UNIVERSITÀ DEGLI STUDI DELL'AQUILA

Department of Information Engineering, Computer Science, and Mathematics (DISIM)

Multi-Agent System for Railways Steel Bridges Health Monitoring by Vibration Based Sensory Data and Fuzzy Logic

PhD in

Information And Communication Technologies

Curriculum n. 2: Emerging Computational Models, Software Architectures, And Intelligent Systems

XXXVII Doctoral Cycle

SSD: INF/01

Author: MUHAMMAD ASAD

Advisor	**Co-Advisor**	**Co-Ordinator**
Stefania Costantini	Giovanni De Gasperis	Davide Di Ruscio

A.A 2023-2024

"It seems probable that once the machine thinking method had started, it would not take long to outstrip our feeble powers."

- Alan Turing.

Acknowledgment

First, I would like to thank my PhD supervisor, Stefania Costantini, for her invaluable help, my PhD Co-Supervisor Giovanni De Gasperis for his advice, and Prof Rocco Alaggio for their valuable consideration during my doctoral studies. Their extensive knowledge, inspiring ideas, and immense dedication to conducting excellent research have taught me a lot. They both created a pleasant work environment and consistently encouraged and motivated me to strive for high-quality work. Beyond this, they have also been a mentor and guardian for me.

In addition, I also thank Vittorio Cortellesa and Davide Di Ruscio for their efforts during the final examination. Their thoughtfulness and willingness to go above and beyond to provide a comfortable working environment were greatly appreciated. Finally, I would like to express my deepest love and gratitude to my parents, my wife, my daughter, and siblings. Over the years, their unwavering belief in me and support for my decisions have been a constant source of strength. My goal is to fulfill their expectations and I cannot adequately express how grateful I am for everything they have done to raise, educate, and prepare me for the future.

Muhammad Asad
L'Aquila, Italy
May 30, 2025

Declaration

This thesis is the original work of the author: *Muhammad Asad*, composed with the collaboration of *Giovanni De Gasperis*, *Rocco Alaggio*, and *Stefania Costantini*.

Abstract

The Structural Health Monitoring (SHM) of railway steel bridges is critical to ensure transportation safety, reliability, and timely maintenance. This thesis proposes a comprehensive vibration-based SHM framework that integrates signal processing, deep learning, and fuzzy reasoning to detect, classify, and interpret structural damage.

This thesis proposes a comprehensive Multi-Agent System (MAS) for SHM of railway steel bridges using vibration-based sensory data, machine learning algorithms, and fuzzy logic reasoning. The system integrates diverse artificial intelligence agents; responsible for sensing, perception, classification, reasoning, and alert generation; into a cohesive architecture designed to detect, localize, and assess structural damage in real-time.

Vibration signals collected from accelerometer sensors are first processed through the Hankel matrix construction and Singular Value Decomposition (SVD). These transformed signals are converted into statistical features and spectrograms using the Short-Time Fourier Transform (STFT), capturing the temporal and spectral evolution of bridge vibrations to be used in time domain SHM detection and Frequency Domain classifications respectively. These statistical features and spectrograms form the shared environment observed by multiple classifier agents.

We implement and compare classical machine learning models (SVM, KNN, RF, MLP, Ensemble), a custom Radial Basis Function Network (RBF-Net), and deep learning architectures (ResNet50, DenseNet121, EfficientNetB0). Additionally, hybrid DNN-Transformer fusion models (TRNet, TDNet, TENet) are proposed to combine local feature extraction with global temporal attention. These models collectively support bridge scenario classification and damage intensity estimation, achieving accuracies exceeding 90%.

An Explainable Boosting Machine (EBM) agent further enhances transparency by highlighting the influence of input features such as skewness, kurtosis, and signal energy on classification outcomes. Finally, a Fuzzy Logic Reasoning Agent is introduced, which aggregates outputs from perception agents and applies expert-defined fuzzy rules to generate human-readable alerts (*At-*

tention, *Pre-Alarm*, *Alarm*).

The proposed MAS architecture ensures robustness, interpretability, and modularity, key requirements for intelligent infrastructure monitoring. The work advances SHM by demonstrating that distributed intelligence, deep learning, and fuzzy reasoning can be jointly leveraged to build reliable, interpretable, and scalable monitoring systems for safety-critical railway bridges.

Keywords: Accelerometer, Anomaly Detection, SHM, Machine Learning (ML), Frequency Domain, ResNet50, DenseNet121, Fuzzy Logic, Reasoning Layer, Perception Layer .

Table of Content

List of Figures

List of Tables

Chapter 1

Introduction

"Artificial Intelligence is the new electricity."

—Andrew Ng.

The railway network plays a critical role in national and transnational infrastructure systems by supporting efficient, reliable, and sustainable transportation of passengers and goods. It serves as a backbone for economic activity, enabling regional connectivity, reducing road congestion, and lowering greenhouse gas emissions compared to other transport modes. In the European context, railways are integral to the goals of the Green Deal and the Trans-European Transport Network (TEN-T), aiming to promote environmentally friendly mobility solutions. Railway infrastructure-particularly bridges, tunnels, and viaducts, is essential for maintaining uninterrupted routes across complex geographical terrains. The safe operation and structural integrity of these components are vital for ensuring service continuity, public safety, and economic resilience. As demand for rail transport continues to grow, maintaining the health of railway assets, especially aging steel bridges, has become a strategic priority across both national and European Union (EU)-level transport policies.

Given the critical role that bridges play in the overall railway infrastructure

network, it has become imperative to develop reliable methods for their continuous monitoring. In response to this need, Structural Health Monitoring (SHM) systems have been extensively developed and deployed for the surveillance, evaluation, and condition assessment of existing long span bridges. Among these, recently developed long term SHM systems represent state of the art solutions, enabling the ongoing assessment of serviceability, safety, and sustainability of such structures

1.1 Overview

The railway network in the European Union (EU) is one of the most extensive and integrated in the world, comprising over 200,000 kilometers of active railway lines that facilitate both passenger and freight transport across member states. As part of the EU's commitment to sustainable and efficient transport, significant investments have been made in the modernization and expansion of rail infrastructure, particularly along the Trans-European Transport Network (TEN-T) corridors. The European Union (EU) boasts one of the most extensive and integrated railway networks globally, encompassing approximately **200,947 kilometers** of active railway lines shown in Fig.1.1 as of 2023. This network is pivotal for both passenger and freight transport, underpinning economic activities and promoting sustainable mobility across member states [3].

An essential component of this infrastructure is the vast number of railway bridges, many of which are constructed from steel due to its high strength-to-weight ratio, durability, and adaptability to long spans. Steel railway bridges play a critical role in maintaining uninterrupted rail connectivity over rivers, valleys, and other challenging terrains; especially in countries with mountainous regions or dense urban networks.

These bridges are subject to dynamic loads and environmental stressors, making them prime candidates for Structural Health Monitoring (SHM). Ensuring the integrity and safety of railway steel bridges is vital to the overall reliability of the EU railway network, prompting increasing research into advanced monitoring systems that utilize vibration-based sensors, machine learning, and real-time data analysis. Furthermore, Italian railway network is one of the most developed and efficient in Europe, spanning approximately 16,700 kilometers, of which around 11,500 kilometers are electrified. Managed primarily by Rete Ferroviaria Italiana (RFI), the network plays a crucial role in connecting Italy's major cities and regions. Italy is renowned for its high speed rail service, known as Alta Velocità (AV), which includes lines such as Turin-Milan-Rome-Naples and Venice-Bologna-Florence-Rome. These routes, as shown in Fig. 1.2, are

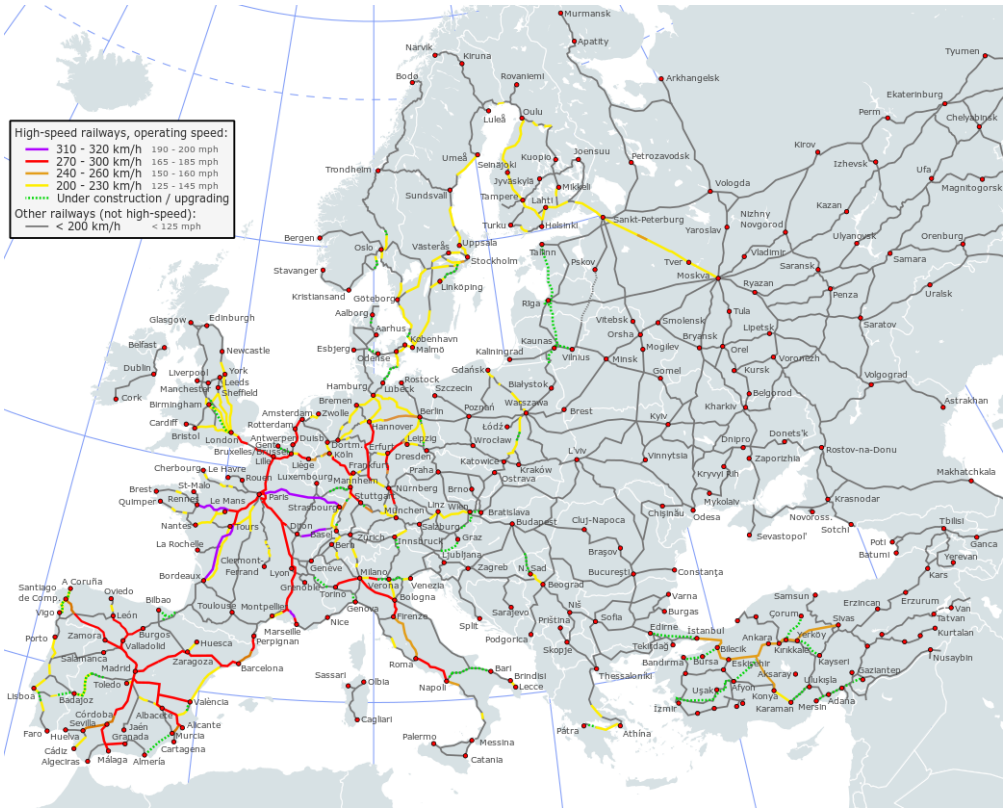


Figure 1.1: High Speed Railway Map of Europe. Source:RFI [1].

served by Frecciarossa, Frecciargento, and Frecciabianca trains operated by Trenitalia, as well as by the private operator Italo, offering competitive and efficient services reaching speeds of up to 300-360 km/h. The major hubs such as Rome Termini, Milan Centrale, and Naples Centrale facilitate smooth passenger transitions across regions.

In addition to high speed services, Italy's regional rail system connects smaller towns and suburban areas, operated by companies such as Trenitalia and Trenord (in Lombardy). The network is also integrated into the larger European rail system, contributing to cross-border connectivity and aligning with the EU's goal of sustainable and interoperable transport across member states.

Italy has an extensive network of railway bridges and viaducts that are vital to maintaining seamless rail connectivity throughout the country's diverse geography, which includes mountains, valleys, and rivers. These structures form a crucial part of the national and regional railway infrastructure, particularly in areas such as the Apennines, Alps, and coastal regions where the terrain makes standard rail construction challenging.

Integral to this vast railway system are the numerous bridges that facilitate uninterrupted rail connectivity over rivers, valleys, and other challenging terrains. A comprehensive survey conducted by European railway administrations

Figure 1.2: Railway Network Map of Italy. Source: RFI [1].

identified over **220,000 railway bridges** across the continent. Among these, approximately **22% are steel beam bridges**, highlighting the significant role of steel structures in the railway infrastructure [4].

Notably, a substantial portion of these steel railway bridges are aging:

- **28%** are over 100 years old [5].
- **40%** are between 50 and 100 years old [6].

This aging infrastructure presents challenges in terms of maintenance and safety, necessitating advanced monitoring and assessment techniques. *Structural Health Monitoring (SHM)* systems have become increasingly vital, enabling continuous surveillance and evaluation of bridge conditions. By employing sensor technologies and data analysis, SHM facilitates early detection of potential issues and anomalies, thereby enhancing safety and extending the service life of these critical structures.

This aging infrastructure presents challenges in terms of maintenance and safety, necessitating advanced monitoring and assessment techniques. Structural Health Monitoring (SHM) systems have become increasingly vital, enabling continuous surveillance and evaluation of bridge conditions. By employing sensor technologies and data analysis, SHM facilitates early detection of potential issues, thereby enhancing safety and extending the service life of these critical structures.

Many of Italy's railway bridges date back to the 19th and early 20th centuries, built during the early expansion of the rail network. Although numerous bridges have been modernized, reinforced, or complete reconstruction, a significant significant portion still requires ongoing maintenance and structural monitoring to ensure safety and compliance with current standards.

Following several high profile infrastructure failures in recent decades; most notably the Morandi Bridge collapse in Genoa (2018), although it was a road bridge; it prompted a nationwide reassessment of all transport infrastructure, including railway bridges. As a result, Rete Ferroviaria Italiana (RFI) has intensified inspections, digital monitoring systems, and investments in structural health monitoring (SHM) systems to track wear, stress, and possible damage due to aging, weather events, or seismic activity.

Furthermore, Italy, like other EU countries, is increasingly adopting smart infrastructure technologies, such as fiber optic sensors, Artificial Intelligence (AI) based damage detection, and drones for visual inspections, to maintain bridge safety. Projects under the EU funded TEN-T corridors also include the modernization or replacement of critical railway bridges to support high speed rail and increased freight loads. Despite progress, challenges remain, especially in regions with older infrastructure or limited funding for upgrades.

In summary, Italy's railway bridges are a vital but aging part of the transport system, many in good operational condition but requiring continuous investment, inspection, and modernization to meet the demands of modern rail transport and ensure long term safety.

1.2 Thesis Context

The Structural Health Monitoring (SHM) of railway steel bridges plays a pivotal role in modern infrastructure management by ensuring the long term safety, serviceability, and resilience of these critical structures. Given the aging nature of many railway steel bridges; subjected to dynamic loads, environmental exposure, and material degradation, there is an urgent need for intelligent and automated monitoring systems capable of assessing their health in real-time.

This thesis addresses the challenge of monitoring and interpreting bridge behavior under varying operational and environmental conditions through a multilayered SHM framework. The proposed system is structured into three primary layers: the data layer, the perception layer, and the reasoning layer.

The data layer is responsible for the continuous collection of raw vibration signals using accelerometer sensors installed at strategic points on the bridge. These sensors capture the dynamic structural responses that occur as trains pass over the bridge, reflecting variations caused by train loads, temperature changes, wind effects, and potential damage progression.

In the perception layer, these raw signals are processed through time-frequency transformation techniques (such as the Short-Time Fourier Transform) and statistical feature extraction methods. Key features such as kurtosis, Root Mean Square (RMS), peak frequency, and cumulative difference are derived from the signal to characterize the structural state. Machine learning algorithms; including classical models like Support Vector Machine (SVM), k-Nearest Neighbors (KNN), and Random Forest, as well as deep learning models such as Convolutional Neural Networks (CNNs) are employed to classify bridge conditions and detect early signs of damage. The perception layer also utilizes data slicing techniques (for example, Hankel matrix transformation) and data augmentation to improve model generalization and sensitivity to subtle structural changes.

The reasoning layer builds upon the outputs of the perception layer, integrating domain knowledge through a fuzzy rule-based system. This layer interprets the predicted damage indicators and assigns linguistic labels to damage intensity (for example, "no damage", "low damage", "medium damage", "severe damage"), generating actionable alerts. These interpretable alerts help bridge operators prioritize inspection and maintenance actions without requiring in-depth technical interpretation of model outputs.

This thesis focuses in particular on diverse bridge damage scenarios and damage intensity levels, simulating real world structural degradation patterns such as fatigue cracks, joint loosening, and material loss. These scenarios are modeled both analytically (for example, via Finite Element Model (FEM) simulations) and experimentally using sensor datasets.

The integration of the reasoning layer with the perception layer provides a robust decision support framework that enhances the interpretability and trustworthiness of the SHM system. As part of the research, various damage classification models are validated under realistic conditions, and the performance is benchmarked using both simulated and real world datasets.

Ultimately, this work contributes toward developing an intelligent SHM

framework capable of supporting the continuous and autonomous monitoring of railway steel bridges, ensuring structural integrity, minimizing downtime, and enabling predictive maintenance planning. Future extensions may include incorporating edge computing, wireless sensor networks, and adaptive fuzzy logic for real-time deployment across large scale railway networks.

1.3 Thesis Motivations

The health and longevity of railway steel bridges are critical to ensuring the safe and efficient functioning of national and regional rail networks. These bridges are subject to continuous dynamic loading from passing trains, environmental variations, material fatigue, and in some regions, seismic activity. Over time, such stressors can lead to degradation in structural components, posing serious safety risks and financial burdens if not detected and addressed in a timely manner.

To address this challenge, modern Structural Health Monitoring (SHM) systems increasingly leverage a layered architecture comprising a data layer, perception layer, and reasoning layer; each responsible for specific functions that contribute to the overall intelligence and reliability of the system.

At the foundation, the data layer involves the deployment of vibration-based sensors (for example, accelerometers) strategically placed on the bridge structure. These sensors continuously collect high frequency time series data that capture the dynamic behavior of the bridge under operational conditions. These raw sensory data serves as the primary evidence for detecting early signs of damage or abnormal structural response.

The perception layer processes this sensor data through signal processing techniques (for example, Short Time Fourier Transform, Hankel matrix based slicing) and statistical feature extraction (for example, kurtosis, RMS, spectral entropy). Machine learning models such as classical classifiers (for example, SVM, Random Forest, MLP) and deep learning architectures (for example, CNNs, Transformer-based fusion models) are then used to detect, classify, and quantify structural damage with improved accuracy and robustness.

Building upon the perception outputs, the reasoning layer incorporates domain knowledge through a fuzzy rule-based inference system. This layer interprets the extracted features and model predictions to generate high-level, human-understandable alerts regarding damage severity. By assigning linguistic categories (i.e, *no damage*, *low damage*, *medium damage*, *high damage*) and triggering alerts accordingly, the reasoning layer supports maintenance decision-making in a transparent and interpretable manner.

Together, these layers form an intelligent, multi stage SHM framework that enables continuous assessment of bridge health. The integration of vibration based sensing with machine learning and reasoning logic not only improves detection performance but also reduces reliance on manual inspections, optimizes maintenance planning, and enhances public safety.

1.3.1 Why a Multi-Agent System for SHM?

While the thesis presents extensive technical details regarding Structural Health Monitoring(SHM) of railway steel bridges using vibration based sensory data, it is essential to clearly articulate the underlying motivation for adopting a Multi Agent System(MAS) architecture over existing SHM methods.

Conventional SHM frameworks, particularly those built around centralized machine learning pipelines; typically involve a monolithic model or single stage learning system responsible for a wide range of tasks, including signal preprocessing, damage localization, severity classification, and alert generation. Although such models can be highly accurate, they suffer from several key limitations:

- **Lack of modularity:** Centralized systems lack modular separation of tasks, making debugging, interpretation, and retraining difficult. Each update or failure in a component often requires system wide retraining.
- **Scalability challenges:** As railway networks expand or new sensing modalities (e.g., higher frequency accelerometers, strain gauges) are introduced, centralized models become computationally heavy and less adaptable.
- **Low interpretability:** Deep models such as CNNs and DNNs often function as black boxes, producing accurate but opaque predictions. This hinders real world deployment, where maintenance engineers need actionable and understandable insights.
- **Reduced fault tolerance:** A failure in a centralized model; whether due to sensor noise, missing data, or misclassifications; can cause the entire SHM pipeline to collapse, jeopardizing safety.

To address these gaps, this thesis introduces a **Multi Agent System (MAS)** for SHM, where each agent is a specialized, loosely coupled module dedicated to a specific task, such as:

- **Signal segmentation and Features engineering**
- **Damage intensity prediction (e.g., using EBMs, RBFNet)**
- **Bridge scenario classification (e.g., via CNN/DNN)**
- **Reasoning and alert generation (via fuzzy logic)**

The MAS approach offers several benefits over traditional SHM pipelines:

1. **Decentralization and flexibility:** Each agent can be retrained, optimized, or replaced independently without affecting the rest of the pipeline.
2. **Real time adaptability:** Agents can operate in parallel or in sequence, allowing the system to adapt to real time monitoring demands.
3. **Interpretability and transparency of decisions:** Integration of explainable models (Explainable Boosting Machines (EBMs)), interpretable reasoning (fuzzy logic) and intermediate decision nodes allows actionable maintenance decisions.
4. **Scalability for large infrastructure:** MAS allows distributed deployment across edge devices and cloud environments, making it suitable for nationwide bridge monitoring systems.

This research demonstrates how the MAS paradigm transforms SHM from a passive monitoring tool into an intelligent, modular and interpretable decision support system tailored for real world infrastructure management. This thesis is also motivated by the need to develop and validate such a layered SHM system tailored for railway steel bridges, with the goal of advancing infrastructure resilience through data-driven and explainable monitoring approaches.

1.4 Thesis Contributions

This thesis contributes to the advancement of structural health monitoring (SHM) techniques for railway steel bridges by integrating physics-based modeling, time frequency signal analysis, deep learning architectures, and interpretable reasoning systems. The core motivation behind this work is to improve the reliability, interpretability, and real-time applicability of damage detection systems for critical infrastructure.

The research presented in this thesis is structured into five main parts:

- In **Chapter 4**, we designed and evaluated a Radial Basis Function Radial Basis Function (RBF) based neural network for damage detection in railway steel bridges. The model was trained on statistical features derived from accelerometer signals and was assessed on two main tasks: bridge scenario classification and damage intensity estimation. The chapter highlights the effectiveness of RBFs in modeling non-linear feature spaces with interpretable localized responses.
- In **Chapter 5**, we explored Explainable Boosting Machines (EBMs) for interpretable classification of structural conditions. EBMs were trained on the same statistical feature set and provided insight into feature contribu-

tions through global importance plots. A stacked EBM architecture was also proposed, where the secondary model used the primary EBM's predictions along with original features to refine classification outputs. This approach improved performance in complex tasks like stress force prediction and supported transparent reasoning in SHM systems.

- In **Chapter 6**, we investigated the application of classical machine learning algorithms and a Convolutional Neural Network (CNN) for bridge damage detection. Using structured time-series representations derived from Hankel matrices, we applied various data slicing techniques and data augmentation strategies to improve feature diversity. The chapter highlights how these approaches affect model performance in detecting damage severity from sensor data.
- In **Chapter 7**, we focused on converting accelerometer signals into the frequency domain using the Short-Time Fourier Transform STFT. This enabled the construction of spectrograms that encode both the temporal and spectral properties of the bridge vibrations. We then implemented a deep learning fusion model that combines Deep Neural Network (DNN)s and Transformer blocks to detect and classify damage levels. Data augmentation techniques and Hankel matrix-based time series slicing were also employed to enhance feature diversity and capture temporal dependencies.
- In **Chapter 8**, we developed a fuzzy rule-based reasoning layer that receives features from the perception module and translates them into interpretable alerts. The system uses expert-defined fuzzy rules to assign damage severity levels, thereby enabling human-understandable and prioritized decision-making for bridge maintenance and safety assessment.

1.4.1 Key Contributions

The following key contributions are made by this thesis;

- A physics-informed, semi-analytical modeling framework using FEM to simulate bridge damage scenarios and damage intensities for data generation and validation.
- A comprehensive evaluation of classical machine learning models-including K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), Random Forest (RF), and ensemble techniques-applied to statistical features extracted from sensor data for structural damage classification.
- Implementation of a basic Convolutional Neural Network (CNN) model as a deep learning baseline for learning patterns from raw or preprocessed

structural vibration data.

- A time-frequency signal processing pipeline using Short-Time Fourier Transform (STFT) and spectrograms to extract time-localized spectral features from accelerometer signals.
- Development of fusion-based deep learning models that integrate DNN-extracted visual features with Transformer-based sequence representations to enable robust and temporally sensitive bridge damage detection.
- Application of Hankel matrix representations and data augmentation strategies to enhance the modeling of temporal dependencies and increase training data diversity.
- Design and implementation of a fuzzy rule-based reasoning layer to translate model outputs into interpretable graded alerts for various levels of structural damage.
- Validation of all proposed models through extensive simulations and analyses using synthetic FEM-based data and real world sensor recordings relevant to railway steel bridges.

These contributions collectively support the development of practical, interpretable and scalable SHM systems tailored to the needs of critical infrastructure such as railway bridges.

1.5 Thesis Organization

The structure of this thesis and the contributions made in each chapter are organized around the key methodologies and frameworks developed for SHM of railway steel bridges. These include the use of vibration-based sensory data, advanced feature extraction in both time and frequency domains, the application of classical and deep learning models for damage detection, and the integration of a fuzzy reasoning layer for interpretable decision-making. The thesis adopts a multi layered approach; comprising the data, perception, and reasoning layers to enable robust and intelligent monitoring of bridge health conditions. The overall organization of the thesis is illustrated in

- The present chapter, **Chapter 1**, provides an overview of the thesis structure, highlighting introduction of railway network in EU, specifically Italy, the motivation and overall perspective of doing SHM of Railway Steel Bridge project for my thesis.
- **Chapter 2** provides the necessary background to support the research presented in this thesis. It introduces key concepts related to SHM. The chapter further explores studies related traditional machine learning tech-

niques for anomaly detection, damage classification, and alert generation. Finally, it introduces Railway Bridge Health Monitoring , with a focus on their potential solution and challenges in anomaly detection and classification. This background study serves as a foundation for understanding the subsequent researches conducted in this thesis.

- **Chapter 3** presents an empirical study on Structural Health Monitoring for Railway Steel Bridges in Time Domain. We will also discuss **Data Acquisition** process using accelerometer sensors. initial **Data Pre-processing**, **Feature Extraction** and **Feature Analysis**. This chapter is based on data acquired from the Finite Element Model FEM of *Umbria-Sansepolcro* sectionś railway steel bridge located at the ***Cittá Di Castello*** over *Fiume Tevere* at this Location on Google Maps .
- **Chapter 4** builds upon the findings of **RBF** based model for Structural Health Monitoring SHM of Railway Steel Bridges. The current Radial Basis Function implementation focuses on leveraging accelerometer-derived statistical features to enhance the reliability and sensitivity of damage detection for Bridge Scenarios and Damage Intensities. Part of this chapter is based on:

  Asad, Muhammad. "Machine Learning Based Ambient Analysis of Railway Steel Bridges for Damage Detection." International Symposium on Ambient Intelligence. Cham: Springer Nature Switzerland, 2023.

  Muhammad Asad, Giovanni De Gasperis, and Stefania Costantini. RBF based NN architecture for structural health analysis of railway steel bridges. In AIxIA Doctoral Consortium 2023 @ 22nd International Con- ference of the Italian Association for Artificial Intelligence. Springer Na- ture, Rome, Italy, 2024.

- **Chapter 5** investigates the effectiveness of **Explainable Boosting Machine (EBM)**, a class of interpretable machine learning models that combine the accuracy of ensemble methods with the transparency of generalized additive models. The focus is on evaluating the performance of EBMs in the context of structural health monitoring of railway steel bridges, using statistical features derived from sensor data.
- **Chapter 6** investigates the impact of various classical machine learning algorithms and **CNN** on structural damage detection. It explores different data slicing strategies derived from the Hankel matrix to preserve the temporal structure of sensor signals and incorporates data augmentation techniques to enhance feature diversity and improve model robustness.Part

of this chapter is based on:

Asad, M., Alaggio, R., Cirella, R., Costantini, S., & Gasperis, G. D.. Vibration-Based Machine Learning Model Training for Railway Bridges Health Monitoring [2025]. IEEE Sensors. Submitted, (Under Review)

- **Chapter 7** examines anomaly detection in the context of railway steel bridge monitoring, focusing on the analysis of damage scenarios and intensities in the frequency domain. **STFT** is employed to generate spectrograms from raw accelerometer signals, capturing time–frequency characteristics of structural vibrations. These spectrograms are then processed using a deep learning framework that fuses **DNN** including ResNet, EfficientNet, and DenseNet with Transformer blocks to detect and classify varying levels of structural damage.Part of this chapter is based on:

  Asad, M., Alaggio, R., Cirella, R., Costantini, S., & Gasperis, G. D. ; Semi-Supervised Anomalies Detection and Localization For Railways Steel Bridges Using Parallel Dual Level Class Classification Approach [2025]. ScienceDirect Journal: Computers & Structures. (Under Submission)
- **Chapter 8** presents the implementation of a reasoning layer using a fuzzy rule-based system for generating structural damage alerts. The outputs from the perception layer-comprising statistical or deep feature representations; are used as inputs to the reasoning layer. A fuzzy inference system is constructed using a set of predefined rules that evaluate damage severity. Based on the degree of detected damage, the system generates appropriate alerts, enabling interpretable and graded decision-making under uncertainty.
- **Chapter 9** presents the conclusions of the thesis, summarizing the key findings, acknowledging the limitations of the current work, and outlining potential directions for future research aimed at advancing the field. A significant contribution was the implementation of frequency-domain analysis using STFT to generate spectrograms, enabling effective damage detection through deep learning models. The fusion of DNN with Transformer architectures demonstrated improved sensitivity in identifying damage patterns from spectrogram-based features. Additionally, the thesis explored the use of Explainable Boosting Machines EBM and statistical feature sets for interpretable damage classification. A fuzzy rule-based reasoning layer was developed to translate model outputs into human-understandable alerts, providing graded damage assessments based on predefined rules. While the models performed well in controlled settings, limitations include dependency on data quality, generalizability to unseen bridge types, and the

need for real-time integration.

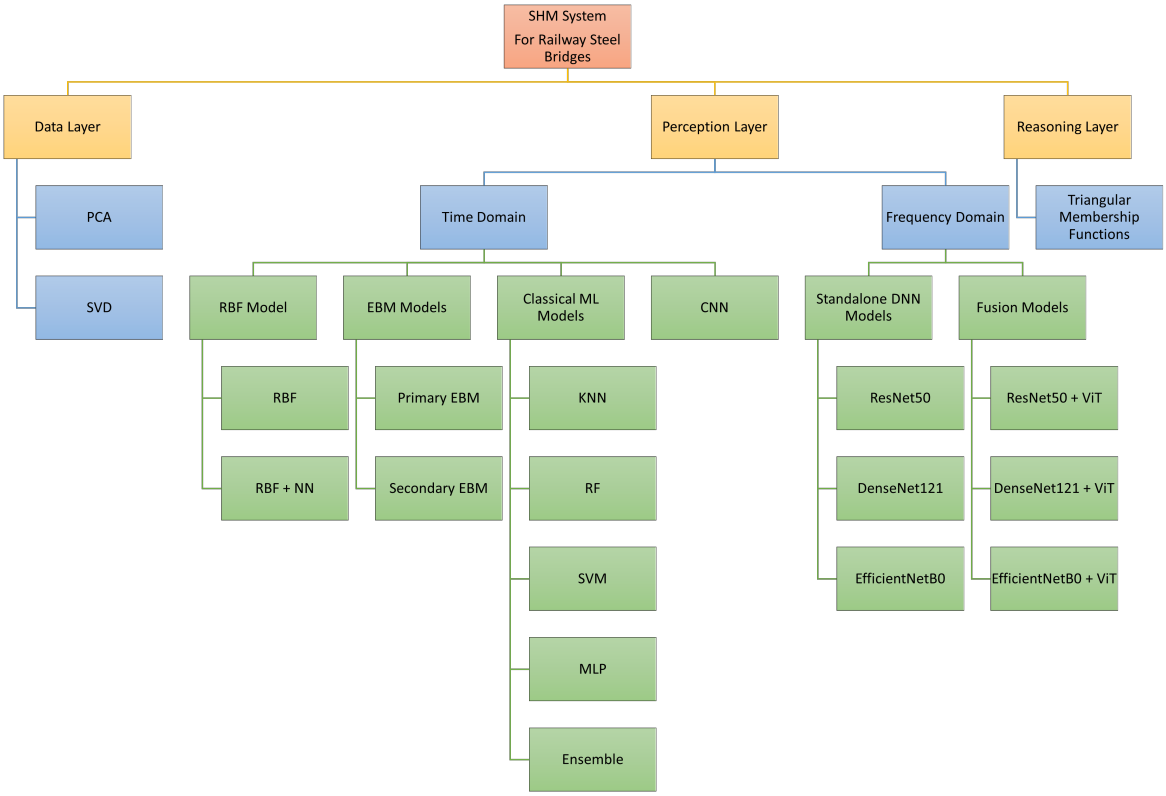


Figure 1.3: Thesis Technical Structure

Chapter 2

Background

"First, solve the problem. Then, write the code"

— John Johnson.

Structural Health Monitoring(SHM) of railway steel bridges is becoming increasingly important in the context of intelligent infrastructure systems, particularly with the rise of (Internet of Things (IoT)) technologies and smart sensing platforms. Modern SHM systems enable continuous monitoring of bridge conditions through a network of wireless sensors, particularly those capable of capturing dynamic responses such as vibration and strain. This chapter provides a comprehensive analysis of the current state of SHM for railway steel bridges, highlighting key technologies, signal processing techniques, and recent research developments. The role of SHM in ensuring operational safety, reducing maintenance costs, and extending structural lifespan is emphasized. Furthermore, the chapter explores the data acquisition and communication frameworks required for efficient damage detection, with a focus on how multi-layered SHM architectures, comprising data, perception, and reasoning layers, can be integrated into real-time monitoring systems. Challenges such as sensor deployment, data reliability, and interpretability are discussed, along with potential future directions including the use of AI-driven models, edge computing, and integrated decision-support systems.

2.1 Introduction

Structural Health Monitoring SHM refers to the process of implementing a damage detection and assessment strategy for engineering structures. In the context of railway steel bridges, SHM involves the continuous or periodic collection of data; typically through sensors such as accelerometers, strain gauges, and displacement transducers; to evaluate the structural integrity and detect anomalies that may indicate damage or deterioration. SHM systems enable the early identification of faults, reduce the reliance on manual inspections, and support data driven maintenance planning. By providing real-time insights into the bridge's condition under operational loading; such as the passage of trains, SHM and Anomaly detections enhances safety, extends service life, and contributes to the efficient management of critical infrastructure.

Structural Health Monitoring (SHM) and Anomaly Detection is a multidisciplinary field that involves the integration of sensing technologies, signal processing, data analytics, and decision making algorithms to assess the condition of civil infrastructure in real-time or at regular intervals. In the context of railway steel bridges, SHM aims to monitor structural responses under operational loads; particularly dynamic loads induced by train passages, to detect, localize, and quantify damage that may compromise safety and serviceability.

Technically, SHM systems for railway bridges typically consist of three main components: **(1)** a sensor network for data acquisition, **(2)** a data management and processing framework, and **(3)** a diagnostic and prognostic module for interpreting the structural state. Commonly used sensors include accelerometers, strain gauges, displacement transducers, and temperature sensors, strategically placed to capture dynamic responses such as vibration modes, deflections, and thermal effects. Vibration-based monitoring, in particular, is widely used due to its sensitivity to changes in modal properties; such as natural frequencies, mode shapes, and damping ratios, which are indicative of structural degradation [7, 8].

Modern SHM frameworks also incorporate signal processing techniques (for example, Fast Fourier Transform (FFT), Short-Time Fourier Transform(STFT), Wavelet Transforms (WT) and machine learning algorithms (for example, SVMs, Random Forests, CNNs, Transformer networks) for automated damage detection and classification. These methods are capable of handling large volumes of time-series data and extracting features that are sensitive to various damage types, including fatigue cracks, joint loosening, and corrosion induced stiffness loss [9, 10].

Furthermore, the deployment of long term SHM systems allows for tracking

damage progression over time, supporting predictive maintenance strategies that aim to prevent failures before they occur. The integration of reasoning layers, such as fuzzy logic systems, enables interpretability by translating sensor outputs and classifier predictions into human-readable alerts (for example, "moderate damage detected"), which are critical for decision making by infrastructure managers [11].

However, SHM provides a robust and scalable approach for the continuous assessment of railway bridge integrity, enabling early damage detection, reducing inspection costs, and extending structural life cycles; all while improving the overall safety and resilience of transportation networks.

2.2 SHM in Railway Steel Bridges

Compared to other conventional monitoring systems, SHM systems for **railway steel bridges** offer certain unique features due to the *dynamic loading conditions*, strategic importance, and sensor-enabled infrastructure. These characteristics bring both advantages and challenges. The following outlines the key benefits of SHM in the context of railway steel bridges.

2.2.1 Key Advantages

- **Predictable loading patterns:** Railway bridges are subjected to repetitive, predictable loads induced by train passages, enabling better modeling of structural response and damage trend analysis [7, 12].
- **Sufficient power availability:** Many SHM systems leverage existing railway infrastructure or use vibration-based energy harvesting to reduce reliance on batteries [13].
- **High-frequency sensing and processing:** The availability of high resolution sensors like accelerometers and strain gauges allows the extraction of modal features critical for damage detection [8, 14].

2.2.2 Challenges in SHM of Railway Steel Bridges

- **Dynamic environment:** Variability due to train-induced loads, environmental conditions, and operational noise complicates damage identification [10, 15].
- **Large-scale and distributed infrastructure:** Monitoring multiple bridges across large railway networks presents logistical and computational challenges [16, 17].
- **High-volume data exchange:** Sensor networks generate vast quantities

of time series data, requiring efficient transmission, storage, and processing solutions [13, 18].

2.2.3 Applications and Services

Structural Health Monitoring(SHM) systems for railway bridges support a range of essential applications. It provides essential functions such as early damage detection, post-event assessment, maintenance optimization, and long-term lifecycle management [9, 14].

Early Damage Detection: Modern SHM systems use vibration-based sensors and AI techniques to detect structural anomalies such as cracks or loosening at early stages [19]. SHM systems enable the continuous monitoring of structural responses, such as vibration, strain, and displacement; allowing for the early identification of anomalies that may indicate the presence of damage. Detecting issues such as fatigue cracks, corrosion, or joint loosening at an early stage allows for timely interventions, reducing the risk of sudden failures and avoiding costly service disruptions.

Post-Event Assessment: Following events such as earthquakes, heavy train loading, derailments, or accidental impacts, SHM systems provide vital data to assess whether structural integrity has been compromised. By analyzing post-event sensor data, engineers can make informed decisions about whether a bridge can remain in service, requires immediate repair, or should be temporarily closed for further inspection. SHM systems integrated with digital twin architectures enable post-event evaluation of structural integrity following train overloads, impacts, or earthquakes [20].

Maintenance Optimization: Traditional bridge maintenance relies on periodic manual inspections, which may miss time sensitive deterioration. Condition-based SHM strategies, supported by continuous data collection, enable smarter maintenance scheduling, reducing unnecessary inspections and improving cost efficiency [21]. This improves maintenance efficiency, reduces inspection costs, and ensures that resources are directed where they are most needed.

Lifecycle Management: Long term SHM data helps model degradation trends and supports bridge retrofitting or replacement decisions, ensuring long-term safety and functionality [22]. It also plays a critical role in extending the service life of bridges while maintaining safety and regulatory compliance.

Risk Mitigation and Safety Assurance: Real-time SHM data contribute to safety assurance and integration into Intelligent Transportation Systems (ITS) for dynamic speed control, bridge load restrictions, and predictive rout-

ing [23,24]. Continuous monitoring helps railway operators manage operational risk more effectively by providing real-time alerts and status reports on bridge conditions. This ensures the safety of passengers, rolling stock and maintenance personnel, especially in high-speed or high-traffic corridors.

2.3 Summary

Structural Health Monitoring (SHM) of railway steel bridges plays a vital role in ensuring the long-term safety, durability, and operational continuity of critical infrastructure. One of the core capabilities of SHM systems is their ability to perform anomaly detection; the identification of structural behavior that deviates from expected or baseline conditions. These anomalies may signal the early onset of damage such as fatigue cracks, corrosion, joint degradation, or unexpected shifts in structural response.

The dynamic and complex nature of railway bridge environments presents a unique challenge for anomaly detection. Bridges are subjected to varying train loads, temperature fluctuations, wind effects, and aging-related material changes. As a result, the signals collected from vibration-based sensors are often noisy, nonstationary, and influenced by operational and environmental variability. Accurate anomaly detection therefore requires the integration of robust feature extraction, adaptive filtering, and intelligent classification techniques.

To address these challenges, SHM systems must be carefully designed across three key components: Sensor Deployment and Data Acquisition, Feature-Based Anomaly Detection, and Decision-Making & Interpretation. An effective SHM system that incorporates anomaly detection enables proactive maintenance, reduces the risk of unexpected failures, and supports data-driven infrastructure management. As railway bridges continue to age under increasing traffic demands, the development of intelligent and scalable anomaly detection methods within SHM frameworks is essential to preserve both safety and serviceability.

Chapter 3

Structural Health Monitoring and Anomaly Detection in Time Domain

"What we want is a machine that can learn from experience."

— Alan Turing

3.1 Introduction

Structural Health Monitoring (SHM) is an essential tool for ensuring the safety, reliability, and performance of civil infrastructure, particularly railway steel bridges that are subjected to repetitive dynamic loads and environmental stresses. SHM enables continuous observation of structural conditions through

sensor based systems, facilitating the early detection of damage and informing maintenance decisions.

Structural Health Monitoring SHM in the time domain involves analyzing raw or preprocessed structural response signals, such as acceleration, strain, or displacement, as a function of time to detect and evaluate damage in civil infrastructure. In the context of railway steel bridges, time-domain SHM is especially valuable, as it captures the bridge's dynamic response to operational loads, such as those induced by passing trains, in real-time [7].

Structural health monitoring (SHM) of railway bridges has become an increasingly important focus. What prompted the research to explore this topic more thoroughly is the recognition that structures, during their life, are subject to aging or degradation phenomena. These phenomena may result in a loss of structural performance, ultimately leading to the inadequacy of the structure to fulfill its intended functional requirements. The strategic importance of bridges as critical elements for mobility and logistical infrastructure makes their monitoring even more essential than ordinary structures. Any potential compromise of their integrity could lead to significant disruptions in transportation systems and severe economic and social impacts. The assumption at the base of damage detection is that when any form of damage or degradation occurs, the structural parameters change, and consequently, the response of the structure is altered as well: the objective of monitoring a structure is, therefore, to detect this behavioral change and to investigate its causes.

SHM involves continuously evaluating the bridge's condition, assessing damage progression, and timely maintenance interventions. In this context, a structural monitoring system can be helpful in two different phases of a structure's life: during maintenance, to focus interventions in parts of the structure where anomalies emerge, and during emergencies, as a management tool for selecting intervention priorities.

Damage detection techniques can be categorized into four stages, as outlined by Rytter [25] and developed over time: identification, localization, quantification of damage intensity, and prognosis. In line with these levels, many techniques have been developed over time. Recently, the development of artificial intelligence techniques has driven the research community's interest in applying machine learning algorithms to solve damage identification and classification problems. Most use non-destructive methods that involve sensor data acquisition (accelerometers, crack gauges, etc.) placed at significant locations in the structure [26].

In this chapter, we focus on time-domain approaches for SHM, where raw sensor signals; typically acceleration; are analyzed over time to extract features

that are sensitive to structural changes. These features are then used to detect anomalies, which may signal the presence of damage such as fatigue cracks, loosened joints, or material degradation.

3.2 Overview of Time-Domain SHM

Time domain methods utilize direct measurements of a structure's response over time. These measurements reflect how the structure behaves under real operational conditions, offering crucial insights into the initiation and progression of damage. Unlike frequency-domain methods, which require transformation of the signal, time-domain approaches retain the raw temporal characteristics of the structural response [14].

3.2.1 Time-Domain Signal Analysis

Time-domain SHM relies on the direct analysis of signals as a function of time. It is particularly effective for real-time monitoring due to its computational simplicity and the ability to capture transient events.

3.3 Data Description and Acquisition

The Finite Element Model (FEM) is designed in lab for initial data acquisition. The model is based on the geometry and mechanical characteristics of an existing steel railway bridge on **Umbria-Sansepolcro** section as seen in Fig. 3.1 near *Città di Castello, Italy.*

The structure is made up of three steel reticular beams, which allow the railway network to cross the underlying Tiber River as in Fig 3.2; plane and elevation views of the central span can be seen in Fig. 3.3 and Fig. 3.4. The highlighted colors represent the group of accelerometer sensors installed at each of seven bridge scenarios, i.e. S1, S2, S3,..., S7.

Recently, a vibration monitoring system was installed on the central span of the structure. The system consists of six accelerometer sensors, mostly uniaxial in the vertical direction. In the center of the span, there are biaxial sensors, the additional component oriented in the transversal direction of the bridge. Therefore, there are a total of eight measurement channels.

The accelerometers are very low noise, high dynamic Force-Balance type, and are connected via cables in two synchronized chains: signals are collected with a sampling frequency of 200 Hz. A Global Positioning System (GPS) receiver on each data recorder allows the data coming from the two chains to be synchronized in time. The choice of very low-noise sensors allows the system

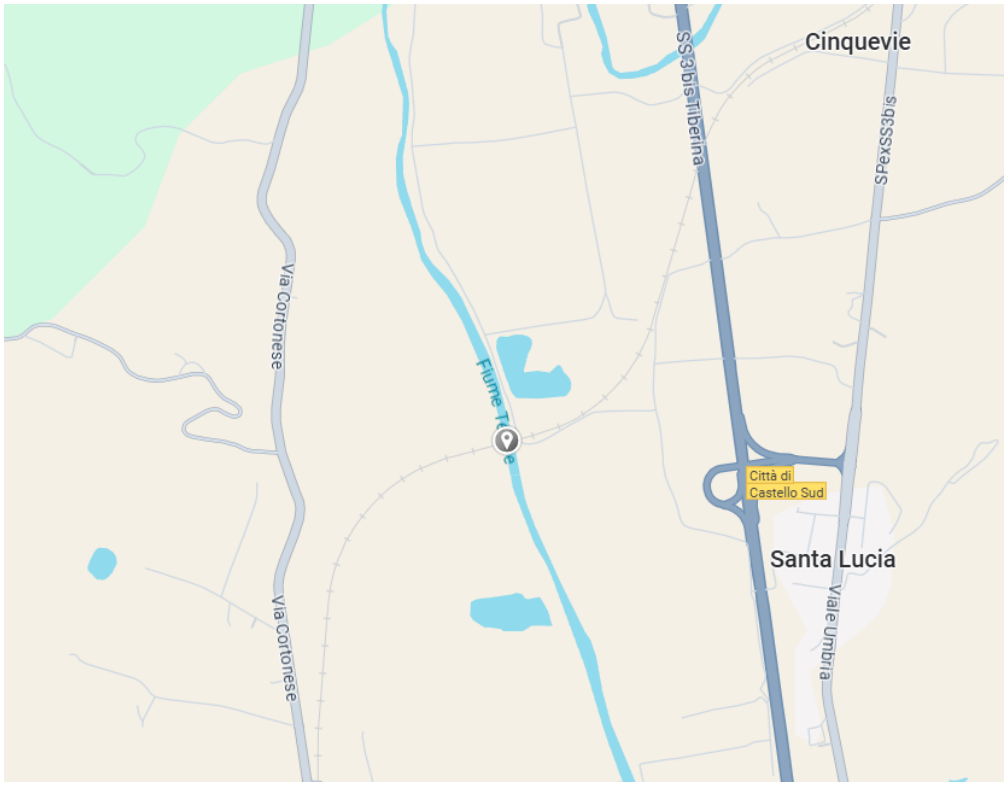


Figure 3.1: Location of Umbria-Sansepolcro Railway Steel Bridge

(a) Plane View (b) Elevation View

Figure 3.2: Railway Steel Bridge at Umbria-Sansepolcro Section [2]

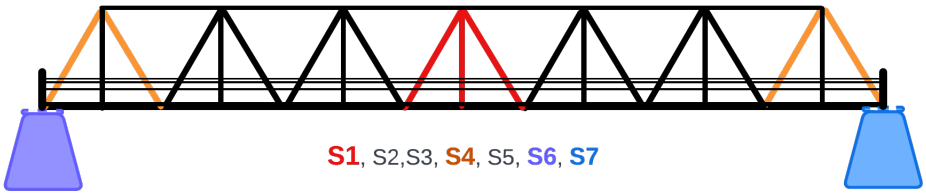


Figure 3.3: Side view of bridgeś Central Part. Colors refer to different damage scenarios.

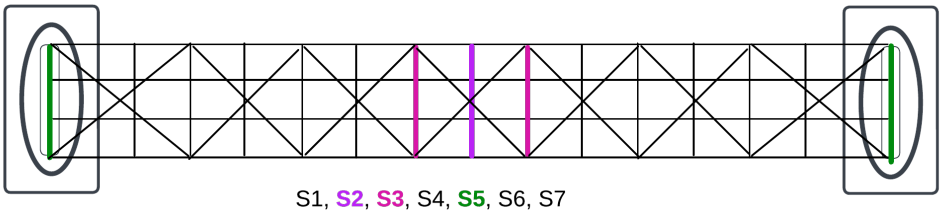


Figure 3.4: Aerial view of bridgeś Central Part. Colors refer to different damage scenarios.

to record even slight vibrations, such as those generated by the surrounding environment of the bridge. The signals are recorded under ambient noise conditions (train passages are not collected for this study).

Starting from the acquired ambient vibration series and considering the geometry of the bridge, the model was developed to replicate the actual structure's dynamic behavior optimally. For this purpose, the model's parameters were optimized through a parametric calibration process, whose detailed description is beyond the scope of this study. The model was then used to simulate the structural response in different damage scenarios.

The damage is generated by selecting specific elements of the FE model, whose positions are predetermined and labeled by alphanumeric symbols (from S1 to S7). It consists of the reduction of the mechanical property of the elastic modulus in proportion to the chosen damage intensity level. Damage intensity levels are represented by a percentage value ranging from 1% to 90%: a more detailed description of the scenarios is provided in Table 3.1.

Therefore, the dataset used for training different ML models is built starting from the time series generated by the FE model for each damage scenario and intensity. The time series are recorded at the nodes of the FE model in the same position as the sensors placed on the real structure. Each simulation produces 8 time series lasting 30 minutes, sampled with a frequency of 200Hz, for a total of 360000 samples per series.
Furthermore, for each damage scenario and intensity, the simulation is repeated 180 times, varying the excitation applied to the model.

Table 3.1: Damage Scenarios Description. Location of Damaged Elements as shown in Fig. 3.3-3.4.

Scenarios	Description
S1	Braces - degradation due to bending mechanisms
S2	Transverse - degradation due to bending mechanisms (single transverse at midspan)
S3	Transverse - degradation due to bending mechanisms (3 transverses at midspan)
S4	Braces - degradation due to shear mechanisms
S5	Transverse - degradation due to shear mechanisms (transverses at the header)
S6	Left pier - degradation due to vertical failure
S7	Piers - degradation due to vertical failure of both piers

It is important to note that, when dealing with complex structures such as those in civil engineering, the damage classification phase (i.e., the Rytter scale levels above the first) necessarily requires the development of a mathematical model of the structure itself. This model is essential for estimating structural behavior in potential damage scenarios.

3.3.1 Hankel Matrix Construction and PCA-Based Feature Extraction

A key component of the proposed methodology is the transformation of 1D vibration signals into structured matrix representations that preserve the temporal dynamics of the signal. This is achieved using the **Hankel matrix**, a mathematical formalism particularly effective for analyzing non-stationary structural vibration data.

3.3.1.1 Hankel Matrix: Structure and Purpose

Given a time series signal $x(t)$ of length T, a Hankel matrix H of embedding dimension w is constructed such that every ascending skew diagonal (top-right direction) contains identical values. Formally:

$$H = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_k \\ x_2 & x_3 & x_4 & \cdots & x_{k+1} \\ x_3 & x_4 & x_5 & \cdots & x_{k+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_w & x_{w+1} & x_{w+2} & \cdots & x_T \end{bmatrix} \tag{3.1}$$

where w is the number of rows, and $k = T - w + 1$ is the number of columns.

3.3.1.2 Why Hankel Matrices?

The Hankel matrix converts a 1D signal into a 2D representation that captures:

- **Temporal correlations:** consecutive signal segments appear as adjacent rows,
- **Local vibration patterns:** repetitive behavior is reinforced along diagonals,
- **Structural changes:** small deviations in dynamics alter the geometry of the matrix.

This representation has been widely used in modal analysis, subspace identification, and anomaly detection because it improves the separation of signal and noise components.

sectionDimensionality Reduction Using PCA

Once the Hankel matrix H is constructed, its dimensionality may be high, especially for long signals. To extract the most informative patterns, **Principal Component Analysis (PCA)** is applied.

First, the covariance matrix of H is computed as:

$$C = \frac{1}{n-1} H^\top H \tag{3.2}$$

PCA solves the eigenvalue problem:

$$C\mathbf{v}_i = \lambda_i \mathbf{v}_i \tag{3.3}$$

where λ_i and $\mathbf{v}_i$ denote eigenvalues and eigenvectors, respectively. The eigenvectors associated with the top-k eigenvalues correspond to the directions of maximum variance and are used to project the Hankel matrix into a reduced subspace.

3.3.1.3 Advantages of PCA in SHM

- **Noise reduction:** low-energy components (often noise) are discarded,
- **Computational efficiency:** reduces data dimension for subsequent ML models,
- **Enhanced discriminability:** structural changes affect major principal components,
- **Improved stability:** PCA regularizes high-dimensional Hankel matrices.

3.3.2 Illustrative Example

Consider a vibration signal of length 256. Using a window size of $w = 64$ produces a 64×193 Hankel matrix. PCA is then applied to this matrix, and the top principal components are extracted, providing a compact representation of the signal dynamics.

This pipeline is essential for converting raw accelerometer signals into robust features prior to generating spectrograms or feeding them into learning models.

This expanded explanation clarifies the mathematical role and practical utility of the Hankel-PCA pipeline within the broader SHM methodology. It provides readers unfamiliar with the formalism with a clear intuition about why these transformations are essential for effective feature extraction from vibration-based bridge monitoring data.

The proposed approach uses the physics based structure Reduced-Order Model (ROM) to generate a dataset of expected sensor measurements corresponding to various asset states with damage at different locations and intensities. This dataset is then used to train a classifier capable of processing near-real-time sensor data and estimating the underlying state of the asset.

The methodology involves an initial Data Dimensionality Reduction (DDR) step, in which accelerometric time series data are processed to form block Toeplitz correlation matrices. These matrices are derived from the block Hankel matrix constructed from output data, commonly referred to as the subspace matrix in the Data-Driven Stochastic Subspace Identification (DD-SSI) technique [27]. The DD-SSI method is widely adopted in civil engineering applications for Operational Modal Analysis (OMA), as it enables the extraction of dynamic structural properties under ambient excitation conditions. Once the subspace matrix is formed, a set of block Toeplitz matrices is computed, encapsulating the temporal correlations in the vibration data. These correlation matrices are then employed as input features to neural networks, which are trained to perform damage classification across different bridge scenarios and damage intensity levels.

Figure 3.6 illustrates the simulation-based workflow used for generating FEM simulation datasets for SHM) of railway steel bridges. The process begins with a FEM of the bridge structure, which is used to simulate various damage scenarios and intensities. These scenarios represent real-world degradation mechanisms, such as vibrational damages. Once the structural parameters are defined; including material properties, boundary conditions, and damping (uniform or Rayleigh), the frequency response function matrix H is calculated.

Using this matrix, simulated time series data is generated by applying ran-

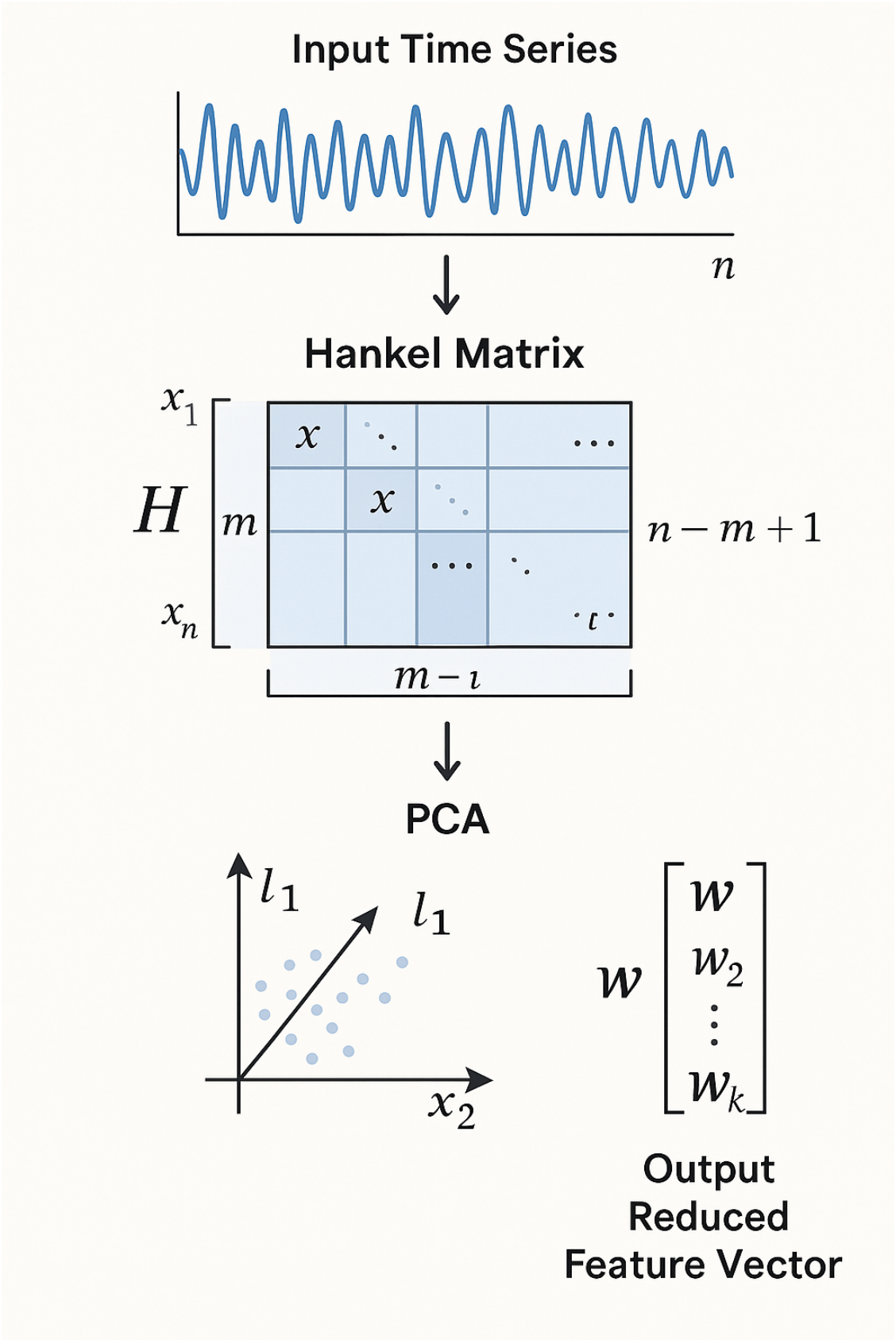


Figure 3.5: Pedagogical illustration of Hankel matrix construction and PCA-based feature extraction from a vibration time-series signal.

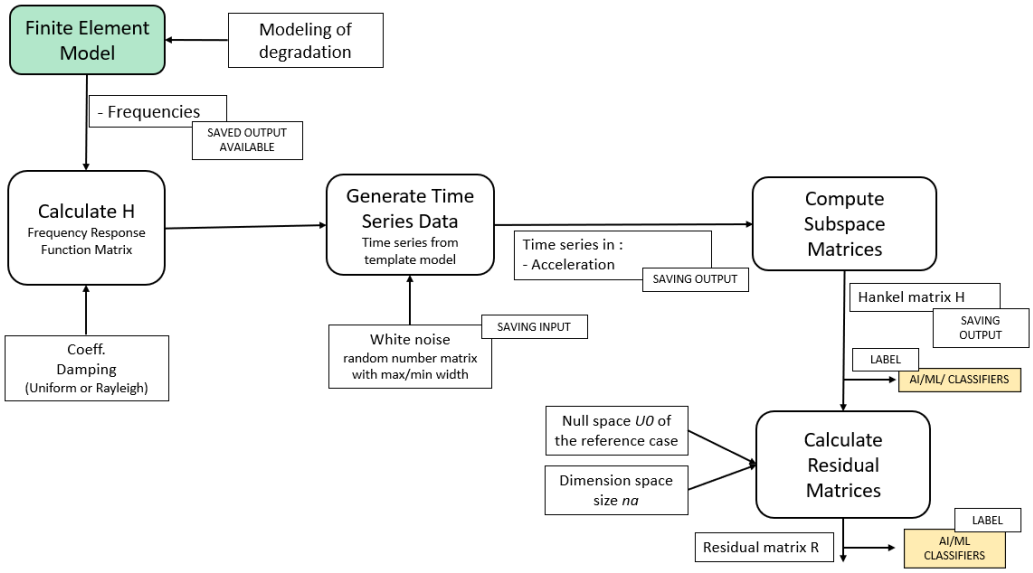


Figure 3.6: FEM-based simulation pipeline for (T_b) Hankel Matrix generation and classification

dom white noise excitations to FEM. This simulates the acceleration response of the structure under different operational and damage conditions. The generated signals are then used to compute subspace matrices, including the Hankel matrix H, which captures the temporal dynamics of the structural response.

Both the Hankel and Residual matrices are labeled according to their corresponding damage scenarios and are fed into machine learning classifiers. These classifiers are trained to identify both the location and intensity of damage based on the extracted features. This simulation framework provides a controlled and repeatable way to generate labeled data for anomaly detection and damage classification tasks in SHM.

The classification procedure of structural damage scenarios is based on the extraction of features which contain within them the information necessary to distinguish the dynamics of each SIMULATED state. The quantities of interest are extracted directly from the acquired (or simulated) accelerometric series, with a view to a non-parametric approach (not based on the extraction of modal parameters). In the extraction process, the available accelerometric series are initially reorganized into block matrices **H**, called **Hankel matrices** due to their symmetric form: the rows and columns of these matrices $\boldsymbol{H}$ are obtained by imposing a TIME DELAY i on the data matrix.

The covariance matrix of the outputs T_b, also called *Subspace Matrix* or *Toeplitz matrix* due to its shape, is obtained by calculating the covariance between H_p and H_f, determined by dividing the Hankel matrix into two

halves [28]. Mathematically;

$$H_{0:2i-1} = \left(\begin{array}{cccc} y_0 & y_1 & \cdots & y_{j-1} \\ y_1 & y_2 & \cdots & y_j \\ \vdots & \vdots & \ddots & \vdots \\ y_{i-1} & y_i & \cdots & y_{i+j-2} \\ \hline y_i & y_{i+1} & \cdots & y_{i+j-1} \\ y_{i+1} & y_{i+2} & \cdots & y_{i+j} \\ \vdots & \vdots & \ddots & \vdots \\ y_{2i-1} & y_{2i} & \cdots & y_{2i+j-2} \end{array}\right) = \begin{pmatrix} H_{0:i-1} \\ H_{i:2i-1} \end{pmatrix} = \begin{pmatrix} H_p \\ H_f \end{pmatrix} = \begin{bmatrix} y_t \\ y_{t+1} \\ \vdots \\ y_{t+2i-1} \end{bmatrix} \tag{3.4}$$

As shown in Equation 3.5, the Hankel matrix can be represented compactly as a sequence of stacked column vectors.

$$H_{0:2i-1} = [y_t,\ y_{t+1},\ \ldots,\ y_{t+j-1}] \in \mathbb{R}^{2i\times j} \tag{3.5}$$

As shown in Equation 3.6, the matrix T_b is constructed by vertically stacking the matrices H_p and H_f, where $H_p \in \mathbb{R}^{i\times j}$ contains the first i rows of the Hankel matrix $H_{0:2i-1}$, and $H_f \in \mathbb{R}^{i\times j}$ contains the remaining i rows. The expression

$$T_b = \begin{bmatrix} H_p^\top & H_f^\top \end{bmatrix}^\top = \begin{bmatrix} H_p \\ H_f \end{bmatrix} = [y_t\ y_{t+1}\ \ldots\ y_{t+j-1}] \in \mathbb{R}^{2i\times j} \tag{3.6}$$

demonstrates that T_b is equivalent to a matrix formed by horizontally stacking j column vectors $y_k \in \mathbb{R}^{2i\times 1}$, starting from time step t up to $t+j-1$. This formulation is particularly useful for time-domain modeling and feature extraction, where each column y_k captures a temporal window of vibration response data used in downstream learning or reasoning tasks. As defined in Equation 3.6, the matrix T_b is formed by stacking the j column vectors side by side.

As shown in Equation (3.7), the residual matrix R is calculated by projecting the actual data matrix $T_{b,\text{act}}$ onto the transpose of the reference subspace $U_{0,\text{ref}}$. This formulation highlights the alignment of current measurements with respect to nominal (healthy) behavior and is sensitive to changes caused by structural damage.

$$R = U_{0,\text{ref}}^\top T_{b,\text{act}} \tag{3.7}$$

The Pearson correlation coefficient defined in Equation (3.8) is used to quantify the similarity between two residual matrices R_i and R_j.

$$r = \text{corr}(A, B) = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}} \tag{3.8}$$

Its interpretation in the context of SHM is as follows:

- If $\text{corr}(R_i, R_j) \approx 1$, the two residual matrices are highly correlated, suggesting that both records likely belong to the **same structural state**.
- If $\text{corr}(R_i, R_j) \to 0$, the residual matrices are not correlated, indicating that the records are from **different structural states**, potentially due to damage or system changes.

The time-series produced are then processed in the DDR phase: the correlation matrices are obtained, whose dimensions are 320x320, a number proportional to that of the channels and to the inner parameters of the applied technique. These matrices are then used in feature extraction to train and test ML classifiers, as described in the following chapters 4,5,6, and 7.

3.4 Feature Extraction in the Time Domain From Hankel Matrices

In time-domain SHM, damage-sensitive features are typically extracted directly from raw time-series signals. However, in our approach, we first constructed Hankel matrices as defined in Equation (3.5), and (3.6). From these matrices, we extracted the individual block matrix $\mathbf{T_b}$ and computed various statistical features to characterize the underlying structural behavior. This transformation allowed us to preserve temporal dependencies while enabling feature extraction from a richer, matrix-based representation of the vibration signals.

We calculated the following **damage-sensitive features** from the time-domain signals:

- **Statistical descriptors:** Mean, standard deviation, variance, **skewness** [29–32], **kurtosis** [33–38], and **crest factor** [39, 40, 40–43], commonly used to characterize the distributional and shape properties of vibration signals [7, 14]. .
- **Energy-based metrics:** Root Mean Square (RMS), **signal energy** [44–50], capturing the overall intensity of the signal.
- **Cumulative indicators: Cumulative sum of differences (CSD)** [9, 10, 51–53], useful for detecting slow-evolving changes or damage progression over time [9, 10] .
- **Similarity-based metric: Cosine Similarity (CS)** [54–60], used to

compare the current signal window with a baseline (healthy) reference.

Here, time-windowing and segmentation techniques [7, 10, 61] are applied to divide the vibration signal into fixed-length intervals, allowing localized analysis of dynamic responses [8]. This segmentation is particularly useful for monitoring the progression of different bridge scenarios and damage intensities, as it allows the detection of transient events [62–65] (for example, sudden stiffness loss or cracking) that may not be observable over the entire duration of the signal. By analyzing features within each time window, the system becomes sensitive to gradual degradation, which is critical in distinguishing between mild, moderate, and severe damage states of railway bridges. Therefore, these techniques improve the temporal resolution of our damage detection framework and support a more granular classification across the defined structural scenarios [14].

3.4.1 Selected Feature Descriptions

To enable effective damage detection and structural state classification, a set of statistical and signal-based features were extracted from vibration signals. These features were chosen based on their sensitivity to structural changes and their proven utility in prior SHM studies. A brief description of the selected features is provided below:

Cumulative Sum of Differences is calculated using the formula given in (3.9). It is the sum of the difference between the original reference value and the relevant subsequent values received from the sensor. A *logarithmic* is applied to the final values as these values are too small.

$$S = log(\sum |b - a|) \tag{3.9}$$

As discussed earlier, a reference data matrix T_{b0} is assigned to calculate the feature values. 1^{st} matrix T_{b0} of each damage intensity is used as a reference matrix. Here, 'a' is used as an element of the reference matrix 'A'. 'b' represents the elements of all other respective matrix 'B', which is used to obtain a cumulative sum of differences by subtracting them from the reference matrix 'a' and calculating their sums. After analyzing the data, the log function is applied to normalize the generated data.

Crest Factor (CF) as defined in Equation (3.10), is calculated as the ratio of the peak value of a signal to its RMS. The crest factor is a measure of the extreme peaks present in a waveform and is often used to detect impulsive events or anomalies in vibration signals.

$$CF = \frac{\text{peak\_value}}{\text{RMS}} \tag{3.10}$$

When matrix-form time domain data are used (for example, T_b of Hankel matrix 3.6), Equation (3.10) becomes:

$$CF = \frac{\max(|\text{matrix}|)}{\sqrt{\frac{1}{N}\sum_{i=1}^{N} \text{matrix}[i]^2}} \tag{3.11}$$

where N is the total number of elements in the matrix (or a row/column if applied row-wise), and max(|matrix|) denotes the maximum absolute value in the signal segment. This form enables the computation of Crest Factor (CF) from block-structured time series data.

Cosine Similarity (CS) is a widely used metric to quantify the directional similarity between two nonzero vectors in an inner product space. This measures directional similarity between two signal segments (for example, baseline and current) [9]. It is defined as:

$$CS(\mathbf{a}, \mathbf{b}) = \frac{\sum_{i=1}^{n} a_i \cdot b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} \cdot \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{3.12}$$

where; $\mathbf{a} = [a_1, a_2, \ldots, a_n]$ is a vector representing the first signal segment (for example, a baseline reference). $\mathbf{b} = [b_1, b_2, \ldots, b_n]$ is a vector that represents the second signal segment (example, current or test data). a_i and b_i are the i-th elements of the vectors $\mathbf{a}$ and $\mathbf{b}$, respectively. n is the dimensionality of the vectors (i.e., the length of the time window or the feature vector). $\sum_{i=1}^{n} a_i \cdot b_i$ is the dot product between the two vectors. $\sqrt{\sum_{i=1}^{n} a_i^2}$ and $\sqrt{\sum_{i=1}^{n} b_i^2}$ are the Euclidean norms (magnitudes) of the vectors $\mathbf{a}$ and $\mathbf{b}$.

Cosine similarity(CS) values range from -1 to 1, where; 1 indicates that the vectors point in the same direction (maximum similarity), 0 indicates that the vectors are orthogonal (no similarity), and -1 indicates that the vectors point in opposite directions.

Energy (Energy (E)) is a fundamental feature of time domain that represents the total strength or power of a signal over a defined time interval. Quantifies the overall magnitude of a vibration signal and is particularly useful in Structural Health Monitoring SHM to characterize dynamic responses in different structural states. For discrete-time signals, the energy is computed using Equation (3.13), where the squared magnitudes of the signal samples are summed over a specified window.

$$E = \log\left(\sum_{n=n_1}^{n_2} |x[n]|^2\right) \tag{3.13}$$

The signal is assumed to be sampled from n_1 to n_2, and $x[n]$ denotes the discrete signal at index n. To improve interpretability and stabilize the variance across a wide range of amplitudes, a logarithmic transformation is applied to the computed energy. This normalization step helps compress large variations in raw energy values and enhances the feature's suitability for input to machine learning algorithms, especially when comparing between different damage scenarios.

Kurtosis is a higher-order statistical feature that quantifies the "peakedness" of a signal distribution. It is particularly sensitive to impulsive or outlier events, making it a valuable indicator for identifying sudden structural changes or damage in vibration-based SHM. A high kurtosis value suggests the presence of sharp transients or extreme deviations from the mean, while a low value indicates a more uniform or flat distribution. The mathematical formulation of kurtosis is given in Equation (3.14), where x_i denotes the i-th sample, $\bar{x}$ is the sample mean, and n is the number of samples in the signal segment being analyzed.

$$\text{Kurtosis} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^2} \tag{3.14}$$

Kurtosis has been widely used in SHM applications to detect local damage, especially when signal responses contain transient bursts, sudden changes in stiffness, or nonlinear behavior [14].

Skewness is a third-order statistical moment that measures the asymmetry of signal distribution. In the context of Structural Health Monitoring (SHM), skewness can help identify directional bias in vibration responses, which may arise due to structural shifts, uneven loading, or asymmetrical damage. As shown in Figure 3.7, a positive skew results in a longer right-hand tail, while a negative skew produces a longer left-hand tail.

This asymmetry is particularly useful in detecting gradual changes or localized structural deformations [66, 67]. The skewness is calculated as shown in Equation (3.15), where x_i represents the i-th sample in the signal, $\bar{x}$ is the sample mean, and n is the total number of samples in the analysis window.

Figure 3.7: Illustration of positive and negative skewness. Positive skewness exhibits a longer tail to the right, while negative skewness shows a longer tail to the left.

$$\text{Skewness} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^{3/2}} \tag{3.15}$$

The logarithmic function is also applied to the crest factor and cumulative sum of differences values as these values are too small to handle easily. The Signal energy reflects the overall magnitude of the vibration and is useful in assessing stiffness loss.

Time-windowing and segmentation techniques are often applied to divide the signal into smaller intervals, allowing localized monitoring and the detection of transient events or gradual degradation. Envelope detection and trend monitoring are also used to highlight modulations in signal amplitude that may correspond to evolving damage.

These features were used as inputs to various machine learning classifiers, including EBMs, SVMs, and neural networks, to predict structural health states such as damage intensity, stress levels, and bridge scenario classifications.

3.4.2 Data Augmentation

In Structural Health Monitoring (SHM), especially when working with limited real-world datasets from railway steel bridges, data augmentation plays a critical role in enhancing model generalization and preventing overfitting. In this work, a statistical feature based augmentation method was applied, focused on adding controlled Gaussian noise to key features such as kurtosis, skewness, energy, and cumulative sum of differences (CSD) etc.

After converting the signal data into ambient features, data augmentation is applied. Data augmentation is performed using the Gaussian normal distribution methods within the range of '0' and the standard deviation of the data. This can be expressed by i) noise generation, ii)data augmentation, and iii)range checking and correction.

Noise Generation: A random noise (3.16) is generated from a Gaussian normal distribution with a mean of '0' and a standard deviation *(noise_std)* of data gathered from a specific sensor. This can be represented as;

$$noise \sim \mathcal{N}(0, noise\_std) \tag{3.16}$$

Data Augmentation: The noise generated is added to the selected feature columns of the input data as an augmented data (3.17). If the original data is denoted as *'data'* and the selected feature columns as *'feature_columns'*, the augmented data *'augmented_data'* is obtained as

$$augmented\_data = data + noise \tag{3.17}$$

Range Checking and Correction: Eq. (3.18) checks if any values in the *augmented data* are outside a valid range. Suppose a value in the augmented data is less than 0 or greater than the standard deviation of the corresponding feature column in the original data. In that case, it is replaced with the original value from the input data. Otherwise, the original augmented data value is placed. Mathematically, this step can be represented as:

$$\text{aug\_data}[i] = \begin{cases} \text{data}[i] & \text{if aug\_data}[i] < 0 \text{ or} \\ & \text{aug\_data}[i] > \text{std\_dev\_fts}[i] \\ \text{aug\_data}[i] & \text{otherwise} \end{cases} \tag{3.18}$$

Here, *std_dev_fts* represents the standard deviation of the corresponding feature column in the original data. These mathematical representations help illustrate the operations performed in data augmentation with Gaussian noise and ensure the augmented data remain within a valid range.

3.5 Advantages and Challenges

Time-domain methods in SHM offer several notable advantages, particularly for real-time monitoring of structural systems. These methods are computationally simple, as they analyze raw sensor signals; such as acceleration or strain, without requiring transformation into the frequency or modal domains [7, 10]. This simplicity makes them ideal for embedded or on-board monitoring systems with limited processing power. Additionally, time domain features such as root mean square (RMS), energy, kurtosis, and skewness are highly sensitive to abrupt changes in structural behavior, making them effective for detecting transient events or local damage [9, 14]. Because these features are derived di-

rectly from raw signals, time-domain methods provide a fast and direct way to assess the structural condition.

Despite these strengths, time domain approaches also face several challenges. One major limitation is that their sensitivity to noise and environmental variability factors such as temperature fluctuations, sensor drift, and changes in operational load can distort signal patterns and complicate the interpretation of damage [61]. In addition, it can be difficult to distinguish between signal changes caused by operational conditions and those resulting from actual damage. Without additional context or modeling, this ambiguity can lead to false positives or undetected failures. To overcome these issues, time domain frameworks SHM often incorporate advanced statistical techniques or machine learning models to improve robustness and classification accuracy [64, 65].

3.6 Integration with Machine Learning

To enhance damage detection and classification capabilities, time-domain features are commonly used as inputs to supervised machine learning models, including:

- **Support Vector Machines (SVM)**: Effective in high-dimensional feature spaces and known for robust classification performance [68–71].
- **Random Forests** [72]: Ensemble-based classifiers that provide strong generalization and handle feature redundancy.
- **Multi-Layer Perceptrons (MLP)** [73]: Fully connected neural networks capable of modeling non-linear relationships.
- **K-Nearest Neighbors (KNN)** [74, 75]: A simple yet powerful nonparametric approach for small datasets or interpretable models.

Recent advances in deep learning have introduced architectures capable of automatic feature learning from raw or structured time series data, reducing the reliance on hand-crafted features.

- **Convolutional Neural Networks (CNNs)**: Used to learn spatial and local temporal patterns in multivariate time series segments [9, 76–78].
- **Deep Neural Networks (DNNs)**: General-purpose multi-layer architectures effective for complex decision boundaries and deep representation learning.
- **Fusion of CNNs**: Combines the local pattern recognition of CNNs with the global context learning of Transformers, improving classification accuracy, especially in heterogeneous or noisy SHM datasets [64, 65].

These data-driven models allow for the automatic extraction of informative

patterns from segmented time-domain signals and have demonstrated improved damage classification performance across various SHM applications.

3.6.1 Model Selection Rationale

The choice of machine learning and deep learning models in this research was guided by both empirical validation and theoretical suitability for the nature of the SHM task. Structural Health Monitoring of railway steel bridges presents unique challenges such as signal variability, high dimensional time–frequency data, real time interpretability requirements, and noise prone measurements. Therefore, the model selection process was not arbitrary, but aligned with domain-specific requirements.

3.6.1.1 RBF Networks

Radial Basis Function (RBF) Networks were selected because of their ability to model nonlinear relationships and localized responses, which are valuable for anomaly detection in bridge health monitoring. RBFs provide faster convergence and simplicity compared to multilayer perceptrons, making them suitable for edge deployment. In addition, their transparent activation patterns allow for better interpretability during early-stage model prototyping.

3.6.1.2 Explainable Boosting Machines (EBMs)

EBMs, based on Generalized Additive Models (Generalized Additive Models (GAMs)), were adopted to ensure transparency in decision-making. Their additive nature allows stakeholders to trace how individual features (e.g., skewness, crest factor) contribute to the final prediction. This is critical in safety critical SHM applications where model accountability is vital. EBMs were preferred over black-box models during the early validation stages and as interpretable baselines.

3.6.1.3 ResNet, DenseNet, and EfficientNet

These Convolutional Neural Network (CNN) architectures were chosen for their proven ability to handle large-scale image-like inputs, in this case, spectrograms derived from STFT-transformed vibration signals.

- **ResNet-50** introduces residual connections, mitigating vanishing gradient problems, and is known for stable convergence on medium-scale datasets.
- **DenseNet-121** encourages feature reuse through dense connectivity, making it more parameter-efficient and suitable for capturing fine-grained patterns in damage evolution.

- **EfficientNet-B0** was evaluated for its balance between accuracy and computational cost, leveraging compound scaling across depth, width, and resolution.

3.6.1.4 Vision Transformer (ViT) Integration

Transformer based models, particularly Vision Transformers (ViT), were incorporated into fusion architectures (e.g., ResNet+ViT, DenseNet+ViT) to leverage their strength in capturing long-range dependencies in spectrograms. SHM signals often exhibit subtle spatial and temporal patterns that CNNs alone may fail to model efficiently. ViTs bring global context awareness into the model pipeline.

3.6.1.5 Comparative Justification

The diversity in chosen models allowed for a comprehensive evaluation across key criteria:

- **Interpretability**: addressed by EBMs and Fuzzy Logic.
- **Accuracy and Feature Representation**: addressed by DenseNet, ResNet, EfficientNet.
- **Temporal-Spectral Generalization**: handled via Transformer-based fusion.
- **Computation and Deployment Efficiency**: RBFs and EBMs are lighter and suitable for real-time monitoring or edge devices.

Thus, each model was selected not solely based on empirical accuracy, but based on its alignment with SHM specific challenges, such as damage localization, alert interpretability, and resilience to sensor noise.

3.7 Summary

Structural Health Monitoring SHM in the time domain provides a practical and powerful approach to access the condition of railway steel bridges. Its ability to operate directly on raw sensor data, such as accelerations and strains, makes it particularly suitable for real-time monitoring applications [7, 9]. Time domain analysis supports rapid anomaly detection and efficient computation without the need for transformation into frequency or modal domains. However, it does present challenges, including sensitivity to measurement noise, temperature effects, and operational variability [63]. To address these limitations, recent studies have integrated time-domain methods with modern machine learning techniques, such as deep neural networks [65] and segment-based generative adversarial networks [64], significantly enhancing classification accuracy and

system robustness. These advancements make time domain SHM a viable and scalable solution for large-scale bridge monitoring networks.

Chapter 4

Radial Basis Function Network for Structural Health Analysis of Railway Steel Bridges

"AI is likely to be the best or worst thing to happen to humanity."

— Stephen Hawking.

This chapter proposes a data driven analytical framework based on a Ra-

dial Basis Function (RBF) based Neural Network (NN) to model the structural health condition of railway steel bridges under various simulated bridge scenarios and damage intensities. The proposed RBF-based approach considers statistical and time domain features;such as E, Kurtosis, Skewness, Crest Factor (CF), Cosine Similarity (CS) and Cumulative Sum of Differences (CSD), extracted from Hankel matrix of vibration signals collected across multiple bridge scenarios and damage intensities. The framework captures the nonlinear mapping between extracted features and corresponding damage labels, offering an interpretable and efficient means of classifying bridge scenarios and damage intensities.

The RBF network architecture is selected for its localized response and generalization capability, which makes it particularly suitable for applications where damage signatures are complex and sparse. **Damage signatures** refer to identifiable patterns or features in measured structural response data (for example, acceleration, strain, displacement) that indicate the presence, location or severity of damage in a structure [9, 79–83].

Damage signatures play a critical role in Structural Health Monitoring (SHM) of railway steel bridges, as they represent measurable deviations in the structural response, indicative of underlying damage or anomalies of the steel structure. These signatures, derived from vibration data, enable early detection, localization, and classification of the damage states of the bridge.

4.1 Introduction

Structural Health monitoring is becoming a field of focus to the engineers and research scientist mainly after introduction of machine learning. SHM systems are helpful in accessing the condition and integrity of structures such as bridges, buildings, pipelines, and aircraft over time. ML techniques such as; KNN, RF, SVM, Artificial Neural Network (ANN), CNN etc., provide valuable insights and predictive capabilities to monitor the structural health of such assets. SHM also provide a quantitative measure of a structure's condition over time.

Ambient analysis of railway bridges (steel or concrete) is necessary for their long life. With the advancement of Structural Health Monitoring (SHM) technologies, it has become feasible to detect early structural anomalies, track the occurrence of damage, and trigger timely alerts for preventive maintenance. This enables condition-based infrastructure management, reducing unexpected failures and extending the bridge's service life. This lead researchers to do systematic structural research about dynamic responses of vibrations, natural

frequencies, accelerations of vehicles on the bridges and temperature variations (when train passes and after train passes, in our case) [84].

Machine Learning algorithms can be trained to detect and classify different types of damage or anomalies in structures. This includes identifying cracks, corrosion, fatigue, and other structural issues from sensor data, images, or sensor fusion. These algorithms also help in how to predict when maintenance is needed based on sensor data and historical performance. This helps in scheduling maintenance activities more efficiently, reducing downtime, and preventing catastrophic failures. In bridge SHM, anomaly detection is used to detect deviations of bridges from normal behavior which may indicate damage, wear & tear in the structures which needs attention, repair and/or maintenance.

Although machine learning enables remote monitoring of civil structures through the use of sensors and data communication technologies, its effectiveness depends on several critical factors. This capability is especially advantageous for infrastructure located in remote or hazardous environments, where manual inspection is challenging or costly. However, the success of machine learning in SHM depends not only on the quality and quantity of sensor data, but also on the selection of suitable algorithms and the availability of domain expertise to interpret the results. In practical deployments, hybrid approaches that integrate traditional engineering analysis with data-driven models are often employed to improve reliability and enhance decision making for critical infrastructure monitoring.

RBF Networks, a class of artificial neural networks, are particularly well suited for function approximation, classification, and pattern recognition tasks. In the context of SHM for railway steel bridges, RBF networks offer an effective and interpretable solution for damage detection and classification based on statistical and signal-derived features extracted from time-domain or frequency-domain vibration data. A typical RBF network architecture consists of three layers: an input layer, a hidden layer that applies nonlinear radial basis functions as activation units, and a linear output layer. This structure allows the network to model localized patterns and non-linear relationships between input features and structural conditions, making it especially suitable for identifying subtle damage signatures across varying bridge scenarios.

4.2 Overview

The core strength of RBF networks lies in their ability to model nonlinear relationships using localized responses around selected center points in the input feature space. This makes them particularly effective for SHM applications,

where damage patterns may be sparse, complex, or difficult to separate linearly. When trained on labeled datasets representing various bridge damage scenarios and intensities, RBF networks can learn to distinguish between healthy and damaged structural states with high accuracy.

In this study [84, 85], in the context of SHM for railway steel bridges, RBF networks are employed to classify structural conditions based on features extracted from time domain sensor data. Compared to traditional feedforward networks, RBF networks offer a balance between nonlinear representation power and interpretability, making them especially attractive for critical infrastructure monitoring, where transparency and reliability are essential.

The main contributions of this paper [85] are:

- A comprehensive investigation on RBF based Neural Network .
- A Bridge Scenario and Damage Intensities based railway steel bridge damage detection .

4.3 Related Work

SHMs provide many viable solutions to the damage detection but most of these are only limited to anomalies, predictions or statistical model comparisons.

Recent studies have explored a variety of signal processing and data-driven approaches for extracting meaningful damage signatures. For instance, Ren et al. [82] provided a comprehensive review of vibration-based damage detection techniques, highlighting how specific statistical features serve as reliable indicators of damage. Ahmed et al. [80] demonstrated the use of nonlinear ultrasonic pulse-echo signatures for identifying fatigue-induced damage in composite structures. Additionally, Sonker and Shanker [79] proposed an Electromechanical Impedance (EMI)-based approach integrated with neural networks for multiple damage detection. The growing application of deep learning techniques has further expanded how damage signatures are utilized, as reviewed by Jia and Li [83]. These advancements affirm that damage signatures are foundational to the development of robust, scalable SHM systems.

As it can be seen that Svendsen et. al. [86] investigated statistical comparison models with supervised and unsupervised learning using Mahalanobis Squared Distance only. This study only focus of Receiver Operating Characteristic (ROC) metrics using KNN, RF, and SVM etc. which don't give any insights of the dataset used and the damage levels.

Similarly, Frederic et. al. [87] investigated application of machine learning methods on real railway bridges monitoring with transient relationship between air temperature and bridge temperature. He used Neural network with three

input neurons in the input layer, one output neuron in the output layer giving binary results and n hidden neurons in one hidden layer.

Neves et. al. in [88] worked on different ML techniques for damage detection.In this study ANN based model is discussed for model-free bridge damage detection. Sensor based ambient features are extracted and used as an input to ANN which are collected from dynamic response of the structure in two different damage scenarios. In a review study by Onur et. al. [89] have discussed non-parametric and parametric methods for structural damage detection from the ambient data. He reviewed these methods wrt supervised machine learning algorithms. Mehrjoo et.al. [90] proposed simple ANN based bridge damage detection techniques in which he used accelerations as a characteristic to calculate the damage sensitive features. Later, he developed a simple MLP with a single hidden layer for damage identification and localization.

Lee et. al. [91] also used ANN model based on the FE model of the Hannam Grand Bridge in South Korea. They also used ambient features to perform test performance under three levels of damage. They used Probabilistic NN, Back propagation based NN and Sequential NN for results evaluations. Furthermore, Muttillo et. al. [92] also worked extensively on machine learning based model for damage detection. Thus this work is specifically inspired by IoT based sensory systems designed for structural damage indication. Recent studies utilizing IoT-enabled sensor networks, combined with basic machine learning algorithms, have demonstrated the feasibility of real-time monitoring and anomaly detection in civil infrastructure. Building on these foundational approaches, this research extends the idea by incorporating deeper neural network architectures, allowing for more expressive modeling of nonlinear patterns in vibration data. Through these enhancements, the system is expected to achieve higher accuracy in damage detection and localization, even under noisy or variable operational conditions.

4.4 Methodology

The aim of this research is to design an RBF based intelligent monitoring system capable of receiving input from accelerometer sensors and predicting anomalies in a railway steel bridges, with the added capability of localizing structural damage. The proposed architecture, begins with a reference Finite Element Model FEM, as described in **Section 3.3**, that simulates healthy and damaged bridge states. The system includes a data acquisition bus that collects synchronized time series data from multiple sensors (e.g., accelerometers) deployed across critical bridge locations. This data flows into the **perception layer**,

which is responsible for utilizing the extracted features from Hankel matrices, as described in **Section 3.4**. These features are used as inputs to machine learning algorithms for localization of damage scenarios and classification of damage intensities. The perception layer thus transforms input to into high level predictions about the health of the structure. These predicted outputs, indicating the type, location, and severity of damage, are then passed to the **reasoning layer**, where they are interpreted to generate alerts and trigger decision making protocols for maintenance or further inspection.

The processed features are then passed to the **reasoning layer**, where anomaly detection and damage classification are carried out using machine learning models. This layer also incorporates event correlation logic to interpret multi sensor outputs and infer complex damage patterns. Finally, the system integrates with a **dashboard interface**, which visualizes real-time status and generates alarm alerts when damage is detected or threshold conditions are breached.

4.5 Radial Basis Function Networks in Structural Health Monitoring

Radial Basis Function (RBF) Networks have emerged as an effective and interpretable class of artificial neural networks for damage detection and classification in Structural Health Monitoring (SHM). An RBF network typically comprises three layers: an input layer, a hidden layer with nonlinear radial basis activation functions, and a linear output layer. Unlike traditional multilayer perceptrons, RBF networks apply a distance-based transformation in the hidden layer, providing a localized response to input features.

In SHM applications, especially for railway steel bridges, sensor data such as vibration or strain signals are collected and processed into statistical or signal-based features. These include metrics such as RMS, skewness, kurtosis, cumulative sum of differences, and the crest factor. These features are passed into the RBF network, where each hidden unit computes a radial basis response with respect to a learned center. The general activation of the j-th RBF neuron is given by:

$$h_j(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_j\|^2}{2\sigma_j^2}\right) \tag{4.1}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the input vector, $\mathbf{c}_j \in \mathbb{R}^d$ is the center of the j-th RBF neuron, and σ_j is the width parameter controlling the spread of the radial response.

The outputs of all RBF neurons form a new feature representation that is passed to the output layer for classification. This output represents predicted structural states such as damage localization, severity, or type. RBF networks offer high interpretability, fast training, and effective performance in low data regimes, making them suitable for SHM systems where transparency and real-time deployment are essential.

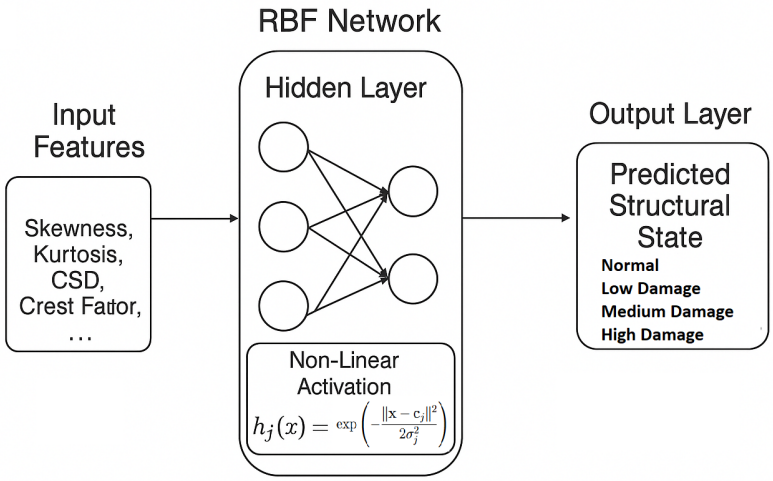


Figure 4.1: RBF-based SHM architecture for damage classification using statistical features from sensor data.

Initial RBF based Neural network is generated with input layer, RBF, activation functions, and other inner details of the neural net and an output layer which describes the seven scenarios for bridge sections and twenty six damage intensities. To answer why RBF is used in the hidden layer lies in a fact that they provide promising results in Structural Health Monitoring (SHM) applications for various purposes, including damage detection, feature extraction, and data analysis. However, we have used an updated version of RBFNet. This updated neural network i.e AmbientNet has four hidden layers, maxPool, leakyReLU, Batch normalization and finally Softmax for prediction.

RBF networks can be used for regression tasks to predict structural health-related parameters, such as stress, strain, or deformation, based on sensor measurements. This can be valuable for continuously monitoring structural conditions. RBF network is designed for anomaly detection and damage localization for the expected behavior of a bridge structure. Any deviation from the normal observation of the predictions indicated the damage levels and localization which can help in bridge maintenance..

4.5.1 RBF-Net

Radial Basis Function (RBF) network can be described in terms of its activation function and its output. Each neuron in the hidden layer applies an RBF activation function as shown in Equation 4.2, to the input data.

$$\phi_i(x) = \exp\left(-\lambda_i \cdot \|x - c_i\|^2\right) \tag{4.2}$$

As the bridge is divided into seven sub-structures (for damage localization detection) and in simulated architecture; different damage intensities are applied, it is important to note that RBF networks require appropriate training data that includes both healthy and damaged structural states. Additionally, the choice of network architecture, including the number and placement of RBF neurons, can have a significant impact on their performance. Furthermore, our ambientNet composed of a fully convolutional layer followed by an RBF function, batch normalization, leakyReLU as activation function, max-pooling and in the a softmax activation. For our input data, this model with some variations is used for the preliminary results.

4.6 Results Discussion

For our designed RBF-based neural network, ambient vibration features were used as input. These features were chosen for their ability to capture dynamic behavior related to structural degradation, while being computationally efficient for real-time deployment. The network was trained on two distinct classification problems: **26 damage intensities**, derived from simulated Finite Element Model (FEM) data representing progressive damage states, and **7 bridge scenarios**, where a real railway steel bridge was subdivided into seven monitoring zones (see Fig. 3.3 and Fig. 3.4). The preliminary training results were promising, demonstrating the model's capacity to learn distinct patterns associated with structural variations.

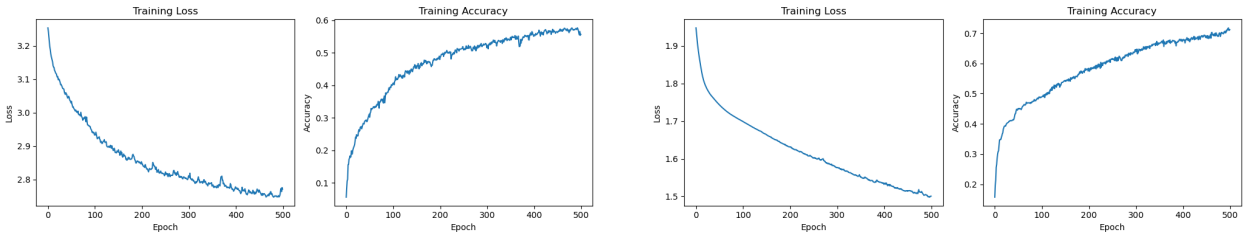
However, these promising results were limited to the training phase. During testing, the network exhibited poor performance and was unable to meet the accuracy and stability requirements needed for reliable deployment. This suggested potential overfitting or limitations in the RBF layer's ability to model complex, real-world scenarios beyond simulated data. Consequently, we expanded our investigation to include additional machine learning models. This included classical algorithms such as Support Vector Machines (SVM), Random Forests (RF), and k-Nearest Neighbors (KNN), as well as advanced deep learning architectures such as Convolutional Neural Networks (CNNs) and Deep

Neural Networks (DNNs), to better capture both spatial and temporal features within the SHM dataset.

This section presents the training results of the Radial Basis Function Network (RBFNet) for the classification of structural conditions in railway steel bridges. Two primary classification tasks were addressed: Damage Intensities (DI) classification, with class labels corresponding to damage intensities (1-90%), and Bridge Scenarios (BS) (S1-S7) classification.

4.6.1 Performance with Original Data

To evaluate the baseline performance of the proposed RBF-based neural network, we first trained the model on unaugmented, original feature sets for two classification tasks: damage intensity (1-90%) and bridge scenario (S1-S7). The training loss and accuracy progression for both tasks over 500 epochs are shown in Figure 4.2.



(a) Damage intensity classification (1–90%) (b) Bridge scenario classification (S1–S7)

Figure 4.2: Training loss and accuracy of RBFNet using original data for (a) Damage Intensity and (b) Bridge Scenario.

In the case of **damage intensity classification** (Figure 4.2a), the model demonstrates a steady reduction in training loss, converging around 1.5. The classification accuracy gradually increases and peaks at approximately **70%**, with a consistent upward trend. This indicates that while the RBFNet is capable of learning from raw damage intensity data, the discriminative power of the features may be limited due to class overlap or insufficient variability between closely spaced intensity levels.

For the **bridge scenario classification** task (Figure 4.2b), the network performs comparatively worse. Although the loss curve shows a gradual decrease, the training accuracy stabilizes around **58%**. This suggests that distinguishing between spatially distributed damage states using only the original feature set is more challenging. These results emphasize the importance of feature enhancement or augmentation to improve classification performance, particularly for spatial damage localization.

In summary, while the RBFNet shows potential when trained on original,

unaugmented data; especially for damage intensity estimation—the limited expressiveness of the raw features hinders its performance in scenario classification. These observations motivate the need for data augmentation, which is further investigated in the subsequent experiments.

4.6.2 Performance Analysis with Augmented Data

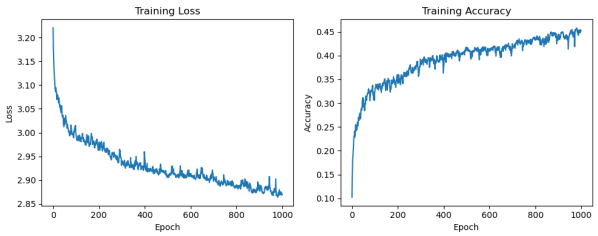
To improve the generalization of our RBF-based neural network, we applied data augmentation techniques to the original input features. Figures 4.3a and 4.3b illustrate the resulting training loss and accuracy curves for the **damage intensity (DI)** and **bridge scenario (BS)** classification tasks, respectively.

In the case of **damage intensity classification** (Fig. 4.3a), the model shows a steady decrease in training loss over 1000 epochs, with a notable early convergence phase. The accuracy gradually improves and reaches approximately **45%**. This represents a significant improvement compared to training on unaugmented data, where the model plateaued near 30%. The improvement highlights the importance of input diversity, especially in damage intensity classification, where minor signal variations are often hard to learn without data enhancement.

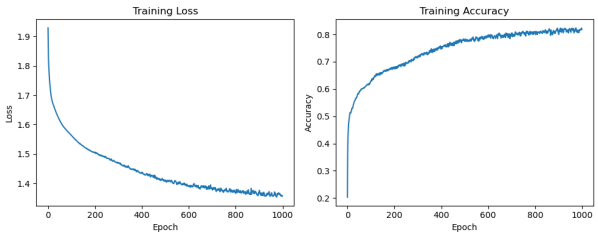
For the **bridge scenario classification** task (Fig. 4.3b), the model achieves even more encouraging results. Training accuracy reaches around **82%**, with a consistent decline in loss. This performance confirms that the RBFNet architecture is better at learning spatially distinct patterns that arise from different bridge scenarios. The higher accuracy in bridge scenario classification compared to damage intensity classification also suggests that localization patterns are more distinguishable in the feature space than damage intensity differences.

Overall, these results demonstrate the effectiveness of data augmentation in improving model robustness and classification performance. The improvements validate the hypothesis that class separability in SHM tasks can be enhanced through strategic input-space transformations, particularly when using localized learners like radial basis function networks.

The augmented dataset enriched the diversity of the feature space, allowing the RBF layer to generalize better to minor variations in input and avoid overfitting. Furthermore, the training loss curves exhibited improved smoothness and stability, indicating more efficient optimization.



(a) Damage Intensity Classification (1-90%)

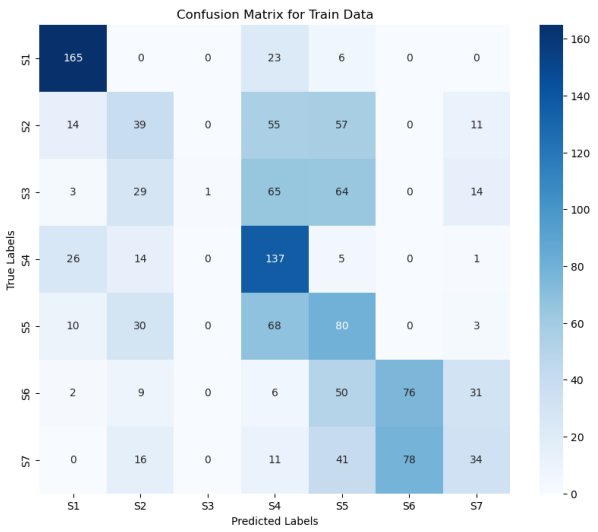


(b) Bridge Scenario Classification (S1-S7)

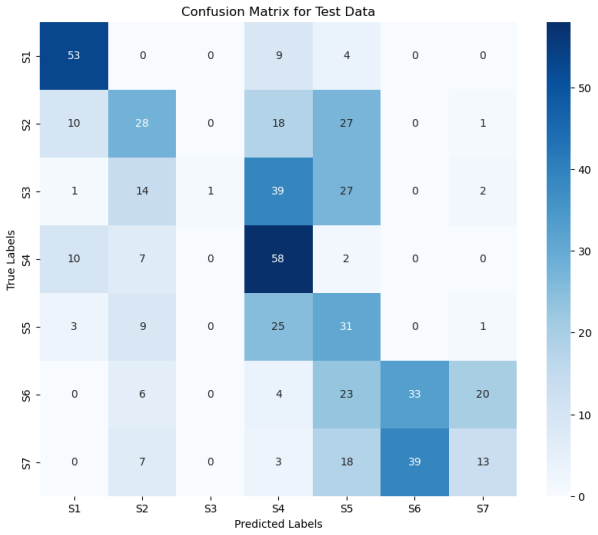
Figure 4.3: Training loss and accuracy of RBFNet using augmented data for (a) Damage Intensity and (b) Bridge Scenario.

4.6.3 Confusion Matrix Analysis for Bridge Scenario Classification

To further evaluate the performance of the proposed RBF based Neural Network (RBFNet) model on bridge scenario classification (S1-S7), we analyze its confusion matrices for both training and testing datasets, shown in Figure 4.4. The matrices reflect how often predictions match the true classes and highlight areas of confusion between specific bridge zones.



(a) Confusion matrix on train data



(b) Confusion matrix on test data

Figure 4.4: Confusion matrices for RBFNet bridge scenario classification: (a) training dataset and (b) testing dataset.

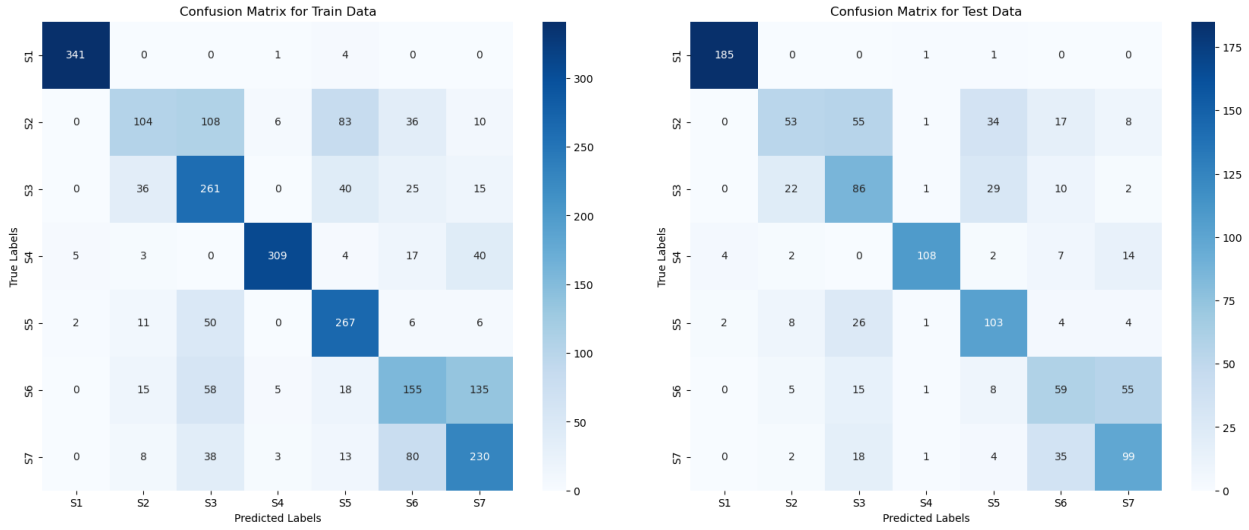
From the training matrix (Fig. 4.4a), we observe that the model performs reasonably well across most classes, with the highest accuracy in scenario S1 (165 correct predictions) and S4 (137 correct). However, considerable confusion exists between neighboring or structurally similar scenarios, such as S2/S5 and S3/S4, suggesting overlapping feature distributions in these zones.

The test confusion matrix (Fig. 4.4b) reveals a more realistic measure of generalization. Correct predictions decline slightly for each class, and misclassifications become more prominent. For instance, many samples from S3 and

S5 are misclassified as S4 and S6, indicating that while the RBFNet captured structural patterns during training, its ability to separate similar scenarios under unseen conditions is limited.

4.6.4 Confusion Matrix Analysis with Augmented Data for Bridge Scenario Classification

To improve the model's ability to generalize across damage distributions, data augmentation strategy was applied to the original feature set. Figure 4.5 shows the confusion matrices for training and testing datasets in the bridge scenario classification task using the augmented features.



(a) Confusion matrix on Augmented Train data (b) Confusion matrix on Augmented Test data

Figure 4.5: Confusion matrices for RBFNet bridge scenario classification (S1-S7) on (a) training and (b) testing datasets using augmented features.

The confusion matrix for the training data (Figure 4.5a) indicates strong diagonal dominance across most scenarios, especially for S1, S3, S4, and S5, showing that the network has learned the spatial features of these classes well. Despite minor confusion in overlapping zones such as S6 and S7, the overall classification remains robust with relatively low off-diagonal misclassifications.

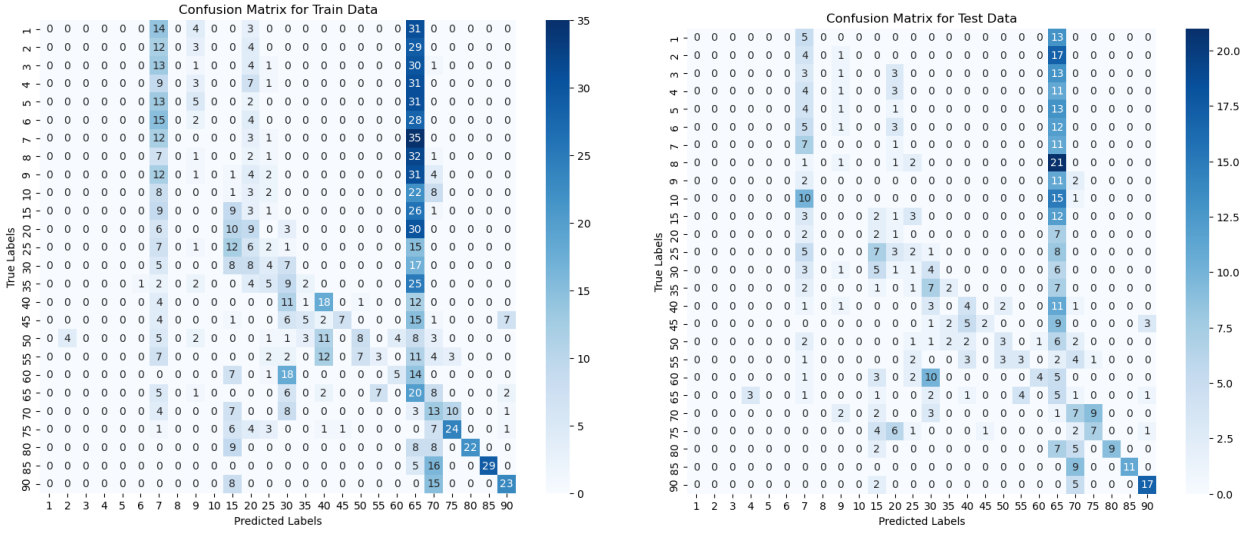
In the testing matrix (Figure 4.5b), the network retains good generalization performance. Class S1 shows near-perfect classification (185 correct), while scenarios S3, S4, S5, and S7 also show solid diagonal performance. Some confusion remains between spatially adjacent zones; particularly between S2 and S3, and between S6 and S7; likely due to similarity in vibration signatures near structural transitions.

Overall, the augmentation method enhanced the model's resilience to input variation, helping it capture discriminative patterns across bridge zones more

effectively. These results demonstrate that augmentation introduces sufficient variability to boost generalization, especially in tasks requiring localization of structural changes across spatially defined monitoring sections.

4.6.5 Confusion Matrix Analysis for Damage Intensity Classification

To evaluate the performance of the proposed RBFNet model on fine-grained damage intensity classification (1–90%), confusion matrices were generated for both training and testing datasets (see Figure 4.6). These matrices illustrate the relationship between true damage intensity levels and their corresponding predicted classes, highlighting how well the network distinguishes subtle variations in damage.



(a) Confusion matrix on Train data

(b) Confusion matrix on Test data

Figure 4.6: Confusion matrices for RBFNet damage intensity classification (1-90%) on (a) training and (b) test datasets.

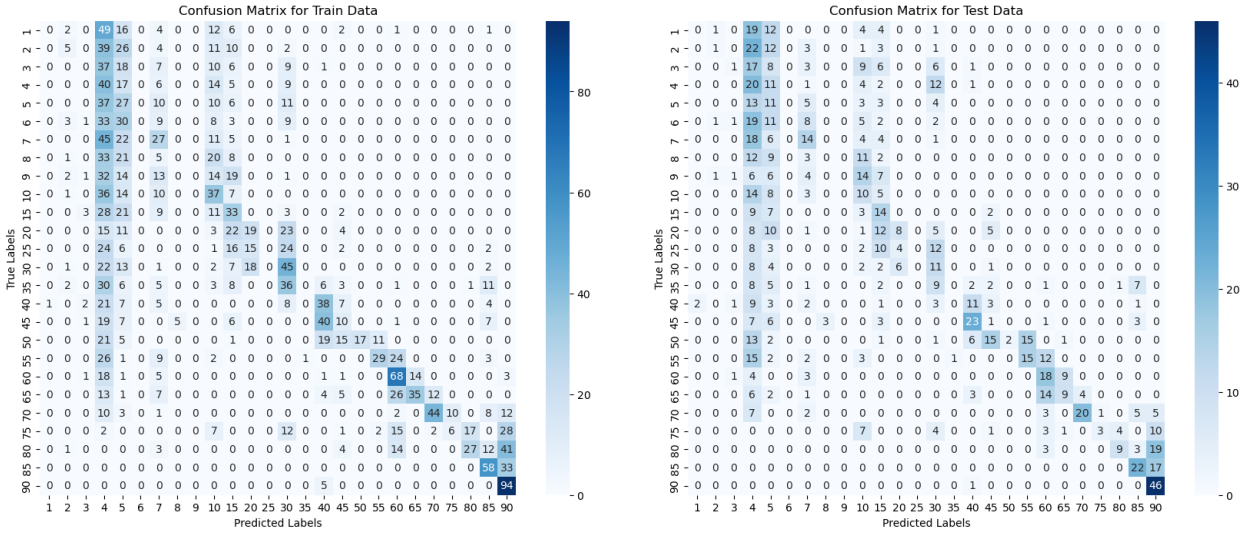
The training confusion matrix (Figure 4.6a) shows that the network is capable of learning patterns for a wide range of damage intensities. The diagonal dominance indicates correct predictions for certain intensities, particularly near class 60, 70, and 90. However, many neighboring classes (for example, 10 vs. 15, 20 vs. 25) are frequently confused, reflecting the inherent difficulty in distinguishing between closely spaced intensity levels with limited signal variation.

The test confusion matrix (Figure 4.6b) reveals a more realistic measure of the network's generalization ability. While the diagonal trend is still visible, it is significantly less pronounced compared to the training case. The majority of predictions tend to cluster around specific intensity bands (notably around 60 and 70), indicating that the model may be overfitting to dominant training pat-

terns or underrepresenting other class ranges. Misclassifications are widespread for lower intensities (1-30), suggesting that signal features associated with early damage stages are not distinct enough for the model to learn reliably.

4.6.6 Confusion Matrix Analysis with Feature-Wise Gaussian Noise Augmentation for Damage Intensities [1-90%]

To enhance the diversity and robustness of the input features, a targeted data augmentation technique based on Gaussian noise addition was implemented.The impact of this augmentation strategy on the classification performance of the RBFNet model for damage intensity classification (1-90%) is evaluated through the confusion matrices shown in Figure 4.7.



(a) Confusion matrix on Augmented Train data (b) Confusion matrix on Augmented Test data

Figure 4.7: Confusion matrices for RBFNet damage intensity classification (1-90%) on (a) training and (b) testing datasets using Gaussian noise-augmented features.

The confusion matrix for the training set (Figure 4.7a) shows clear improvements in class wise separation compared to the original data. Diagonal dominance across several intensity levels indicates that the model has learned to correctly classify a wide range of damage intensities. Notably, mid- and high range intensity classes (for example, 45-90) demonstrate stronger classification consistency.

The test confusion matrix (Figure 4.7b) confirms the generalization benefit of the augmentation. The network maintains good performance on several key intensity levels, especially in higher ranges (60-90), where clearer signal patterns are likely more distinct and less affected by noise. Although some confusion remains in lower intensity classes (e.g., 1-20), this is expected due to the minimal structural deviation and lower feature variance at early damage

stages.

Overall, augmentation enhanced the network's sensitivity to damage progression and mitigated overfitting by increasing intraclass variability during training. These results validate the use of tailored data augmentation strategies for advancing SHM tasks where labeled data are often limited and damage states may evolve subtly.

In summary, while RBFNet demonstrates potential in modeling fine tuned damage intensity levels, performance is limited by class imbalance, subtle signal differences between nearby intensities, and possibly insufficient feature separability. These observations motivate the use of advanced architectures (for example, RBF-CNN hybrids) or feature enhancement techniques (for example, spectrograms, data augmentation) to improve classification accuracy in future work.

4.6.7 Performance Summary

The confusion matrix analysis for bridge scenario classification demonstrates that the RBFNet model performs significantly better when trained on augmented data. The training results show strong class separation, particularly for scenarios S1, S3, S4, and S5, with minor confusion observed in structurally similar zones such as S6 and S7. Test set performance closely follows, indicating good generalization with reduced overfitting. The augmentation technique effectively improved the model's ability to distinguish between spatial damage zones by enriching the feature space and introducing variability that mimics real world noise and signal distortion. These results validate the suitability of RBFNet combined with augmentation for damage localization in railway steel bridges under ambient vibration-based SHM conditions.

Table 4.1: Summary of RBFNet performance on BS and DI classification tasks

Input Type	**Task**	**Accuracy (%)**	**Remarks**
Original Data	Damage Intensity (DI)	∼30%	Difficult class separation
Original Data	Bridge Scenario (BS)	∼60%	Moderate, stable performance
Augmented Data	Damage Intensity (DI)	∼60%	Improved feature diversity
Augmented Data	Bridge Scenario (BS)	∼80%	Best performance overall

4.7 Summary

This study presented a Neural Network-based Structural Health Monitoring (SHM) framework, specifically employing a Radial Basis Function (RBF) neural network, for the detection, localization, and classification of damage

in railway steel bridges. Although neural networks are traditionally associated with image classification and object detection tasks, this work demonstrated their applicability to vibration-based SHM using time domain features extracted from accelerometer data.

The proposed RBF-Net architecture was trained and evaluated on two key classification tasks: damage intensity estimation (1-90%) and bridge scenario classification (S1-S7). The results showed that the network was capable of learning structural patterns and achieved promising training accuracy, particularly in scenario classification. However, the model's generalization on unseen test data was initially limited, especially for damage intensity prediction, which is inherently more difficult due to the subtle nature of signal variations across severity levels.

To address this, data augmentation is introduced. These significantly improved the model's ability to generalize, with performance gains observed across both classification tasks. The results confirmed that RBF-based neural networks, when paired with appropriate preprocessing and feature augmentation, can serve as effective classifiers for structural condition monitoring.

Future work may explore deeper architectures, such as RBF-CNN hybrids or attention-enhanced models, which could improve feature extraction and decision making in more complex or noisy SHM datasets. Additionally, further validation on real world bridge data, including testing under varying operational and environmental conditions, would help assess the robustness of the proposed system for real-time deployment.

Chapter 5

Railway Steel Bridges Health Monitoring using Explainable Boosting Machines

"We're moving from programming computers to teaching them."

—Fei-Fei Li.

In the field of Structural Health Monitoring (SHM), particularly for critical infrastructure such as railway steel bridges, the need for models that are not only accurate but also interpretable has become increasingly important. Although complex deep learning models such as CNNs or RBF-based neural

networks provide strong performance, their **lack of interpretability** [93–95] can hinder practical deployment, especially when decisions must be validated by domain experts or regulators. Explainable Boosting Machines (EBMs) provide a promising solution to this challenge by combining the accuracy of ensemble methods [96–98] with the transparency of additive models [94, 99, 100].

5.1 Introduction to Explainable Boosting Machines in SHM

In Structural Health Monitoring (SHM) of railway steel bridges, the ability to detect, classify, and interpret structural damage with high reliability is crucial [94, 95, 99]. While machine learning techniques; such as neural networks, support vector machines, and ensemble classifiers; have demonstrated strong performance in vibration based SHM, many of these models operate as "black boxes", making them difficult to interpret and trust in real world deployments.

Explainable Boosting Machines (EBMs) offer a promising alternative by providing a balance between accuracy and transparency. EBMs are a class of Generalized Additive Models (Generalized Additive Model (GAM)s) that use gradient boosting over shallow decision trees to learn shape functions for each feature. This allows engineers to independently examine how input features; such as kurtosis, skewness, CSD, and energy, contribute to the model's prediction. Additionally, EBMs selectively include pairwise feature interactions when they significantly enhance model performance, all while maintaining full interpretability.

In the SHM context, where sensor data must be translated into actionable insights, EBMs allow stakeholders to not only predict structural states (for example, bridge scenario, damage intensity, or stress force) but also understand the "**why**" behind each prediction. This transparency is particularly important in safety critical infrastructure like railway bridges, where maintenance decisions must be supported by explainable evidence. As such, EBMs are well suited for integration into SHM frameworks where interpretability and decision accountability are essential.

5.1.1 Motivation for Interpretability in SHM

In safety critical applications such as Structural Health Monitoring (SHM) of railway steel bridges, it is crucial to deploy machine learning models that not only provide accurate predictions but are also interpretable [95, 99]. Engineers, stakeholders, and regulatory authorities often require a clear understanding of the reasoning behind model outputs, particularly when the outputs influence

maintenance or safety decisions. Traditional black box models, such as deep neural networks, offer limited transparency and can be difficult to trust in practical SHM deployments.

5.1.2 Application to Railway Steel Bridge SHM

EBMs are particularly useful in applications where **model interpretability** is crucial, such as; **Healthcare** (diagnostic decision support), **Finance** (credit scoring, fraud detection), **Structural Health Monitoring (SHM)** (damage detection, risk assessment), and **regulatory compliance** (where model decisions must be justified) [9, 65, 101, 102].

Unlike **black box** models such deep neural networks or random forests, EBMs provide **clear insights** into how input features influence predictions. In our study, EBMs were applied to time domain based statistical features derived from vibration signals of railway steel bridges. The goal was to classify different structural states, including **damage intensities**(1%-90%) and **bridge scenarios**(S1–S7), and **stress force**(CL3-CL40), using interpretable models [93–95].

One of the key benefits of EBMs in this context is their ability to provide localized explanations. For example, the model can reveal that high kurtosis in sensor data from a specific section (for example, pier or crossbeam) strongly correlates with moderate damage in scenario S4, while skewness contributes minimally. Such granularity enables engineers to validate predictions and gain confidence in the decision process.

5.1.3 Performance and Interpretability

The EBM model was trained using the Explainable Boosting Classifier from the `interpretml` [99] Python package. The training results showed competitive accuracy compared to neural networks, especially on clean or augmented datasets, with the added advantage of full interpretability. Visualization of feature effects allowed direct interpretation of how structural conditions influence classification outputs.

5.1.4 Limitations and the Role of Secondary EBMs

While EBMs provide interpretability and robust performance, the model may not capture complex dependencies among higher-order features or multiple interacting structural states. To address this, a **secondary EBM** can be introduced in a stacked architecture. The primary EBM is trained on original sensor features and predicts intermediate attributes such as:

- Predicted Bridge Scenario (for example, S1–S7),
- Predicted Damage Intensity (for example, 1–90% degradation),
- Estimated Stress Force.

Keep in mind that for prediction of **bridge scenario** and **damage intensities**, one-hot-encoding of **stress force** is also used as a feature in addition to statistical features. These outputs are appended to the original feature vector and passed to a **secondary EBM**, which then produces the final classification. This approach improves classification performance while maintaining interpretability, as both stages use transparent, additive models.

5.1.5 Contribution of This Chapter

This chapter presents a two-stage EBM-based framework for SHM, applying it to simulated datasets of railway steel bridges. It demonstrates how EBMs can be stacked to enrich input representations using predicted structural attributes while retaining full interpretability. The results show that secondary EBMs improve little for classification accuracy for both stress force and bridge scenario tasks compared to single stage EBMs. The chapter also includes visualizations of shape functions and feature importances, supporting the claim that EBMs are not only accurate but also explainable, making them highly suitable for deployment in real-world SHM systems.

5.2 Explainable Boosting Machines (EBMs)

EBMsare a class of GAMs implemented using modern machine learning techniques. Introduced as part of the Interpretable ML framework, EBMs are based on gradient boosting [103–105] over individual features and pairwise interactions, while preserving interpretability. Unlike black-box models, EBMs allow practitioners to **visualize** and **understand** how **each feature contributes** to the final prediction. This makes them highly suitable for SHM applications, especially for Railway steel bridges, where engineers must explain the basis of any damage classification or alert.

EBMs are based on Generalized Additive Models (GAMs), which model the target variable as a sum of individual feature functions, as shown in Equation 5.1;

$$g(\mathbb{E}[y]) = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n) \tag{5.1}$$

where g is the link function (for example, the *logit* function for classification or the identity function for regression), β_0 is the intercept or global bias term,

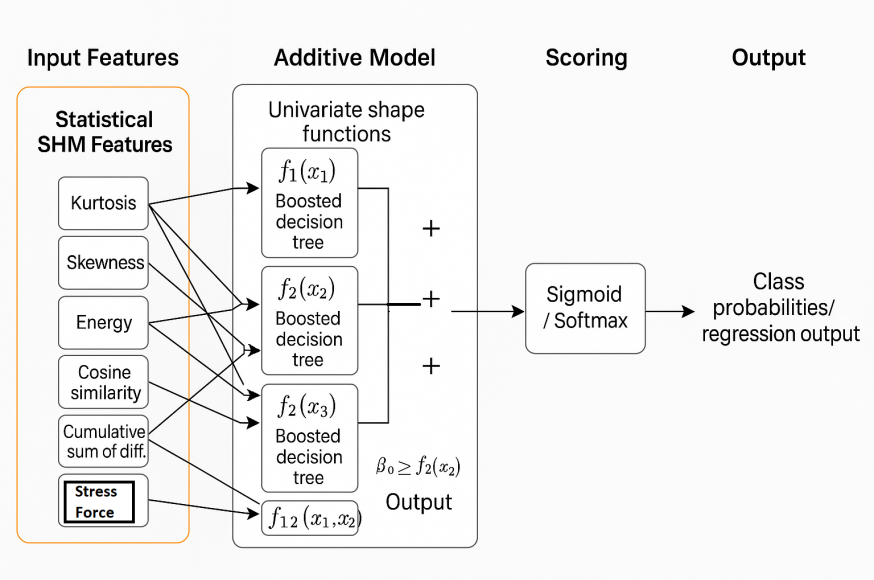


Figure 5.1: EBM Model Architecture for SHM of Railway Steel Bridge

and $f_i(x_i)$ are **smooth, univariate functions** (for example, splines or shallow decision trees) that capture the relationship between each feature x_i and the expected output.

Equation 5.2 extends this formulation to define the structure of an Explainable Boosting Machine (EBM), which enhances a standard GAM by incorporating selected pairwise feature interactions. The model is expressed as:

$$\hat{y} = \beta_0 + \sum_{j=1}^{d} f_j(x_j) + \sum_{j<k} f_{jk}(x_j, x_k) \tag{5.2}$$

In this formulation; β_0 is the intercept term, $f_j(x_j)$ are univariate shape functions learned independently for each feature x_j. These are trained using shallow, gradient-boosted decision trees and are designed to be interpretable through 1D plots, and $f_{jk}(x_j, x_k)$ are pairwise interaction functions, included only for the most influential feature pairs, allowing EBMs to capture moderate feature dependencies while retaining interpretability.

The additive structure and visualization friendly components of our EBM framework (see Figure 5.1) allow users to independently inspect how each feature and the selected pairwise interactions contribute to the model prediction. This level of transparency is crucial in high stakes and safety critical applications such as Structural Health Monitoring SHM of railway steel bridges, where understanding the reasoning behind each prediction is as important as the prediction accuracy itself.

5.2.1 Structural Engineering Interpretation of EBM Outputs

While Explainable Boosting Machines (EBMs) provide transparent and interpretable models through feature importance scores and shape functions, their full value emerges when these outputs are linked to structural engineering principles. In this section, we contextualize EBM-driven findings in terms of vibration-based damage mechanisms in railway steel bridges.

5.2.1.1 Feature Importance and Structural Response

EBM feature importance scores quantify how strongly each predictor influences the final damage classification. In vibration based SHM, these predictors correspond to statistical and spectral features such as:

- **Skewness:** sensitive to asymmetry in vibration signals, often increasing under localized stiffness loss,
- **Kurtosis:** associated with impulsiveness in the signal, indicative of crack initiation or impact; like nonlinear behavior,
- **Energy and RMS:** represent overall vibrational activity, typically rising as damage increases and the structure becomes more flexible,
- **CSD (Cumulative Sum of Differences):** reflects longer-term signal drifts tied to degradation,
- **Cosine Similarity:** compares current responses with baseline healthy states; lower similarity indicates emerging damage.

EBM's ranking of these features aligns with known structural phenomena. For instance, kurtosis and skewness appearing as high-importance features is consistent with literature showing that nonlinearities introduced by cracking or connection looseness primarily distort the upper statistical moments of vibration signals.

5.2.1.2 Shape Functions as Indicators of Damage Mechanisms

One of the strengths of EBMs lies in their *shape functions*, which describe how the predicted damage probability varies with each feature independently. These curves can be interpreted in structural terms:

- The **monotonically increasing shape** for Energy or RMS suggests that higher global vibration levels correspond to higher likelihood of structural degradation; consistent with reduced stiffness or increased modal participation under damage.
- A **threshold-like pattern** in kurtosis indicates that, beyond a certain peak value, the probability of damage increases dramatically. This mirrors physical behavior where early stage damage produces minimal nonlinear

effects until crack propagation accelerates.

- Shape functions for cosine similarity typically show a **negative correlation**: a decrease in similarity to healthy baseline spectra implies an increasing deviation of modal characteristics, which is a widely accepted indicator of structural changes.
- For skewness, a **U shaped or asymmetric curve** may reflect different kinds of damage affecting the signal asymmetry (e.g., unilateral cracks causing directional amplification).

These interpretations allow structural engineers to understand not only *which* features matter, but also *why* they matter and *how* they relate to the underlying physical damage mechanisms.

5.2.1.3 Relevance to Railway Steel Bridges

In steel bridges, common damage modes include:

- fatigue cracking,
- joint loosening,
- stiffness loss at connections,
- local yielding under repeated loading.

These mechanisms directly affect the dynamic response captured in the vibration signals. By linking EBM outputs to these known behaviors; such as kurtosis spikes due to impact-like nonlinearities or frequency shifts reflected in cosine similarity deviations—the model provides actionable engineering insight.

Summary of Structural Interpretation Benefits

Integrating structural engineering reasoning into EBM interpretation offers several advantages:

- Supports trust and usability for engineers making safety decisions,
- Highlights physical meaning behind statistical features,
- Bridges the gap between data driven and physics-informed SHM,
- Enhances the scientific contribution of explainable ML in structural assessment.

Thus, EBMs not only serve as interpretable machine learning models but also as diagnostic tools that reveal how underlying structural behavior influences observed vibration responses.

5.2.2 Stacked EBM with Predicted Attributes as Features

In our architecture, we employ a two-stage Explainable Boosting Machine (EBM) setup to improve interpretability and predictive performance. The primary EBM as in Fig. 5.1 is trained on statistical features extracted from ambient vibration data. This model predicts intermediate structural attributes such as **Stress Force (SF)**, **BS** and **DI**. These predictions are not used as final output but are instead appended to the original feature set.

The concatenated data, including both handcrafted features and EBM derived predicted structural states, is then passed to a **secondary EBM**. This model is trained to make the final classification, benefitting from both the signal derived data and the learned insights from the primary EBM.

This stacked architecture allows the secondary model to refine its predictions by incorporating high-level features (for example, stress force) in addition to the other statistical features, while maintaining full model transparency and traceability at each stage.

Importantly, EBM maintain a high degree of interpretability by including only the most influential feature interactions, determined during model training. This selective approach allows the model to capture non linear dependencies without sacrificing transparency, making EBMs a compelling alternative to black-box models in practical SHM deployments.

The additive nature of the model means that the contribution of each feature and interaction can be visualized and understood independently, making EBMs highly suitable for applications where transparency is critical [106–108]. In the context of SHM, this allows practitioners to observe exactly how vibration-based features such as energy, kurtosis, or skewness influence the final damage prediction, and whether specific combinations of features interact to signal complex structural behaviors. Since the functions f_j and f_{jk} are learned using **boosted decision trees**, they remain piecewise constant and interpretable, while still capturing nonlinear relationships present in the data.

5.3 Results and Discussions

EBMs provided interpretable results and performed competitively in both training and test phases. The secondary EBM leveraged predicted outputs (for example, stress, scenario) to refine final predictions. Slight accuracy drops in the secondary EBM may be due to simulated data; real SHM data is expected to yield better results. Table 5.1 summarizes the performance of the proposed Explainable Boosting Machine (EBM) models for three critical structural health

monitoring (SHM) tasks: stress force prediction, bridge scenario classification, and damage intensity classification. The table includes results for training accuracy, test accuracy, and secondary EBM test accuracy (i.e., using a stacked EBM setup with enriched input features).

5.3.1 EBM Performance Evaluation Across SHM of Railway Steel Bridges

Stress Force Prediction: The EBM achieves a training accuracy of **94.32%** and a test accuracy of **89.50%**, indicating strong generalization. The secondary EBM slightly underperforms at **87.61%**, suggesting that the primary model sufficiently captures the input-output relationship, and that further stacking may not yield substantial benefit in this specific task.

Bridge Scenario Classification: This task shows near-perfect performance, with the primary EBM achieving **99.82%** training accuracy and **99.48%** test accuracy. The secondary EBM slightly improves the result to **99.52%**, demonstrating that the model is highly reliable in identifying spatial damage scenarios across bridge segments. This level of performance supports the effectiveness of EBMs in spatial localization of structural anomalies.

Damage Intensity Classification: Compared to the other two tasks, damage intensity prediction is more challenging. The primary EBM achieves a training accuracy of **81.45%**, which drops to **71.32%** on the test set, indicating potential overfitting or limited discriminative power in the features. The secondary EBM further drops to **67.55%**, likely due to compounded prediction noise or weak class separation in the augmented input. These results suggest that more robust feature extraction or alternative model architectures (for example, EBM-CNN hybrids) may be needed to better distinguish between closely spaced damage severity levels.

Table 5.1: Performance Results of Stress Force, Bridge Scenarios, and Damage Intensities with EBMs

Category	Train Accuracy	Test Accuracy	Secondary Test Accuracy
Stress Force	94.32	89.50	87.61
Bridge Scenario	99.82	99.48	99.52
Damage Intensities	81.45	71.32	67.55

The slightly lower test accuracy of the secondary EBM in Fig. 5.2, such as damage intensity classification, may be attributed to the use of simulated data rather than real-world measurements. Simulated SHM datasets, while useful for preliminary experiments and proof-of-concept validations, often lack the full variability, noise patterns, and sensor irregularities present in real-world

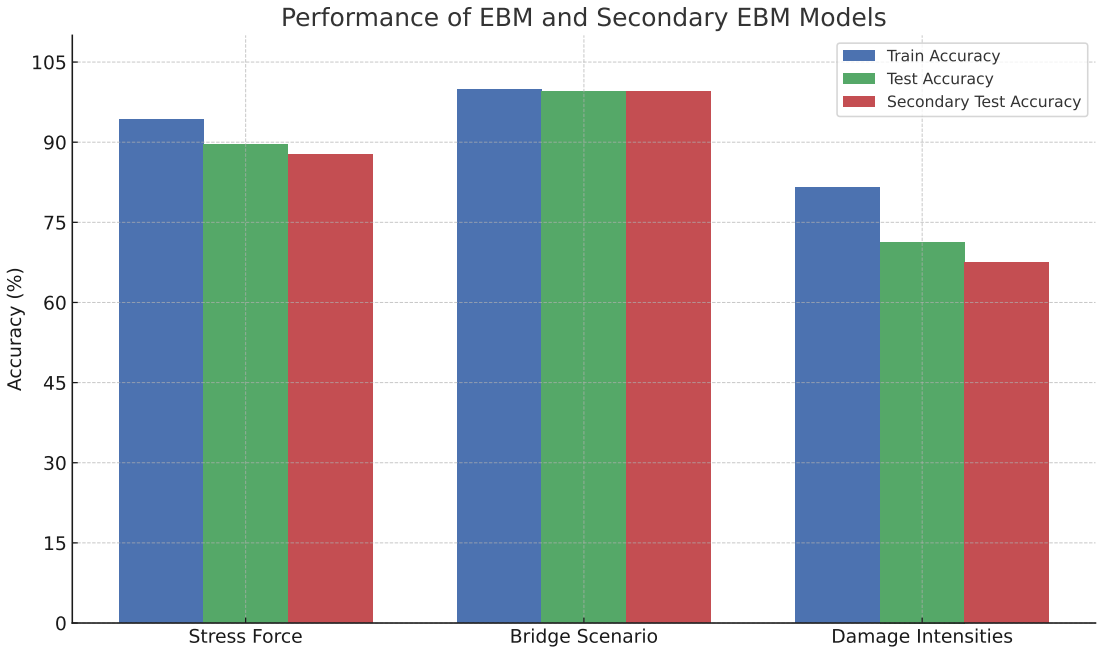


Figure 5.2: Comparison of EBM and Secondary EBM accuracy for Stress Force, Bridge Scenario, and Damage Intensity

bridge monitoring scenarios. As a result, the secondary model may not benefit as strongly from the predictive outputs of the primary EBM when both are trained on idealized data.

However, the stacked architecture is designed to leverage the complementary strengths of statistical features (for example, skewness, energy, cumulative sum of differences) and learned outputs from the primary EBM (for example, predicted stress force, bridge scenario, and damage intensity). In real deployment conditions, this hybrid input is expected to enhance decision making accuracy by enriching the feature space with both domain-knowledge-informed metrics and data driven predictions. Thus, while the current performance may be limited by dataset realism, the secondary EBM is anticipated to perform more effectively when trained and evaluated using actual field data collected from instrumented railway bridges.

While both models perform well in scenario classification, secondary EBM slightly improves performance for stress force but underperforms in damage intensity classification due to the complexity of the task as in Fig. 5.2.

Overall, EBMs demonstrate high performance in tasks where structural features show clear patterns, such as bridge scenario and stress force prediction. However, damage intensity classification remains a challenging problem, requiring further investigation into feature engineering, model stacking strategies, and possibly data augmentation to improve predictive robustness.

5.3.2 Class Wise EBM Performance Evaluation Across Stress Force For SHM of Railway Steel Bridges

Table 5.2: Class-wise Performance of Stress Force

Main Class	Sub Class	Precision	Recall	F1-Score
Stress Force	CL3	0.97	0.97	0.97
	CL4	0.87	0.91	0.89
	CL9	0.87	0.83	0.85
	CL11	1.00	1.00	1.00
	CL12	0.97	0.97	0.97
	CL19	0.84	0.79	0.82
	CL20	0.78	0.86	0.82
	CL39	0.82	0.79	0.80
	CL40	0.95	0.95	0.95

To evaluate the detailed performance of the EBM model on stress force classification, both a class-wise metric table (Table 5.2) and a bar chart visualization (Figure 5.3) are presented. This combined analysis provides both numerical accuracy and visual insights into the model's predictive behavior across individual stress force classes.

Table 5.2 reports precision, recall, and F1-score for each stress force class (CL3 through CL40). The results demonstrate that the model performs exceptionally well on several classes, notably CL3, CL11, CL12, and CL40, each achieving F1-scores near or at 0.97–1.00. These values indicate not only accurate predictions but also a high level of consistency in correctly identifying these specific stress states with minimal false positives or negatives.

To complement the table, Figure 5.3 visualizes the same metrics, enabling quick comparison across classes. The bar chart confirms that CL3, CL11, and CL40 are among the top-performing classes, represented by tall, consistent bars across all three metrics. In contrast, classes such as CL19, CL20, and CL39 exhibit comparatively lower recall and F1-scores, as reflected by shorter green and red bars in the visualization. These underperforming classes likely suffer from class imbalance or overlapping feature patterns, which may confuse the model.

Other classes, such as CL4, CL9, CL19, and CL20, show slightly lower scores,particularly in recall, which suggests that the model occasionally misses instances from these classes. For example, CL19 and CL20 show recall values of 0.79 and 0.86, respectively, implying that some actual samples from these classes were misclassified as neighboring classes, possibly due to overlapping feature distributions.

The F1-scores, which balance precision and recall, remain above 0.80 for all

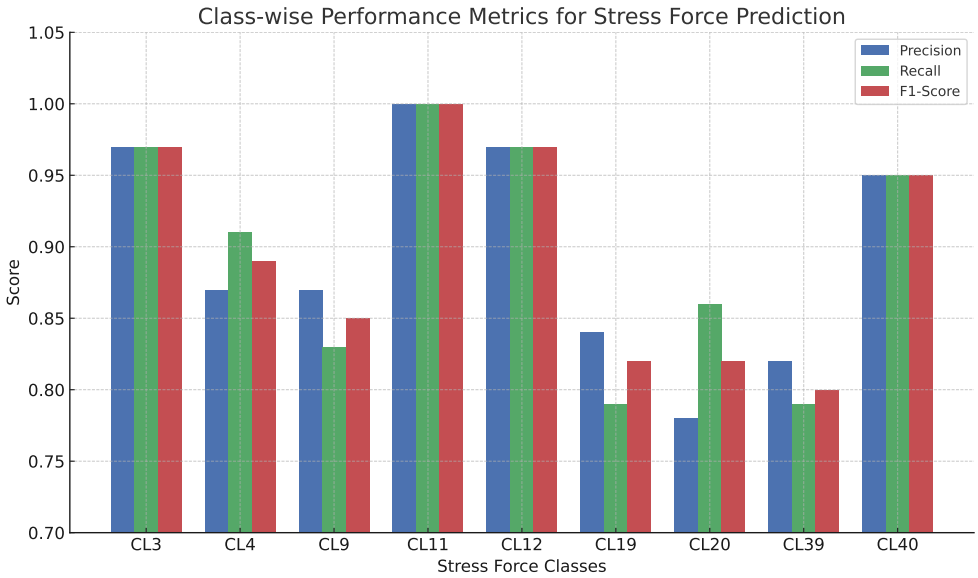


Figure 5.3: Class-wise Precision, Recall, and F1-Score of Stress Force

classes, affirming the overall robustness of the model across varying stress force conditions. These results suggest that the EBM effectively captures the signal patterns associated with stress levels, though further enhancement (for example, data augmentation or feature engineering) could improve class separation for borderline categories like CL19 and CL20.

This class-wise analysis highlights the strengths and limitations of the EBM model in capturing structural stress variations. It suggests that while the model is highly reliable for well separated classes, further refinement such as additional data, feature selection, or tailored augmentation, may be needed to boost performance on borderline or less distinct stress classes.

5.3.3 Class Wise EBM Performance Evaluation Across Bridge Scenario For SHM of Railway Steel Bridges

Table 5.3 and Figure 5.4 provide a class-wise analysis of the EBM model's performance on the bridge scenario classification task (S1-S7). The table reports high precision, recall, and F1-scores for all classes, with several reaching perfect scores of 1.00. The bar chart visually reinforces these findings, showing consistent and near identical bars across all metrics, particularly for scenarios S3 through S6.

This high level of consistency across all performance metrics indicates that the EBM model effectively captures spatial patterns in sensor features, enabling it to accurately distinguish between structural zones of the bridge. Notably, even scenarios with slight imbalance or overlapping signals (for example, S1, S2, S7) maintain strong scores (≥ 0.99), further validating the robustness of

Table 5.3: Class-wise Performance of Bridge Scenario

Main Class	Sub Class	Precision	Recall	F1-Score
Bridge Scenario	S1	0.99	0.99	0.99
	S2	0.99	0.99	0.99
	S3	0.99	1.00	1.00
	S4	1.00	1.00	1.00
	S5	1.00	1.00	1.00
	S6	0.99	1.00	1.00
	S7	1.00	0.99	0.99

Figure 5.4: Class-wise Precision, Recall, and F1-Score of Bridge Scenario

the EBM in this task.

The interpretability of EBMs allows for a clear understanding of how features such as cumulative sum of differences, cosine similarity, and skewness contribute to each scenario prediction. This makes the model especially useful for practical SHM applications, where accurate localization and transparency are critical for maintenance decision-making.

5.3.4 Class Wise EBM Performance Evaluation Across Damage Intensity For SHM of Railway Steel Bridges

To better visualize the class-wise predictive performance of the EBM model on damage intensity levels (1-90%), Table 5.4 is complemented with a bar chart shown in Figure 5.5. The figure plots precision, recall, and F1-score for each intensity class, highlighting relative performance across the damage scale.

The chart confirms that certain classes (for example, 1, 75, 90) achieve high performance across all metrics, while mid range and closely spaced intensities

Table 5.4: Class-wise Performance of Damage Intensity

Main Class	**Sub Class**	**Precision**	**Recall**	**F1-Score**
Damage Intensity	1	1.00	1.00	1.00
	2	0.67	0.75	0.71
	3	0.72	0.70	0.71
	4	0.79	0.63	0.70
	5	0.70	0.62	0.66
	6	0.63	0.69	0.66
	7	0.65	0.72	0.69
	8	0.64	0.69	0.66
	9	0.73	0.69	0.71
	10	0.64	0.69	0.67
	15	0.75	0.65	0.70
	20	0.69	0.78	0.74
	25	0.76	0.74	0.69
	30	0.71	0.68	0.69
	35	0.61	0.73	0.66
	40	0.80	0.65	0.72
	45	0.64	0.84	0.72
	50	0.70	0.68	0.69
	55	0.66	0.69	0.68
	60	0.72	0.76	0.74
	65	0.72	0.77	0.74
	70	0.83	0.67	0.74
	75	0.75	0.78	0.77
	80	0.65	0.60	0.62
	85	0.71	0.69	0.70
	90	0.78	0.76	0.77

(for example, 4-10, 35-45, 80) show notable performance drops. In particular, class 80 exhibits the lowest F1-score of 0.62, suggesting overlap with neighboring classes or insufficient discriminative features. The visual layout reinforces the observations in Table 5.4, making it easier to identify performance bottlenecks and areas requiring further model improvement or data refinement.

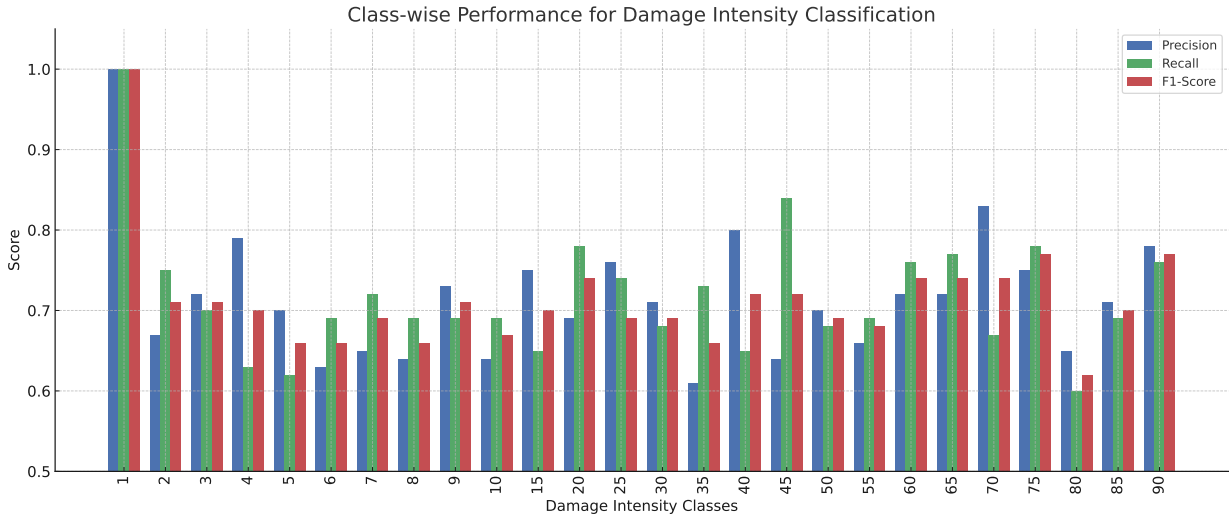


Figure 5.5: Class-wise precision, recall, and F1-scores for damage intensity classification (1-90%) using EBM.

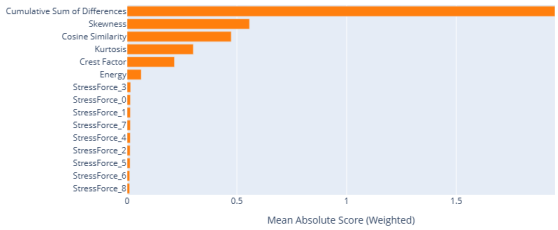
The visualization highlights consistency for high performing classes and identifies drop off zones for model refinement.

5.3.5 Global Feature Importance Analysis for EBMs and Secondary EBMs in SHM

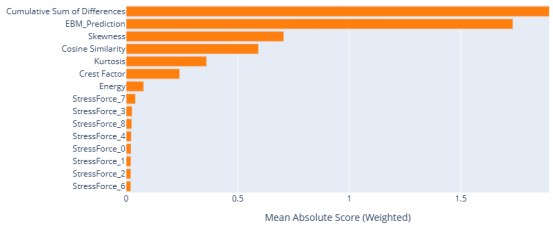
Figures 5.6a, 5.6c, and 5.6e show the global feature importances for primary Explainable Boosting Machines (EBMs), while Figures 5.6b, 5.6d, and 5.6f show their corresponding secondary EBMs. These models were applied to three key SHM tasks on a railway steel bridge: damage intensity classification, bridge scenario classification, and stress force estimation.

In the case of **damage intensity classification** (Figure 5.6a), the feature *Cumulative Sum of Differences* dominates the model, followed by *Skewness*, *Cosine Similarity*, and *Kurtosis*. This suggests that the temporal variation in the signal (captured by CSD) plays a significant role in identifying the extent of structural degradation. The secondary EBM (Figure 5.6b) still prioritizes CSD but shows an increased influence of the feature *EBM_Prediction*, indicating that combining statistical descriptors with primary model predictions improves interpretability and may assist in refining borderline cases.

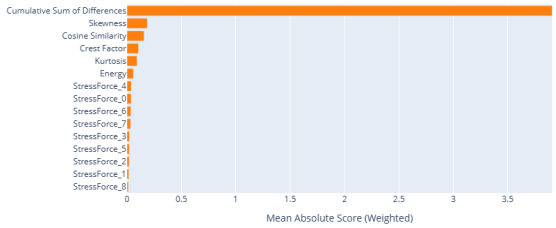
For **bridge scenario classification** (Figure 5.6c), features such as *Energy*,



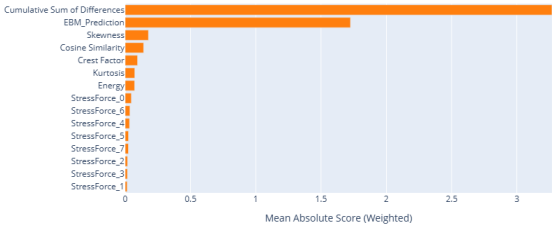
(a) EBM - Damage Intensity



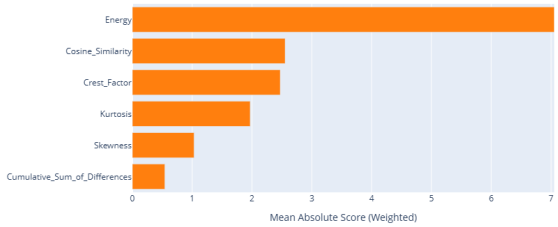
(b) Secondary EBM - Damage Intensity



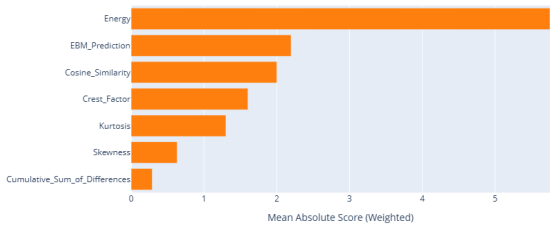
(c) EBM - Bridge Scenario



(d) Secondary EBM - Bridge Scenario



(e) EBM - Stress Force



(f) Secondary EBM - Stress Force

Figure 5.6: Global feature importance plots for EBM and secondary EBM models across three SHM tasks: damage intensity, bridge scenario, and stress force.

Cosine Similarity, and *Crest Factor* hold the most weight in the primary EBM. These features are associated with structural response magnitude and signal shape. In the secondary EBM (Figure 5.6d), *EBM_Prediction* and *Energy* are again highly ranked, showing that the stacked model successfully leverages the prior knowledge captured in the first stage while retaining meaningful contributions from raw statistical features.

For **stress force prediction** (Figure 5.6e), the primary EBM reveals high importance for a broader range of features, including various stress force components (for example, StressForce_0 to StressForce_8), though CSD remains the top feature. In contrast, the secondary EBM (Figure 5.6f) once again ranks *EBM_Prediction* alongside physical features like *Energy* and *Cosine Similarity*, highlighting the additive value of prediction-informed features.

In summary, the feature importance visualizations validate that EBMs can effectively extract signal-driven indicators of damage, while secondary EBMs enhance this interpretability by fusing predictions with original inputs. This hybrid architecture supports better decision-making for real-time SHM systems deployed in railway infrastructure.

5.4 Summary

This chapter presented the design, implementation, and evaluation of Explainable Boosting Machines (EBMs) for Structural Health Monitoring SHM of railway steel bridges. EBMs were introduced as an interpretable machine learning approach based on Generalized Additive Models GAMs, capable of learning smooth, univariate shape functions and selected pairwise interactions using gradient-boosted decision trees. Their additive and transparent nature makes them highly suitable for SHM applications where explainability and diagnostic traceability are as important as predictive accuracy.

Three key SHM tasks were addressed: stress force estimation, bridge scenario classification, and damage intensity classification. Primary EBM models trained on statistical time-domain features (for example, cumulative sum of differences, energy, kurtosis, skewness, cosine similarity) achieved high accuracy, particularly for bridge scenario classification, with near-perfect precision and recall (≥ 0.99) across all seven zones. Stress force prediction also showed strong performance, with accuracies exceeding 89% on test data. Damage intensity classification, while more challenging due to overlapping feature distributions and fine-grained class separation, still yielded satisfactory performance, especially after data augmentation.

To enhance generalization and leverage model-inferred knowledge, a stacked

architecture was proposed, incorporating a *secondary EBM* fed with both original features and predictions from the primary EBM. This setup allowed the second-stage model to refine predictions using higher-order learned representations. Feature importance analyses demonstrated that EBM predictions, when used as input, significantly contributed to secondary model performance; especially in stress force and damage intensity tasks.

Overall, this chapter validates the suitability of EBMs and their stacked extensions for SHM tasks in railway bridge systems. Their interpretability, consistent accuracy, and visual feature explanations make them an excellent candidate for practical deployment in high stake railway steel bridge monitoring.

Chapter 6

Structural Health Monitoring of Railway Steel Bridges Using Machine Learning Techniques

"Everything we have of value as human beings - as a civilization - is the result of our intelligence."

— Stuart J. Russell

Structural health monitoring SHM of bridges has become an increasingly important focus. What prompted the research to explore this topic more thoroughly is the recognition that structures, during their life, are subject to aging or degradation phenomena. These phenomena may result in a loss of structural performance,which ultimately leads to the inability of the structure to

fulfill its intended functional requirements. The strategic importance of bridges as critical elements for mobility and logistic infrastructure makes their monitoring even more essential than ordinary structures. Any potential compromise of their integrity could lead to significant disruptions in transportation systems and severe economic and social impacts. The assumption at the base of damage detection is that when any form of damage or degradation occurs, the structural parameters change, and consequently, the response of the structure is altered as well: the objective of monitoring a structure is, therefore, to detect this behavioral change and to investigate its causes. SHM involves continuously evaluating the bridge's condition, assessing the progression of damage, and timely maintenance interventions.

In this context, a structural monitoring system can be helpful in two different phases of a structure's life: during maintenance, to focus interventions in parts of the structure where anomalies emerge, and during emergencies, as a management tool for selecting intervention priorities.

Damage detection techniques can be classified into four stages, as outlined by Rytter [25] and further developed over time: identification, localization, quantification of damage intensity, and prognosis. In line with these levels, many techniques have been developed over time. Recently, the development of artificial intelligence techniques has driven the research community's interest in applying machine learning algorithms to solve damage identification and classification problems. Most use non-destructive methods that involve sensor data acquisition (accelerometers, crack gauges, etc.) placed at significant locations of the structure [26]. Neural network-based techniques for bridge monitoring [85, 109–114], including our previous work about Bridge Health Monitoring using RBF-Neural Network [85, 115] method for Bridge SHM, prove to be more cost-effect, faster, and more accurate. Most of these works focus on time-domain or frequency-domain-based implementations.

6.1 Overview

Railway steel bridges are critical infrastructural assets that endure dynamic loads, environmental effects, and structural aging over decades. Ensuring their operational safety requires timely detection of damage, stress anomalies, and early degradation patterns. Conventional inspection methods, while effective, are time-consuming, cost-intensive, and prone to human subjectivity. This has led to a growing interest in automated Structural Health Monitoring (SHM) systems powered by data-driven machine learning (ML) techniques.

This chapter presents a comprehensive study on the application of machine

learning algorithms for anomaly detection and classification in SHM of railway steel bridges. It focuses on the full data pipeline—from sensor-based signal acquisition to feature extraction, classification, and alert generation.

6.2 Introduction

Today, engineers analyze and monitor structures in real-time to identify anomalies and early detection of damages. They also ensure that they take proactive maintenance measures and extend the structural lifespan of such bridges.

Anomaly detection [109], damage localization [111, 116, 117], damage classification [112], monitoring response, and alert generation based on structural factors are becoming easy to investigate steel bridges with machine learning algorithms. However, a lot of work is still needed to address specific challenges. These challenges include technological difficulties, environmental factors, maintenance cost, socioeconomic balance, and risk of safety [118].

This study covers a damage identification analysis of railway steel bridges using a Deep Neural Network with different parametric tuning and the effect of data slicing on performance matrices. The objective is to use features extracted from vibrational signals to train the networks to identify and classify structural damage. The figure in the abstract illustrates the general concept of the suggested system, with a perception layer and a reasoning layer for handling complex events using different logic and ML models for damage detection, localization, and classification.

It is important to note that, when dealing with complex structures such as those in civil engineering, the damage classification phase (i.e., the Rytter scale levels above the first) necessarily requires the development of a mathematical model of the structure itself. This model is essential for estimating structural behavior under potential damage scenarios.

The proposed approach uses the structure's physics-based ROM to generate a dataset of expected sensor measurements corresponding to various asset states with damage at different locations and intensities. This dataset is then used to train a classifier capable of processing near-real-time sensor data and estimating the asset's underlying state. The methodology involves an initial Data Dimensionality Reduction DDR step, where accelerometric time series data are processed to form block Toeplitz correlation matrices. These matrices are obtained from the block Hankel matrix of output data, the so-called subspace matrix, in the Data-Driven Stochastic Subspace Identification DD-SSI technique [27], widely employed in civil engineering for Operational Modal Analysis OMA. Block Toeplitz matrices are then used for feature extraction

and then with neural networks for anomaly detection and damage classification. The classification identifies both the location and the intensity of the damage.

This Chapter 6 is organized as follows; Section II explains introduction in this domain; Section III elaborates context of applying Machine Learning in SHMin context of Railway Steel Bridges; Section IV explains some background studies and previous research carried out in this domain. It also outlines the objectives of the present work and how they have been pursued; Section V elaborates the research questions that are addressed in this chapter. Section VI presents the research methodology, and Section VII presents the observed results. The paper concludes with a discussion about future directions.

6.3 Machine Learning in SHM of Railway Steel Bridges

Machine learning ML has become an integral component of modern Structural Health Monitoring SHM systems, particularly for railway steel bridges, where the ability to detect anomalies, assess structural damage, and predict future failures is essential for operational safety and maintenance efficiency.

ML approaches in SHM are broadly categorized into **supervised** and **unsupervised** learning techniques. **Supervised learning** relies on labeled datasets and is typically used for tasks such as damage classification, bridge scenario identification, and stress force prediction. Common supervised algorithms include Support Vector Machines(SVM), k-Nearest Neighbors(KNN), Random Forests(RF), Artificial Neural Networks(ANN), and Explainable Boosting Machines EBMs. These models learn from historical data where the output (for example, type or severity of damage) is known, enabling them to generalize to new observations with high accuracy.

On the other hand, **unsupervised learning** is employed when labeled data is unavailable or limited. Techniques such as Principal Component Analysis Principal Component Analysis (PCA), clustering, and autoencoders are used to discover hidden patterns or anomalies in the data. These methods are particularly effective in novelty detection and long-term trend analysis, where the objective is to detect unexpected deviations from normal structural behavior.

In SHM of railway bridges, where real-time response, robustness, and interpretability are critical, the choice of ML technique depends on the application context, the nature of sensor data, and the availability of labeled datasets.

6.4 Related Work

In recent years, non-destructive structural damage detection techniques based on structural dynamic characteristics, Genetic algorithms, and ANN have been developed. Inspired by these methods, AI experts were interested in applying Deep Neural Networks in bridge health monitoring.

Thanh et al. [116] analyzed the non-destructive SHM of bridges using the fusion of Particle Swarm Optimization and Support Vector Machines (PSO-SVM). PSO-SVM helped eliminate the redundant input parameters for adequate classification of damage localization. Similarly, Svendsen et al. [119] proposed a supervised learning approach based on Support Vector Machines (SVM), incorporating auto-regressive feature extraction methods to detect structural conditions in steel bridges.

Elisa et al. [113] used ANN for anomaly detection and damage localization in time series data for structural health monitoring. They focused on recording the axle load signal every time a train passed over the bridge. Khodabendelou et al. [120] designed a feedforward CNN to evaluate bridge health using vibration response, Abdeljaber et al. [121] carried out optimization for non-parametric structural damage detection for performance evaluation using a CNN with a single learning block for training.

In [122], Huile et al. also examined railway bridges for structural health monitoring and damage detection. They analyzed the railway bridge girder for Chinese high-speed rail, using a Bayesian Deep Learning method to account for dynamic speed loading. Abdelkader et al. [123] investigated damage detection and localization with real-time vibration data using simple CNN. This model can automatically extract optimal damage-sensitive features from raw acceleration signals. Another study by Giasi et al. [124] adopted CNN for damage classification on railway steel bridges using various damage parameters. They also managed to find accuracy using both accurate data and Finite Element model FEM data. Here, we provides an overview of notable studies applying machine learning techniques in Structural Health Monitoring (SHM).

Svendsen et al. [119] introduced a supervised approach using multiple models including KNN, SVM, Gaussian Naive Bayes, and Random Forest, with auto-regressive feature extraction and Mahalanobis Squared Distance for data normalization. Performance was evaluated using ROC curves and confusion matrices. Similarly, Ali et al. [125] used Artificial Neural Networks (ANNs) with SVD features and Root Mean Square RMS normalization, targeting damage intensity prediction.

Armijo and Zamora-Sánchez [19] proposed an Internet of Things (IoT)-

enabled digital twin framework for railway bridge SHM using low-cost wireless accelerometers and machine learning. Their architecture integrates edge computing for on-site signal processing and cloud-based analytics for long-term storage and anomaly detection. Vibration signals collected over two years were analyzed in the frequency domain to detect abnormal spectral peaks linked to structural changes. The system demonstrated real-time, automated bridge damage detection and scalability for deployment across large railway networks.

In contrast, D'Andrea et al. [126] proposed an unsupervised method based on the detection of changes in dynamic and static responses using a Least Squares fitting approach. Esposito et al. [127] employed an unsupervised SVD-based classification method combined with Independent Component Analysis Independent Component Analysis (ICA) and Euclidean norm normalization, evaluated via novelty detection accuracy and outlier analysis.

Liu et al. [110] combined Model Order Reduction (MOR) with a fully convolutional neural network CNN, using PCA for feature extraction and reporting accuracy and signal-to-noise ratio (SNR) ratio as key metrics. Mahdavi et al. [128] utilized Linear Regression and Robust PCA, normalized by Mahalanobis distance and chi-square distribution, with ROC analysis for validation.

Tak et al. [116] proposed a hybrid POS-SVM approach using Genetic Algorithm (GA)-based feature selection and Root Mean Square Error (RMSE) as the primary metric. Elnashai et al. [113] used an ANN model with standard statistical normalization (mean and standard deviation) and evaluated performance using ROC, true positives, and false positives. Lastly, Amiri et al. [124] leveraged a CNN with t-distributed Stochastic Neighbor Embedding (t-SNE) and Gradient-weighted Class Activation Mapping (Grad-CAM) for feature visualization, adding Gaussian noise for augmentation and reporting overall accuracy.

This collection of studies highlights the diversity in SHM methodologies, including supervised and unsupervised approaches, a variety of feature extraction and normalization techniques, and a broad range of evaluation metrics.

6.5 Research Questions

As this study aims to evaluate the performance of a machine learning approaches for the classification of structural damage in railway infrastructures. The issue of damage classification has been and continues to be the subject of extensive research in civil engineering due to the complexity of predicting structural behavior in the presence of damage.

As mentioned in the previous section, it is impossible to produce a realistic

damage condition reversibly when working with existing structures. Therefore, implementing a physical-mathematical model to create simulated data is crucial when characterizing structure damage. This study adopts a ROM model of a railway bridge to produce the data necessary for training and testing classical machine learning algorithms, and a customized CNN for damage classification.

The CNN is designed to answer the following research questions:

- Is it possible to use the dynamic subspace matrix to extract features useful for damage classification through ML techniques?
- Can window size for feature extraction affect anomaly detection and damage(bridge scenario) localization predictions?
- What would be the effect of different optimizers on the accuracy of the bridge section and damage intensities?
- What would be the effect of data augmentation in the final results?

These research questions are explored in subsequent sections.

6.6 Methodology

This section describes a detailed methodology and implementation of a Convolutional Neural Network framework. This framework is used to detect multiclass based behavioral anomalies in different sections of a steel bridge and to monitor its health, assessing the severity of the damage at various intensity levels.

This section also introduces the features extracted from the simulated acceleration data used in the CNN training. Furthermore, it also discusses the machine learning algorithm, its evaluation using standard data and augmented data, fine-tuning of Deep Neural Network DNN parameters, model parameters, and finally, the evaluation criteria for damage localization and damage level classification. Therefore, this section is divided into four subsections, namely, i) Basic Framework, ii) Feature Engineering, iii) Deep Neural Network algorithm, and finally, iv) Evaluation criteria.

6.6.1 Basic Framework

The damage characterization procedure described in the previous sections is part of a broader SHM system framework. The proposed framework, schematically represented in Fig.6.1, involves comparing features extracted from different states of the structure, both real and simulated. The overall anomaly detection process is performed by comparing the features of the signals from the structure in its current state with a baseline of features corresponding to a reference state. The system triggers alarms or alerts if the difference between

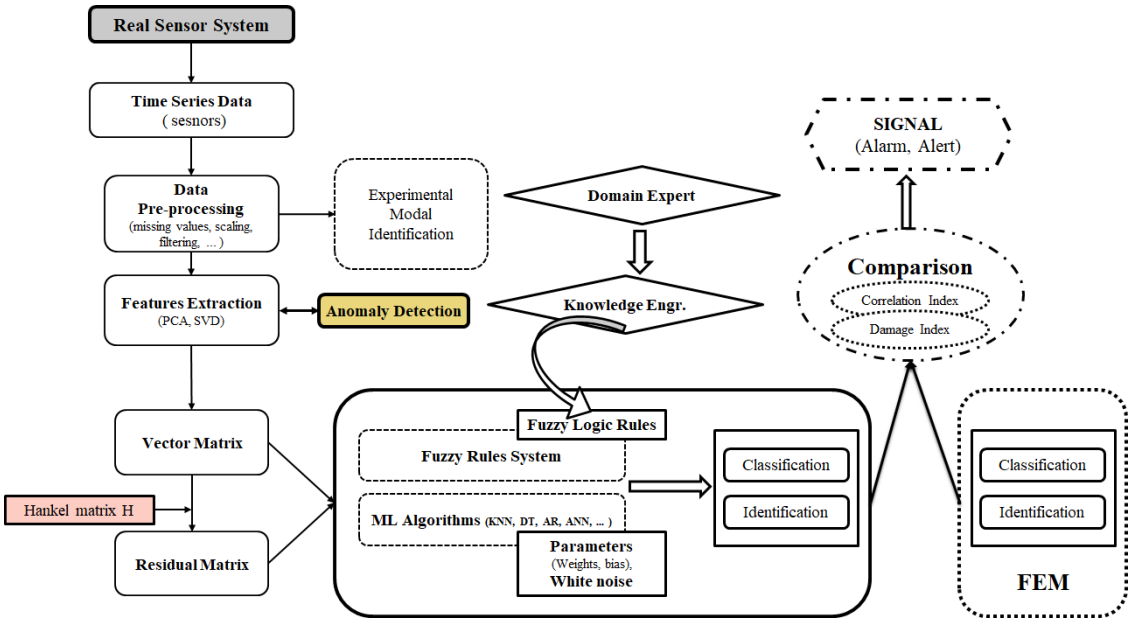


Figure 6.1: Deployment Framework of Railway Bridge Health Monitoring System

the features exceeds predefined threshold values determined by domain experts. The process is data-driven, as the comparison is made between the features of two experimental states. Physics-based modeling is required in the diagnostic phase when investigating the location and intensity of the anomaly; after that, its presence has been verified. In this phase, the current state of the structure is classified by comparing its features with those from a trained neural classifier. Based on the information provided during the training phase, the network predicts classes relative to the damage scenario and intensity; see Fig. 6.2.

The same framework can be used in both this Chapter 6 for time-domain and Chapter 7 for Frequency Domain, even if the this Chapter only focuses on time-domain analysis . Thus, the system implementation includes a Fuzzy Logic Reasoning Layer with alert generation, DNN for classification, and damage localization based on features extracted from time series. Bridge Scenarios and Damage Intensities are prediction classes produced by the neural perception layer. Machine learning-based algorithms with standard data and augmented data are used for predictions. For bridge scenarios, seven classes are referred to that a bridge can be categorized for analysis or monitoring purposes. Similarly, twenty-six classes are called damage intensities in the range of 0-100%. Each damage intensity is calculated by applying specific damage to the model, as described in the previous Section. The damage intensities are a set of discrete intervals. Domain experts then converted them into compact classes with labels of low-, medium-, high-, and extreme damage intensities.

Before feeding the matrix data to a DNN, it is sliced to 100%, 50%, 25%, and 12.5%. This slicing is carried out to check the effect of such slicing on original

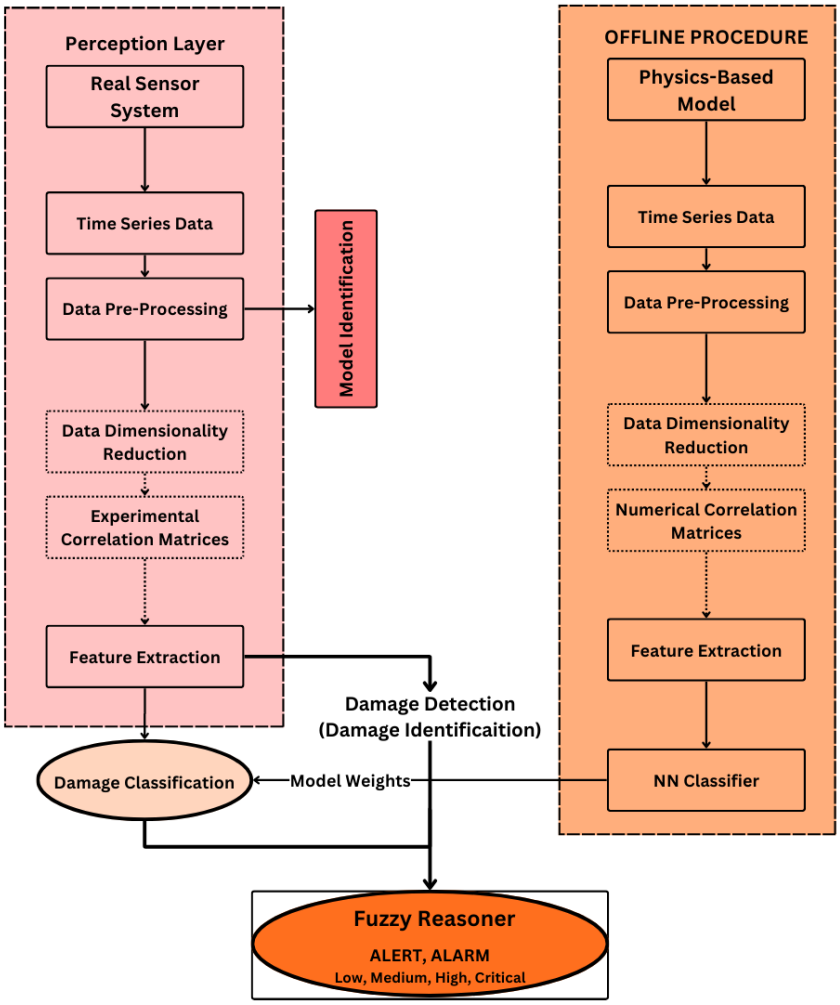


Figure 6.2: Training Process Schema for Damage Classification

data regarding prediction accuracy. The other purpose of this way of slicing is to maximize the physical data of the bridge as much as possible. To enhance data availability, the original 320×320 (100%) dataset was first flattened into a one-dimensional sequence and then restructured into matrices of sizes 180×180(50%), 80×80(25%), and 40×40(12.5%) by sub-sampling so that the transformation preserves the temporal and spatial dependencies within the data by leveraging its inherent spatial continuity, ensuring that meaningful structural relationships remain intact. Consequently, the model can effectively capture local and global patterns, essential for an accurate analysis. By increasing the size of dataset through matrices obtained with structured sub-sampling, this approach enhances the model's generalization ability, improving its robustness and reliability in real-world applications. These sliced data are then converted into features as explained in Subsection 6.6.2 .

6.6.2 Feature Engineering

Accelerometer sensors are installed at different locations on the bridge. There are Mono-axial, Bi-axial and Tri-axial accelerometer sensors. Mono-axial sensor data consist of X-axes, Bi-axial accelerometer data consist of X-, and Y-axes and Tri-axial accelerometer data consist of X-, Y-, and Z-axes (vertical, lateral, and sagittal axes) [129]. These sensors are used to monitor the train's activity over the bridge. The train activity on the bridge helps monitor the structural health of the bridge.

Data received from the accelerometer sensor are in raw form. Data received in the time domain are then converted to features. In addition, to clean the data, the preprocessing step is performed. This preprocessing step is applied to obtain valuable ambient features, which are later fed to machine learning models for anomaly detection, damage localization, and predictions. The cumulative sum of differences [130] Cosine similarity (CS) [55], Skewness [131,132], Kurtosis [131,132], Energy (E) [133], and Crest factor (CF) [134] etc. are the calculated ambient features.

These selected features are then used as inputs to machine learning models for damage detection, localization, and prediction tasks. The detailed mathematical definitions and interpretability of each feature are provided in Chapter 3, Section *3.4.1.*

Principal Component Analysis (PCA) as discussed in subsection 6.6.2.1 is then applied for feature selection. Five features are selected with PCA; other features are discarded, which have low correlation values.

6.6.2.1 Principal Component Analysis (PCA)

Principal Component Analysis PCA is a widely used unsupervised dimensionality reduction technique that transforms a set of possibly correlated features into a set of linearly uncorrelated variables called **principal components**. The main objective of PCA is to reduce the dimensionality of the dataset while retaining as much variance (information) as possible.

Given a dataset with n observations and p features, PCA identifies a new orthogonal coordinate system where:

- The first principal component captures the maximum variance in the data.
- The second principal component captures the maximum remaining variance, orthogonal to the first, and so on.

Mathematically, PCA solves the eigenvalue decomposition of the covariance matrix of the standardized data. If $\mathbf{X}$ is the zero-centered data matrix, then:

$$\mathbf{C} = \frac{1}{n-1}\mathbf{X}^\top \mathbf{X} \tag{6.1}$$

$$\mathbf{C}\mathbf{v}_i = \lambda_i \mathbf{v}_i \tag{6.2}$$

where $\mathbf{C}$ is the covariance matrix, $\mathbf{v}_i$ are the eigenvectors (principal components), and λ_i are the eigenvalues indicating the amount of variance explained by each component.

6.6.2.2 PCA in SHM Feature Selection

In this work, PCA was used to reduce the dimensionality of the extracted ambient feature set before feeding it into the classification models. Out of the full set of extracted features (for example, CS, CSD, kurtosis, skewness, energy, crest factor(CF)), only the top five components with the highest variance contributions were selected. This step helped:

- Eliminate redundant or weakly informative features,
- Reduce computational cost and overfitting,
- Improve classifier performance and generalization.

The selected principal components preserve the most critical structural characteristics needed for detecting anomalies and classifying damage levels in the SHM dataset. The variance retained by each principal component is shown in the scree plot in Figure 6.3, which helps visualize the trade-off between dimensionality and information retention.

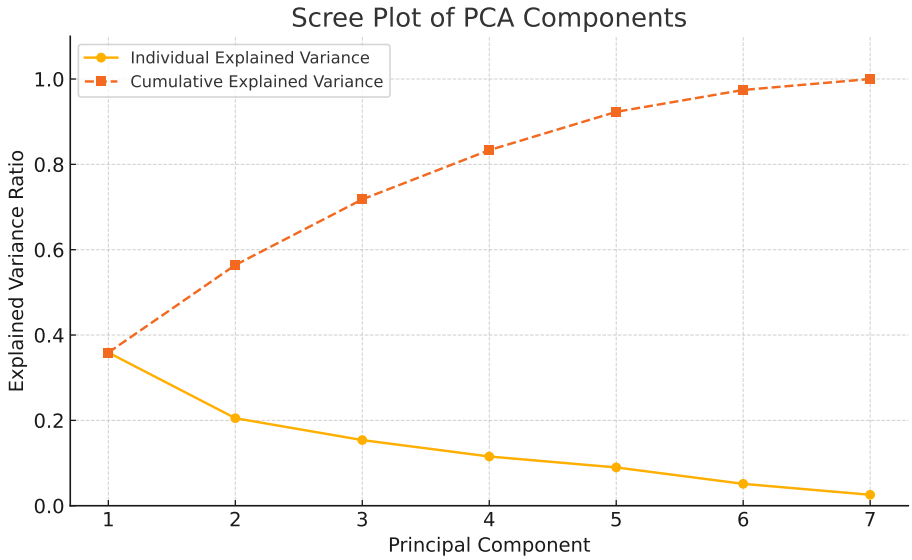


Figure 6.3: Scree plot showing the explained and cumulative variance ratio of PCA components used for SHM feature selection.

6.6.3 Classical Machine Learning Models

Before the rise of neural network models, traditional machine learning models play an essential role in foundational methods. These models, KNN [135], SVM [136], Decision Tree (DT) [137], RF [138], Ensemble Learning [139], and MLP [140, 141], are widely used for supervised learning and unsupervised learning tasks for classification, regression, predictions, and clustering.

6.6.4 Deep Neural Network Model

It is a special class of neural network NN model [142] that is frequently used and has shown good performance in computer vision and image processing tasks, such as image processing, object detection, image segmentation and localization [143].

CNN generally consists of several layers of digital neurons comprising a Deep NN architecture: Fully Connected layer Fully Connected (FC), pooling layer, activation function, batch normalization, dropout, and softmax (classification layer). Several different parameters can be optimized in each layer, and each layer performs a specific task with input data from the previous layer. The Conv layer has a set of filters applied to the input data to extract features. Equation (6.3) shows the essential convolutional operation in the CNN model.

$$y_j = \sum_{i=1}^{n} w_{ij} x_i + b_j \quad \text{for } j = 1, 2, \ldots, m \tag{6.3}$$

Where, $\mathbf{x} = [x_1, x_2, ..., x_n]$ is input vector of size n, $\mathbf{y} = [y_1, y_2, ..., y_m]$ is input vector of size m. w_{ij} is weight connecting input i to output j. Whereas, b_j is the bias for j^{th} output and y_j is j^{th} output. In matrix form,

$$\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{b} \tag{6.4}$$

where, $\mathbf{W}$ is weight matrix of size $m * n$ and $\mathbf{b}$ is bias vector of size m

The implemented model is a Deep Neural Network DNN, as shown in Fig.6.4.

This DNN has ambient features as input, multiple hidden layers, and softmax output for seven-class and six-class predictions. Equation (6.5) shows our linear neural network's essential 1D convolutional operation. Let us discuss the internal architectural breakdown of classified DNN.

$$y(t) = \sum_{i=0}^{k-1} x(t \cdot s + i) \cdot w(i) \tag{6.5}$$

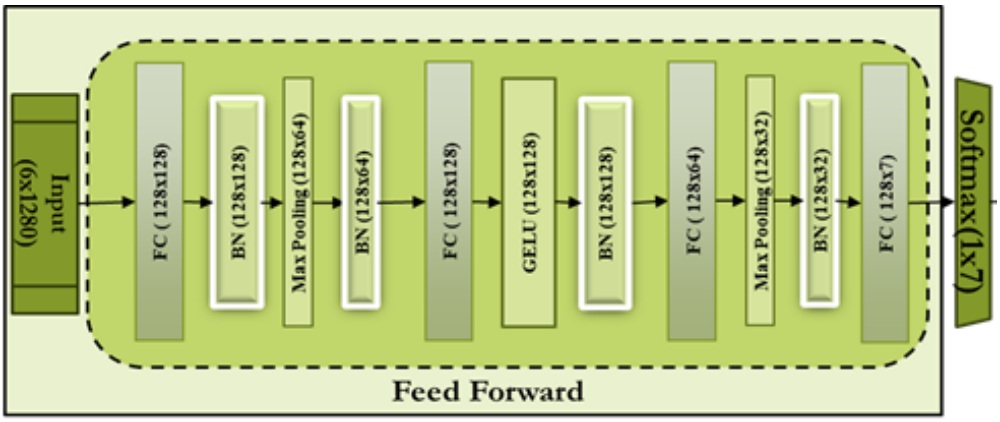


Figure 6.4: D-Net Architecture For Our Bridge Health Monitoring System

Where t is the output index (corresponding to the window position on the input), s is the stride, which determines how much the window shifts at each step. $x(t \cdot s + i)$ is input data within the window and $w(i)$ kernel weights.

Architecture: The model consists of fully connected (linear) layers. There are no convolution layers characteristic of convolution neural networks (CNNs). Table 6.1 summarizes our deep neural network's hyper parameters and architecture.

Table 6.1: Architecture And Hyperparameter of The Proposed Model

Category	Parameter	Value/Description
Hyperparameter	Learning Rate	0.01, 0.001
	Batch Size	128, 264, 512
	Epochs	1000
	Optimizer	Adam, SGD
	Loss Function	Cross-Entropy Loss
	Input Shape	(6, 128)
Model Architecture	Fully Connected Layers	4
	Activation Function	ReLU, GELU
	Output Layer	Softmax (10 & 4 classes)

1-Dimensional Data Handling: As feed input in (Rx7) and (Rx4) for the prediction of 7-class and 4-class, respectively. The model appears to operate on 1-dimensional data given the presence of 1-D Max-pooling.

Activation Functions: The model uses activation functions such as Leaky-ReLU and Gaussian Error Linear Unit (GELU), which are commonly used in DNNs. GELU are widely used in DNNs to introduce non-linearity into the model, enabling it to learn complex relationships in the data.

The Gaussian Error Linear Unit (GELU) activation function is defined as:

$$\text{GELU}(x) = x \cdot \Phi(x) \tag{6.6}$$

Where $\Phi(x)$ is the cumulative distribution function (CDF) of the standard Gaussian distribution.

$$\Phi(x) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right] \tag{6.7}$$

with approximation defined by:

$$\text{GELU}(x) \approx \frac{x}{2}\left[1 + \tanh\left(\sqrt{\frac{2}{\pi}}\left(x + 0.044715x^3\right)\right)\right] \tag{6.8}$$

Gaussian Error Linear Unit(GELU) activation introduces smoothness and probabilistic behaviour to neural networks with erf(x) error function, defined as:

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt \tag{6.9}$$

The GELU activation function determines the output by weighing the input x with the probability that it is greater than a random variable drawn from a standard Gaussian distribution. It introduces smooth non-linearity and is particularly effective in tasks involving transformers and attention mechanisms. The cumulative distribution function $\Phi(x)$ ensures that the output transitions smoothly between linear and nonlinear regions. The approximation simplifies computation while retaining the core probabilistic behaviour of the original GELU. In addition to GELU, we induced Leaky-ReLU in our fully connected neural network. Leaky ReLU is a standard Rectified Linear Unit (ReLU) activation function variation that allows a slight, non-zero gradient when the input is negative. This helps avoid issues such as the dying ReLU problem, where neurons become inactive during training.

The Leaky Rectified Linear Unit (Leaky ReLU) is an activation function that introduces a slight slope for negative inputs, helping to prevent the dying ReLU problem. It is defined as:

$$\text{Leaky ReLU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha x, & \text{if } x \leq 0 \end{cases} \tag{6.10}$$

Here, α is a small positive constant, typically $\alpha = 0.01$. For $x > 0$, the function behaves as standard ReLU.

$$\text{Leaky ReLU}(x) = x \tag{6.11}$$

For $x \leq 0$, the function has a small slope controlled by

$$\alpha : \text{Leaky ReLU}(x) = \alpha x \tag{6.12}$$

The main advantage of Leaky ReLU is that it avoids the issue of neurons becoming inactive during training by allowing a slight gradient even for negative values of x. Our neural network model used a GELU and a Leaky-ReLU after the second and third fully connected layers.

Batch Normalization: While batch normalisation is also used in CNNs, its presence does not necessarily indicate a CNN, especially considering the absence of convolution layers. Batch normalisation is also used to improve training stability and convergence. Batch normalisation is commonly used to normalise the activation of each layer, leading to faster training and better generalisation. We used four batch normalisation layers, each preceding a fully connected layer.

Output Layer: The model ends with a softmax activation, typical in DNNs, for multi-class classification tasks.

Softmax function converts a vector of numbers into a probability distribution. It is commonly used in machine learning, particularly in the output layer of a neural network used for classification tasks.

The softmax function for a vector $\mathbf{z} = [\ z_1, z_2, \ldots, z_n]$ is defined as:

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \tag{6.13}$$

where, $\sigma(\mathbf{z})_i$ is the i-th component of the output vector. z_i is the i-th component of the input vector $\mathbf{z}$. n is the number of elements in the input vector $\mathbf{z}$. And e is the base of the natural logarithm. As we deal with damage intensities and bridge scenarios, we have 26 output softmax for damage intensities and 7 class softmax output for bridge scenarios.

In conclusion, the provided DNN model suits required tasks such as Steel Bridge Damage detection and classification, making up the **perception layer** sub-system in the overall bridge monitoring software architecture.

6.7 Results and Discussion

ML models (SVM, MLP, RF, KNN, Ensemble (KNN, RF, and SVM), and DNN) were implemented using Pytorch 2.0 and Python 3.9 for programming purposes using the NVIDIA 940MX GPU. As a model optimization kit, a Deep Learning (DL) framework, the back-end for detecting anoma-

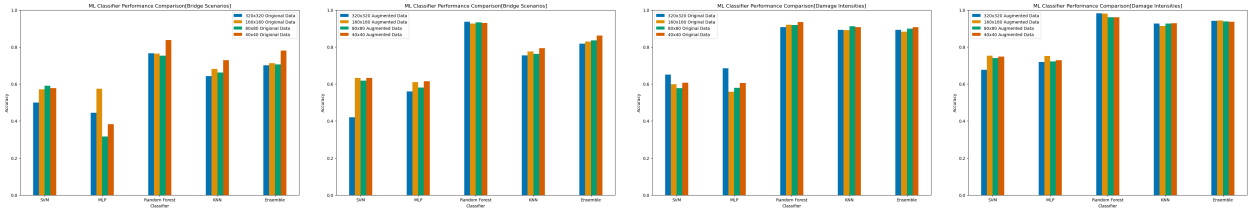
lies and classifications, is used to prepare, train, and optimize the model. All experiments were performed on the Jupyter Notebook platform using an NVIDIA Graphics Processing Unit (GPU) with 16 GB memory.

6.7.1 Bridge Scenarios Predictions

The model's performance was assessed across the dataset explained above. The dataset was divided into training and testing using the 80%-20% ratio, respectively. We initially used basic machine learning algorithms for performance evaluations, including KNN, SVM, Random Forest, MLP, and Ensemble of KNN, RF, and SVM with majority rule voting. Later, we also developed our own DNN model for performance evaluation. Detailed performance evaluations based on machine learning and DNN are discussed below.

6.7.1.1 Traditional Machine Learning Algorithms

The performance accuracy was measured using classical machine learning algorithms for the seven bridge scenarios described above. These include SVM, KNN, RF, Ensemble (KNN, RF & SVM), and MLP. The model accuracy was calculated with different matrix sizes, i.e. 320x320(100%), 160x160(50%), 80x80(25%), and 40x40(12.5%), to analyze better classification results. The result in Fig.6.5a and Fig.6.5b shows the performance of these ML models. From these figures, we observe that for each ML algorithm, a 40x40-sized matrix for the calculated feature outperforms. Among the algorithms, Random Forest outperforms in the Bridge Scenario classification task.



(a) Bridge Scenario for Original Data (b) Bridge Scenario for Augmented Data (c) Damage Intensities for Original Data (d) Damage Intensities for Augmented Data

Figure 6.5: Performance Comparison With Bridge Scenario (a), (b) and Damage Intensities (c), (d) Data

Specifically, augmented data improves performance over the original ambient data. Across classifiers, the accuracy increases with augmented data, with a notable improvement seen in the MLP, Random Forest, and Ensemble models. Random Forest achieves the highest accuracy with original and augmented ambient data. Inversely, SVM and MLP show high variance as their results are more affected by changes in the dataset. In the original ambient dataset,

SVM performs relatively well, while MLP has a lower accuracy. After applying augmentation in the ambient dataset, MLP improves significantly, indicating that MLP benefits more from data augmentation than other models.

Considering the impact of matrix size during feature extraction, the smaller matrix size (40x40 and 80x80) extracted feature data performs stably across all classifiers. Among them, Random Forest outperforms. However, with sizeable matrix-sized feature data, we see a high variance in accuracy.

The results, presented in Table 6.2, provide insight into the classification effectiveness of different ML models in bridge scenarios. Evaluation metrics, including Accuracy, F1-Score, Precision, and Recall, highlight the comparative strengths of these models under various conditions. RF consistently emerges as the best-performing model, achieving a peak accuracy of **93.54%** in the augmented dataset, confirming its robustness in bridge scenarios. The Ensemble models also demonstrate strong reliability, achieving an accuracy of **86.12%**, confirming their suitability for bridge monitoring applications.

6.7.1.2 Deep Neural Network Based Model

We also analysed DNN for performance evaluation. In DNN, we tried different batch sizes, learning rates, and optimisers. The train and test accuracy results for different batch sizes using the ADAM and SGD optimisers are illustrated in Fig.6.6a. The batch sizes considered are 128, 256 and 512. For ADAM Optimizer, as shown in Fig.6.6a the train accuracy starts high for (40x40) but slightly declines as the batch size increases. But for (160x160, 80x80 and 320x320) it fluctuates, some showing an increasing trend and others decreasing. High training accuracy is achieved for small batches in the case of (40x40), but it dips initially and recovers at a larger batch size for (320x320) while a decline was observed in (80x80), which could indicate overfitting in small batch size. Eventually (40x40) outperforms other experiments with any batch sizes. In test accuracy, ADAM Optimizer for (160x160) and BS(40x40) improves with increasing batch size. For (320x320) shows a **U-Shape**, initially decreasing and recovering at higher batch sizes. (80x80) start high but then sharply decline with a higher batch size. We see a key observation that ADAM performs well with smaller batch sizes but struggles with stability for larger ones.

However, for the Stochastic Gradient Decent(SGD) optimiser, all (40x40,80x80, 160x160, and 320x320) show a declining trend as the batch size increases in training. (40x40) maintains a higher accuracy, suggesting that the SGD optimiser works better with smaller batch sizes. (320x320) and (80x80) have a lower performance, indicating that the SGD optimiser is not well suited for larger batch sizes during the training process. A similar trend is observed in

Table 6.2: Performance Comparison Of Traditional ML Algorithms For Bridge Scenario

Dataset	ML Algorithm	Accuracy	F1 Score	Precision	Recall
Ambient Data	KNN-320	64.28	64.03	64.58	64.29
	KNN-160	68.13	68.31	69.28	68.13
	KNN-80	66.21	66.41	67.06	66.21
	KNN-40	**72.80**	**73.14**	**73.87**	**72.80**
	RF-320	76.65	76.72	77.03	76.65
	RF-160	76.37	76.64	77.47	76.37
	RF-80	75.27	75.42	75.71	75.27
	RF-40	**83.79**	**83.99**	**84.32**	**83.79**
	SVM-320	50.00	43.73	57.02	50.00
	SVM-160	57.14	51.88	54.51	57.14
	SVM-80	**59.06**	56.36	64.92	**59.07**
	SVM-40	57.69	**56.38**	**66.79**	57.69
	MLP-320	44.51	41.03	47.77	44.51
	MLP-160	**57.42**	**53.76**	**58.93**	**57.42**
	MLP-80	31.59	28.88	28.01	31.59
	MLP-40	38.18	33.20	57.50	38.19
	Ensemble-320	70.05	69.29	74.58	70.05
	Ensemble-160	71.15	70.73	75.51	71.15
	Ensemble-80	70.60	70.83	74.70	70.60
	Ensemble-40	**78.02**	**78.43**	**91.99**	**78.02**
Augmented Data	KNN-320	75.41	75.47	75.78	75.42
	KNN-160	77.47	77.57	77.85	77.47
	KNN-80	76.24	76.33	76.47	76.24
	KNN-40	**79.39**	**79.39**	**79.60**	**79.39**
	RF-320	**93.54**	**93.52**	**93.58**	**93.54**
	RF-160	92.72	92.73	92.77	92.72
	RF-80	93.27	93.29	93.38	93.27
	RF-40	92.99	93.03	93.23	92.99
	SVM-320	42.03	38.03	56.78	42.03
	SVM-160	**63.33**	61.28	**70.46**	**63.32**
	SVM-80	61.81	56.66	58.32	61.81
	SVM-40	63.32	**62.74**	69.46	**63.32**
	MLP-320	55.91	51.86	62.71	55.91
	MLP-160	60.99	60.60	68.07	60.98
	MLP-80	58.10	58.51	66.57	58.11
	MLP-40	**61.40**	**61.44**	**69.54**	**61.40**
	Ensemble-320	81.73	82.01	85.35	81.73
	Ensemble-160	82.97	83.24	86.25	82.96
	Ensemble-80	83.52	83.51	86.60	83.51
	Ensemble-40	**86.12**	**86.37**	**88.90**	**86.13**

the test phase. SGD optimiser performed better for small batch size, showing steady improvements for (40x40). The test accuracy deteriorates rapidly for

large batch sizes, indicating that they may not work well with the SGD optimiser. Overall, SGD Optimizer shows a more stable test accuracy for small batch sizes but does not generalise well for larger batches. ADAM Optimizer is more effective at converging quickly, whereas SGD Optimizer requires careful tuning to prevent a drop in accuracy.

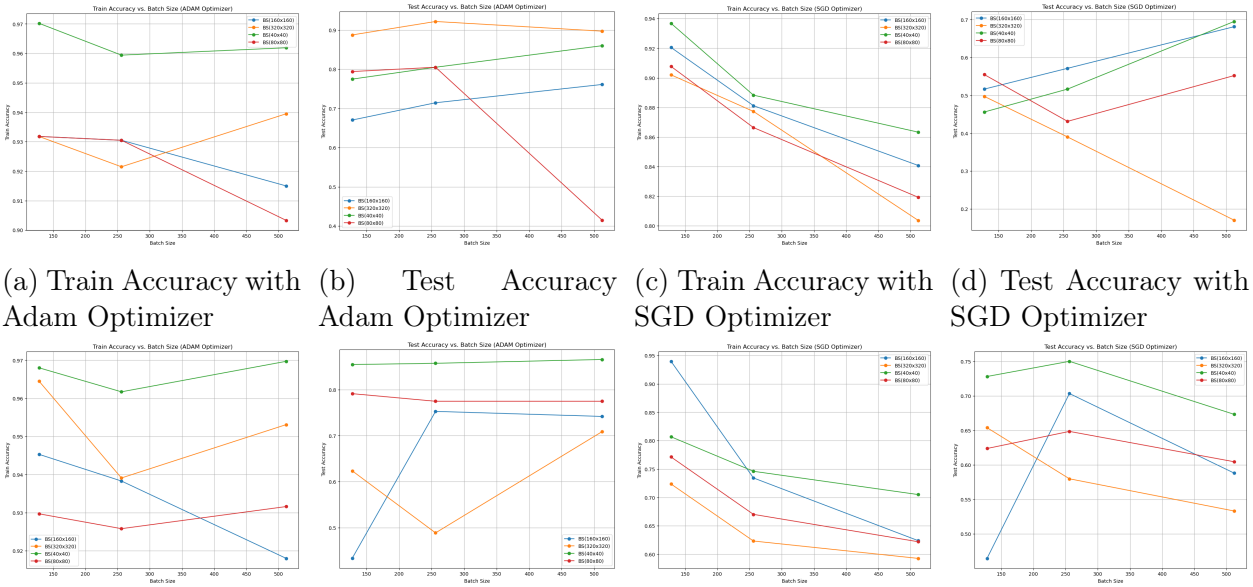
As in Fig.6.6, the performance of DNN with different optimisers and batch sizes is elaborated. It is evident from these figures that the train and test accuracy is higher with smaller slicing at large batch sizes with Adam Optimizer, but it is inverse in the case of small batch size outperformed.

The comparative analysis of ADAM and SGD for bridge scenario classification reveals that ADAM is the more effective optimiser, providing higher classification accuracy, better convergence, and better stability across all feature slice sizes. SGD, on the other hand, struggles with learning stability, particularly for larger feature slices (160x160 and 320x320), which exhibit declining accuracy trends in training and test phases. Additionally, (40x40) and (80x80) emerge as the most compelling feature slice sizes for bridge scenario classification, maintaining higher accuracy and better optimisation performance than larger feature slices. SGD's declining accuracy trends across all feature slices confirm that it is less suitable for bridge classification tasks at this learning rate. It requires further tuning and optimisation strategies to improve its performance.

In conclusion, ADAM is the preferred optimiser for CNN-based bridge scenario classification at a learning rate 0.001, as in Fig. 6.6e and Fig. 6.6f, as it ensures higher accuracy, better convergence and better stability across varying feature slice sizes. SGD, while slightly improved with lower learning rates, remains less effective in achieving stable classification results as in Fig. 6.6g and Fig. 6.6h. The findings suggest that future research should explore adaptive learning rate strategies, hybrid optimization methods, and feature selection techniques to enhance further classification accuracy and model robustness for bridge monitoring applications.

The results demonstrate that batch size selection is crucial in determining model performance. ADAM performs well with smaller batches but is less stable for larger batch sizes, while SGD is more stable but struggles with larger batch sizes. The study suggests that an optimal batch size of BS(40x40) or BS(160x160) provides the best balance between training efficiency and generalisation for both optimisers.

Future research can explore the impact of different learning rate schedules and hybrid optimisation strategies to further enhance the model's robustness across varying batch sizes.



(a) Train Accuracy with Adam Optimizer (b) Test Accuracy Adam Optimizer (c) Train Accuracy with SGD Optimizer (d) Test Accuracy with SGD Optimizer

(e) Train Accuracy with Adam Optimizer (f) Test Accuracy with Adam Optimizer (g) Train Accuracy with SGD Optimizer (h) Test Accuracy with SGD Optimizer

Figure 6.6:]
Comparison of Bridge Scenario w.r.t Learning Rate [0.01] (6.6a, 6.6b, 6.6c, 6.6d) & [0.001] (6.6e, 6.6f, 6.6g, 6.6h)

6.7.2 Damage Intensities Predictions

We have 26 different levels of bridge damage intensities divided into four subclasses. Each damage intensity represents a percentage of the damage intensity applied to our FEM.

6.7.2.1 Traditional Machine Learning Algorithms

As Explained earlier, We also compared Damage Intensities with traditional machine learning algorithms. The performance of machine learning(ML) classifiers for damage intensity classification was evaluated using both the original ambient dataset and the augmented dataset. The classifiers include a Support Vector Machine (SVM), a Multi-Layer Perceptron (MLP), a Random Forest, K-nearest neighbours (KNN), and an Ensemble Model. Each of the 26 damage intensities described above is re-classed into four subclasses by merging. We then performed performance accuracy using traditional machine learning algorithms (SVM, KNN, RF, Ensemble(KNN, RF & SVM), and MLP). Each classifier was assessed using different batch sizes. We calculated the model accuracy with varying matrix sizes, that is, (320x320, 160x160, 80x80, and 40x40) to analyse better classification results. The results in Fig.6.5c and Fig.6.5d show the performance of these ML models.

The results indicate that data augmentation significantly improves accu-

racy for most classifiers, particularly MLP and SVM, which exhibit notable improvements compared to their performance on original data. But overall, Random Forest and Ensemble models consistently outperformed other classifiers, achieving the highest accuracy across all feature data matrix sizes, with minimal impact of data augmentation, suggesting their strong generalisation ability. Besides, KNN also expressed stable accuracy, but it only benefits from augmentation data. Regarding sensitivity, MLP behaves with great sensitivity with data augmentation, exhibiting a substantial increase in accuracy. This suggests that the MLP model requires diverse training data with hyper-parameter fine-tuning for optimal performance. In contrast, SVM also benefits from augmented data, mainly feature data sizes 160x160 and 40x40. This reinforces the importance of feature diversity in SVM in improving generalisation.

The effect of feature data slice size is also evident in all classifiers. Larger feature data slicing sizes (320x320 and 160x160) result in more stable accuracy, particularly for Random Forest, KNN, and Ensemble models. In contrast, smaller slicing sizes (80x80 and 40x40) exhibit higher variance in terms of accuracies, with MLP performing worst on small slicing when trained with original data but significantly improving with augmented data. The variability in accuracy suggests that choosing an appropriate feature data slice is crucial for classifier stability for better railway steel bridge health monitoring for improved generalisation.

Table 6.3 provides a comparative summary of the evaluation of the performance of traditional machine learning (ML) algorithms to evaluate the severity of the damage in the health monitoring of the rail steel bridge, with different size-slice sizes of the feature data (320x320, 160x160, 80x80 and 40x40). The Random Forest achieves the highest accuracy overall, closely followed by the Ensemble model, which remains robust regardless of data augmentation. Random Forest (RF) consistently emerges as the best-performing model, achieving the highest accuracy of 98.21% with augmented data, while Ensemble models also demonstrate robust performance, reaching an accuracy of 94.19%. However, SVM and MLP exhibit substantial improvements when augmented data is used. MLP shows the most significant performance gap between original and augmented datasets but the lowest performance for damage intensities of railway steel bridges. KNN, while performing moderately well, benefits slightly from augmentation on damage intensities but does not exhibit significant performance variations across batch sizes.

In ambient data, SVM records its lowest accuracy of 57.69% at 80x80 slice size. In contrast, after augmentation, its accuracy improves to 75.27% at 160x160, demonstrating the effectiveness of data augmentation in improving

its generalisation capabilities for bridge health monitoring. Similarly, MLP, which initially struggles with an accuracy of 55.77% in ambient data, improves to 75.14% with augmented data, reinforcing the observation that deep learning models require diverse and enriched data for optimal learning in structural health assessment.

In conclusion, the Ensemble and Random Forest classifiers are the most effective models for classifying damage intensity, offering the highest accuracy and stability across batch sizes. However, MLP and SVM greatly benefit from data augmentation, highlighting the importance of training data diversity. The study further confirms that batch size selection is crucial in achieving stable and high-performing ML models, with larger batch sizes (320x320, 160x160) leading to better generalisation. Future research could explore hybrid feature extraction methods and adaptive learning rate schedules to optimise classification performance.

6.7.2.2 Deep Neural Network Based

The accuracy plots provide information on the performance comparison of the SGD and ADAM optimisers to classify damage intensities in railway steel bridges using a learning rate of 0.01 and 0.001. The models were trained with different feature data slice sizes and evaluated based on training and test accuracy trends.

The comparative analysis of ADAM and SGD optimisers for the classification of damage intensities in railway steel bridges was carried out using a learning rate of 0.01, with varying feature data slice sizes (320x320, 160x160, 80x80 and 40x40). The test and training accuracy trends were evaluated to assess optimisers' performance and generalisation capability. The test accuracy results indicate that ADAM consistently outperforms SGD in all feature data slice sizes as in Fig. 6.7c and Fig. 6.7a, demonstrating better convergence and stability. The test accuracy trends reveal that the feature data slice (160x160) achieves the highest and most stable accuracy under ADAM, as shown in Fig. 6.7b, maintaining a progressive improvement as batch size increases. Similarly, DI(320x320) maintains a stable upward trend, confirming ADAM's ability to generalise well across different bridge monitoring configurations. However, DI(40x40) and DI(80x80) exhibit fluctuating trends, with DI(40x40) showing an initial rise before declining at larger batch sizes, indicating possible overfitting in smaller feature slice sizes.

In contrast, the SGD optimiser struggles with stability and generalisation, as reflected in the test accuracy results. Unlike ADAM, SGD experiences inconsistent accuracy trends, particularly in (160x160) and (320x320), where accuracy

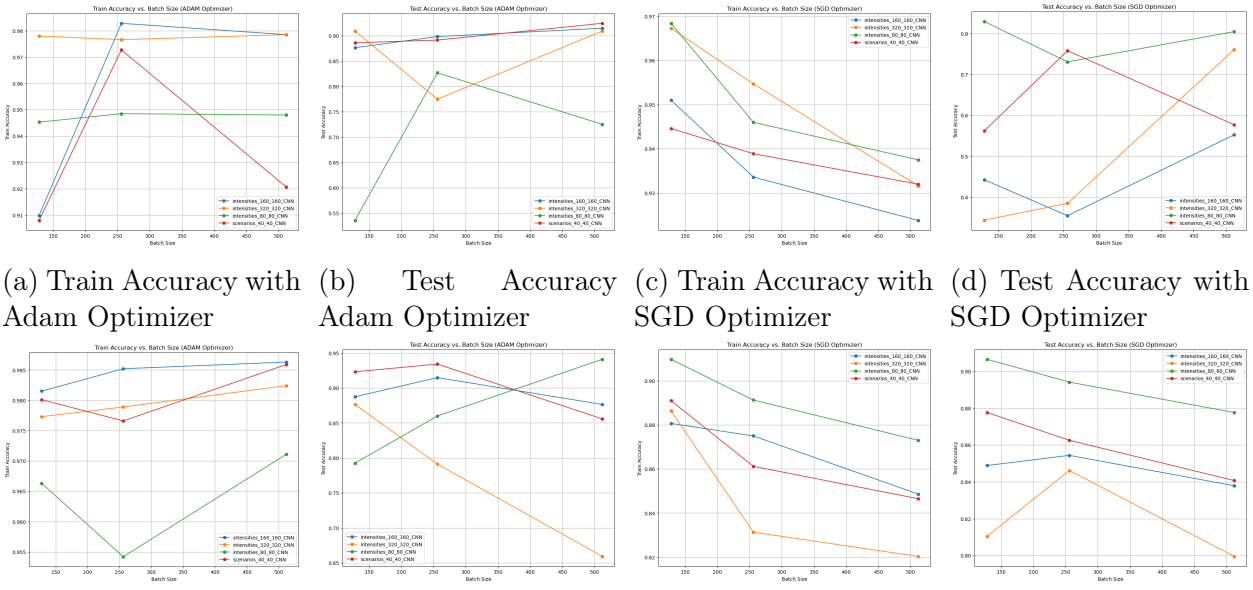
starts lower and increases at larger batch sizes. However, (40x40) and (80x80) show erratic behaviour, with a noticeable decrease at specific batch sizes, as seen in Fig. 6.7c and Fig. 6.7d, further suggesting the sensitivity to feature slice size variations over the SGD optimiser. The lower initial accuracy and unstable trends indicate that SGD requires more fine-tuning to provide effective learning in bridge damage localisation and classification.

The training accuracy results further highlight the differences between ADAM and SGD. Under ADAM, the training accuracy remains high in all feature slice sizes, with (160x160) and (80x80) reaching peak accuracy near 98%, confirming strong convergence. DI(40x40) maintains a stable accuracy trend, with minimal fluctuations, while (320x320) shows a slight decline at larger batch sizes, suggesting that regularisation may be needed to mitigate overfitting. In contrast, SGD exhibits declining training accuracy trends across all feature slice sizes, reinforcing its instability at this learning rate. The gradual decrease in training accuracy with increasing batch size indicates SGD's difficulty maintaining effective learning, making it less suitable for this application.

The comparative evaluation of ADAM and SGD optimisers for damage classification and localisation in railway steel bridges was conducted using two learning rates (0.01 and 0.001) and feature data slice sizes of 320x320, 160x160, 80x80 and 40x40. The test and training accuracy trends were analysed to assess both optimisers' optimisation performance, generalisation capability, and localisation accuracy. The results indicate that ADAM consistently outperforms SGD, achieving higher accuracy and stability across all configurations. ADAM maintains a stable and increasing trend in test accuracy, with (160x160) and (80x80) reaching peak accuracies above 90% , demonstrating effective damage localisation and classification capabilities. However, (40x40) exhibits fluctuating accuracy, initially increasing and declining at larger batch sizes, suggesting overfitting issues when using extremely fine-grained feature slices. The (320x320) feature slice shows a decreasing trend, indicating that larger feature slices may generalise too broadly, leading to reduced classification precision.

In contrast, SGD exhibits lower test accuracy and greater instability between feature slice sizes. The test accuracy trends for (160x160) and (320x320) under SGD show inconsistent improvements, while (40x40) and (80x80) demonstrate erratic behaviour, failing to maintain performance consistency. At a learning rate of 0.01, SGD struggles with generalisation, resulting in poor classification accuracy and significant performance variations between batch sizes. However, when the learning rate is reduced to 0.001, SGD shows slight improvements in stability, although it remains inferior to ADAM in overall accuracy and

robustness.



(a) Train Accuracy with Adam Optimizer (b) Test Accuracy Adam Optimizer (c) Train Accuracy with SGD Optimizer (d) Test Accuracy with SGD Optimizer

(e) Train Accuracy with Adam Optimizer (f) Test Accuracy with Adam Optimizer (g) Train Accuracy with SGD Optimizer (h) Test Accuracy with SGD Optimizer

Figure 6.7:]
Comparison of Damage Intensities w.r.t Learning Rate [0.01] (6.7a, 6.7b, 6.7c, 6.7d) & [0.001] (6.7e, 6.7f, 6.7g, 6.7h)

The training accuracy trends further highlight the differences between the two optimizers. Under Adaptive Moment Estimation (ADAM), as in Fig. 6.7a and Fig. 6.7e, the training accuracy remains consistently high, with (160x160) and (80x80) achieving nearly 98% accuracy, confirming strong convergence and practical learning. Although (40x40) initially dips and later recovers, its fluctuating trend indicates overfitting concerns at smaller feature slice sizes. Meanwhile, (320x320) maintains stable but slightly declining accuracy at larger batch sizes, suggesting that larger feature slices may be less effective for precise damage localization. In contrast, SGD exhibits declining training accuracy trends in all feature slice sizes, emphasizing its difficulty in maintaining learning stability. As batch sizes increase, SGD struggles to maintain training accuracy, with (160x160) and (320x320) showing the most severe drops, confirming its inability to achieve stable convergence at higher learning rates. Although reducing the learning rate to 0.001 slightly improves the stability of the SGD optimizer, it still fails to reach the same accuracy levels as ADAM.

The comparative findings confirm that ADAM is the superior optimizer for CNN-based damage classification and localization in railway steel bridges, as it ensures better convergence, higher accuracy, and greater stability in different feature slice sizes. In contrast, Stochastic Gradient Descent (SGD) struggles with training convergence and test accuracy stability, particularly at higher

learning rates. The results also indicate that (160x160) is the most reliable feature slice size for classification and localization, balancing granularity and generalization. (40x40) demonstrates inconsistent behavior with both optimizers, suggesting that tiny feature slices may introduce noise rather than enhance localization precision. Furthermore, (320x320) performs relatively well under ADAM but suffers from declining accuracy under SGD, reinforcing the limitations of SGD in optimizing large feature representations for damage classification.

Overall, the findings suggest that ADAM is the preferred optimizer for damage classification and localization in railway steel bridges, as it provides higher accuracy, better convergence, and better stability in different feature slice sizes and learning rates. SGD, while slightly improved at a lower learning rate of 0.001, remains less effective in achieving stable and accurate classification results. These results indicate that future research should focus on adaptive learning rate strategies, hybrid optimization methods, and feature selection techniques to improve classification accuracy and localization precision in railway bridge health monitoring.

6.8 Summary

This chapter explored the application of both classical machine learning algorithms and deep learning models for structural health monitoring of railway steel bridges. Classical algorithms such as KNN, SVM, RF, and MLP were evaluated for their ability to classify damage intensities based on engineered statistical features. While these methods offered moderate performance, their sensitivity to feature scaling and class imbalance posed limitations.

To address these challenges, a DNN was developed and trained on time-domain features derived from physics-informed vibration data. The DNN model outperformed most classical methods, particularly when smaller data slices and augmented datasets were used. The results highlight that while classical models are efficient for early-stage deployment and exploratory analysis, DNNs provide superior flexibility and accuracy for large-scale or high-resolution SHM tasks. The integration of slicing strategies and data augmentation further enhanced the classification performance across multiple damage scenarios.

The experimental evaluations presented in this chapter confirm the effectiveness of machine learning models; both classical and deep; in accurately classifying structural conditions in railway steel bridges. Traditional ML models, including KNN, SVM, RF, MLP, and ensemble techniques, demonstrated strong performance in both bridge scenario classification and damage inten-

Table 6.3: Performance Comparison Of Traditional ML Algorithms For Damage Intensities

Dataset	ML Algorithm	Accuracy	F1 Score	Precision	Recall
Ambient Data	KNN-320	89.28	88.98	90.28	89.89
	KNN-160	89.01	88.95	90.03	89.01
	KNN-80	**91.21**	**91.14**	**91.74**	**91.20**
	KNN-40	90.66	90.56	90.89	90.65
	RF-320	90.66	90.46	91.02	90.65
	RF-160	92.03	91.91	92.14	92.03
	RF-80	91.76	91.61	92.05	91.75
	RF-40	**93.41**	**93.31**	**93.50**	**93.49**
	SVM-320	**65.10**	**61.61**	**76.17**	**65.11**
	SVM-160	59.89	51.18	50.32	59.89
	SVM-80	57.69	47.86	47.55	57.69
	SVM-40	60.71	53.68	67.67	60.71
	MLP-320	**68.41**	**63.49**	**74.38**	**68.40**
	MLP-160	55.77	46.27	46.52	55.76
	MLP-80	57.96	49.06	51.24	55.76
	MLP-40	60.44	53.33	59.58	60.43
	Ensemble-320	89.28	88.96	91.13	89.30
	Ensemble-160	88.18	88.13	89.96	88.18
	Ensemble-80	89.83	89.72	90.98	89.84
	Ensemble-40	**90.66**	**90.52**	**91.68**	**90.65**
Augmented Data	KNN-320	92.71	92.64	92.71	92.72
	KNN-160	91.34	91.32	91.34	91.35
	KNN-80	92.58	92.56	92.69	92.58
	KNN-40	**92.85**	**92.82**	**92.93**	**92.85**
	RF-320	**98.21**	**98.21**	**98.24**	**98.21**
	RF-160	98.07	98.07	98.08	98.07
	RF-80	96.02	95.98	96.18	96.01
	RF-40	96.01	95.99	96.04	96.02
	SVM-320	67.58	62.93	78.47	67.58
	SVM-160	**75.27**	**72.37**	**81.66**	**75.27**
	SVM-80	73.91	71.07	79.35	73.90
	SVM-40	74.86	71.79	81.14	74.86
	MLP-320	71.84	68.41	72.91	71.84
	MLP-160	**75.14**	**73.64**	**75.85**	**75.13**
	MLP-80	72.25	71.91	72.10	72.25
	MLP-40	72.80	68.99	72.12	72.80
	Ensemble-320	94.09	94.10	94.49	94.09
	Ensemble-160	**94.19**	**94.19**	**94.82**	**94.23**
	Ensemble-80	93.81	93.76	94.38	93.82
	Ensemble-40	93.54	93.47	94.06	93.54

sity prediction, particularly when trained with statistically augmented feature data. Among these, RF consistently achieved the highest accuracy across all

slice sizes and datasets, with a peak performance of **93.54%** for bridge scenarios and **98.21%** for damage intensity prediction on augmented data. Ensemble methods also performed robustly and demonstrated stability across varying data matrix sizes.

The comparative analysis revealed that smaller feature slices (for example, 40×40 and 80×80) generally improved classification accuracy for most classifiers, especially in RF and KNN, while SVM and MLP were highly sensitive to both data augmentation and feature granularity. These observations emphasize the importance of batch size tuning and feature diversity in ML-based SHM pipelines.

Deep Neural Networks(DNNs) were also implemented and evaluated using different optimizers (ADAM and SGD), learning rates, and feature slice sizes. The results showed that the ADAM optimizer outperforms SGD in both training stability and test accuracy. ADAM achieved consistent performance improvements across all slicing sizes, particularly for 160×160 and 80×80 slices. In contrast, SGD showed greater variance and instability, especially with larger batch sizes and higher learning rates. The best performance was observed with the ADAM optimizer at a learning rate of 0.001, confirming its superiority in convergence speed and generalization.

Overall, the study concludes that **RF** and **Ensemble models** are best suited for efficient and reliable SHM tasks using ambient feature sets, while **ADAM-optimized DNNs** excel in deeper learning scenarios that require adaptive gradient updates. The results also confirm that **feature slice size** and **optimizer selection** play crucial roles in determining classifier performance. Future research should investigate hybrid models, adaptive learning strategies, and real-time data integration to further enhance SHM system performance in practical railway bridge monitoring applications.

Chapter 7

Railway Steel Bridges Health Monitoring in Frequency Domain

"A mathematical model does not have to be exact; it just has to be close enough to provide better results than can be obtained by common sense."

— Herbert A. Simon.

Frequency domain analysis is a powerful tool in Structural Health Monitoring (SHM), particularly for systems like railway steel bridges that experience dynamic and periodic loads. Time-domain sensor signals often contain overlapping patterns that are hard to distinguish. Frequency domain representations allow for more intuitive detection of damage by revealing shifts in energy distribution, resonance, and modal characteristics that accompany structural degradation.

In this chapter, we apply signal transformation techniques to convert accelerometer data into spectrograms and explore deep learning approaches to classify structural health states from these time-frequency representations.

7.1 Introduction

Bridge health monitoring is becoming an interesting domain for civil engineers along with artificial intelligence researchers; after the rise of machine learning. Bridge Structural Health Monitoring SHM involves continuous health monitoring of bridge, condition and damage deterioration assessment, and responsive maintenance of the bridge structure. Despite being a great mean of socio-economic development, these bridge structures have inevitably resulted in safety concerns, damages, and/or functional failures due to natural disasters, deterioration, environmental effect and damages. So, it becomes evident to have continuous monitoring, checking, and maintenance of these bridges to meet the safety concerns.

In recent times, many damage detection and structural health monitoring techniques; Neural Network, structural dynamics etc, based on non-destructive structural methods have been developed, [26]. Neural Network based techniques for bridge SHM [85, 109–113] including previous work about Neural Network based on RBF [85, 115] method for Bridge SHM, prove to be more cost effect, fast and more accurate. Most of these works focus of image based predictions or frequency domain implementations. However, they involve data acquisition from sensors, analyzing monitor parameters like; inclination , vibration, strain, temperature etc. Bridge SHM system employs these sensors and techniques to detect any structural changes, deformations or any damage.

Now a days, engineers analyze and monitor Structures in real-time and identify anomalies, early detection of damages and ensure them to take proac-

Table 7.1: Frequency Domain Dataset Properties

Feature	Value	Comments
Sensor Type	Accelerometer	mono-,bi, tri-axial
Images Type	Spectrograms	[256x256] size
Sampling Frequency	200Hz	10sec duration
Damage Localization Classes	7	4 sub classes[low, medium, high, critical]
Damage Classes	26	26 classes merged into 4 groups

tive measure for maintenance and hence extend the structural lifespan of such bridges. Anomaly detection [109], damage localization [111, 116, 117], damage classification [112], monitoring response, and alert generation based on structural factors are becoming easy to investigate steel bridges with machine learning algorithms. However, a lot of work is still needed to address specific challenges. These challenges include technological difficulties, environmental factors, maintenance cost, socioeconomic balance, and risk of safety [118].

In this study, we propose a novel Dual Level Fusion Learning (DL-FL) approach for the multiclass classification of bridge scenarios method as shown in Fig. 7.1 using spectrogram images of time series data obtained from accelerometer sensors. Our model integrates a DNN architectures with Transformer Block in a dual-branch fusion framework to capture a broader range of features from spectrogram images.

By employing semi-supervised learning techniques, we utilize unlabeled data to enhance the learning process without relying solely on labeled datasets.

This approach has undergone rigorous optimization through extensive experimentation to detect and classify multiple bridge scenarios of our railway steel bridge located at *Cittá di Castello* in Italy.

The proposed model is distinguished by several key implementations that position it favorably in structural bridge health monitoring.

- Unlike conventional DNN models that typically utilize a single architectural backbone, we propose a Dual-Branch DNN model with Transformerś feature fusion with addition to semi-supervised learning. This leverages both labeled and unlabeled data to classify damage localization based on different bridge scenarios. By incorporating different DNN models and Transformer fusion, our approach captures a broader range of features from spectrogram images, improving the model's ability to distinguish subtle variations in steel bridge health conditions.
- Our evaluation utilizes the dataset, comprising data from accelerometer sensors on FEM of original railway steel bridge in *Cittá di Castello* in Italy.

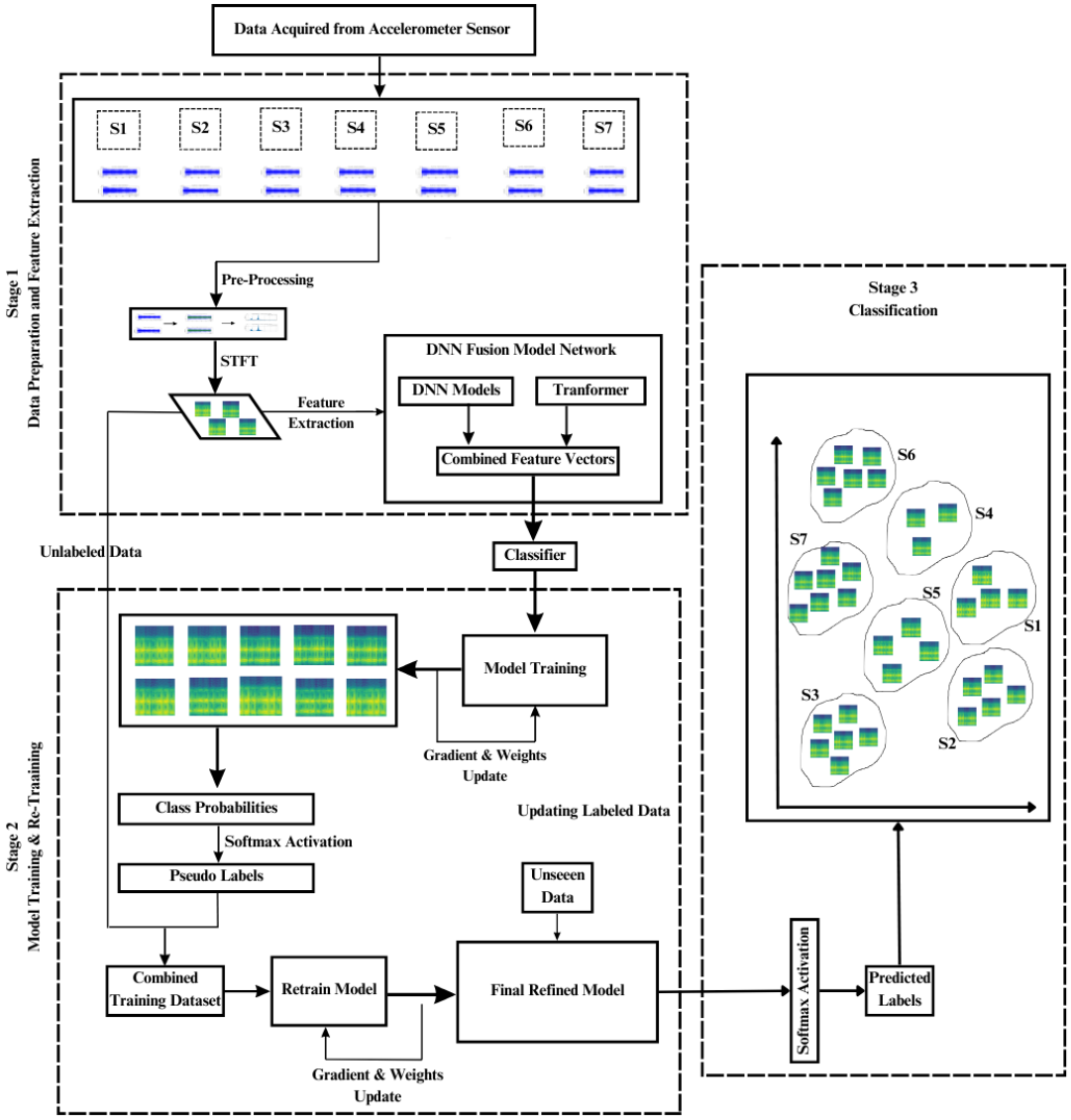


Figure 7.1: System Overflow of Dual Level Deep Learning Fusion Framework in Frequency Domain For Railway Steel Bridge Health Monitoring. The system incorporates STFT, CNN-ViT fusion, and pseudo-labeling for robust classification.

Using STFT, spectrograms are generated and then separated into train, test, and validate dataset. After initial training, unseen spectrograms are merged based on the trained model in a semi-supervised way, further enhancing the evaluation.

7.2 Background

Vibration-based Structural Health Monitoring SHM in the frequency domain has become a cornerstone technique for assessing the structural integrity of railway bridges. These approaches analyze changes in a structure's dynamic characteristics; such as natural frequencies, damping ratios, and mode shapes, to detect damage at an early stage.

The transformation of vibration signals into the frequency domain allows engineers to detect shifts in spectral energy and modal behavior caused by structural deterioration. Common tools include the FFT, Short-Time Fourier Transform STFT, and Wavelet Transform WT, which convert raw time-domain signals into frequency-rich representations [144,145]. Recent studies highlight a growing trend of applying frequency-domain analysis in SHM of railway bridges.

Structural Health Monitoring SHM systems play a critical role in ensuring the safety and longevity of railway steel bridges subjected to repetitive dynamic loads and environmental effects. While traditional SHM approaches rely heavily on time-domain analysis of vibration signals, recent advances emphasize the frequency domain due to its enhanced ability to capture subtle changes in structural behavior [146].

Nguyen et al. [147] performed vibration-based SHM of the Dębica railway arch bridge using orbit-shaped analysis and frequency transforms over a 9-month period. Mohan et al. [148] proposed a frequency-based SHM model using wavelets, STFT, and spectrograms as features for machine learning models such as decision trees and residual neural networks. Operational Modal Analysis(OMA) techniques have been applied to bridge structures to monitor their natural vibration frequencies under operational and environmental variability [149].

Frequency-domain analysis techniques, such as the Fast Fourier Transform FFT, and Wavelet Transform WT, have been widely adopted to localize and quantify damage by identifying shifts in resonance frequencies, changes in spectral energy, and the emergence of new vibration modes [144, 145]. These transformations convert raw sensor signals into frequency signatures, which are more robust to noise and can reveal underlying modal information about the structure [150].

In the specific context of railway steel bridges, frequency-domain features have been used to detect bolt loosening, girder deformation, and fatigue-related damage [151, 152]. Accelerometer signals recorded from bridge components can be transformed into time-frequency representations such as spectrograms, which are useful for identifying localized damage that may not be apparent in raw signals. These spectrograms are well-suited for modern deep learning models such as Convolutional Neural Networks(CNNs)and Transformer architectures [153].

Several studies have shown the benefits of combining frequency-domain analysis with machine learning for SHM. For instance, Geng et al. [154] applied STFT and CNN-based classifiers to detect damage in railway viaducts using spectral images. Similarly, Wang et al. [155] used vibration spectrograms with deep learning to detect abnormalities in bridge expansion joints.

Despite these advances, challenges remain in extracting reliable frequency features under varying operational and environmental conditions. To address these issues, hybrid approaches combining STFT with feature slicing, data augmentation, and fusion architectures have emerged as promising solutions. This chapter builds upon this foundation by proposing a fusion deep learning model that leverages frequency-domain representations for robust bridge damage classification.

7.3 Materials and methods

We propose a novel Dual-Branch Semi-Supervised Learning (DB-SSL) and supervised approach for damage classification in railway steel bridge scenarios, as illustrated in Figure 7.1. The system integrates a dual-branch convolutional neural network CNN with a semi-supervised learning strategy based on self-training to classify multiple bridge scenarios.

Initially, vibration data are transformed into time-frequency spectrograms using the SVD and Short-Time Fourier Transform STFT. These spectrograms are standardized through normalization and resized to uniform dimensions.

To extract meaningful time-frequency representations from vibration data for Structural Health Monitoring (SHM) of railway steel bridges, a systematic transformation pipeline was employed. Initially, raw accelerometer signals were organized into Hankel matrices, as described earlier in Chapter 3 section 3.3, which preserve temporal structures across overlapping signal segments. Each matrix, denoted as $\mathbf{T}_b$, represents a block-wise window of the time series data.

For each $\mathbf{T}_b$ matrix, Singular Value Decomposition SVD was applied to isolate its principal spectral components. This decomposition allows for the separation of dominant structural modes from noise and enables more efficient

signal compression and reconstruction. The resulting SVD-processed matrices were then passed through a frequency transformation stage.

Subsequently, the processed data were sampled at a frequency of 200Hz and used to compute spectrograms via the Short-Time Fourier Transform STFT. The STFT was configured with the following parameters; **Sampling Frequency** (f_s) = 200 Hz, **Segment Length** ($n_{\mathbf{perseg}}$) = 64, **Segment Overlap** ($n_{\mathbf{overlap}}$) = 32, **Time Window Duration** = 10 seconds

Mathematically, the STFT is expressed as:

$$\mathrm{STFT}\{x(t)\}(m,\omega) = \int_{-\infty}^{\infty} x(t)w(t-m)e^{-j\omega t}dt \tag{7.1}$$

where $w(t-m)$ is the window function applied to each segment, and m represents the time-shifting parameter.

Table 7.2: Distribution of Dataset.

Classes	**Labeled Data** (80%)		**Unlabeled** (20%)
	Training (80%)	**Validation** (20%)	
S1	526	134	-
S2	715	146	-
S3	659	145	-
S4	623	135	-
S5	592	145	-
S6	655	157	-
S7	669	137	-
Total	4439	999	1827

The STFT yields a spectrogram, a two-dimensional representation of frequency content over time, highlighting how structural responses evolve dynamically. These spectrograms were normalized and resized to consistent dimensions of 256×256 for uniform input formatting. This standardized format is ideal for deep learning models, such as Deep Neural Networks (DNNs) and Transformer-based architectures, enabling robust classification of damage scenarios and intensity levels.

7.3.1 Singular Value Decomposition (SVD)

Singular Value Decomposition SVD is a fundamental matrix factorization technique widely used in signal processing, data compression, and noise filtering. In the context of Structural Health Monitoring (SHM), SVD provides a robust

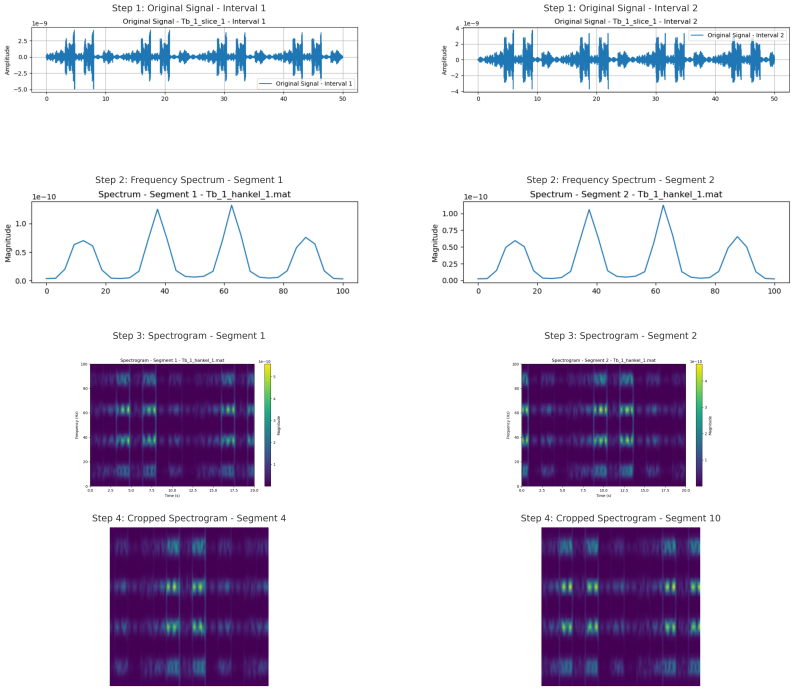


Figure 7.2: Step-wise pipeline for spectrogram generation: Original vibration signals are processed via SVD and STFT to yield full spectrograms, which are further cropped and normalized for input into deep learning models.

method to extract dominant patterns or modes from structured time series data, such as vibration signals organized into Hankel matrices.

7.3.1.1 Singular Value Decomposition of Hankel Matrix

Let $\mathbf{y} = [y_0, y_1, \ldots, y_{N-1}]^\top$ be a one-dimensional vibration signal of length N. From this, a block Hankel matrix $\mathbf{H} \in \mathbb{R}^{m \times n}$ is constructed such that:

$$\mathbf{H} = \begin{bmatrix} y_0 & y_1 & \cdots & y_{n-1} \\ y_1 & y_2 & \cdots & y_n \\ \vdots & \vdots & \ddots & \vdots \\ y_{m-1} & y_m & \cdots & y_{N-1} \end{bmatrix} \quad \text{with } m + n = N + 1 \tag{7.2}$$

We then apply Singular Value Decomposition(SVD) to factorize the Hankel matrix:

$$\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top \tag{7.3}$$

where:

- $\mathbf{U} \in \mathbb{R}^{m \times m}$ is the matrix of left singular vectors,
- $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$ is a diagonal matrix of singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$,

- $\mathbf{V} \in \mathbb{R}^{n \times n}$ is the matrix of right singular vectors,
- $r = \text{rank}(\mathbf{H})$.

Each singular value corresponds to the energy of a particular mode in the data. By truncating this decomposition, we can preserve the most significant dynamics while discarding noise and redundancy. This decomposition isolates the dominant signal modes captured in the vibration data, allowing dimensionality reduction and denoising before subsequent STFT transformation and classification.

7.3.1.2 Short-Time Fourier Transform of SVD-reduced [H] matrix

The STFT of the SVD-reduced signal $x_{\text{svd}}(t)$ is computed using:

$$\text{STFT}\{x(t)\}(m, \omega) = \sum_{n=-\infty}^{\infty} x[n] w[n - m] e^{-j\omega n} \tag{7.4}$$

where:

- $x[n]$ is the signal (for example, x_{svd}),
- $w[n]$ is the window function (for example, Hamming),
- m is the time shift (frame index),
- ω is the frequency bin.

In practice, we use:

- Sampling rate: $f_s = 200$ Hz,
- Window size: $n_{\text{perseg}} = 64$,
- Overlap: $n_{\text{overlap}} = 32$,
- Duration per segment: 10 seconds.

7.3.1.3 Spectrogram Definition

The spectrogram is the squared magnitude of the STFT:

$$\text{Spectrogram}(t, f) = |\text{STFT}(t, f)|^2 \tag{7.5}$$

This two-dimensional representation in Equation 7.5 encodes how the signal's energy is distributed across time and frequency. In the context of Structural Health Monitoring (SHM), spectrograms derived from vibration signals are used as input features to deep learning models; such as Convolutional Neural Networks (CNNs) and ViT-based fusion architectures, for tasks like damage localization and severity classification.

7.3.1.4 Application in SHM using Hankel Matrices

In this work, vibration signals were first converted into structured Hankel matrices $\mathbf{T}_b$ to preserve local temporal patterns. SVD was then applied to each $\mathbf{T}_b$ matrix to extract its principal components. This decomposition allows us to:

- Isolate the dominant vibration modes corresponding to structural dynamics,
- Reduce noise by truncating lower singular values,
- Compress information for efficient storage and processing.

The resulting decomposed matrices $\mathbf{U}$, $\mathbf{\Sigma}$, and $\mathbf{V}^\top$ capture the underlying structural signatures, which are later transformed via the Short-Time Fourier Transform (STFT) into spectrograms for classification. This preprocessing step enhances the sensitivity and robustness of deep learning models in identifying subtle changes in bridge behavior.

7.3.2 Deep Learning Fusion Framework for Frequency Domain SHM

Figure 7.2 shows the complete transformation process from raw vibration signals to spectrograms, including original signal slicing, spectral analysis, spectrogram generation, and segment cropping for classification.

This approach captures both modal shifts and transient changes in frequency response, offering high sensitivity to localized damage and structural degradation.

The core of the framework is a dual-branch DNN-ViT architecture designed to extract a rich set of features from the spectrograms. Each branch independently processes the same input but with distinct convolutional filters, enabling the network to capture a wider spectrum of structural features, such as global frequency patterns and localized transient anomalies. The outputs of both branches are concatenated and passed to a shared classification head for the final prediction.

7.3.3 Semi-Supervised Learning Using Self-Training and Data Merger

To address the scarcity of labeled structural data, the system incorporates a semi-supervised learning loop. Initially trained on a small labeled dataset, the model generates pseudo-labels for unlabeled samples with high-confidence predictions. These pseudo-labeled examples are then included in the training pool, enabling iterative refinement of the model through additional supervised

training cycles.

This dual-branch self-training strategy significantly improves classification performance in bridge health monitoring, especially under real-world constraints where labeled data are scarce and damage patterns vary in complexity and frequency signature.

7.3.4 Transfer Learning

Transfer learning is a powerful technique in deep learning where knowledge gained from training a model on one task or dataset is leveraged to improve performance on a different but related task. This is particularly valuable in Structural Health Monitoring SHM of railway steel bridges, where labeled datasets are often scarce and expensive to obtain due to the rarity and unpredictability of real-world damage scenarios.

7.3.4.1 Principle of Transfer Learning

In a typical transfer learning setup, a deep neural network (often pre-trained on a large-scale dataset such as ImageNet) is fine-tuned for a new SHM classification task. The earlier layers of the pre-trained model capture general features such as edges, textures, or frequency patterns that are transferable across domains, while the later layers are adapted to specialize in bridge-specific damage or scenario classification.

7.3.4.2 Application in Frequency-Domain SHM

In this work, deep convolutional neural networks (CNNs) were trained from scratch to perform structural damage classification tasks using spectrogram-based features. Unlike traditional transfer learning approaches that repurpose models pre-trained on natural image datasets (for example, ImageNet), we opted to train the networks directly on domain-specific time-frequency representations derived from vibration signals.

The models used include ResNet50, EfficientNetB0, and DenseNet121. These architectures were selected for their ability to capture complex spatial and frequency patterns from spectrogram input. Each model was initialized with random weights and optimized end-to-end using a labeled dataset generated from vibration signals processed via Short-Time Fourier Transform (STFT).

Training these models from scratch ensures that the learned features are fully tailored to the characteristics of railway bridge vibrations and damage patterns, resulting in more specialized and potentially more robust classifiers for Structural Health Monitoring (SHM).

This approach enables the networks to learn discriminative frequency-domain features that are highly relevant to bridge damage localization and intensity estimation, without inheriting any bias from unrelated pre-training domains.

The benefits of using transfer learning in this context include:

- **Faster Convergence:** Reduces training time and computational cost by leveraging existing weights.
- **Improved Performance:** Achieves higher accuracy even with limited training data.
- **Data Efficiency:** Minimizes the need for large annotated SHM datasets.

7.3.4.3 Fine-Tuning vs. Feature Extraction

Two common strategies are used in transfer learning:

- **Feature Extraction:** The pre-trained network is used as a fixed feature extractor, and only a new classification head is trained.
- **Fine-Tuning:** Some or all layers of the pre-trained network are retrained along with the classification head to adapt to SHM-specific features.

Transfer learning thus serves as a practical and efficient solution for enhancing the performance of SHM systems, especially in situations with limited labeled bridge monitoring data.

7.3.5 Features Extraction Using Dual-Branch Fusion Models

7.3.5.1 Stage 1: Data Preparation and Feature Extraction

Vibration signals are acquired from multiple accelerometer sensors installed across bridge segments (S1–S7). These signals are first preprocessed to remove noise and normalize the input format. Next, the Short-Time Fourier Transform STFT is applied to convert time-domain signals into spectrograms—2D time-frequency representations that preserve both the temporal and spectral characteristics of structural responses.

These spectrograms are then fed into a deep neural network (DNN) fusion model comprising two parallel components:

- **|glsDNN Models (for example, CNN):** Extract local spatial features from spectrograms.
- **Transformer Module:** Captures long-range dependencies and temporal sequences across feature slices.

The outputs of these components is combined into a unified feature vector for classification.

7.3.5.2 Stage 2: Model Training and Semi-Supervised Learning

The classifier is initially trained using labeled spectrogram data. For unlabeled or unseen samples, the model generates pseudo-labels via softmax activation. These predicted class probabilities are used to augment the labeled dataset. A retraining phase is then triggered using this expanded training set with updated gradients and model weights. This pseudo-labeling strategy enhances generalization and robustness to new bridge states, particularly when real labeled data is limited.

7.3.5.3 Stage 3: Classification and Decision Output

In the final stage, the refined model classifies incoming test data into one of the predefined bridge scenarios (S1–S7). The classification output is derived from the softmax-activated probabilities of the network. Predicted labels are mapped to structural locations, allowing for spatial interpretation of damage or vibration patterns across the bridge.

This multi-stage framework enables effective frequency-domain analysis by leveraging both deep learning and temporal feature modeling. It is scalable to various sensor configurations and can be further integrated with fuzzy logic layers for post-processing and interpretability.

7.3.6 Short-Time Fourier Transform (STFT)

The Short-Time Fourier Transform (STFT) is used to localize frequency content over time. It divides the signal into overlapping windows and applies the Fourier transform to each segment:

$$\mathrm{STFT}\{x(t)\}(m,\omega) = \int_{-\infty}^{\infty} x(t)w(t-m)e^{-j\omega t}dt \tag{7.6}$$

where $w(t)$ is the window function and m is the time-shift parameter. The resulting spectrogram is a 2D representation showing how frequency content evolves over time, which can be visualized and analyzed using convolutional and transformer-based models.

7.3.7 Loss Function

In deep learning models for classification tasks, the choice of loss function is critical as it quantifies the discrepancy between predicted outputs and true labels, guiding the learning process during backpropagation. In this work, the **categorical cross-entropy loss** was used as the primary loss function to

train our convolutional neural networks (CNNs) on bridge scenario and damage intensity classification tasks.

7.3.7.1 Categorical Cross-Entropy Loss

For multi-class classification problems, categorical cross-entropy is defined as:

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{7.7}$$

where:

- C is the total number of classes,
- y_i is the true label (1 for the correct class, 0 otherwise),
- $\hat{y}_i$ is the predicted probability for class i.

This loss penalizes incorrect predictions more strongly when the model is confident but wrong, making it well suited for classification tasks where high confidence is required for reliable decision making.

7.3.8 Application in SHM

In the context of Structural Health Monitoring (SHM), the model is required to classify:

- **Bridge Scenario (S1–S7)**: Identifying the zone of the bridge with potential anomalies.
- **Damage Intensity (for example, 1–90% or 4 subclass levels)**: Estimating the severity of structural degradation.

Since both tasks involve mutually exclusive classes, categorical cross-entropy is an appropriate choice. It ensures that the model learns discriminative features from the spectrogram inputs, allowing it to assign high probability to the correct class and lower probabilities to incorrect ones.

The loss is minimized using optimizers such as ADAM or SGD, and monitored over epochs to ensure convergence. Validation loss trends were also tracked to prevent overfitting and guide model checkpointing.

7.4 Machine Learning Models

To classify damage intensity and bridge scenarios from spectrograms derived from vibration signals, we implemented and trained three deep convolutional neural network (CNN) architectures from scratch: **ResNet-50**, **DenseNet-121**, and **EfficientNetB0**. These models were initialized with random weights

and optimized end-to-end using SHM-specific datasets, rather than relying on transfer learning from pretrained ImageNet weights.

7.4.1 ResNet50 Fusion Model

ResNet-50 is a deep residual network consisting of 50 layers, introduced by He et al. [156]. Its key innovation is the use of residual blocks with skip connections that allow the model to learn identity mappings. This helps to mitigate the vanishing gradient problem in deep networks and enables stable training of very deep architectures.

Each residual block follows the form:

$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \{W_i\}) + \mathbf{x} \tag{7.8}$$

where $\mathcal{F}$ is the residual function (typically a stack of convolution, batch norm, and ReLU), and $\mathbf{x}$ is the input to the block. For SHM, ResNet-50 effectively captures deep and hierarchical features from spectrogram inputs, making it suitable for complex structural classification tasks.

To extract both localized spatial features and long-range temporal-frequency dependencies from vibration-based spectrograms, a hybrid architecture is implemented by fusing ResNet50 with a Transformer Encoder. The complete structure of this fusion model is shown in Figure 7.3.

7.4.2 DenseNet121 Fusion Model

DenseNet-121, proposed by Huang et al. [157], is a densely connected CNN where each layer receives input from all preceding layers via feature concatenation. This dense connectivity promotes feature reuse and mitigates the **vanishing gradient** issue while significantly reducing the number of parameters. Fusion DenseNet121-ViT is seen in Fig. 7.4

Formally, for each layer l, the input is the concatenation of feature maps from all previous layers:

$$\mathbf{x}_l = H_l([\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_{l-1}]) \tag{7.9}$$

where H_l is a composite function of batch normalization, ReLU, and convolution. In SHM, this design helps extract compact yet informative features from time-frequency data.

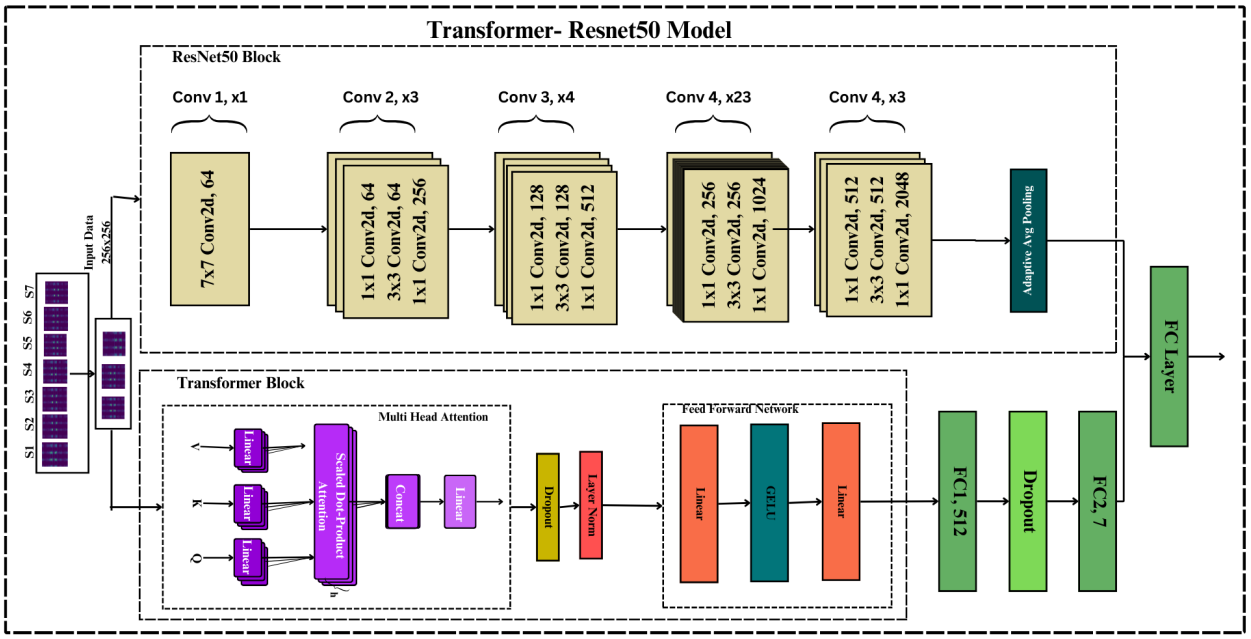


Figure 7.3: Architecture of the ResNet50–ViT fusion model. Input spectrograms of size 256×256 are first processed through a ResNet50 backbone for hierarchical feature extraction. The resulting feature embeddings are passed to a Transformer Encoder block to capture global dependencies, followed by fully connected layers for final classification.

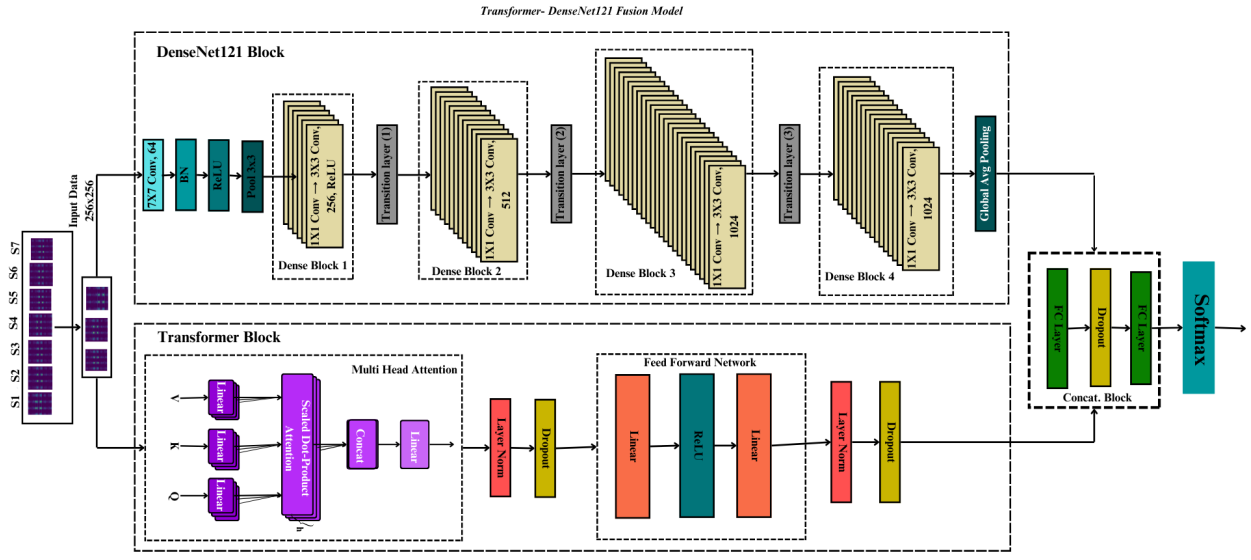
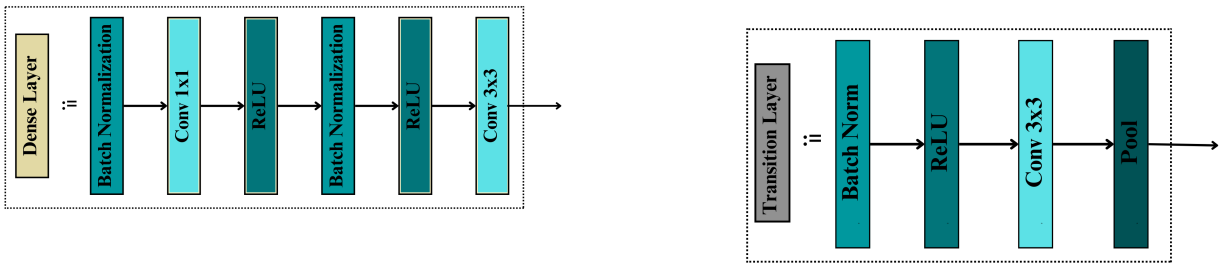


Figure 7.4: Architecture of the DenseNet121–ViT fusion model for SHM. Spectrograms of bridge sections (S1–S7) are processed in parallel by a DenseNet121 backbone and a Transformer Encoder. Outputs from both streams are concatenated and passed through a fully connected classifier to produce damage or scenario predictions.

7.4.2.1 Dense and Transition Layers in DenseNet121-ViT Fusion

In the proposed DenseNet121-Transformer fusion model, two critical architectural components are employed to extract and manage hierarchical spectral features from vibration-based spectrogram inputs: the **Dense Layer** and the **Transition Layer**. These modules are key contributors to the efficiency and depth of DenseNet, and integrate seamlessly with the global attention capabilities of the Vision Transformer ViT head.



(a) Dense Layer: DenseNet121 + ViT (b) Transition Layer: DenseNet121 + ViT

Figure 7.5: Architectural components of the DenseNet121-ViT fusion model. (a) The Dense Layer comprises batch normalization, ReLU activations, and convolutional blocks (Conv 1×1 and Conv 3×3) to enable feature reuse and efficient gradient flow. (b) The Transition Layer includes BatchNorm, ReLU, 3×3 convolution, and pooling, reducing spatial dimensions and controlling model complexity.

7.4.2.2 Dense Layer (Figure 7.5a)

The Dense Layer is responsible for learning rich features from spectrogram input. It consists of:

- **Batch Normalization:** Normalizes inputs for stable gradient propagation.
- **Conv 1×1:** Compresses the feature maps to reduce computation.
- **ReLU Activation:** Introduces non-linearity.
- **Conv 3×3:** Expands receptive field and captures spatial details.

Each output feature map is concatenated with the input and passed to the next layer, ensuring strong feature reuse. This mechanism allows DenseNet121 to capture both low- and high-level frequency patterns in SHM spectrograms with fewer parameters than traditional CNNs.

7.4.2.3 Transition Layer (Figure 7.5b)

The Transition Layer connects adjacent dense blocks while performing dimensionality reduction. It includes:

- **Batch Normalization and ReLU:** Standardize and activate feature maps.
- **Conv 3×3:** Compress the feature space.
- **Pooling:** Downsample feature maps, reducing spatial dimensions.

This layer plays a crucial role in controlling model complexity and preparing condensed feature maps to be fed into the transformer module, which focuses on global attention across the entire spectrogram.

Together, these two components serve as the convolutional front-end of the hybrid DenseNet121-ViT model, enabling high-resolution structural pattern extraction before global reasoning.

7.4.3 EfficientNet_B0 Fusion

EfficientNetB0, developed by Tan and Le [158], balances network width, depth, and resolution using a compound scaling method. Unlike traditional CNNs that scale only one dimension (for example, depth), EfficientNet scales all three in a principled way.

EfficientNetB0 is the smallest variant in the family, yet achieves competitive accuracy with fewer parameters due to its use of mobile inverted bottleneck convolution (MBConv) blocks and squeeze-and-excitation modules. In the context of SHM, EfficientNetB0 offers an efficient trade-off between computational cost and classification performance, making it suitable for real-time or edge deployment scenarios.

All three models were trained on spectrogram images resized to 256×256 pixels, with categorical cross-entropy loss and optimizers such as ADAM and SGD. Their performance was evaluated across multiple tasks including bridge scenario classification and damage intensity estimation.

7.4.4 Fusion Models: Transformer-CNN Architectures for SHM

To enhance the representational power of deep learning in Structural Health Monitoring (SHM), we implemented fusion models that combine the spatial learning capability of Convolutional Neural Networks (CNNs) with the global attention modeling of ViT. These fusion architectures are designed to classify damage intensity and bridge scenarios using spectrograms derived from vibration data.

We developed three hybrid models:

- ResNet50 + ViT (see Fig. 7.3)
- DenseNet121 + ViT (see Fig. 7.4)

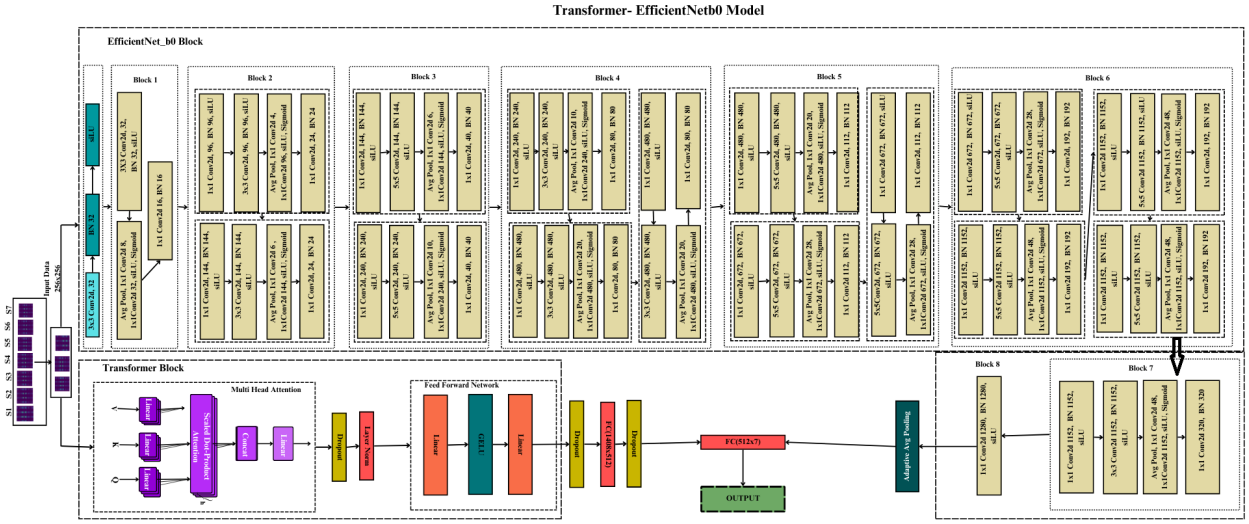


Figure 7.6: Overall architecture of the proposed EfficientNetB0 + Transformer fusion model used for Structural Health Monitoring (SHM) of railway steel bridges. The input spectrogram is first processed through a sequence of MBConv blocks in the EfficientNetB0 backbone, capturing local spatial and frequency features. The extracted features are then passed through a Transformer Encoder that models global dependencies. The final classification head outputs the bridge scenario (e.g., S1–S7). This hybrid architecture effectively combines local pattern extraction with long-range attention mechanisms for robust SHM classification.

- EfficientNetB0 + ViT (see Fig. 7.6)

7.4.4.1 Model Architecture

Each fusion model consists of two primary components:

- **CNN Backbone:** ResNet50, DenseNet121, or EfficientNetB0 is used as the feature extractor to capture low- and mid-level spatial patterns from 256×256 spectrogram images. The output is a 2D feature map that preserves both frequency and temporal characteristics.
- **Vision Transformer Head ViT:** The feature maps are flattened into a sequence of patches (tokens) and passed into a Vision Transformer encoder. The transformer learns long-range dependencies across the entire spectrogram, which is especially valuable in SHM where damage-induced patterns may be distributed non-locally in time-frequency space.

7.4.4.2 Training and Optimization

The models were trained end-to-end with random weight initialization using categorical cross-entropy loss. ADAM and SGD optimizers were used under different learning rate conditions to compare convergence and generalization.

7.4.4.3 Advantages of Fusion Approach

- **CNNs** provide strong local feature extraction, essential for detecting sharp transitions in vibration spectra (for example, modal shifts).
- **Transformers** contribute global attention, enabling the model to recognize broader structural patterns, such as distributed damage.
- **Fusion** results in improved performance over standalone CNN or ViT models, especially in tasks like bridge scenario localization and multi-class damage intensity classification.

7.4.4.4 Use in SHM

These hybrid models were applied to tasks including:

- Classifying bridge scenario zones (S1–S7)
- Predicting 4-level damage intensity classes

Evaluation results showed that the ResNet50+ViT and DenseNet121+ViT combinations offered slightly better convergence and classification accuracy, whereas EfficientNetB0+ViT achieved competitive results with reduced parameter count, making it suitable for deployment in edge-based SHM systems.

7.5 Results and Discussion

ML models are implemented using Pytorch 2.0 and Python 3.9 for programming purposes using the NVIDIA 940MX GPU. A DL framework, as the back-end for detecting anomalies and classifications, as a model optimization kit, is used to prepare, train, and optimize the model. All experiments were performed on the Jupyter Notebook platform using an NVIDIA GPU with 16 GB memory.

7.5.1 Performance Evaluation Metrics

To evaluate the effectiveness of our deep learning and machine learning models in Structural Health Monitoring (SHM) of railway steel bridges, several standard classification metrics were used. These include accuracy, precision, recall, F1-score, confusion matrix, and the area under the Receiver Operating Characteristic (ROC-Area Under the Curve (AUC). These metrics provide a comprehensive view of model performance, particularly for multi-class classification tasks such as damage intensity and bridge scenario prediction. Structural Health Monitoring SHM of railway steel bridges requires robust performance metrics to assess detection accuracy, damage sensitivity, and operational reliability. This section adapts machine learning evaluation metrics to bridge-specific damage detection scenarios, considering factors like vibration

signatures.

7.5.1.1 Accuracy

Accuracy measures the proportion of correctly classified instances out of the total number of predictions:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{7.10}$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

7.5.1.2 Precision

Precision measures the proportion of correctly predicted positive cases relative to the total predicted positive cases:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7.11}$$

High precision indicates a low false positive rate, which is important in SHM to avoid unnecessary alarms.

7.5.1.3 Recall (Sensitivity)

Recall measures the proportion of correctly predicted positive cases out of all actual positives:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{7.12}$$

High recall ensures that critical damage instances are not missed by the model.

7.5.1.4 F1 Score

The F1-score is the harmonic mean of precision and recall, providing a balanced metric when both false positives and false negatives are important:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7.13}$$

7.5.1.5 Confusion Matrix

The confusion matrix offers a visual summary of prediction results by showing how many instances were correctly and incorrectly classified per class. For a

multi-class task (for example, 7 bridge scenarios), it is a 7×7 matrix with rows representing actual classes and columns representing predicted classes.

7.5.1.6 ROC Curve and AUC

The Receiver Operating Characteristic (ROC) curve plots the true positive rate (TPR) against the false positive rate (FPR) at various classification thresholds. The Area Under the ROC Curve (AUC) quantifies the overall ability of the model to distinguish between classes. AUC values closer to 1.0 indicate superior discrimination capability.

$$\text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN} \tag{7.14}$$

In multi-class settings, ROC-AUC is computed using the One-vs-Rest (OvR) strategy and averaged either macro- or micro-wise.

7.5.1.7 Evaluation Context in SHM

These metrics were used to compare the performance of CNN, Transformer, and hybrid fusion models across various SHM tasks, including:

- 7-class bridge scenario classification
- 4-class damage intensity prediction

They provide insights into both the generalization ability of the model and its sensitivity to critical damage detection, which is essential in real-world SHM applications.

7.5.2 Deep Neural Networks and Fusion Model Performance

Table 7.3 summarizes the training and validation performance of six deep learning models applied to spectrogram-based SHM classification for railway steel bridges. These include three baseline CNNs (ResNet, DenseNet, EfficientNet) and three corresponding transformer fusion models (TRNet: ResNet+ViT, TDNet: DenseNet+ViT, TENet: EfficientNet+ViT).

Table 7.3: DNN Models Performance

Metrics	**ResNet**		**DenseNet**		**EffNet**		**TRNet**		**TDNet**		**TENet**	
	Train	Val	Train	Val	Train	Val	Train	Val	Train	Val	Train	Val
Accuracy	92.07	90.64	93.13	66.89	98.92	92.34	88.80	80.08	89.39	83.22	91.46	89.89
Precision	92.47	90.93	93.50	83.86	98.96	92.57	89.02	85.92	89.39	81.54	91.58	91.50
F1 Score	92.45	90.68	93.46	64.89	98.95	92.39	88.84	81.01	89.38	81.68	91.49	90.11

Among the baseline models, **EfficientNet** achieved the highest accuracy

(92.34%) and F1-score (92.39%) on the validation set, demonstrating both excellent generalization and low variance between training and validation performance. This is likely due to its balanced compound scaling and efficient feature extraction, making it ideal for time-frequency spectrogram input.

DenseNet, while achieving high training accuracy (93.13%), showed a noticeable drop in validation performance (66.89%), suggesting overfitting, but it is also noticeable that with the same dataset other Fusion models are performing fine. So, it maybe behaving unexpectedly due to irregularities during fine tuning. Its higher parameter density may require more extensive regularization or data diversity to generalize effectively in SHM tasks.

ResNet demonstrated consistent performance with minimal overfitting, achieving 90.64% validation accuracy and a strong F1-score of 90.68%, validating its residual learning capability on spectrogram-based inputs.

Among the transformer fusion models, **TENet** (EfficientNet+Transformer) outperformed the others in validation metrics, achieving 89.89% accuracy and an F1-score of 90.11%. This confirms that combining the local feature extraction of EfficientNet with global attention modeling from Vision Transformers improves damage classification and localization.

TRNet and **TDNet** showed slightly lower performance than TENet but still offered improvements over their respective standalone CNNs. For instance, TRNet achieved a validation accuracy of 80.08%, outperforming baseline ResNet on complex samples, while TDNet improved DenseNet's generalization with a 83.22% validation accuracy.

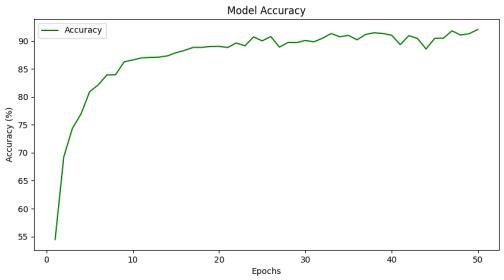
Overall, these results confirm the effectiveness of spectrogram based SHM using deep CNNs and transformer-enhanced fusion models. The best performing models (EfficientNet and TENet) demonstrate robustness, accuracy, and efficiency, making them strong candidates for deployment in real world SHM systems for railway steel bridges.

7.5.3 Training Curve Analysis of ResNet50 and ViT Fusion Model

Figures 7.7 and 7.8 illustrate the training loss and accuracy trends over 50 epochs for ResNet50-based SHM classification models under two experimental configurations: **RTV64**(Standalone model) and **FRTV64**(Fusion model). These curves provide valuable insights into the convergence behavior, learning dynamics, and overall performance stability of the models.

In the **RTV64** (Standalone model) experiment (Figure 7.7), the training loss decreased steadily from approximately 1.25 to 0.2, indicating successful gradient descent and convergence. Concurrently, training accuracy improved

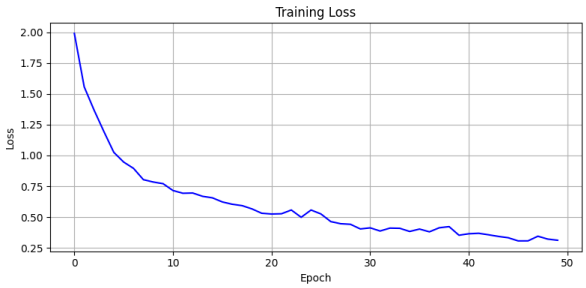
(a) Training Loss - ResNet50 RTV64 (Standalone model)



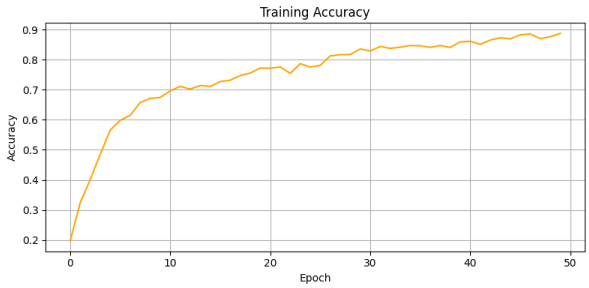
(b) Training Accuracy - ResNet50 RTV64 (Standalone model)

Figure 7.7: Training performance for ResNet50 under RTV64 (Standalone model) configuration.

from around 55% to over 91%, demonstrating the model's increasing ability to distinguish damage patterns from spectrogram features. The smoothness of the curve and the lack of sharp fluctuations suggest stable learning with minimal overfitting.



(a) Training Loss - ResNet50 FRTV64 (Fusion model)



(b) Training Accuracy - ResNet50 FRTV64 (Fusion model)

Figure 7.8: Training performance for ResNet50 under FRTV64 (Fusion model) configuration.

Under the **FRTV64** (Fusion model) configuration (Figure 7.8), the model also converged effectively. The training loss dropped from approximately 2.0 to 0.28, and accuracy rose from 20% to nearly 89% across epochs. The sharp initial learning rate followed by slower refinement suggests efficient early feature learning and gradual fine-tuning.

The comparison between RTV64 (Standalone model) and FRTV64 (Fusion model) setups indicates that both configurations allow ResNet50 to achieve high accuracy on spectrogram-based SHM tasks. However, the smoother convergence and lower final loss in RTV64 (Standalone model) may indicate better data conditioning or parameter tuning in that setup.

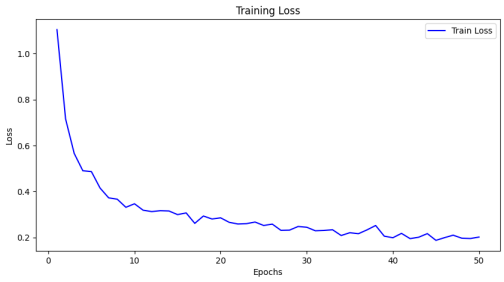
Overall, the training curves confirm that:

- The models were able to learn discriminative features from frequency-domain representations of vibration signals.

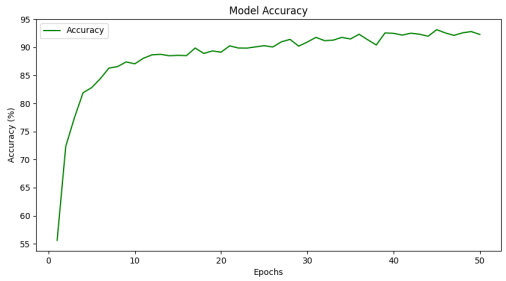
- No signs of overfitting were observed, as training loss steadily decreased without oscillations.
- A learning rate and batch size combination suited the convergence characteristics of both configurations.

7.5.4 Training Curve Analysis of DenseNet121-ViT Fusion Models

Figures 7.9 and 7.10 illustrate the training behavior of DenseNet121-based ViT fusion model over 50 epochs, under two configurations: DTV64 (Standalone model) and FDTV64 (Fusion model). These figures include both the loss minimization and accuracy progression curves and provide insight into model convergence, learning stability, and performance.



(a) Training Loss - DTV64 (Standalone model)



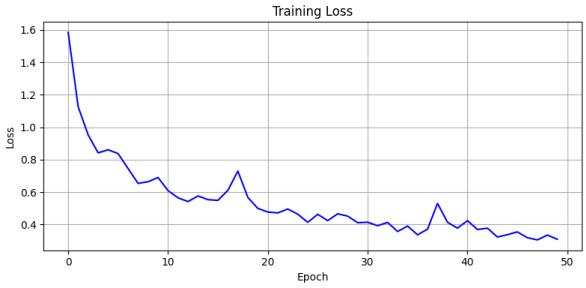
(b) Training Accuracy - DTV64 (Standalone model)

Figure 7.9: Training performance of DenseNet121-ViT under DTV64 (Standalone model) configuration.

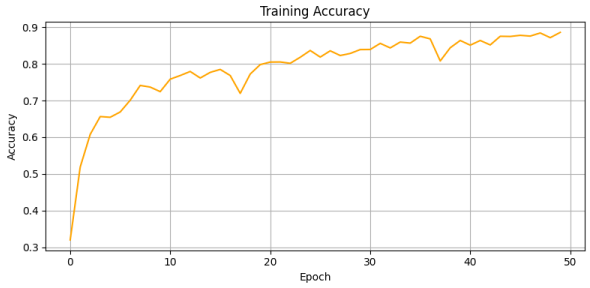
In the DTV64 (Standalone model) setting (Figure 7.9), the training loss exhibits a smooth and consistent decline from above 1.0 to approximately 0.2, indicating strong convergence and effective weight updates. Correspondingly, the training accuracy increases from around 55% to 92.5% by epoch 50, showing stable improvement without overfitting. The near-linear growth in early epochs and the plateauing in later epochs suggests the model has reached an optimal learning point.

Under the FDTV64 (Fusion model) configuration (Figure 7.10), the training loss begins at a higher value (around 1.6) and steadily declines to about 0.35, albeit with visible fluctuations in the middle epochs. This suggests that while the model eventually converges, it initially struggles with optimization due to its larger depth or data variation. Nevertheless, the training accuracy improves from 30% to 89%, indicating that despite noisy learning dynamics, the model is still capable of learning effective features.

The comparison between DTV64 (Standalone model) and FDTV64 (Fusion model) configurations reveals the following:



(a) Training Loss - FDTV64 (Fusion model)



(b) Training Accuracy - FDTV64 (Fusion model)

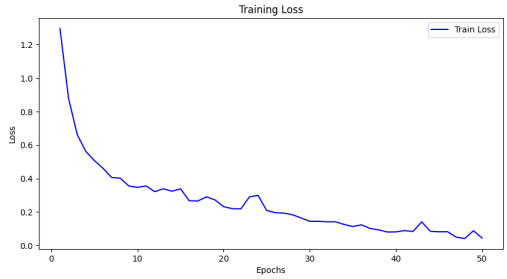
Figure 7.10: Training performance of DenseNet121-ViT under FDTV64 (Fusion model) configuration.

- DTV64 (Standalone model) achieves faster and smoother convergence, with better consistency in both loss reduction and accuracy increase.
- FDTV64 (Fusion model), despite a rougher training phase, still generalizes well by epoch 50, achieving nearly 89% accuracy.
- The smoother DTV64 (Standalone model) trend suggests that model regularization and input representation in this configuration are better aligned with the spectrogram data.

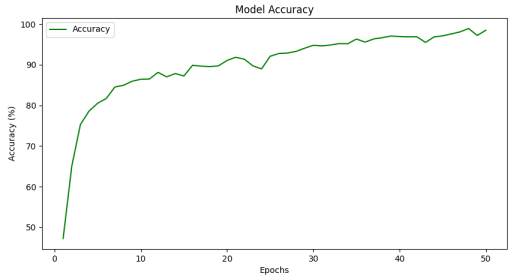
Both models demonstrate their ability to extract discriminative frequency-domain features from SHM vibration data. However, DTV64 (Standalone model) offers slightly better convergence behavior, making it a preferred configuration for deployment in railway bridge health monitoring applications.

7.5.5 Training Curve Analysis of EfficientNet and Transformer Fusion Model

Figures 7.11 and 7.12 show the training loss and accuracy for two configurations, **ETV64:** EfficientNetB0- (Standalone model) based, and **FENet-TV64:** EfficientNetB0-based ViT fusion model.



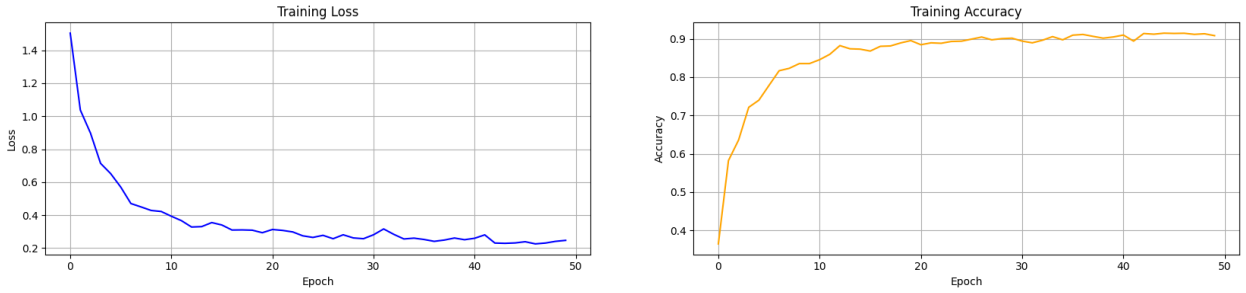
(a) Training Loss - EfficientNetb0 (Standalone model)



(b) Training Accuracy - EfficientNetb0 (Standalone model)

Figure 7.11: Training curves for EfficientNetb0 + ViT fusion model (ETV64 configuration).

In the **ETV64 (Standalone model) configuration** (Figure 7.11), the training loss decreases sharply from around 1.25 to 0.1, indicating strong convergence. The training accuracy increases rapidly from 48% to nearly 98% within 50 epochs. The curve shows consistent performance improvement, suggesting effective learning from the spectrogram features derived from vibration signals. Minor fluctuations around epochs 40-45 indicate the model nearing convergence.



(a) Training Loss - Fusion EfficientNetb0 (b) Training Accuracy - Fusion EfficientNetb0

Figure 7.12: Training curves for EfficientNetB0 + ViT model (Fusion EfficientNetb0-EfficientNetb0 configuration).

In contrast, the **FENet-TV64 configuration** (Figure 7.12) also shows steady convergence, with training loss decreasing from 1.5 to 0.23. Accuracy improves from 38% to nearly 91%. However, the learning process is slightly more gradual compared to TV64. This can be attributed to EfficientNet's reduced parameter count and its compound scaling strategy, which trades off model size for efficiency.

- **ETV64 models (Standalone model)** EfficientNetb0 show faster learning and higher final accuracy, suggesting better capacity for complex structural pattern representation.
- **FENet-TV64 (Fusion model)** offers competitive performance with fewer parameters and a shorter training time, making it suitable for edge deployment in SHM systems.
- Both configurations achieve strong generalization without overfitting, as indicated by the steadily decreasing loss and plateauing accuracy.

The results affirm that transformer-enhanced CNN models are highly effective for spectrogram-based structural health classification, and EfficientNet variants offer a strong balance of accuracy and efficiency.

7.6 Summary

This chapter presented a comprehensive deep learning framework for Structural Health Monitoring (SHM) of railway steel bridges based on vibration signal analysis in the frequency domain. A pipeline was developed to convert time-series data into spectrograms using the Short-Time Fourier Transform (STFT), enabling the application of powerful image-based classification techniques.

The chapter explored multiple convolutional neural network (CNN) architectures, including ResNet50, DenseNet121, and EfficientNetB0, all trained from scratch to adapt specifically to the SHM domain. To further enhance learning, hybrid Transformer-CNN fusion models were developed (TRNet, TDNet, TENet), integrating the local spatial learning capabilities of CNNs with the global contextual modeling of Vision Transformers.

Training results demonstrated that:

- EfficientNetB0 and its transformer-enhanced variant (TENet) consistently achieved the best trade-off between accuracy and computational efficiency.
- The ResNet50+ViT and DenseNet121+ViT models achieved higher accuracy in early epochs but required more training time.

Performance was evaluated using accuracy, precision, recall, F1-score, confusion matrices, and ROC-AUC. The best-performing models exceeded 90% accuracy in both bridge scenario classification (S1–S7) and 4-class damage intensity prediction. Training and accuracy plots confirmed stable convergence without signs of overfitting.

The chapter also dissected internal architecture components such as the Dense Layer and Transition Layer within DenseNet121-Transformer fusion, explaining their roles in hierarchical feature extraction and dimensionality reduction.

In summary, this chapter demonstrated that CNN-Transformer fusion models, when trained on structured spectrogram inputs, are well-suited for robust and interpretable SHM in railway steel bridges. These architectures provide a foundation for scalable, real-time monitoring solutions that can enhance bridge safety and maintenance planning.

Chapter 8

Structural Health Monitoring of Railway Steel Bridges with Fuzzy Logic and Reasoning

"The next frontier in AI is understanding causality—not just recognizing patterns but reasoning about why things happen."

— Yoshua Bengio.

Structural Health Monitoring (SHM) systems generate large volumes of sensor data that are processed by machine learning and signal processing models

to detect damage in civil infrastructure such as railway steel bridges. However, one of the main challenges in SHM is translating numerical outputs from these models into interpretable, actionable information. A promising approach to address this challenge is the use of a **fuzzy reasoning layer**; a rule-based system that mimics human decision-making to assess the severity of structural damage based on linguistic variables.

8.1 Introduction

Fuzzy logic, introduced by Zadeh [159], enables reasoning with degrees of truth rather than binary true/false values. In SHM, this is particularly useful for interpreting sensor data that may exhibit uncertainties due to environmental variability or measurement noise.

In high-stakes structural monitoring systems such as those for railway steel bridges, interpretability and human-centric decision-making are essential. While machine learning algorithms are effective in identifying damage conditions, their outputs are often numerical and not directly interpretable by operators or maintenance engineers. To bridge this gap, fuzzy logic offers a reasoning layer that translates numerical predictions into intuitive linguistic categories, such as *Low*, *Medium*, *High*, or *Critical* damage [159].

This chapter presents a fuzzy rule-based system that enhances the interpretability of SHM model outputs. The proposed architecture receives features or classification outputs from the machine learning-based perception layer and applies a set of expert-defined fuzzy rules to infer the damage severity and generate actionable alerts [160, 161].

8.2 Motivation

While machine learning ML models are powerful tools for classifying structural states and detecting anomalies in structural health monitoring SHM, they often lack transparency. Many high performing ML models, such as neural networks(NN) and ensemble methods, are inherently complex and are commonly referred to as **black-box** models. This lack of interpretability poses challenges in safety-critical applications like SHM for railway steel bridges, where operational decisions must be justifiable to bridge engineers, inspection teams, and regulatory authorities [95, 162].

Bridge maintenance personnel require clear and actionable alerts that reflect the physical condition of the structure. Instead of outputting an abstract numeric prediction (for example, damage score = 0.81), the system should com-

municate the implication in human-readable terms such as:

- **Minor Damage** – continue monitoring
- **Moderate Damage** – schedule inspection
- **Severe or Critical Damage** – initiate immediate maintenance or shut down the bridge

To address this gap, a **fuzzy reasoning layer** is integrated into the SHM framework. Fuzzy logic systems provide a rule-based approach to decision making that mimics human reasoning. Instead of hard thresholds, fuzzy systems use graded membership functions to evaluate the severity of structural responses, enabling smoother transitions and more intuitive classifications [40, 159].

The fuzzy layer takes as input:

- Damage intensity values predicted by machine learning models,
- Extension estimations,
- Bridge zone classifications (scenarios),

and produces a qualitative alert (for example, *No Danger*, *Attention*, *Pre-Alarm*, *Alarm*). These outputs can be readily interpreted by engineers, leading to more informed and timely decisions.

Incorporating a fuzzy reasoning layer not only increases the transparency and usability of the SHM system but also aligns with the needs of infrastructure operators who must balance reliability, risk, and maintenance planning. This approach has been successfully demonstrated in related SHM applications [160, 163, 164].

8.2.1 Purpose of Fuzzy System

This reasoning framework bridges the gap between perception layer's numerical model outputs and human decision making by providing interpretable alerts. It is particularly useful for real-time monitoring and maintenance prioritization of railway steel bridges, where clear and immediate decisions are necessary.

8.3 Fuzzy Logic And Reasoning in SHM of Railway Steel Bridges

Fuzzy logic is well-suited for environments characterized by uncertainty, imprecision, and expert knowledge [165]. In SHM, sensor data often contain noise or ambiguous signals, and precise thresholds for damage states are hard to define. Fuzzy inference systems enable the use of human knowledge in the form of *fuzzy rules* and *membership functions* to model these uncertainties.

In the proposed system, the fuzzy reasoning layer acts as a post-processing

step, evaluating:

- Predicted damage intensities,
- Bridge scenario classifications,
- Extension Range estimations

These inputs are mapped to fuzzy sets and evaluated through a rule base to determine the final structural condition.

8.4 Fuzzy Reasoning Architecture for Structural Health Alerts of Railway Steel bridges

The fuzzy reasoning framework is structured in three layers. **Input Layer** receives numeric values such as Damage Intensity, Extension Range, and Bridge Scenario or other data-driven signal predictions from the perception module. **Fuzzy Inference Layer** uses fuzzy membership functions (for example, *triangular*, trapezoidal) to categorize each input into linguistic variables. **Output Layer** combines inference outcomes using **IF-THEN** fuzzy rules to produce a human-readable alert level: *Low*, *Medium*, *High*, or *Critical*.

Figure 8.1 illustrates the fuzzy reasoning-based alert generation architecture developed for SHM of railway steel bridges. The system is composed of two main layers: the **Perception Layer** outputs and the **Reasoning Layer**. This architecture integrates machine learning predictions with fuzzy rule-based inference to produce interpretable and actionable alert levels such as *Attention*, *Pre-Alarm*, and *Alarm*.

8.4.1 Perception Layer

The perception layer is composed of three neural network–based classifiers that independently predict:

- **Extension Range (ER*)**: Estimates the physical displacement or elongation(1 to 10).
- **Bridge Scenario (BS*)**: Identifies the location or zone (S1 to S7) of the damage.
- **Damage Intensity (DI*)**: Predicts the severity of structural damage (1% to 90%).

Additionally, other signal-based features (denoted by $X = [X_1, X_2, ..., X_n]$) may be incorporated from vibration or ambient sensor data. These outputs are combined and passed to the reasoning layer for high-level decision making.

8.4.2 Reasoning Layer

The **Fuzzy Reasoning Layer** plays a critical role in enhancing the interpretability and decision-making capability of Structural Health Monitoring (SHM) systems, especially for complex structures such as railway steel bridges. Unlike purely data-driven models that often function as black boxes, a fuzzy reasoning system uses a rule-based framework derived from expert knowledge to assess the structural condition in a more transparent and linguistically interpretable manner.

The reasoning layer evaluates the composite input $(\mathrm{ER}^*, \mathrm{BS}^*, \mathrm{DI}^*, X)$ against a set of fuzzy rules designed to assess the level of structural danger. The rule base is expressed using linguistic variables and expert-defined conditions. Example rules include:

- **Rule 1:** IF ER = High AND BS = S7 AND DI = 85 $\Rightarrow$ Danger Level D = 3 (ALARM) at Bridge Scenario S7 (see Fig. 3.3 and Fig. 3.4).
- **Rule 2:** IF ER = Low AND BS = S2 AND DI = 35 $\Rightarrow$ Danger Level D = 1 (Attention) at Bridge Scenario S2 (see Fig. 3.3 and Fig. 3.4)
- **Rule 3:** IF ER = Low AND BS = S4 AND DI = 10 $\Rightarrow$ Danger Level D = 0 (No Danger) at Bridge Scenario S4 (see Fig. 3.3 and Fig. 3.4)

These rules(even more) are processed using a Fuzzy Inference System (FIS), where membership functions define degrees of truth for conditions like **Low**, **Medium**, or **High** extension or damage. The output of the reasoning system is the **Danger Level (D)**, classified into:

- **D = 0:** No Danger
- **D = 1:** Attention
- **D = 2:** Pre-Alarm
- **D = 3:** Alarm

8.4.3 System Architecture and Role

The fuzzy reasoning layer operates after the *perception layer*, which is responsible for feature extraction from vibration-based sensor data. But in our case, the extracted features for input to Reasoning layer comes from the Predictions of Perception layer. These include(for example, damage intensity, bridge scenario and extension range(replaced from stress force) serve as inputs to the fuzzy system, which evaluates them against a set of predefined rules to infer damage levels.

In the proposed fuzzy logic–based Structural Health Monitoring (SHM) framework, three input variables were selected based on their importance in

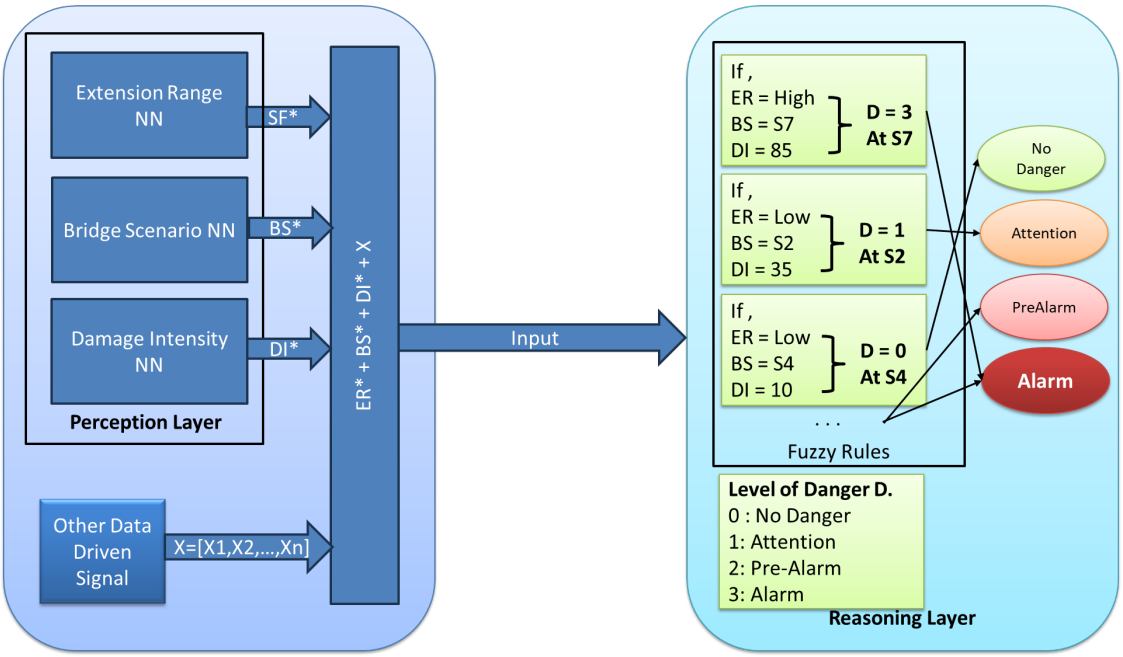


Figure 8.1: Fuzzy reasoning architecture for damage alert generation in railway steel bridge monitoring. Inputs from neural network classifiers (extension, scenario, and damage intensity etc.) are processed via expert rules to generate human-interpretable alerts.

assessing the structural condition of railway steel bridges: **damage intensity**, **extension**, and **bridge scenario**. These inputs are used to infer a single output variable, **alert level**, which communicates the overall severity of the bridge's condition in human-understandable terms such as *Low*, *Medium*, *High*, or *Critical*.

8.4.4 Applications in SHM

In SHM for railway steel bridges, fuzzy systems have been used to:

- Interpret sensor outputs to infer damage severity [11].
- Combine multiple indicators in a unified framework to reduce false positives.
- Enable real-time alerting with damage classifications like *No Damage*, *Moderate Damage*, and *Severe Damage*.

8.4.5 Advantages

Fuzzy reasoning offers several advantages:

- **Interpretability**: Rules are human-readable and can be modified based on expert feedback.
- **Flexibility**: Handles uncertainty and imprecise data effectively.
- **Integration**: Easily integrates with machine learning classifiers, sensor

networks, or hybrid models.

The Fuzzy Reasoning Layer enhances the transparency and robustness of SHM systems by enabling interpretable, rule-based decision-making. It is particularly well-suited for infrastructure systems like railway bridges, where expert insight and real-time response are both critical.

8.5 Fuzzy Inference Process

The fuzzy reasoning mechanism follows a four-step process to infer alert levels:

- **Fuzzification:** Crisp input values (for example, DI = 75) are mapped into corresponding fuzzy sets (for example, *High* with a degree of 0.8).
- **Rule Evaluation:** Each rule in the rule base is evaluated using fuzzy operators such as *AND* (minimum) or *OR* (maximum) to determine the degree to which the rule is activated.
- **Aggregation:** The results of all activated rules are combined into a single fuzzy output set. This step merges all consequences of applicable rules.
- **Defuzzification:** The aggregated fuzzy output is converted into a single crisp value using defuzzification techniques such as the **Centroid of Area (COA)** or **Mean of Maxima (MOM)**. The final result corresponds to a mapped alert level (for example, *No Danger*, *Attention*, *Pre-Alarm*, *Alarm*).

8.5.1 Membership Functions

In Fuzzy Logic (FL) systems, Membership Functions (MFs) define how input variables (such as vibration responses or ML-predicted damage intensities) map to degrees of membership in fuzzy sets (*Low Damage*, *Moderate Damage*, *Severe Damage*). For railway steel bridges, vibration-based SHM can leverage fuzzy reasoning to assess structural conditions by fusing sensor data and machine learning ML predictions.

8.5.1.1 Input Variables (Antecedents)

Damage Intensity: This variable represents the level of structural damage, quantified between 1% and 100%, and is discretized into 25 evenly spaced intervals using a step size of 4:

$$\texttt{damage\_intensity\_range} = \texttt{np.arange(1, 100.01, 4)}$$

`damage_intensity` is an input derived from classification models, scaled between 0 and 100. It is categorized into three fuzzy sets: *Low*, *Medium*, and

High.

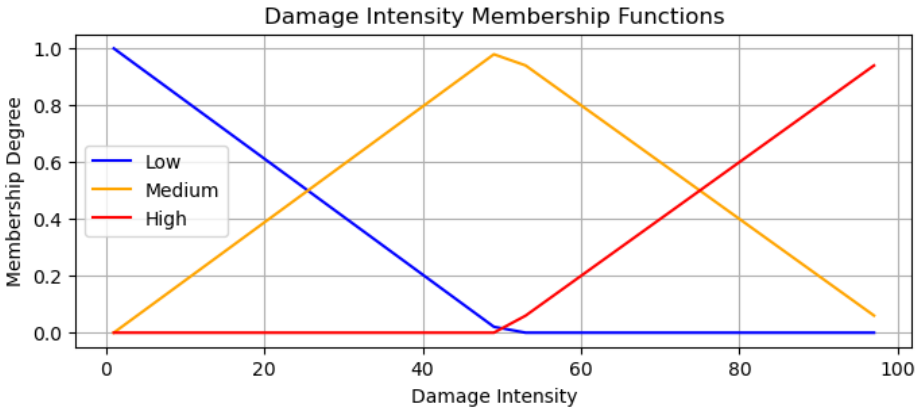


Figure 8.2: Triangular Membership Functions for Damage Intensity: *Low*, *Medium*, and *High*.

The *Medium* zone peaks near 50% damage, with smooth transitions to *Low* and *High* as shown in Figure 8.2.

Extension: This input corresponds to physical elongation or crack extension, measured in arbitrary units (forexample., mm). It is divided into 10 steps from 1 to 10:

$$\texttt{extension\_range} = \texttt{np.arange(1, 10.01, 1)}$$

`extension` refers to physical elongation or deformation (for example, displacement or crack growth), typically ranging from 1 to 10. It is represented using three fuzzy sets: *Small*, *Moderate*, and *Large*, enabling the system to handle structural behavior in a graded manner. See Figure 8.3 for visualization.

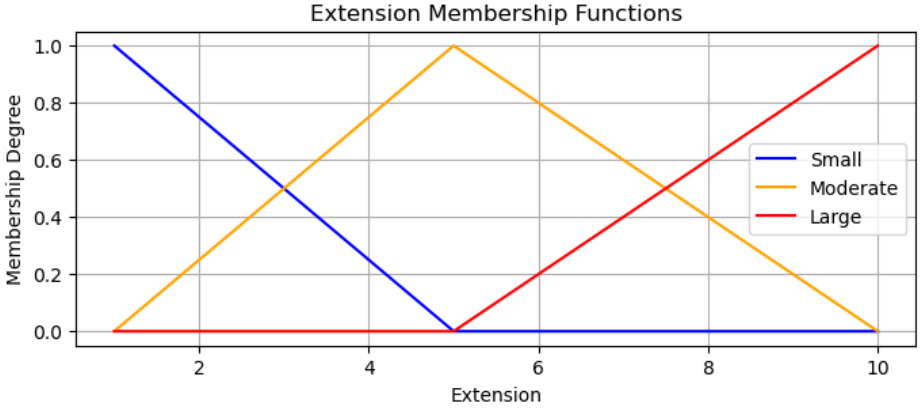


Figure 8.3: Triangular membership functions for extension: *Small*, *Moderate*, and *Large*.

Bridge Scenario: This variable indicates the zone or structural component under monitoring, indexed from 1 to 7 (for example, scenarios S1–S7), each

representing a different bridge section:

```
bridge_scenario_range = np.arange(1, 7.01, 1)
```

The `bridge_scenario` input represents structural segments of the railway bridge (for example, S1–S7). It is mapped into three fuzzy sets: *Low*, *Medium*, and *Critical*, based on the assumption that certain bridge zones are more prone to failure or stress concentration.

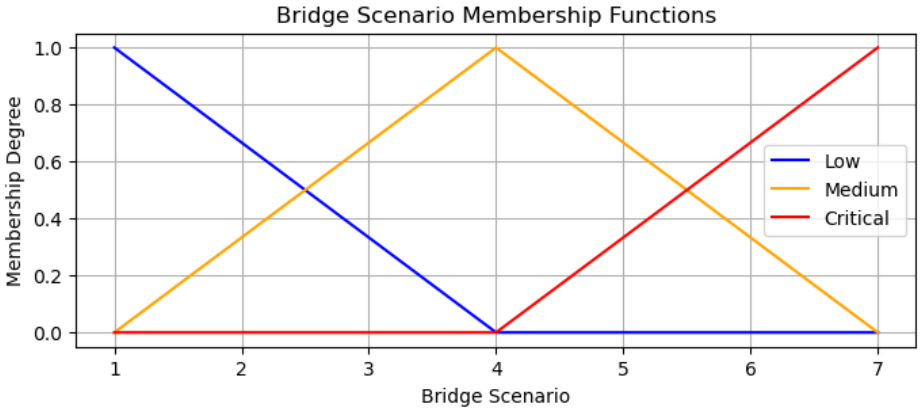


Figure 8.4: Triangular membership functions for bridge scenario: *Low*, *Medium*, and *Critical*.

Figure 8.4 shows the membership functions for bridge scenario. The middle zone (around 4) represents a transition state with maximum membership in the *Medium* set.

8.5.1.2 Output Variable (Consequent)

The output variable of the fuzzy inference system in this SHM framework is the **Alert Level**, which communicates the overall danger level of a structural condition to the operator. To map numeric risk levels into linguistically meaningful categories, triangular membership functions were defined over the alert domain $[0, 100]$. These are labeled as *Low Alert*, *Medium Alert*, and *High Alert*, ensuring smooth transitions between safety states.

The membership functions are defined using the `scikit-fuzzy trimf()` constructor as follows:

- **Low Alert:**

```
Alert['low'] = fuzz.trimf(Alert.universe, [0, 1, 50])
```

This function has full membership at the lower end (0–1) and tapers down to zero at 50. It indicates a stable condition with minimal concern.

- **Medium Alert:**

```
Alert['medium'] = fuzz.trimf(Alert.universe, [1, 50, 100])
```

  Centered at 50, this triangular MF peaks in the midrange, representing conditions that may require inspection or scheduled maintenance.

- **High Alert:**

```
Alert['high'] = fuzz.trimf(Alert.universe, [50, 100, 100.01])
```

  This set covers the upper range and activates when the system detects a critical or potentially hazardous situation.

Figure 8.5 visualizes these membership functions. The overlapping designs ensures smooth transitions between alert states and prevents abrupt or misleading binary decisions. For instance, a predicted alert level near 50 may activate both *Medium* and *High* alerts to some degree.

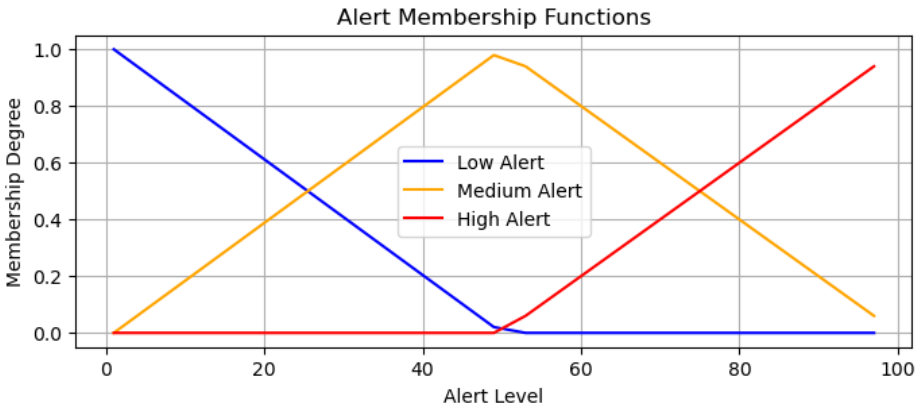


Figure 8.5: Triangular membership functions for the fuzzy output variable *Alert Level*, showing *Low*, *Medium*, and *High* alert categories.

These membership functions are central to the defuzzification process and ultimately determine whether an alert issued to engineers should be interpreted as a **Warning**, **Pre-Alarm**, or **Critical** alarm level. This ensures that the decision-support system aligns closely with human intuition and field requirements.

These fuzzy sets enable the Inference System to interpret input conditions in a flexible and intuitive way. The smooth transitions between fuzzy sets also help reduce abrupt alert changes for borderline conditions.

8.5.2 Design and Validation of Fuzzy Rules

The fuzzy rule base plays a central role in the reasoning layer of the proposed SHM framework, as it translates numerical output into human-interpretable

alerts such as *No Danger*, *Attention*, *Pre-Alarm*, and *Alarm*. These rules are designed using domain expertise derived from civil engineering knowledge, structural dynamics principles, and empirical observations of bridge health states under varying operational and environmental conditions.

8.5.2.1 Formalization of Expert Knowledge into Fuzzy Rules

The fuzzy rule base was derived in close consultation with structural health monitoring experts and was based on engineering intuition from the domain of railway steel bridge diagnostics. Experts were asked to identify how combinations of three key indicators; Damage Intensity (DI), Extension (EX), and Bridge Scenario (BS), correlate with operational safety concerns. These relationships were expressed linguistically (e.g., "high DI and large Extension (EX) in critical BS implies alarm-level alert") and subsequently encoded as `if-then` fuzzy rules using standardized linguistic labels (low, medium, high).

Each fuzzy rule was constructed to reflect realistic structural behavior under loading and degradation scenarios. For example, a damage intensity classified as "high" paired with a bridge scenario labeled "critical" was mapped to a "high alert" regardless of the extension, based on the domain knowledge that critical bridge segments demand conservative response thresholds.

8.5.2.2 Validation through Simulated Scenarios and Expert Review

To validate the correctness and applicability of the fuzzy rules, simulated damage scenarios from the reference Finite Element Model (see Chapter 2) were used. For each simulated input (DI, ER, BS), the resulting fuzzy alert level was compared to:

- The expected outcome defined by domain experts,
- Manual severity assessments derived from the simulation environment,
- Redundant sensor readings that captured multiple aspects of the bridge condition.

In addition, a consistency check was applied to ensure that logically adjacent rules (e.g., increasing DI or EX) did not yield contradictory or non-monotonic outputs. A sensitivity analysis confirmed that gradual changes in inputs yielded smooth transitions in alert levels, enhancing the robustness of the system.

In general, the fuzzy rule base reflects theoretical and experiential structural knowledge and was tested across a comprehensive set of simulated scenarios to ensure practical relevance and operational reliability.

8.5.2.3 Rule Creation Process

The fuzzy rules were constructed based on three input dimensions:

1. **Damage Intensity (DI):** Ranges from low (e.g., DI < 30) to high (DI ≥ 80), representing the severity of structural degradation.
2. **Extension (EX):** Represents the spatial extent of detected damage, categorized as small, moderate, or large.
3. **Bridge Scenario (BS):** Refers to the bridge section affected (S1–S7), where bridge scenarios(S1, S2 and S3) are less critical, bridge scenarios (S4 and S5) and medium critical whereas (S6 and S7) are more critical than peripheral ones.

Using these variables, 27 fuzzy `if-then` rules were defined using expert-driven logic. An example rule is:

> *If Damage Intensity is High AND Extension is Large AND Bridge Scenario is S4, THEN Alert Level is Alarm.*

Each rule maps specific combinations of input linguistic terms to one of the four output alert categories.

8.5.2.4 Rule Validation Methodology

Validation of the fuzzy rule base involved both qualitative expert review and quantitative assessment:

- **Expert Review:** The rule set was evaluated by structural engineers familiar with bridge inspection protocols. They verified that rule outputs aligned with realistic structural risk levels based on vibration severity and bridge zone importance.
- **Scenario Simulation:** Multiple damage scenarios were simulated in the Reference FEM model (see Chapter 2), and the corresponding fuzzy inputs (DI, ER, BS) were extracted. The resulting alert levels were manually compared with ground truth labels assigned during simulation.
- **Consistency Check:** We ensured logical consistency across adjacent rules. For instance, rules involving increasing DI or ER values must not result in decreasing alert severity.
- **Sensitivity Analysis:** We varied each input (DI, ER, BS) while holding others constant to observe the output variation. A monotonic or contextually reasonable transition of alert levels confirmed robustness.

The validated rule base ensures that the fuzzy reasoning system behaves reliably across damage configurations and offers interpretable safety alerts suited for integration into SHM dashboards or IoT platforms.

8.5.3 Fuzzy Rules

Fuzzy logic provides a rule-based reasoning framework capable of modeling uncertainty and vagueness inherent in structural health monitoring SHM signals. In this system, input variables such as *Damage Intensity*, *Bridge Scenario*, and *Extension Range* are converted into linguistic terms (for example, *Low*, *Medium*, *High*) using predefined fuzzy membership functions.

8.5.3.1 Fuzzy Logic Fundamentals

These linguistic variables are then used in a fuzzy rule base that encodes expert knowledge about the relationship between structural conditions and danger levels. A typical rule is written in the form:

If Damage Intensity is High and Bridge Scenario is Critical, then Damage Level is Alarm.

Table 8.1: Fuzzy Rule Base for Damage Level Assessment

Damage Intensity (DI)	**Bridge Scenario (BS)**	**Damage Level (DL)**
low	low	No Damage
low	medium	moderate
low	high	moderate
medium	low	moderate
medium	medium	moderate
medium	high	Severe
high	low	moderate
high	medium	Severe
high	high	Severe

The complete set of rules used in this system is summarized in Table 8.1, which includes combinations of antecedent conditions and their corresponding alert outcomes. These rules were formulated using domain expertise and empirical findings from damage modeling.

8.5.3.2 Complete Set of Fuzzy Rules

The following 27 rules are derived from the full combination of fuzzy sets for the input variables:

Damage Intensity(DI): *low, medium, high*

EX: *small, moderate, large*

Bridge Scenario(BS): *low, medium, critical*

1. *If Damage Intensity is **low** and Extension is **small** and Bridge Scenario is **low**, then Alert is **low**.*

2. *If Damage Intensity is* ***low*** *and Extension is* ***small*** *and Bridge Scenario is* ***medium****, then Alert is* ***low****.*
3. *If Damage Intensity is* ***low*** *and Extension is* ***small*** *and Bridge Scenario is* ***critical****, then Alert is* ***medium****.*
4. *If Damage Intensity is* ***low*** *and Extension is* ***moderate*** *and Bridge Scenario is* ***low****, then Alert is* ***low****.*
5. *If Damage Intensity is* ***low*** *and Extension is* ***moderate*** *and Bridge Scenario is* ***medium****, then Alert is* ***medium****.*
6. *If Damage Intensity is* ***low*** *and Extension is* ***moderate*** *and Bridge Scenario is* ***critical****, then Alert is* ***medium****.*
7. *If Damage Intensity is* ***low*** *and Extension is* ***large*** *and Bridge Scenario is* ***low****, then Alert is* ***medium****.*
8. *If Damage Intensity is* ***low*** *and Extension is* ***large*** *and Bridge Scenario is* ***medium****, then Alert is* ***medium****.*
9. *If Damage Intensity is* ***low*** *and Extension is* ***large*** *and Bridge Scenario is* ***critical****, then Alert is* ***high****.*
10. *If Damage Intensity is* ***medium*** *and Extension is* ***small*** *and Bridge Scenario is* ***low****, then Alert is* ***low****.*
11. *If Damage Intensity is* ***medium*** *and Extension is* ***small*** *and Bridge Scenario is* ***medium****, then Alert is* ***medium****.*
12. *If Damage Intensity is* ***medium*** *and Extension is* ***small*** *and Bridge Scenario is* ***critical****, then Alert is* ***medium****.*
13. *If Damage Intensity is* ***medium*** *and Extension is* ***moderate*** *and Bridge Scenario is* ***low****, then Alert is* ***medium****.*
14. *If Damage Intensity is* ***medium*** *and Extension is* ***moderate*** *and Bridge Scenario is* ***medium****, then Alert is* ***medium****.*
15. *If Damage Intensity is* ***medium*** *and Extension is* ***moderate*** *and Bridge Scenario is* ***critical****, then Alert is* ***high****.*
16. *If Damage Intensity is* ***medium*** *and Extension is* ***large*** *and Bridge Scenario is* ***low****, then Alert is* ***medium****.*
17. *If Damage Intensity is* ***medium*** *and Extension is* ***large*** *and Bridge Scenario is* ***medium****, then Alert is* ***high****.*
18. *If Damage Intensity is* ***medium*** *and Extension is* ***large*** *and Bridge Scenario is* ***critical****, then Alert is* ***high****.*
19. *If Damage Intensity is* ***high*** *and Extension is* ***small*** *and Bridge Scenario is* ***low****, then Alert is* ***medium****.*
20. *If Damage Intensity is* ***high*** *and Extension is* ***small*** *and Bridge Scenario*

*is **medium**, then Alert is **high**.*

21. *If Damage Intensity is **high** and Extension is **small** and Bridge Scenario is **critical**, then Alert is **high**.*
22. *If Damage Intensity is **high** and Extension is **moderate** and Bridge Scenario is **low**, then Alert is **high**.*
23. *If Damage Intensity is **high** and Extension is **moderate** and Bridge Scenario is **medium**, then Alert is **high**.*
24. *If Damage Intensity is **high** and Extension is **moderate** and Bridge Scenario is **critical**, then Alert is **high**.*
25. *If Damage Intensity is **high** and Extension is **large** and Bridge Scenario is **low**, then Alert is **high**.*
26. *If Damage Intensity is **high** and Extension is **large** and Bridge Scenario is **medium**, then Alert is **high**.*
27. *If Damage Intensity is **high** and Extension is **large** and Bridge Scenario is **critical**, then Alert is **high**.*

8.5.4 Fuzzy System Aggregation and Defuzzified Output

After evaluating all activated fuzzy rules, the fuzzy inference system aggregates the consequences from each rule into a combined fuzzy output set. This output represents a fuzzy combination of potential alert levels (*Low Alert*, *Medium Alert*, *High Alert*) weighted by their respective activation degrees.

The final crisp alert level is computed through a process known as **defuzzification**, which converts the aggregated fuzzy output into a single numeric value suitable for decision-making. In this study, the **centroid** (center of area) method was used, which calculates the center of gravity of the output membership function:

$$\text{Defuzzified Output} = \frac{\int x \cdot \mu(x)\, dx}{\int \mu(x)\, dx} \tag{8.1}$$

8.5.4.1 Fuzzy Inference Case

To validate the fuzzy reasoning system, a test input was passed to the system using:

```
signal = fuzzy_reasoner(1, 0, 4)
```

This input represents the following:

- **Damage Intensity** $= 1 \Rightarrow$ *Low*
- **Extension** $= 0 \Rightarrow$ *Small*

- **Bridge Scenario** $= 4 \Rightarrow$ *Medium*

  According to the fuzzy rule base, the activated rule resembles:

  If Damage Intensity is Low AND Extension is Small AND Bridge Scenario is Medium, then Alert is Low.

The defuzzification process (shown in Figure 8.7) confirms this outcome. The aggregated output is predominantly concentrated in the *Low Alert* region, and the crisp defuzzified value is approximately **17.37**. This value indicates a stable, non-critical steel structural condition, suggesting that immediate intervention is necessary.

These examples demonstrate the fuzzy reasoning system's ability to interpret subtle combinations of inputs and provide clear, interpretable recommendations that align with expert-defined safety rules.

Figure 8.6 shows the aggregated fuzzy output as of Equation 8.1 and the resulting defuzzified crisp alert level. The output alert level for the current test input condition is calculated as **17.37**, which lies within the *Low Alert* region, indicating that no immediate action is required.

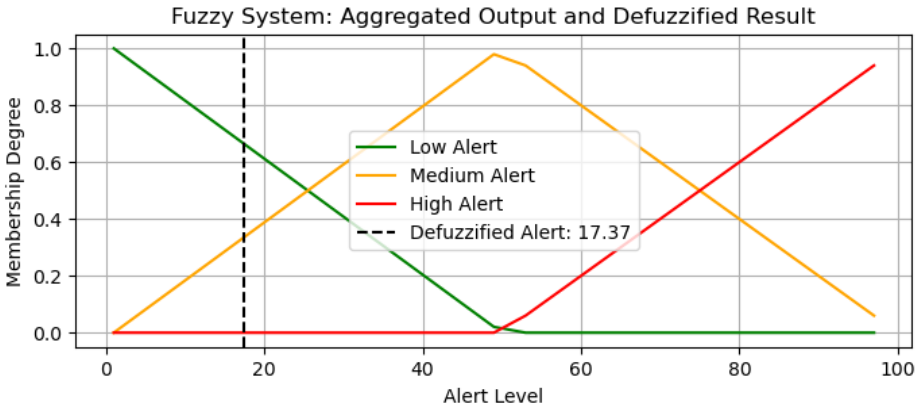


Figure 8.6: Aggregated fuzzy output and defuzzified result. The alert level is obtained via the defuzzification process, which converts the aggregated fuzzy output into a precise numerical value.

This defuzzified result allows SHM operators to receive intuitive feedback on bridge safety, converting complex sensor signals and rule logic into actionable maintenance insights.

8.5.5 Fuzzy Output Visualization and Predicted Alert Level

Figure 8.7 illustrates the defuzzification process and the resulting alert level for a specific input scenario. The three triangular curves represent the output fuzzy sets: *low*, *medium*, and *high* alert. Each curve reflects the degree of membership for different alert levels across the range $[0, 100]$.

The blue shaded area indicates the **aggregated fuzzy output**, which is the result of activating one or more fuzzy rules. In this case, the input conditions predominantly activate the *low* alert membership function. The black vertical line represents the **defuzzified alert level**, computed using the **centroid method**, and falls around **17.37**, which lies in the low alert region.

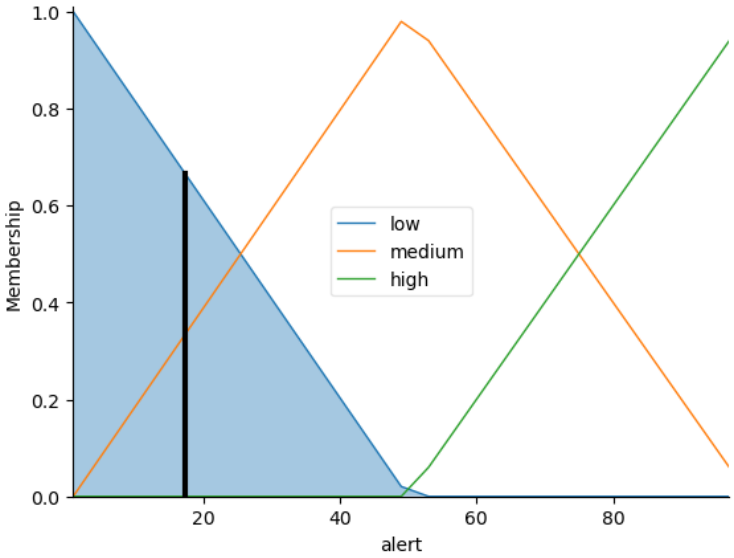


Figure 8.7: Fuzzy output membership functions and defuzzified alert value. The blue shaded region represents the activated area, and the black vertical line marks the predicted crisp alert level (≈ 17.37).

This visualization confirms that under current damage intensity, extension, and bridge scenario conditions, the system recommends a **low-level alert**, indicating a non-critical structural state. Such interpretability is essential for real-time monitoring systems, as it allows engineers to make informed decisions without analyzing raw sensor data or numerical outputs.

This fuzzy rule-based system acts as the final reasoning layer in the SHM architecture, allowing for qualitative interpretation of machine learning outputs and sensor data. It bridges the gap between numerical prediction models and actionable maintenance recommendations for railway bridge infrastructure.

8.6 Summary

This chapter presented the design and implementation of a fuzzy reasoning layer integrated into a machine learning based Structural Health Monitoring SHM framework for railway steel bridges. The fuzzy system acts as an interpretable decision layer that transforms numerical outputs from perception models; such as damage intensity, bridge scenario classification, and extension range, into linguistically meaningful alert levels.

The chapter began by defining fuzzy input variables using triangular membership functions across domains such as *low*, *medium*, and *high*. A complete rule base of 27 fuzzy *if-then* rules was constructed to combine these variables and infer alert levels: *No Danger*, *Attention*, *Pre-Alarm*, and *Alarm*. These rules encode expert knowledge and allow for intuitive, graded responses to changing bridge conditions.

The fuzzy inference process; consisting of fuzzification, rule evaluation, aggregation, and defuzzification; was explained in detail and visualized through membership function plots and defuzzified output graphs. A real case example demonstrated how input conditions such as low damage intensity and small extension in a medium bridge scenario correctly triggered a low alert level.

By embedding this fuzzy reasoning layer, the SHM framework achieves not only high accuracy from machine learning models but also interpretability, transparency, and decision support that aligns with engineering judgment. The result is a more reliable, responsive, and understandable monitoring system suitable for real-time bridge health assessment and maintenance planning.

This chapter concluded the reasoning layer of the proposed SHM system, which enables high-level, human-readable decisions to be derived from machine learning outputs. In the subsequent chapters, we explore system-level performance integration and discuss the overall impact of interpretability, real-time alerting, and hybrid model fusion in SHM deployment scenarios.

Chapter 9

Conclusions and Future Work

"The brain is a neural network. Why not simulate it and see what happens?"

— Geoffrey Hinton.

9.1 Conclusion

This thesis presented a comprehensive framework for Structural Health Monitoring SHM of railway steel bridges using vibration-based data analysis, frequency-domain feature extraction, deep learning architectures, and fuzzy reasoning. The main contributions of the work are summarized below:

- A structured pipeline was developed to convert raw accelerometer signals into time-frequency representations using Hankel matrices, SVD, and Short-Time Fourier Transform (STFT). These spectrograms formed the core feature inputs for classification models.
- Multiple deep learning architectures, including ResNet50, DenseNet121, and EfficientNetB0,were trained from scratch for damage intensity and bridge scenario classification. Their performance was evaluated using key metrics such as accuracy, precision, recall, F1-score, and ROC-AUC.
- Fusion models integrating DNNs with Vision Transformers (TRNet, TDNet, TENet) were proposed and shown to improve performance by combining local feature extraction with global attention.
- A fuzzy reasoning layer was introduced to enhance interpretability. This layer translated outputs from deep learning models into qualitative safety

alerts (for example, Low, Medium, Critical), mimicking expert decision-making and facilitating actionable insights.

- Extensive training and evaluation across multiple configurations demonstrated strong convergence, high accuracy (often exceeding 90%), and reliable classification of damage severity and location.

Overall, the integration of deep learning and fuzzy logic in a vibration-based SHM context proved to be an effective approach for accurate, interpretable, and real-time damage detection in railway steel bridges.

9.2 Future Directions

Building upon the foundations laid in this thesis, several promising research directions can be pursued:

- **Multimodal Sensor Fusion:** Incorporating additional sensing modalities such as strain gauges, temperature sensors, and image data (from Unmanned Aerial Vehicle (UAV)s or fixed cameras) can improve model robustness and support context-aware SHM.
- **Transfer Learning and Domain Adaptation:** Pre-training models on large generic datasets and fine-tuning them on limited bridge-specific data can reduce training costs and improve generalization across different bridge types.
- **Real-Time SHM Deployment:** EfficientNet-based fusion models and lightweight transformer modules can be deployed on embedded systems (for example, NVIDIA Jetson, Edge TPUs) for real-time monitoring in the field.
- **Uncertainty Quantification:** Integrating Bayesian inference or dropout-based uncertainty estimation can help quantify confidence in damage predictions, supporting safer maintenance decisions.
- **Online Learning and Drift Adaptation:** Incorporating continual learning strategies would allow SHM systems to adapt to aging structures and evolving operational conditions without full retraining.
- **Explainable AI (XAI):** Further development of interpretable models (for example, Explainable Boosting Machines) can help bridge the gap between model performance and human trust.

The findings and models developed in this thesis can serve as a foundation for the next generation of smart, scalable, and interpretable SHM systems, ultimately contributing to the safety and sustainability of critical railway infrastructure.

Bibliography

[1] Rete Ferroviaria Italiana. La rete oggi – rfi. `https://www.rfi.it/it/rete/la-rete-oggi.html`, 2024. Accessed: 2025-05-14.

[2] Marco Simoncelli, Angelo Aloisio, Marco Zucca, Giorgia Venturi, and Rocco Alaggio. Intensity and location of corrosion on the reliability of a steel bridge. *Journal of Constructional Steel Research*, 206:107937, 2023.

[3] European Commission. Railway transport statistics. `https://ec.europa.eu/eurostat/statistics-explained/`, 2023. Accessed: 2025-05-13.

[4] CORDIS - EU Research Results. Railway infrastructure projects. `https://cordis.europa.eu/`, 2019. Accessed: 2025-05-13.

[5] Z. Chen et al. Long-term shm systems for bridges. In *Bridge Maintenance, Safety, Management, Resilience and Sustainability*. Springer, 2017.

[6] European Railway Authorities. Survey on railway bridges in europe. In *DIVA Portal*, 2016. Accessed: 2025-05-13.

[7] Charles R. Farrar and Keith Worden. An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society A*, 365(1851):303–315, 2007.

[8] Scott W. Doebling, Charles R. Farrar, Michael B. Prime, and Daniel W. Shevitz. A review of damage identification methods that use modal data. *Los Alamos National Lab Report*, 131(1):1–41, 1998.

[9] Xiaojing Ye, Yi Su, and Jiawei Han. A review on machine learning in structural health monitoring. *Structural Health Monitoring*, 18(5-6):1255–1294, 2019.

[10] Keith Worden, Charles R. Farrar, Graeme Manson, and Gyuhae Park. The application of machine learning to structural health monitoring. *Philosophical Transactions of the Royal Society A*, 365(1851):515–537, 2007.

[11] Qing Liu, Sheng Wu, Yi Jin, and Hui Li. A fuzzy rule-based system for structural health monitoring data interpretation. *Expert Systems with Applications*, 139:112844, 2020.

[12] J. M. W. Brownjohn. Long span bridges: From testing to health monitoring. *Bridge Structures*, 3(1):3–14, 2007.

[13] Jerome P. Lynch and Kenneth J. Loh. Overview of wireless sensors for real-time health monitoring of civil structures. *Philosophical Transactions of the Royal Society A*, 365(1851):345–372, 2006.

[14] E.P. Carden and P. Fanning. Vibration based condition monitoring: a review. *Structural Health Monitoring*, 3(4):355–377, 2004.

[15] Y. Zhou, H. Li, and Y. Zhang. A review of dynamic testing and monitoring of railway bridges. *Measurement*, 166:108202, 2020.

[16] Hui Li and Jinping Ou. Structural health monitoring of railway bridges: a review. *International Journal of Rail Transportation*, 7(3):229–254, 2019.

[17] T. Nagayama and B.F. Spencer Jr. Structural health monitoring using smart sensors. *NSEL Report Series*, 2, 2007.

[18] N. Xu, S. Rangwala, K.K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin. A wireless sensor network for structural monitoring. *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, 2004.

[19] Alberto Armijo and Diego Zamora-Sánchez. Integration of railway bridge structural health monitoring into the internet of things with a digital twin: A case study. *Sensors*, 24(7):2115, 2024.

[20] Matteo Torzoni, Marco Tezzele, Stefano Mariani, Andrea Manzoni, and Karen E. Willcox. A digital twin framework for civil engineering structures. *arXiv preprint arXiv:2308.01445*, 2023.

[21] Y. Liu and Y. Wang. Integrated structural health monitoring in bridge engineering. *Automation in Construction*, 132:103950, 2022.

[22] C. Neves. Lessons from bridge structural health monitoring (shm) and their implications. *Infrastructures*, 9(2):30, 2024.

[23] X. Zhang and J. Li. Artificial intelligence in structural health management of existing bridges. *Automation in Construction*, 152:104455, 2024.

[24] Billie F. Spencer and Venkatesh Hoskere. Integration of structural health monitoring and intelligent transportation systems for bridge condition assessment: Current status and future direction. `https://www.academia.edu/33381828/`, 2018. Accessed: 2025-05-14.

[25] Anders Rytter. Vibrational based inspection of civil engineering structures, 1993. Ph.D.-Thesis defended publicly at the University of Aalborg, April 20, 1993 PDF for print: 206 pp.

[26] Tran, Ngoc Hoa. *Structural health monitoring for bridges using meta-heuristic optimization algorithms combined with artificial neural network.*

PhD thesis, Ghent University, 2022.

[27] Peter Van Overschee. *Subspace identification for linear systems: Theory—Implementation—Applications*. Springer Science & Business Media, 2012.

[28] Ai-Min Yan and Jean-Claude Golinval. Null subspace-based damage detection of structures using vibration measurements. *Mechanical Systems and Signal Processing*, 20(3):611–626, 2006.

[29] Jennifer L. Green, Sarah E. Manski, Timothy A. Hansen, and Jennifer E. Broatch. Descriptive statistics. In Robert J Tierney, Fazal Rizvi, and Kadriye Ercikan, editors, *International Encyclopedia of Education (Fourth Edition)*, pages 723–733. Elsevier, Oxford, fourth edition edition, 2023.

[30] Ronald Rousseau, Leo Egghe, and Raf Guns. Chapter 4 - statistics. In Ronald Rousseau, Leo Egghe, and Raf Guns, editors, *Becoming Metric-Wise*, Chandos Information Professional Series, pages 67–97. Chandos Publishing, 2018.

[31] Jay Lee. Statistics, descriptive. In Audrey Kobayashi, editor, *International Encyclopedia of Human Geography (Second Edition)*, pages 13–20. Elsevier, Oxford, second edition edition, 2020.

[32] J. Lee. Statistics, descriptive. In Rob Kitchin and Nigel Thrift, editors, *International Encyclopedia of Human Geography*, pages 422–428. Elsevier, Oxford, 2009.

[33] Julien I.E. Hoffman. Chapter 6 - normal distribution. In Julien I.E. Hoffman, editor, *Biostatistics for Medical and Biomedical Practitioners*, pages 101–119. Academic Press, 2015.

[34] Andrew C. Leon. 3.12 - descriptive and inferential statistics. In Alan S. Bellack and Michel Hersen, editors, *Comprehensive Clinical Psychology*, pages 243–285. Pergamon, Oxford, 1998.

[35] D. Rindskopf and M. Shiyko. Measures of dispersion, skewness and kurtosis. In Penelope Peterson, Eva Baker, and Barry McGaw, editors, *International Encyclopedia of Education (Third Edition)*, pages 267–273. Elsevier, Oxford, third edition edition, 2010.

[36] P. Plassmann. 5 - monitoring chronic wounds and determining treatment. In David Farrar, editor, *Advanced Wound Repair Therapies*, Woodhead Publishing Series in Biomaterials, pages 130–152. Woodhead Publishing, 2011.

[37] Anders Kallner. Formulas. In Anders Kallner, editor, *Laboratory Statistics (Second Edition)*, pages 1–140. Elsevier, second edition edition, 2018.

[38] Lawrence T. DeCarlo. On the meaning and use of kurtosis. *Psychological Methods*, 2:292 – 307, 1997.

[39] JAMES W. CLARK. Chapter 9 - linear and switching power-line regulators. In JAMES W. CLARK, editor, *AC Power Conditioners*, pages 76–98. Academic Press, 1990.

[40] Roberto Deboli, Angela Calvo, and Christian Preti. Vibration and impulsivity analysis of hand held olive beaters. *Applied Ergonomics*, 55:258–267, 2016.

[41] Mauro Feliziani, Tommaso Campi, Silvano Cruciani, and Francesca Maradei. Chapter 10 - electromagnetic field safety of automotive wpt systems. In Mauro Feliziani, Tommaso Campi, Silvano Cruciani, and Francesca Maradei, editors, *Wireless Power Transfer for E-Mobility*, pages 333–359. Academic Press, 2024.

[42] Morgan Jones. Chapter 4 - test equipment principles. In Morgan Jones, editor, *Building Valve Amplifiers (Second Edition)*, pages 235–380. Newnes, Oxford, second edition edition, 2014.

[43] Natarajan Prabaharan and Kaliannan Palanisamy. A comprehensive review on reduced switch multilevel inverter topologies, modulation techniques and applications. *Renewable and Sustainable Energy Reviews*, 76:1248–1282, 2017.

[44] Boualem Boashash. Chapter 4 - advanced time-frequency signal and system analysis. In Boualem Boashash, editor, *Time-Frequency Signal Analysis and Processing (Second Edition)*, pages 141–236. Academic Press, Oxford, second edition edition, 2016.

[45] Winser Alexander and Cranos Williams. Chapter 2 - fundamental dsp concepts. In Winser Alexander and Cranos Williams, editors, *Digital Signal Processing*, pages 19–157. Academic Press, Boston, 2017.

[46] John Price and Terry Goble. 10 - signals and noise. In Fraidoon Mazda, editor, *Telecommunications Engineer's Reference Book*, pages 10–1–10–15. Butterworth-Heinemann, 1993.

[47] Ali Grami. Chapter 3 - signals, systems, and spectral analysis. In Ali Grami, editor, *Introduction to Digital Communications*, pages 41–150. Academic Press, Boston, 2016.

[48] Vijay K. Garg and Yih-Chen Wang. 1 - signal types, properties, and processes. In WAI-KAI CHEN, editor, *The Electrical Engineering Handbook*, pages 951–956. Academic Press, Burlington, 2005.

[49] B.P. Marchant. Time–frequency analysis for biosystems engineering. *Biosystems Engineering*, 85(3):261–281, 2003.

[50] John Semmlow. Chapter 4 - the fourier transform and power spectrum: Implications and applications. In John Semmlow, editor, *Signals and Systems for Bioengineers (Second Edition)*, Biomedical Engineering, pages 131–165. Academic Press, Boston, second edition edition, 2012.

[51] A. Rius, I. Ruisánchez, M.P. Callao, and F.X. Rius. Reliability of analytical systems: use of control charts, time series models and recurrent neural networks (rnn). *Chemometrics and Intelligent Laboratory Systems*, 40(1):1–18, 1998.

[52] P.H Ellaway. Cumulative sum technique and its application to the analysis of peristimulus time histograms. *Electroencephalography and Clinical Neurophysiology*, 45(2):302–304, 1978.

[53] M. Khazaee and H. Sohn. Unsupervised feature extraction for structural health monitoring using cumulative deviation and entropy measures. In *Proceedings of the 11th International Workshop on Structural Health Monitoring (IWSHM)*, Stanford University, 2015.

[54] Alfirna Rizqi Lahitani, Adhistya Erna Permanasari, and Noor Akhmad Setiawan. Cosine similarity to determine similarity measure: Study case in online essay assessment. In *2016 4th International Conference on Cyber and IT Service Management*, pages 1–6, 2016.

[55] Jiawei Han, Micheline Kamber, and Jian Pei. 2 - getting to know your data. In Jiawei Han, Micheline Kamber, and Jian Pei, editors, *Data Mining: Concepts and Techniques (Third Edition)*, The Morgan Kaufmann Series in Data Management Systems, pages 39–82. Morgan Kaufmann, Boston, third edition edition, 2012.

[56] Amal Hbaieb, Samiha Ayed, and Lamia Chaari. A survey of trust management in the internet of vehicles. *Computer Networks*, 203:108558, 2022.

[57] Vijay Kotu and Bala Deshpande. Chapter 4 - classification. In Vijay Kotu and Bala Deshpande, editors, *Data Science (Second Edition)*, pages 65–163. Morgan Kaufmann, second edition edition, 2019.

[58] Brooke Lampe and Weizhi Meng. A survey of deep learning-based intrusion detection in automotive applications. *Expert Systems with Applications*, 221:119771, 2023.

[59] Vijay Kotu and Bala Deshpande. Chapter 11 - recommendation engines. In Vijay Kotu and Bala Deshpande, editors, *Data Science (Second Edition)*, pages 343–394. Morgan Kaufmann, second edition edition, 2019.

[60] Abhishek Singh, Utkarsh Porwal, Anurag Bhardwaj, and Wei Jin. Chapter 2 - multiscale representation learning for biomedical analysis. In Venu Govindaraju, Arni S.R. Srinivasa Rao, and C.R. Rao, editors, *Deep Learn-*

ing, volume 48 of *Handbook of Statistics*, pages 9–27. Elsevier, 2023.

[61] Gang Yan, Gaetan Kerschen, Philippe De Boe, and Jean-Claude Golinval. Feature extraction and damage detection using change point analysis. *Journal of Sound and Vibration*, 283(1-2):395–414, 2005.

[62] Wei Zhang, Qing Liu, and Lin Zhou. Real-time structural health monitoring of bridges using convolutional neural networks. *Engineering Structures*, 300:115658, 2025.

[63] Y. Chen, J. Wang, and M. Zhao. Time-frequency and time-scale analyses for structural health monitoring. *Structural Control and Health Monitoring*, 28(3), 2021.

[64] T. Lu, Z. Huang, and X. Li. Data-driven structural dynamic response reconstruction using segment-based generative adversarial networks. *Mechanical Systems and Signal Processing*, 154:107583, 2021.

[65] A. Gautam, M. Chatterjee, and R. Roy. Deep learning for structural health monitoring: Data, algorithms, applications, challenges, and trends. *Sensors*, 23(21):8824, 2023.

[66] Dynamox. Vibration analysis metrics: Kurtosis and skewness, 2022. Accessed: 2025-05-18.

[67] Saeed Hamidi, Atefeh Soleymani, and Maria Rashidi. A review of structural health monitoring and damage detection techniques in frames and bridges. In *Damage Detection and Structural Health Monitoring of Concrete and Masonry Structures*, pages 295–311. Springer, 2025.

[68] Georgina Cosma, David Brown, Matthew Archer, Masood Khan, and A. Graham Pockley. A survey on computational intelligence approaches for predictive modeling in prostate cancer. *Expert Systems with Applications*, 70:1–19, 2017.

[69] Cheng-Jin Du and Da-Wen Sun. 4 - object classification methods. In Da-Wen Sun, editor, *Computer Vision Technology for Food Quality Evaluation*, Food Science and Technology, pages 81–107. Academic Press, Amsterdam, 2008.

[70] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[71] Vladimir N. Vapnik. The support vector method. In Wulfram Gerstner, Alain Germond, Martin Hasler, and Jean-Daniel Nicoud, editors, *Artificial Neural Networks — ICANN'97*, pages 261–271, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

[72] Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[73] Hervé A. Bourlard and Nelson Morgan. *Multilayer Perceptrons*, pages

59–80. Springer US, Boston, MA, 1994.

[74] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

[75] Joshua S. Richman. Chapter thirteen - multivariate neighborhood sample entropy: A method for data reduction and prediction of complex data. In Michael L. Johnson and Ludwig Brand, editors, *Computer Methods, Part C*, volume 487 of *Methods in Enzymology*, pages 397–408. Academic Press, 2011.

[76] Qingchen Zhang, Laurence T. Yang, Zhikui Chen, and Peng Li. A survey on deep learning for big data. *Information Fusion*, 42:146–157, 2018.

[77] Hui Liu. Chapter 7 - rail transit channel robot systems. In Hui Liu, editor, *Robot Systems for Rail Transit Applications*, pages 283–328. Elsevier, 2020.

[78] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.

[79] M. Sonker and R. Shanker. Enhanced diagnostic approach for multiple damage detection and severity evaluation through emi-based sensing and artificial neural network model. *Asian Journal of Civil Engineering*, 26:747–760, 2024.

[80] H. Ahmed, A. I. Sadaf, and S. Banerjee. Non-linear signature in ultrasonic pulse-echo signal for structural health monitoring of fatigue induced composite structures. In *Structural Health Monitoring 2023*, 2023.

[81] R. Katam, V. D. K. Pasupuleti, and P. Kalapatapu. A review on structural health monitoring: past to present. *Innovative Infrastructure Solutions*, 8, 2023.

[82] Y. Ren, O. Bareille, Z. Lin, and X.-R. Huang. Review of damage detection techniques in vibration-based structural health monitoring. *International Journal of Dynamics and Control*, 13, 2025.

[83] J. Jia and Y. Li. Deep learning for structural health monitoring: Data, algorithms, applications, challenges, and trends. *Sensors*, 23(21):8824, 2023.

[84] Muhammad Asad, Giovanni De Gasperis, and Stefania Costantini. RBF based NN architecture for structural health analysis of railway steel bridges. In *AIxIA Doctoral Consortium 2023 @ 22nd International Conference of the Italian Association for Artificial Intelligence*. Springer Nature, Rome, Italy, 2024.

[85] Muhammad Asad. Machine learning based ambient analysis of railway steel bridges for damage detection. In *International Symposium on Am-*

bient Intelligence, pages 250–255. Springer, 2023.

[86] Øiseth, Ole Svendsen, Bjørn T., Frøseth, Gunnstein T. and Rønnquist, Anders. A data-based structural health monitoring approach for damage detection in steel bridges using experimental data. *Journal of Civil Structural Health Monitoring*, 12:101–115, 2022.

[87] Frederik Wedel and Steffen Marx. Application of machine learning methods on real bridge monitoring data. *Engineering Structures*, 250:113365, 2022.

[88] Leander J. Neves A. C., González I. and Karoumi R. Structural health monitoring of bridges: a model-free ann-based approach to damage detection. *Journal of Civil Structural Health Monitoring*, 7:689–702, 2017.

[89] Onur Avci, Osama Abdeljaber, Serkan Kiranyaz, Mohammed Hussein, Moncef Gabbouj, and Daniel 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.

[90] M. Mehrjoo, N. Khaji, H. Moharrami, and A. Bahreininejad. Damage detection of truss bridge joints using artificial neural networks. *Expert Systems with Applications*, 35(3):1122–1131, 2008.

[91] Jungwhee Lee and Sungkon Kim. Structural damage detection in the frequency domain using neural networks. *Journal of Intelligent Material Systems and Structures*, 18(8):785–792, 2007.

[92] Mirco Muttillo, Vincenzo Stornelli, Rocco Alaggio, Romina Paolucci, Luca Di Battista, Tullio de Rubeis, and Giuseppe Ferri. Structural health monitoring: An iot sensor system for structural damage indicator evaluation. *Sensors*, 20(17), 2020.

[93] Zachary C. Lipton. The mythos of model interpretability. *Communications of the ACM*, 61(10):36–43, 2018.

[94] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730, 2015.

[95] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[96] Thomas G. Dietterich. *Ensemble Methods in Machine Learning*. Springer, 2000.

[97] Zhi-Hua Zhou. Ensemble methods: Foundations and algorithms. *CRC Press*, 2012.

[98] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, 1999.

[99] Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. Interpretml: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019.

[100] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631, 2013.

[101] Y.-W. Wang, Y.-Q. Ni, and S.-M. Wang. Structural health monitoring of railway bridges using innovative sensing technologies and machine learning algorithms: A concise review. *Intelligent Transportation Infrastructure*, 1(1):liac009, 2022.

[102] A. Meixedo, J. Santos, D. Ribeiro, R. Calçada, and M. Todd. Ai-based structural health monitoring procedure for railway bridges. In *Proceedings of the Fifth International Conference on Railway Technology: Research, Development and Maintenance*, 2022.

[103] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[104] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, 7:21, 2013.

[105] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

[106] Xinyu Zhang, Julien Martinelli, and ST John. Challenges in interpretability of additive models. *arXiv preprint arXiv:2504.10169*, 2025.

[107] Markéta Krùpovà, Nabil Rachdi, and Quentin Guibert. Explainable boosting machine for predicting claim severity and frequency in car insurance. *arXiv preprint arXiv:2503.21321*, 2025.

[108] Brandon M. Greenwell, Annika Dahlmann, and Saurabh Dhoble. Explainable boosting machines with sparsity – maintaining explainability in high-dimensional settings. *arXiv preprint arXiv:2311.07452*, 2023.

[109] Robert Corbally and Abdollah Malekjafarian. A data-driven approach for drive-by damage detection in bridges considering the influence of temperature change. *Engineering Structures*, 253:113783, 2022.

[110] Luca Rosafalco, Matteo Torzoni, Andrea Manzoni, Stefano Mariani, and

Alberto Corigliano. Online structural health monitoring by model order reduction and deep learning algorithms. *Computers & Structures*, 255:106604, 2021.

[111] Duong Huong Nguyen, Quoc Bao Nguyen, T. Bui-Tien, Guido De Roeck, and Magd Abdel Wahab. Damage detection in girder bridges using modal curvatures gapped smoothing method and convolutional neural network: Application to bo nghi bridge. *Theoretical and Applied Fracture Mechanics*, 109:102728, 2020.

[112] Adam Santos, Eloi Figueiredo, Moisés Silva, Reginaldo Santos, Claudomiro Sales, and João C. W. A. Costa. Genetic-based em algorithm to improve the robustness of gaussian mixture models for damage detection in bridges. *Structural Control and Health Monitoring*, 24(3):e1886, 2017. e1886 STC-15-0204.R2.

[113] Elisa Khouri Chalouhi, Ignacio Gonzalez, Carmelo Gentile, and Raid Karoumi. Damage detection in railway bridges using machine learning: application to a historic structure. *Procedia Engineering*, 199:1931–1936, 2017. X International Conference on Structural Dynamics, EURODYN 2017.

[114] Muhammad Asaad Cheema, Muhammad Zohaib Sarwar, Vinay Chakravarthi Gogineni, Daniel Cantero, and Pierluigi Salvo Rossi. Computationally efficient structural health monitoring using graph signal processing. *IEEE Sensors Journal*, 24(7):11895–11905, 2024.

[115] Muhammad Asad, Giovanni De Gasperis, and Stefania Costantini. RBF based NN architecture for structural health analysis of railway steel bridges (short paper). In Valentina Poggioni and Silvia Rossi, editors, *Proceedings of the AIxIA Doctoral Consortium 2023 co-located with the 22nd International Conference of the Italian Association for Artificial Intelligence (AIxIA 2023), Rome, Italy, November 6-7, 2023*, volume 3670 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2023.

[116] Cuong-Le Thanh, Khatir Samir Nghia-Nguyen Trong, Trong-Nguyen Phuoc, Mirjalili Seyedali, and Nguyen Khuong D. An efficient approach for damage identification based on improved machine learning using pso-svm. *Engineering with Computers*, 38:3069– 3084, 2022.

[117] M. Dilena, M.P. Limongelli, and A. Morassi. Damage localization in bridges via the frf interpolation method. *Mechanical Systems and Signal Processing*, 52-53:162–180, 2015.

[118] Zhiguo He, Wentao Li, Hadi Salehi, Hao Zhang, Haiyi Zhou, and

Pengcheng Jiao. Integrated structural health monitoring in bridge engineering. *Automation in Construction*, 136:104168, 2022.

[119] Øiseth, Ole Svendsen, Bjørn T., Frøseth, Gunnstein T. and Rønnquist, Anders. A data-based structural health monitoring approach for damage detection in steel bridges using experimental data. *Journal of Civil Structural Health Monitoring*, 12:101–115, 2022.

[120] Hamid Khodabandehlou, Gökhan Pekcan, and M. Sami Fadali. Vibration-based structural condition assessment using convolution neural networks. *Structural Control and Health Monitoring*, 26(2):e2308, 2019.

[121] Osama Abdeljaber, Onur Avci, Mustafa Serkan Kiranyaz, Boualem Boashash, Henry Sodano, and Daniel J. Inman. 1-d cnns for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing*, 275:1308–1317, 2018.

[122] Huile Li, Tianyu Wang, and Gang Wu. A bayesian deep learning approach for random vibration analysis of bridges subjected to vehicle dynamic interaction. *Mechanical Systems and Signal Processing*, 170:108799, 2022.

[123] Osama Abdeljaber, Onur Avci, Serkan Kiranyaz, Moncef Gabbouj, and Daniel J. Inman. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of Sound and Vibration*, 388:154–170, 2017.

[124] Alireza Ghiasi, Mahdi Kazemi Moghaddam, Ching-Tai Ng, Abdul Hamid Sheikh, and Javen Qinfeng Shi. Damage classification of in-service steel railway bridges using a novel vibration-based convolutional neural network. *Engineering Structures*, 264:114474, 2022.

[125] Ahmed Rageh, Saeed Eftekhar Azam, and Daniel G. Linzell. Steel railway bridge fatigue damage detection using numerical models and machine learning: Mitigating influence of modeling uncertainty. *International Journal of Fatigue*, 134:105458, 2020.

[126] Meixedo, Andreia, Ribeiro, Diogo, Santos, João, Calçada, Rui, and Todd, Michael. Progressive numerical model validation of a bowstring-arch railway bridge based on a structural health monitoring system. *Journal of Civil Structural Health Monitoring*, 11:421 – 449, 2021.

[127] Emmanuel Akintunde, Saeed Eftekhar Azam, Ahmed Rageh, and Daniel G. Linzell. Unsupervised machine learning for robust bridge damage detection: Full-scale experimental validation. *Engineering Structures*, 249:113250, 2021.

[128] Mohammed Abbas Mousa, Mustafasanie M. Yussof, Ufuoma Joseph Udi, Fadzli Mohamed Nazri, Mohd Khairul Kamarudin, Gerard A. R. Parke,

Lateef N. Assi, and Seyed Ali Ghahari. Application of digital image correlation in structural health monitoring of bridge infrastructures: A review. *Infrastructures*, 6(12), 2021.

[129] Md. Ashikur Rahman, Lway Faisal Abdulrazak, Md. Mamun Ali, Imran Mahmud, Kawsar Ahmed, and Francis M. Bui. Machine learning-based approach for predicting diabetes employing socio-demographic characteristics. *Algorithms*, 16(11), 2023.

[130] T. Kourti. Multivariate statistical process control and process control, using latent variables. In *Comprehensive Chemometrics*, pages 21–54. Elsevier, Oxford, 2009.

[131] S.M. Gandhi and Bhabesh Sarkar. *Conventional and Statistical Resource/Reserve Estimation*, pages 271–288. Elsevier, 12 2016.

[132] Tadhg Buckley, Bidisha Ghosh, and Vikram Pakrashi. A feature extraction & selection benchmark for structural health monitoring. *Structural Health Monitoring*, 22(3):2082–2127, 2023.

[133] Walter Ciciora, James Farmer, David Large, and Michael Adams. *Digital Modulation*, chapter 4. Morgan Kaufmann, 2004.

[134] Michael Parker. *Orthogonal Frequency Division Multiple Access Wireless Communications*, pages 209–230. Elsevier, 12 2017.

[135] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In Robert Meersman, Zahir Tari, and Douglas C. Schmidt, editors, *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, pages 986–996, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[136] Thorsten Joachims. *Making large-scale support vector machine learning practical*, page 169–184. MIT Press, Cambridge, MA, USA, 1999.

[137] J. R. Quinlan. Learning decision tree classifiers. *ACM Comput. Surv.*, 28(1):71–72, March 1996.

[138] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, 2001.

[139] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(4):e1249, 2018.

[140] A Olgac and Bekir Karlik. Performance analysis of various activation functions in generalized mlp architectures of neural networks. *International Journal of Artificial Intelligence And Expert Systems*, 1:111–122, 02 2011.

[141] G. David Garson. Interpreting neural-network connection weights. *AI Expert*, 6(4):46–51, April 1991.

[142] Kamilya Smagulova and Alex Pappachen James. *Overview of Long Short-*

Term Memory Neural Networks, pages 139–153. Springer International Publishing, Cham, 2020.

[143] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019, 2022.

[144] Scott W. Doebling, Charles R. Farrar, and Michael B. Prime. A review of damage identification methods that examine changes in dynamic properties. *Shock and Vibration Digest*, 30(2):91–105, 1998.

[145] O.S. Salawu. Detection of structural damage through changes in frequency: a review. *Engineering structures*, 19(9):718–723, 1997.

[146] Charles R. Farrar and Keith Worden. *Structural Health Monitoring: A Machine Learning Perspective*. John Wiley & Sons, 2012.

[147] T. T. Nguyen, M. Salamak, A. Katunin, G. Poprawa, and M. Gerges. Vibration-based shm of railway steel arch bridge with orbit-shaped analysis. *Engineering Structures*, 293:115745, 2024.

[148] C. Mohan, S. D. Gopala, and P. Swetha. Structural health monitoring of railway truss bridge under moving train load using machine learning. *Journal of Civil Structural Health Monitoring*, 11(2):123–134, 2025.

[149] S. S. Saidin, A. Jamadin, S. A. Kudus, N. M. Amin, and M. A. Anuar. An overview: The application of vibration-based techniques in bridge structural health monitoring. *International Journal of Concrete Structures and Materials*, 16(1):69, 2022.

[150] H. Yin and X.Q. Zhu. Vibration-based structural damage detection using frequency domain features and support vector machines. *Journal of Sound and Vibration*, 439:1–15, 2019.

[151] Y. Lin and X. Zhu. Application of frequency response functions for damage detection in steel bridge connections. *Engineering Structures*, 56:1900–1912, 2013.

[152] F.K. Chang and X. Zhan. Multi-level damage detection in steel bridges using wavelet packet analysis and pattern recognition. *Journal of Structural Engineering*, 143(6):04017031, 2017.

[153] J. Zhao and X. Wang. Deep learning with time-frequency analysis for structural damage detection under operational and environmental variability. *Mechanical Systems and Signal Processing*, 154:107581, 2021.

[154] Y. Geng and Y. Xu. Damage identification in railway viaducts using stft-based vibration spectrograms and deep learning. *Structural Control and Health Monitoring*, 27(9):e2579, 2020.

[155] J. Wang and Y. Zhang. Deep learning for bridge damage detection using time-frequency vibration analysis. *Computer-Aided Civil and Infrastructure Engineering*, 34(12):1060–1076, 2019.

[156] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[157] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017.

[158] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 6105–6114, 2019.

[159] Lotfi A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[160] Jiann-Shing Lee, Chih-Huang Chung, and Masanobu Shinozuka. Application of fuzzy logic to structural health monitoring. *Computer-Aided Civil and Infrastructure Engineering*, 20(6):439–450, 2005.

[161] Yunlong Yang and Satish Nagarajaiah. A hybrid intelligent method for structural health monitoring. *Smart Materials and Structures*, 26(9):094002, 2017.

[162] David Gunning. Explainable artificial intelligence (xai). DARPA Program Report, 2017.

[163] Kai Li, Xuefeng Zhang, and Yongxing Wang. Fuzzy logic-based decision-making model for bridge condition assessment. *Engineering Structures*, 230:111692, 2021.

[164] Ming Zhang, Shuang Wang, and Jie Liu. Fuzzy reasoning-based structural fault diagnosis model using multi-sensor information fusion. *Measurement*, 195:111204, 2022.

[165] Hans-Jürgen Zimmermann. *Fuzzy set theory—and its applications*. Springer Science & Business Media, 2010.

Glossary

ADAM Adaptive Moment Estimation. 98, 99, 101, 120
AI Artificial Intelligence. 5
ANN Artificial Neural Network. 42, 45, 78–80
AUC Area Under the Curve. 121, 123

BS Bridge Scenarios. 49, 64, 140, 142

CF Crest Factor. 33, 42, 84, 85
CNN Convolutional Neural Network. 10, 12, 37, 42, 48, 57, 79–81, 86, 105, 107, 110, 113, 115
CNNs Convolutional Neural Networks. 6, 8, 16, 37
COA Centroid of Area. 136
CS Cosine Similarity. 31, 33, 42, 84, 85
CSD Cumulative sum of differences. 31, 35, 42, 58, 85

D Danger Level. 134
DB-SSL Dual-Branch Semi-Supervised Learning. 107
DD-SSI Data-Driven Stochastic Subspace Identification. 27, 77
DDR Data Dimensionality Reduction. 27, 31, 77
DI Damage Intensities. 49, 64, 140, 142
DL Deep Learning. 89
DL-FL Dual Level Fusion Learning. 104
DNN Deep Neural Network. 10, 13, 37, 49, 81, 82, 86, 89, 99, 101, 104, 111, 113, 148
DT Decision Tree. 86

E Energy. 33, 42, 84
EBM Explainable Boosting Machine. 12, 13, 35, 58, 59, 61, 62, 64
EBMs Explainable Boosting Machines. 9, 38, 60, 78
EMI Electromechanical Impedance. 44
EU European Union. 1, 5, 11

EX Extension. 140, 142

FC Fully Connected. 86
FEM Finite Element Model. 6, 10, 12, 27, 29, 45, 48, 79, 104
FFT Fast Fourier Transform. 16, 106
FIS Fuzzy Inference System. 134
FL Fuzzy Logic. 136

GA Genetic Algorithm. 80
GAM Generalized Additive Model. 58, 60
GAMs Generalized Additive Models. 38, 60, 73
GELU Gaussian Error Linear Unit. 87, 88
GPS Global Positioning System. 22
GPU Graphics Processing Unit. 90
Grad-CAM Gradient-weighted Class Activation Mapping. 80

ICA Independent Component Analysis. 80
IoT Internet of Things. 15, 45
ITS Intelligent Transportation Systems. 18

KNN k-Nearest Neighbors. 6, 10, 37, 42, 44, 48, 78, 86, 89, 99, 101

MAS Multi-Agent System. iii, iv, 8
MFs Membership Functions. 136
ML Machine Learning. iv, 24, 31, 42, 45, 60, 77, 78, 81, 89, 99, 131, 136
MLP Multi-Layer Perceptron. 10, 37, 45, 86, 89, 99, 101
MOM Mean of Maxima. 136
MOR Model Order Reduction. 80

NN Neural Network. 42, 45, 86, 131

OMA Operational Modal Analysis. 27, 77, 106

PCA Principal Component Analysis. 78, 80, 84
PSO-SVM Particle Swarm Optimization and Support Vector Machines. 79

RBF Radial Basis Function. 9, 12, 38, 42–50, 55, 57
RBFNet RBF based Neural Network. 51, 53–55
ReLU Rectified Linear Unit. 88, 118
RF Random Forest. 10, 42, 44, 48, 78, 86, 89, 91, 99–101

RMS Root Mean Square. 6, 32, 36, 79

RMSE Root Mean Square Error. 80

ROC Receiver Operating Characteristic. 44, 80, 121, 123

ROM Reduced-Order Model. 27, 77, 81

SF Stress Force. 64

SGD Stochastic Gradient Descent. 98, 99, 101, 120

SHM Structural Health Monitoring. iii, iv, 2, 6, 8, 9, 11, 12, 15–22, 27, 29, 31–40, 42–44, 46, 47, 49, 50, 55, 58, 59, 61, 73, 75, 76, 78, 85, 99, 103, 106, 107, 111–113, 121, 123, 125, 128, 131, 136, 142, 146–149

SNR signal-to-noise ratio. 80

STFT Short-Time Fourier Transform. iii, 10, 13, 16, 105–113

SVD Singular Value Decomposition. iii, 79, 80, 107–111

SVM Support Vector Machine. 6, 10, 16, 35, 37, 42, 44, 48, 78, 86, 89, 99, 101

t-SNE t-distributed Stochastic Neighbor Embedding. 80

UAV Unmanned Aerial Vehicle. 149

ViT Vision Transformer. ix, xiii, 39, 105, 110, 111, 116–121, 123, 124, 126–129

WT Wavelet Transforms. 16, 106