Machine learning based health monitoring of composite structures

Zeyu Liu

To cite this version:

Zeyu Liu. Machine learning based health monitoring of composite structures. Other. Université de Lyon, 2021. English. NNT : 2021LYSEC001 . tel-03462633

HAL Id: tel-03462633
https://theses.hal.science/tel-03462633

Submitted on 2 Dec 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Numéro d'ordre NNT: 2021LYSEC01 Année: 2021

THESE de DOCTORAT DE L'UNIVERSITÉ DE LYON
OPÉRÉE AU SEIN DE L'ÉCOLE CENTRALE DE LYON

ÉCOLE DOCTORALE MEGA
Mécanique, Enegétique, Génie civil et Acoustique

Spécialité: Mécanique

Soutenue le 21/ 01/ 2021 par

Zeyu LIU

Machine Learning Based Health Monitoring of Composite Structures

---

Devant le jury composé de:

Walid Larbi	Professeur, LMSSC, CNAM	Président/Rapporteur
Sergio De Rosa	Professeur, University of Naples Federico II	Rapporteur
Lin LI	Professeur, Beihang University, Chine	Examinatrice
Mohamed Ichchou	Professeur, LTDS, ECL	Directeur
Mohsen Ardabilian	Maître de Conférence, LIRIS, ECL	Co-directeur
Abdelmalek Zine	Maître de Conférence, ICJ, ECL	Co-directeur

Acknowlegements

I would like to express my earnest gratitude to my advisors Professor Mohamed ICHCHOU, Professor Mohsen ARDABILIAN and Professor Abdelmalek ZINE, for providing me the opportunity to work on a multi-disciplinary project and guiding me with patience, motivation and immense knowledge all along the three years. Thanks for their discussions and exchanges in different subject areas, which gave me a broader perspective. Without their valuable suggestions during each meeting, it is impossible for me to make continuous progress and successfully complete my Ph.D study.

I would also like to thank all the members of the jury, Professeur Lin LI and the reviewers Professor Sergio DE ROSA and Walid LARBI for their insightful comments and precious advices on this dissertaton.

Besides, I appreciate the help from my colleagues Christophe Droz and Nassardin Guenfoud. They helped me a lot in the modelling of the studied structure in software at the beginning of my research.

Also, I will express my thankfulness to all my friends who supported me and filled these years with happiness and fun.

Most of all, I would like to thank my family for their unconditional support and encouragement. I am grateful to my parents and my sister for their concern. My profound thankfulness is for my wife and best friend, Guang ZHU. Thank you for being by my side, understanding and supporting me during the three years. Your support is priceless.

At last, I would like to acknowledge China Scholarship Council (CSC) for the financial support of this thesis.

LIU Zeyu
Ecully, France

Abstract

Health monitoring of structures is a key issue for their integrity and reliability. The problem is exacerbated with the use of composites in many cases highly anisotropic heterogeneous structures that meet often conflicting design constraints. Efficient and reliable Structural Health Monitoring (SHM) systems are required to provide a probabilistic interpretation of diagnostics. This work is a special issue of the health monitoring of composite materials. The goal of the study is to obtain a data-driven probabilistic interpretation of damage detection. More precisely, the idea is to develop an automated methodology evolution of defects based on machine learning approaches.

The investigation is carried out based on healthy and damaged status of a composite sandwich panel with a honeycomb core modeled in ANSYS. Instantaneous signals with different frequencies are applied to the structure and finite element analysis is performed to obtain vibration responses in both statuses. Two machine learning methods are adopted for crack damage detection successively: a conventional machine learning method Gaussian Process (GP), and a deep learning algorithm Convolutional Neural Network (CNN). Features extracted by Discrete Wavelet Transform (DWT) from structural vibration responses are used to train and test the GP model to assess the health status of the structure.

Then another feature extraction method Continuous Wavelet Transform (CWT) is used to convert the time domain vibration response signal into a two-dimensional image, which is then fed into the proposed CNN model to predict the occurrence and location of potential crack damages. In addition, the applicability of the CNN method to different structures and different types of data is studied, i.e., using modal data in a beam.

The proposed GP-based method is effective for crack-type damage detection in the studied composite sandwich structure while CNN-based method is capable of detecting crack damage occurrence and its potential location. Besides, it is proposed that the appropriate data types can be selected according to different situations to obtain an optimal performance using the CNN-based method. It is expected that the proposed methods are suitable for damage detection of more complex structures.

Keywords: Structural Health Monitoring, Machine Learning, Composite sandwich structure, damage detection and localization

Résumé

La surveillance de l'état des structures est un enjeu clé pour leur intégrité et leur fiabilité. Le problème est exacerbé par l'utilisation de composites dans de nombreux cas où des structures hétérogènes sont hautement anisotropes. Des systèmes efficaces et fiables de surveillance de l'état de la structure (SHM) sont nécessaires pour fournir une interprétation probabiliste des diagnostics. Ce travail est un sujet spécial de la surveillance et le suivi de l'état des matériaux composites, dont le but est d'obtenir une interprétation probabiliste de la détection des dommages basée sur les données. Plus précisément, l'idée est de développer une méthodologie automatisée d'évolution des défauts basée sur des approches d'apprentissage automatique.

La recherche est réalisée sur la base de l'état sain et endommagé d'un panneau sandwich composite à âme en nid d'abeille modélisé dans ANSYS. Des signaux instantanés de différentes fréquences sont appliqués à la structure et une analyse par éléments finis est effectuée pour obtenir des réponses aux vibrations dans les deux états. Deux méthodes d'apprentissage automatique sont adoptées successivement pour la détection des dommages aux fissures: une méthode d'apprentissage automatique classique Processus Gaussien (GP) et un algorithme d'apprentissage profond le Réseau de Neurones Convolutifs (CNN). Les caractéristiques extraites par la transformation en ondelettes discrète (DWT) des réponses aux vibrations structurelles sont utilisées pour entraîner et tester le modèle GP afin d'évaluer l'état de santé de la structure.

Ensuite, une autre méthode d'extraction de caractéristiques la transformation en ondelettes continue (CWT) est utilisée pour convertir le signal de réponse aux vibrations dans le domaine temporel en une image bidimensionnelle, qui est ensuite introduite dans le modèle CNN proposé pour prédire l'apparition et l'emplacement des dommages potentiels de type fissure. De plus, l'applicabilité de la méthode CNN à différentes structures et différents types de données est étudiée, c'est-à-dire en utilisant des données modales dans une poutre.

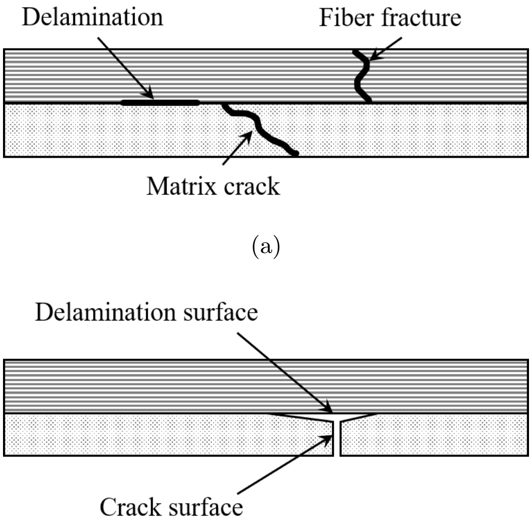
La méthode basée sur la GP est efficace pour la détection des dommages de type fissure dans la structure sandwich composite étudiée, tandis que la méthode basée sur CNN est capable de détecter l'apparition de dommages de fissure et son emplacement potentiel. En outre, il est proposé que les types de données appropriés puissent être sélectionnés en fonction de différentes situations pour obtenir une performance optimale en utilisant la méthode basée sur CNN. On s'attend à ce que les méthodes proposées conviennent pour la détection des dommages de structures plus complexes.

Mots-clés: Surveillance de santé structurale, apprentissage automatique, Structure sandwich composite, détection et localisation des dommages

Contents

Introduction

Advanced composite materials have been widely used in different areas, including aerospace, medicine, machinery, construction and other industries due to the advantageous characteristics of light weight, good ductility, corrosion resistance, heat insulation, sound insulation, shock absorption and high (low) temperature resistance, etc. However, due to the synthesis method of composite materials, they are susceptible to several structural damages, such as fiber fracture, matrix crack and delamination, as well as many other kinds of damages, e.g., blistering, crushing, etc. [1]. These types of damage sometimes appear in composite materials at the same time as shown in figure 1(a), and sometimes appear in sequence as the damage evolves as shown in figure 1(b). In symmetric composite laminates, transverse crack may induce delaminations [2] and in graphite/epoxy (Gr/Ep), fiber breaks may induce debonding, followed by matrix cracking [3]. These damages are usually caused by fatigue and impact events. In the early stage of the damages, they are very small and barely visible to visual inspections, but they could further lead to disasters, especially for aircrafts, causing huge loss of people lives and money. As a result, the development of *Structural Health Monitoring* (**SHM**) systems has been considerable concerns in the last 2 decades.



(a)

(b)

Figure 1: Illustration of composite material damages, (a) matrix crack, fiber fracture and delamination, (b) delamination induced by transverse crack

The definition of Structural Health Monitoring is given by D. Balageas et al.[4]: SHM aims to give, at every moment during the life of a structure, a diagnosis of the "state" of the constituent materials, of different parts, and of the full assembly of these parts constituting the structure as a whole. The motivation to develop SHM is that an appropriate SHM system allows to reduce the maintenance cost and increase in-service time by replacing the periodic inspection with condition-based maintenance and performing targeted maintenance on damaged parts, to improve the safety through an alarm system and increase the longevity of the structure by improving the construction, especially to avoid significant losses due to damage that can be predicted in advance before failures [5]. A typical structural failure example is shown in figure 2, NASA's solar-powered Helios wing which is made mostly of composite materials broke up because of deformation beyond limit. With appropriate SHM system, abnormalities in the structure can be detected and failure might be prevented.


(a) (b)

Figure 2: (a) NASA's Helios wing and (b) its wreckage in the Pacific [6]

SHM has great potential in damage diagnosis and avoidance of failures in various fields and in turn reducing maintenance cost as well as increasing structural reliability, especially with the increasing development in industries such as civil engineering, transportation, manufacturing, etc.

The present thesis focused on machine learning based structural health monitoring of a composite sandwich structure that is modeled based on finite element method. The thesis mainly includes six chapters, which are organized as follows:

The research background of composite structures, structural health monitoring methods, the development of machine learning and its applications are firstly reviewed in Chapter 1. The survey starts with the wide application of composite structures and structure damage problems. Then a review of existing structural health monitoring methods, including physical-based and data driven-based methods is given. The introduction of machine learning and its applications in structural health monitoring as well as other fields is then discussed.

Chapter 2 focuses on the implement of a numerical approach to construct a database for SHM of composite sandwich structure. Models are constructed and validated by comparing the physical properties obtained numerically with analytical methods. Different damage cases are modeled followed by finite element analysis. Some commonly used feature extraction techniques through FE simulation are then introduced, which help to construct a database for further study.

Chapter 3 develops a conventional machine learning algorithm: Gaussian Process-based method for structural health monitoring of the studied composite sandwich structure. Discrete wavelet transform (DWT) is used for feature extraction, and GP is used to detect the occurrence of potential crack damages in the structure. Several factors that have impact on classification accuracy have been investigated. The pertinence of data from different sampling points on the structure is also investigated. In addition, the effectiveness of the proposed method is verified under simulated environmental noise.

A Convolutional Neural Network (CNN)-based structural health monitoring system is proposed in Chapter 4 due to the advantages of the deep learning algorithm in image recognition and classification. One-stage CNN model and two-stage CNN model using images processed by Continuous wavelet transform (CWT) are proposed successively. In the one-stage CNN model, most samples with ground truth "crack" are incorrectly classified as "healthy", leading to a poor capability of detecting and localizing crack damage at the same time. However, a two-stage CNN model is then proposed to address the problem, in which the first CNN is used to detect the occurrence of crack damage while the second is used for crack localization. The enrichment of the database is carried out. Accordingly, the lack of variability of the database is solved and the capability of the two-stage CNN model of detecting and localizing crack damage is improved significantly.

Chapter 5 extends the two-stage CNN-based SHM method using modal data at low frequencies in a simply-supported beam to verify the wide applicability of the CNN-based method to different structures and different types of data. Crack localization and damage size prediction in a simply-supported beam is conducted based on mode shapes and stress distribution in the beam instead of the vibration data used in the sandwich structure in Chapter 3 and Chapter 4. Results suggest that stress distribution may contain more discriminative information relative to crack location and size than mode shapes. In addition, it is necessary to find a suitable compromise between the classification accuracy and the number of sampling points according to different situations to obtain an optimal performance.

Finally, conclusions and perspectives are drawn after chapter 5.

Chapter 1

Literature survey on SHM of composite structures and machine learning

---

Contents

1.1 Composite structures

Composite materials are formed by combining materials together to form an overall structure with properties that differ from that of the individual components. They have the characteristics of light weight, good ductility, corrosion resistance, heat insulation, sound insulation, shock absorption and high (low) temperature resistance, etc. These features typically meet the material requirements in specific working environments [7]. Therefore, composite materials are increasingly applied in industries rather than conventional metal alloys. Advanced composite materials are more and more adopted as aircraft materials for economic reasons. The primary reason why they are applied in aircrafts is the weight reduction. In civil aviation, aircraft weight reduction means less fuel is consumed and more cargo can be loaded, resulting in considerable profits and more environmentally friendly transport form. In military aviation, aircraft weight reduction means that extra loads can be carried and more complex maneuvers can be performed. For example, the use of composite materials in the Boeing 787 is about 50%, and the weight percentage of the composites of the aero-structure went from 5% for the A300 to more than 50% for the latest model A350 XWB, gradually replacing other materials as illustrated in figure 1.1 and 1.2. The structural mass of the A350 is 15 tonnes lighter than it would have been without composite - in total the resulting weight gain is 15 tonnes. It is estimated that one ton of saved mass translates into a saving of 6,000 tons of kerosene over the lifetime of an aircraft [8]. Considering that under proper maintenance an aircraft can be used for more than 20 years. Even if the aircraft weight reduction is 1%, the profit can be considerable.

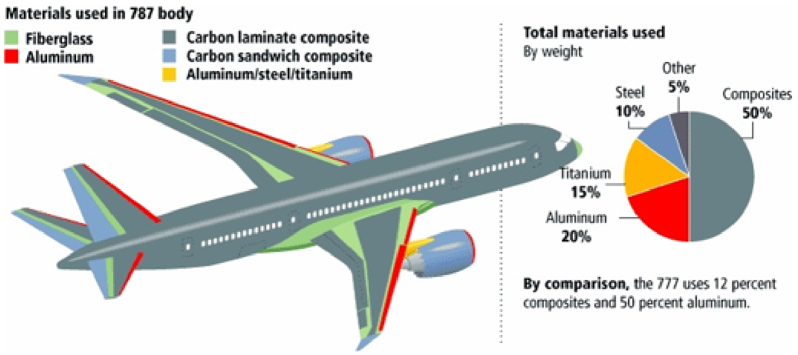


Figure 1.1: Application of composite materials in Boeing aircraft. The 787 uses 50% composites and 20% aluminium while the 777 uses only 12% composites and 50% aluminium.

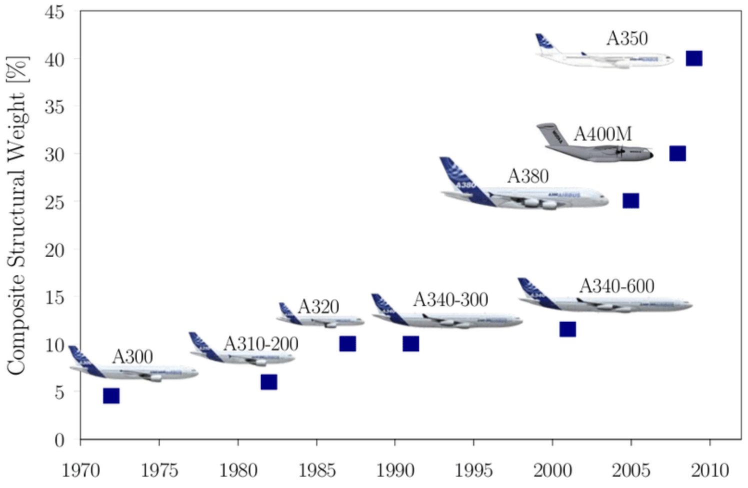


Figure 1.2: Evolution of the overall composite structural weight in Airbus air-crafts. Composite weight went from 5% for A300 to more than 50% for A350 XWB.

Composite sandwich structures have attracted increasing attention due to the high performance in industries. They are usually fabricated by attaching two thin but stiff skins to a lightweight but thick core. The two skins, also called face sheets, are identical in material and geometry properties. In composite sandwich structures, the face sheets are fabricated by laminated plies with specified stack sequence for specific purposes, which can resist the in-plane and bending loads. The core of foam-shaped or regular shapes as shown in figure 1.3, can resist transverse shear load and has different properties depending on the configuration.

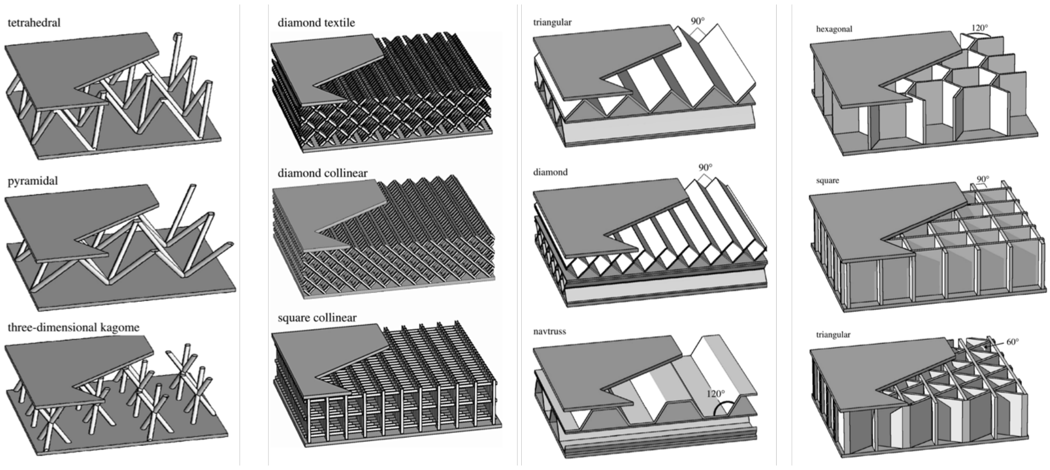


Figure 1.3: Sandwich structures with core of different shapes[9]

Sandwich structures are widely adopted as architectural materials due to its high flexural stiffness-to-weight ratio compared to monocoque structures. These properties make the sandwich structure material show significant advantages over other materials in the same working environment. The applications of sandwich structures can date back to the World War II, when such structures were adopted in the construction of aircrafts. A ply-balsa-ply sandwich structure made of wood was designed for the fuselage of the famous de Havilland DH.98 Mosquito multi-role combat aircraft, see figure 1.4. Later, sandwich structures with better performance was proposed and developed, i.e. carbon fiber reinforced polymer/plastic (CFRP). Nowadays advanced composite sandwich structures are more and more adopted in various civil domains, such as aircrafts, trains, ect. An example of the application of composite sandwich structures in transportation is illustrated in figure 1.5. The advanced aerodynamic nose of high speed train ETR 675 is made of composite sandwich structures aiming at reducing weight while keeping the stiffness of the material.

With the increasing application of composite materials in structures, a deeper understanding of the properties of composite materials is needed for different structural design. Numerous studies have been conducted to investigate the static and dynamic properties of composite structures.

Figure 1.4: Ply-balsa-ply sandwich structure designed for the fuselage of de Havilland DH.98 Mosquito multi-role combat aircraft

Figure 1.5: Advanced aerodynamic nose of high speed train ETR 675

To study the effects of core thickness and face sheet thickness of a cellular foam core sandwich structure on its dynamic behaviour, static structural and modal analysis was carried out through finite element simulation software [10]. It is found that the overall stiffness of the sandwich structure is directly impacted by core thickness and skin thickness. In the case where the skin mass is dominant in a sandwich structure, the core thickness and foam structure will be crucial to the overall stiffness of the sandwich structure by keeping the skin thickness at a minimum level. Another research focusing on the finite element evaluation of composite sandwich panels under static bending load was conducted by M.M.Venugopal et al. [11]. Finite element analysis results are compared with experimental data on sandwich panel bending properties and the maximum percentage error is 11.27%, which suggests that the modeling approach is reliable in structural design of CFC sandwich panels.

For sandwich structures in which the face sheets can resist the in-plane and bending loads and the core can resist transverse shear loads, the vibroacoustic properties of both parts should be studied in order to understand its behaviours in some circumstances, such as shear stresses in honeycomb sandwich panels by analytical, numerical and experimental approaches [12], the shear core effects on sound transmission loss through honeycomb sandwich panels [13], flexural properties [14], bending-shear core transition [15] as well as structural vibroacoustic optimization of sandwich panels [16].

Due to the complex nature of composite structures, one major interest in the academic community is to study the wave propagation characteristics, which may provide insight into the structural dynamic properties. Various works have been done to investigate wave propagation in composite sandwich plates.

Numerical simulation was conducted in [17] to investigate the mechanism of guided wave propagation in honeycomb sandwich structures. Global guided waves in the structure at low frequencies as well as guided wave leak in the face sheet at high frequencies are observed. The cell geometry is demonstrated to have impact on the wave propagation. Finally experimental data are use to validate simulations and results shown that the numerical simulations are reliable to represent wave propagation in the composite sandwich structure. Wave propagation in sandwich composite plates is also investigated by V. N. Smelyanskiy et al. using analytical approach [18]. It is found that at low frequencies, the sandwich's motion is mainly governed by the rigid face sheet but with the increase of frequencies, the motion of the sandwich plate tends to be governed by the shear of the core, which is consistent with the analytical results in [19]. Besides, analytical results are in agreement with the results of numerical simulations and experiments. However, as the propagation time increases, reflections from boundaries in simulation will lead to phase changes of strain oscillations, and it does not always match analytical and experimental signals well. Nevertheless, the obtained results open up prospects for obtaining wave propagation in composite structures through modeling methods.

Numerical simulations of Lamb wave propagation in honeycomb sandwich panels and metallic foam sandwich structures are also conducted in [20] and [21] with similar approaches. It is demonstrated that for both foam sandwich panels and honeycomb sandwich panels, wave propagation is influenced by the geometrical properties and material properties of the plates, which provides valuable information in structural design and SHM systems. Researchers in [22] focused on radiation and attenuation of Lamb waves in fiber-reinforced composites. It is demonstrated by numerical simulations and experiments that wave attenuation, wave velocities and acoustic energy for different modes of propagation are strongly impacted by the material anisotropic characteristics. A thorough review on damping in composite materials is referred to [23].

1.2 Structural health monitoring methods

In recent years, researchers from various fields have been concentrating on developing structural health monitoring systems for composite structures with increasing utilisation of these materials, to guarantee structure safeties. The definition of structural health monitoring is given by D. Balageas et al. [4]: SHM aims to give, at every moment during the life of a structure, a diagnosis of the "state" of the constituent materials, of different parts, and of the full assembly of these parts constituting the structure as a whole.

The motivation to develop SHM is that an appropriate SHM system allows to reduce the maintenance cost and increase in-service time by replacing the periodic inspection with condition-based maintenance and performing targeted maintenance on damaged parts, to improve the safety through an alarm system and increase the longevity of the structure by improving the construction, especially to avoid significant losses due to damage that can be predicted in advance before failures [5]. In the sectors that have high safety requirements, especially in the domains closely related to human safeties, such as buildings, cars, trains, airplanes and even spaceships, efficient and reliable SHM technology is demanded to provide a probabilistic interpretation of diagnostics so as to prevent loss of human lives and properties.

Current and emerging structural damage detection methods are summarized by Avci et al. [24], as shown in figure 1.6. Damage detection methods in SHM were categorized into local methods and global methods, where the former focuses on detecting structural damage in a relatively smaller scale through visual testing, ultrasonic testing, infrared thermography, radiographic testing etc., while the latter focuses on time, frequency and modal information to detect, localize and assess severity of damage based on vibration response of structures by traditional methods or machine learning methods.

Although the benefits of appropriate SHM is evident, it appears that a real-time monitoring system for a on-line structure is not yet available. The existing non-destructive testing (NDT) methods such as X-ray inspection, ultrasonic testing and thermography are time-consuming, labor-intensive and highly costly, and they are applied during the short inspection intervals [25]. An appropriate SHM system should be able to collect data during operating time of structures and provide real-time information about the structure. Four levels of assessment of SHM are proposed

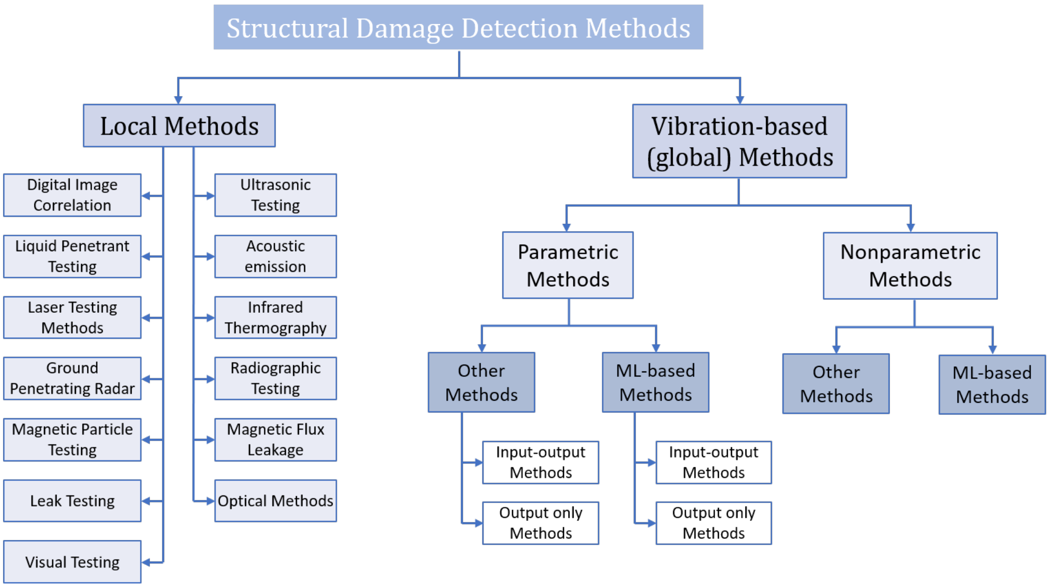


Figure 1.6: Structural damage detection methods [24]

as below:

- Detect if a damage exists
- Localize the damage area
- Identify the type of the damage
- Quantify the severity of the damage

The first level provides the information that a damage appears on the structure, while the second level tells in which region on the structure the damage occurs. Level three is dedicated to the identification of the damage, e.g., crack, delamination, debonding. In the fourth level, more information about the structure is provided. Several parameters are needed to describe the size of the damage, the physical properties of the structure such as stiffness or mass. Sometimes the assessment of the remaining life time of the structure is included in level four, which usually involves fatigue analysis and damage evolution analysis. At each level there are still many challenges. For example at the first level, how to increase the SHM system sensitivities to detect small damages is still a tricky issue.

SHM refers to the process of implementing a damage detection and characterization strategy for engineering structures. The basic idea behind the SHM technology is that, modal parameters are functions of the physical properties of the structure, such as natural frequencies, mode shapes and modal damping. Therefore, changes to the material or structure will adversely cause detectable changes in modal properties and affect the structure performance.

Thanks to various types of actuators and sensors, one is able to collect information relative to the state of structures. Due to the sensors, physical properties of a structure become detectable. A review of recent advances and applications of sensors for SHM is referred to [26, 27]. SHM can be divided into passive monitoring and active monitoring according to the initiative. For passive monitoring, an examiner focuses on the evolution detected from the sensors. On the contrary, perturbations are generated through an actuator to the structure, then structural responses will be collected by sensors. In this case, the SHM is called 'active monitoring'. In an active monitoring of aerospace composite structures, an investigation on the interaction between embedded sensors and composite structures is conducted since there will be stress concentrations in all composites that contain embedded sensors and transducers [28]. A design of reducing the effect of the embedded sensor on the strength of composite structures is proposed.

In fact, SHM appeared in the 1970s and has been studied and developed for several decades. It is a multidisciplinary research field, involving material science, mechanics, structural vibrations, signal processing, computer science, sensor technology, etc.

Dating back to the 19th century, railway workers used a qualitative way to evaluate if damage was present by striking the train wheel with hammers and listening to the sound, which can be considered as the initial form of SHM. Later in the 1970s and 1980s when the oil industry was well developed, workers and scientists in this field have made considerable efforts to create SHM systems for health monitoring of offshore platforms. However, measurement difficulties caused by machine noise, mass change due to marine tides, instability of the platform structures and fluid wave impact prevented the application of this technology on offshore platforms, resulting in large abandonment in the 1980s.

In aerospace industry, vibration-based damage identification approaches became a research field in the 1970s with the development of man-made aircrafts launched to the space. Fatigue damages are identified in components of space shuttles using this technique. Later the application of space station promoted the development of SHM due to space debris impacts. Both experimental and analytical approached were used to measure damaged and undamaged structures. Differences in modal properties were emphasized to identify damages.

SHM was also applied in civil engineering in the 1990s. During the 1994 Northridge earthquake in America, cracks were found in many buildings as shown in figure 1.7, and it was able to detect damages only by visual inspection, which was also by qualitative manner. Since then, it was proposed to develop methods that enable an immediate reliable and efficient evaluation of a structural health state after such an event or accident instead of relying on visual inspection which is time and labor-consuming.

(a)

(b)

Figure 1.7: Damage caused by the 1994 Northridge earthquake

With the evolution and development of sensor technology and signal processing techniques, various types of sensors are employed in SHM systems. One of the most successful application of SHM is on the rotating machinery based on pattern recognition of structural vibration features, such as displacement, velocity and acceleration (e.g., SHM on rotating machinery in figure 1.8). However, in the early age SHM in this manner was also based on visual inspection of signals in time or frequency domains, by comparing the current signal features with features extracted during damage-free operating conditions. This was still qualitative approach. More recently, with the rapid development of computer science, pattern recognition-based SHM has become a popular research field. With the significant improvement of computer capabilities dealing with information processing, the efficiency and accuracy of SHM have been greatly improved.

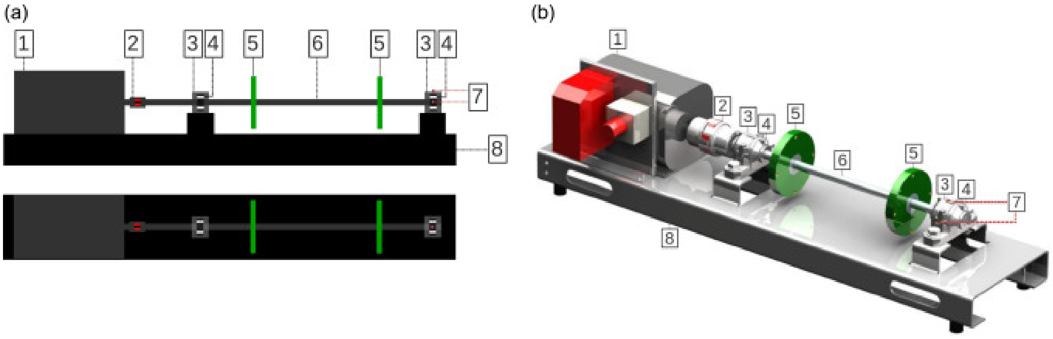
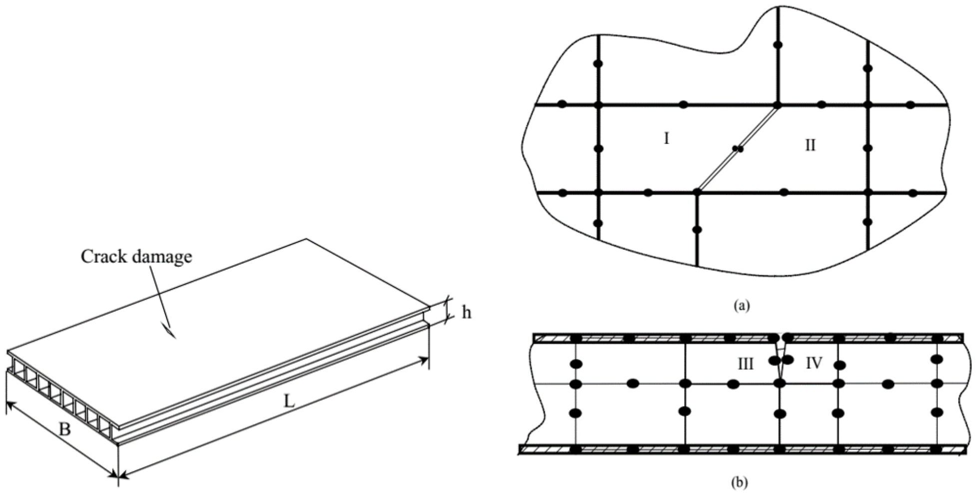


Figure 1.8: Rotating machinery used in [29]. 1. servo-motor; 2. coupling; 3. bearing housing; 4. bearing; 5. disk; 6. shaft; 7. accelerometer 8. metal plate

In order to improve the efficiency and reliability of SHM, it is generally necessary to have a prior knowledge about the damages. However, damage cases are not always available in interested structures and sometimes are difficult and expensive to create different types and severity of damages in real structures. Therefore, the modeling of damages has become a promising way to study various commonly seen damages.

A general way to model a damage is using finite element method-based software. For a damage whose nature is local discontinuity, which is a frequently encountered situation, the modeling is often achieved by disconnecting local nodes, for example the crack damage modeling in composite structures in [30] as shown in figure 1.9. Similarly, crack in composite material shell structures is modeled by disconnecting local nodes in [31], barely visible low-speed impact damage (BVLID) in 3D-core sandwich composite structure is modeled by demerging element nodes at the interphase of skin and core [32].

Delamination damages in laminated composites can also be modeled by disconnecting local nodes so that the delamination region is made up of two separate component segments above and below the delamination surface [33, 34]. Another damage modeling approach is to modify local physical parameters. As it is demonstrated that delamination in a structure will lead to local loss of stiffness, researchers attempt to reduce local stiffness so as to model delamination damages [35]. In Ref.[36], skin and core defects in a composite sandwich panel are modeled by local reduction of their isotropic Young's and Shear coefficients and delamination between the face sheet and core is modeled by a disconnection between the skin and core layers, as shown in figure 1.10.



(a) (b)

Figure 1.9: Crack modeling in composite structures by disconnecting local nodes [30]

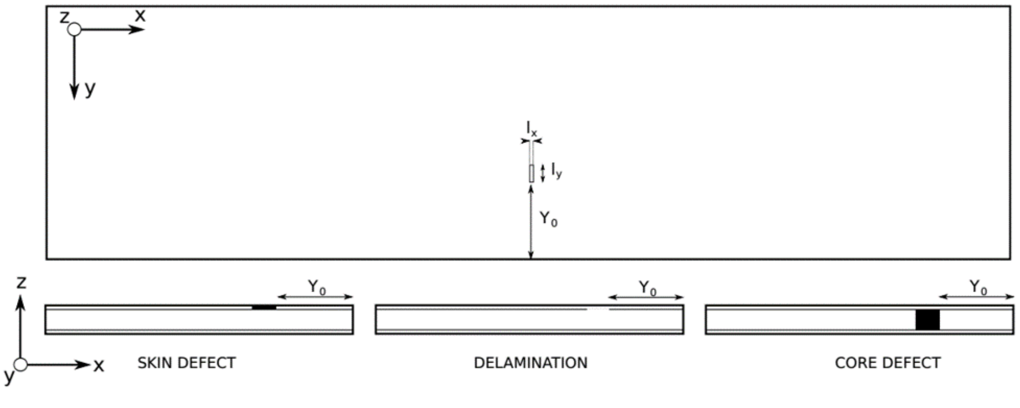


Figure 1.10: Skin and core defects and delamination modeling in a composite sandwich panel [36]

Two of NDT methods, vibration-based technique and guided wave-based technique have been developed for the extraction of damage-sensitive information about the health state of structures. They are the most commonly used among others. A general process of the SHM involves collecting relevant data, which is the structure response, from an array of sensors attached on the structure. Then necessary signal processing is carried out for the purpose of data reduction and key feature extraction from these measurements. Finally, the healthy state of the structure is determined by statistical analysis of these features.

Vibration-based damage detection focuses on the detection of the mode shape singularity and natural frequency changes created by local discontinuity caused by damages. Several researches have successfully implemented this method in SHM of composite materials.

In [37], vibration-based NDT is adopted to detect debonding in honeycomb sandwich beams based on the natural frequency changes caused by damages. Honeycomb sandwich beam is considered equivalent to homogeneous materials in low frequency because the local periodic structure is much smaller than the wavelength. However, the proposed method cannot effectively detect small damages in large structures. Detection of structural changes based on an artificial immune system and a statistical data-driven modeling on an aircraft skin panel is conducted in [38]. Structural vibration responses are fused and feature extraction together with pattern recognition are performed to detect damage and structural changes. The methodology is able to accurately detect damage but unable to identify several simultaneous damage patterns. Another vibration-based abnormality detection method is employed in [39] for health monitoring of in-service offshore wind turbines. Operational modal analysis (OMA) is used to monitor the resonance frequencies of wind turbines due to its capability of monitoring the boundary conditions of civil structures despite its non-sensitivity to small cracks or very local phenomena. The contribution shows the possibility of monitoring wind turbines.

The effect of environmental condition variations on structural dynamic characteristics are taken into account in [40] for SHM of railway bridges based on vibration approaches. Environmental conditions are used as input together with other loads applied to the structure to a machine learning algorithm so that the influence of environmental conditions is not considered in the prediction and the result indicating damages is only due to damages. The proposed method is proven capable to detect and localize damage. In addition, neglecting the environmental factor, a noticeable decrease of the prediction power is met.

It is worth mentioning that in vibration-based NDT, wavelet analysis has been applied in many studies for the post-processing of vibrational mode shapes to extract features for damage detection.

In [41], vibration-based NDT with wavelet analysis is applied to a composite sandwich plate to detect different types of damages by extracting modal shapes of vibration. Sandwich plate with damages is scanned by 2 laser Doppler vibrometers (LDV) by experiment and the amplitudes of wavelet analysis coefficients are used to represent the presence and location of damages. Results show that the proposed method is capable to detect and localize different damages using wavelet analysis. But the proposed method should not be limited to the laboratory scale test.

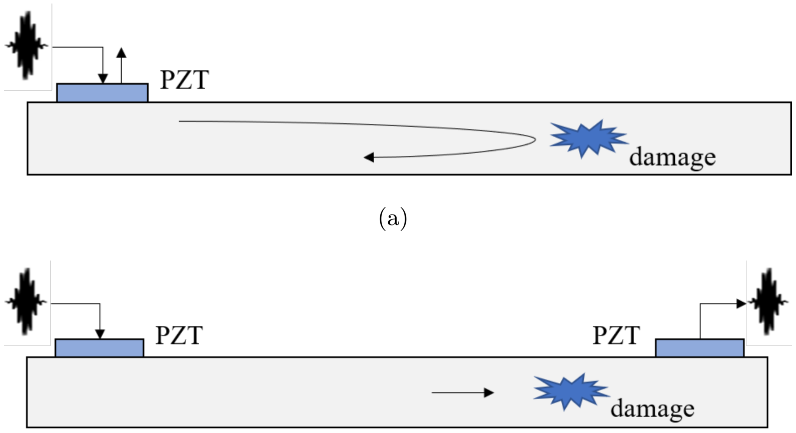
A novel method for identification of multiple damage by combining shearographic NDT and 2D undecimated wavelet transform based on modal data was proposed by Katunin et al. [42]. The proposed method with wavelet analysis shows high sensitivity compared to the analysis of raw shearographic results. Similar result was observed in another research conducted by Zhou et al. [43] based on continuous wavelet transform, which shows that the sensitivity for damage detection is increased by wavelet analysis.

Wavelet transform (WT) is used in [30] for signal processing of the vibration response of a composite sandwich panel. Numerical simulations are firstly performed to obtain a database of the sandwich panel. The database is in turn used to train an ANN to establish the mapping relationship between structural damage features and damage states. Experimental study is then conducted where vibration responses of sandwich plates with damages of different severity are extracted with WT and are fed into the trained ANN. Crack damage location and severity can be detected. Results show that the method developed can be applied to online structural health monitoring for various industrial structures.

Vibration-based damage identification using wavelet analysis is employed in [44, 45] for SHM of concrete plate and steel plate. Continuous wavelet transform is used to extract time-frequency information from the response data. "Zeroth-order moment" (ZOM) is proposed as damage index to identify damages. The study shows that WT is a powerful tool in extracting features, and the proposed method used in this study is more sensitive to structural damage than other methods. A review of wavelet analysis in SHM is referred to [46, 47]. A thorough review of vibration-based damage detection is presented in [48, 49, 50].

Guided wave-based damage detection focuses on the detection of elastic wave energy variation and changes in wave propagation pattern due to the interaction between propagated waves and material discontinuity where damage occurs [51, 52, 53].

In [54] ultrasonic feature guided waves (FGW) are used which focuses on the wave propagation energy to detect damages on quasi-isotropic composite laminates. The interaction of the identified FGW mode with different types of defects is studied by both simulation and experiment. Close agreement is observed between the numerical measurement and experimental measurement. It is demonstrated that the proposed FGW method has good potential for efficient damage detection in composite bends. A model-based method for damage detection with guided wave is



(a)

(b)

Figure 1.11: Guided wave-based damage detection methods. (a) pulse-echo, (b) pitch-catch

proposed in [55]. Normally guided wave-based method is conducted by comparing a baseline signal recorded for a damage-free structure with or subtracted from the signal recorded during the inspection. However, the accuracy of this approach is affected by uncontrollable factors such as temperature variation, sensor errors and material property changes due to degradation. In the proposed method, the effect of these factors can be compensated. Experimental and numerical approaches are conducted and demonstrate the feasibility of the proposed method. Nevertheless, the utilisation of 3D measurement system together with transient finite element simulations will significantly increase the cost.

Theoretical and numerical studies are firstly conducted in [51] to identify disbond and high density core region in a honeycomb composite sandwich structure using ultrasonic guided waves. Laboratory experiment is then carried out to validate theoretical and numerical results. Interaction of guided waves with damages is analyzed through the structural response signals. A good agreement is observed between analytical, numerical and experimental results. It is found that the presence of high density core region results in a decrease in amplitude of the propagating guided wave modes and the presence of disbond results in a significant amplification of the primary anti-symmetric mode. Similar method is also adopted in [17, 56, 57]. A thorough review of guided wave-based damage detection methods is presented in [58, 59, 60].

To have a higher accuracy in damage detection, combination of vibration-based and guided wave-based approaches was proposed by Maciej Radzienski et al. [61]. As vibration-based method and guided wave-based method can both accomplished by the same setup: actuators and 3D measurement system, they are combined for a double verification with different NDT techniques. Data are registered simultaneously for both methods. Filtering techniques are employed to obtain damage maps. Results prove that the combined method does not reduce the effectiveness of NDT techniques for detection of debonding in honeycomb core panels. Therefore, the proposed method can be used to provide higher defect detection reliability by a double verification. Another research in [62] adopts various NDT techniques for off-line damage detection in three composite structures.

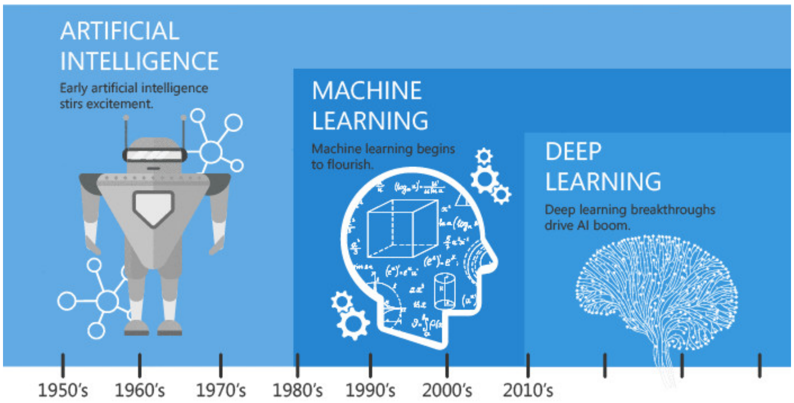
Both vibration-based and guided wave-based techniques have been proved capable of detecting damages occurred in composite structures and other materials. They have good performance in some certain conditions and show a good potential in the application in more fields. However, a common process of these two methods are obvious: the vibration responses of a structure should be analysed all by human labor. Thus, the limitations of these methods are obvious: firstly, they are time-consuming and labor-intensive. Secondly, there will be high requirements of expertise for practitioners, especially for complex structures, which is not always available.

With the development of Artificial Intelligence (AI) in the last two decades, the ability of computers to process information has greatly improved. It is possible to reduce labor loss and improve efficiency, and the advantages of artificial intelligence can sometimes avoid the high requirements of expertise for practitioners. Therefore, it provides a more efficient solution to the problems encountered in traditional SHM approaches.

1.3 Machine learning and its applications

Artificial Intelligence (AI) is the simulation of human intelligence processes by machines, especially computer systems. These processes include learning (the acquisition of information and rules for using the information), reasoning (using rules to reach approximate or definite conclusions) and self-correction.

Dating back to the 1950's when there were several waves of optimism, AI was founded. It was used to learn simple tasks that a man could do such as checkers strategies and solving word problems in algebra. Later in the 1980's, a sub-field of AI, machine learning (ML), begun to flourish. Machine learning uses algorithms to parse data,learn from it, and then make a determination or prediction about something in the world. With the boom of industries, data has exploded and conventional AI techniques are not powerful enough when confronting new complex problems. Therefore, as a sub-field of machine learning, deep learning encounters



Since an early flush of optimism in the 1950's, smaller subsets of artificial intelligence - first machine learning, then deep learning, a subset of machine learning - have created ever larger disruptions.

Figure 1.12: Evolution of Artificial Intelligence

opportunities for rapid development due to its capability of gathering huge amount of data from industrial machines and turning the big data into actionable information, see figure 1.12. Due to its powerful data mining capabilities by means of hierarchical representation of features, deep learning has gained increasing interest in various industry problems concerning decision-making and classification.

With the emergence of ML, various algorithms are constantly being studied. Since ML came into being with the booming development of industry, academic and industrial communities are always studying how to apply ML to industrial development in view of its powerful ability in processing data so as to improve efficiency and reduce costs. Therefore, research on ML is very hot, especially in the recent period of rapid development of information technology. It is widely applied, covering almost every field, such as agriculture, industrial production, banking, financial market analysis, economic forecasting, medicine, robotics and autonomous driving, social security, structural health testing, etc. It is for this reason that it is difficult to track its development status all the time. A recent review paper [63] that systematically and clearly sorted out the methods of machine learning, and listed some commonly used methods in various sub-fields, can provide a global understanding about ML, as shown in figure 1.13.

Machine learning can be traditionally categorized as supervised learning, unsupervised learning and reinforcement learning, according to the type of data as input and output, the type of tasks to be solved. In supervised learning, a model is constructed with a set of data including input and output. The purpose is to obtain an optimal mapping function that can match the input and output pair. According to different purposes, the model can be used as classifier (e.g., Bayesian, Neural networks, decision trees or deep learning, etc.) or as regression model (e.g.,

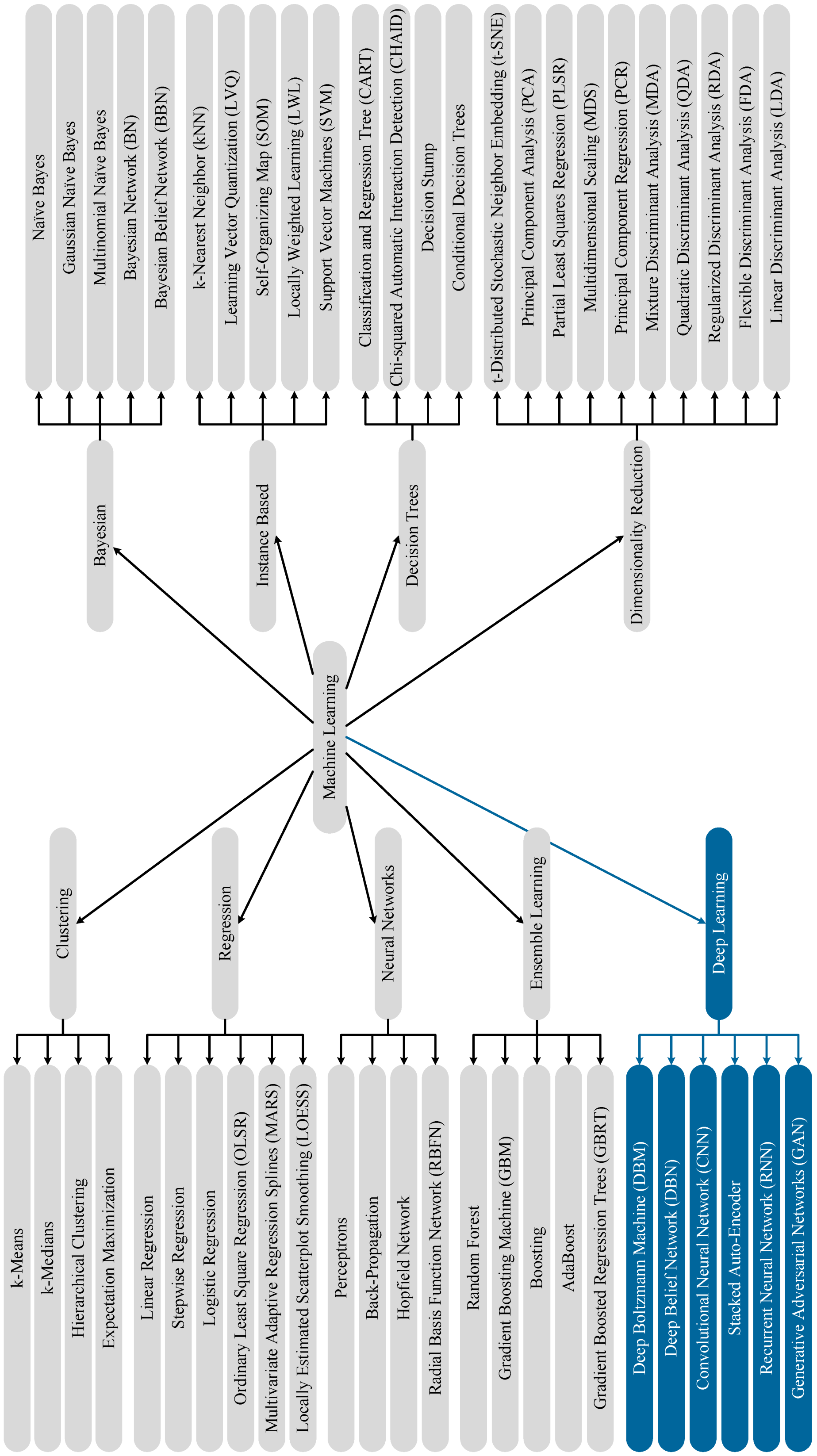


Figure 1.13: Machine Learning algorithms mind map [63]

linear and nonlinear regression, Bayesian, decision trees, deep learning, etc.). In unsupervised learning, the model clusters datasets by finding structure in its input without the input-output pairs through k-means, hierarchical clustering, etc. The idea of reinforcement learning is to make a computer program interact with the environment (e.g., autopilot, play with human go player) and provide feedback which is considered as reward to be maximized. Other approaches have also been developed such as dimensionality reduction, which aims to extract important information from data containing a large number of redundant variables or features through Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Multidimensional Scaling (MDS), etc.

1.3.1 Machine learning

Various types of machine learning models have been established and studied in the last few years. They have been trained with some training data and then can make predictions with additional data. The predictions in machine learning models are mainly for two purposes: regression and classification. Machine learning model is used for regression when the outputs are continuous values, and it is used for classification when the outputs are discrete or categorical variables. Several commonly used models will be briefly introduced.

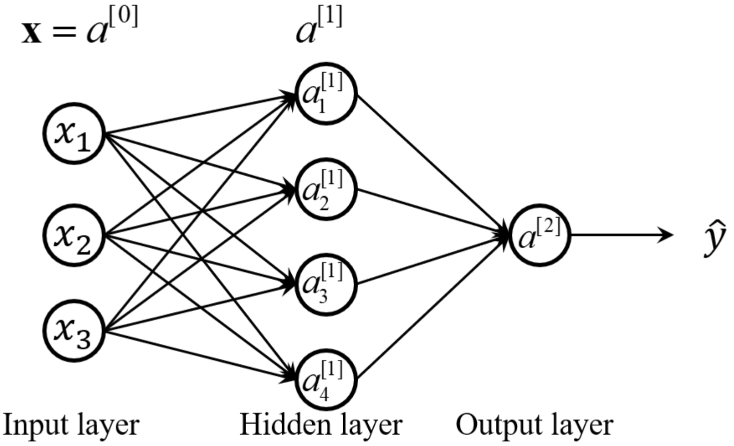


Figure 1.14: Representation of a 2 layered neural network

Artificial neural networks (ANN), or neural networks (NN), is a machine learning model inspired by biological neural networks that constitutes animal nervous systems. An ANN model consists of multiple layers including input layer, hidden layers and output layer where each layer consists of several "artificial neurons". These neurons in different layers are connected so that information can be transmitted from one layer to another. Hidden layers are proposed to extract hidden information of the analyzed data. Typically each neuron has a weight, which is used to adjust the learning process. In the learning process, a forward propagation is implemented to calculate the output. Then a backward propagation is implemented for the purpose of learning, which means to adjust weights in each neuron so that the predicted output fits the ground truth. A simple two-layered ANN is illustrated in figure 1.14. It should be noted that in ANN the input layer is not counted while counting the number of layers. The first layer is denoted by $\mathbf{a}^{[0]}$, second layer by $\mathbf{a}^{[1]}$ and third layer by $a^{[2]}$. Here 'a' indicates activations, which are the values that different layers pass on to the next layer. The mathematical operations of neural networks are introduced that will be useful for understanding deep learning mechanisms. In the hidden layer, each neuron is expressed as:

$$\begin{aligned} a_1^{[1]} &= \sigma\left(\mathbf{w}_1^{[1]T}\mathbf{x} + b_1^{[1]}\right), \\ a_2^{[1]} &= \sigma\left(\mathbf{w}_2^{[1]T}\mathbf{x} + b_2^{[1]}\right), \\ a_3^{[1]} &= \sigma\left(\mathbf{w}_3^{[1]T}\mathbf{x} + b_3^{[1]}\right), \\ a_4^{[1]} &= \sigma\left(\mathbf{w}_4^{[1]T}\mathbf{x} + b_4^{[1]}\right) \end{aligned} \tag{1.1}$$

where $\mathbf{w}_1^{[1]}$ and $b_1^{[1]}$ are the weight and bias for the first neuron of the first layer, respectively, and the same for other neurons. In this example the input scalars x_i with $i = 1, 2, 3$ is of dimension 1×1, thus the weight $\mathbf{w}_1^{[1]}$ is of dimension 3×1 and the bias b is of dimension 1×1. The output layer is expressed as:

$$a^{[2]} = \sigma\left(\mathbf{w}^{[2]T}\mathbf{a}^{[1]} + b^{[2]}\right) \tag{1.2}$$

where $\mathbf{a}^{[1]}$ is the output of the precedent layer of dimension 4×1, $\mathbf{w}^{[2]}$ is the weight of the second layer of dimension 4×1, $b^{[2]}$ is the bias, and the output $\widehat{y} = a^{[2]}$ is a scalar. ANN have been applied in various fields, including computer vision, speech recognition, machine translation, etc.

Decision tree learning is one of the predictive modeling approaches used in statistics, data mining and machine learning. A decision tree is a model that mimics a tree with root, branches and leaves, but is drawn upside down with its root at the top. Decision trees are of two main types: classification tree and regression tree. Classification tree is used when the target variable is a set of discrete value while regression tree is used when target take continuous values. In both models, the root represent the input data, branches represent different features and leaves represent class labels for classification or continuous values for regression. Decision tree models are easy to understand and interpret, but they can be very non-robust because a small change in the training data may result in large difference in predictions. A simple example of decision tree in the domain of cardiovascular diseases is illustrated in figure 1.15. It is a classification tree.

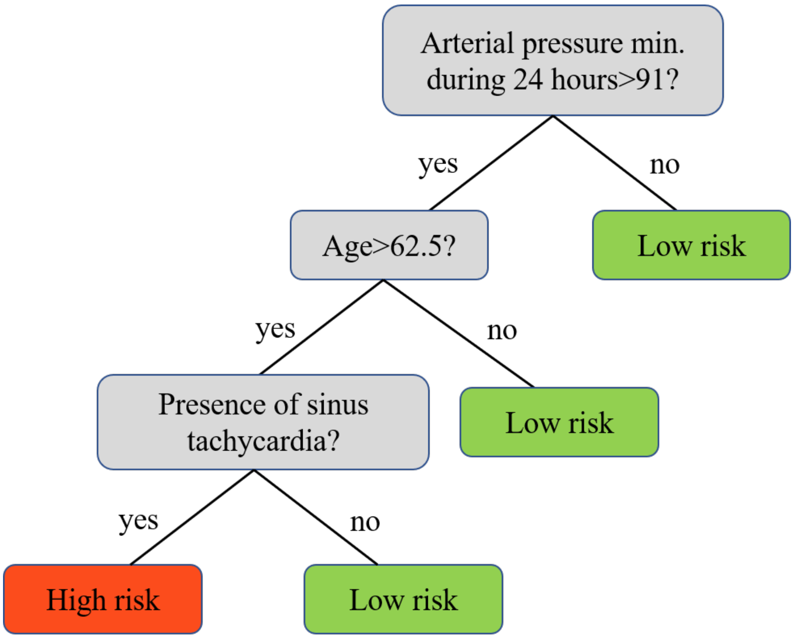


Figure 1.15: Example of decision tree used in the domain of cardiovascular diseases

Support vector machines (SVM) is a machine learning method based on statistical learning theory developed in the mid-1990s. It seeks to maximize the learning machine's generalization ability by minimizing the expected risk [64]. A SVM training algorithm is a non-probabilistic, binary, linear classifier. It constructs a hyper-line or a hyper-plane in a higher dimensional feature space to separate training data belonging to two different categories. The classifier is trained so that the training data is well separated, that's to say, the distance to the nearest training data point of any class is maximized. SVMs can be applied to solve various problems such as text categorization[65], image classification[66] and biological problems[67].

There are some other conventional machine learning approaches like Bayesian networks, Naïve Bayes, Random forest, Gaussian Processes, etc. Among them, Gaussian Process will be introduced in detail in chapter 3.

1.3.2 Deep learning

As the hottest sub-field of machine learning, deep learning has gained increasing attention in recent years and has shown great potential in many fields including biology, medicine, architecture, transportation, defense, etc. Here three commonly used deep learning approached will be briefly introduced to provide a preliminary understanding.

Auto-encoders (AE) are neural networks that aim to efficiently compress and encode data into a latent space representation and then reconstruct the output from this representation. The objective is to make the output as close to the original signal (the input) as possible. It's an unsupervised neural network. AE consists of two main parts: (1)Encoder, in which the model learns how to reduce the input data dimensions and compress it into a latent-space representation. (2)Decoder, in which the model learns how to recover the input data as accurately as possible from the encoded representation. In applications, AE can exponentially reduce the computational cost by reducing data dimensions. But it learns to capture as much information as possible rather than as much relevant information. It might not be able to determine what information is relevant[68]. In applications, AE can also be used for noise reduction purpose due to its capability of reducing data dimensions. Noise in the data can be ignored by learning. An example of image denoising using AE is illustrated in figure 1.16. The input/output image is from the MNIST dataset.

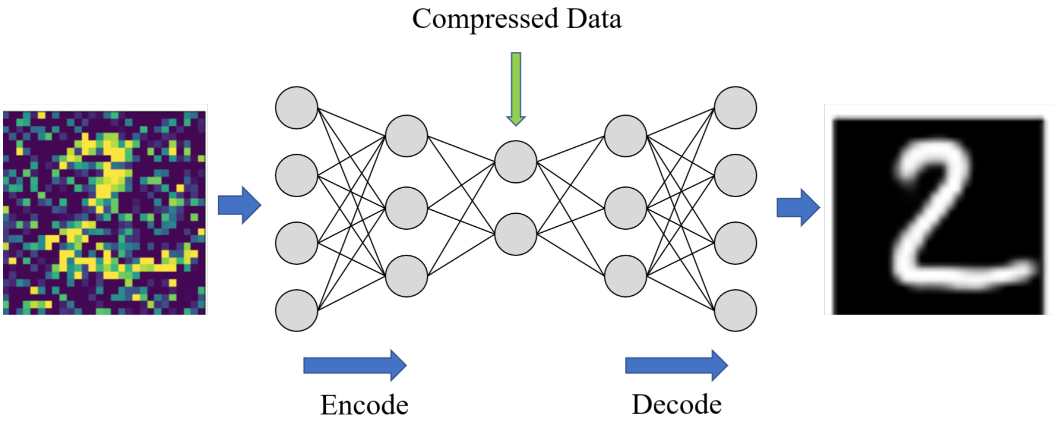


Figure 1.16: Example of Auto-encoders for image denoising in MNIST dataset

Convolutional Neural Network (CNN) is a kind of neural networks in deep learning. It's similar to the ordinary neural networks consisting of neurons which have adjustable weights and biases. The difference is that in an ordinary neural network, every neuron in each layer is fully connected with all neurons in the neighbour layer, which makes the network prone to overfitting data, while in CNN filters are used between layers and for each layer, each output is only related to a portion of the input, instead of taking into account all the inputs. In CNN, there are three main layers besides the input layer and the output layer, that is convolutional layer, pooling layer and fully connected layer. Detailed interpretation of CNN will be given in chapter 4.

Recurrent Neural Networks (RNN) is the deepest of all neural networks[69]. It is a deep learning approach to deal with sequential data. Consider a simple one-layered neural network, the hidden layer has one input vector and one output vector. A RNN is a combination of N such networks and each hidden layer's output is propagated through time to be used as input for the next hidden layer. It means that the output vector's contents are not only dependent on the input you just fed in, but also on the entire history of inputs you have fed in in the past. To make it easier to understand, imagine that when you are reading a sentence, you can understand every word that you have seen based on your knowledge, but you need to recall the previous words so that you might predict probable words in the next sequence of the sentence. This is the biggest difference compared to other machine learning and deep learning approaches. Nevertheless, there is a short-term memory problem for RNN, the short-term memory has a large impact but long-term memory effects are small. That means, the output of the first layer has the lowest impact on the final result while the output of the penultimate layer has the largest impact on the final result. This problem makes it difficult for RNN to deal with very long input sequences. An example of RNN is shown in figure 1.17. In the input layer, x_t is an input at time step t. In the hidden layer, $h_t$represents a hidden state at time t and acts as "memory" of the network. It is calculated based on the current input x_t as well as the previous time step's hidden state: $h_t = f(W_x x_t + W_h h_{t-1})$, where the function f is a nonlinear transformation, e.g., sigmoid, ReLU (Rectified Linear Unit), tanh. It should be noted that in RNN, weights exist not only in the connection of input layer and hidden layer, and the connection between hidden layer and output layer, they also exist between two adjacent hidden states. Thus,

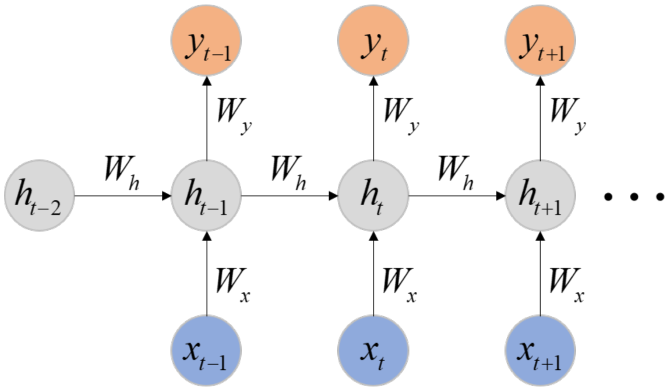


Figure 1.17: A Recurrent Neural Network with input layer, hidden layer and output layer

there are three types of weights, (W_x, W_h, W_y). These weights are shared across the whole time series. The output layer is calculated based on the current hidden state which has already taken the previous time step's hidden state and current input into account: $y_t = f(W_y h_t + b)$ where b is a bias term.

1.3.3 Applications of machine learning

AI approaches including conventional machine approaches and deep learning approaches have been well developed and applied in SHM for multiple materials and structures. In this part, applications of AI approaches in SHM will be reviewed, and SHM for composite materials will be emphasized.

A general process of a data-driven based SHM is shown in figure 1.18. Firstly, relevant data from various sensors attached on the structure is collected. Then, necessary process is conducted including signal preprocessing, feature extraction, feature classification and predictions by means of conventional machine learning approaches or deep learning approaches.

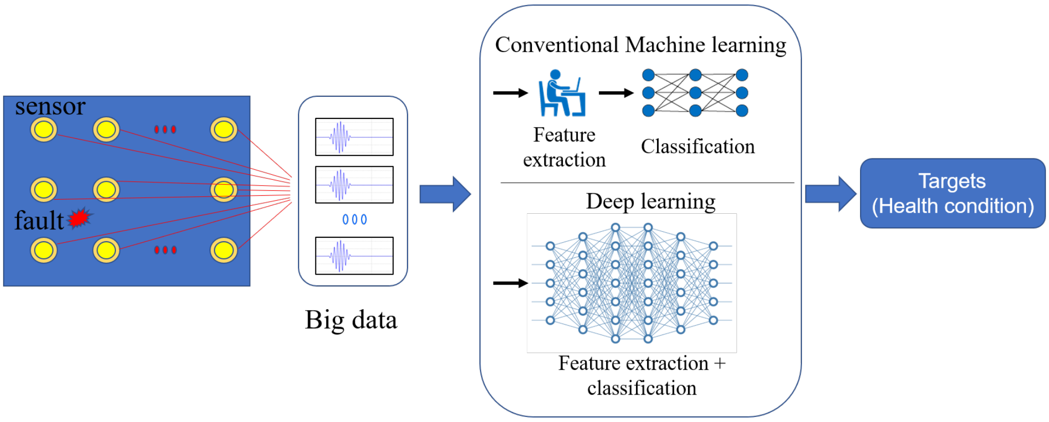


Figure 1.18: Illustration of Structural Health Monitoring (SHM) system

An ANN model was proposed by L.H. Yam et al.[30] to detect damages occurred on a composite sandwich plate by both numerical and experimental approaches. A validated numerical model of the sandwich structure is used to simulate the vibration responses of the structure. 121 sets of data in which 1 set is of intact structure and 120 sets are of damaged structure are obtained and used for the training and validation of ANN. In turn, the trained ANN is used for testing experimental data. Results show that ANN is powerful to establish mapping relationships between damage feature and structural status, but the accuracy of the result depends on the damage severity. Another ANN model was proposed by O'Brien R. J. et al. in [70]. Healthy and damaged composite beam's vibration responses are experimentally collected by a microphone. Singular Value Decomposition (SVD) and Principal

Component Analysis (PCA) are implemented to filter the signals and extract relevant information, which in turn are fed to the ANN model for training and test. Results show that the proposed approach is capable to detect damages with accuracy greater that 97% for carbon fiber and glass fiber beams. The generalization of the proposed method to more complex composite structures should be investigated.

In [35] an optimized methodology for delamination identification on laminated carbon fiber reinforced polymer (CFRP) plates using Genetic Algorithm (GA) and Artificial Neural Networks (ANN) was proposed. The acceptable accuracy of damage location prediction shows that the proposed method GA-ANN is effective for the detection of delamination damage in laminates, but it cannot provide precise location and severity predictions simultaneously. A Bayesian framework in the context of Gaussian Process was proposed in [71] to detect the existence of damages in a CFRP sandwich structure of dimensions 217 mm × 217 mm × 31 mm. Experimental data is obtained and used for the training of Gaussian Process model. The assessment of the damage detection capabilities of the presented method is performed by Receiver operating characteristics (ROC). A limitation of the proposed method is that experimental data for a larger and more complex structure is not always available.

In [72] authors proposed a Gaussian process–based approach to deal with uncertainty in the pattern recognition problem in structural health monitoring of a multi-layer composite airfoil structure. The proposed method shows obvious advantage compared to other neural networks dealing with noisy data. It is also suggested to assess the method in other cases in order to investigate its performance. Simulated vibration responses of smart composite laminates were performed in [73]. The discriminative features extracted from structures in different health status are used for the training and cross-validation of various supervised machine learning classifiers and an optimal classifier is identified. The Support Vector Machine (SVM) and Multilayer Perceptron (MLP) are the optimal classifiers. Results show that the proposed method can be employed as a reliable tool for NDT of smart composite laminates. Several other works in which machine learning approached are implemented for SHM of structures such as gas turbine engines[74, 75, 76] and bearings[77, 78, 79] made of other materials rather than composites have also been widely conducted .

As the limitations for traditional machine learning methods can be avoided through deep learning, such as the hand-crafted features, deep learning approaches are increasingly applied in various fields, like medical field for diagnoses and decision-making, engineering field for vision-based SHM and vibration-based SHM, etc. Based on the powerful data processing and feature learning capabilities of deep learning, it has been adopted in medical field to assist the diagnoses of cardiovascular diseases [80], distinguishing inflammation caused by bacteria or viruses based on images [81], as well as healthcare decision making based on electronic medical records.

Most traditional damage detection in infrastructures relies on visual assessments, which required the expertise of practitioners. It is time costly and labor-consuming. With the development of deep learning, algorithms becomes an alternative to humans and are able to perform inspections on raw images. A CNN-based method was proposed in [82] to classify crack and non-crack noise patterns from raw concrete images. The crack identification approach is proven to have a strong potential for automated crack assessment of concrete structures. Chen and Jahanshahi [83] proposed a CNN-based crack detection method to inspect nuclear power plant components while using Naive Bayes data fusion scheme from each video frame. The proposed framework achieves an accuracy of 98.3% per frame, which is significantly higher than other methods presented in the research.

It is sometimes important to have a global view when dealing with SHM of a complex system instead of a component of the system, since it is necessary to understand the relationship between the whole system and each component as well as the relationship between a damage in a component and the safety of the whole system. Therefore, the recognition of components should be put before damage detection of components in specific situations. Yeum et al. [84] developed a CNN-based automated region-of-interest localization and classification method for infrastructures using raw images. After that, highly relevant regions of interest can be extracted for further visual inspections. Narazaki et al. [85] proposed a CNN and RNN-based approach for automated bridge component recognition using video datasets.

In some cases, however, necessary data for the vision-based SHM are not available, for example in some historical sites, protective buildings or bridges of which some parts are not accessible for inspectors. The development of Unmanned Aerial Vehicles (UAVs) has shown the ability to overcome the drawbacks of manual visual inspection and has broadened the vision-based SHM. Kim and Cho [86] demonstrated that real-time video frames acquired by UAV from concrete surfaces can be used in a CNN-based deep learning algorithm for crack detection. Huynh et al. [87] evaluated a RCNN-based deep learning algorithm for the detection of bolt-loosening on a real-scale girder bridge using images captured with an UAV. Hoskerre et al. [88] used UAVs to acquire data and used deep learning algorithm to achieve rapid and automated post-earthquake inspections.

In vibration-based SHM in various engineering sections including bearings, rotating machines, bridges, steel frames, etc., deep learning is also widely applied. To the authors' knowledge, although there are some related studies of SHM for composite materials, the research in this area is very scarce, and there is no attempt to apply deep learning in SHM for composite sandwich structures.

A CNN-based approach was proposed for the classification and prediction of various types of in-plane and through-the-thickness delamination in smart composite laminates [89]. A total of 13 cases including one healthy and 12 delaminated cases were investigated. Short Time Fourier Transform (STFT) is employed to transform the vibration responses of healthy and damaged structures into 2D images, which in turn are fed to CNN for training and classification. The trained CNN classifier is then evaluated with unseen cases of delamination. A total of 13 classification labels corresponding to the 13 cases were used, where one case indicates the healthy state of the structure while the others indicate delaminated cases and each label corresponds to a specific damage location. Therefore, the prediction using the proposed neural network can provide both damage occurrence and damage location information. The CNN showed an overall classification accuracy of 90.1% which is acceptable compared to results obtained by other machine learning approaches. The proposed approach could be extended to assess other types of damages in composite laminates.

A deep learning-based framework was proposed by Meng et al. [90] for the classification of ultrasonic signals from CFRP laminates with void and delamination. CNN is employed to learn high-level representation for each signal from wavelet transform. A linear SVM is used on the top layer for classification tasks. Results are compared with those using handcrafted features, different loss functions and different classifiers, respectively. It's shown that the chosen methods including Wavelet transform, CNN, SVM have optimal performance. Deep learning is employed in SHM for various other types of materials.

CNN-based approach was proposed in [91] and [92] for joint damage detection on a steel frame of Qatar University grandstand simulator. Damages are simulated by loosening the bolts at connection joints. A unique 1D CNN for each one of the 30 joints instrumented with accelerometers is designed. Raw 1D signals from healthy and damaged structures are directly fed into the related CNN. Therefore, the health state of each joint can be detected by the corresponding CNN. Results show that the proposed approach is capable of extracting optimal damage-sensitive features automatically from the raw acceleration signals, allowing it to be used for real-time damage detection.

In order to overcome the two major deficiencies of ANN: (1) manually extracted features require expertise, (2) ANN is not capable to learn complex non-linear relationships in fault diagnosis issues, Jia et al. [93] proposed a DNN-based approach to mine damage-sensitive information from raw data and make predictions of structural health state for bearings. Results show that the proposed method can not only mine available fault characteristics from raw signals, but also obtain superior detection accuracy compared with existing methods.

X. Guo et al. proposed a hierarchical adaptive deep convolution neural network (ADCNN) for SHM of bearings [94]. In this work, experimental dataset provided by Case Western Reserve University is adopted, in which there are 3 types of damage, each of which has three severity levels, along with 1 healthy state. ADCNNs were constructed for recognizing fault types and predicting fault size, respectively. The first ADCNN architecture has seven main layers, where the first layer is used to convert the signal-vector input into a matrix. The followed 3 layers are ConvNet layers, each of which consists of a convolution layer and a max-pooling layer. Two fully connected layers are followed by a softmax layer after the ConvNet layers. After fault-pattern recognition in the first ADCNN, 3 ADCNNs are constructed for each fault type. They have the same architecture as the first ADCNN. After training, test samples are used in each ADCNN and the output provides the probability that the given sample belongs to each class of the three severity levels.

Zhang et al. proposed an end-to-end deep CNN-based method which takes environmental noise and working load changes into consideration for fault diagnosis of bearings in [95]. Experimental dataset provided by Case Western Reserve University is adopted to train and validate the proposed method named Convolution Neural Networks with Training Interference (TICNN). In the dataset, there are three types of damage, each with 3 severity, plus a healthy model, for a total of 10 cases, corresponding to 10 classification classes. The proposed model takes raw temporal signals as inputs and doesn't need any denoising pre-processing. It can achieve pretty high accuracy under noisy environment and even when working load is changed, which outperforms DNN. They proposed a similar approach in [96] that can achieve 100% classification accuracy on raw 1D vibration signals.

T. Ince et al. proposed a 1D CNN for real-time motor fault detection in [97]. Raw signals are pre-processed and only two classes are introduced - health, damage. CNN-based fault detection was proposed in [98, 29] for SHM of bearings. In these works raw amplitudes of the frequency spectrum of vibration signals are used as input to proposed models during training and testing steps. Although CNN is a powerful tool of classification for images, these works prove its capability of classifying 1D signals.

Ding and He [99] proposed an energy-fluctuated multi-scale feature learning approach based on wavelet packet energy (WPE) image with deep CNN for intelligent spindle bearing fault diagnosis. 2D WPE image of the frequency subspaces is used as input to the deep CNN. The comparison of the proposed feature mining approach outperforms others and is quite suitable for spindle bearing fault diagnosis.

Vision-based method using deep CNN architecture was proposed in [100, 101] for concrete crack damage detection. Intact and crack images of concretes are fed into CNN models to learn damage-related features and detect the presence of cracks. Results show that only using vision images, deep CNN models can provide reliable crack detection in concretes and can also minimize false alarms. Other related works in which deep learning is employed for SHM are referred to [102] for surface fatigue crack identification in steel box girder of bridges using CNN based on consumer-grade camera images, and [103] data anomaly detection based on acceleration data from the structural health monitoring system of a real long-span bridge using DNN.

1.4 Conclusion

In this chapter, the research background of composite structures, especially composite sandwich structures and its advantageous properties compared with traditional alloy materials, is firstly presented. However, since composite materials are susceptible to several damages, the concept structural health monitoring (SHM) is then introduced and SHM methods that have been studied in recent years are reviewed. With the development of artificial intelligence, it is more and more used in engineering problems, providing new possible solutions to the disadvantages of traditional structural health detection methods. Several machine learning and deep learning methods are introduced and its applications in engineering problems, especially in SHM are emphasized.

CHAPTER 2

Structure modeling and features extraction techniques

Contents

In data driven-based SHM systems, information containing the structure's health status is quite important for the performance of the system. Thus, as many structural status as possible should be collected to construct a reliable database because different sets of data correspond to different healthy conditions of a structure. However, it is neither economic nor realistic to build real structures of different health status, especially for huge and expensive structures to construct a database. To understand the behaviour of a structure, numerical simulation of structures in different conditions is a promising idea. Therefore, SHM achieved by simulation becomes a compromise way.

2.1 Finite element method

Understanding the behaviors and characteristics of a complex object is often impossible to achieve through a simple act. A natural way is to subdivide a large system into finite number of smaller, simpler components or elements, whose behavior is readily known. This process is called "discretization". Then the original system is rebuilt from such components and elements in order to study the behavior of the entire system as a whole. Based on this idea, the Finite element method (FEM) is proposed to solve problems of engineering and mathematical models encountered in structural mechanics, heat transfer, electric and magnetic potential and fluid flow, etc [104].

The characteristics of an object are usually determined by the laws of physics. The description of the laws of physics related to space-time problems is generally expressed by *partial differential equations* (PDEs). Therefore, for most geometries and problems, important variables and parameters can be obtained by solving these PDEs. However, analytical solutions for these PDEs are not always available. Instead, they can be approximated by introducing approximation equations based on different types of discretization. The FEM subdivides the studied object into small finite elements, which can be expressed by a system of numerical model equations. These simple equations that model the finite elements then constitute a larger system of equations that models the entire problem. The global system of equations has known solution techniques and can be calculated from initial conditions to obtain a numerical answer. The solution to the system of equations is an approximation of the real solution to the PDEs. FEM is used to compute such approximations.

A variety of softwares for solving engineering problems have been formed based on the laws of physics in different fields using FEM. The laws of physics are expressed in mathematical models in corresponding fields such as structural mechanics and electromagnetics. Numerical model equations are then obtained from mathematical models discretized by the FEM. The solutions of the numerical model equations from the software are described in space and time, which will be used to understand the behavior of the studied system as well as making predictions about it.

ANSYS is one of the most commonly used softwares for simulation. The Ansys Parametric Design Language (APDL) is a powerful scripting language dedicated to solving a series of problems such as thermal, structural, acoustic, piezoelectric and some others that can be simulated. One essential task for FEM software is to build a model of the studied object with appropriate elements in the analysis of structural mechanics since different types of element may lead to different performances of the simulation. In this study, two elements SHELL 181 and SOLSH 190 in ANSYS are mainly used to build the studied structure - composite sandwich plate. The comparison of these two elements in [105] suggests that material properties and geometries of these elements are similar to those used in aerospace industry. A brief introduction of these two elements will be conducted as follows.

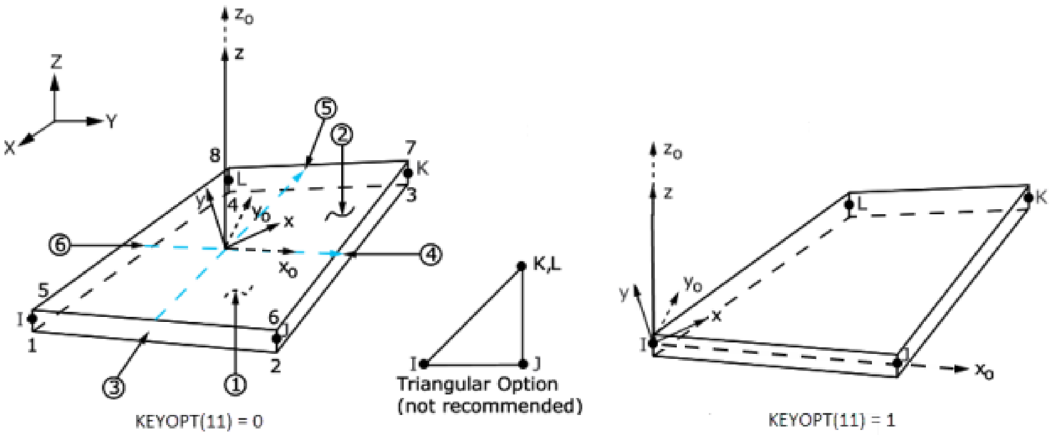


Figure 2.1: Geometry of shell element SHELL 181 [106]

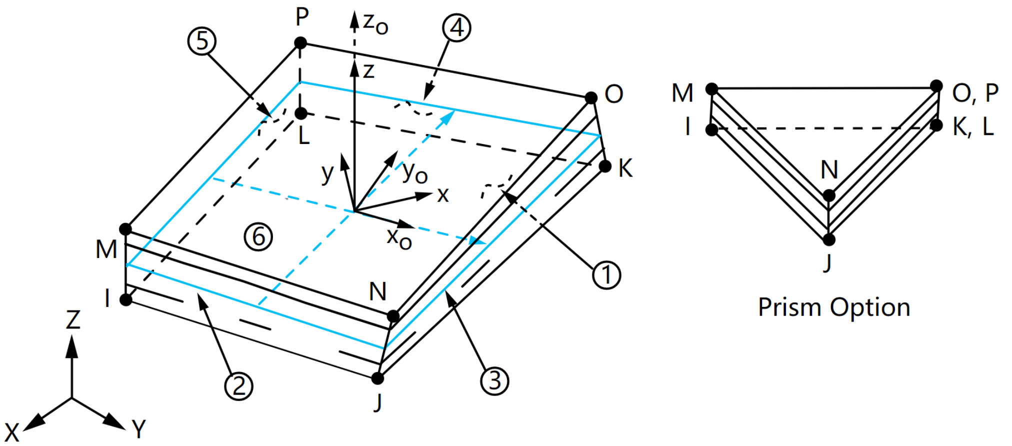


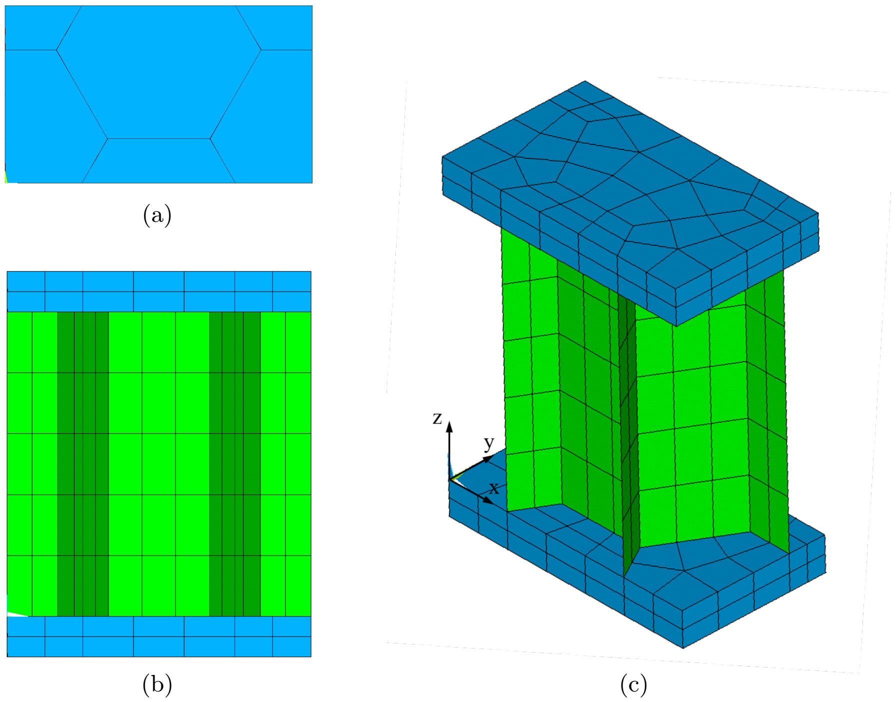
Figure 2.2: Geometry of solid-shell element SOLSH 190 [106]

Shell element SHELL 181 is a shell element suitable for analyzing thin to moderately-thick shell structures. It has four nodes with six degrees of freedom at each node: three translations and three rotations along x, y and z axis, as shown in figure 2.1. The element is defined by four nodes I, J, K and L. It can be used for layered applications in composite shell models and sandwich construction. The bending deformation is taken into consideration while the transverse shear deformation is neglected.

Solid-shell element SOLSH 190 is suitable for simulating shell structures with a wide range of thickness. The element has a topology of continuum solid element. It has eight nodes with three degrees of freedom at each node: translations in the x, y and z directions. The geometry, node locations and the element coordinate system of the element are shown in figure 2.2. The connection of this element with other continuum elements does not need extra efforts. This element has the properties of plasticity, hyperelasticity, stress stiffening, creep, large deflection and large strain. It can be used to simulate laminated shells and sandwich construction.

2.2 Structure modeling

In the present work, a hexagonal shaped Nomex honeycomb core (3D core) and CFRP-skin made composite sandwich plate is selected as the study target, based on which machine learning-based Structural health monitoring is conducted. The composite sandwich plate is a periodic structure, in which each unit cell structure has the form of honeycomb. The geometry of a unit cell is illustrated in figure 2.3 with a top view, a front view and a corner view. Each cell is composed of a honeycomb core and two face sheets. The height of the honeycomb core is $H = 15mm$ in which the radius and wall thickness are $r = 5mm$ and $t = 0.2mm$, respectively. The thickness of each face sheet is $h = 2mm$.



(a)

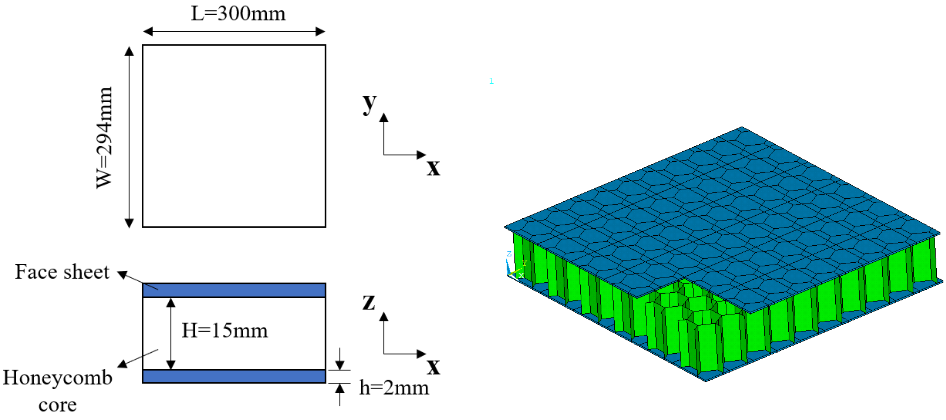
(b)

(c)

Figure 2.3: Geometry of a unit cell of the composite sandwich plate with a core in the form of honeycomb (green) and two face sheets (blue). (a) Top view, (b) Front view, (c) Corner view

The unit cell structure is repeated 20 times in x direction and 34 times in y direction to form a rectangular plate. The plate of spatial dimensions 300 mm ×

294 mm × 19 mm consists of a honeycomb core and two face sheets. The Nomex core's material properties are: $\rho_1 = 1240 kg/m^3$, $\nu_1 = 0.33$, $E_1 = 5.5 \times 10^9 Pa$. The face sheets are laminate panels made of carbon/epoxy materials whose properties are: $\rho_2 = 1850 kg/m^3$, $\nu_2 = 0.3$, $E_2 = 70 \times 10^9 Pa$. A simplified illustration of the sandwich structure is referred to figure 2.4(a).



(a) (b)

Figure 2.4: (a) Simplified geometry from top view and front view and (b) numerical model of the composite sandwich panel from corner view

The honeycomb sandwich plate consisting of a honeycomb core and two skin plates is shown in figure 2.4(b). Within our study the dimensions of the face sheets and the core will not change. The structure is modeled in commercial software ANSYS®.

The face sheets are modeled by the SOLSH190 solid-shell element which has 8 nodes with 3 degrees of freedom at each node: translation in x, y and z directions. In this manner, face sheet shear phenomena can be taken into account when a signal with high frequency propagates in the structure. 2D shell element SHELL181 is used to model the honeycomb cells. The transition area where the face sheet connect to the honeycomb cell is often the weakest region. In our study, the elements of shell and solid-shell are connected with an automatic constrain equation developed by ANSYS®. The face sheets and core share the same nodes and the mesh is compatible at the interface between the two parts of the sandwich plate. The total number of meshing elements is about 177000, which guarantees a minimum mesh density of 10 elements per wavelength for the model.

2.3 Model validation

Whatever modeling technique is being used, model validation is an essential procedure to assess if a numerical model is a "good" representation of the reality. There is no doubt that judging a numerical model is "good" or not is subjective. However, from the perspective of model performance, the quality of a model can be measured according to the difference between the model's performance and the represented system. In other words, it can be achieved by checking if the numerical model possesses essential features of a real model. In this study, the FEM model is validated by two approaches: comparing wave propagation velocity of simulation with analytical results, comparing the first 5 natural frequencies with a simplified homogenized sandwich model.

2.3.1 Group velocity of wave propagation

Group velocity of the wave propagation in the sandwich structure is evaluated by analytical and numerical methods. In numerical method, as the honeycomb structure has a orthotropic property due to its geometry, which means the panels react differently depending on the orientation of the structure. It is therefore necessary to distinguish between the directions of symmetry, the so-called L and W-direction. In the present study, only the wave velocity along the L-direction (x direction) is considered.

The governing equations for vibration of sandwich structures are described in [107]. The total deflection of a sandwich plate w is due to the bending of the core denoted by β and the angular displacement due to the shear in the core denoted by γ:

$$\frac{\delta w}{\delta x} = \beta + \gamma \tag{2.1}$$

The differential equation governing w, β and γ can be obtained by Hamilton's principle:

$$\delta \int \int (U + V - T)\, dxdt = 0 \tag{2.2}$$

where U is the elastic energy per unit length, V is work per unit length done by external forces and T is the kinetic energy per unit length. Here external forces are not considered, so $V = 0$. Combining equation 2.1 and 2.2, the following equations governing w, β and γ can be obtained:

$$-G_e H \left(\frac{\delta^2 w}{\delta x^2} - \frac{\delta \beta}{\delta x} \right) + 2D_2 \left(\frac{\delta^4 w}{\delta x^4} - \frac{\delta^3 \beta}{\delta x^3} \right) + \mu \frac{\delta^2 w}{\delta t^2} = 0 \tag{2.3}$$

$$-D_1\frac{\delta^2\beta}{\delta x^2}+2D_2\left(\frac{\delta^3 w}{\delta x^3}-\frac{\delta^2\beta}{\delta x^2}\right)+I_\rho\frac{\delta^2\beta}{\delta t^2}-G_eH\left(\frac{\delta w}{\delta x}-\beta\right)=0 \tag{2.4}$$

where G_e is the effective shear stiffness of the core, H and h are the height of the core and thickness of the face sheet, respectively. D_1 and D_2 are the overall bending stiffness of the core and the bending stiffness of the face sheet, I_ρ is the mass moment of inertia per unit width, and μ is the mass per unit area. The expression of D_1, D_2, I_ρ and μ are given in [19] as below:

$$D_1=E_1\frac{H^3}{12}+E_2h\left(\frac{H^2}{2}+Hh+\frac{2h^2}{3}\right) \tag{2.5}$$

$$D_2=E_2\frac{h^3}{12} \tag{2.6}$$

$$I_\rho=\rho_{1e}\frac{H^3}{12}+\rho_2h\left(\frac{H^2}{2}+Hh+\frac{2h^2}{3}\right) \tag{2.7}$$

$$\mu=2h\rho_2+H\rho_{1e} \tag{2.8}$$

where ρ_{1e} is the effective density of the core. The deviation of the expression of the core effective density ρ_{1e} as well as the core effective shear stiffness G_e can be found in [108], and they are written as below:

$$\rho_{1e}=\rho_1\left(1-\left(\frac{l-\frac{t}{2\cos\theta}}{l}\right)^2\right) \tag{2.9}$$

where l is the side length of a honeycomb cell, which here is a regular hexagon, thus $l=r$. $\theta=30°$.

$$G_e=E_1\left(\frac{t}{l}\right)^3\frac{1+\sin\theta}{\cos\theta}\frac{1}{C} \tag{2.10}$$

where C is expressed as:

$$C=\left[3+\left(\frac{t}{l}\right)^2\left((2.4+1.5\nu_1)(3+\sin\theta)+(1+\sin\theta)\left((1+\sin\theta)\tan^2\theta+\sin\theta\right)\right)\right] \tag{2.11}$$

Assuming that plane waves propagate along direction x and have a form $w(x,t)=\omega_0e^{-j(\omega t-kx)}$. The wave number k must satisfy the 6^{th} order dispersion relation in the sandwich plate:

$$2D_2k^6-\frac{2D_2}{D_1}I_\rho k^4\omega^2-\left(\mu+\frac{2D_2}{D_1}\mu+\frac{I_\rho G_eH}{D_1}\right)k^2\omega^2+G_eH\left(k^4-\frac{\mu}{D_1}\omega^2\right)+\frac{I_\rho\mu}{D_1}\omega^4=0 \tag{2.12}$$

By introducing two hypotheses: the rotation inertia per unit width is small compared with the shear effects $I_\rho \omega^2 \ll G_e H$, and the bending stiffness per unit width of the face sheet is negligible in comparison with the overall bending stiffness $D_2 \ll D_1$. Therefore the 4^{th} order dispersion relation of the sandwich plate is obtained:

$$Sk^4 - \mu\omega^2 k^2 - \mu\omega^2 S/D_1 = 0 \tag{2.13}$$

where $S = G_e H$ is the transverse shear rigidity of the plate. Therefore the group velocities corresponding to the 6^{th} and 4^{th} dispersion relation can be obtained by the relation of wave number k and angular frequency ω defined by:

$$v_g = \frac{\delta k}{\delta \omega} \tag{2.14}$$

Material characteristics are listed in table 2.1. The core is described with effective properties after homogenization in the table.

Table 2.1: Dimensions and material properties for the sandwich plate. Here the properties of the core are results after homogenization.

Material	Thickness	Density	E (Pa)	G (Pa)
Skin	2 mm	$1850 kg/m^3$	$E_2 = 70$	$G_2 = 26.9$
Core	15 mm	$56.6 kg/m^3$	$E_e = 8.63 \cdot 10^5$	$G_e = 2.1658 \cdot 10^5$
Overall	19 mm	μ =$8.25 kg/m^2$		

Group velocities obtained by the 4^{th} order dispersion relation for the sandwich structure in x direction is shown in figure 2.5.

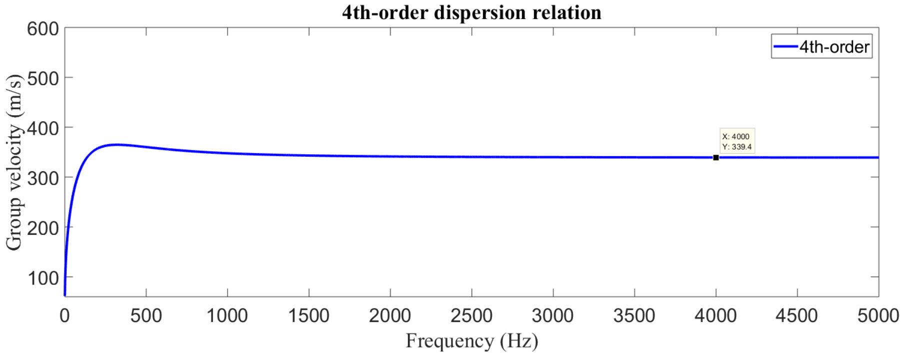


Figure 2.5: Group velocities obtained by the 4th order dispersion relation for the sandwich structure.

To obtain the group velocity along direction x in a numerical sandwich model, a 3D sandwich beam that has the same cell structure as the sandwich plate is

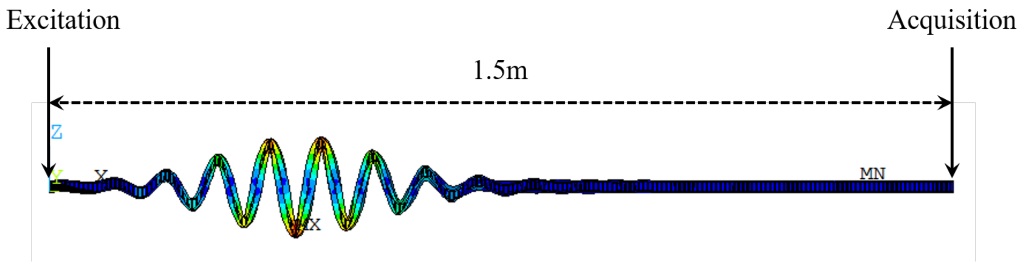
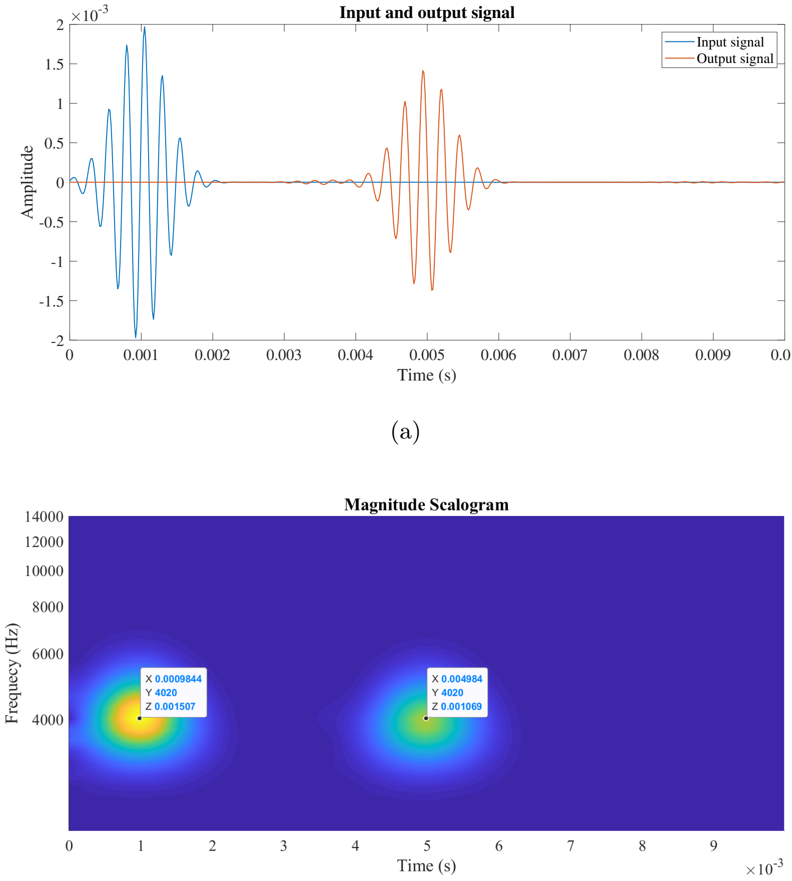


Figure 2.6: Illustration of the beam for the measurement of group velocity in direction x. The length is L=1.5m. Excitation signal is a pulse of frequency 4000 Hz



(a)

(b)

Figure 2.7: Continuous Wavelet Transform of the input and output signal in the sandwich beam, with input signal 4kHz. (a) Vibration signal, (b) CWT

constructed with length L=1.5m, as shown in figure 2.6. Note that a beam is used for the sake of saving computational cost. A pulse excitation signal with frequency

$f = 4000Hz$ is applied on one edge of the beam. Simulation is conducted for the propagation of the signal, with a total calculation time of 0.01 s, which allows the signal propagate to the other edge. Propagated signal is collected at the other edge with a distance of 1.5 m from the input point. Continuous Wavelet Transform is employed to extract the Time of Flight (ToF) of the signal. The ToF is defined as the time difference between the maximum energy of the signal at the central frequency of the output signal and input signal.

As shown in figure 2.7 obtained from Continuous Wavelet Transform, the ToF is obtained as $\Delta t = 3.9996ms$. Therefore, the group velocity for a signal of frequency 4kHz in the sandwich beam is :

$$v_{gs} = \frac{L}{\Delta t} = 375.1m/s$$

where the index gs indicates group velocity by simulation. Comparing to the analytical group velocity corresponding to the frequency 4kHz obtained in figure 2.5,

$$v_{ga} = 339.4m/s$$

where the index ga indicates analytical group velocity. There is a error of 9.5% between numerical and analytical group velocities.

2.3.2 Homogenized model

Construction of a homogenized sandwich model is conducted aiming at comparing the first 5 natural frequencies with those of the sandwich plate with honeycomb core in ANSYS. The approach is to replace the honeycomb core with an equivalent solid. Therefore the important task is to obtain reliable effective material parameters, including effective density, effective Young's modulus and effective shear modulus of the honeycomb core.

For a honeycomb core in the shape of general hexagon, as shown in figure 2.8, the effective bending lengths of the cell wall are denoted by l_b and h_b. They are expressed as

$$l_b = l - t/\left(2\cos\theta\right) \tag{2.15}$$

$$h_b = h - t\left(1 - \sin\theta\right)/\cos\theta \tag{2.16}$$

The effective density ρ_1^* for a unit cell highlighted in figure 2.9 can be determined by dividing the mass of the material in an unit cell by the total volume of the material and the void part, given the material density and the volume of the cell and the

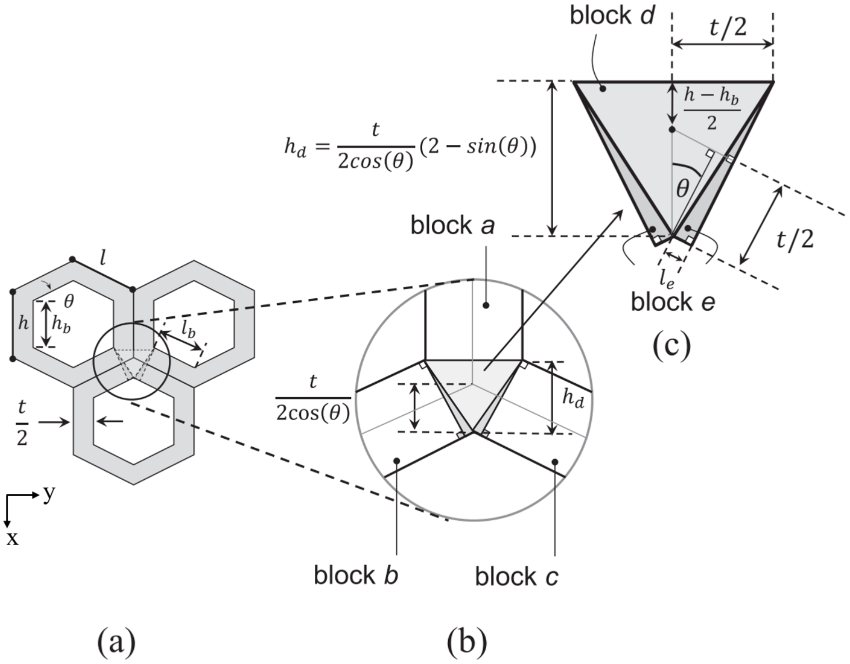


Figure 2.8: Schematic representation of periodic hexagonal honeycombs with uniform wall thickness. (a) honeycomb cells, (b) geometrical dimensions and (c) building blocks of the node region [108]

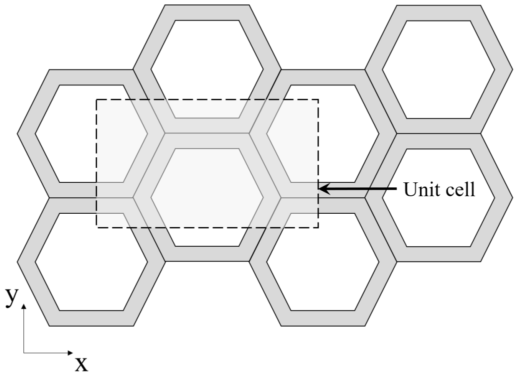


Figure 2.9: Periodic honeycomb cells in the shape of hexagon

void:

$$\frac{\rho_1^*}{\rho_1} = 1 - \frac{A_{void}}{A_{cell}} = 1 - \frac{l_b\left(l_b + l_b \sin\theta\right)}{l\left(l + l\sin\theta\right)} \tag{2.17}$$

The out-of-plane Young's modulus in the z direction can be determined by assuming that the honeycomb deformation is axial under normal loading in the z direction. Hence, the longitudinal modulus can be expressed as:

$$E_z^* = E_1 \left(\frac{\rho^*}{\rho_1} \right) \tag{2.18}$$

The effective Young's modulus in x direction and y direction as well as the effective shear moduli in in different directions are given by [108], expressed as:

$$E_x^* = E_1 \left(\frac{t}{l_b} \right)^3 \frac{\cos\theta}{(h/l + \sin\theta)\sin^2\theta} \left[\frac{1}{1 + (2.4 + 1.5\nu_1 + \cot^2\theta)\left(\frac{t}{l_b}\right)^2} \right] \tag{2.19a}$$

$$E_y^* = E_1 \left(\frac{t}{l_b} \right)^3 \frac{(h/l + \sin\theta)}{\cos^3\theta} \left[\frac{1}{1 + \left(2.4 + 1.5\nu_1 + \tan^2\theta + \frac{2(h_b/l_b)}{\cos^2\theta}\right)\left(\frac{t}{l_b}\right)^2} \right] \tag{2.19b}$$

$$G_{xy}^* = E_1 \left(\frac{t}{l_b} \right)^3 \frac{(h/l + \sin\theta)}{\left(\frac{h_b}{l_b}\right)^2 \cos\theta} \left(\frac{1}{C} \right) \tag{2.19c}$$

$$\begin{aligned} G_{xz}^* =& G_1 \left(\frac{t/l}{(h/l + \sin\theta)\cos\theta} \right) \\ & \left[\cos^2\theta \left(\frac{l_b}{l} \right) + \frac{3}{4} \left(\frac{t}{l} \right) \tan\theta - \frac{\cos\theta}{2} \left(\frac{t}{l} \right) (2\sin\theta - 1) \right] \end{aligned} \tag{2.19d}$$

$$\begin{aligned} G_{yz}^* =& G_1 \left(\frac{t/l}{(h/l + \sin\theta)\cos\theta} \right) \\ & \left[\sin^2\theta \left(\frac{l_b}{l} \right) + \frac{h_b}{2l} + \frac{3}{4} \left(\frac{t}{l} \right) \tan\theta - \frac{\sin^2\theta}{2\cos\theta} \left(\frac{t}{l} \right) (2\sin\theta - 1) \right] \end{aligned} \tag{2.19e}$$

The term C in (2.19c)is expressed as

$$\begin{aligned} C =& 1 + 2\frac{h_b}{l_b} + \left(\frac{t}{l_b} \right)^2 \left[\frac{(2.4 + 1.5 * \nu_1)}{h_b/l_b} \left(2 + \frac{h}{l} + \sin\theta \right) \right. \\ & \left. + \frac{(h/l + \sin\theta)}{(h_b/l_b)^2} \left[(h/l + \sin\theta)\tan^2\theta + \sin\theta \right] \right] \end{aligned} \tag{2.20}$$

As the core of the structure studied in the present work is a regular hexagonal honeycomb, we have $l = h$, $l_b = h_b$, $\theta = \frac{\pi}{6}$. Substituting the properties of material described in section 2.2 in the above equations, the effective parameters can be calculated and are listed in table 2.2.

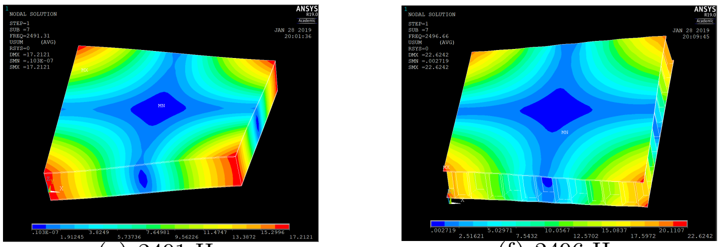
Table 2.2: Effective characteristics of the honeycomb core

Parameter	E_x^*	E_y^*	E_z^*
Value	8.63×10^5 Pa	8.63×10^5 Pa	2.51×10^8 Pa
Parameter	G_{xy}^*	G_{xz}^*	G_{yz}^*
Value	2.1658×10^5 Pa	4.775×10^7 Pa	4.775×10^7 Pa

A modal analysis is performed in ANSYS for the honeycomb sandwich plate and the homogenized sandwich plate, respectively. The first 5 natural frequencies and corresponding mode shapes of each model are extracted to verify the reliability of the theoretical formulas mentioned previously, as shown in figure 2.10. The first five natural frequencies and mode shapes of the homogenized model are on the left in the figure while the other five on the right are for the honeycomb plate model. The first 5 natural frequencies for the two models and the percentage errors between them are listed in table 2.3. The results show that the errors are below 3.4% for the first five natural frequencies. As the frequency increases, the error tends to increase. It's an acceptable accuracy for the homogenized model, which demonstrates that the theoretical formulas to derive effective characteristics are reliable.

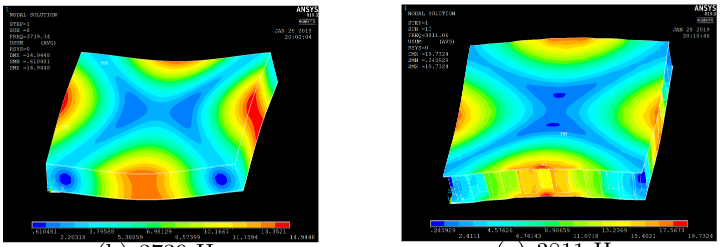
Table 2.3: Natural frequencies of the honeycomb sandwich plate and the homogenized sandwich plate

Order	Homogenized (Hz)	Honeycomb (Hz)	Error (%)
1	2491	2496	-0.2%
2	3739	3811	-1.9%
3	4313	4332	-0.4%
4	5043	4919	2.5%
5	5166	4994	3.4%



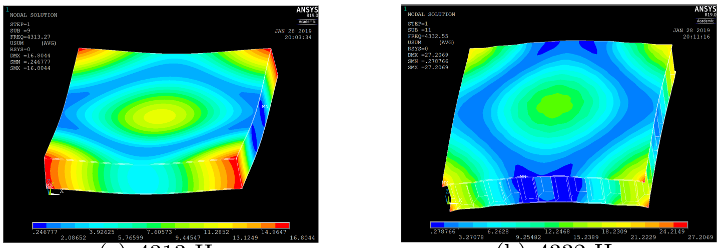
(a) 2491 Hz

(f) 2496 Hz



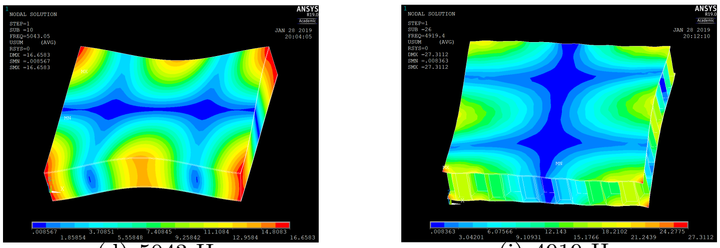
(b) 3739 Hz

(g) 3811 Hz



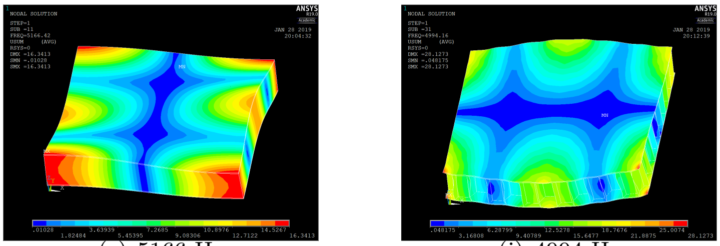
(c) 4313 Hz

(h) 4332 Hz



(d) 5043 Hz

(i) 4919 Hz



(e) 5166 Hz

(j) 4994 Hz

Figure 2.10: The first five natural frequencies with corresponding mode shapes of (a-e) a homogenized sandwich plate and those of (f-j) a honeycomb sandwich plate

2.4 Damage

Composite materials exposed to long term loads and critical working environment may have damages due to fatigue, impact or material degradation. Cracks and delamination are two most common damages for composites. In the first attempt of this study, crack damage is designed in the sandwich structure and investigation of the capability of detecting this kind of damage is conducted. After the validation of the composite sandwich plate model without damage, models with cracks in different positions are constructed. Five crack damages in the x direction and one crack damage in y direction are created. Cracks in x direction are located at ($x = \frac{L}{4}$, $y = \frac{W}{4}$), ($x = \frac{L}{4}$, $y = \frac{W}{2}$), ($x = \frac{L}{2}$, $y = \frac{W}{4}$), ($x = \frac{L}{2}$, $y = \frac{W}{2}$), ($x = \frac{3L}{4}$, $y = \frac{3W}{4}$), respectively. Crack in y direction is located at ($x = \frac{L}{2}$, $y = \frac{W}{2}$). The length of cracks is 30mm which is $\frac{L}{10}$ in x direction, and crack length in y direction is 43mm, as illustrated in figure 2.11. Cracks are designed through the core and the top face sheet so that the interaction of the propagated signal with the damage is detectable. Cracks are created by disconnecting local nodes.

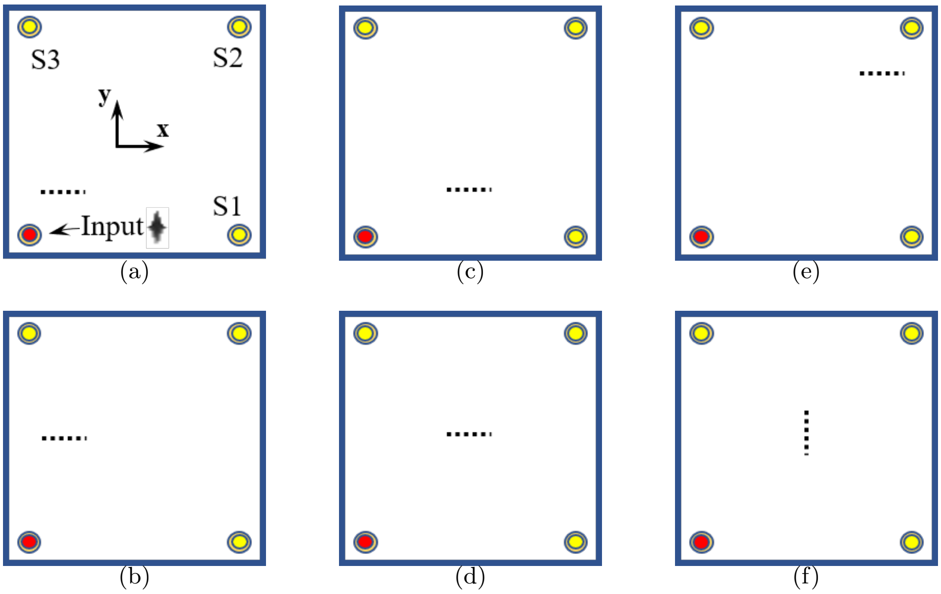


Figure 2.11: Composite sandwich models with crack damage in different positions: (a) in x direction $x = \frac{L}{4}$, $y = \frac{W}{4}$ (b) in x direction $x = \frac{L}{4}$, $y = \frac{W}{2}$ (c) in x direction $x = \frac{L}{2}$, $y = \frac{W}{4}$ (d) in x direction $x = \frac{L}{2}$, $y = \frac{W}{2}$ (e) in x direction $x = \frac{3L}{4}$, $y = \frac{3W}{4}$ (f) in y direction $x = \frac{L}{2}$, $y = \frac{W}{2}$. The setup of the model consists of 1 actuator (red) and 3 sensors (yellow) attached on the face-sheet

For each model, actuator and sensors are attached on the top face sheet. They are located in four corners. One sensor is used as actuator through which the pulse excitation signal is inserted, while the other three sensors are used to collect propagated signals through the structure, as shown in figure 2.11(a). In actual cases, the actuator converts the input voltage signal into a displacement signal, while the sensor converts the received displacement signal into a voltage signal. Thus in the current study, for the sake of convenience, we directly input the displacement signal on the structure and extract the displacement signal at the corresponding position to represent the work of actuator and sensors. The vertical displacement of nodes are collected for further signal processing.

2.5 Simulation

In the simulation of each model, a pulse excitation signal with 7 cycles is applied on the actuator to stimulate vertical displacement. Transient analysis is performed on the sandwich plate structure. Damping coefficients are set as follows: matrix multiplier is set as $\alpha = 0$ and stiffness matrix multiplier as $\beta = 6.37 \times 10^{-7}$. The calculation time is set as 0.01s, which is long enough for the wave to propagate through the whole plate. The sampling frequency f_s should be no less than 2 times the signal frequency f according to the Nyquist's rule. In the present research the sampling frequency is set as $f_s = 8f$ in order to make the recorded signal be representative. In each simulation case, vibration responses of one structure are collected at three sensors for further use. They are saved in one file to represent the structural health status in that condition. Same simulations are performed for the same structure with input signals of different frequencies. Then the same simulations are conducted for structures with different health status successively. As mentioned above, we have 7 composite sandwich plate models among which one healthy and six with crack damages at different positions. On each model 3 different input signals with frequency 8kHz, 6kHz and 4kHz are applied successively corresponding to 3 simulation cases. So far a total of 21 simulation cases are obtained. The objective is to perform the least cases with which machine learning approaches can detect the structure's health status and determine probable damaged regions. The information of the computer used for simulation is as follows: System Windows 10, Processor Intel Core i7-7700 CPU, 16G RAM. The computation time for each case is around 2.5 hours and 3.5 hours depending on different frequencies.

A comparison of the vibration responses of a healthy structure and the structure in figure 2.11(b) to a same input signal of 8kHz with 7 cycles at the same position is firstly conducted, as shown in figure 2.12. The compared signals are from three sensors on the structure together with the input signal. First of all, in both structures it can be seen that the original input excitation signal waveform has changed after propagating along different directions by comparing the signals in figure 2.12 (b-d) with figure 2.12 (a). This is due to the wave dispersion in the structure. Besides, as the core of the structure is honeycomb that has anisotropic properties, the dispersive behavior of the structure is direction-dependant, which results in the differences in the vibration responses from 3 sensors in figure 2.12 (b) to figure 2.12 (d). Moreover, comparing the vibration responses of healthy and damaged structures from sensor 3, we can find that in the first wave packet, the amplitude of the signal from damaged structure is slightly higher than that from healthy structure, as shown in 2.12 (d). It is due to the fact that the existence of the crack damage in this position reduces the bending stiffness of the structure in y direction from the actuator to sensor 3. Furthermore, comparing the healthy and damaged structures in 2.12 (b) and (d), the signal difference occurring between wave packets might be caused by the interaction of input signal with the crack, but as the difference is not evident, machine learning algorithm should be used to learn such kind of features.

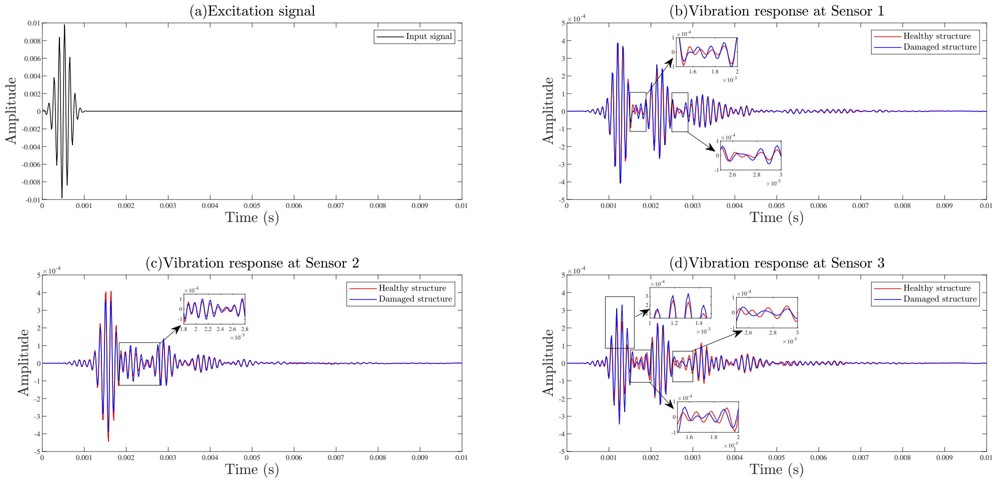


Figure 2.12: Comparison of structural vibration responses of a healthy model and a model with crack damage to an excitation signal of frequency 8 kHz. (a) Excitation signal, (b) Vibration response at Sensor 1, (c) Vibration response at Sensor 2, (d) Vibration response at Sensor 3

The DWT coefficients are then compared in figure 2.13. As mentioned in 2.6.4, DWT can extract information in both time and frequency domain. The mismatch between the vibration responses of healthy and damaged structures can be extracted, as shown in figure 2.13 (b) and (d). It should be noted that only the first 256 coefficients, which correspond to lower frequencies of the analysis, carry relevant information and the rest of the signal can be discarded without any major loss of information. In addition, discarding half of the signal can speed up the computation in further step.

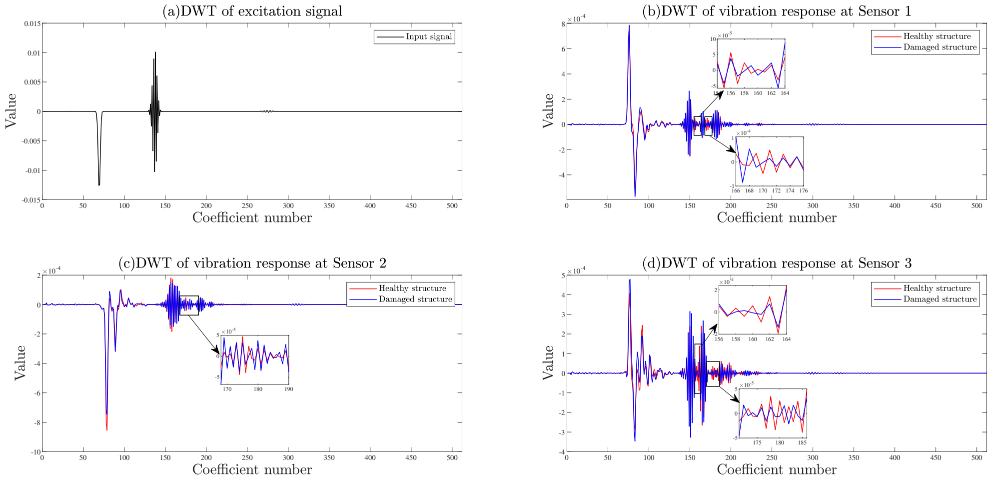


Figure 2.13: Comparison of DWT coefficients of the vibration responses in Figure 2.12. (a) DWT of excitation signal, (b) DWT of vibration response at Sensor 1, (c) DWT of vibration response at Sensor 2, (d) DWT of vibration response at Sensor 3

2.6 Features extraction techniques through FE simulation

In the current research, conventional machine learning method Gaussian Process and deep learning method Convolutional Neural Networks will be applied successively. Before introducing the machine learning methods, a signal processing and feature extraction method is firstly introduced.

There exist various signal processing methods, such as one of the most commonly used Fourier Transform (FT), Short Time Fourier Transform (STFT), wavelet transform (WT) methods like Continuous Wavelet Transform (CWT) as well as Discrete Wavelet Transform (DWT). These processing methods are often used to extract frequency information of a time-domain signal, because some distinguished information is not visible in the time domain, but can be seen in the frequency domain, such as graphical recording of heart's electrical activity in medicine, where symptoms of a pathological condition are shown in the frequency domain. A frequency spectrum of a signal is the frequency components of the signal. In fact there are many other transforms that are used in mathematical and physical problems except these four mentioned methods. But for a better understanding, the principle of each method mentioned above will be introduced in this section, along with the advantages and disadvantages of each method.

2.6.1 Fourier transform

Fourier Transform (FT) is widely used due to its convenience in implementation and its capability of extracting frequency information of a time-series signal. The FT decomposes a time-domain signal f into its constituent frequencies, which is achieved by:

$$\widehat{f}(\xi) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i t \xi} dt \tag{2.21}$$

for any real number ξ, where $\widehat{f}$ indicates the FT of the function f at hand, t stands for time and ξ stands for frequency. It's worthy of note that the signal $f(t)$ is multiplied by an exponential term with some certain frequency ξ, and then integrated over all time instances. If the result of the integration is relatively large, then it means that the frequency ξ is a dominant spectral component in the signal $f(t)$. If the integration result is small or equal to zero, then it means that the signal $f(t)$ does not have a frequency component of ξ.

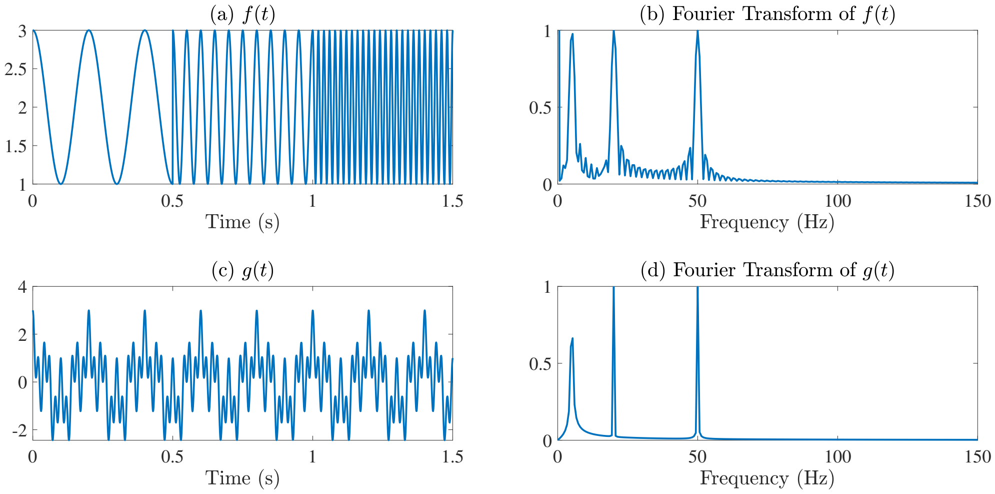


Figure 2.14: An example of Fourier Transform. (a) function $f(t)$ containing frequency components 5Hz, 20Hz and 50Hz in different time intervals, (b) Fourier Transform of $f(t)$, (c) function $g(t)$ containing frequency components 5Hz, 20Hz and 50Hz in all time instances, (d) Fourier Transform of $g(t)$.

The information provided by FT corresponds to all time instances. In other words, regardless of the frequency component present in this or that time interval, the FT provides the same result. Thus, FT is not suitable if the signal is non-stationary, which means frequency components are time dependence. FT is only suitable for stationary signals, which is usually not the case, because it can only determine which frequency component exists in a signal. An example is illustrated in figure 2.14. In the sub-figure (a), the function $f(t)$ contains frequency components of 5Hz, 20Hz and 50Hz. The component of 5Hz occurs in the first 1/3 time interval, 20Hz occurs in the second 1/3 time interval and 50Hz occurs in the last 1/3 time interval. The corresponding FT is illustrated in sub-figure (b) where the three peaks correspond to the three different frequencies. The existence of the small noise-like peaks between those three major peaks are due to the sudden change of frequencies. In sub-figure (c), the function $g(t)$ contains the same frequency components as $f(t)$, but in the entire time domain. Sub-figure (d) is the corresponding FT and the three peaks correspond to the three different frequencies. It is evident that the two functions $f(t)$ and $g(t)$ are totally different in time domain, but their FT results are the same, which demonstrates the limitations of FT in processing non-stationary signals.

2.6.2 Short Time Fourier Transform

Short Time Fourier Transform (STFT) is then proposed to obtain both time and frequency domain information by involving a window function. As FT is not suitable for non-stationary signals, it is assumed that some portion of a non-stationary signal is stationary. The signal can be divided into several segments with equal length by a window function w. It's assumed that the window is narrow enough that each segment is a stationary signal, thus FT can be performed in each segment and frequency information can be then extracted. STFT is achieved by:

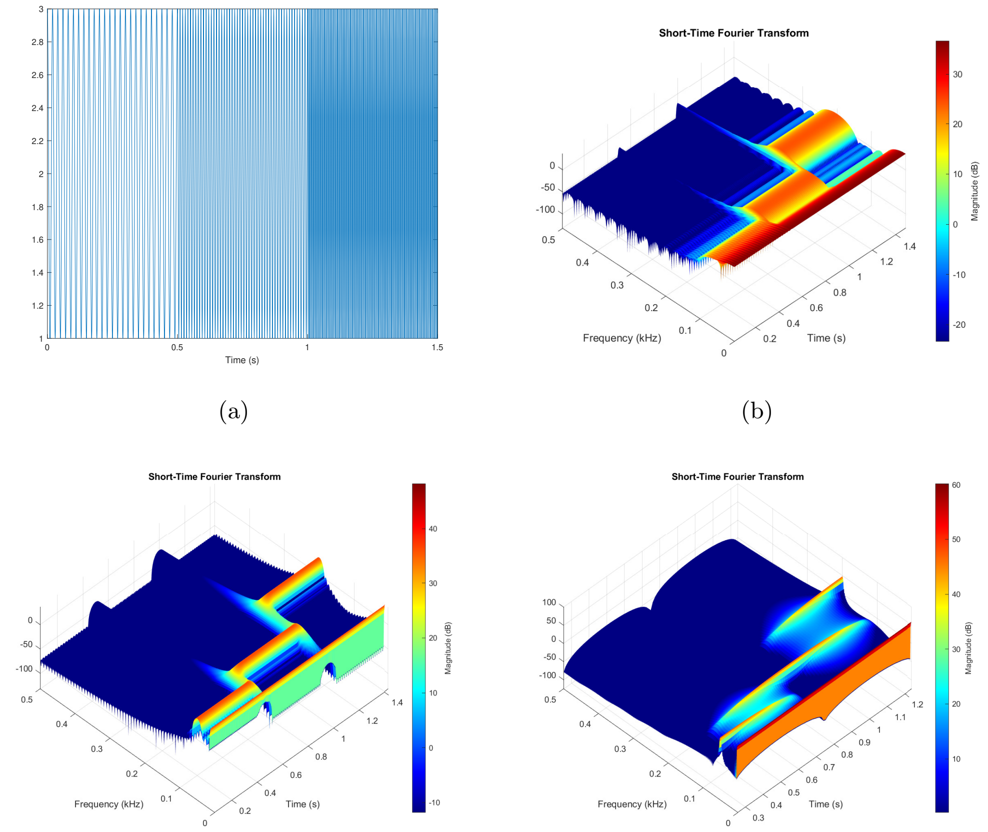
$$\mathbf{STFT}\{f(t)\}(\tau,\xi)=\int_{-\infty}^{\infty} f(t)w(t-\tau)e^{-i2\pi\xi t}dt \tag{2.22}$$

where $f(t)$ is the function at hand, $w(t)$ is the window function. It can be seen that STFT of a signal is nothing but the FT of the signal multiplied by a window function. The window function can be a Gaussian function in the form:

$$w(t)=e^{-a\frac{t^2}{2}} \tag{2.23}$$

where the term a determines the width of the window. It should be noted that the width of the window is inversely proportional to the value of a.

The window function is firstly located at $t=0$ of the signal $f(t)$. Suppose the width of the window function is '**T**' second, then the window will overlap with the first $\mathbf{T/2}$ seconds. The product of the window function and the first $\mathbf{T/2}$ seconds of the signal is chosen to take the FT. Then the result is the FT of the first $\mathbf{T/2}$ seconds of the signal. The next step is to shift the window by a step of τ shown in Eq.(2.22) to a new location. This procedure is repeated until the window overlaps the end of the signal. Thus, the STFT of the signal at hand is accomplished step by step. However, the implicit problem of STFT is that the window width is constant for all segments. Hence, the Heisenberg uncertainty principle becomes a problem for STFT. In other words, one can know in which time intervals certain frequency bands exist, but one cannot know what frequency components exist at what time instances, which is a resolution problem. More precisely, if the window is too wide, a good frequency resolution and poor time resolution will be obtained, and in an extreme case, if the width of the window is infinite, FT will be obtained; otherwise if it is too narrow, a poor frequency resolution and a good time resolution will be obtained.



(a) (b)

(c) (d)

Figure 2.15: Short Time Fourier Transform of a time domain signal containing three frequencies. (a) Signal containing 50Hz, 100Hz and 200Hz in three successive intervals, (b)STFT with a window of length 64, (c)STFT with a window of length 256, (d)STFT with a window of length 1024

In order to interpret the issue more intuitively, an example of STFT using three windows with different lengths is taken as illustrated in figure 2.15. Figure 2.15(a) is a time domain signal containing frequency 50 Hz in the first 1/3 time interval, frequency 100 Hz in the second 1/3 time interval and 200 Hz in the last 1/3 interval. STFT is conducted on the time domain signal using windows of three different widths, 64, 256, 1024, respectively. Corresponding results are illustrated from a top corner view for better interpretation in figure 2.15(b), figure 2.15(c) and figure 2.15(d), respectively. With a narrow window, the STFT has a good time resolution as the intervals in which different frequencies exist are well separated and correspond to the original signal time intervals, but in frequency domain, each peak covers a range of frequencies instead of a single frequency, as shown in figure 2.15(b). As the width of the window increases from 64 to 256, the three peaks are not well separated in time domain compared with the previous one, but the resolution in frequency domain gets better, as shown in figure 2.15(c). With a wider window whose width is 1024, a very good resolution in frequency domain can be obtained as shown in figure 2.15(d), but in time domain the time intervals covered by the peak overlap each other. Therefore, the task of STFT will be a result of choosing an appropriate window function.

2.6.3 Continuous Wavelet Transform

Multi-resolution Analysis (MRA) is an alternative approach to analyze signals, allowing to analyze signals with different resolutions at different frequency bands, which differs from STFT. Continuous Wavelet Transform (CWT) is widely used as a MRA method. It is defined by:

$$CWT\{f(t)\}(\tau, s) = \frac{1}{\sqrt{|s|}} \int f(t)\psi^*(\frac{t-\tau}{s})dt \tag{2.24}$$

where $f(t)$ is the signal at hand, $\psi(t)$ is the mother wavelet, τ and s are the translation and scale, respectively, which are related to time and frequency. Mother wavelet means that wavelets used in different regions of the signal are derived from this prototype wavelet, and wavelet means the wave is of finite length, similar to the window function in STFT. The difference is that the length of the wavelet will change depending on the location of the wavelet in the signal.

There are various options for mother wavelet, among which some commonly used are Morlet wavelet, Daubechies wavelet, Coiflet wavelet, Symlets wavelet, etc. The scale s is proportional to the reciprocal of frequency. In other words, low scale means high frequency, while high scale means low frequency. The term translation τ is the same as in STFT, it is used to shift the wavelet to next location. It should be noted that in CWT, there are two variables: scale and translation, which are used to scan the time and frequency domains of the signal $f(t)$.

CWT of a signal is computed as the following: the wavelet is firstly located at the beginning of the signal at $t = 0$ with the scale $\tau = 1$. The wavelet is multiplied by the signal and integrated over the whole time field. The final result corresponds to the CWT at $t = 0$ and $s = 1$ in the time-scale plane. Then the wavelet is shifted to the next location by a step of τ. The same procedure is performed and repeated until the wavelet overlaps the whole signal in time field for $s = 1$. One row of the time-scale plane is completed. Then the scale s is increased by a sufficiently small step size so that the computation seems "continuous", and the wavelet locates again at $t = 0$. The CWT is computed for the new scale by shifting the wavelet towards the right. This procedure is repeated for each scale until the entire time-scale plane is completed.

It is worth mentioning that the computation of CWT of a signal for a certain scale representing a certain frequency, is a comparison of similarity between the wavelet and the signal in frequency field. More precisely, if the frequency corresponding to the scale in the wavelet is a major component in the signal $f(t)$, then the result of CWT will be relatively large, otherwise it will be small or zero. This is quite similar to FT and STFT. However, as the transform is continuous, wavelet with different frequency contents will be shifted for the entire time-domain signal. Thus, the information provided by CWT will be highly redundant, which will also be computation time-consuming.

2.6.4 Discrete Wavelet Transform

DWT was proposed, on one hand, to extract information in both time and frequency domain, on the other hand to reduce the dimension of data, which resolves the problem of redundancy in CWT. Since the time domain signal is a discrete time sequence, we denote the sequence as $x[n]$ with n an integer. The transform is computed by passing the signal through a half band digital lowpass filter with impulse response $h[n]$ and a half band highpass filter with impulse response $g[n]$ simultaneously.

The filtering, from a mathematical point of view, is convolution of the signal with the filter. A half band lowpass filter removes all frequencies above half of the highest frequency in the signal while a half band highpass filter removes all frequencies below half of the highest frequency. After passing the signal through filters, detail coefficients from the output of highpass filter and approximation coefficients from the output of lowpass filter are obtained. Since half the frequencies of the signal have been removed, half the samples can be discarded according to Nyquist's theorem. The output of lowpass filter is subsampled by 2. This constitutes one decomposition level. This decomposition has halved the time resolution since the number of samples has been halved, but has doubled the frequency resolution since the frequency band has been halved. The subsampled output of lowpass filter is further processed by passing it again through a new lowpass filter $h[n]$ and a highpass filter $g[n]$ constituting another decomposition level. It should be noted that the highpass and lowpass filters are known as the Quadrature Mirror Filters (QMF) and are related by:

$$g[L-n-1] = (-1)^n \times h[n] \tag{2.25}$$

where L is the filter length. This whole procedure is illustrated in figure 2.16.

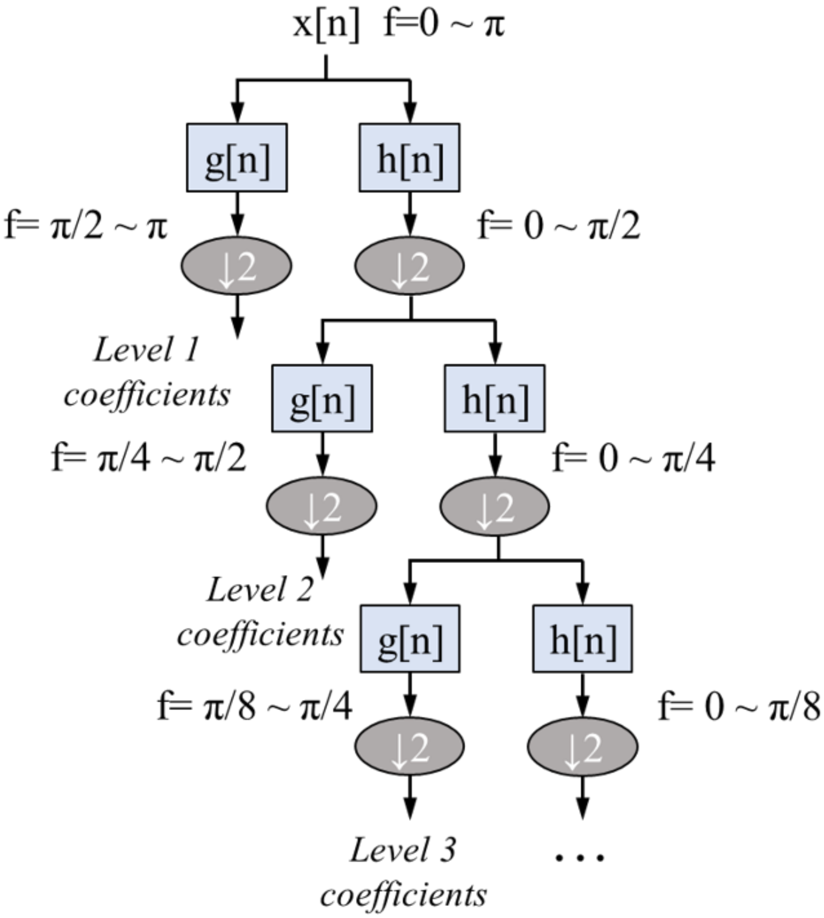


Figure 2.16: The DWT algorithm

The selection of mother wavelet of DWT has an important impact on the results because the highpass and lowpass filters are determined by the mother wavelet. Then the filters will affect the result of the transform, in turn affect the final results. Basically there are two approaches to select an appropriate mother wavelet for different signal analysis purposes: qualitative approach and quantitative approach [109]. The former is usually based on visual inspection of the similarity between the mother wavelet and the signal. The latter involves several methods such as Maximum Relative Wavelet Energy (RWE) criterion, Energy to Shannon entropy ratio criterion, which should be studied in detail.

Chapter 3

Gaussian Process-based structural health monitoring

Contents

3.1 Introduction

Supervised machine learning has been widely used to learn a function that maps an input to an output based on example input-output pairs [110]. As one sub-field of machine learning, it helps to improve the efficiency and reduce error in problem solving. It can be divided into regression and classification problems. The outputs for regression are continuous values whereas for classification are discrete class labels. The idea of machine learning-based SHM is to learn the relations between input variables and output variables. There are several Gaussian process (GP) models for regression problems, but in this study, a GP model for classification is employed.

A Gaussian Process for classification can be considered as a Gaussian distribution over functions rather than over variables, and inference takes place directly in the space of functions. The machine learning algorithm involving GP takes a measure of the similarity between points to predict the value for an unseen point from the training data.

3.1.1 Gaussian process regression

Introduction of a GP model for regression problems is firstly carried out because it is a necessary step for understanding GP model for classification problems. The simplest regression model is a linear regression which involves the linear combination of input variables

$$y = w_0 + w_1 x_1 + ... + w_n x_n = w_0 + \sum_{i=1}^{n} w_i x_i \tag{3.1}$$

where $\mathbf{x} = (x_1, ..., x_n)^{\mathrm{T}}$ is the input vector and $w_1, ..., w_n$ are the corresponding coefficients, also called weights. The parameter w_0 is a bias parameter. This function is a linear function of the coefficients w_i, as well as a linear function of the input variables x_i. The application of this linear regression model is too limited. Thus, an extension of the model is conducted by considering a linear combination of nonlinear functions of the input variables in the form of

$$y = w_0 + \sum_{i=1}^{n} w_i \phi_i(\mathbf{x}) \tag{3.2}$$

where ϕ_i is a basis function. It is often practical to define another function $\phi_0(\mathbf{x}) = \mathbf{1}$ as an additional 'basis function' so that Eq. (3.2) can be simplified as

$$y = \sum_{i=0}^{n} w_i \phi_i(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}) \tag{3.3}$$

where $\phi = (\phi_0(\mathbf{x}), ..., \phi_n(\mathbf{x}))^{\mathrm{T}}$ and $\mathbf{w} = (w_0, ..., w_n)^{\mathrm{T}}$. By introducing nonlinear basis functions, the output y becomes a nonlinear function of the input vector $\mathbf{x}$, which is commonly encountered in solving engineering problems. Now consider a Gaussian prior distribution over the weights $\mathbf{w}$ with zero mean and covariance matrix $\alpha^{-1}\mathbf{I}$ in the form of

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-\mathbf{1}}\mathbf{I}) \tag{3.4}$$

where α is hyperparameter representing the inverse variance of the distribution that governs the distribution and $\mathbf{I}$ is a identity matrix. For a particular value of weights $\mathbf{w}$, Eq. (3.3) defines a particular function of x. Thus, from the Gaussian distribution over $\mathbf{w}$ expressed in Eq. (3.4), one can induce a probability distribution over the function $y(\mathbf{x})$. Practically, it is expected to evaluate this function at specific data points of $\mathbf{x}$ so that we can obtain the joint probability distribution of $y(\mathbf{x}_1), ..., y(\mathbf{x}_N)$. We define a vector $\mathbf{y} = (y_1, ..., y_N)$ where each element $y_i = y(\mathbf{x}_i)$ for $i = 1, ..., N$.

Similar to Eq. (3.3), then this vector can be rewritten in matrix notation as

$$\mathbf{y} = \mathbf{\Phi}\mathbf{w} \tag{3.5}$$

where $\mathbf{\Phi}$ is a matrix whose elements are $\Phi_{ij} = \phi_j(\mathbf{x}_i)$. Note that $\mathbf{y}$ is a linear combination of $\mathbf{w}$ whose probability distribution is Gaussian, as expressed in Eq. (3.4). Thus, $\mathbf{y}$ is also a Gaussian distribution. Based on Eq. (3.4) and Eq. (3.5), the mean and covariance of $\mathbf{y}$ can be derived as:

$$\mathrm{E}[\mathbf{y}] = \mathbf{\Phi}\mathrm{E}[\mathbf{w}] = \mathbf{0} \tag{3.6}$$

$$cov[\mathbf{y}] = \mathrm{E}[\mathbf{y}\mathbf{y}^{\mathrm{T}}] = \mathbf{\Phi}\mathrm{E}[\mathbf{w}\mathbf{w}^{\mathrm{T}}]\mathbf{\Phi}^{\mathrm{T}} = \frac{1}{\alpha}\mathbf{\Phi}\mathbf{\Phi}^{\mathrm{T}} = \mathbf{K} \tag{3.7}$$

where $\mathbf{K}$ is the Gram matrix whose elements are in the form of

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{\alpha}\phi(\mathbf{x}_i)^{\mathrm{T}}\phi(\mathbf{x}_j) \tag{3.8}$$

here $k(\mathbf{x}_i, \mathbf{x}_j)$ is the kernel function.

Similarly, for a Gaussian process which is defined as a probability distribution directly over functions, given an input vector variable $\mathbf{x}$ with dimension D, the corresponding output target t is related with the input vector by a nonlinear smooth mapping function f with an additional Gaussian noise ε

$$t = f(\mathbf{x}) + \varepsilon \tag{3.9}$$

In the same way, for a given input training data set $\mathbf{D} = \{\mathbf{X}, \mathbf{t}\}$ containing input training matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]^T$ constituted by vectors $\mathbf{x}_i$ and corresponding training target vector $\mathbf{t}$, in which each element is expressed as $t_i = f(\mathbf{x}_i) + \varepsilon$, $i = 1, 2, ..., N$, we are interested in making inferences about the relationship between inputs and targets as well as making predictions for a new input $\mathbf{x}_{i+1}$. Therefore, GP is involved by modeling the mapping function f with a zero mean and covariance matrix $\mathbf{K}$, see Eq. (3.10):

$$p(\mathbf{f}|\mathbf{D}) = \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K}) \tag{3.10}$$

where $\mathbf{f} = [f(\mathbf{x}_1), f(\mathbf{x}_2), ..., f(\mathbf{x}_N)]^T$, $\mathbf{K}$ is the covariance matrix, computed by covariance function $k(\mathbf{x}_i, \mathbf{x}_j)$, also called kernel function, expressed by Eq. (3.11). In the present study, a covariance function is chosen to express the property that for similar points $\mathbf{x_i}$ and $\mathbf{x_j}$, the corresponding target values t_i and t_j will be more strongly correlated than for dissimilar points. Therefore, the squared exponential with automatic relevance determination distance (SE-ARD) measure expressed by

Eq. (3.11) is adopted:

$$k(\mathbf{x}_i, \mathbf{x}_j) = cov(f(\mathbf{x}_i), f(\mathbf{x}_j)) = \sigma_f^2 exp(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M}(\mathbf{x}_i - \mathbf{x}_j)) \tag{3.11}$$

where $\mathbf{M} = diag(\lambda_1, \lambda_2, ..., \lambda_N)$, with each λ_i corresponding to input dimension characteristic length scale. σ_f^2 is the signal variance. λ and σ_f^2 are hyperparameters. In the training step, the objective is to model an appropriate mapping function f so that the training target t_i corresponds well to the training input vector $\mathbf{x_i}$. The learning task is achieved by tuning the hyperparameters denoted by Θ in the covariance function. And the optimization is accomplished by minimizing the negative logarithmic likelihood function with respect to the hyperparameters in Eq. (3.12), which corresponds to choosing the value of Θ for which the probability of the observed data set is maximized.

$$L\left(\Theta\right) = -\log\left(p\left(\mathbf{t}|\mathbf{X}\right)\right) = \frac{1}{2}\log|\mathbf{K}| + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{K}^{-1}\mathbf{t} + \frac{N}{2}\log\left(2\pi\right) \tag{3.12}$$

where $|\mathbf{K}|$ refers to the determinant of the matrix $\mathbf{K}$.

The role of the model is to use known data $\mathbf{X}$ and $\mathbf{t} = [t_1, t_2, ..., t_N]$ to predict the output target of a new input vector $\mathbf{x}_{N+1}$. As the covariance matrix $\mathbf{K}$ is composed of the covariance between each of the first N input vectors, for a new input vector $\mathbf{x}_{N+1}$, the joint probability distribution can be expressed as:

$$p(\mathbf{t}_{N+1}) = \mathcal{N}(\mathbf{t}_{N+1}|\mathbf{0}, \mathbf{K_{N+1}}) \tag{3.13}$$

where $\mathbf{t}_{N+1} = [t_1, t_2, ..., t_{N+1}]^{\mathrm{T}}$, and the covariance matrix $\mathbf{K}_{N+1}$ of dimension $(N+1) \times (N+1)$ is expressed as:

$$\mathbf{K}_{N+1} = \begin{bmatrix} \mathbf{K} + \sigma_{\mathbf{n}}^{\mathbf{2}}\mathbf{I} & \mathbf{k} \\ \mathbf{k}^{\mathrm{T}} & c \end{bmatrix} \tag{3.14}$$

with $\mathbf{k}$ a column vector with elements $k(\mathbf{x}_n, \mathbf{x}_{N+1})$ for $n = 1, 2, ..., N$, and c a scalar with value $c = k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) + \sigma_n^2$. It's clear to see that the vector $\mathbf{k}$ and the scalar c are both dependant on the test point input vector $\mathbf{x}_{N+1}$. The predictive conditional distribution over t_{N+1} is a Gaussian distribution with mean and covariance given by [111]:

$$\begin{aligned} \mu\left(\mathbf{x}_{N+1}\right) &= \mathbf{k}^{\mathrm{T}}\mathbf{K}\mathbf{t}, \\ \sigma^2\left(\mathbf{x}_{N+1}\right) &= c - \mathbf{k}^{\mathrm{T}}\mathbf{K}\mathbf{k} \end{aligned} \tag{3.15}$$

These two items are the most important for the regression because they provide both prediction mean value and uncertainty information. In a GP regression example, see figure 3.1, the regression problem is resolved by making a prediction of the new target for a new input with a predictive mean value and uncertainty information indicated by a confidence interval based on known data. In the figure, points denoted by (+) are training data, the blue line is the prediction generated by Gaussian process and the grey region represents the confidence interval (mean value ± two standard deviations: $\mu \pm 2\sigma$). In general, the more detailed information we know, the more reliable the prediction will be, which means the more sparse the data, the larger the uncertainty interval of the prediction. We can see that the lack of data in the interval $x \leq -1.5$ and $0.6 \leq x \leq 1.8$ leads to more uncertain predictions in these regions while in other regions the prediction generated by Gaussian process is more reliable.

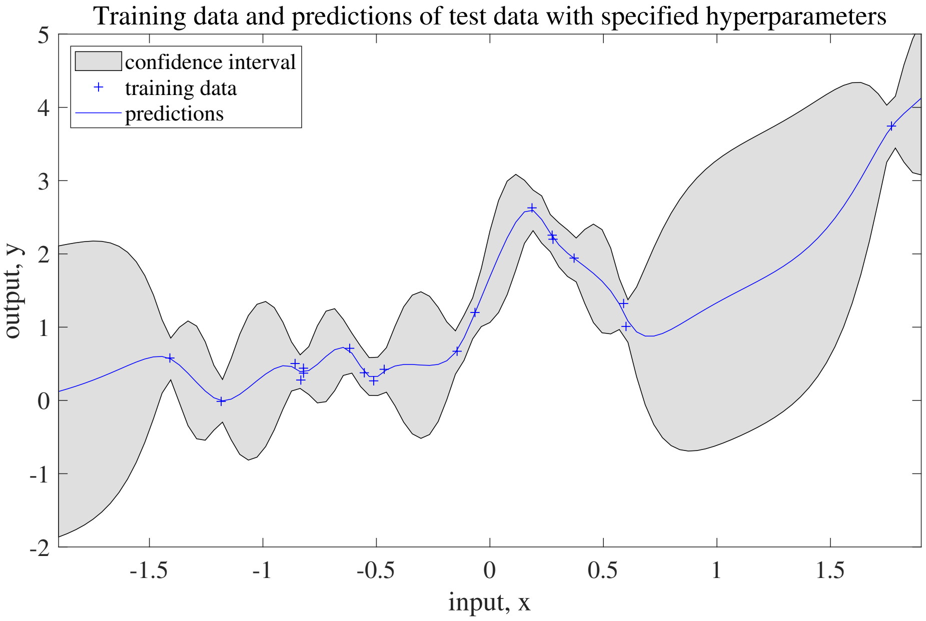


Figure 3.1: An example of GP regression. + is training data, blue line — is predictions using GP model based on training data, and the grey region is the confidence interval (mean value ± two standard deviations)

3.1.2 Gaussian process classification

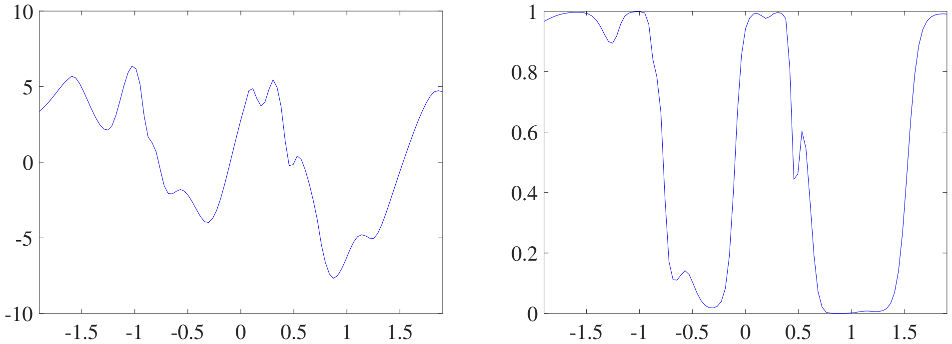
Different from Gaussian process regression, outputs of classification are discrete class labels. For a classification problem, our goal is to model the posterior probability distribution of the target for a new input vector, given a set of training data. The predictive probability should be in the interval [0, 1], but the prediction result of the

Gaussian process regression is within the entire real range, as mentioned in 3.1.1. However, GP model can be adapted to classification problems by transforming the output using an appropriate nonlinear activation function, such as logistic sigmoid:

$$\sigma(f) = \frac{1}{1 + exp(-f)} \tag{3.16}$$

Consider a binary classification problem with target value $t \in \{-1, 1\}$. A Gaussian process is defined over a function $f(x)$ and then this function is transformed by a logistic sigmoid $y = \sigma(f)$, with $y \in (0, 1)$. A case of 1-D input space is presented in figure 3.2, where 3.2(a) illustrates a sample from a Gaussian process prior over function $f(x)$ and 3.2(b) illustrates the result of transforming the function $f(x)$ using a logistic sigmoid function σ. The probability distribution of the target variable is given by the Bernoulli distribution:

$$p(t|f) = \sigma(f)^t(1 - \sigma(f))^{1-t} \tag{3.17}$$



(a) (b)

Figure 3.2: Example of (a) a sample from a Gaussian process prior over function $f(x)$ and (b) Result of transforming the function $f(x)$ using a logistic sigmoid function

Similar to the regression, the training input vectors are denoted by $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N$, the corresponding observed targets are denoted by $\mathbf{t}_N = [t_1, t_2, ..., t_N]^{\mathrm{T}}$. The objective is to determine the predictive distribution of $p(t_{N+1}|\mathbf{t}_N)$. In classification problems, the Gaussian noise ε in covariance matrix no longer exists because it is assumed that every training input vector is correctly labelled. In a two-class problem, predicting $p(t_{N+1} = 1|\mathbf{t})$ is enough because the probability of $p(t_{N+1} = -1|\mathbf{t})$ is obtained by $1 - p(t_{N+1} = 1|\mathbf{t})$. The expected predictive distribution is obtained by a marginal distribution expressed as:

$$p(t_{N+1} = 1|\mathbf{t}) = \int p(t_{N+1} = 1|f_{N+1})p(f_{N+1}|\mathbf{t})df_{N+1} \tag{3.18}$$

where $f_{N+1} = f(\mathbf{x}_{N+1})$. However, this integral is analytically intractable. To approximate the integral, several inference methods can be taken into account, such as Laplace approximation [111], variational inference [112], expectation propagation [113], etc.

3.2 Proposed Gaussian process classification

The proposed Gaussian Process includes 2 main steps: training step and testing step for the purpose of classification using the processed signals from sensors. In the first step, input-output pairs of known data are used to train Gaussian process model for each case corresponding to healthy and damaged structural status. As the training data is correctly labelled, the aim is to determine an optimal mapping function between the inputs and outputs. In this step, the optimal mapping function should be able to identify discriminative features that can indicate health or damage status of the structure. In the second step, the performance of the GP model trained in the previous step is evaluated when facing unknown features so as to determine if damages appear in the structure or not.

3.2.1 Training phase

The inspection of the structural health status is achieved by a SHM system consisting of several sensors permanently attached on the structure. Each sensor can be used as an actuator to apply a pulse excitation signal or as a receiver to collect data. In each simulation scenario, a pre-defined pulse signal is applied on the actuator, which in turn transmits the signal to the structure in the form of vibration. The signal interacts with elements of the structure during the propagation over the whole structure. Sensors located at three corners are considered capable of collecting signals that have interacted with potential damages without missing information. These raw time series signals are considered to contain discriminative information about damages and signal processing is then conducted. This process is achieved by Discrete Wavelet Transform. The signal processing step can be regarded as reduction of data dimension and feature extraction because DWT is used to extract information in both time and frequency domains as well as discarding data that is not dominant without any major loss of information. Data after signal processing are further used in machine learning algorithms to learn models. In DWT, the

Coiflet 5 is chosen as mother wavelet after comparing with other wavelets. This wavelet is the optimal among others in terms of errors, which will be explained in detail in section 3.3.1.

In each simulation case, one file containing signals collected by three sensors is obtained. It should be noted that as mentioned in section 2.5, the duration of each simulation is 0.01s and the sampling frequency depends on the input signal frequency, thus the number of sampling points for a fixed time duration is different for different signals.

Note that in DWT the successive subsampling is by 2, the collected signal length should be a power of 2 or a multiple of power of 2 in order to make the scheme efficient. Thus, the first 512 sampled points for signals whose length is more than 512 are used for DWT, while for collected signals whose length is less than 512, zeros are added to the end of each signal until its length becomes 512. DWT is employed for each collected signal, and the corresponding coefficients are saved as one vector with the same length as the original signal. Since the frequency bands that are not very prominent in the original signal will have very low amplitudes, that part of the DWT signal can be discarded without any major loss of information. For this reason, only the first 256 coefficients are used, as shown in figure 3.3, where the upper figure is a vibration signal collected at sensor 2 (on the diagonal of the actuator) while the lower figure is the corresponding DWT signal.

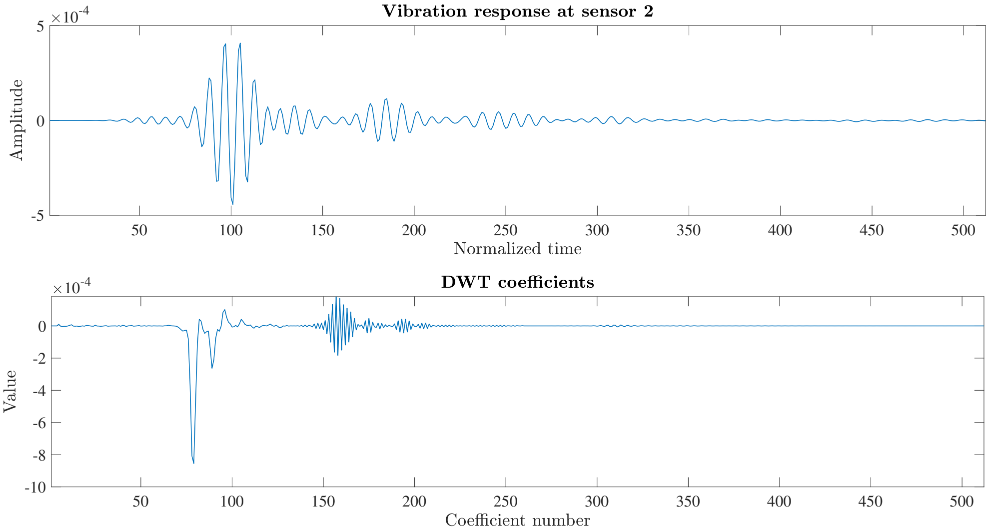


Figure 3.3: Example of a DWT for a signal collected at sensor 2 (on the diagonal of the actuator)

In each simulation, the three vectors after dimension reduction obtained by DWT are merged into one vector which contains all features from different sensors in the order of the sensors. The same procedure is performed for all simulation cases. All the vectors are finally organized in one matrix in which each row consists of one transposed vector representing one simulation. This matrix contains all features of the structures extracted by DWT from every simulation. As data from different sensors are merged in one vector, and all vectors for different simulation cases are merged in one matrix before the employment of Gaussian process, this procedure is considered as early data fusion.

The matrix after data fusion has a dimension of $I \times 3K$, where I is the number of simulations, K is the number of coefficients retained after signal processing by DWT on the signal collected from each sensor, and $3K$ represents the vector length after data fusion of three sensors in one simulation. The obtained matrix is in the form of:

$$\mathbf{X}=\begin{pmatrix} x_{11}^1 & \cdots & x_{1K}^1 & x_{11}^2 & \cdots & x_{1K}^2 & & x_{11}^J & \cdots & x_{1K}^J \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ x_{I1}^1 & \cdots & x_{IK}^1 & x_{I1}^2 & \cdots & x_{IK}^2 & & x_{I1}^J & \cdots & x_{IK}^J \end{pmatrix} \tag{3.19}$$

where the element is denoted by x_{ik}^j with i indicating the simulation number (i^{th} simulation), j indicating the sensor number (j^{th} sensor) and k indicating the DWT coefficient number (k^{th} coefficient). Group scaling is performed on this matrix in order to remove the mean trajectories of each sensor and to make sure that data from any sensor have the same variance[114]. Group scaling is conducted with respect to sensors, which means that data collected from the same sensor during every simulation are scaled. As the matrix can be seen as composed of three sub-matrices representing data from three sensors, the element in each sub-matrix is scaled by:

$$\bar{x}_{ik}^j = \frac{x_{ik}^j - \mu_k^j}{\sigma^j} \tag{3.20}$$

where μ_k^j is the mean of the I measurements of sensor j for the k^{th} coefficient, in the form of

$$\mu_k^j = \frac{1}{I}\sum_{i=1}^{I} x_{ik}^j \tag{3.21}$$

σ^j is the standard deviation of all measurements of sensor j expressed as

$$\sigma^j = \sqrt{\frac{1}{IK}\sum_{i=1}^{I}\sum_{k=1}^{K}\left(x_{ik}^j - \mu^j\right)^2} \tag{3.22}$$

and μ^j is the mean of all measurements of sensor j. This scaled matrix $\bar{\mathbf{X}}$ composed of $\bar{x}^j_{ik}$ is used as input to the Gaussian Process for subsequent analysis.

In GP training step, input vectors in the matrix $\mathbf{X}$ are correctly labelled by the ground truth (GT) t with $t = 1$ indicating *healthy status* and $t = -1$ indicating *damaged status*. Likelihood function together with inference method are employed. These two functions are specified in the training step. Hyperparameters are predefined randomly for the purpose of initialization. The input is then fed into the GP model and in turn an output predicted by the model is given denoted by $\hat{t}$. During the learning process, negative logarithmic likelihood function (3.12) is used as loss function to describe the error between the predictive output $\hat{t}$ derived from GP model and the corresponding ground truth t. As the predictive output $\hat{t}$ is a function of hyperparameters in the kernel function and likelihood function denoted by Θ, the negative logarithmic likelihood function is also a function of the hyperparameters. The optimization process is achieved by minimizing the negative logarithmic likelihood function with respect to the hyperparameters Θ over a series of iterations, which corresponds to choosing the value of Θ for which the probability of the observed data set is maximized. As a result, the error between the predictive output and the ground truth is minimized. This procedure is illustrated in figure 3.4.

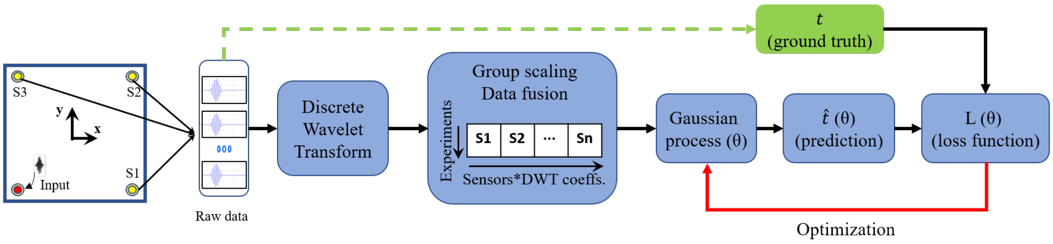


Figure 3.4: Gaussian Process training step

3.2.2 Testing phase

Once the hyperparameters are optimized for the training data, the validated GP model can be used to make predictions for new inputs. In this study, the GP model is used to predict if damages occur in the structure. The testing procedure is the same as that of training before inserting the transformed input into the GP model. However, in the testing phase, only the trained GP model is used. Besides, different from the training phase where optimized hyperparameters are as output, the trained GP model will provide a predictive mean value corresponding to the new input and a variance of the predictive distribution based on the training data, as described in section 3.1.1. The predictive mean value indicates in which label class the input is most likely distributed, that's to say healthy or damaged class, while the variance provides an information of confidence, which indicates the reliability of the classification result. The predictive label from the GP model can be compared to the corresponding ground truth so that the effectiveness of the trained GP model can be evaluated. It should be noted that the GP model is used to predict the appearance of damages, but unable to localize the probable region where damages occur. The procedure of the testing step is presented in figure 3.5.

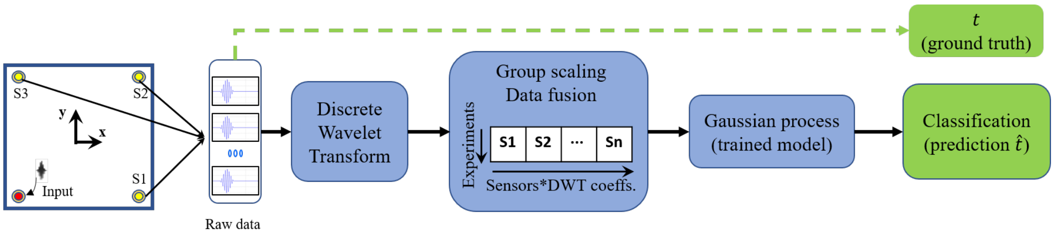


Figure 3.5: Gaussian Process testing step

For the six sandwich models with cracks, simulations are conducted with excitation signals of three different frequencies successively. 30 data sets for each model are collected. For the damage-free sandwich model, same simulations are conducted with excitation signals of three different frequencies successively as for the damage models. 90 data sets are obtained. Thus a data base is constructed with a total number of 270 data sets, in which 90 data for healthy model labelled by '1' and 180 data for damaged models labeled by '-1'. 70% of the data base is used for training the GP model and the rest 30% is used for testing. The testing step is a step for evaluating the GP model for damage predictions in composite sandwich plates. In both training and testing step, several factors will affect the results, such as the type of mother wavelet for the DWT, the amount of data discarded by DWT, the selection of likelihood function and inference method for GP, and the number of function evaluations to optimize hyperparameters during the training step, etc. The influence of these factors will be evaluated successively. It is worth mentioning that although discarding the coefficients that have low amplitude in DWT may not lead to major loss of information, as has been mentioned in the section 2.6.4, it will slightly reduce the accuracy of the results, but the effect is negligible.

3.3 Results and discussions

In this section, the influence of previously mentioned factors on the classification results is discussed. In binary classification case, 1 represents healthy status while

-1 represents damaged status. It is expected that the predictive mean value is as close to 1 as possible for the case whose ground truth is healthy, while the predictive mean value is as close to -1 as possible for the case whose ground truth is damaged. However, in GP classification model, the predictive mean value is not necessarily 1 or -1. Therefore, it is necessary to define a threshold to determine if the prediction from the GP model is correct or not. In the testing step, an overall predictive mean value for each class can be calculated, μ_d for damaged class and μ_h for healthy class. The threshold to separate two classes is defined as the mean value of μ_d and μ_h. The distance from overall mean value to the threshold is called class distance, indicating how far the two classes are separated, which is considered as a measure of the quality of the classifier. If the predictive mean value for a case whose ground truth is damaged is less than the threshold, the prediction is considered as correct, indicating a damaged status. The same, if the predictive mean value for a case whose ground truth is healthy is greater than the threshold, then it means that the status of the structure is predicted as healthy by GP model, which is correct. Otherwise, it is considered that the GP model misclassifies the case.

3.3.1 Classification with three sensors

Firstly, the influence of the selection of mother wavelet is investigated. As mentioned in 2.6.4, the selection of mother wavelet has an important impact on the results because the highpass and lowpass filters used during the discrete wavelet transform are determined by the mother wavelet. Different filters will perform differently in filtering, thus will affect the transform results, which in turn will affect the result of Gaussian process classification.

In the current study, control variate method is employed, that's to say in each analysis, only one factor is considered as a variable while others are invariant. Daubechies 8 (D8) wavelet, Coiflet 5 (C5) wavelet, Haar wavelet and Symlet 10 (S10) wavelet are compared with 512 saved coefficients in DWT, logistic function as likelihood function, Variational Bayesian (VB) as inference method. 40 iterations are adopted in the optimization step. Results are shown in figure 3.6. The abbreviation of mother wavelet, saved amount of data in DWT, likelihood function, inference method and iteration numbers are successively listed in the title of each sub-figure, followed by the predictive mean values for two classes, e.g., 'D8-512-Logistic-VB-40, $\mu_h = 0.697$, $\mu_d = -0.858$' in figure 3.6(a). It should be noted that in all the following classification figures in this section, the first 54 testing points (denoted by □) along the x-axis are with ground truth 'Damage' while the last 27 testing points (denoted by $*$) are with ground truth 'Health'. The global mean value of predictive mean value for all testing damaged cases and that for healthy cases are expressed

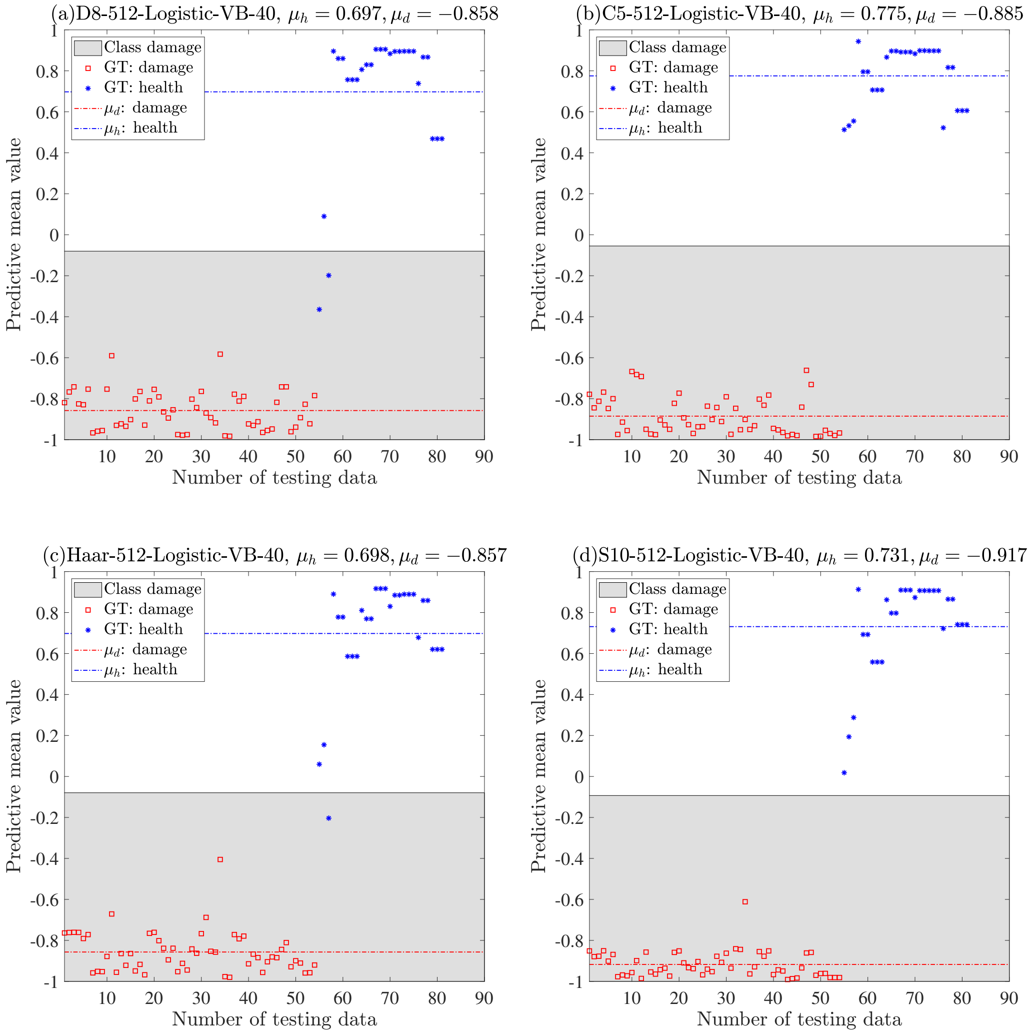


Figure 3.6: Influence of the selection of mother wavelet on the classification accuracy. (a) Daubechies 8, (b) Coiflet 5, (c) Haar, (d) Symlet 10

as

$$\mu_d = \frac{1}{n}\sum_{i=1}^{n}\mu_{di} \tag{3.23}$$

$$\mu_h = \frac{1}{m}\sum_{i=1}^{m}\mu_{hi} \tag{3.24}$$

where the subscripts $_d$ and $_h$ represent *Damage* and *Health*, respectively. The global mean value of predictive mean value for all testing damaged cases and that for healthy cases are -0.885 and 0.775 using Coiflet 5 wavelet, while the results using Daubechies 8 wavelet are -0.858 and 0.697, Haar -0.857 and 0.698, Symlet 10

−0.917 and 0.731, respectively, as shown in figure 3.6(a-d). The class distance using Coiflet 5 is 0.830, larger than that using D8 0.778, Haar 0.777 and S10 0.8244. Meanwhile, the mean square error for Coiflet 5 is 0.019, much smaller than 0.067 for D8, 0.044 for Haar and 0.030 for S10. In addition, every single case is correctly classified using Coiflet 5 wavelet, which suggests that Coiflet 5 is more powerful in extracting features.

It should be noted that some other mother wavelets are also evaluated but have poorer performance and the corresponding results are listed in Table 3.1. The classification accuracy of GP model using Haar wavelet and D8 wavelet are 96.30% and 98.77%, respectively. By contrast, the accuracy using other wavelets are 100%. However, among all wavelets, the class distance using Coiflet 5 is larger than others while the mean square error (MSE), which measures the average squared difference between the estimated values and the actual value, is lower, indicating that using Coiflet 5 wavelet can make the distinction between the two classes more obvious, and the samples of each category are more concentrated. Therefore, the GP model performs better in the classification for both healthy and damaged structures using Coiflet 5 wavelet.

Table 3.1: Performance of different wavelets in GP classification.

	Haar	Beylkin	Coiflet5	Coiflet3	Coiflet1
Class distance	0.5261	0.8344	0.8803	0.8161	0.8697
MSE	0.0799	0.0374	0.0087	0.0214	0.0135
Accuracy	96.30%	100%	100%	100%	100%
	D20	D16	D12	D8	D4
Class distance	0.8397	0.8380	0.8408	0.8363	0.8268
MSE	0.0172	0.0140	0.0133	0.0354	0.0251
Accuracy	100%	100%	100%	98.77%	100%
	Symlet10	Symlet8	Symlet6	Symlet4	Vaidyanathan
Class distance	0.8738	0.8323	0.8575	0.8578	0.8320
MSE	0.0145	0.0329	0.0121	0.0172	0.0162
Accuracy	100%	100%	100%	100%	100%

Secondly, the influence of the amount of data discarded by DWT on the classification accuracy is discussed. According to the previous study, Coiflet 5 mother wavelet, Logistic likelihood function, Variational Bayesian inference method and 40 iterations are used due to the outstanding performance. An example of DWT coefficients for a signal collected at sensor 2 (on the diagonal of the actuator) together with the original signal is illustrated in figure 3.3. The length of the DWT coefficients is the same as the signal. It is shown that the first 256 coefficients are more dominant in amplitude than the last 256 ones. Therefore, all 512 coefficients and the first 256 coefficients are separately used for GP model to study the influence of the amount of data discarded by DWT on the classification accuracy. The class distance of classification result with all 512 coefficients and that with the last 256 coefficients discarded are almost equal, with a value of 0.830 and 0.835, while the former MSE is 0.019, smaller than the latter 0.038. Result shows that without discarding coefficients, GP performs slightly better than that discarding the last 256 coefficients that have low amplitude as shown in figure 3.7, which is reasonable and easy to understand, because although discarding the coefficients that have low amplitude in DWT may not lead to major loss of information, it will slightly reduce the accuracy of the results, but the effect is negligible.

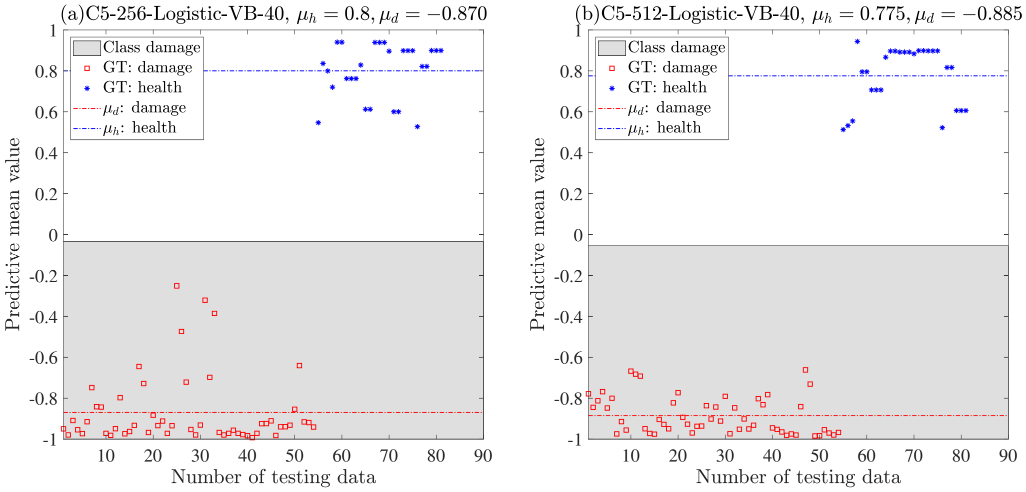


Figure 3.7: Influence of the amount of data saved by DWT on the classification accuracy. (a) 256 coefficients, (b) 512 coefficients

Thirdly, the influence of likelihood function and inference method is investigated simultaneously because there is an issue of compatibility among different likelihood functions and inference methods. Four combinations of likelihood function and inference method are presented in figure 3.8, including Logistic-VB in figure 3.8(a), Logistic-Expectation Propagation (EP) in figure 3.8(b), Logistic-Laplace in figure 3.8(c) and Error function (Erf)-EP in figure 3.8(d). The class distance for the combination from (a) to (d) are 0.779, 0.630, 0.251 and 0.684, respectively. The combination of Logistic-VB performs better in distinguishing two classes than others. The corresponding mean square error are 0.047, 0.100, 0.017 and 0.114, respectively. Although the MSE for Logistic-Laplace is the lowest, the corresponding class distance

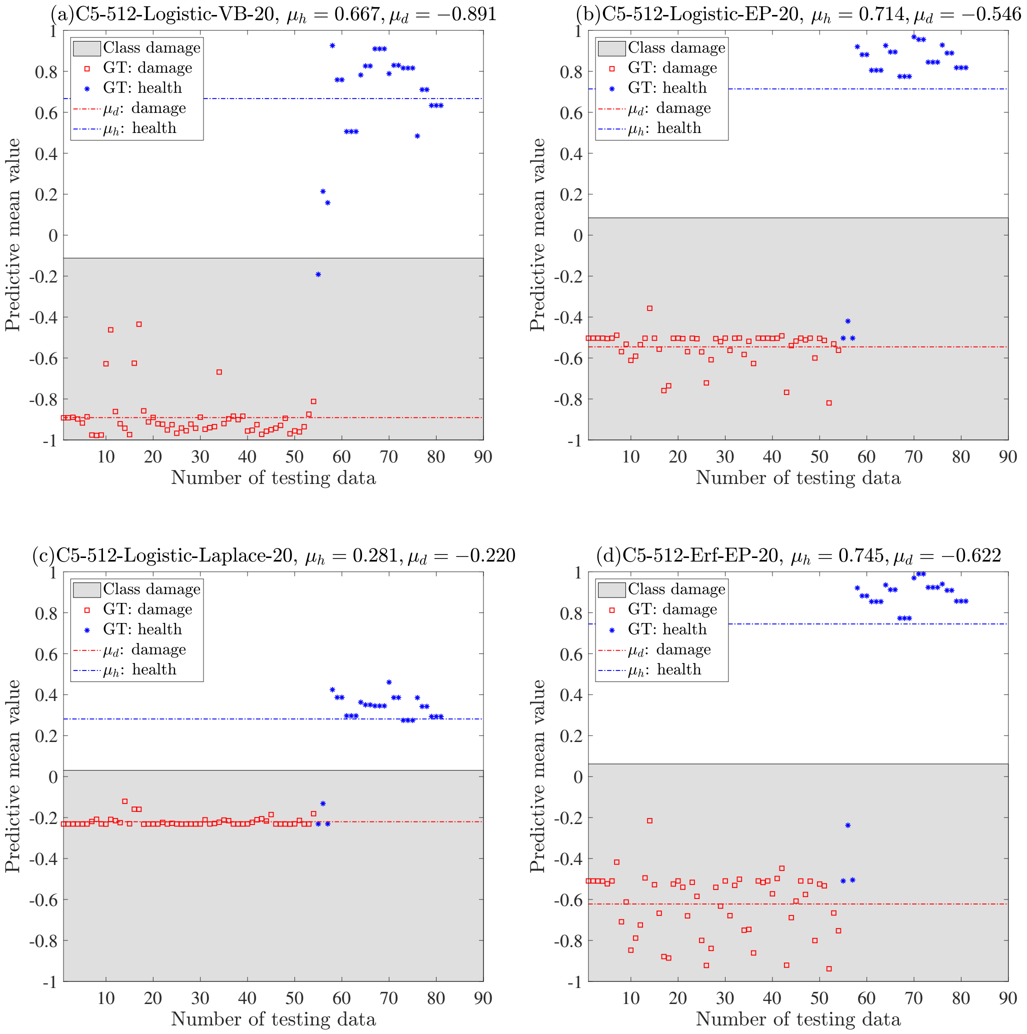


Figure 3.8: Influence of likelihood function and inference method on the classification accuracy. (a) Logistic-Variational Bayesian (VB), (b) Logistic-Expectation Propagation (EP), (c) Logistic-Laplace and (d) Error function (Erf)-Expectation Propagation

is the smallest, with global predictive mean value for healthy and damaged cases 0.281 and -0.22 respectively, around 0 and far from 1 or -1, indicating that this combination has the worst performance in classifying different samples. The Logistic-EP combination and Erf-EP combination perform better than Logistic-Laplace, with a global mean value $\mu_h = 0.714, \mu_d = -0.546$ and $\mu_h = 0.745, \mu_d = -0.622$, respectively. However, 3 cases are misclassified with Logistic-EP combination and 3 cases are misclassified with Erf-EP combination. A better result is achieved by Logistic-VB combination with global mean value $\mu_h = 0.667, \mu_d = -0.891$ where the class distance is 0.779, larger than others, with a smaller MSE 0.047, indicating a better classification quality, and only 1 case is misclassified.

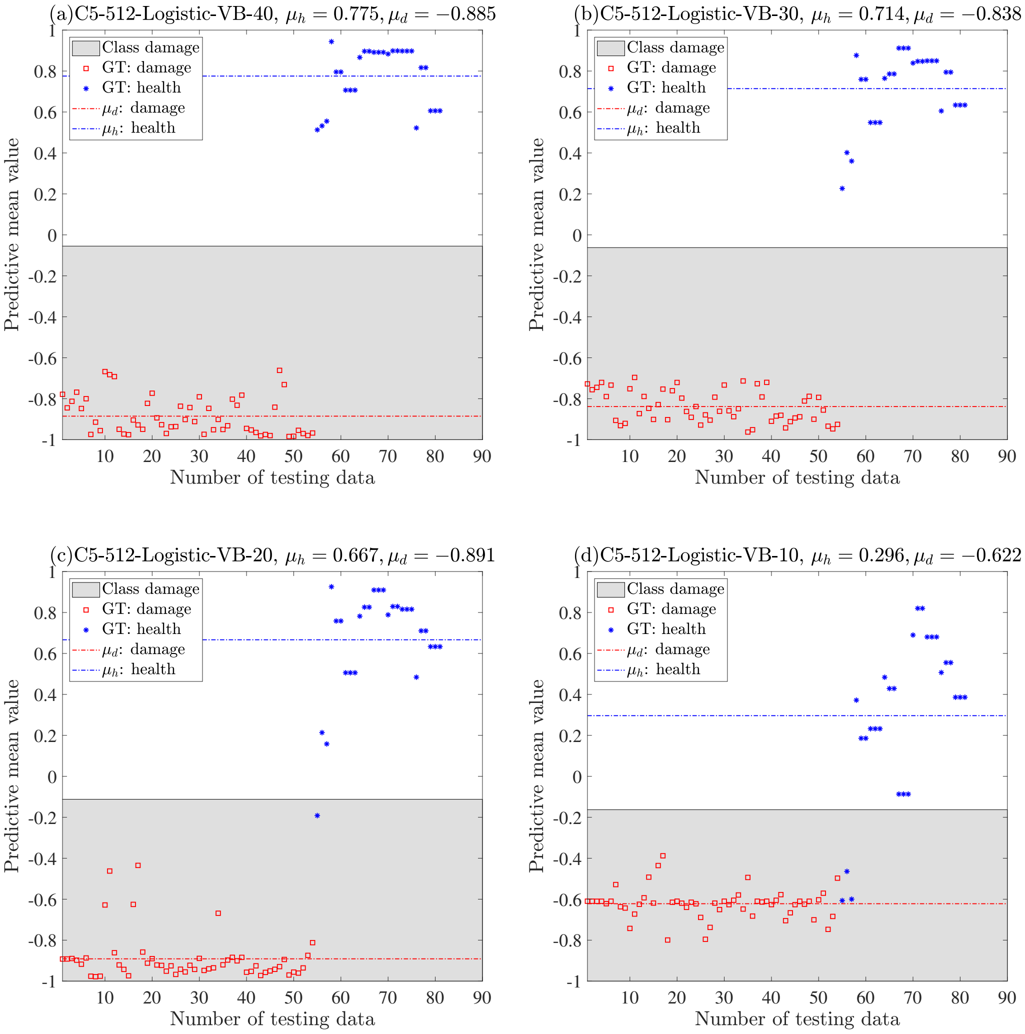


Figure 3.9: Influence of iteration numbers to optimize hyperparameters during the training step on the classification accuracy. (a) 40 iterations, (b) 30 iterations, (c) 20 iterations, (d) 10 iterations

Finally, the influence of iteration numbers on the accuracy of the classification results is discussed. The iteration number represents the number of optimization cycles in the training step of GP model. Normally, the more iterations there are, the more accurate the result should be. Finding an appropriate number of iterations can not only ensure the quality of the results but also avoid wasting computing time. The result is presented in figure 3.9, where the vertical axis indicates intervals of the predictive mean value μ and the horizontal axis indicates the number of testing data. A comparison of 40, 30, 20 and 10 iterations is carried out. Here the Coiflet 5 (C5) mother wavelet is chosen in the DWT and the first 512 coefficients are saved to be used as input for GP. Logistic function is chosen as the likelihood function and Variational Bayesian (VB) is selected as the inference method. The global mean value of predictive mean value for all testing damaged cases and that for healthy cases $\mu_g = \frac{1}{n}\sum_{i=1}^{n}\mu_i$ are -0.885 and 0.775 for 40 iterations, while the results for 30 iterations are -0.838 and 0.714, for 20 iterations -0.891 and 0.667, for 10 iterations -0.622 and 0.296, respectively. Besides, 1 case with ground truth (GT) 'healthy' is misclassified in the 'damaged' class for 20 iterations and 3 cases misclassified for 10 iterations, while every case is correctly classified with 40 iterations and 30 cases. In addition, the class distance for 40 iterations is 0.830, larger than that for 30, 20 and 10 iterations. Moreover, the MSE is 0.019 for 40 iterations, smaller than 0.022, 0.047, 0.084 for 30, 20 and 10 iterations. It shows that with more iterations, the result is slightly better than with less iterations, which is consistent with common sense. 40 iterations are enough to obtain an acceptable classification result.

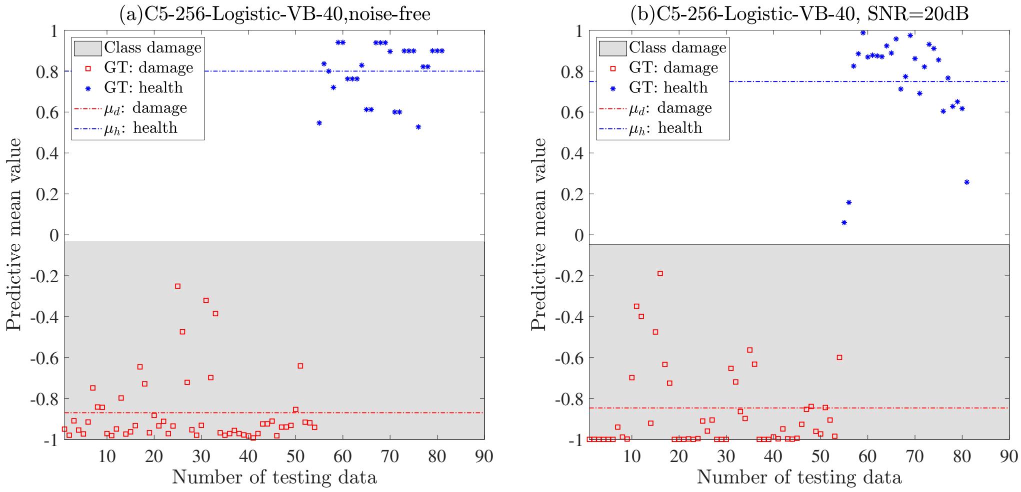


Figure 3.10: Comparison of classification result based on (a) noise-free data and (b) data with noise.

It should be noted that the current research is based on simulation, and there is no doubt that there will always be differences between simulation and experiments. In real experimental data the impact of environmental noise should be taken into consideration. Therefore, it is necessary to add noise to the simulation data to simulate the influence of environmental noise, so as to approximate experiments and verify the stability of the proposed method. White Gaussian noise is added into the raw vibration data that are collected from sensors. Signal-noise ratio (SNR) is set as 20dB. Then the same procedure of signal processing with DWT and Gaussian Process classification mentioned in Section 3.2 is conducted. A comparison of classification result based on noise-free data and data with noise is illustrated in figure 3.10. Despite the added noise, the class distance has only slightly decreased from 0.835 to 0.798, and all data can be correctly classified. This proves that the proposed Gaussian process-based method is still practical under simulated environmental noise.

3.3.2 Classification with one sensor

Reducing the number of sensors used can not only save equipment costs, but also reduce the computational cost of data processing and training machine learning models. Now that an acceptable classification result has been obtained with a certain setup of the GP model using the data from 3 sensors in Section 3.3.1, a discussion concerning the reduction of the number of sensors while ensuring the classification result is conducted in this section.

Based on the results obtained in 3.3.1, the model configuration with mother wavelet C5, 256 DWT coefficients, Logistic likelihood function, VB inference method and 40 iterations is maintained. The pertinence of data from three sensors at different locations to the classification result is investigated. Data from three sensors are processed and used for training and testing of the GP model independently. Hence three GP models are constructed. Classification results from the three GP models are compared between each other and also with the results with all data from sensor 1, 2 and 3, as shown in figure 3.11. It shows that with data from all three sensors the GP model achieves the best result in the current study. However, with data only from sensor 1, the classification accuracy can be 100%, but it should be noted that as there exist several testing points so close to the threshold in both classes, they could probably be misclassified. With data only from sensor 3 , at least 2 point among 81 are misclassified, but with data only from sensor 2 located at the diagonal of the actuator, the result is much worse where at least 6 testing points are misclassified.

In this section, Gaussian Process is evaluated by the data base collected through sensors attached on the sandwich plate. 70% of the database is used for training and the rest 30% is used for testing. Several factors are evaluated successively and it is found that: to optimize hyperparameters during the training step, the more iterations, the better the results; during the signal processing and feature extraction, the mother wavelet Coiflet 5 in DWT is outperforming other mother wavelets; in the whole GP, the combination of Logistic function and Variational Bayesian inference method is outstanding compared to other combinations; Reducing the amount of data by DWT will result in a slight loss of accuracy of the predictive mean value,

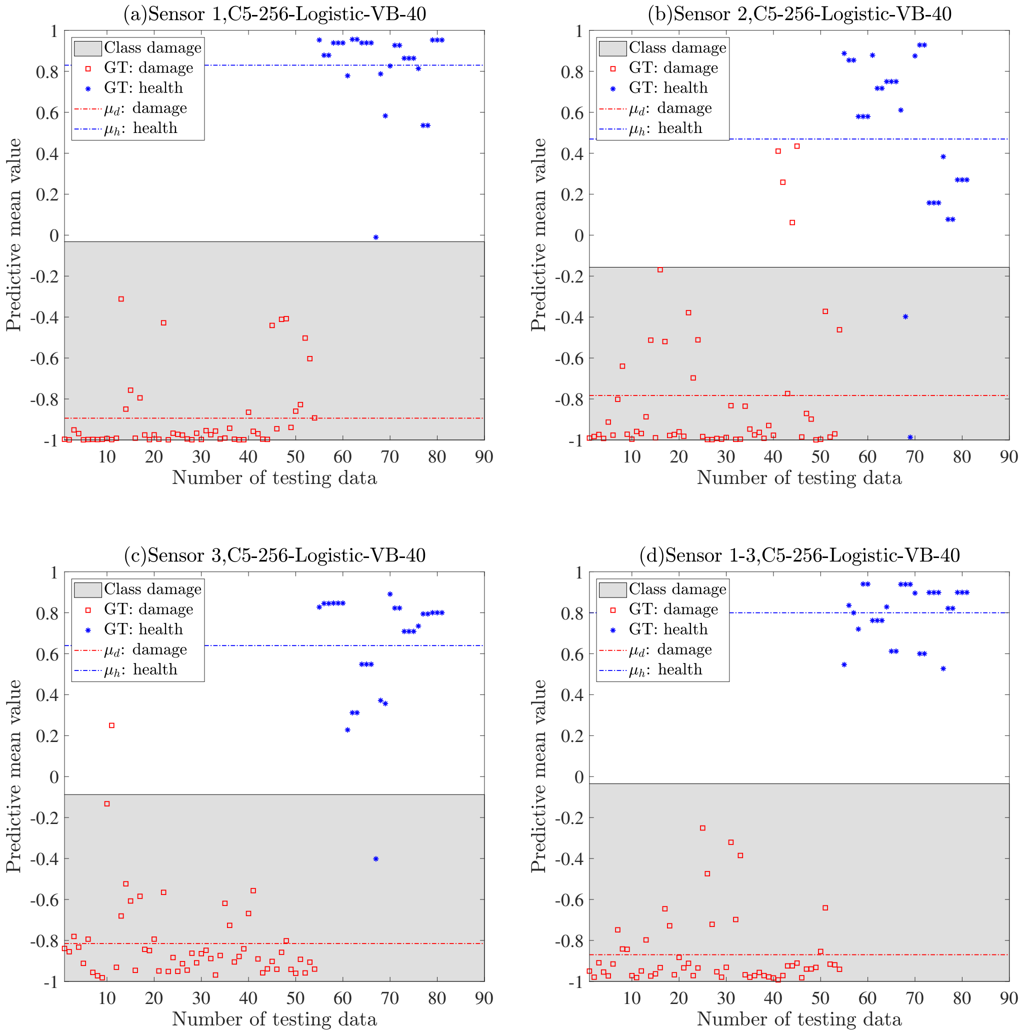


Figure 3.11: Investigation of the pertinence of three different sensors to the classification result. Classification using (a) data only from sensor 1, (b) data only from sensor 2, (c) data only from sensor 3 and (d) data from sensor 1-3

but it can greatly reduce the computation time while ensuring the accuracy of classification, which is worthwhile for processing big data. Therefore, a GP model with mother wavelet C5, 256 DWT coefficients, Logistic likelihood function, VB inference method and 40 iterations is adopted. The corresponding predictive mean values for healthy and damaged structure are 0.800 and -0.870, respectively. As for the impact of the number of sensors, the result shows that data from sensor 1 and sensor 3 are more pertinent than that from sensor 2. It shows also that reducing of the number of sensors in the current study will lead to worse results.

3.4 Conclusions

This section addresses the difficulty in constructing a database for structural health monitoring of real composite structures and the inconvenience in conventional expertise-based SHM approaches. A data-driven approach GP for damage detection in a composite sandwich plate by simulation approach is presented. Several factors that have impact on classification accuracy are investigated, including the selection of mother wavelet during the signal processing and feature extraction by DWT, the amount of data discarded by DWT, the likelihood function and inference method that are used to make predictions, as well as iteration numbers in the training step of GP model. Based on the present results, some conclusions can be drawn as below:

- The proposed method is proven effective for crack-type damage detection in the studied composite sandwich plate.
- The selection of mother wavelet in discrete wavelet transform has an important impact on the classification accuracy. Coiflet 5 performs better than others for the classification of both healthy and damaged structure.
- Discarding the coefficients that have low amplitude in DWT can speed up the computation time and may not lead to major loss of information. The resulted reduction of classification accuracy is negligible.
- The likelihood function Logistic function and inference method Variational Bayesian perform better than other combinations in the present study.
- With more iterations, the classification is slightly better than with less iterations, which is consistent with common sense.
- Data from sensor 1 and sensor 3 are more pertinent than that from sensor 2. Reducing the number of sensors in the current study will lead to worse results.
- The effectiveness of the proposed method is verified under simulated environmental noise.

Although the proposed approach was achieved based on simulation results, it is applicable for real experimental data. The same procedure of data collection, signal processing and the use of GP model can be conducted as for simulation data for the purpose of damage detection. This proposed method is capable to detect crack-type damages for a composite sandwich panel. It is expected to be suitable for damage detection of other kind of damages and for more complex structures.

Chapter 4

Convolutional Neural Networks based structural health monitoring at mid-high frequencies

Contents

4.1 Introduction

For an ANN, with the increase of hidden layers and neurons in each layer, the number of hyperparameters will increase dramatically, which results in high computational and memory requirements. Convolutional Neural Networks (CNN) have emerged with distinct advantages such as high efficiency and low computation cost compared to ANN.

A CNN is a deep learning algorithm which can take in an input image, assign importance by adjustable weights and biases to various contents in the image and be able to distinct one from the other. In computer vision, an image is nothing but a matrix of pixel values. A CNN interprets an input as hierarchical representations by convolving it with multiple filters. CNN has been well developed in computer vision and has brought breakthroughs in processing images, video and speech [115]. In CNN, there are basically three types of layers:

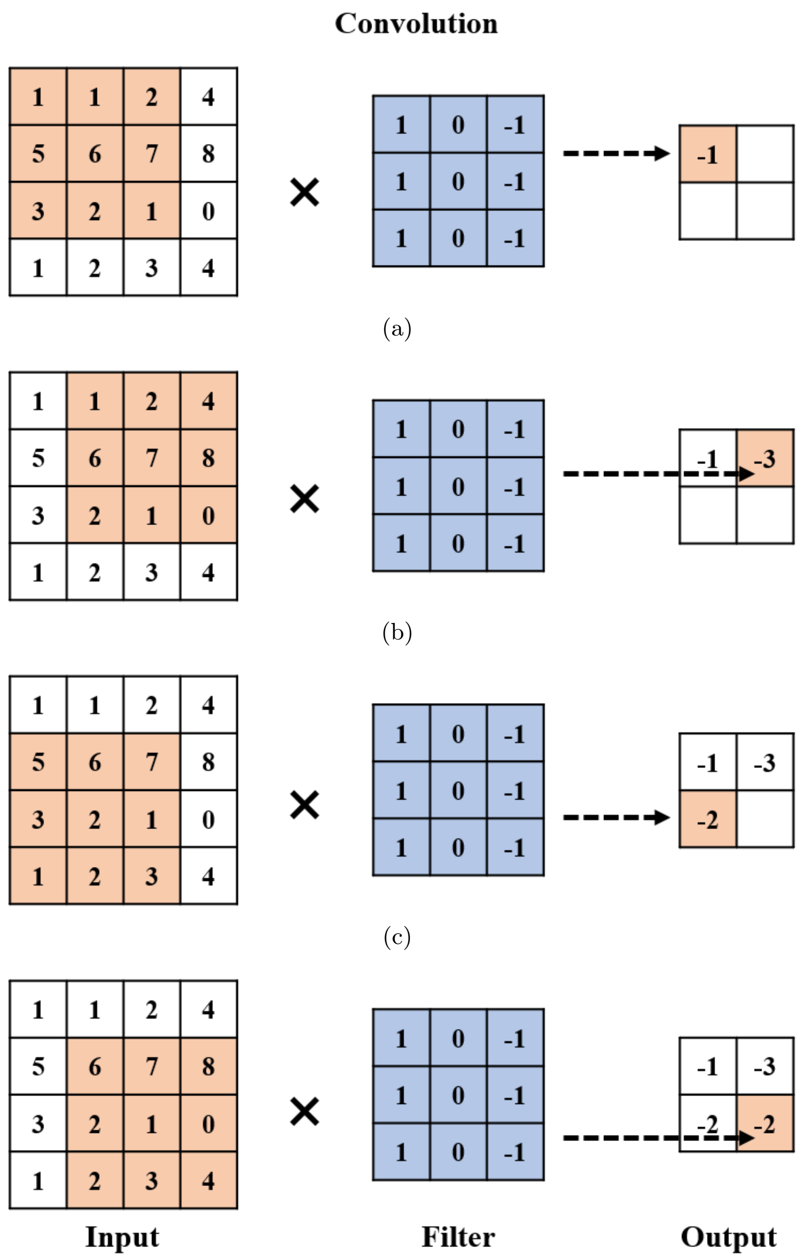
- Convolution layer
- Pooling layer
- Fully connected layer

4.1.1 Convolution layer

Convolution layer is used to extract key features in the input images by sliding a filter from left to right, from top to bottom by small steps called stride, until it covers the whole image through a convolution operation. Suppose we have a gray scale image with dimension 4×4 and a filter of dimension 3×3. When this image convolves with the filter, we get a 2×2 image. The convolution operation is computed as follows: by multiplying the first 3×3 matrix of the 4×4 matrix with the filter, the output is the sum of the element-wise product of these values. This is the first element of the 2×2 matrix. To calculate the second element of the 2×2 output, the filter is shifted one step towards the right and again get the sum of the element-wise product, as shown in figure 4.1. Two simple and commonly used filters for detecting vertical edges and horizontal edges are illustrated in 4.2(a) and 4.2(b), respectively. In the filter, higher pixel values represent the brighter portion of the image and lower values represent darker portions.

4.1.2 Stride and padding

Stride denotes the number of steps we are moving in each step in convolution. In the above example the stride is 1. We denote the stride with s. It is observed that



(a)

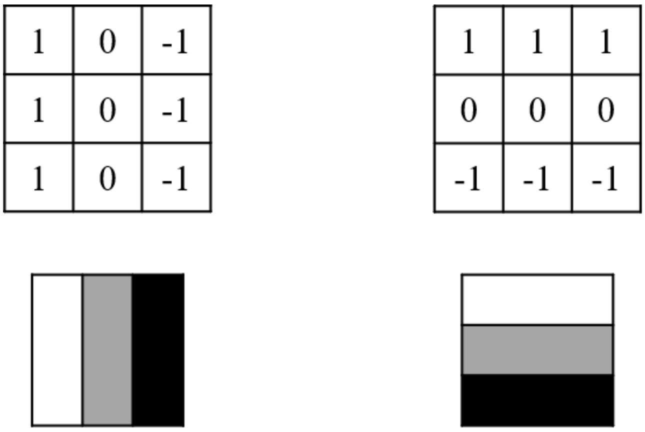
(b)

(c)

(d)

Figure 4.1: Illustration of the convolution operation of an input image with a filter of dimension 3 by 3. stride = 1, padding = 0

the size of the output image is smaller than the input image after convolution in figure 4.1. In addition, pixels in the corner of the image are used less often than the central pixels during the convolution, which may result in information loss. In some cases it is necessary to maintain the size of the output as input, padding is



1	0	-1
1	0	-1
1	0	-1

1	1	1
0	0	0
-1	-1	-1

(a) (b)

Figure 4.2: Two commonly used filters for detecting (a) vertical edges and (b) horizontal edges in images

used. Padding is an operation to add zeros to the borders of the input matrix symmetrically so that output and input will have the same dimension. We denote padding as p, input size as $n \times n$, filter size as $f \times f$. Then the size of output can be obtained as $(n + 2p - f + 1) \times (n + 2p - f + 1)$. The size of output is the same as the input if $p = (f - 1)/2$.

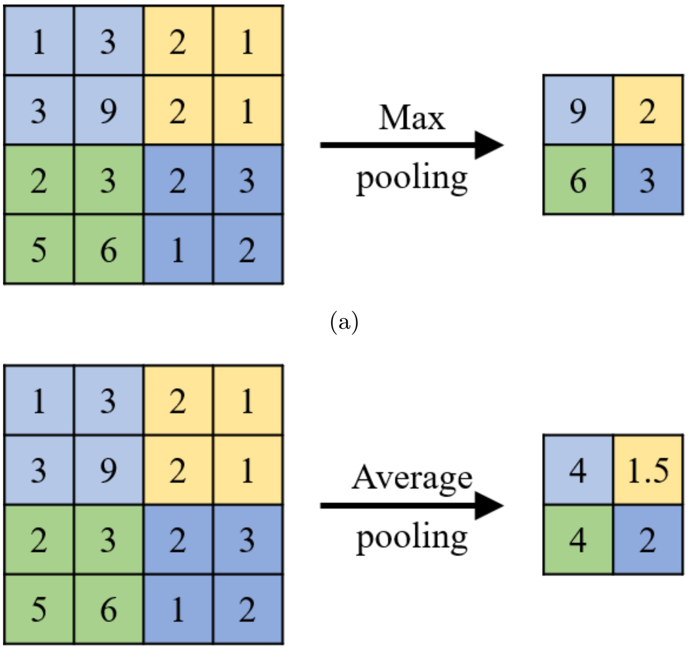
4.1.3 Pooling layer

Similar to convolution layer, the pooling layer is used to reduce the spatial size of width and height of the output after convolution. It is used between two convolution layers. Another purpose of using pooling layer is to extract dominant features. From a mathematical point of view, it picks the dominant value from a certain size of window and discard other values in the same window.

There are two types of pooling: max pooling and average pooling. Max pooling, as the name suggests, is to extract the maximum value from the portion of the image covered by the window (filter), while average pooling extracts the mean value of all elements in the portion covered by the filter, as illustrated in figure 4.3(a) and 4.3(b). Max pooling is widely used to suppress noise along with dimensionality reduction.

4.1.4 Fully connected layer

A fully connected (FC) layer is similar to a layer in ANN, where each neuron in the layer is connected to all neurons in the previous layer. The FC layer is employed to classify images between different categories by training as it learns the high-level



(a)

(b)

Figure 4.3: Two types of pooling: (a) Max pooling, (b) Average pooling

features represented by the output of the convolution layer. The output of the last pooling layer is flattened into a column vector which in turn is fed as input to the FC layer for learning. Over a series of forward and backward propagation of training, the CNN model is able to distinguish high-level and low-level features and classify the image in a corresponding class, as illustrated in figure 4.4.

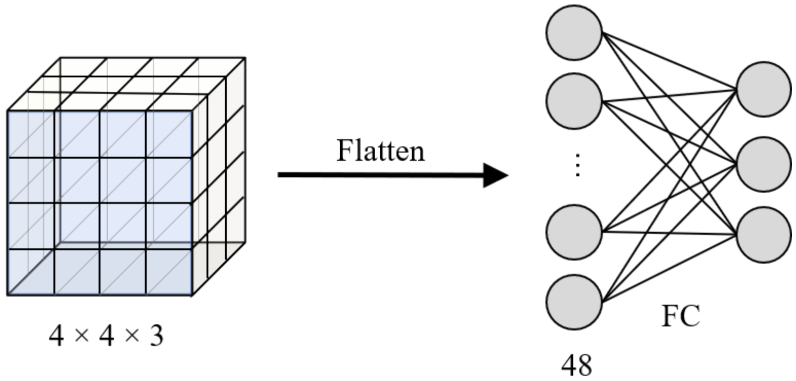


Figure 4.4: Flattening an output of the last pooling layer and passing it to a fully connected layer before classification

Figure 4.5 illustrates an example of a classic CNN architecture LeNet-5 for the recognition of a handwritten number. In the first convolution layer, a filter of dimension 5×5 is applied with a stride 1. The output is in turn transformed with an average pooling with filer size 2×2 and stride 2. The output of pooling layer is again fed into the second convolution layer and pooling layer. The output of the second pooling layer is flattened into a 400×1 vector, which is passed successively through two FC layers until the final classification of the image. During the training process, filters in each layer together with the bias item constitute the hyperparameters denoted by Θ, which will be tuned such that the prediction label can match the ground truth.

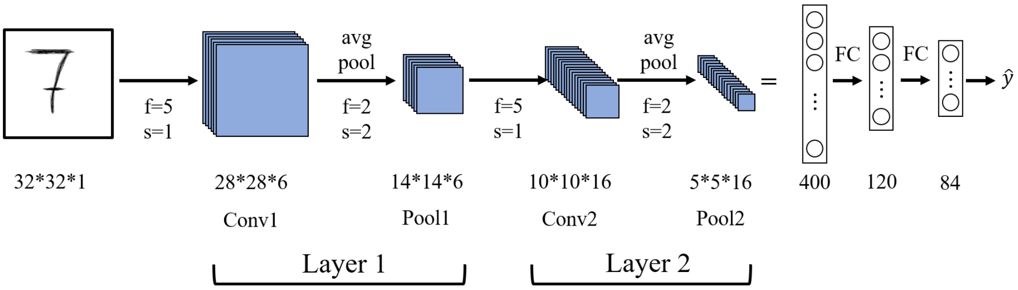


Figure 4.5: Architecture of a classic CNN: LeNet-5

CNN is widely used in computer vision due to its powerful feature extraction and classification capabilities, especially in classification of images of dimension 2D for gray scale images or 3D for RGB images. Although CNN has been approved capable of classifying raw 1D time series signals in several works, it is recommended to perform a pre-processing of signals in order to take advantage of CNN in image classification. Time series signal can be transformed into images using CWT. The image obtained by CWT is thought of as transformed information of the original signal and representing time-frequency information. The representation in the form of images allows the CNN to get high-level features.

4.2 One-stage CNN

In the first attempt, a one-stage CNN is employed to detect and localize crack damage in the composite sandwich structure. After collecting the vibration responses from three sensors attached on the structure, data processing is carried out by CWT which transforms 1-dimensional vibration responses of the intact structure and cracked structures from transient analysis into a 2-dimensional matrix, in which the x-axis represents time, y-axis represents frequency and the value represents the amplitude. The matrix is then visualized as a 2-dimensional image, as shown in figure 4.6. The mother wavelet of CWT is by default, Morse wavelet in Matlab. The frequency band of CWT is between 2000Hz and 14000Hz, which is the vertical axis interval of the 2-D image in figure 4.6, covering the frequencies of input signals, and the horizontal axis represents time. It should be pointed out that axes in the CWT image have the same scale for all cases and that the axes are hidden in the employment of CNN. The first reason of representing the data by CWT is to maintain the information in both time and frequency domains. The second is that it can take advantage of the image classification capability by CNN. All these 2-D images constitute the database, which will be used for the training and testing of the proposed CNN model.

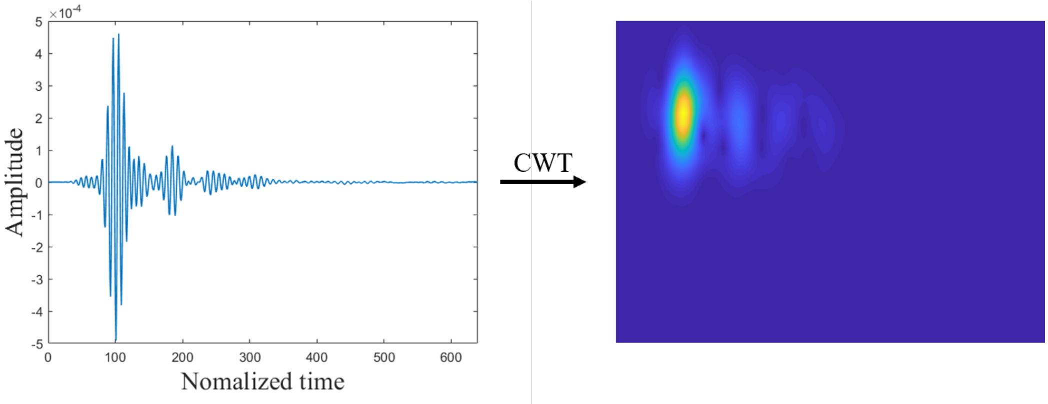


Figure 4.6: Transformation of transient vibration response into time-frequency image by CWT

4.2.1 Training and validation

The proposed CNN architecture is introduced as follows. It contains five convolution layers and pooling layers that are employed to extract key features from time-frequency plane images of the vibration responses of the composite sandwich structure with and without cracks. A fully connected layer and a Softmax classifier are employed to classify the extracted discriminant features through convolution layers and pooling layers into corresponding classes. Detail of the CNN architecture is summarized in Table 4.1.

As described in section 2.4, one healthy model and six crack models are created. CNN is proposed for damage detection as well as damage localization. Therefore, the 2-D images from healthy structure are labelled as "health", while others from crack structures are labelled with corresponding crack number indicating its location: crack 1, crack 2, crack 3, crack 4, crack 5, crack 6. In order to visualize different damages and make the relevant position of each damage clear at a glance, all six damages are illustrated in one diagram, as shown in figure 4.7.

The CNN is used to detect if damages occur in the structure and predict the

Table 4.1: One-stage CNN configuration

Layer	Description
Input	$656 \times 656 \times 3$ CWT images
Conv1, Pool1	Convolution filter 5×5, stride 1, Filter number = 16, Batch normalization, ReLU, Max Pooling filter 2×2, strides 2
Conv2, Pool2	Convolution filter 3×3, stride 1, Filter number = 32, Batch normalization, ReLU, Max Pooling filter 2×2, strides 2
Conv3, Pool3	Convolution filter 3×3, stride 1, Filter number = 64, Batch normalization, ReLU, Max Pooling filter 2×2, strides 2
Conv4, Pool4	Convolution filter 3×3, stride 1, Filter number = 96, Batch normalization, ReLU, Max Pooling filter 2×2, strides 2
Conv5, Pool5	Convolution filter 3×3, stride 1, Filter number = 128, Batch normalization, ReLU, Max Pooling filter 2×2, strides 2
Fully connected softmax	Output = 7, WeightLearnRateFactor = 10, BiasLearnRateFactor = 10

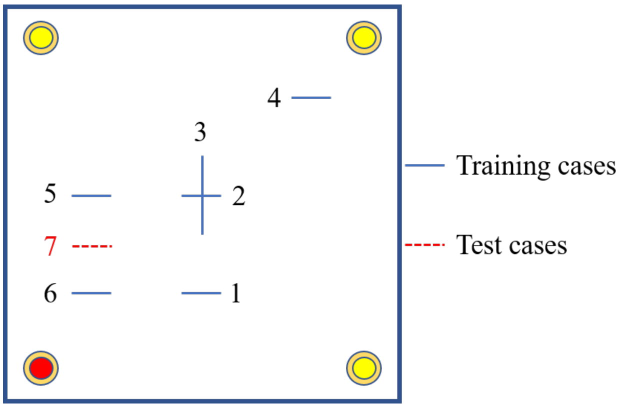


Figure 4.7: Crack damage 1-6 for CNN training, damage 7 for testing CNN

location of the damage at the same time. Therefore, there are 7 classes in the output layer: healthy, crack 1, crack 2, crack 3, crack 4, crack 5, crack 6. A simplified process is shown in figure 4.8. An image with ground truth is fed into the CNN model in which hyperparameters in hidden layers are denoted by Θ. The CNN will give a predictive label of the input image depending on the hyperparameters. A loss function $L(\Theta)$ is defined as a measure of the error between the ground truth and the prediction label, which is a function of the hyperparameters Θ in CNN. The aim is to minimize the error, i.e. the loss function. The loss function is minimized through the optimization process, which is achieved by updating the hyperparameters Θ over a series of epochs until the CNN model is able to classify the image in the correct class.

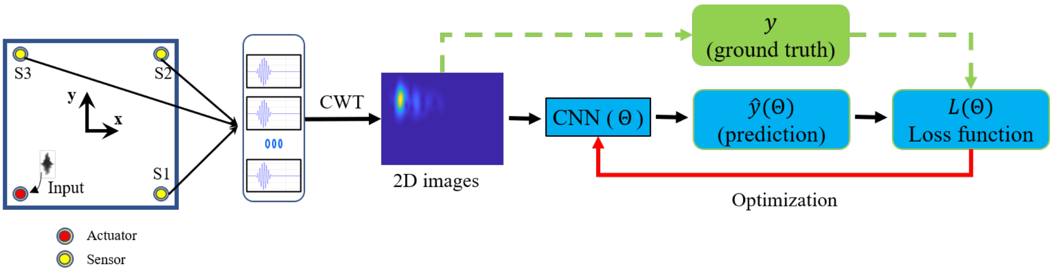


Figure 4.8: CNN training phase

In the training step, we would like to evaluate the model skill after training, but the evaluation on the training dataset may lead to a biased score. Therefore, the database is split into two parts: one part for training, the other part for validation. 90% of the data in the database is randomly selected to train the CNN model, and the rest 10% of the data is for validation. It should be noted that data used for training and validation covers all categories. Adam optimizer is adopted to speed up the calculation, a mini-batch size of 32 is set to train CNN models. The initial learning rate is 0.00001. The training is not stopped until it reaches the max epoch number.

4.2.2 Testing on unseen cases

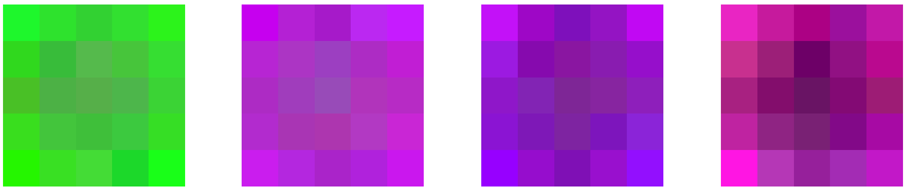
After training the proposed CNN model with training dataset, the trained CNN model is considered to be capable of accurate classification based on extracted discriminant features. One approach to verify the effectiveness of the trained model is to test it with unseen cases, which means the samples for testing are never used during the training phase. In the current study, an unseen case crack 7 is created between crack 5 and crack 6, as shown in figure 4.7. It is utilized to test the trained CNN model.

In order to understand how CNN extracts low level and high level features through hidden layers, some features are shown in figure 4.9. It can be seen that from convolution layer 1 to convolution layer 5, the extracted features become more and more complex, which means that as the number of hidden layers increases, higher level features can be extracted from the input images.

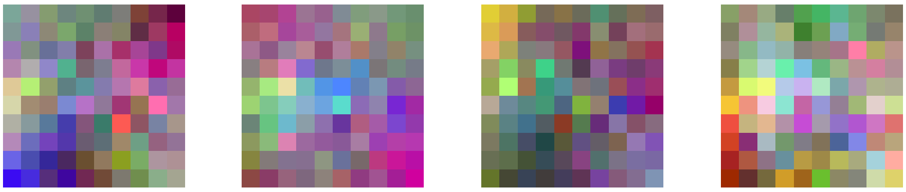
4.2.3 Results and discussions

In the training and validation step, 90% of the data in the database is randomly selected to train the CNN model, and the rest 10% of the data is for validation. It should be noted that data used for training and validation covers all categories. After training, the CNN model is evaluated by the validation samples. Each sample for each category will be given one prediction label by the CNN model. Each prediction label may be either "healthy" or any crack position. A confusion matrix is obtained based on the validation result, as shown in figure 4.10. The row corresponds to the predicted label and the columns correspond to the ground truth (target class). Diagonal cells correspond to correctly classified observations (samples). The off-diagonal cells correspond to observations that are misclassified. The number and percentage in each cell correspond to the number of observations and the percentage of the total number of observations. The rightmost column shows the percentages of all the examples predicted by the CNN model to belong to each class that are correctly and incorrectly classified. The row at the bottom of the matrix shows the proportion of the labels that the network correctly and incorrectly predicts. The cell in the bottom right of the plot shows the overall accuracy. An overall validation accuracy of 91.4% is obtained. The training progress is plotted in figure 4.11. The figure marks each training epoch using a shaded background. An epoch is a full pass through the entire data set. The maximum number of epoch is set as 30. It can be seen that the loss function and training accuracy are both approaching convergence.

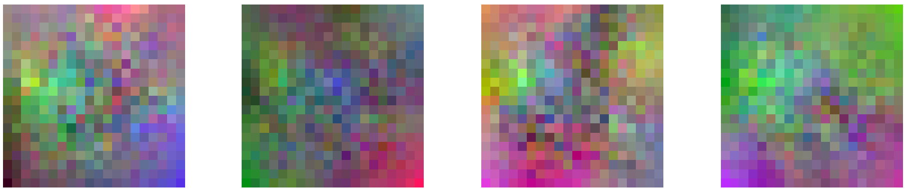
The unseen case crack 7 is used to test the pre-trained one-stage CNN model to verify its effectiveness. The CNN will give a prediction label for every sample of crack 7. A summary of the prediction result is listed in table 4.2. 43.33% instances of crack 7 are classified as "healthy", which occupies the largest portion. 2.22% instances are classified as crack 1, 1.11% instances are classified as crack 2, 2.22% instances are classified as crack 3, 2.22% instances are classified as crack 4. 7.78% instances are classified as crack 5 and 41.11% instances are classified as crack 6, occupying the second largest portion. It should be noted that crack 7 should be classified as crack 5 or crack 6, since the dynamic characteristics of crack 7 should be similar to either 5 or 6 crack case from the physics point of view. As crack 7 is not at the exact middle position between crack 5 and 6, slightly closer to crack 6, more instances are classified as crack 6 than crack 5. The instances that are classified as crack 2 and 3 may help to localize the crack damage area. However, the problem that cannot be ignored is that most of the instances corresponding crack 7 are misclassified as "healthy", which is not acceptable.



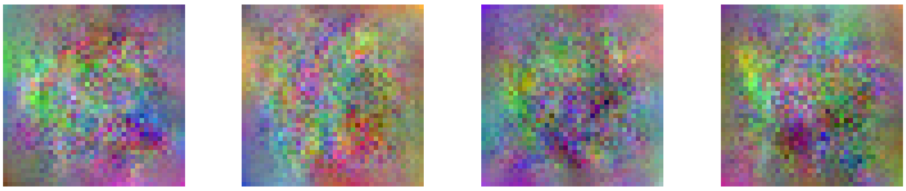
(a) Features extracted by convolution layer 1



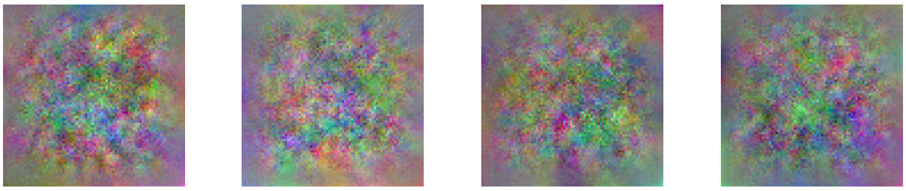
(b) Features extracted by convolution layer 2



(c) Features extracted by convolution layer 3

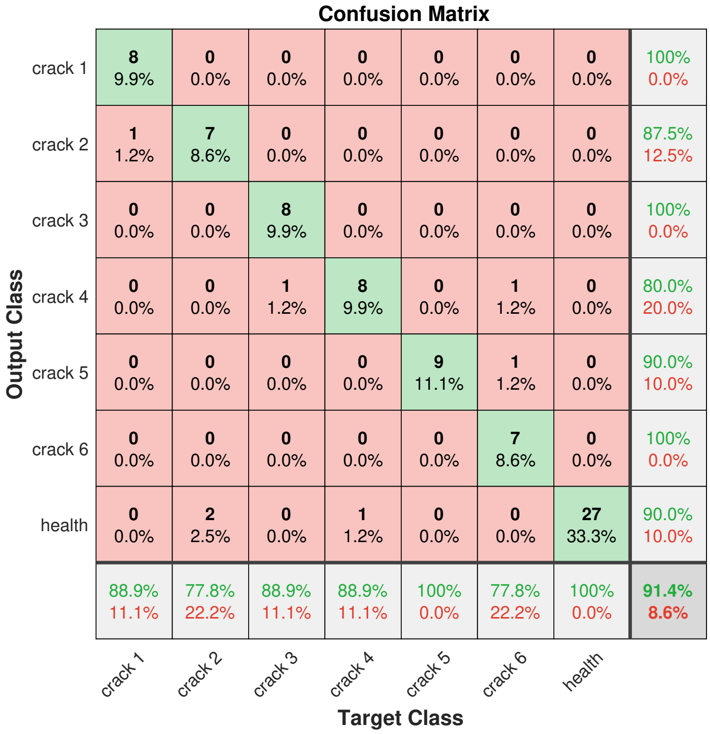


(d) Features extracted by convolution layer 4



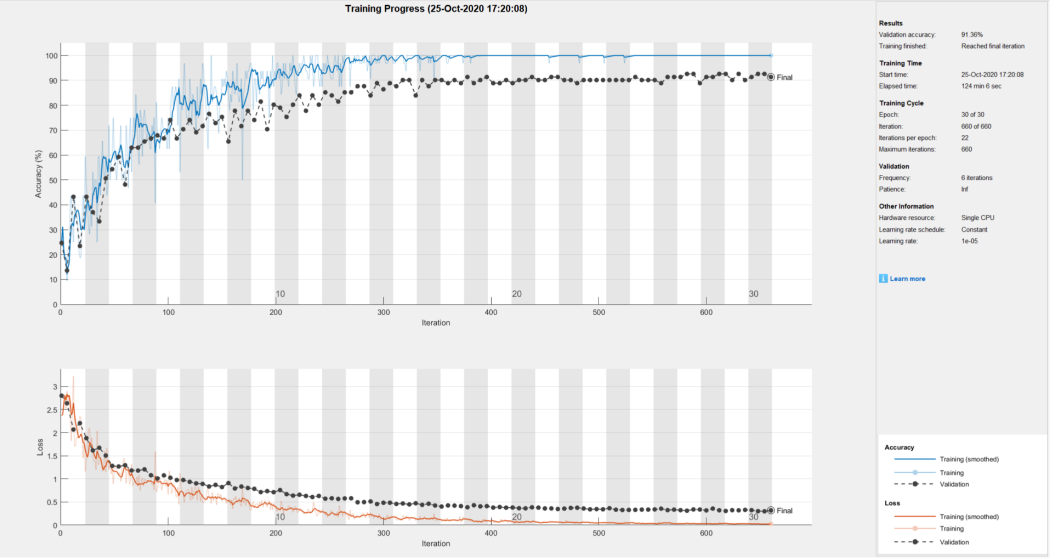
(e) Features extracted by convolution layer 5

Figure 4.9: Samples of features extracted from low level to high level through different convolution layers



	crack 1	crack 2	crack 3	crack 4	crack 5	crack 6	health	
crack 1	**8** 9.9%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	100% 0.0%
crack 2	**1** 1.2%	**7** 8.6%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	87.5% 12.5%
crack 3	**0** 0.0%	**0** 0.0%	**8** 9.9%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	100% 0.0%
crack 4	**0** 0.0%	**0** 0.0%	**1** 1.2%	**8** 9.9%	**0** 0.0%	**1** 1.2%	**0** 0.0%	80.0% 20.0%
crack 5	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**9** 11.1%	**1** 1.2%	**0** 0.0%	90.0% 10.0%
crack 6	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**7** 8.6%	**0** 0.0%	100% 0.0%
health	**0** 0.0%	**2** 2.5%	**0** 0.0%	**1** 1.2%	**0** 0.0%	**0** 0.0%	**27** 33.3%	90.0% 10.0%
	88.9% 11.1%	77.8% 22.2%	88.9% 11.1%	88.9% 11.1%	100% 0.0%	77.8% 22.2%	100% 0.0%	**91.4%** **8.6%**

Figure 4.10: Confusion matrix from validation result of the one-stage CNN



Results
Validation accuracy: 91.36%
Training finished: Reached final iteration

Training Time
Start time: 25-Oct-2020 17:20:08
Elapsed time: 124 min 6 sec

Training Cycle
Epoch: 30 of 30
Iteration: 660 of 660
Iterations per epoch: 22
Maximum iterations: 660

Validation
Frequency: 6 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 4.11: Training progress of one-stage CNN for detection and localization of crack damage

If the instances that are misclassified as "healthy" are not taken into account,

Table 4.2: Labels predicted by pre-trained CNN model for unseen cases of crack damage in the one-stage CNN method

Ground truth	Predicted label	Percentage
Crack 7	Healthy	43.33%
Crack 7	Crack 1	2.22%
Crack 7	Crack 2	1.11%
Crack 7	Crack 3	2.22%
Crack 7	Crack 4	2.22%
Crack 7	Crack 5	7.78%
Crack 7	Crack 6	41.11%

the classification accuracy might be much better. With this assumption, the classification accuracy is recalculated only based on the predictions whose labels are "crack". Thus, the classification result can be redisplayed in table 4.3.

Table 4.3: Redisplayed result based on Table 4.2 by discarding predicted "healthy" labels

Ground truth	Predicted label	Percentage
Crack 7	Crack 1	3.92%
Crack 7	Crack 2	1.96%
Crack 7	Crack 3	3.92%
Crack 7	Crack 4	3.92%
Crack 7	Crack 5	13.73%
Crack 7	Crack 6	72.56%

Although processing data in this way is not scientific and rigorous, it provides a feasible idea to solve the problem. It is necessary to take into account the "healthy" classification labels and meanwhile, improve the classification accuracy. A feasible method is to divide the one-step classification method into two steps. That is to say, in the first step, the database is split into two classes: healthy and damaged. They can be used to train a CNN model to have an output with only two classes: healthy and crack. This is to detect the occurrence of crack in the structure. Once data for testing is classified as "crack", we proceed to the second step: remove the health label data in the database, and split the remaining data into different categories according to their corresponding crack positions, i.e. 6 crack cases in the present study. Train another CNN model with these data that do not contain any samples representing healthy case. Thus, the classification output will have 6 classes: crack 1, crack 2, ..., crack 6. Therefore, the classification output in the second step will not include "healthy" label and the impact of healthy data can be eliminated. A possible approach to realize this idea is to build a two-stage CNN-based crack detection model: coarse to fine classification, which will be explained in next section.

4.3 Two-stage CNN: coarse to fine classification

In this study, a two-stage CNN is proposed to detect the occurrence and localization of the crack damage. The basic idea behind the two-stage CNN is that, a first CNN is constructed and dedicated to detecting whether crack damage occurs in the structure. This step is named as coarse classification. After fault-pattern recognition, a second CNN with the same architecture as the first one is constructed to predict the location of the crack damage. In this step, data for training the CNN model contain those corresponding to different crack cases. An overview of data pre-processing and the proposed two-stage CNN-based crack detection algorithm is illustrated in figure 4.12.

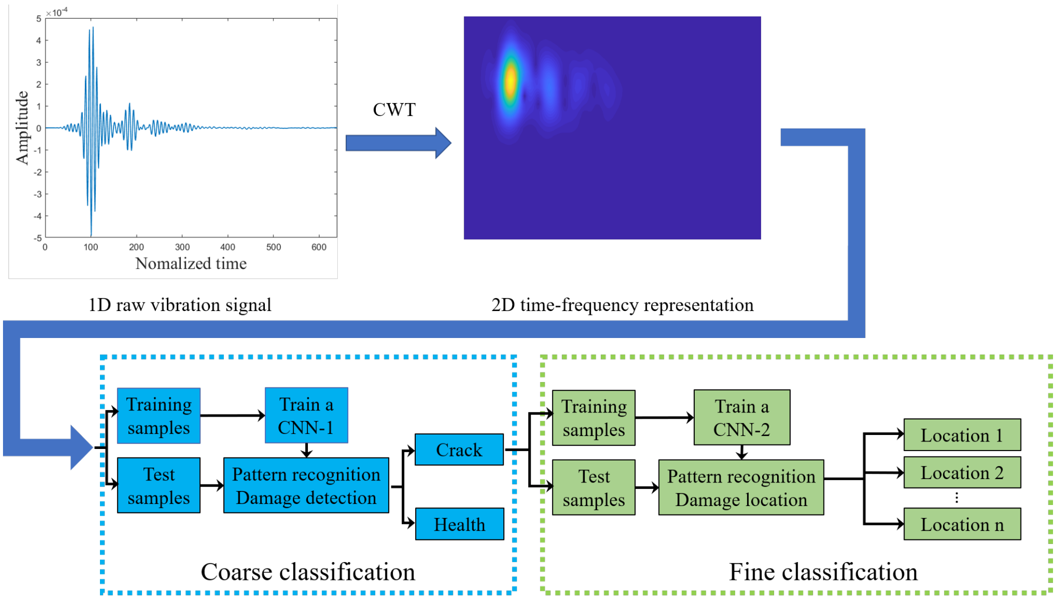


Figure 4.12: Overview of data pre-processing and the proposed two-stage CNN-based crack detection algorithm

4.3.1 Coarse classification

In the first step of the two-stage CNN-based crack detection algorithm: coarse classification, damage cases are the same as those in section 4.2. The same database as in section 4.2 is used for training and testing the CNN-1 model. However, it should be mentioned that in coarse classification, the aim is to determine if crack damage occurs in the structure. Thus, the database is split into two categories: healthy and crack, where the crack category contains all data corresponding to the six crack cases. In consequence, the output of the classifier in CNN-1 has only two classes: healthy and crack. The architecture of CNN-1 is the same as the one used in Table 4.1 except that the number of classes in the output layer is 2 instead of 7. An appropriate value of maximum epoch number in the training process allows to save computing time while ensuring the convergence of the results. In coarse classification it is set as 25. An iteration corresponds to a mini-batch. The validation frequency value corresponds to the number of iterations between evaluations of validation metrics.

A confusion matrix is obtained based on the validation results as shown in figure 4.13. It is shown that all samples with ground truth label "healthy" are correctly classified as "healthy" and all samples with ground truth label "crack" are also correctly classified, which suggests that the trained network performs well enough to be further used for testing with unseen cases. The training progress is plotted in figure 4.14. The validation loss gradually decreases and tends to converge while the validation accuracy gradually grows and tends to converge.

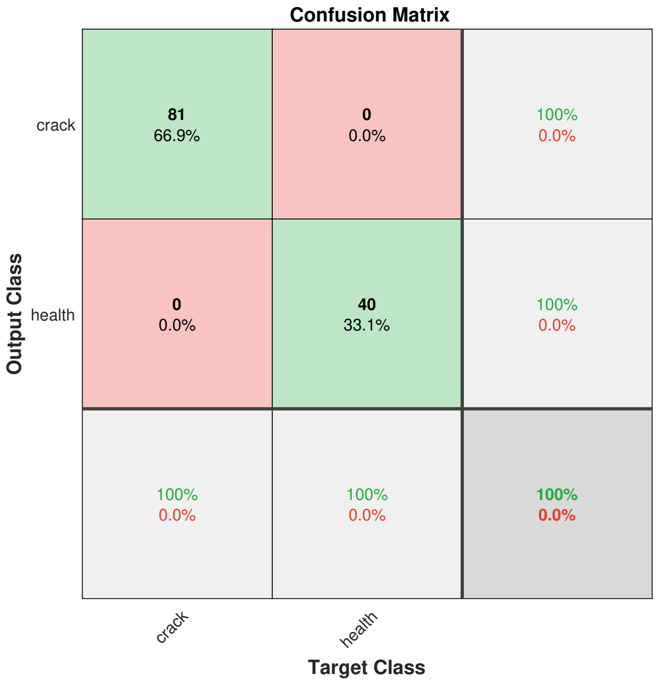
The unseen case crack 7 is then used as testing sample to verify the effectiveness of the pre-trained CNN-1 model. The prediction results are listed in table 4.4, where every single sample is correctly predicted with a label "crack". The result suggests that in the present condition, CNN-1 model is capable of predicting the occurrence of crack damages in the structure. Then we can proceed to the second step, fine classification by CNN-2 model for the localization of the crack damage in the structure.

Table 4.4: Labels predicted by pre-trained CNN-1 for unseen cases of crack damage in the coarse classification of the two-stage CNN method

Ground truth	Predicted label	Percentage
Crack 7	Healthy	0%
Crack 7	Crack	100%

4.3.2 Fine classification

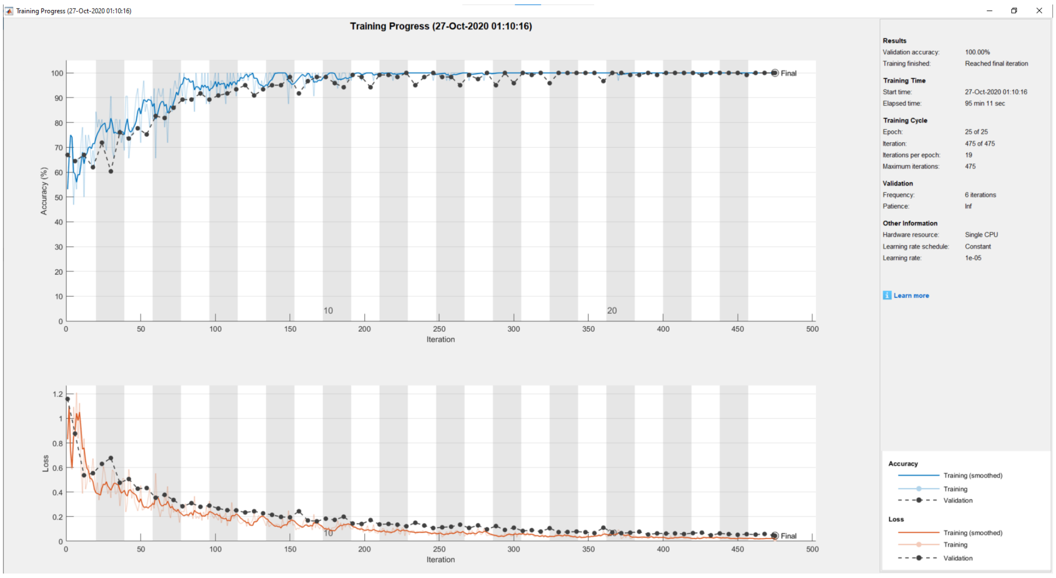
Following the fault-pattern recognition in coarse classification which is dedicated to detecting the occurrence of crack damage in the composite sandwich structure, another CNN model denoted by CNN-2 is proposed to localize the crack damage



Confusion Matrix

Output Class \ Target Class	crack	health	
crack	**81** 66.9%	**0** 0.0%	100% 0.0%
health	**0** 0.0%	**40** 33.1%	100% 0.0%
	100% 0.0%	100% 0.0%	**100%** **0.0%**

Figure 4.13: Confusion matrix from validation result of CNN-1 in the two-stage CNN



Results
Validation accuracy: 100.00%
Training finished: Reached final iteration

Training Time
Start time: 27-Oct-2020 01:10:16
Elapsed time: 95 min 11 sec

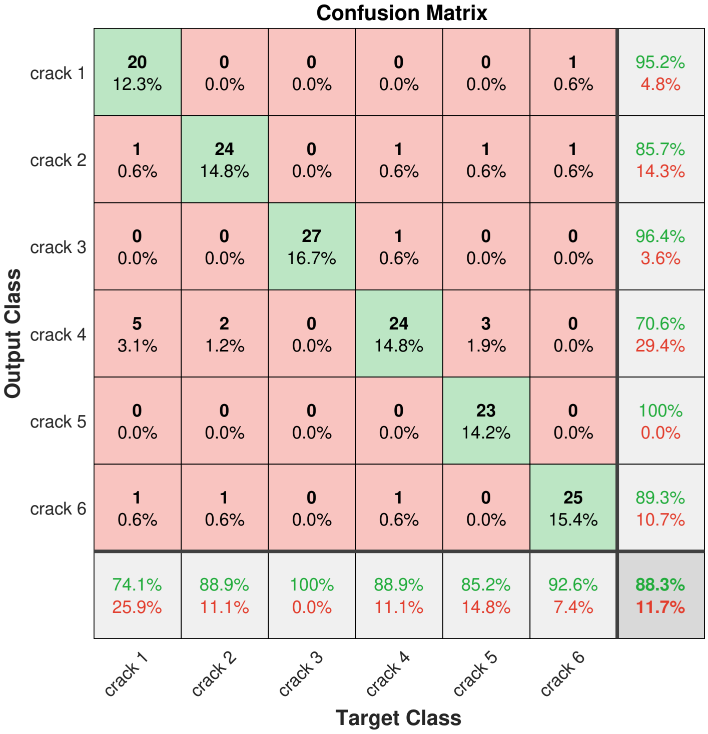
Training Cycle
Epoch: 25 of 25
Iteration: 475 of 475
Iterations per epoch: 19
Maximum iterations: 475

Validation
Frequency: 6 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 4.14: Training progress of CNN-1 in the two-stage CNN for crack damage detection

in the structure. In this step, data in the database corresponding to crack 1, crack 2, ..., and crack 6 are used to train the CNN-2 model, while data corresponding to healthy structure is excluded. In consequence, the output layer has six classes corresponding to different positions, instead of two classes "healthy" and "crack" in the coarse classification. The rest of the architecture of CNN-2 is the same as the previous CNN-1.

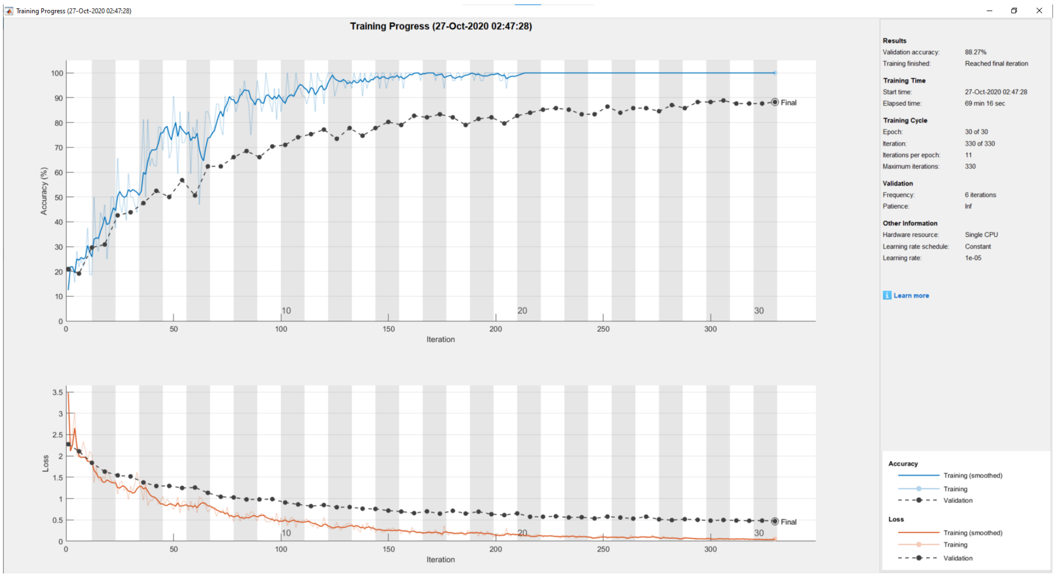


	crack 1	crack 2	crack 3	crack 4	crack 5	crack 6	
crack 1	**20** 12.3%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**1** 0.6%	95.2% 4.8%
crack 2	**1** 0.6%	**24** 14.8%	**0** 0.0%	**1** 0.6%	**1** 0.6%	**1** 0.6%	85.7% 14.3%
crack 3	**0** 0.0%	**0** 0.0%	**27** 16.7%	**1** 0.6%	**0** 0.0%	**0** 0.0%	96.4% 3.6%
crack 4	**5** 3.1%	**2** 1.2%	**0** 0.0%	**24** 14.8%	**3** 1.9%	**0** 0.0%	70.6% 29.4%
crack 5	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**23** 14.2%	**0** 0.0%	100% 0.0%
crack 6	**1** 0.6%	**1** 0.6%	**0** 0.0%	**1** 0.6%	**0** 0.0%	**25** 15.4%	89.3% 10.7%
	74.1% 25.9%	88.9% 11.1%	100% 0.0%	88.9% 11.1%	85.2% 14.8%	92.6% 7.4%	**88.3%** **11.7%**

Figure 4.15: Confusion matrix from validation result of CNN-2 in the two-stage CNN

In fine classification training phase, the maximum epoch number is set as 30. The validation frequency is 6 iterations, the same as in coarse classification. A confusion matrix based on the validation result is obtained, as shown in figure 4.15. The classification accuracy of crack 1, ... and crack 6 in data for validation are 74.1%, 88.9%, 100%, 88.9%, 85.2%, 92.6%, respectively. An overall validation accuracy of 88.3% is obtained. The training process is plotted in figure 4.16. The loss function and training accuracy together with validation accuracy tend to converge.

After model validation, the unseen case 7 is used to test the performance of CNN-2 model in reality. A summary of the prediction result is listed in table 4.5. 3.33% instances of crack 7 are classified as crack 1, 3.33% instances are classified as crack 2, 20.00% instances are classified as crack 3, and 10.00% instances are classified as crack 4. 36.67% instances are classified as crack 5, occupying the largest portion



Results
Validation accuracy: 88.27%
Training finished: Reached final iteration

Training Time
Start time: 27-Oct-2020 02:47:28
Elapsed time: 69 min 16 sec

Training Cycle
Epoch: 30 of 30
Iteration: 330 of 330
Iterations per epoch: 11
Maximum iterations: 330

Validation
Frequency: 6 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 4.16: Training progress of CNN-2 in the two-stage CNN for crack damage localization

in the prediction result, and 26.67% instances are classified as crack 6, the second largest portion in the prediction result.

Table 4.5: Labels predicted by pre-trained CNN-2 for unseen cases of crack damage in the fine classification of the two-stage CNN method

Ground truth	Predicted label	Percentage
Crack 7	Crack 1	3.33%
Crack 7	Crack 2	3.33%
Crack 7	Crack 3	20.00%
Crack 7	Crack 4	10.00%
Crack 7	Crack 5	36.67%
Crack 7	Crack 6	26.67%

4.3.3 Discussions

For the coarse classification by CNN-1, it can correctly predict an unseen case to be damaged or not with a classification accuracy of 100%. It is clear that CNN-1 has the capability of distinguishing crack and healthy cases, although it cannot yet predict the damage area. Therefore, it is supposed to be able to detect if crack damage occurs in the structure for other unseen cases.

As for CNN-2, which is dedicated to localizing the crack damage, the prediction portion of crack 5 and crack 6 for the unseen case 7 are 36.67% and 26.67%, respectively, occupying the largest and second largest portion among others. As mentioned in section 4.2.3, the case of crack 7 being classified as crack 5 and 6 is consistent with the physics of the problem because the dynamic characteristics of crack 7 should be similar to either crack 5 or crack 6 case in a location near crack 7.

The analysis of the testing result of CNN-2 can be viewed from two aspects: physics and machine learning aspects. From a physics point of view, a total of 63.34% samples classified as crack 5 and crack 6 can directly provide a clear guidance to determine the approximate location of the crack damage in the structure, which outperforms the one-stage CNN-based crack detection model in section 4.2 that misclassifies most of the unseen crack 7 samples as "healthy". It is a significant result and can provide a useful reference for future decision-making, such as using traditional methods to scan and detect the damaged area more precisely, repair the damaged component, or replace the damaged part with healthy component, etc. However, from a machine learning point of view, although the results can provide a useful reference, the accuracy of 63.34% is not satisfying. It should be understood that in the health monitoring of structures, especially for structures that play an important role in a system, the reliability of the testing result is crucial. Therefore, improving the classification accuracy is always the constant pursuit.

It should also be noted that in the current database that is used to train the network, the number of crack cases is too limited. It is expected that the database can cover as many cases as possible so that the CNN model can still provide reliable prediction results when dealing with more unknown cases. Accordingly, the enrichment of the database is proposed by enriching crack cases in the structure. It will be discussed in detail in the next section.

4.4 Two-stage CNN with enriched database

The enrichment of the database is achieved by creating more crack damages which can cover the entire composite sandwich structure instead of limited and unevenly distributed zones in figure 4.7. The added crack cases are shown in figure 4.17. In this figure, the number of each crack is rearranged for the sake of description. Compared to the precedent section, four crack cases are added, i.e. crack 1, crack 2, crack 6 and crack 9. Cracks 1-9 are almost evenly distributed in the structure. Two other crack cases, crack 12 and crack 13, are created between crack 5 and crack 8, crack 6 and crack 9, respectively. Raw vibration response signals corresponding to different crack cases are pre-processed by CWT in the same way as previously. Data corresponding to crack 1 to crack 13 together with healthy data constitute the enriched database. All data except crack 11, crack 12 and crack 13 in the enriched database are used as training cases, while data corresponding to crack 11 to crack 13 are used as test cases.

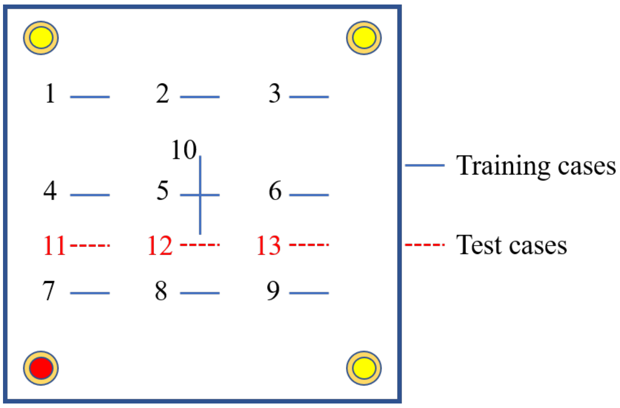
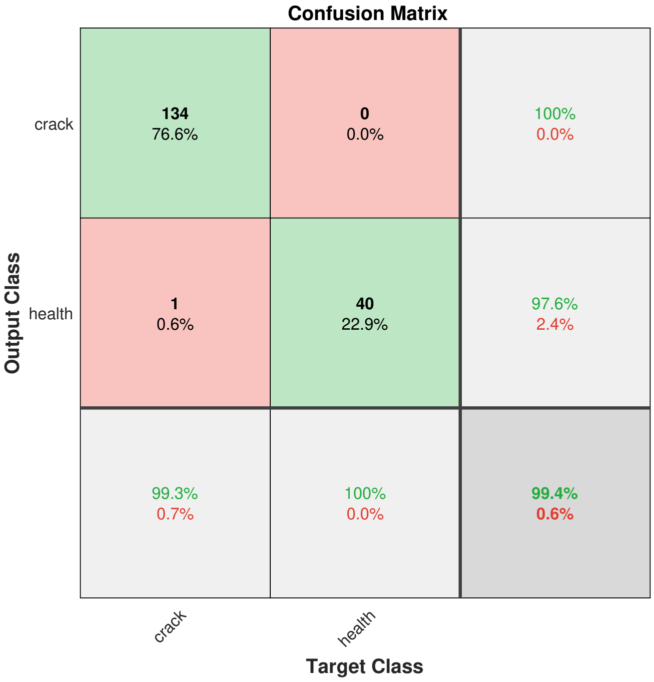


Figure 4.17: Crack damage 1-10 for CNN training, damage 11-13 for testing CNN

4.4.1 Coarse classification

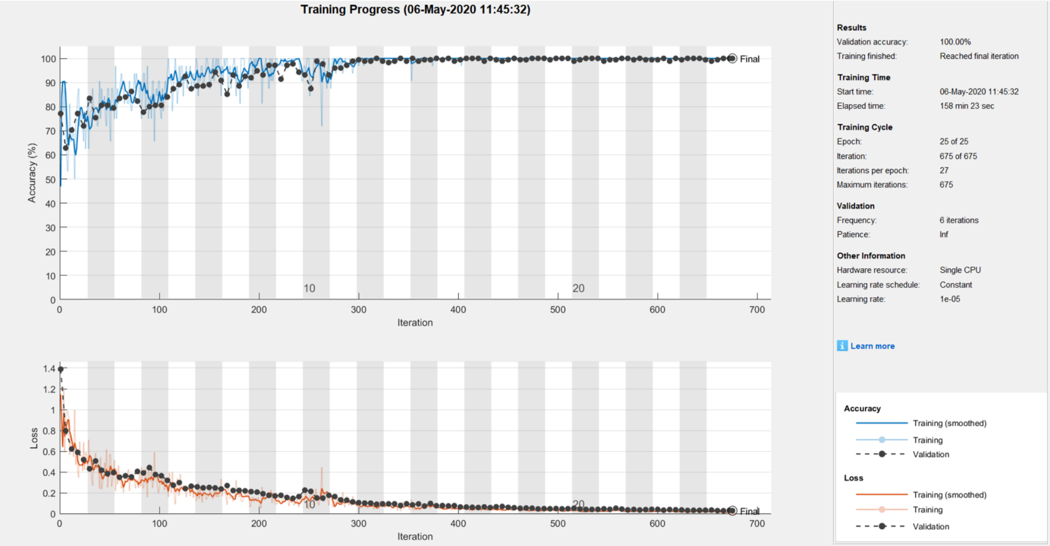
In the first step, coarse classification, data corresponding to crack 1 to crack 10 are combined with only one label "crack". Data corresponding to healthy case together with crack data from crack 1 to 10 are used to train CNN-1 model where 90% are training samples and the other 10% are validation samples. It should be pointed out that the CNN-1 model has the same architecture as that in section 4.3.1 but it is not the pre-trained model. The number of maximum epochs is set as 25 so that the training and validation accuracy together with the loss function can converge. The validation frequency is set as 6 iterations. A confusion matrix is obtained based on the validation result, as shown in figure 4.18. 99.3% validation samples with ground truth label "crack" are correctly classified by the trained CNN-1 and all validation samples with ground truth label "health" are correctly classified. An overall classification accuracy of 99.4% is obtained. The training process is plotted in figure 4.19.

After model validation, data corresponding to unseen cases crack 11, 12 and 13 are combined with only one label "crack" to test the effectiveness of the pre-trained CNN-1 model. The output of CNN-1 has two classes: healthy and crack. The prediction results are listed in table 4.6. 94.81% of the samples are correctly classified as crack, although 5.19% are misclassified as "healthy" case. The classification accuracy is slightly lower than that before database enrichment in the previous section, but it is still convincing enough to predict the occurrence of crack damage in the composite sandwich structure. Then we proceed to the next step, fine classification to predict the location of unseen crack damages.



Output Class \ Target Class	crack	health	
crack	134 76.6%	0 0.0%	100% 0.0%
health	1 0.6%	40 22.9%	97.6% 2.4%
	99.3% 0.7%	100% 0.0%	99.4% 0.6%

Figure 4.18: Confusion matrix from validation result of CNN-1 in the two-stage CNN with enriched database



Results

Validation accuracy:	100.00%
Training finished:	Reached final iteration

Training Time

Start time:	06-May-2020 11:45:32
Elapsed time:	158 min 23 sec

Training Cycle

Epoch:	25 of 25
Iteration:	675 of 675
Iterations per epoch:	27
Maximum iterations:	675

Validation

Frequency:	6 iterations
Patience:	Inf

Other Information

Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	1e-05

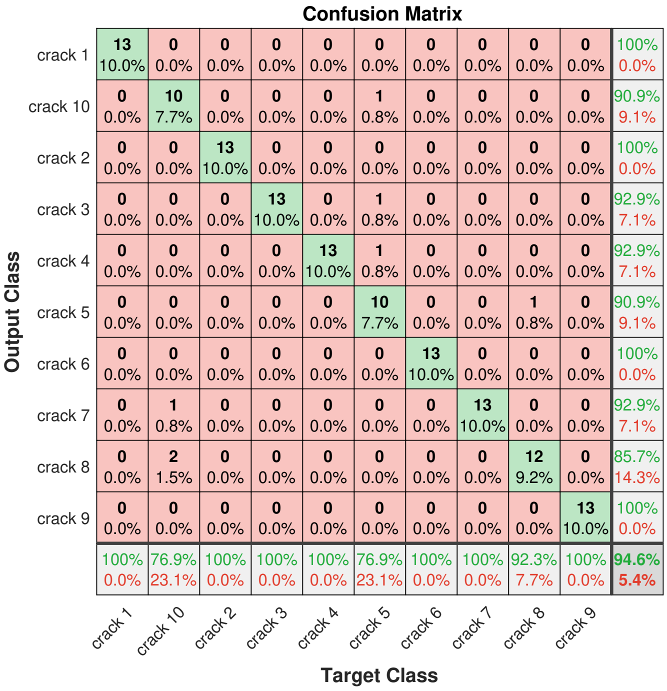
Figure 4.19: Training progress of CNN-1 in the two-stage CNN for crack damage detection with enriched database

Table 4.6: Labels predicted by pre-trained CNN-1 for unseen cases of crack damage in the coarse classification of the two-stage CNN method with enriched database

Ground truth	Predicted label	Percentage
Crack	Healthy	5.19%
Crack	Crack	94.81%

4.4.2 Fine classification

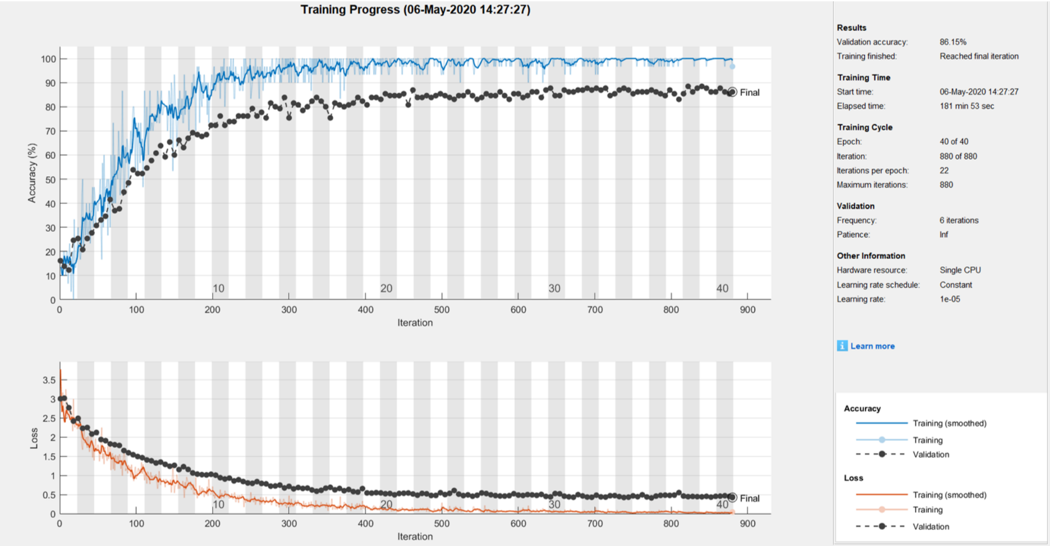
After fault-pattern recognition in the coarse classification with CNN-1 for detecting the occurrence of crack damage, another CNN model denoted by CNN-2 with the same architecture as CNN-1 is proposed to predict the location of unseen crack damages. In this step, data for training CNN-2 model include only data corresponding to crack 1 to crack 10. Thus, the output of CNN-2 has 10 classes, representing crack 1, crack 2, ... and crack 10, respectively. In fine classification, 70% data are used to train the CNN-2 model while the other 30% are used for validation. The maximum epoch number is set as 40 so that the training and validation accuracy together with loss function can converge.



Output Class \ Target Class	crack 1	crack 10	crack 2	crack 3	crack 4	crack 5	crack 6	crack 7	crack 8	crack 9	
crack 1	13 10.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
crack 10	0 0.0%	10 7.7%	0 0.0%	0 0.0%	0 0.0%	1 0.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	90.9% 9.1%
crack 2	0 0.0%	0 0.0%	13 10.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
crack 3	0 0.0%	0 0.0%	0 0.0%	13 10.0%	0 0.0%	1 0.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	92.9% 7.1%
crack 4	0 0.0%	0 0.0%	0 0.0%	0 0.0%	13 10.0%	1 0.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	92.9% 7.1%
crack 5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	10 7.7%	0 0.0%	0 0.0%	1 0.8%	0 0.0%	90.9% 9.1%
crack 6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	13 10.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
crack 7	0 0.0%	1 0.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	13 10.0%	0 0.0%	0 0.0%	92.9% 7.1%
crack 8	0 0.0%	2 1.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	12 9.2%	0 0.0%	85.7% 14.3%
crack 9	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	13 10.0%	100% 0.0%
	100% 0.0%	76.9% 23.1%	100% 0.0%	100% 0.0%	100% 0.0%	76.9% 23.1%	100% 0.0%	100% 0.0%	92.3% 7.7%	100% 0.0%	94.6% 5.4%

Figure 4.20: Confusion matrix from validation result of CNN-2 in the two-stage CNN with enriched database

A confusion matrix is obtained based on the validation result, as shown in figure 4.20. The classification accuracy of crack 1, crack 2, crack 3, crack 4, crack 6, crack 7 and crack 9 in the validation phase are 100%. The classification accuracy of crack 5, crack 8 and crack 10 are 76.9%, 92.3% and 76.9%, respectively. An overall validation accuracy of 94.6% is obtained. The training process is plotted in figure 4.21. The loss function and training accuracy together with validation accuracy tend to converge.



Results
Validation accuracy: 86.15%
Training finished: Reached final iteration

Training Time
Start time: 06-May-2020 14:27:27
Elapsed time: 181 min 53 sec

Training Cycle
Epoch: 40 of 40
Iteration: 880 of 880
Iterations per epoch: 22
Maximum iterations: 880

Validation
Frequency: 6 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 4.21: Training progress of CNN-2 in the two-stage CNN for crack damage detection with enriched database

After model validation, the pre-trained CNN-2 classifier is employed to predict labels for unseen cases of crack 11, 12 and 13 indicating their locations. As the unseen case crack 11 is designed between crack 4 and crack 7, it should have similar dynamic characteristics either to crack 4 or crack 7. Thus, it is expected that the predicted label for the unseen crack 11 is crack 4 or crack 7. It should be the same for the other two unseen cases crack 12 and crack 13. Samples for crack 11, 12 and 13 are labelled with the corresponding crack locations and are fed into the pre-trained CNN-2 classifier. Predictions given by CNN-2 are listed in table 4.7, 4.8 and 4.9.

For crack 11 in table 4.7, 37.78% samples are classified as crack 7, which occupied the largest portion in the predicted labels, and 5.56% samples are classified as crack 4. Crack 7 and crack 4 are located around the unseen case crack 11. 24.44% and 21.11% samples are classified as crack 6 and crack 1, respectively. The percentage of samples classified as crack 8, crack 9 and crack 10 are relatively small, 3.33% , 5.56% and 2.22%, respectively.

For crack 12 in table 4.8, the percentage of samples classified as crack 5, crack

Table 4.7: Labels predicted by pre-trained CNN-2 for unseen case of crack 11 in the fine classification of the two-stage CNN method with enriched database

Ground truth	Predicted label	Percentage
Crack 11	Crack 1	21.11%
Crack 11	Crack 2	0.00%
Crack 11	Crack 3	0.00%
Crack 11	Crack 4	5.56%
Crack 11	Crack 5	0.00%
Crack 11	Crack 6	24.44%
Crack 11	Crack 7	37.78%
Crack 11	Crack 8	3.33%
Crack 11	Crack 9	5.56%
Crack 11	Crack 10	2.22%

Table 4.8: Labels predicted by pre-trained CNN-2 for unseen case of crack 12 in the fine classification of the two-stage CNN method with enriched database

Ground truth	Predicted label	Percentage
Crack 12	Crack 1	3.33%
Crack 12	Crack 2	18.89%
Crack 12	Crack 3	12.22%
Crack 12	Crack 4	5.56%
Crack 12	Crack 5	21.11%
Crack 12	Crack 6	0.00%
Crack 12	Crack 7	11.11%
Crack 12	Crack 8	20.00%
Crack 12	Crack 9	7.78%
Crack 12	Crack 10	0.00%

8 are 21.11% and 20.00% respectively, which occupy the first largest portions in all samples. They are the cases nearest to the unseen case. 18.89% samples are classified as crack 2, 12.22% samples are classified as crack 3, 11.11% samples are classified as crack 7, 7.78% samples are classified as crack 9, 5.56% samples are classified as crack 4 and 3.33% samples are classified as crack 1.

For crack 13 in table 4.9, there are 34.44% samples classified as crack 6 and 30% samples classified as crack 9. From figure 4.17, the unseen case crack 13 is located between crack 6 and crack 9. The percentage of samples classified as crack 3 and

Table 4.9: Labels predicted by pre-trained CNN-2 for unseen case of crack 13 in the fine classification of the two-stage CNN method with enriched database

Ground truth	Predicted label	Percentage
Crack 13	Crack 1	4.44%
Crack 13	Crack 2	0.00%
Crack 13	Crack 3	13.33%
Crack 13	Crack 4	0.00%
Crack 13	Crack 5	2.22%
Crack 13	Crack 6	34.44%
Crack 13	Crack 7	10.00%
Crack 13	Crack 8	5.56%
Crack 13	Crack 9	30.00%
Crack 13	Crack 10	0.00%

crack 7 are 13.33% and 10.00%. Predicted labels as crack 1, crack 5 and crack 8 occupy 4.44%, 2.22% and 5.56%, respectively.

4.4.3 Discussions

The classification accuracy of unseen cases crack 11, 12 and 13 can reach 94.81% by classifier CNN-1 in the proposed two-stage CNN-based method for crack damage detection in the composite sandwich structure with enriched database. This result shows the damage detection capability of the classifier.

On the contrary, the classifier CNN-2 performs much worse after database enrichment. Compared to the result in section 4.3.2, where the total percentage of samples classified as crack 4 and crack 7 is 63.34% for the unseen case crack 11, the classification accuracy after database enrichment is only 43.34% in this section, which decreases dramatically. Although the result for crack 13 can reach 64.44%, it can't be ignored that the accuracy of crack 12 is only 41.11%.

On the one hand, these classification accuracy rates such as 43.34%, 64.44% and 41.11% can provide a certain reference for future decision-making; but on the other hand, for some cases, the percentage of samples that are correctly classified is not much different from the percentage of samples that are incorrectly classified, which may mislead decision-making. Especially if the damage approximate location information is not known in advance, the crack location cannot be determined reliably. In addition, the performance of CNN-2 classifier is not stable according to the classification accuracy for crack 11, crack 12 and crack 13, as the classification accuracy fluctuates from around 40% to around 60%.

In deep learning algorithms, the classification performance on simulation-based dataset is essential for further applications. The robustness of the classifier should be guaranteed since when deployed in real environments, small perturbations on the samples should not cause significant loss of performance.

The performance of the classifier CNN-2 on unseen cases suggests that when perturbations are applied on the test samples, the classifier cannot make stable predictions accordingly. However, when the database is not enriched in section 4.3, the performance of the classifier CNN-2 is much better. This can be explained by the fact that when there are few cases in the database, the distribution of crack damage is more scattered, and the number of predictable classification labels is also small, the classifier is more inclined to classify the test samples into a category that is more similar to it, that is, the crack cases near it. When the number of cases in the database is appropriately increased, there are more classification labels. At this time, the variability of the database is more important, because the classifier needs to identify the matching test sample among many categories. If a small disturbance is applied to the test sample, the classifier cannot distinguish which category the sample should be classified into, then the variability of the database is not sufficient. Therefore, it is very important to increase the variability of the database so as to improve the performance and robustness of the classifier.

4.5 Two-stage CNN with further enriched database

To address the previously mentioned issue, namely the lack of variability of the database, we try to increase the database variability by further enrichment of the database based on the database in section 4.4. The enrichment is achieved by creating more damage cases that are more densely distributed in the structure. In this section, the structure is divided into nine zones, each of which represents one location of crack damages. We enrich crack cases in each zone, and cases in the same zone have a same label that indicates the crack location, as shown in figure 4.22. The enriched crack cases in each zone have the same size and are equidistantly distributed in the vertical direction, approximately 17.32 mm between two adjacent cracks. In this way, the variability of the database can be increased.

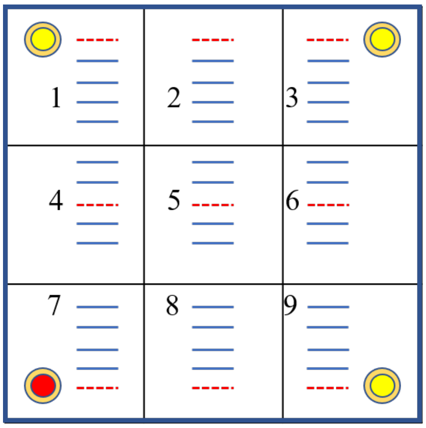
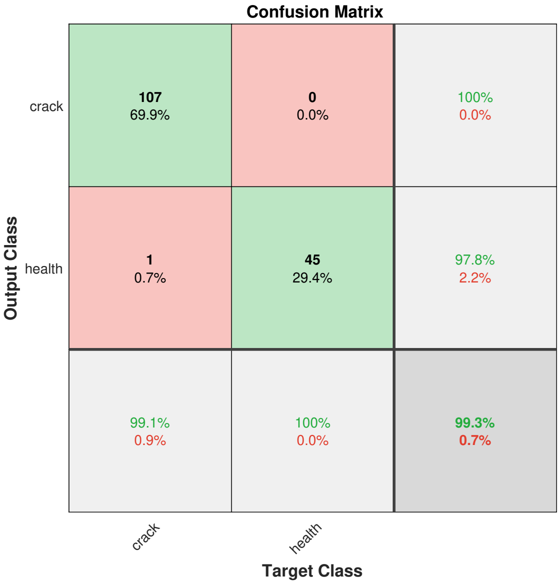


Figure 4.22: Further enriched crack damage cases. Damage denoted by blue solid line for CNN training, damage denoted by red dash line for CNN testing

After database enrichment, 4/5 crack cases in each zone are used to train the proposed two-stage CNN model and the rest 1/5 cases are used as test sample to evaluate the effectiveness of the two-stage CNN model in crack damage detection and localization.

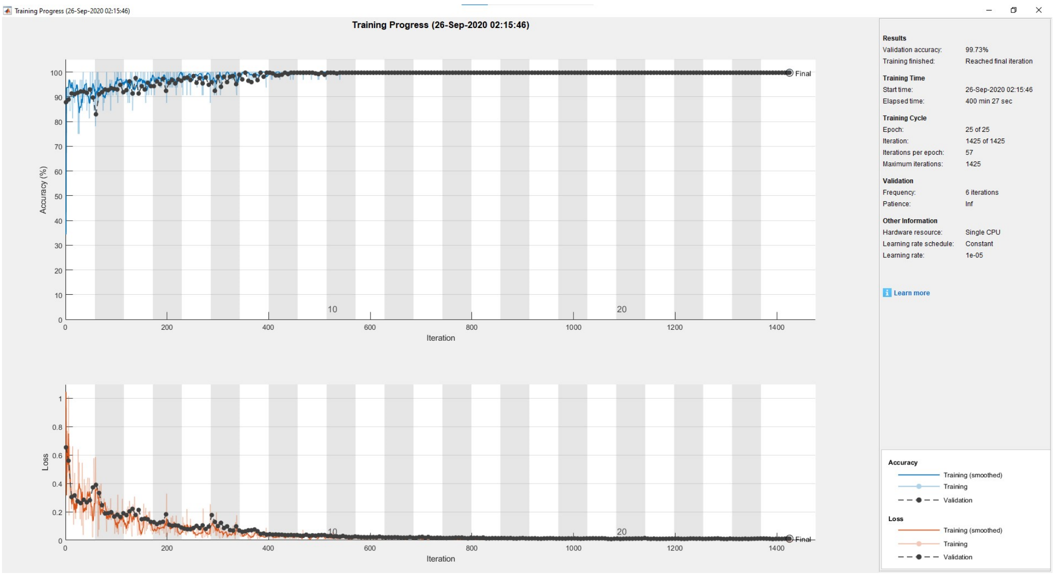
4.5.1 Coarse classification

In the first stage of the two-stage CNN-based crack damage detection and localization algorithm, coarse classification, crack cases that are used as training samples are combined and labelled as "crack", while the healthy case is labelled as "health". A first neural network CNN-1 is constructed to detect the occurrence of crack damage. The output layer has two classes: health and crack. The maximum epoch number is set as 15 with a validation frequency of 6 iterations. Based on the validation result, a confusion matrix is obtained showing the effectiveness of the trained CNN-1 on validation samples, as shown in figure 4.23. 99.1% samples with ground truth label "crack" are correctly classified and the classification accuracy for all samples with ground truth "health" is 100%. The training process is plotted in figure 4.24.



Output Class \ Target Class	crack	health	
crack	107 69.9%	0 0.0%	100% 0.0%
health	1 0.7%	45 29.4%	97.8% 2.2%
	99.1% 0.9%	100% 0.0%	**99.3%** **0.7%**

Figure 4.23: Confusion matrix from validation result of CNN-1 in the two-stage CNN with further enriched database



Results
Validation accuracy: 99.73%
Training finished: Reached final iteration

Training Time
Start time: 26-Sep-2020 02:15:46
Elapsed time: 400 min 27 sec

Training Cycle
Epoch: 25 of 25
Iteration: 1425 of 1425
Iterations per epoch: 57
Maximum iterations: 1425

Validation
Frequency: 6 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 4.24: Training progress of CNN-1 in the two-stage CNN for crack damage detection with further enriched database

After CNN-1 model validation, unseen crack cases in each zone are combined as test samples to evaluate the effectiveness of the pre-trained model on detecting the occurrence of crack damage in the structure. It is expected that labels predicted by CNN-1 for test samples are the same as the ground truth of test samples: "crack". Prediction results are listed in table 4.10. All test samples can be correctly predicted as "crack" as expected, which shows the capability of CNN-1 in detecting the occurrence of crack damage in the structure. Then we proceed to the next step to determine the location of the unseen crack.

Table 4.10: Labels predicted by pre-trained CNN-1 for unseen cases of crack damage in the coarse classification of the two-stage CNN method with further enriched database

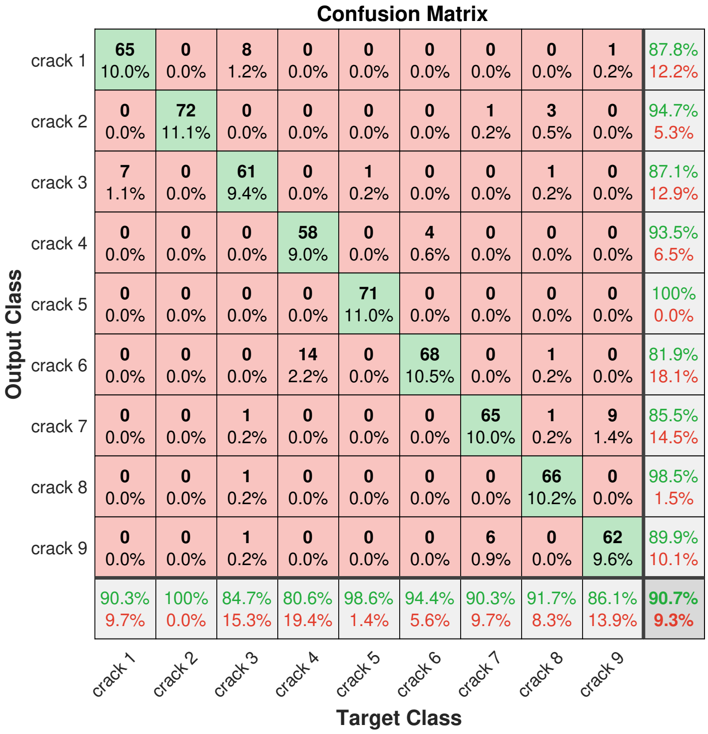
Ground truth	Predicted label	Percentage
Crack	Healthy	0%
Crack	Crack	100%

4.5.2 Fine classification

After fault-pattern recognition by CNN-1 in coarse classification to detect the occurrence of crack damage in the structure, a second CNN model denoted by CNN-2 with the same architecture as CNN-1 except the output layer is constructed to localize the crack damage. Datasets representing nine crack locations are used to train the CNN-2 and test its effectiveness, given that with this enriched database the variability of the database should be increased. The output layer of CNN-2 has nine classes, i.e., crack 1, crack 2, ..., crack 9.

In the fine classification training phase, the maximum epoch number is set as 15 such that the training process can converge. The validation frequency is set as 6 iterations, the same as before. A confusion matrix based on the validation result is obtained indicating the performance of CNN-2 on the validation samples, as shown in figure 4.25. The lowest classification accuracy is 84.7%, corresponding to crack 3, but an overall average accuracy is obtained as 90.7%. The training process is plotted in figure 4.26.

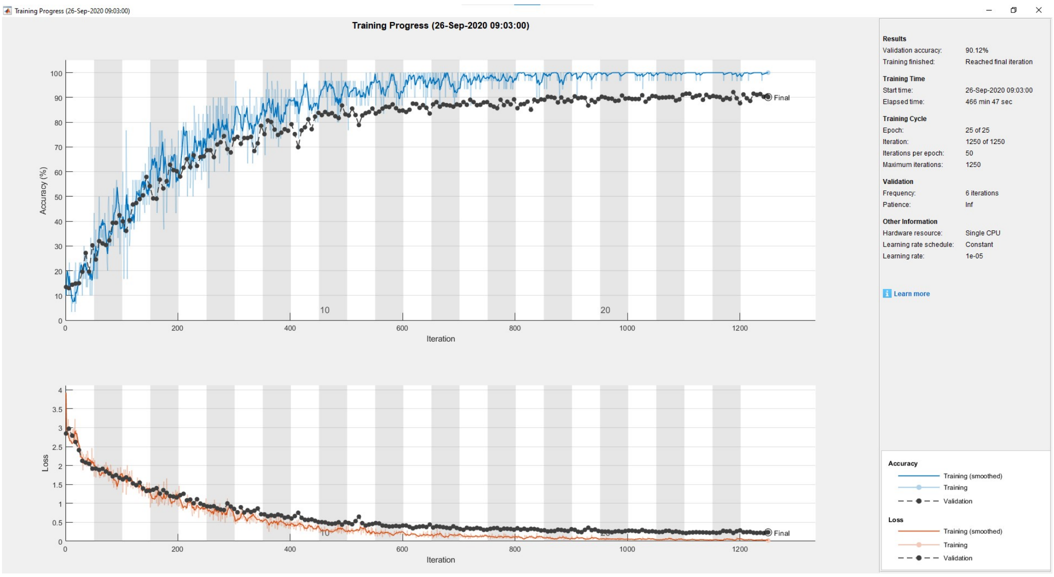
After model validation, the pre-trained CNN-2 classifier is employed to predict labels for unseen cases in each zone. Nine cases at nine different positions are used separately as input to CNN-2, and the output will have nine classes for each unseen case, indicating the portion of samples that are classified in different classes. Results are summarized in table 4.11. The row corresponds to the predicted label and the columns correspond to the target class (ground truth). Highlighted diagonal cells



Confusion Matrix

Output Class \ Target Class	crack 1	crack 2	crack 3	crack 4	crack 5	crack 6	crack 7	crack 8	crack 9	
crack 1	**65** 10.0%	**0** 0.0%	**8** 1.2%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**1** 0.2%	87.8% 12.2%
crack 2	**0** 0.0%	**72** 11.1%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**1** 0.2%	**3** 0.5%	**0** 0.0%	94.7% 5.3%
crack 3	**7** 1.1%	**0** 0.0%	**61** 9.4%	**0** 0.0%	**1** 0.2%	**0** 0.0%	**0** 0.0%	**1** 0.2%	**0** 0.0%	87.1% 12.9%
crack 4	**0** 0.0%	**0** 0.0%	**0** 0.0%	**58** 9.0%	**0** 0.0%	**4** 0.6%	**0** 0.0%	**0** 0.0%	**0** 0.0%	93.5% 6.5%
crack 5	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**71** 11.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	100% 0.0%
crack 6	**0** 0.0%	**0** 0.0%	**0** 0.0%	**14** 2.2%	**0** 0.0%	**68** 10.5%	**0** 0.0%	**1** 0.2%	**0** 0.0%	81.9% 18.1%
crack 7	**0** 0.0%	**0** 0.0%	**1** 0.2%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**65** 10.0%	**1** 0.2%	**9** 1.4%	85.5% 14.5%
crack 8	**0** 0.0%	**0** 0.0%	**1** 0.2%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**66** 10.2%	**0** 0.0%	98.5% 1.5%
crack 9	**0** 0.0%	**0** 0.0%	**1** 0.2%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**6** 0.9%	**0** 0.0%	**62** 9.6%	89.9% 10.1%
	90.3% 9.7%	100% 0.0%	84.7% 15.3%	80.6% 19.4%	98.6% 1.4%	94.4% 5.6%	90.3% 9.7%	91.7% 8.3%	86.1% 13.9%	**90.7%** **9.3%**

Figure 4.25: Confusion matrix from validation result of CNN-2 in the two-stage CNN with further enriched database



Training Progress (26-Sep-2020 09:03:00)

Training Progress (26-Sep-2020 09:03:00)

Results
Validation accuracy: 90.12%
Training finished: Reached final iteration

Training Time
Start time: 26-Sep-2020 09:03:00
Elapsed time: 466 min 47 sec

Training Cycle
Epoch: 25 of 25
Iteration: 1250 of 1250
Iterations per epoch: 50
Maximum iterations: 1250

Validation
Frequency: 6 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 4.26: Training progress of CNN-2 in the two-stage CNN for crack damage detection with further enriched database

correspond to correctly classified observations. The off-diagonal cells correspond to incorrectly classified observations. The percentage in each cell indicates what percentage of samples belonging to that column class are classified as the label of the row of the cell.

In the first column which shows the classification result of testing samples with ground truth "Crack 1". 63.33% samples are correctly classified while 18.33% samples are incorrectly classified as "Crack 3", 16.67% samples are incorrectly classified as "Crack 7" and 1.67% samples are incorrectly classified as "Crack 6". The classification accuracy is dominant among all predicted labels. In the second column, 73.33% samples with ground truth label "Crack 2" are correctly classified, while 10.00% samples are misclassified as "Crack 1", 13.33% samples are misclassified as "Crack 8" and 3.33% samples are misclassified as "Crack 3". The percentage of correctly classified samples is much higher than incorrectly classified ones. For testing samples with ground truth label "Crack 3" in the third column, 91.67% predictions are correct, with only 6.67% and 1.67% samples incorrectly classified as "Crack 1" and "Crack 9", respectively. In the forth column which shows the classification result for samples with ground truth label "Crack 4", 48.33% samples are correctly classified, larger than any other predicted labels. However, in the last five columns which that contain the results for crack 5, crack 6, ..., and crack 9, only 28.33%, 38.33%, 20.00%, 31.67% and 26.67% samples are correctly classified in the corresponding class, respectively. The classification accuracy for the last five cases are lower than misclassified portion.

Nevertheless, it has been verified that by increasing the variability of the database, the classification accuracy of the proposed two-stage CNN-based crack damage detection and localization model can be improved, especially for the crack cases 1, 2, 3 and 4. But it can't be ignored that the localization accuracy of crack damage in other locations still needs to be improved.

Table 4.11: Labels predicted by pre-trained CNN-2 for unseen cases of crack damage in the fine classification of the two-stage CNN method with further enriched database

		Target Class (Ground Truth)								
		Crack 1	Crack 2	Crack 3	Crack 4	Crack 5	Crack 6	Crack 7	Crack 8	Crack 9
Output Class (Prediction)	Crack 1	63.33%	10.00%	6.67%	6.67%	1.67%	3.33%	20.00%	0.00%	33.33%
	Crack 2	0.00%	73.33%	0.00%	6.67%	3.33%	0.00%	0.00%	66.67%	0.00%
	Crack 3	18.33%	3.33%	91.67%	11.67%	23.33%	11.67%	33.33%	0.00%	33.33%
	Crack 4	0.00%	0.00%	0.00%	48.33%	3.33%	45.00%	0.00%	0.00%	0.00%
	Crack 5	0.00%	0.00%	0.00%	1.67%	28.33%	0.00%	0.00%	0.00%	0.00%
	Crack 6	1.67%	0.00%	0.00%	3.33%	0.00%	38.33%	0.00%	0.00%	0.00%
	Crack 7	16.67%	0.00%	0.00%	15.00%	11.67%	0.00%	20.00%	1.67%	6.67%
	Crack 8	0.00%	13.33%	0.00%	6.67%	28.33%	1.67%	10.00%	31.67%	0.00%
	Crack 9	0.00%	0.00%	1.67%	0.00%	0.00%	0.00%	16.67%	0.00%	26.67%

4.6 Conclusions

In this section, a CNN-based structural health monitoring system is proposed due to the advantages of the deep learning algorithm Convolutional Neural Network in image recognition and classification. One-stage CNN model and two-stage CNN model are proposed successively.

In the first attempt, a one-stage CNN model is employed to detect and localize crack damage in the composite sandwich structure in section 4.2. Vibration responses of the structure are collected and pre-processed by Continuous Wavelet Transform to obtain time-frequency plane 2-D images that can be used as input to the CNN. One healthy case and six crack cases are used to train the model and an unseen case crack 7 is used to evaluate the effectiveness of the network in crack damage detection and localization at the same time. The largest portion of predicted label is "Healthy", which is not correct. The second largest portion is "Crack 6" which is expected since crack 6 is close to the unseen case and their dynamic characteristics should be similar from the physics point of view.

Based on the current results, an idea emerged, which is to divide the one-step classification method into two steps where the first step is used to distinguish healthy case and crack cases while the second step is used to localize the crack damage among crack cases. In this way the prediction accuracy of crack location could be improved. Therefore, a two-stage CNN model is proposed in section 4.3. In the first step, coarse classification, the model is capable of detecting the occurrence of crack damage in the structure. In the second step, fine classification, the portion of testing samples that are correctly classified is 62.34% and is dominant among all predicted labels.

It should be noted that the number of crack cases is too limited in the current database. It is expected that the CNN model can still provide reliable prediction results when dealing with more unknown cases. Accordingly, the two-stage CNN model is evaluated with enriched database in section 4.4. The performance of the coarse classification to detect the occurrence of crack damage is still excellent with an accuracy of 94.81%, but in the fine classification to localize the crack damage, the performance of CNN-2 is not stable, which may be caused by the lack of variability of the database. Therefore, it is very important to increase the variability of the database so as to improve the performance and robustness of the classifier. Thus, the effectiveness of the two-stage CNN model is evaluated with further enriched database in section 4.5.

The structure is divided into nine zones, each of which represents one location of crack damages. Each zone is enriched with new crack cases that have the same label. 4/5 cases are used to train the two-stage CNN model and the rest 1/5 are used as testing samples. The performance of CNN-1 is perfect with an accuracy 100%. In the fine classification, it is verified that the classification accuracy of the proposed two-stage CNN-based crack damage detection and localization model can be improved by increasing the variability of the database, especially for the crack cases 1, 2, 3 and 4. But it can't be ignored that the localization accuracy of crack damage in other locations still needs to be improved. In addition, the two-stage CNN model has advantages in both crack detection and localization compared to the one-stage CNN model.

CHAPTER 5

Convolutional Neural Networks based structural health monitoring at low frequencies

---

Contents

5.1 Introduction

Modal parameters are functions of the physical properties of the structure, such as natural frequencies, mode shapes and modal damping. Therefore, changes to the material or structure will adversely cause detectable changes in modal properties and affect the structure performance. One method of structural health monitoring is to extract damage-related information from the wave that propagates in the structure and interacts with the damage site, provided that it is assumed that the propagated waves contain information related to damage. This method was used in Chapter 3 and Chapter 4. Another method is to directly detect the physical parameter abnormality of the structure caused by the damage, such as mode shape, stress, strain that may be sometimes more sensitive to damage than displacement from a signal point. Meanwhile, the issues that need to be studied also include the need to select appropriate data to achieve SHM according to different situations.

In this chapter, it is proposed to use modal parameters for structural health monitoring under low frequencies, different from the previous two chapters where wave propagation is used for structural health monitoring at high frequencies. At the same time, the wide applicability of the proposed CNN-based method for damage detection can be verified, e.g. for different structures and different types of data. The aim is to detect crack damage and predict the damage size in a simply-supported beam based on mode shapes and stress distribution in the beam instead of the vibration data used in the sandwich structure. Mode shapes and stress corresponding to different vibration modes of the beam are used successively to train the CNN. The position of the crack as well as its size are predicted by the CNN model.

5.2 Beam modeling and simulation

A simply-supported beam is considered as the research target. The beam is made of isotropic and homogeneous steel. The length of the beam is $1200mm$, and the cross section is $50 \times 50mm$. For steel material, the density is $\rho = 7850kg/m^3$, the Young's Modulus is $E = 2 \cdot 10^{11}Pa$ and the Poisson's ratio is $\gamma = 0.3$. A simplified illustration of the beam is referred to figure 5.1. Solid 185 element is used to build the structure in ANSYS. The structure is meshed with square elements with size of $10mm$.

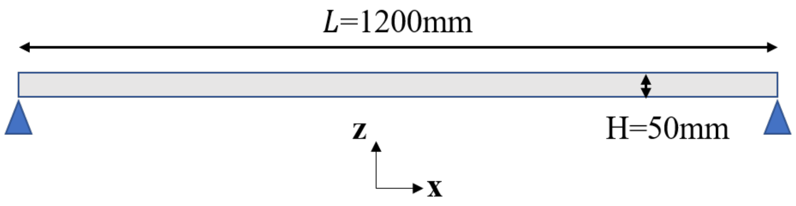


Figure 5.1: Simply-supported beam made of steel of dimensions $1200 \times 50 \times 50mm$

The intact structure is used as baseline model. Crack damage is designed at four locations, $x_1 = 140mm$, $x_2 = 270mm$, $x_3 = 450mm$ and $x_4 = 550mm$, respectively. In addition, four levels of crack damage is implemented for each location. Crack depth varying from $10mm$ to $40mm$ with $10mm$ increment is considered to investigate the damage severity, as shown in table 5.1. Crack severity index is the ratio of crack depth to the beam height where H represents the beam height and h represents crack depth. Therefore, there are a total of 17 models, one of which is an intact model, and each of the 4 crack sites has 4 models with different levels of severity.

Table 5.1: Crack severity index

Crack size (mm)	Damage Index (Di) (h/H)
10	0.2
20	0.4
30	0.6
40	0.8

As the cross section of the beam is square, only the vibration modes in the x-z plane are considered. Modal analysis is firstly carried out to extract the first ten natural frequencies of each model. Figure 5.2 illustrates the first ten natural frequencies of all the four models with different damage size but at the same location $x = 550mm$ together with those of the healthy structure. It is observed that natural frequencies will decrease with the extension of the crack damage. Similar phenomena have been found in other cases.

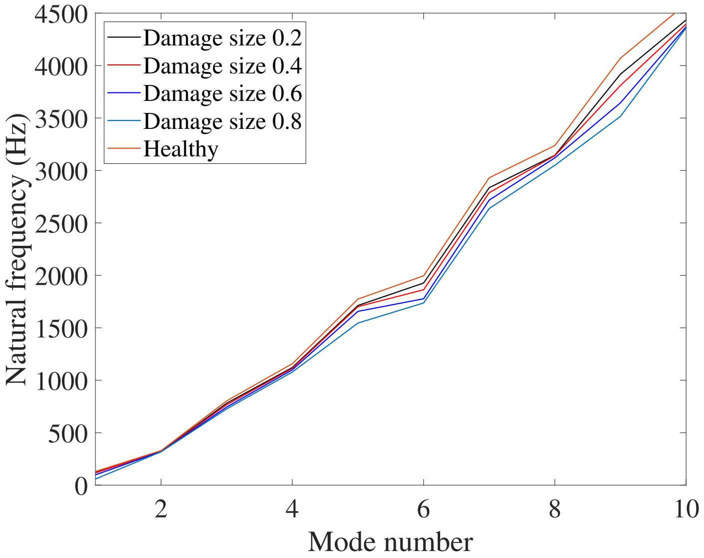


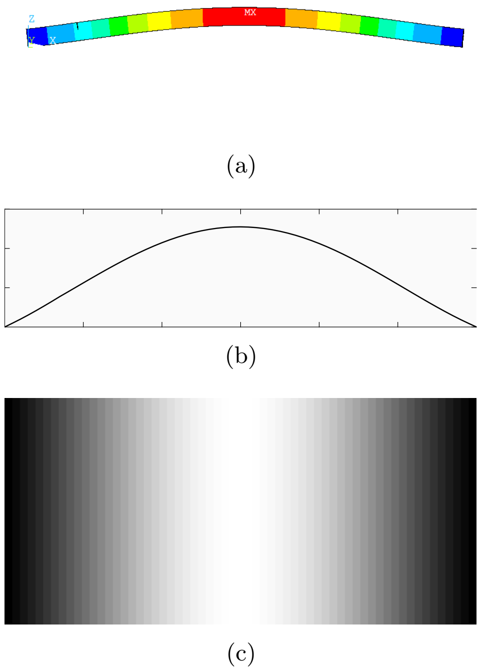
Figure 5.2: Comparison of the first ten natural frequencies of different damage severity of crack at $x = 550mm$

The first ten natural frequencies for all damage cases and the intact one is listed in table 5.2. Numbers in the first column of the table indicate the damage location in the structure with unit *meter* and Di in the second column indicates the damage

Table 5.2: First ten natural frequencies (Hz) of the healthy beam and all 16 models with different damage severity at four locations

Loc.	Di	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6	Mode 7	Mode 8	Mode 9	Mode 10
Healthy		129.91	329.06	803.60	1157.8	1773.6	1995.4	2930.4	3237.6	4070.9	4588.9
0.14	0.2	127.04	320.03	778.57	1117.2	1705.4	1925.3	2819.4	3144.3	3919.3	4451.6
	0.4	126.82	312.93	747.09	1097.5	1652.8	1873.4	2730.6	3143.2	3845	4441.4
	0.6	125.23	293.4	668.21	1069.8	1518.8	1838.8	2625.9	3139.6	3761.2	4421.8
	0.8	110.22	237.32	539.52	1049.3	1363.7	1830.4	2549	3131.5	3693.6	4387.8
0.27	0.2	126.54	318.08	776.52	1124.1	1702.8	1945.9	2841	3140.3	3919.5	4453.6
	0.4	123.99	304.87	743.53	1123.8	1659	1943.8	2814.4	3131.3	3835.6	4451.9
	0.6	114.67	278.22	685.61	1122.3	1583	1941.7	2787.2	3122.7	3703.9	4442.3
	0.8	79.398	243.35	619.27	1117.9	1498.8	1936.7	2777.4	3118.1	3542.7	4365.4
0.45	0.2	125.45	320.01	786.14	1113.3	1710.1	1946.2	2832.9	3114.2	3938.3	4452.7
	0.4	118.86	313.9	785.53	1078.1	1696.3	1944.7	2747.9	3036.9	3922.6	4442.8
	0.6	101.44	303.61	784.97	1017.1	1680.2	1942.8	2548.3	2973.5	3917.4	4412.6
	0.8	61.668	293.42	783.94	949.71	1670.9	1937	2302.4	2932.9	3913.7	4344.8
0.55	0.2	125.09	321.65	781.29	1121.8	1712.8	1926.8	2835.7	3144.4	3918.1	4434.8
	0.4	117.35	320.81	766.34	1114.1	1701.8	1862.5	2788.5	3141.2	3810.6	4395.6
	0.6	98.319	319.34	744.34	1099.5	1656.8	1776.8	2717.5	3120.1	3644.8	4369.7
	0.8	58.522	317.79	726.85	1078.2	1545.3	1736.6	2637	3049.4	3515.8	4358.5

index. The unit of frequencies in the table is Hz.



(a)

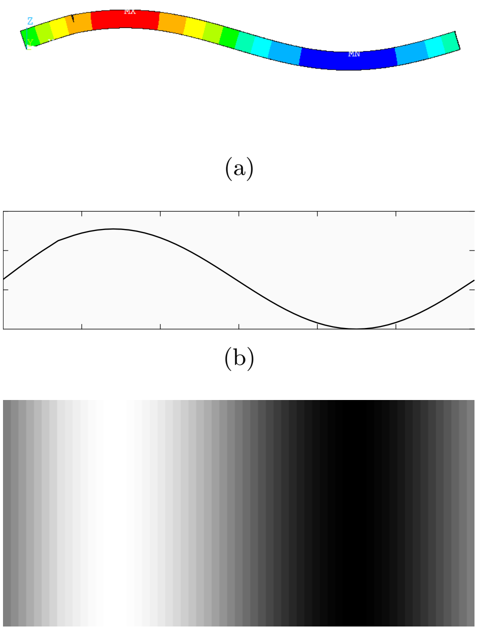
(b)

(c)

Figure 5.3: First vibration mode for the case where crack is located at $x = 140mm$ with damage index Di=0.4. (a) Mode shape, (b) Reconstructed mode shape from sampling points with z-component displacement, (c) Gray image converted from the reconstructed mode shape curve

It is assumed that the changes in the physical properties of the structure caused by the structural discontinuity due to the damage will be reflected in mode shapes. Therefore, mode shapes will be extracted in each damage scenario in the modal analysis and will be used in this section to train the two-stage CNN model. 61 sampling points are selected equidistantly from the beginning to the end on the upper surface of the beam model along the x direction. The mode shape can be reconstructed by the displacement along z direction of all sampling points. The first mode shape of crack at $x = 140mm$ with damage index Di=0.4 is illustrated in figure 5.3(a), which plots the z-component displacement of the beam. A vector constituted of z-component displacement from sampling points is normalized and projected to the interval 0-255, then it is used to reconstruct the mode shape as shown in figure 5.3(b). In order to take advantage of image processing capabilities of CNN, the mode shape curve is converted into a gray image, as shown in figure 5.3(c), where the x axis indicates the direction of the beam and bright color indicates high vibration amplitude while black color indicates low vibration amplitude corresponding to the curve in figure 5.3(b).

Similarly, the second vibration mode of the same case where crack is located at $x = 140mm$ with damage index Di=0.4 is illustrated in figure 5.4. In this case, singularity caused by structural discontinuity due to the crack can be observed in the reconstructed curve and the gray image as shown in figure 5.4(b) and 5.4(c), corresponding to the crack damage in figure 5.4(a). Mode shapes in grey image for each crack model and the healthy model are extracted to constitute the mode shape-based database.


(a)

(b)

(c)

Figure 5.4: Second vibration mode for the case where crack is located at $x = 140mm$ with damage index Di=0.4. (a) Mode shape, (b) Reconstructed mode shape from sampling points with z-component displacement, (c) Gray image converted from the reconstructed mode shape curve

Stress distribution in the beam, as an alternative to mode shapes based on displacement, can also be extracted to represent another kind of features that describe the status of the structure. In order to obtain stress distribution on the beam, harmonic analysis is carried out. An external force $F = -100$ is applied to nodes located on the bottom surface at $x = 100mm$, as illustrated in figure 5.5. The frequency range of the harmonic analysis is within the first ten natural frequencies of each beam model obtained in table 5.2. Ten sub-steps are set, In ten substeps, the frequency of each substep load corresponds to a natural frequency. Consequently, stress distribution corresponding to the first ten vibration modes of the beam is obtained.

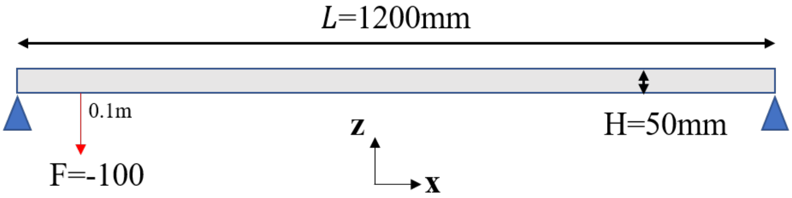
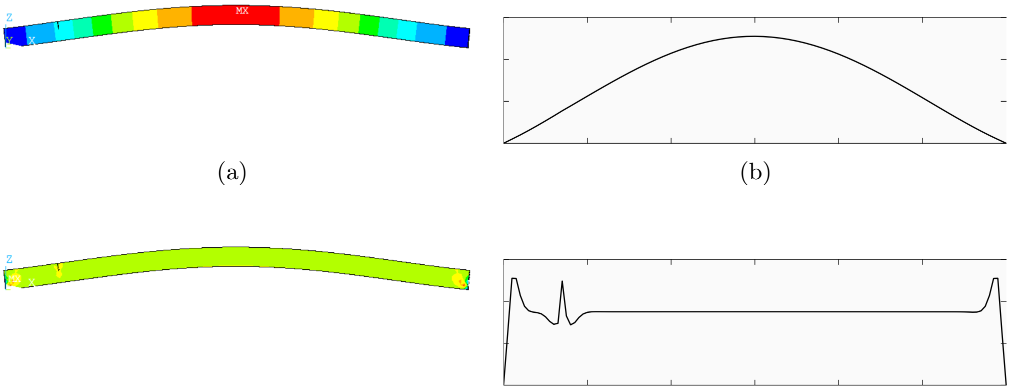


Figure 5.5: Harmonic analysis of the simply-supported beam with external load

121 sampling point are selected equidistantly from the beginning to the end at the height $z = 30mm$ in the beam along the x direction. Six stress components are extracted for each mode, i.e., $\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{xz}, \sigma_{yz}$. A comparison of the stress distribution of component σ_{zz} with the corresponding mode shape is illustrated in figure 5.6 and figure 5.7. It can be observed that the zz-component stress distribution in figure 5.6(c) is different from the mode shape based on z-direction displacement in figure 5.6(a), where the singularity at the crack position $x = 140mm$ in stress is more dominant than in mode shape. It can be verified from the reconstructed stress distribution in figure 5.6(d) and the reconstructed mode shape in figure 5.6(b).

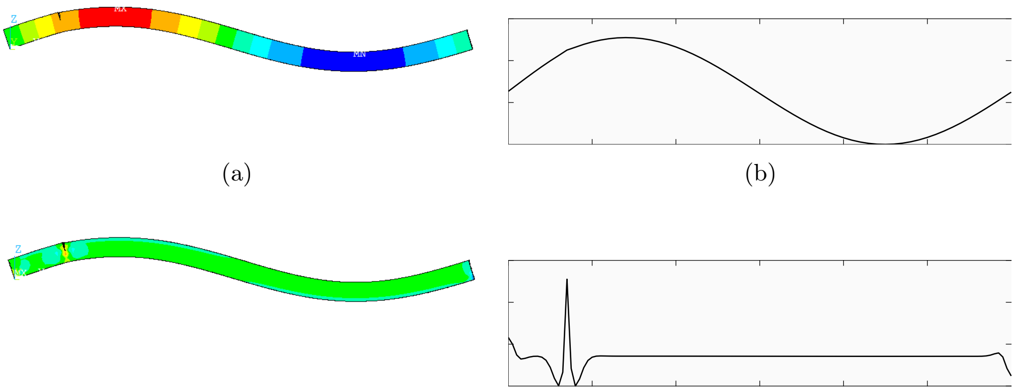


(a) (b)

(c) (d)

Figure 5.6: First vibration mode for the case where crack is located at $x = 140mm$ with damage index Di=0.4. (a) Mode shape, (b) Reconstructed mode shape from sampling points with z-component displacement, (c) Stress component σ_{zz} distribution, (d) Reconstructed stress distribution from sampling points with σ_{zz} component

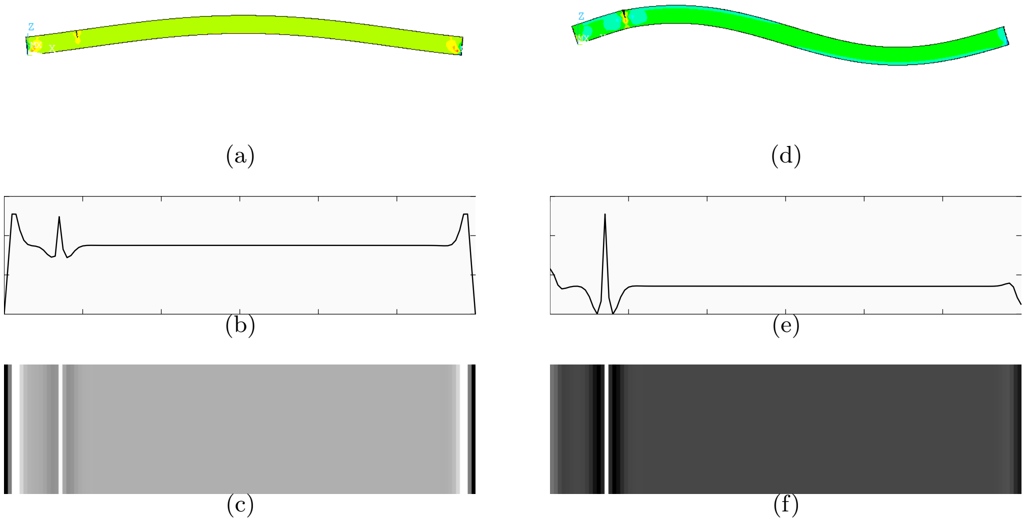
The singularity at the peak indicates that the stress here is very large, which is consistent with the phenomenon of stress concentration on the damage position. The same phenomenon can be found in the second mode in figure 5.7.



(a) (b)

(c) (d)

Figure 5.7: Second vibration mode for the case where crack is located at $x = 140mm$ with damage index Di=0.4. (a) Mode shape, (b) Reconstructed mode shape from sampling points with z-component displacement, (c) Stress component σ_{zz} distribution, (d) Reconstructed stress distribution from sampling points with σ_{zz} component



(a) (d)

(b) (e)

(c) (f)

Figure 5.8: Stress component σ_{zz} distribution corresponding to the first two modes for crack at $x = 140mm$ with damage index Di=0.4. (a) σ_{zz} distribution in mode 1, (b) Reconstructed stress distribution in mode 1, (c) converted Grey image from curve 1, (d) σ_{zz} distribution in mode 2, (b) Reconstructed stress distribution in mode 2, (c) converted Grey image from curve 2

Similar to mode shapes, reconstructed stress distribution curves should be converted into grey images in order to take advantage of image processing capabilities of CNN. Figure 5.8 illustrates the stress distribution of σ_{zz} in the first mode and the second mode. The reconstructed stress distribution curve for the first mode in figure 5.8(b) is converted into a grey image in figure 5.8(c) and the reconstructed stress distribution curve for the second mode in figure 5.8(e) is converted into a grey image in figure 5.8(f). In fact, the six stress components in all the first ten vibration modes of each model are extracted and converted into grey images. The pertinence of different stress components in CNN will be investigated in the next section.

5.3 Two-stage CNN using mode shapes

The two-stage CNN-based crack detection model: coarse to fine classification used in Chapter 4, is proposed in this section for damage detection in a beam using mode shapes. The first stage is employed to predict the crack location on the beam. After fault-pattern recognition, the second stage is employed to predict the crack damage size. An overview of data pre-processing and the proposed two-stage CNN-based crack detection algorithm is presented in figure 5.9.

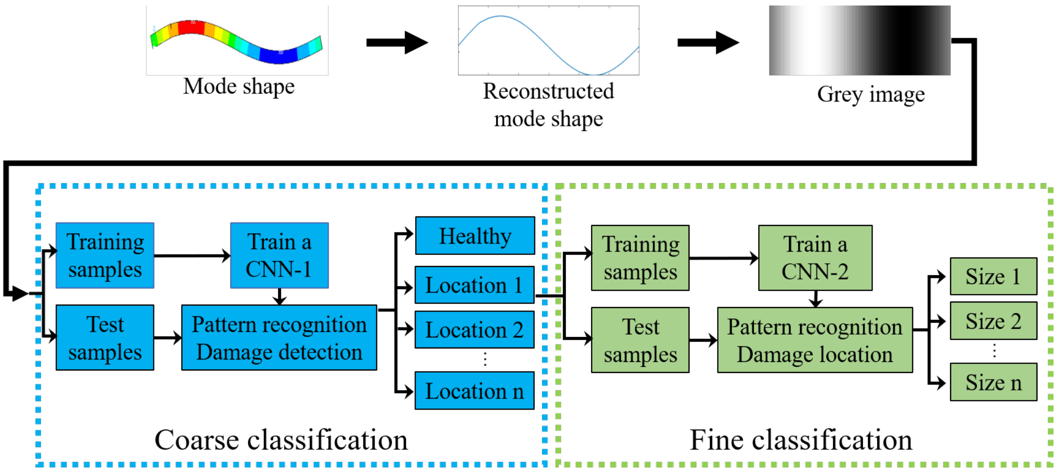


Figure 5.9: Overview of data pre-processing and the proposed two-stage CNN-based crack detection algorithm using mode shapes

5.3.1 Coarse classification

In the first step of the two-stage CNN-based crack detection algorithm, the aim is to predict the crack location. The database contains 5 cases (i.e., healthy and 4 crack cases at different locations), each crack case includes four levels of crack severity. The first ten mode shapes of each cases are extracted and transformed into grey images. Some samples from the database are illustrated in figure 5.10, where mode shapes in grey images for all crack locations are with damage index

Di=0.4. The database is split into five classes: healthy, 0.14m, 0.27m, 0.45m and 0.55m, indicating "healthy" status and four damage locations respectively. Thus the output layer of the CNN-1 model has five classes corresponding to those in the database. 60%, 20% and 20% of the database are randomly selected as training data, validation data and testing data, respectively. In this way, none of the samples in validation data and testing data appear in the training phase. It is expected that the CNN-1 model can distinguish features of the samples belonging to different classes after training. The mini-batch size is set as 32 and the maximum epoch number is set as 40.

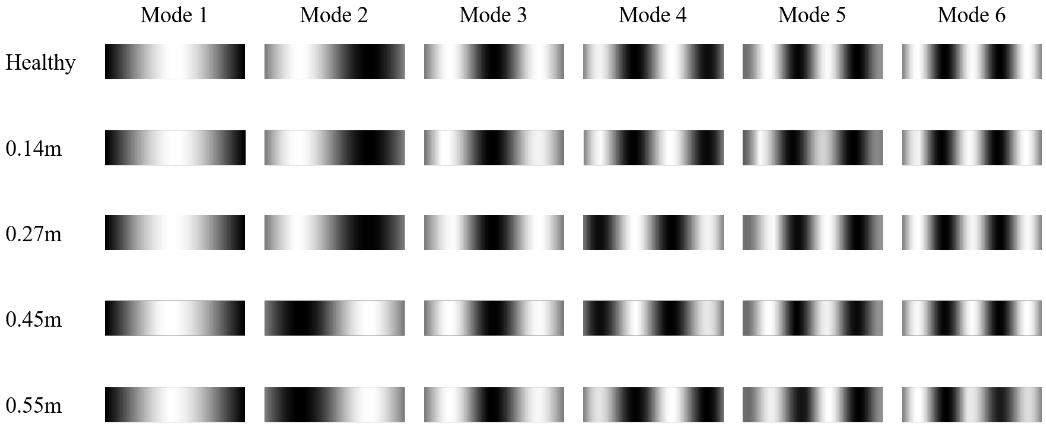
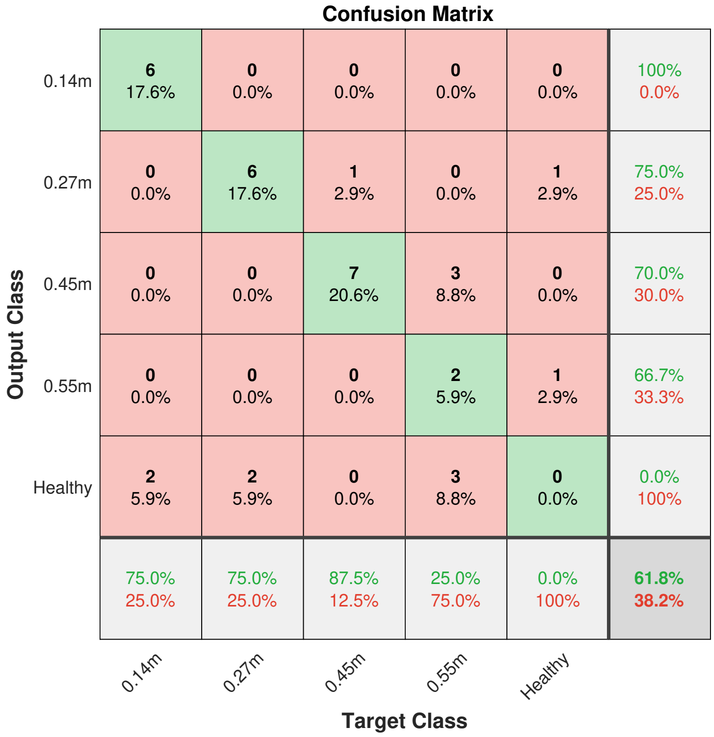


Figure 5.10: Mode shapes in grey images corresponding to different classes. Samples of crack cases are with damage index Di=0.4.

A confusion matrix can be obtained from the validation results as shown in figure 5.11. It can be observed that none of samples with ground truth label "Healthy" are correctly classified. For samples with ground truth label "0.14m", the classification accuracy is 75%. For samples with ground truth label "0.27m", "0.45m" and "0.55m", the corresponding classification accuracy is 75%, 87.5% and 25%, respectively. An overall average validation accuracy is obtained as 61.8%. The training process is plotted in figure 5.12. It can be seen that the validation accuracy doesn't converge to a stable value but it remains at a relative level around 60%. Meanwhile, the value of validation loss function is kept at a level, around 1.5.

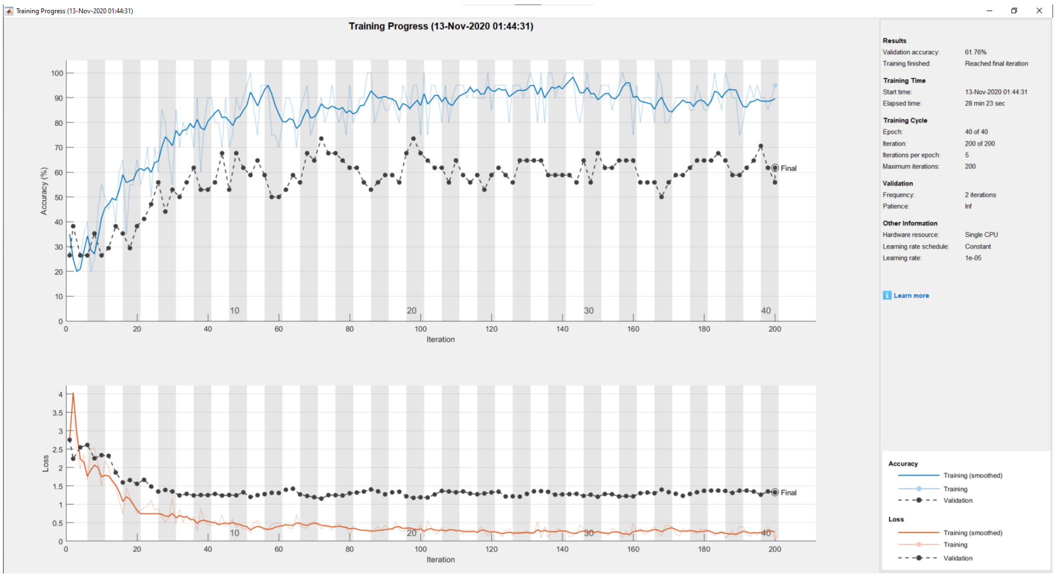
In the testing phase, the result is slightly better. As summarized in the confusion matrix in figure 5.13, an overall test accuracy of 64.7% is obtained: all samples belonging to the case 0.14m are correctly classified, but healthy samples are all misclassified. 50% samples belonging to the case 0.27m are correctly classified with 2 samples classified as 0.45m and 1 sample classified as 0.14m and 0.55m respectively. For the case 0.45m, 75% samples are correctly classified while one sample classified as "healthy" and one classified as 0.27m. The classification accuracy for the case



Confusion Matrix

Output Class \ Target Class	0.14m	0.27m	0.45m	0.55m	Healthy	
0.14m	**6** 17.6%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	100% 0.0%
0.27m	**0** 0.0%	**6** 17.6%	**1** 2.9%	**0** 0.0%	**1** 2.9%	75.0% 25.0%
0.45m	**0** 0.0%	**0** 0.0%	**7** 20.6%	**3** 8.8%	**0** 0.0%	70.0% 30.0%
0.55m	**0** 0.0%	**0** 0.0%	**0** 0.0%	**2** 5.9%	**1** 2.9%	66.7% 33.3%
Healthy	**2** 5.9%	**2** 5.9%	**0** 0.0%	**3** 8.8%	**0** 0.0%	0.0% 100%
	75.0% 25.0%	75.0% 25.0%	87.5% 12.5%	25.0% 75.0%	0.0% 100%	**61.8%** **38.2%**

Figure 5.11: Confusion matrix from validation result of CNN-1 in the two-stage CNN for crack localization in a beam



Results
Validation accuracy: 61.76%
Training finished: Reached final iteration

Training Time
Start time: 13-Nov-2020 01:44:31
Elapsed time: 28 min 23 sec

Training Cycle
Epoch: 40 of 40
Iteration: 200 of 200
Iterations per epoch: 5
Maximum iterations: 200

Validation
Frequency: 2 iterations
Patience: Inf

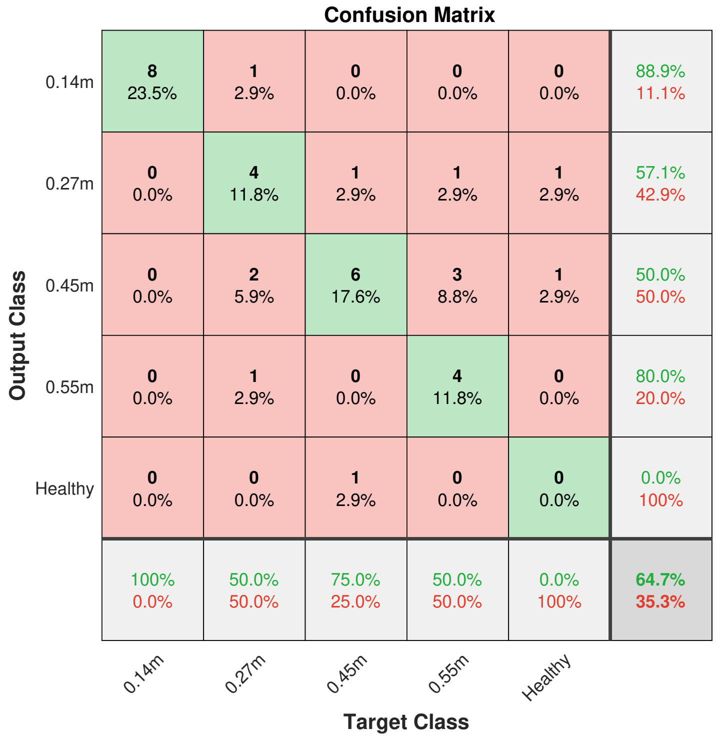
Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Accuracy
Training (smoothed)
Training
Validation

Loss
Training (smoothed)
Training
Validation

Figure 5.12: Training process of CNN-1 for crack localization in the beam

0.55m is 50%, but 37.5% samples are classified as 0.45m, which is the case next to the location 0.55m. It should be mentioned that the false alarm may lead to interference in decision-making of practitioners.



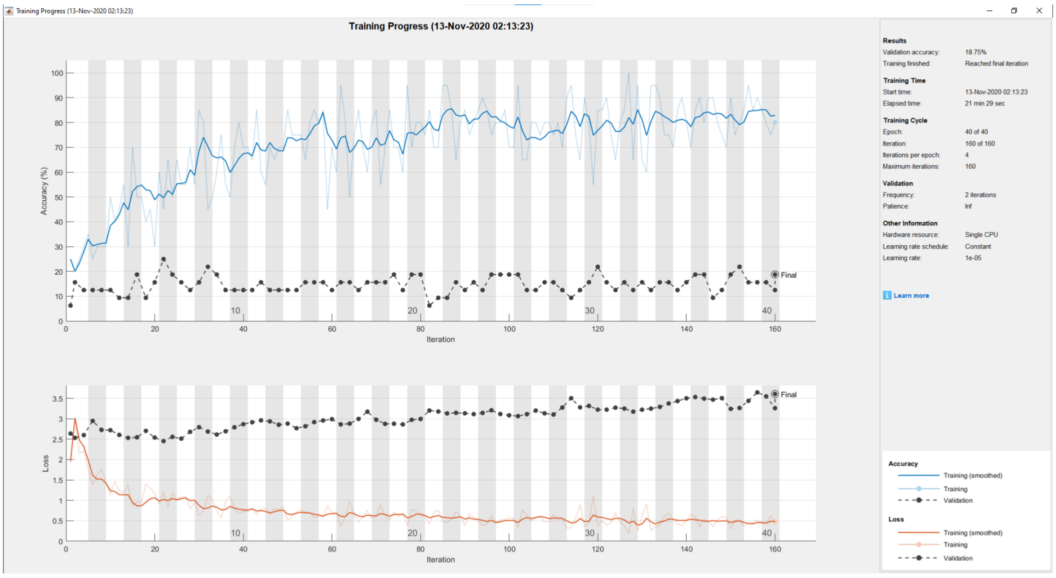
	0.14m	0.27m	0.45m	0.55m	Healthy	
0.14m	**8** 23.5%	**1** 2.9%	**0** 0.0%	**0** 0.0%	**0** 0.0%	88.9% 11.1%
0.27m	**0** 0.0%	**4** 11.8%	**1** 2.9%	**1** 2.9%	**1** 2.9%	57.1% 42.9%
0.45m	**0** 0.0%	**2** 5.9%	**6** 17.6%	**3** 8.8%	**1** 2.9%	50.0% 50.0%
0.55m	**0** 0.0%	**1** 2.9%	**0** 0.0%	**4** 11.8%	**0** 0.0%	80.0% 20.0%
Healthy	**0** 0.0%	**0** 0.0%	**1** 2.9%	**0** 0.0%	**0** 0.0%	0.0% 100%
	100% 0.0%	50.0% 50.0%	75.0% 25.0%	50.0% 50.0%	0.0% 100%	**64.7%** **35.3%**

Figure 5.13: Confusion matrix from testing result of CNN-1 for crack localization in the beam

5.3.2 Fine classification

In fine classification which is dedicated to predicting the crack size, data with "Healthy" label are eliminated and data corresponding to crack cases "0.14m", "0.27m", "0.45m" and "0.55m" are kept, but are rearranged with respect to damage size indicated by damage index Di. Accordingly, data are split into four classes: 0.2, 0.4, 0.6 and 0.8, which are also the classes of the output layer of CNN-2.

In the database, 60% are used as training data, while data used for verification and testing each account for 20% of the database. The maximum epoch number is set as 40 and validation frequency is set as 2 iterations. The training and validation process is plotted in figure 5.14. The validation accuracy is at a low level between 10% and 20%, and the loss function continues to increase, indicating that the pre-trained CNN model cannot properly distinguish features between different crack sizes from the mode shape grey images.



Results
Validation accuracy: 18.75%
Training finished: Reached final iteration

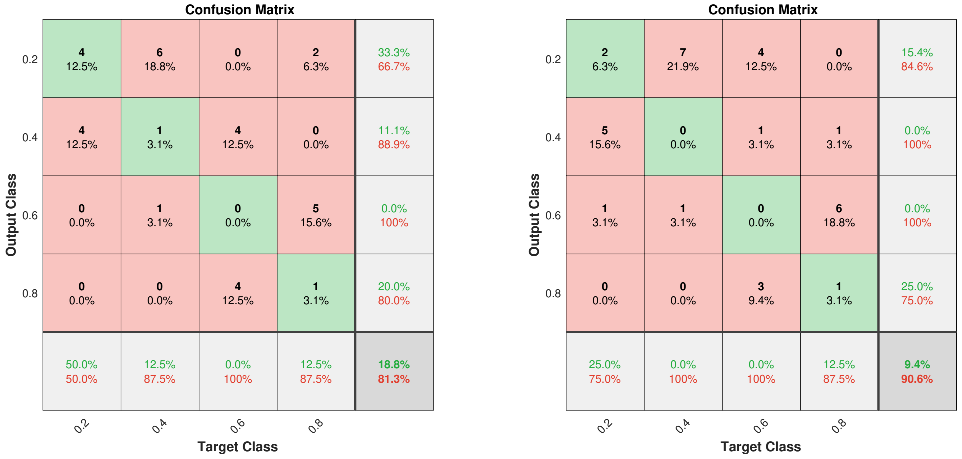
Training Time
Start time: 13-Nov-2020 02:13:23
Elapsed time: 21 min 29 sec

Training Cycle
Epoch: 40 of 40
Iteration: 160 of 160
Iterations per epoch: 4
Maximum iterations: 160

Validation
Frequency: 2 iterations
Patience: Inf

Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 1e-05

Figure 5.14: Training process of CNN-2 for crack size prediction in the beam



Output Class \ Target Class	0.2	0.4	0.6	0.8	
0.2	4 12.5%	6 18.8%	0 0.0%	2 6.3%	33.3% 66.7%
0.4	4 12.5%	1 3.1%	4 12.5%	0 0.0%	11.1% 88.9%
0.6	0 0.0%	1 3.1%	0 0.0%	5 15.6%	0.0% 100%
0.8	0 0.0%	0 0.0%	4 12.5%	1 3.1%	20.0% 80.0%
	50.0% 50.0%	12.5% 87.5%	0.0% 100%	12.5% 87.5%	**18.8%** **81.3%**

(a) Validation

Output Class \ Target Class	0.2	0.4	0.6	0.8	
0.2	2 6.3%	7 21.9%	4 12.5%	0 0.0%	15.4% 84.6%
0.4	5 15.6%	0 0.0%	1 3.1%	1 3.1%	0.0% 100%
0.6	1 3.1%	1 3.1%	0 0.0%	6 18.8%	0.0% 100%
0.8	0 0.0%	0 0.0%	3 9.4%	1 3.1%	25.0% 75.0%
	25.0% 75.0%	0.0% 100%	0.0% 100%	12.5% 87.5%	**9.4%** **90.6%**

(b) Testing

Figure 5.15: (a) Confusion matrix of validation result, (b) Confusion matrix of testing result of CNN-2 for crack size prediction in the beam

A confusion matrix based on the validation result and testing result can be obtained in figure 5.15(a) and figure 5.15(b), respectively. In the validation phase, an overall classification accuracy of 18.8% is obtained, in which 50% for Di=0.2, 12.5% for Di=0.4 and 0.8, 0% for Di=0.6. In the testing phase, the overall accuracy is only 9.4%, in which 25% for Di=0.2, 0% for Di=0.4 and 0.6, 12.5% for Di=0.8.

Compared to the result in the first stage CNN-1 where the validation and testing classification accuracy is 61.8% and 64.7%, the CNN-2 is not capable of predicting the crack size with an acceptable accuracy.

The results obtained by the two-stage CNN method using mode shapes in this section indicate that, mode shapes or displacements of the structure may not be as sensitive to damages as expected. Therefore, it is necessary to use alternative types of data that are more sensitive to damage in crack localization and size prediction in the beam. A two-stage CNN using stress data will be discussed in section 5.4.

5.4 Two-stage CNN using stress in each mode

From harmonic analysis in section 5.2, stress distribution corresponding to the first ten vibration modes of the beam is obtained, including six stress components: $\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{xz}, \sigma_{yz}$. Some stress distribution samples in grey images in the beam corresponding to different cases of mode 1 with Di=0.2 are illustrated in figure 5.16. The rows correspond to different cases, i.e., healthy case and crack at different locations. The columns correspond to the six stress components. It can be observed that the color changes smoothly in the healthy case except for σ_{xy} and σ_{yz} where the singularity is due to the external force at $x = 100mm$. The same singularity is found in other crack cases in stress components σ_{xy} and σ_{yz}. In other columns corresponding to other stress components, a sudden change in color can be observed at a specific location, which is the location of the crack damage, indicating a sudden change of stress that may be caused by a potential damage at that position.

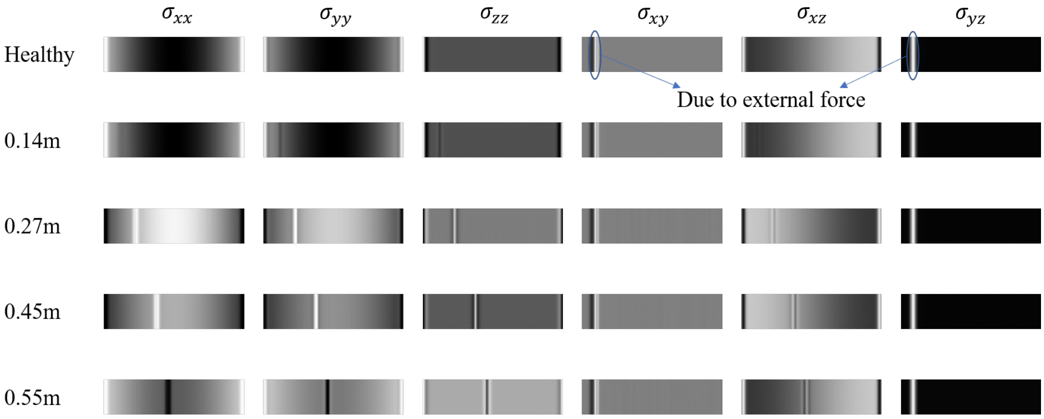


Figure 5.16: Stress distribution in grey images in the beam corresponding to different cases of mode 1 with Di=0.2

Different from previous section where the database contains mode shapes of different cases, in this section it is composed of stress distribution grey images. The process of crack localization and crack size prediction remains the same as illustrated in figure 5.9. Pertinence of different stress components will be discussed in the coarse classification by CNN-1 and fine classification by CNN-2.

5.4.1 Coarse classification

In coarse classification, CNN-1 is employed to predict the crack location in the beam based on stress distribution grey images. We have a total of 10×6 (10 represents the first ten modes in the healthy case, 6 represents the six stress components in each mode) and $4 \times 4 \times 10 \times 6$ (4 crack locations, 4 crack sizes at each location, 10 modes in each case, 6 stress components in each mode) data. Since the six stress components may contain different information representing characteristics of the structure, the role played in SHM may also be different. In order to study the effects of different stress components, different databases have been established to train and test CNN-1 respectively, namely a database containing all components, 6 databases each containing only one component, and a database without the components σ_{xy} and σ_{yz}. It should be pointed out that each database is used to train its corresponding CNN model. 60% of each database are used to train the CNN-1 model, 20% and 20% are used as validation data and testing data for each database.

Table 5.3: Prediction accuracy of crack location in validation and testing phase using different databases by CNN-1

Stress component σ_{**}	Coarse classification	
	Validation	Testing
ZZ	100%	100%
Without XY and YZ	95.1%	91.18%
XZ	92%	64.71%
YY	88%	82.35%
XX	84%	76.47%
All components	75.16%	70.59%
XY	56%	82.35%
YZ	48%	47.06%

The prediction accuracy of crack location in validation and testing phase using different databases is summarized in table 5.3 sorted in descending order of validation accuracy. In validation phase, the database containing only σ_{zz} has the best performance with a prediction accuracy of 100%. Stress component σ_{xy} and σ_{yz} have the worst performance with an accuracy of 56% and 48% respectively. In testing phase, stress component σ_{zz} has the best performance with a prediction accuracy of 100% while the component σ_{yz} has the worst performance with an accuracy of 47.06%. The difference between databases suggests that features representing different crack locations in some stress components are more discriminative than in others.

5.4.2 Fine classification

In fine classification, CNN-2 is employed to predict the crack size in the beam using stress distribution grey images. The database in this section only contains data corresponding to crack cases. Similar to coarse classification, databases are established to study the importance of different stress components. The difference to coarse classification is that data in each database are split into four classes corresponding to four crack sizes indicated by the damage index Di (0.2, 0.4, 0.6 and 0.8). In each database, 60% data are used to train a CNN-2 model, 20% and 20% are used as validation data and testing data.

Table 5.4: Prediction accuracy of crack size in validation and testing phase using different databases by CNN-2

Stress component σ_{**}	Fine classification	
	Validation	Testing
ZZ	83.33%	62.50%
YY	75%	68.75%
XZ	70.83%	75.00%
XX	66.67%	50.00%
Without XY and YZ	64.58%	65.63%
XY	62.50%	50.00%
YZ	62.50%	50.00%
All components	58.33%	55.21%

The prediction accuracy of crack size in validation and testing phase using different databases is summarized in table 5.4 sorted in descending order of validation accuracy. Stress component σ_{zz} has a prediction accuracy of 83.33%, outperforming other components and the database containing all components performs the worst, with an accuracy 58.33%. But in testing phase, the database of σ_{xz} and that of σ_{yy} have the best performance with 75.00% and 68.75% respectively. The classification accuracy of other databases is no less than 50%.

5.5 Discussions

The results obtained from mode shapes and stress distribution in the beam have big difference. For the prediction of the crack location, an overall accuracy of 64.7% is obtained using mode shapes, but using stress distribution, the best performance can have an accuracy of 100%. For the prediction of crack size, the overall accuracy with mode shapes is at a very low level of 9.4%, indicating the incapability of prediction, but the accuracy is no less than 50% using stress distribution, and can even reach 75%. It is clear that the classification accuracy for both crack location and crack size using stress distribution has been improved significantly compared with the results using mode shapes, which suggests that stress distribution may contain more discriminative information relative to crack location and size than mode shapes.

Although high accuracy can be achieved with stress distribution, what cannot be ignored is that the number of sampling points to reconstruct the stress distribution on the beam is 121. This means that if it is under real experimental conditions, it is necessary to arrange sensors at so many points to collect data, which will greatly increase the cost of experimental equipment and will burden data storage, processing and subsequent computation. Therefore, under the premise of ensuring the accuracy of the results, appropriately reducing the number of sampling points should be studied. Figure 5.17 illustrates the stress distribution in the beam with the reduction of the number of sampling points corresponding to crack damage at 0.27m with crack size Di=0.2 of the first mode. It can be seen that with sampling point reduction, the features indicating damage in the stress distribution grey image become fuzzier, which may lead to poorer classification accuracy.

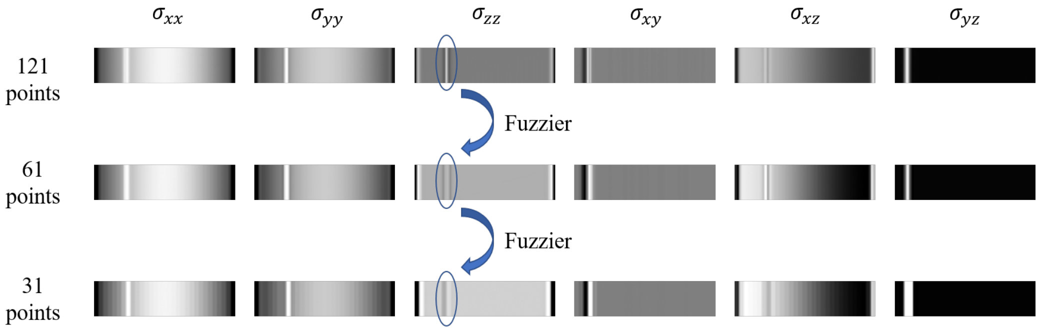


Figure 5.17: Stress distribution in the beam with the reduction of the number of sampling points. Images correspond to crack damage at 0.27m with crack size Di=0.2 of the first mode

The classification accuracy of crack location and crack size with database from reduced sampling points predicted by the two-stage CNN model is listed in table

5.5. It can be found that the classification accuracy decreases with the reduction of sampling points as expected. It should be due to the fact that stress containing important damage information near the damage area are more likely not to be collected with less sampling points. However, the classification accuracy in the prediction of crack location remains 82.35% at least while that in the prediction of crack size remains 50.00% at least. Therefore, it is possible to reduce the number of sampling points, but it is necessary to find a suitable compromise between the accuracy of classification and the number of sampling points.

Table 5.5: Prediction accuracy of crack location and crack size with the reduction of sampling points

Sampling points (database σ_{**})	Coarse classification		Fine classification	
	Validation	Testing	Validation	Testing
121 points (Without XY and YZ)	95.10%	91.18%	64.58%	65.63%
61 points (Without XY and YZ)	88.24%	94.12%	56.25%	51.56%
31 points (Without XY and YZ)	78.43%	82.35%	51.04%	50.00%

5.6 Conclusions

In this chapter, it is proposed to use modal parameters for structural health monitoring under low frequencies, different from the previous two chapters where wave propagation is used for structural health monitoring at high frequencies. At the same time, the wide applicability of the proposed two-stage CNN-based method for damage detection can be verified, e.g. for different structures and different types of data. Thus, crack localization and damage size prediction in a simply-supported beam is conducted based on mode shapes and stress distribution in the beam instead of the vibration data used in the sandwich structure.

The classification accuracy for both crack location and crack size using stress distribution in the two-stage CNN model has been improved significantly compared with the results using mode shapes, suggesting that stress distribution may contain more discriminative information relative to crack location and size than mode shapes.

In addition, the reduction of the number of sampling points to reconstruct the stress distribution in the beam is discussed. It is found that with sampling point reduction, the features indicating damage in the stress distribution grey image become fuzzier, which has led to the decrease of classification accuracy as expected. Therefore, it is necessary to find a suitable compromise between the accuracy of classification and the number of sampling points according to different situations.

The wide applicability of the proposed CNN-based method for damage detection has be verified for a different structure (i.e., beam) and different types of data (i.e., modal data). Different from Chapter 3 and Chapter 4 where propagated wave at high frequencies are used for damage detection, it has been verified in this chapter that the utilization of modal data at low frequencies in the CNN model is effective for damage detection.

Chapter 6

Conclusions and perspectives

6.1 Conclusions

The thesis project is a special issue of health monitoring of composite materials. This work addresses the difficulty in constructing a database for structural health monitoring (SHM) of real composite structures and the inconvenience in conventional expertise-based SHM approaches. Crack damage detection of a thick composite sandwich structure based on a conventional machine learning algorithm Gaussian Process and a deep learning algorithm Convolutional Neural Networks using simulation data was reported successively. The developed machine learning based damage detection model can help detect and localize crack damages in composite sandwich structures.

The research background of composite structures, structural health monitoring methods, the development of machine learning and its applications are firstly reviewed in Chapter 1. The survey starts with the wide application of composite structures and structure damage problems. Then a review of existing structural health monitoring methods, including physical-based and data driven-based methods is given. The introduction of machine learning and its applications in structural health monitoring as well as other fields is then discussed.

The implement of the numerical approach to construct a database for SHM of composite sandwich structure has been introduced in Chapter 2. Models have been constructed and validated by comparing the physical properties obtained numerically with analytical methods. Different damage cases are modeled followed by finite element analysis. Some commonly used feature extraction techniques through FE simulation are then introduced, which help to construct a database for further study.

A conventional machine learning algorithm: Gaussian Process-based method was developed for structural health monitoring of the studied composite sandwich structure in Chapter 3. Discrete wavelet transform (DWT) is used for feature extraction, and GP is used to detect the occurrence of potential crack damages in the structure. Several factors that have impact on classification accuracy have been investigated. The proposed method is proven effective for crack-type damage detection in the studied structure with appropriate factors, such as mother wavelet in DWT, likelihood function and inference method in GP and the number of iterations in the learning process. The pertinence of data from different sampling points on the structure is also investigated. In addition, the effectiveness of the proposed method is verified under simulated environmental noise.

A CNN-based structural health monitoring system is proposed in Chapter 4 due to the advantages of the deep learning algorithm Convolutional Neural Network in image recognition and classification. One-stage CNN model and two-stage CNN model using images processed by Continuous wavelet transform (CWT) are proposed successively. In the one-stage CNN model, most samples with ground truth "crack" are incorrectly classified as "healthy", leading to a poor capability of detecting and localizing crack damage at the same time. However, a two-stage CNN model is then proposed to address the problem, in which the first CNN is used to detect the occurrence of crack damage while the second is used for crack localization. The enrichment of the database is carried out. Accordingly, the lack of variability of the database is solved and the capability of the two-stage CNN model of detecting and localizing crack damage is improved significantly.

Chapter 5 extends the two-stage CNN-based SHM method using modal data at low frequencies in a simply-supported beam to verify the wide applicability of the CNN-based method to different structures and different types of data. Crack localization and damage size prediction in a simply-supported beam is conducted based on mode shapes and stress distribution in the beam instead of the vibration data used in the sandwich structure in Chapter 3 and Chapter 4. Results suggest that stress distribution may contain more discriminative information relative to crack location and size than mode shapes. In addition, it is necessary to find a suitable compromise between the classification accuracy and the number of sampling points according to different situations to obtain an optimal performance.

It is proven that machine learning based health monitoring of composite structures is effective and efficient. Besides, the effectiveness of using modal parameters under low frequencies and propagated waves in high frequencies provides more flexibility and possibilities for SHM.

6.2 Perspectives

From the results obtained in the present research, some perspectives can be planned:

- Evaluate the robustness of the CNN-based method under simulated environmental noise;
- Enrich the crack damage characteristics, e.g., with different orientations, irregular shapes. Extend the machine learning based SHM model to identify more complex characteristics of the damage;
- Conduct experiments on real structures and evaluate the applicability of the proposed damage detection method based on simulated data;
- Construct composite sandwich models with delamination damage which is also common in composite materials. Detect and localize delamination in x-y plane (in the face sheet of sandwich structure) and in z direction (depth).

List of Figures

List of Tables

List of publications

International Journals

- **Z. Liu**, M. Ardabilian, A. Zine, M. Ichchou. Crack damage identification of a thick composite sandwich structure based on Gaussian Processes classification. Composite Structures, 2020: 112825.

  https://doi.org/10.1016/j.compstruct.2020.112825.
- **Z. Liu**, A. Zine, M. Ichchou, M. Ardabilian. Crack damage detection and localization of a thick composite sandwich structure based on Convolutional Neural Network. (To be submitted to Structural Health Monitoring)
- **Z. Liu**, A. Zine, M. Ichchou, M. Ardabilian. Convolutional Neural Network-based structural health monitoring of a beam at low frequencies. (To be submitted to Structural Health Monitoring)

International Conferences and Workshops

- **Z. Liu**, M. Ichchou, M. Ardabilian, A. Zine. Structural Health Monitoring of Composite Sandwich Panels using Machine Learning. 10th Workshop Groupe des Écoles Centrales-Centrale Beihang pour les Sciences de l'ingénieur, 4-5 July 2019, Nante, France.
- **Z. Liu**, A. Zine, M. Ardabilian, M. Ichchou. Structural damage predictions of a composite sandwich panel based on Gaussian Processes. 7th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering, 24-26 June 2019, Crete, Greece.
- **Z. Liu**, M. Ardabilian, M. Ichchou, A. Zine. Simulation-based Health Monitoring of a Composite Sandwich Structure using Convolutional Neural Network. Proceedings of MEDYNA2020, 3rd Euro-Mediterranean Conference on Structural Dynamics and Vibroacoustics, Edited by S. De Rosa, F. Franco, M. Guida, F. Marulo and G. Petrone, Napoli, 2020, ISBN: 9788890648465.
- **Z. Liu**, M. Ardabilian, A. Zine, M. Ichchou.Crack-type damage detection and localization of a thick composite sandwich structure based on Convolutional Neural Networks. International Conference on Noise and Vibration Engineering, 7-9 September 2020, Leuven, Belgium.

- **Z. Liu**, M. Ardabilian, A. Zine, M. Ichchou.CNN-based damage detection and localization of a thick composite sandwich structure. The 13th International Conference on Structural Safety and Reliability (ICOSSAR 2021), June 21-25, 2021, Shanghai, P.R. China.

Bibliography

[1] R. B. Heslehurst, Defects and damage in composite materials and structures, CRC press, 2014. (Cited on page xi.)

[2] J. Zhang, J. Fan, K. Herrmann, Delaminations induced by constrained transverse cracking in symmetric composite laminates, International Journal of Solids and Structures 36 (6) (1999) 813–846. (Cited on page xi.)

[3] A. Bussiba, M. Kupiec, S. Ifergane, R. Piat, T. Böhlke, Damage evolution and fracture events sequence in various composites by acoustic emission technique, Composites science and technology 68 (5) (2008) 1144–1155. (Cited on page xi.)

[4] D. Balageas, C.-P. Fritzen, A. Güemes, Structural health monitoring, Vol. 90, John Wiley & Sons, 2010. (Cited on pages xii and 6.)

[5] S. Alokita, V. Rahul, K. Jayakrishna, V. Kar, M. Rajesh, S. Thirumalini, M. Manikandan, Recent advances and trends in structural health monitoring, in: Structural health monitoring of biocomposites, fibre-reinforced composites and hybrid composites, Elsevier, 2019, pp. 53–73. (Cited on pages xii and 6.)

[6] R. Di Sante, Fibre optic sensors for structural health monitoring of aircraft composite structures: Recent advances and applications, Sensors 15 (8) (2015) 18666–18713. `doi:10.3390/s150818666`.
URL `https://www.mdpi.com/1424-8220/15/8/18666` (Cited on pages xii and 135.)

[7] D. Gay, Composite materials: design and applications, CRC press, 2014. (Cited on page 1.)

[8] R. COURTEAU, Sur les perspectives d'évolution de l'aviation civile à l'horizon 2040 : préserver l'avance de la france et de l'europe, `http://www.senat.fr/rap/r12-658/r12-65815.html` (2013). (Cited on page 2.)

[9] H. N. Wadley, Multifunctional periodic cellular metals, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 364 (1838) (2005) 31–68. (Cited on pages 3 and 135.)

[10] R. Potluri, M. N. Raju, K. R. P. Babu, et al., Finite element analysis ofcellular foam core sandwich structures, Materials Today: Proceedings 4 (2) (2017) 2501–2510. (Cited on page 4.)

[11] M. Venugopal, S. Maharana, K. Badarinarayan, Finite element evaluation of composite sandwich panel under static four point bending load, Journal of Engineering, Science and Technology Management 2 (1) (2013) 1–6. (Cited on page 4.)

[12] L. Wahl, S. Maas, D. Waldmann, A. Zürbes, P. Frères, Shear stresses in honeycomb sandwich plates: Analytical solution, finite element method and experimental verification, Journal of Sandwich Structures & Materials 14 (4) (2012) 449–468. (Cited on page 5.)

[13] Z. Zergoune, M. Ichchou, O. Bareille, B. Harras, R. Benamar, B. Troclet, Assessments of shear core effects on sound transmission loss through sandwich panels using a two-scale approach, Computers & Structures 182 (2017) 227–237. (Cited on page 5.)

[14] L. Guillaumie, Vibroacoustic flexural properties of symmetric honeycomb sandwich panels with composite faces, Journal of Sound and Vibration 343 (2015) 71–103. (Cited on page 5.)

[15] O. Baho, Z. Zergoune, M. Ichchou, B. Harras, R. Benamar, B. Troclet, O. Bareille, On global bending–shear core transition effects for the vibroacoustic of sandwich structures: Analytical and numerical investigations, Composite Structures 154 (2016) 453–463. (Cited on page 5.)

[16] C. Droz, Z. Zergoune, R. Boukadia, O. Bareille, M. Ichchou, Vibro-acoustic optimisation of sandwich panels using the wave/finite element method, Composite Structures 156 (2016) 108–114. (Cited on page 5.)

[17] F. Song, G. Huang, K. Hudson, Guided wave propagation in honeycomb sandwich structures using a piezoelectric actuator/sensor system, Smart Materials and Structures 18 (12) (2009) 125007. (Cited on pages 5 and 14.)

[18] V. Smelyanskiy, V. Hafiychuk, D. Luchinsky, R. Tyson, J. Miller, C. Banks, Modeling wave propagation in sandwich composite plates for structural health monitoring, in: Annual Conference of the Prognostics and Health Management Society, 2011, pp. 1–10. (Cited on page 5.)

[19] C. Droz, O. Bareille, M. N. Ichchou, A new procedure for the determination of structural characteristics of sandwich plates in medium frequencies, Composites Part B: Engineering 112 (2017) 103–111. (Cited on pages 5 and 35.)

[20] S. M. H. Hosseini, U. Gabbert, Numerical simulation of the lamb wave propagation in honeycomb sandwich panels: a parametric study, Composite Structures 97 (2013) 189–201. (Cited on page 5.)

[21] S. M. H. Hosseini, A. Kharaghani, C. Kirsch, U. Gabbert, Numerical simulation of lamb wave propagation in metallic foam sandwich structures: a parametric study, Composite Structures 97 (2013) 387–400. (Cited on page 5.)

[22] M. A. Torres-Arredondo, H. Jung, C.-P. Fritzen, A study of attenuation and acoustic energy anisotropy of lamb waves in multilayered anisotropic media for ndt and shm applications, in: Proceedings of the 6th International Workshop NDT in Progress, Prague, Czech Republic, 2011, pp. 103–115. (Cited on page 5.)

[23] A. Treviso, B. Van Genechten, D. Mundo, M. Tournour, Damping in composite materials: Properties and models, Composites Part B: Engineering 78 (2015) 144–152. (Cited on page 6.)

[24] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, M. Gabbouj, D. J. Inman, A review of vibration-based damage detection in civil structures: From traditional methods to machine learning and deep learning applications, Mechanical Systems and Signal Processing 147 (2021) 107077. `doi:https://doi.org/10.1016/j.ymssp.2020.107077`.
URL `http://www.sciencedirect.com/science/article/pii/S0888327020304635` (Cited on pages 6, 7 and 135.)

[25] R. Lammering, U. Gabbert, M. Sinapius, T. Schuster, P. Wierach, Lamb-Wave Based Structural Health Monitoring in Polymer Composites, Research Topics in Aerospace, Springer International Publishing, 2017.
URL `https://books.google.fr/books?id=zXIzDwAAQBAJ` (Cited on page 6.)

[26] R. Di Sante, Fibre optic sensors for structural health monitoring of aircraft composite structures: Recent advances and applications, Sensors 15 (8) (2015) 18666–18713. (Cited on page 8.)

[27] H. Guo, G. Xiao, N. Mrad, J. Yao, Fiber optic sensors for structural health monitoring of air platforms, Sensors 11 (4) (2011) 3687–3705. (Cited on page 8.)

[28] C. Paget, Active health monitoring of aerospace composite structures by embedded piezoceramic transducers, Ph.D. thesis, Institutionen för flygteknik (2001). (Cited on page 8.)

[29] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. Van de Walle, S. Van Hoecke, Convolutional neural network based fault detection for rotating machinery, Journal of Sound and Vibration 377 (2016) 331–345. (Cited on pages 10, 27 and 135.)

[30] L. Yam, Y. Yan, J. Jiang, Vibration-based damage detection for composite structures using wavelet transform and neural network identification, Composite Structures 60 (4) (2003) 403 – 412. `doi:https://doi.org/10.1016/S0263-8223(03)00023-0`.
URL `http://www.sciencedirect.com/science/article/pii/S0263822303000230` (Cited on pages 10, 11, 13, 23 and 135.)

[31] S. Charoenphan, M. Plesha, L. Bank, Simulation of crack growth in composite material shell structures, International journal for numerical methods in engineering 60 (14) (2004) 2399–2417. (Cited on page 10.)

[32] S. Sikdar, P. Kudela, M. Radzieński, A. Kundu, W. Ostachowicz, Online detection of barely visible low-speed impact damage in 3d-core sandwich composite structure, Composite Structures 185 (2018) 646–655. (Cited on page 10.)

[33] A. Klepka, L. Pieczonka, W. J. Staszewski, F. Aymerich, Impact damage detection in laminated composites by non-linear vibro-acoustic wave modulations, Composites Part B: Engineering 65 (2014) 99–108. (Cited on page 10.)

[34] P. M. Mujumdar, S. Suryanarayan, Flexural vibrations of beams with delaminations, Journal of sound and vibration 125 (3) (1988) 441–461. (Cited on page 10.)

[35] G. F. Gomes, F. A. de Almeida, D. M. Junqueira, S. S. da Cunha Jr, A. C. Ancelotti Jr, Optimized damage identification in cfrp plates by reduced mode shapes and ga-ann methods, Engineering Structures 181 (2019) 111–123. (Cited on pages 10 and 24.)

[36] C. Droz, O. Bareille, J.-P. Lainé, M. N. Ichchou, Wave-based shm of sandwich structures using cross-sectional waves, Structural Control and Health Monitoring 25 (2) (2018) e2085. (Cited on pages 10, 11 and 135.)

[37] K. Zhu, M. Chen, Q. Lu, B. Wang, D. Fang, Debonding detection of honeycomb sandwich structures using frequency response functions, Journal of Sound and Vibration 333 (21) (2014) 5299 – 5311. `doi:https://doi.org/10.1016/j.jsv.2014.05.023`.
URL `http://www.sciencedirect.com/science/article/pii/S0022460X14004131` (Cited on page 12.)

[38] M. Anaya, D. A. Tibaduiza, F. Pozo, A bioinspired methodology based on an artificial immune system for damage detection in structural health monitoring, Shock and Vibration 2015 (2015). (Cited on page 12.)

[39] W. Weijtjens, T. Verbelen, E. Capello, C. Devriendt, Vibration based structural health monitoring of the substructures of five offshore wind turbines, Procedia engineering 199 (2017) 2294–2299. (Cited on page 12.)

[40] E. K. Chalouhi, I. Gonzalez, C. Gentile, R. Karoumi, Vibration-based shm of railway bridges using machine learning: The influence of temperature on the health prediction, in: International Conference on Experimental Vibration Analysis for Civil Engineering Structures, Springer, 2017, pp. 200–211. (Cited on page 12.)

[41] A. Katunin, Vibration-based spatial damage identification in honeycomb-core sandwich composite structures using wavelet analysis, Composite Structures 118 (2014) 385 – 391. `doi:https://doi.org/10.1016/j.compstruct.2014.08.010`.
URL `http://www.sciencedirect.com/science/article/pii/S0263822314004000` (Cited on page 12.)

[42] A. Katunin, H. Lopes, J. A. dos Santos, Identification of multiple damage using modal rotation obtained with shearography and undecimated wavelet transform, Mechanical Systems and Signal Processing 116 (2019) 725–740. (Cited on page 13.)

[43] J. Zhou, Z. Li, J. Chen, Damage identification method based on continuous wavelet transform and mode shapes for composite laminates with cutouts, Composite Structures 191 (2018) 12–23. (Cited on page 13.)

[44] N. D. A. Farrar C R, Doebling S W, Vibration-based structural damage identification, Philosophical Transactions of the Royal Society of London A. 359 (1778) (2001) 131–149. `doi:https://doi.org/10.1098/rsta.2000.0717`. (Cited on page 13.)

[45] W. Bayissa, N. Haritos, S. Thelandersson, Vibration-based structural damage identification using wavelet transform, Mechanical Systems and Signal Processing 22 (5) (2008) 1194 – 1215. `doi:https://doi.org/10.1016/j.ymssp.2007.11.001`.
URL `http://www.sciencedirect.com/science/article/pii/S0888327007002324` (Cited on page 13.)

[46] A. Katunin, Nondestructive damage assessment of composite structures based on wavelet analysis of modal curvatures: state-of-the-art review and description of wavelet-based damage assessment benchmark, Shock and Vibration 2015 (2015). (Cited on page 13.)

[47] M. R. Taha, A. Noureldin, J. Lucero, T. Baca, Wavelet transform for structural health monitoring: a compendium of uses and features, Structural health monitoring 5 (3) (2006) 267–295. (Cited on page 13.)

[48] Y. Zou, L. Tong, G. P. Steven, Vibration-based model-dependent damage (delamination) identification and health monitoring for composite structures—a review, Journal of Sound and vibration 230 (2) (2000) 357–378. (Cited on page 13.)

[49] S. W. Doebling, C. R. Farrar, M. B. Prime, et al., A summary review of vibration-based damage identification methods, Shock and vibration digest 30 (2) (1998) 91–105. (Cited on page 13.)

[50] S. Das, P. Saha, S. Patro, Vibration-based damage detection techniques used for health monitoring of structures: a review, Journal of Civil Structural Health Monitoring 6 (3) (2016) 477–507. (Cited on page 13.)

[51] S. Sikdar, S. Banerjee, Identification of disbond and high density core region in a honeycomb composite sandwich structure using ultrasonic guided waves, Composite Structures 152 (2016) 568 – 578. `doi:https://doi.org/10.1016/j.compstruct.2016.05.064`.
URL `http://www.sciencedirect.com/science/article/pii/S0263822316306444` (Cited on pages 13 and 14.)

[52] S. Mustapha, L. Ye, D. Wang, Y. Lu, Assessment of debonding in sandwich cf/ep composite beams using a0 lamb wave at low frequency, Composite structures 93 (2) (2011) 483–491. (Cited on page 13.)

[53] K. Diamanti, C. Soutis, J. Hodgkinson, Lamb waves for the non-destructive inspection of monolithic and sandwich composite beams, Composites Part A: Applied science and manufacturing 36 (2) (2005) 189–195. (Cited on page 13.)

[54] X. Yu, M. Ratassepp, Z. Fan, Damage detection in quasi-isotropic composite bends using ultrasonic feature guided waves, Composites Science and Technology 141 (2017) 120 – 129. `doi:https://doi.org/10.1016/j.compscitech.2017.01.011`.
URL `http://www.sciencedirect.com/science/article/pii/S0266353816312192` (Cited on page 13.)

[55] P. Aryan, A. Kotousov, C.-T. Ng, B. Cazzolato, A model-based method for damage detection with guided waves, Structural Control and Health Monitoring 24 (3) (2017) e1884, e1884 STC-15-0156.R2. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/stc.1884, doi:10.1002/stc.1884.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/stc.1884 (Cited on page 14.)

[56] N. Chakraborty, V. Rathod, D. R. Mahapatra, S. Gopalakrishnan, Guided wave based detection of damage in honeycomb core sandwich structures, NDT & E International 49 (2012) 27 – 33. doi:https://doi.org/10.1016/j.ndteint.2012.03.008.
URL http://www.sciencedirect.com/science/article/pii/S0963869512000400 (Cited on page 14.)

[57] S. Thwaites, N. Clark, Non-destructive testing of honeycomb sandwich structures using elastic waves, Journal of Sound and Vibration 187 (2) (1995) 253 – 269. doi:https://doi.org/10.1006/jsvi.1995.0519.
URL http://www.sciencedirect.com/science/article/pii/S0022460X85705196 (Cited on page 14.)

[58] M. Mitra, S. Gopalakrishnan, Guided wave based structural health monitoring: A review, Smart Materials and Structures 25 (5) (2016) 053001. (Cited on page 14.)

[59] C. Willberg, S. Duczek, J. M. Vivar-Perez, Z. A. B. Ahmad, Simulation Methods for Guided Wave-Based Structural Health Monitoring: A Review, Applied Mechanics Reviews 67 (1), 010803 (01 2015). arXiv:https://asmedigitalcollection.asme.org/appliedmechanicsreviews/article-pdf/67/1/010803/2938559/amr_067_01_010803.pdf, doi:10.1115/1.4029539.
URL https://doi.org/10.1115/1.4029539 (Cited on page 14.)

[60] Z. Su, L. Ye, Y. Lu, Guided lamb waves for identification of damage in composite structures: A review, Journal of sound and vibration 295 (3-5) (2006) 753–780. (Cited on page 14.)

[61] M. Radzienski, M. Cao, X. Wei, P. Kudela, W. Ostachowicz, Combined vibration and guided wave-based approach for composite panels health assessment, in: T. Kundu (Ed.), Health Monitoring of Structural and Biological Systems 2017, Vol. 10170, International Society for Optics and Photonics, SPIE, 2017,

pp. 499 – 506. `doi:10.1117/12.2260425`.
URL `https://doi.org/10.1117/12.2260425` (Cited on page 14.)

[62] A. Katunin, K. Dragan, M. Dziendzikowski, Damage identification in aircraft composite structures: A case study using various non-destructive testing techniques, Composite structures 127 (2015) 1–9. (Cited on page 15.)

[63] M. Azimi, A. D. Eslamlou, G. Pekcan, Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review, Sensors 20 (10) (2020) 2778. (Cited on pages 16, 17 and 135.)

[64] N. Cristianini, J. Shawe-Taylor, D. Shawe-Taylor, I. Books24x7, C. U. Press, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, 2000.
URL `https://books.google.fr/books?id=_PXJn_cxv0AC` (Cited on page 20.)

[65] S. S. Pradhan, W. H. Ward, K. Hacioglu, J. H. Martin, D. Jurafsky, Shallow semantic parsing using support vector machines, in: Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004, 2004, pp. 233–240. (Cited on page 20.)

[66] G. M. Foody, A. Mathur, A relative evaluation of multiclass image classification by support vector machines, IEEE Transactions on geoscience and remote sensing 42 (6) (2004) 1335–1343. (Cited on page 20.)

[67] M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, D. Haussler, Knowledge-based analysis of microarray gene expression data by using support vector machines, Proceedings of the National Academy of Sciences 97 (1) (2000) 262–267. (Cited on page 20.)

[68] S. Khan, T. Yairi, A review on the application of deep learning in system health management, Mechanical Systems and Signal Processing 107 (2018) 241–265. (Cited on page 21.)

[69] J. Schmidhuber, Deep learning in neural networks: An overview, Neural networks 61 (2015) 85–117. (Cited on page 22.)

[70] R. J. O'Brien, J. M. Fontana, N. Ponso, L. Molisani, A pattern recognition system based on acoustic signals for fault detection on composite materials, European Journal of Mechanics-A/Solids 64 (2017) 1–10. (Cited on page 23.)

[71] M. A. Torres-Arredondo, D. A. Tibaduiza-Burgos, An acousto-ultrasonics approach for probabilistic modelling and inference based on gaussian processes, Structural Control and Health Monitoring 25 (6) (2018) e2178. (Cited on page 24.)

[72] H. Teimouri, A. S. Milani, J. Loeppky, R. Seethaler, A gaussian process–based approach to cope with uncertainty in structural health monitoring, Structural Health Monitoring 16 (2) (2017) 174–184. (Cited on page 24.)

[73] A. Khan, H. S. Kim, Assessment of delaminated smart composite laminates via system identification and supervised learning, Composite Structures 206 (2018) 354–362. (Cited on page 24.)

[74] P. R. Spina, G. Torella, M. Venturini, The use of expert systems for gas turbine diagnostics and maintenance, in: ASME Turbo Expo 2002: Power for Land, Sea, and Air, American Society of Mechanical Engineers Digital Collection, 2002, pp. 127–134. (Cited on page 24.)

[75] W. Yan, Application of random forest to aircraft engine fault diagnosis, in: The Proceedings of the Multiconference on" Computational Engineering in Systems Applications", Vol. 1, IEEE, 2006, pp. 468–475. (Cited on page 24.)

[76] M. A. Zaidan, R. F. Harrison, A. R. Mills, P. J. Fleming, Bayesian hierarchical models for aerospace gas turbine engine prognostics, Expert Systems with Applications 42 (1) (2015) 539–553. (Cited on page 24.)

[77] P. Kankar, S. C. Sharma, S. Harsha, Fault diagnosis of ball bearings using continuous wavelet transform, Applied Soft Computing 11 (2) (2011) 2300 – 2312, the Impact of Soft Computing for the Progress of Artificial Intelligence. `doi:https://doi.org/10.1016/j.asoc.2010.08.011`.
URL `http://www.sciencedirect.com/science/article/pii/S1568494610002139` (Cited on page 24.)

[78] H. Zhou, J. Chen, G. Dong, H. Wang, H. Yuan, Bearing fault recognition method based on neighbourhood component analysis and coupled hidden markov model, Mechanical Systems and Signal Processing 66 (2016) 568–581. (Cited on page 24.)

[79] H. Kumar, T. Ranjit Kumar, M. Amarnath, V. Sugumaran, Fault diagnosis of bearings through vibration signal using bayes classifiers, International Journal of Computer Aided Engineering and Technology 6 (1) (2014) 14–28. (Cited on page 24.)

[80] C. Krittanawong, K. W. Johnson, R. S. Rosenson, Z. Wang, M. Aydar, U. Baber, J. K. Min, W. W. Tang, J. L. Halperin, S. M. Narayan, Deep learning for cardiovascular medicine: a practical primer, European heart journal 40 (25) (2019) 2058–2073. (Cited on page 25.)

[81] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, et al., Identifying medical diagnoses and treatable diseases by image-based deep learning, Cell 172 (5) (2018) 1122–1131. (Cited on page 25.)

[82] H. Kim, E. Ahn, M. Shin, S.-H. Sim, Crack and noncrack classification from concrete surface images using machine learning, Structural Health Monitoring 18 (3) (2019) 725–738. (Cited on page 25.)

[83] F.-C. Chen, M. R. Jahanshahi, Nb-cnn: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion, IEEE Transactions on Industrial Electronics 65 (5) (2017) 4392–4400. (Cited on page 25.)

[84] C. M. Yeum, J. Choi, S. J. Dyke, Automated region-of-interest localization and classification for vision-based visual assessment of civil infrastructure, Structural Health Monitoring 18 (3) (2019) 675–689. (Cited on page 25.)

[85] Y. Narazaki, V. Hoskere, T. A. Hoang, B. F. Spencer Jr, Automated bridge component recognition using video data, arXiv preprint arXiv:1806.06820 (2018). (Cited on page 25.)

[86] B. Kim, S. Cho, Automated vision-based detection of cracks on concrete surfaces using a deep learning technique, Sensors 18 (10) (2018) 3452. (Cited on page 25.)

[87] T.-C. Huynh, J.-H. Park, H.-J. Jung, J.-T. Kim, Quasi-autonomous bolt-loosening detection method using vision-based deep learning and image processing, Automation in Construction 105 (2019) 102844. (Cited on page 25.)

[88] V. Hoskere, Y. Narazaki, T. A. Hoang, B. F. Spencer Jr, Towards automated post-earthquake inspections with deep learning-based condition-aware models, arXiv preprint arXiv:1809.09195 (2018). (Cited on page 25.)

[89] A. Khan, D.-K. Ko, S. C. Lim, H. S. Kim, Structural vibration-based classification and prediction of delamination in smart composite laminates using deep learning neural network, Composites Part B: Engineering 161 (2019) 586–594. (Cited on page 26.)

[90] M. Meng, Y. J. Chua, E. Wouterson, C. P. K. Ong, Ultrasonic signal classification and imaging system for composite materials via deep convolutional neural networks, Neurocomputing 257 (2017) 128–135. (Cited on page 26.)

[91] O. Avci, O. Abdeljaber, S. Kiranyaz, D. Inman, Structural damage detection in real time: implementation of 1d convolutional neural networks for shm applications, in: Structural Health Monitoring & Damage Detection, Volume 7, Springer, 2017, pp. 49–54. (Cited on page 26.)

[92] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D. J. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, Journal of Sound and Vibration 388 (2017) 154–170. (Cited on page 26.)

[93] F. Jia, Y. Lei, J. Lin, X. Zhou, N. Lu, Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data, Mechanical Systems and Signal Processing 72 (2016) 303–315. (Cited on page 26.)

[94] X. Guo, L. Chen, C. Shen, Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis, Measurement 93 (2016) 490–502. (Cited on page 27.)

[95] W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load, Mechanical Systems and Signal Processing 100 (2018) 439–453. (Cited on page 27.)

[96] W. Zhang, G. Peng, C. Li, Y. Chen, Z. Zhang, A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals, Sensors 17 (2) (2017) 425. (Cited on page 27.)

[97] T. Ince, S. Kiranyaz, L. Eren, M. Askar, M. Gabbouj, Real-time motor fault detection by 1-d convolutional neural networks, IEEE Transactions on Industrial Electronics 63 (11) (2016) 7067–7075. (Cited on page 27.)

[98] D. Lee, V. Siu, R. Cruz, C. Yetman, Convolutional neural net and bearing fault analysis, in: Proceedings of the International Conference on Data Mining (DMIN), The Steering Committee of The World Congress in Computer Science, Computer . . . , 2016, p. 194. (Cited on page 27.)

[99] X. Ding, Q. He, Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis, IEEE Transactions on Instrumentation and Measurement 66 (8) (2017) 1926–1935. (Cited on page 27.)

[100] Y.-J. Cha, W. Choi, O. Büyüköztürk, Deep learning-based crack damage detection using convolutional neural networks, Computer-Aided Civil and Infrastructure Engineering 32 (5) (2017) 361–378. (Cited on page 28.)

[101] K. Jang, N. Kim, Y.-K. An, Deep learning–based autonomous concrete crack evaluation through hybrid image scanning, Structural Health Monitoring 18 (5-6) (2019) 1722–1737. (Cited on page 28.)

[102] Y. Xu, Y. Bao, J. Chen, W. Zuo, H. Li, Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images, Structural Health Monitoring 18 (3) (2019) 653–674. (Cited on page 28.)

[103] Y. Bao, Z. Tang, H. Li, Y. Zhang, Computer vision and deep learning–based data anomaly detection method for structural health monitoring, Structural Health Monitoring 18 (2) (2019) 401–421. (Cited on page 28.)

[104] O. C. Zienkiewicz, R. L. Taylor, The finite element method for solid and structural mechanics, Elsevier, 2005. (Cited on page 30.)

[105] B. Banerjee, J. Chen, A. Kathirgamanathan, Comparison of ansys elements shell181 and solsh190, Res. Rep., Univ. of Auckland, New Zealand (2011). (Cited on page 30.)

[106] P. Swanson, Ansys, inc. theory, Theoretical Manual, Release 11 (2007). (Cited on pages 31, 135 and 136.)

[107] E. Nilsson, A. Nilsson, Prediction and measurement of some dynamic properties of sandwich structures with honeycomb and foam cores, Journal of sound and vibration 251 (3) (2002) 409–430. (Cited on page 34.)

[108] S. Malek, L. Gibson, Effective elastic properties of periodic hexagonal honeycombs, Mechanics of Materials 91 (2015) 226–240. (Cited on pages 35, 39, 40 and 136.)

[109] W. K. Ngui, M. S. Leong, L. M. Hee, A. M. Abdelrhman, Wavelet analysis: mother wavelet selection methods, in: Applied mechanics and materials, Vol. 393, Trans Tech Publ, 2013, pp. 953–958. (Cited on page 52.)

[110] S. J. Russell, P. Norvig, Artificial intelligence: a modern approach, Malaysia; Pearson Education Limited, 2016. (Cited on page 55.)

[111] C. M. Bishop, Pattern recognition and machine learning, springer, 2006. (Cited on pages 58 and 61.)

[112] M. J. Beal, Variational algorithms for approximate bayesian inference, Ph.D. thesis, UCL (University College London) (2003). (Cited on page 61.)

[113] T. P. Minka, Expectation propagation for approximate bayesian inference, arXiv preprint arXiv:1301.2294 (2013). (Cited on page 61.)

[114] D. Tibaduiza, L. Mujica, J. Rodellar, Damage classification in structural health monitoring using principal component analysis and self-organizing maps, Structural Control and Health Monitoring 20 (10) (2013) 1303–1316. (Cited on page 63.)

[115] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (7553) (2015) 436–444. (Cited on page 78.)

dernière page de la thèse

AUTORISATION DE SOUTENANCE

Vu les dispositions de l'arrêté du 25 mai 2016,

Vu la demande du directeur de thèse

Monsieur M. ICHCHOU

et les rapports de

M. S. DE ROSA
Professeur - Dept of Industrial Engineering - Aerospace section
University of Naples Federico II - Via Giulio Cesare Cortese, 29 - 80133 Napoli NA -Italie

et de

M. W. LARBI
Professeur - Laboratoire en Mécanique des Structures et des Systèmes Couplés - CNAM
292 rue Saint-Martin - 75141 Paris cedex 03

Monsieur LIU Zeyu

est autorisé à soutenir une thèse pour l'obtention du grade de **DOCTEUR**

Ecole doctorale Mécanique, Energétique, Génie Civil et Acoustique

Fait à Ecully, le 5 janvier 2021

P/Le directeur de l'E.C.L.
Le directeur des Etudes

Grégory VIAL