

Fall 2012

Structural Damage Classification using Support Vector Machines

Xiang Li

Embry-Riddle Aeronautical University - Daytona Beach

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Aerospace Engineering Commons](#)

Scholarly Commons Citation

Li, Xiang, "Structural Damage Classification using Support Vector Machines" (2012). *Dissertations and Theses*. 92.

<https://commons.erau.edu/edt/92>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

STRUCTURAL DAMAGE CLASSIFICATION USING SUPPORT VECTOR MACHINES

by

Xiang Li

A thesis Submitted to the
Aerospace Engineering Department
In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Aerospace Engineering

Embry-Riddle Aeronautical University
Daytona Beach, Florida
Fall 2012

Structural Damage Classification using Support Vector Machines

by

Xiang Li

This thesis was prepared under the direction of the candidate's thesis committee chair, Dr. Daewon Kim, Department of Aerospace Engineering, and has been approved by the members of his thesis committee. It was submitted to the Department of Aerospace Engineering and was accepted in partial fulfillment of the requirements for the Degree of Master of Science in Aerospace Engineering

THESIS REVIEW COMMITTEE:



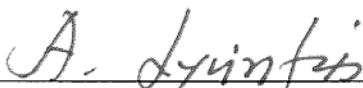
Dr. Daewon Kim
Committee Chair



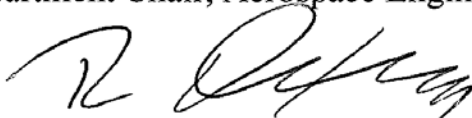
Dr. Frank J. Radosta
Committee Member



Dr. Yi Zhao
Committee Member



Dr. Tasos Lyrintzis
Department Chair, Aerospace Engineering



Dr. Robert Oxley
Associate Vice President of Academics

12-11-12

Date

ACKNOWLEDGEMENTS

I dedicate this thesis to my advisor, Dr. Daewon Kim. I would like express my enormous thanks to him for the huge time and efforts that he has devoted. His inspiring suggestions, patient teachings, and consistent encouragements have been crucial in the completion of this thesis.

I sincerely thank Dr. Yi Zhao and Dr. Frank Radosta for their assistance and feedback to this thesis. I also would like to thank Gregory Pike, Balaji Sivasubramanian, and Yi Zhang in the structures lab for kindly helping me in the process of this thesis.

Last but not least, I want to thank my beloved parents, Huizhuan and Chunrong, and my fiancé Meng for their continuous support and encouragement throughout my study in the college. Furthermore, I would like to thank my friends for sharing the best memories.

ABSTRACT

Researcher: Xiang Li

Title: Structural Damage Classification using Support Vector
Machine

Institution: Embry-Riddle Aeronautical University

Degree: Master of Science in Aerospace Engineering

Year: 2012

In this research, a methodology to classify crack and corrosion metallic damages using a time-frequency representation method and support vector machines is investigated. Piezoelectric ceramic actuators are utilized to generate guided wave signals on a set of aluminum beam coupons with different damage features, such as types, locations, and thicknesses. The short-time Fourier transform is applied to analyze the measured signals. For damage classification, the spectrograms obtained from finite element models are employed to train a two-class support vector machine learning classifier. The classifier is able to correctly classify different types of damages based upon the measured signals collected from the unknown damage sources. A multiple-class classifier is also generated to predict the damage extent of crack samples.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	iii
ABSTRACT.....	iv
TABLE OF CONTENTS.....	v
LIST OF TABLES.....	vii
LIST OF FIGURES.....	viii
LIST OF ABBREVIATIONS.....	xii
1. Introduction.....	1
1.1 Structural health monitoring.....	3
1.1.1 Process of Structural Health Monitoring.....	4
1.1.2 Challenges for Structural Health Monitoring.....	5
1.2 Structural Health Monitoring Methods.....	6
1.2.1 Guided Wave.....	7
1.2.2 Guided Wave Structural Health Monitoring.....	11
1.3 Transducer Technology.....	12
1.3.1 Piezoelectric Effect.....	12
1.3.2 Piezoelectric Materials.....	14
1.4 Literature Review.....	15
1.5 Damage Types in Metallic Materials.....	18
1.6 Research Objective.....	19
1.7 Thesis Outline.....	21
2. Support Vector Machines.....	23
2.1 Support Vector Machine Classification.....	24
2.2 Two-Class Support Vector Machine Classification.....	25
2.2.1 Linearly Separable Support Vector Machine.....	26
2.2.2 Non-linearly Separable Support Vector Machines.....	35
2.2.3 Dual problem.....	41
2.2.4 Decomposition method.....	47
2.2.5 Kernel Methods.....	49
2.2.6 Parameters Selection.....	51

2.3 Multiple-Class Support Vector Machine Classification.....	55
2.3.1 One-Against-The-Rest Strategy	55
2.3.2 One-Against-One Strategy	58
2.4 Chapter Recap	60
3. Time-Frequency Representations	61
3.1 TFR methods.....	61
3.1.1 Short Time Fourier Transform	62
3.1.2 Wigner-Ville Distribution.....	64
3.1.3 Wavelet Transform.....	66
3.2 TFR Method Selection	68
3.3 Chapter Recap	72
4. Damage Classification and Result Analysis	73
4.1 Training Samples using ABAQUS®	73
4.2 Feature Extraction	83
4.3 Two-Class Classification and Results.....	85
4.4 Multiple-Class Classification and Results.....	88
5. Conclusion and Recommendations.....	98
5.1 Summary and conclusion	98
5.2 Recommendations	100
References:.....	101

LIST OF TABLES

Table 1. 10 training samples consisting of 3 features	53
Table 2. Training samples divided into two subsets	53
Table 3. 6 two-class decision functions obtained in a 4-class SVM classification.....	59
Table 4. 6 predicted class labels obtained by substituting data x into the 6 different decision functions.....	59
Table 5. 64 training samples simulated using ABAQUS [®]	86
Table 6. 16 testing samples simulated using ABAQUS [®]	87
Table 7. Training samples for the 4-class classification	89
Table 8. Testing Samples for the 4-class classification	90
Table 9. The cross validation accuracy, ratio of correctly classified testing samples, and running time for various training cases	95
Table 10. The training samples for a three-class classifier including cracks of 2/8, 4/8, and 6/8 of the beam thickness	96
Table 11. Testing samples for a three-class classifier including cracks of 2/8, 4/8, and 6/8 of beam thickness	96

LIST OF FIGURES

Figure 1. Catastrophic accidents due to structural failures (a) the decompression of Aloha Airlines Flight 243 fuselage [1] (b) a collapsed bridge in China due to poorly controlled construction process [2].....	2
Figure 2. Structural health monitoring applications (a) a composite wing [6] (b) the Jindo bridge in South Korea [7]	3
Figure 3. Symmetric and anti-symmetric bounded waves (a) a symmetric Lamb wave (b) an anti-symmetric Lamb wave [18].....	9
Figure 4. Dispersion curves for lamb waves in an aluminum plate of thickness = 0.15-mm (a) phase velocity (b) group velocity [19].....	10
Figure 5. Molecule models for the piezoelectric effect (a) an unstressed molecule (b) a molecule exerted upon by an external stress (c) the polarizing effect of the material	13
Figure 6. Direct piezoelectric effect demonstrated by a piezoelectric material (a) the material in original shape (b) the material compressed (c) the material stretched	13
Figure 7. Converse piezoelectric effect demonstrated by a piezoelectric material (a) applied voltage of the same polarity as the material (b) applied voltage of the opposite polarity of the material	14
Figure 8. Schematic outline of the method presented in the research	21
Figure 9. Circles and triangles in an n-dimensional feature space	26
Figure 10. Linearly separable training data	27

Figure 11. The original separating straight line moving towards two directions (a) towards the circles (b) towards the triangles.....	29
Figure 12. Both the solid line and the dashed line fully separate all training samples.....	31
Figure 13. Definition of the margin for a SVM classifier.....	32
Figure 14. Projection of vector $(\mathbf{x}_1 - \mathbf{x}_2)$ to the perpendicular direction of the separating line.....	33
Figure 15. Linearly non-separable training samples in a two-dimensional space (a) the training samples cannot be separated by a straight line (b) the training samples fully separated by a curve.....	36
Figure 16. Definition of slack variables.....	38
Figure 17. The support vectors on a soft margin.....	39
Figure 18. Parameter selection using grid search [46].....	52
Figure 19. The first step to solve a multi-class classifier.....	55
Figure 20. The second step to solve a multi-class classifier.....	56
Figure 21. The final step to solve a multi-class classifier.....	57
Figure 22. The Wigner-Ville distribution at time t_0	65
Figure 23. A Mexican hat wavelet.....	66
Figure 24. The basic principle of Wavelet Transform.....	67
Figure 25. Two damage scenarios simulated in ABAQUS [®] (a) a crack scenario (b) a corrosion scenario.....	69
Figure 26. Plots of the sensed signal, damage signal, and the spectrogram of the damage signal (a) the sensed signal (b) the reflected damage signal only (c) the Spectrogram of the damage signal.....	70

Figure 27. Plots of the sensed signal, damage signal, and the Wigner-Ville distribution of the damage signal (a) the sensed signal (b) the reflected damage signal only (c) The Wigner-Ville distribution of damage signal.....	71
Figure 28. Plots of the sensed signal, damage signal, and the wavelet transform of the damage signal (a) the sensed signal (b) the reflected damage signal only (c) the wavelet transform of damage signal.....	72
Figure 29. The workstation utilized for running ABAQUS®	74
Figure 30. Cross sections of a crack and corrosion in a real metallic material and the ABAQUS simulations (a) a crack in a metallic material (b) a crack simulated in ABAQUS® (c) a piece of corrosion in a metallic material (d) a piece of corrosion simulated in ABAQUS®	75
Figure 31. ABAQUS® models of a crack and a piece of corrosion (a) a crack (b) a piece of corrosion.....	76
Figure 32. 50 kHz center frequency, Hanning windowed, tone-burst excited signal	76
Figure 33. The guided wave propagation in an ABAQUS model (scale: x600,000) (a) a crack and PZT actuators (b) the guided wave generated by the PZT actuators (c) the reflected wave and the residual of the excited wave	77
Figure 34. The sensed signal and its spectrogram (a) the sensed signal (b) STFT spectrogram of the sensed signal (d= 15-cm, 3/8 of the beam thickness).....	79
Figure 35. The reflected damage and its spectrogram (a crack of d= 20-cm, depth=3/8 of the beam thickness) (a) the reflected signal (b) the spectrogram of the reflected signal	80

Figure 36. Spectrograms of 4 crack signals of d=15-cm (a) a crack of 1/8 beam thickness (b) a crack of 2/8 beam thickness (c) a crack of 3/8 beam thickness (d) a crack of 4/8 beam thickness	81
Figure 37. Spectrograms of 4 crack signals of d=20-cm (a) a crack of 1/8 beam thickness (b) a crack of 2/8 beam thickness (c) a crack of 3/8 beam thickness (d) a crack of 4/8 beam thickness	82
Figure 38. The spectrogram of a corrosion signal and a magnified region from the spectrogram (a) the spectrogram (b) a magnified region from the spectrogram	84
Figure 39. The generation of a feature vector from a spectrogram	85
Figure 40. The locations of the maximum energy densities in two different spectrograms for the two-class classification (a) spectrogram of a crack (d=15-cm and depth=1/8 of beam thickness) (b) spectrogram of another crack (d=15-cm and depth =4/8 of beam thickness).....	91
Figure 41. Spectrograms of different crack signals (d=15-cm) for the multiple-class classification (a) the spectrogram of a crack (1/8 of the beam thickness) (b) the spectrogram of a crack (2/8 of the beam thickness) (c) the spectrogram of a crack (3/8 of the beam thickness) (d) the spectrogram of a crack (4/8 of the beam thickness)	93

LIST OF ABBREVIATIONS

Support Vector Machine	SVM
Structural Health Monitoring	SHM
Nondestructive Evaluation	NDE
Time Frequency Representation	TFR
Finite Element Analysis	FEA
Cross Validation	CV
Radial Bias Function	RBF
Short Time Fourier Transform	STFT
Surface Acoustic Wave	SAW
Lead Zirconate Titanate	PZT
Guided Wave	GW
Wavelet Transform	WT
Wigner-Ville Distribution	WVD
Matching Pursuit	MP
Electromagnetic Acoustic Transducer	EMAT
Polyvinylidene Fluoride	PVDF

Chapter 1

1. Introduction

In the past decades, developments in the transportation, infrastructure, and manufacturing industries have brought great advantages to humanity. However, catastrophic incidents of structural failure, such as bridge collapses and aircraft crashes, have caused huge losses of human lives and significant impacts to the economy. Figure 1 (a) depicts the decompression of the Aloha Airlines Flight 243 fuselage due to structural failure, and Figure 1 (b) shows a collapsed bridge due to a poorly controlled construction process in Harbin, a city of China, in August 24th, 2012. Such incidents have emphasized the importance of structural safety and promoted the development of damage detection and identification methods. One of these methods is nondestructive evaluation (NDE).



Figure 1. Catastrophic accidents due to structural failures (a) the decompression of Aloha Airlines Flight 243 fuselage [1] (b) a collapsed bridge in China due to poorly controlled construction process [2]

Nondestructive evaluation, as it turns out, is the examination of an object using technology without affecting the object's future usefulness [3]. An application of nondestructive evaluation in everyday life is the examination of a watermelon's ripeness. By visually inspecting flaws on the rind and listening to the acoustic response by tapping the surface, the ripeness of a watermelon can be judged without cutting it open. Some widely utilized nondestructive evaluation methods include visual inspection, acoustic emission, penetrant testing, magnetic field methods, radiography, eddy-current methods, and thermal field methods.

Since nondestructive evaluation methods can be conducted without affecting the tested objects' future functions, they are widely used in the construction, manufacturing, and transportation industries. However, the evaluation reliability can be affected by various factors. These factors include the physical and mental attributes of the inspector, working environment, and inevitable human errors introduced by the inspector. To

overcome these shortages of NDE, a novel damage identification approach called structural health monitoring has been developed.

1.1 Structural Health Monitoring

Structural health monitoring (SHM) is commonly defined as a process of implementing damage detection strategies for engineering infrastructures [4]. The development of SHM is generally motivated by economic and life-safety requirements. As a multidisciplinary research field, SHM involves various classical areas such as structural vibration, nondestructive evaluation, materials science, signal processing, sensor technology, etc. An SHM technique is generally integrated with sensors, data transmission, computational power, and processing ability inside the tested structures. The objective of SHM is to provide real time information and diagnosis about the state of safety and reliability of a structure during its lifetime [5]. Figure 2 (a) shows an SHM system integrated on a composite wing that calculates the residual fatigue life of the wing. Figure 2 (b) shows the Jindo bridge in South Korea which is equipped with one of the largest civil infrastructure SHM systems.

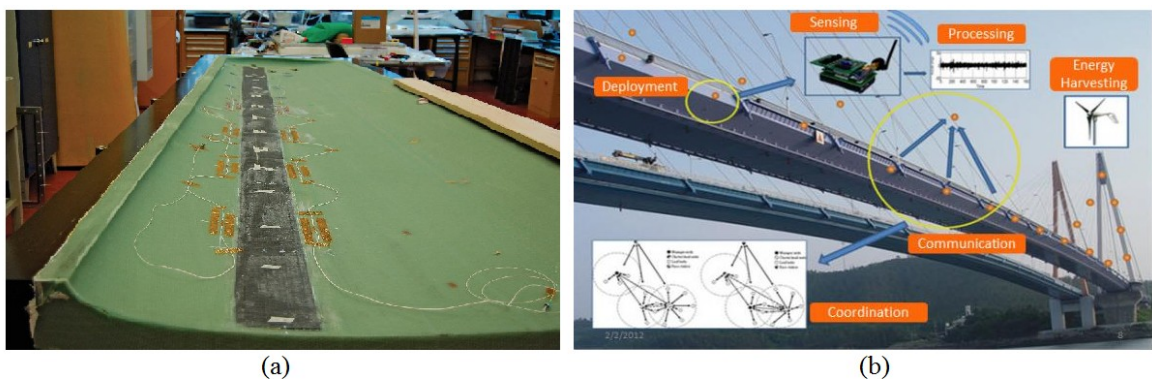


Figure 2. Structural health monitoring applications (a) a composite wing [6] (b) the Jindo bridge in South Korea [7]

SHM can be generally classified as passive or active. In passive SHM, a tested structure is equipped with sensors. The interaction between the tested structure and the surrounding environment is monitored by these sensors. On the contrary, in active SHM, a tested structure is equipped with both sensors and actuators. The actuators are used to generate perturbations to the tested structure, and the responses of the structure to these perturbations are monitored by the sensors. An active SHM method is applied in this research.

The development of SHM was primarily motivated by its economic potential and safety features. By applying SHM techniques, human errors induced by traditional damage detection methods can be eliminated. This improves the safety and reliability of the tested structures. Also, structure offline time and human labor involvement during maintenance can be minimized, and the maintenance cost reduced. In addition to this, the maintenance procedure of structural systems can be switched from conventional schedule-driven to condition-based.

1.1.1 Process of Structural Health Monitoring

SHM is commonly described as a process involving damage detection, diagnosis, and prognosis [8]. In the first step, existence and location of damage are evaluated. In the second step, damage type and damage extent are assessed. In the third step, the remaining useful life of a structural system is predicted.

Conjointly SHM can furthermore be described in terms of a four-step process. In the first step, also known as the operational evaluation step, the life-safety and economic criteria for performing SHM are evaluated to check if performing SHM to the system is

necessary. Damage is defined for the investigated system, and the most concerned damage types are determined for multiple damage cases.

In the second step, the excitation method, sensor type, number of sensors, sensor locations, data acquisition, data storage, and transmittal hardware are selected. Measured data from the sensors under various operational and environmental conditions are normalized and cleansed for further processing.

In the third step, data features that can be used to distinguish between undamaged and damaged structures are extracted. Since SHM diagnosis and prognosis technologies involve considerably more data than traditional damage detection methods, data condensation and reduction techniques are also required in this step.

In the final step, a statistical model is established to quantify the damage state of structures by implementing statistical pattern recognition algorithms. Pattern recognition algorithms are generally divided into two categories. The first category is known as supervised learning. In this category, samples from both undamaged and damaged state of a structure are available. In the second category, also known as unsupervised learning, samples of the damaged state in a structure are unknown. In this research, a supervised pattern recognition algorithm named support vector machines is utilized for damage classifications.

1.1.2 Challenges for Structural Health Monitoring

In recent years, SHM has become an attractive research topic in aerospace, civil and mechanical engineering. However, some challenges have been posed to the future development of SHM. The first challenge to the development of SHM is the difficulty in

capturing low-frequency responses of a system. Properties of a structural system, including the changes of stiffness, mass, or energy dissipation properties, can be altered by damages in the structure. The changes in the dynamic response of the structural system caused by these damages generally do not have a significant influence over the global response of the structure.

The second challenge is the difficulty in identifying damages in a system. Many SHM systems are based on an unsupervised learning mode, in which the samples of the damaged state are unknown. Damages may accumulate or evolve in a long-term monitoring process. It is challenging to precisely detect and track damages in long-term monitoring.

The third challenge comes from the requirement that the sensor network in a SHM system should be fail-safe. It implies that sensors should not get damaged after being deployed in a structure. Alternatively, if sensors are allowed to get damaged, a redundancy algorithm is supposed to be introduced to adapt to the new sensor network once one or several sensors are damaged.

Although difficulties are posed to the future developments of SHM technology, many SHM methods have been investigated and developed.

1.2 Structural Health Monitoring Methods

Various structural health monitoring methods have been developed in the past two decades. The impedance based method, the vibration based method, and the guided wave based method are some of the extensively investigated SHM approaches. The development of the vibration based method is based on the assumption that structural

modal parameters are functions of physical parameters [9-11]. Changes of physical parameters, such as mass, stiffness caused by damages, consequently alter the dynamic responses of the tested structure. These changes in the structural dynamic responses are used to identify damages in the tested structure. The drawback of the vibration based method is the modal parameters' low sensitivity to small flaws in a structure. In the impedance based method, physical changes in a tested structure induce changes in mechanical impedance of the tested structure [12-14]. These mechanical impedance changes are sensed by changes in the electro-mechanical coupling of transducers. In this research, an SHM method based on the guided wave method is utilized. To understand this method, an understanding of a guided wave is first required.

1.2.1 Guided Wave

Guided waves are commonly defined as stress waves that are forced to propagate along a path defined by the material boundaries of a structure [15-17]. A guided wave can be generally classified into two categories. The first category of guided wave is known as the surface acoustic wave (SAW). It was first discovered by Lord Rayleigh in 1885 [3]. A surface acoustic wave can propagate on the surface of an object for a very long distance, and its amplitude decays exponentially with the depth of the object. Different types of surface acoustic waves are defined according to the material nature at the interfaces. For example, a Rayleigh wave propagates along an interface between a semi-infinite solid medium and air; a Scholte wave travels along a boundary between water and a solid medium. Other types of surface acoustic waves include Stoneley waves, Love waves, and longitudinal creeping waves.

The second category of guided wave is known as the bounded wave because these waves are bounded between the boundaries of the finite medium. A bounded wave in a multilayer structure is defined as a plate wave, and bounded waves propagating in a single layer structure are referred as Lamb waves.

The shear wave mode and the longitudinal wave mode are the two modes that exist in ultrasonic guided waves. In the shear wave mode, motions of the medium's particles are perpendicular to the direction of wave propagation. In the longitudinal mode, the particles in the medium move parallel to the direction of wave propagation.

Guided waves can be generated using an angle beam transducer or a combo transducer [13-15]. In a plate structure, a longitudinal wave is generated by a transducer and hits the top surface of the plate. A longitudinal and a shear wave are reflected from the top surface. Four other waves (two longitudinal waves and two shear waves) are reflected when these two waves reach the bottom surface. This process is repeated, and infinite reflected waves interfere with each other to generate a resonant wave named the bounded wave. A bounded wave can either be symmetric or anti-symmetric about the center line of the plate structure. Figure 3 (a) describes a symmetric Lamb wave, while Figure 3 (b) shows an anti-symmetric Lamb wave.

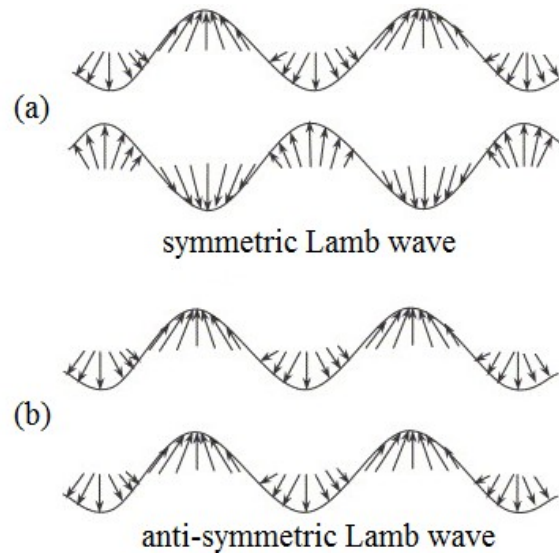


Figure 3. Symmetric and anti-symmetric bounded waves (a) a symmetric Lamb wave (b) an anti-symmetric Lamb wave [18]

Two wave modes that exist in guided waves are group velocity and phase velocity. Group velocity represents the speed of a wave packet that contains several individual waves. Phase velocity indicates the speed of an individual wave. Imagine that a group of people in a marathon race start the race from the same starting point at the same time. It appears that all of these runners are running at the same speed in the beginning. As time goes on, the group would stretch out because each individual runner is running at a different speed. The speed of an individual runner can be regarded as the phase velocity, and the group velocity could be regarded as the speed of the entire group of runners. Group velocity represents the velocity of energy propagation and is essentially used for damage detection in guided wave SHM.

Infinite numbers of symmetric wave modes S_x and anti-symmetric wave modes A_x exist in a bounded wave. Different wave modes have different group and phase velocity

characteristics. The velocity of bounded waves is related to the wave frequency, thickness of the plate medium, the properties of the medium material, and the wave mode. Velocities of various wave modes can be determined using dispersion curves.

Dispersion curves relate the velocity v and the product of plate thickness t and the wave frequency f . Figure 4 shows dispersion curves (phase and group velocity) of Lamb waves in an aluminum plate. It can be observed from the dispersion curves that at least two modes of waves can propagate at any frequency. In Figure 4 (b), waves with frequencies in the range of 0-10 MHz are more preferable for guided wave SHM because only two fundamental wave modes, S_0 and A_0 , exist in this region. The fundamental symmetric wave mode S_0 is mostly employed to detect surface cracks in metallic materials. The fundamental anti-symmetric mode A_0 is sensitive to delamination and is often used to detect damages in composites materials.

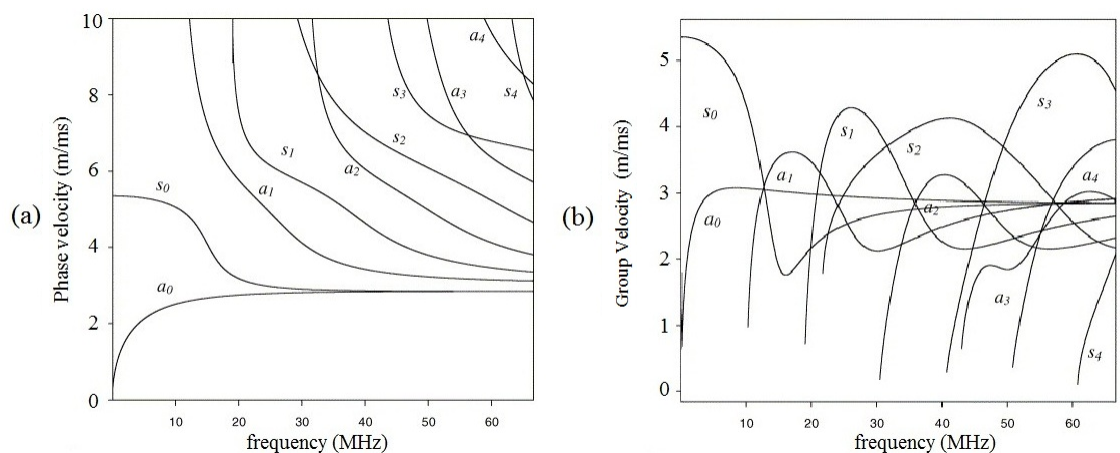


Figure 4. Dispersion curves for lamb waves in an aluminum plate of thickness = 0.15-mm

(a) phase velocity (b) group velocity [19]

1.2.2 Guided Wave Structural Health Monitoring

The guided wave (GW) based method is a well-established technology in traditional non-destructive evaluation [12]. It is more appropriate for far-field damage detection since the guided waves can be excited and propagated over a long distance with very little loss of amplitude. This feature gives the GW based SHM method the capability of a rapid global inspection of the structural health [17]. In the past two decades, the guided wave based method has been widely applied for damage detection in long structures, such as tubes, pipes, and plate structures.

In the guided wave SHM, an actuator is excited by a high frequency pulse signal to stimulate guided waves. When the guided waves encounter a structural discontinuity (such as damage or boundary in the structure), they are scattered in all directions. A sensor collocated with the actuator is utilized to sense echoes of the pulse signal reflected by the structural discontinuity (or damage). The excited signal is eliminated from the echoes to obtain the pure component of the damage signal. After that, damage sensitive features are extracted by applying some signal-processing algorithm. Finally, a pattern recognition technique is utilized to identify the damage type and extent.

In this research, a guided wave SHM method based on Lamb waves is utilized. A time-frequency representation technique is chosen to process damage signals. A pattern recognition algorithm named Support Vector Machine is applied for damage diagnosis. These techniques will be scrutinized in Chapter 2, 3, and 4.

1.3 Transducer Technology

Guided wave signals are generated by transducers from other forms of signals. For example, the piezoelectric transducer is used to transform electrical signals to guided wave signals, and the fiber optics transducer is used transform optical signals to guided wave signals. Some common materials used for producing transducers are piezoelectric ceramics, fiber composites, magnetostrictive materials, and so on. Piezoelectric transducers are the most extensively used transducers nowadays for generating and detecting guided waves. A specific type of piezoelectric transducer made from lead zirconate titanate (PZT) is used in this research. To understand the mechanisms of the piezoelectric transducer, the piezoelectric effect is investigated.

1.3.1 Piezoelectric Effect

The word “piezo” comes from the Greek word “piezeion”, which means to press or squeeze. The piezoelectric effect was first discovered by Jacques and Pierre Curie in 1880. They found that certain types of crystalline minerals became electrically polarized when subjected to mechanical stresses. The piezoelectric effect is generally known as the linear interaction between mechanical and electrical states. The molecular model of a piezoelectric material is depicted in Figure 5 to demonstrate this effect. The gravity centers of the positive and negative charges coincide before an external stress is exerted, as shown in Figure 5 (a), and the molecule appears electrically neutral. A separation of the gravity centers of positive and negative charges is caused by exerting an external stress, as shown in Figure 5 (b), and a dipole is generated. The neighboring dipoles inside a material cancel each other, and the charges on the material surfaces generate an electric field, as shown in Figure 5 (c). The material is polarized and an electric field is generated.

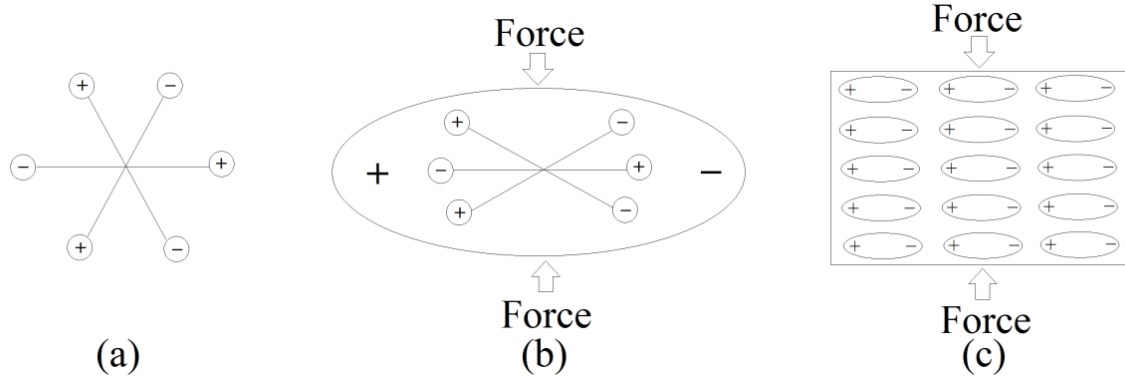


Figure 5. Molecule models for the piezoelectric effect (a) an unstressed molecule (b) a molecule exerted upon by an external stress (c) the polarizing effect of the material

As illustrated in Figure 6 (a) and (b), electric charges accumulate on material surfaces when piezoelectric ceramic materials are mechanically strained or deformed by external stresses. The polarities of the electric charges are reversed correspondingly if the external stress is reversed (from compression to tension) [20], as shown in Figure 6 (c). This phenomenon is called the direct piezoelectric effect.

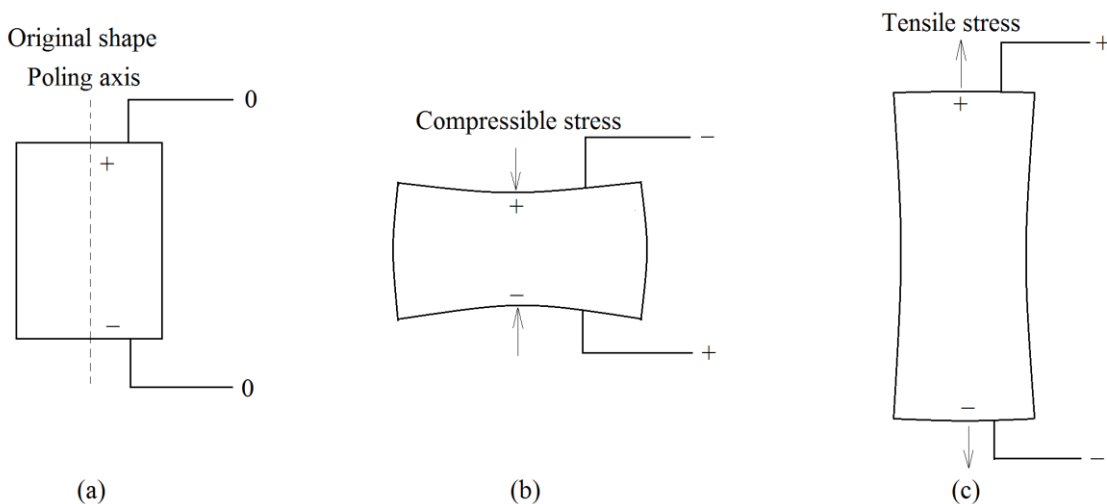


Figure 6. Direct piezoelectric effect demonstrated by a piezoelectric material (a) the material in original shape (b) the material compressed (c) the material stretched

In contrast to the direct effect, piezoelectric materials exhibit strain when placed in an electric field or when charges are applied. This phenomenon is called the converse piezoelectric effect. In Figure 7 (a), the material is stretched when the applied voltage polarity is same as the material polarity. In Figure 7 (b), the material compresses when the applied voltage's polarity is opposite to that of the material.

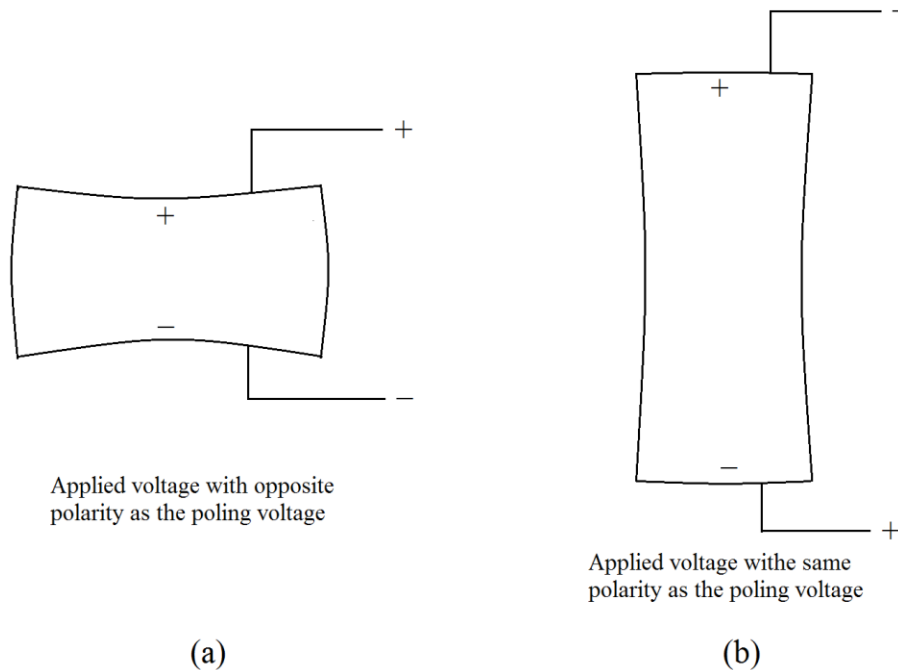


Figure 7. Converse piezoelectric effect demonstrated by a piezoelectric material (a) applied voltage of the same polarity as the material (b) applied voltage of the opposite polarity of the material

1.3.2 Piezoelectric Materials

Materials that exhibit the piezoelectric effect are named piezoelectric materials. Some naturally occurring piezoelectric materials include quartz, tourmaline, and sodium

potassium tartarate. Some common synthetic piezoelectric materials include piezoelectric ceramics and polyvinylidene fluoride (PVDF) [21].

One of the most widely utilized piezoelectric ceramic materials is lead zirconate titanate, also known as PZT. It was first created in the Tokyo Institute of Technology around the year 1952. The PZT material is made by combining lead and zirconium using a chemical compound called titanate. Compared to traditional piezoelectric materials, PZT exhibits greater sensitivity and a higher operating temperature. PZTs are widely used in ultrasonic nondestructive testing applications and accurate inspection of infrastructure and aerospace products [22].

Some new piezoelectric materials include single crystal and relaxor [22]. Single crystals can be produced using lead magnesium niobate, lead zirconate niobate, lithium niobate, and quartz. Some popular relaxor materials include lead magnesium niobate and lead nickel niobate. Relaxor materials are not sensitive to temperature, have high electromechanical coupling factors and are widely utilized to produce piezoelectric transducers.

1.4 Literature Review

Damage detection and identification methods using machine learning methods have been extensively investigated by researchers. Liu and Meng [23] developed a structural health monitoring system to detect approximate damage locations in beam structures using a vibration-based damage detection technique and a machine learning algorithm called Support Vector Machines (SVM). Their system was further developed to predict the damage location more accurately using support vector regression. Esterline et al. [24]

classified the acoustic emission signals generated by different sources, such as damage growth, using Support Vector Machines. Das et al. [25] developed an SHM system to classify damage signatures in composite structures. The system detected responses of surface-mounted piezoelectric transducers and used them to establish a classifier based on one-class support vector machines algorithm. Ying et al. [26] investigated a reliable SHM application to monitor damages within a pipeline using three classification algorithms including adaptive boosting, SVM, and a method called AdaSVM that combines the two methods. The system was robust enough to resist interferences like changes of internal air pressure in a pipe. The analysis result showed the AdaSVM algorithm achieved the best classification accuracy. Michaels and Michaels [27] studied a method to determine the exact location and the area extent of corrosion in an aluminum plate using Lamb waves generated by a sparse array of ultrasonic transducers. Dua et al. [28] developed a classification method to classify impact-induced damages in composite plates using the finite element analysis technique and artificial neural networks. Kim and Philen [29] investigated a damage classification method that classifies crack and corrosion using spectrogram and Adaboost. Their method was able to classify two different damages of crack and corrosion and showed confidence levels of each testing sample. Garpinteri et al. [30] investigated a methodology to identify damage in reinforced concrete structures and masonry buildings using acoustic emissions. They also developed a method to predict damage evolution status based upon fracture mechanics concepts. Yuan and Chu [31] developed a method for fault diagnosis of turbo-pump rotors. Their method utilized the Support Vector Machine “one against other” algorithm to classify vibration signals generated by different fault modes of rotors. Wang et al. [32] utilized

the least square support vector machine and acoustic signal to develop a damage signal classification method. Their method exhibited good performance in classifying different types of damage activities over oil transmission pipelines, such as drilling, hammering, and excavating. Widodo and Yang [33] investigated a method that can correctly diagnose faults in an induction machine based on transient current signals. The wavelet transform, Support Vector Machine, and feature extraction methods were utilized in their research. Banerjee and Das [34] developed a motor fault diagnosis method based on short time Fourier transforms and Support Vector Machine. Their method incorporated information collected from multiple sensors to detect and identify motor faults. Nguyen et al. [35] investigated a damage identification method for composite structures based on the distributed sensor network and the Bayesian network. In their method, the damage features used for identification include damage location, dimension, area, and damage signal characteristics. Coelho et al. [36] developed a methodology to classify the extent of growing cracks in lug joints that incorporated Support Vector Machine and guided-wave. In the method, matching pursuit algorithm is utilized to extract feature vectors from raw signals. Vines-Cavanaugh et al. [37] investigated a method to detect the abnormal status of a cable-stayed bridge in Zhanjiang, China based upon parameters collected from various sensors attached to the bridge. Their method utilized a Support Vector Machine algorithm called SVM20 for status classification and a finite element tool named ANSYS for generating training samples. Farooq et al. [38] developed a damage identification strategy for carbon fiber reinforced polymer composite materials. In their method, static strains sensed at predefined locations are measured in different damage scenarios and utilized as features in feature vectors. Support Vector Machine and

artificial neural networks are both employed to examine the classification accuracies. Xiao and Qu [39] investigated a damage detection method based on vibration based structural health monitoring and Support Vector Machine. Different damage states and positions were correctly identified in their method by analyzing vibration signals in different damage scenarios. Matic et al. [40] developed a system for broken bar detection in an electrical induction machine by analyzing current signals generated by the motor. The system was able to correctly identify the damage extent of the broken bar using Support Vector Machine and Hilbert transform. Xie [41] proposed a damage detection method for composite laminated plates. Least square support vector machine, guided-wave method, and Hilbert transform are utilized in his method. Yang [42] investigated a damage identification method using Support Vector Machine and the vibration based method. Random noises were added into training samples to obtain finite element models for consistency with the real testing sample. Damage location and its extent were able to be predicted by this method.

1.5 Damage Types in Metallic Materials

Damage is generally defined as changes introduced to a structural system that adversely affects its current or future performance [4]. Two common damage types within metallic materials are crack and corrosion. The presence of a crack in aerospace structures is often caused by cyclic loading, such as pressurizations and depressurizations of a fuselage or cyclic impact loadings on the landing gear. Catastrophic consequences are usually incurred if an initial crack evolves to a critical size. Therefore, crack detection is vital for the aerospace industry.

Corrosion within a metallic material usually involves reactions between the material and the environment. The corrosion process in nature is usually an electrical process. The cause of corrosion in nature is because an unstable refined metal tends to transform to a stable compound. The common types of corrosion include uniform, pitting, localized, intergranular, galvanic erosion, and fretting.

As previously mentioned, crack detection is vital to flight safety in the aerospace industry, while corrosion generally accounts for a large portion of the maintenance cost. Therefore, a reliable damage classification method for distinguishing between a crack and corrosion exhibits great potential value in the aerospace industry.

1.6 Research Objective

A great amount of research has been conducted on damage detection and diagnosis in various materials. Many of these researches have investigated various methods to distinguish different types of damages. Some researchers developed approaches to determine damage locations. Some others investigated methods to monitor the initiation of damage. However, it appears that not much attention has been paid to damage extent evaluation using guided-wave based SHM method and multiple-class SVM classification algorithms. As previously mentioned, damage diagnosis, including classification and extent estimation, is an important step in structural health monitoring. In this step, vital information is extracted for predicting the remaining life of structures. Hereby, the first objective of this research is to explore and develop a robust damage identification method that can be used to correctly classify crack and corrosion in metallic materials. The second objective is to extend the method to evaluate crack evolution extent.

The method can be described as a 5-step process. In the first step, the guided-wave based structural health monitoring method is used. The guided wave excited by PZT actuators propagates along an aluminum beam. Some portions of the guided wave are reflected by damage on the beam, and the reflected damage signal is collected by measuring the in-plane displacement of a fixed location on the beam. In the second step, the raw damage signal is processed using a short time Fourier transform to generate a time-frequency representation image. The third step is about feature extraction. In this step, STFT images generated in various damage scenarios are scanned and processed to generate training and testing data. In the fourth step, training data are utilized to construct a support vector machine classifier. In the final step, the testing data is validated using the classifier, and the performance and accuracy of the classifier is examined. The damage diagnosis method presented in the thesis is schematically outlined in Figure 8.

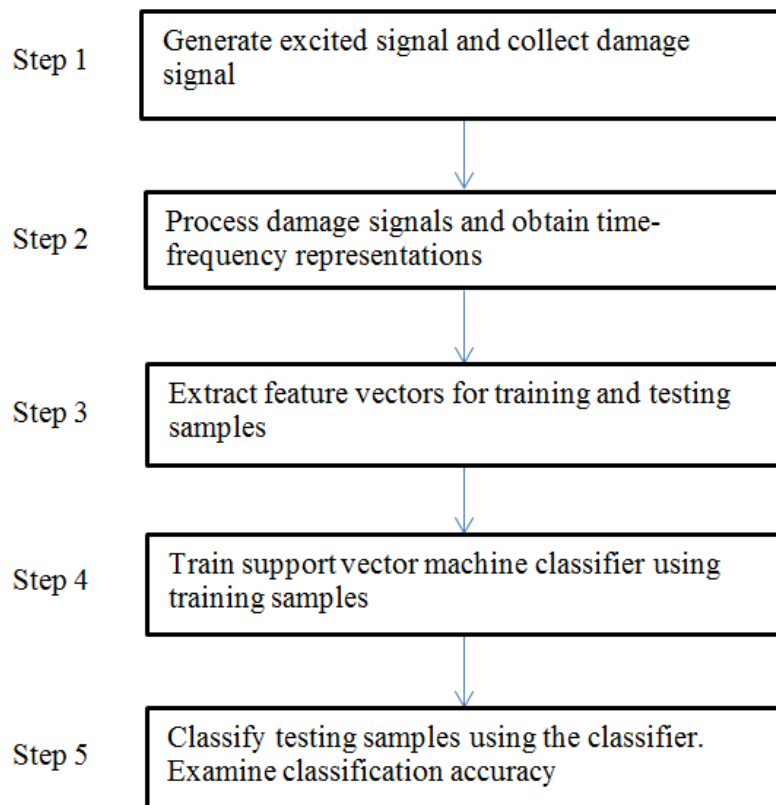


Figure 8. Schematic outline of the method presented in the research

1.7 Thesis Outline

The thesis consists of 5 chapters that are organized as follows. In the first chapter, background knowledge about structural health monitoring is introduced. Literature reviews about damage detection and diagnosis using SHM methods are presented.

In the second chapter, theory about two-class and multiple-class support vector machine classification has been comprehensively discussed.

In the third chapter, several time-frequency analysis methods, including short time Fourier transform, Wigner-Ville distribution, Wavelet transform, and Matching Pursuit are investigated. The time-frequency plots using different methods are compared with one another.

In the fourth chapter, damage samples simulated in finite element analysis software named ABAQUS[®] are presented. The damage signals obtained from ABAQUS[®] simulations are transformed into time-frequency representations. A two-class damage classification program is developed to predict damage type (crack or corrosion). A multiple-class damage classification method is also investigated to estimate crack evolution extent. The performance and accuracy of these classification methods are examined.

Finally, the fifth chapter summarizes the research and presents recommendations for future work.

Chapter 2

2. Support Vector Machines

In human history, a huge number of inventions have been created to mimic human or animal behavior. However, very few inventions have been created to simulate the function of the human brain because of its extremely complex working mechanism. However, the invention of the first computer system in 1945 made the simulation of the human brain possible for the first time. Since then, artificial intelligence has been extensively used in almost every field. Machine learning is a branch of artificial intelligence. It is a system that learns from experience, training, and analytic observation, and it continuously self-improves [43]. Some popular machine learning algorithms include decision tree learning, artificial neural networks, and Bayesian networks.

In a machine learning-pattern classification problem, a classifier is developed to correctly classify an object with promising accuracy. Inputs to the classifier are called features, and outputs from the classifier are called class labels. Features are discreetly chosen so that they represent data of different classes distinguishably. In other words, features are chosen such that data belonging to different classes are well separated in the feature space.

The essence of a pattern classification problem is to solve for the decision function. Determination of the decision function using input and output pairs is called training. Predicting the output or class label of an input using the decision function is called testing. The input-output pairs utilized to determine the decision function are called training samples and the data that are input to the decision function are called testing samples.

As mentioned in Section 1.2.2, a pattern recognition technique is utilized in a guided wave structural health monitoring method. In this research, a specific pattern recognition algorithm named Support Vector Machine (SVM) is utilized. The underlying theory of SVM was introduced by Cortes and Vapnik based upon the statistical learning theory [43]. SVM has been extensively applied to pattern classification and regression. It has several obvious advantages compared to conventional pattern classification methods. First, SVM classification can be conducted based on a very small amount of training samples [44]. Also, the feature space of SVM can be high dimensional, and SVM exhibits excellent generalization performance and high accuracy over a wide range of classification problems [45]. Finally, unlike some traditional classification methods, SVM does not suffer from problems like over-fitting and local minimization. Since SVM outperforms conventional pattern classification methods, it is chosen as the classification algorithm for this research. The fundamentals of SVM classification are illustrated in the following sections.

2.1 Support Vector Machine Classification

As previously mentioned, a classifier is generated using input-output pairs in machine learning pattern classification. For SVM classification, an input is also known as

a feature vector, and an output is known as a class label. A feature vector describes an object in each dimension of the feature space. In other words, the feature of an object is used to characterize the particular object from other objects. The label of an object, on the other hand, represents the class that the object belongs to.

A simple example is shown here to demonstrate the SVM classification. In this example, a classifier is generated to classify two types of fruits: apples and oranges. The classification is a two-step process that includes training and prediction. In the training step, the feature vectors and the class labels of training fruits are utilized to generate a classifier. The features of a fruit would include shape, size, color, weight, and smell. The Label of a fruit is either “apple” or “orange”. In the testing step, the feature vector of an unknown fruit is given as the input to the classifier. The classifier predicts whether this fruit is an apple or an orange.

2.2 Two-Class Support Vector Machine Classification

As discussed in the previous section, a SVM classifier is generated using training samples. Each training sample is represented by a feature vector and a class label. A decision function is generated using the feature vectors and class labels of the training samples. The class label of an unknown sample can be obtained by inputting its feature vector into the decision function. In this section, the underlying theory of SVM classification is investigated. Furthermore, the procedures to solve the decision function are demonstrated in detail.

2.2.1 Linearly Separable Support Vector Machine

The equations of SVM classification are commonly derived from a Euclidean point of view, as shown in Figure 9.

In the figure, training samples (circles and triangles) are distributed in a three dimensional feature space. The features of the training sample are its x, y, and z axis coordinates. The objective of SVM classification is to find a separating plane that fully separates all the circles from the triangles.

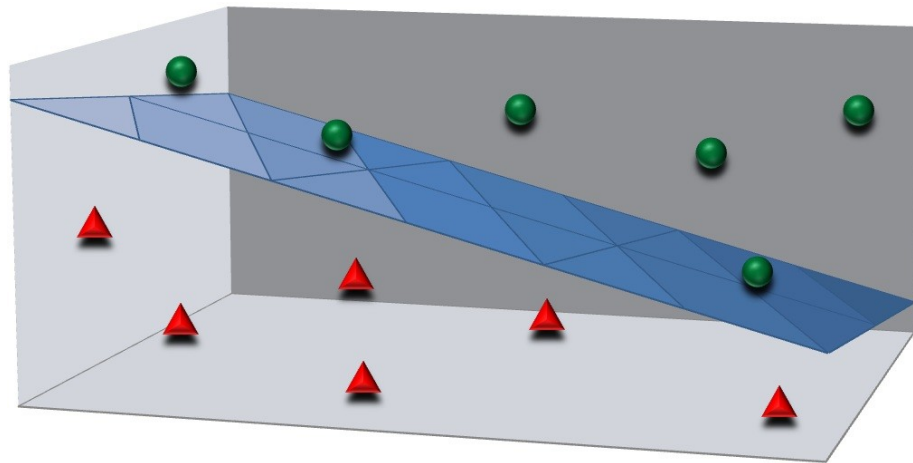


Figure 9. Circles and triangles in an n-dimensional feature space

In an n-dimensional space, each training sample has n number of feature elements in feature vector: $\mathbf{x}_i = \{x_1, x_2, \dots, x_n\}$ ($i = 1, \dots, l$). “ l ” is the amount of training samples. The scalar “ x_i ” in the feature vector represents the i^{th} axis coordinate of the training sample. Each training sample is assigned with a class label y_i .

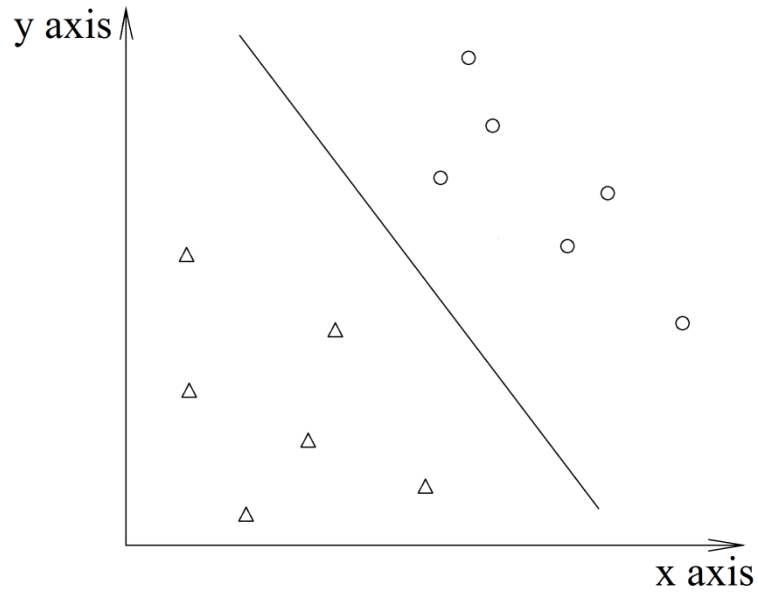


Figure 10. Linearly separable training data

For simplicity, a two-dimensional problem is considered as shown in Figure 10. In the figure, circles are defined as Class 1, and triangles are defined as Class 2. The label of each class is defined as follows:

$$y_i = 1 \text{ for Class 1 (circles)}$$

$$y_i = -1 \text{ for Class 2 (triangles)}$$

The objective of SVM classification is to find a separating straight line that fully separates all the circles from the triangles. The separating straight line in a two-dimensional feature space satisfies

$$\mathbf{w}^T \cdot \mathbf{x} + b = 0 \tag{2.1}$$

In this equation, \mathbf{w} is a two-dimensional vector called “weight”, and “ b ” is a scalar called bias. Mathematically, \mathbf{w} is a vector normal to the straight line, and $\frac{b}{\|\mathbf{w}\|}$ is the distance from the straight line to the origin. \mathbf{w}^T is the transposed vector of \mathbf{w} . \mathbf{x} is a vector called the feature vector, which contains all the features of a training sample.

Since circles are in the upper right region to the separating straight line in Figure 10, they should satisfy Eq. (2.2).

$$\mathbf{w}^T \cdot \mathbf{x} + b > 0 \quad (2.2)$$

Similarly, the triangles in the lower left region of the separating straight line satisfy Eq. (2.3),

$$\mathbf{w}^T \cdot \mathbf{x} + b < 0 \quad (2.3)$$

Therefore, Eq. (2.2) and Eq. (2.3) can be written in a more compact form as below.

$$\mathbf{w}^T \cdot \mathbf{x} + b > 0 \text{ for circles} \quad (2.4)$$

$$\mathbf{w}^T \cdot \mathbf{x} + b < 0 \text{ for triangles} \quad (2.5)$$

As previously mentioned, the label of a circle is $y_i = 1$, and the label of a triangle is $y_i = -1$. Therefore, Eq. (2.4) and Eq. (2.5) can be modified as shown below.

$$\mathbf{w}^T \cdot \mathbf{x} + b > 0 \text{ if } y_i = 1 \quad (2.6)$$

$$\mathbf{w}^T \cdot \mathbf{x} + b < 0 \text{ if } y_i = -1 \quad (2.7)$$

Furthermore, Eq. (2.6) and (2.7) can be combined into one equation as shown in Eq. (2.8).

$$(\mathbf{w}^T \cdot \mathbf{x} + b)y_i > 0 \quad (i=1, \dots, l) \quad (y_i = +1 \text{ or } -1) \quad (2.8)$$

The original separating straight line is named H . H is moved in a parallel direction towards the circles until it meet a circle. This new line is named as H_1 , as shown in Figure 11 (a). The equation of H_1 is expressed in Eq. (2.9).

$$\mathbf{w}^T \cdot \mathbf{x} + b = k_1 \quad (2.9)$$

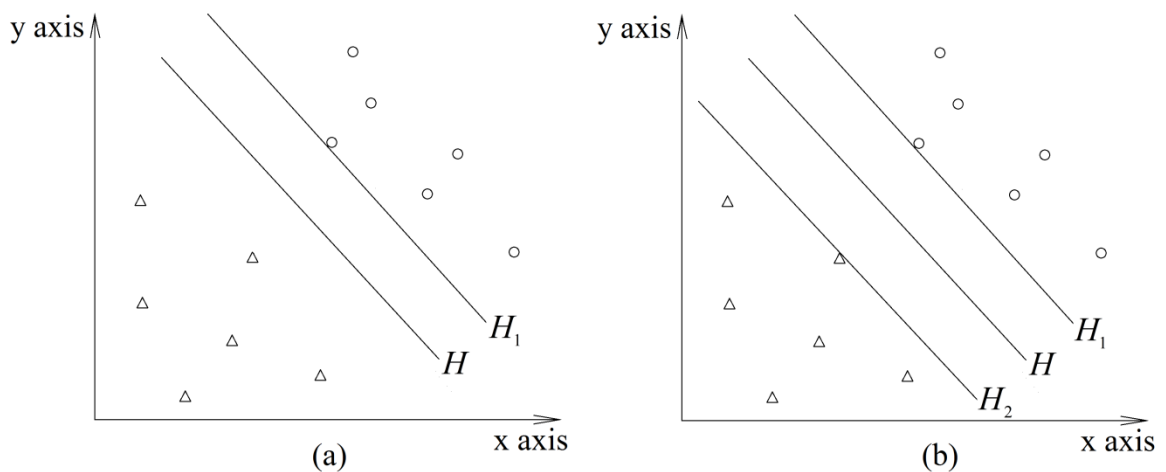


Figure 11. The original separating straight line moving towards two directions (a) towards the circles (b) towards the triangles

Similarly, the straight line H is moved towards the triangles in parallel until it touches the first triangle. This straight line is named as H_2 , as shown in Figure 11 (b). The equation of H_2 is given by Eq. (2.9).

$$\mathbf{w}^T \cdot \mathbf{x} + b = k_2 \quad (2.9)$$

In these two equations, k_1 and k_2 are constant scalars. By constraining the straight line H in the middle of H_1 and H_2 , k_1 and k_2 are forced to be equal to each other.

$$k_1 = k_2 = k \quad (2.10)$$

Then, Eq. (2.8) and Eq. (2.9) are modified as shown

$$\begin{aligned} \mathbf{w}^T \cdot \mathbf{x} + b &= k \text{ for } H_1 \\ \mathbf{w}^T \cdot \mathbf{x} + b &= -k \text{ for } H_2 \end{aligned} \quad (2.11)$$

By dividing both sides by k , Eq. (2.11) is transformed.

$$\begin{aligned} \mathbf{w}'^T \cdot \mathbf{x} + b' &= 1 \text{ for } H_1 \\ \mathbf{w}'^T \cdot \mathbf{x} + b' &= -1 \text{ for } H_2 \end{aligned} \quad (2.12)$$

The difference between Eq. (2.11) and Eq. (2.12) is just a matter of scaling.

Therefore, the equations for the straight lines H_1 and H_2 are simplified as

$$\begin{aligned} \mathbf{w}^T \cdot \mathbf{x} + b &= 1 \text{ for } H_1 \\ \mathbf{w}^T \cdot \mathbf{x} + b &= -1 \text{ for } H_2 \end{aligned} \quad (2.13)$$

The circles or triangles falling on H_1 and H_2 are called support vectors. To ensure that all circles are in the upper region of H_1 , a circle \mathbf{x} must satisfy the constraint in Eq.

(2.14)

$$\mathbf{w}^T \cdot \mathbf{x} + b > 1 \text{ if } y_i = 1 \quad (2.14)$$

Similarly, to ensure all triangles are in the lower region of H_2 , a triangle \mathbf{x} must satisfy Eq. (2.15)

$$\mathbf{w}^T \cdot \mathbf{x} + b < -1 \text{ if } y_i = -1 \quad (2.15)$$

Eq. (2.14) and Eq. (2.15) are combined as a single equation, as shown in Eq. (2.16).

$$(\mathbf{w}^T \cdot \mathbf{x} + b)y_i > 1 \quad (i = 1, \dots, l) \quad (y_i = +1 \text{ or } -1) \quad (2.16)$$

Eq. (2.16) is known as the constraint of a SVM classifier. As previously mentioned, the objective of a SVM classification is to find a separating straight line which fully separates all the training instances. However, there exists more than one separating straight line that fully separates all the circles from the triangles. For example, both the solid line and the dashed line in Figure 12 fully separate the triangles from the circles. Therefore, the optimal separating line needs to be determined.

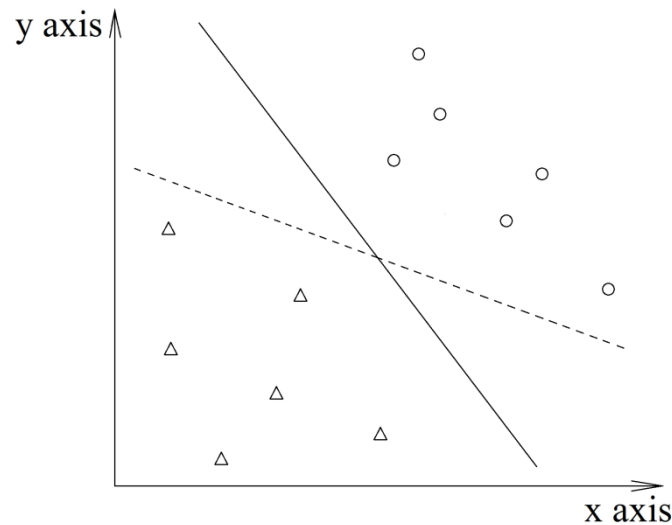


Figure 12. Both the solid line and the dashed line fully separate all training samples

To decide the optimal separating line, the concept of margin is introduced. The distance between the separating line and its nearest training sample is called a margin. More concisely, the margin is defined as the distance between the lines H_1 and H_2 , as

shown in Figure 13. The margin in a linearly-separable SVM classifier is also known as a hard margin. A contrast between a hard margin and a soft margin will be given in the next section.

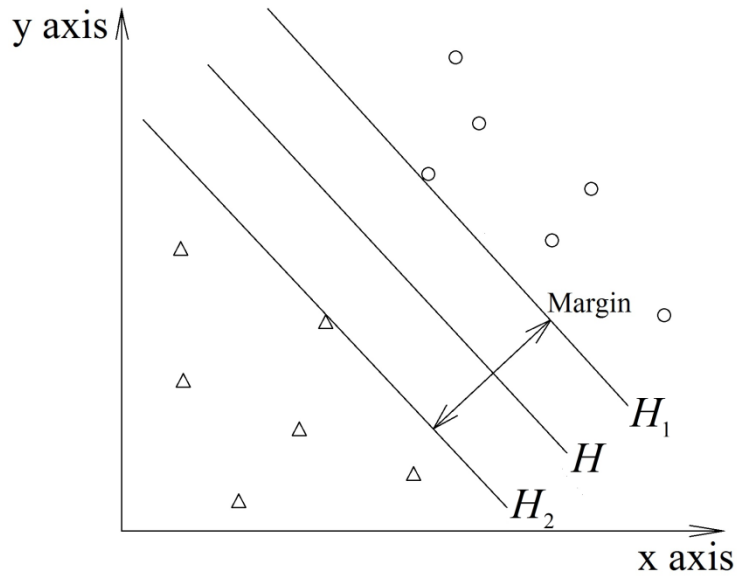


Figure 13. Definition of the margin for a SVM classifier

A random point \mathbf{x}_1 on H_1 and a random point \mathbf{x}_2 on H_2 are chosen, as shown in Figure 14. Due to \mathbf{x}_1 being on the H_1 , it should satisfy the equation of H_1

$$\mathbf{w}^T \cdot \mathbf{x}_1 + b = 1 \quad (2.17)$$

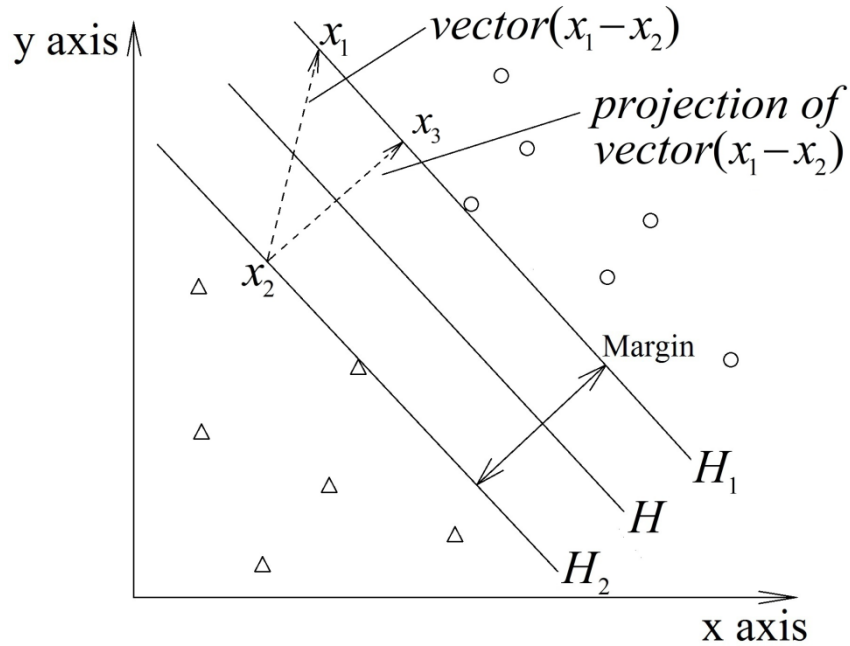


Figure 14. Projection of vector $(\mathbf{x}_1 - \mathbf{x}_2)$ to the perpendicular direction of the separating line

Similarly, \mathbf{x}_3 should satisfy the equation of H_1

$$\mathbf{w}^T \cdot \mathbf{x}_3 + b = +1 \quad (2.18)$$

Assume the distance between H_1 and H_2 (or the margin) is m . Vector $(\mathbf{x}_3 - \mathbf{x}_2)$ is perpendicular to the straight line H , and it is the product of the margin m and the normal vector of \mathbf{w} . Therefore, the equation of vector $(\mathbf{x}_3 - \mathbf{x}_2)$ is $m \frac{\mathbf{w}}{\|\mathbf{w}\|}$. Vector \mathbf{x}_3 is obtained by adding vector \mathbf{x}_2 and vector $(\mathbf{x}_3 - \mathbf{x}_2)$

$$\mathbf{x}_3 = \mathbf{x}_2 + (\mathbf{x}_3 - \mathbf{x}_2) = \mathbf{x}_2 + m \frac{\mathbf{w}}{\|\mathbf{w}\|} \quad (2.19)$$

By substituting Eq. (2.19) into Eq. (2.18)

$$\mathbf{w}^T \cdot \mathbf{x}_3 + b = \mathbf{w}^T \cdot \left(\mathbf{x}_2 + m \frac{\mathbf{w}}{\|\mathbf{w}\|} \right) + b = \mathbf{w}^T \cdot \mathbf{x}_2 + b + m \mathbf{w}^T \cdot \frac{\mathbf{w}}{\|\mathbf{w}\|} = +1 \quad (2.20)$$

Due to \mathbf{x}_2 being on the H_2 , it should satisfy $\mathbf{w}^T \cdot \mathbf{x}_2 + b = -1$. By substituting this equation into Eq. (2.20)

$$\begin{aligned} -1 + m \mathbf{w}^T \cdot \frac{\mathbf{w}}{\|\mathbf{w}\|} &= +1 \\ m \mathbf{w}^T \cdot \frac{\mathbf{w}}{\|\mathbf{w}\|} &= +2 \\ m &= \frac{2 \|\mathbf{w}\|}{\mathbf{w}^T \cdot \mathbf{w}} = \frac{2 \sqrt{\mathbf{w}^T \cdot \mathbf{w}}}{\mathbf{w}^T \cdot \mathbf{w}} = \frac{2}{\sqrt{\mathbf{w}^T \cdot \mathbf{w}}} = \frac{2}{\|\mathbf{w}\|} \end{aligned} \quad (2.21)$$

As previously illustrated, the objective of the SVM classifier is to maximize the margin. It is equivalent to minimize the reciprocal of the margin, as shown in Eq. (2.22)

$$\text{reciprocal of } \frac{2}{\|\mathbf{w}\|} = \frac{\|\mathbf{w}\|}{2} = \frac{\sqrt{\mathbf{w}^T \cdot \mathbf{w}}}{2} \quad (2.22)$$

Minimizing $\frac{\sqrt{\mathbf{w}^T \cdot \mathbf{w}}}{2}$ is equivalent to minimizing $\frac{\mathbf{w}^T \cdot \mathbf{w}}{2}$. Therefore, the objective

of the SVM classifier in Eq. (2.22) can be transformed to

$$\min \varphi(\mathbf{w}) = \frac{\mathbf{w}^T \cdot \mathbf{w}}{2} \quad (2.23)$$

Thus, the optimization problem for the SVM classifier, as shown in Eq. (2.24), is obtained by combining the constraint in Eq. (2.16) and the objective in Eq. (2.23).

$$\begin{aligned} \min \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} \\ \text{subject to } y_i(\mathbf{w}^T \cdot \mathbf{x} + b) y_i \geq 1 (i=1, \dots, l) \quad (y_i = +1 \text{ or } -1) \end{aligned} \quad (2.24)$$

The decision function $f(\mathbf{x})$ can be obtained by solving for \mathbf{w} and b from Eq. (2.24).

$$f(\mathbf{x}) = \text{sgn}(\mathbf{w}^T \cdot \mathbf{x} + b) \quad (2.25)$$

The “sgn” function in Eq. (2.55) is called the sign function. The class label of an unknown sample \mathbf{x}_u can be determined by substituting \mathbf{x}_u into the decision function in Eq. (2.25). A positive value returned from the decision function indicates that the unknown sample \mathbf{x}_u is above the optimal separating line H , as shown in Eq. (2.26). This implies that \mathbf{x}_u is classified as a circle by the classifier.

$$\mathbf{w}^T \cdot \mathbf{x}_u + b > 0 \quad (2.26)$$

Similarly, an unknown sample is classified to be a triangle if the decision function returns a negative value, i.e.

$$\mathbf{w}^T \cdot \mathbf{x}_u + b < 0 \quad (2.27)$$

So far, the linearly-separable two-class SVM classifier has been illustrated in a Euclidean point of view. A more complicated case called non-linearly separable SVM is introduced.

2.2.2 Non-linearly Separable Support Vector Machines

As mentioned in the previous section, training samples of a linearly separable SVM classifier can be fully separated by a separating line. In the real world, however, training

samples are not often linearly separable. An example of linearly non-separable training samples in a two-dimensional feature space is shown in Figure 15. In the figure, the triangles are generally distributed within the group of circles. Therefore, it is not possible to simply use a straight line to fully separate the circles from the triangles. A solution to completely separate those samples is by using a curve, as illustrated in Figure 15 (b).

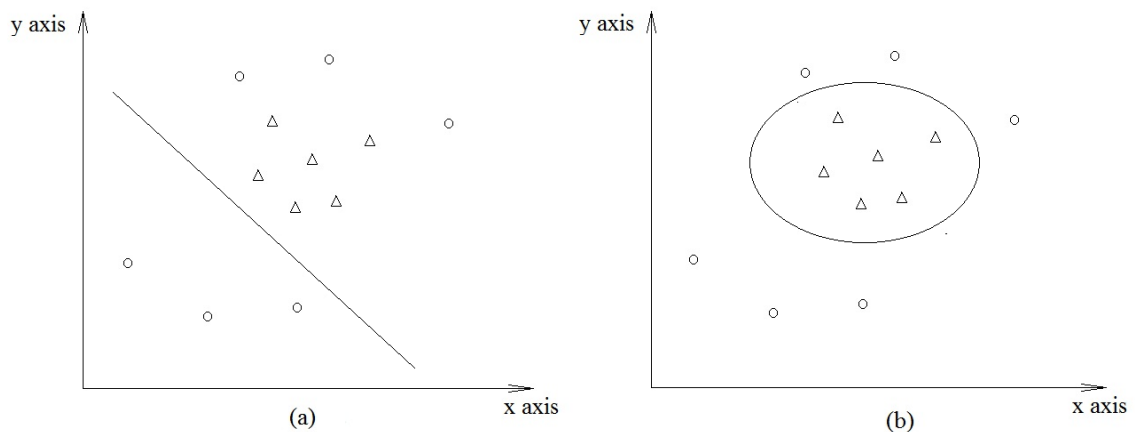


Figure 15. Linearly non-separable training samples in a two-dimensional space (a) the training samples cannot be separated by a straight line (b) the training samples fully separated by a curve

Obviously, the SVM classifier for such training samples cannot be solved using the linear-separable solution demonstrated in the previous section. Two strategies are introduced for solving this type of classifier.

The first strategy is mapping data samples in a low-dimensional feature space to a high-dimensional feature space. By doing so, linearly non-separable data samples in the low-dimensional space are likely to become linearly-separable in the high-dimensional feature space. They are hence called a nonlinearly separable sample.

An example of mapping is illustrated here. Data sample \mathbf{x} is in a three-dimensional feature space ($\mathbf{x} \in R^3$). The vector \mathbf{x} has three components, as shown in Eq. (2.28).

$$\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\} \quad (2.28)$$

A mapping function $\phi(\mathbf{x})$ is defined in Eq. (2.29). After mapping, the vector \mathbf{x} is transformed to a new vector \mathbf{x}' . The new vector \mathbf{x}' is in a 6-dimensional feature space, and it has 6 components. In the new feature space, features of the original data samples are combined to generate new features. This is because a single feature may not provide sufficient information for solving a SVM classifier. However, by combining features together, it may provide more feature information to separate all the training samples.

$$\mathbf{x}' = \phi(\mathbf{x}) = \{\mathbf{x}_1, \sqrt{2}\mathbf{x}_2, \sqrt{3}\mathbf{x}_3, \mathbf{x}_1\mathbf{x}_2, \mathbf{x}_1\mathbf{x}_3, \mathbf{x}_2\mathbf{x}_3\} (\mathbf{x}' \in R^6) \quad (2.29)$$

By mapping, all the data samples are mapped into a higher-dimensional feature space. The SVM classifier can now be solved in the new feature space. As previously mentioned, Eq. (2.24) is the optimization problem for solving the SVM classifier. By introducing a mapping function $\phi(\mathbf{x})$, Eq. (2.24) is transformed into a new optimization problem in a high-dimensional feature space, as illustrated in Eq. (2.30).

$$\begin{aligned} \min \varphi(\mathbf{w}) &= \frac{\mathbf{w}^T \cdot \mathbf{w}}{2} \\ \text{subject to } &y_i(\mathbf{w}^T \cdot \phi(\mathbf{x}) + b) y_i \geq 1 (i = 1, \dots, l) \end{aligned} \quad (2.30)$$

As discussed in the previous section, training samples are linearly-separable in hard-margin support vector machines. In the real world, however, training samples may still be not separable even though they are transferred to a high-dimensional feature space.

It implies that no feasible solution of a SVM classifier can be obtained by using a hard margin. Therefore, the second strategy is to introduce a soft margin and allow training errors.

A hard margin is illustrated in Figure 16. In the figure, most circles are in the upper-right region to H_1 , except for “Circle1”. Similarly, triangles are in the lower-left region to H_2 , with the exemption of “Triangle1”. Obviously, Circle 1 and Triangle 1 do not satisfy Eq. (2.31). Therefore, Circle 1 and triangle 1 are known as training errors. It implies that a hard-margin classifier does not correctly separate all the training samples.

$$(\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b)y_i \geq 1 \quad (2.31)$$

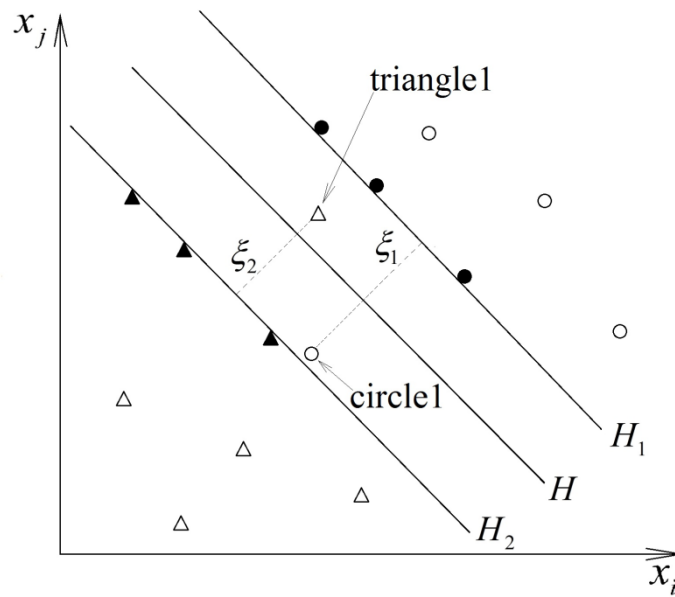


Figure 16. Definition of slack variables

To allow training errors, nonnegative slack variables ξ_i are introduced. In Figure 16, slack variable ξ_1 indicates the distance from the H_1 to Circle 1. ξ_2 represents the

distance between H_2 and Triangle 1. By doing so, Triangle 1 and Circle 1 are accepted as support vectors, as demonstrated by the filled circles and triangles in Figure 17.

To ensure Circle 1 and Triangle 1 are correctly separated by the classifier, they must satisfy the constraint of the optimization problem. Hence, the original constraint in Eq. (2.31) is modified to Eq. (2.32).

$$(\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b)y_i \geq 1 - \xi_i \quad (2.32)$$

As shown in Figure 16, the filled circles and triangles represent the support vectors of the classifier. For a hard margin case, all the filled circles (or triangles) are distributed on the classifier's boundaries. Therefore, the margin of the classifier appears to be hard and solid. In a soft margin case, as shown in Figure 17, the support vectors are distributed in a curved-surface, and the margin appears to be soft.

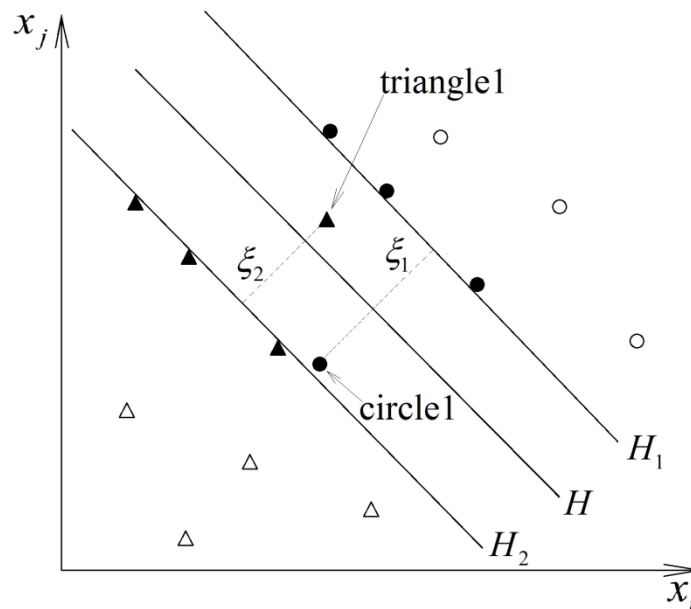


Figure 17. The support vectors on a soft margin

As previously mentioned, slack variables are introduced to allow training errors. However, not too many training errors are expected in a SVM classifier because most of the training samples are supposed to be correctly separated by the classifier. Therefore, the optimization problem in Eq. (2.30) is transformed into Eq. (2.33) by adding the summation of the slack variables.

$$\min_{\mathbf{w}, b, \xi} \varphi(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} + C \sum_{i=1}^l \xi_i \quad (2.33)$$

In Eq. (2.33), the constant C is called the penalty term. It is a trade-off between the maximization of the classifier margin and the minimization of the training errors. By changing the value of C , the extent of allowable training errors can be adjusted. As previously discussed, the objective of a SVM classifier is to minimize the function $\varphi(\mathbf{w})$ in Eq. (2.33). If C is adjusted to a large value, most of the slack variables ξ_i should be very small to ensure $\varphi(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} + C \sum_{i=1}^l \xi_i$ in Eq. (2.33) is minimized. In contrast, if the penalty term C is adjusted to a relatively small number, slack variables are allowed to become relatively large.

Up to now, the optimization problem in Eq. (2.33) and the constraint in Eq. (2.32) could be combined to solve the SVM classifier, as shown in Eq. (2.34).

$$\begin{aligned} \min_{\mathbf{w}, b, \xi} \quad & \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} + C \sum_{i=1}^l \xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i (\xi_i \geq 0, i = 1, \dots, l) \end{aligned} \quad (2.34)$$

This Eq. (2.34) is also known as the primal problem of a SVM classifier. Once \mathbf{w} and b are solved from Eq. (2.34), they are substituted into Eq. (2.35) to construct the decision function.

$$f(\mathbf{x}) = \text{sgn}(\mathbf{w}^T \cdot \phi(\mathbf{x}) + b) \quad (2.35)$$

2.2.3 Dual problem

As previously discussed, the decision function of a SVM classifier in Eq. (2.35) can be obtained by solving for \mathbf{w} and b from the primal problem in Eq. (2.34). However, solving the primal problem is not an easy task. This is because the training samples in Eq. (2.34) are mapped into a high-dimensional feature space using a mapping function $\phi(\mathbf{x})$. Therefore, the vector \mathbf{w} has the same dimension as the new feature space. Commonly, the dimension of the vector \mathbf{w} is too high to conduct any traditional optimization method to solve it from Eq. (2.34). So, the primal problem in Eq. (2.34) is usually transformed to a dual problem. Then, \mathbf{w} and b are solved from the dual problem to generate a decision function.

For simplicity, the derivation of the dual problem is based on the primal problem without the slack variables

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} \\ \text{subject to} \quad & y_i (\mathbf{w}^T \cdot \phi(\mathbf{x}) + b) y_i \geq 1 (i = 1, \dots, l) \end{aligned}$$

A new equation is obtained by moving the right hand side of the second equation given above to the left hand side, as shown in Eq. (2.37)

$$y_i(\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b) - 1 \geq 0 \quad (2.36)$$

To derive the dual problem, the Lagrange dual is introduced as shown in Eq. (2.37). The Lagrange dual is generated by subtracting the first equation in Eq. (2.35) by the left hand side of Eq. (2.36).

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^l \alpha_i (y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) - 1) (\alpha_i \geq 0) \quad (2.37)$$

In Eq. (2.37), \mathbf{w} and b are the weight vector and the bias from the original primal problem in Eq. (2.34) where α_i is called the nonnegative Lagrange multiplier. According to the Karush-Kuhn-Tucker (KKT) conditions, the minimum of the primal problem ($\min_{\mathbf{w}, b, \xi} \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w}$) can be solved in a two-step process. In the first step, the minimum of the Lagrange dual is found with respect to \mathbf{w} and b . In the second step, the maximum of Lagrange dual is found with respect to the Lagrange multiplier $\boldsymbol{\alpha}$. This criterion is illustrated in Eq. (2.38).

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} = \max_{\boldsymbol{\alpha} > 0} \{ \min_{\mathbf{w}, b} L(\mathbf{w}, b, \boldsymbol{\alpha}) \} \quad (2.38)$$

First Step

In the first step for solving Eq. (2.38), $\boldsymbol{\alpha}$ is assumed to be a fixed constant. The Lagrange dual in Eq. (3.37) is transformed to Eq. (2.39) by eliminating the parentheses.

$$\begin{aligned} L(\mathbf{w}, b, \boldsymbol{\alpha}) &= \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^l \alpha_i (y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) - 1) \\ &= \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^l \alpha_i y_i \mathbf{w}^T \phi(\mathbf{x}_i) - b \sum_{i=1}^l \alpha_i y_i + \sum_{i=1}^l \alpha_i \end{aligned} \quad (2.39)$$

If $\sum_{i=1}^l \alpha_i y_i > 0$, the Lagrange dual in Eq. (2.39) is equal to $-\infty$ by making $b = +\infty$. If

$\sum_{i=1}^l \alpha_i y_i < 0$, the Lagrange dual in Eq. (2.39) is also equal to $-\infty$ by making $b = -\infty$, as

shown in Eq. (2.40)

$$\min_{\mathbf{w}, b} L(\mathbf{w}, b, \boldsymbol{\alpha}) = -\infty \text{ if } \sum_{i=1}^l \alpha_i y_i \neq 0 \quad (2.40)$$

Then, a more complicated case is considered if $\sum_{i=1}^l \alpha_i y_i = 0$. In this case, it can be

observed that the term $b \sum_{i=1}^l \alpha_i y_i$ in Eq. (2.39) is equal to 0. The minimum of the Lagrange

dual with respect to \mathbf{w} and b (Lagrange multiplier $\boldsymbol{\alpha}$ is assumed to be fixed) in Eq. (2.39)

can be simplified, as shown in Eq. (2.41).

$$\begin{aligned} \min_{\mathbf{w}, b} L(\mathbf{w}, b, \boldsymbol{\alpha}) &= \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^l \alpha_i y_i \mathbf{w}^T \phi(\mathbf{x}_i) + \sum_{i=1}^l \alpha_i \\ &= \min_{\mathbf{w}, b} \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} - \sum_{i=1}^l \alpha_i (y_i \mathbf{w}^T \phi(\mathbf{x}_i) - 1) \end{aligned} \quad (2.41)$$

Now, the minimization of the Lagrange dual can be written in a new form as shown in Eq. (2.42).

$$\min_{\mathbf{w}, b} L(\mathbf{w}, b, \boldsymbol{\alpha}) = \min_{\mathbf{w}, b} \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} - \sum_{i=1}^l \alpha_i (y_i \mathbf{w}^T \phi(\mathbf{x}_i) - 1) \quad (2.42)$$

The Lagrange dual $L(\mathbf{w}, b, \boldsymbol{\alpha})$ in Eq. (2.41) is a strict convex function about the vector \mathbf{w} . It implies $\min_{\mathbf{w}, b} L(\mathbf{w}, b, \boldsymbol{\alpha})$ can be obtained when the partial derivative of $L(\mathbf{w}, b, \boldsymbol{\alpha})$ with respect to \mathbf{w} is equal to 0, as shown in Eq. (2.43).

$$\frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0 \quad (2.43)$$

Eq. (2.43) is expanded as

$$\frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \cdot 2\mathbf{w} - \sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i) = 0 \quad (2.44)$$

The value of vector \mathbf{w} can be obtained by solving Eq. (2.44)

$$\mathbf{w} = \sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i) \quad (2.45)$$

Eq. (2.45) implies that \mathbf{w} is a linear combination of the training instances (\mathbf{x}_i and y_i) and the Lagrange multiplier (α_i). Till now, the two conditions for $\sum_{i=1}^l \alpha_i y_i > 0$ and $\sum_{i=1}^l \alpha_i y_i < 0$ have both been investigated. By substituting the expression of \mathbf{w} in Eq. (2.45) into the right hand side of Eq. (2.42), the Lagrange dual can be simplified as shown in Eq. (2.46).

$$\begin{aligned} & \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} - \sum_{i=1}^l \alpha_i [y_i \mathbf{w}^T \phi(\mathbf{x}_i) - 1] \\ &= \frac{1}{2} \left[\sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i) \right]^T \left[\sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i) \right] - \sum_{i=1}^l \alpha_i y_i \left[\sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i) \right]^T \phi(\mathbf{x}_i) + \sum_{i=1}^l \alpha_i \quad (2.46) \\ &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \end{aligned}$$

By combining Eq. (2.40) and (2.46), Eq. (2.38) can be rewritten as shown in Eq. (2.47).

$$\max_{\alpha > 0} \left\{ \min_{w, b} L(w, b, \alpha) \right\} = \max_{\alpha \geq 0} \left\{ \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \text{ if } \sum_{i=1}^l \alpha_i y_i = 0 \quad -\infty \text{ if } \sum_{i=1}^l \alpha_i y_i \neq 0 \right\} \quad (2.47)$$

Second Step

In the second step for solving Eq. (2.38) or Eq. (2.47), the maximum of the Lagrange dual needs to be found with respect to the Lagrange multiplier α . In Eq. (2.47), minus infinity is obviously not the maximum of the dual problem. Therefore, the maximum of the Lagrange dual problem occurs only if $\sum_{i=1}^l \alpha_i y_i = 0$, as illustrated in Eq. (2.48).

$$\max_{\alpha > 0} \left\{ \min_{w, b} L(w, b, \alpha) \right\} = \max_{\alpha > 0} \left\{ \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \text{ if } \sum_{i=1}^l \alpha_i y_i = 0 \right\} \quad (2.48)$$

For simplicity, $\sum_{i=1}^l \alpha_i y_i$ is converted into vector form, as shown in Eq. (2.49).

$$\sum_{i=1}^l \alpha_i y_i = \mathbf{y}^T \cdot \boldsymbol{\alpha} \quad (2.49)$$

Then, Eq. (2.48) is simplified by substituting Eq. (2.49), as shown in Eq. (2.50).

$$\begin{aligned} & \max_{\alpha \in R^l} \left\{ \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \right\} \\ & \text{subject to } \mathbf{y}^T \cdot \boldsymbol{\alpha} = 0 (\alpha_i \geq 0, i = 1, \dots, l) \end{aligned} \quad (2.50)$$

To further simplify Eq. (2.50), a unit vector is defined in Eq. (2.51).

$$\mathbf{e} = [1, \dots, 1]^T \quad (2.51)$$

Then, $\sum_{i=1}^l \alpha_i$ can be written as the product of \mathbf{e} and $\boldsymbol{\alpha}$ in vector form, as shown in Eq.

(2.52).

$$\sum_{i=1}^l \alpha_i = \mathbf{e} \cdot \boldsymbol{\alpha} \quad (2.52)$$

A matrix Q is defined as shown in Eq. (2.53).

$$Q_{ij} = y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \quad (2.53)$$

The Lagrange dual can be simplified by substituting Eq. (2.51), Eq. (2.52), and Eq. (2.53) into Eq. (2.50), as shown in (2.54).

$$\begin{aligned} & \max_{\alpha \in R^l} \left\{ \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \right\} \\ & = \max_{\alpha \in R^l} \mathbf{e} \cdot \boldsymbol{\alpha} - \frac{1}{2} \boldsymbol{\alpha}^T Q \boldsymbol{\alpha} \\ & = \min_{\alpha \in R^l} \frac{1}{2} \boldsymbol{\alpha}^T Q \boldsymbol{\alpha} - \mathbf{e} \cdot \boldsymbol{\alpha} \end{aligned} \quad (2.54)$$

The final format of the SVM dual problem after all simplifications is shown in Eq. (2.55).

$$\begin{aligned} \min_{\alpha \in R^l} \quad & \frac{1}{2} \boldsymbol{\alpha}^T Q \boldsymbol{\alpha} - \mathbf{e} \cdot \boldsymbol{\alpha} \\ \text{subject to} \quad & \mathbf{y}^T \cdot \boldsymbol{\alpha} = 0 \quad (\alpha_i \geq 0, i = 1, \dots, l) \end{aligned} \quad (2.55)$$

As mentioned at the beginning of this section, slack variables ξ_i and the penalty term are neglected in the derivations. The final form of the dual problem is shown in Eq. (2.56) considering all these terms.

$$\begin{aligned} \min_{\alpha \in R^l} \quad & \frac{1}{2} \boldsymbol{\alpha}^T Q \boldsymbol{\alpha} - \mathbf{e} \cdot \boldsymbol{\alpha} \\ \text{subject to} \quad & \mathbf{y}^T \cdot \boldsymbol{\alpha} = 0 \quad (0 \leq \alpha_i \leq C, i = 1, \dots, l) \end{aligned} \quad (2.56)$$

The Lagrange multipliers α_i can be solved from Eq. (2.56). The method for solving Lagrange multipliers will be illustrated in next section. Once the Lagrange multipliers are obtained, they are substituted into the Eq. (2.45) to generate the weight \mathbf{w} .

$$\mathbf{w} = \sum_{i=1}^l y_i \alpha_i \phi(\mathbf{x}_i) \quad (2.45)$$

The new decision function, as shown in Eq. (2.57), is obtained by substituting the expression of \mathbf{w} into the decision function in Eq. (2.35).

$$\begin{aligned} & \text{sgn}(\mathbf{w}^T \cdot \phi(\mathbf{x}) + b) \\ & = \text{sgn}\left(\sum_{i=1}^l y_i \alpha_i \phi(\mathbf{x}_i) \phi(\mathbf{x}) + b\right) \end{aligned} \quad (2.57)$$

2.2.4 Decomposition Method

Since it is often difficult to directly solve \mathbf{w} from the primal problem, the primal problem of SVM classification is transformed to the dual problem. Computer programs

are employed to solve the Lagrange multiplier α_i from the dual problem. However, an immediate difficulty to solve the dual problem in Eq. (2.56) is that Q is a $l \times l$ fully dense matrix, as shown in Eq. (2.58).

$$Q_{ij} = y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \quad (i=1, \dots, l \text{ and } j=1, \dots, l) \quad (2.58)$$

As an example, consider that a classification problem has 30,000 training samples ($l=30,000$). The Q matrix in this problem is a $30,000 \times 30,000$ fully dense matrix. It takes $30,000 \times 30,000 \times 8/2$ bytes (about 3GB) of computer RAM to store this matrix. Normal personal computers usually do not have free memory space of this size. Therefore, it is difficult to store the matrix Q in a personal computer.

A solution to overcome this difficulty is the decomposition method. Instead of directly solving the entire vector α from the dual problem, the decomposition method works only on a subset of the Lagrange multiplier α per iteration. In this way, only a few columns of the Q matrix are needed per iteration. The indexes of the subset are defined as a working set B . The indexes that are not in the working set B are defined as an idle set $N = \{1, \dots, l\} - B$. Using this partition method, the Lagrange multiplier vector α is divided into two sub vectors as $[\alpha_B, \alpha_N]$. Similarly, the Q matrix is divided into 4 sub matrixes including Q_{BB} , Q_{BN} , Q_{NB} , and Q_{NN} , as shown in Eq. (2.59).

$$\min_{\alpha_B} \frac{1}{2} [\alpha_B^T \quad \alpha_N^T] \begin{bmatrix} Q_{BB} & Q_{BN} \\ Q_{NB} & Q_{NN} \end{bmatrix} \begin{bmatrix} \alpha_B \\ \alpha_N \end{bmatrix} - \begin{bmatrix} e_B^T & e_N^T \end{bmatrix} \begin{bmatrix} \alpha_B \\ \alpha_N \end{bmatrix} \quad (2.59)$$

Subject to $0 \leq \alpha_t \leq C, t \in B, y_B^T \alpha_B + y_N^T \alpha_N = 0$

An iteration solution is utilized in the decomposition method. At each iteration, the Lagrange multipliers in the idle set ($\alpha_k, k \in N$) are treated as constants. Only those Lagrange multipliers in the working set ($\alpha_k, k \in B$) are treated as variables and solved in a sub optimization problem. In this way, Eq. (2.59) can be further simplified to a sub problem shown in Eq. (2.60) by merging all the constant terms.

$$\begin{aligned} \min_{\alpha_B} \quad & \frac{1}{2} \alpha_B^T Q_{BB} \alpha_B + (-e_B \alpha_B + Q_{BN} \alpha_N) + \text{constant} \\ \text{Subject to} \quad & 0 \leq \alpha_t \leq C, t \in B, y_B^T \alpha_B + y_N^T \alpha_N = 0 \end{aligned} \quad (2.60)$$

In an extreme situation, the working set B contains only two objects: i and j. Therefore, only two Lagrange multipliers need to be solved per iteration. Compared with the original problem in Eq. (2.56), this sub problem is a much smaller optimization problem and can be solved using a common personal computer. Objects in the working set B are different per iteration. The iteration process is terminated if all the Lagrange multipliers are solved. The composition method exhibits promising convergence and is adopted by the SVM classification software.

2.2.5 Kernel Methods

As discussed in section 2.2.3, mapping functions are introduced to the SVM classification to transfer training and testing samples to a high-dimensional feature space. In this section, kernel functions are introduced to replace mapping functions. This is because the kernel calculation is more effective than the mapping function. The basic format of a kernel is defined in Eq. (2.61). In the equation, the inner product of the two mapping functions can be calculated by the kernel function $K(\mathbf{x}, \mathbf{y})$ using only one

calculation. Therefore, computing time can be generally saved by using kernels to replace mapping functions.

$$K(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) \quad (2.61)$$

The simplest kernel is the linear kernel. The equation of the linear kernel is shown in Eq. (2.62). It can be observed that no mapping functions are used on the right hand side of Eq. (2.62). Hence, the linear kernel is generally used in the linearly separable SVM classification.

$$K(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y} \quad (2.62)$$

An extensively utilized kernel function is called the polynomial kernel, as shown in Eq. (2.63). In the equation, d is a natural number called the degree.

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \mathbf{y} + 1)^d \quad (2.63)$$

Another extensively used kernel is the radial basis function kernel (RBF), as shown in Eq. (2.64). In the equation, γ is a positive parameter of radius controlling.

$$K(\mathbf{x}, \mathbf{y}) = e^{-\gamma \|\mathbf{x} - \mathbf{y}\|^2} \quad (2.64)$$

Once kernels are introduced, the Q matrix in Eq. (2.53) can be rewritten in the kernel form, as shown in Eq. (2.65). In the equation, the inner product of the two mappings of the two feature vectors, $\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$, are replaced by the kernel function.

$$\begin{aligned} Q_{ij} &= y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \\ &= y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \end{aligned} \quad (2.65)$$

Similarly, the decision function in Eq. (2.57) can be written in a simpler kernel form, as shown in Eq. (2.66). By introducing kernel functions, all mapping calculations in the dual problem can be replaced by a simple kernel calculation, and the computational complexity is therefore reduced.

$$\begin{aligned}
 & \text{sgn}(\mathbf{w}^T \cdot \phi(\mathbf{x}) + b) \\
 &= \text{sgn}\left(\sum_{i=1}^l y_i \alpha_i \phi(\mathbf{x}_i) \phi(\mathbf{x}) + b\right) \\
 &= \text{sgn}\left\{\sum_{i=1}^l y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b\right\}
 \end{aligned} \tag{2.66}$$

Kernel parameters are introduced to SVM if kernel methods are used. For example, if the RBF kernel ($K(\mathbf{x}, \mathbf{y}) = e^{-\gamma \|\mathbf{x} - \mathbf{y}\|^2}$) is used, the positive kernel parameter γ is added to the dual problem. Or, if the polynomial kernel is utilized, the kernel parameter d is introduced to the dual problem. To solve the dual problem, kernel parameters like γ and d must be assigned with a specific value.

2.2.6 Parameters Selection

As demonstrated in the previous section, kernel parameters in the dual problem must be assigned with a specific value. For example, the RBF kernel is used in the SVM dual problem, as shown in Eq. (2.64). In this case, the value of the kernel parameter γ should be determined.

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2} \tag{2.64}$$

Besides the kernel parameter, the penalty parameter C in Eq. (2.56) should also be assigned with a specific value. The process to determine the parameters like C and γ in

the dual problem is called parameter selection. A method of parameter selection is called a grid search (Figure 18). By assigning the parameters C and γ with different values, the SVM classifier achieves different levels of cross-validation (CV) accuracies. In this method, all grid points of (C, γ) are examined to find an optimal point (or a region) that returns the highest cross validation accuracy. The C and γ values represented by this optimal point (or this region) are substituted into the dual problem to generate the final classifier.

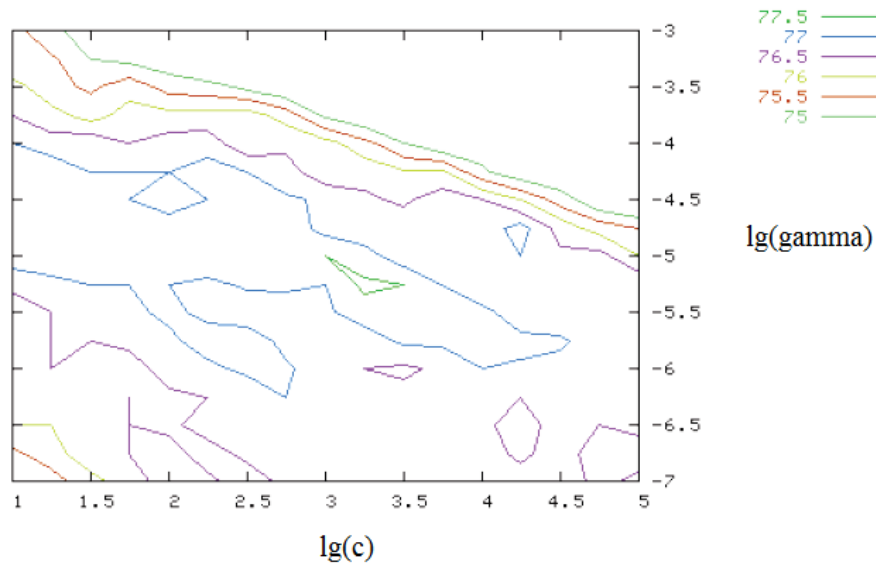


Figure 18. Parameter selection using grid search [46]

An example is shown here to illustrate the cross validation accuracy. Here, “people” are used as training samples. Features of a sample (or a person) include personal information like weight, height, and age. A sample’s class label is represented by the person’s gender.

Table 1. 10 training samples consisting of 3 features

Gender	Height	Weight	Age
M	6'2"	190	23
M	6'1"	198	30
M	5'8"	169	15
M	5'7"	159	26
M	4'	135	12
F	5'11"	199	52
F	5'11"	180	33
F	5'3"	152	16
F	5'	139	25
F	3'11"	99	9

A SVM classifier named “Classifier 1” is established based upon the training data in Table 1. A two-fold cross validation is conducted here to obtain the cross validation accuracy. In the two-fold cross validation, all the training samples are separated into two subsets, as shown in Table 2. Each subset contains 5 training samples.

Table 2. Training samples divided into two subsets

(a) Subset 1

Gender	Height	Weight	Age
M	6'2"	190	23
M	6'1"	198	30
M	4'	135	12
F	5'3"	152	16
F	5'	139	25

(b) Subset 2

Gender		Height	Weight	Age
M	M	5'8"	169	15
M	F	5'7"	159	26
F	F	5'11"	199	52
F	F	5'11"	180	33
F	M	3'11"	99	9

In the cross validation, the class labels of the samples in Subset 2 are assumed to be unknown. The samples in the Subset 1 are used to train a new SVM classifier named “Classifier 2”. Classifier 2 is then utilized to predict the class labels of the samples in Subset 2. These labels predicted by Classifier 2 are marked in bold in Table 2 (b). It can be observed that the second and fifth samples are incorrectly predicted. Therefore, the cross validation accuracy in this case is three out of 5, which is 60%.

Cross validation accuracy and testing accuracy are not always the same in value. However, they are tightly correlated to each other. In general, a classifier with a high cross validation accuracy finally achieves a promising testing accuracy. In the same way, a classifier with poor cross validation accuracy is more likely to obtain a low testing accuracy. Therefore, the parameter combination that achieves the highest cross validation accuracy is always preferred in the parameter selection process.

2.3 Multiple-Class Support Vector Machine Classification

In Section 2.2, the solution for the two-class SVM classifier was demonstrated. However, a SVM classification problem sometimes has more than two classes of samples. This type of problem is known as a multiple-class SVM classification problem. Two methods have been developed for solving a multiple-class SVM classifier, and they are shown in the following sections.

2.3.1 One-Against-The-Rest Strategy

The first method for solving the multi-class support vector machine classifier is known as the one-against-the-rest strategy. A k ($k > 2$) class multiple-class classification problem is shown here to demonstrate this strategy.

In the first step of the one-against-the-rest strategy, samples in the entire k classes are divided into two sub-classes: Class A_1 and Class B_1 . Sub-class A_1 contains the original Class 1, and Sub-class B_1 contains the original Class 2 to Class k , as shown in Figure 19.

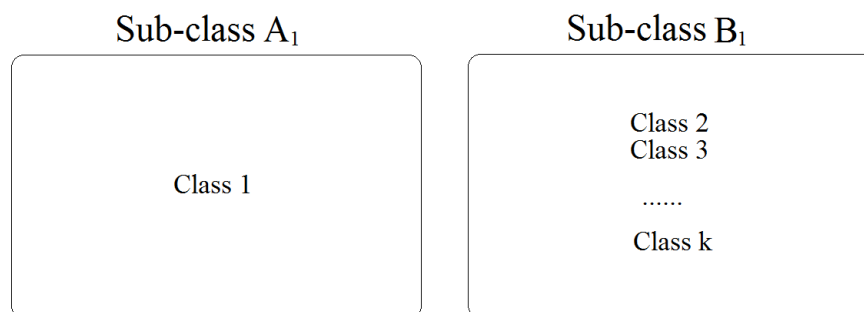


Figure 19. The first step to solve a multi-class classifier

A two-class classifier called “Classifier 1” is constructed based on the samples of Sub-class A_1 and Sub-class B_1 . The class label of Sub-class A_1 is defined as +1, and the label of Sub-class B_1 is defined as -1. The decision function d_1 of Classifier 1 is shown in Eq. (2.67).

$$d_1 = \mathbf{w}_1^T \cdot \phi(\mathbf{x}) + b_1 \quad (2.67)$$

In the second step, samples in the entire k classes are also divided into 2 sub-classes, as shown in Figure 20. Sub-class A_2 contains the original Class 2. Sub-class B_2 contains the original Class1, Class 3 and the rest till Class k.

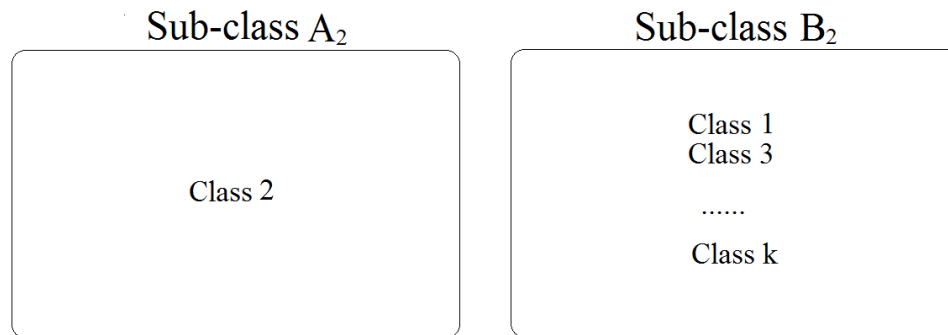


Figure 20. The second step to solve a multi-class classifier

A second two-class classifier named “Classifier 2” is generated based upon training samples in Sub-class A_2 and Sub-class B_2 . The decision function d_2 of Classifier 2 is shown in Eq. (2.68)

$$d_2 = \mathbf{w}_2^T \cdot \phi(\mathbf{x}) + b_2 \quad (2.68)$$

Using the same approach, in the last step (or k^{th} step), samples in the original k classes are divided into Sub-class A_k and B_k , as shown in Figure 20.

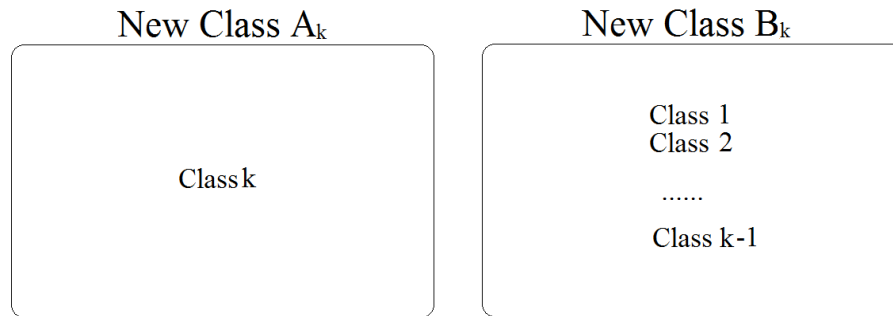


Figure 21. The final step to solve a multi-class classifier

The decision function of the k^{th} classifier is shown in Eq. (2.69)

$$d_k = \mathbf{w}_k^T \cdot \phi(\mathbf{x}) + b_k \quad (2.69)$$

Therefore, k decision functions are finally obtained, as shown in Eq. (2.70)

$$\begin{cases} d_1 = \mathbf{w}_1 \cdot \phi(\mathbf{x}) + b_1 \\ d_2 = \mathbf{w}_2 \cdot \phi(\mathbf{x}) + b_2 \\ \dots \\ d_k = \mathbf{w}_k \cdot \phi(\mathbf{x}) + b_k \end{cases} \quad (2.70)$$

A testing sample \mathbf{x}_j that belongs to the original Class j is substituted into all the decision functions in (2.70). Obviously, the 1st decision function d_1 returns a negative class label because \mathbf{x}_j does not belong to the original Class 1, as shown in Eq. (2.71)

$$\mathbf{w}_1^T \cdot \phi(\mathbf{x}_j) + b_1 < 0 \quad (2.71)$$

Similarly, the 2nd decision function d_2 also returns a negative class label because \mathbf{x}_j does not belong to the original Class 2, as shown in Eq. (2.72)

$$\mathbf{w}_2^T \cdot \phi(\mathbf{x}_j) + b_2 < 0 \quad (2.72)$$

In fact, all other decision functions return negative class labels except the j^{th} decision functions d_j . Decision function d_j returns a positive value because \mathbf{x}_j is in the original Class j , as shown in Eq. (2.73). Therefore, it is concluded that the testing sample \mathbf{x}_j belongs to Class j .

$$\mathbf{w}_j^T \cdot \phi(\mathbf{x}_j) + b_j > 0 \quad (2.73)$$

2.3.2 One-Against-One Strategy

The second method to solve a multiple-class SVM classifier is called the one-against-one strategy. The principle of this method is to select training samples from only two classes at each time, and a two-class classifier is generated based upon these samples.

In a k -class ($k > 2$) SVM classification problem, $\frac{k \times (k-1)}{2}$ different two-class classifiers are generated using this method. Therefore, a total of $\frac{k \times (k-1)}{2}$ different decision functions are generated. For example, in a 4-class SVM problem, 6 two-class SVM classifiers are generated based upon sample combinations as shown below. The 6 decision functions of these 6 two-class classifiers are shown in Table 3.

$(Class1, Class2); (Class1, Class3); (Class1, Class4); (Class2, Class3); (Class2, Class4); (Class3, Class4)$

Table 3. 6 two-class decision functions obtained in a 4-class SVM classification

1 st class	2 nd class	Decision functions
Class 1	Class 2	$f_{12} = \mathbf{w}_{12}^T \cdot \phi(\mathbf{x}) + b_{12}$
Class 1	Class 3	$f_{13} = \mathbf{w}_{13}^T \cdot \phi(\mathbf{x}) + b_{13}$
Class 1	Class 4	$f_{14} = \mathbf{w}_{14}^T \cdot \phi(\mathbf{x}) + b_{14}$
Class 2	Class 3	$f_{23} = \mathbf{w}_{23}^T \cdot \phi(\mathbf{x}) + b_{23}$
Class 2	Class 4	$f_{24} = \mathbf{w}_{24}^T \cdot \phi(\mathbf{x}) + b_{24}$
Class 3	Class 4	$f_{34} = \mathbf{w}_{34}^T \cdot \phi(\mathbf{x}) + b_{34}$

To predict the class of a testing sample \mathbf{x} , it is substituted into the 6 different decision functions. The 6 decision functions return 6 predicted class labels, as shown in Table 4.

Table 4. 6 predicted class labels obtained by substituting data \mathbf{x} into the 6 different decision functions

1 st class	2 nd class	Decision function	Predicted class
Class 1	Class 2	$f_{12} = \mathbf{w}_{12}^T \cdot \phi(\mathbf{x}) + b_{12}$	2
Class 1	Class 3	$f_{13} = \mathbf{w}_{13}^T \cdot \phi(\mathbf{x}) + b_{13}$	1 or 3
Class 1	Class 4	$f_{14} = \mathbf{w}_{14}^T \cdot \phi(\mathbf{x}) + b_{14}$	1 or 4
Class 2	Class 3	$f_{23} = \mathbf{w}_{23}^T \cdot \phi(\mathbf{x}) + b_{23}$	2
Class 2	Class 4	$f_{24} = \mathbf{w}_{24}^T \cdot \phi(\mathbf{x}) + b_{24}$	2
Class 3	Class 4	$f_{34} = \mathbf{w}_{34}^T \cdot \phi(\mathbf{x}) + b_{34}$	3 or 4

Assume a testing sample \mathbf{x} is in Class 2 (but the classifier does not know it). The first decision function f_{12} correctly predicts its class label to be 2. Since the second

decision function f_{13} is generated based on Class 1 and Class 3, f_{13} cannot correctly predict the class label of sample x . Therefore, f_{13} predicts its label to be either Class 1 or Class 3. Finally, three out of the 6 decision functions, f_{12} , f_{23} , and f_{24} , correctly predict the class label of x . The other three decision functions, f_{13} , f_{14} , and f_{34} , simply predict some random class labels. However, no matter what class labels they predict, Class 2 wins the maximum votes from the 6 decision functions (three out of 6). The class label that receives the most votes is chosen to be the class label of the unknown sample. Therefore, the unknown sample x is predicted to belong to Class 2.

2.4 Chapter Recap

In this chapter, the primal problem of SVM classification was derived from a Euclidean point of view. The primal problem was transformed to the dual problem for simplicity. A decomposition strategy was introduced for solving the SVM dual problem. The parameter selection technique was utilized to determine kernel parameters and the penalty parameter. Finally, two multiple-class SVM classification methods were discussed.

Chapter 3

3. Time-Frequency Representations

As mentioned in Section 1.2.2, damage sensitive features are extracted by applying signal-processing algorithms in the guided wave SHM method. In this research, the time-frequency representation (TFR) method is chosen for analyzing the damage signals. In this chapter, some common time-frequency representation methods are investigated. Time-frequency representations using different methods based upon a damage signal are generated and compared. The optimal TFR method is finally chosen for the damage signals analyses in this research.

3.1 TFR methods

The two classical signal representation methods are the time-domain representation and the frequency domain representation. The time-domain representation is not localized with respect to frequency. In the time-domain representation, a signal is represented as wave amplitude versus time. The time-domain representation is direct and easy to understand, but it is not capable of diagnosing complex signals with multiple frequency components [47]. The frequency representation is non-localized with respect to time. A signal is represented as its amplitude versus its frequency in this representation. The frequency-domain analysis is capable of identifying the signal components in complex signals, but it doesn't represent the signal's time-varying phenomenon [48]. The time-

frequency representation is developed because of these drawbacks of the time-domain and frequency-domain representations. Time-frequency representation describes a signal as a function in a two-dimensional (time and frequency) space. Several popular time-frequency analysis methods including Short Time Fourier Transform, Wigner-Ville Distribution, and Wavelet Transform are discussed in the following sections.

3.1.1 Short Time Fourier Transform

The Short time Fourier transform (STFT) method is widely used for analyzing time-varying signals that contain multiple frequency components. The basic idea of this method is to break up a signal into small time segments and conduct Fourier analysis based upon each time segment to obtain the frequency spectrum in each. By studying the spectrum of each time segment, the frequencies that exist in each time segment can be observed. By combing the spectrums in all the small time segments, a frequency spectrum varying with time is obtained.

To study a signal $f(\tau)$ around a specific time point t , the original signal is modified by multiplying a window function $h(t)$. The modified signal is shown in Eq. (3.1).

$$f_t(\tau) = f(\tau)h(\tau - t) \quad (3.1)$$

In Eq. (3.1), t represents a fixed time point around which the original signal is analyzed. τ represents the running time. $h(t)$ is called the window function, and it implies that only a small portion of the original signal is observed, just like only a portion of the entire scenery can be seen through a window. The window function $h(t)$ keeps the

modified signal $f_t(\tau)$ almost unchanged from the original signal wave $f(\tau)$ around the fixed time t . But, it converts the original signal $f(\tau)$ to 0 in the time regions far from the fixed time point t [47]. The modified signal $f_t(\tau)$ is illustrated in Eq. (3.2).

$$f_t(\tau) = \begin{cases} f(\tau) & \text{round time point } t \\ 0 & \text{away from time point } t \end{cases} \quad (3.2)$$

The Fourier transform is applied to the modified signal in Equation (3.1) to obtain the frequency distribution around the fixed time point t , as shown in Eq. (3.3).

$$\begin{aligned} S_f(t, \omega) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-j\omega\tau} f_t(\tau) d\tau \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-j\omega\tau} f(\tau) h(\tau-t) d\tau \end{aligned} \quad (3.3)$$

The term $S_f(t, \omega)$ is known as the STFT in vicinity to the fixed time point t . The energy density spectrum P_{sp} at the fixed time point t is shown in Eq. (3.4).

$$\begin{aligned} P_{sp}(t, \omega) &= |S_f(t, \omega)|^2 \\ &= \left| \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-j\omega\tau} f(\tau) h(\tau-t) d\tau \right|^2 \end{aligned} \quad (3.4)$$

Using Eq. (3.4), a graph of the energy density spectrum distribution in the entire time domain and frequency domain can be generated. This graph is also known as the spectrogram [47]. The spectrogram represents how the signal energy density varies with respect to time and frequency.

If the window size is increased too large in the spectrogram, the frequency components that change with time cannot be clearly observed. It implies that the time

resolution is decreased by increasing the window size. In contrast, the frequency resolution is decreased if the windows size is decreased. Therefore, the windows size of STFT has to be carefully chosen to obtain both promising time and frequency resolutions.

3.1.2 Wigner-Ville Distribution

Due to the resolution limitation of the STFT method, some other time-frequency representation methods have been developed. One of the most studied methods is the Wigner-Ville distribution (WVD). It was first introduced by Eugene Wigner for an application of quantum mechanics in 1932, and it was derived independently by J. Ville using a different approach in 1948. WVD has recently been recognized as a powerful tool for analyzing time-varying signals.

The analytic associate of a time-varying signal $f(t)$ is described in Eq. (3.5).

$$z(t) = f(t) + jH[f(t)] \quad (3.5)$$

In Eq. (3.5), $H[f(t)]$ is the Hilbert transform of the real signal $f(t)$. In Eq. (3.6), ζ is the Cauchy principal value of the integral.

$$H[f(t)] = \zeta \left(\int_{-\infty}^{+\infty} \frac{s(u)}{\pi(t-u)} du \right) \quad (3.6)$$

The equation of WVD in terms of a signal $z(t)$ is described in Eq. (3.7).

$$W(t, f) = \int_{-\infty}^{+\infty} z\left(t + \frac{\tau}{2}\right) z^*\left(t - \frac{\tau}{2}\right) e^{-j2\pi f\tau} d\tau \quad (3.7)$$

Equivalently, the WVD can be written in terms of the signal's spectrum $F(\omega)$ in Eq. (3.8).

$$W(t, f) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega + \frac{\gamma}{2}) F^*(\omega - \frac{\gamma}{2}) e^{j\gamma t} d\gamma \quad (3.8)$$

To obtain the WVD attribute of a signal at a specific time point t , the signal pieces at a past time $(t - \frac{\tau}{2})$ and a future time $(t + \frac{\tau}{2})$ are multiplied and added up [48]. Therefore, the WVD attribute at a time point t can be simply examined by folding the left side of the signal to the right side and see if any overlap exists. The WVD of a signal is permanently zero at any time point before the signal starts or after the signal ends. This is because nothing overlaps by folding the signal with respect to these time points.

The WVD is usually nonzero at a time point t where the signal amplitude is zero. As an example, a signal is shown in Figure 22. The signal amplitude is zero between the time points t_1 and t_2 . The time point t_0 is at the center of t_1 and t_2 . Obviously, overlaps exist by folding the signal with respect to the time point t_0 . Therefore, WVD is not usually zero at the time points where no signals exist. This phenomenon is an obvious shortcoming of the WVD known as the interference or cross term.



Figure 22. The Wigner-Ville distribution at time t_0

3.1.3 Wavelet Transform

The Wavelet transform (WT) method is extensively used to analyze signals that exhibit noisy, aperiodic, intermittent, and transient features [48-49]. The WT analysis was first developed to study seismic signals in the middle 1980s. The applications of the WT rapidly grew in the beginning of the 1990s. In this method, small wavelike functions also known as wavelets, are utilized to analyze signals. Common wavelets are Haar, Morlet, and Mexican hat. A Mexican hat is depicted in Figure 23.

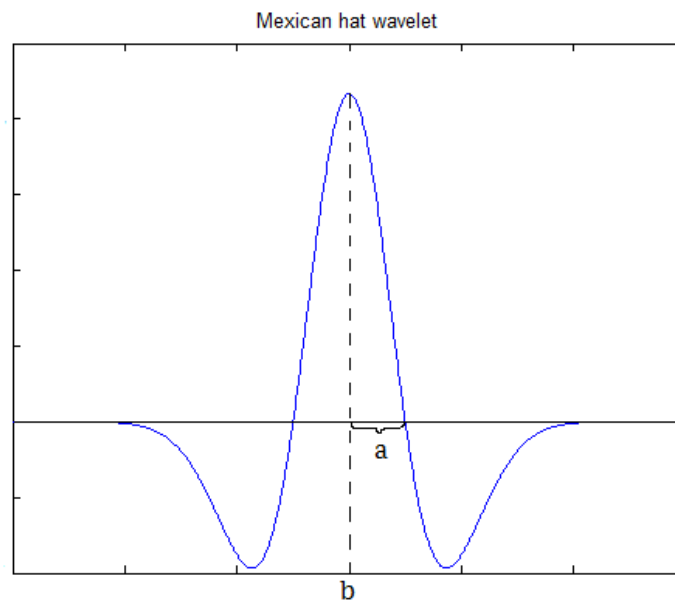


Figure 23. A Mexican hat wavelet

A wavelet can be manipulated in two ways. First, a wavelet can be moved to various locations along a signal. As shown in Figure 23, the location of the wavelet is dominated by parameter b . Secondly, a wavelet can be stretched or squeezed, and the dilation of a wavelet is controlled by the parameter a . The basic idea of the WT method is to compare the local matching between the wavelet and the signal. If the wavelet matches

the shape of the signal well at a certain location (parameter b) and dilation (parameter a), a large wavelet transform value is obtained, as depicted by point 1 in Figure 24. On the contrary, if the wavelet and the signal do not match well at a certain location and dilation, a low wavelet transform value is obtained, as shown at point 2 in Figure 24.

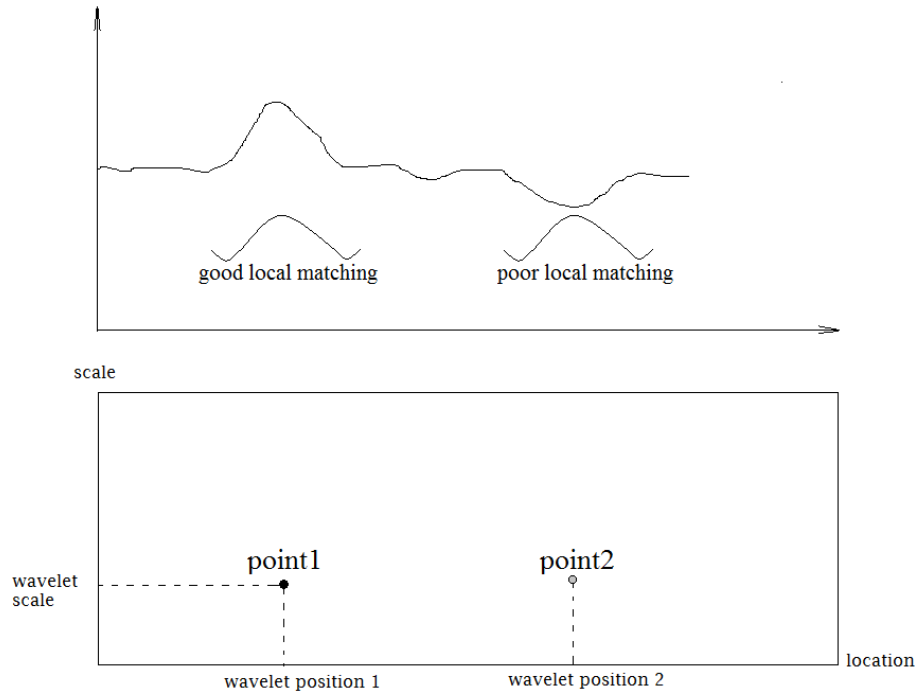


Figure 24. The basic principle of Wavelet Transform

To become a wavelet, a function $\psi(t)$ should meet some requirements. The first requirement is that its energy must be finite, as shown in Eq. (3.9).

$$E = \int_{-\infty}^{+\infty} |\psi(t)|^2 dt < \infty \quad (3.9)$$

The second requirement is that the function $\psi(t)$ must satisfy Eq. (3.10).

$$C_g = \int_{-\infty}^{+\infty} \frac{|\hat{\psi}(f)|^2}{f} df < \infty \quad (3.10)$$

$\hat{\psi}(f)$ is the Fourier transform of the wavelet, as shown in Eq. (3.11).

$$\hat{\psi}(f) = \int_{-\infty}^{+\infty} \psi(t) e^{-i2\pi ft} dt \quad (3.11)$$

The WT of a signal is defined as the convolution of the wavelet function and the signal. This is illustrated in Eq. (3.12).

$$WT(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) \psi^* \left(\frac{t-b}{a} \right) dt \quad (3.12)$$

The function $\psi(t)$ in (3.12) is called the mother wavelet. The energy density of the WT is demonstrated in Eq. (3.13). The plot of the energy density $E(a, b)$ is also known as the scalogram, and it is equivalent to the spectrogram of the STFT.

$$E(a, b) = |WT(a, b)|^2 \quad (3.13)$$

3.2 TFR Method Selection

Three time-frequency analysis methods were illustrated in the previous sections. In this section, a damage signal is generated using a finite element analysis tool named ABAQUS[®]. The three time-frequency analysis methods are conducted based upon the damage signal to generate different time-frequency representations (TFR) graphs. These TFR graphs are compared to choose the optimal TFR method.

To examine the TFRs of a damage signal, a damage scenario is simulated using a finite element analysis tool named ABAQUS[®]. In the ABAQUS[®] model, a 2-m long aluminum beam is simulated. The beam thickness is 3-mm. Two 1-cm by 1-cm PZT transducers are attached to the center of the beam. A corrosion of 3-cm wide is made on the beam at a distance of 15-cm from the transducers, as shown in Figure 25 (b).

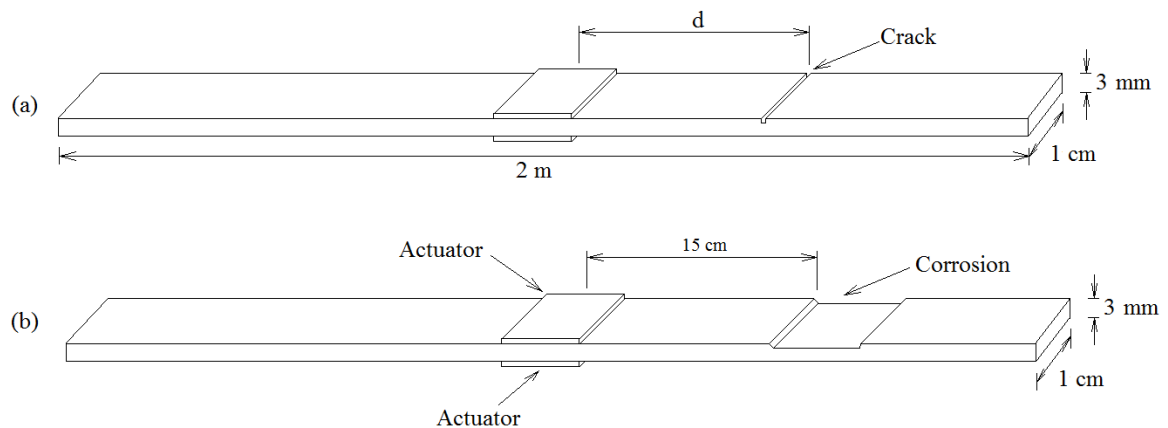


Figure 25. Two damage scenarios simulated in ABAQUS[®] (a) a crack scenario (b) a corrosion scenario

A 50 kHz center frequency, Hanning windowed, tone-burst signal is generated by the PZT actuators. As the excited signal travels along the beam, some portions of the signal are reflected by the corrosion. The in-plane beam displacement excited by the reflected signal is measured at a node close to the PZT actuator on the upper surface of the beam, as shown in Figure 26 (a). The sensed signal contains two portions namely, the actuating signal and the damage signal reflected by the corrosion. Since only the reflected signal is useful for characterizing the damage features, the actuating signal is eliminated from the entire sensed signal. Only the corrosion signal is kept for the STFT, as shown in Figure 26 (b). The spectrogram of the corrosion signal is depicted in Figure 26 (c). In the

spectrogram, different levels of spectrogram intensity represent different levels of energy density.

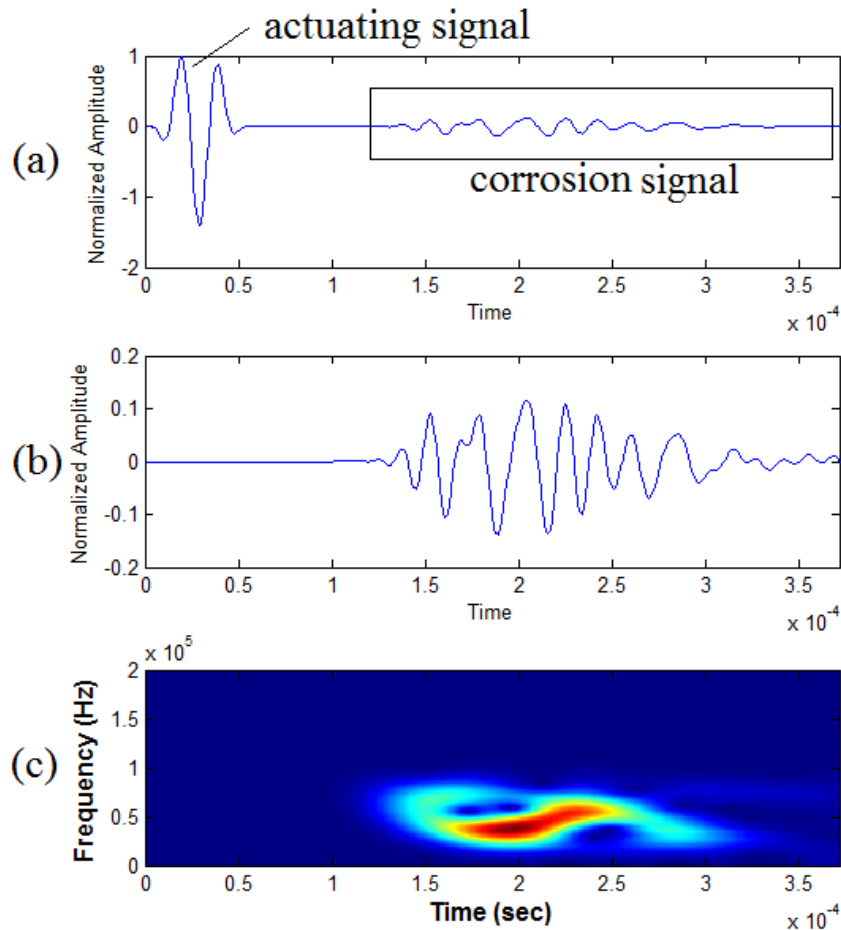


Figure 26. Plots of the sensed signal, damage signal, and the spectrogram of the damage signal (a) the sensed signal (b) the reflected damage signal only (c) the Spectrogram of the damage signal

The same corrosion signal is utilized to conduct the Wigner-Ville distribution, as shown in Figure 27. Figure 27 (a) illustrates the sensed signal, and Figure 27 (b) shows the corrosion signal only by eliminating the actuating signal. The WVD of the corrosion signal is depicted in Figure 27 (c). In this figure, a cross term can be observed.

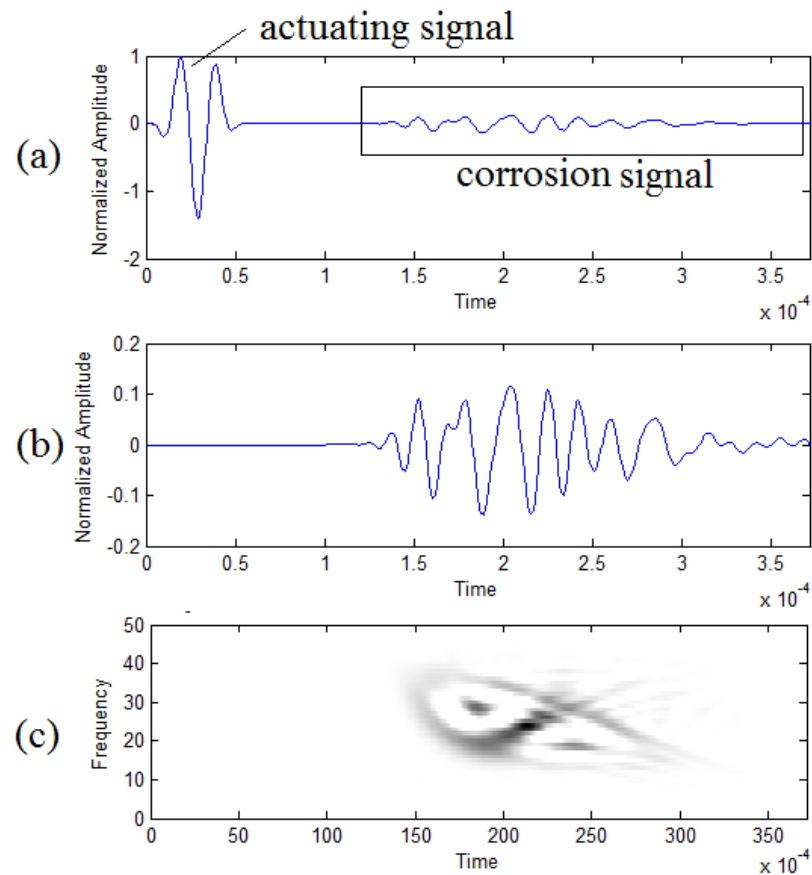


Figure 27. Plots of the sensed signal, damage signal, and the Wigner-Ville distribution of the damage signal (a) the sensed signal (b) the reflected damage signal only (c) The Wigner-Ville distribution of damage signal

In Figure 28 (c), the wavelet transform based upon the same signal is shown. It can be seen that both the time domain and the frequency domain resolutions are not promising compared to the STFT spectrogram. Therefore, the STFT is chosen as the TFR method for this research. The Spectrograms of damage signals are used for damage classifications, as will be demonstrated in the following chapter. By choosing an appropriate window size, the STFT is able to provide good resolutions in both time and frequency domains.

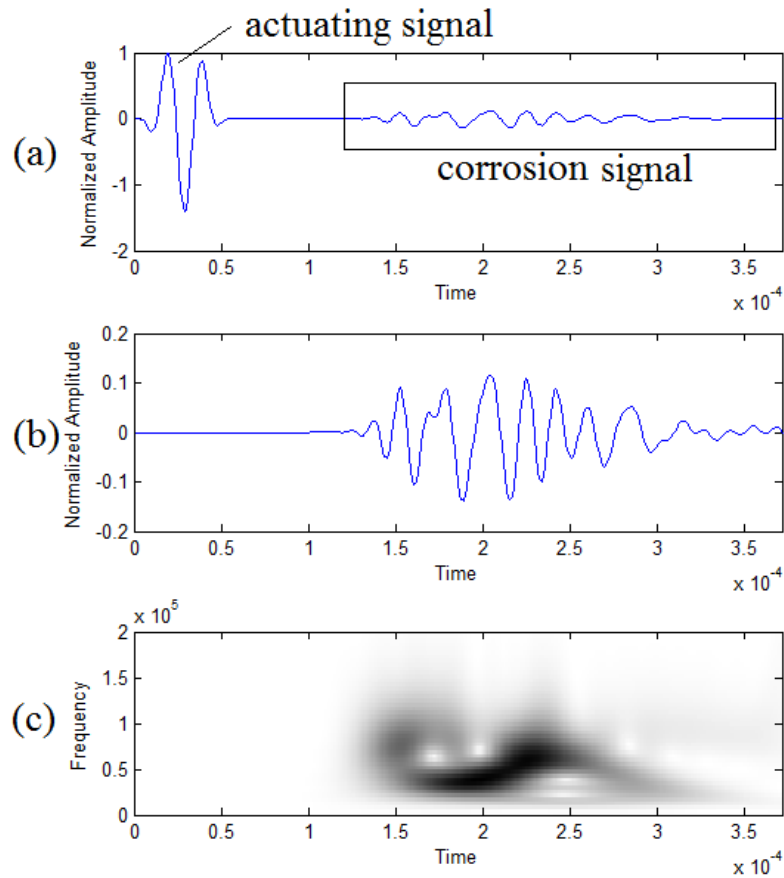


Figure 28. Plots of the sensed signal, damage signal, and the wavelet transform of the damage signal (a) the sensed signal (b) the reflected damage signal only (c) the wavelet transform of damage signal

3.3 Chapter Recap

In this chapter, several common time frequency representation methods have been discussed. Time-frequency representations using different methods are generated and compared. Finally, the STFT was chosen for transforming damage signals into the Spectrograms. Various damage signals were processed using the STFT to generate training and testing data for the SVM classification, as will be discussed in the following chapter.

Chapter 4

4. Damage Classification and Result Analysis

The SVM classification theories and TFR methods were demonstrated in the previous chapters. Damage signals are processed using the STFT to generate training and testing data. In order to conduct the SVM classification, a certain quantity of training samples is required. However, generating a large number of training and testing samples in the experimental environment is time and money consuming. Therefore, a finite element analysis tool named ABAQUS[®] is utilized to generate training and testing samples in this research. ABAQUS[®] simulation, feature extraction, and damage classification are illustrated in detail in the following sections.

4.1 Training Samples using ABAQUS[®]

ABAQUS[®] is a computer software tool for finite element analysis that was first released in 1978. ABAQUS[®] has been extensively applied in aerospace, automotive, and civil engineering due to its strong modeling and simulation capabilities. In this research, all damage scenarios are simulated in ABAQUS[®] V6.12. The computer platform used for running ABAQUS[®] is a Dell PRECISION T6600 workstation equipped with an Intel[®] Xeon[®] E5-2603 CPU, 4 GB RAM, and an AMD[®] FirePro[®] V4900 graphic adapter. The

operation system is a Windows[®] 7 Enterprise edition. A picture of the workstation is shown in Figure 29.



Figure 29. The workstation utilized for running ABAQUS[®]

To simulate the damage scenarios in ABAQUS[®], the outlines of the damages should be determined first. Figure 30 (a) shows the cross section outline of a crack in a metallic material. In the figure, the cross section of a crack looks like a thin cleft. For this reason, a crack is represented as a thin notch in ABAQUS[®] simulation, as shown in Figure 30 (b). Similarly, the cross section of a pitting corrosion in a metallic material is depicted in Figure 30 (c). It can be observed that the width of the corrosion is generally much larger than that of the crack. Also, the corrosion has the shape of a concave depression. Therefore, corrosion is represented as a trapezoidal concave in ABAQUS[®], as shown in Figure 30 (d).

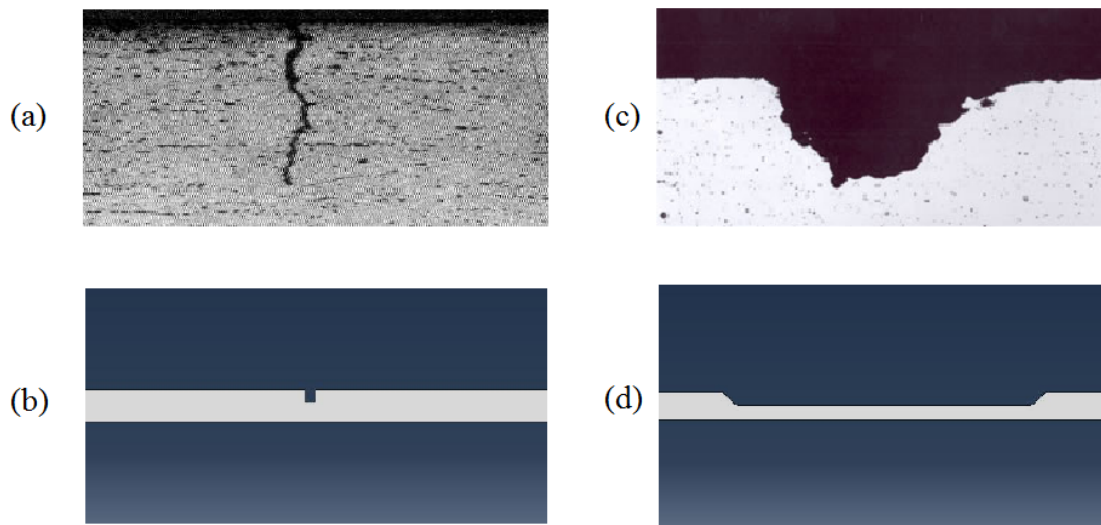


Figure 30. Cross sections of a crack and corrosion in a real metallic material and the ABAQUS simulations (a) a crack in a metallic material (b) a crack simulated in ABAQUS[®] (c) a piece of corrosion in a metallic material (d) a piece of corrosion simulated in ABAQUS[®]

In a 2-dimensional ABAQUS[®] model for simulating a crack scenario, two 1-cm by 1-cm PZT actuators are attached to the center of a thin 2-m long aluminum beam, as shown in Figure 31 (a). The thickness of the beam is 3-mm. The positive poles of the PZT actuators face the opposite direction to each other. A crack is located at a distance “d” from the actuators. The width of a crack is 1mm. The depth “t” of a crack varies from 1/8 to 1/2 of the beam thickness (3-mm). A corrosion scenario is shown in Figure 31 (b). The width of corrosion is 3-cm. The cross section of the corrosion is shaped as a trapezoidal channel. The other settings of the corrosion scenario are similar to the crack scenario.

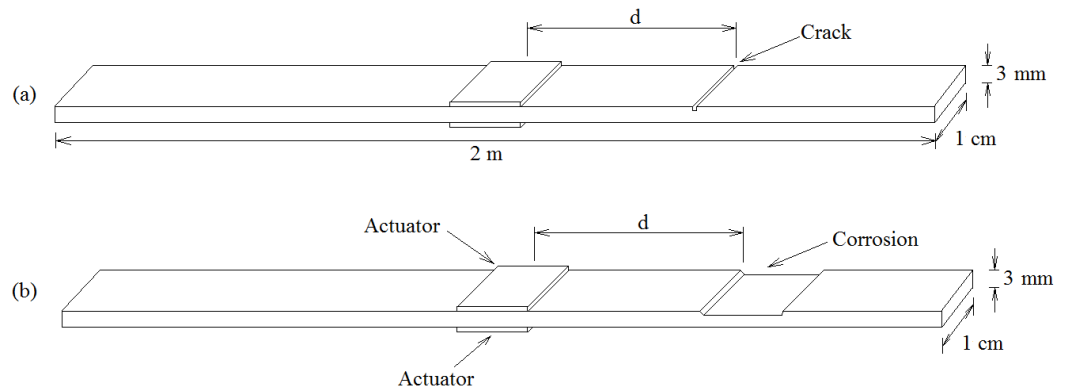


Figure 31. ABAQUS[®] models of a crack and a piece of corrosion (a) a crack (b) a piece of corrosion

A 50 kHz center frequency, Hanning windowed, tone-burst signal is generated by the PZT actuators. The signal is depicted in Figure 32.

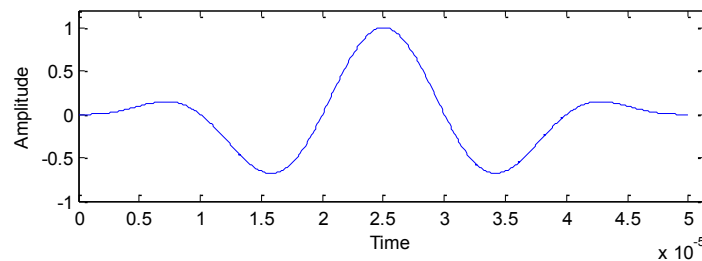


Figure 32. 50 kHz center frequency, Hanning windowed, tone-burst excited signal

Figure 33 (a) shows the aluminum beam before the excited signal is applied to the PZT actuators. A guided wave is generated after the excited signal is applied, as shown in Figure 33 (b). As the excited guided wave propagates along the aluminum beam, some portions of the wave are reflected by the crack on the beam, and the residual from the wave propagates through the damage, as shown in Figure 33 (c). In these figures, the

scale factor of the beam displacement is 600,000. It means the beam displacement in the figures is 600,000 times as large as the actual beam displacement.

The sensed signal is obtained by measuring the in-plane beam displacement at a node close to the PZT actuator on the upper surface of the beam. As a result, no PZT sensor is used in the ABAQUS[®] model. The entire model, including the beam and the PZT actuators, is meshed using triangular elements. A damage model consists of about 12,000 mesh elements. It takes the workstation about three hours to complete analyzing a single model.

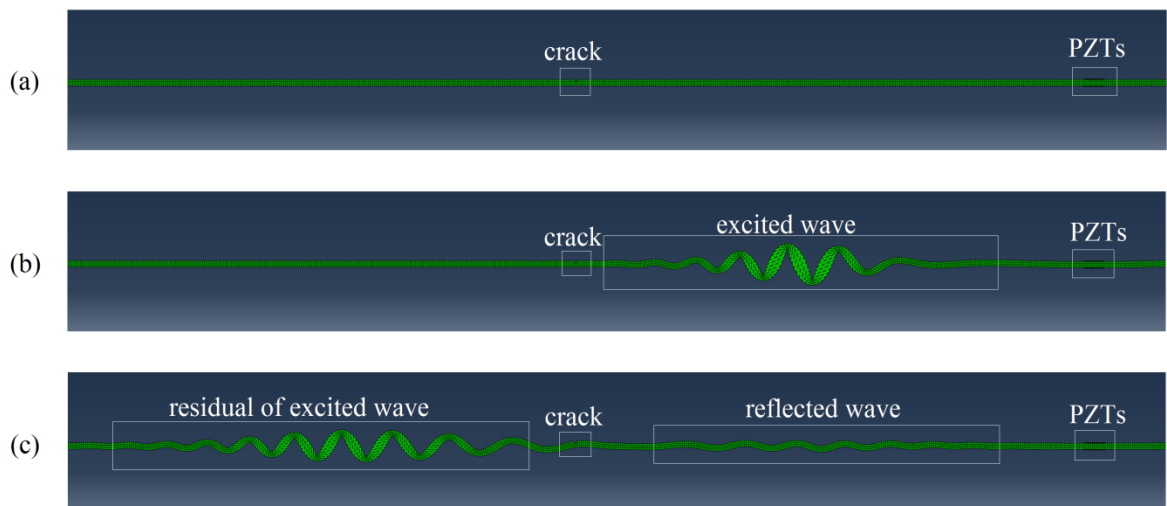


Figure 33. The guided wave propagation in an ABAQUS model (scale: x600,000) (a) a crack and PZT actuators (b) the guided wave generated by the PZT actuators (c) the reflected wave and the residual of the excited wave

A crack located at a distance of 15-cm from the PZT actuators ($d=15\text{-cm}$) with a depth of 1.125-mm ($3/8$ of the beam thickness) is shown here as an example. As demonstrated in the previous chapter, the spectrogram based on the STFT is chosen to

analyze the sensed signal for damage classification. The spectrogram of the entire signal, including the excited and the reflected signal, is shown in Figure 34 (a). Both the excited signal and the reflected crack signal can be seen in this figure. The reflected crack signal begins at around $80 \mu s$. The crack signal is longer than the excited signal because the crack signal contains both the A0 and S0 modes. The crack signal is marked inside a dotted rectangle in Figure 34 (a). The spectrogram of the crack signal is marked inside a white rectangle in Figure 34 (b). In this spectrogram, different levels of spectrogram intensity represent different levels of energy density. Figure 34 (b) indicates that the energy is mostly concentrated within the excited signal. The energy density of the reflected signal is “weak” compared to the excited signal. The reflected signal is highlighted in the “blurry” and light blue region inside the white rectangle in Figure 34 (b).

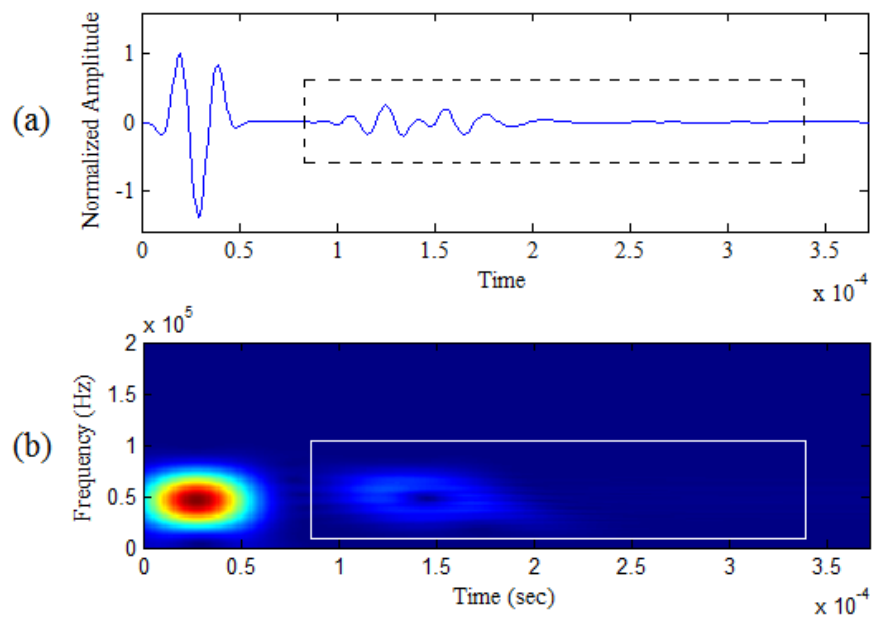


Figure 34. The sensed signal and its spectrogram (a) the sensed signal (b) STFT spectrogram of the sensed signal ($d=15\text{-cm}$, $3/8$ of the beam thickness)

Since only the reflected signal from the damage is helpful for characterizing the damage, the excited signal is eliminated from the whole signal. Only the reflected signal is reserved (the region from $81\ \mu\text{s}$ to $372\ \mu\text{s}$ of the original signal), as shown in Figure 35 (a). The spectrogram of the reflected signal alone is depicted in Figure 35 (b).

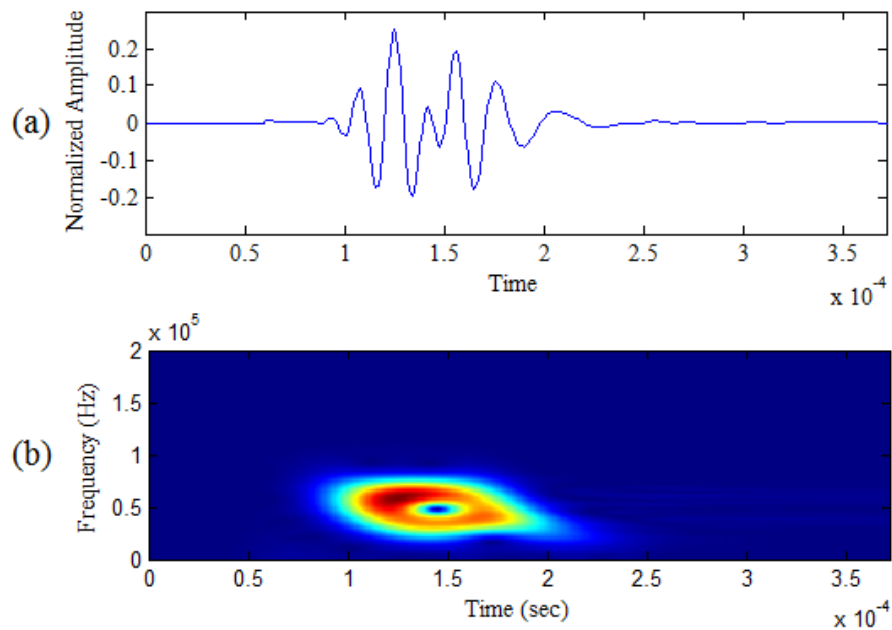


Figure 35. The reflected damage and its spectrogram (a crack of $d=20$ -cm, depth= $3/8$ of the beam thickness) (a) the reflected signal (b) the spectrogram of the reflected signal

Several spectrograms of crack signals are depicted in Figure 36. In the figure, the distances between the cracks and the PZT actuators are $d=15$ -cm. The depths of the cracks change from $1/8$ to $4/8$ of the beam thickness. It can be observed that outlines of these spectrograms are similar. But, positions of the maximum spectrogram energy density in these spectrograms are slightly different.

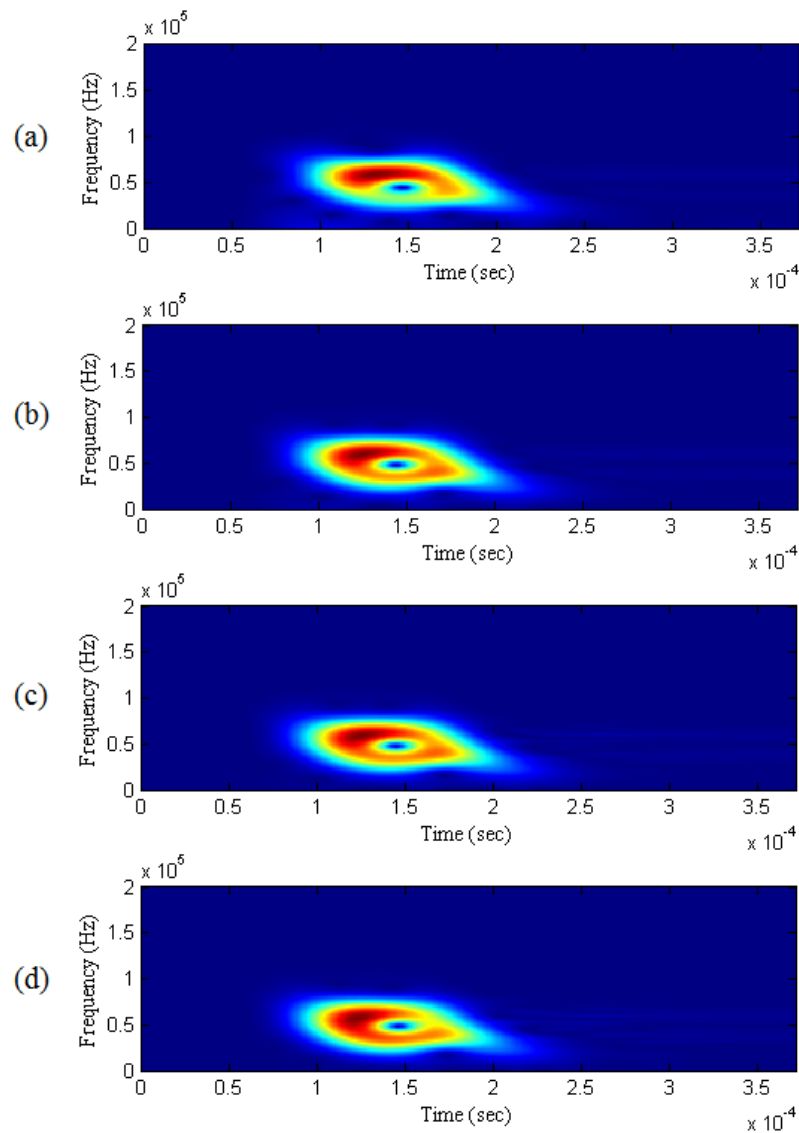


Figure 36. Spectrograms of 4 crack signals of $d=15$ -cm (a) a crack of $1/8$ beam thickness (b) a crack of $2/8$ beam thickness (c) a crack of $3/8$ beam thickness (d) a crack of $4/8$ beam thickness

Figure 37 shows the spectrograms of the crack signals of $d=20$ -cm. The depths of these cracks vary from $1/8$ to $4/8$ of the beam thickness. By comparing Figure 36 and Figure 37, it can be observed that the spectrogram outlines in Figure 37 are generally

“wider” than Figure 36. This is because the reflected wave signal contains multiple wave modes with different frequencies. Different wave modes travel at different velocities. As the travelling distance increases, the reflected wave becomes more dispersive.

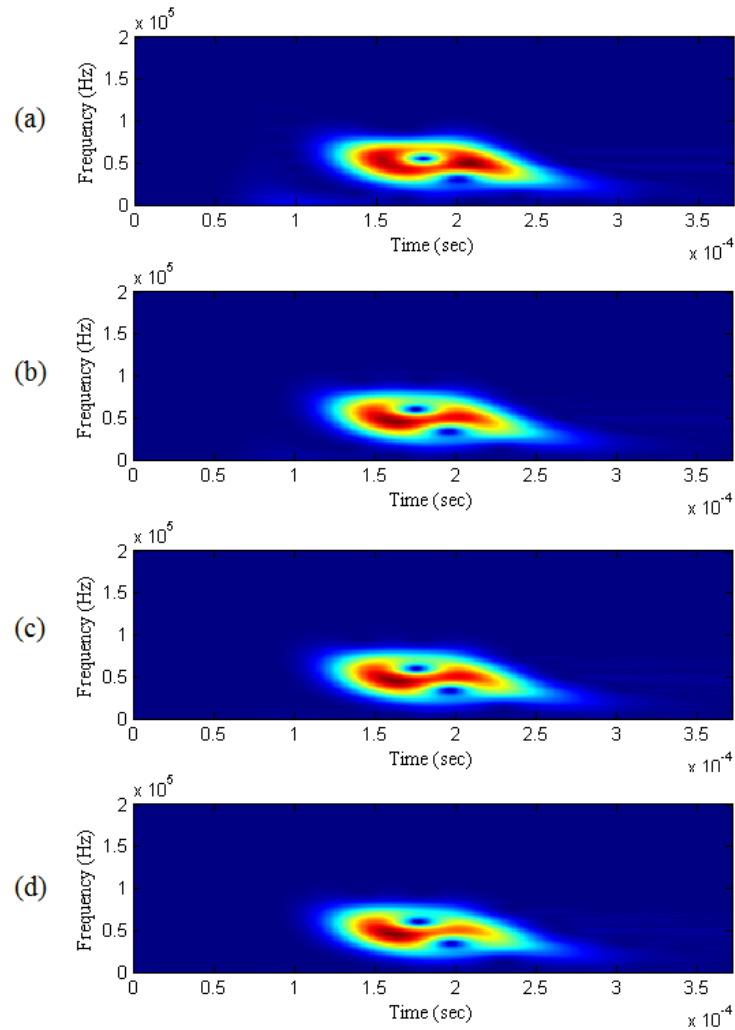


Figure 37. Spectrograms of 4 crack signals of $d=20$ -cm (a) a crack of $1/8$ beam thickness (b) a crack of $2/8$ beam thickness (c) a crack of $3/8$ beam thickness (d) a crack of $4/8$ beam thickness

4.2 Feature Extraction

As previously discussed, the spectrograms of the damage signals are processed to generate the training and testing data for the SVM classification. In this section, the algorithm to transform spectrograms into training and testing feature vectors is demonstrated in detail.

In this research, the spectrogram of a damage signal is saved as a 24 bit (true color) TIFF no compression file. In a spectrogram image, each of the red, green, and blue (RGB) color channels has 256 (as 2^8) levels of color depth. Different colors can be described by different combinations of RGB color channel depths. For example, the “black” color is represented by Red 0, Green 0, and Blue 0, while the “white” color is represented by Red 255, Green 255, and Blue 255. A spectrogram image is composed of pixels, and RGB color depths can be extracted from any pixel in a spectrogram. For example, Figure 38 (a) shows the spectrogram of a sensed signal collected in a corrosion scenario. The region in the black rectangle is magnified so that the pixels in this region become visible, as illustrated in Figure 38 (b). In the figure, the color channel depth information of the “blue” pixel marked by the white square can be extracted as: Red 0, Green 36, and Blue 255.

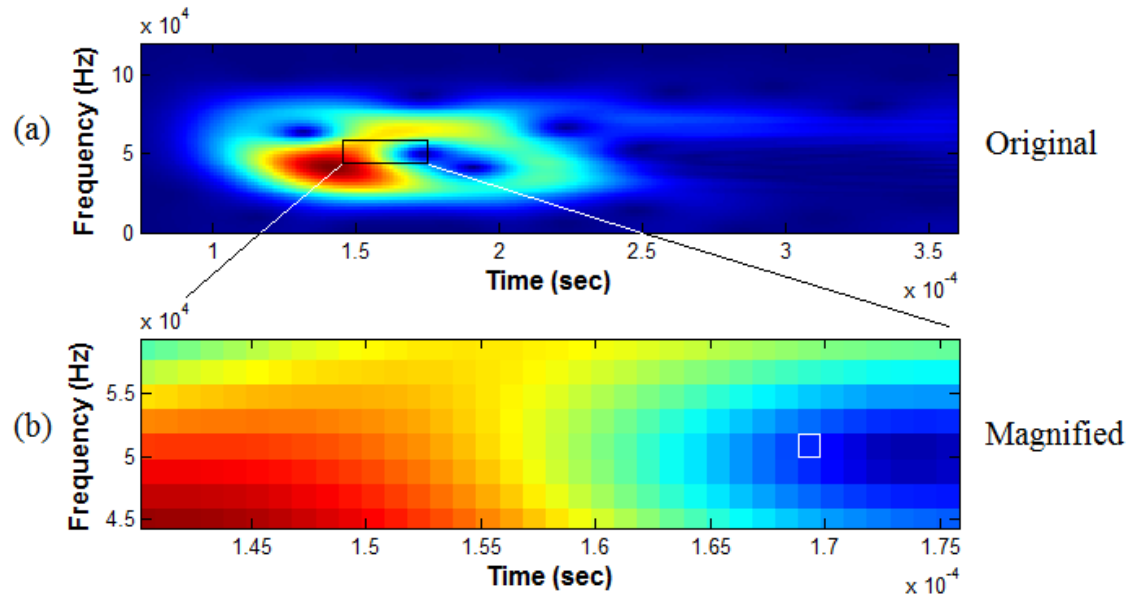


Figure 38. The spectrogram of a corrosion signal and a magnified region from the spectrogram (a) the spectrogram (b) a magnified region from the spectrogram

This process can be used to extract the color depth information of all the pixels in a spectrogram. Different spectrograms have different distributions of color channel depth information. The difference in the distribution of the color depth information can be used to describe the difference between different spectrograms. Therefore, the distribution of the RGB color depth in the pixels can be used as “features” to describe a spectrogram. Color depth information of all the pixels in a spectrogram can be reorganized to generate a feature vector for SVM training and testing. To improve the SVM classification accuracy, signal frequencies represented by pixels in a spectrogram are also used as additional features in the feature vectors. For example, the pixel in the white rectangle in Figure 39 (b) represents a frequency level around 5.1×10^4 Hz.

The algorithm for organizing color depth information and frequency information and generating a SVM feature vector is demonstrated in Figure 39. In the figure, the color

depth and frequency information of all the pixels in a spectrogram are chained together to generate a lengthy feature vector. The RGB color depth information of the first pixel in the spectrogram (the pixel at the upper left corner) is extracted and placed in the first three elements of the feature vector. The frequency represented by the first pixel is calculated and placed in the fourth element of the feature vector. The color depths and frequency of the second pixel are stored in the next adjacent 4 elements. This process is repeated to all other pixels in the spectrogram until the RGB color depth and the frequency of the last pixel is stored in the feature vector. The length of a feature vector is related to the size of a spectrogram image. As an example, if the size of a spectrogram is 500×250 pixels, the length of the features vector is $500 \times 250 \times 4 = 500000$.

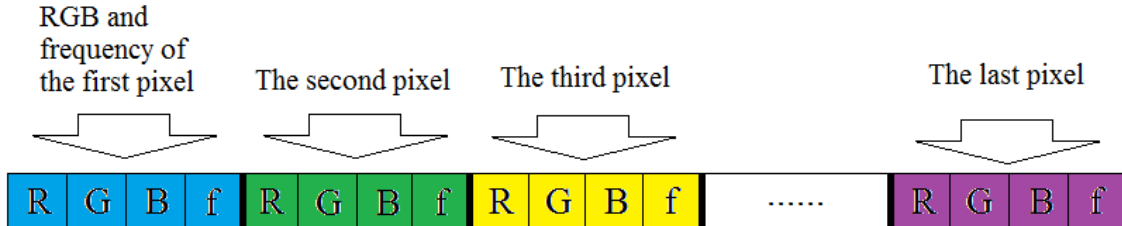


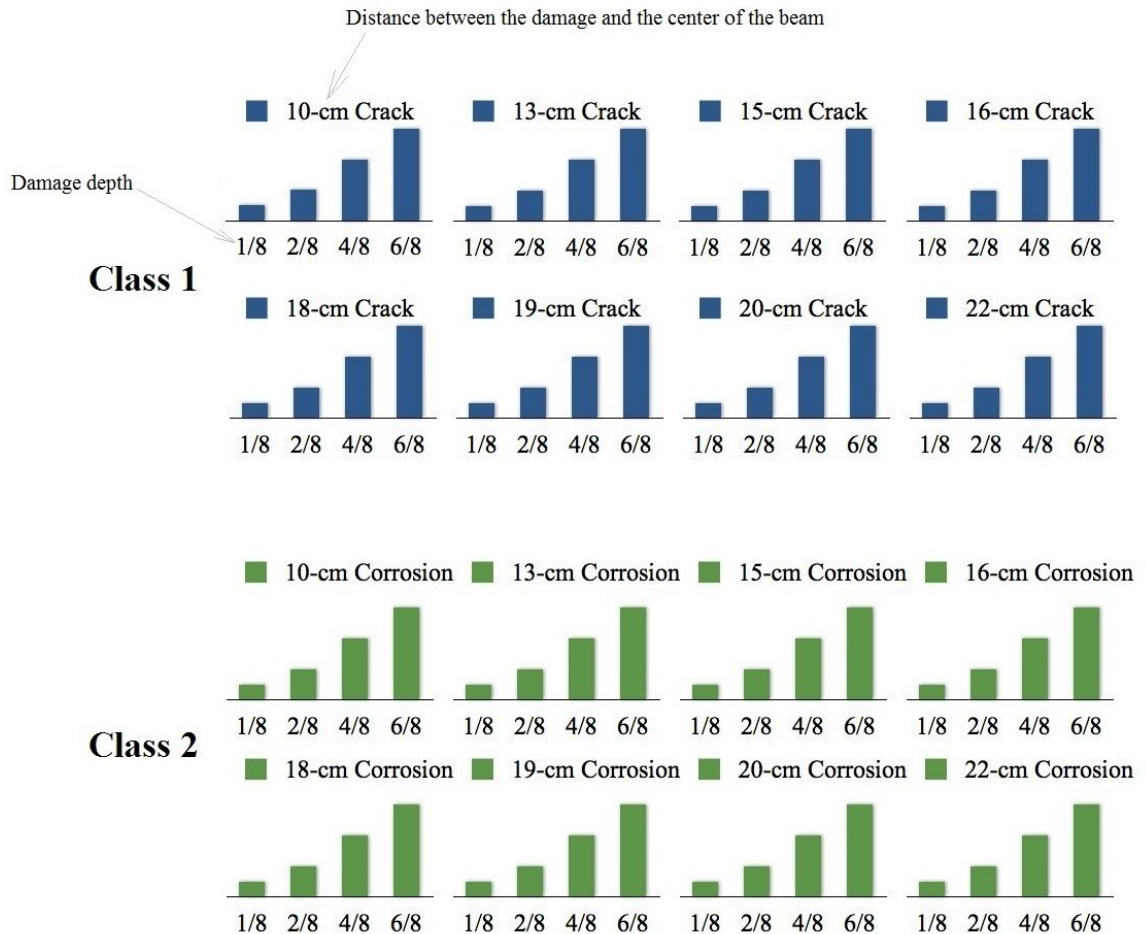
Figure 39. The generation of a feature vector from a spectrogram

4.3 Two-Class Classification and Results

The algorithm for transforming a spectrogram to a feature vector was demonstrated in the previous section. Feature vectors of other damage scenarios can be obtained using the same algorithm. In this research, 80 damage samples (40 cracks and 40 corrosions) have been simulated using ABAQUS[®]. The distances between the damages and the PZT actuators of these 80 samples vary from 10-cm to 22-cm. The depths of the damages vary

among $1/8$, $2/8$, $3/8$, $4/8$ and $6/8$ of the beam thickness. In the two-class classification, 64 out of the 80 samples are used as training samples. The damage types, locations, and depths of these training samples are listed in Table 5.

Table 5. 64 training samples for the SVM two-class classification

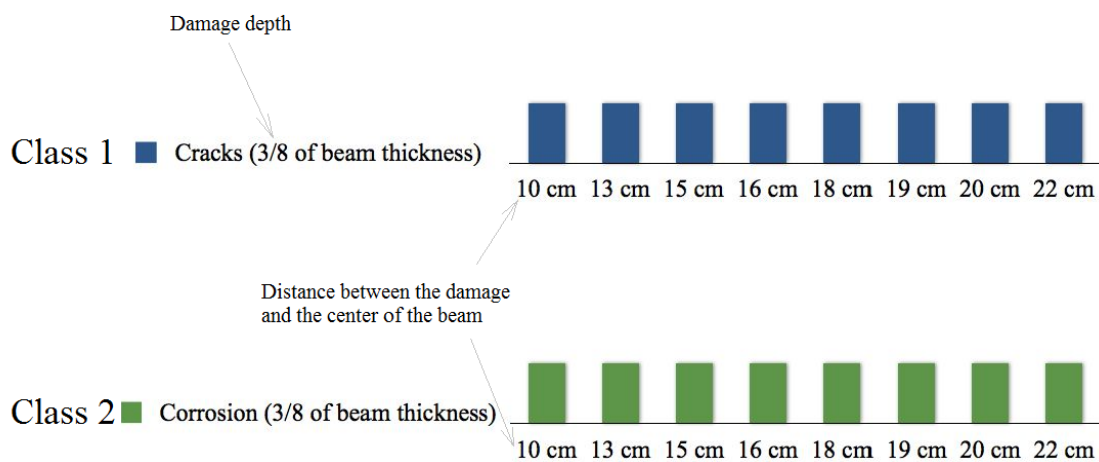


16 out of the 80 damage samples are used as testing sample. The distances from the PZT actuators to the damages of these testing samples vary from 10-cm to 22-cm. The damage thicknesses of the testing samples are $3/8$ of the beam thickness. The damage type, location, and depth of these testing samples are listed in Table 6. All the

spectrograms of the training and testing samples for the two-class classification are shown in Appendix A.

In the training process of the two-class SVM classification, all the 64 training samples are utilized to train a two-class classifier. Cross validation accuracy is checked in the training process because high cross validation accuracy is also expected. In the testing process, each of the 16 testing samples is predicted by the classifier. The testing sample's predicted label is compared with its original label to examine if the prediction is correct.

Table 6. 16 testing samples for the SVM two-class classification



A two-class damage classification algorithm is developed based upon support vector machine theory on the MATLAB[®] platform. The computer utilized to run the SVM classification code is a SONY[®] VPCCW21FX laptop equipped with an INTEL[®] i3 2.13 GHz processor and 4 GB RAM. The operation system is a 64-bit Windows 7 Ultimate Edition. It took the computer 37.49 seconds to complete running the code. The cross validation accuracy was 89.06%. All the 16 testing samples are correctly classified.

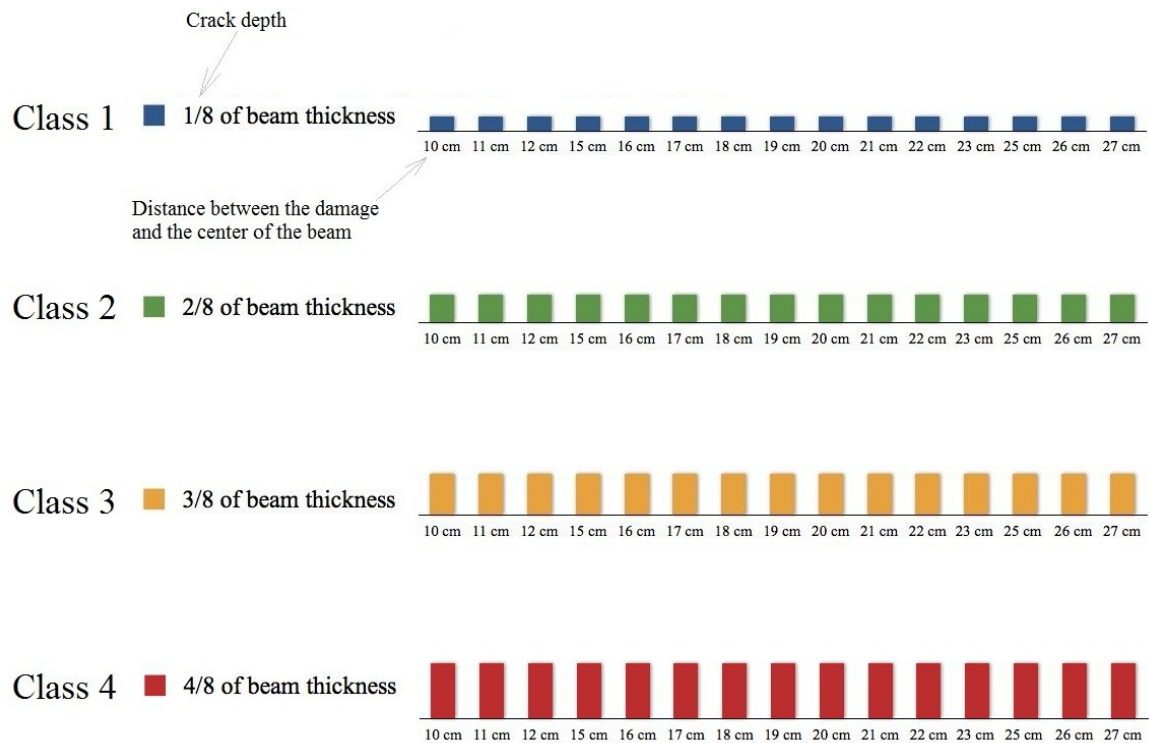
Therefore, it can be concluded this two-class classifier exhibited good generalization capability.

4.4 Multiple-Class Classification and Results

As mentioned in Chapter 1, crack detection is vital for the aerospace industry because catastrophic consequences are usually incurred if an initial crack evolves to a critical size. Therefore, a multiple-class damage classification method is developed to evaluate the evolution extent of a crack.

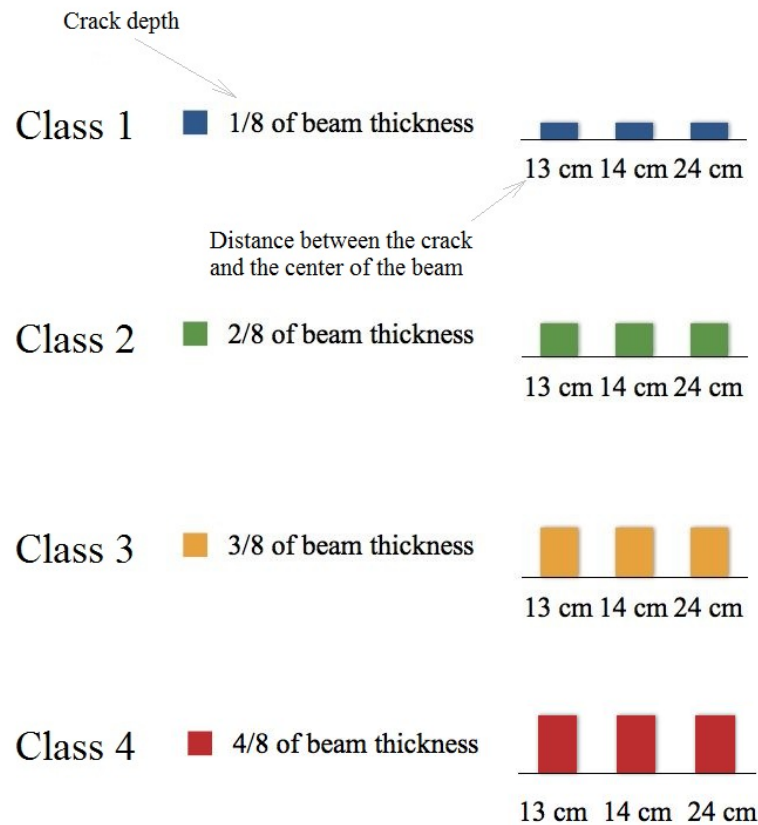
As discussed in the previous section, the damage samples are divided into only two classes by the damages' type in the two-class classification. For the multiple-class classification, however, all the damage samples are divided into 4 classes by the damage depths. The first class consists of damages with depth of $1/8$ of the beam thickness. Similarly, the second, third, and fourth class consist of damages with depths of $2/8$, $3/8$ and $4/8$ of the beam thickness, respectively. In this research, 60 crack samples are used as training samples. The damage locations of the 60 samples vary from 10-cm to 27-cm. The features of these training samples, including damage type, location, and thickness, are listed in Table 7.

Table 7. Training samples for the 4-class classification



A 4-class classifier is established by training the 60 training samples listed in Table 7. The 12 damage samples in Table 8 are used as testing samples to examine the performance of the classifier. The damage locations of testing samples vary among 13-cm, 14-cm, and 24-cm. The thicknesses of the testing samples vary among 1/8, 2/8, 3/8, and 4/8 of the beam thickness.

Table 8. Testing Samples for the 4-class classification



As previously mentioned, the energy density is normalized by the maximum energy density in each spectrogram for the two-class classification. It means that the maximum energy density in any spectrogram is always normalized to 1, no matter what its actual value is. It also implies that the points that represent the maximum energy densities in different spectrograms always exhibit the same color (the most intense color), no matter how much difference exists among the actual values of the energy densities. This is because, for the two-class classification, the outlines of spectrograms are more concerning than the energy densities. Figure 40 (a) shows the spectrogram of a crack of $d=15$ -cm and $\text{depth}=1/8$ of the beam thickness. Figure 40 (b) shows the spectrogram of a crack of $d=15$ -cm and $\text{depth}=4/8$ of beam thickness. Although the maximum energy

density in the second crack is considerably larger than the first crack, the maximum energy density regions in both spectrograms have the similar “bronzing” color, as highlighted inside the white rectangles in Figure 40 (a) and 40 (b).

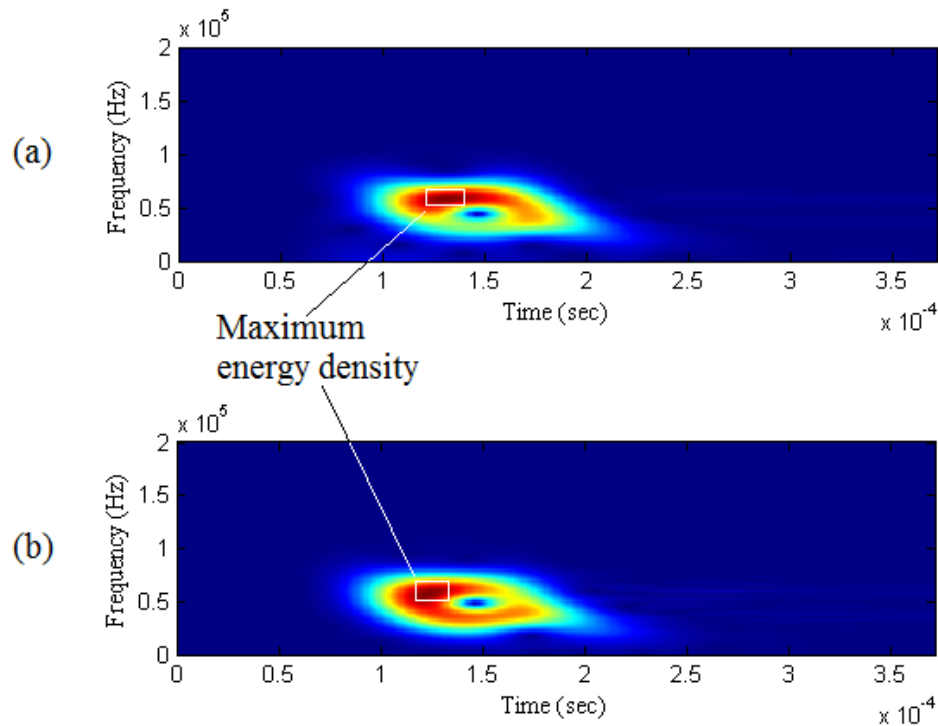


Figure 40. The locations of the maximum energy densities in two different spectrograms for the two-class classification (a) spectrogram of a crack ($d=15$ -cm and depth= $1/8$ of beam thickness) (b) spectrogram of another crack ($d=15$ -cm and depth= $4/8$ of beam thickness)

To conduct the multiple-class classification, the energy densities in a spectrogram are no longer normalized by the maximum energy density in the spectrogram. Instead, they are normalized by some predefined maximum energy densities. This implies that points that represent the maximum energy densities in different spectrograms no longer have the same intense color. This is because the energy density's actual values of the

reflected signals are essential to distinguish damages with different depths. If the excited signals are the same, the signal reflected by a deep (or severe) damage generally has more intense energy density than a superficial (or initial) damage.

The spectrograms normalized by a predefined maximum energy density of 4 crack signals are illustrated in Figure 41. In these damage scenarios, the distances between cracks and PZT actuators are 15-cm. The depths of cracks vary from 1/8 to 1/2 of the beam thickness. It can be observed that the colors in the spectrogram become more intense as the crack depth increases. This is because the energy density of the reflected signal becomes stronger as the damage depth increases. The spectrograms in Figure 41 can be compared with the spectrograms (normalized to 1) in Figure 36 to examine the differences.

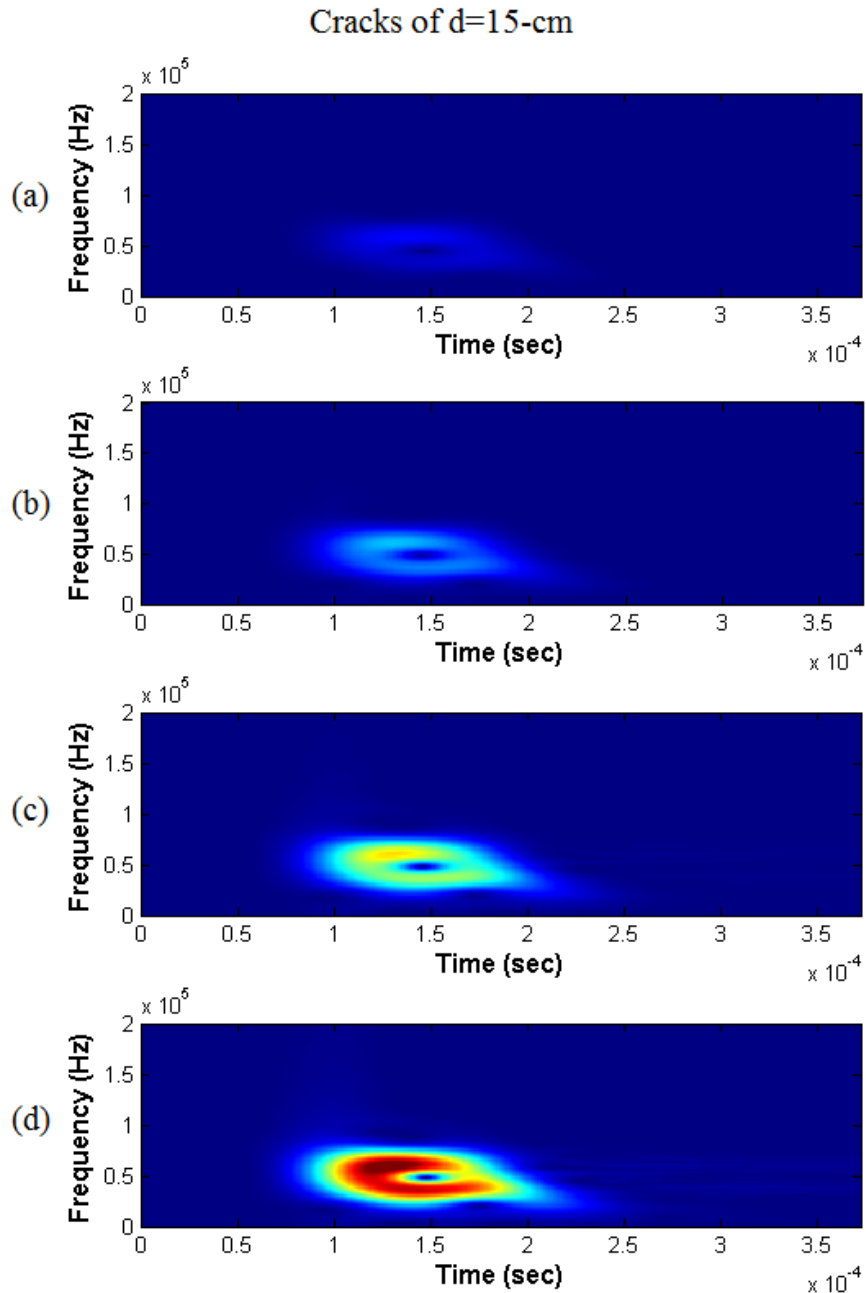


Figure 41. Spectrograms of different crack signals ($d=15\text{-cm}$) for the multiple-class classification (a) the spectrogram of a crack (1/8 of the beam thickness) (b) the spectrogram of a crack (2/8 of the beam thickness) (c) the spectrogram of a crack (3/8 of the beam thickness) (d) the spectrogram of a crack (4/8 of the beam thickness)

A 4-class classifier is established based upon the training samples in Table 7. The classifier is used to classify unknown testing samples into one of the 4 classes. Equivalently, the classifier is used to predict the thickness of a crack sample. Just like the two-class classification, a multiple-class damage classification algorithm is developed based upon Support Vector Machine theory on the MATLAB[®] platform. The cross validation accuracy is examined in the multiple-class classification. It took the computer 15.83 seconds to run the SVM classification code. The cross validation accuracy was 70.00%. Only 9 out of the 12 testing samples are correctly classified. It turns out that the cross validation accuracy is not satisfying.

To analyze the reason for the poor cross validation accuracy, the 15 training samples of Class 1 in Table 4.3 are taken out from the 60 training samples. The training samples of Class 2, Class 3, and Class 4 are used to train a three-class classifier. It took the computer 11.23 seconds to run the code. The cross validation of this 3-class classifier was 91.11%. All the 9 testing samples were correctly classified 100%. Similarly, more combinations of the training samples are used to train some more three-class classifiers, and the results are shown in Table 9.

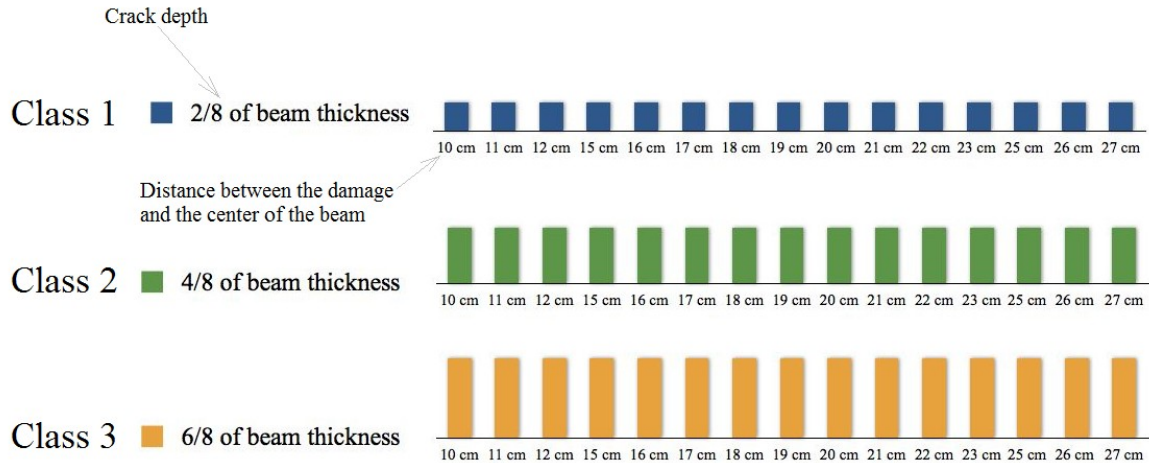
Table 9. The cross validation accuracy, ratio of correctly classified testing samples, and running time for various training cases

Case Number	Training Sample Combination	Cross Validation Accuracy	Correctly classified testing samples	Code Running Time (seconds)
1	Class 1, 2, 3, 4	70.00%	12/16	15.83
2	Class 2, 3, 4	91.11%	9/9	11.23
3	Class 1, 2, 3	62.22%	6/9	12.18
4	Class 1, 2, 4	66.67%	6/9	12.28
5	Class 1, 3, 4	91.11%	9/9	11.40

From Table 9, it turns out that the cross validation accuracies are generally low if the training samples contain samples from both Class 1 and Class 2, such as Case 1, 3, and 4 in Table 9. However, a three-class classifier can provide good cross validation accuracy, as long as the samples from Class 1 and Class 2 are not used for training simultaneously. For examples, Case 2 and Case 5 in Table 9 exhibit good cross validation accuracy.

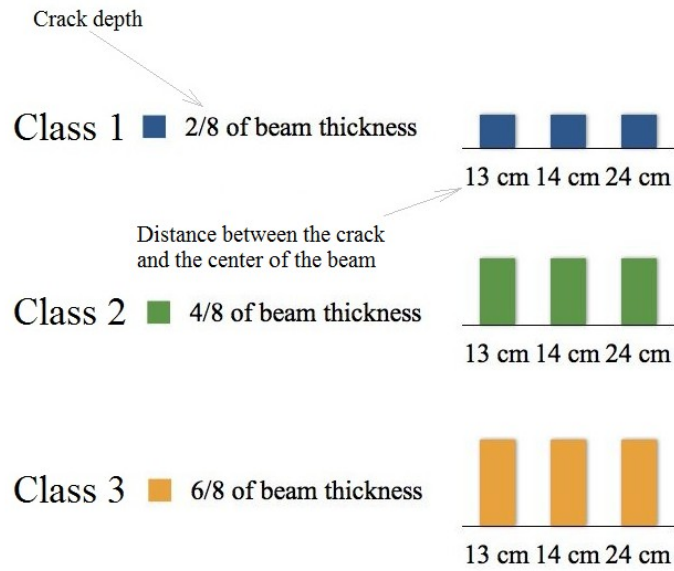
To prove this assumption, the cracks with 1/8 of the beam thickness are removed from training samples, and the cracks of 6/8 of beam thickness are introduced. The cracks with 2/8, 4/8, and 6/8 of the beam thicknesses are used as training samples to conduct another three-class damage classification. The features including crack depth, location of 45 training samples are listed in Table 10.

Table 10. The training samples for a three-class classifier including cracks of $2/8$, $4/8$, and $6/8$ of the beam thickness



Nine testing samples used in this three-class classification are listed in Table 11. The damage locations of testing samples vary among 13-cm, 14-cm, and 24-cm. The depths of testing samples vary among $2/8$, $4/8$, and $6/8$ of the beam thickness. All the spectrograms of the training and testing samples for this multiple-class classification are shown in Appendix B.

Table 11. Testing samples for a three-class classifier including cracks of $2/8$, $4/8$, and $6/8$ of beam thickness



This time, it took the computer 10.98 seconds to run the code and generate the three-class classifier. The cross validation accuracy was 88.89%. All the samples in Table 4.5 are correctly classified using this classifier. Therefore, it can be concluded that this multiple-class classification method exhibits poor performance if the training samples contain more than one class of superficial damages (such as the cracks with depths of 1/8 or 2/8 of the beam thickness). However, this method exhibits good performance if the training samples contain only one class of superficial damage.

Chapter 5

5. Conclusion and Recommendations

Structural health monitoring is a novel damage detection and identification methodology in the aerospace, civil, and mechanical engineering industries. Structural health monitoring is generally described by steps including damage detection, diagnosis, and prognosis. In the first step, the existence and location of damage are evaluated. In the second step, the damage type and extent of damage are assessed. In the third step, the remaining useful life of the structural system is predicted. The damage classification in the diagnosis step is important because it is a premise to the prediction of the remaining life of the structure. Damage classification methods have been extensively investigated using various machine learning algorithms. However, not much attention has been paid to multiple-class damage classification. In this research, a multiple-class damage classification method has been developed based upon the SVM theory and the guided wave technique.

5.1 Summary and conclusion

To develop the multiple-class damage classification method, the guided wave based structural health monitoring method is investigated. In an experiment using the guided wave method, PZT actuators are used to excite a guided wave signal that propagates

along an aluminum beam. Some portions of the excited signal are reflected by damage on the beam. The reflected damage signal is examined by measuring the in-plane displacement of the beam at a location close to the PZT actuators.

Damage signals are transformed to time-frequency representations to better understand the characteristics of the signals. Several time-frequency analysis methods including short time Fourier transform, Wigner-Ville distribution, and Wavelet transform are investigated. The time-frequency plots using these methods are compared with one another. By comparisons, the spectrogram based upon the short time Fourier transform is chosen.

To develop a database for damage signals, a finite element tool named ABAQUS[®] is used to simulate various damage scenarios. The damage signals obtained from the ABAQUS[®] simulations are transformed into spectrograms using the short time Fourier transform. A damage classification algorithm is developed based upon Support Vector Machine theory on the MATLAB[®] platform. Support vector machine classifiers are established by inputting training samples to the algorithm. A two-class classifier is generated to predict the testing samples' type. A multiple-class classifier is generated to predict the damage extent of cracks. The performance and accuracy of the classifiers are finally examined. The two class classifier exhibits good classification accuracy. The multiple-class classifier also shows good accuracy as long as the training samples contain only one class of superficial damage.

5.2 Recommendations

As demonstrated in Section 4.5, the accuracy of the multiple-class classification is affected by superficial damage samples. To solve this problem, a coarse-fine classification method is recommended. In the method, a damage signal is first examined by a coarse multiple-class classification method to determine the approximate depth of the damage. Then, the damage signal is substituted into a finer two-class classification to predict the more accurate depth of the damage.

In this research, only 80 damage samples have been simulated in ABAQUS[®] due to the limitations of time and hardware. However, considerably more training samples are utilized in typical Support Vector Machine classifications. Therefore, if possible, more damage samples are recommended to be generated and used in the future work,

As discussed in Section 4.3, the signal frequency and energy density distribution in spectrograms are used to generate the training and testing data for the Support Vector Machine classification. It is desired that more features are extracted from damage signals and supplemented to the training and testing data. By doing so, the classification accuracy is suspected to improve.

In this research, the support vector machine multiple-class classification is utilized for evaluating the cracks' extents. Instead of outputting some discrete class labels, the support vector machine regression can be used to output continuous numbers. Therefore, for future work, it is recommended to develop a support vector machine regression algorithm to predict the actual depths of the cracks.

References:

- [1] *The scene after an expressway collapsed this morning in Harbin, Heilongjiang province, China.* N.d. The GuardianWeb. <<http://www.guardian.co.uk/news/2012/aug/24/best-news-pictures-of-the-day>>.
- [2] *Aloha Airlines Flight 243 at Kahului Airport on April 28, 1988, after its fuselage was ripped apart during flight.* . N.d. The Epoch TimesWeb. <<http://www.theepochtimes.com/n2/science/oxygen-in-the-skies-30202.html>>.
- [3] Shull, Peter J. *Nondestructive Evaluation Theory, Techniques, and Applications.* New York: Marcel Dekker, Inc., 2002.
- [4] Sohn, H., C.R. Farrar, et al. *A review of structural health monitoring literature: 1996-2001.* Los Alamos: Los Alamos National Laboratory, 2004.
- [5] Balageas, Daniel, Claus-Peter Fritzen, and Alfredo Güemes. *Structural Health Monitoring.* Wiley-ISTE, 2006.
- [6] *The Largest Civil Infrastructure Monitoring System to Date, Deployed on Jindo Bridge in South Korea.* N.d. University of Illinois Board of Trustees. <<http://www.iti.illinois.edu/tech-transfer/technologies/wireless-monitoring-critical-infrastructures>>.
- [7] *Intelligent load-monitoring sensor technologies are being developed to detect or calculate the residual fatigue life and structural health of a composite wing.* N.d. SAE International. <Intelligent load-monitoring sensor technologies are being developed to detect or calculate the residual fatigue life and structural health of a composite wing>.
- [8] Johnson, Stephen B., Thomas Gormley, Seth S. Kessler, Ann Patterson-Hine, Karl Reichard, and Philip A. Scandura. *System Health Management: with Aerospace Applications.* Wiley; 1st edition, 2011.
- [9] Rytter, A. *Vibration based inspection of civil engineering structures.* Ph.D Dissertation, Department of Building Technology and Structural Engineering, Aalborg University, Denmark, 1993.
- [10] Doebling, Scott W., Charles R. Farrar, et al. "Summary review of vibration-based damage identification methods." *Shock and Vibration Digest, Sage Sci Press*, 91-105, 1998.
- [11] Wang, Liang and Chan, Tommy H.T., "Review of vibration-based damage detection and condition assessment of bridge structures using structural health monitoring." *The Second Infrastructure Theme Postgraduate Conference: Rethinking Sustainable Development: Planning, Engineering, Design and Managing Urban Infrastructure*, 2009.
- [12] F. P. Sun, Z. Chaudhry, C. Liang, and C. A. Rogers, "Truss structure integrity identification using PZT sensor-actuator," *Journal of Intelligent Material Systems and Structures*, vol. 6, pp. 134-139, 1995.

- [13] G. Park, H. Sohn, C. R. Farrar, and D. J. Inman, "Overview of piezoelectric impedance-based health monitoring and path forward," *Shock and Vibration Digest*, vol. 35, pp. 451-463, 2003.
- [14] Rim, Mi-Sun, Seung-Jae Yoo, et al. "Impedance-based damage assessment using piezoelectric sensors." *Proceedings of SPIE - The International Society for Optical Engineering*, v 7981, 2011.
- [15] Rose, Joseph L. "A baseline and vision of ultrasonic guided wave inspection potential." *Journal of Pressure Vessel Technology, Transactions of the ASME*, v 124, pp. 273-282, 2002
- [16] A. Raghavan and C. E. S. Cesnik, "Finite-dimensional piezoelectric transducer modeling for guided wave based structural health monitoring," *Smart Materials and Structures*, vol. 14, pp. 1448-1461, 2005.
- [17] Zhao, Xiaoliang, and Joseph L. Rose. "Boundary element modeling for defect characterization potential in a wave guide." *Elsevier Ltd.* vol.40, pp. 2645-2658,2003.
- [18] Su, Zhongqing, Lin Ye, and Ye Lu. "Journal of Sound and Vibration." *Journal of Sound and Vibration*. v 295, pp. 753-780, 2006.
- [19] Ho, K.S., D.R. Billson, and D.A. Hutchins. "Journal of FoodEngineering." *Journal of Food Engineering*. v 80, pp. 431-444, 2006.
- [20] Vives, Antonio Arnau. Piezoelectric transducers and applications. 2. New York: Springer, 2008.
- [21] APC International, Ltd., Piezoelectric ceramics: principles and applications. *Mackeyville, PA: APC International*, 2002.
- [22] Tichý, Jan, Jiří Erhart, Erwin Kittinger, and Jana Přívratská. Fundamentals of Piezoelectric Sensorics. *Berlin Heidelberg: Springer*, 2010.
- [23] Long, Liu, and Meng Guang. "Localization of damage in beam-like structures by using support vector machine." *Proceedings of International Conference on Neural Networks and Brain Proceedings*, v.2, 919-924, 2005.
- [24] Esterline, Albert, Kiran Krishnamurthy, Mannur Sundaresan, Tariqul Alam, Duwarahan Rajendra, and William Wrigft. "Classifying acoustic emission data in structural health monitoring using support vector machines." *AIAA Infotech at Aerospace*, 2010.
- [25] Das, Santanu, Ashok N. Srivastava, and Aditi Chattopadhyay. "Classification of damage signatures in composite plates using one-class SVMs." *IEEE Aerospace Conference Proceedings*, 2007.
- [26] Ying, Yujie, Joel Harley, James H. Garrett Jr., Yuanwei Jin, Irving J. Oppenheim, Jun Shi, and Lucio Soibelman. "Applications of machine learning in pipeline

monitoring." *Congress on Computing in Civil Engineering, Proceedings*, pp. 242-249, 2011 .

[27] Michaels, Thomas E., and Michaels, Jennifer E.. "Monitoring and characterizing corrosion in aluminum using Lamb waves and attached sensors." *Proceedings of SPIE - The International Society for Optical Engineering*, v 6532, 2007.

[28] Dua, R., S.E. Watkins, D.C. Wunsch, K. Chandrashekhara, and F. Akhavan. "Detection and classification of impact-induced damage in composite plates using neural networks." *Proceedings of the International Joint Conference on Neural Networks*, v 1, pp. 681-686, 2001.

[29] Kim, Daewon and Philen, Michael. "Damage classification using adaboost machine learning for structural health monitoring." *Proceedings of SPIE - The International Society for Optical Engineering*, v 7981, 2011.

[30] Carpinter, A., G. Lacidogna, and N. Pugno. "Structural damage diagnosis and life-time assessment by acoustic emission monitoring." *Engineering Fracture Mechanics*. v 74, pp. 273-289, 2007.

[31] Yuan, Sheng-Fa, and Fu-Lei Chu. "Support vector machines-based fault diagnosis for turbo-pump rotor." *Mechanical Systems and Signal Processing*, v 20, pp. 939-952, 2005.

[32] Wang, Qiang, Changmin Yuan, and Jianyun Zhu. "Buried pipeline third-party damage signals classification based on LS-SVM." *Proceedings of the World Congress on Intelligent Control and Automation*, v 1, pp. 5032-5036, 2006.

[33] Widodo, Achmad, and Bo-Suk Yang. "Wavelet support vector machine for induction machine fault diagnosis based on transient current signal." *Expert Systems with Applications*. v 35, pp. 307-316, 2008.

[34] Banerjee, Tribeni Prasad, and Swagatam Das. "Multi-sensor data fusion using support vector machine for motor fault detection." *Information Sciences*. v 217, pp. 96-107, 2012.

[35] Nguyen, Minh, Xiaoming Wang, Zhongqing Su, and Lin Ye. "Damage identification for composite structures with a Bayesian network." *Proceedings of the 2004 Intelligent Sensors*, pp. 307-311, 2004.

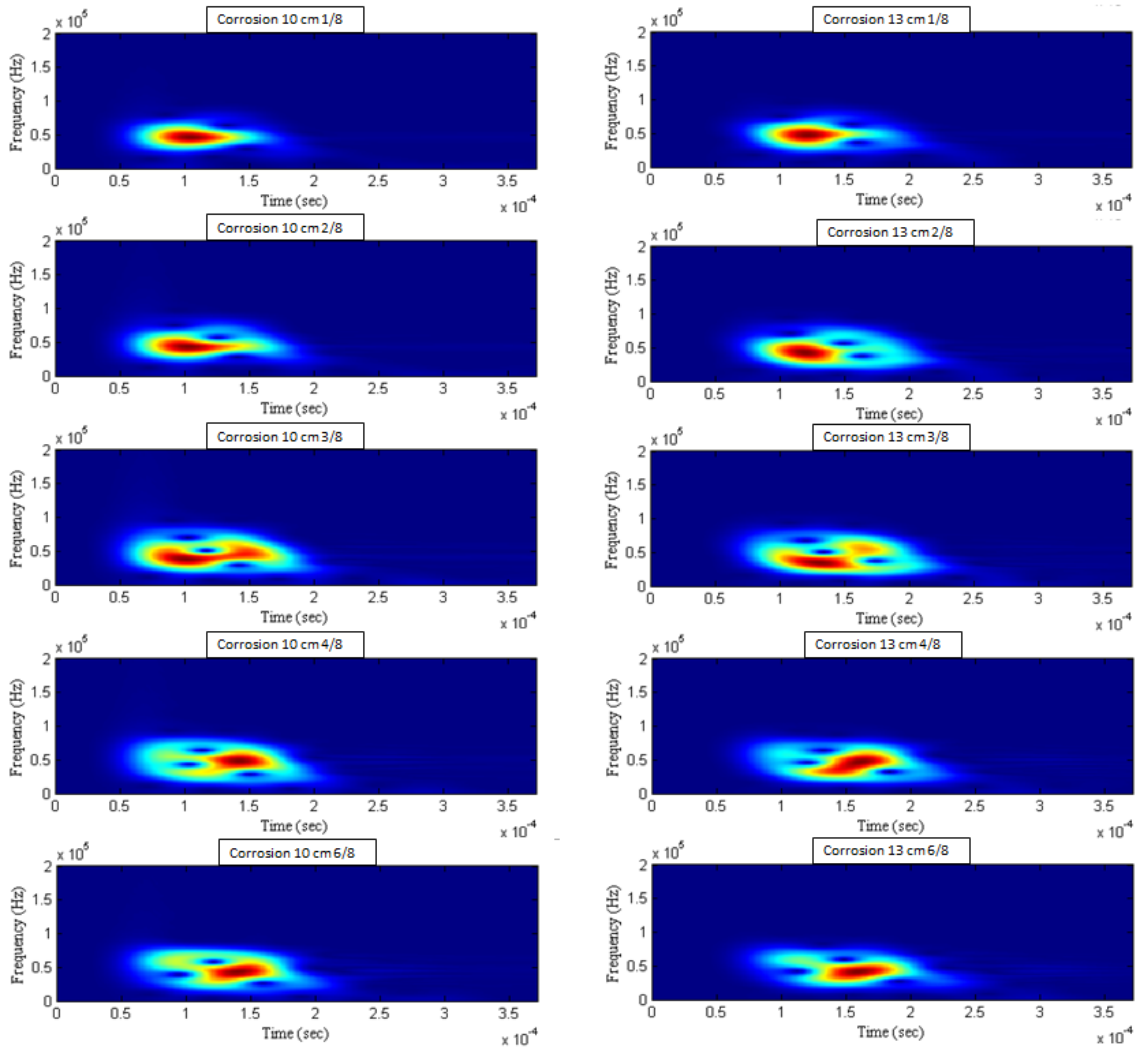
[36] Coelho, Clyde K., Santanu Das, and Aditi Chattopadhyay. "Binary tree SVM based framework for mining fatigue induced damage attributes in complex lug joints." *Proceedings of SPIE - The International Society for Optical Engineering*, v 6926, 2008.

[37] Vines-Cavanaugh, David, Yinghong Cao , and Ming L. Wang. "Support vector machine for abnormality detection on a cable-stayed bridge." *Proceedings of SPIE - The International Society for Optical Engineering*, v 7647, 2010.

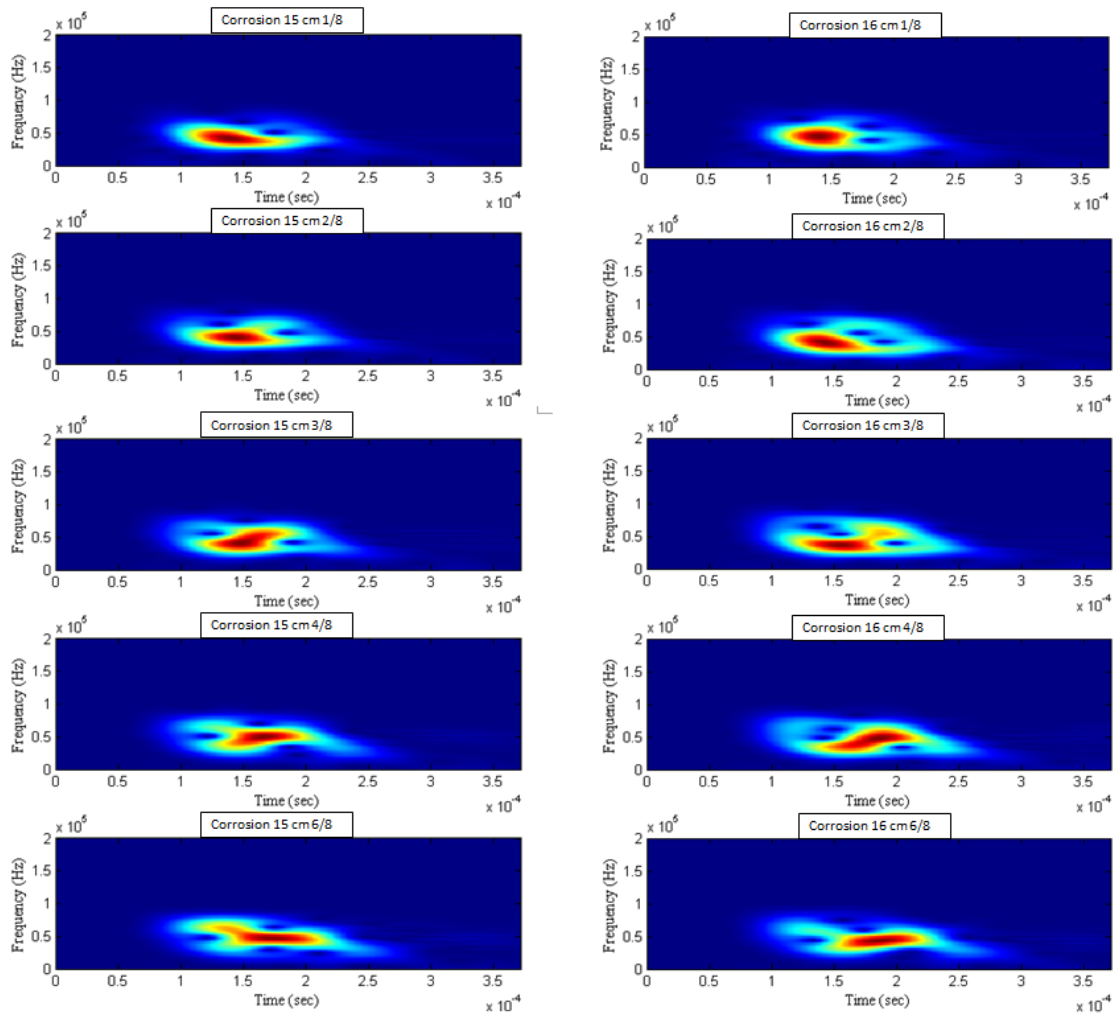
- [38] Farooq , M., H. Zheng, A. Nagabhushana, S. Roy na, S. Burkett , M. Barkey, S. Kotru, and E. Sazonov. "Damage detection and identification in smart structures using SVM and ANN." *Proceedings of SPIE 8346, Smart Sensor Phenomena, Technology, Networks, and Systems Integration*, 2012.
- [39] Xiao, Li, and Wenzhong Qu. "Nonlinear structural damage detection using support vector machines." *Proceedings of SPIE - The International Society for Optical Engineering*, v 8348, 2012.
- [40] Matic, Dragan, Filip Kulic, Manuel Pineda-Sánchez, , and Ilija Kamenko. "Support vector machine classifier for diagnosis in electrical machines: Application to broken bar." *Expert Systems with Applications*, v 39, pp. 8681-8689, 2012.
- [41] Xie, Jianhong. "Improved least square support vector machine for structural damage detection." *ICCET 2010 - 2010 International Conference on Computer Engineering and Technology*, v 6, pp. 6237-6240, 2010.
- [42] Yang, Yan, and Tian-Yi Liu. "Application of support vector machine in structure damage identification." *Proceedings - 2009 International Conference on Information Engineering and Computer Science*, 2009.
- [43] Vapnik, Vladimir, Steven E. Golowich, and Alex Smola. "Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing." *Advances in Neural Information Processing Systems*, pp. 281-287, 1997
- [44] Cho, Sohyung, Shihab Asfour, Arzu Onar, and Nandita Kaundinya. "Tool breakage detection using support vector machine learning in a milling process." *International Journal of Machine Tools and Manufacture*, v 45, pp. 241-249, 1997
- [45] Oh, Chang Kook, and Hoon Sohn. "Damage diagnosis under environmental and operational variations using unsupervised support vector machine." *Journal of Sound and Vibration*, v 325, pp. 224-239, 2009
- [46] Hsu, Chih-Wei, Chih-Chung Chang, and Chih-Jen Lin. "Department of Computer Science National Taiwan University." *Department of Computer Science National Taiwan University*, 2010.
- [47] L. Cohen, Time-frequency Analysis. Upper Saddle River: Prentice-Hall. Inc., 1995.
- [48] W. J. Staszewski and A. N. Robertson, "Time–frequency and time–scale analyses for structural health monitoring," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, pp. 456-460, 2007.
- [49] Addison, Paul. *The Illustrated Wavelet Transform Handbook*. 1. 2. New York: Taylor & Francis Group, pp. 51-55, 2002.

APPENDIX A. Spectrograms of Training and Testing Samples for Two-class Classification

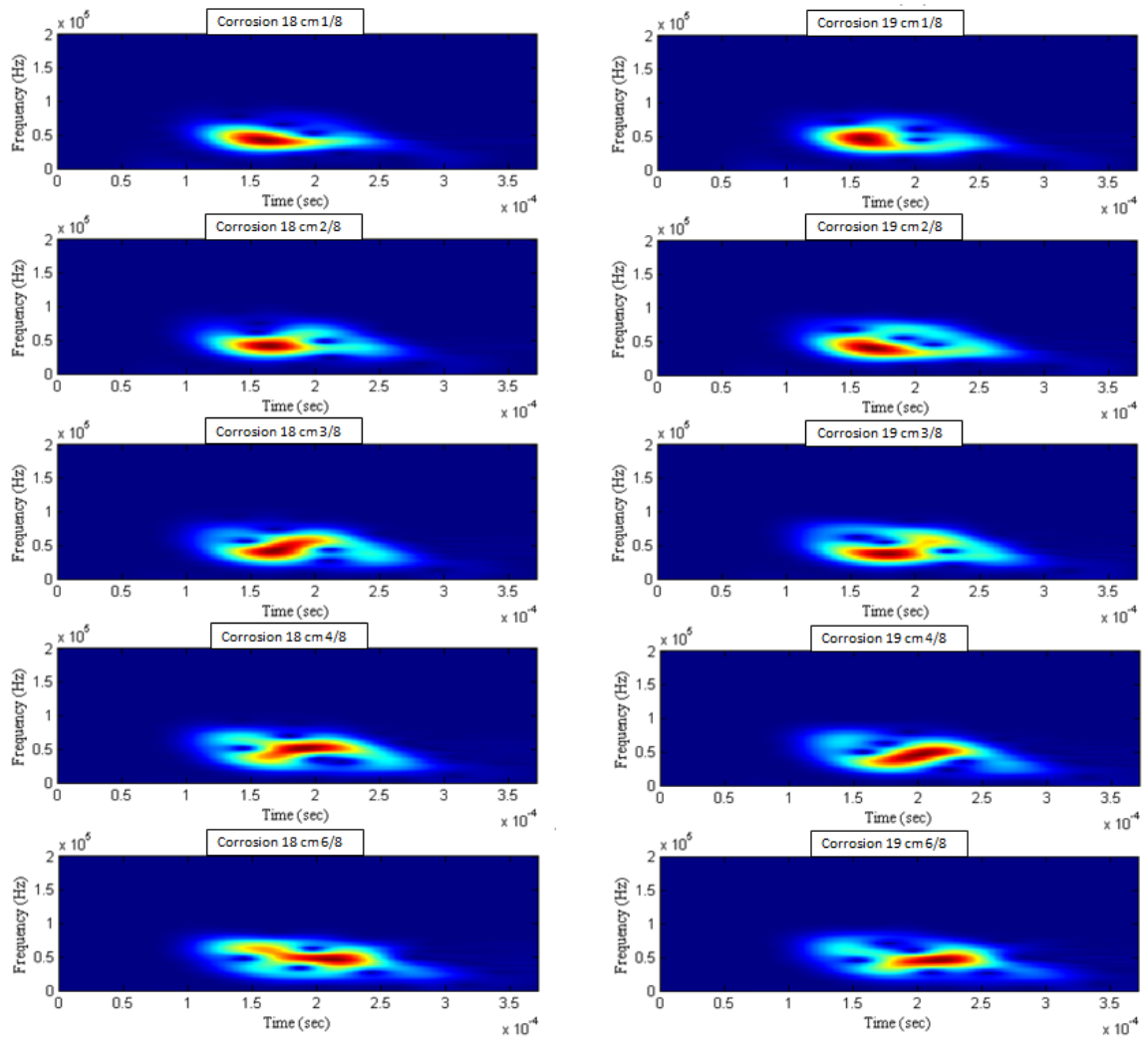
1. Corrosion signal spectrograms of d=10-cm and d=13-cm



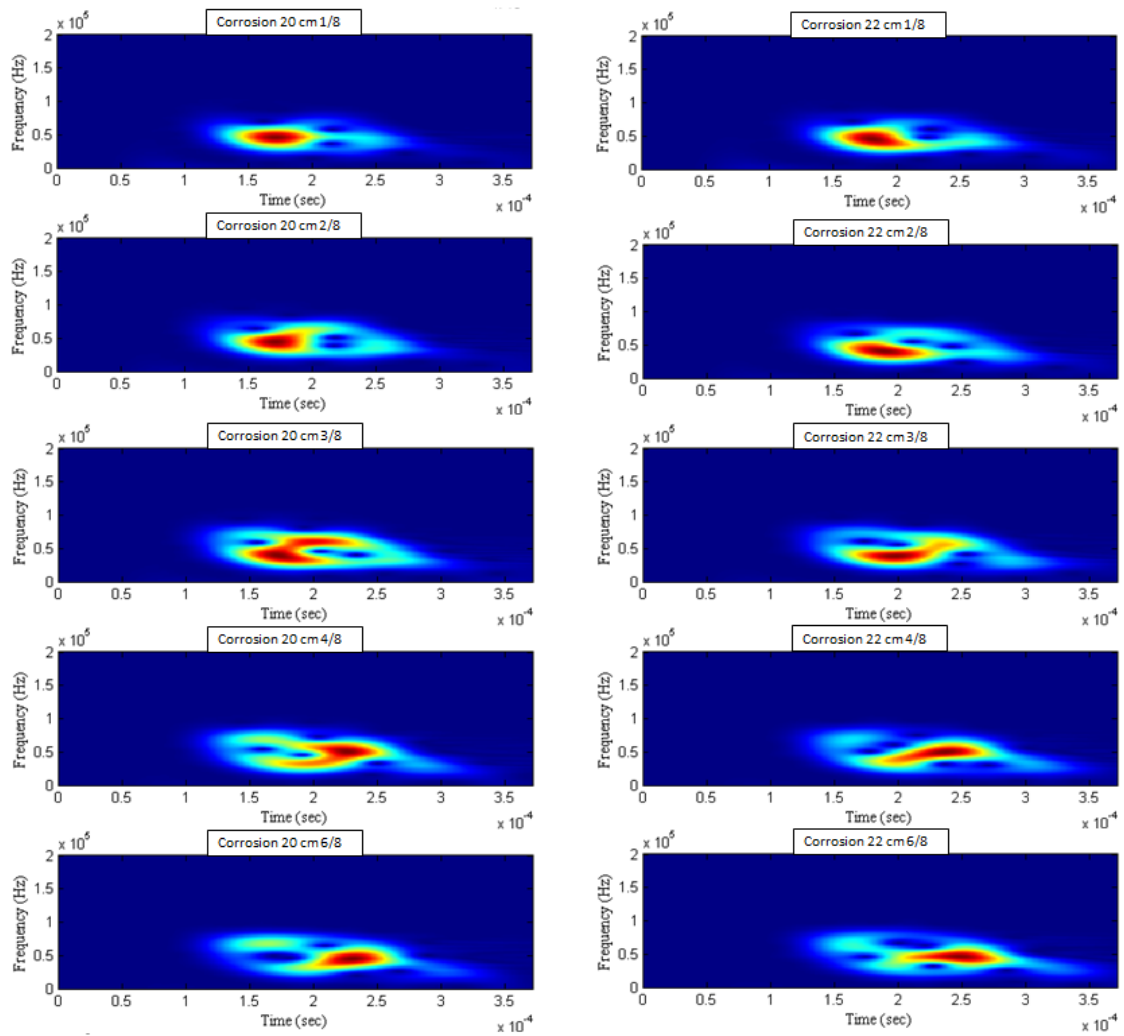
2. Corrosion signal spectrograms of d=15-cm and d=16-cm



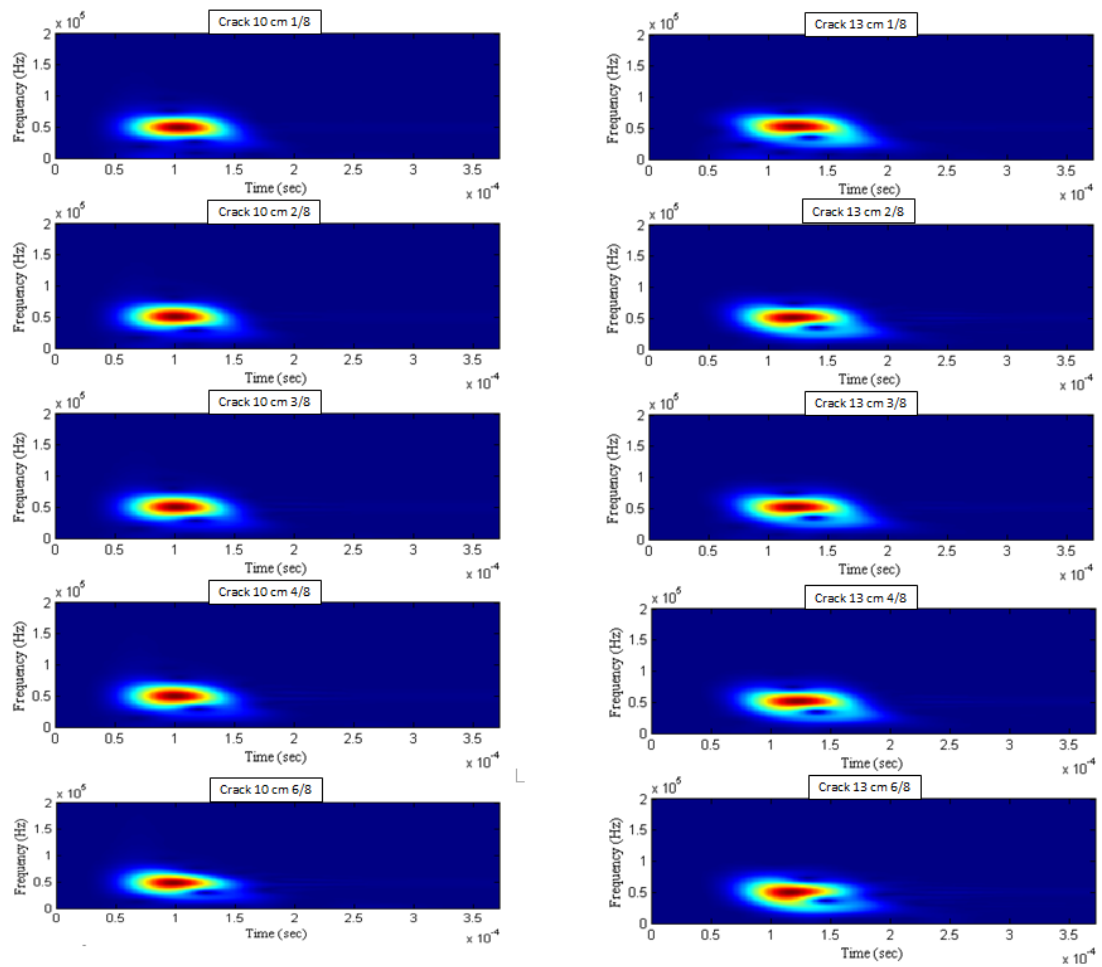
3. Corrosion signal spectrograms of d=18-cm and d=19-cm

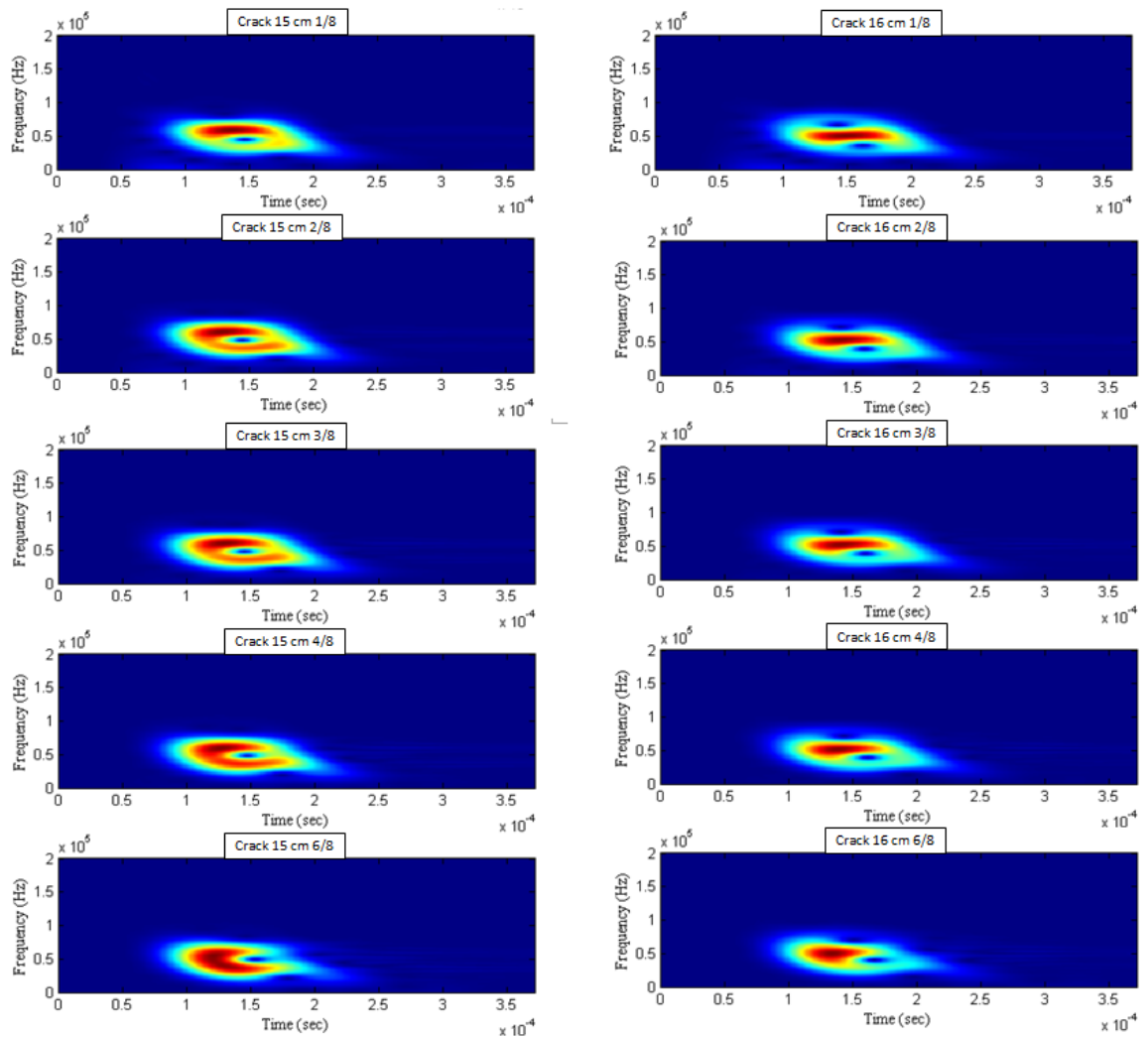


4. Corrosion signal spectrograms of d=20-cm and d=22-cm

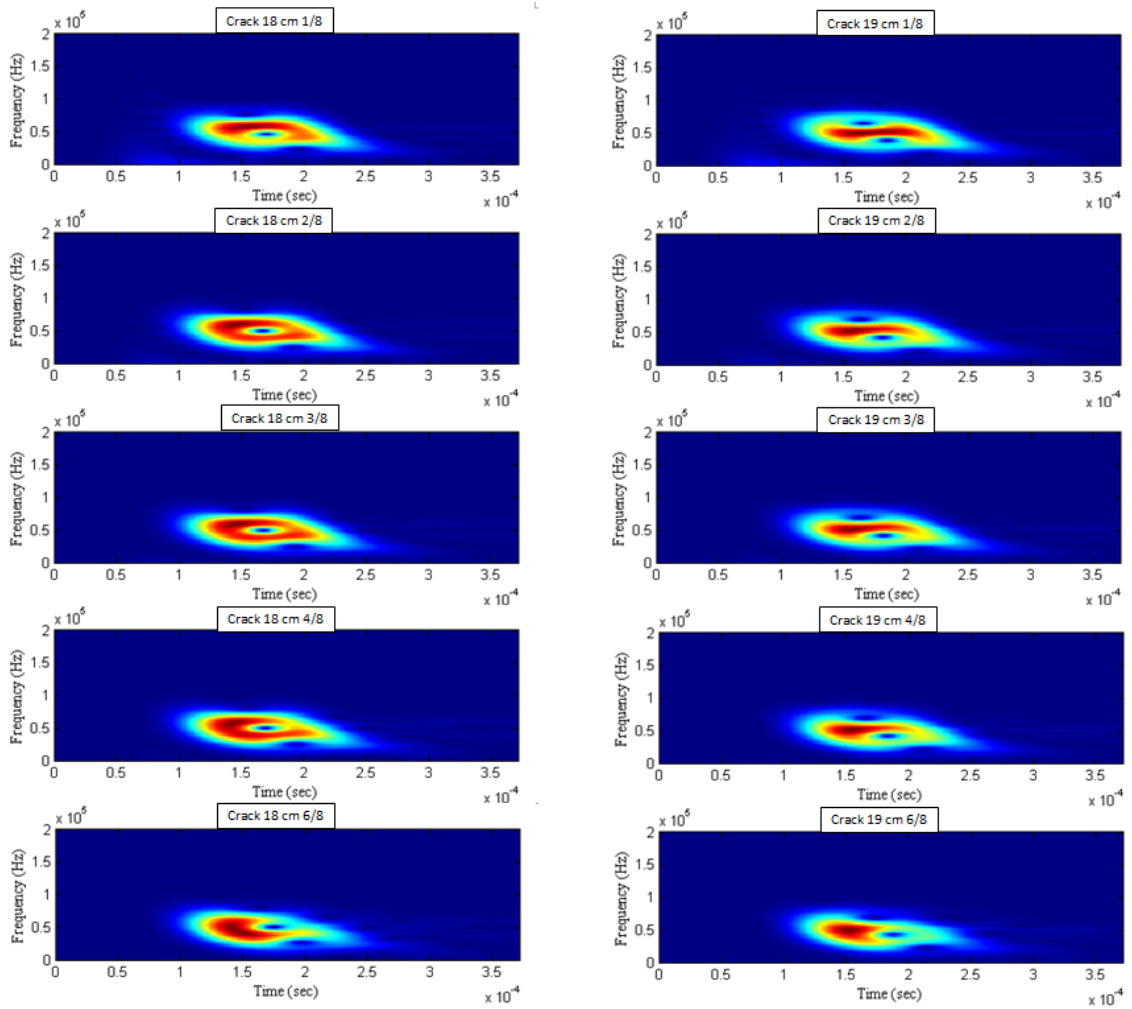


5. Crack signal spectrograms of d=10-cm and d=13-cm

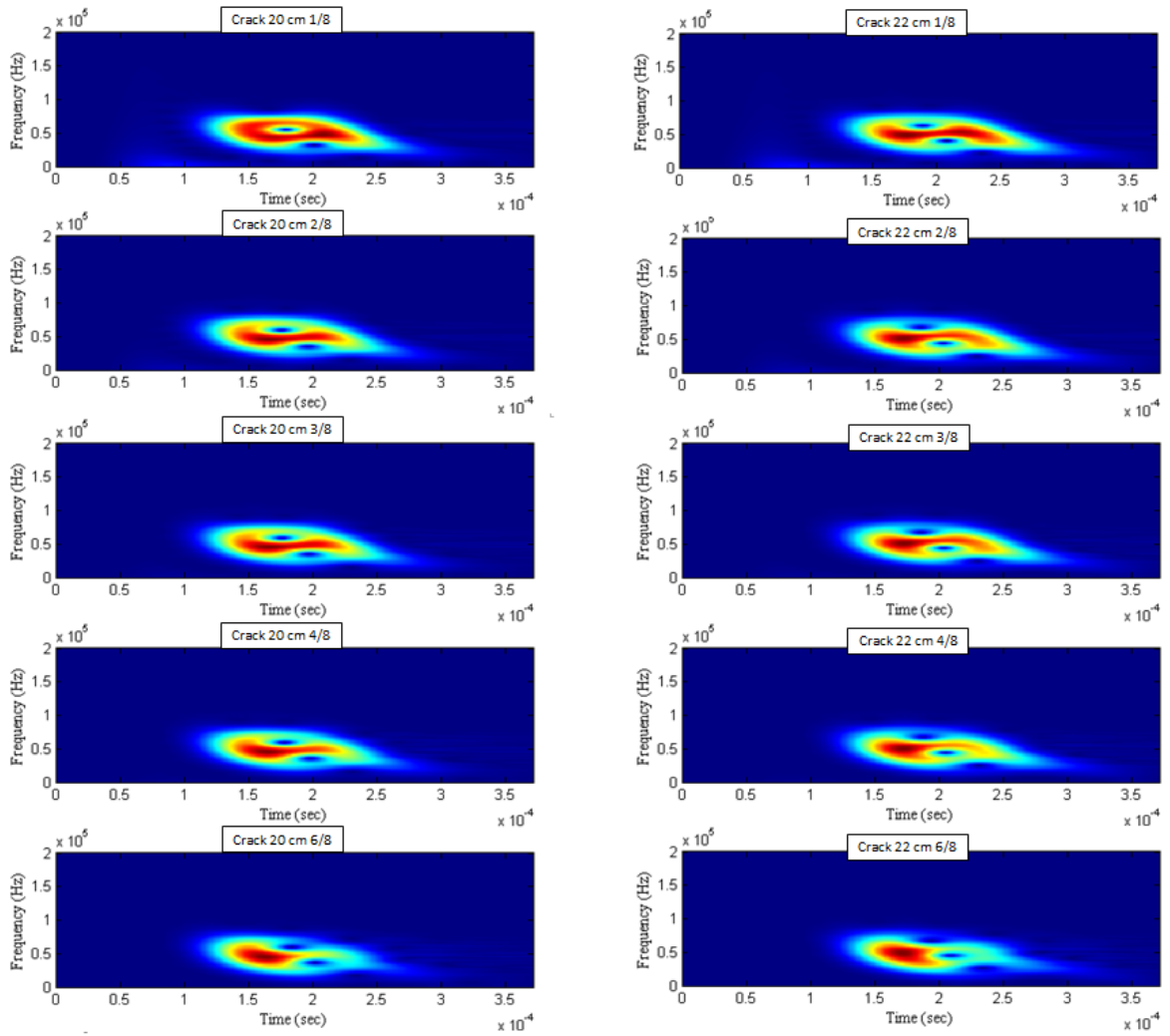


6. Crack signal spectrograms of $d=15$ -cm and $d=16$ -cm

7. Crack signal spectrograms of d=18-cm and d=19-cm

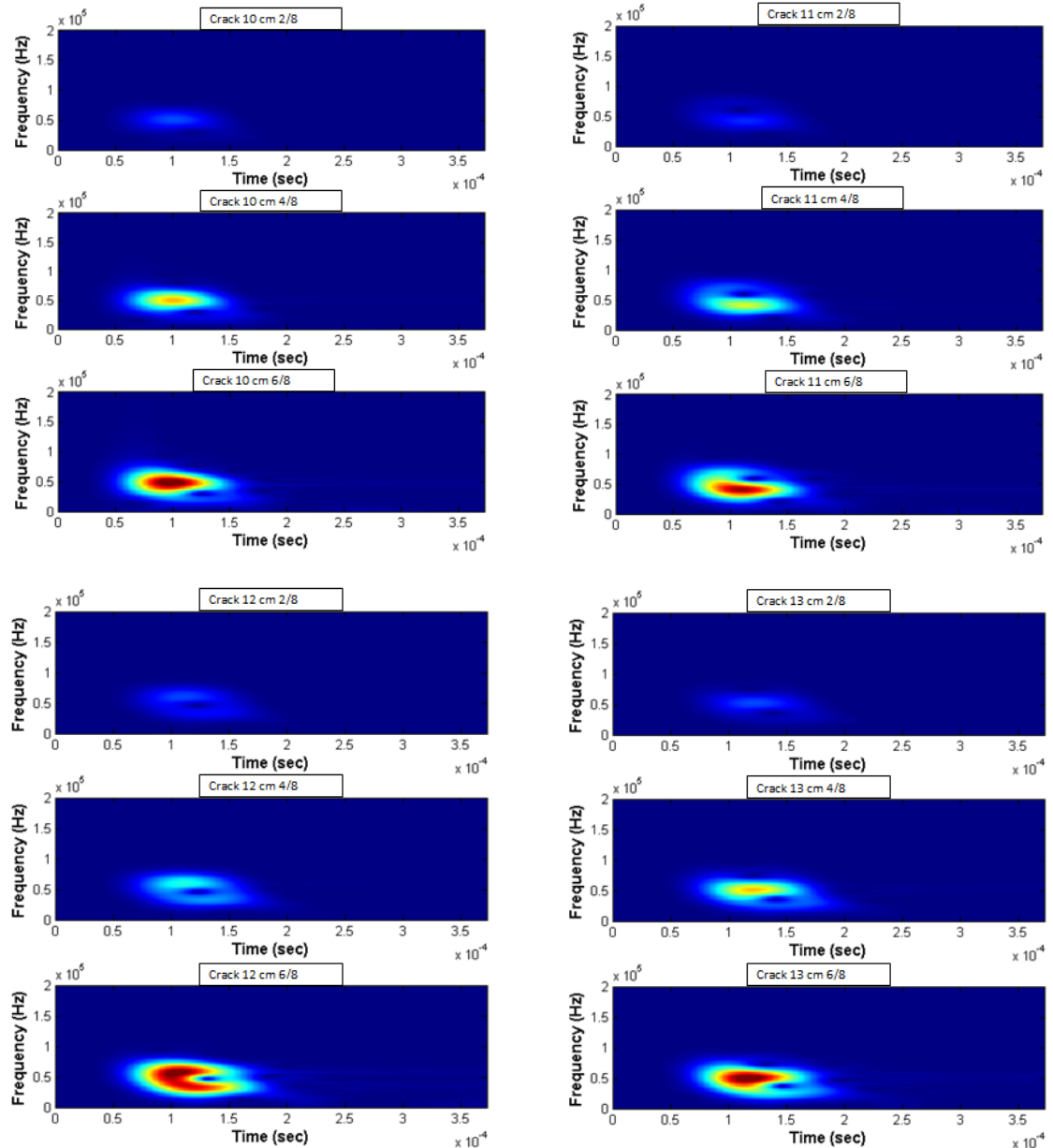


8. Crack signal spectrograms of d=20-cm and d=22-cm

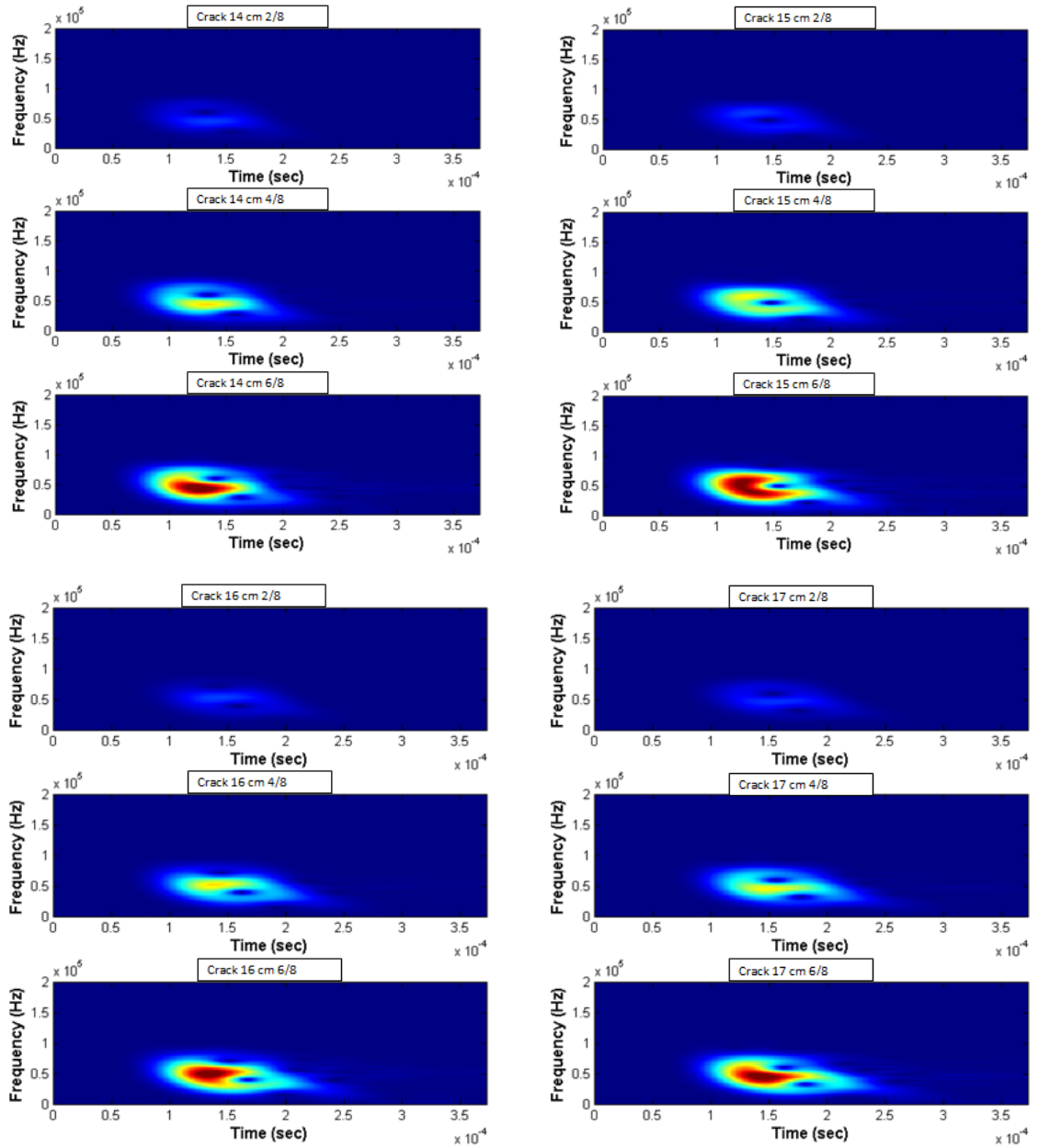


APPENDIX B. Spectrograms of Training and Testing Samples for Multiple-class Classification

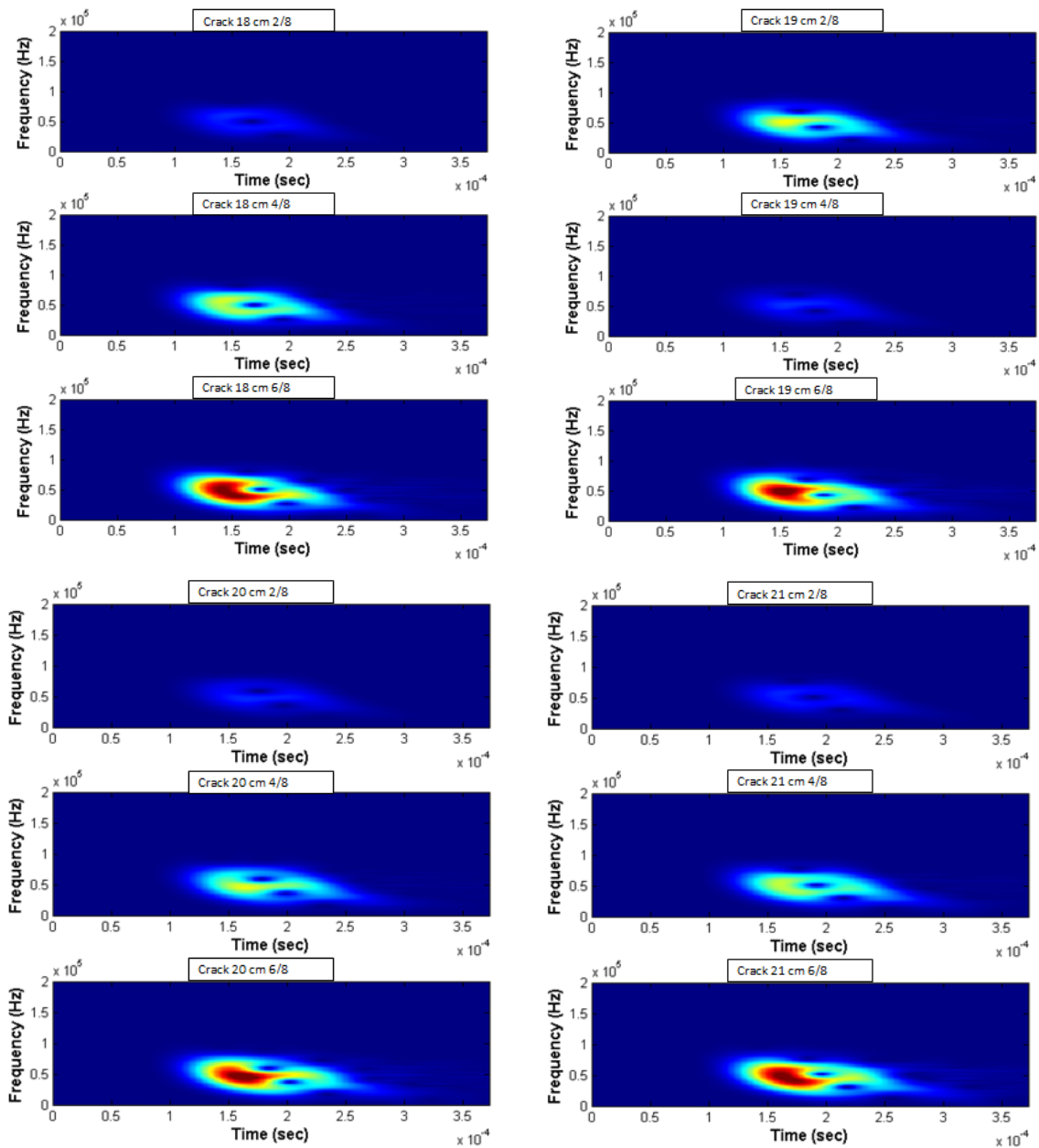
1. Crack signal spectrograms of d=10-cm, 11-cm, 12-cm, and 13-cm



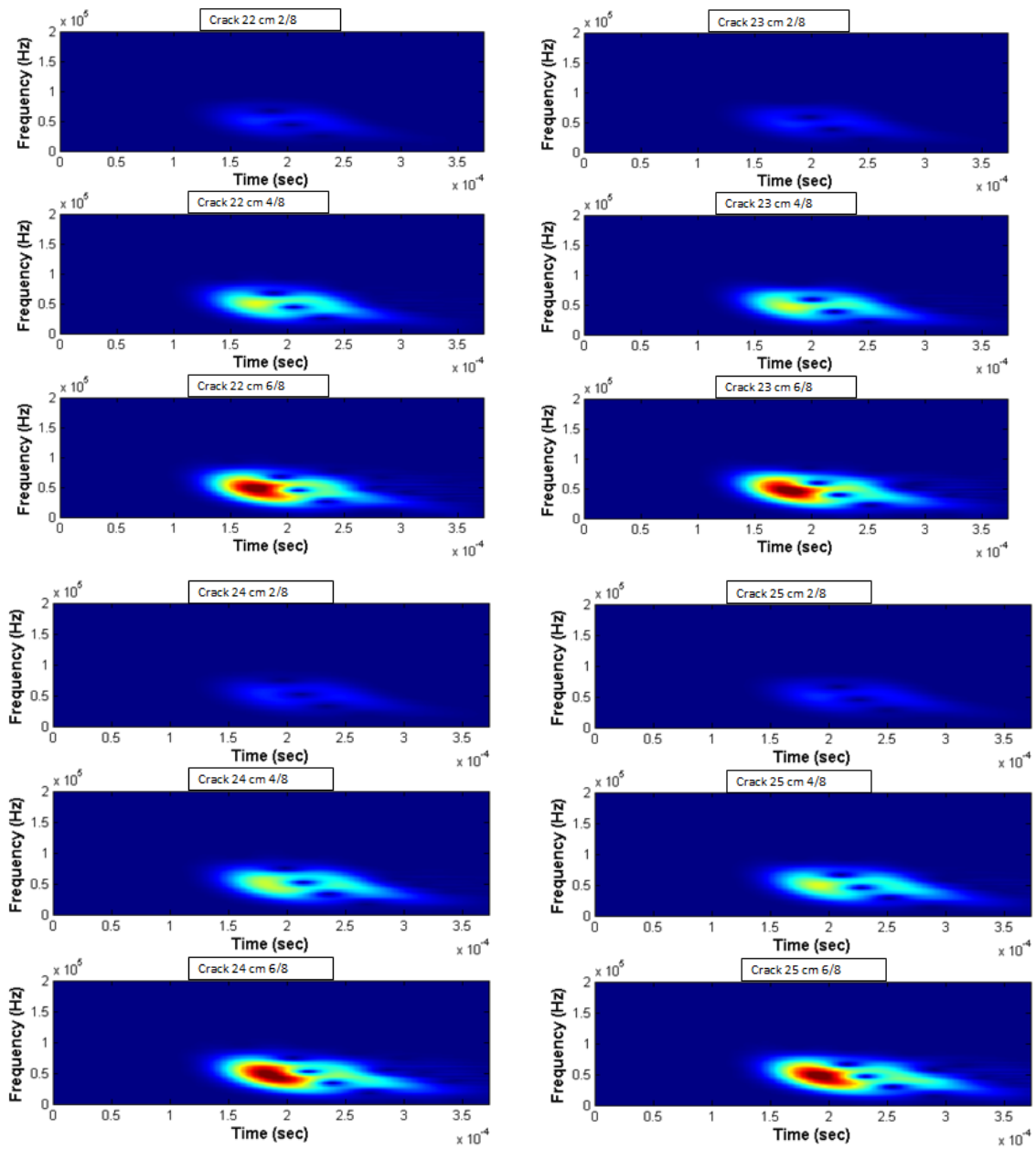
2. Crack signal spectrograms of d=14-cm, 15-cm, 16-cm, and 17-cm



3. Crack signal spectrograms of d=18-cm, 19-cm, 20-cm, and 21-cm



4. Crack signal spectrograms of d=22-cm, 23-cm, 24-cm, and 25-cm



5. Crack signal spectrograms of d=26-cm and 27-cm

