vector to determine the amount of new input data that can be retained. This output is added to the cell state. The current cell state, $C_t$ is given by:

$$C_t = C_{t-1} * f_t + I_N * i_t \tag{5.6}$$

The output gate is the last gate; it determines the new hidden state. It uses the current input data and previous hidden states like the forget and input gates. The equation for

the output gate can be expressed as:

$$o_t = \sigma(x_t \times W_0 + h_{t-1} \times W_{ho}) \tag{5.7}$$

The new hidden state will be the output of the pointwise multiplication of the output gate and the new cell state. The *tanh* function is applied to the current cell state resulting in new output in the range of -1 and +1. The new hidden state, $h_t$ can be expressed as:

$$h_t = \sigma(x_t \times w_o + h_{t-1} \times w_{h0}) * tanh\,(C_t) \tag{5.8}$$

LSTM auto-encoder is an LSTM-based architecture (also called encoder-decoder LSTM); it is shown in Figure 5.2. The encoder reads the input sequence and encodes it into a vector (Hochreiter & Schmidhuber, 1997). The decoder decodes this vector and outputs the predicted sequence. LSTM auto-encoders have been used for unsupervised problems, especially those related to the video generation (Srivastava et al., 2015) and speech synthesis (Wan et al., 2017).

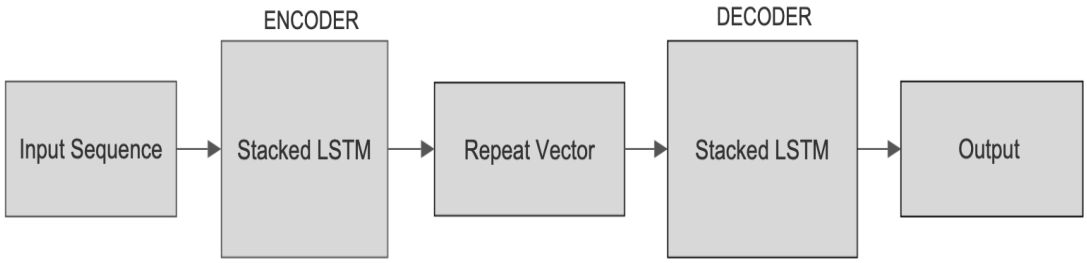



Figure 5.2 LSTM Auto-encoder

## 5.3 Proposed network configuration

The proposed LSTM autoencoder consists of two phases. Figure 5.3 shows the proposed network implemented using Keras Functional API (Fchollet, 2019) for deep learning. The first phase involves reconstructing the input sequence to extract the best feature representing the input sequence, and the second phase consists of the relationship learning for structural damage identification. A network may learn effectively from a training dataset to train the input samples and the corresponding outputs. Still, it may perform poorly on new datasets, such as testing and validation datasets. This results in a generalisation error, which can often be improved by adding random noise, resulting in effective network learning of the training dataset (Brownlee,

2019b). The Keras API enables the addition of white noise to the network via a Gaussian layer, as shown in Figure 5.3.

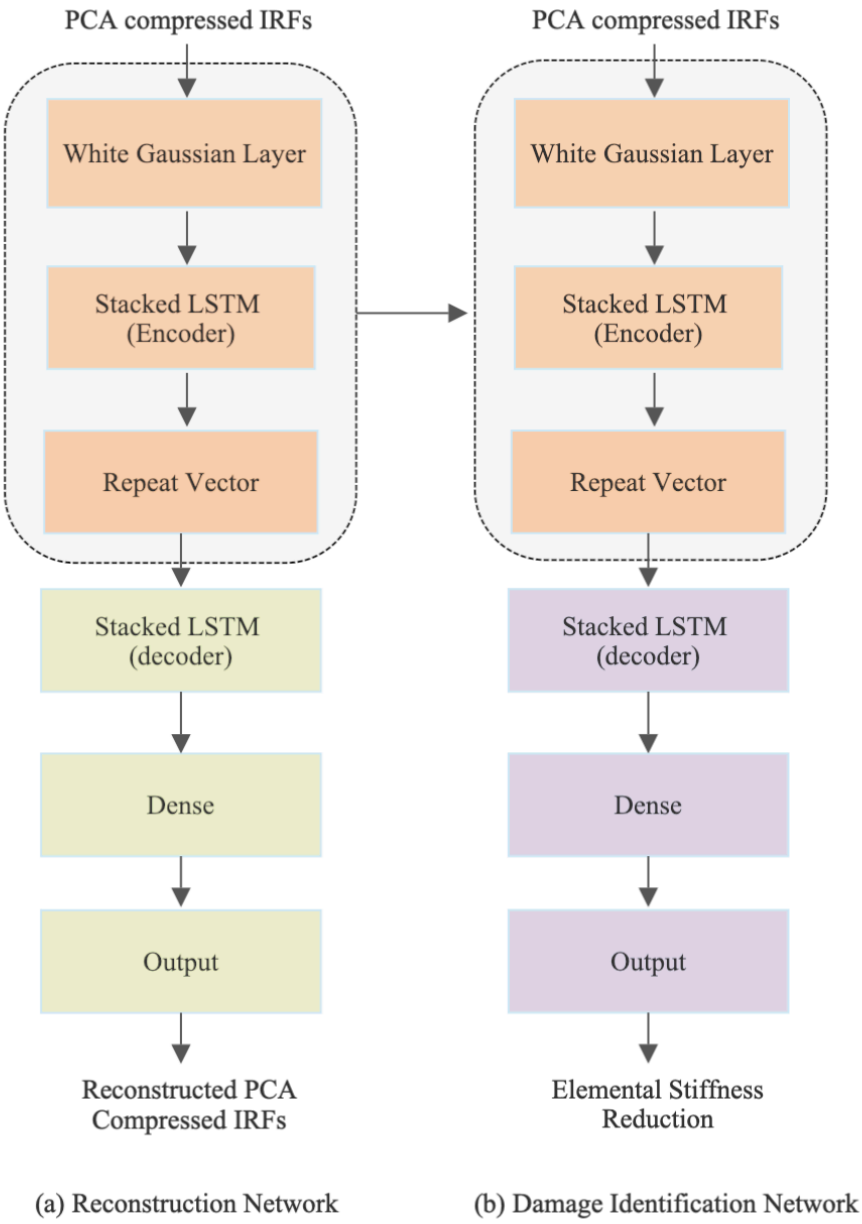



Figure 5.3 The proposed LSTM-Autoencoder

The addition of a Gaussian layer adds noise with a mean of zero. The standard deviation of noise can be tuned to obtain the best results for the network. For the proposed network, a standard deviation of 0.03 is used, and the Gaussian layer is added as an input layer to the encoder. The encoder outputs internal representations of the input sequences. The input to the network for both the network requires 3-dimensional input of [samples, timestep and features]. Here the samples denote the observation, the

timestep is the length of the sequence, and features is the number of features in the input. Both encoder and decoder are stacked LSTM network. A stacked LSTM network has multiple hidden layers, and each layer will have numerous memory cells. The output from the encoder is 2-dimensional output, and the decoder requires 3-dimensional input. Repeat Vector layer is used here to create 2-dimensional output from the encoder to 3-dimensional data for the decoder. The decoder maps the learned internal representation of the input sequence to the output sequence. A dense layer is used as the output for the network.

In the first phase, the encoder and decoder are tuned to obtain the maximum R-squared for reconstructing the input sequence. The input sequence is the PCA compressed IRFs. IRFs are extracted from acceleration measurements from the structure. IRFs extraction is described in Chapter 4. The decoder uses the learned internal representations of the input signal by the encoder as the input to reconstruct the input sequence. Both the encoder and decoder are tuned to get a reasonable reconstruction of the input sequence from the decoder. The damage identification network is developed using the same encoder when the maximum R-squared is achieved in the rebuilding input sequence. The output is to predict structural elemental stiffness reduction. The decoder of the damage identification network are tuned to get good damage identification results. Therefore, in the damage identification phase, the learned internal representation of the input sequence is mapped to stiffness reduction. Figure 5.3(b) shows the network model of the proposed method for structural damage identification.

## 5.4 Methodology

Under this section, data generation, data pre-processing and the network configuration are explained.

### 5.4.1 Data Generation and Pre-Processing

The study is carried out on a simply supported beam with 10 elements used in Chapter 4. The same dataset generated for ERT is used here to improve the damage identification further. Figure 5.4 shows the beam structure and selected sensor locations. IRFs is generated using equation (4.8) of Chapter 4. A summary of data generation is given below.

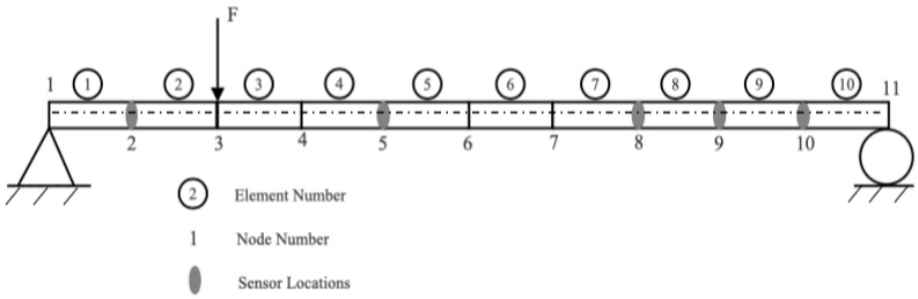



Figure 5.4 Simply supported beam and the sensor locations

IRFs extracted from five sensor locations shown in Figure 5.4 are concatenated, making the sequence length 500 sample points. Moving averaging removes random variations by selecting a suitable window size. 500 sample points in a sequence are pretty long for the LSTM, which can increase the computational cost of the LSTM network. PCA is performed on the concatenated input sequence to reduce the overall computational cost by reducing the sample length. 19 principal components are selected. A minimum total variance of 98.71% of the data is preserved while selecting the number of principal components. These 19 principal components are taken as the input sequence to the LSTM auto-encoder network. Therefore, the input to the LSTM Autoencoder will have 19-time steps and one feature. The entire dataset is reshaped into [*samples, timesteps, features]*.

The previous study considers noise measurement and uncertainty in system modelling. The damages in the structural elements are defined as the stiffness reduction by reducing the stiffness parameter, Young's modulus. The Young's modulus of all ten elements cannot be the same due to impurity in the material or manufacturing quality, which is inevitable. This was defined herein as 'uncertainty'. Accordingly, four scenarios were considered for measuring the performance of the proposed framework.

Scenario 1: Acceleration responses measured from selected sensor locations without noise or uncertainty. The IRFs are extracted from the measurement responses.

Scenario 2: White noise is added to acceleration responses recorded in Scenario 1. The IRFs are extracted from the acceleration responses along with noise measurements. Two different levels of white noise, 5% and 10%, are considered to determine the effect of the noise level on the performance.

Scenario 3: Acceleration responses measured with uncertainty. Uncertainty ranging ±1%–3% are included randomly in the stiffness parameters in the FEM. The IRFs are extracted from the acceleration responses. No noise measurements are considered herein.

Scenario 4: Acceleration responses are measured considering the uncertainty and noise measurements. Acceleration responses are measured, including uncertainties ranging from ±1%–3%. Noises of 5% and 10% are added to the measured acceleration responses, and the IRFs are extracted.

Only the IRFs extracted from Scenarios 1 and 3 are used for training and testing the framework. For Scenario 1, 11,300 damage cases of single- and multiple-element damage cases are simulated. Three samples are generated for each damage case, and each sample is generated under a different impact force. For the same damage cases, uncertainty (ranging ±1%–3%) is included in the stiffness parameter, and the acceleration responses are measured, from which the IRFs are extracted and considered as Scenario 3. Scenarios 1 and 3 had the same damage cases (11,300). Both the scenarios are combined to yield 22,600 damage cases, and with three samples for each damage case, 67,800 samples are generated. Four undamaged samples are added to the dataset, forming the final dataset with shape *[67804, 19, 1]* for the study.

Scenarios 1 and 3 are combined to train and test the proposed framework. The network is trained using 85% of the entire dataset. The remaining 15% is used as one set for testing the proposed model. Fifteen percentage of the training dataset was used to validate the proposed method. Three sets of test data are used to test the performance of the proposed method.

i. Random 15% of the dataset to test Scenarios 1 and 3
ii. Scenario 2 dataset
iii. Scenario 4 dataset

### 5.4.2 Results

During the reconstruction phase, the 19 principal components are used as the input sequence to the LSTM auto-encoder, and the same sequence is used as the output for the reconstruction. The first and second principal components are considered the values for the first- and second-time steps, respectively. The number of LSTM cells and layers in both the encoder and decoder are tuned to obtain the maximum R-square

for 15% of the test dataset. The reconstruction network is compiled using the Adam optimiser by considering the mean square (MSE) error as the loss function. An R-squared of 0.996 is achieved with the reconstruction network configuration. Table 5.1 presents the reconstruction results of the two random samples; the reconstructed sequence is almost identical to the input sequence.

Table 5.1 Input sequence and reconstructed sequences

| Time steps | Sample 1 | | Sample 2 | |
|---|---|---|---|---|
| | Input Seq. | Reconstructed Seq. | Input Seq. | Reconstructed seq. |
| 1 | -9.05 | -9.07 | -11.31 | -11.24 |
| 2 | 1.70 | 1.69 | -10.72 | -10.72 |
| 3 | -0.68 | -0.68 | 6.77 | 6.89 |
| 4 | -1.49 | -1.52 | 5.00 | 5.07 |
| 5 | 0.89 | 0.84 | 0.99 | 0.89 |
| 6 | 0.10 | 0.11 | 1.16 | 1.26 |
| 7 | -1.24 | -1.24 | -1.83 | -1.88 |
| 8 | -1.97 | -1.98 | 2.27 | 2.25 |
| 9 | 0.87 | 0.83 | -1.62 | -1.65 |
| 10 | -0.28 | -0.29 | -1.26 | -1.31 |
| 11 | 0.05 | 0.02 | -1.19 | -1.19 |
| 12 | 0.32 | 0.35 | -0.59 | -0.57 |
| 13 | 0.06 | 0.10 | 0.37 | 0.29 |
| 14 | -0.09 | -0.10 | -0.17 | -0.24 |
| 15 | 0.21 | 0.19 | 0.38 | 0.41 |
| 16 | -0.72 | -0.69 | 0.02 | 0.03 |
| 17 | 0.11 | 0.10 | -0.69 | -0.66 |
| 18 | -0.22 | -0.19 | 0.39 | 0.33 |
| 19 | -0.12 | -0.12 | -0.07 | -0.03 |

The learning plot for the reconstruction network over 100 epochs is shown in Figure 5.5. The plot provides information on the network performance of the training and validation datasets and can be used to determine whether the network underfits, overfits, or learns well. The training curve (obtained from the training dataset) provides

information on the learning performance of the network. In contrast, the validation curve (calculated for 15%–85% of the total dataset) includes information on the generalisation performance of the network. The plot shows that training and validation losses decrease up to their respective stability points; a small gap exists between the training and validation loss curves. This curve nature indicates a good fit.

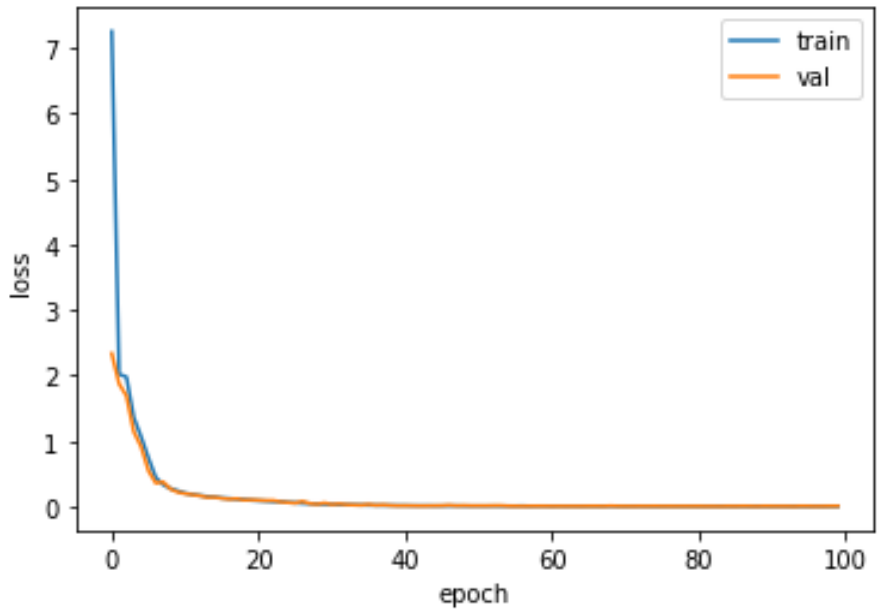



Figure 5.5 Loss curve for reconstruction network

After achieving an R-squared value of 0.996, the decoder of the reconstruction network is replaced with another stacked LSTM network. The decoder is tuned further to map the features extracted by the encoder to stiffness reduction. However, the same decoder, which comprised a layer less than that in the reconstruction network, yields good results. The performance of the LSTM auto-encoder in damage identification is measured using the MSE and R-square for the scenarios. For Scenario 2, 5% and 10% noises are considered for investigating the effect of the measurement noise. Table 5.2 lists the MSE and R-squared for scenarios 1 and 2. Table 5.3 shows the results for scenarios 3 and 4. The proposed method exhibits good damage-identification performance. The performance is slightly affected when noise is present in the acceleration response. The performance further degrades with the increasing noise level. The learning curve for damage identification is shown in Figure 5.6, which indicates a good fit.

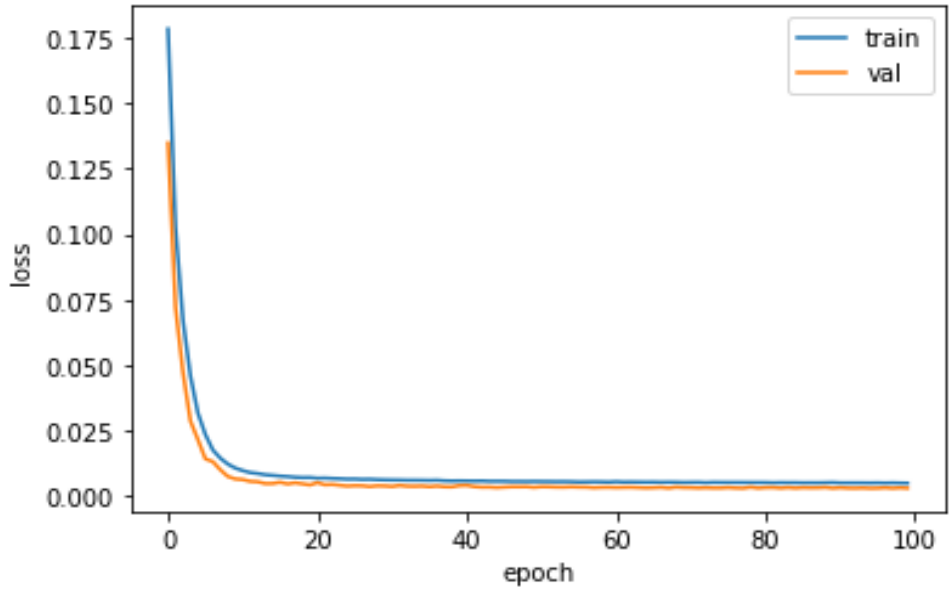



Figure 5.6 Loss curve for damage identification network

Table 5.2 Performance measurements using five sensor measurements for scenario one and Scenario 2

| Performance Metrics | Scenario 1 | Scenario 2 | |
|---|---|---|---|
| | | 5% Noise | 10% Noise |
| MSE | $3.21 \times 10^{-06}$ | $6.15 \times 10^{-06}$ | $2.5 \times 10^{-05}$ |
| R-Squared | 0.998 | 0.996 | 0.985 |

Table 5.3 Performance measurements using five sensor measurements for scenario 3 and Scenario 4

| Performance Metrics | Scenario 3 | Scenario 4 | |
|---|---|---|---|
| | | 5% Noise | 10% Noise |
| MSE | 4.16 x 10-05 | 4.9 x 10-05 | 7.08 x 10-05 |
| R-Squared | 0.974 | 0.971 | 0.953 |

The R-square decreases by 0.2% and 1.3% at 5% and 10% noise level, respectively. When uncertainty is considered without any noise measurement in the acceleration response, as in Scenario 3, the R-squared decreases by 2.4%. With both noisy measurement and uncertainty effect, the performance degrades further, depending on the noise level. Decreases of 2.7% and 4.6% in R-squared values are observed at 5% and 10% noise levels, respectively. Some negative stiffness reductions are observed in the non-damaged elements, but they are nearly zero.

The damage identification results for the single- and multiple-element cases for all scenarios are presented below. Damages of the stiffness reductions smaller than 10% are defined herein as minor and major damages, respectively. Figure 5.7 shows the damage identification results for the single-element damage case with a 7% decrease in the stiffness reduction in element No. 4. The predicted damage is 6.75% when no noise measurement is considered. For the 5% and 10% noise measurements, the predicted damage identification results are 6.03% and 5.87%, respectively. In Scenario 3, the predicted result is 8.8%. The performance degrades when both the uncertainty and noisy measurements are considered. With uncertainty and noise measurements of 5% and 10% noise levels, the predicted results are 9.09% and 9.24%, respectively. Some false positive damage identification results are observed in the undamaged elements, but the values are less than 0.5%.

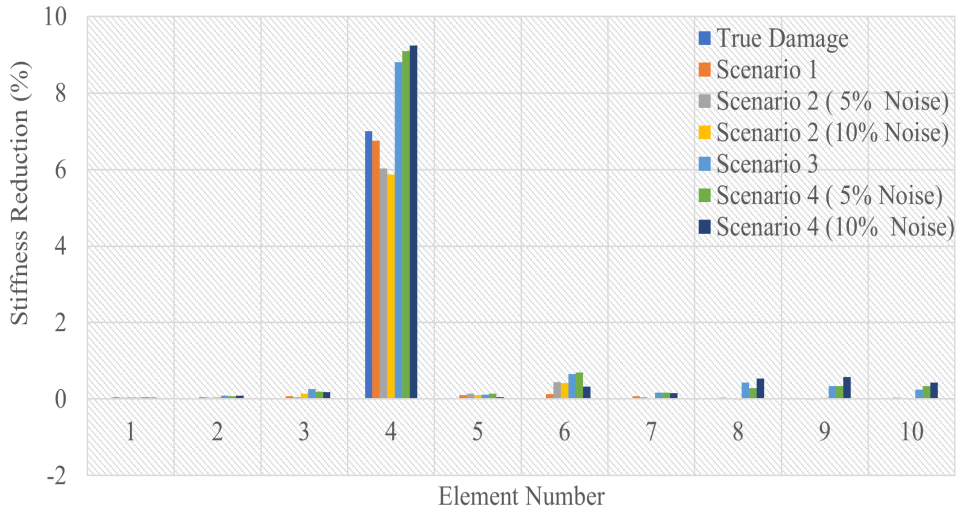



Figure 5.7 Damage identification of single element damage (minor)

Figure 5.8 shows the single-element damage identification results with a 20% stiffness reduction in the first element under different scenarios. In all the scenarios, the introduced damage can be reliably identified. The performance degrades with an increase in the noise level and degrades further when uncertainty is considered.

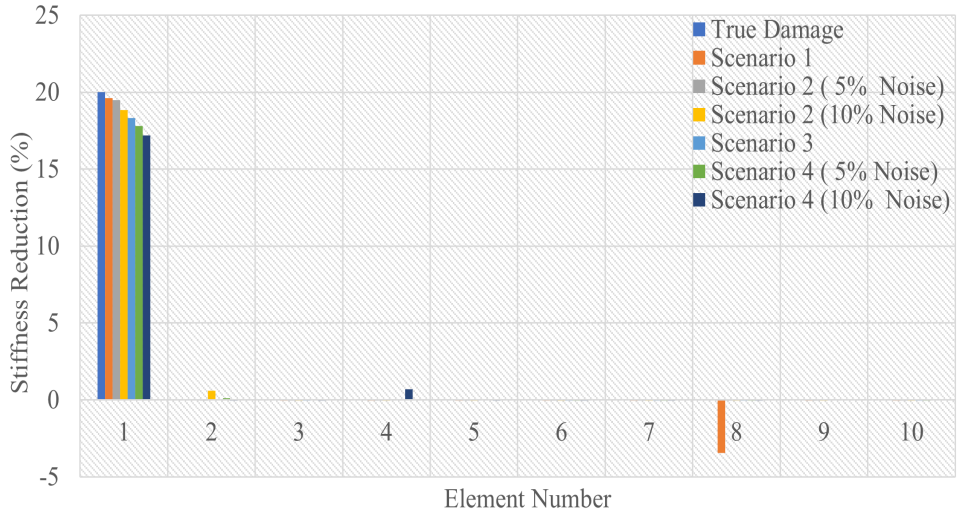



Figure 5.8 Damage identification of single element damage (major)

For the multiple-element damage cases, two-and three-element damage cases are considered. In Figure 5.9, stiffness reductions are observed in elements No. 1 and 5. True damages of 6% and 9% stiffness reductions are considered for elements No. 1 and 5, respectively. Figure 5.9 shows the damage identification results in another two-element damage case. The stiffness reductions are 13.5% for element No. 1 and 12% for element No. 6. As shown in these figures, the proposed approach obtains good damage identification results. However, the identification results are good even when the measurement noise and uncertainty in the stiffness parameter are considered. Very few and small false positive stiffness reductions are observed in Figure 5.10

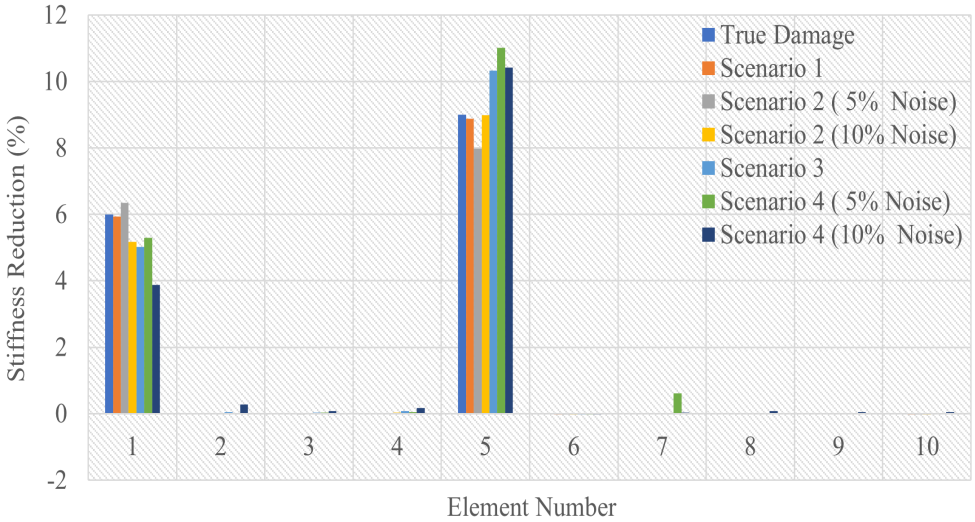



Figure 5.9 Damage identification of two element damage (minor)

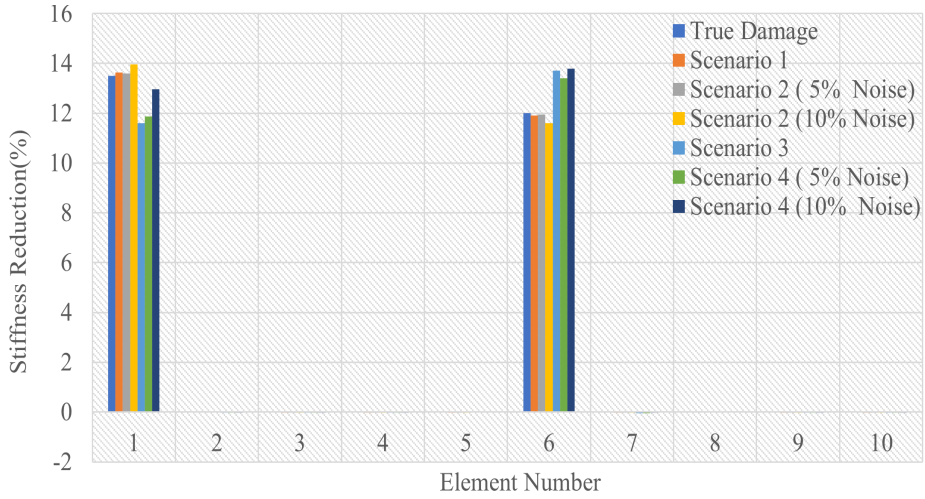



Figure 5.10 Damage identification of two element damage with more than 10% stiffness reduction

For a three-element minor damage case, stiffness reductions of 6%, 10.5%, and 6% are introduced in elements No. 2, 4, and 7, respectively. For a major damage case, the stiffness reductions are 10.5%, 15%, and 15% for elements No. 2, 4, and 9, respectively. The identification results for minor and major damages herein are consistent with those in the single- and two-element damage cases. Figures 5.11 and 5.12 show the identification results for the minor and major three-element damage cases, respectively. The prediction results for both single- and multiple-element damage cases are pretty close to those of the true damage case, even when the measurement is considered.

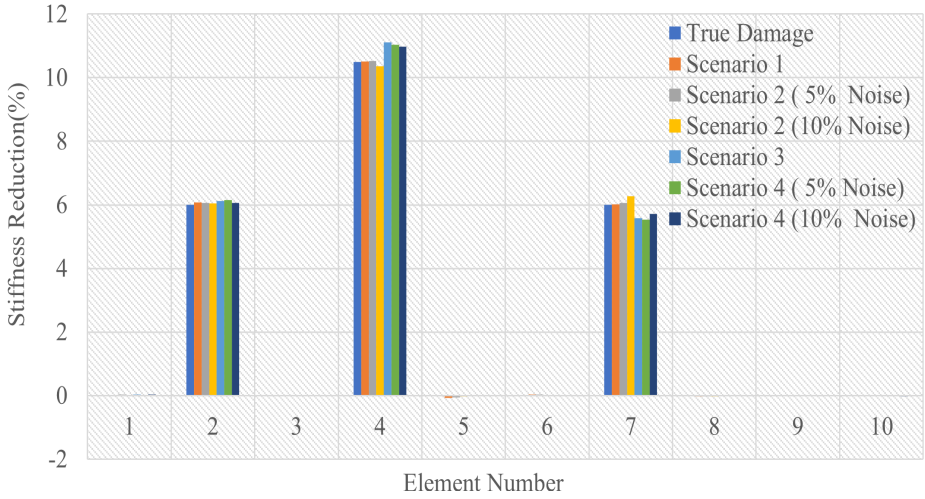



Figure 5.11 Damage identification of three element damage (minor)

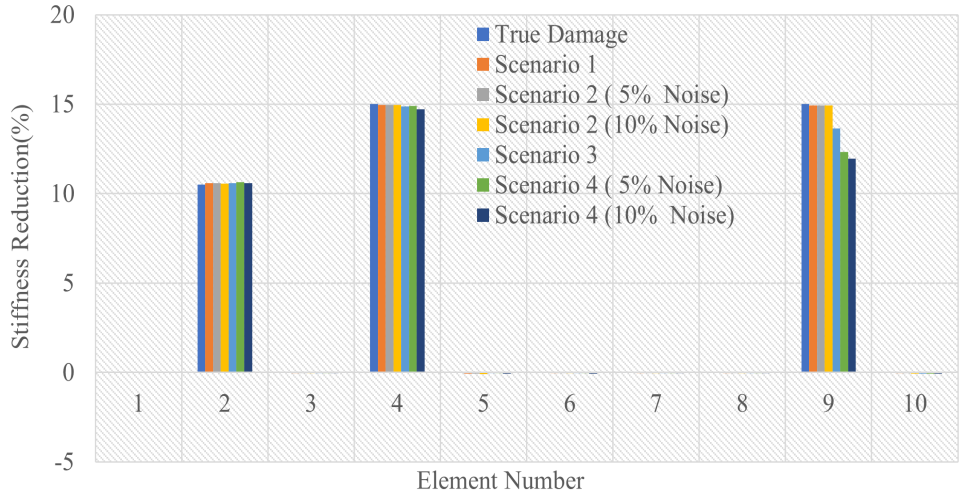



Figure 5.12 Damage identification of three element damage (major)

Next, the number of sensors is reduced from five to three. The vertical acceleration responses are measured from three sensors at nodes. No. 2, 8 and 9, and these responses are processed in a similar way as in the above cases of using five sensors. The same damage cases in the training, validation and testing datasets of the above case with five sensors are used. The 19 principal components selected from the concatenated IRFs are used as inputs to the network with the same architecture, and the performance is measured. For five sensor cases, the variance retained is 98.71% with 19 principal components. When the sensor number is reduced to three, with 19 principal components, 96.68% variance is preserved. The proposed approach is trained and tested using 19 principal components to compare the performance with the extremely randomised tree in Chapter 4 using the same sensor locations. In Chapter 4, when 19 principal components are used using three sensors, the performance is not good. The performance is improved using 40 principal components which retain 98.71% variance. It is observed that the proposed approach in this chapter can still give good damage identification close to using five sensor datasets. However, as presented in Figure 5.13, 40 principal components must be considered if the number of principal components is selected to retain 98.71% variance using three sensors. This will increase the training time of the proposed model since the sequence length is more than twice the length using five sensors.

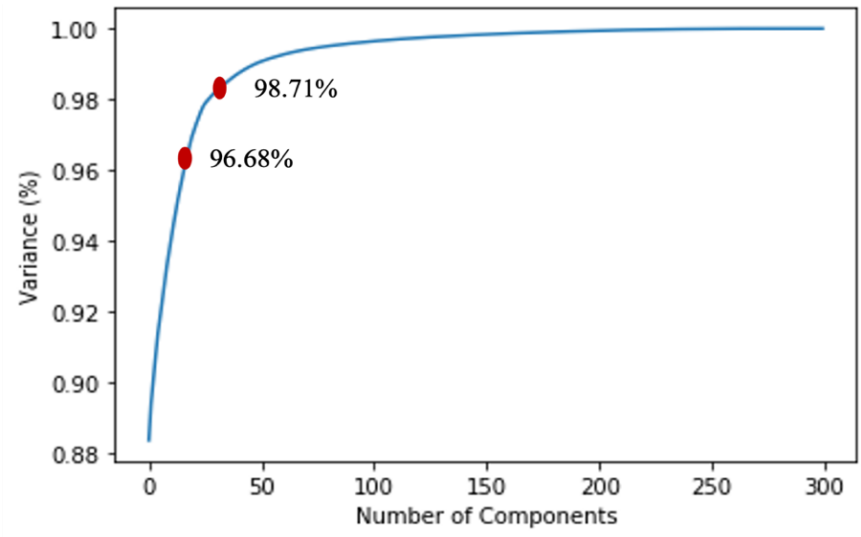



Figure 5.13 Variance plot

Tables 5.4 and 5.5 present the performance measurements of the proposed method using three sensor measurements. R-squared and MSE values are measured for all scenarios considered in the five-sensor measurement case. Both the MSE and R-squared values indicate good damage identification. The results are close to those obtained using the five-sensor measurements.

Table 5.4 Performance measurements using three sensor measurements for scenario 1 and 2

| Performance Metrics | Scenario 1 | Scenario 2 | |
|---|---|---|---|
| | | 5% Noise | 10% Noise |
| MSE | $5.7 \times 10^{-06}$ | $8.5 \times 10^{-06}$ | $2.7 \times 10^{-05}$ |
| R-Squared | 0.996 | 0.994 | 0.984 |

Table 5.5 Performance measurements using three sensor measurements for scenario 3 and 4

| Performance Metrics | Scenario 3 | Scenario 4 | |
|---|---|---|---|
| | | 5% Noise | 10% Noise |
| MSE | $5.3 \times 10^{-05}$ | $6.21 \times 10^{-05}$ | $8.69 \times 10^{-05}$ |
| R-Squared | 0.967 | 0.961 | 0.945 |

Figures 5.14 – 5.16 shows the comparative damage identification results for Scenario 1 using five- and three-sensor measurements. The introduced stiffness reduction in element No. 5 is 9.5%. The predicted stiffness reductions are 9.18% and 9.07% when

using five and three sensors, respectively. For a randomly selected two-element damage case, the actual stiffness reductions are 13.5% for element No. 2 and 6% for element No. 8. The predicted stiffness reductions are 13.3% for element No. 2 and 5.91% for element No. 8 (five sensors), and 13.4% for element No. 2 and 5.17% for element No. 8 (three sensors), respectively.

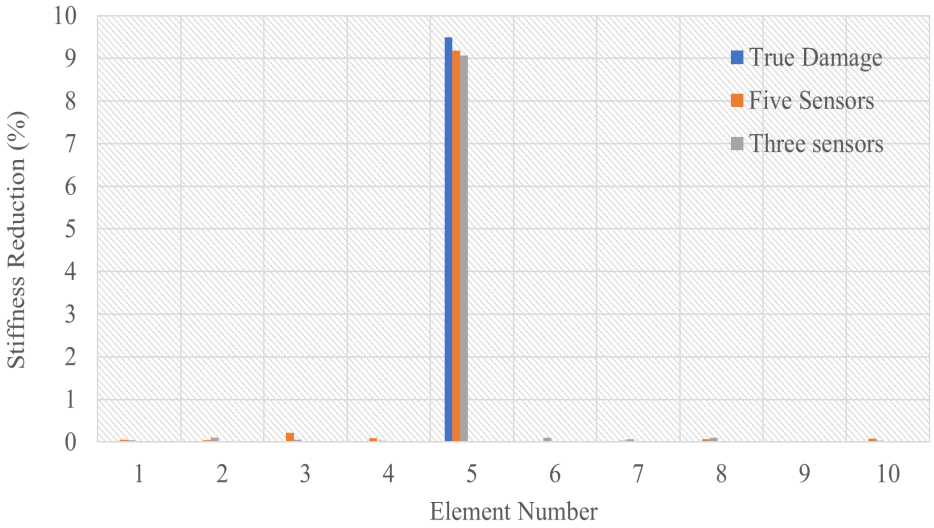



Figure 5.14 Damage identification for single element damage case (scenario 1)

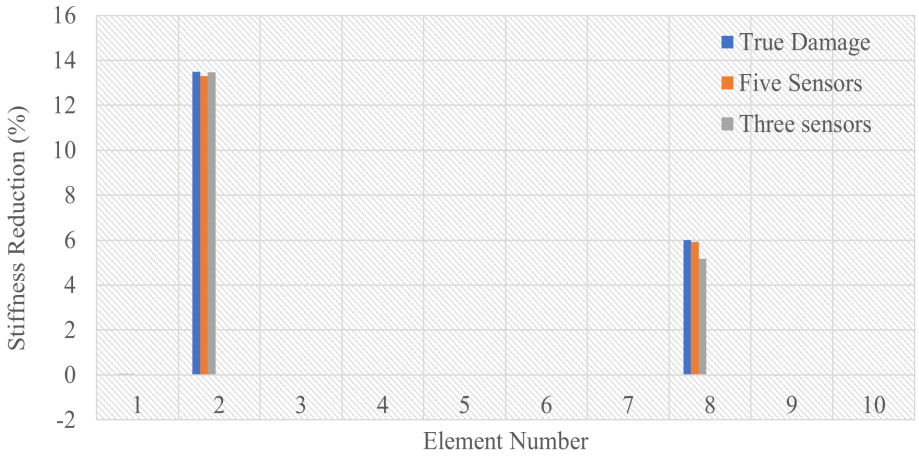



Figure 5.15 Damage identification for two-element damage for case (scenario 1)

The results for a three-element damage case are also shown here. The simulated stiffness reductions in the three elements, No. 2, 5 and 9, are 10.5%, 15%, and 15%, respectively. Five-sensor measurements yield predicted stiffness reductions of 10.44%, 14.91%, and 14.90% for elements No. 2, 5, and 9, respectively. On the other hand, the three-sensor measurements yield predicted stiffness reductions of 10.46%, 14.97%, and 14.96% for elements No. 2, 5, and 9, respectively. Some false positive

damages are detected in both single- and multiple-element damage cases. However, the results show that the proposed approach enables good damage identification using a small number of sensors.

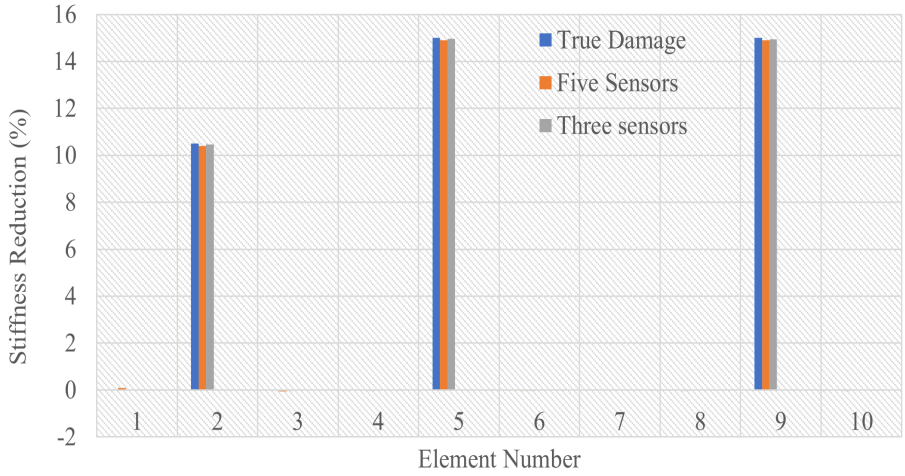



Figure 5.16 Damage identification for three-element damage case (scenario 1)

Figure 5.17 - 5.19 show the damage identification results for Scenario 2 with a 10% noise level. The damage prediction results are like those of true stiffness reduction. Some false positives with higher values than those in Scenario 1 are observed. The true damage in element No. 6 is 5%. The prediction results demonstrate that using five sensors can provide stiffness reduction close to the true damage. However, it is observed that using three sensors can also give entirely accurate stiffness reduction, and very few false identification results can be obtained.

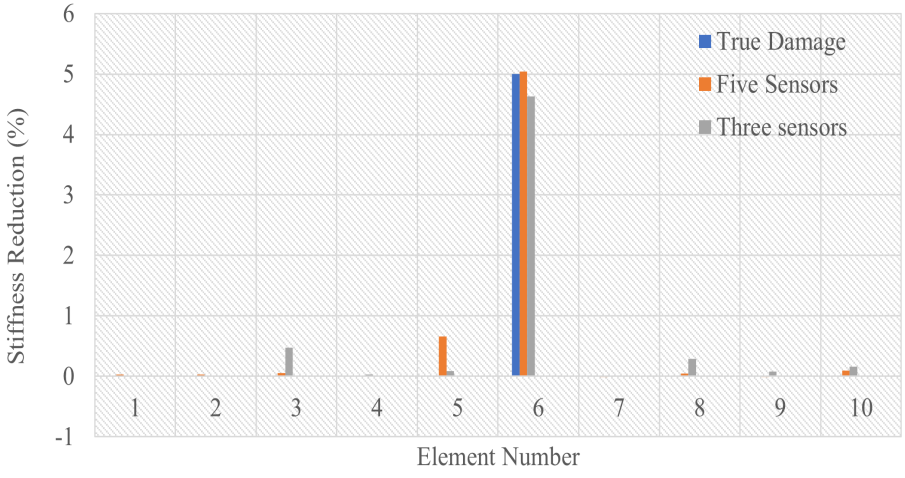



Figure 5.17 Damage identification for single element damage case (scenario 2)

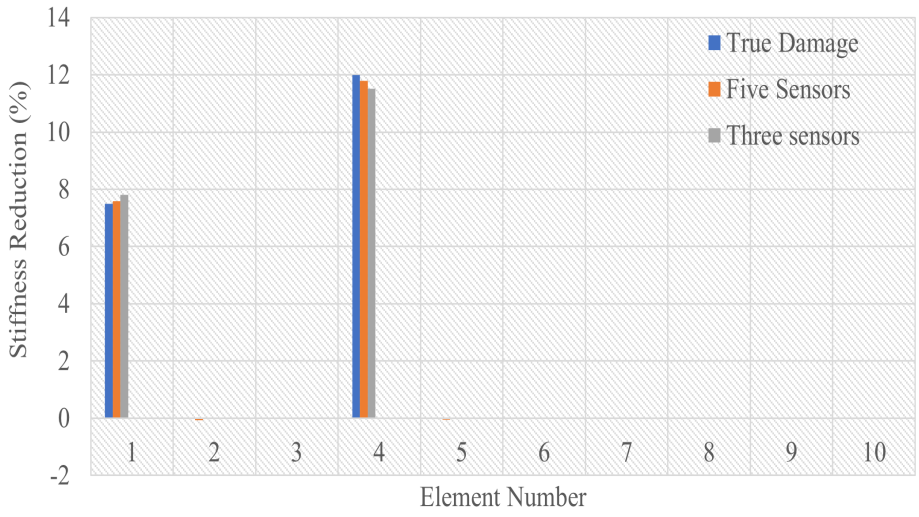



Figure 5.18 Damage identification for two-element damage case (scenario 2)

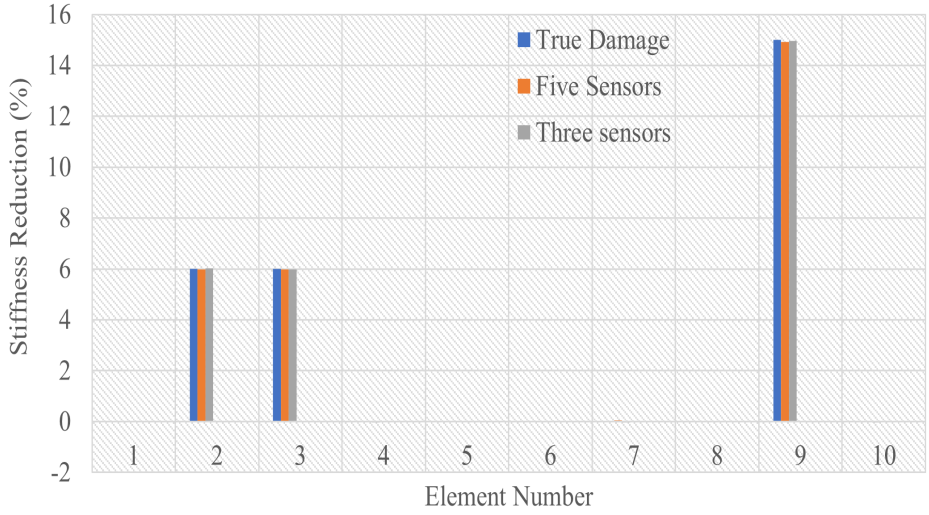



Figure 5.19 Damage identification for three-element damage case (scenario 2)

As in scenario 1, the prediction results for the multi-element pattern are like single-element damage cases. The prediction results are close to the true damage using either five or three sensors. Figure 5.20 – 5.22 show the damage identification examples for both single- and multiple-element damage cases in scenario 3. As in the case of five sensors, the prediction results show a similar pattern, and the results are pretty close to those of the true damage case with some false positive damages in the undamaged elements.

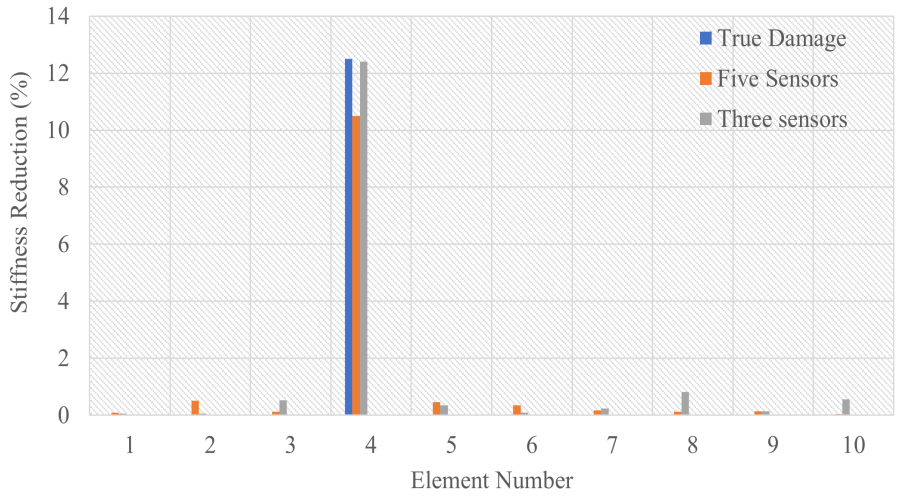



Figure 5.20 Damage identification for single element damage case (scenario 3)

Figure 5.20 shows the stiffness reduction in the elements for the single-element damage case. The true damage is in element No. 4. The prediction results in element No.4 using three sensors look closer to true damage than the five sensors. Still, it is observed that using three sensors gives higher value of false positive in undamaged elements. The damage identification result for multi-element damage cases shows good damage identification results using a small number of sensors. It can be observed in both Figures 5.21 and 5.22 that the damage prediction results are close to actual damage using both five and three sensors.

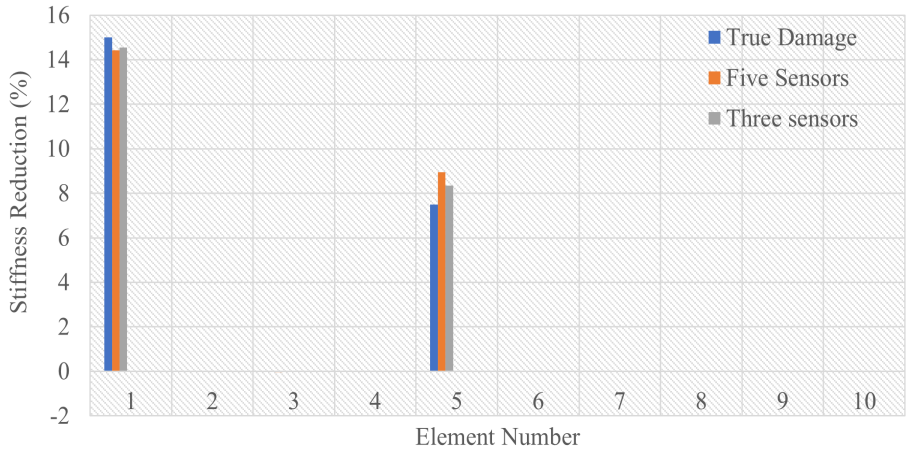



Figure 5.21 Damage identification for two-element damage case (scenario 3)

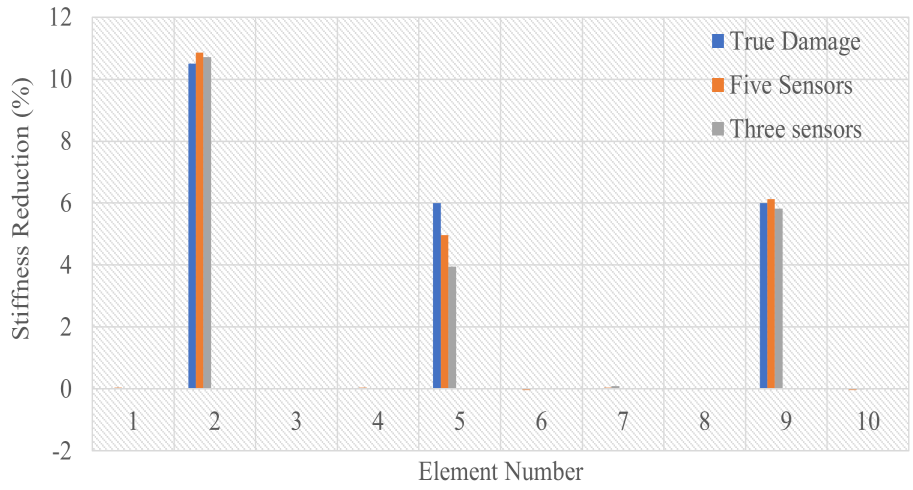



Figure 5.22 Damage identification for three-element damage case (scenario 3)

Figures 5.23–5.25 shows the damage identification results for scenario 4 with 10% noise for single-, two-, and three-element damage cases. As observed in previous scenarios, the damage prediction results are close to true damage, or the results obtained using five sensors. Some false positive damages are observed in the undamaged elements, but the damages are less than 1%, which is negligible.

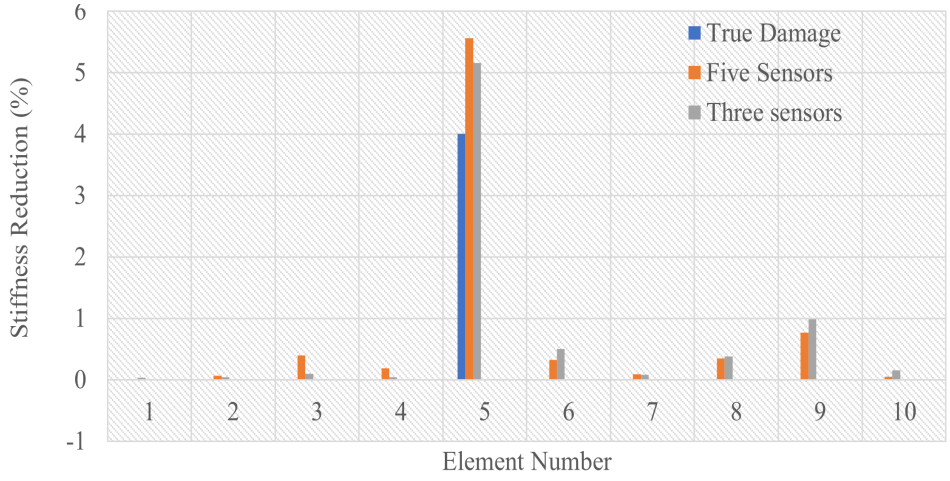



Figure 5.23 Damage identification for single element damage case (scenario 4)

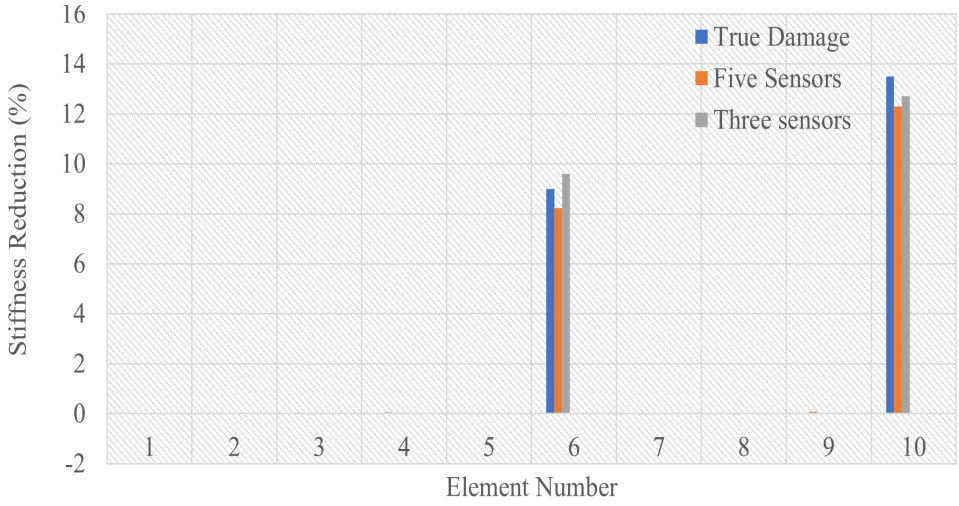



Figure 5.24 Damage identification for two-element damage case (scenario 4)

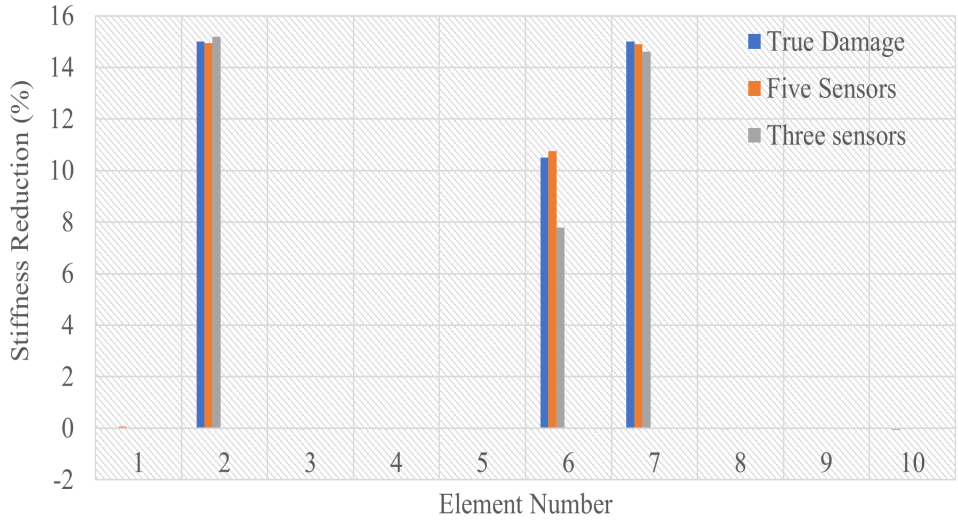



Figure 5.25 Damage identification for three-element damage case (scenario 4)

The model herein implements the same network architecture and pre-processing data methods using five sensors. The damage identification results indicated that the developed model could provide good damage identification in structural elements using a small number of sensors. The identification results show that even reduced number of sensors, the performance of the proposed approach is not significantly affected when the measurement noise and uncertainty in the stiffness parameter are considered. Further, the model is tested with one sensor, but the performance is not good. Therefore, the number of sensors in a practical application can be decided by carrying out a numerical study. The measurements from the FEM analysis can be used to train and test the model using a different number of sensors.

## 5.5 Conclusion

The performance measurement for the proposed structural elemental damage identification approach using IRF and LSTM autoencoder indicated good damage identification and outperformed the traditional machine learning model proposed in chapters 3 and chapter 4. The computation time for the proposed model is reduced by performing PCA. The reconstruction network after tuning can give good reconstruction results, and the same internal vector representation of the input sequence is used for damage identification. The results for all the four scenarios considered are close to true damage. Further, the model is trained with one dataset and tested with another scenario's dataset. The performance demonstrated that the model could generalise quite well. The study considers the noise measurement and uncertainty in stiffness parameters. The performance of the proposed method is also measured, reducing sensor numbers from five to three. The damage identification results demonstrated that with three sensors, enough information can be obtained with the proposed method to give good damage prediction in structural elements. Tables 5.4 and 5.5 shows that there is no significant difference in MSE and R-Squared using five sensors and three sensors.

The damage is defined in terms of stiffness reduction in structural elements considering only noise and uncertainties in the stiffness parameter. A large-scale infrastructure like a bridge can be exposed to several external conditions such as vehicle loads and weather. It can also experience different types of damage and different levels for each damage type. The proposed multioutput regressions models may not be suitable or can become too complex due to identifying the damage and then quantifying the damage when the data available is insufficient. The next chapter of this thesis considers a real civil infrastructure and proposes a suitable machine learning model.

# CHAPTER 6

# 6 Structural Damage Classification of a Large-scale Bridge using 1D Convolutional Neural Network and Time Domain Responses

## 6.1 Introduction

Convolutional neural networks are widely used in SHM due to their capability of efficient and robust feature learning (Wang et al., 2020). It is also commonly used in SHM (Abdeljaber et al., 2017; Khodabandehlou et al., 2019). 1D-CNN networks have the advantage over the other CNN work in terms of computational complexities and shallow architecture; this makes the network easy to train and implement. Further, 1D-CNN does not require high system configuration, making it suitable for real-time and low-cost applications (Mitiche et al., 2020). This chapter proposes a 1D-CNN for the damage of a large-scale civil structure considering several damage scenarios using fewer time domain responses from the Z24 bridge. The measurements made from the Z24 bridge in Switzerland during the short-term progressive damage test (PDT) are used for the study. The PDT took over a month. Two hundred ninety-one degrees of freedom have been measured for both forced and ambient excitation. The study considers only the measurement under ambient excitation, which has the advantage over forced excitation of not needing to arrange the excitation for the measurement. Further, to reduce the overall cost of monitoring the structure, the study attempts to use a small number of sensor measurements. The performance of the proposed method is measured in terms of accuracy supported by a confusion matrix. It is compared with the existing study conducted on Z24 by Hung et al. (2020).

## 6.2 Proposed 1D-CNN

The acceleration responses from three sensors are pre-processed and used as input to the proposed 1D-CNN. The number of sensor measurements used is less than that in the previous study (Hung et al., 2020) on the Z24 bridge. In addition, no data augmentation is performed, and only real measurement data are used for training and validation. The data pre-processing technique is straightforward, involving autocorrelation, normalisation and splitting into smaller lengths. Figure 6.1 shows the basic architecture of 1D-CNN. It has building layers, namely, convolutional, pooling, flattening and fully connected neural networks.

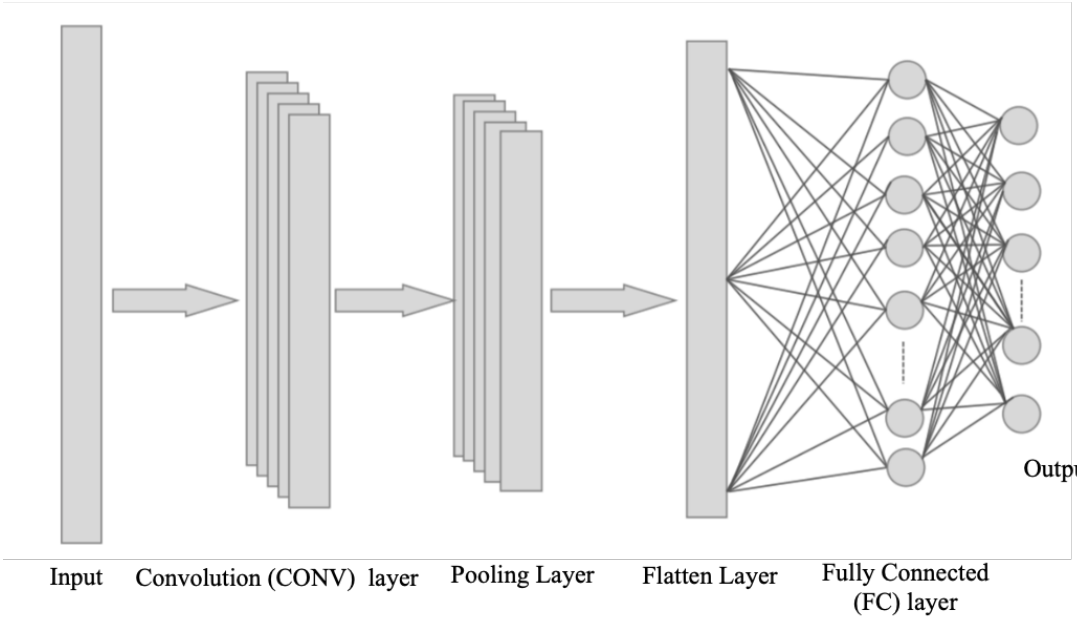



Figure 6.1 Convolution Neural Network

## 6.3 Convolution layer

The convolutional layer is the core layer of the CNN, and it performs convolution operations on the input data (Wang et al., 2020). Convolution is performed on the input dimension and involves dot product multiplication between the input data and the filters or kernel, with an array of weights (Xu et al., 2018). The kernels have a small receptive field, but the operation occurs at different points on the input. Each filter produces an activation or feature map by convolving across the input volume. The dimension of the activation map depends on the type of CNN. A CNN can learn multiple features using multiple filters, which allows the network to be equipped with different means of extracting features from the input. Multiple convolutional layers can be employed depending on the problem to be solved. This allows a hierarchical decomposition of the raw input and the extraction of higher-level features.

The convolution layer has the activation function as its last component to increase non-linearity in the output and is obtained through a nonlinear activation function. Linear activation functions are easy to train but are challenging to be used to learn complex mapping functions. Nonlinear activation functions allow the nodes to remember more complex mapping functions. Three widely used nonlinear activation functions are the sigmoid, hyperbolic tangent (*'tanh'*) and rectified linear unit (RELU). The sigmoid function transforms the input into a value between zero and one. The *'tanh'* function

converts the input to values between -1 and 1. The *'sigmoid'* and *'tanh'* functions become saturated since the input values are between 1 and 0 or -1, respectively. RELU outputs positive values when inputs are positive and outputs zero otherwise. The RELU function is nearly linear for input values greater than zero, which makes it easy to train the model. However, it is a nonlinear function since the negative output values are always taken as zero.

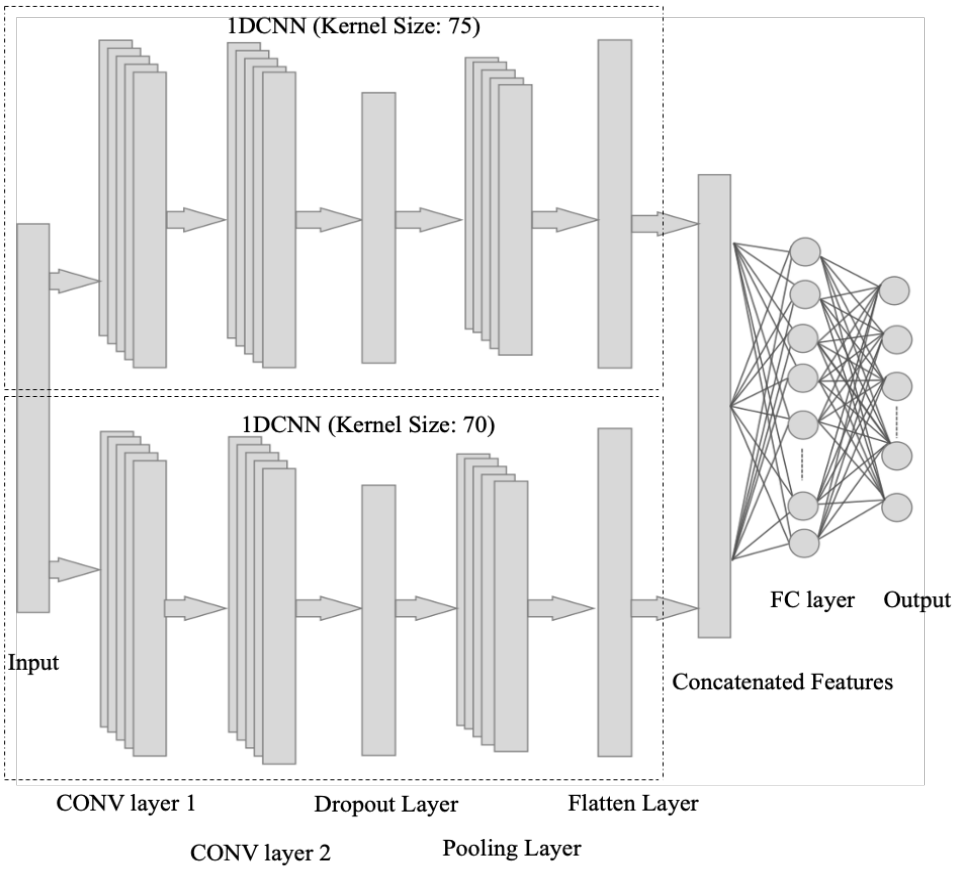



Figure 6.2 The proposed 1D-CNN

Figure 6.2 shows the structure of the proposed 1D-CNN model. The proposed 1D-CNN framework consists of two 1D-CNN networks with the same number of filters but different kernel lengths. Using different kernel lengths can interpret input data at various resolutions. Each 1D-CNN network comprises two convolutional layers with the same number of filters. A total of 256 filters are used in each 1D-CNN, allowing 256 different kernel models to extract features from pre-processed data. The length of the 1D convolution window of the first 1D-CNN is 75, and that of the other 1D-CNN is 70 and is selected based on the rial-and error; increasing the length further does not improve the performance while decreasing degrades the performance. A stride length

of one is used for the proposed framework. The convolution output is obtained through a RELU activation function to introduce nonlinearity.

### 6.4 Pooling layer

The pooling layer reduces the dimensionality of a given mapping, consolidating the prominent features, and is performed after the convolution layer (Xu et al., 2018). Pooling is performed on each feature map independently and reduces the number of parameters and computational load. Different types of pooling methods are used in CNNs, such as max pooling, average pooling, and global pooling, as shown in Figure 6.3. When using the max pooling, the maximum element is selected from the region covered by the pooling window. Average pooling takes the average of the elements covered by the pooling window, and global pooling reduces each channel to a single element. Max pooling provides the most prominent feature and is one of the most common pooling methods used in CNNs. This method leaves the maximum activation in the rectangular neighbourhood (Goodfellow et al., 2016).

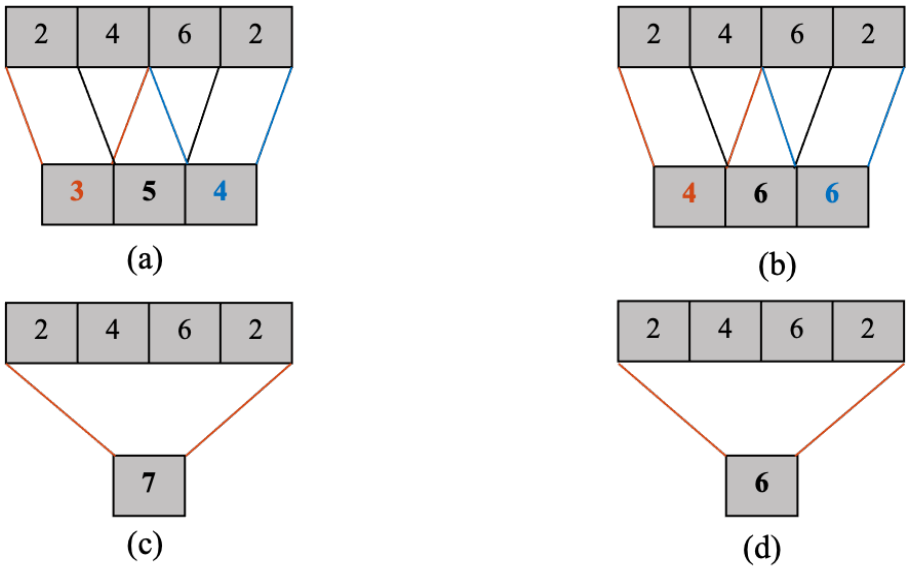



Figure 6.3 (a) Average Pooling (b) Max Pooling (c) Global Average Pooling (d) Global Max Pooling

### 6.5 Flatten layer and fully connected component

Two main processes in a CNN are feature learning and classification. The convolutional and pooling layers learn the feature representation of the datasets. The classification is performed by the flattening layer and fully connected layer. The

pooling layer's output, a pooled feature map, is converted into a long one-dimensional vector by the flattening layer, making it suitable for the subsequent fully connected layer. The next layer is a fully connected artificial neural network. It can have several densely connected layers to learn the nonlinear properties of the feature representation. However, it cannot use multi-dimensional data. Therefore, flattening layers are incorporated before the fully connected artificial neural network. Activation functions, such as the sigmoid or SoftMax function, can be used in a fully connected layer to obtain the probabilities of the classes. It is noted that SoftMax is used in this study.

### 6.6 Regularisation

Deep neural networks face generalisation challenges. A neural network with a strong capacity may cause overfitting, whereas a network with limited power cannot learn the relationship effectively. The number of neural network nodes, the used layers of the network and the weights of parameters define the network's capacity. A good model should learn the mapping relationship from the training dataset and generalise well on the testing dataset. Underfitting problems can be addressed by varying the network structure, such as increasing the numbers of network nodes and layers (Brownlee, 2019b). Using a large dataset and changing the complexity of the network can reduce the overfitting problem in deep learning networks. The network complexity can be altered by changing the network structure and the parameters (Bishop, 1995). Regularisation can also be used to reduce overfitting and provide better overall performance. There are many regularisation methods, such as weight regularisations ($L_1$ regularisation, $L_2$ regularisation and soft weight sharing) (L1, L2 and soft weight sharing), (Nowlan & Hington, 1992), activity regularisation, weight constraints, dropout, make robust with noise, and stop training at the right time with early stopping. Each method aids the network in generalisation performance, configuration, and computational complexity (Zhang et al., 2019). Training several models and averaging the outputs improve the network's performance and help prevent overfitting. However, training different networks or architectures is computationally expensive since the optimal hyperparameters must be determined for each network (Srivastava et al., 2014).

Srivastava et al. (2014) demonstrated that a restricted Boltzmann machine (RBM) performed better with a dropout layer, and experimental studies are conducted with

neural networks for classification problems on different datasets. Dropout is defined as the temporary deactivation of some nodes in neural networks. This includes incoming and outgoing connections (Brownlee, 2019b; Xu et al., 2018). Figure 6.4 shows a thinned network resulting from applying the dropout technique. Dropout can be used to feed-forward networks and graphical models.

A good dropout value in the hidden layer is between 0.5 and 0.8 for the input layers, where 1.0 represents no dropout and 0 means no output from the layer. 1D-CNNs learn rapidly from a dataset, which can make the model prone to overfitting. In this study, a dropout layer is used for regularisation to reduce overfitting and improve the generalisation of the proposed framework. Thus, some of the outputs are dropped during the training process, as shown in Figure 6.4. The best method to determine the optimal dropout value is to perform a grid search between 1.0 and 0.1. Dropout layers can be used after the convolutional layers or the pooling layer. In the proposed framework, the dropout layer is included after the convolutional layers, as shown in Figure 6.2, since its placement after the pooling layer or convolution layers does not affect the performance of the classification problem considered here classification problem's performance. A dropout layer is used in this study with the Keras application programming interface (API). The definition of the dropout rate discussed previously is different in the Keras API. The dropout rate of 0.2 means that 20% of data are set to zero and 80% of the input is retained. The proposed model is trained with 90% of the dataset and tested with the remaining 10%. A mini-batch gradient descent of 128 is used and trained, and testing is carried out using 100 epochs. An epoch means that the proposed algorithm runs through the dataset once.

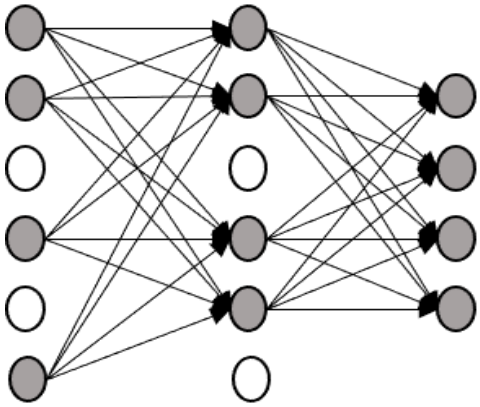


Figure 6.4 Dropout Technique

## 6.7 Experimental Verifications on Z24 Bridge

The performance of the proposed method for the classification of damage scenarios is evaluated using real bridge data from the classical post-tensioned two-cell box-girder Z24 bridge in Canton Bern, Switzerland. The bridge is part of the road connecting Koppigen and Utzenstorf, passing over the A1 highway between Bern and Zurich. This bridge has three spans, a middle span of 30 m and two lanes, as shown in Figure 6.5 (Kramar et al., 1999). It has an overall length of 60 m and rests on four pillars. More detailed descriptions of the Z24 bridge and the tests are provided in a previous study by (Roeck, 2003).

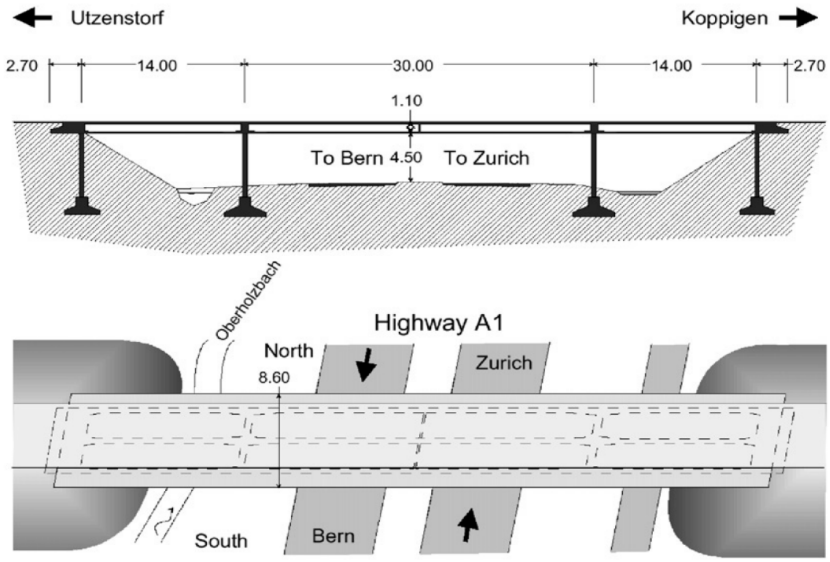



Figure 6.5 Z24 Bridge (Kramar et al., 1999)

## 6.8 On-site test descriptions

The bridge was tested under the European Brite EuRam research project BE-3157, “System Identification to Monitor Civil Engineering Structures” (SIMCES). Dynamics tests were conducted on this bridge to verify the feasibility of vibration-based health monitoring and civil engineering infrastructure damage detection. The measured vibration data from full-scale, long-term, and progressive failure tests of this bridge structure are used for validating the accuracy of the proposed approach for damage detection. Several damage scenarios are considered. Table 6.1 lists the types of damage introduced in sequence on the bridge, which required over a month to complete. The test setups and safety considerations can be found in previous reports (Kramar et al., 1999; Roeck, 2003).

Table 6.1 Types of damage scenarios

| Sl. No | Damage Scenario | Units |
|---|---|---|
| 1 | Settlement of foundation | m |
| 2 | Tilt of foundation | degree |
| 3 | Spalling of concrete of soffit | $m^2$ |
| 4 | Landslide | $m^3$ |
| 5 | Failure of concrete hinges | pieces |
| 6 | Failure of anchor heads | pieces |
| 7 | Failure of post tensioning tendons | Numbers |

The complete damage scenarios and levels introduced in the progressive damage tests are listed in Table 6.2. Vibration measurement data are obtained from ambient and forced vibration conditions for the introduced damage scenarios with different damage types. The vibration response measurements under ambient conditions are used in the present study.

Vibration measurements in the vertical, transverse, and longitudinal directions are collected from a $2 \times 8$ grid on each of the two columns and a regular $3 \times 45$ grid on the top deck. Testing data are recorded in nine setups, owing to the limitation of the number of accelerometers and data acquisition channels. Vibration responses are collected at a sampling rate of 100 Hz using a 30 Hz cut-off frequency antialiasing filter. Each response measurement comprises 65,536 samples. Five sensors, namely, R1-V, R2-T, R2-V, R2-L and R3-V, of the 33 total sensors are commonly used between the test setups. R1, R2 and R3 indicate the locations of the sensors, and V, T and L denote the vertical, transverse, and longitudinal direction measurements, respectively.

Hung et al. (2021) proposed a damage classification framework using a hybrid deep learning model consisting of a 1D-CNN and LSTM network. Three different techniques extracted features: an autoregressive model, discrete wavelet transform, and empirical mode decomposition. These features were then fused and used as inputs to the hybrid model. The developed framework was tested on the Z24 bridge for

damage classification considering the first 16 damage scenarios listed in Table 6.2. The performance of the proposed approach will be compared with the existing study.

Table 6.2 Complete damage scenarios in the progressive damage tests

| Sl. No | Damage Scenario | Sl. No | Damage Scenario |
|---|---|---|---|
| 1 | Undamaged Condition | 10 | Spalling of concrete of soffit, 24 $m^2$ |
| 2 | Installation of pier settlement system | 11 | Landslide of 1 m at abutment |
| 3 | Lowering of pier, 20 mm | 12 | Failure of concrete hinge |
| 4 | Lowering of pier, 40 mm | 13 | Failure of 2 anchor heads |
| 5 | Lowering of pier, 80 mm | 14 | Failure of 4 anchor heads |
| 6 | Lowering of pier, 95 mm | 15 | Rupture of 2 out of 16 tendons |
| 7 | Lifting of pier, tilt of foundation | 16 | Rupture of 4 out of 16 tendons |
| 8 | New reference condition | 17 | Rupture of 6 out of 16 tendons |
| 9 | spalling of concrete at soffit, 12 $m^2$ | | |

## 6.9 Data Pre-processing

This study aims to obtain better classification results using a smaller number of sensors than the existing study. In the proposed framework, three vertical measurements, R1-V, R2-V, and R3-V, common sensors in nine setups, are used. The nine test setups and sensor locations are shown in Figure 6.6 (Kramar et al., 1999).

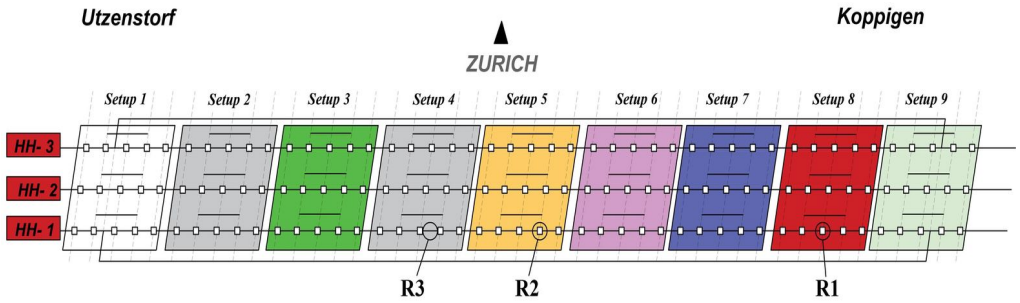



Figure 6.6 Test setup and sensor measurement location field (Hung et al., 2021)

Each time series signal has 65536 sample points. The raw measurements are detrended first and then processed further. The detrended measurements for the three selected sensors are shown in Figures 6.7, 6.8, and 6.9, respectively.

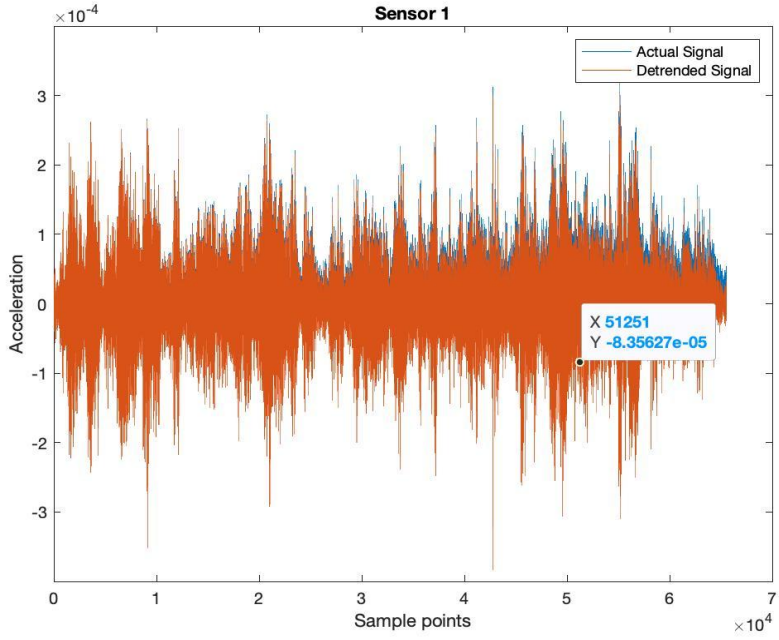



Figure 6.7 R1-V Measurement

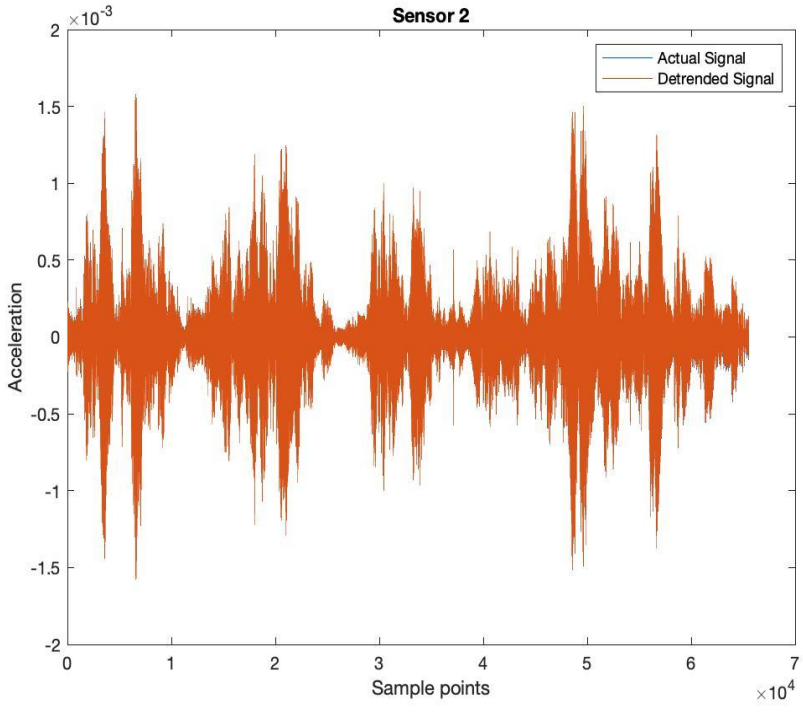



Figure 6.8 R2-V Measurement

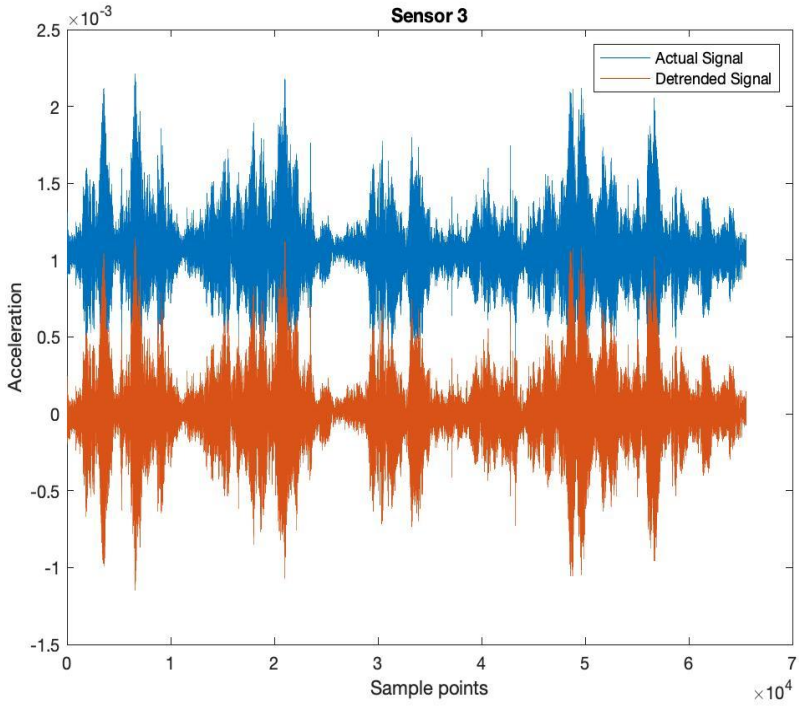
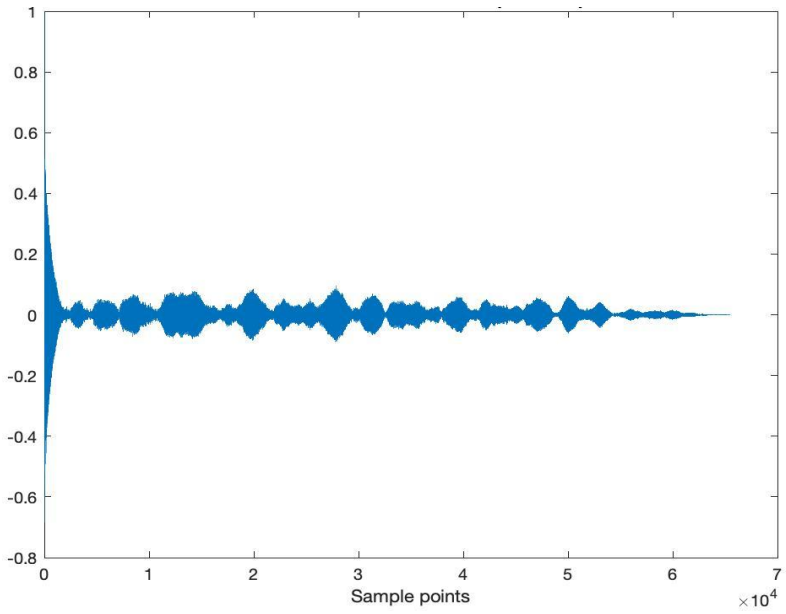
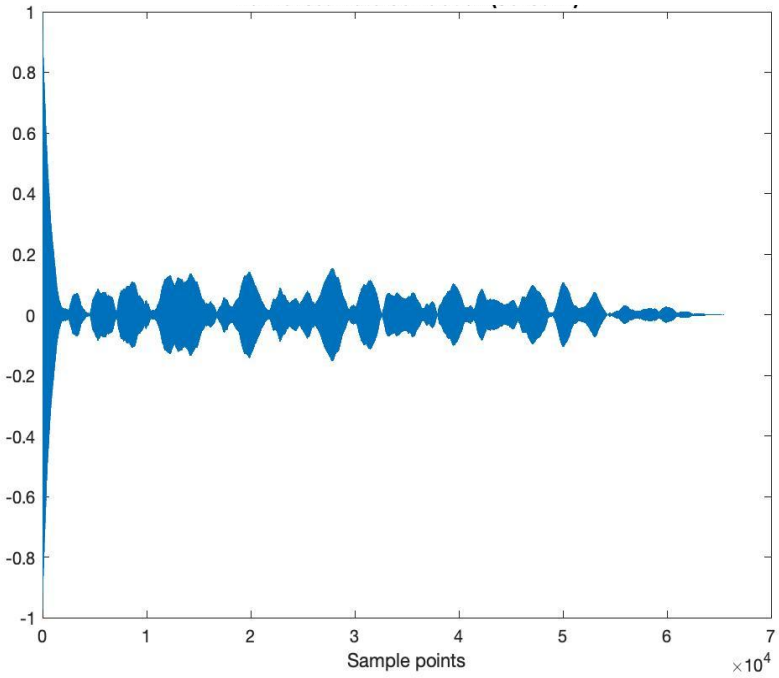



Figure 6.9 R3-V Measurement

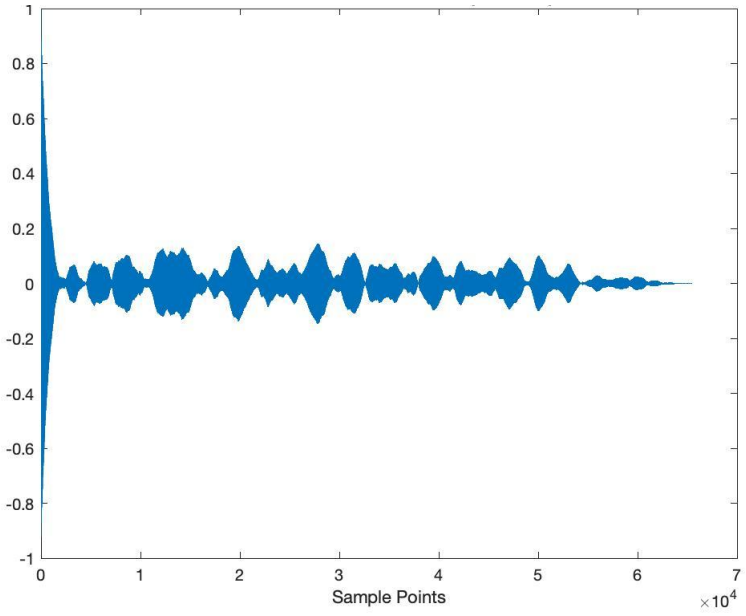
Autocorrelation is calculated for the detrended signal and is normalised. The autocorrelation function helps reduce the noise in the sensor measurements and is much faster. Figures 6.10 (a), 6.10 (b), and 6.10 (c) are the normalised signal for R1-V, R2-Vand R3-V, respectively.




(a)




(b)

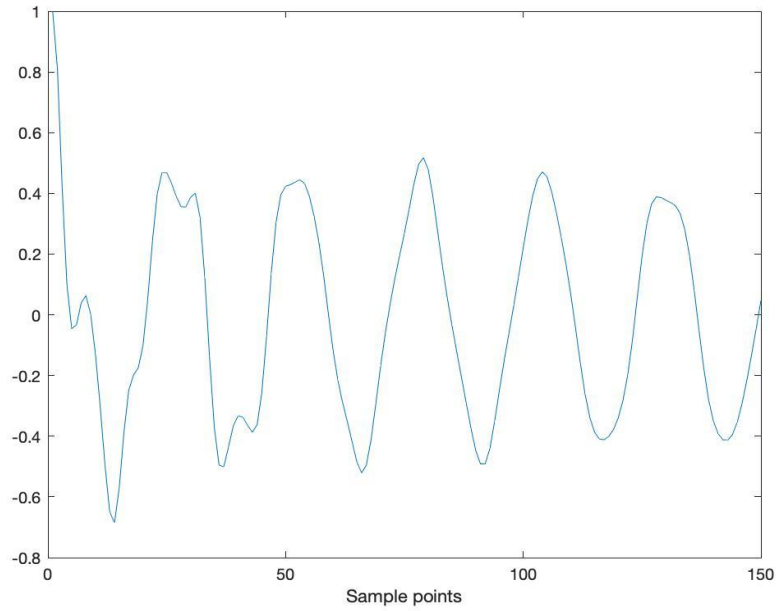



(c)

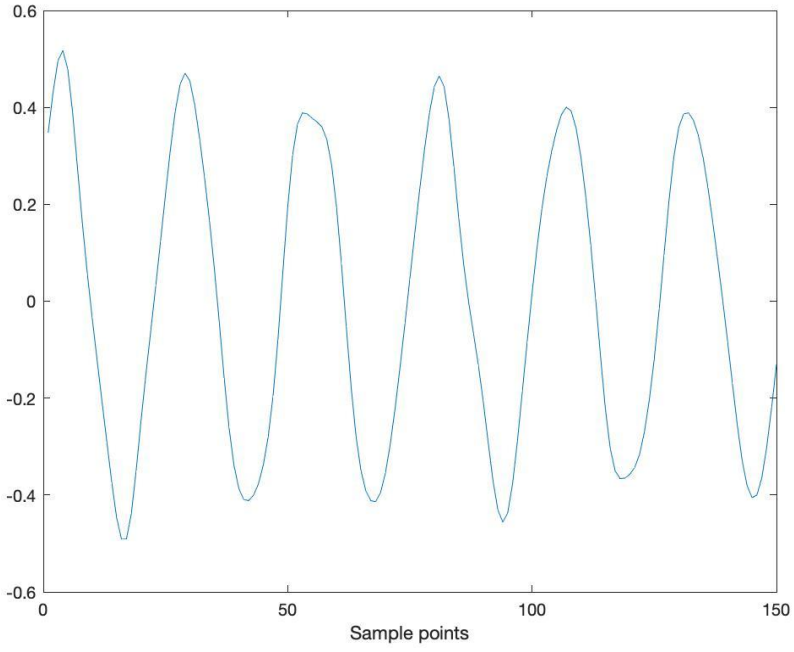
Figure 6.10 Normalised Autocorrelation Signal (a) R1-V (b) R2-V and (c) R3-V

The autocorrelation measures the correlation between the samples at points $x_t$ and $x_{t+k}$, where $k$ denotes the time lag ($k$= *0, 1, 2, K*), considering that $x_t$ is a stochastic process. Further, each of the normalised signals is split into smaller lengths of 150 sample points with an overlap of 75 sample points in the previous and next segmented

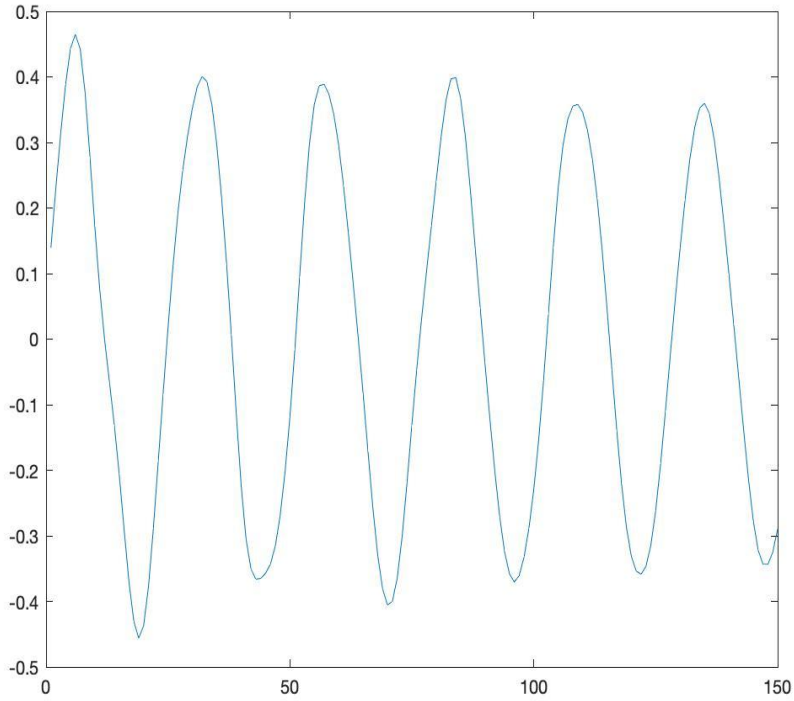
signals, i.e., 50% overlap. The first three splits of R1-V for Damage Scenario 2 are shown in Figure 6.11.




(a)




(b)




(c)

Figure 6.11 First three samples of R1-V measurement after split: (a) first 150 sample points, (b) second 150 sample points with 50% overlap with the first sequence; (c) third 150 sample points with 50% overlap with the second sequence

The measurement data from all three sensors for each damage case are grouped into a single three-dimensional array. The measurement data with shape *[Samples, Sample points, Number of sensors]* is used to train and test the proposed 1D-CNN model for damage classification. The data split of 90% for training and 10% for testing is selected to compare the proposed approach with the existing study by Hung et al. (2020).

The performance measurement is performed for three cases:

1. Case 1: Autocorrelation is calculated over the entire sequence length of raw acceleration response. The data is pre-processed and split into smaller lengths of 150 sample points with a 50% overlap. There are 5400 samples generated from one sensor measurement for each damage scenario making 86,400 samples in total for 16 damage scenarios.
2. Case 2: From case 1, only 50% of a sample size of each damage scenario is considered, that is 2700 for each damage scenario which makes a total sample size of 43,200.

3. Case 3: Considered only 50% length of the raw acceleration response to calculate autocorrelation used in case 1 and case 2. For each damage scenario, 2700 samples are used for the study.

Figure 6.12 shows the architecture of the proposed framework, including the dimensions of the input and output of each layer. The performance is measured for the above three cases and is presented in the results section.

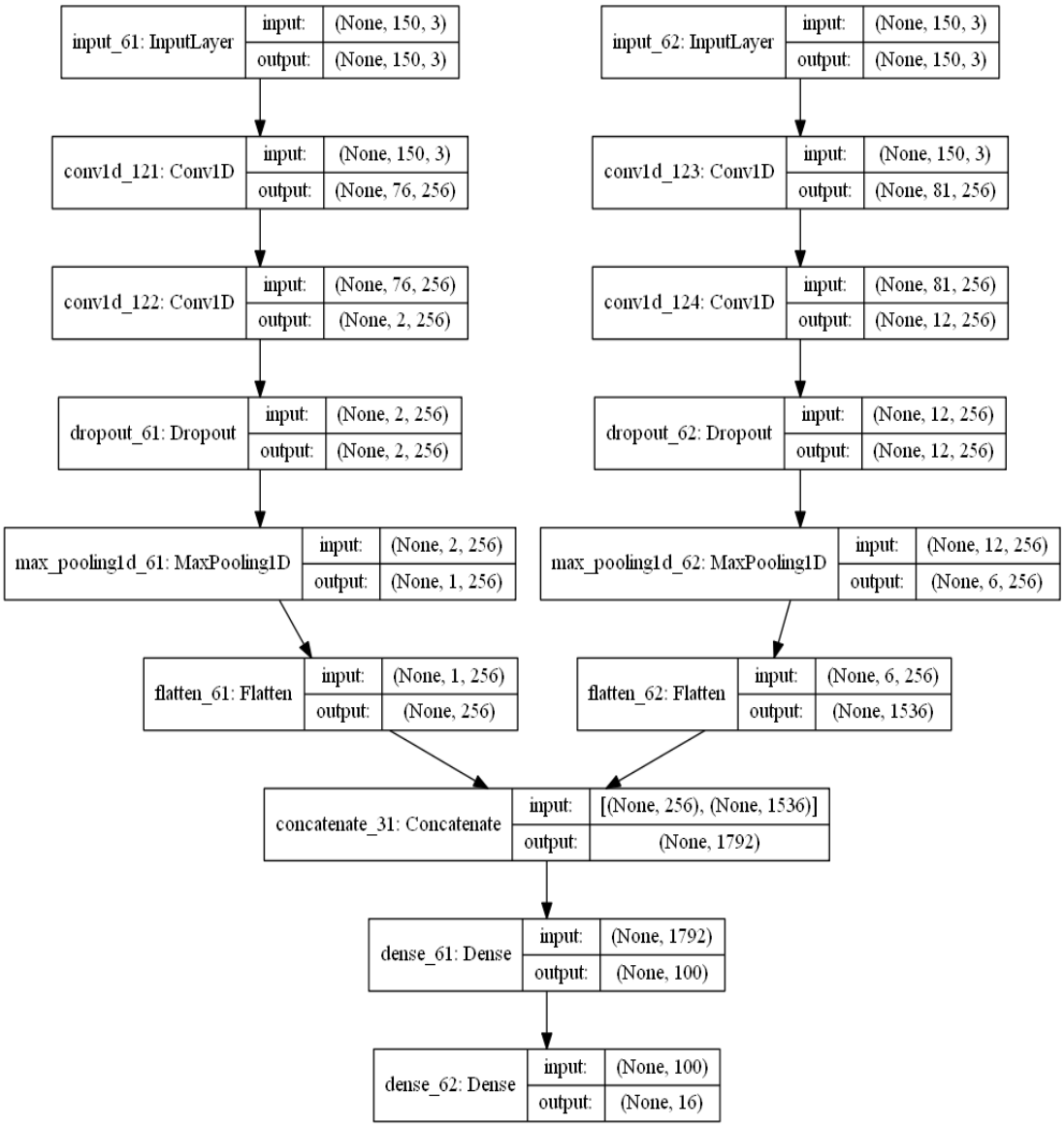



Figure 6.12 The proposed 1D-CNN architecture

## 6.10 Results and Discussions

The performance of the proposed framework is measured in terms of the accuracy supported by the confusion matrix, precision, recall, and F-1 score. The accuracy

metric indicates the number of classes that are correctly predicted. With more classes considered, the accuracy alone cannot define the model's performance well, especially when there are unequal samples for different classes. The confusion matrix can describe the overall performance of the proposed approach for the damage classification. It can be used to determine a classifier's performance on the test data set for which true values are known. Precision, recall, and F-1 score are also calculated to indicate and measure the accuracy and performance of the proposed approach.

### 6.10.1 Case 1

The proposed framework is fine-tuned to obtain good results. A total of 256 filters or kernels are used in the 1D-CNN networks, with kernel sizes of 70 and 75 in the two 1D-CNNs. A damage classification accuracy of 93.65% is achieved with the proposed framework, which is 3.55% higher than the previous damage classification results field (Hung et al., 2020). The training time by using the proposed approach is 4 hours and 42 min with a computer of an Intel® Core (TM) i7-0750H, a graphical card Nvidia RTX2070.

It can be seen in the confusion matrix presented in Figure 6.13 that there are some false positive damage classifications. These are observed more frequently in the last four damage scenarios. Damage scenarios 13 and 14 represent the failures of two and four anchor heads, respectively. In Damage Scenario 13, 679 out of 809 samples are classified as true positives, and 106 samples are classified as Damage Scenario 14, a similar damage type. It can be seen that the prediction accuracy for Damage Scenario 13 is only 83.9%. However, 13% of the samples are classified as Damage Scenario 14.

Similarly, for Damage Scenario 14, the accuracy is 89.9%, and 7.4% is classified as Damage Scenario 13. Similar results are observed for Damage Scenarios 15 and 16, which are similar damage types. The proposed model can provide a good overall accuracy, and many false positives are observed in the same kind of damage scenarios.

| Actual | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | Total Samples |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **813** | 20 | 6 | 5 | 0 | 0 | 0 | 3 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 851 |
| | **2** | 7 | **818** | 1 | 20 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 5 | 0 | 854 |
| | **3** | 2 | 1 | **797** | 0 | 5 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 809 |
| | **4** | 4 | 38 | 1 | **753** | 1 | 0 | 0 | 3 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 | 803 |
| | **5** | 0 | 0 | 6 | 2 | **799** | 1 | 1 | 3 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 813 |
| | **6** | 0 | 0 | 0 | 0 | 1 | **774** | 12 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 788 |
| | **7** | 0 | 0 | 1 | 0 | 0 | 16 | **733** | 3 | 1 | 4 | 3 | 1 | 1 | 0 | 0 | 0 | 763 |
| | **8** | 2 | 0 | 5 | 2 | 4 | 2 | 7 | **767** | 2 | 12 | 11 | 0 | 0 | 1 | 0 | 0 | 815 |
| | **9** | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 6 | **819** | 19 | 9 | 4 | 0 | 1 | 0 | 0 | 859 |
| | **10** | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 4 | 9 | **760** | 6 | 2 | 0 | 0 | 2 | 0 | 785 |
| | **11** | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 8 | 1 | 11 | **744** | 2 | 0 | 1 | 0 | 0 | 774 |
| | **12** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | **750** | 11 | 23 | 5 | 12 | 803 |
| | **13** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | **679** | 106 | 11 | 9 | 809 |
| | **14** | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | 59 | **718** | 14 | 0 | 798 |
| | **15** | 1 | 4 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 12 | 18 | 42 | **703** | 39 | 821 |
| | **16** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 3 | 16 | 6 | 8 | 37 | 742 | 815 |
| | | Predicted | | | | | | | | | | | | | | | | |

Figure 6.13 Confusion Matrix for case 1

Table 6.3 lists the per class precision, recall and F-1 scores for the damage scenarios in Case 1. The ability of the classifier to avoid labelling a negative sample as positive is defined by precision, and its ability to classify all positive samples as positive is reflected in the recall. For an ideal classifier, both precision and recall are 1. The *F-1* score is the harmonic mean of the accuracy and recall and can be expressed as

$$F1 = 2 * \frac{(P*R)}{(P+R)} \tag{6.1}$$

where $P$ and $R$ are the precision and recall, respectively. F-1 score is 1 for a good ideal classifier which can be achieved if precision and recall are 1.

For a class, precision is given as

$$P = \frac{T_p}{T_p+F_p}, \tag{6.2}$$

The recall is calculated as follows

$$R = \frac{T_P}{T_P+F_N} \tag{6.3}$$

where $T_P$, $F_P$ and $F_N$ are true positive, false positive and false negative, respectively.

Table 6.3 Precision, Recall and F-1 score for case 1

| Class | Precision | Recall | F-1 Score |
|---|---|---|---|
| 1 | 0.98 | 0.96 | 0.97 |
| 2 | 0.93 | 0.96 | 0.94 |
| 3 | 0.97 | 0.99 | 0.98 |
| 4 | 0.96 | 0.94 | 0.95 |
| 5 | 0.99 | 0.98 | 0.98 |
| 6 | 0.97 | 0.98 | 0.98 |
| 7 | 0.97 | 0.96 | 0.96 |
| 8 | 0.96 | 0.94 | 0.95 |
| 9 | 0.98 | 0.95 | 0.97 |
| 10 | 0.94 | 0.97 | 0.95 |
| 11 | 0.96 | 0.96 | 0.96 |
| 12 | 0.94 | 0.93 | 0.94 |
| 13 | 0.88 | 0.84 | 0.86 |
| 14 | 0.8 | 0.9 | 0.84 |
| 15 | 0.9 | 0.86 | 0.88 |
| 16 | 0.92 | 0.91 | 0.92 |

For the multiclass problem, as in the study with 16 classes, the macro-average F-1 score with the proposed framework is 0.939. The macro-average precision and recall are 0.940 and 0.939, respectively. For case 1, it is observed that the proposed method can provide good damage classification.

### 6.10.2 Case 2

An accuracy of 92.03% is achieved with 50% fewer data than in case 1. The accuracy is decreased by 1.62%. However, it is still higher than that previous study by 1.93%. In addition, the training time using the same computer configuration is 2 hours and 20 minutes. Similar patterns of classification results are observed as in Case 1. Most false positives are attributed to identical damage scenarios for the last four damage scenarios. Figure 6.14 shows the confusion matrix obtained in Case 2.

| Actual | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | Total Samples |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | **360** | 10 | 3 | 8 | 0 | 0 | 0 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 384 |
| | 2 | 1 | **361** | 2 | 32 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 402 |
| | 3 | 3 | 0 | **374** | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 0 | 398 |
| | 4 | 0 | 18 | 0 | **406** | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 6 | 1 | 433 |
| | 5 | 0 | 1 | 4 | 2 | **379** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 386 |
| | 6 | 0 | 0 | 0 | 0 | 0 | **391** | 1 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 395 |
| | 7 | 2 | 0 | 0 | 0 | 0 | 17 | **417** | 3 | 1 | 0 | 3 | 1 | 0 | 0 | 0 | 0 | 444 |
| | 8 | 5 | 0 | 0 | 2 | 1 | 2 | 4 | **387** | 2 | 3 | 0 | 1 | 0 | 0 | 0 | 2 | 409 |
| | 9 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | **396** | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 408 |
| | 10 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 7 | 9 | **355** | 4 | 2 | 0 | 0 | 0 | 0 | 380 |
| | 11 | 0 | 1 | 0 | 0 | 0 | 2 | 0 | 12 | 2 | 12 | **423** | 1 | 0 | 0 | 0 | 2 | 455 |
| | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | **406** | 10 | 7 | 6 | 5 | 435 |
| | 13 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | **329** | 41 | 7 | 7 | 386 |
| | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 42 | **320** | 8 | 3 | 377 |
| | 15 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 8 | 32 | **340** | 29 | 410 |
| | 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 8 | 1 | 13 | **352** | 378 |
| | | Predicted | | | | | | | | | | | | | | | | |

Figure 6.14 Confusion Matrix for case 2

Table 6.4 lists the precision, recall and F-1 scores for each damage scenario. The F-1 score is 0.925. Therefore, the proposed framework can still provide good damage classification results using 50% of the datasets. The difference in accuracy is mainly due to the size of the used datasets. The proposed framework tends to learn better with more datasets, which is understandable. However, there is a trade-off between the accuracy and computational demand. The training time for this case with 50% of datasets is approximately 50% shorter than using the whole datasets.

For this case, the macro-average precision, recall and F-1 score are 0.925, 0.926, and 0.925, respectively. It can be concluded that even with 50% less data size, the proposed model can still provide good damage classification.

Table 6.4 Precision, Recall and F-1 score for case 2

| Class | Precision | Recall | F-1 Score |
|---|---|---|---|
| 1 | 0.97 | 0.95 | 0.95 |
| 2 | 0.92 | 0.91 | 0.91 |
| 3 | 0.97 | 0.94 | 0.96 |
| 4 | 0.89 | 0.94 | 0.91 |
| 5 | 0.98 | 0.98 | 0.98 |
| 6 | 0.95 | 0.99 | 0.97 |
| 7 | 0.98 | 0.94 | 0.96 |
| 8 | 0.91 | 0.95 | 0.93 |
| 9 | 0.96 | 0.97 | 0.97 |
| 10 | 0.95 | 0.93 | 0.94 |
| 11 | 0.97 | 0.93 | 0.95 |
| 12 | 0.96 | 0.93 | 0.95 |
| 13 | 0.83 | 0.85 | 0.84 |
| 14 | 0.79 | 0.85 | 0.82 |
| 15 | 0.89 | 0.83 | 0.86 |
| 16 | 0.88 | 0.93 | 0.9 |

### 6.10.3 Case 3

In this case, only 50% of the acceleration response is considered rather than taking the whole length of raw acceleration measurements from the sensors. It has the advantage over the previous two cases in that it needs half the size of raw acceleration responses. The accuracy obtained is 91.77% which is not significantly less than the last two cases. However, in the previous two cases, the entire length of the time series is considered to calculate the autocorrelation. Figure 6.15 shows the confusion matrix, and Table 6.5 lists the precision, recall, and F-1 scores.

Like in the previous two cases, it is seen that some samples are not predicted correctly even when an acceleration response of shorter length is used. However, most classes predicted wrong falls in a similar type of damage. The performance is still better than

the existing study and is also very close to case 2 in this study. The macro-average precision is 0.921; recall is 0.917 and F-1 score is 0.921. All three values are above 0.9, which shows that the proposed framework can provide good damage classification when the acceleration measurement duration is 50% less than in the previous two cases.

| Actual | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | Total Samples |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **379** | 11 | 0 | 1 | 0 | 0 | 1 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 397 |
| | **2** | 4 | **365** | 1 | 16 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 388 |
| | **3** | 2 | 23 | **373** | 2 | 6 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 411 |
| | **4** | 0 | 8 | 3 | **415** | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 429 |
| | **5** | 0 | 2 | 1 | 3 | **406** | 2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 416 |
| | **6** | 0 | 0 | 0 | 0 | 5 | **373** | 6 | 1 | 1 | 2 | 7 | 0 | 0 | 0 | 0 | 0 | 395 |
| | **7** | 0 | 0 | 0 | 0 | 0 | 1 | **394** | 1 | 6 | 8 | 0 | 0 | 0 | 0 | 0 | 0 | 418 |
| | **8** | 1 | 1 | 2 | 0 | 0 | 0 | 1 | **391** | 2 | 5 | 8 | 0 | 0 | 0 | 1 | 0 | 412 |
| | **9** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | **398** | 13 | 1 | 1 | 0 | 0 | 0 | 0 | 416 |
| | **10** | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 3 | 5 | **395** | 10 | 0 | 0 | 0 | 1 | 0 | 415 |
| | **11** | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 6 | 9 | **381** | 0 | 0 | 0 | 0 | 0 | 399 |
| | **12** | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 2 | 1 | 1 | 2 | **368** | 6 | 4 | 9 | 3 | 397 |
| | **13** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | **347** | 20 | 7 | 16 | 395 |
| | **14** | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8 | 38 | **303** | 30 | 9 | 389 |
| | **15** | 0 | 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | 16 | 8 | **363** | 13 | 408 |
| | **16** | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 8 | 22 | 4 | 6 | 353 | 395 |
| | | Predicted | | | | | | | | | | | | | | | | |

Figure 6.15 Confusion Matrix for case 3

Table 6.5 Precision, Recall and F-1 score for case 3

| Class | Precision | Recall | F-1 Score |
|---|---|---|---|
| 1 | 0.98 | 0.95 | 0.97 |
| 2 | 0.88 | 0.91 | 0.94 |
| 3 | 0.98 | 0.91 | 0.94 |
| 4 | 0.94 | 0.97 | 0.96 |
| 5 | 0.97 | 0.98 | 0.97 |
| 6 | 0.99 | 0.94 | 0.97 |
| 7 | 0.97 | 0.94 | 0.96 |
| 8 | 0.93 | 0.95 | 0.94 |
| 9 | 0.96 | 0.96 | 0.96 |
| 10 | 0.92 | 0.95 | 0.93 |
| 11 | 0.81 | 0.88 | 0.84 |
| 12 | 0.89 | 0.78 | 0.83 |
| 13 | 0.87 | 0.89 | 0.88 |
| 14 | 0.89 | 0.89 | 0.89 |
| 15 | 0.87 | 0.89 | 0.88 |
| 16 | 0.89 | 0.89 | 0.89 |

The variation in the performance of the proposed framework is dependent on the data size and the autocorrelation over the length of the available time series data. Cases 1 and 2 found that the data size affects the accuracy, but the training time required is more significant with larger data size. In addition, the performance is slightly dependent on the length of the data used for calculating autocorrelation. The accuracy of Case 2 is slightly higher than that of Case 3, although the dataset size is the same. The difference is 0.26%, and the training time is the same.

## 6.11 Conclusion

This chapter presents the damage classification of a large-scale bridge 1D-CNN and time domain vibration response measured from three sensors. The study considers different types of damage scenarios under ambient excitation of the Z24 bridge. The pre-processing consists of detrending and finding the autocorrelation of the measurements. Further, it is normalised and split into shorter lengths. The performance measurement for all three cases demonstrated that the proposed framework can

accurately classify damage scenarios. There are some false positive damages observed but mostly in the damages of a similar type. Therefore, the result can still do good damage type classification.

# CHAPTER 7

## 7 Conclusion

Recent studies on applying autoencoders and residual neural networks (ResNet) using modal information have been explored for structural elemental damage identification and quantification. However, obtaining modal information from a civil engineering structure requires a lot of measurement, which is time-consuming. Further, it isn't easy to take the measurement from a large and complex infrastructure. For this reason, the use of acceleration responses measured randomly from the structure and with a small number of sensors is explored for structural elemental damage identification using a densely connected convolutional network (Dense net). Deep learning models require a considerable dataset to give good results which ultimately increases the computational cost for training the model. Using expensive graphics processing units (GPUs) can help reduce the computational cost, but it becomes costly for the model developers.

Therefore, the use of decision tree-based ensemble traditional machine learning models is explored for structural elemental damage identification using acceleration responses from a small number of sensors. An ensemble of deep learning models can be used for the same problem, but this can increase the computational cost due to the requirement of training all the deep learning models in the ensemble. The decision tree-based ensemble method, RF, is a popular machine learning model that performs well with both categorical and continuous values. Moreover, it does not require a lot of data pre-processing. It is robust to outliers and has a low risk of overfitting. The computational cost is more than a decision tree because the result is the average of all decision trees or the majority vote if the problem is a classification problem. The dataset considered in the study has continuous values. Chapters 3, 4 and 5 presented the proposed models as the multioutput regression model, i.e., it must output stiffness reduction in all the structural elements for every instance or sample.

Chapter 3 provided a detailed study on the development of RF for structural elemental damage identification and quantification using acceleration response. The numerical and experimental studies demonstrated a good prediction of stiffness reductions in the structural elements with less training time. The training time was further reduced using

PCA for dimensionality reduction. RF may require a large volume of memory for storage due to storing information of many decision trees in it. This may increase the system cost but may not be as expensive as GPUs. Further, it may not be visualised like a decision tree. It is less interpretable.

Chapter 4 presented the development and application of ERT ,which is also an ensemble machine learning model with decision trees as the base learners and is like RF. The training time required for ERT is significantly less than the RF and deep learning models due to random splitting at every node by selecting a random feature from the random subset of the features. In RF, the best split takes place at every node, i.e., it selects the best feature from a random subset of features. Moreover, in RF, it builds multiple decision trees from multiple datasets drawn with replacements from the input dataset. So, there will be a repetition of observations. This is not the case with ERT. There is no repetition of observations in the datasets used for building the decision trees. The raw acceleration responses are not used here. IRFs, one of the dynamic characteristics of a structure, is extracted from the acceleration response and processed further for use in RF and ERT. Moving averaging is performed on the IRFs to remove random variations, and then PCA is computed for dimensionality reduction. Using IRF along with moving averaging and PCA has shown better structural elemental damage identification and quantification with RF than using acceleration responses. ERTs outperform the RF with significantly less training time and more accurate damage predictions in all four scenarios considered. However, ERT also requires a large volume of memory to keep information of all the trees in it.

In all the existing structural elemental damage identification and quantification methods, both deep learning models and ensemble methods presented in Chapter 3 and Chapter 4 of this thesis are trained and tested with the specific scenario dataset and require training and testing again for other scenarios to measure the performance. Chapter 5 presented a solution to this by using LSTM-autoencoder with a noise layer which can provide good generalisation when tested using measurement datasets with noise. The computation of time of the LSTM autoencoder is reduced by computing PCA on the input observations. The same dataset in Chapter 4 is used with this method to compare the performance of RF and ERT. The data pre-processing technique is the same as in Chapter 4. The model was trained using 85% of the combined dataset of

scenario 1 and scenario 3 and tested with the remaining 15% of the combined dataset, scenario 2 and scenario 4 dataset. The performance measurement indicated that it is more robust to the noise measurement and has shown better damage prediction results than RF and ERT. The performance is further measured with fewer data, assuming the available number of sensors is three instead of five. There is no significant difference in performance using data from five and three sensors. The proposed model has the advantage of using fewer sensors and being robust to noise measurements. It does not need to be trained and tested again for all four scenarios considered.

In all the methods presented in Chapters 3, 4, and 5 and the existing deep learning models for damage identification and quantification of civil engineering infrastructure, the damage is usually defined in terms of reduction in structural element stiffness. However, a civil engineering structure may experience different damage types in real-world settings. It, therefore, can also be defined according to the damage types; each type can have several levels of damage. Z24 bridge dataset has considered different damage scenarios during the PDT, which took over a month. The measurement was done for both forced excitation and ambient condition. The proposed study used measurements under ambient conditions. The methods explained earlier may not be suitable when the bridge experiences combined damages of different types and levels.

Moreover, the measurement taken from the Z24 bridge is undoubtedly exposed to external conditions such as the environment and vehicle loading. This can make it difficult for the proposed model to map the input pattern to the labelled output. Therefore, the proposed 1D-CNN is developed ,taking the problem as a classification problem. The presented data processing methods and 1D-CNN model demonstrated good damage classification. The classification accuracy is more than the existing hybrid model. The proposed model also used fewer sensors than the current methods and did not use any data augmentation method. Autocorrelation is computed on the acceleration responses before splitting them into smaller lengths.

**Future work**

The model for structural elemental damage identification and quantification is developed from the training and testing dataset generated from the updated FEM model and is further validated with the structure model fabricated in the laboratory. The studies considered only the measurement noise and modelling error associated

with the uncertainty, i.e., material properties and their influences on the stiffness parameter. The experimental results have demonstrated suitable damage identification. However, In the real-world setting, there can be other uncertainties arising from environment loading such as wind and vehicles on the bridges, construction flaws, boundary conditions etc. These sources of errors can be sensitive to the acceleration response and IRFs extracted from it and may affect the performance of the models proposed. There is a need to consider more uncertainties in the FEM which can help reduce errors in the training data. This can help get more accurate damage identification and quantification in the test dataset obtained from the fabricated structure in the laboratory.

Further, the acceleration responses are measured using forced excitation. The acceleration responses and IRF extracted from it under ambient conditions are not considered. It would be an exciting topic in future to consider exploring this for the proposed methods for damage identification and quantification of structural elements.

Chapter 6 presented a supervised damage classification method using the dataset measured from a large-scale bridge, Z24. The performance measured has shown better accuracy than the current study on the Z24 bridge. However, there are still spaces to improve to achieve more accurate damage classification. Not getting optimal classification accuracy could be due to uncertainties from non-structure components and environmental conditions, which makes it hard for the model to map the internal pattern of input variables with the labelled output. Autocorrelation is computed from the acceleration response. It would be interesting to obtain IRFs from the acceleration response under ambient conditions and measure the performance in the future.

The methods proposed are all supervised. It may not be possible to have measurements for all the damage scenarios considered in the Z24 bridge, e.g., data from the entire structure, which could only be simulated from a numerical model. It is also challenging to include all the uncertainties relating to external conditions in the real-world settings in the simulation. Therefore, unsupervised methods can be explored for damage detection using the undamaged dataset to detect damage in the structure. Moreover, finding ways to include uncertainties in the simulation model as much as possible could be another study to explore in the future.

## References


Abdeljaber, O., & Avci, O. (2016). Nonparametric structural damage detection algorithm for ambient vibration response: Utilizing artificial neural networks and self-organizing maps. *Journal of Archetectural Engineering 22*(2). doi:https://doi.org/10.1061/(ASCE)AE.1943-5568.0000205

Abdeljaber, O., Avci, O., Kiranyaz, M. S., Boashash, B., Sodano, H., & Inman, D. J. (2018). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing, 275*, 1308-1317. doi:https://doi.org/10.1016/j.neucom.2017.09.069

Abdeljaber, O., Avci, O., Kiranyaz, S., Gabbouj, M., & Inman, D. J. (2017). Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of sound and vibration, 388*, 154-170. doi:https://doi.org/10.1016/j.jsv.2016.10.043

Aldrich, C., & Auret, L. (2010). Fault detection and diagnosis with random forest feature extraction and variable importance methods. *IFAC Proceedings Volumes, 43*(9), 79-86.

Alvandi, A., & Cremona, C. (2006). Assessment of vibration-based damage identification techniques. *Journal of sound and vibration, 292*(1-2), 179-2002. doi:https://doi.org/10.1016/j.jsv.2005.07.036

Amiri, G. G., Jalalinia, M., Hosseinzadeh, A. Z., Nasrollahi, A., & 503–515. (2015). Multiple Crack Identification in Euler Beams by Means of B-Spline Wavelet. *Archive of Applied Mechanics 85*, 503-515. doi:https://doi.org/10.1007/s00419-014-0925-z

Avci, O., Abdeljaber, O., Kiranyaz, S., Hussein, M., Gabbouj, M., & Inman, D. J. (2021a). 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*. doi:https://doi.org/10.1016/j.ymssp.2020.107077

Avci, O., Abdeljaber, O., Kiranyaz, S., Hussien, M., Gabbouj, M., & Inman, D. J. (2021b). 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*. doi:https://doi.org/10.1016/j.ymssp.2020.107077

Azimi, M., & Pekcan, G. (2020). Structural health monitoring using extremely compressed data through deep learning. *Computer - Aided Civil and Infrastructure Engineering, 35*(6), 597-614. doi:https://doi.org/10.1111/mice.12517

Betti, M., Facchini, L., & Biagini, P. (2015). Damage detection on a three-story steel frame using artificial neural networks and genetic algorithms. *Meccanica, 50*, 875-886. doi:https://doi.org/10.1007/s11012-014-0085-9

Bishop, C. (1995). *Neural Networks for Pattern Recognition*: Oxford University Press.

Breiman, L. (1996). Bagging predictors. *Mach Learn, 24*, 123-140. doi:https://doi.org/10.1023/A:1018054314350

Brownlee, J. (2018). *Long short-term memory network with python: Develop sequence prediction models with deep learning* (v1.5 Ed.): Jason Brownlee.

Brownlee, J. (2019a). *Basics of Linear Algebra for Machine Learning* (v1.7 ed.): Jason, Brownlee.

Brownlee, J. (2019b). *Better deep learning* (v1.3 ed.): Jason, Brownlee.

Brownlee, J. (2019c). *Machine Learning Mastery with Python* (v1.7 ed.): Jason, Brownlee.

Cao, J., & Liu, X. (2016). *Wireless Sensor Networks for Structural Health Monitoring*: Cham : Springer International Publishing : Imprint: Springer.

Chae, M. J., Yoo, H., Kim, J. C., & Cho, M. J. (2012). Development of a Wireless Sensor Network System for Suspension bridge health monitoring. *Automation in Construction, 21*, 237-252. doi:https://doi.org/10.1016/j.autcon.2011.06.008

Chencho, Li, J., Hao, H., Wang, R., & Li, L. (2020). Development and application of random forest technique for elemental level structural damage quantification. *Structural Control Health Monitoring, 28*(3). doi:https://doi.org/10.1002/stc.2678

Cury, A., & Cremona, C. (2012). Pattern recognition of structural behaviors based on learning algorithms and symbolic data concepts. *Structural Control and Health Monitoring, 19*, 161-186. doi:https://doi.org/10.1002/stc.412

Dackermann, U., Li, J., & Samali, B. (2016). Dynamic-based damage identification using neural network ensembles and damage index method. *Advances in Structural Engineering, 13*(6), 1001-1016.

doi:https://doi.org/10.1260%2F1369-4332.13.6.1001

Doebling, S. W., Farrar, C. R., & Prime, M. B. (1998). A summary review of vibration based damge identification methods. *Shock and vibration digest, 30*(2), 91-105. doi:http://dx.doi.org/10.1177/058310249803000201

Drakos, G. (2019). Random forest regression model explained in depth. In: mimeo.

Esfandiari, A., Nabiyan, M. S., & Rofooei, F. R. (2020). Structural damage detection using principal component analysis of frequency response function data. *Structural control and health monitoring, 27*(7). doi:https://dx.doi.org/10.1002/stc.2550

Fallahian, M., Khoshnoudian, F., & Meruane, V. (2017). Ensemble classification method for structural damage assessment under varying temperature. *Structural Health Monitoring, 17*(4), 747-762. doi:https://doi.org/10.1177%2F1475921717717311

Fallahian, M., Khoshnoudian, F., Talaei, S., Meruane, V., & Shadan, F. (2018). Experimental validation of a deep neural network—sparse representation classification ensemble method. *The Structural Design of Tall and Special Building, 27*(15). doi:https://doi.org/10.1002/tal.1504

Fan, W., & Qiao, P. (2011). Vibration-based damage identification method: a review and comparative study. *Structural Health Monitoring, 10*(1), 83-111. doi:https://doi.org/10.1177%2F1475921710365419

Farrar, C. R., & Worden, K. (2007). An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 365*(1851), 303-315.

Fchollet. (2019, 12/04/2020). Complete guide to the functional API. Retrieved from https://keras.io/guides/functional_api/

Figueiredo, E., Moldovan, I., Santos, A., Campos, P., & Costa, J. (2019). Finite element-based machine-learning approach to detect damage in bridge under operational and environment variations. *Journal of Bridge Engineering, 24*(7). doi:https://dx.doi.org/10.1061/(ASCE)BE.1943-5592.0001432

Francesco, C., Sbarufatti, C., Matteo, C., & Marco, G. (2017). A particle filter-based model selection algorithm for fatigue damage identification on aeronautical structure. *Structural Control and Health Monitoring, 24*(11). doi:https://doi.org/10.1002/stc.2002

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning, 63*(1), 3-42. doi:https://dx.doi.org/10.1007/s10994-006-6226-1

Giagopoulos, D., Arailopoulos, A., Dertimanis, V., Papadimitrious, C., Chatzi, E., & Grompanopoulos, K. (2018). Structural health monitoring and fatigue damage estimation using vibration measurements and finite element model updating. *Structural Health Monitoring, 18*(4), 1189-1206. doi:https://doi.org/10.1177%2F1475921718790188

Goh, L. D., Bakhary, N., Rahman, A. A., & Ahmad, B. H. (2013). Prediction of unmeasured mode shape using neural network for damage detection. *Journal Teknol, 61*, 57-66. doi:https://doi.org/10.11113/jt.v61.1624

Goodfellow, I. J., Bengio, Y., & Courville, A. (2016). *Deep learning*.

Gopalakrishnan, S., Ruzzene, M., & Hanagud, S. (2011). *Computational Techniques for Structural Health Monitoring*: Springer Science & Business Media.

Haidarpour, A., & Tee, K. F. (2020). Finite element updating for structural health monitoring. *Structural Durability and Health Monitoring, 14*(1). doi:https://www.techscience.com/sdhm/v14n1/38469

Hartmann, D., Smarsly, K., & Law, K. H. (2011). *Coupling sensor-based structural health monitoring with finite element model updating for probabilistic lifetime estimation of wind energy converter structures.* Paper presented at the Proceedings of the 8th international workshop on structural health monitoring.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*: Springer Science & Business Media.

Heng, A., Zhang, S., Tan, A. C. C., & Mathew, J. (2009). Rotating machinery prognostics: state of the art, challenges and opportunities. *Mechanical Systems and Signal Processing, 23*(3), 724-739. doi:https://doi.org/10.1016/j.ymssp.2008.06.009

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735-1780. doi:https://doi.org/10.1162/neco.1997.9.8.1735

Hu, X. W. (2013). A wireless Sensor Network -Based Structural Health Monitoring System for Highway Bridges. *Computer Aided Civil and Infrastructure Engineering, 28*(3). doi:https://doi.org/10.1111/j.1467-8667.2012.00781.x

Hung, V. D., Hoa, T. N., Tung, V. N., T, B.-T., Guido, D. R., & Nguyen, H. X. (2021). Data-Driven Structural Health Monitoring Using Feature Fusion and Hybrid Deep Learning. *IEEE transactions on automation science and engineering, 18*(4), 2087-2013. doi:https://doi.org/10.1109/TASE.2020.3034401

Hung, V. D., Hung, H. M., Anh, P. H., & Thang, N. T. (2020). Structural damage detection using hybrid deep learning algorithm. *Journal of Science and Technology in Civil Engineering(STCE)-NUCE, 14*(2). doi:http://dx.doi.org/10.31814/stce.nuce2020-14(2)-05

Jardine, A. K. S., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementation condition-based maintenance. *Mechanical Systems and Signal Processing, 20*(7), 1483-1510. doi:https://doi.org/10.1016/j.ymssp.2005.09.012

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. doi:https://doi.org/10.1098/rsta.2015.0202

Juang, J.-N. (1994). *Applied System Identification*. Englewood Cliffs: Prentice Hall.

Kan, M. S., Tan, A. C. C., & Mathew, J. (2015). A review on prognostic techniques for non-stationary and non-linear rotating systems. *Mechanical Systems and Signal Processing, 62*, 1-20. doi:https://doi.org/10.1016/j.ymssp.2015.02.016

Karbhari, V. M., & Ansari, F. (2009). *Structural Health Monitoring of Civil Infrastructure Systems*: Cambridge: Elsevier Science and Technology.

Kim, C. W., & Kawatani, M. (2998). Pseudo-static approach for damage identification of bridges based on coupling vibration with movign vehicle. *Structure and Infrastructure Engineering, 4*(5), 371-379. doi:https://doi.org/10.1080/15732470701270082

Koehrsen, W. (2018). Hyperparameter tuning the random forest in python. *Towards Data Science*.

Kramar, C. D. S., de Smet, C. A. M., & Guido, D. R. (1999). *Z24 bridge damage detection tests*. Paper presented at the IMAC 17, the International Modal Analysis Conference, Kissimmee, FI, USA.

Law, S. S., & Li, X. Y. (2007). Wavelet- based sensitivity analaysis of the impulse response function for damage detection. *Journal of applied mechanics, 74*(2), 375-377. doi:https://dx.doi.org.10.1115/1.2189875

Lawson, E., Smith, D., Sofge, D., Elmore, P., & Petry, F. (2017). Decision forests for machine learning classification of large, noisy seafloor feature sets. *Computers & Geosciences, 99*, 116-124. doi:https://dx.doi.org/10.1016/j.cageo.2016.10.013

Lecun, Y., Bengio, Y., & Hington, G. (2015). Deep learning. *Nature 521*, 436-444. doi:https://doi.org/10.1038/nature14539

Lee, E. W. M., & Lam, H. F. (2011). Intelligent-based structural damage detection model. *Mechanics of advanced materials and structures, 18*(8), 590-596. doi:https://doi.org/10.1080/15376494.2011.621838

Li, J., Hao, H., & Xingyu, F. (2015). Structural damage identification with extracted impulse response functions and optimal sensor locations. *Electronic Journal of Structural Engineering, 14*(1), 123-132.

Li, J., Law, S. S., & Ding, Y. (2012). Substructure damage identification based on response reconstruction in frequency domain and model updating. *Engineering Structure, 41*, 270-284. doi:https://dx.doi.org/10.1016/j.engstruct.2012.03.035

Li, X. Y., & Law, S. S. (2008). Structural damage detection with statistical analysis from support excitation. *Mechanical Systems and Signal Processing, 22*, 1793-1808. doi:https://doi.org/10.1016/j.ymssp.2008.03.003

Li, Y., Zhang, M., & Yang, W. (2018). Numerical and experimental investigation of modal-energy-based damage localization for offshore wind turbine structures. *Advances in Structural Engineering, 21*(10), 1510–1525. doi:https://doi.org/10.1177%2F1369433217750725

Lin, E., Mukherjee, S., & Kannan, S. (2021). A deep adversarial variation autoencoder model for dimensionality reduction in single-cell RNA sequencing analysis. *BMC Bioinformatics, 21*(1). doi:DOI:10.1186/s12859-020-3401-5

Lin, J., Wang, J., Wang, L., & Law, S. (2019). Structural damage diagnosis-oriented impulse response function estimation under seismic excitation. *Sensors, 19*(24). doi:https://dx.doi.org/10.3390%2Fs19245413

Magalhaes, F., Cunha, A., & Caetano, E. (2012). Vibration based structural health monitoring of an arch bridge: From automated OMA to damage detection. *Mech. Syst. Signal Process, 28*, 212-228. doi:https://doi.org/10.1016/j.ymssp.2011.06.011

Malekzadeh, M., & Catbas, F. N. (2016). A machine learning framework for automated functionality monitoring of movable bridges. In *Dynamics of Civil Structures* (Vol. 2, pp. 57-63).

Mansouri, M., Avci, O., Nounou, H., & Nounou, M. (2015). *A comparative assessment of nonlinear state estimation methods for structural health monitoring.* Paper presented at the 33rd IMAC, A conference and exposition of structural dynamics, Orlando.

Mariniello, G., Pastore, T., Menna, C., Festa, P., & Asprone, D. (2020). Structural damage detection and localization using decision tree sensemble and vibration data. *Computer Aided Civil Infrastructure Engineering, 36*(9), 1129-1149. doi:https://doi.org/10.1111/mice.12633

Mitiche, I., Nesbitt, A., Conner, S., Boreham, P., & Morison, G. (2020). 1D-CNN based real-time fault detection system for power asset diagnostics. *IET generation, transmission & distribution, 14*(24), 5766-5773. doi:https://doi.org/10.1049/iet-gtd.2020.0773

Moore, M., Phares, B. M., Graybeal, B., Rolander, D., Washer, G., & Wiss, J. (2001). *Reliability of visual inspection for highway bridges, volume I*. Retrieved from

Ng, C. T. (2014). Application of Bayesian-designed artificial neural networks in Phase II structural health monitoring benchmark studies. *Australian Journal of structural engineering, 15*(1), 27-36. doi:https://doi.org/10.7158/13287982.2014.11465144

Nicknam, A., Hosseini, M. H., & Bagheri, A. (2011). Damage Detection and Denoising in Two-Dimensional Structures using Curvelet Transform by Wrapping Method. *Archive of Applied Mechanics, 81*(12), 1915–1924. doi:https://dx.doi.org/10.1007/s00419-011-0527-y

Nie, Z., Guo, E., Li, J., Hao, H., Ma, H., & Jian, H. (2020). Bridge condition monitoring using fixed moving principal component analysis. *Structural Control and Health Monitoring, 27*(6).

Nowlan, S. J., & Hington, G. E. (1992). Simplifying neural networks by soft weight sharing. *Neural Computation, 4*(4). doi:https://doi.org/10.1162/neco.1992.4.4.473

Pal, A. K. (2017, 11 Dec 2017). Understading Dimension Reduction with Principle Component Analysis. Retrieved from https://blog.paperspace.com/dimension-reduction-with-principal-component-analysis/

Pathirage, C. S. N., Li, J., Hao, H., Li, L., Liu, W. Q., & Ni, P. (2018). Structural damage identification based on autoencoder neural networks and deep learning. *Engineering Structures, 172*, 13-28.

doi:https://dx.doi.org/10.1016/j.engstruct.2018.05.109

Pathirage, C. S. N., Li, J., Li, L., Hao, H., Liu, W. Q., & Wang, R. (2019). Development and application of a deep learning-based sparse autoencoder framework for structural damage identification. *Structureal Health Monitoring, 18*(1), 103-122. doi:https://doi.org/10.1177%2F1475921718800363

Rafiei, M. H., & Adeli, H. (2019). A novel unsupervised deep learning model for global and local health condition assessment of Structure. *Engineering Structures, 156*, 598-607. doi:https://dx.doi.org/10.1016/j.engstruct.2017.10.070

Reddy, G., Reddy, M., Lakshmanna, K., Kaluri, R., Rajput, D., Srivastava, G., & Baker, T. (2020). Analysis of Dimensionality Reduction Techniques on Big Data. *IEEE Access, 8*, 54776-54788.

doi: 10.1109/ACCESS.2020.2980942

Robertson, A., Park, K., & Alvin, K. (1998). Extraction of impulse response data via wavelet transfrom for structural system identification. *Journal of vibration and acoustics, 120*(1), 252-260. doi:https://dx.doi.org/10.1115/1.2893813

Roeck, G. D. (2003). The state-of-art of damage detection by vibration monitoring: the SIMCES experience. *Journal of Structural Control 10*, 127-134. doi:https://doi.org/10.1002/stc.20

Roy, K. (2017). Structural Damage Identification using Mode Shape Slope and Curvature. *Journal of Engineering Mechanics 143*. doi:https://doi.org/10.1061/(ASCE)EM.1943-7889.0001305

Rucevskis, S., Janeliukstic, R., Akishin, P., & Chate, A. (2016). Mode shape-based damage detection in plate structure without baseline data. *Structural control and health monitoring, 23*(9), 1180-1193. doi:https://doi.org/10.1002/stc.1838

Sarbu, C., & Pop, H. (2005). Principal component analysis versus fuzzy principal component analysis: A case study: The quality of danube water (1985-1996). *Talanta, 65*(5), 1215-1220. doi:https://doi.org/10.1016/j.talanta.2004.08.047

Sbarufatti, C. (2017). Optimization of an artificial nueral network for fatigue damage identification using analysis of variance *Structural Control and Health Monitoring, 29*(9). doi:https://doi.org/10.1002/stc.1964

Sbarufatti, C., Manes, A., & Giglio, M. (2013). Performance optimization of a diagnostic system based upon a simulated strain field for fatigue damage characterization. *Mechanical Systems and Signal Processing, 40*(2), 667-690. doi:https://dx.doi.org/10.1016/j.ymssp.2013.06.003

Seventekidis, P., Giagopoulos, D., Arailopoulos, A., & Markogiannaki, O. (2020). Structural health monitoring using deep learning with optimal finite element model generated data. *Machanical Systems and Signal Processing 145*. doi:https://doi.org/10.1016/j.ymssp.2020.106972

Shadan, F., Khoshnoudian, F., & Estandiari, A. (2016). A frequency response-based stuctural damage identification using model updating method. *struct. Control Health Monit, 23*, 286-302. doi:https://doi.org/10.1002/stc.1768

Sipple, J. D., & Sanayei, M. (2013). Finite element model updating using frequency response functions and numerical sensitivities. *Structural Control Health Monitoring, 21*(5), 784-802. doi:https://doi.org/10.1002/stc.1601

Smarsly, K., Dragos, K., & Wiggenbrock, J. (2016). *Machine learning techniques for structural health monitoring.* Paper presented at the 8th EWSHM, Bilbao, Spain.

Smarsly, K., Hartmann, D., & Law, K. H. (2013). A computational framework for life-cycle management of wind turbines incorporating structural health monitoring. *Structural Health Monitoring, 12*(4), 359-376. doi:http://dx.doi.org/10.12989/sss.2019.24.5.567

Srivastava, N., Hington, G., A, K., Sutsekever, I., & Salakhudinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*, 1929-1958.

Srivastava, N., Mansimov, R., & Salakhudinov, R. (2015). *Unsupervised Learning of Video representations using LSTMs.* Paper presented at the ICML

Sun, T. Z., Zhi-Hua. (2018). Structural diversity for decision tree ensemble learning. *Frontiers of Computer Science, 12*(3), 560-570.

Todorovska, M. I., & Trifunac, M. D. (2008). Earth quake damage detection in the Imperial County Service Building III: Analysis of wave travel times via impulse response function. *Soil Dynamics and Earthquake Engineering, 28*(5), 387-404. doi:https://doi.org/10.1016/j.soildyn.2007.07.001

Wan, V., Yannis, A., Hanna, S., & Jakub, V. (2017). Google's next-generation real-time unit-selection synthesizer using sequence-to-sequence LSTM-based autoencoder. *IEEE INTERSPEECH* 1143-1147.

Wang, R., Chencho, An, S., Li, J., Li, L., Hao, H., & Liu, W. Q. (2020). Deep residual network framework for structural health monitoring. *Structural Health Monitoring, 20*(4), 1443-1461. doi:https://doi.org/10.1177%2F1475921720918378

Wang, R., Li, J., Chencho, An, S., Hao, H., Liu, W.-Q., & Li, L. (2021). Densely connected convolutional networks for vibration based structural damage identification. *Engineering Structures, 245*. doi:https://doi.org/10.1177%2F1475921720918378

Wang, R., Li, L., & Li, J. (2018). A novel parallel auto-encoder framework for multi-scale data in civil structural health monitoring. *Algorithms, 11*(8). doi:https://doi.org/10.3390/a11080112

Weber, B., & Paultre, P. (2010). Damage Identification in a truss tower by regularized model updating. *Journal of Structural Engineering, 136*(3), 307-316. doi:https://dx.doi.org/10.1061/(ASCE)ST.1943-541X.0000105

Wu, R.-T., & Jahanshahi, M. R. (2018). Data fusion approaches for sturctural health monitoring and system identification: Past, present, and future. *Structural Health Monitoring, 19*(2). doi:https://doi.org/10.1177%2F1475921718798769

Wu, X. G., J; Garrett Jr, J.H. (1992). Use of neural networks in detection of structural damage. *Computers and structures, 42*(4), 649-659. doi:https://doi.org/10.1016/0045-7949(92)90132-J

Xhou, X. T., Ni, Y. Q., & Zhang, F. L. (2014). Damage localization of cable-supported bridges using modal frequency data and probabilistic neural network. *Mathematical Problems in Engineering, 2014*. doi:https://doi.org/10.1155/2014/837963

Xu, Q. (2021). Damage Index Analysis of Retaining Wall Structures Based on the Impulse Response Function and Virtual Impulse Response Function. *Shock and Vibration Digest*. doi:https://dx.doi.org/10.1155/2021/9741732

Xu, X., & Yonedal, M. (2021). *Multitask Air-Quality Prediction Based on LSTM-Autoencoder Mode*. Paper presented at the IEEE Transactions on Cybernetics.

Xu, Y., Bao, Y., Chen, J., Zuo, W., & Li, H. (2018). Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images. *Structural Health Monitoring, 18*(3), 653-674. doi:https://doi.org/10.1177%2F1475921718764873

Ye, X. W., Jin, T., & Yun, C. B. (2019). A review on deep learning-based structural health monitoring of civil infrastructures. *Smart Structures and Systems, 24*(5), 1738-1584. doi:http://dx.doi.org/10.12989/sss.2019.24.5.567

Yong, X., & Hao, H. (2003). Statistical Damage Identification of Structures with Frequency Changes. *Journal of Sound and Vibration, 263*(4), 853–870. doi:https://dx.doi.org//10.1016/S0022-460X(02)01077-5

Yuan, F. G., Zargar, S. A., Chen, Q., & Wang, S. (2020). *Machine learning for structural health monitoring: challenges and opportunities*. Paper presented at the SPIE SMART STRUCTURES + NONDESTRUCTIVE EVALUATION, California.

Zabalza, J., Jinchang, R., Zheng, J., Zhao, H., Yang, Z., Du, P., & Marshall, S. (2016). Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. *Neurocomputing, 185*, 1-10. doi:https://doi.org/10.1016/j.neucom.2015.11.044

Zhang, Y., Miyamori, Y., Mikami, S., & Saito, T. (2019). Vibration-based structural state identification by a 1-dimensional convolution nueral network. *Computer Aided Civil Infrasture Engineering, 34*(9), 822-839. doi:https://doi.org/10.1111/mice.12447

Zhang, Y., Qin, C., Srivastava, A. K., Jin, C., & Sharma, R. K. (2020). Data-Driven Day-Ahead PV Estimation Using Autoencoder-LSTM and Persistence Model. *IEEE Transactions on Industry Applications, 56*(6), 7185-7192. doi:https://doi.org/10.1109/TIA.2020.3025742

Zhou, Q., Ning, Y., Zhou, Q., Luo, L., & Lei, L. (2014). Structure damage detection method based on random forest recursive forests and data fusion. *Mechanical systems and signal processing, 46*(1), 82-90. doi:https://doi.org/10.1016/j.ymssp.2013.12.013

Zhou, Q., Zhou, H., Zhou, Q., Yang, F., & Luo, L. (2013). Structural damage detection method based on random forests and data fusion. *Structural Health Monitoring, 12*(1), 48-58. doi:https://doi.org/10.1177%2F1475921712464572

Zhu, S., Levinson, D., Liu, H. X., & Harder, K. (2010). The traffic and behavioral effects of the I-35W Mississippi River bridge collapse. *Transportation research part A: policy and practice, 44*(10), 771-784.

**APPENDIX I: Attribution of authorship**

To whom it may concern,

I, Chencho, conducted numerical, experimental investigations, data processing, analysis and wrote manuscripts of the papers titled as follows, which were revised and edited by the second and third co-authors. They also provided insights on experimental preparation, data processing and data analysis. The fourth and fifth authors helped with the ideas of data processing and analysis.

**Development and application of random forest technique for element level structural damage quantification**

(…….)

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

(Associate Prof. Jun Li) (……...…...)

(Prof. Hong Hao) (……...……...)

(Dr. Ruhua Wang) (……...)

(Prof. Ling Li) (……...……...)

To whom it may concern,

I, Chencho, conducted numerical investigations, data processing, analysis and wrote manuscripts of the papers titled as follows, which were revised and edited by the second and third co-authors. They also provided insights on data processing and data analysis.

**Long short-term memory auto-encoder based damage quantification using impulse response functions**

(.....)

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

(Associate Prof. Jun Li) (……...…...)

(Prof. Hong Hao) (……...……...)

## APPENDIX II: Permission to use copyright material



JOHN WILEY AND SONS LICENSE
TERMS AND CONDITIONS

Apr 17, 2022

This Agreement between Mr. Chencho Chencho ("You") and John Wiley and Sons ("John Wiley and Sons") consists of your license details and the terms and conditions provided by John Wiley and Sons and Copyright Clearance Center.

| | |
|---|---|
| License Number | 5291140890862 |
| License date | Apr 17, 2022 |
| Licensed Content Publisher | John Wiley and Sons |
| Licensed Content Publication | Structural Control and Health Monitoring |
| Licensed Content Title | Development and application of random forest technique for element level structural damage quantification |
| Licensed Content Author | Ling Li, Ruhua Wang, Hong Hao, et al |
| Licensed Content Date | Dec 6, 2020 |
| Licensed Content Volume | 28 |
| Licensed Content Issue | 3 |
| Licensed Content | 19 |

| | |
|---|---|
| Pages | |
| Type of use | Dissertation/Thesis |
| Requestor type | Author of this Wiley article |
| Format | Print and electronic |
| Portion | Full article |
| Will you be translating? | No |
| Title | Efficient Supervised Machine Learning Techniques for Structural Health Monitoring |
| Institution name | Curtin University |
| Expected presentation date | Jun 2022 |
| Order reference number | DIS2022 |
| Requestor Location | Mr. Chencho Chencho<br>23A Stonehouse Crescent<br><br>Bentley, WA 6102<br>Australia<br>Attn: Mr. Chencho Chencho |
| Publisher Tax ID | EU826007151 |
| Total | 0.00 AUD |

Terms and Conditions

ELSEVIER LICENSE
TERMS AND CONDITIONS

May 23, 2022

---

This Agreement between Mr. Chencho Chencho ("You") and Elsevier ("Elsevier") consists of your license details and the terms and conditions provided by Elsevier and Copyright Clearance Center.

| | |
|---|---|
| License Number | 5314650651810 |
| License date | May 23, 2022 |
| Licensed Content Publisher | Elsevier |
| Licensed Content Publication | Mechanical Systems and Signal Processing |
| Licensed Content Title | DESCRIPTION OF Z24 BENCHMARK |
| Licensed Content Author | J. MAECK,G. DE ROECK |
| Licensed Content Date | Jan 1, 2003 |
| Licensed Content Volume | 17 |
| Licensed Content Issue | 1 |
| Licensed Content Pages | 5 |
| Start Page | 127 |
| End Page | 131 |

| | |
|---|---|
| Type of Use | reuse in a thesis/dissertation |
| Portion | figures/tables/illustrations |
| Number of figures/tables/illustrations | 1 |
| Format | electronic |
| Are you the author of this Elsevier article? | No |
| Will you be translating? | No |
| Title | Efficient Supervised Machine Learning Techniques for Structural Health Monitoring |
| Institution name | Curtin University |
| Expected presentation date | Jun 2022 |
| Portions | Figure 2 (Top view and Cross Section) |
| Requestor Location | Mr. Chencho Chencho<br>23A Stonehouse Crescent<br><br>Bentley, WA 6102<br>Australia<br>Attn: Mr. Chencho Chencho |
| Publisher Tax ID | GB 494 6272 12 |
| Total | 0.00 AUD |

CCC RightsLink®

Home | Help | Email Support | Chencho Chencho

IEEE Requesting permission to reuse content from an IEEE publication

## Data-Driven Structural Health Monitoring Using Feature Fusion and Hybrid Deep Learning

**Author:** Hung V. Dang
**Publication:** IEEE Transactions on Automation Science and Engineering
**Publisher:** IEEE
**Date:** Oct. 2021



### Thesis / Dissertation Reuse

**The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:**

*Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:*

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

*Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:*

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK | CLOSE WINDOW



### Papers now Available: AIAC19 & ACAM10

EV Elena Vvedenskaia <EVvedenskaia@engineersaustralia.org.au>
To: Chencho
Wed 5/25/2022 12:40 PM

Dear Chencho,

Thank you for your enquiry.
You are welcome to use your conference paper in your PhD Thesis, you don't need our permission if you are using it for this purpose.

Regards,
**Elena Vvedenskaia, FIEAust | Library Manager**
Learned Society Business Unit
Professional Standards & Practice Group
Engineers Australia | 11 National Circuit BARTON ACT 2600 Australia
t:(61) 26270 6535 | Ext: 2535 | f: (61) 26273 1958
w: www.engineersaustralia.org.au

Reply | Forward

**From:** Chencho <chencho@postgrad.curtin.edu.au>
**Sent:** Saturday, 21 May 2022 11:28 AM
**To:** National Events <NationalEvents@engineersaustralia.org.au>
**Subject:** Re: Papers now Available: AIAC19 & ACAM10

Hi,

Greetings again.

I am Chencho and attended ACAM10 and presented a paper. A full paper was submitted and has been published with RMIT. The paper was a part of my PhD study. I am currently writing a thesis and I would to know if I need to get copyright approval to use the paper as a part of my Thesis chapter. The paper is tittle as

**"Long short-term memory auto-encoder based damage quantification using impulse response functions"**

Best regards

Chencho