ABSTRACT

Title of Dissertation:	SMART STRUCTURAL CONDITION ASSESSMENT METHODS FOR CIVIL INFRASTRUCTURES USING DEEP LEARNING ALGORITHM
	Heng Liu, Doctor of Philosophy, 2018
Dissertation directed by:	Professor Yunfeng Zhang, Department of Civil and Environmental Engineering

Smart Structural Health Monitoring (SHM) technique capable of automated and accurate structural health condition assessment is appealing since civil infrastructural resilience can be enhanced by reducing the uncertainty involved in the process of assessing the condition state of civil infrastructures and carrying out subsequent retrofit work. Over the last decade, deep learning has seen impressive success in traditional pattern recognition applications historically faced with long-time challenges, which motivates the current research in integrating the advancement of deep learning into

SHM applications. This dissertation research aims to accomplish the overall goal of establishing a smart SHM technique based on deep learning algorithm, which will achieve automated structural health condition assessment and condition rating prediction for civil infrastructures.

A literate review on structural health condition assessment technologies commonly used for civil infrastructures was first conducted to identify the special need of the proposed method. Deep learning algorithms were also reviewed, with a focus on pattern recognition application, especially in the computer vision field in which deep learning algorithms have reported great success in traditionally challenging tasks. Subsequently, a technical procedure is established to incorporate a particular type of deep learning algorithm, termed Convolutional Neural Network which was found behind the many success seen in computer vision applications, into smart SHM technologies. The proposed method was first demonstrated and validated on an SHM application problem that uses image data for structural steel condition assessment. Further study was performed on time series data including vibration data and guided Lamb wave signals for two types of SHM applications - brace damage detection in concentrically braced frame structures or nondestructive evaluation (NDE) of thin plate structures. Additionally, discrete data (neither images nor time series data), such as the bridge condition rating data from National Bridge Inventory (NBI) data repository, was also investigated for the application of bridge condition forecasting. The study results indicated that the proposed method is very promising as a data-driven structural health condition assessment technique for civil infrastructures, based on research findings in the four distinct SHM case studies in this dissertation.

SMART STRUCTURAL CONDITION ASSESSMENT METHODS FOR CIVIL INFRASTRUCTURES USING DEEP LEARNING ALGORITHM

by

Heng Liu

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2018

Advisory Committee:
Professor Yunfeng Zhang, Chair
Professor Amde M. Amde
Professor Bilal M. Ayyub
Research Professor Chung C. Fu
Professor Sung Lee

Dedication

To my loving, caring wife and daughter.

To my beloved parents, on both sides.

To my precious memory at University of Maryland, College Park.

Acknowledgements

I would like to place on record my sincere thanks to Professor Yunfeng Zhang, for his countless advice, encouragement, and support throughout my graduate studies at the University of Maryland, College Park. It is my greatest honor to pursue my Ph.D. study under his supervision. His academic merit and passion have greatly inspired me in many ways.

I would also like to thank the efforts of my committee members, Dr. Amde, Dr. Ayyub, Dr. Fu, and Dr. Lee. Their valuable time and precious comments are greatly appreciated.

I gratefully acknowledge the fellowship supports from the Stanley R. Zupnik Foundation, the Bechtel Fellowship Fund, and the Guilian Foundation during my graduate study at the University of Maryland, College Park. Their encouragements on my research work are greatly appreciated. Financial support from the Department of Civil and Environmental Engineering are also appreciated.

I cherish my days at the University of Maryland, College Park. Special thanks to everyone who helped me during this time.

Table of Contents

Dedication ii
Acknowledgements iii
Table of Contents iv
List of Tables viii
List of Figures ix
Chapter 1: Introduction 1
1.1 Research Background and Motivation 1
1.2 Research Objectives 3
1.3 Dissertation Organization 5
Chapter 2: Literature Review 9
2.1 Common SHM Techniques for Civil Infrastructures 9
2.1.1 Structural Steel Fuse Condition Assessment in EBF 9
2.1.2 Brace Damage Condition Assessment in CBF 11
2.1.3 NDE in Thin Plate Structures 13
2.1.4 Bridge Condition Data Modeling 15
2.2 Deep Learning Algorithms for Pattern Recognition 17
2.2.1 A little History about Deep Learning 18
2.2.2 Deep Learning based Pattern Recognition 19
2.3 Deep Learning based SHM Techniques 23
2.4 Summary 25
Chapter 3: Image-driven Structural Steel Damage Condition Assessment Method ... 39
3.1 Introduction 39
3.2 Deep Learning Model Description 40
3.2.1 Architecture 41
3.2.2 Layers 42
3.2.3 Training CNN 43

3.2.4 Visualization check 46
3.3 Deep Learning based Structural Condition Assessment Procedure 48
3.3.1 Data Collection 48
3.3.2 Data Labeling 49
3.3.3 Defining and Training of Deep Learning Model 51
3.4 Case Study I: Perforated Steel Shear Link Beam 52
3.4.1 Specimen and FE Modeling 53
3.4.2 Deep Learning Implementation 54
3.4.3 Results and Discussion 55
3.5 Case Study II: Dogbone Steel Plate Specimen 60
3.5.1 Experimental Test Description 61
3.5.2 Deep Learning Implementation 62
3.5.3 Results and Visualizations 63
3.6. Concluding Remarks 64
Chapter 4: Brace Damage Detection for Concentrically Braced Frame Structures under Seismic Loadings 85
4.1. Introduction 85
4.2 Deep Learning based Brace Damage Detection 86
4.2.1 Brace Damage Detection Procedure 86
4.2.2 Deep Convolutional Neural Network 87
4.2.3 Visual Interpretation 89
4.3 Case Study: 6-story Steel Braced Frame Building 90
4.3.1 Prototype Building and Numerical Modeling 91
4.3.2 Ground Motions and Seismic Analysis 93
4.3.3 Potential Damage Scenario 94
4.3.4 Evaluation Tasks 95
4.3.5 Deep Learning Implementation 96
4.4 Discussions 99
4.5 Concluding Remarks 106

Chapter 5: Nondestructive Evaluation using Guided Ultrasonic Lamb Wave Signals for Plate Structures 121
5.1 Introduction 121
5.2 Deep Learning based Procedure for Guided Wave Damage Detection 122
5.2.1 Technical Procedure 122
5.2.2 Deep Learning Model Description 123
5.3 Analytical Simulation of Lamb Wave Interaction with Notch Damages 129
5.3.1 Piezoelectric Fundamentals 130
5.3.2 Guided Lamb Wave Propagation in Pristine Plate 130
5.3.3 Lamb Wave Interacting with Damages 135
5.4 Case Study of Deep Learning Method: Guided Lamb Wave based NDE 138
5.4.1 Test Setup 138
5.4.2 Data Collection 141
5.4.3 Data Preparation 142
5.4.3 Deep Learning Implementation 142
5.4.4 Results and Discussions 144
5.5 Concluding Remarks 149
Chapter 6: Bridge Condition Rating Data Modeling 181
6.1 Introduction 181
6.2. Deep Learning based Bridge Condition Assessment 182
6.2.1 General Procedure 182
6.2.2 Convolutional Neural Network 184
6.2.3 Cost Function and Statistical Interpretation 185
6.2.4 *K*-fold Cross Validation 188
6.3 Case Study on NBI Bridges in Maryland 188
6.3.1 Data Preparation 188
6.3.2 Deep Learning Implementation 190
6.3.3 Bridge Condition Assessment Results and Discussion 190
6.3.4 Steel Bridge Condition Deterioration Model 198

6.3.5 Bridge Condition Deterioration Modeling: Individual Bridge 199
6.3.6 Comparison with Markovian Method 201
6.4 Concluding Remarks 203
Chapter 7: Summary, Conclusion and Future Work 229
7.1 Summary 229
7.2 General Conclusion 235
7.3 Research Contribution 237
7.4 Future Work 239
References 241

List of Tables

Table 3-1. Categorization of damage levels 80

Table 3-2. Material hardening parameters of G20Mn5QT cast steel 80

Table 3-3. Loading protocols used for generating the training dataset..... 81

Table 3-4. Test results on FE simulation generated dataset..... 82

Table 3-5. Test results on photos of printed FE simulation generated images: Fracture with no residual drift..... 82

Table 3-6. Test results on photos of printed FE simulation generated images: Fracture with residual drift..... 83

Table 3-7. Test results on photos of printed FE simulation generated images: Fracture with no residual drift..... 84

Table 4-1. Deep CNN model validation test results in two evaluation tasks 120

Table 5-1. Damage parameters for a notch type of damage 180

Table 5-2. Damage parameters for breathing crack..... 180

Table 5-3. Damage region based Labeling 180

Table 6-1. Data sample size regarding to different record length..... 221

Table 6-2. Evaluation results for bridge deck (mean ± std., %) 222

Table 6-3. Evaluation results for bridge superstructure (mean ± std., %) 224

Table 6-4. Evaluation results for bridge substructure (mean ± std., %) 226

Table 6-5. Comparison between the Markovian method and proposed method 228

List of Figures

Figure 1-1. Impact on system resilience curves of two SHM technologies (Zhang et al., 2016)..... 2

Figure 2-1. Ultra-low cycle fatigue damages to a structural steel fuse member: the shear link beam designed for the eccentrically braced framed structures (Tong et al., 2018) 27

Figure 2-2. Damages in building structures during the 1994 Northridge Earthquake: (a) the 4-story steel framed HP building (Hall et al., 1995); (b) damaged brace was revealed after the dry wall has been removed (Sabelli, 2013) 28

Figure 2-3. Non-destructive testing of plate structures using guided lamb waves: (a) Piezoelectric Wafer transducers; (b) calibration on theoretical simulation of guided lamb waves (Shen & Giurgiutiu, 2014) 29

Figure 2-4. Deteriorating bridges: (a) the outer road bridge over the gasconade river in Laclede county (courtesy Missouri department of transportation); (b) a bridge over Monroe Street in Minneapolis (Jeffrey Thompson, MPR News) 30

Figure 2-5. Deep learning powered techniques for challenging computer vision tasks: (a) image classification (Krizhevsky et al., 2012); (b) object detection using Faster R-CNN (Ren et al., 2015); (c) duplicate work for bolt detection using Faster R-CNN . 32

Figure 2-6. Go knowledge learned by AlphaGo Zero (Silver et al., 2017) 33

Figure 2-7. Visual intuition for general features learned by different layers in the CNN model (Zeiler & Fergus, 2014) 34

Figure 2-8. Feature related pattern visualization for interoperating results from deep CNN models (Yu et al., 2016) 35
Figure 2-9. Difference between deconvolutional neural network, gradient backpropagation and guided backpropagation..... 36
Figure 2-10. Deep learning based artistic style: (a) images that combine the content of a photograph with the style of several well-known artworks (Gatys et al., 2015); (b) Glenn L. Martin Hall (2015) at University of Maryland, College Park; (c) artistic style of the Martin Hall..... 38
Figure 3-1. Architecture of adapted VGG-16 model based on the work by Simonyan and Zisserman (2014) 66
Figure 3-2. Illustration of mathematical operation of convolutional, ReLu and max pooling layer 66
Figure 3-3. Difference between deconvolutional neural network, gradient backpropagation and guided backpropagation..... 67
Figure 3-4. Image-driven structural damage condition assessment flowchart using deep learning algorithm..... 67
Figure 3-5. Cast steel shear link with perforated web: (a) picture of test specimen; (b) perforation pattern and dimensions; (c) finite element model..... 68
Figure 3-6. Hysteresis curves of perforated steel link specimen from FE analysis and experimental testing..... 69
Figure 3-7. Evolution of damage index of perforated shear link specimen under AISC loading protocol 69

Figure 3-8. Sample FE simulation-generated images for CNN model training 70
Figure 3-9. Sample learning curve from fine-tuning the CNN model with 8,600 training images for shear link 71
Figure 3-10. Visualization check on four sample images: (a) well-trained model with 92.3% testing accuracy; (b) poorly-trained model with 18.2% testing accuracy 72
Figure 3-11. Illustration of object distance and camera inclination angle 72
Figure 3-12. Samples of printed FE generated image 75
Figure 3-13. Visualization check results of experimental test image 76
Figure 3-14. Dogbone steel plate specimen: (a) picture of specimen in test machine; (b) picture of fractured specimen; (c) specimen geometry and dimensions 77
Figure 3-15. FE simulation of dogbone specimen: (a) meshed FE model; (b) hysteresis curves from FE simulation and experimental data; (c) damage index evolution and predicted crack initiation 78
Figure 3-16. Visualization of sample images with different damage severities in the dogbone steel plate specimen (the coordinates referring to 224x224 pixels) 79
Figure 3-17. Learning curve from fine-tuning the CNN model with 8,259 images for dogbone specimen 79
Figure 4-1. Deep learning based brace damage detection procedure 109
Figure 4-2. Brace damage detection procedure in the case study of a 6-story CBF building 109
Figure 4-3. Configurations of prototype 6-story CBF building model 110

Figure 4-4. Sensor configuration in proposed brace damage detection scheme for model validation..... 111
Figure 4-5. Pre-processing of acceleration response data for CNN model training: (a) original acceleration time history; (b) the short-time Fourier spectrum (STFS) in grayscale 111
Figure 4-6. Learning curves of trained CNN models for Task I and Task II 113
Figure 4-7. Feature visualization of three randomly picked input images 114
Figure 4-8. Schematic illustration of the optimal decision boundary to minimize the misclassification rate under the assumption of a uniform prior..... 115
Figure 4-9. Statistical analysis of the computed volume feature: (a) histogram; (b) estimation of class conditional probability density (H = Healthy; D = Damaged; V = Volume; P = Probability; pdf = probability density function)..... 116
Figure 4-10. Statistical analysis of the computed modified volume feature: (a) histogram; (b) estimation of class conditional probability density..... 117
Figure 4-11. Statistical analysis of the computed FNN feature: (a) histogram; (b) estimation of class conditional probability density..... 118
Figure 4-12. Statistical analysis of the computed CNN feature: (a) histogram; (b) estimation of class conditional probability density..... 119
Figure 5-1. Implementing procedure of deep learning based nondestructive damage evaluation for plate structures using guided Lamb wave signals 152
Figure 5-2. The adapted CNN model for nondestructive damage detection using Lamb wave signals based on the work by Simonyan and Zisserman (2014) 152

Figure 5-3. Lamb Wave excited by piezoelectric based sensor: (a) shear force interaction between sensor and plate surface; (b) decomposition of shear force into symmetric and antisymmetric mode 153

Figure 5-4. Experimental setup for damage detection using a pair of piezoelectric based sensors (a transmitter T- and a receiver) in a pitch-catch configuration..... 154

Figure 5-5. Damaged plates with notch (top) and breathing crack (bottom)..... 154

Figure 5-6. Three counts tone-burst input signal with central frequency of 100kHz 155

Figure 5-7. Dispersion curve of the aluminum plate with thickness (a) 2 mm (b) 3.17 mm 156

Figure 5-8. Tuning Curve of the aluminum plate with thickness (a) 2 mm (b) 3.17 mm 157

Figure 5-9. White noise effects on the collected Lamb wave signal from a pristine 2 mm-thick-plate: (a) noise free (b) SNR 15 dB (c) SNR 20 dB (d) SNR 25dB..... 159

Figure 5-10. Typical Lamb wave signals in notch damage scenarios: (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm 161

Figure 5-11. Typical Lamb wave signals in breathing crack damage scenarios (f_c=100 kHz, SNR=15dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm 163

Figure 5-12. Typical Lamb wave signals in notch damage scenarios (f_c=220 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm 165

Figure 5-13. Typical Lamb wave signals in breathing crack damage scenarios (f_c=220 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm 167

Figure 5-14. Typical Lamb wave signals in breathing crack damage scenarios (f_c=300 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm 169

Figure 5-15. Typical Lamb wave signals in breathing crack damage scenarios (f_c=300 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm 171

Figure 5-16. Learning curves for examining the effects from center frequencies: (a) 100kHz (b) 220 kHz (c) 300kHz 172

Figure 5-17. Learning curves for examining the effects from white noises (a) SNR 15dB (b) SNR 20 dB (3) SNR 25 dB 173

Figure 5-18. Learning curves for examining the effects from the number of damage bins (a) 8 bins (b) 16 bins 174

Figure 5-19. Visual features recognized by the trained CNN model for the notch damage scenarios (fc=100kHz, SNR=15dB) 175

Figure 5-20. Visual features recognized by the trained CNN model for the breathing crack damage scenarios (fc=100kHz, SNR=15dB) 175

Figure 5-21. Visual features recognized by the trained CNN model for the notch damage scenarios (x_d =150mm, fc=100kHz): (a) SNR=15dB; (b) SNR=20dB; (c) SNR=25dB 176

Figure 5-22. Visual features recognized by the trained CNN model for the breathing crack damage scenarios (x_d =150mm, fc=100kHz): (a) SNR=15dB; (b) SNR=20dB; (c) SNR=25dB 177

Figure 5-23. Visual features recognized by the trained CNN model for the notch damage scenarios (x_d =150mm, SNR=25dB): (a) fc=100kHz; (b) fc=220kHz; (c) fc=300kHz 178

Figure 5-24. Visual features recognized by the trained CNN model for the breathing crack damage scenarios (x_d =150mm, SNR=25dB): (a) fc=100kHz; (b) fc=220kHz; (c) fc=300kHz 179

Figure 6-1. Deep learning based bridge condition assessment 206

Figure 6-2. Data preparation for deep CNN model training 206

Figure 6-3. The architecture of the developed CNN model: (a) Model A with 1 convolutional layer (m=1); (b) Model B with 2 convolutional layers (m=2); (c) Model C with 3 convolutional layers (m=3) 207

Figure 6- 4. Sample learning curves of the developed CNN model for concrete bridge condition assessment: (a) deck; (b) superstructure; (c) substructure 208

Figure 6-5. The effect of training sample size on the prediction accuracy of developed CNN model for concrete bridge condition assessment (a) deck; (b) superstructure (c) substructure 209

Figure 6-6. Learned model distribution from concrete bridge data with the input: (a) [6,6,6,6,6]; (b) [7,7,7,7,7]; (c) [4,4,4,4,4]; (d) [5,5,5,5,5] 211

Figure 6-7 Illustration of the marking process to distinguish data samples with the same condition history: (a) input [8,8,8,7,7] with label [7]; (b) input [8,8,8,7,7] with label [6] 212

Figure 6-8. Sample learning curves of the developed CNN model for steel bridge condition assessment: (a) deck; (b) superstructure; (c) substructure 213

Figure 6-9. The effect of training sample size on the prediction accuracy of developed CNN model for steel bridge condition assessment: (a) deck; (b) superstructure (c) substructure 214

Figure 6-10. Learned model distribution from steel bridge data with the input: (a) [6,6,6,6,6]; (b) [7,7,7,7,7]; (c) [4,4,4,4,4]; (d) [5,5,5,5,5] 216

Figure 6-11. Typical deterioration curves for single bridge predicted by CNN model: (a) slowly degrading bridges; (b) notable degrading bridges; (c) bridges with repair actions; (d) bridges with unnormal interventions 218

Figure 6-12. Determined transition matrix of concrete bridges' (a) deck (b) superstructure (c) substructure; steel bridges' (d) deck (e) superstructure (f) substructure 220

Chapter 1: Introduction

1.1 Research Background and Motivation

Natural disasters have historically caused billions of dollars in direct economic loss due to damaged civil infrastructures all over the world. Structural condition assessment in such hazardous events has always been concerning the public and property owners with the uncertain risks to the public safety and the long-term property management. In order to rationally manage such uncertain hazardous risks, especially for the civil infrastructures, system resilience metrics have adequately been established such as the ones by Ayyub (2014 and 2015). Recently, researchers have also studied the feasibility of integrating the Structural Health Monitoring (SHM) technologies with resilient civil infrastructure design and establishing a resilience metric for rating candidate SHM technologies (Zhang et al., 2016). By explicitly considering the SHM technologies in the building design with resilience metrics, civil infrastructure's resilience level can be enhanced with rapid and accurate structural health condition assessment, especially for those post-hazardous inspections (see Figure 1-1). Therefore, research on smart SHM technology capable of rapid and accurate structural health condition assessment is appealing so that enhancement of the civil infrastructural resilience can be achieved by

reducing the uncertainty involved in the process of assessing the condition state of civil infrastructures and carrying out subsequent retrofit work.

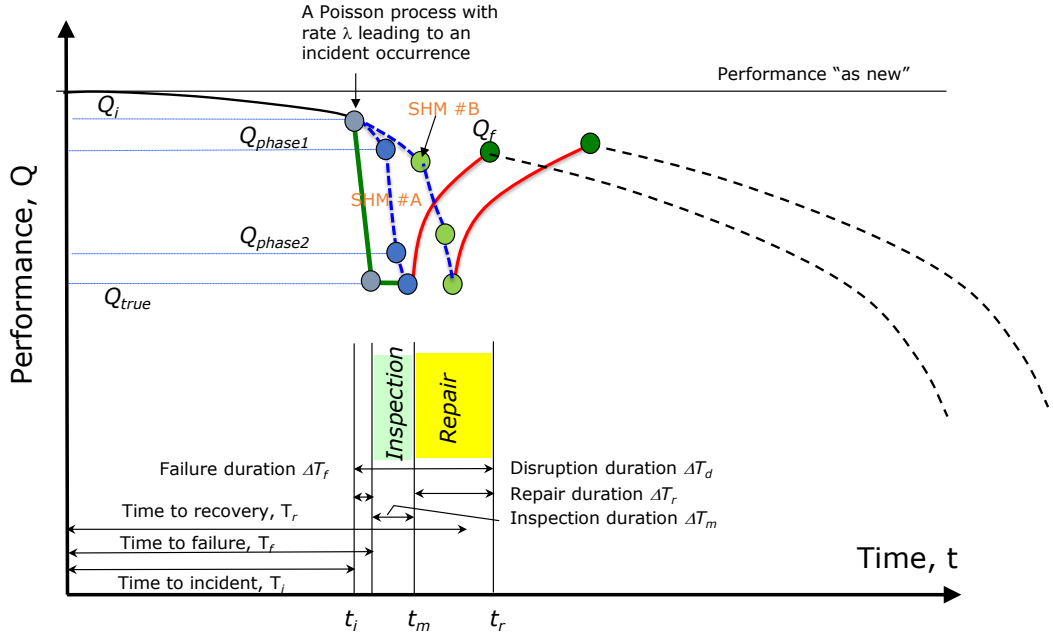


Figure 1-1. Impact on system resilience curves of two SHM technologies (Zhang et al., 2016)

Popularized by Hinton and others over the last decade, deep learning based algorithms have been reported recently to have achieved impressive success in many challenging fields (Hinton et al., 2006; LeCun et al., 2015). It is reported that deep learning powered image recognition is now performing better than human vision on many tasks (Krizhevsky et al., 2012). The AlphaGo and AlphaZero are other examples of the tremendous achievements by deep learning in recent years (Silver et al., 2017). Incorporating deep learning algorithms into SHM technologies for civil infrastructures

is promising with standing from two aspects: (1) deep learning based technologies have achieved impressive success in these challenging fields, suggesting the strong potential of success if extended to the SHM application; (2) deep learning algorithms have been found to be very effective in representing the hard-to-describe relationship or features in high dimensional data and thus can automate the feature-mining process without the need for explicit definition of features in advance. This has a strong potential in representing SHM data such as photo images, vibration data, and ultrasonic surface guided wave signals, which are typically in high dimension.

1.2 Research Objectives

This dissertation research aims to accomplish the overall goal of establishing a smart SHM technique based on the selected deep learning algorithm, which is expected to capable of automated structural health condition assessment and condition rating prediction for civil infrastructures. Specifically, the following objectives will be investigated,

i. Performing a literature review on structural health condition assessment technologies commonly used for civil infrastructures, so the special need of the proposed method can be identified. Also, with the specific application background clearly defined, the current research study can be targeted onto

more specific condition assessment tasks, so the effectiveness and practicality of the proposed method can be more conveniently evaluated;

ii. Performing a literature review on deep learning algorithms, with an emphasis on its pattern recognition application, especially in the field of computer vision where deep learning algorithms have been reported with the impressive success in traditionally challenging tasks;

iii. Performing an application study of selected deep learning algorithms, with an emphasis on a particular deep learning model termed Convolutional Neural Network (CNN), which has seen many successful applications in the computer vision field;

iv. Establishing a working procedure that incorporates the selected deep learning algorithm into a smart SHM technique for typical structural health condition assessment applications.

v. Applying the proposed method to solve a similar application problem that uses image data for structural steel condition assessment. Examine the effectiveness of the proposed method in such an image-driven application, and further study the implementation requirements such as how to effectively train the deep learning model, etc.

vi. Extending the study to time series data such as vibration data or guided ultrasonic surface wave signals for specific applications of structural condition assessment such as brace damage detection in concentrically braced frame

structures or nondestructive damage evaluation in thin plate structures. Examine the current research work in such applications using time series data, including working procedure for particular SHM tasks dealing with time series data, modification to accommodate time series input data, training method, and strategies, etc.

vii. Further extending the study to discrete data (neither images nor time series data), such as the bridge condition rating data from the National Bridge Inventory (NBI) data repository, for a specific application of bridge condition assessment. Examine again the components in the proposed method, including the working procedure for the SHM task dealing with discrete data, modification on input data and deep learning model architecture, training method and strategies, etc.

viii. Summarizing the findings and developing a general procedure for structural health condition assessment and condition deteriorating modeling using the deep CNN algorithm. Identifying the advantages and limitations of the method and propose potential improvements for future works.

1.3 Dissertation Organization

In this dissertation research, a smart SHM technique using deep learning algorithm has been proposed and studied in addressing the need for rapid and accurate structural

health condition assessment for civil infrastructures. The following chapters are organized as follows.

Chapter 2 reviews the previous research works in common structural health condition assessment problems in civil infrastructures, including structural steel fuse condition assessment in eccentrically braced frame structures, brace damage detection in concentrically braced frame structures, guided lamb wave based non-destructive testing (NDT) for plate structures, and bridge condition rating deterioration modeling based on the National Bridge Inventory (NBI) database. Additionally, previous research works in deep learning are reviewed with emphasis on computer vision related topics. Deep learning based SHM technologies published recently are also covered.

Chapter 3 presents the research work on the topic of image-driven structural steel damage condition assessment using deep learning algorithm. In this study, the deep learning algorithm has been applied to a similar application that uses image data for condition assessments. This study first examines the specific deep learning model used for solving the application problem. The proposed method is demonstrated and validated in two independent case studies using different sources of image data - finite element simulation generated contour plot images, printed photo images, and photo images taken in structural test experiments. As the first study in the dissertation, the research aims to examine the effectiveness of the proposed method in such an image-driven application, and further study the implementation requirements such as how to effectively train the deep learning model, etc.

Chapter 4 presents the research work on the topic of deep learning based brace damage detection for concentrically braced frame structures under seismic loadings. This chapter extends the proposed method to time series data such as the vibration response of structural frames subjected to earthquake ground motion. The current study further examined the research work on deep learning method for SHM, including working procedure, training method and strategy. Modification on the method of accommodating the time series data is also presented. The modified method is demonstrated and validated in a numerical simulation based case study of a 6-story CBF structure under seismic excitation. A statistical approach is used to analyze the results from the deep learning method.

Chapter 5 presents the research work on deep learning based non-destructive evaluation (NDE) in plate structures using guided Lamb wave signals. This chapter extends the research work to another form of time series data, guided Lamb wave signals. This study further examines the proposed method for this particular NDE application, including the working procedure, modification to time-series data, training method and strategies. The proposed method is demonstrated and validated in a case study of damage detection on aluminum plates with notch or breathing crack damages. The proposed method can effectively identify the occurrence, the location and the severity (or type) of the concerned damage in thin plates. The results from the deep learning model are examined by comparing with manual damage analysis.

Chapter 6 presents the research work on the topic of NBI bridge condition rating data modeling using the CNN method. This chapter aims to extend the application of the proposed method to discrete data (neither images nor time series data), such as the bridge condition rating data from the NBI data repository. This study examined the components in applying the deep learning method, including the working procedure, modification on input data, training method and strategies. Modification on the method including the architecture of the CNN model is needed and presented. The modified method is demonstrated and validated in a case study of the Maryland federal-aid state highway concrete and steel highway bridges from the NBI database. Discussion of the results is also included to offer insights into the proposed method.

Chapter 7 summarizes the research findings from the previous chapters and finalizes the general conclusions. The original contributions of the dissertation research are also presented. The advantages and limitations are discussed, with future work proposed for further improvement.

Chapter 2: Literature Review

2.1 Common SHM Techniques for Civil Infrastructures

In this section, a literature review has been performed in structural health condition assessment techniques commonly used for civil infrastructures, including structural steel fuse condition assessment in eccentrically braced frame structures, brace damage condition assessment in concentrically braced frame structures, nondestructive damage detection in plate structures, and bridge condition assessment. These four topics are selected in the basis of: (1) common civil infrastructures such as building structures, plate structures, and bridges, and their primary structural health condition assessment demand; (2) with the access to the necessary data resources for training the deep learning algorithm, including experimental data, numerical simulation or theoretical modeling data, and public panel data such as the National Bridge Inventory database, so the research work can be conveniently investigated and validated.

2.1.1 Structural Steel Fuse Condition Assessment in EBF

New design trends for seismic resistant structures is to incorporate fuse members (sacrificial elements to dissipate energy) into structures so that damage can be confined to a limited number of structural fuse members while other structural members would

remain undamaged during design level earthquakes. Examples of structural fuses are buckling restrained braces, steel shear links (Figure 2-1) and slit steel plate wall with buckling restrained cover plate (see, e.g., Hajjar et al., 2013; Vargas & Bruneau, 2009). For example, Qu et al. (2018) developed buckling-restrained brace (BRB) with replaceable steel angle fuses which offer ease of post-earthquake examination of fuse damage, convenient and prompt replacement of damaged fuses, and reuse of the buckling restraining elements. Such structure design concept is also appealing to structural health monitoring (SHM) since condition assessment or monitoring work can now be concentrated to a limited number of fuse members. For such smart fuse members instrumented with sensors, rapid inspection of fuse members for damages likely inflicted by strong earthquakes could be carried out in an efficient way, which would accelerate damage condition assessment and thus enhance structural resilience through quick and less subjective inspection practice. Currently, visual inspection is the typical practice for post-hazard condition assessment of structural fuses (e.g., Clifton et al., 2012). Intensive labor, high cost, and variable results are typical of such manual operation. Furthermore, in buildings, structural members are often hidden behind fireproof coating and drywall, and thus damage of these hidden steel fuse members are difficult to detect, often requiring removal of coverings and thus time-consuming and costly. For efficient operation with instrument-assisted inspection, researchers have been looking into automated structural health monitoring technology such as computer vision based or acoustic emission based method (Sun et al., 2017).

In assessing the damage condition of structural fuse members, image-based structural condition assessment shows strong potential in addressing the rapid inspection needs in practical applications: convenient data collection by snapping photos (e.g., using smartphones, wearable imaging devices or drones) and availability of well-designed algorithms for accurate image pattern recognition. Several research works have been reported in using images to assist with structural condition assessment (Jacobsen et al., 2010; Zhang & Bai, 2015; Li et al., 2017; Cha et al., 2017; Chen & Jahanshahi, 2017). For examples, Jacobsen et al. (2010) identified the maximum drift angle during an earthquake event by using strain patterns in steel slit plate walls to train an artificial neural network. Despite the promising potential of using image-based method for condition assessment, a robust algorithm to identify structural damage from collected data is difficult to define because the detection results are usually sensitive to such pre-defined damage features. To address this need of high-level features that might not be accurately described with current mathematical models, deep learning methods have been adopted which can learn such features from training data.

2.1.2 Brace Damage Condition Assessment in CBF

Destructive earthquakes have posed a continuous threat to lives and properties worldwide, so the structural damage condition assessment plays an essential role during the post-hazard repair and recovery efforts (e.g., see Figure 2-2). Structural health

monitoring can reduce economic loss and minimize service interruption to civil infrastructures. A new trend in structural health monitoring of civil infrastructures is to use sensor arrays to automate structural abnormally or damage detection. An automated damage condition assessment technique with a consistent level of accuracy is desirable for post-hazard inspection with reduced inspection time and thus enhanced the resilience of civil infrastructures.

Popular structural damage detection methods for buildings during strong earthquakes can be broadly classified into parametric model-based and nonparametric methods. Parametric model-based structural identification methods involve the use of mathematical models to represent structural system behavior in either time or frequency domain. Damage to structures is usually reflected by changes in structural parameters such as modal parameters, stiffness and damping indices. The benefits of using parametric models for structural identification-based damage detection include their direct relationship with physically meaningful quantities such as stiffness and mass, and their suitability for analysis, prediction, fault diagnosis and control (Petsounis & Fassois, 2001; Ni et al., 2009; Vanik et al., 2000; Xia & Hao, 2000). Non-parametric damage detection methods do not require *prior* information on the nature of the structural model but suffer from their inability to determine the severity of damages.

2.1.3 NDE in Thin Plate Structures

Nondestructive evaluation (NDE) is appealing to structural health monitoring (SHM) of civil infrastructures because structural health condition can be evaluated without causing destructive damage, which can be both economic and time efficient. Ultrasonic damage detection and characterization are commonly used in NDE of aerospace structures. Lamb wave was first discovered by H. Lamb (1917), and it was found to be existing in thin plates or shells. The Lamb wave theory has been well-established and can be found in numerous textbooks (e.g., Achenbach, 2012; Rose, 2014). The mechanisms of the Lamb wave propagation in thin plates or shells have been rigorously examined over the years, aiming to provide better guidance for damage identification. The modeling and simulation of the Lamb wave propagation are primarily through analytical or numerical approaches. Many advanced finite element modeling (FEM) techniques have been investigated in parallel for the application of damage identification (Hayashi et al., 2002; Su & Ye, 2004; Gibson & Popovics, 2005; Lu et al., 2008). Analytical approaches have also shown good performance in simulating Lamb wave propagation in plates (Li & Zhang, 2008; Shen & Giurgiutiu, 2014).

Since the Lamb wave can propagate over a large distance in plates or shells, it is particularly advantageous for ultrasonic scanning and has been widely used as an NDE tool for SHM purposes (Raghavan & Censnik, 2007; Leckey & Parker, 2014; Li et al., 2018). In plate structures, Lamb wave can be excited and measured in a variety of means. Piezoelectric sensors such as the Piezoelectric lead zirconate titanate (PZT)

element (Dimitriadis & Fuller, 1991; Badcock & Birt, 2000; Kee & Zhu 2013; Zhou & Zhang, 2014;) and piezoelectric paint sensor (Li & Zhang, 2008), have been shown to be effective in Lamb wave generation and reception, and are especially suitable for the application as an active SHM technique because of their cost efficiency, compact size and many other appealing aspects (Su et al., 2006). Applying Lamb wave for damage identification is an interpretation process of damage features contained within the guided wave signals. Identifying the damage feature is critical to the success of detecting and characterizing the structural damage. Suitable features should be selected from raw wave signals according to engineering experience and professional knowledge, which makes feature selection a challenging and time-consuming task. Conventional features, such as the wave energy, wavenumber, frequency, etc., are typically discovered with certain signal processing methods, which can be categorized in: (1) time domain methods (Sohn & Farrar, 2001; Zang et al., 2004); (2) frequency domain methods (Heller et al., 2000; Kessler et al., 2002; Ruzzene, 2007); (3) time-frequency domain methods (Wang et al., 2004; Park et al., 2007; Poddar et al., 2011). To automate the process of damage feature identification, several researchers have started to investigate machine learning based approaches. For examples, Agarwal and Mitra (2014) studied a Lamb wave based automatic damage detection that uses machine pursuit to preprocess the wave signals as input and then uses support vector machine (SVM) or artificial neural network (ANN) as a classifier for the damage detection. The method is verified based on a finite element simulation-based case study

of a thin aluminum plate. The results reflected that the machine learning algorithm is very suitable for such a classification problem with a successful detection rate near 95%. Dworakowski et al. (2015) also studied a Lamb wave based damage detection using the ANN combining with a predefined damage index as input. The ANN was trained with numerical simulation generated wave signals and tested with experimentally collected data. The results indicated that it is capable of performing successful classification that uses simulation generated data for the training of ANN, but the results of such a classification depend on the degree of similarity between the training and the real collected dataset. As an evolved form of ANN, deep learning has improved the ability of algorithms to find optimal features and started demonstrating predominant performance in many tasks (e.g., Krizhevsky et al., 2012).

2.1.4 Bridge Condition Data Modeling

Bridge condition assessment is crucial to a successful Bridge Management System (BMS) as a proper measurement of bridge performance in communication with bridge engineers and administrations (e.g., see Figure 2-4). Since 1972, the National Bridge Inventory (NBI) initiated by the Federal Highway Administration (FHWA) provides the primary source of data for bridge management in the United States (Chase et al., 1999; Frangopol et al., 2001). According to the National Bridge Inspection Standards (NBIS), the federal-aid highway system bridges shall be routinely inspected every two

years and the condition ratings of bridge components (e.g., bridge deck, superstructure, and substructure) will be recorded in the NBI database. The condition ratings are one-digit numbers that characterize the general condition of the entire component being rated. For example, the condition of three subsystems of a bridge may be rated on a scale from 0 to 9 (with 9 being “excellent” and 0 being “failed”).

A time-dependent reliability analysis requires the knowledge of deterioration models which predict the future structural condition and form the basis for life-cycle monitoring and maintenance planning. The condition ratings have been utilized as the basis of bridge deterioration modeling to assist with the BMS in optimizing maintenance, rehabilitation, and replacement decisions. Markovian methods are reported as the most common probabilistic model and have been extensively studied in modeling such deterioration process (Cesare et al., 1992; Frangopol et al., 2004; Morcous, 2006; Ranjith et al., 2013). The Markovian methods assume the bridge deterioration process as a Markov-Chain based stochastic process that describes the condition deterioration with a probabilistic transition from one state to another during one inspection interval. The probabilistic transition of the facility condition state is defined by the transition matrix, which can be obtained through regression-based nonlinear optimization, Bayesian maximum likelihood, and Markov-Chain Monte Carlo method (Micevski et al., 2002; Wellalage et al., 2014). Through probabilistic modeling, Markovian methods can capture the uncertainties within the bridge deterioration due to unobserved explanatory variables, the presence of measurement

errors, and the inherent stochasticity of the deterioration process (Madanat et al., 1995). However, there exist certain limitations concerning the use of Markovian methods. For example, the Markov-Chain model assumes the future condition depends only on the current facility condition other than the condition history (Madanat et al., 1997).

To circumvent the limitations of Markovian methods, machine learning such as the Artificial Neural Network (ANN) based method has been investigated by many researchers as an alternative approach to bridge deterioration modeling (Sobanjo, 1997; Tokdemir et al., 2000; Li & Burgueño, 2010; Huang, 2010). For example, Sobanjo (1997) utilized a multilayer ANN to predict the condition rating of the bridge superstructure based on the bridge age. Tokdemir et al. (2000) used the ANN model for predicting the bridge sufficiency rating using age, traffic, geometry, and structural attributes as explanatory variables. Li and Burgueño (2010) conducted comparative studies of several soft computing methods (including ANN) for predicting bridge abutment condition ratings in the state of Michigan. Huang (2010) analyzed historical maintenance and inspection data of concrete decks in Wisconsin using an ANN model with a reported accuracy of 75% in classifying the true condition rating.

2.2 Deep Learning Algorithms for Pattern Recognition

Popularized by Hinton and others over the last decade, deep learning methods have been reported recently to have achieved impressive success in many challenging fields

(Hinton et al., 2006; Lecun et al., 2015). It is reported that deep learning powered image recognition is now performing better than human vision on many tasks (Krizhevsky et al., 2012; Ren et al., 2015, also see Figure 2-5). The AlphaGo and AlphaZero are other examples of the tremendous achievements by deep learning in recent years (Silver et al., 2017, also see Figure 2-6).

2.2.1 A little History about Deep Learning

Deep learning, as one of the many streams within machine learning, has had a long and rich history but gone by many different names. The artificial neural networks (ANN) is one of them and has been well-known since the1980s. The modern term, deep learning, is recognized by the machine learning community starting with the paper (Hinton et al., 2006), in which the very deep neural nets was found to be capable of well-trained if proper strategies were applied. However, in the early days, conventional machine learning techniques were limited in processing high-dimensional data in its raw form due to inefficient training algorithms and lacking computational power. Hence, conventional machine learning based technique is typically used as a semi-automated pattern recognition system, because a hand-craft feature is usually desired to reduce the data dimension and such feature needs careful engineering and considerable domain expertise.

Since the training issue was reported with working solutions in 2006 (Hinton et al., 2006), deep learning has been found to be very good at discovering intricate structure or recognizing patterns in high-dimensional data (Lecun et al., 2015). Impressive successes have been reported in many research fields, such as those beating-records applications in image classification, object detection, speech recognition (Krizhevsky et al., 2012; Hinton et al., 2012; Girshick, 2015). Unsurprisingly, deep learning powered technology has also been applied in the medical field to support biomedical diagnosis (Mirowski et al., 2008; Ma et al., 2015).

2.2.2 Deep Learning based Pattern Recognition

Pattern recognition, namely, is the recognition of patterns or regularities in data, which is the typical research focus in many deep learning based applications, especially for the image classification and object detection tasks in computer vision. The image classification tasks can be challenging as the ImageNet (Deng et al., 2009) which requires the computer vision model to classify 1,000 objects among 1-million images. In a more complicated task, the object detection needs to precisely mark the location of one or more objects occurred within the image. Deep learning powered image classification has been reported with performing better than humans in the visual recognition tasks (Krizhevsky et al., 2012; He et al., 2016), and the performance has continuously been reported with beating records (Lecun et al., 2015).

The Convolutional Neural Network (CNN) is found to be the deep learning model behind the many success in computer vision based challenging tasks. The CNN is a specialized kind of neural network (NN) for processing data that has a known grid-like topology, such as time-series data, which can be treated as the 1-D grid of samples at given time intervals, and image data, which can be thought of 2-D grid of pixel information. The examples of typical CNN models with reported beat-record performance can be the AlexNet (Krizhevsky et al., 2012), GoogleNet (Szegedy et al., 2014), VGGNet (Simonyan & Zisserman, 2014), and ResNet (He et al., 2016). It can be found that the architecture becomes deeper and deeper, starting from 8 layers in the AlexNet to 156 layers in the ResNet. These advanced CNN models also integrated the latest deep learning technologies, such as the rectified linear unit (ReLu) (Nair & Hinton, 2010), the Xavier initialization (Glorot & Bengio, 2010), and the ResNet block (He et al., 2016). These technologies (e.g., the ReLu and the Xavier) either helps to accelerate the training of CNN models or provide improvements of data flows in CNN model architectures (e.g., the ResNet block).

Another appealing property of these advanced deep learning models is the transfer learning. Transfer learning refers to the situation where what has been learned in one setting (base dataset) is exploited to improve the generalization in another setting (target dataset) (Goodfellow et al., 2016). It has been observed that the generalized features in early layers appear to be general regardless of the exact cost function and natural image dataset, and in later layers seem to be task-specific (Yosinski et al.,

2014). In image classification, the general feature can be an edge detector while the task-specific feature can be the face of a human or the wheel of a car. Visual evidence (see Figure 2-7) of this finding can be found in the reference by Zeiler and Fergus (2014). When the target dataset is significantly smaller than the base dataset, transfer learning enables training a large target network without overfitting; recent studies that have taken advantage of this fact obtained state-of-the-art results from transfer learning (Yosinski et al., 2014; Zeiler & Fergus, 2014). Transfer learning is also appealing to the application of structural health condition assessment for two reasons: (1) the sophisticated design of the architecture of those successful CNN models can be taken advantage of, and (2) the size of acquirable database for brace damage detection (order of thousands) is far smaller than the image data (order of millions) in the ImageNet database, from which those sophisticated deep CNN models were trained.

While deep CNN models have demonstrated impressive performance on image classification and object detection, there is a lack of deep insight in how and why the CNN models work so well and as such model improvement is restricted to trial-and-error process (Zeiler & Fergus, 2014). To address the need for visualizing features, researchers have investigated visualization techniques that give insight into the function of intermediate feature layers and the feature evolution during training (Zeiler & Fergus, 2014; Simonyan et al., 2013; Springenberg et al., 2014). Visual examples can be seen in Figure 2-8. The deconvolutional neural network used by Zeiler and Fergus (2014) provides a non-parametric view of invariance, showing which patterns

from the training set activate the feature map. Simonyan et al. (2013) further elucidated the function of the deconvolutional neural network for feature-related pattern visualization by computing image-specific class saliency map, which approximates the highly non-linear scoring function mapped from CNN model by evaluating its first-order Taylor expansions as Eqn. (2-1):

$$\begin{aligned} S_C(I) &\approx S_C(I_0) + \frac{\partial S_C}{\partial I}|_{I_0} \cdot (I - I_0) \\ &= \frac{\partial S_C}{\partial I}|_{I_0} \cdot I + \left(S_C(I_0) - \frac{\partial S_C}{\partial I}|_{I_0} \cdot I_0 \right) \end{aligned} \tag{2-1}$$

where S_c represents the scoring function; I denotes input variable (images) and I_0 is a given specific image. The saliency map is established based on the derivative respect to the input image of the score function that its magnitudes indicate which pixel needs to be adjusted the least to affect the score. They pointed out that the deconvolution layer is either equivalent or similar to computing the gradient of the visualized layer. It is effectively corresponding to the gradient back-propagation through a convolutional layer. The reason for the "either equivalent or similar to computing the gradient" is the ReLU layer. Therefore, guided-backpropagation (Springenberg et al., 2014) can be utilized to achieve enhanced accuracy in calculating the gradient back-propagation through a convolutional layer. According to the comparison made by Springenberg et al. (2014), the differences between these techniques are shown in Figure 2-9.

In addition to feature related pattern visualization techniques, researcher have also investigated the deep learning algorithm in artistic style representations with

aiming to understand the mechanism of the deep CNN model (Gatys et al., 2015). In their research, an artificial system based on a Deep Neural Network (i.e., the pre-trained VGGNet by Simonyan and Zisserman in 2014) was proposed to create artistic images of high perceptron quality. The neural representations were used to separate and recombine the content and style of arbitrary images, providing a neural algorithm for the creation of artistic images. Moreover, in light of the striking similarities between performance-optimized artificial neural networks and biological vision, their work offers a path forward to an algorithmic understanding of how humans create and perceive artistic imagery. Their research work also provides visual intuition of how the deep CNN models process high-dimensional pixel information (e.g., an image with 224 pixels by 224 pixels by 3 color channels) regarding the image contents and style, which are hard to be described in conventional mathematical models (see Figure 2-10).

2.3 Deep Learning based SHM Techniques

Appealing by the deep learning algorithms, a few related research works that also investigate deep learning based SHM techniques, especially using the CNN, have been published recently (Liang et al., 2016; Cha et al., 2017; Kim et al., 2017; Kumar et al., 2018). For examples, Cha et al. (2017) proposed a deep learning based crack damage detection approach for concrete structures. The study proposed a vision-based method using a deep architecture of CNN for detecting concrete cracks without calculating the

defect features. Comparative studies were conducted to examine the performance of the proposed CNN using traditional Canny and Sobel edge detection methods. The results indicated that the proposed method shows quite better performances and can indeed find concrete cracks in realistic situations. Kumar et al. (2018) developed a prototype system that uses the deep convoluted neural network (CNN) to classify multiple defects in sewer CCTV images including root intrusions, deposits, and cracks. The viability of this approach in the automated interpretation of sewer CCTV videos was demonstrated with an average testing accuracy of 86.2% using trained CNN model based on 12,000 images collected from over 200 pipelines. Kim et al. (2017) used a region-based CNN (i.e., the faster R-CNN proposed by Ren et al. (2015)) for detecting equipment on construction sites by taking advantage of transfer learning. It was reported with a mean average precision of 96.33% in detecting five classes in the benchmark data, including a dump truck, excavator, loader, concrete mixer truck, and road roller. Liang et al. (2016) conducted a study on big data-enabled multiscale serviceability analysis for aging bridges. The study investigated deep learning based techniques to evaluate the bridge serviceability according to real-time sensory data or archived bridge-related data such as traffic status, weather conditions, and bridge structural configuration.

2.4 Summary

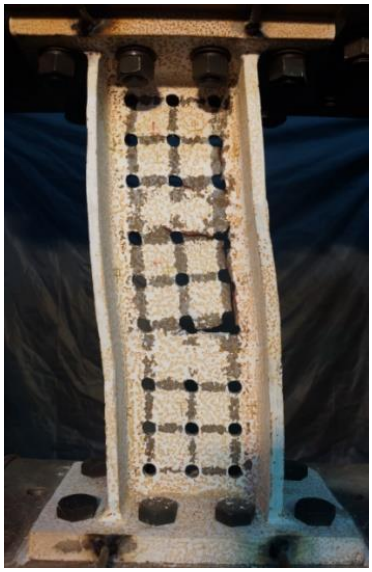
Through the literature review, it has been identified that there exists a demand of advanced SHM techniques with being capable of rapid and accurate structural health condition assessment for civil infrastructures, at least the four common topics within the area of building structures, bridges and plate structures that have been investigated in this dissertation. Comparing to traditional SHM technologies, deep learning algorithms are promising to be utilized on SHM applications with automated damage pattern recognition.

In terms of deep learning algorithms for pattern recognition, the Convolutional Neural Network (CNN) is found to be the foundation of many success behind the challenging tasks in computer vision. Transfer learning may facilitate the beginning of the research since the demand for knowledge is softened regarding the design of a sophisticated CNN architecture, and thus a greater chance to succeed. Feature related pattern visualization techniques may offer the path to the insights of the CNN's mechanism so that the results can be interpretable and better guidance may be offered to the community.

It is noticed that the community of Civil Engineering started the investigation of deep learning algorithms for engineering applications. Comparing to the related research works that have been published recently, especially to those who also used CNN, the dissertation research distinguishes from the following aspects: (1) for image-driven applications, the proposed SHM method for structural health condition

assessment is aiming to provide more quantitative damage information the ultra-low cycle fatigue condition of structural steel fuse members in eccentrically braced framed structures; (2) various types of SHM data have been investigated in the dissertation, including images, time series data and discrete data. Modification of each data formats to achieve optimal results is also presented; (3) results interpretation has been implemented in a different angle, such as the feature related pattern visualization techniques for image-driven applications with comparing to visual intuitions and the statistical interpretations for non-image-driven applications.

(a) 

(b) 

(c) 

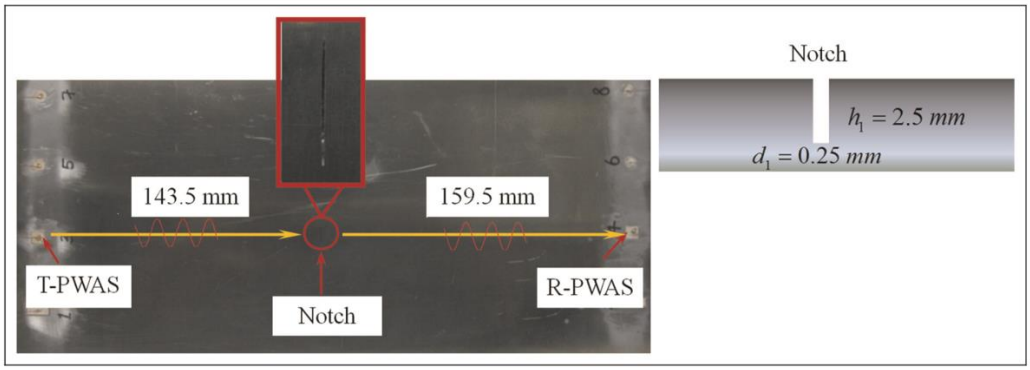
Figure 2-1. Ultra-low cycle fatigue damages to a structural steel fuse member: the shear link beam designed for the eccentrically braced framed structures (Tong et al., 2018)

(a) 

(b) 

Figure 2-2. Damages in building structures during the 1994 Northridge Earthquake: (a) the 4-story steel framed HP building (Hall et al., 1995); (b) damaged brace was revealed after the dry wall has been removed (Sabelli, 2013)

(a)



(b)

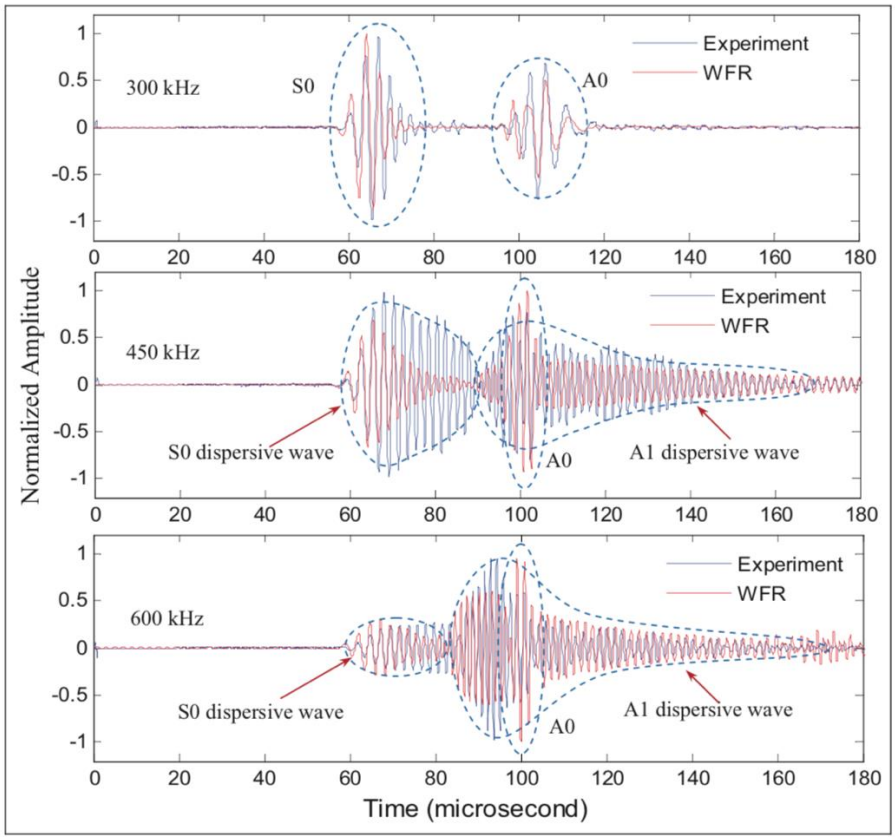
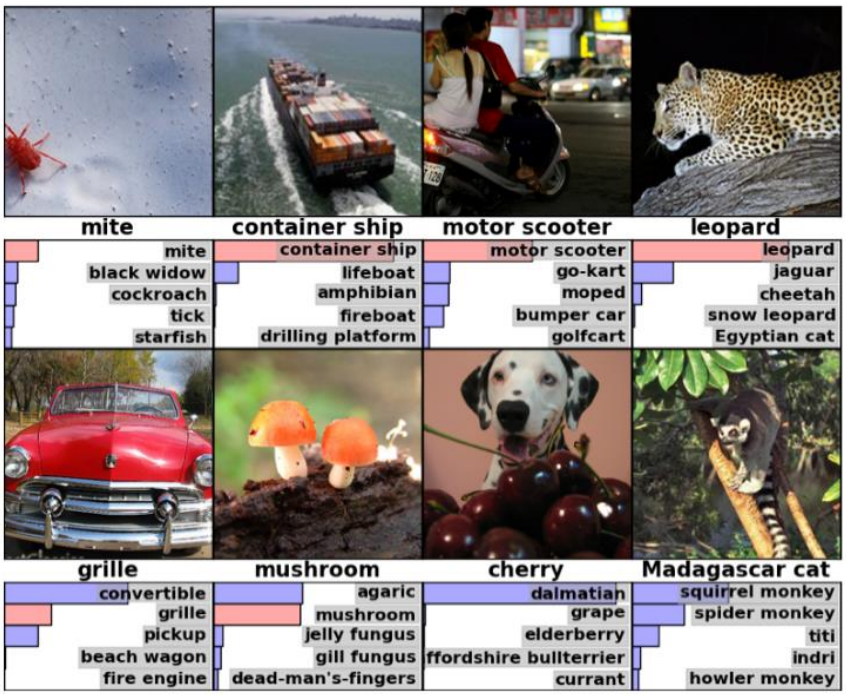


Figure 2-3. Non-destructive testing of plate structures using guided lamb waves: (a) Piezoelectric Wafer transducers; (b) calibration on theoretical simulation of guided lamb waves (Shen & Giurgiutiu, 2014)

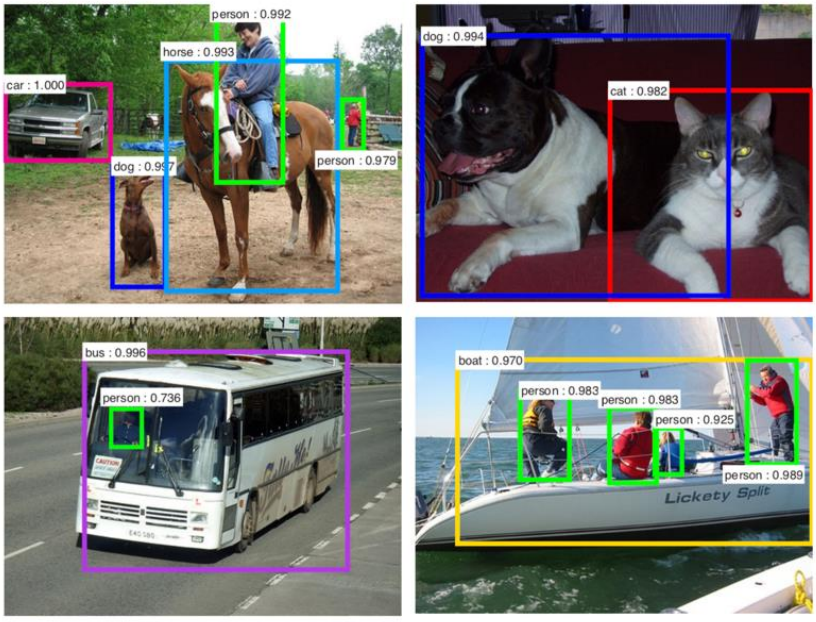
(a) 

(b) 

Figure 2-4. Deteriorating bridges: (a) the outer road bridge over the gasconade river in Laclede county (courtesy Missouri department of transportation); (b) a bridge over Monroe Street in Minneapolis (Jeffrey Thompson, MPR News)


mite
mite
black widow
cockroach
tick
starfish
container ship
container ship
lifeboat
amphibian
fireboat
drilling platform
motor scooter
motor scooter
go-kart
moped
bumper car
golfcart
leopard
leopard
jaguar
cheetah
snow leopard
Egyptian cat
grille
convertible
grille
pickup
beach wagon
fire engine
mushroom
agaric
mushroom
jelly fungus
gill fungus
dead-man's-fingers
cherry
dalmatian
grape
elderberry
ffordshire bullterrier
currant
Madagascar cat
squirrel monkey
spider monkey
titi
indri
howler monkey

(a)


person : 0.992
horse : 0.993
car : 1.000
dog : 0.957
person : 0.979
dog : 0.994
cat : 0.982
bus : 0.996
person : 0.736
boat : 0.970
person : 0.983
person : 0.983
person : 0.925
person : 0.989
Lickety Split

(b)

(c)

Figure 2-5. Deep learning powered techniques for challenging computer vision tasks: (a) image classification (Krizhevsky et al., 2012); (b) object detection using Faster R-CNN (Ren et al., 2015); (c) duplicate work for bolt detection using Faster R-CNN

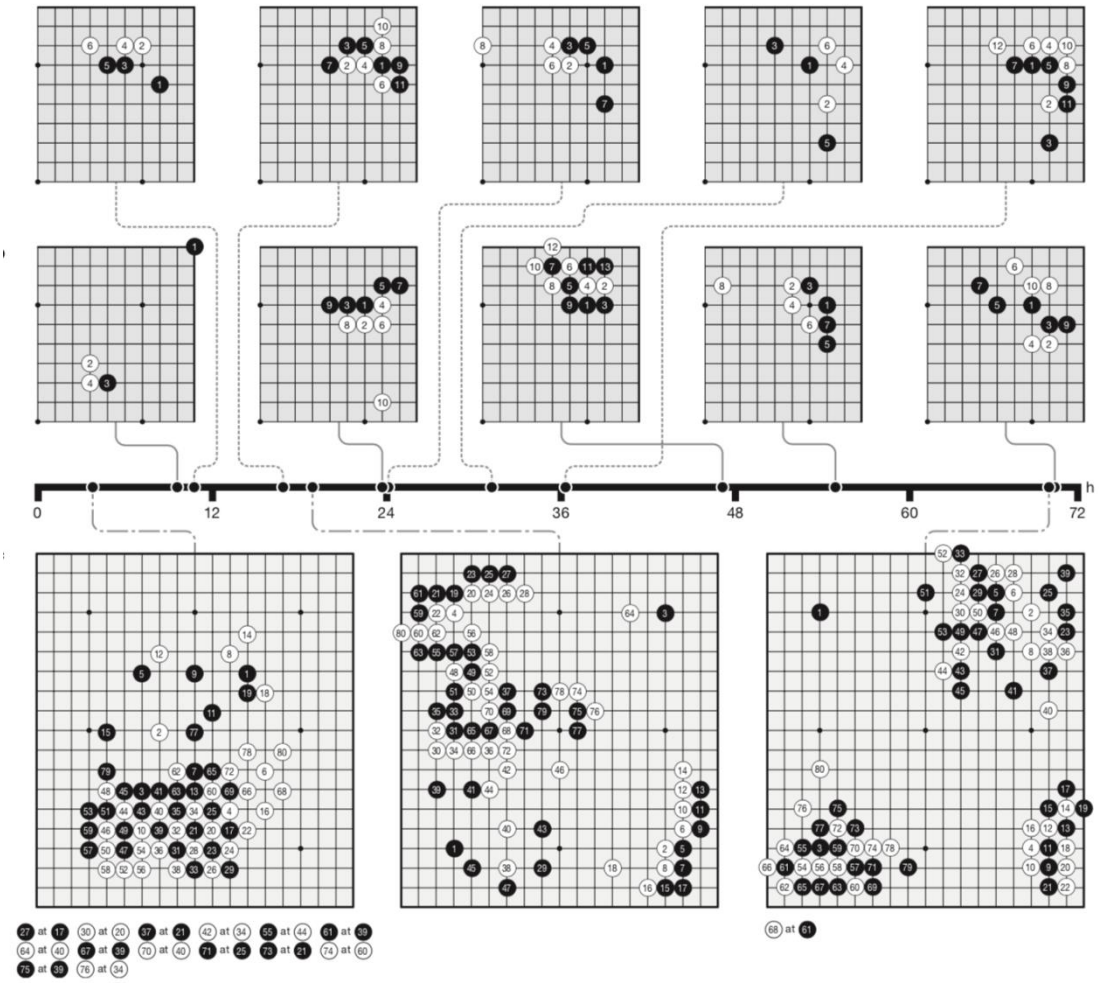


Figure 2-6. Go knowledge learned by AlphaGo Zero (Silver et al., 2017)

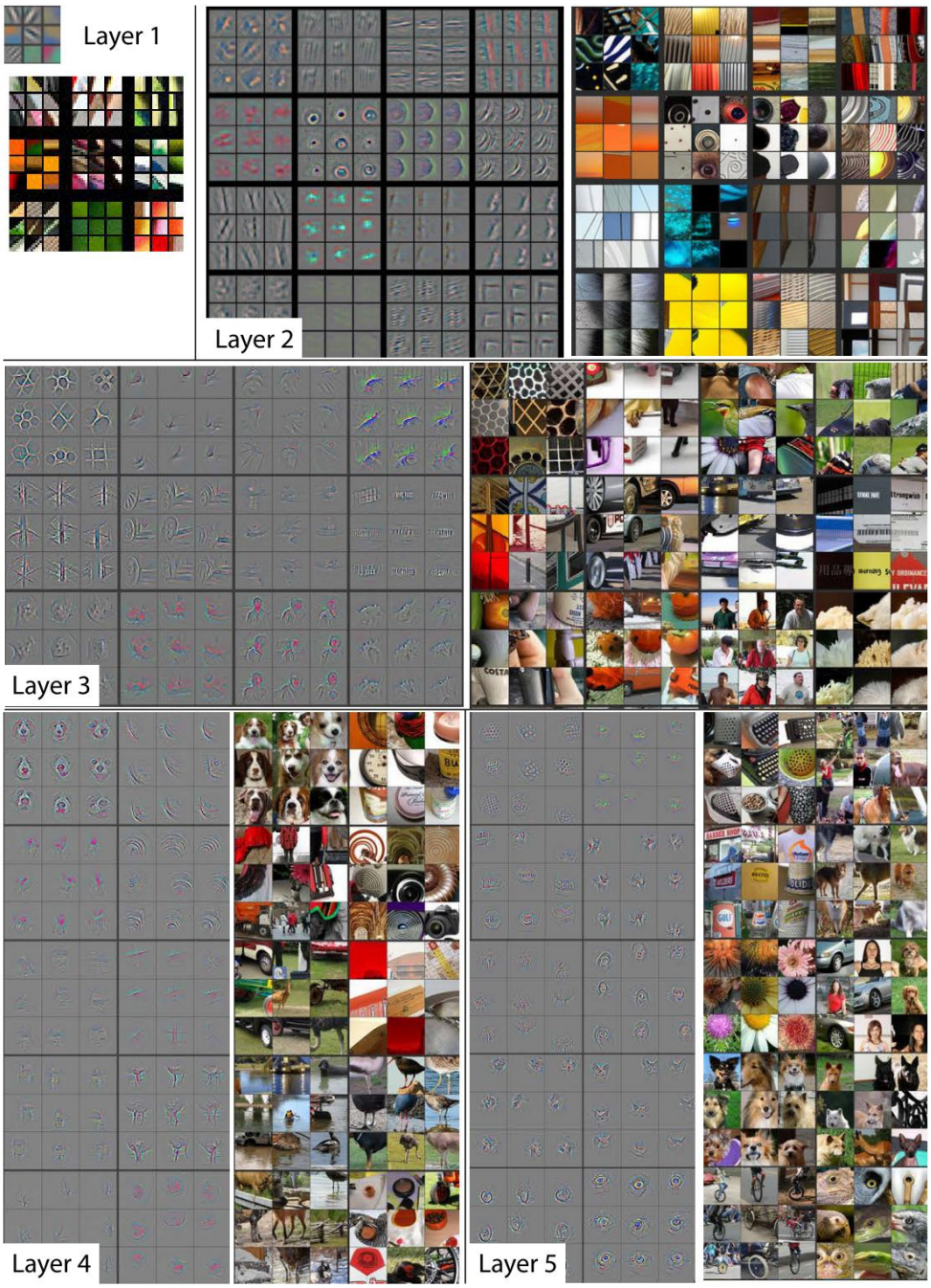
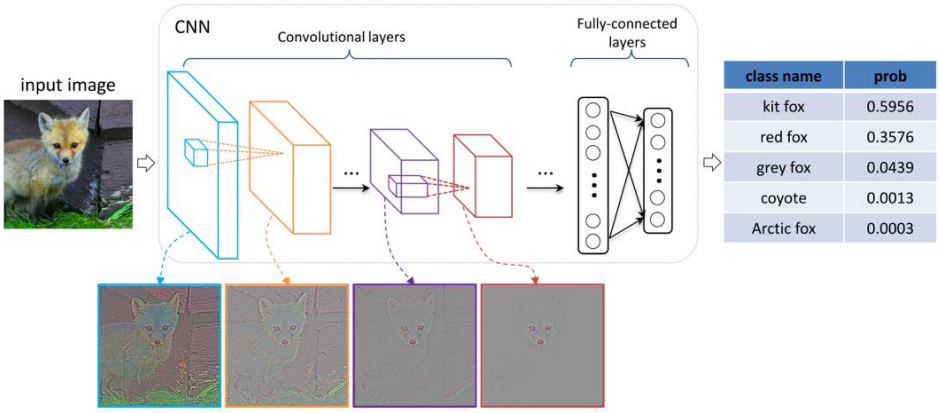
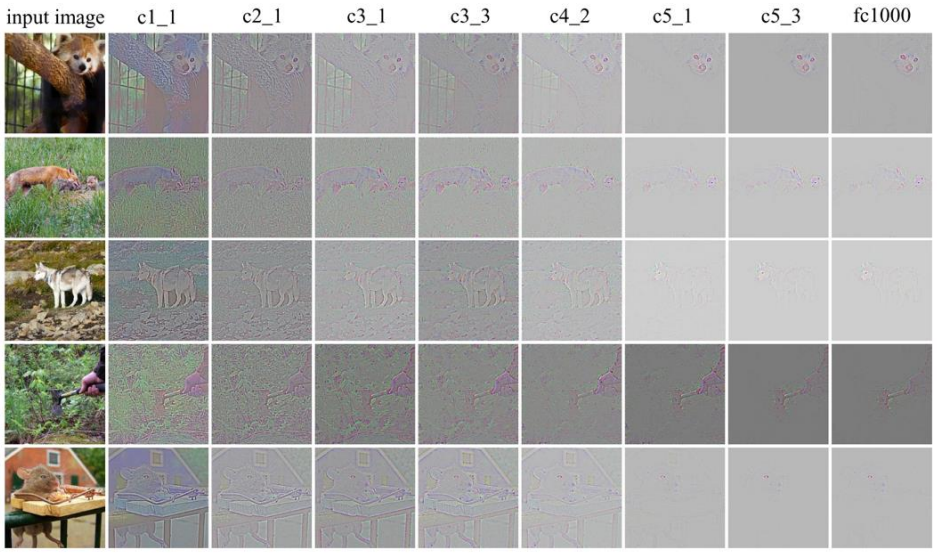


Figure 2-7. Visual intuition for general features learned by different layers in the CNN model (Zeiler & Fergus, 2014)



(a)



(b)

Figure 2-8. Feature related pattern visualization for interoperating results from deep CNN models (Yu et al., 2016)

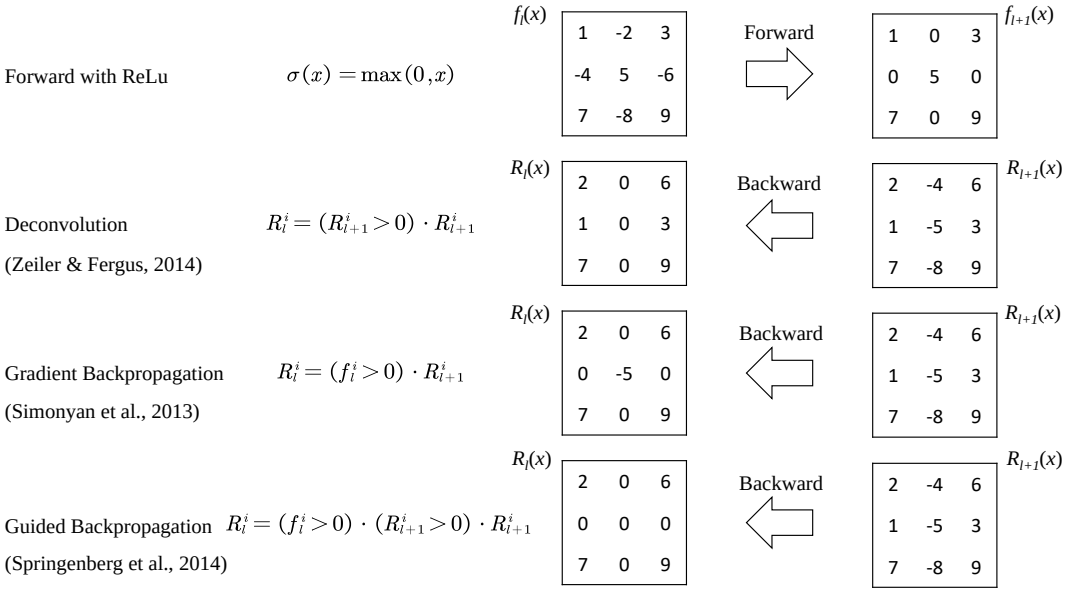
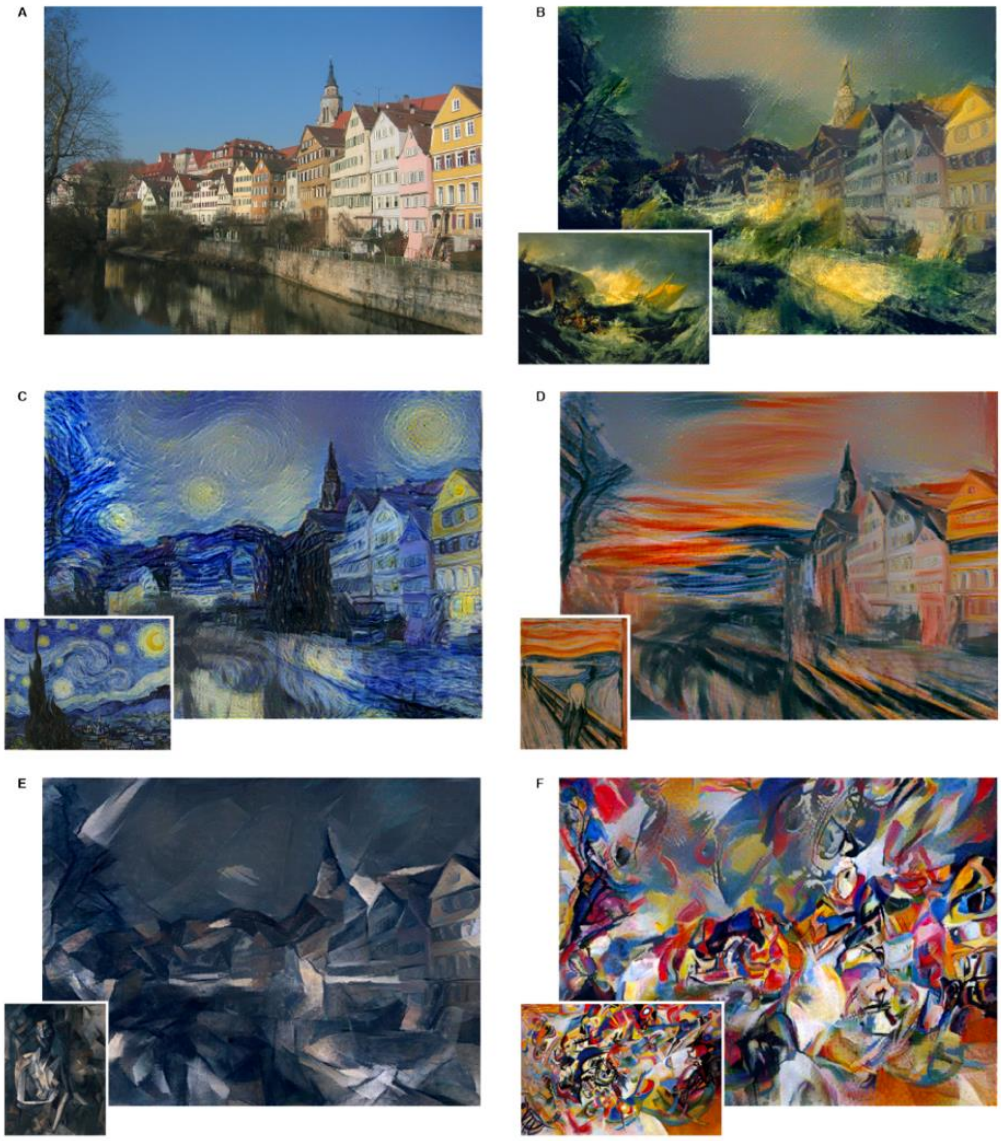


Figure 2-9. Difference between deconvolutional neural network, gradient backpropagation and guided backpropagation



(a)

(b)

(c)

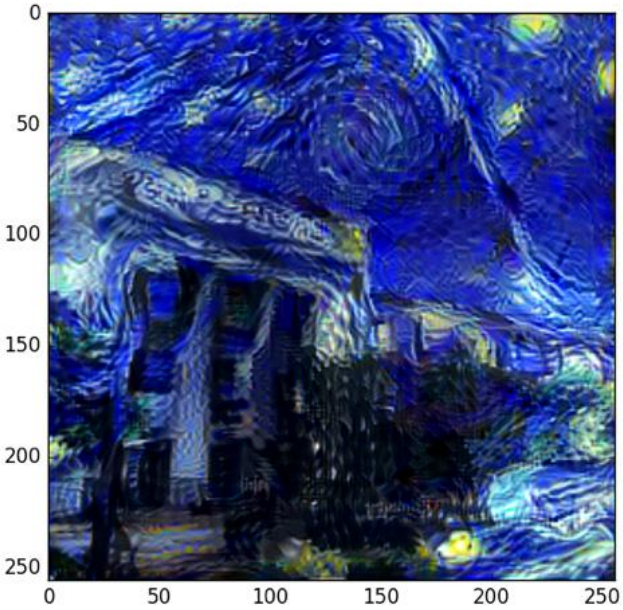


Figure 2-10. Deep learning based artistic style: (a) images that combine the content of a photograph with the style of several well-known artworks (Gatys et al., 2015); (b) Glenn L. Martin Hall (2015) at University of Maryland, College Park; (c) artistic style of the Martin Hall

Chapter 3: Image-driven Structural Steel Damage Condition Assessment Method

3.1 Introduction

This study investigates applying deep learning method to an application of image-driven rapid inspection of structural steel fuse damage condition. Firstly, a procedure for incorporating a customary deep learning model into structural condition assessment is presented, in which a micromechanical fracture index is defined to label the damage condition of the structural steel fuse member. The proposed method is demonstrated in a case study of a replaceable shear link beam, which is designed as a structural fuse member for eccentrically braced frames. A deep Convolutional Neural Network (CNN) model is adapted for the image-driven structural damage condition recognition in this case study. This case study validates and optimizes the method by using FE simulation-generated images as data resources. The printed photos of selected FE simulation-generated images and one experimental image are also used to investigate optical effects and to identify potential issues for using realistic images. The method is further verified by a second case study involving photo images taken during experimental testing of a dogbone steel plate specimen. The results from both case studies suggest

that the proposed method can effectively identify damages in structural steel fuse members.

3.2 Deep Learning Model Description

Deep CNN is adapted here for image-based structural condition assessment because of its advancement and demonstrated success in computer vision field. However, a few issues discussed next distinguish the present study from the research works on image classification in the computer vision field. The first one is that the label of the structural condition images is assigned to reflect the quantitative damage condition of the structural elements; therefore, a damage index has to be established first to characterize the damage. Another issue is that generating the millions of training image data as observed in the ImageNet database (Deng et al., 2009) is not practical in near future simply because of the limited availability of field reconnaissance data and experimental test data. Due to the limited amount of available training data, the training strategy needs to be adjusted to avoid the overfitting problem. Finally, no benchmark testing data is available for evaluating the performance of the trained model. Thus, the testing data needs to be adequately designed to fit the objective of damage condition assessment (e.g., considering the relation of damage condition to the loading history and different loading protocols).

Details of deep learning or the CNN can be found in references (e.g., the review article by LeCun et al., 2015 or the book by Goodfellow et al., 2016). A brief description that helps readers to gain necessary background knowledge on the specific algorithm of the deep learning model used in this study is presented next. Additionally, a saliency map based visualization technique is also used to interpret results. The adapted CNN model is based on the VGG-16 model pre-trained by Simonyan and Zisserman (2014).

3.2.1 Architecture

The architecture of CNN is specialized for processing the 2D grid-like topology of input images, and the CNN model usually includes multiple stacks of alternating convolution and max-pooling layers to function as feature extractors, followed by a small number of fully connected layers as the classifier (Goodfellow et al., 2016). The architecture of the adapted CNN model is shown in Figure 3-1. The input images are converted into numerical vectors with RGB values at a size of 224 (pixels) x 224 (pixels) x 3 (color channels: red, green and blue). The vectors are then forwarded into multiple convolutional and pooling layers, outcoming with features followed by fully connected layers. Finally, a SoftMax layer is used to score the activations into a vector consisting of C (possible classes) values with each ranging from 0 to 1 and all summating to 1. The prediction is made by choosing the class with the highest score.

3.2.2 Layers

Convolutional layer, max pooling layer, and fully connected layer are frequently used in the adapted CNN model. The convolutional operation computes features based on a local field of the preceding layer as described in Eqn. (3-1),

$$y = \sum (X_{F\times F} \cdot w_{F\times F}) + b \tag{3-1}$$

where y is the output of the convolutional computation on a local receptive field ($X_{F\times F}$) of input data (X); $w_{F\times F}$ is the weights of the filter; b is the filter bias; $\sum$ operates an element-wise summation; F is the kernel size of the local field window. The additional control parameters of a convolutional layer are: (1) the stride (S), controlling how many pixels the filter moves one step; (2) the zero padding (P), which is to make the output data size as integer (i.e., $(W\text{-}F+2P)/S+1$ and $(H\text{-}F+2P)/S+1$) by padding zeros at the each side of the input data; and (3) W and H, which are the width and height of input data respectively.

The max pooling layer performs maximum operation over a pooling window. The fully connected layer is the conventional artificial neural network with full connection to all parameters in the preceding layer as described by Eqn. (3-2),

$$Y_{n\times 1} = W_{n\times m} \cdot X_{m\times 1} + B_{n\times 1} \tag{3-2}$$

where Y is the output of fully connected layer; W and B are weights and biases matrix; m and n are the size of input and output vector respectively.

The mathematical operation described in Eqn. (3-1) and (3-2) are linear and thus, without any further operation, a linear function is characterized by combinations

of those layers, which makes the network eventually be a linear regression model. Therefore, an activation layer is introduced with non-linearity into the network to describe a nonlinear function. In the adapted CNN model, ReLu, which is abbreviated from Rectified Linear Units introduced by Nair and Hinton (2010), is extensively used as the activation layer because it has been observed that deep CNN with ReLu layers can be trained several times faster than sigmoid function (Nair & Hinton, 2010; Krizhevsky et al., 2012; He et al., 2016). The operation of an ReLu Layer is shown in Eqn. (3-3),

$$\sigma(x) = \max(0, x) \tag{3-3}$$

where x represents each element in the input vectors. An example of using a 5x5 input matrix to illustrate the operation of a convolutional, ReLu and max pooling layer is shown in Figure 3-2.

3.2.3 Training CNN

Training CNN for representation learning is an optimization process, which aims to minimize the output loss (or error) on the given training data sets. The first step of this process is to determine which part (i.e., CNN model parameters) contribute most to the output loss so that adjustments (or weights update) can be applied to those sensitive areas. Mathematically this is achieved by computing the gradients of loss function at the CNN parameter values, which are determined by using the backpropagation

method. The backpropagation refers to the method for computing the first order gradients of CNN through repeatedly applying chain rule (Goodfellow et al., 2016). For example, the backpropagation can be used for computing the gradient of function ***f(x)***, where ***x*** is the vector of inputs. In CNN, this function would be the loss function, and the input vectors consist of the image pixel RGB values, as well as the weights and biases in the neural network. In the adapted CNN models, the SoftMax with the Multinomial Logistic Loss (Bishop, 2006) is used as the loss function defined in Eqn. (3-4) below,

$$\begin{cases} z^i = \dfrac{\exp(y^i)}{\sum_{i=1}^{n} \exp(y^i)} \\ L(W) = -Y_g^T \cdot \ln(Z) \end{cases} \tag{3-4}$$

where y^i (i=1,2, 3, …, C) are the features fed into the SoftMax layer; C is the number of possible classes; W denotes the weights and biases, and z^i represents a SoftMax function that scores proceeding features with each ranging from 0 to 1 and all summating to 1; Z describes the score vector (z_1, z_2, z_3, …, z_c) and Y_g is assigned with ground truth vector (y_{g1}, y_{g2} , y_{g3}, …, y_{gc}); $L(W)$ is the loss function or objective function that evaluates the distance from the current model to the target. In the classification problem, the ground truth is usually an indicator vector with 1 in the target neuron and 0 in the rest. For example, in a 5-classes classification problem, given the ground truth in the 3rd neuron, the vector is (0,0,1,0,0). If the score vector Z = (0.1,0.1,0.6,0.1,0.1),

the loss value L(W) =-[1*ln(0.6) +0*ln(0.1)*4]=-ln(0.6). It can be noticed that the closer the score to 1 in the target neuron, the less the loss value would be.

Parameter adjustments or weights update are then orchestrated based on calculated gradients with an attempt to minimize the loss. Generally, the adjustments would follow the steepest direction (gradients) with respect to the trend of decreasing the loss as Eqn. (3-5),

$$p_{new} = p_{old} - \eta \frac{\partial L}{\partial p_{old}} \tag{3-5}$$

where *p* denotes each parameter (including weights and biases) in the CNN model and the learning rate (η) is a parameter specified by the user. A higher learning rate means that bigger steps are taken in the weight updates and thus, it would take less time for the model to converge on an optimal set of weights, while too small a learning rate will slow down the gradient descent and hence the overall training process.

While the training of a deep CNN model seems to be straight-forward by repeatedly applying backpropagation and weights update to eventually minimize the value of the loss function, convergence or over-fitting is a common problem obstructing the training process which in turn requires a large training dataset. Fine-tuning or transfer learning has been suggested to circumvent this issue by many researchers. The fine-tuning or transfer learning means that the initialization of the CNN model is based on a pre-trained model rather than from random generation. In practice, fine-tuning is appealing because of two restraints: large enough training dataset (on the order of

millions) to avoid overfitting problem and computing demand. Fine-tuning of a pre-trained CNN model can substantially reduce the required training images and training time by transferring the well-trained ability of feature extraction.

3.2.4 Visualization check

While deep CNN models have demonstrated impressive performance on image classification and object detection, there is a lack of deep insight in how and why the CNN models work so well and as such model improvement is restricted to trial-and-error process (Zeiler & Fergus, 2014). To address the need for visualizing features, researchers have investigated visualization techniques that give insight into the function of intermediate feature layers and the feature evolution during training (Zeiler & Fergus, 2014; Simonyan et al., 2013; Springenberg et al., 2014). The deconvolutional neural network used by Zeiler and Fergus (2014) provides a non-parametric view of invariance, showing which patterns from the training set activate the feature map. Simonyan et al. (2013) further elucidated the function of the deconvolutional neural network for feature-related pattern visualization by computing image-specific class saliency map, which approximates the highly non-linear scoring function mapped from CNN model by evaluating its first-order Taylor expansions as Eqn. (3-6):

$$S_C(I) \approx S_C(I_0) + \frac{\partial S_C}{\partial I}|_{I_0} \cdot (I - I_0)$$
$$= \frac{\partial S_C}{\partial I}|_{I_0} \cdot I + \left(S_C(I_0) - \frac{\partial S_C}{\partial I}|_{I_0} \cdot I_0\right) \quad (3\text{-}6)$$

where S_c represents the scoring function; I denotes input variable (images) and I_0 is a given specific image. The saliency map is established based on the derivative respect to the input image of the score function that its magnitudes indicate which pixel needs to be adjusted the least to affect the score. They pointed out that the deconvolution layer is either equivalent or similar to computing the gradient of the visualized layer. It is effectively corresponding to the gradient back-propagation through a convolutional layer. The reason for the "either equivalent or similar to computing the gradient" is the ReLU layer. Therefore, guided-backpropagation (Springenberg et al., 2014) was proposed to achieve enhanced accuracy in calculating the gradient back-propagation through a convolutional layer. According to the comparison made by Springenberg et al. (2014), differences between these methods are shown in Figure 3-3. The guided back-propagation method will be used to visualize the feature-related patterns recognized by the CNN model.

3.3 Deep Learning based Structural Condition Assessment Procedure

The general procedure for deep learning based structural condition assessment is illustrated in Figure 3-4: (1) collecting data; (2) labeling images; (3) defining the deep learning model; (4) training and testing the model, while some modification and customization steps are also included for structural condition assessment. It is also seen that the feature visualization technique prescribed in Section 3.2.4 is employed to understand the high-level features identified by the deep learning model.

3.3.1 Data Collection

In this study, the focus is given to replaceable steel structural fuse members as seismic energy dissipation devices, in which the damage condition is usually controlled by ultra-low cycle fatigue failure. Whitewash paintings have been widely used in laboratory testing for visualizing the plastic deformation distribution in steel structures; thus, the flaking patterns of the whitewash coatings on steel surface provide a potential source of image data for such damage assessment purpose. However, due to the unknown amount of required training data and high costs of doing experiment tests, the contour plots of the accumulated equivalent plastic strain from FE simulation of structural fuse members are first employed as the training image data for demonstrating and optimizing the proposed damage assessment procedure. Contours plots of peak strains (including elastic and plastic strains) were used by Jacobsen et al. (2010) to

estimate drift angle identification. Afterward, real experimental images of flaking patterns from whitewash paintings will be used to further examine the procedure.

3.3.2 Data Labeling

Data labeling clusters the data into desired groups of damage conditions to supervise training of the deep learning model in this application. A micromechanical damage index is adapted to characterize the damage of structural fuse members so that the clustering of different damage conditions can be quantitatively evaluated. The critical failure of structure fuses is usually governed by ductile fracture or ultra-low cycle fatigue since such sacrificial structural element is designed to dissipate energy by allowing significant plastic deformation under seismic loading. Definition of ultra-low-cycle fatigue can be found in references (e.g., Bleck et al., 2009). Analytical studies (Rice & Tracey, 1969; Hancock & Mackenzie, 1976) indicate that the ductile fracture initiation depends on the interaction of triaxiality and the equivalent plastic strain, with a larger triaxiality hastening the onset of the fracture, where the triaxiality refers to the mean or hydrostatic stress normalized by the effective or Von Mises stress. Consequently, the fracture is predicted when the equivalent plastic strain exceeds a critical value. Henceforth, the damage index is computed as the ratio of accumulated equivalent plastic strain to critical plastic strain in Eqn. (3-7). Similar damage index

has also been used by other researchers (e.g., Prinz & Richards, 2006; Fell et al., 2006; Chi et al., 2006) for evaluating seismic steel fuse devices.

$$\text{Damage index (DI)} = \frac{\text{Accumulated equivalent plastic strain}}{\varepsilon_{p,critical}} \tag{3-7}$$

and the critical plastic strain is taken as,

$$\varepsilon_{p,critical} = \alpha \exp\left(-1.5\,\frac{\sigma_m}{\sigma_e}\right) \tag{3-8}$$

where σ_m is the hydrostatic stress, σ_e is the von Mises stress, and α is a material constant. The ratio of the hydrostatic and von Mises stress is the stress triaxiality. Fracture initiation is declared once the above damage index exceeds 1.0 over a characteristic length of l^*. In this study, the material constant α and the characteristic length l^* are set to be 2.6 and 0.2 mm respectively.

For ease of structural condition scoring, the extent of damage is classified into five damage level groups: no damage, minor damage, moderate damage, severe damage, and imminent fracture. For each group, the damage index range is given in Table 3-1. The index range is assumed to be evenly divided into five damage level groups while an uneven division is also possible if deemed necessary. Additionally, under cyclic loading, the triaxiality value can switch signs in tension and compression stages (positive in tension and negative in compression), then it will cause $\varepsilon_{p,\text{critical}}$ to have the same sign as the triaxiality. However, the accumulated equivalent plastic strain is monotonically increasing with positive values, thus the damage index would

decrease during the compression cycle. In this study, the damage level label for each image is assumed to be based on the maximum value of the historical damage index values in the fuse member. This assumption is made to mimic the flaking behavior of whitewash paint which is irreversible in paint flaking occurrence.

3.3.3 Defining and Training of Deep Learning Model

A 16-layer CNN model (VGG-16) pre-trained by Simonyan and Zisserman (2014) is adapted as the CNN model for transfer learning in this study. The original application of this model is to classify 1,000 classes from a large-scale image classification database. Adapting this model for structural damage condition assessment, the possible prediction classes (i.e., the output number of the last layer) is modified to five damage condition groups.

Training of the deep learning model depends on the training dataset size. Generally, a deep learning model can be trained by starting from scratch in which all parameters are randomly initialized, however doing so usually requires a large training dataset to avoid overfitting problem. To circumvent this issue, fine-tuning of a pre-trained model using a smaller dataset has been employed as a solution by many researchers (Goodfellow et al., 2016; Yosinski et al., 2014; Zeiler & Fergus, 2013). In fine-tuning, the parameters of adopted layers are initialized from the pre-trained model. However, which part of the model should be fine-tuned is still unclear in the beginning

and needs to be optimized for best results. Therefore, to find the best suitable training strategy, different training strategies were employed: (1) training from scratch; (2) treat the pre-trained CNN model as fixed feature extractor and only train the classifier at the end of the neural network; (3) fixing a few early convolutional layers due to its generalized feature extraction and only training the later layers; (4) fine-tune the CNN model on all the layers.

Testing is to evaluate the performance of the trained CNN model on an independent dataset which should also have a sufficiently large size for statistical significance.

3.4 Case Study I: Perforated Steel Shear Link Beam

In this section, the proposed method will be used for damage condition assessment of a steel shear link beam. Link beams are employed to serve as fuse members in steel eccentrically braced frames (EBF). In the EBF structures, shear links (its length e is shorter than $1.6*M_p/V_p$, referred to as "shear link" or short link) are generally preferred, since they are more ductile than the long links ($e > 2.4*M_p/V_p$, referred to as "flexural link" or long link) with the same section sizes. M_p and V_p denote the plastic moment and shear capacity of the steel section (AISC, 2010).

3.4.1 Specimen and FE Modeling

The dimensions of the perforated link specimen (G20Mn5QT cast steel) tested by Zhang (2015) are shown in Figure 3-5. Cyclic reversal displacement loading protocol prescribed by the AISC Seismic Provisions (2010) was adopted for both FE analysis and experimental tests. The FE model of the same shear link was built in a general FE software - ANSYS Mechanical (2017) using Solid185, an 8-node brick solid element. The combined Chaboche kinematic hardening and nonlinear isotropic hardening rules (Chaboche, 1989 and 1991) are adapted to simulate the hardening effects as described by Eqn. (3-9),

$$\sigma = \sigma_0 + \frac{C_1}{\gamma_1}(1 - e^{-\gamma_1 \varepsilon_p}) + R_\infty (1 - e^{-b\hat{\varepsilon}_p}) \tag{3-9}$$

where σ_0 denotes the initial yield stress; C_1 and γ_1 are the Chaboche kinematic hardening parameters; R_∞ and b characterize the nonlinear isotropic hardening behavior. These parameters were calibrated with experimental data as listed in Table 3-2 and good agreement between FE analysis results and experimental data can be seen in Figure 3-6. Details of the experimental setup, test specimens, and loading protocols can be found in Tong, et al. (2018). The calibrated FE simulation results were then used to calculate the damage index. The evolution of the damage index was plotted along with the applied AISC loading protocol in Figure 3-7. It should be noted that the FE simulation results under the particular AISC loading protocol were only used for generating the CNN model test database that is representative of the link beam behavior

in an earthquake. However, the actual loading condition of the link beam may differ from the AISC loading protocol. To partially remove this effect, twelve loading protocols listed in Table 3-3 were used to generate training data that are independent of the testing data (from AISC).

3.4.2 Deep Learning Implementation

The accumulated plastic strain contours from FE simulation were used to generate the image data. Although the FE generated images are employed for training the CNN model, replicating as many details in real images as possible is always preferred to make the images realistic. Here, the accumulated equivalent plastic strain contours are generated in gray-scale with nine shades as shown in Figure 3-8. For the testing database, 1,000 images were generated by capturing contour plots at each load step until the fracture initiation (the end of 0.02 rad cycle for this link specimen, see Figure 3-7). The size of the testing database was fixed in the following parametric study. However, the size of the training database varied (3,200 vs. 8,600) to investigate the effects of different training data size.

The training of the CNN model was implemented in the Caffe Framework (Jia et al., 2014) using a workstation with a single GPU (model: Nvidia GeForce Titan X) under the Ubuntu 14.04 LTS operation system. The learning rate was set to 0.001 as the initial value and decayed with a ratio of 0.1 per 300 steps. The initialization of the

CNN model varied with different training strategies, and the convergence of the training process is indicated by the consistently high testing accuracy followed with relatively low training loss as shown in Figure 3-9 for a typical learning curve.

3.4.3 Results and Discussion

3.4.3.1 Training Data Size Effect

As the training data sets in this case study include several thousand images, which is far less than the typical order of millions of images (e.g., in the ImageNet database), overfitting would be expected if the model was trained from scratch via such small data set. Also, Xavier parameter initialization (Glorot & Bengio, 2010) and the drop-out technology (Srivastava et al., 2014) applied in the VGG16 would randomly affect the gradient updating and eventually make the performance of the trained model slightly different from each training episode. Considering these effects, the testing results were averaged on the 1,000 testing images over 10 training episodes for each batch of training data and training strategies. The average testing accuracy results are given in Table 3-4. It is found from this study that as the size of the training data increases, the average testing accuracy keeps improving. It is recommended to have a training data size of at least several thousand images if a good pre-strained CNN model is available for fine-tuning.

3.4.3.2 Training strategy

As mentioned in Section 3.3.4, two strategies are available for the training: training from scratch or fine-tuning. For the fine-tuning, it can also be subdivided into three: fine-tuning all layers; fine-tuning all layers except for the first two layers; fine-tuning only the classifier (the last fully connected (fc) layer). The testing results for different training strategies are presented in Table 3-4. The results suggest that "fine tuning all layers" is the best strategy here. Another interesting note is that fine-tuning the classifier and all layers except for the first two layers also yields acceptable performance comparable to the "fine tuning all layers" option when the model is trained with 8,600 images.

3.4.3.3 Feature-related Pattern Visualization

Feature-related pattern visualization has been performed for two reasons: (1) providing an intuitive visual indication whether the CNN model is identifying the relevant features for structural damage condition assessment; (2) using the visual indication, rank the adopted strategy for optimizing the CNN model by tuning the training parameters. The guided backpropagation in Section 3.2.4 offers such a tool to visualize the gradients of the prediction corresponding to each pixel of the input image. The saliency map can show the discriminative patterns that the trained CNN model has used for classification. The visualization results of the trained CNN model in this case study are shown in Figure 3-10. It is seen that the well-trained CNN model is based on visual

patterns similar to the human visual inspection. For this particular damage assessment of perforated steel link beam, human intuition would consider the rectangular bar patterns between the holes in the link beam web as the features for damage level evaluation where the plastic deformation concentration occurs. The width and color shades of those bars would be closely related to the target damage condition under consideration. These bar patterns in the link beam web can also be explained by the physical mechanism underlying the low-cycle fatigue induced damage in the steel link beam. On the other hand, the models with poor performance can be seen to have looked into irrelevant visual patterns. Therefore, this visualization tool can be used for validating the training strategy as well as checking whether model optimization is guided in the right direction.

3.4.3.4 Optical Effects

In order to verify the proposed deep learning based structural damage condition assessment procedure under more realistic photograph settings, the trained model was tested with three types of photo images of printed pictures selected from the generated FE image testing database. The three image categories are characterized as follows: (1) damage level 5 with no residual drift, (2) damage level 5 with residual drift and (3) moderate damage levels with no residual drift. The damage level and corresponding damage index of these three image categories are listed in Table 3-5. The FE generated images were first printed out using a laser printer (model: Brother HL-L2380DW) with

a resolution of 600 x 600 dpi. The size of the printed pictures was scaled to match the real dimensions of the perforated steel shear link specimen in the experimental testing. The distance between the holes was measured as 62 mm which is close to the design value of 60 mm (Figure 3-5 (b)).

The investigation focuses on the optical effect (e.g., light condition, optical distortion, etc.) to see that has any effect on the assessment result, a camera was used to take pictures of the printed pictures which were then used as test images. Considering the popularity and convenience of smartphone camera, rear camera on an iPhone 6S (Apple model A1634) was used here. Three different object distance values (28 cm, 35 cm, and 42 cm) from camera to object and four different camera inclination angles (0°, 5°, 10° and 15° horizontal angle between image sensor axis and perpendicular line to the object) were considered in the test to account for possible camera imaging condition during field inspection. An iPhone tripod was used so these values can be precisely controlled. The definitions of object distance and camera inclination angle are sketched in Figure 3-11. The camera picture mode is set as square size and mono-color with room light condition. The images are taken at the original size of 640 pixels x 640 pixels and are cropped manually to keep only the perforated shear link portion and resized to 224 pixels x 224 pixels (to fit the input size of VGG-16). Sample images are shown in Figure 3-12.

The test results are summarized in Table 3-5 to 3-7. It shows that the method is quite robust in categorizing the fracture damage images for all considered values of

object distance and camera inclination angles. Interestingly, some of the images (e.g., with 15° inclination angle) are apparently distorted by the resizing operation, but the results are still encouraging with little change in the evaluation result. For the moderately damaged images, the prediction results are in good agreement with the ground truth at the object distance of 28 cm and 35 cm combined with some camera inclination angle values. However, the results are missed by one mark for the remaining cases. The results are expected since the training database didn't include images with the inclination angles. Furthermore, these kinds of uncertainties can be reduced by increasing the training data set size with the images under different situations. However, missing by one mark still suggests that the method is very promising for structural damage condition assessment and could be further improved by using more data for training.

For further investigation under real-world setting, a real photo image taken during lab cyclic load testing of the perforated steel link beam is used as a test image to identify potential issues in evaluating the proposed method in future field implementation. The original image is shown in Figure 3-13. The white paint on the link beam web was a water based latex paint. Potentially factors such as paint quality and optical characteristics of photographing could have a negative impact on the performance of the proposed method. For instance, the white paint on the link web specimen is dotted with yellowish oxidization blemish and the paint flaking pattern images is not used for the training of the CNN model but the FE generated plastic strain contour images.

Additionally, the deflected shape of the shear link shows residual deformation since the load on the link specimen was completely removed when the picture was taken.

Before using this image to evaluate the proposed method, the image is converted to grayscale and resized to 224 pixels x 224 pixels. The CNN model still successfully predicts the damage condition as Level 5. The confidence level or the probability output (40%) is lower than those FE generated images and printed photos but still the highest one among other damage levels. The result of this exploratory test suggests that the proposed method is promising for real-world implementation and issues like paint quality could affect the performance. However, further study is needed once more experiment collected images are available. The feature visualization of the test result is shown in Figure 3-13. It can be seen that the model predicts the damage level by focusing on the rectangular bar patterns between the holes. From the observations made in Figures 3-10 and 3-13**,** similar key features are consistently extracted independently of the test images.

3.5 Case Study II: Dogbone Steel Plate Specimen

In the previous section, a case study of a replaceable steel shear link beam has been presented by using FE simulation generated images and printed photos. The results from the parametric study suggest that at a minimum several thousand images are needed if fine-tuning of a pre-trained model is adopted and fine-tuning all layers yields

the best accuracy. Since the plastic strain contour images are ideally generated from FE simulations, realistic paint that can faithfully reflect such damage pattern is still needed. In this section, the white latex coating is examined, and experimentally collected images from tensile loading test of a dogbone steel plate coupon specimen are used to further verify the performance of the proposed damage identification method.

3.5.1 Experimental Test Description

A dogbone steel plate coupon specimen made of low yield point steel Q225LY was used for the experimental test under cyclic tensile loadings so that real video images can be recorded during the test. Before the experiment, white latex coating was brushed onto the surface of the dogbone segment of the steel plate coupon specimen so that the paint flaking pattern can be used to visualize the pattern change of the test specimen's plastic zone during the cyclic loading test. Pictures of the white flaking pattern in the dogbone segment were recorded using a digital camera with 720P resolution at a speed of 33 frames per second. Pictures of the dogbone steel plate coupon specimen at the beginning and the end of this experiment are shown in Figure 3-14 with dimensions marked. FE simulation results were used to calculate the damage index value at corresponding load steps in order to label the images. It should be noted that the FE simulation results used in this section are only for labeling purpose while the data (images) are collected through video stream taken during the experimental test of the

specimen. The FE simulation was performed in ANSYS Mechanical (2017) using solid element and employing large deformation formulation. The meshed FE model is plotted in Figure 3-15 (a). Mesh refinement studies were conducted to determine the level of refinement necessary to achieve reasonable accuracy for hysteresis behavior of the specimen and its fracture initiation point estimation. The calibrated hysteresis results are shown in Figure 3-15 (b). The evolution of the damage index from the FE simulation for the prescribed loading protocol is also shown in Figure 3-15 (c). The FE predicted crack initiation is 24.67 mm in terms of cumulative displacement (12 mm for the first two cyclic cycles plus an additional 14.67 mm in the last tensile excursion; 24.67=4×3+14.67), while the first visible crack in the experiment is around 26.5 mm. The FE prediction error is 7% in terms of cumulative displacement.

3.5.2 Deep Learning Implementation

A total of 8,259 images, on a similar order of magnitude to those reported earlier in the previous section, were extracted from the video stream taken during the experimental test of the dogbone steel plate coupon specimen. From these 8,259 images, 80% (6607) and 20% (1652) of the images were randomly selected for the training and test dataset respectively. This training to test data ratio (i.e., 80% to 20%) remains the same for the data set at each damage level. The images were then cropped and resized to 224 pixels by 224 pixels to match the input requirement of the adapted CNN model. The cropping

window was set as rectangular (600 pixels by 600 pixels) to keep the width to height ratio of the specimen unchanged for resize operation since the ratio would change without cropping when the original video size was 1280 pixels by 720 pixels. The training and testing process were similar to those reported in the previous section. The converged learning curve is plotted in Figure 3-17.

3.5.3 Results and Visualizations

The trained CNN model was tested using the 1,652 testing images, which achieved an overall 99.87% testing accuracy, indicating that the proposed method is doing well in the case of using real-world photo images collected from experiment tests.

The guided backpropagation method described in Section 3.2.4 was then used to interpret the results by visualizing the feature-related pattern identified by the CNN model. Four sample images representing different damage levels (from level 2 to level 5) randomly selected from the 1,652-image test database were plotted with feature-related patterns in Figure 3-16. It is seen that feature-related pattern similar to those of human visual inspection were identified: the flaking patterns of the white latex coating were highlighted in the pattern plots for damage condition assessment. Interestingly, the cable behind the notched specimen was also identified as the discriminative feature-related pattern by the CNN model. This would make sense because the cable length

shown in the video image can be correlated with the loading displacement amplitude and thus also the damage severity in the test specimen.

3.6. Concluding Remarks

This study investigates an image-driven ultra-low cycle fatigue induced structural steel damage condition assessment method using a deep learning algorithm. The technical procedure for incorporating deep learning model into image-based structural steel low cycle fatigue induced damage condition assessment method is developed. To demonstrate the usage of this method and its effectiveness in low cycle fatigue induced damage identification, a case study of structural steel fuse device was carried out using finite element (FE) simulation generated contour plot images. Parametric studies were performed in order to investigate the optimal parameter values in this method. The findings are further verified with another case study of a dogbone steel plate specimen using photo images taken during the experimental test. The proposed method is found to be effective for damage condition assessment in both case studies. The main findings are summarized as follows,

(I) The technical procedure for implementing the deep learning model for structural condition assessment application has been developed, including data labeling with quantitative damage index, training strategy and feature-related pattern

visualization. Quantitative damage index is defined to label structural condition images.

(II) Fine-tuning of all layers in a pre-trained CNN model is the recommended training strategy for structural damage condition assessment application; by doing so the training data size can be made far less than the super large data size that would be typically required for training a deep learning model from scratch.

(III) Training data size is an important factor affecting the performance of the CNN model for image driven low cycle fatigue induced damage condition assessment. The order of magnitude for the training dataset is found to be a few thousand images when fine-tuning of adapted pre-trained CNN model, to ensure the ninety percent accuracy seen in the case studies.

(IV) Saliency map offers a visual tool to evaluate whether the trained deep learning model is doing the right thing and distinguish well-trained deep learning models from those poorly-performing models by looking at the feature-related patterns aligned with the human visual inspection. The guided back-propagation method provides the salience map showing the feature-related patterns in the original images recognized by the deep learning model.

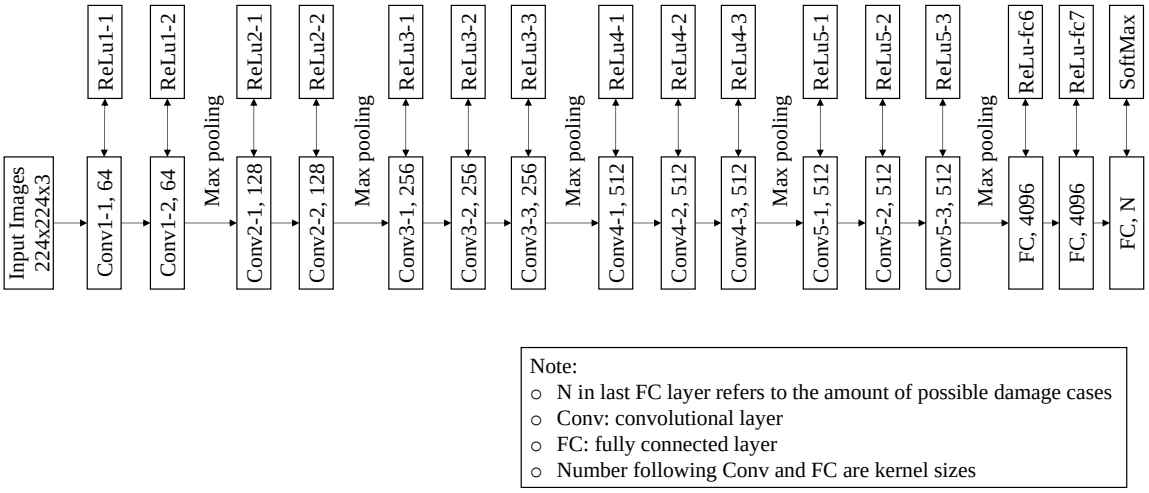


Figure 3-1. Architecture of adapted VGG-16 model based on the work by Simonyan and Zisserman (2014)

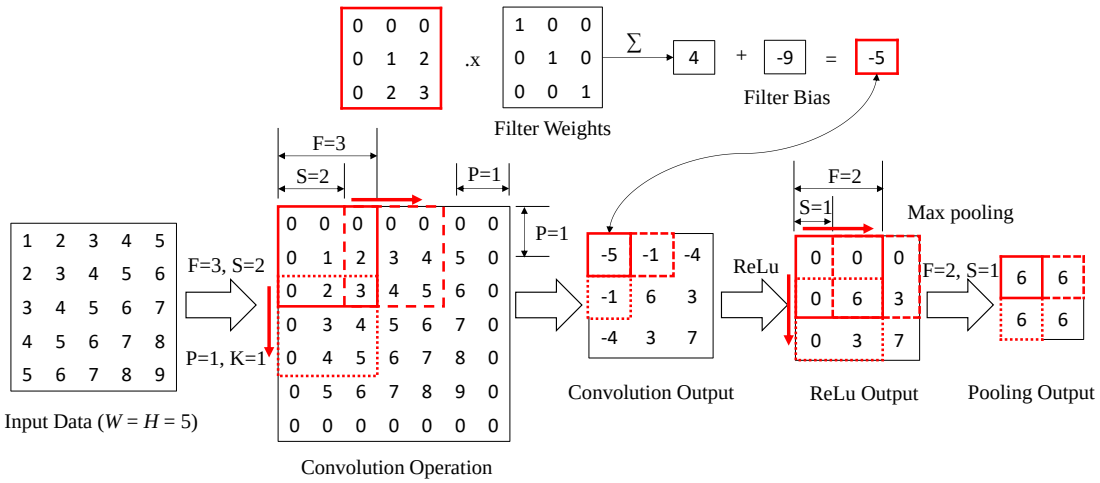


Figure 3-2. Illustration of mathematical operation of convolutional, ReLu and max pooling layer

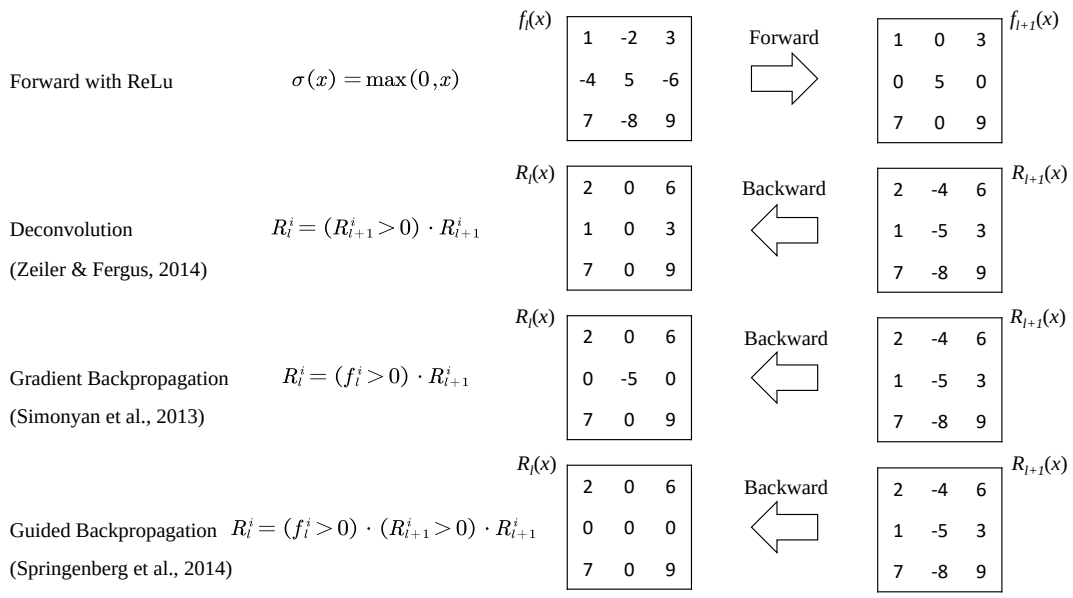


Figure 3-3. Difference between deconvolutional neural network, gradient backpropagation and guided backpropagation

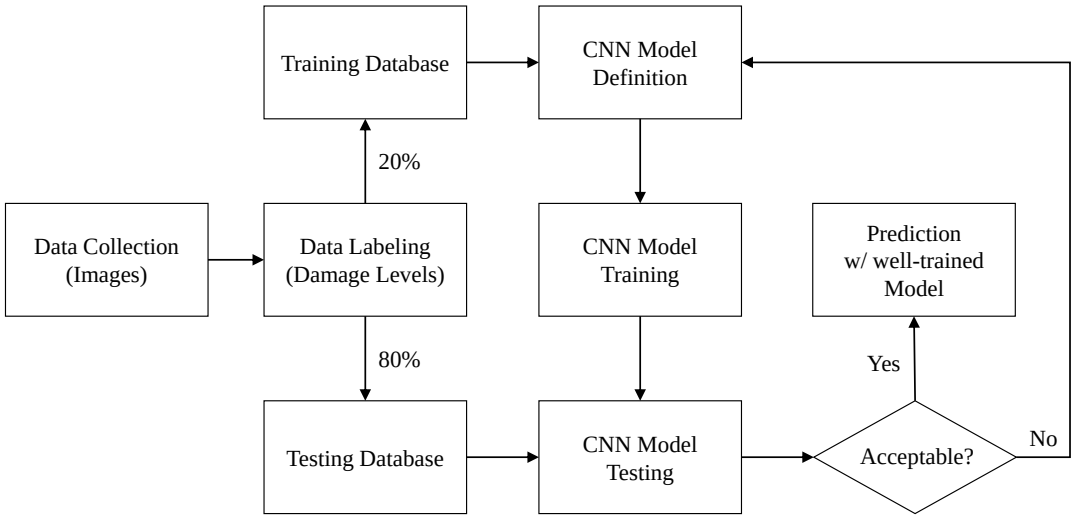
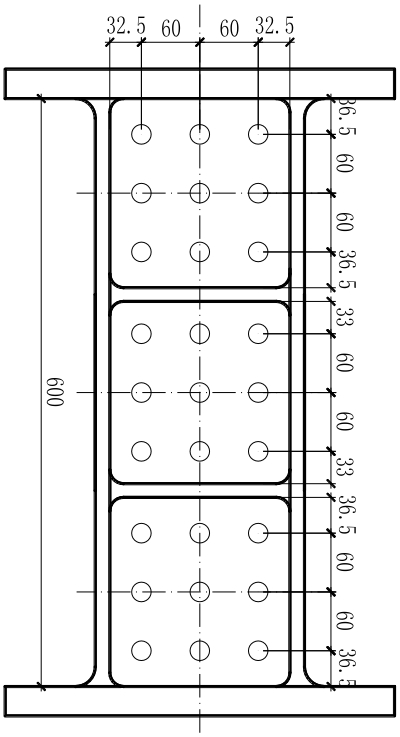
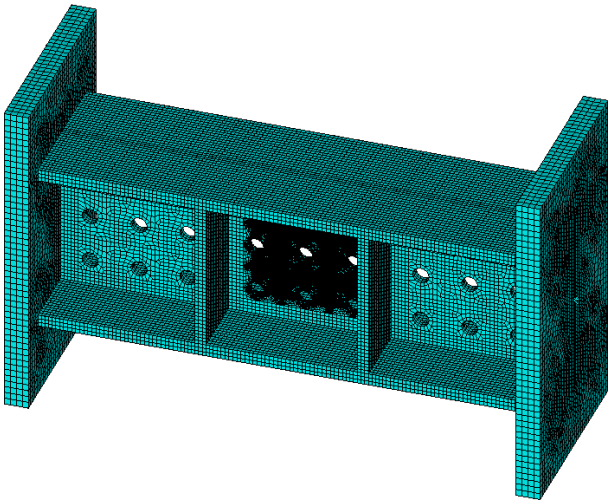


Figure 3-4. Image-driven structural damage condition assessment flowchart using deep learning algorithm

(a)

(b)





(c)

Figure 3-5. Cast steel shear link with perforated web: (a) picture of test specimen; (b) perforation pattern and dimensions; (c) finite element model

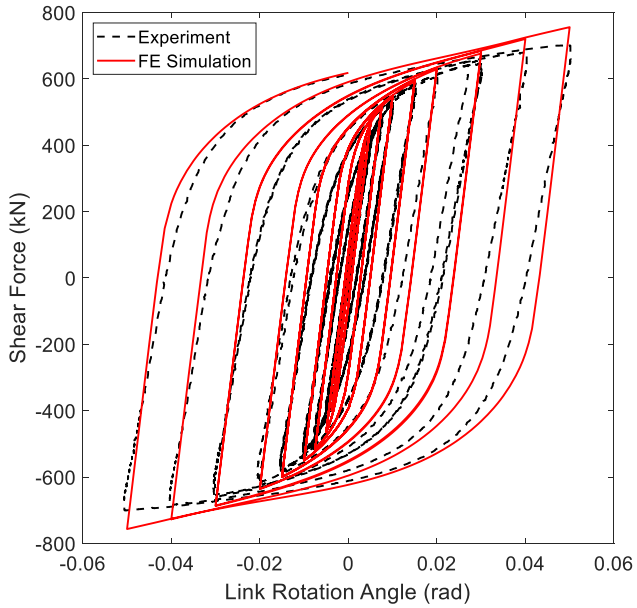


Figure 3-6. Hysteresis curves of perforated steel link specimen from FE analysis and experimental testing

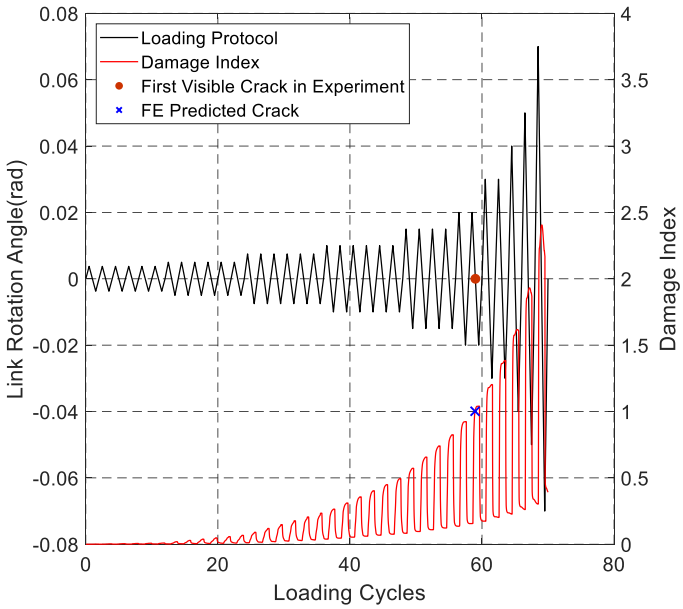
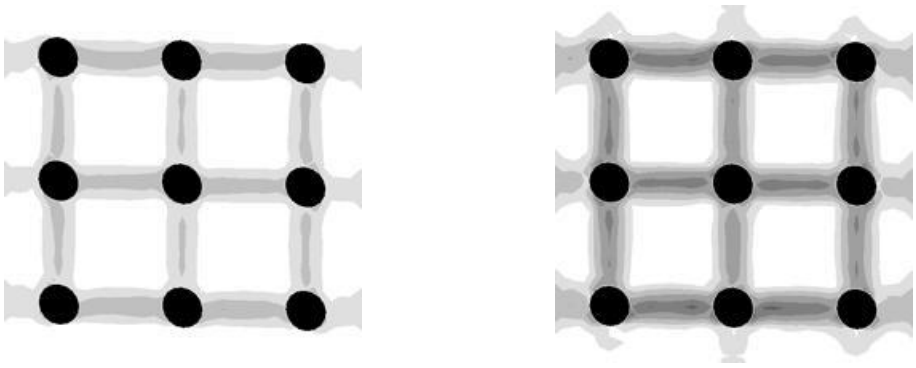
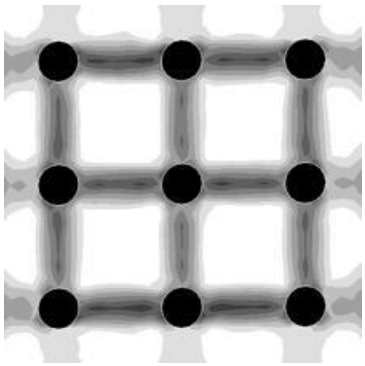


Figure 3-7. Evolution of damage index of perforated shear link specimen under AISC loading protocol

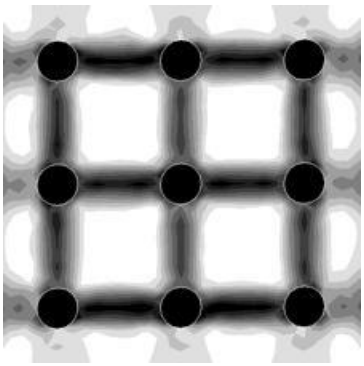


(a) Damage Index 0.11 (Damage Level 1)

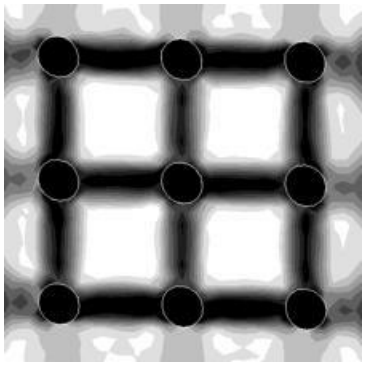
(b) Damage Index: 0.31 (Damage Level 2)



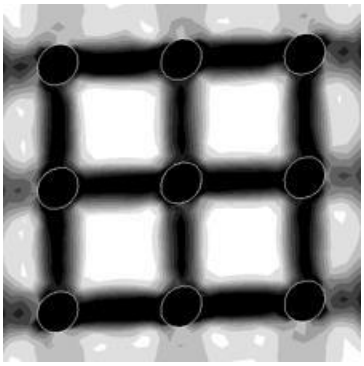
(c) Damage Index: 0.50 (Damage Level 3)



(d) Damage Index: 0.67 (Damage Level 4)



(e) Damage Index: 0.85 (Damage Level 5)



(f) Damage Index: 0.94 (Damage Level 5)

Figure 3-8. Sample FE simulation-generated images for CNN model training

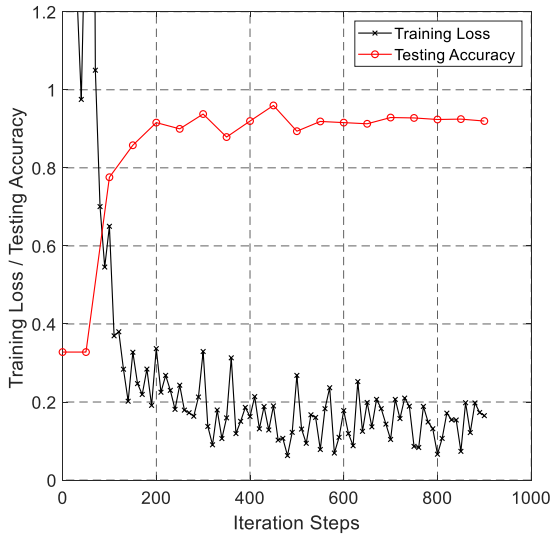
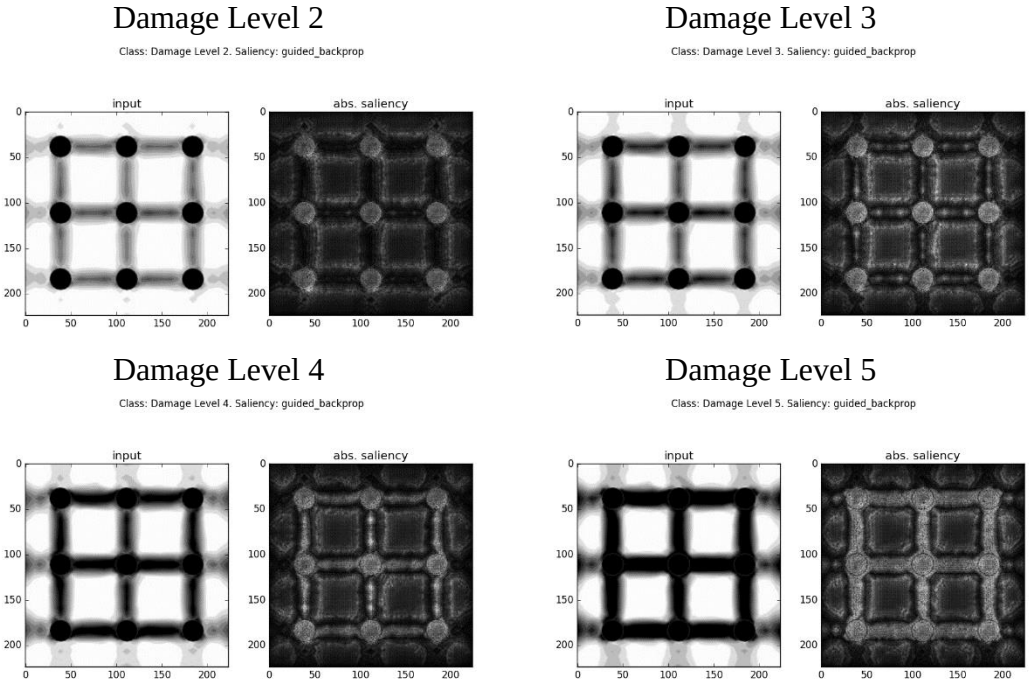
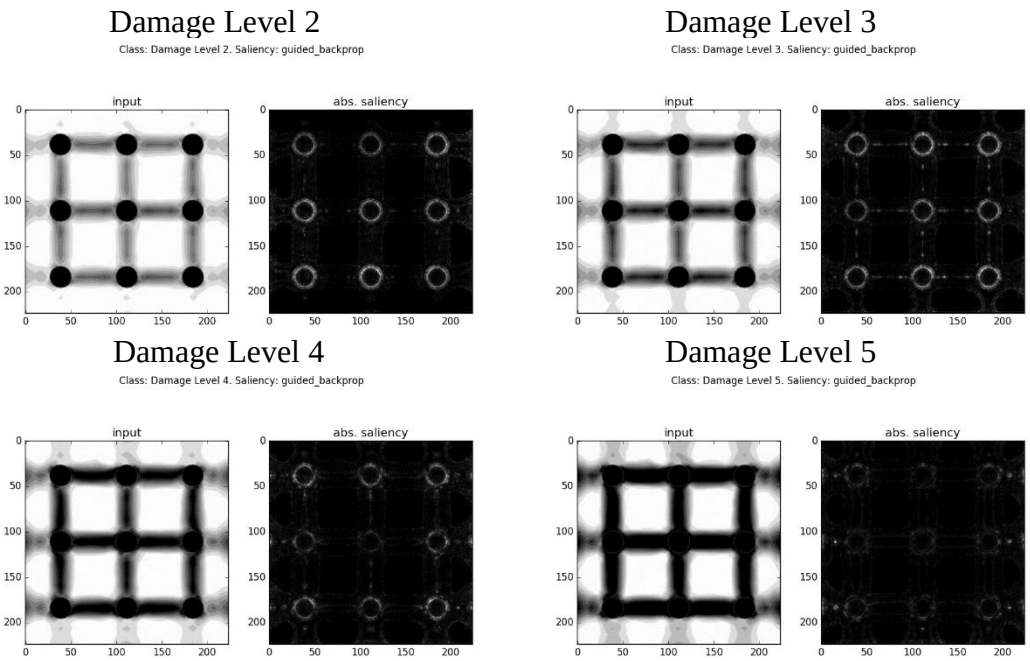


Figure 3-9. Sample learning curve from fine-tuning the CNN model with 8,600 training images for shear link



(a) well-trained model



(b) poorly-trained model

Figure 3-10. Visualization check on four sample images: (a) well-trained model with 92.3% testing accuracy; (b) poorly-trained model with 18.2% testing accuracy

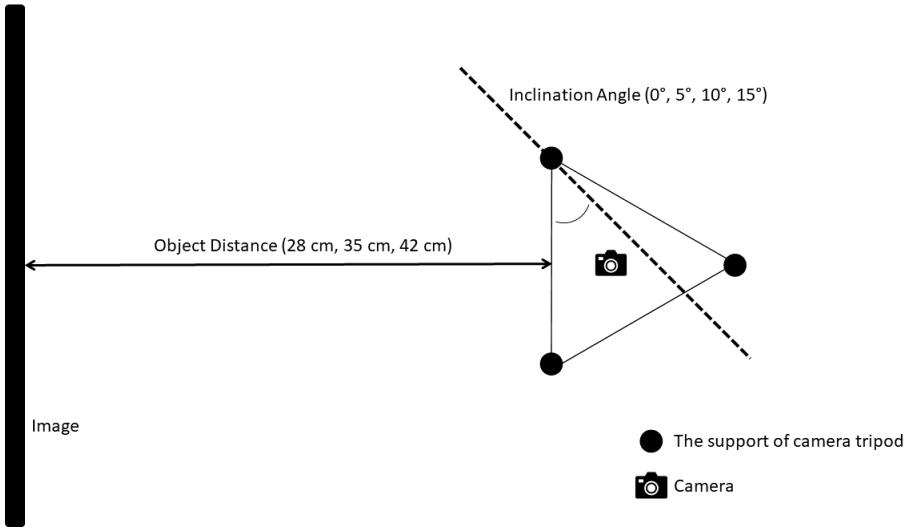
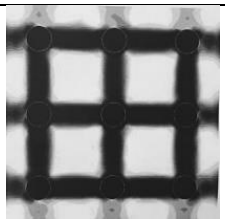
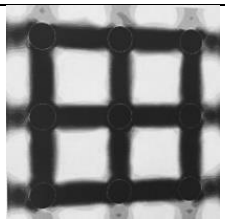
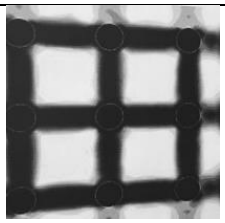
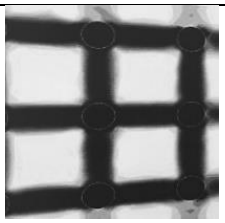
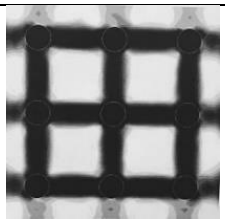
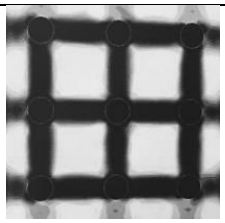
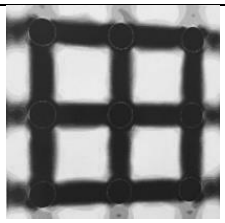
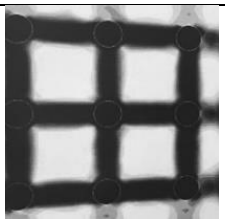
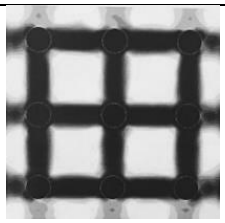
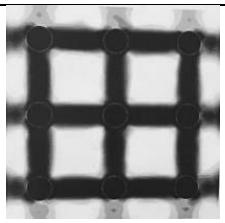
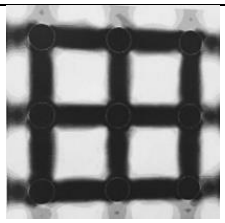
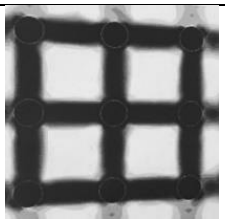
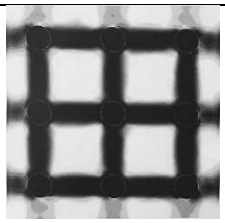
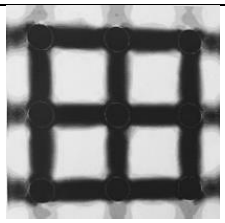
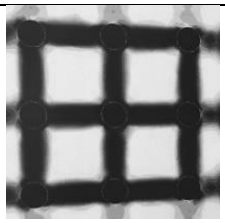
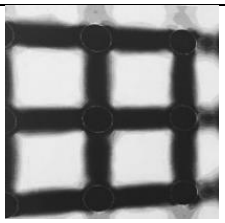
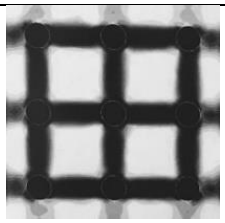
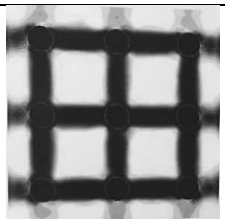
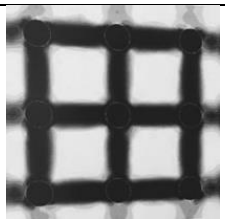
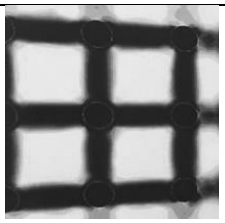
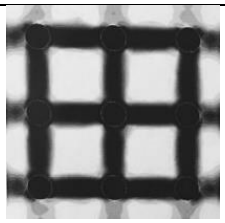
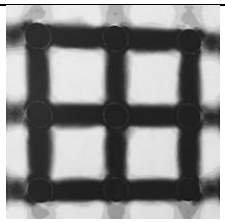
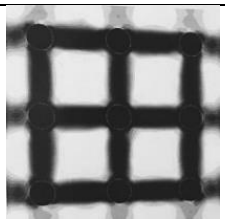
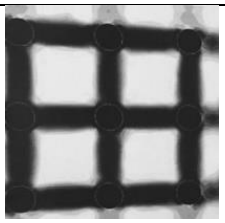


Figure 3-11. Illustration of object distance and camera inclination angle

(a) Photos of FE generated images (Fracture Damage with no Residual Drift)

Camera Inclination Angle			
0°	5°	10°	15°
Object Distance = 28 cm			
			
Object Distance = 35 cm			
			
Object Distance = 42 cm			
			

(b) Photos of FE generated images (Fracture Damage with Residual Drift)

Camera Inclination Angle			
0°	5°	10°	15°
Object Distance = 28 cm			
			
Object Distance = 35 cm			
			
Object Distance = 42 cm			
			

(c) Photos of FE generated images (Moderate Damage Image with no Residual Drift)

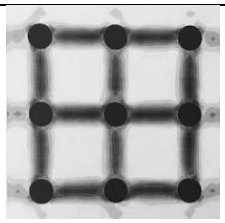
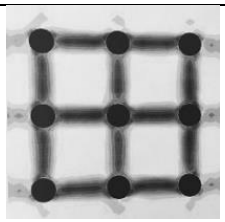
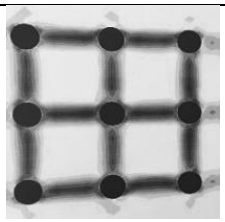
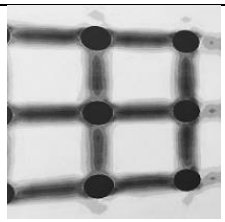
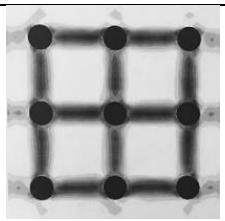
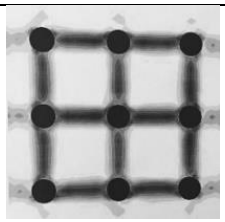
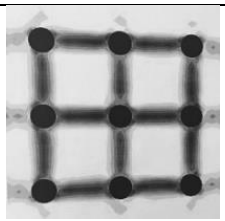
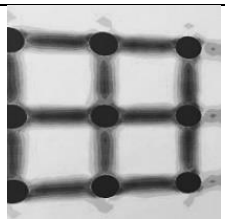
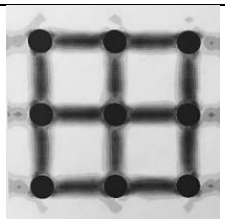
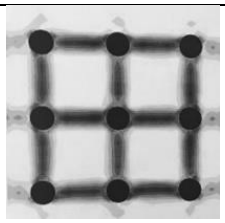
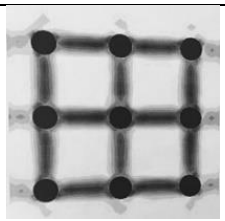
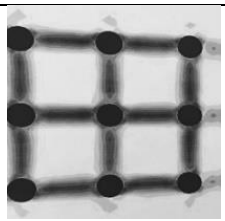
Camera Inclination Angle			
0°	5°	10°	15°
Object Distance = 28 cm			
			
Object Distance = 35 cm			
			
Object Distance = 42 cm			
			

Figure 3-12. Samples of printed FE generated image

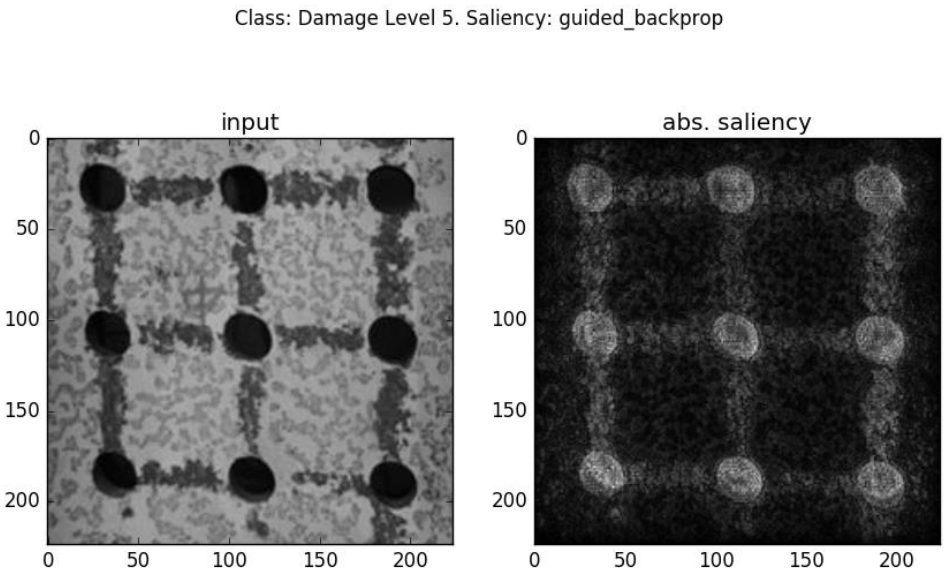


Figure 3-13. Visualization check results of experimental test image

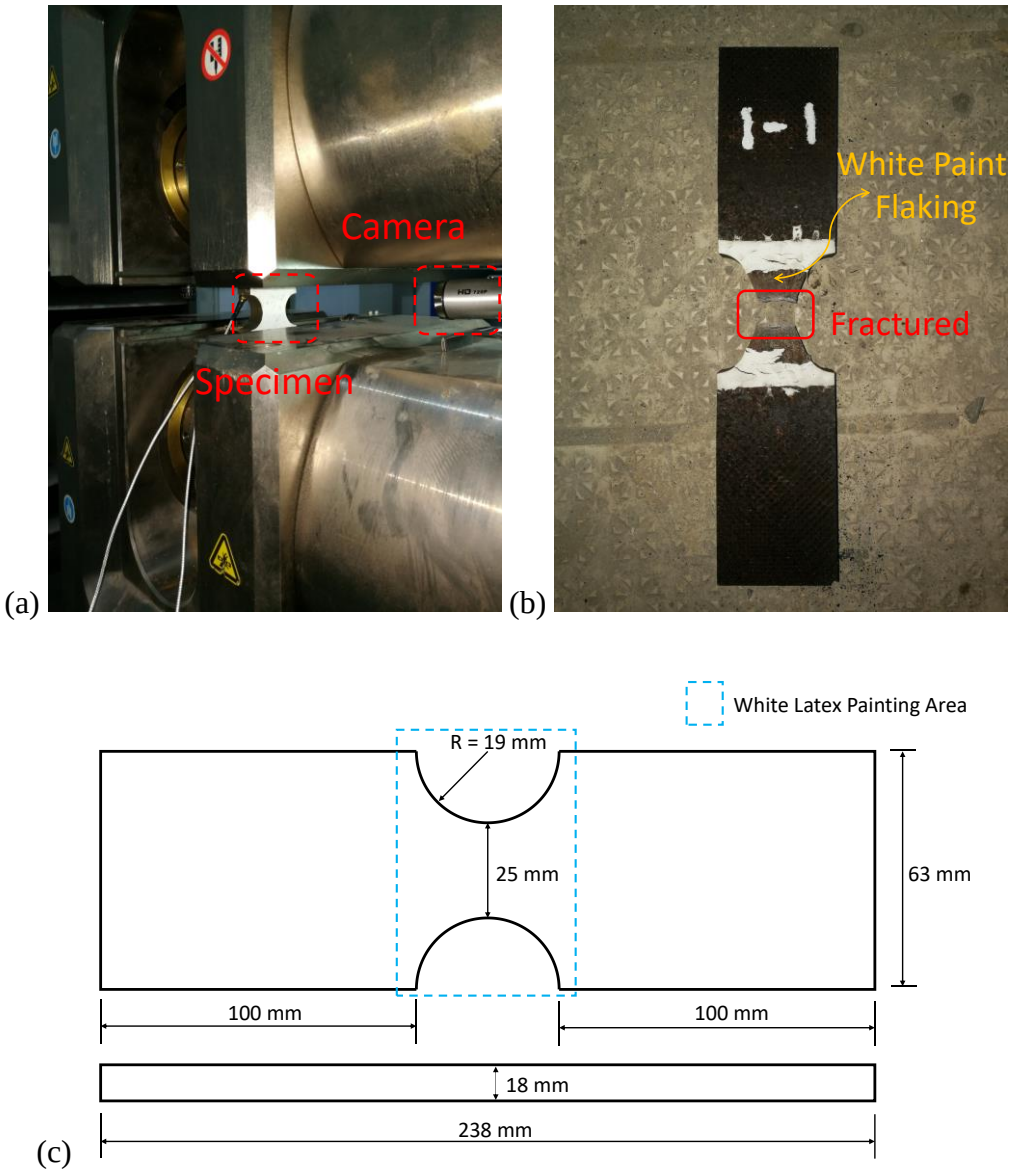
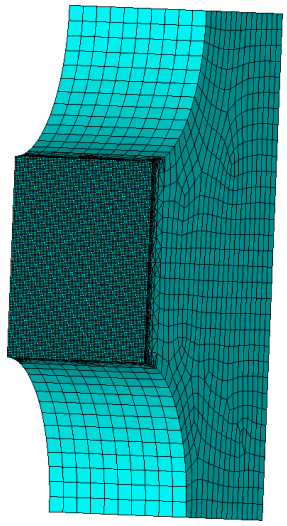
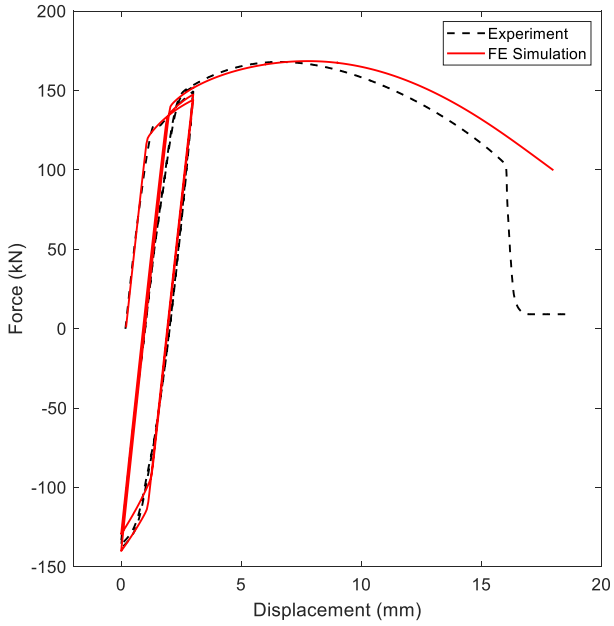


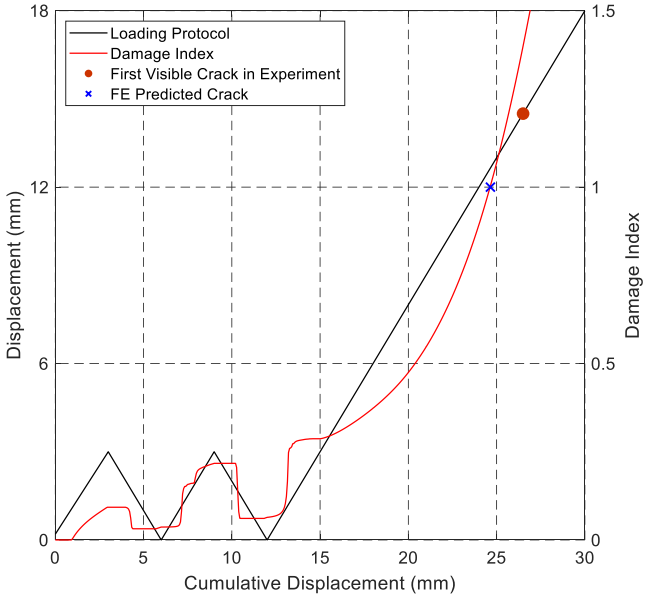
Figure 3-14. Dogbone steel plate specimen: (a) picture of specimen in test machine; (b) picture of fractured specimen; (c) specimen geometry and dimensions



(a)



(b)



(c)

Figure 3-15. FE simulation of dogbone specimen: (a) meshed FE model; (b) hysteresis curves from FE simulation and experimental data; (c) damage index evolution and predicted crack initiation

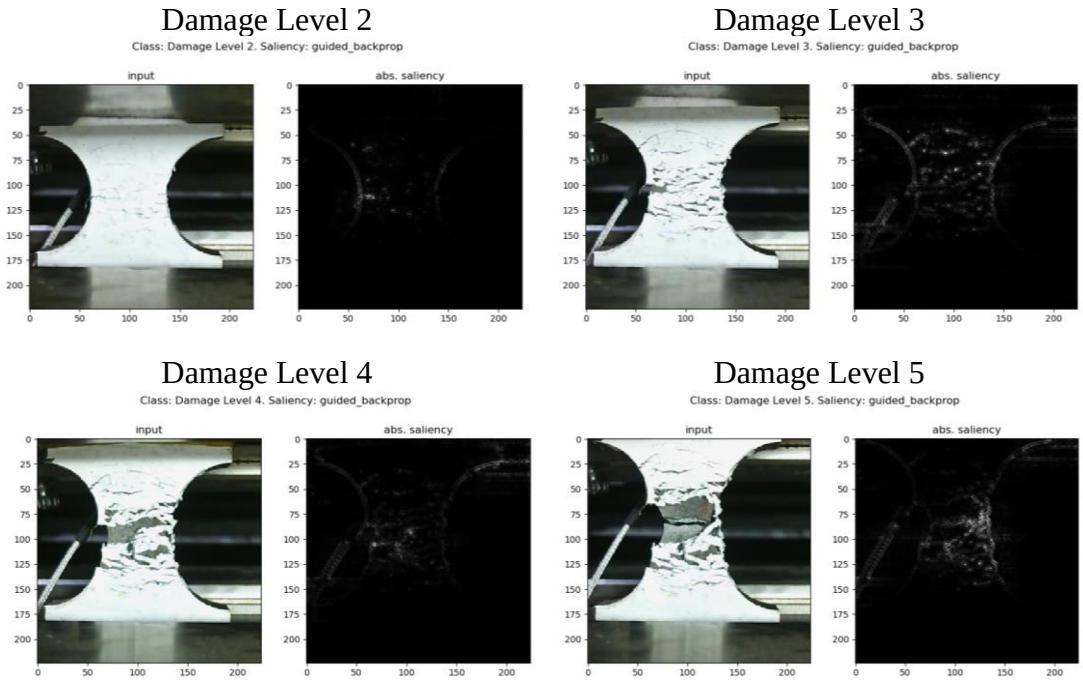


Figure 3-16. Visualization of sample images with different damage severities in the dogbone steel plate specimen (the coordinates referring to 224x224 pixels)

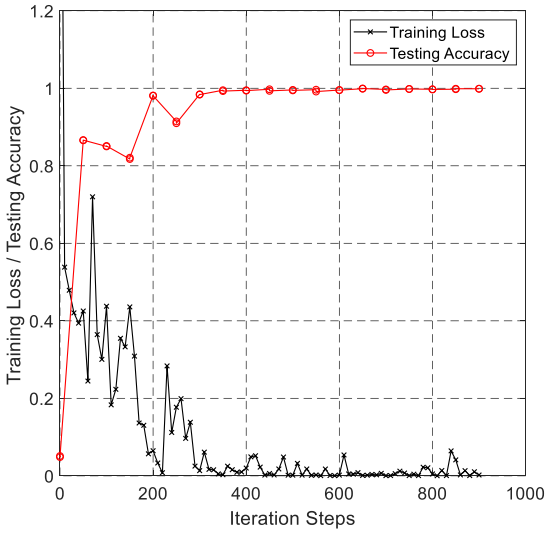


Figure 3-17. Learning curve from fine-tuning the CNN model with 8,259 images for dogbone specimen

Table 3-1. Categorization of damage levels

Damage Condition	No Damage (Level 1)	Minor Damage (Level 2)	Moderate Damage (Level 3)	Severe Damage (Level 4)	Fracture (imminent) (Level 5)
Damage Index value	0-0.2	0.2-0.4	0.4-0.6	0.6-0.8	>=0.8

Table 3-2. Material hardening parameters of G20Mn5QT cast steel

$\sigma_{0\ (MPa)}$	C1	$\gamma 1$	R_{∞}	b
483	5872.7	31.51	21	1.2

Table 3-3. Loading protocols used for generating the training dataset

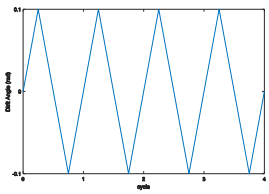
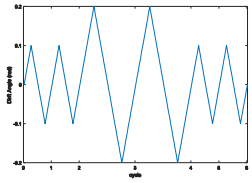
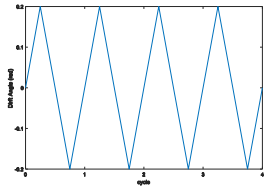
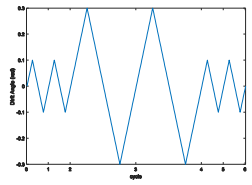
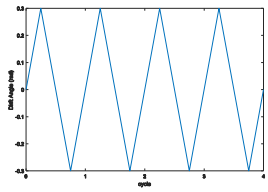
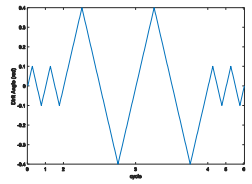
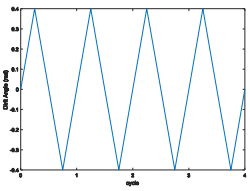
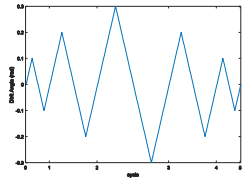
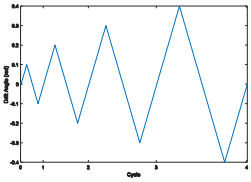
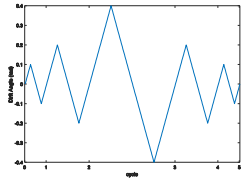
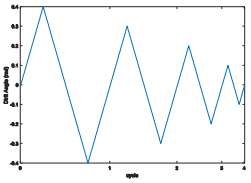
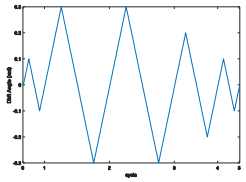
Amplitude Cycles	Waveform	Amplitude Cycles	Waveform
0.01-0.01-0.01-0.01		0.01-0.01-0.02-0.02-0.01-0.01	
0.02-0.02-0.02-0.02		0.01-0.01-0.03-0.03-0.01-0.01	
0.03-0.03-0.03-0.03		0.01-0.01-0.04-0.04-0.01-0.01	
0.04-0.04-0.04-0.04		0.01-0.02-0.03-0.02-0.01	
0.01-0.02-0.03-0.04		0.01-0.02-0.04-0.02-0.01	
0.04-0.03-0.02-0.01		0.01-0.03-0.03-0.02-0.01	

Table 3-4. Test results on FE simulation generated dataset

Training Data Size (total no. of images)	Averaged Testing Accuracy over 10 training episodes			
	Training from Scratch	Fine Tuning		
		Classifier only	Excluding First 2 layers	All Layers
3200	77.3%	79.5%	82.4%	**83.4%**
8600	80.6%	89.2%	89.4%	**91.2%**

Table 3-5. Test results on photos of printed FE simulation generated images:

Fracture with no residual drift

Damage Level (Damage Index)	Object Distance (cm)	Inclination Angle	Evaluation Results (Probability)
Damage Level 5 (Damage Index=1.04)	28 cm	0°	5 (0.867)
		5°	5 (0.712)
		10°	5 (0.815)
		15°	5 (0.703)
	35 cm	0°	5 (0.849)
		5°	5 (0.830)
		10°	5 (0.737)
		15°	5 (0.787)
	42 cm	0°	5 (0.681)
		5°	5 (0.668)
		10°	5 (0.734)
		15°	5 (0.753)

Table 3-6. Test results on photos of printed FE simulation generated images:

Fracture with residual drift

Damage Level (Damage Index)	Object Distance (cm)	Inclination Angle	Evaluation Results (Probability)
Damage Level 5 (Damage Index=0.92)	28 cm	0°	5 (0.713)
		5°	5 (0.739)
		10°	5 (0.787)
		15°	5 (0.670)
	35 cm	0°	5 (0.815)
		5°	5 (0.715)
		10°	5 (0.739)
		15°	5 (0.753)
	42 cm	0°	5 (0.853)
		5°	5 (0.749)
		10°	5 (0.710)
		15°	5 (0.821)

Table 3-7. Test results on photos of printed FE simulation generated images:

Fracture with no residual drift

Damage Level (Damage Index)	Object Distance (cm)	Inclination Angle	Evaluation Results (Probability)
Damage Level 3 (Damage Index=0.45)	28 cm	0°	3 (0.595)
		5°	4 (0.485) / 3 (0.463)
		10°	3 (0.672)
		15°	3 (0.631)
	35 cm	0°	3 (0.529)
		5°	4 (0.633) / 3 (0.268)
		10°	4 (0.615) / 3 (0.273)
		15°	4 (0.534) / 3(0.287)
	42 cm	0°	4 (0.744) / 3 (0.220)
		5°	3 (0.595)
		10°	4 (0.485) / 3 (0.463)
		15°	3 (0.672)

Chapter 4: Brace Damage Detection for Concentrically Braced Frame Structures under Seismic Loadings

4.1. Introduction

Automated and robust damage detection tool is needed to enhance the resilience of civil infrastructures. In this study, a deep learning based damage detection procedure using acceleration data is proposed as an automated post-hazard inspection tool for rapid structural condition assessment. The procedure is investigated with a focus on application in concentrically braced frame (CBF) structure, a commonly used seismic-force-resisting structural system with bracing as fuse members. A case study of 6-story CBF building was selected to validate and demonstrate the proposed method. The deep learning model – a Convolutional Neural Network (CNN), was trained and tested using a numerically generated dataset from over two thousand sets of nonlinear seismic simulation, and an accuracy of over ninety percent was observed for bracing buckling damage detection in this case study. The results of the deep learning model were also discussed and extended to define other damage feature index. This study shows that the proposed procedure is promising for rapid bracing condition inspection in CBF structures after earthquakes.

4.2 Deep Learning based Brace Damage Detection

4.2.1 Brace Damage Detection Procedure

The proposed brace damage detection method involves training a deep CNN with acceleration response data for brace damage detection tasks (see Section 4.3.4) for a series binary classification tasks (i.e., damaged or not), including (1) use roof acceleration data to detect the occurrence of brace damage in the building and (2) use the acceleration data at each floor to detect brace damages in the lower story. The training of the deep CNN model uses well-labeled images to discover features or pixel patterns in order to distinguish between images. The feature generalization is optimized through an iteration process called back-propagation. In order to improve the generalization capability of the CNN model, it is necessary to use training samples that are representative of structural response under interested ground motions (see Section 4.3.2). Since the acceleration response data is time series in nature, it is also necessary to pre-process the raw input data into standardized input format of the deep CNN model (see Section 4.3.5).

The brace damage detection process is illustrated in Figure 4-1. First, a sufficiently large data set with rectified data samples is built for the training of the deep CNN model. In this study, nonlinear time-series finite element simulation has been performed to generate training data samples representative of structural responses. A total of 2,880 sets of nonlinear time history analysis were conducted to create the

training dataset. Next, these acceleration response data from the time history analysis results are pre-processed and labeled for the training of the deep CNN model. Then, the deep CNN model is optimized through back-propagation in training. Finally, the trained CNN model is evaluated using test data samples which are generated separately from the training data samples. The validated CNN model can be used for brace damage detection in the CBF structure.

4.2.2 Deep Convolutional Neural Network

The proposed brace damage detection method relies on the deep CNN model to detect the features related to brace damage using a strategy similar to CNN-based image classification. The CNN is a specialized kind of neural network for processing data that has a known grid-like topology (Goodfellow et al., 2016). Examples of deep CNN models for image classification tasks are AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan & Zisserman, 2014), and ResNet (He et al., 2016). These CNN models have sophisticated network architectures in stacking of computational layers, including convolutional layer, pooling layer, ReLu layer, and fully connected layer. Details of these computational layers can be found in the reference by Goodfellow et al. (2016). It is observed that the generalized features in early layers appear to be general regardless of exact cost function and natural image dataset, and in later layers seem to be task-specific (Yosinski et al., 2014). In image classification, the general feature can

be an edge detector while the task-specific feature can be the face of a human or the wheel of a car. The visual intuition of this finding can be found in the reference by Zeiler and Fergus (2014). This observation on the feature generalization enables the use of transfer learning to facilitate the training of CNN models. Transfer learning refers to the situation where what has been learned in one setting (base dataset) is exploited to improve the generalization in another setting (target dataset) (Goodfellow et al., 2016). When the target dataset is significantly smaller than the base dataset, transfer learning enables training a large target network without overfitting; recent studies that have taken advantage of this fact obtained state-of-the-art results from transfer learning (Yosinski et al., 2014; Zeiler & Fergus, 2014). Transfer learning is also appealing to the application of brace damage detection in this study for two reasons: (1) the sophisticated design of the architecture of those successful CNN models can be taken advantage of, and (2) the size of acquirable database for brace damage detection (order of thousands) is far smaller than the image data (order of millions) in ImageNet database (Deng et al., 2009), from which those sophisticated deep CNN models were trained. Transfer learning has shown promising results in medical applications also with a limited number of available data samples, such as melanoma screening (Menegola et al., 2017) and skin lesion classification (Lopez et al., 2017). Therefore, transfer learning is adapted in this study, and the pre-trained deep CNN model (VGG-16) by Simonyan and Zisserman (2014) has been used in this study

as the base model due to its advanced network structure and manageable computational cost.

4.2.3 Visual Interpretation

With demonstrated impressive performance of deep CNN models for large-scale image recognition, the problem of understanding the aspects of visual appearance captured inside a deep learning model has become related, so the improvement would not be restricted to trial-and-error process (Simonyan et al., 2013; Zeiler & Fergus, 2014). To address the problem, researchers have investigated visualization techniques that gives insight into the function of intermediate feature layers and the feature evolution during training (Zeiler & Fergus, 2014; Simonyan et al., 2013; Springenberg et al., 2014). The deconvolutional neural network used by Zeiler and Fergus (2014) provides a non-parametric view of invariance, showing which patterns from the training set activate the feature map. Simonyan et al. (2013) further elucidated the function of the deconvolutional neural network for feature visualization by computing image-specific class saliency map, which approximates the highly non-linear scoring function mapped from CNN model by evaluating its first-order Taylor expansions as Eqn. (4-1):

$$\begin{aligned} S_C(I) &\approx S_C(I_0) + \frac{\partial S_C}{\partial I}\Big|_{I_0} \cdot (I - I_0) \\ &= \frac{\partial S_C}{\partial I}\Big|_{I_0} \cdot I + \left(S_C(I_0) - \frac{\partial S_C}{\partial I}\Big|_{I_0} \cdot I_0 \right) \end{aligned} \tag{4-1}$$

where S_c represents the scoring function; I denotes input variable (images) and I_0 is a given specific image. The saliency map is established based on the derivative respect to the input image of the score function that its magnitudes indicate which pixel needs to be adjusted the least to affect the score. They pointed out that the deconvolution layer is either equivalent or similar to computing the gradient of the visualized layer. It is effectively corresponding to the gradient back-propagation through a convolutional layer. The reason for the "either equivalent or similar to computing the gradient" is the ReLU layer. Afterward, guided-backpropagation (Springenberg et al., 2014) was proposed to achieve enhanced accuracy in calculating the gradient back-propagation through a convolutional layer. The difference between deconvolutional neural network, gradient backpropagation and guided backpropagation were compared in the reference (Springenberg et al., 2014). The guided back-propagation method is used in this study to visualize the feature patterns identified in the CNN model, so the classification results for damage pattern identification can be more easily assessed.

4.3 Case Study: 6-story Steel Braced Frame Building

The validation procedure is illustrated in Figure 4-2. A 6-story CBF building is selected as the prototype structure and nonlinear time history analysis of the prototype CBF structure subjected to earthquake ground motions were performed to generate the data for training the deep learning model. Then, the performance of the CNN model is

evaluated based on two evaluation tasks. Details and results are presented in this section.

4.3.1 Prototype Building and Numerical Modeling

To demonstrate and validate the proposed bracing damage detection method, a six-story CBF (chevron type, 6V) structure was selected as the prototype building. This building was designed for a site in the metropolitan Los Angeles area with site class D (firm soil) by Sabelli et al. (2003). To generate the data records, a finite element (FE) model for the prototype structure that can simulate the nonlinear failure behavior of bracings and other structural members needs to be established first; without loss of generality, only a single braced bay was modeled in the FE software and later used for performing the nonlinear time history analysis under an ensemble of ground motion records, as shown in Figure 4-3. Similar to the model by Sabelli et al. (2003), all beam-to-column connections with gusset plates attached (i.e., all connections except for the roof level joints) were modeled as rigid joints. Braces are modeled with the pin-pin connection at their ends. The rigid floor diaphragm was assumed for all floors; thus, all nodes at the same floor level would experience the same lateral drift in the horizontal direction. The columns were fixed at the base on a rigid foundation. The global P-Δ effects were considered via a lean-on column. The seismic mass was modeled as a lumped mass with one-sixth of the total floor mass assigned to the node of the lean-on

column because a total of six braced bays contribute to lateral force resistance in each direction.

The finite element model (FEM) of the building was built using a general FE modeling software - OpenSees (McKenna et al., 2000). The displacement based nonlinear Beam-Column element with discretized fiber section was used to model all the beams, columns and braces. The rigid truss element was used to model the lean-on columns. The uniaxial Giuffre-Menegotto-Pinto nonlinear material model for steel and large displacement effect by using co-rotational transformation were considered in the model. An initial imperfection with one-thousands of the initial brace length was assigned to the braces to simulate the brace buckling behavior. The yield stress of the steel material was assumed as 55 ksi (379.21 MPa) for all beams and columns as used by Uriz (2008), and 60 ksi (413.69 Mpa) for braces based on the experimental data from Yang and Mahin (2005). Panel-zone effects were not considered. The strain hardening ratio (ratio between the post-yield tangent stiffness and the initial elastic stiffness) is assumed to be 0.05. It was found that 10 elements for columns and 20 elements for braces and beams were sufficient to capture the nonlinear response behavior of the CBF structure subjected to seismic ground motion while the computation efforts are not unreasonably high for generating the massive training data through nonlinear time history analysis.

4.3.2 Ground Motions and Seismic Analysis

A total of 192 historical earthquake ground motion records were selected from the PEER NGA database (https://ngawest2.berkeley.edu/) according to the following criteria: (i) The selected earthquake records are free of any forward directivity effects; (ii) All earthquake ground motion accelerograms were recorded on soil type D; (iii) were generated by earthquakes of moment magnitude M_w ranging from 5.7 to 7.3; and (iv) The hypocentral distance for these records ranges between 3.4 and 59.7 km. Subsequently, all the earthquake records were scaled to a target spectrum that represents the design basis earthquake (DBE) in Van Nuys, California conforming to the ASCE 7-10 seismic code. The scaling factor for each of the selected earthquake records was calculated by matching the averaged 5% damped response spectrum to the target spectrum at the pre-specified period range. The interested periods were set as 0.55 sec, 0.6 sec, 0.65 sec, 0.7 sec and 0.75 sec based on the fundamental period of the prototype structure (T_{n1} = 0.65 sec from eigenvalue analysis). After each ground motion was scaled to DBE, additional records were created by scaling the corresponding DBE earthquake using scaling factors ranging from 0.1 to 1.5 so that different damage level scenarios could be simulated for the prototype structure under varying seismic intensity levels. In total, 2,880 ground motion records were used for the nonlinear seismic analysis data generation.

In the nonlinear time history analysis, the Newmark constant average acceleration method (β=0.25 and γ=0.50) was employed. The Krylov-Newton method

(Scott & Fenves, 2009) was used as the solution algorithm, which uses a low-rank least-squares analysis to advance the search for equilibrium at the degrees of freedom (DOFs) where the largest changes in structural state occur; then it corrects for smaller changes at the remaining DOFs using a modified Newton computation. The algorithm is suited to simulating brace buckling in CBF structure where yielding and local collapse mechanisms form at a small number of DOFs while the state of the remaining structural components is relatively linear. The analysis results of the prototype structure revealed that the method had satisfactory convergence performance.

4.3.3 Potential Damage Scenario

In CBF structures, seismic loads are mainly resisted by pairs of braces, in which the potential damage would initiate as compression braces buckle and tension braces yield. It is noted from nonlinear analysis results that brace buckling always happens first because the inter-story drift limit for brace buckling is smaller than the corresponding yielding limit for a specific brace. Other types of damage such as plastic hinge formation would then occur at beam-to-column joints at large inter-story drifts. Among the different damage types considered, brace buckling occurs first and has the most severe effect to structural properties, thus the status of brace buckling is used to label the input data for the CNN model so that the deep CNN model has been trained to

identify the occurrence of brace buckling. The brace tension yielding and plastic hinge formation near beam-to-column connections were also included the FE model.

4.3.4 Evaluation Tasks

In buildings, structural members are often hidden behind fireproof coating and drywall, and thus buckling or yielding of these hidden steel members are difficult to detect, often requiring removal of coverings and thus time-consuming and costly. For example, in the 1994 Northridge earthquake, a 4-story office building with fractured HSS bracing remained plumb following the earthquake and the initial assessment of the structure by the owner's representative prior to the review engineer arriving on site was that the structure had not sustained much damage (Tremblay et al., 1995). Only after the drywall with some cosmetic damage was removed, the fractured bracing could be inspected. Therefore, visual inspection of the damage condition of each brace in the building structure might be neither economical nor time efficient due to difficult access to braces after strong earthquakes. Therefore, as shown in Figure 4-4, two simplified damage detection schemes are proposed and examined in place of all possible permutations: (Task 1) use the roof acceleration record only to detect the occurrence of brace damage (at least one brace damaged in the building); This can be viewed as a yes/no question to be addressed during the initial quick screening stage of structural condition inspection; (Task 2) uses the acceleration data at each floor to detect brace

damages in the lower story (at least one of the two braces are identified as damaged). Once the first scheme raised a red flag for the post-disaster inspection that signals the retrofit or close-up inspection priority, the second scheme can be carried out to reveal the damage location which would help engineers rapidly locate the damage and justify the need for close-up inspection in the building. These two damage detection schemes are developed to address different needs and resource availability: more accurate damage information needs more sensors which increase the installation and maintenance costs. These two damage detection schemes are investigated to evaluate the damage detection procedure.

4.3.5 Deep Learning Implementation

4.3.5.1 Input Data: format and preprocessing

The floor acceleration response data from nonlinear seismic analysis of the prototype CBF structure subjected to earthquake ground motion base excitation was selected as the raw input data in this study. However, the image data was stipulated as the input to the particular deep learning model (i.e., VGG-16) adapted for this study while the acceleration data is time series in nature. Thus, the short-time Fourier transformation (STFT, see Figure 4-5) was applied first to transform the acceleration time series data to image format compatible with the adapted CNN model. The STFT spectrum plot can be defined with two coordinates: time (x-axis) and frequency (y-axis), while its

magnitude is represented by different color shades. Because different time-intervals were used for the original 192 ground motion records, all acceleration response data were interpolated with a uniform time interval value of 0.01 seconds to standardize the output STFT images. Additionally, all acceleration time series data were tailored to a constant duration of 70 seconds by padding zeros if the original duration is shorter.

4.3.5.2 Dataset creation

As mentioned above, a total of 192 ground motion records were scaled to the design basis earthquake (DBE), and they were further scaled using scaling factors ranging from 0.1 to 1.5 to generate different damage scenarios for the prototype CBF structure. Nonlinear time history analysis of the CBF structure was then performed using each scaled ground motion record (total number of 2,880). For 40 records out of the 2,880 sets of scaled ground motion records, the displacement response of the structure became overly large, and the numerical analysis did not converge in the later stage of the seismic duration. Therefore, these 40 cases were dropped, and the remaining 2,840 sets of nonlinear time history analysis results were selected for the damage detection study. The acquired dataset was randomly shuffled before the splitting for the training and testing subsets, and the ratio of training to testing data was set at 80% to 20% for the respective class. Details about the size of training and testing samples can be found in Table 4-1.

4.3.5.3 Model Training and Validation Test

The deep learning model's training and testing were implemented in the Caffe Framework (Jia et al. 2014) using a workstation with a single Graphic Processing Units (GPU) card (model: Nvidia GeForce Titan X) under the Ubuntu 16.04 LTS system. The learning rate was set to be 0.001 as initial value and decayed with a ratio of 0.5 per every 300 steps (e.g., after 300 iterations, the learning rate was adjusted to 0.0005, and after 600 iterations, the rate decayed to 0.00025, etc.). The training batch size was set as 64 due to graphic memory limit. The deep learning model was fine-tuned from the adapted VGG-16 using the training data generated above. The last fully connected layer in the original VGG-16 was replaced by a binary output for brace damage detection. The training strategy of fine-tuning all layers is used here. It can be seen from the learning curve in Figure 4-6 that the model converged very well, indicated by consistent high training and testing accuracy.

4.3.5.4 Validation Test Results

Seven CNN models were trained separately for the above-mentioned evaluation tasks. The testing results were summarized in Table 4-1. It can be found that the proposed method can successfully identify damage or healthy cases with consistently high accuracy in Tasks I and II, except for the damage case in the 4^{th} story of Task II. This was due to the facts that the amount of data samples was too few in training for this case, and this particular dataset cannot be augmented by adding more sets of ground

motions because of the specific design of this prototype building that made the brace in this story less likely to buckle. It should be noted that the 192 DBE level ground motions with a scaling factor of 1.5 represented the maximum considered earthquake (MCE) level events. Thus, further scaling the ground motions would not benefit the filed practice since the ground motions greater than this level are out of the design scope. However, this issue can be circumvented by implementing the brace damage detection for two adjacent floors. For instance, the 4^{th} and the 5^{th} floor can be combined for the damage detection; the acceleration signals at the 5^{th} floor were used to detect the occurrence of brace buckling within the two stories. A separate CNN model was trained based on this damage detection scheme, and the results turned out to be effective with a testing accuracy of 95.07% (135/142) for the healthy case and 98.36% (419/426) for the damaged case.

4.4 Discussions

With the encouraging accuracy observed in the previous section, the potential of using the proposed method for brace damage detection appears to be promising. However, the reason for such success remains unclear. To address this question, the discussion in this section is attempting to interpret the results by visualization in order to understand why the CNN model can do well and how to optimize. The attempt was first trying to identify a recurring pattern through visual inspection of the results from the trained

CNN model. This pattern was then used to formulate a handcrafted feature for classification using Bayesian decision theory, so the comparison can be made between the handcrafted feature and the feature identified by the trained CNN model in order to explain the reason for the high accuracy of brace damage detection. A modified version of the handcrafted feature and a feedforward neural network is also included for comparison.

The results from the trained deep learning model were visually assessed by using the guided backpropagation method described in Section 4.2.3; the saliency map recognized by the trained CNN model for brace damage detection is visualized here. The saliency map was plotted using the same method as described in the reference by Simonyan et al. (2013). Given an input STFT image, the saliency map represents the most relevant pixel information regarding brace damage detection.

Through visual inspection of the saliency maps, a recurring feature pattern was observed as the "volume" of the STFTs if the grayscale intensity of the spectrum is viewed as the height of the bottom time-frequency surface for a 3-D representation. For demonstration, three STFT plots were randomly picked from testing data, and the saliency maps for the selected STFTs were shown in Figure 4-7. In order to quantify this feature, the volume (V) of a given STFT plot can be calculated using the summation expression in Eqn. (4-2) below,

$$V = \sum I = \sum_{m=1}^{224} \sum_{n=1}^{224} I_{mn} \tag{4-2}$$

where I represents the STFT image matrix, and I_{mn} denotes the gray scale intensity of the pixel with coordinates of m and n.

Applying Eqn. (4-2), the STFTs can be transformed into the feature space, so the classification problem can be restated as: given a feature value (i.e., the volume), how to make the prediction? According to Bayesian based decision theory, the prediction can be made based on the rule of *Maximum a Posterior* (MAP), which assigns the prediction to the class with the highest posterior probability. The posterior probability can be calculated based on the Bayes theorem as Eqn. (4-3),

$$P(C_i|v) = \frac{P(v,C_i)}{P(v)} = \frac{P(v|C_i)P(C_i)}{P(v)},\ i=1,2 \tag{4-3}$$

where $P(v,C_i)(i=1,2)$ denotes the joint probability; $P(v|C_i)(i=1,2)$ represents the class conditional probability (or likelihood); $P(C_i)$ describes the prior probability; v signifies the feature (volume) value; $C_i(i=1,2)$ is the binary class. The ratio of posterior probability can then be formulated as Eqn. (4-4),

$$\frac{P(C_1|v)}{P(C_2|v)} = \frac{P(v,C_1)}{P(v,C_2)} = \frac{P(v|C_1)P(C_1)}{P(v|C_2)P(C_2)} \tag{4-4}$$

If a uniform prior is assumed, which means the prior probability of the two classes is equal (i.e. $P(C_1)=P(C_2)=0.5$) when no prior information is given. Eqn. (4-4) can be restated as Eqn. (4-5),

$$\frac{P(C_1|v)}{P(C_2|v)} = \frac{P(v|C_1)}{P(v|C_2)} \tag{4-5}$$

Since the feature variable is assumed to be continuous, $P(v=v_x|C) = 0$ for a given value of v (e.g., $v=v_x$). The L'Hospital's rule is then used to calculate Eqn. (4-5) that,

$$\frac{P(C_1|v_x)}{P(C_2|v_x)} = \frac{P(v_x|C_1)}{P(v_x|C_2)} = \lim_{v \to v_x} \frac{P(v|C_1)}{P(v|C_2)} = \lim_{v \to v_x} \frac{p(v|C_1)}{p(v|C_2)} = \frac{p(v_x|C_1)}{p(v_x|C_2)} \quad (4\text{-}6)$$

where $p(v|C_i)(i=1,2)$ represents the class-conditional probability density (or likelihood density), which can be established based on the observed data (feature distribution). Henceforth, the classification can be made based on Eqn. (4-7),

$$\text{Prediction} = \begin{cases} C_1, \text{ if } p(v_x|C_1) > p(v_x|C_2) \\ C_2, \text{ if } p(v_x|C_1) < p(v_x|C_2) \end{cases} \quad (4\text{-}7)$$

A decision boundary can also be found so that $p(v|C_1) = p(v|C_2)$, which is the cross intersect of the two probability density functions. And the prediction can also be made based on Eqn. (4-8),

$$\text{Prediction} = \begin{cases} C_1, \text{ if } v_x \overset{<}{>} v_0, \text{ so } p(v_x|C_1) > p(v_x|C_2) \\ C_2, \text{ if } v_x \overset{>}{<} v_0, \text{ so } p(v_x|C_1) < p(v_x|C_2) \end{cases} \quad (4\text{-}8)$$

Interestingly, it is noticed that the rule of *Maximum a Posterior* (MAP) is equivalent to the rule of minimizing the misclassification rate, with the assumption of a uniform prior. The proof about this statement can be found in the reference by Bishop (2006). The misclassification occurs when an input feature belonging to class C_1 is assigned to class C_2 or vice versa. The probability of making the mistake is given by Eqn. (4-9),

$$P(\text{misclassification}) = P(v \in R_1, C_2) + P(v \in R_2, C_1)$$
$$= P(C_2)\int_{R_1} p(v|C_2)dv + P(C_1)\int_{R_2} p(v|C_1)dv \quad (4\text{-}9)$$

As illustrated in Figure 4-8, the optimal decision boundary v_0 is the cross intersect of the two class-conditional probability density functions; the probability of misclassification will always be larger with assigning a boundary on the left (v_1) or right (v_2) side of the v_0, since an additional area region (yellow or blue) will occur while the sum of the area of red and green regions remains constant. With the optimal decision boundary v_0, it can also be found that the larger the intersection area of the two curves (i.e., the summation of the two integrals in Eqn. (4-9); the sum of the area of the red and green regions), the higher the misclassification probability will be. Since the feature distribution predominately influences the intersection area, a better feature will always have a smaller intersection area, so as the probability of making mistakes; a comparison can be made between features.

The observed volume feature formulation was applied to the 2,840 sets of acceleration data in Task 1. The histogram of the feature distribution and the estimated class-conditional probability density function were shown in Figures 4-9(a), (b) respectively. The establishment of the class-conditional probability density function was based on the kernel smoothing function estimation to fit the observed feature distribution. The decision boundary was found to be 26.37, as shown in Figure 9(b). The predicting accuracy per class was then found to be 86.13% (healthy) and 87.30%

(damaged), which were close to the estimated rate 85.39% (healthy) and 86.95% (damaged). The estimated accuracy was calculated based on the cumulative probability using the estimated class-conditional density function. It should be noted that the accuracy here was referred to the whole dataset rather than the testing dataset used in Section 4.3.

As discussed before, feature comparison results show that an ideal feature would be the one formulated to reduce the intersection area in the class-conditional probability density function curve plots. An example of this can be modifying the volume feature with a modifier (s), which was set as the scaling factor of the input earthquake ground motion with respect to the design earthquake level (i.e., a value between 0.1 to 1.5), as expressed in Eqn. (4-10) below,

$$\hat{V} = s \cdot V = s \cdot \sum I = s \cdot \sum_{m=1}^{224} \sum_{n=1}^{224} I_{mn} \tag{4-10}$$

This modification is inspired by the fact that stronger earthquakes would have a higher likelihood to cause brace damaged in the CBF structure. The data distribution of the modified feature is shown in Figure 4-10, with fewer intersection area comparing to the original volume feature in Figure 4-9(b). When the decision boundary was set as 12.72, the predicting accuracy per case was found to be 95.48% (healthy) and 92.43% (damaged), which were close to the estimated rate 95.58% (healthy) and 91.54% (damaged).

The modification of the "volume" feature suggests that an optimization can be made on the feature function in Eqn. (4-2) to further boost the classification performance by mining features with the least intersection area. The optimization can be achieved by training a feedforward neural network (FNN) with no hidden neuron but two outputs. The output neuron (before the SoftMax layer) calculated feature *F* from the input STFT using Eqn. (4-11),

$$F = \sum WI + b = \sum_{m=1}^{224} \sum_{n=1}^{224} W_{mn} I_{mn} + b \tag{4-11}$$

where *W* and *b* are the weight matrix and bias term in the FNN. It is found that Eqn. (4-11) is reduced to Eqn. (4-2) if the weight matrix *W* is set as all-ones matrix and the bias term *b* is set as zero. The activations in the output neuron was extracted and the feature distribution are shown in Figure 4-11. The intersection area is further decreased (Figure 4-11(b)) comparing to the previous handcrafted features in Figures 4-9(b) and 10(b). When the boundary was set as 0.451, the predicting accuracy per case was found to be 98.23% (healthy) and 95.95% (damaged), which were close to the estimated 98.27% (healthy) and 95.63% (damaged).

Next, the activations in the last feature layer of the CNN model used in this study were extracted, and the feature distribution is shown in Figure 4-12. An even better feature distribution can be qualitatively assessed in Figure 4-12(b). When the boundary was set as -0.38, the predicting accuracy per case was found to be 99.23%

(healthy) and 98.41% (damaged), which were close to the estimated rate 99.19% (healthy) and 98.42% (damaged).

It was found that the trained deep CNN model achieved the best classification performance by generalizing the most discriminative feature distribution with the least intersection area, comparing to the features of volume, modified volume, and trained FNN. The difference between the trained deep CNN model and the FNN was found to be the definition of the feature function. In the FNN, the feature function was restricted to the form as expressed in Eqn. (4-11), while a more advanced feature function is believed to be generalized by the CNN model through stacking different computation layers. The advancement of the feature function had potential to increase the classification performance.

4.5 Concluding Remarks

A deep learning based damage detection procedure using acceleration data is presented as a nondestructive evaluation tool for rapid post-hazard inspection and condition assessment task. In this study, the procedure is demonstrated with a case study of CBF structure, which is a widely used seismic load resisting system; a 6-story CBF building is selected as the prototype structure for the validation study. The deep learning model is trained and tested based on nonlinear time history analysis data from a total of 2,840 cases. For practical issues, two representative evaluation tasks are selected: Task (1)

detect the occurrence of bracing damage in the entire building using roof acceleration data only, and Task (2) not only detect the occurrence but also locate the damaged in the building using acceleration data measured from each floor. The results from the trained deep learning model were visually assessed on the recognized features through the saliency map. In order to interpret the results, a comparison was made between the inspired feature volume, modified volume, trained FNN and trained deep CNN model. The study shows that the proposed procedure is promising in offering an automated damage detection tool for rapid condition assessment. Specific findings are summarized as follows:

(I) The proposed damage detection method is effective in identifying the presence of seismic-induced brace damage (buckling) in the CBF structures, with the accuracy over 95.60% in this case study.

(II) The inspiration from the saliency map revealed a simple feature that was tested with acceptable accuracy, which showed the potential to learn from the deep learning. However, the performance of the inspired feature yielded to the optimization of the feature function.

(III) The deep CNN model was trained to discover more discriminative features comparing to the simple FNN through the optimization of more advanced feature function, which was generalized by the stacking of different computational layers.

(IV) The transfer learning was shown to be effective in training the deep CNN model for the detection of seismic-induced brace damage (buckling) in the CBF

structures. The transfer learning can simplify the design and training of the deep CNN model. The adapted CNN model here offered an option with fine-tuning a well pre-trained model on a database at an order of a few thousand training samples for this specific brace damage detection problem.

Figure 4-1. Deep learning based brace damage detection procedure

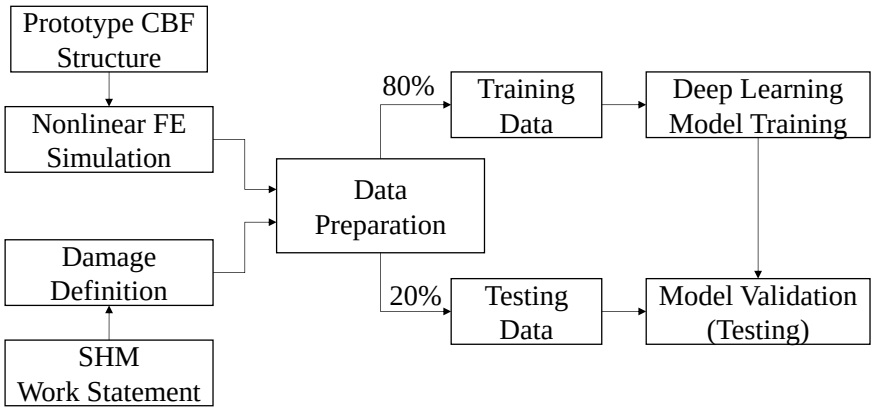


Figure 4-2. Brace damage detection procedure in the case study of a 6-story CBF building

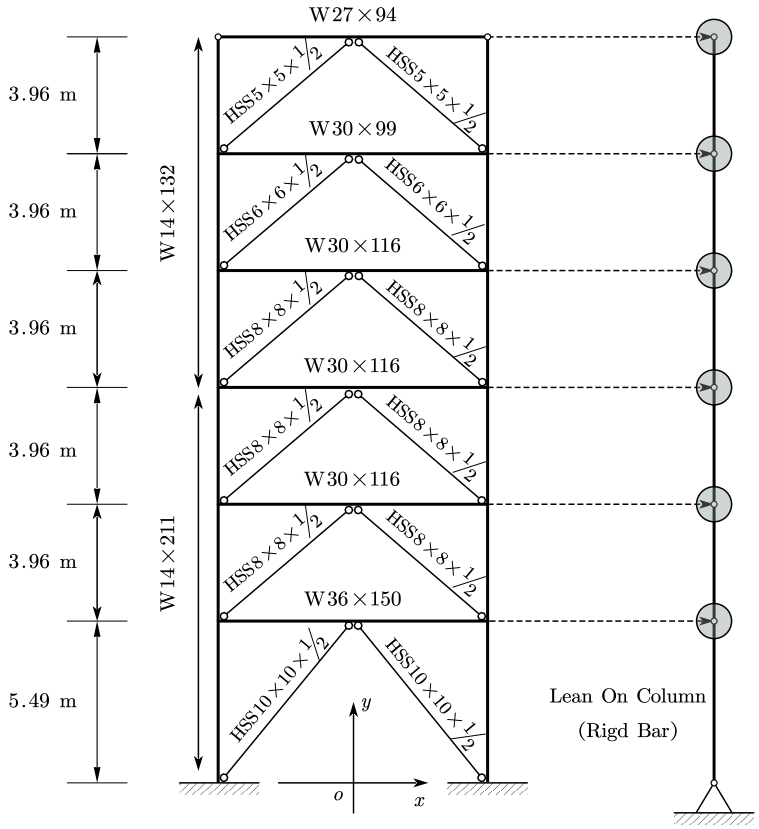


Figure 4-3. Configurations of prototype 6-story CBF building model

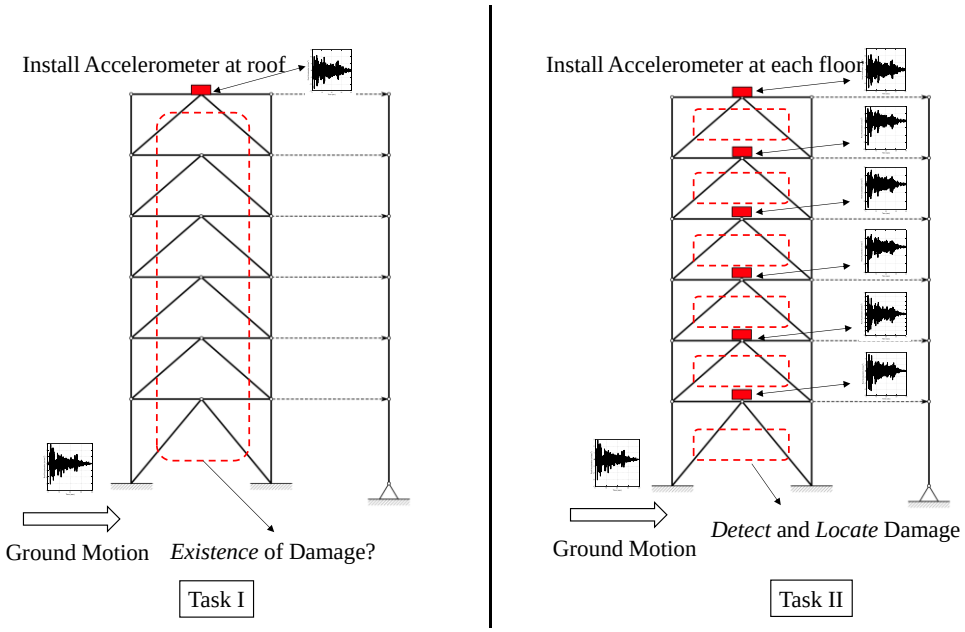
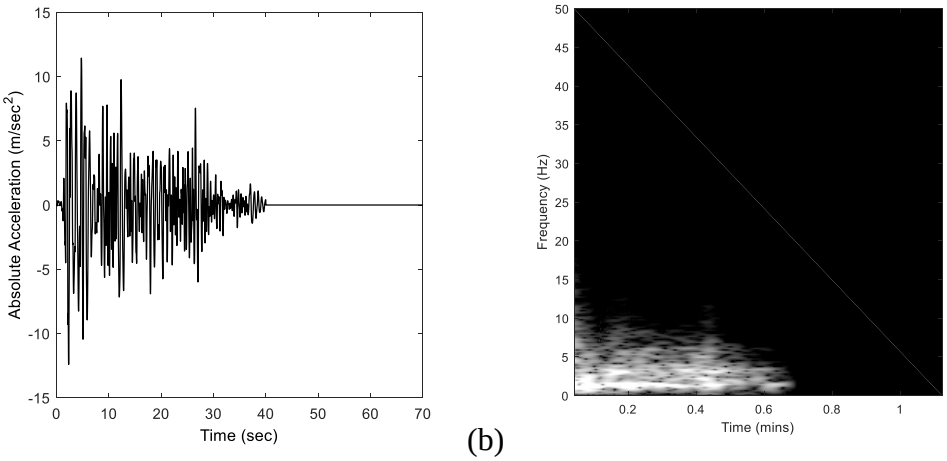


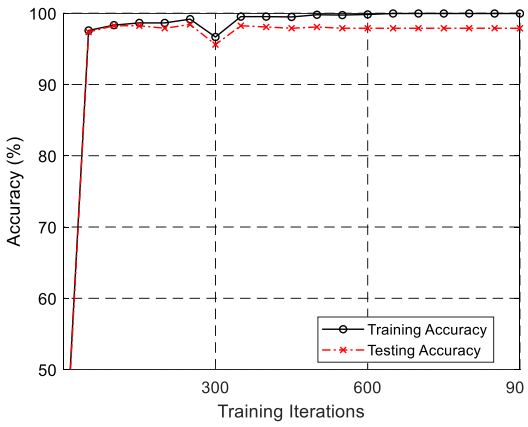
Figure 4-4. Sensor configuration in proposed brace damage detection scheme for model validation



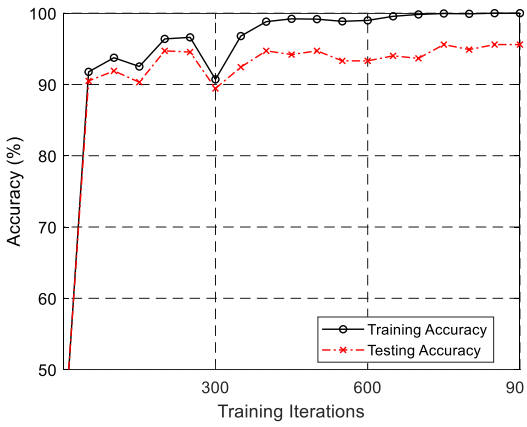
(a) (b)

Figure 4-5. Pre-processing of acceleration response data for CNN model training: (a) original acceleration time history; (b) the short-time Fourier spectrum (STFS) in grayscale

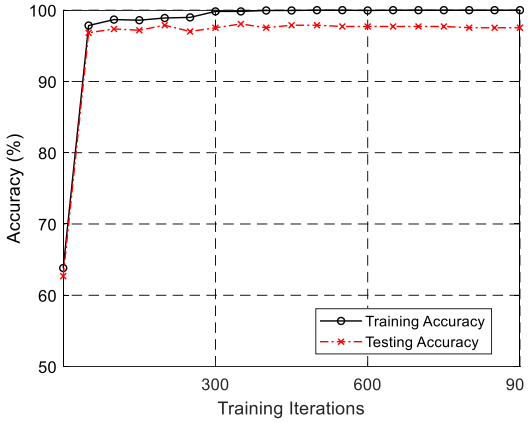
Task I



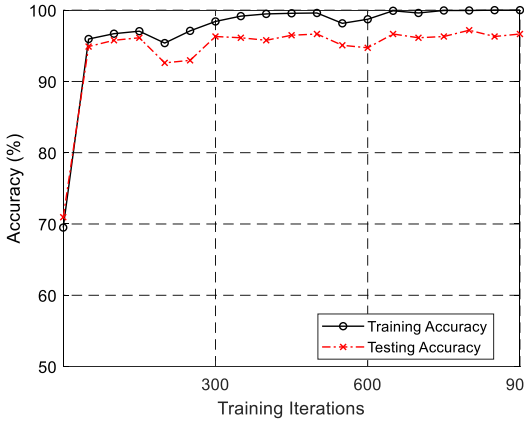
Task II (Floor 1)



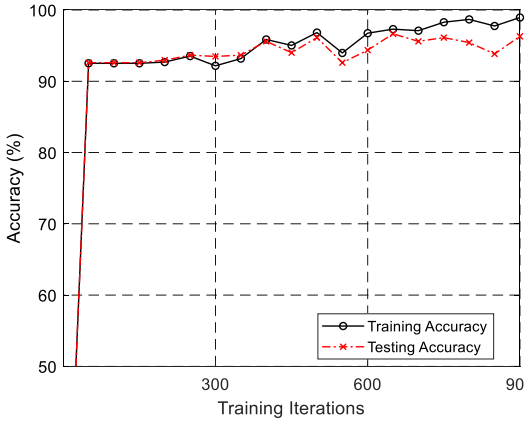
Task II (Floor 2)



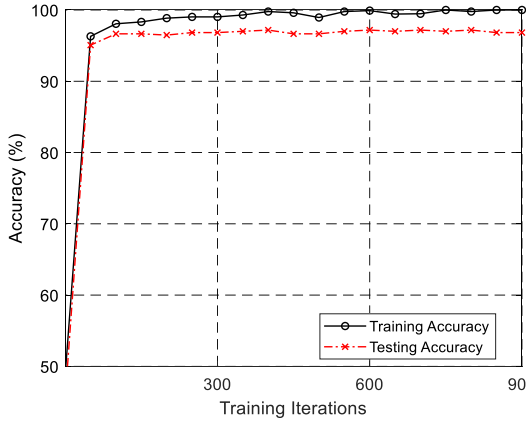
Task II (Floor 3)



Task II (Floor 4)



Task II (Floor 5)



Task II (Floor 6)

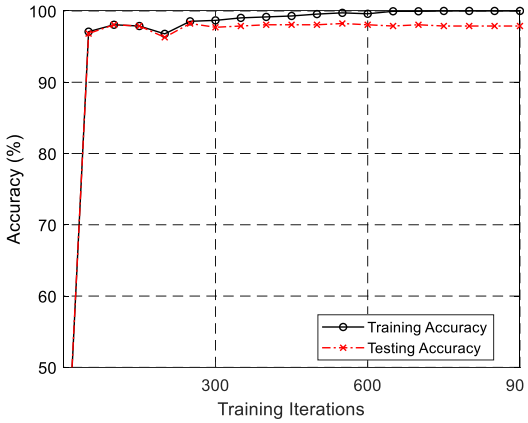


Figure 4-6. Learning curves of trained CNN models for Task I and Task II

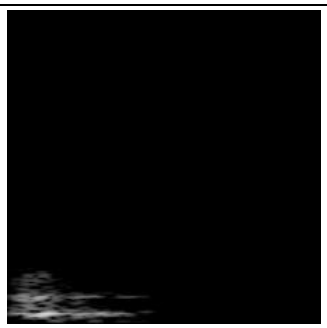
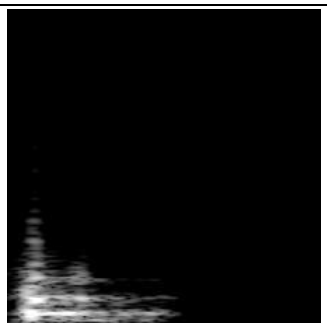
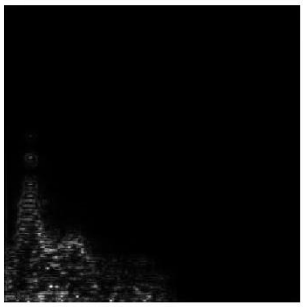
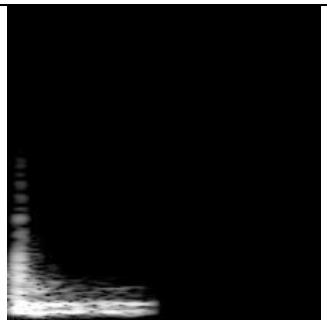
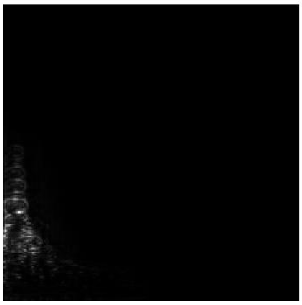
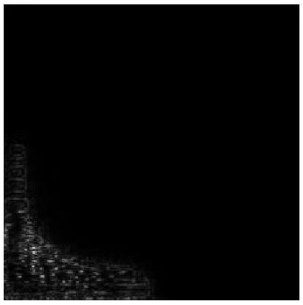
Input Image		Saliency map for Task I	Saliency map for Task II
#1			
#2			
#3			

Figure 4-7. Feature visualization of three randomly picked input images

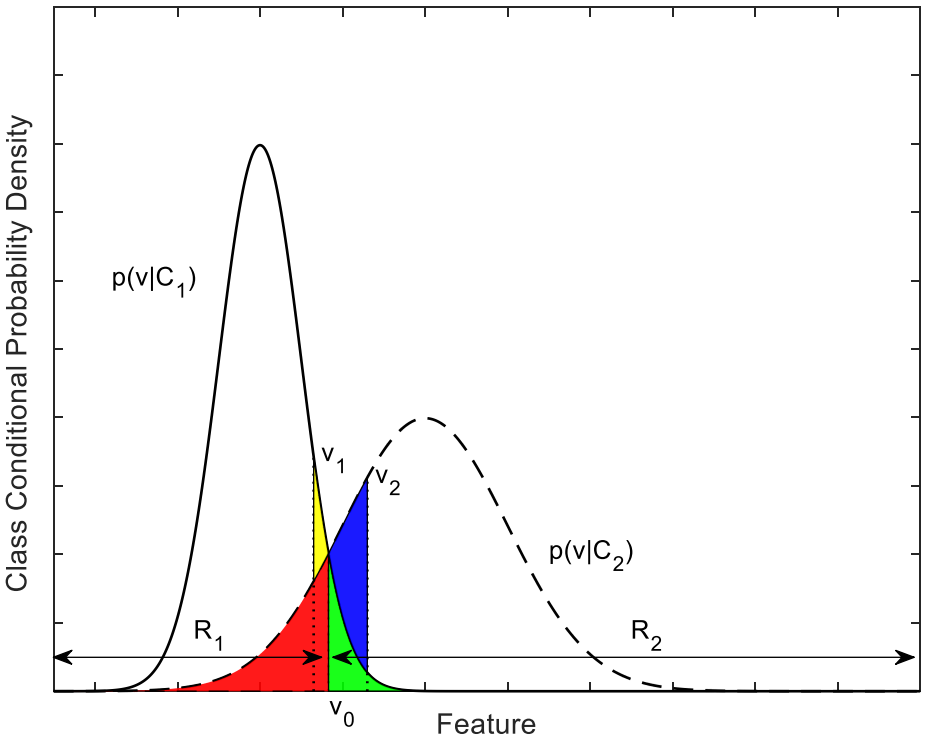
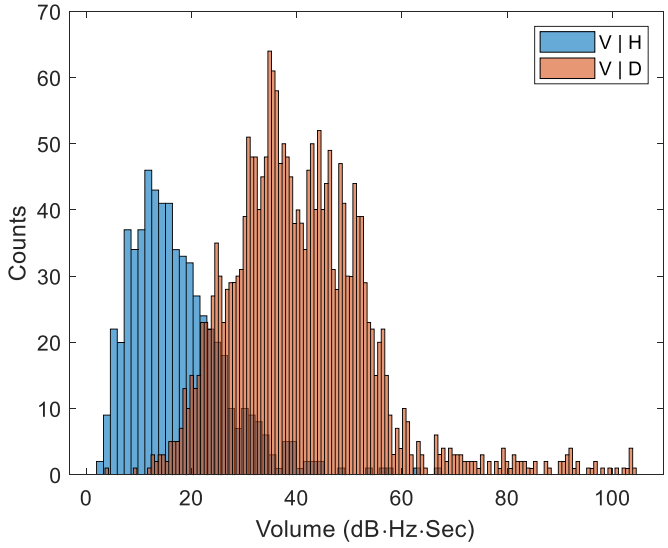
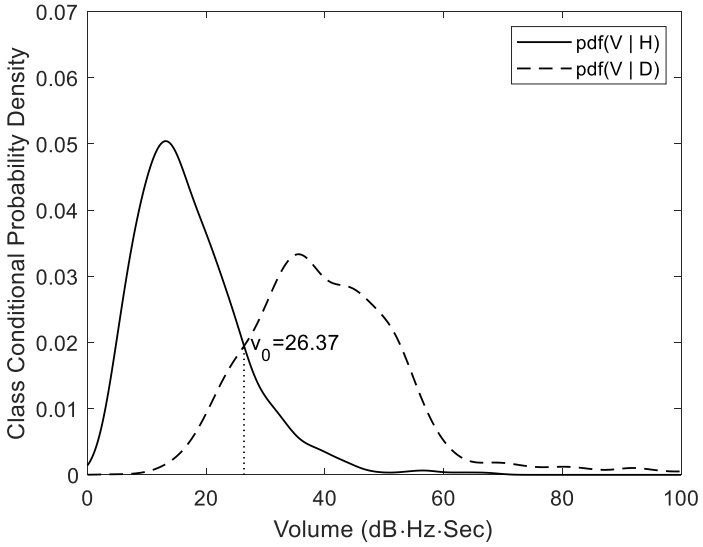


Figure 4-8. Schematic illustration of the optimal decision boundary to minimize the misclassification rate under the assumption of a uniform prior

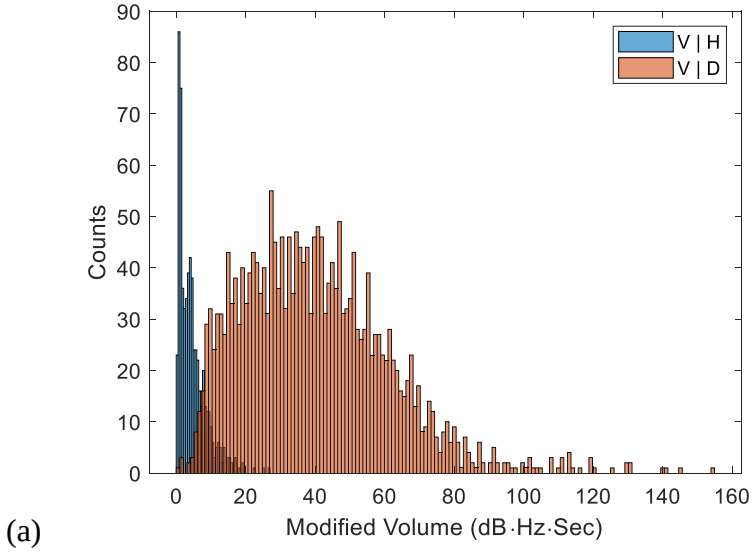


(a)

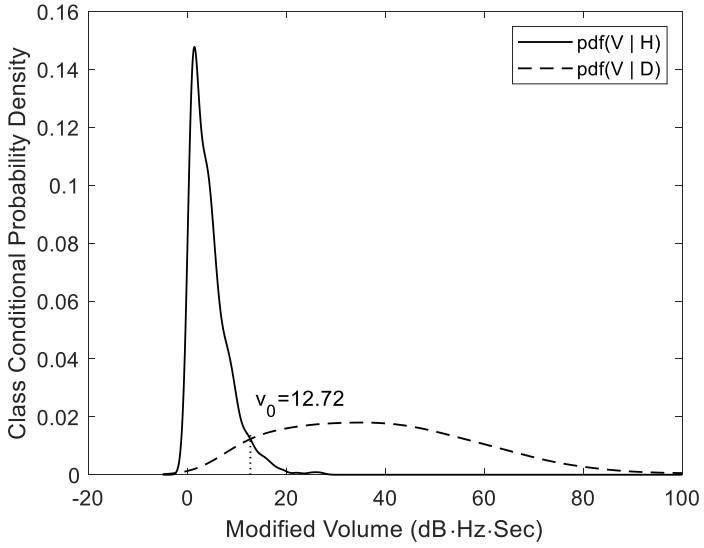


(b)

Figure 4-9. Statistical analysis of the computed volume feature: (a) histogram; (b) estimation of class conditional probability density (H = Healthy; D = Damaged; V = Volume; P = Probability; pdf = probability density function)

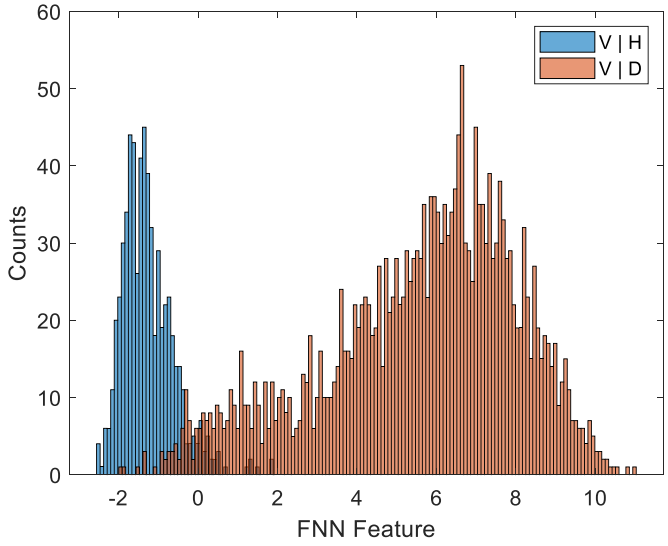


(a)

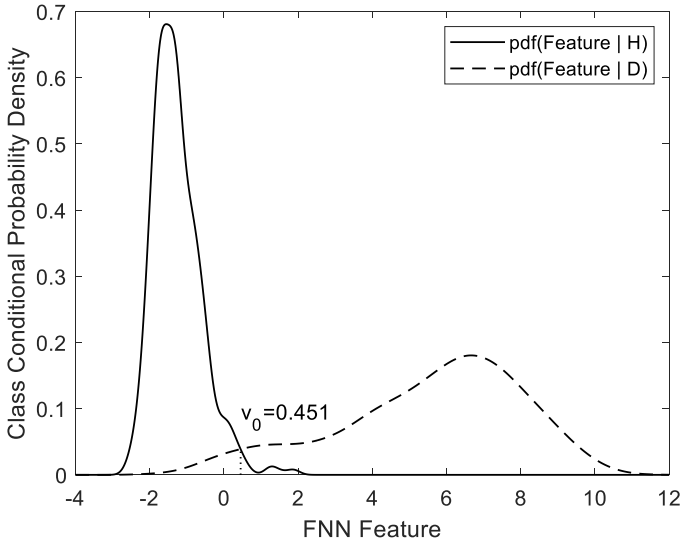


(b)

Figure 4-10. Statistical analysis of the computed modified volume feature: (a) histogram; (b) estimation of class conditional probability density

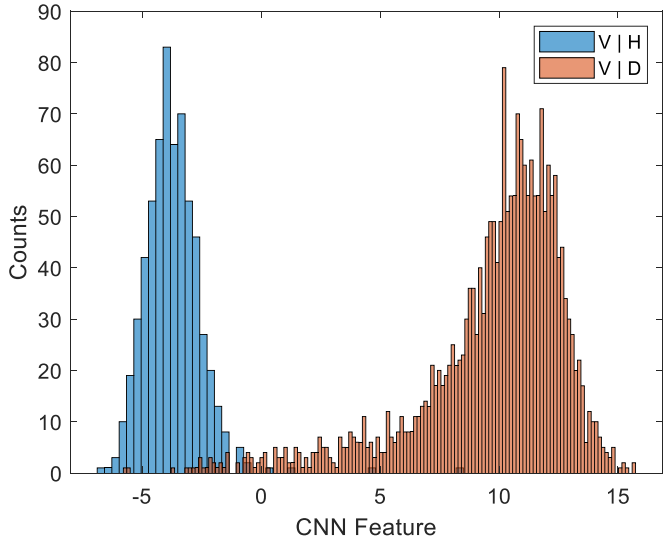


(a)

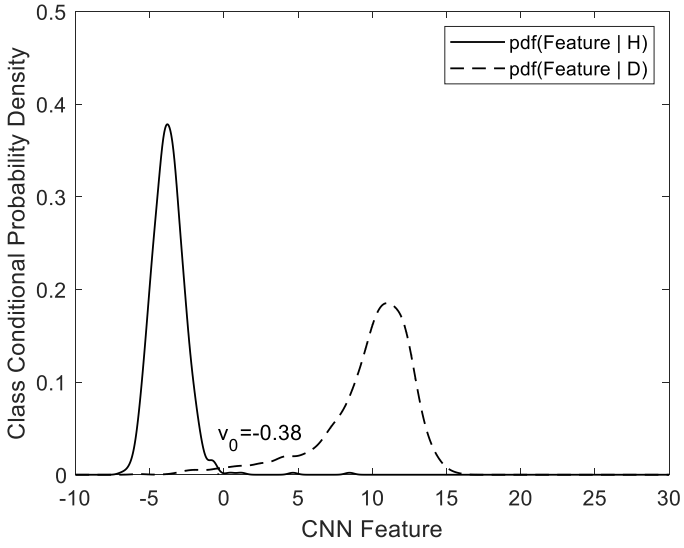


(b)

Figure 4-11. Statistical analysis of the computed FNN feature: (a) histogram; (b) estimation of class conditional probability density



(a)



(b)

Figure 4-12. Statistical analysis of the computed CNN feature: (a) histogram; (b) estimation of class conditional probability density

Table 4-1. Deep CNN model validation test results in two evaluation tasks

Tasks	Floor Level	# of Training Samples (Healthy/Damaged)	Test Accuracy of Brace Damage Detection (# of successful detection cases/total case #)		
			Healthy	Damaged	Overall
Task 1	-	2,272 (496/1,776)	94.35% (117/124)	98.87% (439/444)	97.89% (556/568)
Task 2	1	2,272 (924/1,348)	93.94% (217/231)	96.74% (326/337)	95.60% (543/568)
	2	2,272 (844/1,428)	96.21% (203/211)	98.32% (351/357)	97.53% (554/568)
	3	2,272 (1,400/872)	98.57% (345/350)	93.58% (204/218)	96.65% (549/568)
	4	2,272 (2,102/170)	98.29% (517/526)	71.43% (30/42)	96.30% (547/568)
	5	2,272 (564/1,708)	91.49% (129/141)	98.59% (421/427)	96.83% (550/568)
	6	2,272 (520/1,752)	91.54% (119/130)	99.77% (437/438)	97.89% (556/568)

Chapter 5: Nondestructive Evaluation using Guided Ultrasonic Lamb Wave Signals for Plate Structures

5.1 Introduction

This study aims to demonstrate and establish technical procedures so that deep learning can boost the significance of NDE of plate structures, increase effectiveness and reduces cycle times and reduce the need for manual intervention in damage detection and characterization of plate structures. The scope of the research is to enhance the damage identification process for plate-like civil infrastructures. The proposed method includes formalizing the NDE problem of structural damage condition assessment into an image classification problem in machine learning domain, which is then solved by transfer learning a pre-trained deep CNN model using the time-frequency spectrum images of the Lamb wave signals. The proposed method is validated using simulated data from a test setup with aluminum plates. Parametric studies have been performed to investigate the effects of central frequency, data noise level, and damage locations. The study results show that the proposed method is promising as an automated NDE tool for plate structures using guided Lamb waves.

5.2 Deep Learning based Procedure for Guided Wave Damage Detection

5.2.1 Technical Procedure

In practice, three aspects of the damage information are of particular interest: (1) the occurrence, (2) the location, and (3) the severity of the damage. For instance, cracks are typical damage forms in thin plate or shell structures, and an ideal damage detection tool shall as a minimum be able to report the occurrence of the crack, and if possible, the detection should also report the crack location in the plate and the severity of the crack (e.g., length and depth of crack).

The implementing procedure is illustrated in Figure 5-1. In this study, the proposed nondestructive damage evaluation method involves training a deep Convolutional Neural Network (CNN) with guided Lamb wave signals as input. The deep learning model is trained such that it is capable of reporting the anticipated damage information (e.g., occurrence, location, and severity) in the concerned thin plate structure. Since the adapted deep CNN model has demonstrated success in image classification tasks, it would be beneficial to present the damage detection task in the image classification framework, and thus the Lamb wave signal is first transformed into a time-frequency spectrum image. To alert the damage occurrence event, Lamb wave signals collected from an intact aluminum plate are also included in the training data of the CNN model as a baseline case. To report the damage location, the plate is

divided into a specified number of potential damage zones, and Lamb waves collected from several scenarios with different damage location (i.e., identical damage but occurring in different zones) are labeled correspondingly. After training, the validated CNN model is supposed to identify the damage location by distinguishing the Lamb waves interacting with the damage. To assess the severity of the damage, the Lamb waves collected from different scenarios of damage severity (i.e., different severities of damage occurred at the same location) are labeled accordingly. By doing so, the CNN model is trained and validated to distinguish the Lamb waves interacting with different severities of damage.

5.2.2 Deep Learning Model Description

5.2.2.1 Convolutional Neural Network

A typical application of deep learning can be found in computer vision field such as image classification (Krizhevsky et al., 2012) or object detection (Ren et al., 2015). CNN has been found to be very successful in the computer vision field, and thus it is also adapted in this study for the damage pattern recognition task in guided Lamb wave based NDE. The CNN is a specialized type of neural network (NN) for processing data with a known grid-like topology such as time-series data and image data. Examples of CNN models with reported record-setting performance are the AlexNet (Krizhevsky et al., 2012), GoogleNet (Szegedy et al., 2014), VGGNet (Simonyan & Zisserman, 2014),

and ResNet (He et al., 2016). These CNN models are featured with sophisticated network architectures in stacked computational layers including convolutional layer, pooling layer, ReLu layer, and fully connected layer. Details of these computational layers can be found in the book by Goodfellow et al. (2016).

5.2.2.2 Transfer Learning

An appealing aspect of the deep CNN models is their capability of transfer learning. Transfer learning refers to the situation where what has been learned in one setting (base dataset) is exploited to improve the generalization in another setting (target dataset) (Goodfellow et al., 2016). It is observed that the generalized features in early layers appear to be general regardless of the cost function and natural image dataset, and in later layers seem to be task-specific (Yosinski et al., 2014). For example, in the context of image classification, a general feature can be an edge detector while the task-specific feature can be the face of a human or the wheel of a car. Visual evidence of this finding can be found in the reference by Zeiler and Fergus (2014). Transfer learning is also important to the application of lamb wave based damage detection in this study for two reasons: (1) sophisticated architecture design of those validated CNN models with demonstrated success can be taken advantage of, and (2) the size of acquirable database for Lamb wave signals (e.g., order of thousands in this study) can be substantially reduced comparing to huge-sized image database (order of millions) typically seen in computer vision field such as the ImageNet (Deng et al., 2009), from

which these sophisticated deep CNN models were trained. When available data is limited in size, transfer learning has yielded promising results in image-driven structural steel damage condition assessment application reported earlier in this dissertation. Therefore, transfer learning is adapted here, and the pre-trained deep CNN model (VGG-16) by Simonyan and Zisserman (2014) has been used in this study as the base model due to its advanced network structure and manageable computational cost. The architecture of the adapted CNN model is plotted in Figure 5-2.

5.2.2.3 Cost Function

The cost function is the objective function that defines the training target of the CNN model. In this study, the CNN model is trained in the classification scheme for damage pattern recognition. For classification tasks, the SoftMax with cross-entropy is reported as the most frequently used cost function (Goodfellow et al., 2016). The Softmax function is used to represent the probability distribution over C different classes as described in Eqn. (5-1),

$$P\left(\hat{y}=i|x,\theta\right)=\frac{\exp\left(z^{i}\right)}{\sum_{i=1}^{C}\exp\left(z^{i}\right)} \tag{5-1}$$

where $P(\hat{y}=i|x,\theta)$ (i=1,2,…,C) represents the probability of the output-class of i when given the input x and model parameters θ; z^i denotes the activations in the proceeding layer (e.g., a fully connected layer). The cross-entropy loss is used to represent the

dissimilarity between the empirical distribution (p_{data}) defined by the training data and the model distribution estimated by the CNN model, formulated as Eqn. (5-2),

$$L(x) = E_{x \sim p_{data}}[-\log p_{model}(x,\theta)] = -\sum_{x} y_x \log p_{model}(x,\theta) \quad (5\text{-}2)$$

where y_x is the true label of the data sample x and the $p_{model}(x)$ can be represented using the SoftMax function. With the SoftMax with cross-entropy loss, the CNN model is trained to search for a feature space (i.e. z^i) that can make better classifications (i.e., higher prediction probability $p_{model}(x)$) over the training data x.

5.2.2.4 Training of Deep Learning Model

Mathematically, the training of the deep learning model is to search for the CNN model parameters θ that has the least cost function value. This is achieved through an iteration process by adjusting the parameters through gradient descent based backpropagation approach, as described in Eqn. (5-3),

$$p_{new} = p_{old} - \eta \frac{\partial L}{\partial p_{old}} \quad (5\text{-}3)$$

where p denotes each parameter (including weights and biases) in the CNN model, the derivative is calculated through backpropagation method (Rumelhart et al., 1986) and the learning rate (η) is a parameter specified by the user. A high learning rate means that bigger steps are taken in the weight updates, so it would take less time for the model to converge on an optimal set of weights, while too small a learning rate will slow down the gradient descent and hence the overall training process. The

conventional gradient descent requires the computation of the gradients over all training samples, so the backpropagation has to be repeated that many times. When the training data size increases, this process will become slower. The stochastic gradient descent (SGD) approach (Lecun et al., 1998) has been proposed to accelerate this process by calculating the gradients from a batch of randomly selected sub-dataset of the entire training database. The sub-dataset is called the mini-batch, and its size is called mini-batch size. However, since the gradients from SGD only depends on the mini-batch of the training data, the variance of the stochastic gradient would also slow or even diverge the training process. The SGD with momentum is introduced to solve this problem. Formally, the momentum introduces a variable that plays the role of velocity which will guide the direction (gradients) of parameter updates. Details regarding the algorithm can be found in the reference (Goodfellow et al., 2016). The SGD with moment will be used as the training algorithm in this study.

5.2.2.5 Damage Feature Pattern Visualization

While deep learning models have demonstrated impressive performance on image classification and object detection, there is a lack of deep insight in how and why the deep learning models such as CNN worked so well and as such model improvement has been restricted to trial-and-error process (Zeiler & Fergus, 2014). To address the need for visualizing features, researchers have investigated visualization techniques that gives insight into the function of intermediate feature layers and the feature

evolution during training (Zeiler & Fergus, 2014; Simonyan et al., 2013; Springenberg et al., 2014). The deconvolutional neural network used by Zeiler and Fergus (2014) provides a non-parametric view of invariance, showing which patterns from the training set activate the feature map. Simonyan et al. (2013) further elucidated the function of the deconvolutional neural network for feature visualization by computing image-specific class saliency map, which approximates the highly non-linear scoring function mapped from CNN model by evaluating its first-order Taylor expansions as Eqn. (5-4),

$$\begin{aligned} S_C(I) &\approx S_C(I_0) + \frac{\partial S_C}{\partial I}|_{I_0} \cdot (I - I_0) \\ &= \frac{\partial S_C}{\partial I}|_{I_0} \cdot I + \left(S_C(I_0) - \frac{\partial S_C}{\partial I}|_{I_0} \cdot I_0 \right) \end{aligned} \tag{5-4}$$

where S_c represents the scoring function; I denotes input variable (images) and I_0 is a given specific image. The saliency map is established based on the derivative respect to the input image of the score function that its magnitudes indicate which pixel needs to be adjusted the least to affect the score. They pointed out that the deconvolution layer is either equivalent or similar to computing the gradient of the visualized layer. It is effectively corresponding to the gradient back-propagation through a convolutional layer. The reason for the "either equivalent or similar to computing the gradient" is the ReLU layer. Therefore, guided-backpropagation (Springenberg et al., 2014) was proposed to achieve enhanced accuracy in calculating the gradient back-propagation through a convolutional layer. The difference between deconvolutional neural network,

gradient backpropagation and guided backpropagation were compared in the reference (Springenberg et al., 2014). The guided back-propagation method is used in this study to visualize the feature patterns identified in the CNN model, so the classification results for damage pattern identification can be more easily assessed.

5.3 Analytical Simulation of Lamb Wave Interaction with Notch Damages

Comparing to other methods of generating the Lamb wave, the piezoelectric sensors, such as the Piezoelectric lead zirconate titanate (PZT) element (Dimitriadis & Fuller, 1991; Badcock & Birt, 2000), piezoelectric paint sensor (Li & Zhang, 2008), and the Piezoelectric Waver Active Sensors (PWAS) (Giurgiutiu, 2005), have been found to deliver excellent performance in Lamb wave generation and acquisition. To acquire the data required for the training of the deep CNN model, analytical simulation of guided Lamb wave signals measured by piezoelectric wafers mounted on the surface of thin plates is formulated here in a time efficient way. It is noted that the formulation is adapted from the work by Giurgiutiu (2005) as well as Shen and Giurgiutiu (2014), in which the formulas have been validated with experimental data on aluminum plates.

5.3.1 Piezoelectric Fundamentals

Piezoelectric material based sensors can produce electric charge signals proportional to the applied deformation in its coupled surface based on the principles described in Eqn. (5-5),

$$\begin{aligned} D_i &= \varepsilon_{ij}^{T} E_j + d_{im}^{d} \sigma_m \\ S_k &= d_{jk}^{c} E_j + s_{km}^{E} \sigma_m \end{aligned} \tag{5-5}$$

where i, j = 1, 2, 3 and k, m = 1, 2, …, 6. Here, D, E, T and S are the dielectric displacement, electric field, stress, and strain, respectively; ε^{T}_{ij}, d^{c}_{jk} (or d^{d}_{im}), and s^{E}_{km} are the dielectric permittivity (Farad/m), the piezoelectric coefficients (Coulomb/N or m/Volt), and the elastic compliance (m^2/N); the piezoelectric coefficient d^{c}_{jk} defines the strain per unit electric field at constant stress and d^{d}_{im} defines electric displacement per unit stress at constant electric field. The superscripts c and d are used to distinguish between the converse and direct piezoelectric effects. The superscripts σ and E indicate that the quantity is measured at constant stress and constant electric field, respectively (Li & Zhang, 2008).

5.3.2 Guided Lamb Wave Propagation in Pristine Plate

Consider the plate surface with piezoelectric wafer sensor shown in Figure 5-3 (a), the plate is subjected to exciting shear force transferred from the piezoelectric wafer sensor which is excited by external voltage signal. Under the piezoelectric transduction

principle, the equation of Lamb Wave propagation excited by the piezoelectric wafer can be described as Eqn. (5-6):

$$\begin{aligned} \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\omega^2}{c_p^2}\phi = 0 \\ \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} + \frac{\omega^2}{c_p^2}\psi = 0 \end{aligned} \tag{5-6}$$

where $c_L^2 = (\lambda+2\mu)/\rho$ and $c_T^2 = \mu/\rho$; c_p and c_s are the longitudinal and transverse wave speed; λ and μ are Lame constants and ρ is the mass density of the material. By applying the Fourier transform on the potential function ϕ and ψ in the domain of wavenumber k ($k = \omega/c_p$ and c_p is the phase velocity)

$$\begin{aligned} \tilde{\phi}(k) = \int_{-\infty}^{+\infty} \phi(k) e^{-ikx} dx \ , \ \tilde{\psi}(k) = \int_{-\infty}^{+\infty} \psi(k) e^{-ikx} dx \\ \phi(k) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \tilde{\phi}(k) e^{ikx} dk \ , \ \psi(k) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \tilde{\psi}(k) e^{ikx} dk \end{aligned} \tag{5-7}$$

yields:

$$\begin{aligned} -k^2\tilde{\phi} + \frac{\mathrm{d}^2\tilde{\phi}}{\mathrm{d}y^2} + \frac{\omega^2}{c_L^2}\tilde{\phi} = 0 \\ -k^2\tilde{\psi} + \frac{\mathrm{d}^2\tilde{\psi}}{\mathrm{d}y^2} + \frac{\omega^2}{c_T^2}\tilde{\psi} = 0 \end{aligned} \tag{5-8}$$

By introducing the notations below,

$$p^2 = \frac{\omega^2}{c_L^2} - k^2$$
$$q^2 = \frac{\omega^2}{c_T^2} - k^2 \tag{5-9}$$

The general solution of Eqn. (5-8) can be expressed as,

$$\tilde{\phi} = A_1 \sin(py) + A_2 \cos(py)$$
$$\tilde{\psi} = B_1 \sin(qy) + B_2 \cos(qy) \tag{5-10}$$

The above four integration constants, A_1, A_2, B_1, B_2, are determined through boundary conditions associated with displacement, strain and stress functions in the wavenumber domain as shown in Eqn. (5-11):

$$\tilde{u}_x = ik\tilde{\phi} + \frac{\mathrm{d}\tilde{\psi}}{\mathrm{d}y}$$
$$\tilde{u}_y = \frac{\mathrm{d}\tilde{\phi}}{\mathrm{d}y} - ik\tilde{\psi}$$
$$\tilde{\tau}_{yx} = \mu\left(2ik\frac{\mathrm{d}\tilde{\phi}}{\mathrm{d}y} + k^2\tilde{\psi} + \frac{\partial^2\tilde{\psi}}{\partial y^2}\right) \tag{5-11}$$
$$\tilde{\tau}_{yy} = \lambda\left(-k^2\tilde{\phi} + \frac{\mathrm{d}^2\tilde{\phi}}{\mathrm{d}y^2}\right) + 2\mu\left(-k^2\tilde{\phi} - ik\frac{\mathrm{d}\tilde{\psi}}{\mathrm{d}y}\right)$$
$$\tilde{\varepsilon}_x = iku_x$$

As the excitation from the piezoelectric wafer is acted to the top surface of the plate, the signal can be decomposed into symmetric and antisymmetric mode (see Figure 5-3 (b)), which yield the boundary conditions expressed in Eqn. (5-12) and Eqn. (5-13), where τ is the bonding shear stress between the piezoelectric wafer and plate.

For symmetric mode,

$$\begin{aligned} \tilde{u}_x(k)\,|_{y=d} &= \tilde{u}_x(k)\,|_{y=-d} \\ \tilde{u}_y(k)\,|_{y=d} &= -\tilde{u}_y(k)\,|_{y=-d} \\ \tilde{\tau}_{yx}(k)\,|_{y=d} &= -\tilde{\tau}_{yx}(k)\,|_{y=-d} = \frac{\tilde{\tau}}{2} \\ \tilde{\tau}_{yy}(k)\,|_{y=d} &= \tilde{\tau}_{yy}(k)\,|_{y=-d} = 0 \end{aligned} \tag{5-12}$$

For antisymmetric mode,

$$\begin{aligned} \tilde{u}_x(k)\,|_{y=d} &= -\tilde{u}_x(k)\,|_{y=-d} \\ \tilde{u}_y(k)\,|_{y=d} &= \tilde{u}_y(k)\,|_{y=-d} \\ \tilde{\tau}_{yx}(k)\,|_{y=d} &= \tilde{\tau}_{yx}(k)\,|_{y=-d} = \frac{\tilde{\tau}}{2} \\ \tilde{\tau}_{yy}(k)\,|_{y=d} &= -\tilde{\tau}_{yy}(k)\,|_{y=-d} = 0 \end{aligned} \tag{5-13}$$

Therefore, the complete solution of Eqn. (5-8) can be achieved by solving for the four integration constants A_1, A_2, B_1 and B_2 based on the boundary conditions expressed as Eqn. (5-12) and Eqn. (5-13). As the piezoelectric wafer sensor is a strain measurement based sensor whose electrical signal is in direct relation with strains (i.e., Eqn. (5-5)) on the surface of the plate, the solution in the form of strain is in the most interested (Eqn. (5-14)):

$$\tilde{\varepsilon}_x = -i\frac{\tilde{\tau}}{2\mu}\left(\frac{N_S}{D_S} + \frac{N_A}{D_A}\right) \tag{5-14}$$

where

$$
\begin{aligned}
N_S &= kq(k^2+q^2)\cos(pd)\cos(qd) \\
D_S &= (k^2-q^2)^2\cos(pd)\sin(qd) + 4k^2pq\sin(pd)\cos(qd) \\
N_A &= kq(k^2+q^2)\sin(pd)\sin(qd) \\
D_A &= (k^2-q^2)^2\sin(pd)\cos(qd) + 4k^2pq\cos(pd)\sin(qd)
\end{aligned}
\tag{5-15}
$$

By applying the inverse Fourier transform on Eqn. (5-14) and residue theorem, also assuming the ideal bonding between the piezoelectric wafer sensor and plate so that the bonding force can have closed-form Fourier transform (Giurgiutiu, 2005), then the solution of strain in time domain will be Eqn. (5-16):

$$
\begin{aligned}
\varepsilon_x(x,t) = &-i\frac{a\tau_0}{\mu}\sum_{k^S}\frac{\sin(k^S a)N_S(k^S)}{D_S'(k^S)}e^{i(k^S x-\omega t)} \\
&-i\frac{a\tau_0}{\mu}\sum_{k^A}\frac{\sin(k^A a)N_A(k^A)}{D_A'(k^A)}e^{i(k^A x-\omega t)}
\end{aligned}
\tag{5-16}
$$

where a is the half size of the piezoelectric wafer sensor; $a\tau_0$ is the pin force applied at the piezoelectric based sensor; k^S, k^A are the all singular roots of $D_S=0$ and $D_A=0$, which are also the alternative form of Rayleigh-Lamb equation for symmetric and antisymmetric mode (i.e., Eqn. (5-17), +1 for symmetric and -1 for antisymmetric, and d represents the thickness of the plate).

$$
\frac{\tan(pd)}{\tan(qd)} = -\left[\frac{4k^2pq}{(k^2-q^2)^2}\right]^{\pm 1} \tag{5-17}
$$

If applying the Fourier transform in frequency domain (not wavenumber domain) on both the input signal and response, the transfer function $G(x, \omega)$ from inputting the

piezoelectric wafer excitation signal to output strain response can be written as Eqn. (5-18).

$$G(x,w) = -i\frac{a\tau_0}{\mu}\left(\sum_{k^S}\frac{\sin(k^S a)N_S(k^S)}{D_S'(k^S)}e^{ik^S x} + \sum_{k^A}\frac{\sin(k^A a)N_A(k^A)}{D_A'(k^A)}e^{ik^A x}\right) \tag{5-18}$$

Therefore, for any given input time series signal $V(t)$ (or $V(\omega)$ in frequency domain), the output response can be obtained based on transfer function $G(x, \omega)$ by using Eqn. (5-19), if the Fourier transform is applicable .

$$\varepsilon_x(x,t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} G(x,\omega)V(w)e^{iwx}\mathrm{d}x \tag{5-19}$$

5.3.3 Lamb Wave Interacting with Damages

When the plate is damaged, Eqn. (5-19) will be no longer applicable. Shen and Giurgiutiu (2014) proposed a theoretical model to represent the interaction between Lamb Wave and damages. The idea is to consider the damage as new wave source by introducing wave transmission, reflection, mode conversion and higher harmonic effects on the wave signal arrived at the damage location. The method has been validated by comparing with experimental results showing good agreements. Thus, the method is adopted here to simulate the Lamb Waves interacting with damages.

For a given damaged plate as shown in Figure 5-4, the Lamb Wave signal arrives at the damage location (x_d) will be described in time domain by Eqn. (5-20) and in frequency domain by Eqn. (5-21).

$$\varepsilon_x(x_d,t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} G(x_d,\omega)V(w)e^{iwx}\mathrm{d}x \tag{5-20}$$

Eqn. (5-21) is used to introduce the wave transmission and mode conversion effects,

$$\begin{aligned}
\tilde{\varepsilon}_x(x_d,\omega) &= G(x_d,\omega)V(w) = \tilde{\varepsilon}_x^S(x_d,\omega) + \tilde{\varepsilon}_x^A(x_d,\omega)\\
\tilde{\varepsilon}_x^S(x_d,\omega) &= -i\frac{a\tau_0}{\mu}\left(\sum_{k^S}\frac{\sin(k^S a)N_S(k^S)}{D_S'(k^S)}e^{ik^S x}\right)V(w)\\
\tilde{\varepsilon}_x^A(x_d,\omega) &= -i\frac{a\tau_0}{\mu}\left(\sum_{k^A}\frac{\sin(k^A a)N_A(k^A)}{D_A'(k^A)}e^{ik^A x}\right)V(w)\\
\tilde{\varepsilon}_{x,new}(x_r,x_d,\omega) &= \sum_{k^S}\left[C_{SST}^N e^{-i\varphi_{SST}^N}\tilde{\varepsilon}_x^S(x_d,\omega) + C_{AST}^N e^{-i\varphi_{AST}^N}\tilde{\varepsilon}_x^A(x_d,\omega)\right]e^{-ik^S(x_r-x_d)}\\
&+ \sum_{k^A}\left[C_{AAT}^N e^{-i\varphi_{AAT}^N}\tilde{\varepsilon}_x^A(x_d,\omega) + C_{SAT}^N e^{-i\varphi_{SAT}^N}\tilde{\varepsilon}_x^S(x_d,\omega)\right]e^{-ik^A(x_r-x_d)}
\end{aligned} \tag{5-21}$$

where C_{SST}, C_{SAT}, C_{AAT}, C_{AST}, φ_{SST}, φ_{SAT}, φ_{AAT}, φ_{AST} are damage coefficients with first letter denoting the incident wave type and second letter for resulting wave type and third letter for propagation direction respectively. Specifically, S is symmetric, A is antisymmetric, and T is transmission. C_{SST} will represent the incoming symmetric wave (S) resulting symmetric wave (S) in transmission (T).

To introduce the higher order harmonic effects, Eqn. (5-21) is modified by moving the central frequency axis to higher order frequencies as shown in Eqn. (5-22)

. In the work by Shen and Giurgiutiu (2014), *h* equals to 1 or 2. However, Eqn. (5-22) gives a more general form in which any higher order frequency can be modeled. For example, if *h* is 0, there is no higher order harmonic effect, then Eqn. (5-22) and Eqn. (5-21) are identical. The damage coefficients in Eqn. (5-22) have the same meaning and denotations as those in Eqn. (5-21). In numerical implementation, moving frequency axis can be achieved by padding a certain length of zeros in front to the Fourier spectrum of Lamb Wave arriving at the damage location (i.e. $\tilde{\varepsilon}_x^H(x_d,\omega)$ in Eqn. (5-22)).

$$\tilde{\varepsilon}_x^H(x_d,\omega) = \tilde{\varepsilon}_x(x_d,\omega - h^*\omega_c),\ h \in R^+$$

$$\tilde{\varepsilon}_x^{SH}(x_d,\omega) = -i\frac{a\tau_0}{\mu}\left(\sum_{k^S}\frac{\sin(k^S a)N_S(k^S)}{D_S'(k^S)}e^{ik^S x}\right)V(\omega - h^*\omega_c)$$

$$\tilde{\varepsilon}_x^{AH}(x_d,\omega) = -i\frac{a\tau_0}{\mu}\left(\sum_{k^A}\frac{\sin(k^A a)N_A(k^A)}{D_A'(k^A)}e^{ik^A x}\right)V(\omega - h^*\omega_c)$$

$$\tilde{\varepsilon}_{x,new}^H(x_r,x_d,\omega) = \sum_{k^S}\left[C_{SST}^N e^{-i\varphi_{SST}^N}\tilde{\varepsilon}_x^{SH}(x_d,\omega) + C_{AST}^N e^{-i\varphi_{AST}^N}\tilde{\varepsilon}_x^{AH}(x_d,\omega)\right]e^{-ik^S(x_r - x_d)}$$

$$+\sum_{k^A}\left[C_{AAT}^N e^{-i\varphi_{AAT}^N}\tilde{\varepsilon}_x^{AH}(x_d,\omega) + C_{SAT}^N e^{-i\varphi_{SAT}^N}\tilde{\varepsilon}_x^{SH}(x_d,\omega)\right]e^{-ik^A(x_r - x_d)}$$

$$\tilde{\varepsilon}_{x,new}^{HT}(x_r,x_d,\omega) = \sum_h \tilde{\varepsilon}_{x,new}^H(x_r,x_d,\omega)$$

(5-22)

Then, the strain response at receiving location can be obtained through the inverse Fourier transform as shown in Eqn. (5-23).

$$\varepsilon_x(x_r,x_d,t)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}\tilde{\varepsilon}_{x,new}(x_r,x_d,\omega)e^{iwx}\mathrm{d}x$$
$$\text{or }\ \varepsilon_x^H(x_r,x_d,t)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}\tilde{\varepsilon}_{x,new}^{HT}(x_r,x_d,\omega)e^{iwx}\mathrm{d}x \tag{5-23}$$

Based on the analytical modeling of Lamb Wave interacting with damages, the Lamb wave signals from different damage scenarios (e.g., different severity of damage or different damage locations) can be collected.

5.4 Case Study of Deep Learning Method: Guided Lamb Wave based NDE

5.4.1 Test Setup

The test setup is illustrated in Figure 5-4. A pair of piezoelectric wafers (one as the exciter, and the other as the receiving sensor) are installed on the same surface of a thin plate structure. A pitch-catch configuration is adopted, which consists of a transmitter on one side to excite a Lamb wave signal and a receiver on the other side to capture the Lamb wave propagation at a predefined distance. Two aluminum plates with different severities of damage are examined: a 3.17 mm-thick-plate with a notch (0.25 mm in width and 2.5 mm in depth), and a 2-mm-thick-plate with a breathing crack (0 mm in width and 1.2 mm in depth), as sketched in Figure 5-5. The choice of these two plates is made by considering the availability of calibrated damage coefficients found in the

analytical modeling study by Shen and Giurgiutiu (2014). The distance between the two piezoelectric wafers is prescribed as 400 mm, and the plate within the two wafers is evenly divided into four damage regions. Hence, for one type of damage, it has five potential damage conditions, including one pristine state (the damage not happened) and four damaged conditions representing its location within different damage regions. Henceforth, the performance of the proposed damage evaluation method can be primarily examined in the following: (1) detect the occurrence of the specified damage; (2) If damage occurred, detect the location of the damage (one of the four potential damage region). This damage detection task is converted into a classification task by training a deep CNN model. The classification on different types of damage is usually desired in practice, so the potential of the proposed method for detecting different types of damage is also examined in the experiment by combing all potential condition states or damage cases from the two plates into one CNN model for classification, however, the pristine case of the two plates with different thickness are grouped into one class. As such, the CNN model is trained to classify a total of 9 condition classes.

The Lamb wave mode selection is another critical issue for field implementation. The research work indicated that a proper Lamb mode for damage detection should feature non-dispersion, and it has been found that a narrow bandwidth input signal is able to effectively prevent wave dispersal (Wilcox et al., 2001). Thus, windowed tone-burst, rather than the pulse, is frequently utilized as the diagnostic Lamb wave signal. In this study, a three-counts tone-burst signal is used to excite the

Lamb waves as shown in Figure 5-6. The tone-burst signal is generated by multiplying a single frequency carrier with a hanging window. The number of counts equals the number of periods and the frequency of the carrier is called the center frequency. It can be found that the center frequency will have an effect on the phase velocity of the A0 and S0 mode in the dispersion curve of the plate as shown in Figure 5-7. The effect, in turn, will change the mode amplitude in the collected Lamb wave signals as shown in the tuning curves in Figure 5-8 and the corresponding visual features (as discussed in Section 5.4.4.2). Hence, in order to investigate the preference of the proposed method on center frequencies of the input signal, three center frequency values are considered based on the tuning curve: 100 kHz, 220 kHz, and 300 kHz.

The sensitivity to white noises is also examined here. The artificial white noises are created and superimposed to the collected Lamb wave signals. The white noise effects are controlled by adjusting the signal-to-noise-ratio (SNR), as described in Eqn. (5-24),

$$\mathrm{SNR} = 10\log_{10}\left[\left(\frac{\mathrm{RMS(signal)}}{\mathrm{RMS(noise)}}\right)^2\right] \tag{5-24}$$

where RMS is the root mean square amplitude of the signal, and a larger SNR value indicates a signal with better quality. Three level of white noises (with an SNR of 15, 20, or 25 dB) are considered in the study. In terms of the RMS of the original signal amplitude, the SNR ratio of 15, 20, or 25 dB represents an RMS ratio (noise to signal) of 17.78%, 10%, or 5%. A set of Lamb wave signal collected from the pristine plate

with a thickness of 2 mm is plotted in Figure 5-9 to show the effects of white noises with different SNR values.

5.4.2 Data Collection

The analytical formulas in Section 5.3 are used to generate the Lamb wave signals, with and without damage interactions in the aluminum plates. It can be seen that the parameter x_d controls the damage location and the damage coefficients in Eqn. (5-21) and Eqn. (5-22) control the damage severity. The adopted damage coefficients for the notch and the breathing crack are listed in Table 5-1 and 5-2. Hence, the Lamb wave signals with different severities of damage at different locations can be collected by choosing the corresponding damage coefficients and adjusting the value of x_d. Here, the value of x_d varies from 0 to 400 mm with an interval of 0.1 mm, so a particular type of damage (the notch or the breathing crack) occurring in the same damage region can be collected with 1,000 sets of data. Thus, a total of 8,000 sets of Lamb wave signals are collected to represent 8 damage condition classes (i.e., 2 damaged types x 4 potential damage regions). In order to consider the noise effects, each set of data is also superimposed with white noise to a certain level of SNR. For the pristine condition state, 500 sets of data are collected separately from the two plates by adding a certain level of white noises, so the data for each condition case has equal size (i.e., 1,000 sets of data per each class). Therefore, a total of 9,000 sets of data are collected in the test.

5.4.3 Data Preparation

The raw form of the Lamb wave signal needs to be preprocessed since a pre-trained CNN model is adapted for transfer learning on damage pattern recognition, and the model has the image as stipulated input, while the Lamb wave signals are time-series data in nature. Thus, short-time Fourier transformation (STFT) is used to preprocess the raw data of the Lamb wave signal from time-series to images in the form of STFT spectrum. The STFT spectrum represents the time-series data in an image format with three coordinates: time (x-axis), frequency (y-axis), magnitude (color shades). To be consistent, a unique STFT process has been applied to all collected Lamb wave signals. The STFT spectrum image size is 224 pixels by 224 pixels.

The processed Lamb wave signals also need labeling to supervise the training of the CNN model. Since the damage location and damage types are known, the labeling process is simply to assign the numerical representation (e.g., Class 1 to 9) for each damage condition class as listed in Table 5-3.

5.4.3 Deep Learning Implementation

For a certain level white noises, a total of 9,000 STFT spectrum images can be obtained from the collected Lamb wave signals. The 9,000-image database is then randomly divided into a training and testing dataset. The ratio of training data size to testing data is consistently set as 80% to 20% for each damage condition class. Thus, a total of

7,200 images are used to train the deep learning model, which will be then tested for accuracy using an out-of-sample testing dataset with 1,800 images. The out-of-sample data refers to the holdout samples that have not been used during the training stage.

The training and validating of the deep CNN model are performed in a deep learning software library - the Caffe Framework (Jia et al., 2014), using a UNIX (i.e., Ubuntu 16.04 LTS) workstation, with accelerated Graphics Processing Units (GPU) implementation on a Nvidia GeForce Titan X GPU. The stochastic gradient descent with moment is chosen as the training algorithm to minimize the cost function, with the mini-batch size of 64 due to graphics memory limit and the moment coefficient as 0.9. The learning rate is set as 0.001 for the initial 300 iteration steps and the value will decay in a ratio of 0.5 every 300 steps in the after (e.g., for 301-600 iteration steps, the learning ratio is 0.001×0.5). The strategy that fine-tuning all layers are applied during the training. The total steps of training iteration are set to be large enough so that the testing accuracy in the learning curve converges to a stable value; in this study, the maximum training iteration is 900. As shown in Figures 5-16, 5-17 and 5-18, the learning curves consistently showing high testing accuracy along with relatively low training loss indicate the convergence of the training process.

5.4.4 Results and Discussions

5.4.4.1 Damage analysis and data screening

Prior to applying the deep learning based NDE procedure, manual data screening has been performed on the collected Lamb wave signals, so the advantages and limitations of the proposed method can be better understood.

In the analytical modeling, the interaction between the Lamb wave and the damage can be described in three aspects: (1) wave transmission, (2) mode conversion, and (3) higher harmonic effects. The damage coefficients described in Eqn. (5-21) and Eqn. (5-22) control the wave transmission and mode conversion effects by affecting the signal components in the Lamb wave. The parameter h in Eqn. (5-22) controls the higher harmonic effects. The damage location x_d controls when such effects would happen. Therefore, potential damage features should have a correlation with such effects for damage detection. By comparing the calibrated damage coefficients listed in Table 5-1 and 5-2, it is found that the major difference between the breathing crack and the notch is the higher harmonic effects, which can be used as a potential damage feature to distinguish the two damage types. However, finding a feature that can identify the damage location is felt to be more challenging.

Visual screening of the plots is also implemented. Typical plots for the cases using the excitation with center frequency (f_c) of 100kHz under SNR 15dB white noise level have been shown in Figure 5-10 and 5-11. In time domain or its raw time-series form, at first glance, it is very difficult to find visual features (or patterns) with

correlation to the damage locations. However, due to the higher frequency components in breathing crack damage mode, the differences between the two damage types can be easily identified based on the fluctuations from higher frequency components. Benefiting from such patterns in the time-frequency domain (i.e., the STFT spectrums), the difference between the two damage types becomes easier to identify. Additionally, for the notch damage, it is observed that the signal pattern in between the two wave packets varies along with damage location while it is still difficult to distinguish the cases where the damage is close to either the transmitter or the receiver. A similar pattern can be observed in the STFTs of the Lamb waves from the breathing crack damage case. With the aid of such observed image patterns, its corresponding time series pattern can also be identified that additional wave packet may be present in between the two wave packets because of the damage location. Similar observations are also made in the cases using excitation with higher center frequency (i.e., fc = 220kHz and fc = 300kHz) in Figures 5-12, 5-13, 5-14, and 5-15.

Based on the above analysis, it can be concluded that such vision-based approach can be utilized to correlate the Lamb wave signals to the damage information, including its occurrence, location, and severity (or type). However, proposing a mathematical model to represent such a relationship is rather difficult and time-consuming, since such a process usually requires trial-and-error of many times. The deep learning based damage pattern recognition is used to resolve this issue since the

identification of the hidden visual features correlated with the damage location can be automated after the training process.

5.4.4.2 Sensitivity to Center frequency

First, the proposed method is validated at a low noise level (25 dB SNR), but with different center frequencies for the excitation signal, in order to exam the sensitivity to different center frequencies. The results indicate that the testing accuracy of the trained CNN model would not be affected much by the selection of the center frequency. The testing accuracy in the cases of center frequency at 100kHz, 220 kHz, and 300 kHz are 98.5, 99.8%, and 99.8% respectively. Thus, the proposed method is found to be insensitive to the center frequency of the input. The learning curves in this section are shown in Figure 5-16.

5.4.4.3 Sensitivity to White Noises

Second, the proposed method is examined in a multi noise level (i.e., 15, 20, or 25 dB) environment, but the center frequency of the excitation signal is fixed, so the sensitivity to different levels of white noise can be examined. Since it has been found that the center frequency will not affect the performance of the CNN model in this application, the center frequency of the excitation signal is fixed at 100 kHz in the current study. The results indicate that the testing accuracy from the trained CNN model is also not affected by the adopted white noise level. The testing accuracy in the scenario of white

noise level at SNR 15, 20 or 25 dB is 98.4%, 98.3%, or 98.5% respectively. Thus, the proposed method is found to be insensitive to the white noise levels considered here. The learning curves in this section are shown in Figure 5-17.

5.4.4.4 Sensitivity to Damage Region Size

Next, the proposed method is further validated in the situation that requires more accurate results regarding the damage location. This demand can be resolved by refining the size of potential damage regions (or increasing the number of damage location bins). Initially, the number of such damage case bins is set as 4, meaning the range of the damage region is 100 mm. The number of bins is then increased to 8 or 16, which represents the range of damage region is reduced to 50 mm or 25 mm. In the meantime, the number of potential damage condition classes is also increased to 17 (=8+8+1) in the case of 8 bins and 33 (=16+16+1) in the case of 16 bins. It should be noted that a random guess for this scenario would yield an accuracy of around 6% (1/17) or 3% (1/33). During the tests, the size (9,000) of collected Lamb wave signals remains unchanged so as to the training (7,200) and testing (1,800) dataset, while the labeling for the data for has been changed accordingly. Since it has been found that both the center frequency and white noise level do not affect the performance in the prior study, the center frequency of the tone-burst signal is fixed at 100 kHz and the white noise level is locked at 25 dB of SNR in this section. The results indicate that the testing accuracy for the trained CNN model rarely drops with increased bin numbers,

also with the reduced data size per each class. The testing accuracy is 98.17% for 8 bins and 97.5% for 16 bins. The learning curves in this section are shown in Figure 5-18.

5.4.4.5 Damage Pattern Visualization

The proposed method is found to be very robust to the potential effects considered in the previous sections. In this section, the damage patterns recognized by the validated CNN model are visualized through the guided backpropagation method described in Section 5.2.2.5. The visualization can reveal the visual feature patterns used by the trained CNN model in prediction. Intuitive assessment of the visualized features can assist with double-checking the results of the trained deep learning model.

First, the visualizations of the CNN models that have been trained for the scenario of using the excitation signal with 100 kHz center frequency under SNR 15 dB white noise environment is investigated. The visualizations of Lamb waves under typical damage conditions are shown in Figures 5-19 and Figure 5-20. It can be found that the trained CNN model is searching for similar visual patterns found in the previous manual screening. To identify the damage location, the model focuses on the visual patterns in between the two wave packets. To distinguish the notch and the breathing crack damage, the higher frequency components are identified by the validated CNN model. More interestingly, the background noises have been automatically removed by CNN model.

Second, the visualizations of the CNN models that have been trained for the scenarios under different levels of white noises are investigated. The center frequency of the excitation signal has been controlled during the investigation. It has been found that the trained CNN models discovered the same visual features with minor differences for the same type of damage. For demonstration purpose, the results for the damage location (x_d=150 mm) are shown in Figures 5-21 and 5-22. It is also shown that the background noises have been gone in all cases.

Third, the visualizations of the CNN models that have been trained for the scenarios using different center frequencies are investigated. The white noise level has been controlled during the investigation. It has been found that for the same damage location but different center frequencies within the excitation signal, the visual features recognized by the validated CNN models are different. However, they are all highlighting the pattern changes between the two wave packets. For demonstration purpose, the results for the damage location (x_d=150 mm) are shown in Figures 5-22 and 5-22. This finding indicates that the CNN model is data-driven, and the identified visual features are highly dependent on the training data.

5.5 Concluding Remarks

This NDE study investigates a deep learning based nondestructive damage evaluation method for plate structures using guided Lamb wave signals. The damage evaluation

has been converted into an image classification task and solved by transfer learning a pre-trained deep CNN model. Lamb wave signals thus need to be transformed into the time-frequency spectrum images using the short-time Fourier transformation (STFT) algorithm. To locate the damage, the plate is first divided into a specified number of damage regions and the predicted damage will be discerned inside one particular damage region. The proposed NDE method is validated in a test setting with aluminum plates with notch or breathing crack damages. For training of the deep CNN model, the analytical model of Lamb wave interaction with damages is adopted to generate the required data in an efficient way. Parametric studies are performed to investigate the effects of considered factors, including the center frequency, the white noises, and the number of damage regions. The proposed method is found to be effective in its damage detection capability of reporting: (1) the occurrence, (2) the damage region if occurred, (3) the severity (or type) of the damage with an accuracy over 97.5%. The main findings are summarized as follows,

(I) The implementation procedure of integrating the deep learning algorithm with NDE method for plate structures using guided Lamb wave signals is validated and examined, including the data preprocessing and labeling. The Lamb wave signal is time series in nature and needs to be preprocessed into the STFT spectrum images. The damage severity and location information are used for labeling the images.

(II) The proposed method is found to be insensitive to white noises. Three level of white noises with an SNR of 15, 20, or 25 dB have been used to investigate the

effect, in which the center frequency of the excitation signal is fixed at 100kHz. The proposed method is validated with a testing accuracy of 98.43% 98.33% or 98.5% respectively.

(III) The proposed method is found to have little preference on the choice of center frequency within the excitation signal. Three different excitation signals with a center frequency of 100 kHz, 220 kHz, or 300 kHz have been used to investigate the effect, in which the white noise level is locked at 25 dB SNR. The proposed method is validated with a testing accuracy of 98.5% 99.8% or 99.8% respectively.

(IV) The damage evaluation results of the proposed method can be improved regarding the damage location with minor loss of performance. The number of potential damage regions (or bins) is increased to 8 or 16, so the range of the damage region is refined. The proposed method is validated with a testing accuracy of 98.17% in the case of 8 bins and 97.5% in the case of 16 bins.

(V) The saliency map based visualization technique can be used to visualize the relevant features recognized by the validated CNN model for the NDE task. By comparing the recognized visual features with manual screening, it is found that validated CNN model is searching for features consistent with human intuition, and this search process can be automated after training and validation process. Furthermore, its execution does not require a prior feature definition or engineering.

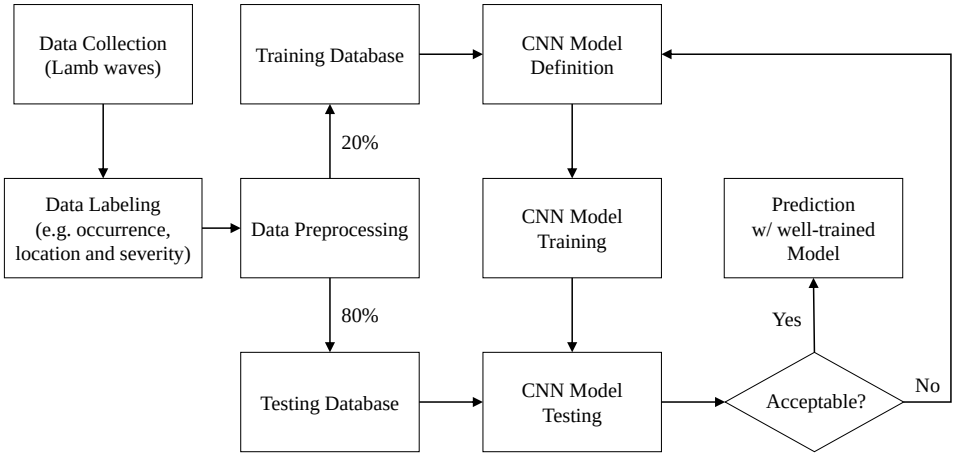


Figure 5-1. Implementing procedure of deep learning based nondestructive damage evaluation for plate structures using guided Lamb wave signals

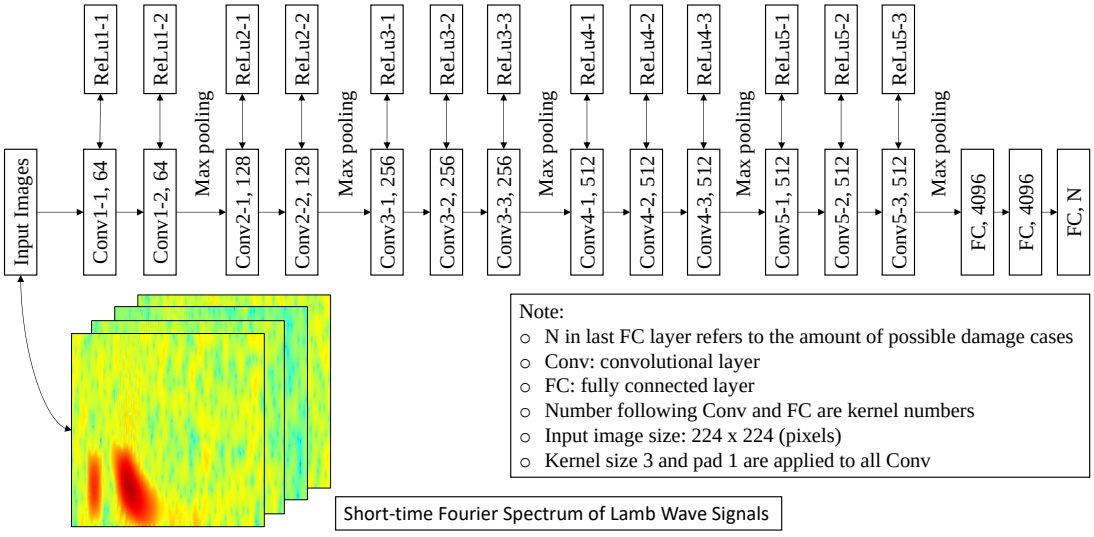


Figure 5-2. The adapted CNN model for nondestructive damage detection using Lamb wave signals based on the work by Simonyan and Zisserman (2014)

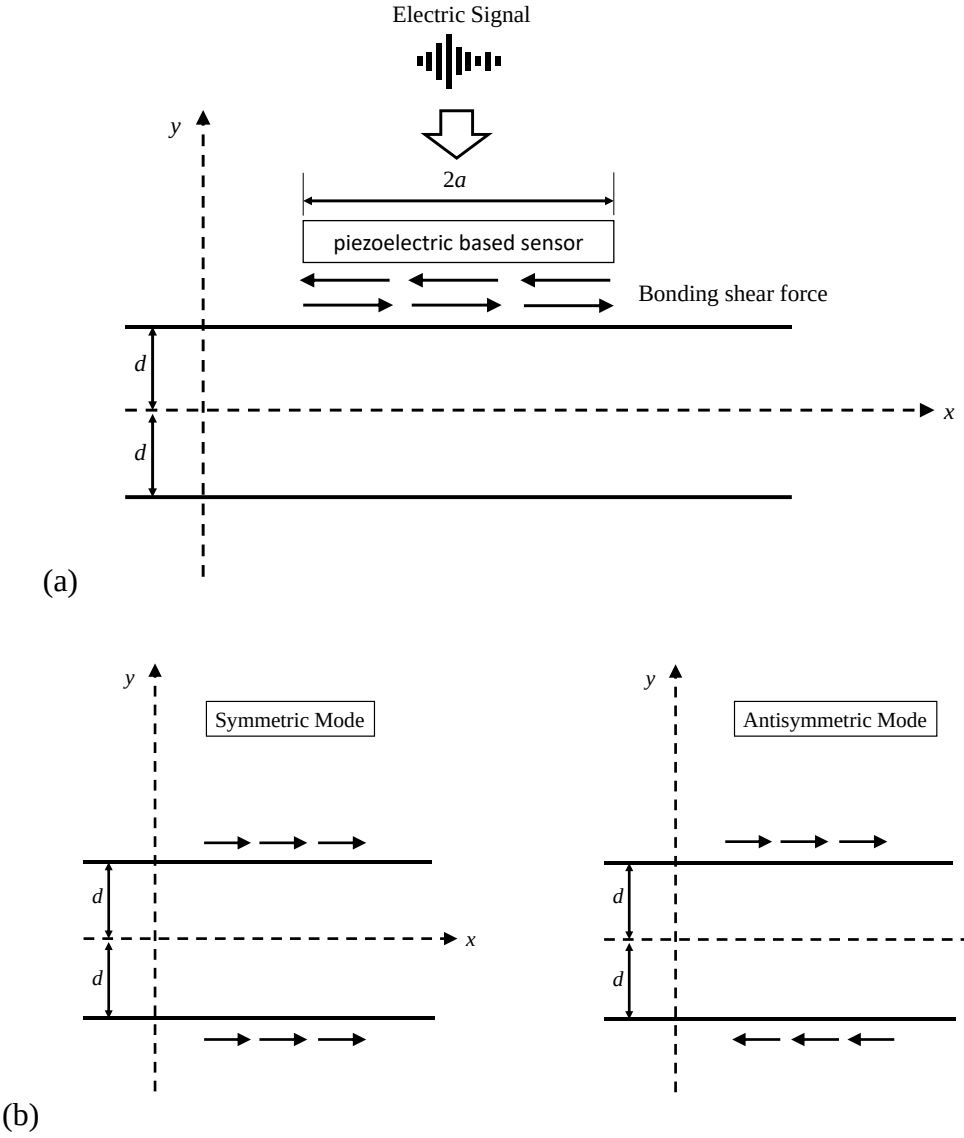


Figure 5-3. Lamb Wave excited by piezoelectric based sensor: (a) shear force interaction between sensor and plate surface; (b) decomposition of shear force into symmetric and antisymmetric mode

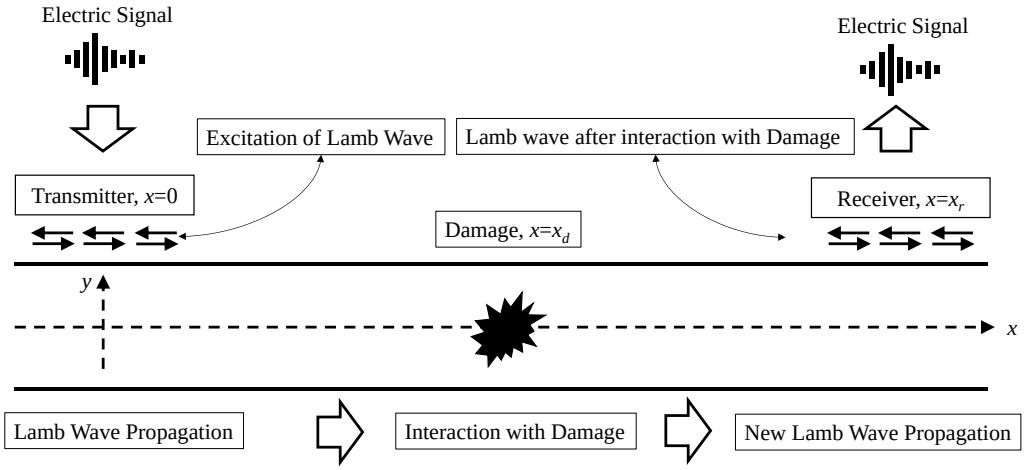


Figure 5-4. Experimental setup for damage detection using a pair of piezoelectric based sensors (a transmitter T- and a receiver) in a pitch-catch configuration.

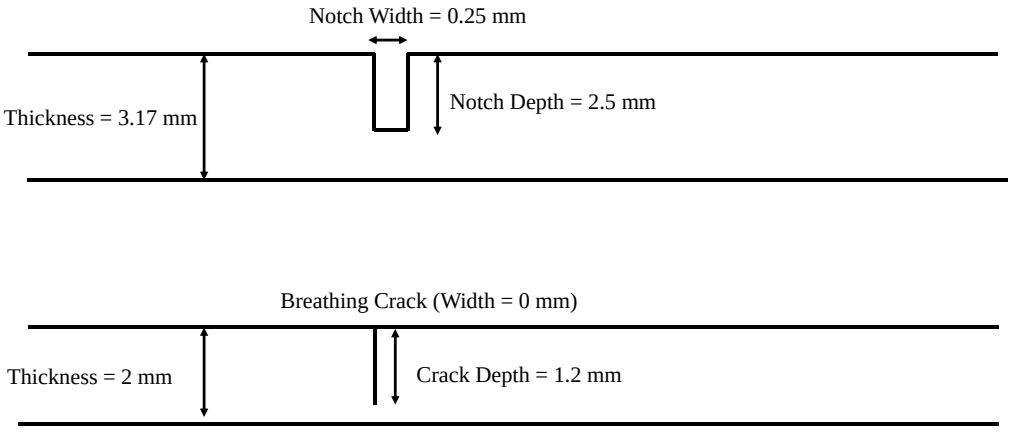


Figure 5-5. Damaged plates with notch (top) and breathing crack (bottom)

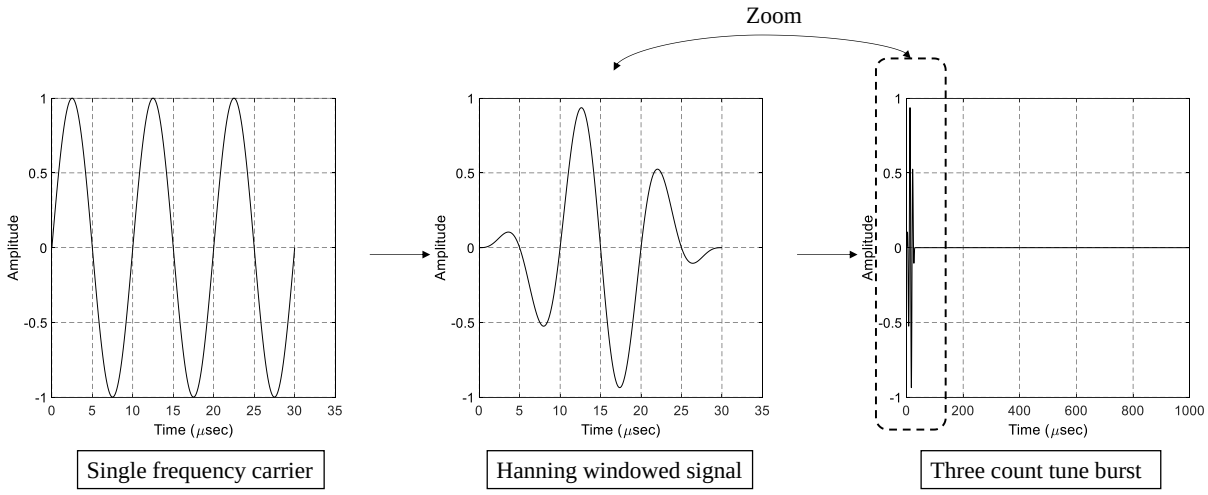
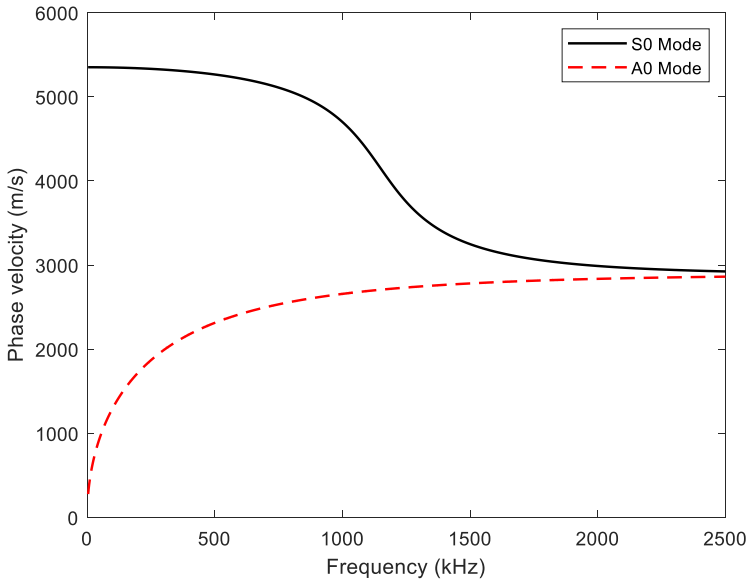
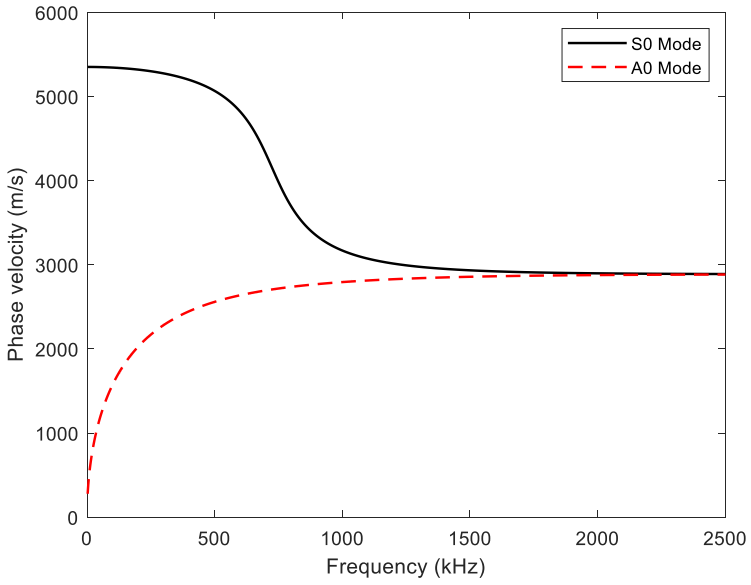


Figure 5-6. Three counts tone-burst input signal with central frequency of 100kHz

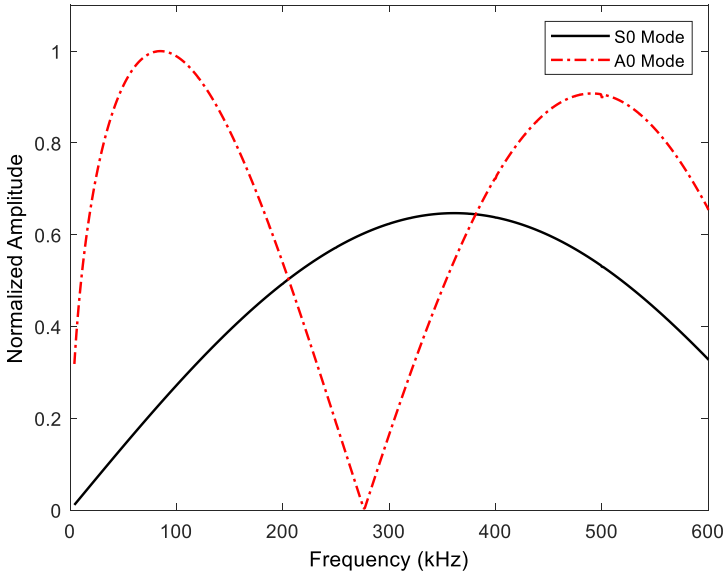


(a)

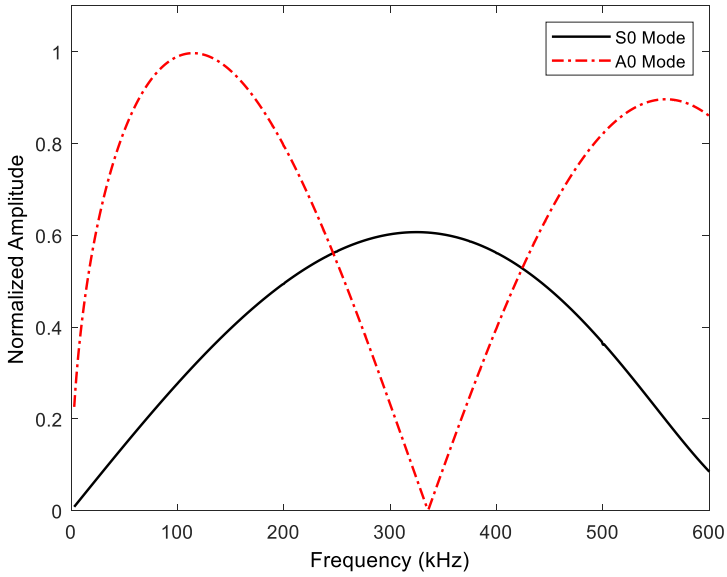


(b)

Figure 5-7. Dispersion curve of the aluminum plate with thickness (a) 2 mm (b) 3.17 mm

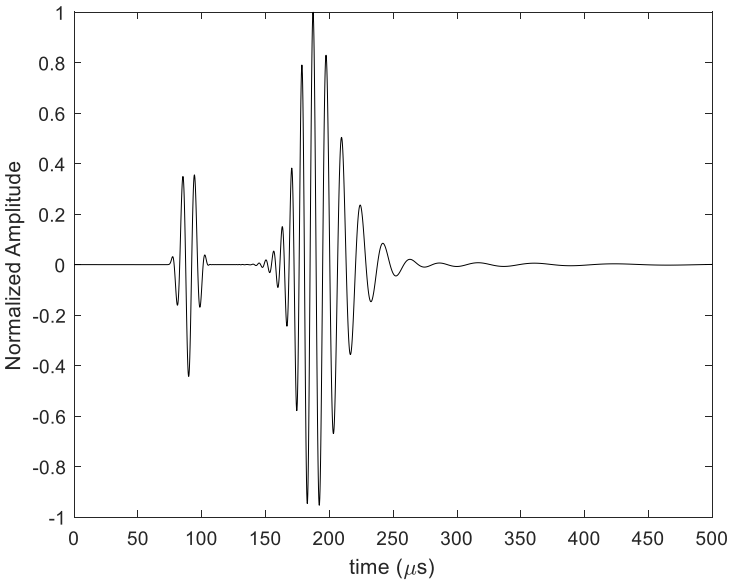


(a)

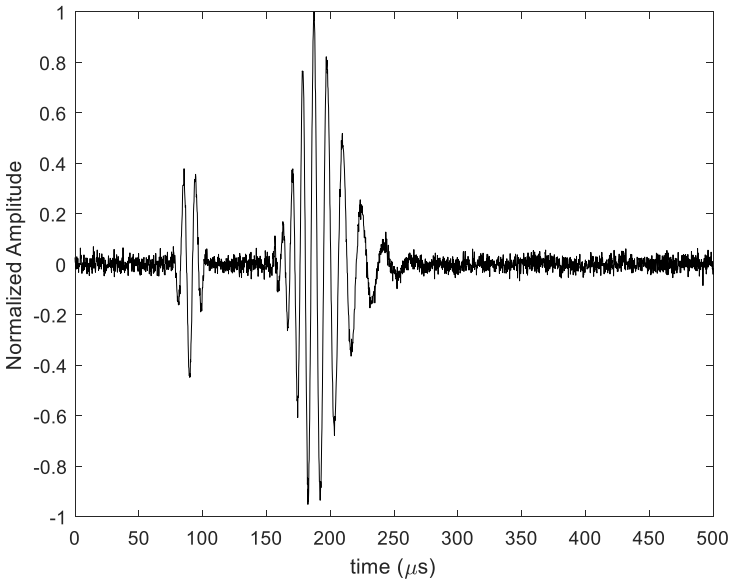


(b)

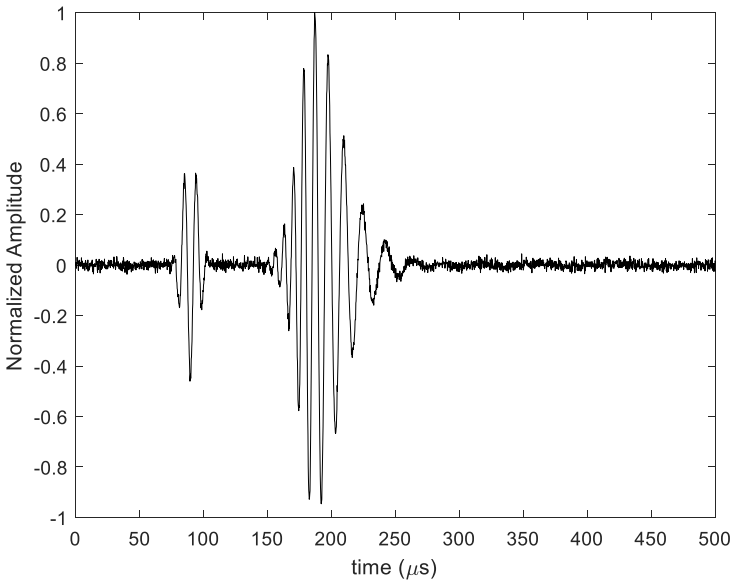
Figure 5-8. Tuning Curve of the aluminum plate with thickness (a) 2 mm (b) 3.17 mm


Normalized Amplitude
time (μs)

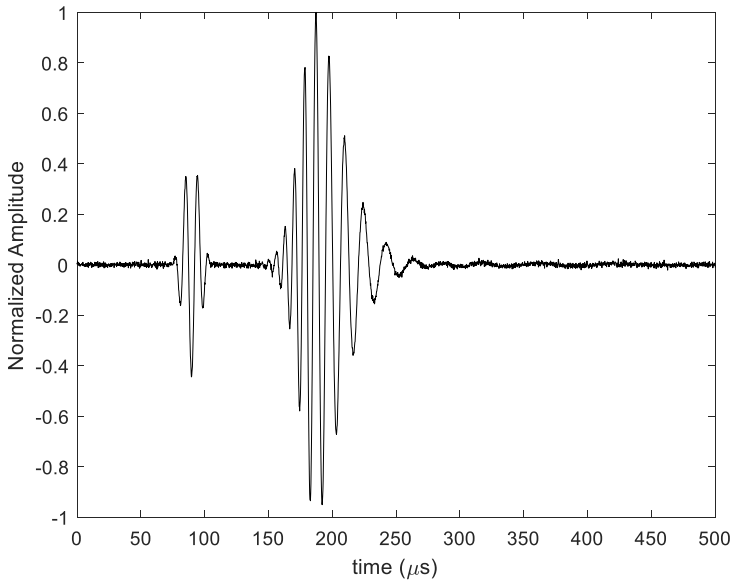
(a)


Normalized Amplitude
time (μs)

(b)

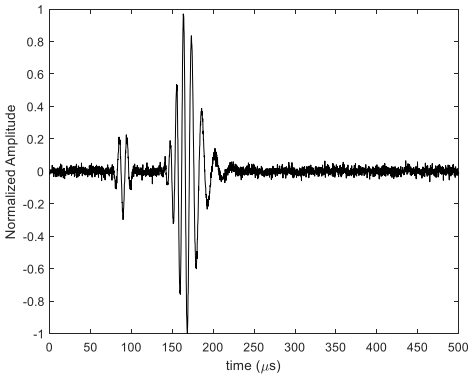


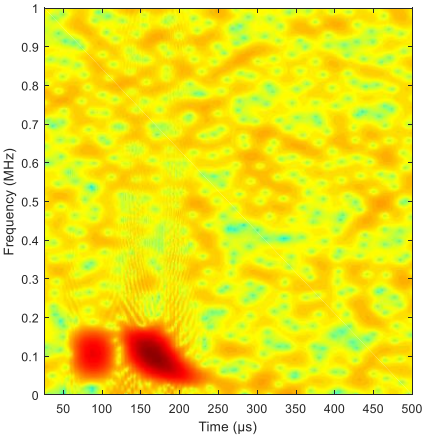
(c)



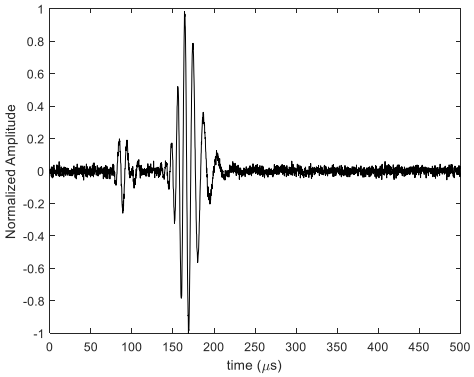
(d)

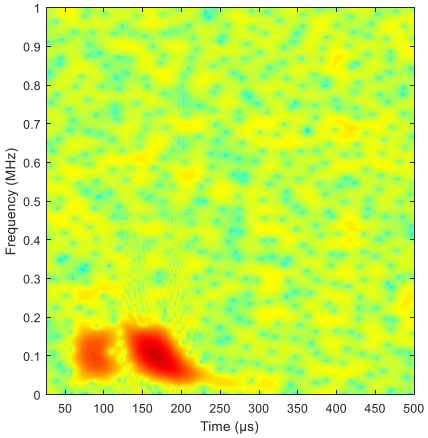
Figure 5-9. White noise effects on the collected Lamb wave signal from a pristine 2 mm-thick-plate: (a) noise free (b) SNR 15 dB (c) SNR 20 dB (d) SNR 25dB


Normalized Amplitude
time (μs)

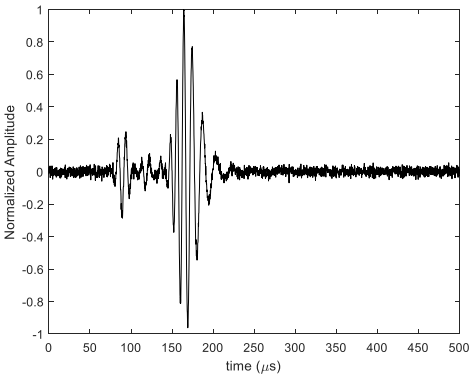

Frequency (MHz)
Time (μs)

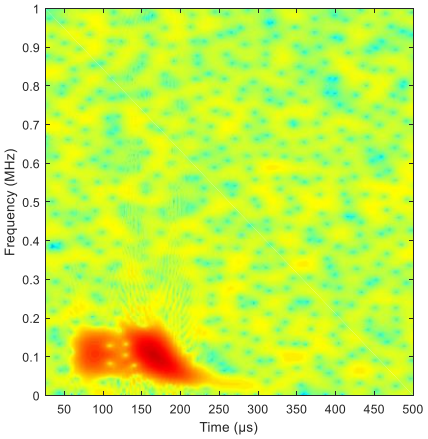
(a)


Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

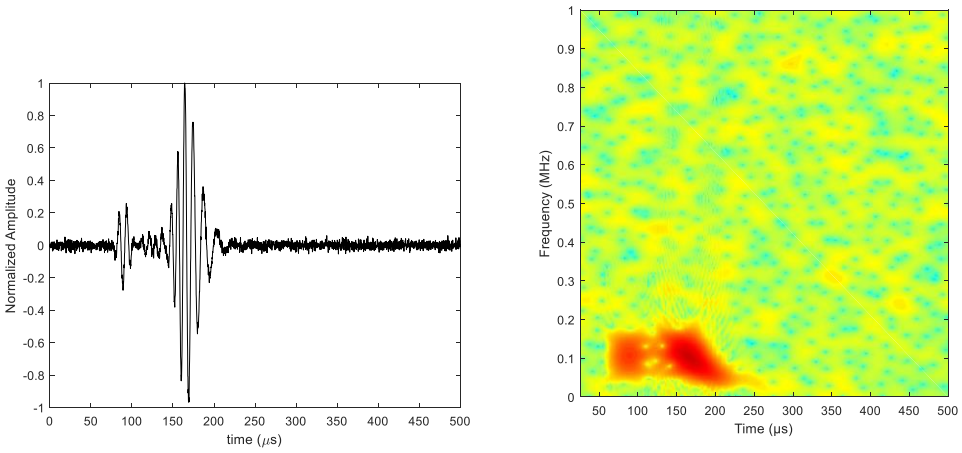
(b)


Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

(c)

(d)



(e)

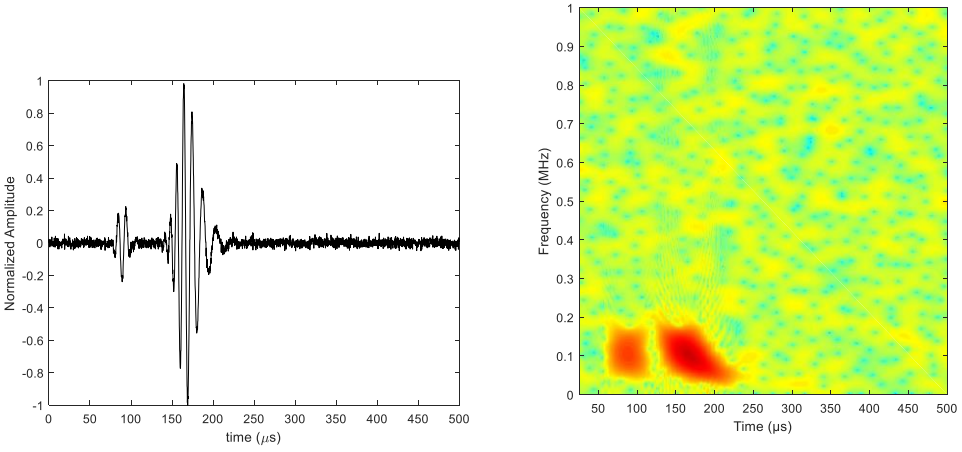
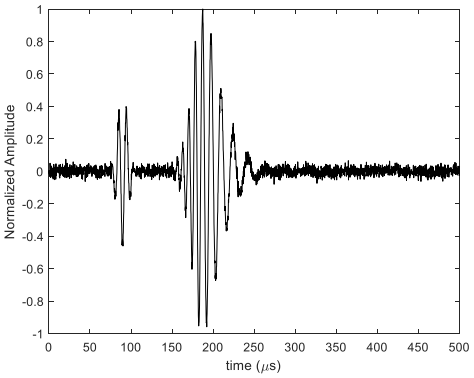
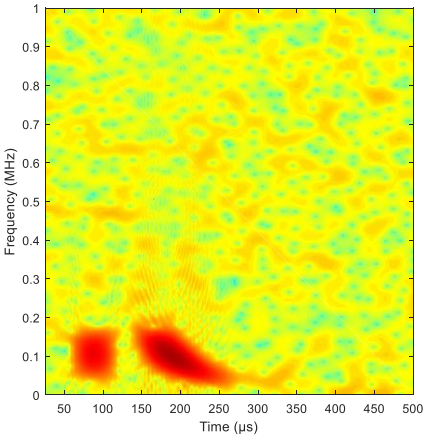
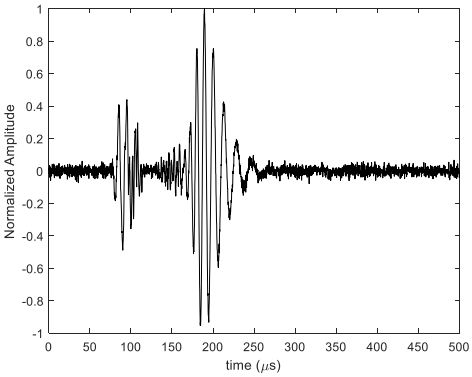


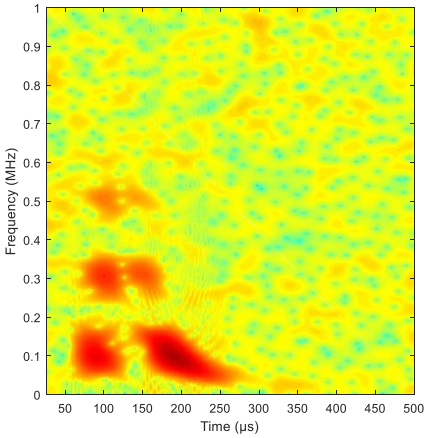
Figure 5-10. Typical Lamb wave signals in notch damage scenarios: (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm


Normalized Amplitude
time (μs)

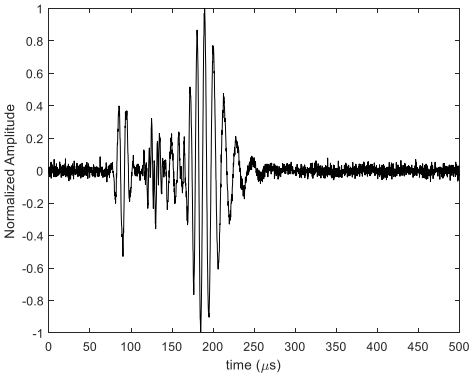

Frequency (MHz)
Time (μs)

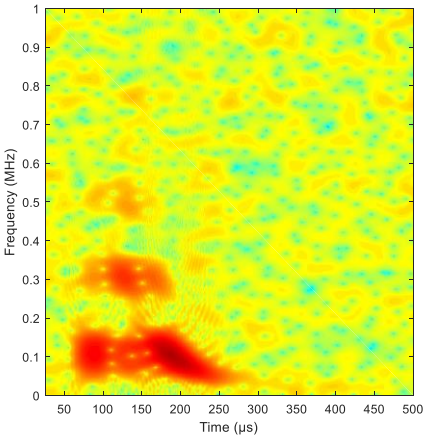
(a)


Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

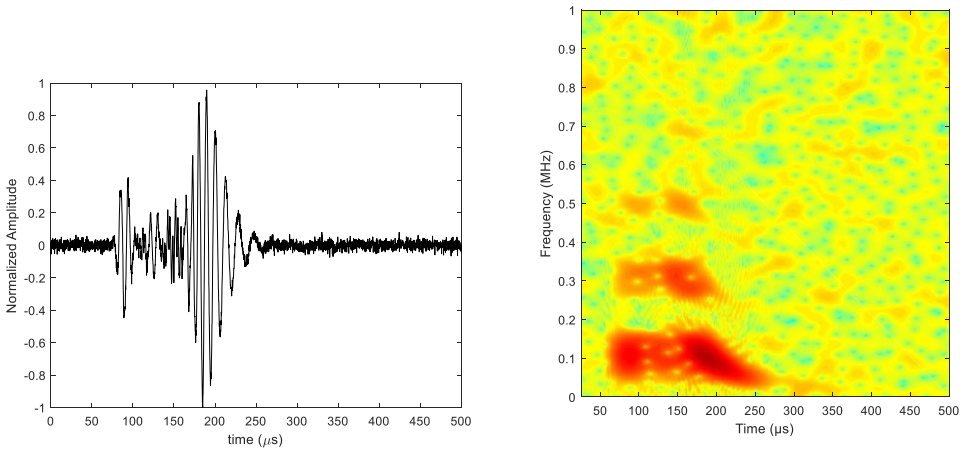
(b)


Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

(c)

(d)



(e)

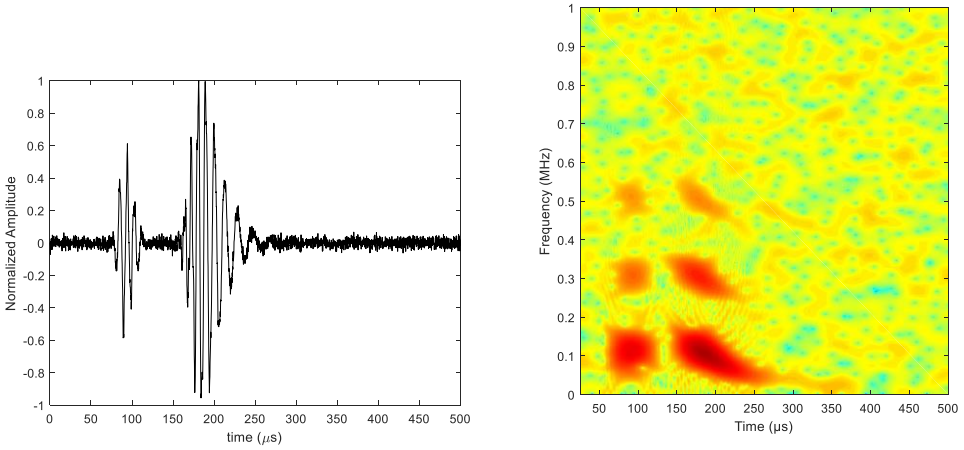
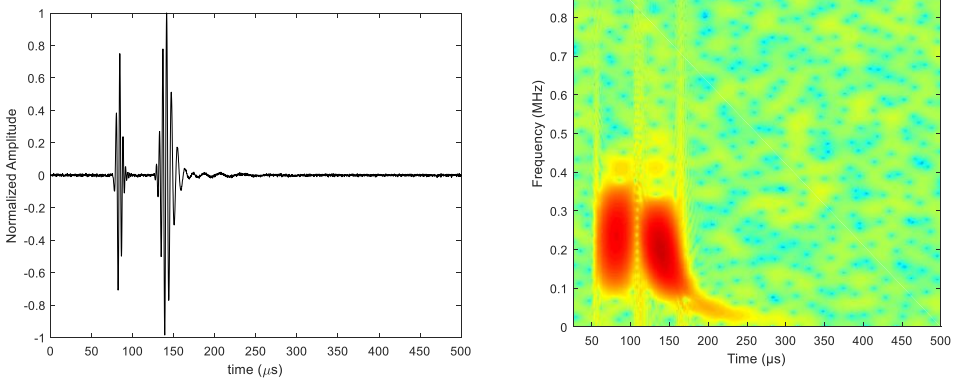
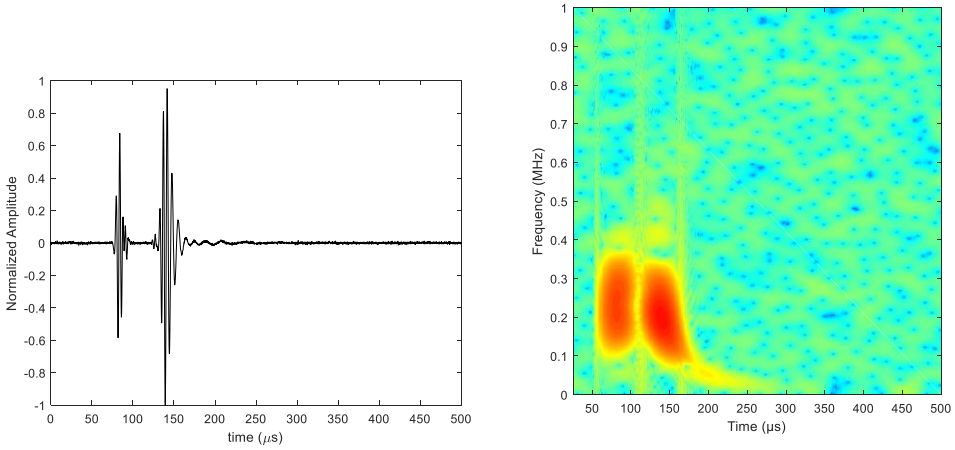


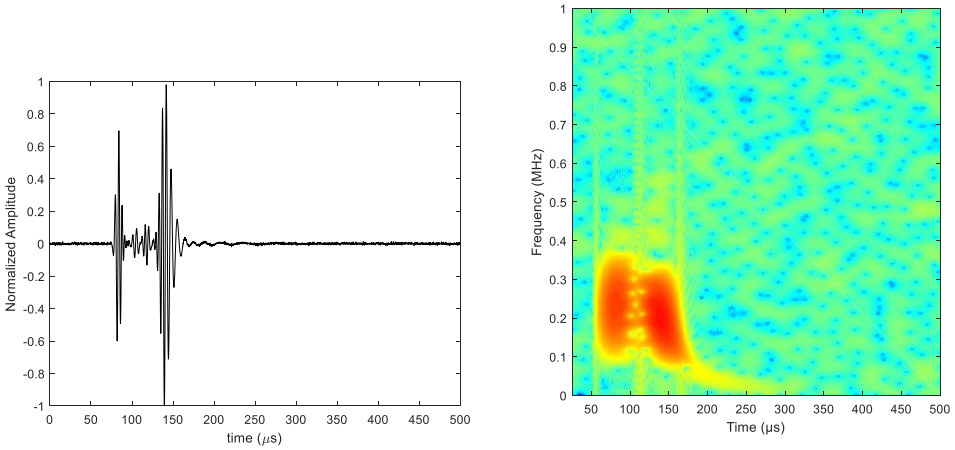
Figure 5-11. Typical Lamb wave signals in breathing crack damage scenarios (f_c=100 kHz, SNR=15dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm


Normalized Amplitude
time (μs)
Frequency (MHz)
Time (μs)

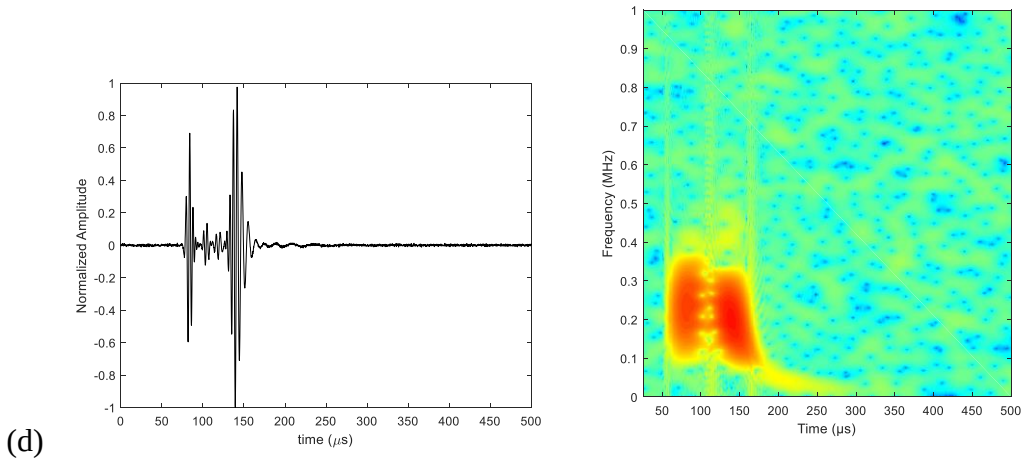
(a)


Normalized Amplitude
time (μs)
Frequency (MHz)
Time (μs)

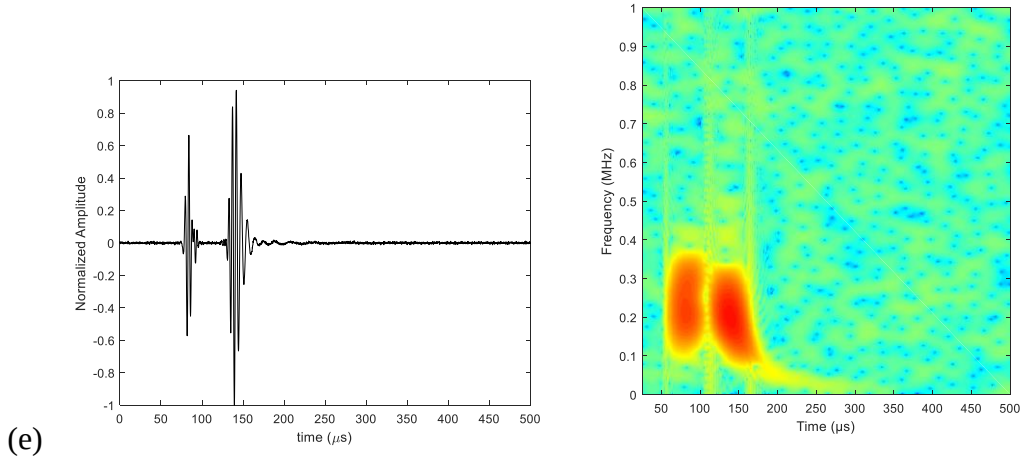
(b)


Normalized Amplitude
time (μs)
Frequency (MHz)
Time (μs)

(c)

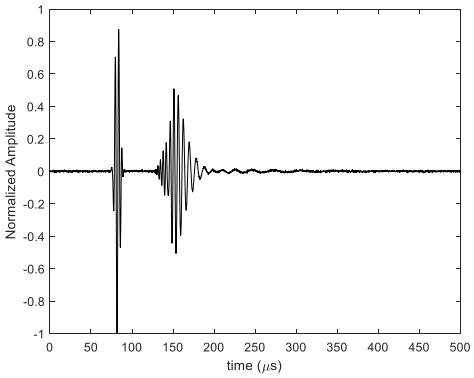


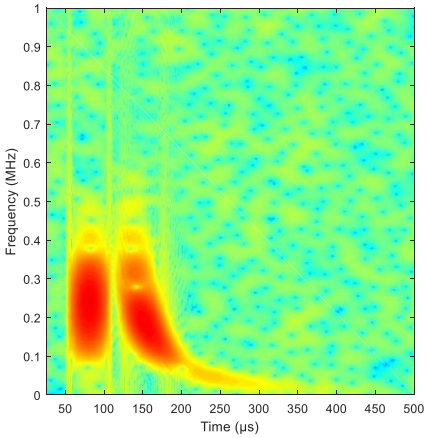
(d)



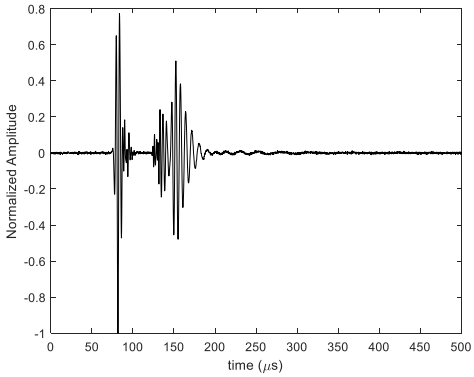
(e)

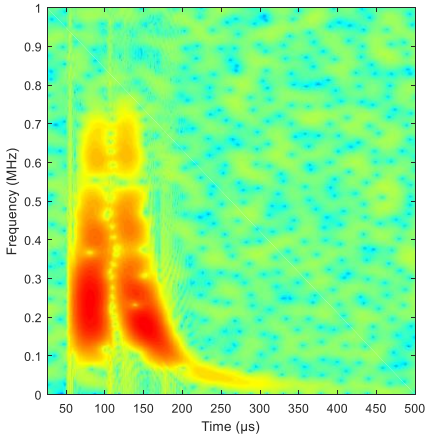
Figure 5-12. Typical Lamb wave signals in notch damage scenarios (f_c=220 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm


Normalized Amplitude
time (μs)

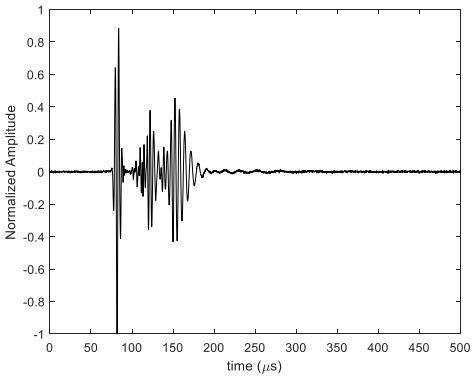

Frequency (MHz)
Time (μs)

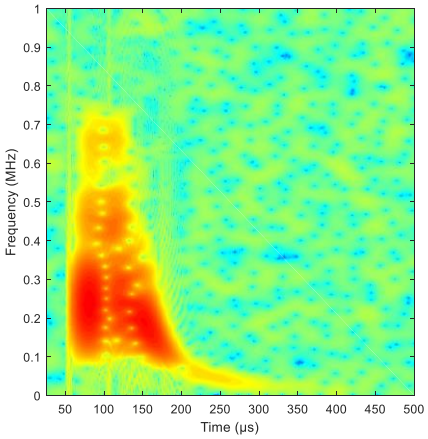
(a)


Normalized Amplitude
time (μs)

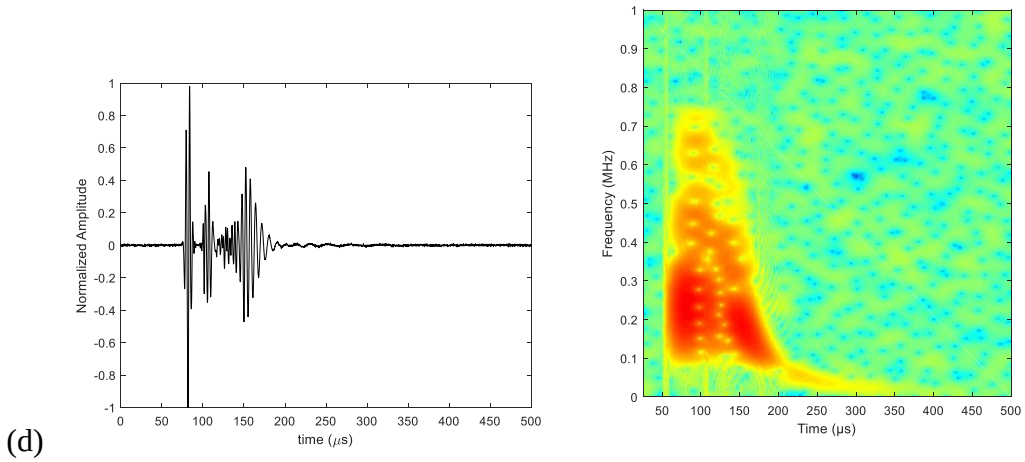

Frequency (MHz)
Time (μs)

(b)

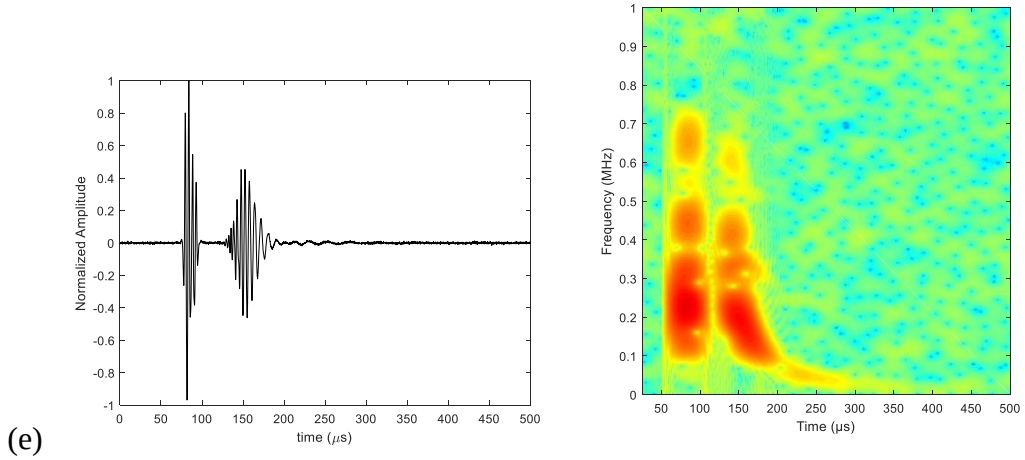

Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

(c)

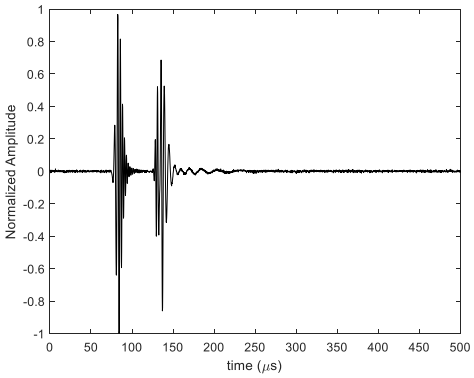


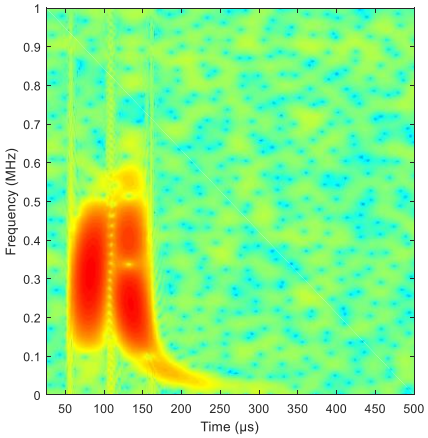
(d)



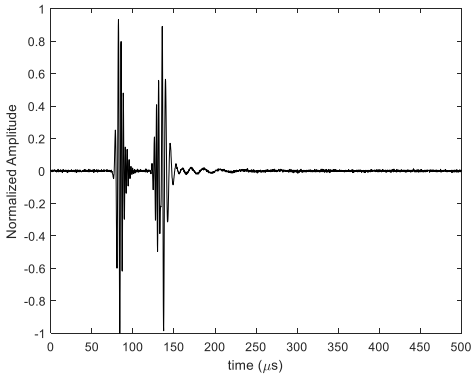
(e)

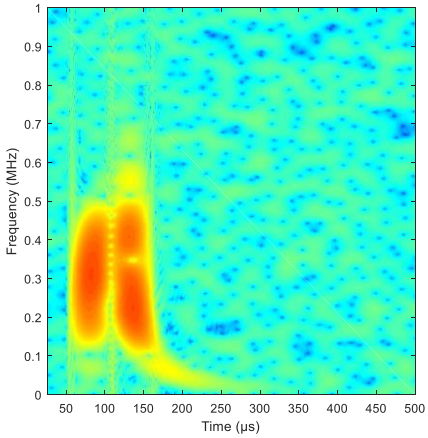
Figure 5-13. Typical Lamb wave signals in breathing crack damage scenarios (f_c=220 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm


Normalized Amplitude
time (μs)

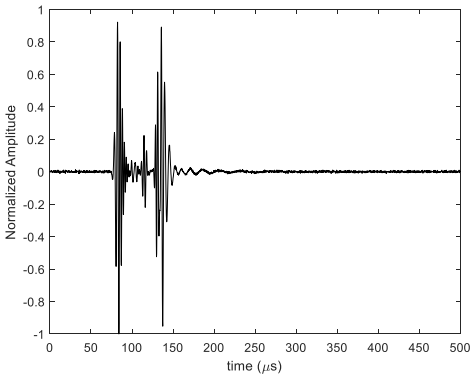

Frequency (MHz)
Time (μs)

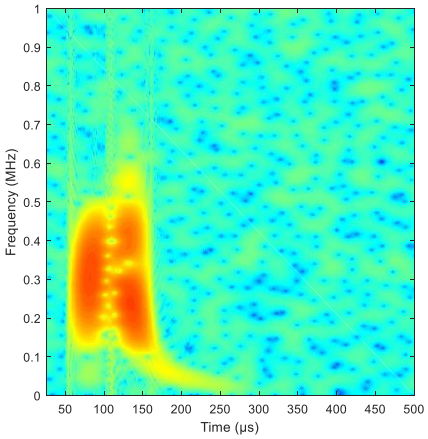
(a)


Normalized Amplitude
time (μs)

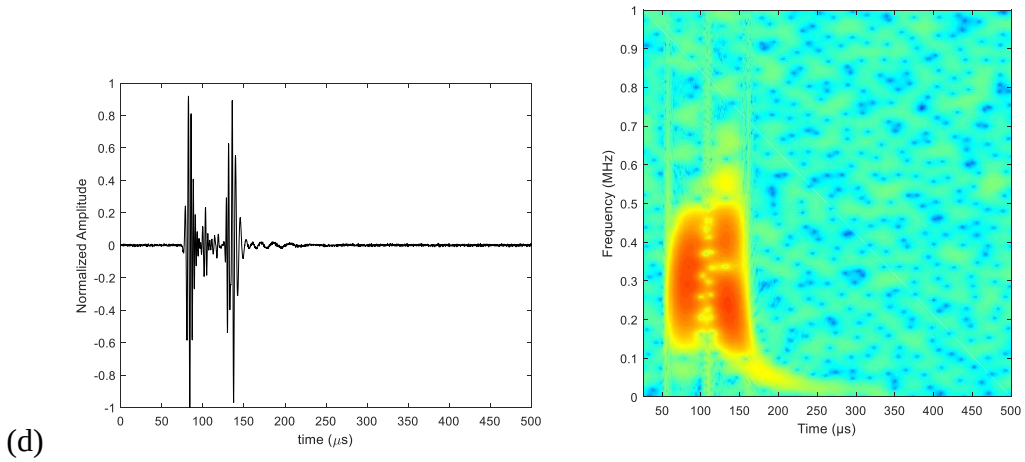

Frequency (MHz)
Time (μs)

(b)

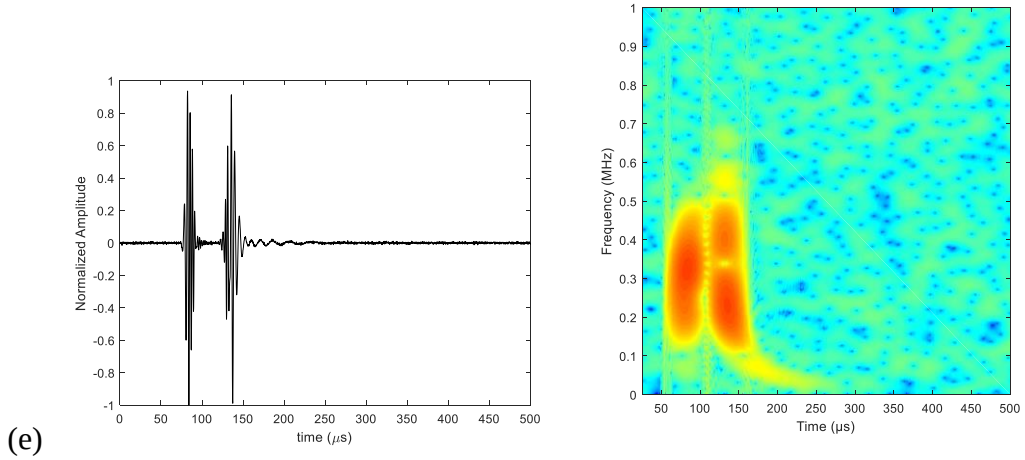

Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

(c)

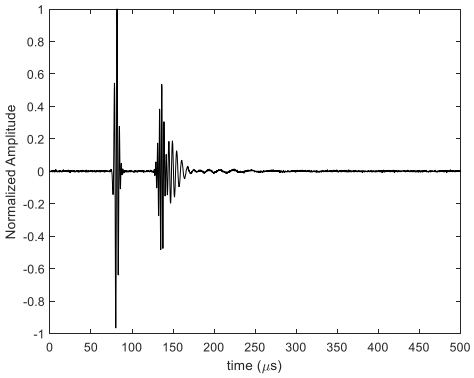


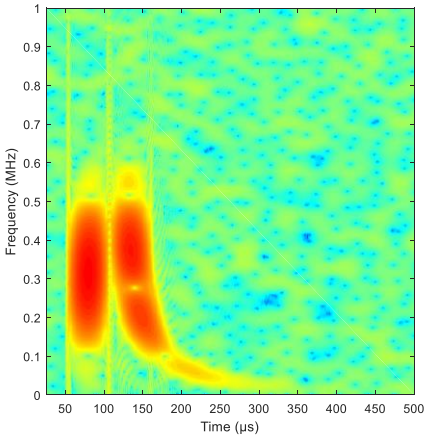
(d)



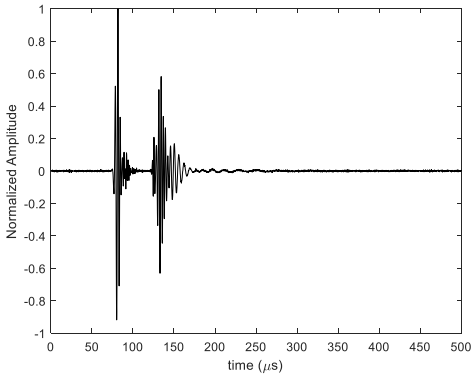
(e)

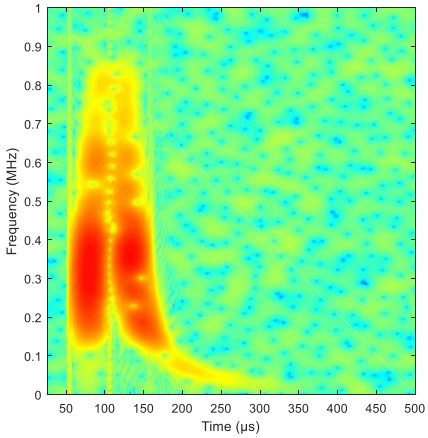
Figure 5-14. Typical Lamb wave signals in breathing crack damage scenarios (f_c=300 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm


Normalized Amplitude
time (μs)

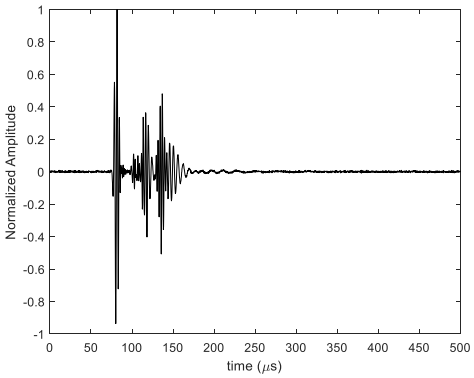

Frequency (MHz)
Time (μs)

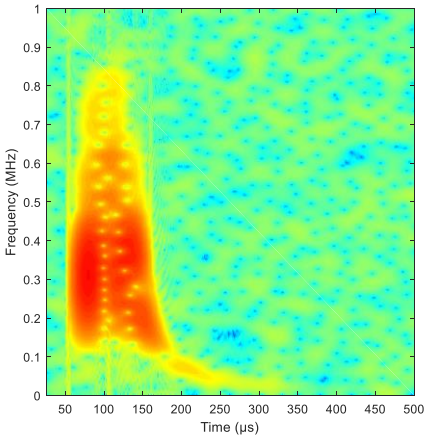
(a)


Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

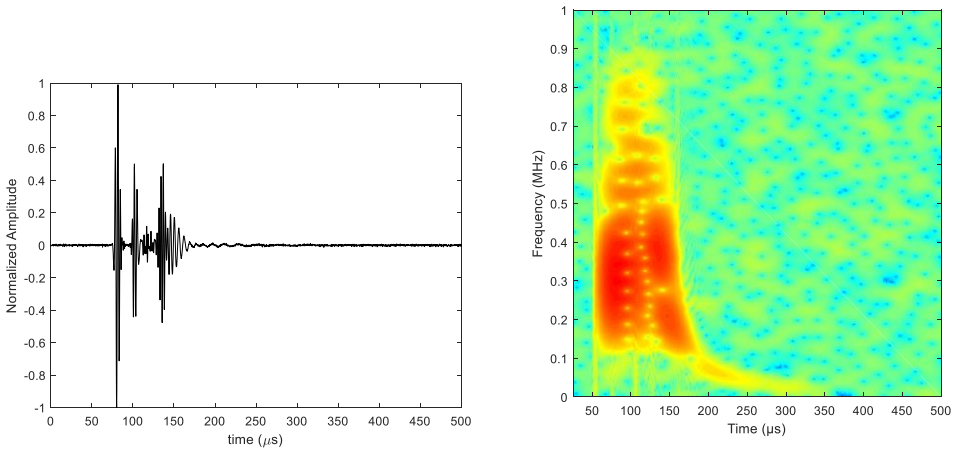
(b)


Normalized Amplitude
time (μs)


Frequency (MHz)
Time (μs)

(c)

(d)



(e)

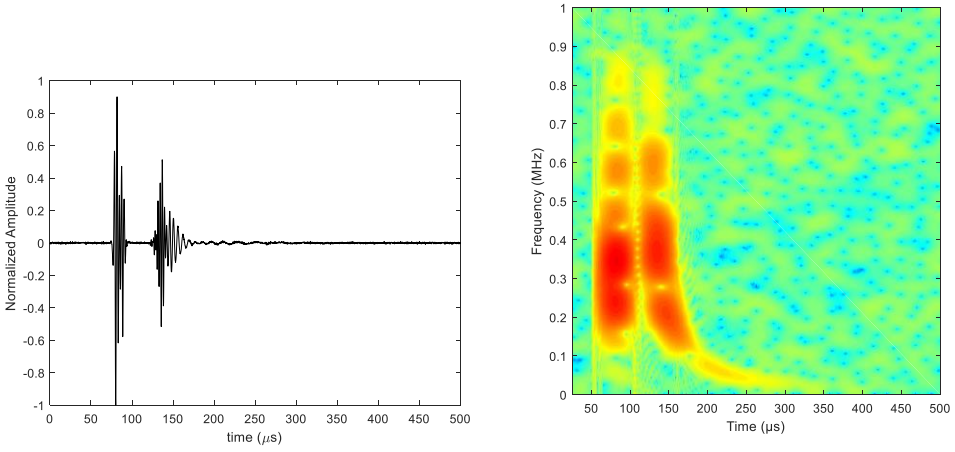
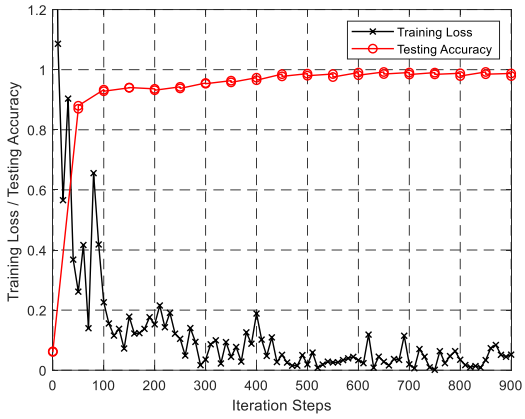
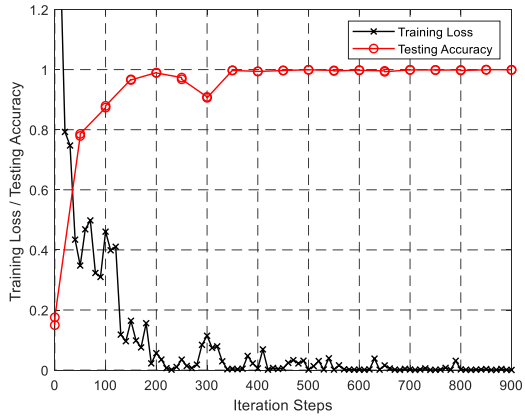


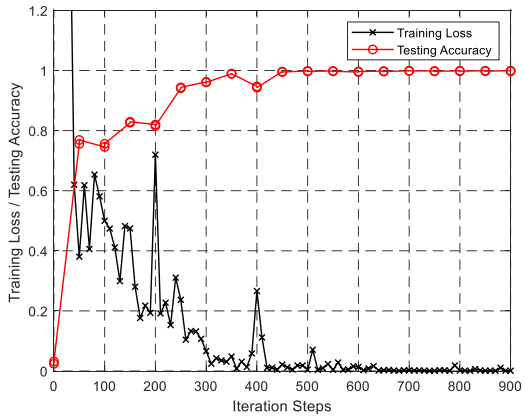
Figure 5-15. Typical Lamb wave signals in breathing crack damage scenarios (f_c=300 kHz, SNR=25dB): (a) pristine; (b) x_d = 50 mm; (c) x_d = 150 mm; (d) x_d = 250 mm; (e) x_d = 350 mm



(a)

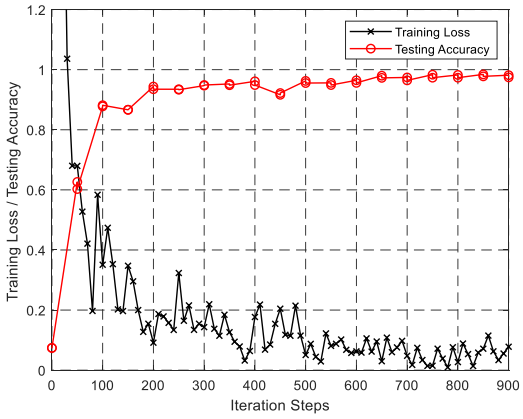


(b)

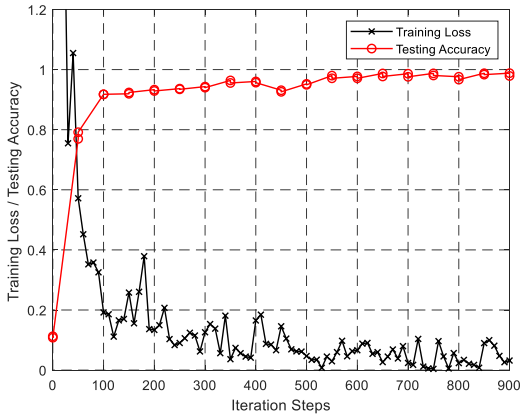


(c)

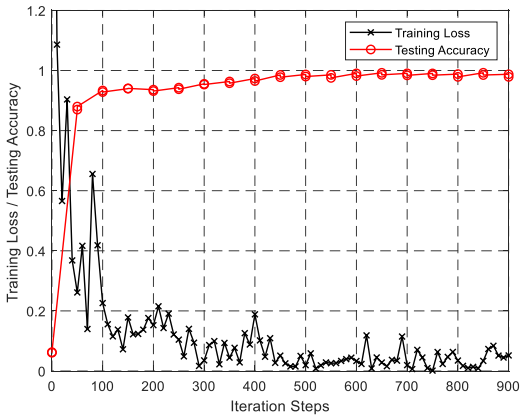
Figure 5-16. Learning curves for examining the effects from center frequencies: (a) 100kHz (b) 220 kHz (c) 300kHz



(a)

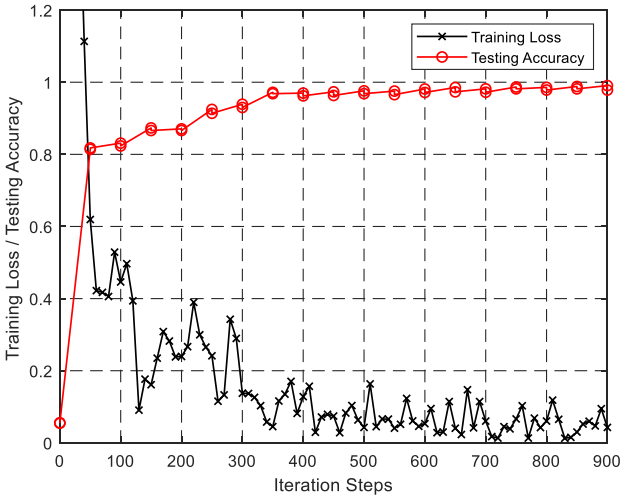


(b)

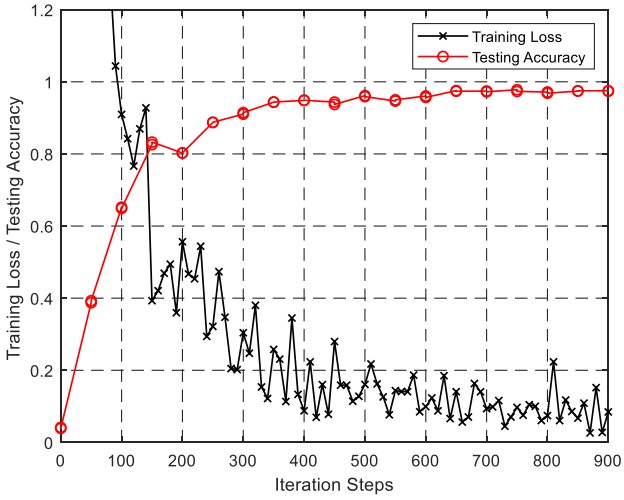


(c)

Figure 5-17. Learning curves for examining the effects from white noises (a) SNR 15dB (b) SNR 20 dB (3) SNR 25 dB



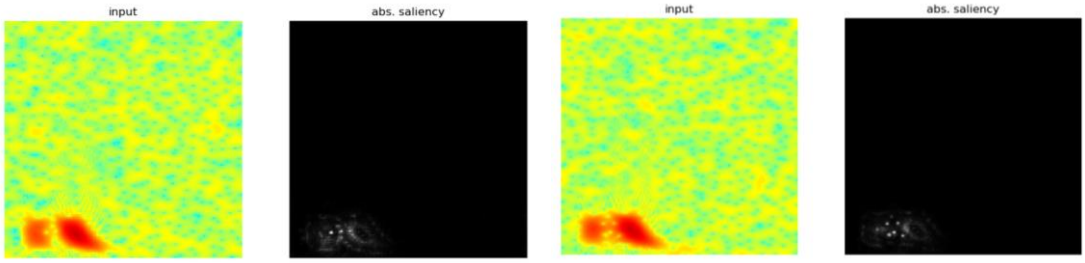
(a)



(b)

Figure 5-18. Learning curves for examining the effects from the number of damage bins (a) 8 bins (b) 16 bins

(a) x_d = 50 mm (b) x_d = 150 mm



(c) x_d = 250 mm (d) x_d = 350 mm

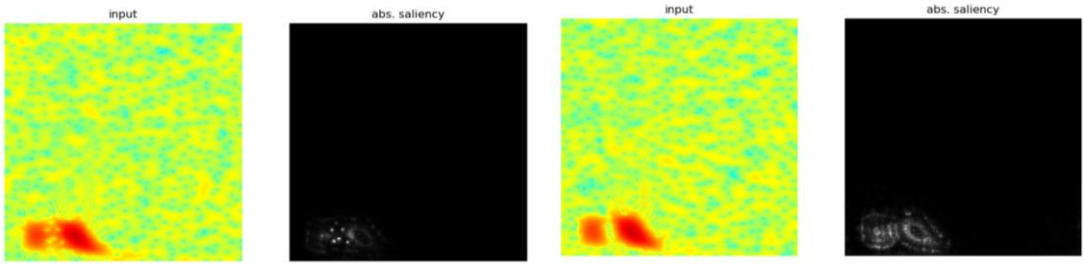
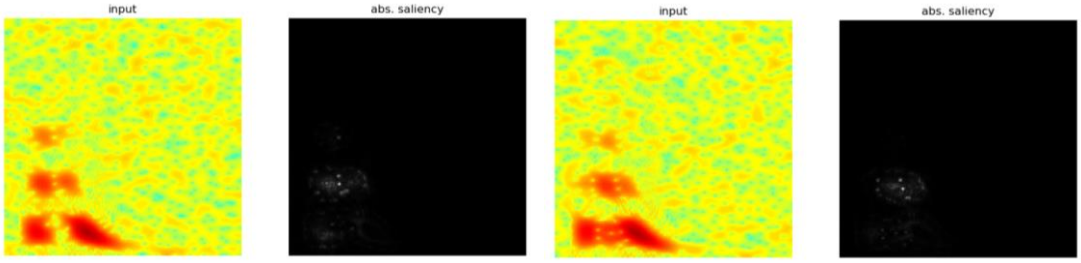


Figure 5-19. Visual features recognized by the trained CNN model for the notch damage scenarios (fc=100kHz, SNR=15dB)

(a) x_d = 50 mm (b) x_d = 150 mm



(c) x_d = 250 mm (d) x_d = 350 mm

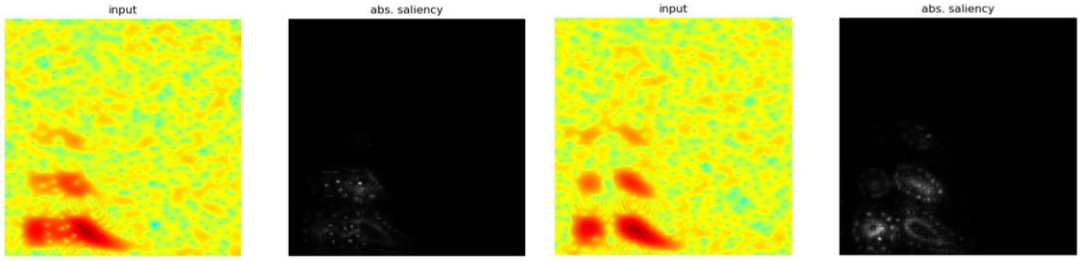
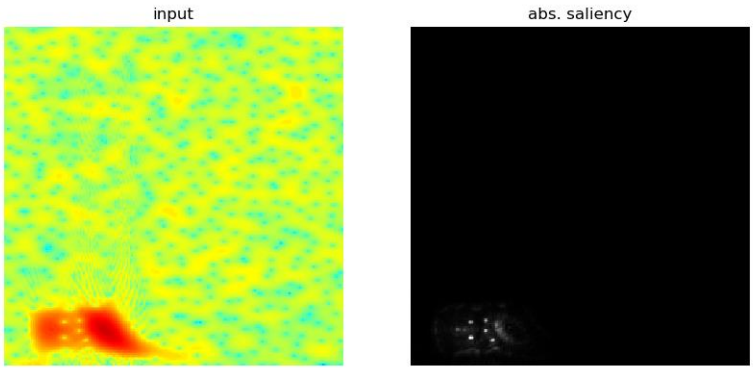
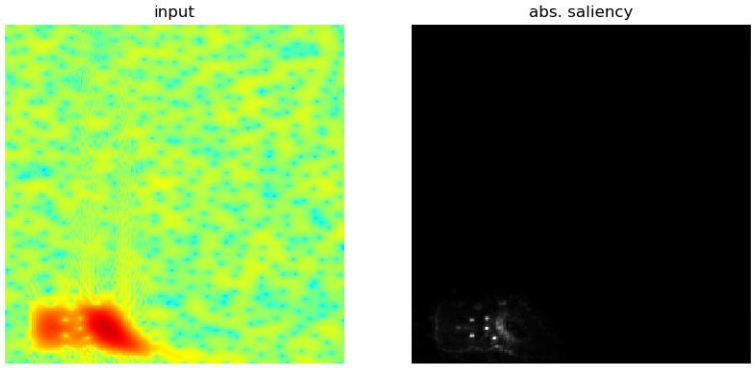


Figure 5-20. Visual features recognized by the trained CNN model for the breathing crack damage scenarios (fc=100kHz, SNR=15dB)

(a)



(b)



(c)

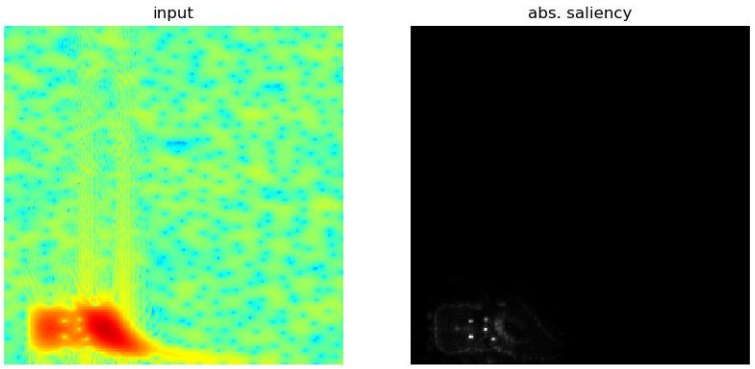
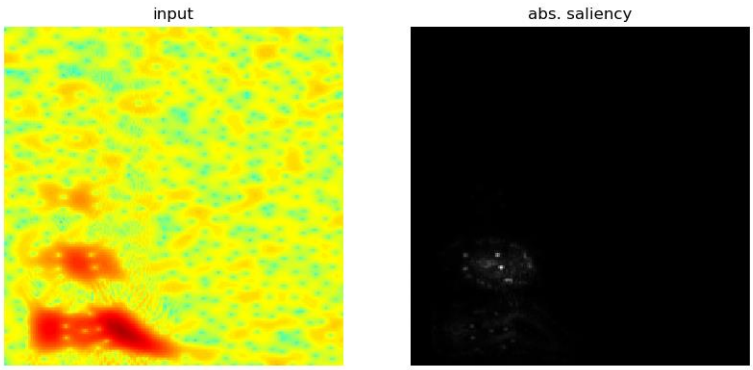
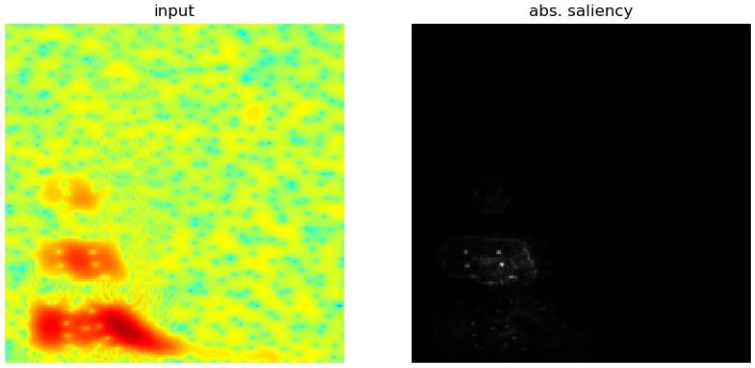


Figure 5-21. Visual features recognized by the trained CNN model for the notch damage scenarios (x_d =150mm, fc=100kHz): (a) SNR=15dB; (b) SNR=20dB; (c) SNR=25dB

(a)



(b)



(c)

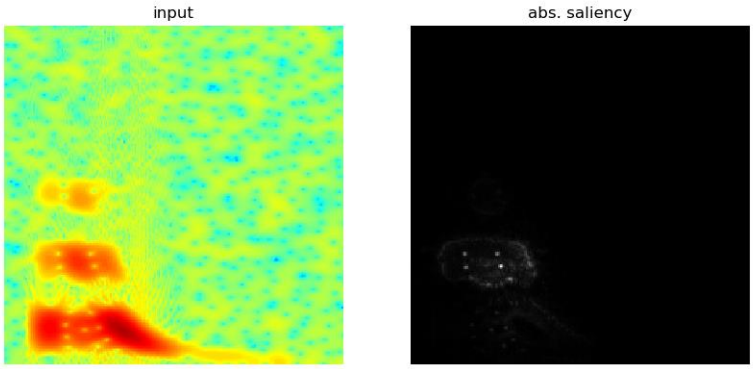
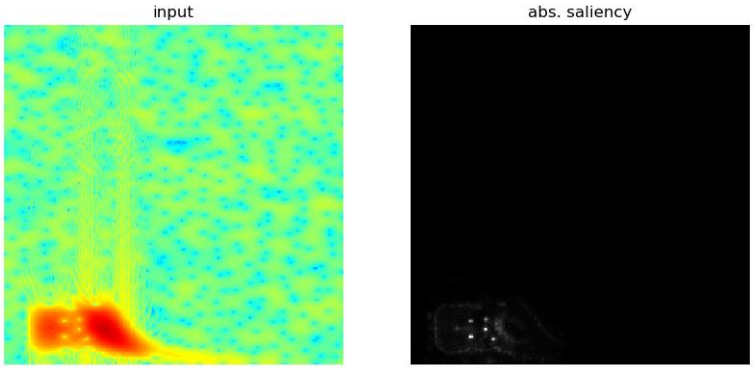
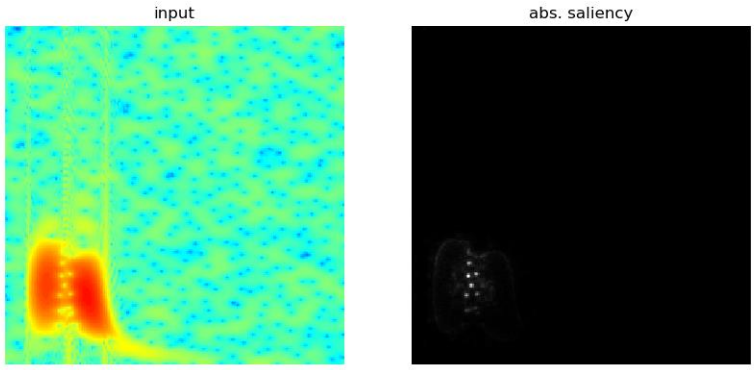


Figure 5-22. Visual features recognized by the trained CNN model for the breathing crack damage scenarios (x_d =150mm, fc=100kHz): (a) SNR=15dB; (b) SNR=20dB; (c) SNR=25dB

(a)



(b)



(c)

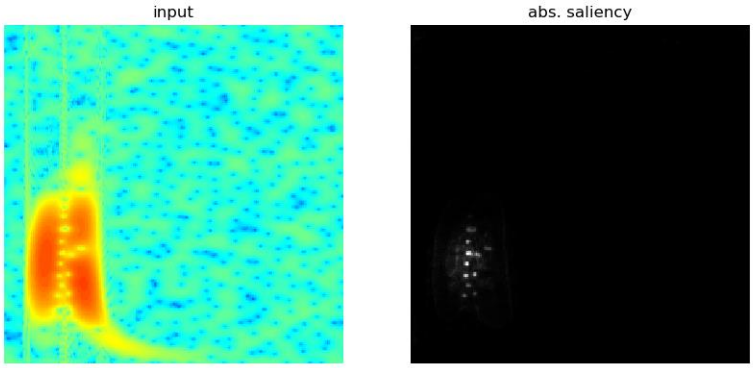
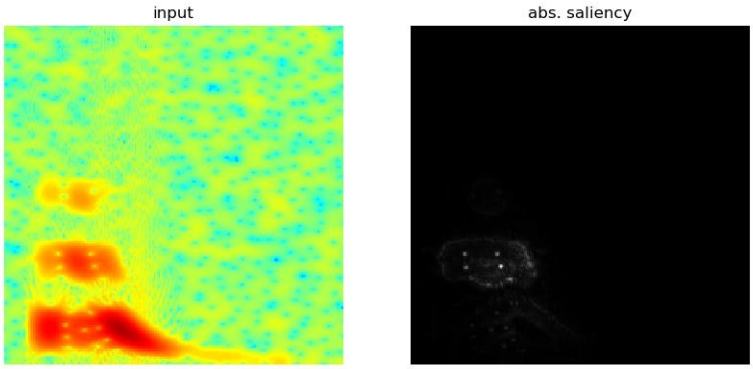
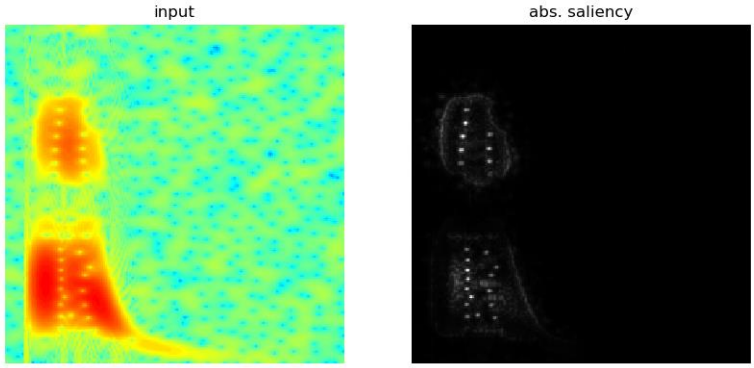


Figure 5-23. Visual features recognized by the trained CNN model for the notch damage scenarios (x_d =150mm, SNR=25dB): (a) fc=100kHz; (b) fc=220kHz; (c) fc=300kHz

(a)



(b)



(c)

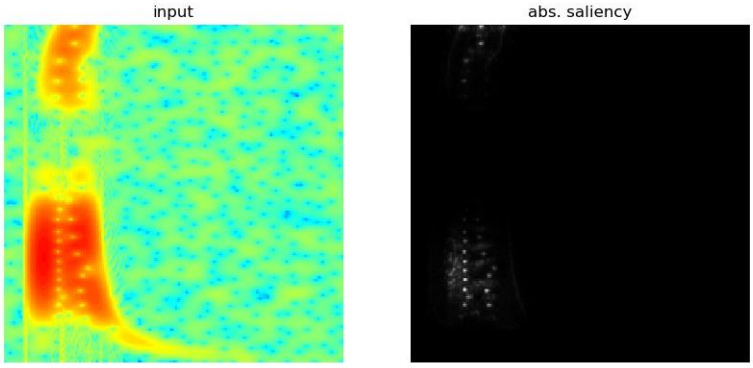


Figure 5-24. Visual features recognized by the trained CNN model for the breathing crack damage scenarios (x_d =150mm, SNR=25dB): (a) fc=100kHz; (b) fc=220kHz; (c) fc=300kHz

Table 5-1. Damage parameters for a notch type of damage

Magnitude Coefficients	C_{SST}	C_{SAT}	C_{AAT}	C_{AST}
	0.55	0.11	0.8	0.06
Phase Angle Coefficients (°)	φ_{SST}	φ_{SAT}	φ_{AAT}	φ_{AST}
	-30	30	0	30

Table 5-2. Damage parameters for breathing crack

h=0	Magnitude Coefficients	C^0_{SST}	C^0_{SAT}	C^0_{AAT}	C^0_{AST}
		0.900	0.420	0.820	0.100
	Phase Angle Coefficients (°)	φ^0_{SST}	φ^0_{SAT}	φ^0_{AAT}	φ^0_{AST}
		0	100	-35	90
h=1	Magnitude Coefficients	C^1_{SST}	C^1_{SAT}	C^1_{AAT}	C^1_{AST}
		0.082	0.100	0.050	0.110
	Phase Angle Coefficients (°)	φ^1_{SST}	φ^1_{SAT}	φ^1_{AAT}	φ^1_{AST}
		0	0	120	90
h=2	Magnitude Coefficients	C^2_{SST}	C^2_{SAT}	C^2_{AAT}	C^2_{AST}
		0.032	0.038	0.005	0.025
	Phase Angle Coefficients (°)	φ^2_{SST}	φ^2_{SAT}	φ^2_{AAT}	φ^2_{AST}
		0	0	0	0

Table 5-3. Damage region based Labeling

Damage Region	Damage Location x_d
1	$0< x_d \leq 100$
2	$100< x_d \leq 200$
3	$200< x_d \leq 300$
4	$300< x_d \leq 400$

Chapter 6: Bridge Condition Rating Data Modeling

6.1 Introduction

This study is aimed at developing a deep learning based method using Convolutional Neural Network (CNN) to predict the future condition of the bridge components. The scope of the research is to enhance the decision-making process of road authorities by improving the reliability of the forecasts, which is achieved by training a high-performance CNN model with available condition rating data. A case study of Maryland State highway bridges is presented to demonstrate and validate the proposed method using historical data deposited in the National Bridge Inventory (NBI) database from the Year 1992 to 2017. This methodology has been applied to derive the CNN models for three different components of both concrete and steel bridges. The study results show that the proposed deep learning method is promising as a data-driven bridge deterioration modeling approach with improved accuracy and prediction capability. Discussion of the results is also included to offer insights into the proposed method.

6.2. Deep Learning based Bridge Condition Assessment

6.2.1 General Procedure

The proposed bridge condition deterioration modeling method utilizes a deep CNN model which is trained with historical condition rating data from a bridge inventory in order to predict future condition ratings of three primary highway bridge components (i.e., NBI bridge subsystems: bridge deck, superstructure, and substructure). The input to the model in this study is either the latest single rating or multiple inspection ratings for a selected bridge component and the output is defined as the predicted condition rating in the next inspection. Hence, the historical database is reorganized into data samples that each consist of one or multiple condition ratings records consistent with the prescribed record length, and these data samples are labeled with actual condition ratings reported in their corresponding next inspection. Since the historical data records are from real inspection events, the data is believed to reflect the actual bridge condition including the effect of natural deterioration and man-made interventions such as regular maintenance and repair. Henceforth, the CNN model is trained to model the evolution of bridge deterioration based on historical condition rating data and such generalization is optimized through an iteration process termed back-propagation. To improve the reliability of the generalization, it is necessary to use training samples that are representative of the actual bridge deterioration process. As a result, the historical records from the Year 1992 to 2017 of the highway bridges in the State of Maryland

deposited in the NBI database are retrieved and used as the training data for the developed CNN model.

The bridge condition deterioration modeling process is illustrated in Figure 6-1. First, a sub-inventory is extracted from the bridge inventory database based on a set of prescribed similarity criteria defined for best representation. Since the current prediction is made based on only a limited number of condition rating records, such input data does not explicitly contain those factors potentially affecting the bridge deterioration process such as geographical locations, surrounding environments, bridge sizes, construction materials, highway classes, average daily traffic and others listed in the NBI. Thus, the specificity of the output (i.e., the prediction) from the CNN model depends on the selected similarity criteria: the more restricted the criteria are, the more specific of the prediction will reflect on a group of bridges. However, in practice the higher specificity might result in only a small sub-inventory that can be retrieved from the same database. For example, there are fewer data records of concrete bridges located in the City of College Park, Maryland than those within the entire state of Maryland, while the prediction results can be more specific (i.e., concrete bridges in the City of College Park) using the CNN model trained with the municipal dataset rather than the state model. Therefore, a trade-off has to be negotiated between the specificity and the size of data available for model training. Second, the extracted subdivision from the bridge inventory is preprocessed into discrete data samples complying with the designated formats and being properly labeled, as illustrated in

Figure 6-2, which is described in more details in the next section. Next, the processed data samples are randomly distributed into the training and testing datasets for respectively optimizing and evaluating the CNN model. The validated CNN model is then ready for use by engineers for bridge deterioration modeling and future condition rating prediction.

6.2.2 Convolutional Neural Network

CNN is a special type of neural networks for processing data with a known grid-like topology, such as time-series data and image data. The name of the CNN indicates that the network employs a mathematical operation called convolution, and the CNN perform the convolution operation in at least one of its layers. Examples of the computational layer in a CNN model include the convolutional layer, pooling layer, rectified linear unit (ReLU) layer, and fully connected layer. Details regarding the mathematical operation of these layers can be found in the book by Goodfellow, et al. (2016). The design of the architecture is usually developed for the specific task through experiments guided by monitoring the validation error, such as the deep CNN models reported in the ILSVRC or ImageNet (Deng et al., 2009) for image classification, including AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan & Zisserman, 2014), and ResNet (He et al., 2016). The common architecture of a CNN model stacks a few convolution-ReLU layers, follows with a pooling layer, and repeats this pattern until it

merges to the fully connected layers. As a result, the structure of the CNN model considered in this study is designed based on this common scheme as illustrated in Figure 6-3, while a few modifications have been made: (1) 1x1 convolution (Lin, et al., 2013) is employed throughout all the convolutional layers in the model to increase the representation power of the neural network; (2) the pooling layer which is designed to reduce the spatial size of representations is discarded in the model to keep the simplicity of the structure. Springenberg et al. (2014) suggest that a homogeneous CNN solely consisting of convolutional layers can achieve the state-of-the-art performance without the need for max-pooling layers; (3) the batch normalization (Loffe & Szegedy, 2015) is embedded between the convolutional layer and the ReLU layer to accelerate the training of the CNN model.

6.2.3 Cost Function and Statistical Interpretation

The cost function is the objective function that defines the target of the training of the CNN model. In this study, the CNN model is trained to represent the bridge condition deterioration model which can be used to predict future condition scores based on a set of historical condition rating records. Since the condition score is a discrete integer number ranging from 0 to 9, such prediction can be considered as a classification problem with each class representing one condition score. For classification purpose, the SoftMax with cross-entropy is most often used as the cost function. The SoftMax

function is thus used here to represent the probability distribution over C different classes (Goodfellow et al., 2016), as described in Eqn. (6-1),

$$p(\hat{y}|x^j,\theta) = \frac{1}{\sum_{i=1}^{C}\exp(z^i)}\begin{bmatrix}\exp(z^1)\\ \exp(z^2)\\ \vdots \\ \exp(z^C)\end{bmatrix} \tag{6-1}$$

where $p(\hat{y}|x^j,\theta)$ represents the probability distribution over C different classes when given one input x^j (j=1,2,…,N) among N training samples and model parameters θ; z^i (i=1,2,…,C) denotes the activations in the preceding layer (e.g., a fully connected layer). The cross-entropy loss is used to represent the dissimilarity between the empirical distribution (p_{data}) defined by the training data and the model distribution (p_{model}) estimated by the CNN model, and the degree of dissimilarity between the two measured by the Kullback–Leibler (KL) divergence (Goodfellow et al., 2016) is expressed in Eqn. (6-2) below,

$$D_{KL}(p_{data}\|p_{model}) = E_{x\sim p_{data}}[\log p_{data}(x) - \log p_{model}(x,\theta)] \tag{6-2}$$

And the cross-entropy loss is formulated as Eqn. (6-3),

$$L(x) = E_{x\sim p_{data}}[-\log p_{model}(x,\theta)] = -\frac{1}{N}\sum_{j=1}^{N} y_{x^j}\log p_{model}(x^j,\theta) \tag{6-3}$$

where y_{xj} is the true label of the data sample x^j and the $p_{model}(x^j)$ can be represented using Eqn. (6-1). The label y_{xj} is a one-hot encoding vector. For instance, for the data that belongs to the 3rd class in a classification problem with C equal to 5, the one-hot encoding vector label y is equal to [0,0,1,0,0]. Since the term related to p_{data} is fixed when the training data is given, the optimization (minimizing) of the Softmax with

cross-entropy loss corresponds to minimizing the KL divergence between the distributions.

To illustrate the training process of the CNN model, it is useful to describe the *Maximum Likelihood Estimation* (*MLE*). Let $p_{model}(x,\theta)$ denotes a parametric family of the probability distributions over the same sample space indexed by the model parameters θ. The MLE for the model parameters θ over a total of N samples x^j (j=1,2,…,N) is governed by Eqn. (6-4),

$$\begin{aligned} \theta_{MLE} &= \underset{\theta}{\operatorname{argmax}} \prod_{j=1}^{N} p_{model}(x^j,\theta) = \underset{\theta}{\operatorname{argmax}} \sum_{j=1}^{N} \log\left(p_{model}(x^j,\theta)\right) \\ &\Leftrightarrow \underset{\theta}{\operatorname{argmax}} \frac{1}{N} \sum_{j=1}^{N} y_{x^j} \log\left(p_{model}(x^j,\theta)\right) = \underset{\theta}{\operatorname{argmax}} E_{x\sim p_{data}}[\log p_{model}(x,\theta)] \end{aligned} \tag{6-4}$$

An equivalent optimization process is carried out such the MLE can be viewed as an attempt to make the model distribution match the empirical distribution, which decreases the KL divergence towards the zero. Such optimization can be achieved through the iteration process using the gradient based approach such as the gradient descent through backpropagation. Hence, the training process of the CNN model for a classification problem using the SoftMax with cross-entropy loss can be interpreted as the MLE for the model parameters θ to match the empirical distribution defined by the training dataset. Once the CNN model is trained, the predictions on test data is simply assigned to the class with the highest probability based on Eqn. (6-5) below.

$$\text{Prediction} = \operatorname{argmax}[p_{model}(x^j,\theta)] \tag{6-5}$$

6.2.4 *K*-fold Cross Validation

The *K*-fold cross validation is a statistical technique to assess the performance of a CNN model. The *K*-fold cross validation refers to a cross validation method that the original dataset is randomly partitioned into *K* equal-sized sub-datasets, among which the k^{th} sub-dataset is retained for validating the model while the remaining *K*-1 sub-dataset are used as training data, and such process is repeated K times (i.e., k=1,2,…,K) (Hastie et al., 2009). The value of K is usually empirically chosen. In this study, the K is set as 5 so that the ratio of testing data can remain at 20%. The results from the K-fold cross validation can be used to assess the performance of the trained CNN model. The averaged testing error (or accuracy) across the K trials can be used to estimate the prediction error (or accuracy) of the trained CNN model. The standard deviation of the results can be used to find the most parsimonious model whose error is no more than one standard error above the error of the best model, which refers the "one-standard-error" rule (Hastie et al., 2009).

6.3 Case Study on NBI Bridges in Maryland

6.3.1 Data Preparation

First, a sub-inventory is grouped from the NBI data depository according to the following criteria: (1) highway bridges in the State of Maryland; (2) the construction

material of the bridge is concrete; (3) the designated inspection frequency is 24 months. In practice, the criteria can be defined based on the need of bridge asset management, while it should be noted that there exists a trade-off between the stringency of the adopted data selection criteria and the amount of available data.

Second, the sub-inventory is preprocessed into data samples consistent with the designated data formats and with proper labeling. The input to the model in this study is either the latest single rating or multiple inspection ratings for a selected bridge component and the output is defined as the predicted condition rating in the next inspection. Hence, the data samples are built, as illustrated in Figure 6-2, using a sliding window with a length L (e.g., if L equals 5, the input data sample contains the condition rating scores from the last five inspections), and each data sample is labeled with the true condition score for the next inspection. Additionally, the condition score of 1 and 0 are combined and recorded as 1 in the data samples since both scores reflect the bridge is in failure condition. As a result, when L equals 5, a total of 5341, 5561, and 5612 sets of data sample are retrieved for bridge deck, superstructure, and substructure respectively, from the sub-inventory containing 1,634 concrete highway bridges with designated inspection frequency of 2 years in the State of Maryland. Details for the data set with different L size can be found in Table 6-1. It is seen that there exists a large amount of NA (not applicable) data in the NBI, which also limits the size of retrieved data samples.

6.3.2 Deep Learning Implementation

The training and validation of the developed CNN model are performed in MATLAB and Deep Learning Toolbox (2018), with accelerated Graphics Processing Units (GPU) implementation using a Nvidia GeForce Titan X GPU. The stochastic gradient descent with moment is chosen as the solution algorithm, with the mini-batch size of 2048 and the moment coefficient set as 0.9. A constant learning rate of 0.01 is adopted for the entire training process. The total number of training iterations is set to be large enough so that the learning curve converges to a stable value; in this study, the maximum training epoch is set as 200 (one training epoch means the model is non-repeatedly trained over all the training examples for once). As shown in Figure 6-4, the learning curves showing consistent high testing accuracy along with relatively low training loss indicate that the training of the model converges well.

6.3.3 Bridge Condition Assessment Results and Discussion

6.3.3.1 Overall Observations

The overall effectiveness of the proposed method in modeling concrete bridge condition deterioration can be seen from the validation results listed in Tables 6-2 to 6-4. These tables present the 5-fold cross-validation accuracy for the cases in which a series of CNN models trained with different input length. The results show that the proposed method has an averaged prediction accuracy varying from 82.4% to 89.7%,

and low standard deviation values from 0.3% to 1.5%. It is worth noting that the accuracy of a random draw would yield a prediction accuracy of 11.1% (one out of nine classifications).

6.3.3.2 Effect of CNN architecture

In this study, the main consideration of the CNN model architecture is its width of each layer and the depth of the network due to the adopted homogeneous design as described in Section 6.2.2. Three candidate models have been illustrated in Figure 6-2. It is seen that the depth of the network varies from 4 to 6 layers, and the width of each layer can be adjusted with the base kernel size *k*. Subsequently, the effect of CNN architecture on the prediction performance using the trained CNN model can be investigated through a series of parametric studies. The parametric study results for bridge deck, superstructure and substructure are listed in Tables 6-2 to 6-4 respectively. The comparison of the study results indicates that the effect of CNN architecture in terms of the width and depth on the performance of the trained model is minor. The mean testing accuracy variation is within 1.5% for the evaluation tasks with the same input record length.

6.3.3.3 Effect of the input record length

The effect of the input data length is also investigated through a parametric study and the results are listed in Tables 6-2 to 6-4. With the input data length (L) equals 1, 3 or

5, the prediction accuracies from the developed CNN model are around 83%, 87%, and 89% respectively. Hence, the following conclusions can be drawn: (i) the CNN model has the power to predict the future bridge condition ratings in the next inspection based on a set of historical condition rating records, with an accuracy over 83% in this case study; (ii) the prediction accuracy improves with increased input record length (L), and the best improvement observed in this study is around 6% when L is equal to five; (iii) Even if using only the latest condition rating score as the input, this particular CNN model still yields a 83% accuracy.

6.3.3.4 Effect of training sample size

The reported bridge condition deterioration model test results are based on the developed CNN model trained with a dataset containing 6800, 5600, 4400 (80% of the total data) training data samples respectively for different input length (i.e., L=1, 3, or 5). In order to discuss the effect of training dataset sizes, the evaluation task with record length (L) of 5 is revisited and the training dataset size is reduced to 1/2, 1/4, 1/8, 1/16, 1/32, or 1/64 of the original dataset size, corresponding to the reduced training data size of around 2000, 1000, 500, 250, 125, or 65 data samples respectively. Since it has been found that the considered CNN model architecture has minor effects on the prediction accuracy of the trained CNN model, only one candidate model is used to investigate the effects of training dataset size. Specifically, the Model B with its base kernel size (k) equal to 128 is trained independently and separately for condition rating modeling

of bridge deck, superstructure, and substructure, while it is later validated with the same size of testing data using the 5-fold cross-validation method. To reduce the uncertainty in the subsampling process of the training fold, the entire 5-fold cross validation is repeated for another 5 times and the results are shown in Figure 6-5. It can be seen that the best performance is always achieved by using the full size of training data in all condition rating prediction cases, and the performance becomes worse when using fewer data for the training. In general, the performance of the developed CNN model converges to the best with the increased size of training data. It can be noted that an order of 2000 (ratio of 1/2) samples can also achieve an accuracy close to the best when using the full size of training data.

6.3.4.5 Error analysis

While the developed CNN model performs well for the bridge condition deterioration modeling, there are several issues in its application: (i) what causes the model's wrong prediction? and (ii) how to improve the method with decreased error rate? A manual inspection of the results has been performed in an attempt to address these questions. The inspection focuses on the wrong predictions made by the CNN model, so the error analysis is performed in the following steps: (1) record every input sample that has been misclassified by the trained CNN model; (2) observe the model's behavior (i.e., the prediction and its associated output probabilities) regarding each recorded input sample. Specifically, the CNN model B with a base kernel size of 256 (k=256) that has

been trained for bridge deck condition rating prediction with input records length (L) of 5 is revisited. It is found that the selected CNN model misclassified 100 out of 1068 testing samples (i.e., the error rate is 9.36%) in one randomly picked training episode. Among the 100 misclassifications, it is also found that the CNN model consistently made the prediction of a set of condition history by assigning the model of a sample, which refers to the condition score that most frequently occurs in the training database, when the specific set of input condition sample is selected. Thus, the cause to the wrong prediction is identified: if the upcoming condition score is not the mode of the sample in the history (training data), the CNN model will make a wrong prediction. This finding is also observed in the results from the cases of bridge superstructure and substructure.

The cause of this behavior is found to be related to the objective (cost) function of the developed CNN model. Recalling the descriptions in Section 6.2, the training of the CNN model using the SoftMax with cross-entropy as the cost function is equivalent to the MLE of the CNN model parameters to match the model distribution with the empirical distribution defined by the training data, and thus the prediction is simply choosing the class with the highest probability. For discrete variables (such as the condition rating scores in this study), the empirical distribution is essentially the frequency (or occurrences) of a sample normalized by the total number of samples and the sample with the highest probability is equivalent to the mode of a sample that occurs most frequently. Here the input samples are a set of condition scores with a predefined

length L, and for a certain set of condition history $h = [CR_1,\ldots,CR_L]$ (L = 1, 3 or 5), the empirical distribution can be determined by Eqn. (6-6),

$$P_{hs} = \frac{N_{hs}}{N_h} \tag{6-6}$$

where N_{hs} = number of facilities transitioned from a condition history h to state s (s=1,2,3,…,9) within one inspection interval and N_h = number of facilities with a condition history h before the transition. Since the CNN model's prediction is assigned to the mode of a sample (i.e., Eqn. (6-5)) based on the model distribution, which is trained to mirror the empirical distribution. For demonstration purpose, four typical sets of input samples are selected to make comparisons between the model distribution (CNN output probabilities) and the empirical distribution (defined by the training data), and the results are plotted in Figure 6-6, which shows that the model distribution is very close to the empirical distribution. For non-mode of a sample, the CNN model will make error predictions. For instance, data samples sharing the same history may have a broad empirical distribution, such as the samples with input [5,5,5,5,5] shown in Figure 6d, but the CNN model will consistently assign the prediction to the mode of the sample (i.e., assign 5 for each input with [5,5,5,5,5]).

This finding can be taken advantage of to propose solutions that would improve the prediction performance of the developed CNN model. Let $p_{data}(x^j)$ denote the empirical distribution of a data sample x^j in the training data x, the prediction of the trained CNN model can be made as Eqn. (6-7),

$$\text{Prediction}(x^j) = \text{argmax}[p_{model}(x^j)] \approx \text{argmax}[p_{data}(x^j)] \tag{6-7}$$

where p_{model} represents the model distribution learned by the CNN model. The reason for using the symbol of approximately equal is due to a nonzero training loss in this study. For instance, a training loss of 0.4 in a minibatch with 2048 samples means that the training loss per sample equals 2×10^{-4} (= 0.4/2048). Hence, the prediction accuracy of the trained CNN model can be estimated using Eqn. (6-8),

$$\begin{aligned} \text{Prediction Accuracy} &= \frac{1}{N}\sum_{j=1}^{N} N_j \cdot \max[p_{model}(x^j)] \\ &\approx \frac{1}{N}\sum_{j=1}^{N} N_j \cdot \max[p_{data}(x^j)] \end{aligned} \tag{6-8}$$

where N denotes the total number of samples, and N_j represents the number of samples with input x^j. It can be revealed that a desired empirical distribution with the property describe by Eqn. (6-9),

$$\max[p_{data}(x^j)] \to 1,\ j = 1, 2, ..., N \tag{6-9}$$

can significantly improve the prediction accuracy. It is also found that the empirical distribution is affected by the entries in the input x^j that adding additional entries or deleting some existing entries will change the empirical distribution. An ideal empirical distribution will have $\max[p_{data}(x^j)]$ (j =1,2,…,N) equal to one. The attempt is made by adding another entry to the input data for the CNN model to distinguish those data samples sharing the same condition history. The marking process is carried out as following (also see the example in Figure 6-7): (1) marking 1 for bridges would behave normal during the inspection interval. This mark will indicate that the bridge

deterioration will follow the general trend in the bridge inventory, and the general trend can be statistically determined based on empirical distribution. This marking is the default setting; (2) marking 2 for bridges with change of condition rating which indicate anticipated abnormal events during the inspection interval. The events might be related to the approval of budgets for maintenance and rehabilitation, abnormal change in traffic volumes, or environmental loadings such as earthquakes or hurricanes. By doing so, the empirical distribution is modified towards to the desired empirical distribution. Consequently, the prediction accuracy can be increased using the CNN model that is trained on such data with the additional mark. For example, it is found that the testing accuracy can be boosted up to 96.7%, 96.2% and 96.0% for condition rating prediction of the bridge deck, superstructure and substructure respectively. Since the marking step is considered to reflect the desired empirical distribution, the error rate of the developed CNN model has been demonstrated to be reduced. This finding reflects that since the CNN model is a data-driven model, its prediction accuracy is closely related to its ability to capture the empirical distribution of the data set. The true empirical distribution can be better represented by adding additional marks provided by the inspection engineers to distinguish data samples with the same condition rating history. Potential marks might be related to traffic volumes, maintenance budget, environmental loading such as earthquakes or hurricanes.

6.3.4 Steel Bridge Condition Deterioration Model

In the preceding case study of concrete highway bridges in the State of Maryland, the use of the proposed deep learning based method has been demonstrated to be effective in bridge condition deterioration modeling. As another effectiveness study, the steel highway bridges with the same routine inspection interval (i.e., 24 months) in the State of Maryland are also considered, with updated CNN model using new training data on steel bridges. In this section, a parametric study on the CNN model architecture effects, training samples size effects, error analysis, and possible improvement procedure are performed but using new data set of NBI steel highway bridges in Maryland.

First, the data samples with a record length (L) are extracted from a sub-inventory of 3,187 steel highway bridges as listed in Table 6-1. It is noted that the data sizes are about twice that of the NBI concrete highway bridges in Maryland. The training convergence can be visualized with the sample learning curves shown in Figure 6-8 that the testing accuracies converged to a stable value. The parametric study results on the CNN model architecture effects are summarized side-by-side with the concrete bridges in Tables 6-2 to 6-4. It shows that the proposed method is also effective as a condition rating prediction tool for steel highway bridges, with an accuracy varying from 83% to 88%. The prediction accuracy is improved by using more historical condition rating records as input (a larger record length L) to the CNN model, and the largest improvement is around 5% when using the latest 5 inspection records. The training sample size effects are plotted in Figure 9, showing that a similar order of 2,000

sample size (ratio of 1/8) can also achieve an accuracy close to the best when using the full size of training data. The error analysis indicates that the cause of wrong predictions is also induced by unanticipated errors in the data empirical distribution. Typical samples for the model distribution learned by the CNN model are plotted in Figure 6-10. Using the marking method, the performance of the re-trained CNN model can also be significantly improved. For instance, the prediction accuracy for condition ratings of bridge deck, superstructure and substructure can be boosted up to 96.6%, 95.9%, 96.2% respectively.

6.3.5 Bridge Condition Deterioration Modeling: Individual Bridge

Once the CNN model has been trained, the validated CNN model can be used as a condition rating prediction tool for major components (deck, superstructure or substructure) of individual bridge in the NBI. Since it has been demonstrated that using more historical inspection records as input can improve the prediction accuracy, the CNN model with an input record length L of 5 is considered for bridge condition deterioration modeling study here. The bridge condition deterioration curve of individual bridge's major component can be obtained (or predicted) through repeated application of the validated CNN model using the latest 5 inspection records as input.

Typical condition deterioration curves predicted by the developed CNN model are plotted in Figure 6-11 along with the real inspection data for the corresponding bridges.

Here, two types of CNN model were considered for the prediction task: (1) with input of real inspection data only (denoted as type I); (2) with input of inspection data augmented with the additional mark (denoted as type II). It can be noticed that both types of CNN model can capture a variety of bridge condition deterioration relationships during the inspection window from the Year 1992 to 2017, such as natural deterioration (undetected degrading in Figure 6-11a, or noticeable decay in Figure 6-11b) and man-made interventions (Figure 6-11c, d). However, Type I model may mis-predict those data points that do not align with the general bridge condition deterioration trend in the selected NBI bridge inventory. This is because the data-driven prediction made by the CNN model heavily relies on the empirical distribution in the training data set. For example, in the given training data, bridges with a historical inspection data of [4,4,4,4,4] would most likely remain in a condition rating of 4 in the next inspection, while a few bridges may be rehabilitated and brought back up to a condition rating of 8 (see Figure 6-6c, Figure 6-10c and Figure 6-11c). Also, it is rarely expected that a bridge will be intervened to a condition rating of 9, if giving an inspection history of [5,5,5,5,5] (see Figure 6-6d, Figure 6-10d and Figure 6-11d). In order to predict such intervention action effect (comparing to most bridges in the inventory), additional information such as the additional mark in Type II model needs to be included in the input data to improve the model, which yield better prediction accuracy compared to Type I model as shown in Figure 6-11. It is also noted that the inspection interval is not always 2 years, since some bridges may require special non-

scheduled inspections after hazardous events such as floods, earthquakes, fires or collisions. If information containing such rehabilitation action is included in the training data set, the validated CNN model can better reflect the empirical bridge condition deterioration relationship even for data with uneven inspection time intervals.

6.3.6 Comparison with Markovian Method

A comparative study with the Markov-Chain model has also been conducted using the same NBI bridge condition rating dataset. The Markov-Chain model assumes the bridge deterioration process as a Markov-Chain based stochastic process that describes the condition deterioration with a probabilistic transition from one state to another during one inspection interval. The probabilistic transition of the facility condition state is defined by the transition matrix, which can be obtained through percentage method, regression-based nonlinear optimization, Bayesian maximum likelihood, and Markov-Chain Monte Carlo method (Micevski et al., 2002; Ranjith et al., 2013; Wellalage et al., 2014). The percentage method is chosen here since the transition matrix can be directly determined from the data and its formulation is correlated to the empirical distribution used by the CNN model. The percentage method defines the probabilistic elements in the transition matrix as Eqn. (6-10),

$$P_{ij} = \frac{N_{ij}}{N_i} \tag{6-10}$$

where N_{ij} = number of facilities transitioned from state i to state j within one inspection interval and N_i = number of facilities in state i before the transition (in this study, i and j = 1,2,…,9). When L = 1, Eqn. (6-6) is found to be identical to Eqn. (6-10). Using a one-hot vector to represent the current condition state S, the prediction of the condition state after one inspection interval can be determined using the transition matrix T by Eqn. (6-11),

$$\text{Prediction} = \text{argmax}(S \cdot T) \tag{6-11}$$

in which the prediction is made by assigning the condition state with the highest probability; the same criterion is used by the CNN model. The multiplication of a one-hot vector with the transition matrix gives a vector with respect to one row of the transition matrix. One example is illustrated in Eqn. (6-12),

$$[0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0] \cdot \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{19} \\ P_{21} & P_{22} & \cdots & P_{29} \\ \vdots & \vdots & \vdots & \vdots \\ P_{91} & P_{92} & \cdots & P_{99} \end{bmatrix} = [P_{31} \quad P_{32} \quad \cdots \quad P_{39}] \tag{6-12}$$

Hence, for the case L=1, if the CNN model is properly trained with the model distribution mirroring the empirical distribution, the mechanism of the trained CNN model is then equivalent to the presented Markovian method. For verification, the determined transition matrix for bridge deck, superstructure, and substructure of concrete and steel bridges are shown in Figure 6-12, in which the percentage method has been used for the calculation. The determined transition matrixes are then used for condition rating prediction using Eqn. (6-11) for each component respectively and the

results are listed in Table side-by-side with the results from the CNN model. It can be found that the results from the two methods are near the same. The minor differences are due to (1) only partial (80%) of the dataset is used for training the CNN model, while the transition matrix is determined using the entire dataset; (2) the converged training loss is not zero, so the model distribution is slightly different from the empirical distribution (e.g., Figure 6-6 and 6-10).

However, for the case L = 3 or 5, the mechanism of the CNN model cannot be compared with the Markovian method since the input of the CNN model is no longer a single condition state. Since more information has been provided as input such as using latest 3 or 5 inspection records, the uncertainties are believed to be reduced, and the prediction accuracy has been demonstrated to be increased with an improvement up to 6%. If further information such as the additional mark is included in the input data, the prediction performance of the CNN model can be further improved, up to 96.7% as the observation from proceeding study results.

6.4 Concluding Remarks

A deep learning based approach involves training a CNN model to predict future conditions of bridge components has been proposed in this study. The study aims to enhance the decision-making process of road authorities by improving the reliability of the forecasts, which can be achieved by training a high-performance CNN model. To

demonstrate and validate the proposed method, a case study of NBI bridges in the state of Maryland has been conducted. The historical condition data in NBI from the Year 1992 to 2017 is used to generate training data for the CNN model. The method has been used to derive CNN models for predicting future conditions of three major bridge components in both concrete and steel highway bridges. The proposed method is found to be effective in predicting the future condition of three primary highway bridge components (i.e., NBI bridge subsystems: bridge deck, superstructure, and substructure) by using either the latest single rating or multiple inspection ratings, with a prediction accuracy up to 96.7% when using the last five inspection records and the additional mark. A comparative study with the Markovian method is also conducted. Specific findings are summarized as follows,

(I) The input data length (L) affects the prediction accuracy of the CNN model. The CNN model with a larger L can give more reliable predictions. In this study, the best results are achieved when L = 5, and the largest improvement is 6% comparing to the cases when L = 3 or 1.

(II) The considered CNN model architecture has minor effects on the prediction accuracy of the trained CNN models. In this study, the main considerations of the model architecture are the width and depth of the CNN model, and the parametric study results indicated that the variation of the mean testing accuracies is within 1.5% due to these two factors.

(III) The size of the training data affects the prediction accuracy of the CNN model. In this study, an order of 2000 training samples is needed to ensure the reported accuracies in the case study.

(IV) The prediction of the trained CNN model is data-driven, and the performance of the trained CNN model can be potentially enhanced by using training data with better empirical distribution. The desired empirical distribution could be established by including additional information in the training data. In this study, the prediction accuracy of the CNN model is improved up to 96.7%.

(V) When using the latest single condition rating for prediction (i.e., L=1), the mechanism of the trained CNN model is found to be similar to a Markov-Chain model, in which the transition matrix is determined using the percentage prediction method.

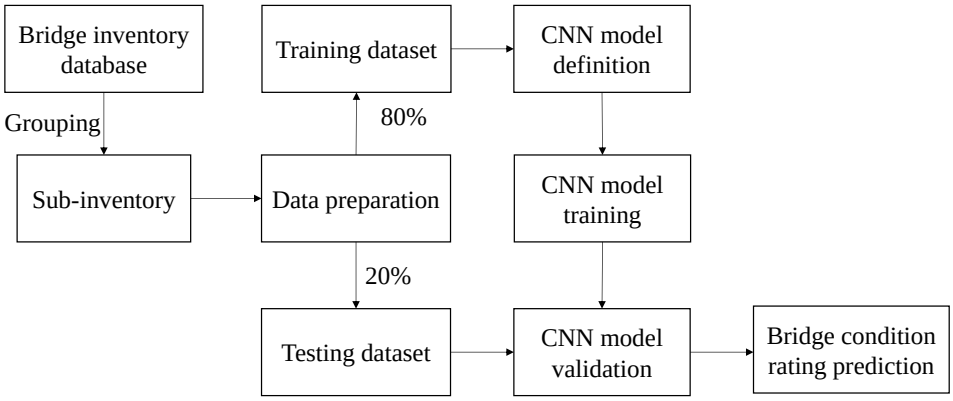


Figure 6-1. Deep learning based bridge condition assessment

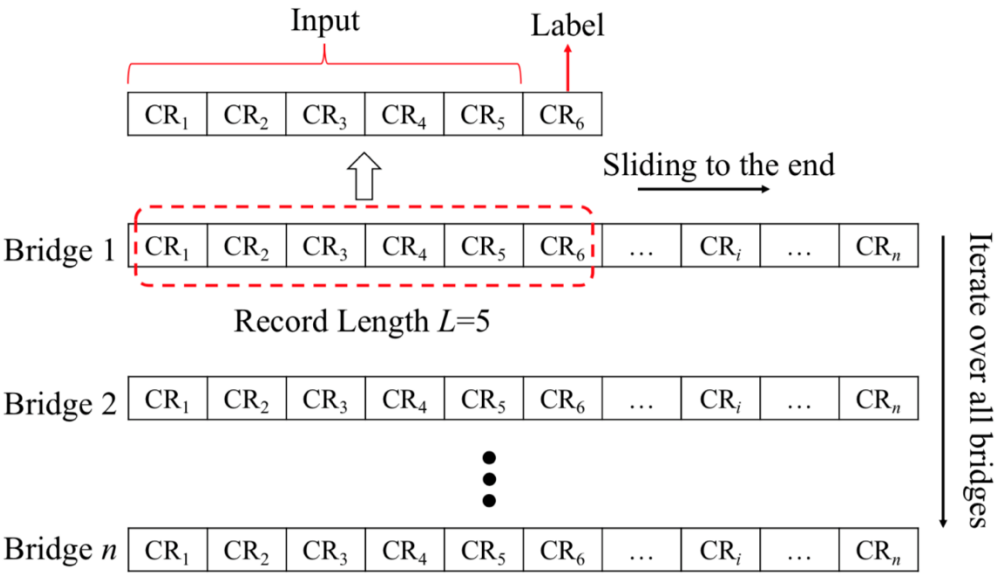
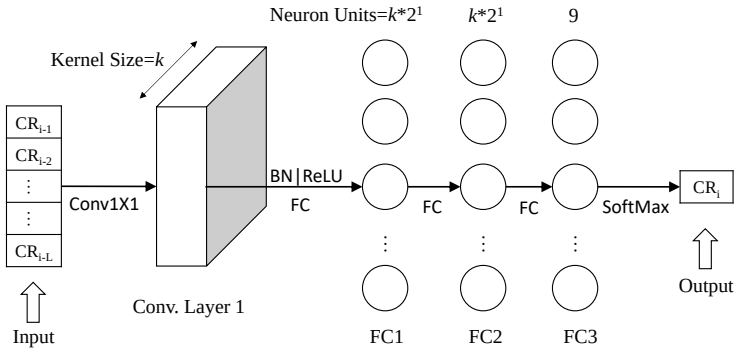
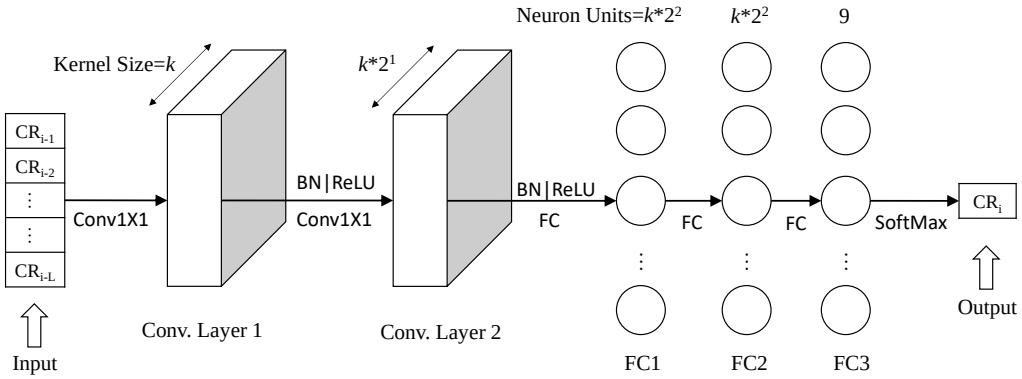


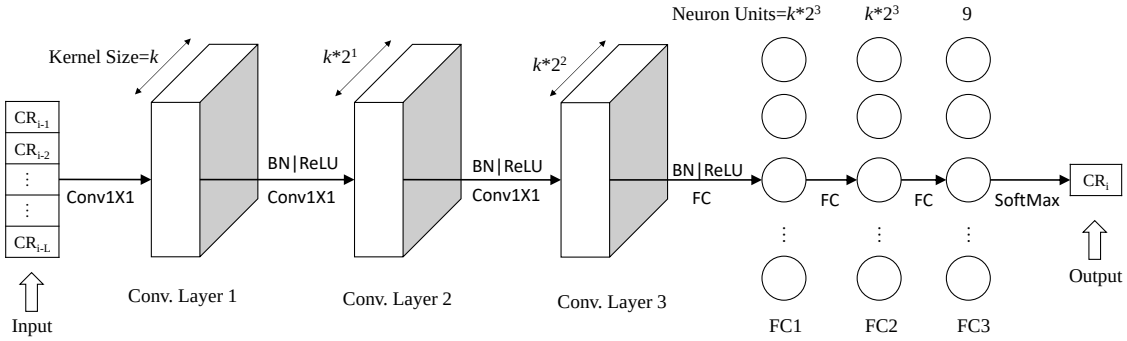
Figure 6-2. Data preparation for deep CNN model training



(a) Model A

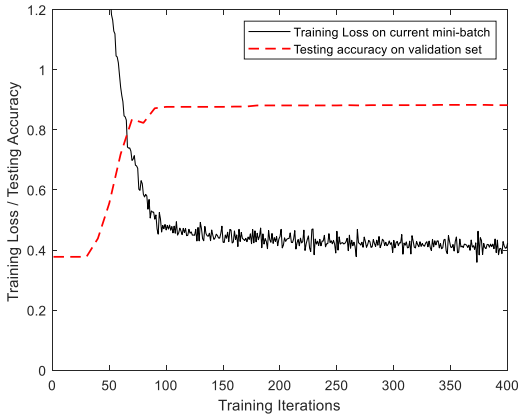


(b) Model B

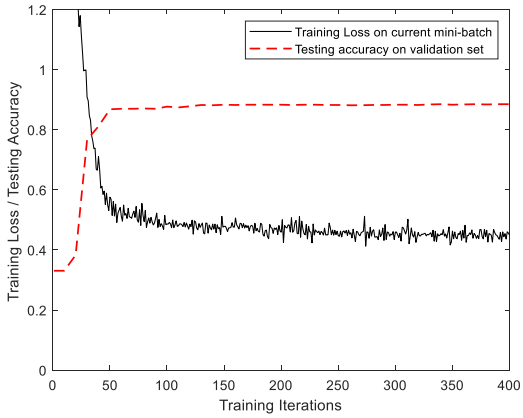


(c) Model C

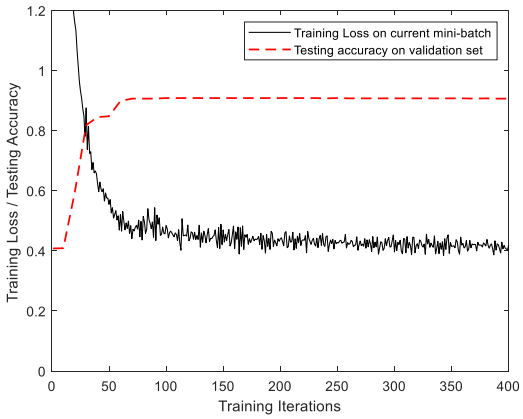
Figure 6-3. The architecture of the developed CNN model: (a) Model A with 1 convolutional layer (m=1); (b) Model B with 2 convolutional layers (m=2); (c) Model C with 3 convolutional layers (m=3)



(a)

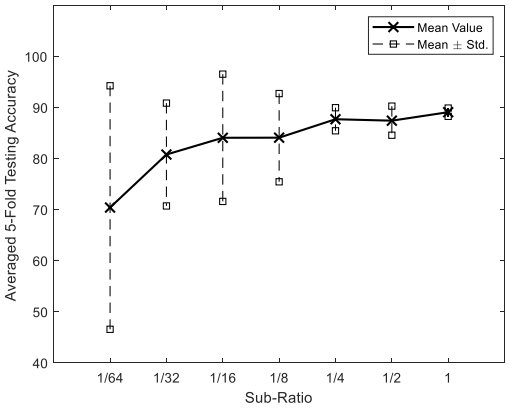


(b)

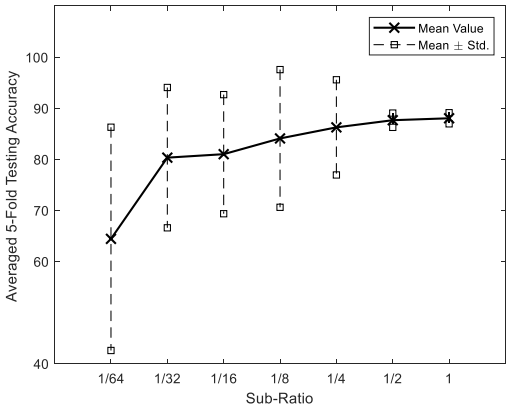


(c)

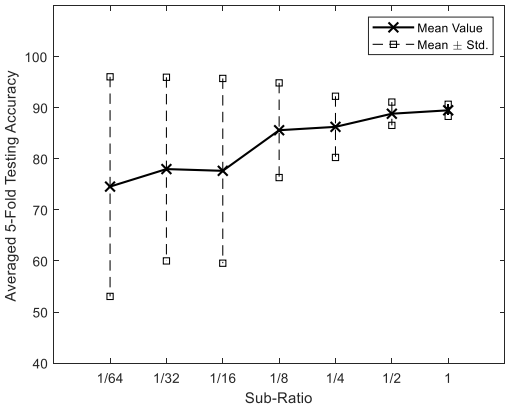
Figure 6- 4. Sample learning curves of the developed CNN model for concrete bridge condition assessment: (a) deck; (b) superstructure; (c) substructure



(a)

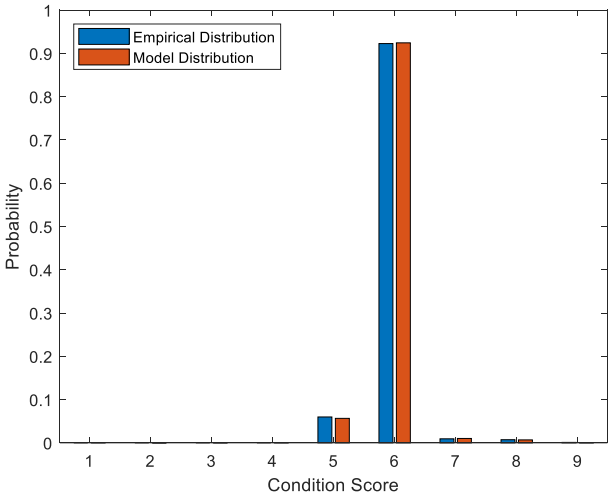


(b)

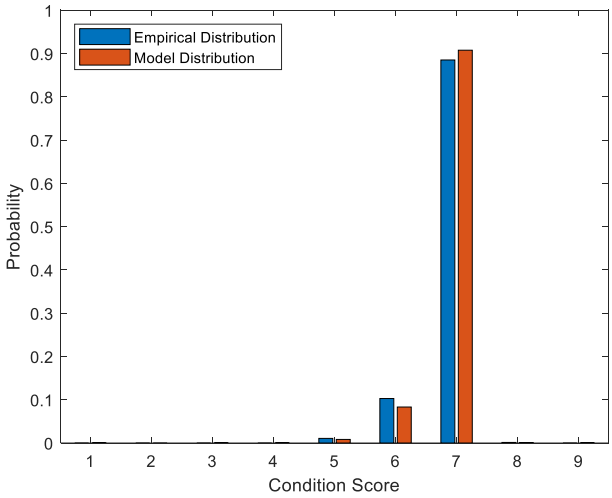


(c)

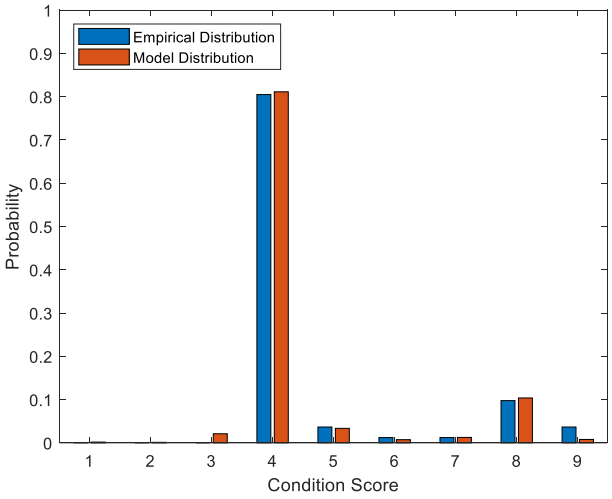
Figure 6-5. The effect of training sample size on the prediction accuracy of developed CNN model for concrete bridge condition assessment (a) deck; (b) superstructure (c) substructure


Empirical Distribution
Model Distribution
Probability
Condition Score
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1
2
3
4
5
6
7
8
9

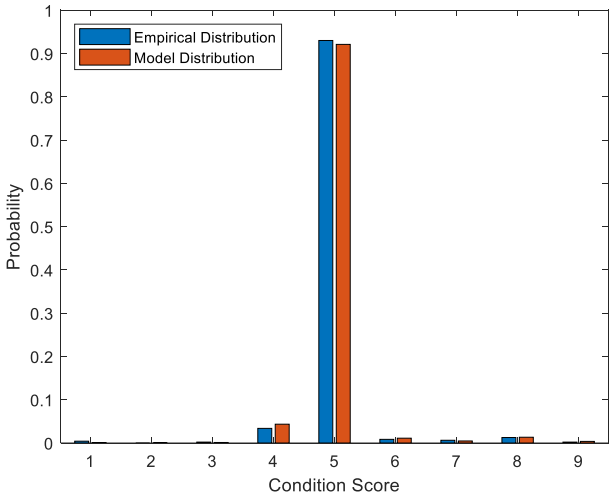
(a)


Empirical Distribution
Model Distribution
Probability
Condition Score
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1
2
3
4
5
6
7
8
9

(b)

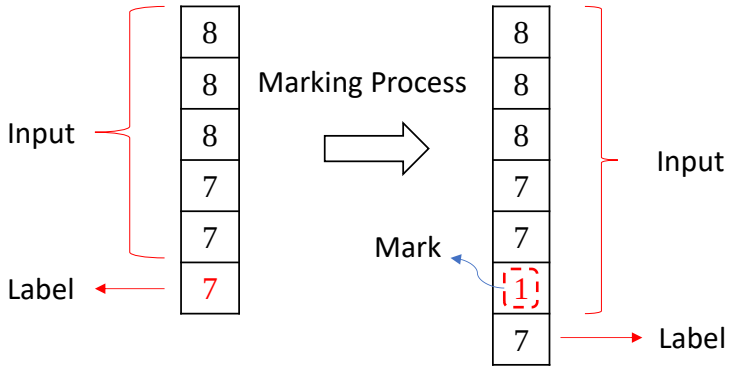


(c)

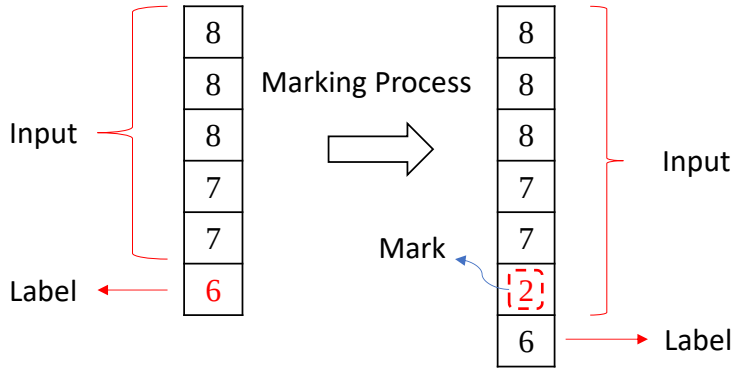


(d)

Figure 6-6. Learned model distribution from concrete bridge data with the input: (a) [6,6,6,6,6]; (b) [7,7,7,7,7]; (c) [4,4,4,4,4]; (d) [5,5,5,5,5]

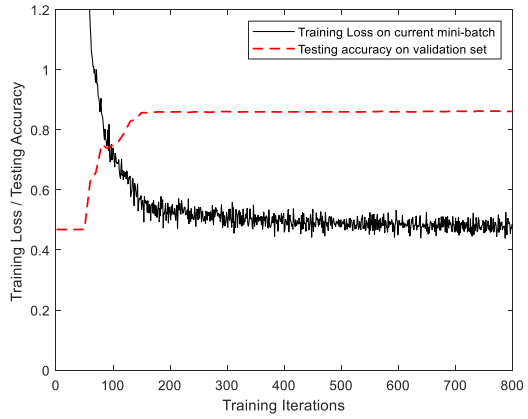


(a)

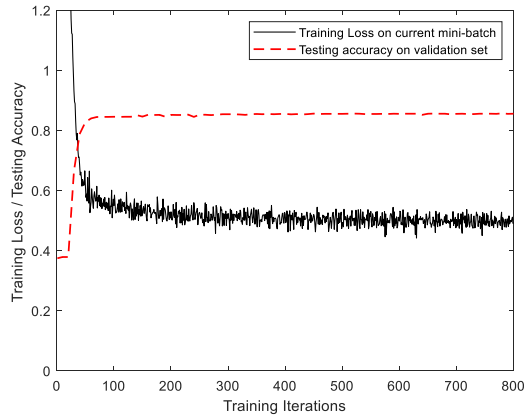


(b)

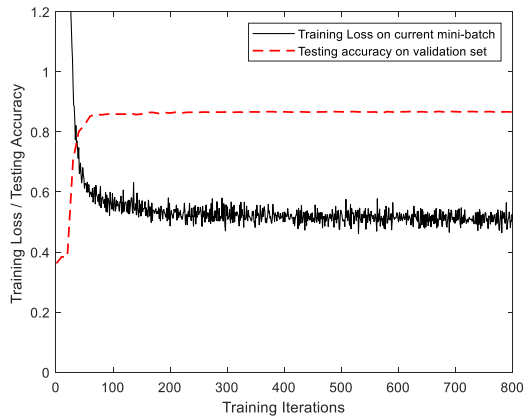
Figure 6-7 Illustration of the marking process to distinguish data samples with the same condition history: (a) input [8,8,8,7,7] with label [7]; (b) input [8,8,8,7,7] with label [6]



(a)

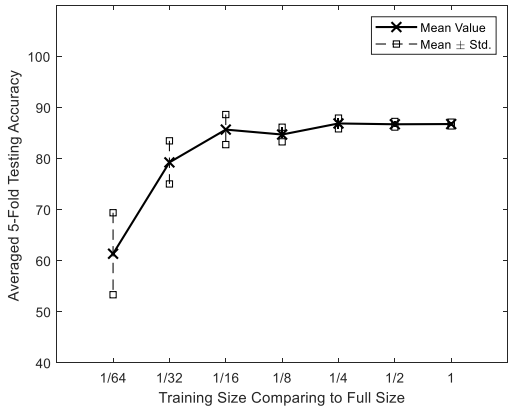


(b)

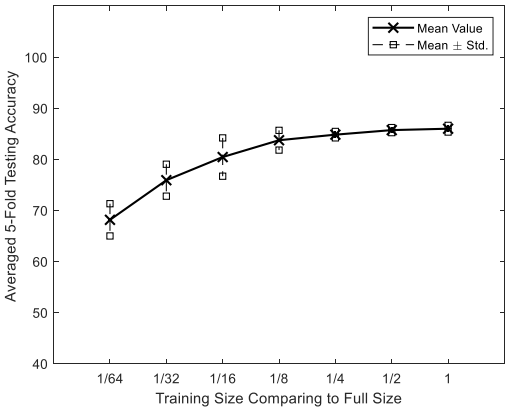


(c)

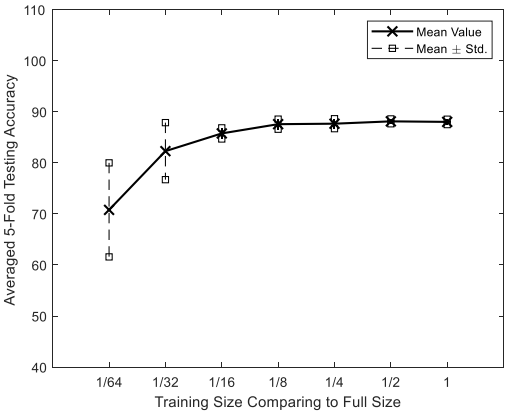
Figure 6-8. Sample learning curves of the developed CNN model for steel bridge condition assessment: (a) deck; (b) superstructure; (c) substructure



(a)

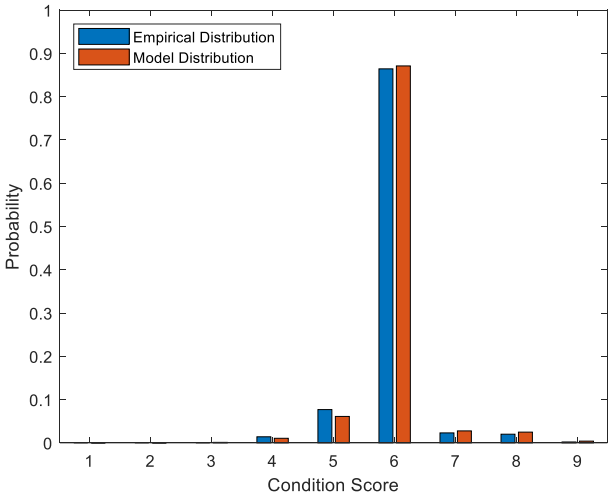


(b)

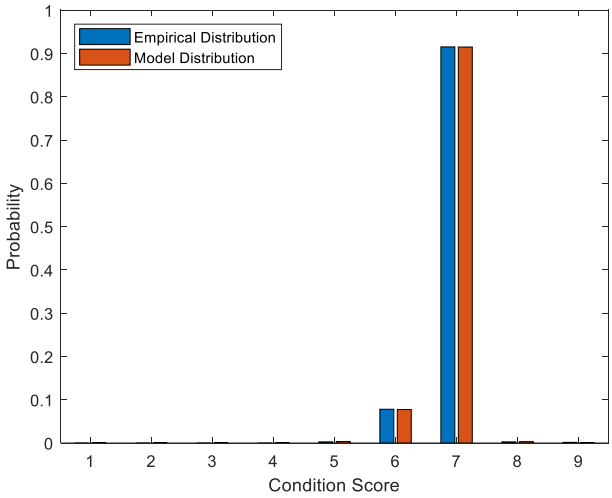


(c)

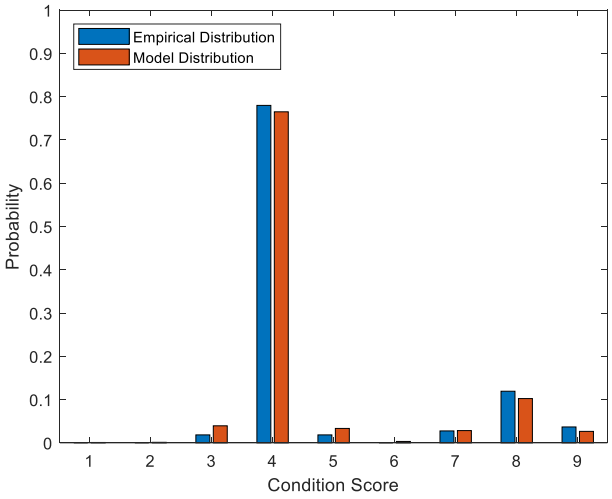
Figure 6-9. The effect of training sample size on the prediction accuracy of developed CNN model for steel bridge condition assessment: (a) deck; (b) superstructure (c) substructure


Empirical Distribution
Model Distribution
Probability
Condition Score
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1
2
3
4
5
6
7
8
9

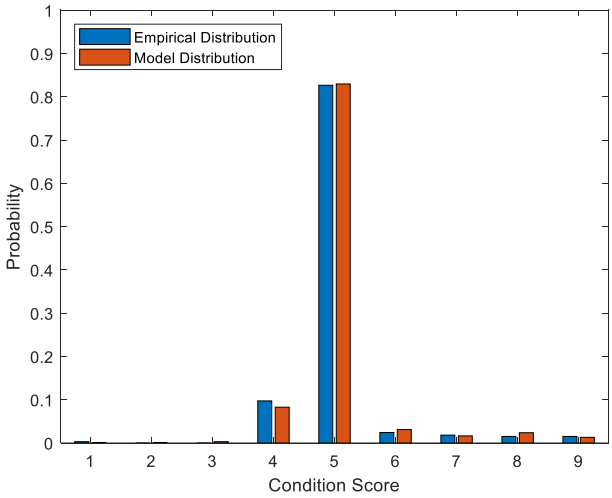
(a)


Empirical Distribution
Model Distribution
Probability
Condition Score
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1
2
3
4
5
6
7
8
9

(b)

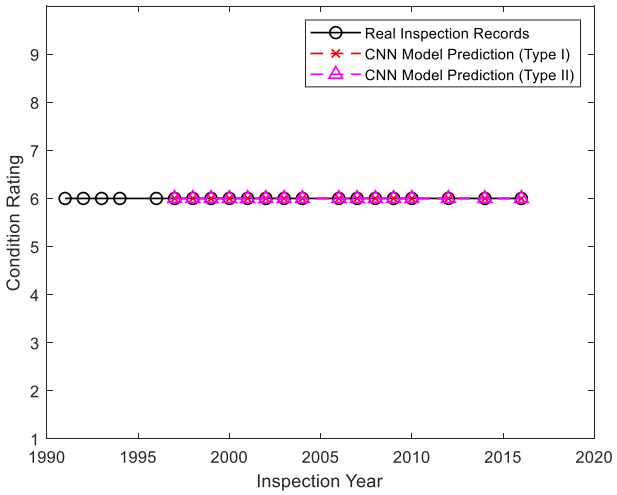


(c)

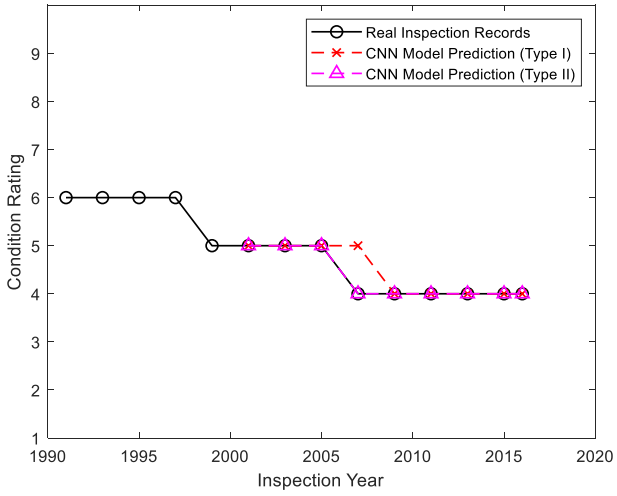


(d)

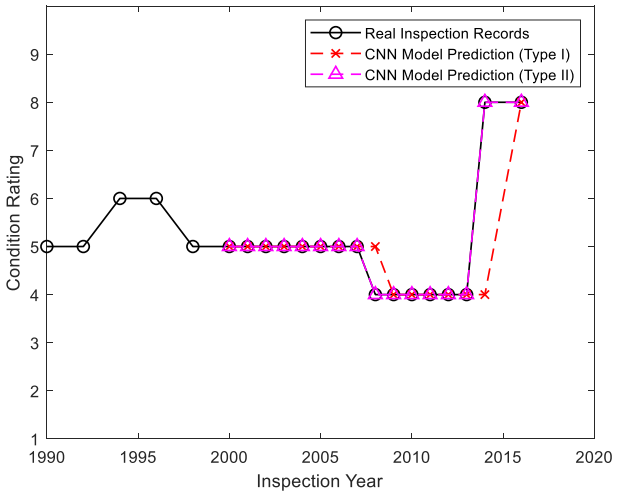
Figure 6-10. Learned model distribution from steel bridge data with the input: (a) [6,6,6,6,6]; (b) [7,7,7,7,7]; (c) [4,4,4,4,4]; (d) [5,5,5,5,5]


Real Inspection Records
CNN Model Prediction (Type I)
CNN Model Prediction (Type II)
Condition Rating
Inspection Year
1
2
3
4
5
6
7
8
9
1990
1995
2000
2005
2010
2015
2020

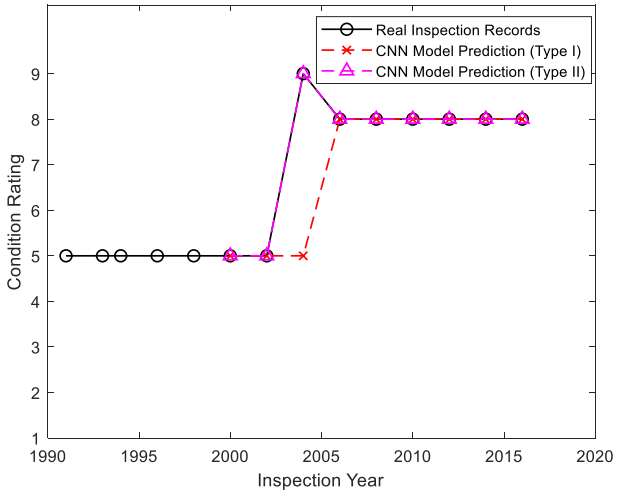
(a)


Real Inspection Records
CNN Model Prediction (Type I)
CNN Model Prediction (Type II)
Condition Rating
Inspection Year
1
2
3
4
5
6
7
8
9
1990
1995
2000
2005
2010
2015
2020

(b)



(c)



(d)

Figure 6-11. Typical deterioration curves for single bridge predicted by CNN model: (a) slowly degrading bridges; (b) notable degrading bridges; (c) bridges with repair actions; (d) bridges with unnormal interventions

(a)

0.83	0	0	0	0	0	0	0.17	0
0	0.44	0	0	0	0	0	0.44	0.11
0	0.03	0.78	0.05	0	0	0.02	0.08	0.05
0	0.00	0.03	0.76	0.05	0.02	0.03	0.08	0.01
0.00	0	0.00	0.05	0.86	0.05	0.01	0.01	0.01
0.00	0	0.00	0.01	0.07	0.87	0.05	0.01	0.00
0.00	0	0	0.00	0.01	0.11	0.86	0.01	0.00
0	0	0	0.00	0.00	0.03	0.26	0.69	0.01
0	0	0	0	0	0	0.14	0.47	0.39

(b)

0.71	0	0	0	0	0	0	0.29	0
0	0.63	0	0.03	0	0.03	0.07	0.17	0.07
0.01	0.04	0.74	0.04	0.03	0	0.04	0.08	0.03
0	0.00	0.04	0.75	0.07	0.03	0.01	0.07	0.02
0.00	0.00	0.00	0.06	0.87	0.05	0.01	0.01	0.00
0.00	0.00	0.00	0.01	0.08	0.86	0.04	0.01	0.00
0.00	0.00	0	0.00	0.01	0.12	0.85	0.01	0.00
0	0	0	0.00	0.01	0.02	0.22	0.74	0.00
0	0	0	0	0	0	0.08	0.49	0.43

(d)

0.66	0	0	0	0.03	0	0.10	0.21	0
0	0.58	0	0.05	0	0	0	0.21	0.16
0.02	0.05	0.57	0.06	0.02	0.02	0.06	0.14	0.06
0.00	0.00	0.04	0.69	0.07	0.02	0.03	0.12	0.03
0.00	0	0.00	0.08	0.79	0.06	0.03	0.03	0.01
0.00	0	0.00	0.01	0.06	0.85	0.06	0.02	0.01
0.00	0	0.00	0.00	0.01	0.08	0.89	0.02	0.00
0	0	0	0.00	0.00	0.02	0.29	0.69	0.00
0	0	0	0	0	0.00	0.09	0.56	0.34

(e)

0.58	0.02	0	0.02	0	0.07	0.05	0.21	0.05
0	0.47	0	0.05	0.04	0.05	0.05	0.22	0.11
0.04	0.03	0.61	0.07	0.05	0.01	0.04	0.11	0.05
0.01	0.01	0.06	0.70	0.06	0.03	0.02	0.08	0.03
0.00	0.00	0.00	0.06	0.81	0.07	0.02	0.02	0.01
0.00	0.00	0.00	0.01	0.07	0.86	0.04	0.01	0.00
0	0	0.00	0.00	0.01	0.10	0.86	0.02	0.00
0	0	0	0.00	0.00	0.02	0.22	0.76	0.00
0	0	0	0	0	0	0.07	0.57	0.36

(c)

0.63	0	0	0	0	0	0	0.25	0.13
0	0.67	0	0	0	0	0	0.17	0.17
0	0	0.64	0.11	0.07	0.03	0.05	0.08	0.03
0.01	0.00	0.03	0.66	0.12	0.06	0.04	0.06	0.02
0.00	0	0.00	0.04	0.85	0.07	0.02	0.02	0.01
0.00	0.00	0.00	0.00	0.06	0.89	0.04	0.01	0.00
0	0	0	0.00	0.01	0.10	0.86	0.02	0
0	0	0.00	0.00	0.01	0.03	0.25	0.71	0
0	0	0	0	0	0	0.08	0.48	0.44

(f)

0.46	0	0	0.08	0.08	0	0.17	0.13	0.08
0.05	0.73	0	0.03	0.03	0	0.03	0.11	0.03
0.01	0.02	0.54	0.08	0.06	0.03	0.09	0.17	0.02
0.01	0.01	0.03	0.62	0.10	0.08	0.05	0.08	0.03
0.00	0.00	0.01	0.04	0.80	0.10	0.03	0.02	0.01
0.00	0.00	0.00	0.00	0.06	0.88	0.05	0.01	0.00
0	0	0.00	0.00	0.01	0.09	0.88	0.02	0.00
0	0	0.00	0.00	0.00	0.01	0.25	0.72	0.00
0	0	0	0	0	0	0.08	0.50	0.42

Figure 6-12. Determined transition matrix of concrete bridges' (a) deck (b) superstructure (c) substructure; steel bridges' (d) deck (e) superstructure (f) substructure

Table 6-1. Data sample size regarding to different record length

Bridge Material	Bridge Element	Total Samples		
		Record Length L=1	Record Length L=3	Record Length L=5
Concrete	Deck	8432	6754	5341
	Superstructure	8661	7010	5561
	Substructure	8745	7062	5612
Steel	Deck	20925	15569	11727
	Superstructure	21085	15734	11880
	Substructure	21071	15718	11865

Table 6-2. Evaluation results for bridge deck (mean ± std., %)

Bridge Material	CNN Model Architecture Configuration					
Concrete	Input Record Length $L = 1$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	83.36±0.75	83.36±0.75	83.48±0.79	83.59±0.82	83.70±0.95
	B (5)	83.92±1.01	83.92±1.01	83.92±1.01	83.95±1.00	83.98±1.03
	C (6)	83.95±0.83	83.98±0.83	84.00±0.81	84.00±0.81	83.98±0.83
	Input Record Length $L = 3$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	87.15±0.50	87.22±0.44	87.36±0.58	87.37±0.67	87.37±0.63
	B (5)	87.87±0.70	87.85±0.70	87.93±0.68	88.10±0.66	88.08±0.68
	C (6)	87.83±1.19	88.07±1.29	88.08±1.29	88.08±1.33	87.95±1.24
	Input Record Length $L = 5$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	88.65±0.29	88.69±0.34	88.73±0.22	88.63±0.47	88.80±0.64
	B (5)	88.59±1.04	88.64±1.11	89.07±1.06	89.20±1.06	89.44±0.97
	C (6)	89.35±1.34	89.40±1.28	89.38±1.23	89.37±1.26	89.12±1.15

	Input Record Length $L = 1$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	82.65±0.51	82.63±0.50	82.69±0.47	82.58±0.53	82.73±0.59
	B (5)	82.89±0.62	82.89±0.71	82.90±0.61	82.90±0.64	82.77±0.59
	C (6)	82.90±0.59	82.90±0.60	82.90±0.65	82.92±0.61	82.87±0.67
	Input Record Length $L = 3$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
Steel	A (4)	85.13±0.43	85.09±0.39	85.11±0.40	85.14±0.43	85.12±0.43
	B (5)	85.05±0.50	85.09±0.50	85.16±0.52	85.13±0.52	85.23±0.33
	C (6)	85.49±1.18	85.48±1.15	85.39±1.19	85.45±1.21	85.41±1.15
	Input Record Length $L = 5$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	86.61±0.43	86.58±0.43	86.60±0.43	86.68±0.25	86.52±0.34
	B (5)	86.60±0.43	86.64±0.46	86.67±0.62	86.59±0.49	86.57±0.47
	C (6)	86.88±0.89	86.87±0.89	86.83±0.85	86.71±0.80	86.63±0.69

Table 6-3. Evaluation results for bridge superstructure (mean ± std., %)

Bridge Material	CNN Model Architecture Configuration					
Concrete	Input Record Length L = 1	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	82.44±0.64	82.58±0.82	82.74±0.78	82.84±0.77	82.58±0.82
	B (5)	83.15±0.93	83.15±0.93	83.19±0.86	83.32±0.96	83.43±0.93
	C (6)	83.33±0.57	83.25±0.67	83.29±0.69	83.27±0.69	83.24±0.74
	Input Record Length L = 3	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	85.79±0.65	85.82±1.01	86.23±0.68	85.82±0.87	85.44±2.38
	B (5)	86.41±1.13	86.46±1.19	86.50±1.11	86.78±1.23	86.80±1.36
	C (6)	86.42±0.70	86.58±0.62	86.62±0.62	86.66±0.33	86.48±0.59
	Input Record Length L = 5	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	87.23±1.57	87.65±1.74	87.83±1.38	87.54±1.55	87.86±1.35
	B (5)	88.10±1.04	88.04±1.06	87.95±1.02	88.15±0.96	88.06±0.96
	C (6)	88.22±1.23	88.28±1.48	88.26±1.30	87.90±1.37	87.70±1.45

	Input Record Length $L = 1$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	81.67±0.75	81.88±0.83	81.84±0.80	81.88±0.69	81.77±0.74
	B (5)	82.12±0.76	82.10±0.79	81.96±0.83	82.10±0.77	82.16±0.71
	C (6)	82.13±0.28	82.17±0.31	82.17±0.34	82.15±0.34	82.21±0.30
	Input Record Length $L = 3$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
Steel	A (4)	83.75±0.47	83.96±0.51	84.07±0.46	83.92±0.48	84.14±0.63
	B (5)	84.38±0.68	84.40±0.64	84.46±0.67	84.42±0.67	84.44±0.63
	C (6)	84.43±0.44	84.49±0.47	84.55±0.45	84.49±0.43	84.43±0.46
	Input Record Length $L = 5$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	85.39±0.89	85.53±0.99	85.46±1.02	85.66±0.76	85.86±0.70
	B (5)	85.94±0.81	86.07±0.91	86.09±0.86	86.06±0.93	86.05±0.90
	C (6)	86.14±0.89	86.09±1.03	86.05±0.97	85.84±0.98	85.66±0.90

Table 6-4. Evaluation results for bridge substructure (mean ± std., %)

Bridge Material	CNN Model Architecture Configuration					
Concrete	Input Record Length L = 1	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	84.24±0.81	84.24±0.81	84.24±0.81	84.24±0.81	84.24±0.81
	B (5)	84.48±0.72	84.69±0.74	84.69±0.74	84.69±0.74	84.67±0.74
	C (6)	84.71±0.65	84.72±0.65	84.64±0.69	84.65±0.71	84.78±0.62
	Input Record Length L = 3	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	88.22±0.78	88.19±0.76	88.19±0.76	88.09±0.79	88.05±0.84
	B (5)	88.53±0.98	88.28±0.78	88.40±0.99	88.67±1.02	88.67±1.00
	C (6)	88.40±0.55	88.49±0.71	88.70±0.63	88.59±0.64	88.57±0.63
	Input Record Length L = 5	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	89.42±1.11	89.40±1.09	89.34±1.14	89.22±1.23	89.04±0.96
	B (5)	89.40±0.87	89.43±0.91	89.61±0.84	89.70±0.92	89.65±0.90
	C (6)	89.40±0.88	89.58±0.79	89.72±0.79	89.42±0.86	89.20±0.89

	Input Record Length $L = 1$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	83.72±0.32	83.68±0.27	83.74±0.33	83.72±0.29	83.73±0.33
	B (5)	83.85±0.56	84.04±0.52	84.00±0.52	84.03±0.49	84.13±0.44
	C (6)	84.03±0.72	83.96±0.80	84.16±0.66	84.11±0.69	84.13±0.71
	Input Record Length $L = 3$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
Steel	A (4)	86.50±0.43	86.49±0.41	86.47±0.36	86.49±0.45	86.39±0.38
	B (5)	86.69±1.20	86.70±1.20	86.83±1.10	86.89±1.09	86.80±1.11
	C (6)	86.76±0.88	86.85±0.82	86.85±0.83	86.84±0.75	86.89±0.93
	Input Record Length $L = 5$	Width (base kernel size, k)				
	Model (# of layers)	16	32	64	128	256
	A (4)	88.02±1.06	87.98±1.03	87.86±1.09	87.91±1.08	87.88±1.11
	B (5)	88.05±0.92	88.17±0.90	88.21±1.08	88.22±0.96	88.26±0.81
	C (6)	88.24±0.50	88.29±0.43	88.23±0.34	88.16±0.36	87.98±0.26

Table 6-5. Comparison between the Markovian method and proposed method

Bridge Material	Bridge Components	Prediction Accuracy (%)	
		Markov Chain Model	CNN Model (L=1)
Concrete	Deck	84.03	84.00
	Superstructure	83.43	83.43
	Substructure	84.79	84.78
Steel	Deck	82.98	82.92
	Superstructure	82.25	82.21
	Substructure	84.18	84.16

Chapter 7: Summary, Conclusion and Future Work

7.1 Summary

The proposed structural health condition assessment methods using deep learning algorithm are very promising as smart Structural Health Monitoring (SHM) techniques for civil infrastructures in addressing the need of rapid and accurate structural health condition assessment. Its successful development aims to reduce the uncertainties in assessing the true condition state of civil infrastructures and to reduce inspection time, so that civil infrastructures resilience can be enhanced with a shortened recovery phase. A pilot study is conducted here to investigate the effectiveness of the proposed method for commonly encountered structural health condition assessment problems in civil infrastructures, such as damage condition assessment of seismic fuse members, brace damage detection in braced framed building structures, Guided Lamb wave based damage detection in plate structures, and bridge condition deterioration modeling. The main objective of this study is to establish an effective implementation procedure for integrating the deep learning algorithm with SHM applications. A summary of the conducted research work towards this objective is listed as follows:

In Chapter 2, a literature review has been performed on structural health condition assessment technologies commonly used for civil infrastructures, including structural steel fuse condition assessment in eccentrically braced frame structures,

brace damage detection in concentrically braced frame structures, guided lamb wave based non-destructive testing (NDT) for plate structures, and bridge condition rating deterioration modeling based on the National Bridge Inventory database. Additionally, previous research works in deep learning are reviewed with emphasis on the computer vision related topics. Deep learning based SHM technologies published recently are also covered. The previous works reveal that the Convolutional Neural Network has the strongest potential to be used for the SHM application here. The transfer learning can be utilized for simplifying and accelerating the training of the deep learning model. The visualization technique can be taken advantage of to interpret results from deep learning modeling by comparing with human intuitions.

In Chapter 3, the research work focuses on the topic of image-driven structural steel damage condition assessment using deep learning algorithm. In this study, the deep learning algorithm has been applied to a similar application that uses image data for condition assessments. This study first examines the specific deep learning model used for solving the application problem. The proposed method is demonstrated and validated in two independent case studies using different sources of image data - finite element simulation generated contour plot images, printed photo images, and photo images taken in structural test experiments. As the first study in the dissertation, the research aims to examine the effectiveness of the proposed method in such image-driven application, and further study the implementation requirements such as how to effectively train the deep learning model, etc. The study results indicated that the

proposed method is effective (achieving a damage identification accuracy of over 95%) for damage condition assessment in both case studies. Fine-tuning of all layers in a pre-trained CNN model is the recommended training strategy for structural damage condition assessment application; by doing so the training data size can be made far less than the super large data size that would be typically required for training a deep learning model from scratch. Training data size is an important factor affecting the performance of the CNN model for image driven low cycle fatigue induced damage condition assessment. The order of magnitude for the training dataset is found to be a few thousand images when fine tuning of pre-trained CNN model is adopted, to ensure the ninety percent accuracy seen in the case studies. Saliency map offers a visual tool to evaluate whether the trained deep learning model is doing the right thing and distinguish well-trained deep learning models from those poorly-performing models by looking at the feature-related patterns aligned with human visual inspection. The guided back-propagation method provides the salience map showing the feature-related patterns in the original images recognized by the deep learning model.

In Chapter 4, presents the research work on the topic of deep learning based brace damage detection for concentrically braced frame structures under seismic loadings. This chapter extends the proposed method to time series data such as the vibration response of structural frames subjected to earthquake ground motion. The current study further examined the research work on deep learning method for SHM, including working procedure, training method and strategy. Modification of the method

to accommodating the time series data is also presented. The modified method is demonstrated and validated in a numerical simulation based case study of a 6-story CBF structure under seismic excitation. A statistical approach is used to analyze the results from the deep learning method. The study results indicate that proposed damage detection method is effective in identifying the presence of seismic-induced brace damage (buckling) in the CBF structures, with an accuracy over 95.60% in this case study. The inspiration from the saliency map revealed a simple feature that was tested with acceptable accuracy, which showed the potential to learn from the deep learning. However, the performance of the inspired feature yielded to the optimization of the feature function. The deep CNN model was trained to discover more discriminative features comparing to the simple FNN through the optimization of more advanced feature function, which was generalized by the stacking of different computational layers. The transfer learning was shown to be effective in training the deep CNN model for the detection of seismic-induced brace damage (buckling) in the CBF structures. The transfer learning can simplify the design and training of the deep CNN model. The adopted CNN model here offered an option with fine tuning a well pre-trained model on a database at order of a few thousand training samples for this specific brace damage detection problem.

In Chapter 5, research extends to the topic of deep learning based nondestructive evaluation (NDE) of crack damage in thin plate structures using guide Lamb wave signals. The study further examines the proposed method, including

technical procedure, adaptation to time-series data, training method and strategies. The proposed method has been demonstrated and validated in a case study of damage detection on aluminum plates with notch or breathing crack damages, in which the calibrated analytical model has been used to generate the wave signal data. The study results indicate that the proposed method can identify the occurrence, location and severity of the concerned damage, with an accuracy level of over 97.5% in the case study. The proposed method is also found to show robust performance under white noise influence. Three levels of white noises with an SNR of 15, 20, or 25 dB respectively have been included in the original input Lamb wave data to investigate the noise effect, with a testing accuracy of 98.43% 98.33% or 98.5% respectively. The proposed method is found not sensitive to the selected center frequency value of the excitation signal. Three excitation signals with different center frequency values of 100 kHz, 220 kHz, or 300 kHz have been considered, in which the white noise level is locked at 25 dB SNR, and result in testing accuracy of 98.5% 99.8% or 99.8% respectively. The damage evaluation results of the proposed method can be enhanced regarding the damage location with minor loss of performance. The number of potential damage regions (or bins) is increased to 8 or 16, so the range of each damage region is further refined. The proposed method is validated with a testing accuracy of 98.17% in the case of 8 bins and 97.5% in the case of 16 bins. The saliency map based visualization technique can be used to reveal the recognized visual features by the trained CNN model for the damage evaluation task. By comparing the recognized

visual features with manual damage analysis, it is found that the validated CNN model is searching for visual features consistent with human intuition, and this search process can be automated and does not require any prior feature definition task.

In Chapter 6, research is carried out on the topic of NBI bridge condition rating data modeling using the Convolutional Neural Network model. This chapter aims to extend the proposed method to discrete data (neither images nor time series data) such as the bridge condition data in NBI. The study covers a discussion of the technical procedure, adaptation to different input data format, training method and strategies. The modifications including changes made on the architecture of the CNN model is presented. The modified method is demonstrated and validated in a case study of Maryland federal-aid state highway concrete and steel highway bridges. Discussion of the results are also included to offer insights to the proposed method. The study results indicate that the proposed method can effectively predict the future condition of the three major bridge components using a set of condition history records, with an accuracy near 90% in the case study. Training data size is an important factor affecting the performance of the CNN model for the NBI bridge condition rating data application. The required order of magnitude for the training dataset is found to be a few thousands when training the selected CNN model, to ensure the near ninety percent accuracy seen in the case studies. The performance of the CNN model can be enhanced by including more information in the input data.

7.2 General Conclusion

Based on the results from the above four distinct SHM case studies commonly seen in civil infrastructure's structural health condition assessment application, the following findings can be drawn:

- The investigated deep learning algorithm is very promising in developing smart SHM techniques for rapid and accurate structural health condition assessment of civil infrastructures, based on the reported success in the four distinct SHM tasks. The demand for rapid and accurate structural health condition assessment has been addressed by first converting the SHM task into a classification problem, which can be elegantly solved using adapted deep learning algorithm. In this dissertation research, the deep Convolutional Neural Network is adapted as the deep learning model.
- Training of the deep learning model in the considered application has been conveniently completed through transfer learning from the selected CNN model pre-trained for other fields. In practice, the transfer learning can benefit the application in two folds: (1) challenges in the design of a sophisticated deep learning model can be circumvented by fine-tuning a pre-trained deep CNN model with validated performance; (2) the typical requirement on large training data size can be substantially reduced. Regarding the training strategy in transfer learning, the one with fine-tuning of all layers is found to display the best performance, which

is recommended here. In terms of required training data size, it is found that an order of a few thousand is necessary to achieve the reported performance in the case studies.

- The proposed method can take as input various types of SHM data including 2-D photo images, 1-D time series (e.g., vibration and Lamb wave time series signals), and discrete data (e.g., NBI bridge condition rating data). Since the pre-trained CNN model was originally developed for computer vision tasks with images as input, raw SHM data needs pre-processing to some extent. Therefore, while the photo images only require the resize operation, time series data needs to be transformed into image format. The short-time Fourier transformation algorithm is used here to preprocess the time series data into time-frequency spectrum images. The results are found to be effective in the reported case studies. For discrete data, customary developed CNN model is studied and validated, which provides a reference for related applications.
- The saliency map based visualization technique, such as the guided-backpropagation used in the study, can offer visual clues on the results from the deep learning model, which can be used to evaluate its reasonability with an intuitive comparison. The use of the saliency map also shows a potential application in learning from the deep learning so that a simple feature can be identified from the saliency map.

7.3 Research Contribution

This dissertation research mainly contributes to the technical field of Structural Health Monitoring (SHM). To bridge the current gap between recent advancement in machine learning and SHM needs, customization of deep learning algorithms and big data technology for civil infrastructure SHM is being focused here to enable automated and accurate structural condition assessment and condition deterioration data modeling. The original contributions of this research work include:

- Different SHM data types require algorithm customization and different data preparation such as labeling techniques. In some situations, well-labeled data is not available for immediate implementation. Efficient data generation technique is therefore needed. For example, in this study, finite element simulation has been examined as an alternative method for data generation in the image-driven structural steel damage identification application. Also, a micromechanical fracture damage index has been tested here to label the image data in the application of structural steel fuse damage condition assessment. The effectiveness of the proposed methods in algorithm customization and different data preparation has been demonstrated with high accuracy observed in four different SHM case studies including image-driven damage condition assessment of structural steel fuse member, time series data for bracing damage

detection and Lamb wave based NDE in thin-plate structures, as well as bridge condition deterioration modeling.

- Due to lack of well-labeled SHM data for deep learning model training, an alternative technique is desired for effective training of the deep learning model with data size much smaller than those typically encountered in computer vision field. In this direction, the transfer learning from validated CNN model from other fields is tested for SHM data application. With transfer learning, the typical requirement on large training data size can be substantially reduced to a few thousand in the considered SHM problems. Different training strategies in transfer learning have been investigated in this study and it is found that the one with fine-tuning of all layers has shown strong performance.
- Since the pre-trained CNN model was originally developed for computer vision tasks with images as input, raw SHM data has to be prepared to fit the CNN model. For example, those time series data including vibration and ultrasonic surface wave data need to be first transformed into the image format. Different transformation methods have been investigated, among which the short-time Fourier transformation algorithm is found to be most effective.
- Saliency map based visualization technique can offer visual clues on understanding why the adapted deep learning model works effectively in damage pattern identification. After visualizing the identified feature by the saliency map, simplification and explicit definition of features can be achieved

so that a simple feature can be developed for SHM without the need for deep CNN Model in certain applications without required computing resource available. This concept of SHM feature inspiration and modification from saliency map is believed to be the first attempt in the field of SHM of civil infrastructures.

- Customized deep CNN models have been developed in addressing the need for inputting discrete data in the application of bridge condition data modeling, where the above-mentioned transfer learning technique is no longer applicable. Different architectures have been tested in order to obtain the most effective CNN models for condition deterioration modeling in both concrete and steel highway bridges. Along with this line, the effort has also been made to fully understand the adapted deep learning algorithm, so modifications can be made to optimize the model configuration for this particular application.

7.4 Future Work

While the current study has done a pilot investigation of four commonly seen SHM applications using the deep learning algorithms, several recommendations worth of further study are listed as follows,

- The proposed method involves an important but time-consuming process of data generation and labeling, which is very costly if done manually. Developing an

efficient automation procedure that can accelerate this process is highly desired for wide adoption of the deep learning models in civil infrastructure application.

- In the current research, a special type of deep learning model termed the Convolutional Neural Network is investigated. Future study may also look at other types of deep learning models with demonstrated success in certain applications involving time series data. For example, the long-time short memory (LSTM) is another type of deep neural network model specializing in processing time series data and it has been successfully used for solving problems in the field of natural language processing, such as speech recognition and machine translation.
- The study on bridge condition rating data modeling only used the NBI highway bridge data in the State of Maryland. The current study can be expanded to more States or even to other nations. A comprehensive study can be conducted to investigate major factors potentially affecting bridge condition deterioration modeling and incorporation into bridge management systems, such as the operating environments, design and construction practice in different States or geographic regions.

References

Achenbach, J. (2012). *Wave propagation in elastic solids*. North Holland: Elsevier.

Agarwal, S., & Mitra, M. (2014). Lamb wave based automatic damage detection using matching pursuit and machine learning. *Smart Materials and Structures*, *23*(8), 085012.

AISC. (2010). Seismic provisions for structural steel buildings. ANSI/AISC 341-05. Chicago, IL.

ANSYS. (2017). ANSYS Academic Research, ANSYS Inc, Canonsburg, Pennsylvania, USA.

Ayyub, B. M. (2014). Systems resilience for multihazard environments: Definition, metrics, and valuation for decision making. *Risk Analysis*, *34*(2), 340-355.

Ayyub, B. M. (2015). Practical resilience metrics for planning, design, and decision making. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, *1*(3), 04015008.

Badcock, R. A., & Birt, E. A. (2000). The use of 0-3 piezocomposite embedded Lamb wave sensors for detection of damage in advanced fiber composites. *Smart Materials and Structures*, *9*(3), 291-297.

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. New York: Springer-Verlag New York

Bleck, W., Dahl, W., Nonn, A., Amlung, L., Feldmann, M., Schäfer, D., & Eichler, B. (2009). Numerical and experimental analyses of damage behaviour of steel moment connection. *Engineering Fracture Mechanics*, *76*(10), 1531-1547.

Cao, P., Zhang, S., & Tang, J. (2018). Preprocessing-Free Gear Fault Diagnosis Using Small Datasets with Deep Convolutional Neural Network-Based Transfer Learning. *IEEE Access*, *6*, 26241-26253.

Cesare, M. A., Santamarina, C., Turkstra, C., & Vanmarcke, E. H. (1992). Modeling bridge deterioration with Markov chains. *ASCE Journal of Transportation Engineering*, *118*(6), 820-833.

Cha, Y. J., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, *32*(5), 361-378.

Chaboche, J. L. (1989). Constitutive equations for cyclic plasticity and cyclic viscoplasticity. *International Journal of Plasticity*, *5*(3), 247-302.

Chaboche, J. L. (1991). On some modifications of kinematic hardening to improve the description of ratchetting effects. *International Journal of Plasticity*, *7*(7), 661-678.

Chase, S. B., Small, E. P., & Nutakor, C. H. R. I. S. (1999). An in-depth analysis of the national bridge inventory database utilizing data mining, GIS and advanced statistical methods. *Transportation Research Circular*, *498*, 1-17.

Chen, F. C., & Jahanshahi, M. R. (2018). NB-CNN: deep learning-based crack detection using convolutional neural network and naive Bayes data fusion. *IEEE Transactions on Industrial Electronics*, *65*(5), 4392-4400.

Chi, W. M., Kanvinde, A. M., & Deierlein, G. G. (2006). Prediction of ductile fracture in steel connections using SMCS criterion. *ASCE Journal of Structural Engineering*, *132*(2), 171-181.

Clifton, G. C., Nashid, H., Ferguson, G., Hodgson, M., Seal, C., Bruneau, M., ... & Gardiner, S. (2012). Performance of eccentrically braced framed buildings in the Christchurch earthquake series of 2010/2011. *Proceedings of 15th World Conference on Earthquake Engineering.* Lisbon, Portugal.

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,* Miami, Florida, United States.

Dimitriadis, E. K., Fuller, C. R., & Rogers, C. A. (1991). Piezoelectric actuators for distributed vibration excitation of thin plates. *Journal of Vibration and Acoustics*, *113*(1), 100-107.

Dworakowski, Z., Ambrozinski, L., Packo, P., Dragan, K., & Stepinski, T. (2015). Application of artificial neural networks for compounding multiple damage indices in Lamb-wave-based damage detection. *Structural Control and Health Monitoring*, *22*(1), 50-61.

Fell, B. V., Myers, A. T., Deierlein, G. G., & Kanvinde, A. M. (2006). Testing and simulation of ultra-low cycle fatigue and fracture in steel braces. *Proceedings of the*

8th National Conference on Earthquake Engineering. San Francisco, California, United States.

Frangopol, D. M., Kallen, M. J., & Noortwijk, J. M. V. (2004). Probabilistic models for life-cycle performance of deteriorating structures: review and future directions. *Progress in Structural Engineering and Materials*, *6*(4), 197-212.

Frangopol, D. M., Kong, J. S., & Gharaibeh, E. S. (2001). Reliability-based life-cycle management of highway bridges. *Journal of Computing in Civil Engineering*, *15*(1), 27-34.

Gatys, L. A., Ecker, A. S., & Bethge, M. (2015). A neural algorithm of artistic style. *arXiv:1508.06576*.

Gibson, A., & Popovics, J. S. (2005). Lamb wave basis for impact-echo method analysis. *ASCE Journal of Engineering mechanics*, *131*(4), 438-443.

Girshick, R. (2015). Fast r-cnn. *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile.

Giurgiutiu, V. (2005). Tuned Lamb wave excitation and detection with piezoelectric wafer active sensors for structural health monitoring. *Journal of Intelligent Material Systems and Structures*, *16*(4), 291-305.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, Sardinia, Italy.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge: MIT press.

Hajjar J. F., Sesen A. H., Jampole E., & Wetherbee, A. (2013). A synopsis of sustainable structural systems with rocking, self-centering, and articulated energy-dissipating fuses. Department of Civil and Environmental Engineering Reports. Report No. NEU-CEE-2013-01. Department of Civil and Environmental Engineering, Northeastern University, Boston, Massachusetts.

Hancock, J. W., & Mackenzie, A. C. (1976). On the mechanisms of ductile failure in high-strength steels subjected to multi-axial stress-states. *Journal of the Mechanics and Physics of Solids*, *24*(2-3), 147-160.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. New York: Springer-Verlag New York

Hayashi, T., & Kawashima, K. (2002). Multiple reflections of Lamb waves at a delamination. *Ultrasonics*, *40*(1-8), 193-197.

He, J., & Yuan, F. G. (2015). Damage identification for composite structures using a cross-correlation reverse-time migration technique. *Structural Health Monitoring*, *14*(6), 558-570.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, Las Vegas, Nevada, United States.

Heller, K., Jacobs, L. J., & Qu, J. (2000). Characterization of adhesive bond properties using Lamb waves. *NDT & E International*, *33*(8), 555-563.

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527-1554.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv:1207.0580*.

Huang, Y. H. (2010). Artificial neural network model of bridge deterioration. *ASCE Journal of Performance of Constructed Facilities*, *24*(6), 597-602.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167*.

Jacobsen, A., Hitaka, T., & Nakashima, M. (2010). Online test of building frame with slit-wall dampers capable of condition assessment. *Journal of Constructional Steel Research*, *66*(11), 1320-1329.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... & Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *Proceedings of the 22nd ACM International Conference on Multimedia*, Orlando, Florida, United States.

Kee, S. H., & Zhu, J. (2013). Using piezoelectric sensors for ultrasonic pulse velocity measurements in concrete. *Smart Materials and Structures*, *22*(11), 115016.

Kessler, S. S., Spearing, S. M., & Soutis, C. (2002). Damage detection in composite materials using Lamb wave methods. *Smart materials and structures*, *11*(2), 269-278.

Kim, H., Kim, H., Hong, Y. W., & Byun, H. (2017). Detecting Construction Equipment Using a Region-Based Fully Convolutional Network and Transfer Learning. *ASCE Journal of Computing in Civil Engineering*, *32*(2), 04017082.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Proceedings of Neural Information Processing Systems, Lake Tahoe, December 2012*. USA: NIPS

Kumar, S. S., Abraham, D. M., Jahanshahi, M. R., Iseley, T., & Starr, J. (2018). Automated defect classification in sewer closed circuit television inspections using deep convolutional neural networks. *Automation in Construction*, *91*, 273-283.

Leckey, C. A., & Parker, F. R. (2014). NDE and SHM Simulation for CFRP Composites. *Proceedings of 29th American Society for Composites Technical Conference*. La Jolla, CA, United States.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

Li, G., Zhao, X., Du, K., Ru, F., & Zhang, Y. (2017). Recognition and evaluation of bridge cracks with modified active contour model and greedy search-based support vector machine. *Automation in Construction, 78*, 51-61.

Li, S., Xu, J., & Tang, J. (2018). Tunable modulation of refracted lamb wave front facilitated by adaptive elastic metasurfaces. *Applied Physics Letters*, *112*(2), 021903.

Li, X., & Zhang, Y. (2008). Analytical study of piezoelectric paint sensor for acoustic emission-based fracture monitoring. *Fatigue & Fracture of Engineering Materials & Structures*, *31*(8), 684-694.

Li, X., & Zhang, Y. (2008). Feasibility study of wide-band low-profile ultrasonic sensor with flexible piezoelectric paint. *Smart Structures and Systems*, *4*(5), 565-582.

Li, Z., & Burgueño, R. (2010). Using soft computing to analyze inspection results for bridge evaluation and management. *ASCE Journal of Bridge Engineering*, *15*(4), 430-438.

Liang, Y., Wu, D., Liu, G., Li, Y., Gao, C., Ma, Z. J., & Wu, W. (2016). Big data-enabled multiscale serviceability analysis for aging bridges. *Digital Communications and Networks*, *2*(3), 97-107.

Lin, M., Chen, Q., & Yan, S. (2013). Network in network. *arXiv:1312.4400*.

Lopez, A. R., Giro-i-Nieto, X., Burdick, J., & Marques, O. (2017, February). Skin lesion classification from dermoscopic images using deep learning techniques. *Proceedings of 2017 13th IASTED International Conference on Biomedical Engineering*, Innsbruck, Austria.

Lu, Y., Ye, L., Su, Z., & Yang, C. (2008). Quantitative assessment of through-thickness crack size based on Lamb wave scattering in aluminum plates. *Ndt & E International*, *41*(1), 59-68.

Ma, J., Sheridan, R. P., Liaw, A., Dahl, G. E., & Svetnik, V. (2015). Deep neural nets as a method for quantitative structure–activity relationships. *Journal of Chemical Information and Modeling*, *55*(2), 263-274.

Madanat, S., Karlaftis, M., & McCarthy, P. (1997). Probabilistic infrastructure deterioration models with panel data. *ASCE Journal of Infrastructure Systems*, *3*(1), 4-9.

Madanat, S., Mishalani, R., & Ibrahim, W. (1995). Estimation of infrastructure transition probabilities from condition rating data. *ASCE Journal of Infrastructure Systems*, *1*(2), 120-125.

MATLAB (2018). MATLAB and Deep Learning Toolbox Release 2018a, The MathWorks, Inc., Natick, Massachusetts, United States.

McKenna, F., Fenves, G. L., & Scott, M. H. (2000). Open system for earthquake engineering simulation. *University of California, Berkeley, California, United States.*

Menegola, A., Fornaciali, M., Pires, R., Bittencourt, F. V., Avila, S., & Valle, E. (2017). Knowledge transfer for melanoma screening with deep learning. *Proceedings of 2017 IEEE 14th International Symposium on Biomedical*, Melbourne, Australia.

Micevski, T., Kuczera, G., & Coombes, P. (2002). Markov model for storm water pipe deterioration. *ASCE Journal of Infrastructure Systems*, *8*(2), 49-56.

Mirowski, P. W., LeCun, Y., Madhavan, D., & Kuzniecky, R. (2008). Comparing SVM and convolutional networks for epileptic seizure prediction from intracranial EEG. *Proceedings of IEEE Workshops on Machine Learning for Signal Processing*, Cancun, Mexico

Morcous, G. (2006). Performance prediction of bridge deck systems using Markov chains. *ASCE Journal of Performance of Constructed Facilities*, *20*(2), 146-155.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. *Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel.

Ni, Y. Q., Xia, Y., Liao, W. Y., & Ko, J. M. (2009). Technology innovation in developing the structural health monitoring system for Guangzhou New TV Tower. *Structural Control and Health Monitoring*, *16*(1), 73-98.

Park, H. W., Sohn, H., Law, K. H., & Farrar, C. R. (2007). Time reversal active sensing for health monitoring of a composite plate. *Journal of Sound and Vibration*, *302*(1-2), 50-66.

Petsounis, K. A., & Fassois, S. D. (2001). Parametric time-domain methods for the identification of vibrating structures—a critical comparison and assessment. *Mechanical Systems and Signal Processing*, *15*(6), 1031-1060.

Poddar, B., Kumar, A., Mitra, M., & Mujumdar, P. M. (2011). Time reversibility of a Lamb wave for damage detection in a metallic plate. *Smart Materials and Structures*, *20*(2), 025001.

Prinz, G. S., & Richards, P. W. (2009). Eccentrically braced frame links with reduced web sections. *Journal of Constructional Steel Research*, *65*(10-11), 1971-1978.

Qu, B., Liu, X., Hou, H., Qiu, C., & Hu, D. (2018). Testing of buckling-restrained braces with replaceable steel angle fuses. *ASCE Journal of Structural Engineering*, *144*(3), 04018001.

Raghavan, A., & Cesnik, C. E. (2007). Review of guided-wave structural health monitoring. *The Shock and Vibration Digest*, *39*(2), 91-114.

Ranjith, S., Setunge, S., Gravina, R., & Venkatesan, S. (2013). Deterioration prediction of timber bridge elements using the Markov chain. *Journal of Performance of Constructed Facilities*, *27*(3), 319-325.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Proceedings of Advances in Neural Information Processing Systems 28*, Montreal, Canada.

Rice, J. R., & Tracey, D. M. (1969). On the ductile enlargement of voids in triaxial stress fields. *Journal of the Mechanics and Physics of Solids*, *17*(3), 201-217.

Rose, J. L. (2014). *Ultrasonic guided waves in solid media*. New York, USA: Cambridge university press.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533-536.

Ruzzene, M. (2007). Frequency–wavenumber domain filtering for improved damage visualization. *Smart Materials and Structures*, *16*(6), 2116-2129.

Sabelli, R., Mahin, S., & Chang, C. (2003). Seismic demands on steel braced frame buildings with buckling-restrained braces. *Engineering Structures*, *25*(5), 655-666.

Sabelli, R., Roeder, C. W., & Hajjar, J. F. (2013). Seismic design of steel special concentrically braced frame systems. *NEHRP Seismic Design Technical Brief No. 8*, Gaithersburg, Maryland, United States.

Scott, M. H., & Fenves, G. L. (2009). Krylov subspace accelerated Newton algorithm: application to dynamic progressive collapse simulation of frames. *ASCE Journal of Structural Engineering*, *136*(5), 473-480.

Shen, Y., & Giurgiutiu, V. (2014). WaveFormRevealer: An analytical framework and predictive tool for the simulation of multi-modal guided wave propagation and interaction with damage. *Structural Health Monitoring*, *13*(5), 491-511.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Chen, Y. (2017). Mastering the game of Go without human knowledge. *Nature*, 550(7676), 354-359.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*.

Simonyan, K., Vedaldi, A., & Zisserman, A. (2013). Deep inside convolutional networks: visualizing image classification models and saliency maps. *arXiv:1312.6034*.

Sobanjo, J. O. (1997). A neural network approach to modeling bridge deterioration. *Proceedings of the 4th Congress on Computing in Civil Engineering*, ASCE, Reston, VA, United States.

Sohn, H., & Farrar, C. R. (2001). Damage diagnosis using time series analysis of vibration signals. *Smart Materials and Structures*, *10*(3), 446-451.

Sohn, H., & Lee, S. J. (2009). Lamb wave tuning curve calibration for surface-bonded piezoelectric transducers. *Smart Materials and Structures*, *19*(1), 015007.

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: the all convolutional net. *arXiv:1412.6806*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929-1958.

Su, Z., & Ye, L. (2004). Lamb wave-based quantitative identification of delamination in CF/EP composite structures using artificial neural algorithm. *Composite Structures*, *66*(1-4), 627-637.

Su, Z., Ye, L., & Lu, Y. (2006). Guided Lamb waves for identification of damage in composite structures: A review. *Journal of Sound and Vibration*, *295*(3-5), 753-780.

Sun, F., Xiao, B., & Zhang, Y. (2017). Acoustic emission signal characteristics of damage accumulation in non-buckling steel plate shear wall. *Special Issue: Proceedings of Eurosteel 2017, 1(2-3), 2444-2450.* Copenhagen, Denmark.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, Boston, Massachusetts, United States.

Tokdemir, O. B., Ayvalik, C., & Mohammadi, J. (2000). Prediction of highway bridge performance by artificial neural networks and genetic algorithms. *Proceedings of 17th International Symposium on Automation and Robotics in Construction*, Taipei, Taiwan.

Tong, L., Zhang, Y., Zhang, L., Liu, H., Zhang, Z., & Li, R. (2018). Ductility and energy dissipation behavior of G20Mn5QT cast steel shear link beams under cyclic loading. *Journal of Constructional Steel Research*, *149*, 64-77.

Tremblay, R., Filiatrault, A., Timler, P., & Bruneau, M. (1995). Performance of steel structures during the 1994 Northridge earthquake. *Canadian Journal of Civil Engineering*, *22*(2), 338-360.

Uriz, P. (2008). Toward earthquake-resistant design of concentrically braced steel-frame structures. *PEER Report 2008/08*, Pacific Earthquake Engineering Research Center.

Vanik, M. W., Beck, J. L., & Au, S. (2000). Bayesian probabilistic approach to structural health monitoring. *Journal of Engineering Mechanics*, *126*(7), 738-745.

Vargas, R., & Bruneau, M. (2009). Experimental response of buildings designed with metallic structural fuses. II. *ASCE Journal of Structural Engineering*, *135*(4), 394-403.

Wang, C. H., Rose, J. T., & Chang, F. K. (2004). A synthetic time-reversal imaging method for structural health monitoring. *Smart materials and structures*, *13*(2), 415-423.

Wellalage, N. K. W., Zhang, T., & Dwight, R. (2014). Calibrating Markov Chain–based deterioration models for predicting future conditions of railway bridge elements. *ASCE Journal of Bridge Engineering*, *20*(2), 04014060.

Wilcox, P. D., Lowe, M. J. S., & Cawley, P. (2001). Mode and transducer selection for long range Lamb wave inspection. *Journal of Intelligent Material Systems and Structures*, *12*(8), 553-565.

Xia, Y., & Hao, H. (2000). Measurement selection for vibration-based structural damage identification. *Journal of Sound and Vibration*, *236*(1), 89-104.

Xu, B., Wu, Z., Chen, G., & Yokoyama, K. (2004). Direct identification of structural parameters from dynamic responses with neural networks. *Engineering Applications of Artificial Intelligence*, *17*(8), 931-943.

Yang, F., & Mahin, S. A. (2005). Limiting net section failure in slotted HSS braces. *Structural Steel Education Council*, Moraga, California, United States.

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? *Proceedings of Advances in Neural Information Processing Systems 27*, Montreal, Canada.

Zang, C., Friswell, M. I., & Imregun, M. (2004). Structural damage detection using independent component analysis. *Structural Health Monitoring*, *3*(1), 69-83.

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. *Proceedings of European Conference on Computer Vision*, Zurich.

Zhang, Y., & Bai, L. (2015). Rapid structural condition assessment using radio frequency identification (RFID) based wireless strain sensor. *Automation in Construction*, *54*, 1-11.

Zhang, Y., Ayyub, B., & Huang, H. (2016). Enhancing Civil Infrastructure Resilience with Structural Health Monitoring. *Proceedings of 2016 International Workshop on Resiliency of Urban Tunnels. Reston, September 2016*. USA: ASCE.

Zhang, Z. (2015). Experimental Study on Hysteretic Behavior of Replaceable Cast Steel Shear Link Beams. Master Thesis, Department of Structural Engineering, Tongji University, Shanghai, China.

Zhou, C., & Zhang, Y. (2014). Near-field acoustic emission sensing performance of piezoelectric film strain Sensor. *Research in Nondestructive Evaluation*, *25*(1), 1-19.