TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

Thesis No.: M-412-MSREE-2024-2026

VIBRATIONAL DATA AND MACHINE LEARNING BASED MULTI BOLT LOOSENESS LEVELS CLASSIFICATION

BY

MANU ARYAL

A THESIS

SUBMITTED TO THE DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN RENEWABLE ENERGY ENGINEERING

DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

LALITPUR, NEPAL

APRIL, 2026

COPYRIGHT

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

The undersigned certify that they have read, and recommended to the Institute of Engineering for acceptance, a thesis entitled **"Vibrational Data and Machine Learning Based Multi Bolt Looseness Levels Classification"** submitted by Manu Aryal (080MSREE011) in partial fulfilment of the requirements for the degree of Master of Science in Renewable Energy Engineering.

Supervisor: Prof. Dr. Mahesh Chandra Luintel
Department of Mechanical and Aerospace Engineering
IOE, Pulchowk Campus

Supervisor: Asst. Prof. Laxman Motra
Department of Mechanical and Aerospace Engineering
IOE, Pulchowk Campus

External Examiner: Asst. Prof. Bikki Chhantyal
Department of Automobile and Mechanical Engineering
IOE, Thapathali Campus

Tribhuvan University
Institute of Engineering
Department of Mechanical and Aerospace Engineering
Pulchowk Campus

Committee Chairperson: Asst. Prof. Dr. Sudip Bhattrai
Head of Department
Department of Mechanical and Aerospace Engineering
IOE, Pulchowk Campus

Date: April 29, 2026

ABSTRACT

The loss of bolt preload in multi-fastener structural joints is a gradual failure process that affects load transfer and overall stiffness. The response in such systems is often governed by coupled structural dynamics rather than local behavior. Percussion-based inspection methods have focused mostly on single-bolt systems under controlled conditions, limiting their use in realistic assemblies where multiple bolts interact and measurements are affected by environmental noise. This study models the acoustic response induced by impacts as a stochastic excitation of an underlying low-dimensional structural system and employs a state-space learning framework for robust state identification. A hybrid model combining one-dimensional convolutional neural networks with input-selective state space models is used to capture both local transient features and long range temporal dependencies. Contrary to the traditional spectral or cepstral methods, the proposed approach adapts its internal state to preserve useful structural information while reducing the effect of noise. The method is tested on a cantilever-based multi-bolt system with dominant global modes, where joint states are defined by different preload levels. All models achieve high accuracy (>90%) under the noise free conditions but, at 20 dB signal-to-noise ratio, MFCC-based methods accuracy drop to 9.1% and PSD-based methods to 76.3%. In contrast, the proposed framework maintains 98.9% accuracy across SNR levels from 20 to 60 dB. These results show that state-space modeling offers a reliable and noise-robust solution for percussion-based structural health monitoring in practical environments.

Keywords: *bolt preload classification; percussion-based SHM; selective state space model; multi-bolt structures; acoustic signal processing; noise robustness*

ACKNOWLEDGEMENT

I would like to thank Professor Dr. Mahesh Chandra Luintel for recommending this topic and his constant guidance throughout the duration of this thesis. I would like to express my sincere gratitude to Assistant professor Laxman Motra for constantly helping me throughout this work. This research work couldn't be in this place without my supervisors. I would like to thank all my professors who were there in different aspects of my master's journey guiding me towards becoming a better researcher and scholar. I would also like to thank Yatri Design Studio for providing me with the facility and computational resources for data acquisition and model training. Throughout the course of this research, my workspace colleagues Ranjan Pandey, Chirag G Shrestha, Anmol Shrestha, and Hemanta Neupane provided their tangible and intangible support without which this project would not have been what it is today. I also want to recognize my father Pusparaj Aryal and mother Namina Ghimire Aryal for always pushing me towards academics and motivating me at times when I lose confidence in myself. Lastly, I am indebted to the Department of Mechanical and Aerospace Engineering, IOE Pulchowk campus and the head of department Dr. Sudip Bhattarai for providing me with this opportunity to carry out my research work.

I would also like to acknowledge "UCT 34673: Transforming Energy Access - Learning Partnership (TEA-LP)" project and Center for Energy Studies (CES) for monetary support aiding the completion of this thesis.

Sincerely,

Manu Aryal (080MSREE011)

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

SNR	Signal-to-noise ratio
FFT	Fast Fourier Transform
PSD	Power Spectral Density
MFCC	Mel Frequency Cepstral Coefficients
STFT	Short-Time Fourier Transform
f	Frequency
As	Signal amplitude
An	Noise amplitude
k	Structural stiffness
c	Damping coefficient
m	Mass
ω	Angular frequency
τ	Bolt torque (preload)
nb	Number of bolts
ns	Number of states
LHS	Latin Hypercube Sampling
fs	Sampling frequency
$\mathbf{X} \in \mathbb{R}^{T \times D}$	Input feature sequence

ReLU	Rectified Linear Unit activation
θ	Model parameters
$\mathscr{L}$	Loss function
η	Learning rate
A	State transition matrix
B	Input projection matrix
C	Output projection matrix
Δ	Time step / discretization parameter
$\bar{A}$	Discretized state transition matrix
$\bar{B}$	Discretized input matrix
B(t)	Input-dependent projection matrix
C(t)	Output-dependent projection matrix
$\Delta(t)$	Input-dependent step size
H(t)	Updated hidden state
x(t)	Input signal (percussion acoustic signal)
y(t)	Output signal / model prediction
h(t)	Hidden state vector
t	Time index

CHAPTER ONE: INTRODUCTION

1.1 Background

The ongoing development in infrastructure and machinery can't be imagined without one tiny but significant component, the bolt. Bolts have been upholding the integrity of massive structures for the better part of the past few centuries having the first printed record of threaded bolts only dating back to the 15th Century. One of the major advantages of using nuts and bolts over other rigid connectors is their ease of dismantling while maintaining the integrity in the form of bolt pretension. This is also a major drawback, as during operation, the bolt loses pretension and hampers the intended holding of components in place. Some of the most prominent causes of bolt loosening include vibrations, stresses, material failure, self-loosening, etc., with the major contributing factor being relative motion within the threads due to the sliding of the nut or bolt head relative to the fastened joint, reducing the clamping force. [1]

Identifying whether a bolt is loose or not has always been significant to track the defects and damage progression in the structures. Several structural health monitoring techniques have been developed and used over the past two decades. A review by Wang (2013) categorised these techniques into three major categories that are most commonly used: Acoustoelastic effect-based methods, piezoelectric active sensing method and the piezoelectric impedance methods. [22] Among these, acoustoelastic methods are impractical due to their high operational cost and impracticality during real-time monitoring. Thus, several other methods like modal analysis, wave propagation and percussion-based methods are actively used, driven by their simplicity and relevance in real structures. With the latest advances in machine learning methods, analysing large clusters of data and using them for signal processing and classification has seen a new paradigm in structural state detection. Following that, techniques such as Artificial Neural Networks and Convolutional Neural Networks, Decision Trees, Random Forests and Support Vector Machines have been actively being used for bolt looseness detection. A visual overview of these methodologies as defined by Nikesh (2023) is presented in Figure 1.1. [23]

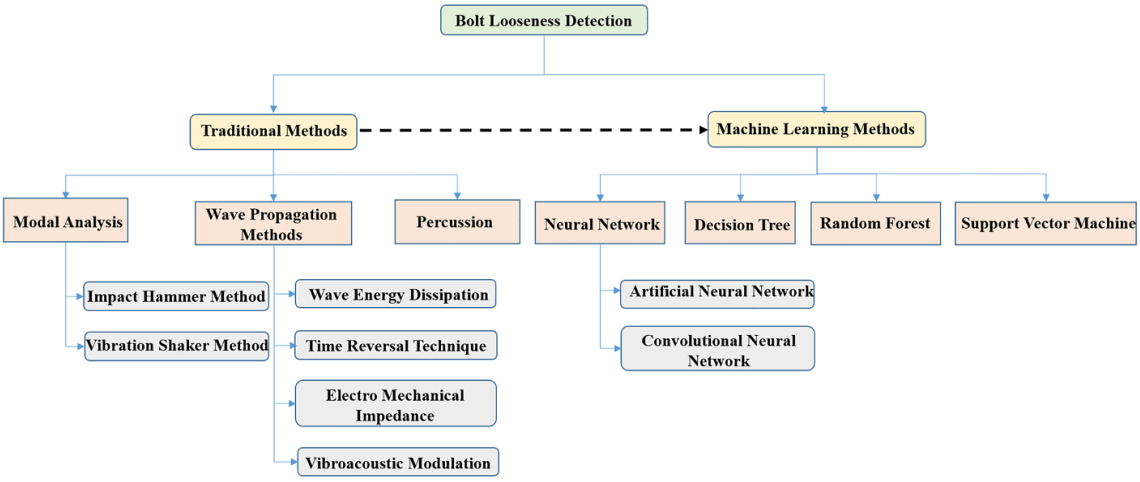

Figure 1.1: Types of bolt loosening detection techniques

1.2 Research Statement

By analysing the impact-induced acoustic and vibrational response of a joint, percussion-based structural health monitoring (SHM) provides a non-destructive approach to detecting bolt looseness. The previous studies have demonstrated promising results using signal processing and machine learning techniques, but most investigations are limited to single-bolt assemblies under controlled laboratory conditions. The complexity of real engineering structures, where multiple bolts collectively determine structural stiffness, are not reflected by these setups. Thus, systematic study for simultaneously classifying discrete preload levels across multiple bolts using percussion-based methods is necessary.

Additionally, the influence of impact location on classification performance is yet to be quantified. Since the excitation point can alter the wave propagation paths and modal participation significantly, understanding its effect is essential for practical usage which has not been addressed in prior research.

The choice of structure is yet another limitation in existing research. Most experiments use rigid, low-noise specimens where the percussion responses dominate the signal. In contrast, cantilever structures have significant self-excited vibration and low damping which introduces dynamic components that might suppress preload-related features. The extent to which the classification accuracy is affected by such structural vibration has not been clearly studied.

The recent advances in state space models involving the Mamba architecture have provided an opportunity to address these challenges due to their strong sequence modelling and noise-handling capabilities. This study applies the Mamba model to multi-bolt preload classification and evaluates the performance under cantilever-induced vibration and environmental noise while advancing percussion-based SHM toward realistic structural conditions. [24]

1.3 Objective of this research

1.3.1 Main Objective

The main objective of this research is to develop and evaluate a percussion-based structural health monitoring framework for accurate classification of discrete bolt preload levels in multi-bolt structures and test its classification viability in noisy environmental conditions.

1.3.2 Specific Objectives

- To construct and validate a percussion-based setup to acquire structural vibration data from a multi-bolt structural across three impact locations.
- To develop a labeled dataset representing multiple discrete torque levels and distinct impact positions for robust supervised learning.
- To implement the Mamba State Space Model for sequence-based preload classification and compare its performance against classical machine learning architectures.
- To investigate the influence of impact location and environmental noise on signal characteristics to evaluate classification accuracy under both controlled and noisy conditions.

1.4 Limitations of the study

- The study is conducted on a specific multi-bolt configuration and a cantilever setup. Although this introduces dynamic complexity, the findings might not directly translate to other joint geometries, materials, or boundary conditions.

- The classification framework is built around discrete torque levels while in reality, bolt preload degrades gradually and may vary across fasteners in an uneven manner. This continuous nature of preload loss may reduce classification clarity compared to idealized discrete states.

CHAPTER TWO: LITERATURE REVIEW

Many studies under Structural Health Monitoring (SHM) have been conducted to precisely identify the bolt loosening and mitigate it before any significant damage is done. In the late 60s, Gerhard Junker first discussed the theory of transverse vibration-induced self-loosening of preloaded bolted connections. He proposed a simplified method for large-scale testing using a vibration machine where 20, 50 and 80% preloads were used as parameters, and the bolt behavior under various cyclic loadings was studied. [2] It was in 2007 when the use of Piezoceramic (PZT) sensors was proposed by Gao to estimate the bolt condition quantitatively. These PZT transducers monitor the bolt preload by measuring the changes in electrical impedance as the preload changes and are sensitive to loose bolts. [3] Following up on the usage of PZT for active sensing, many researchers studied the effect of bolt state on impedance change by employing finite element analysis, frequency shift analysis, attaching PZT to washers, and using time reversal techniques. [4,5,6,7] Advancing on it further, Wongi in 2021 combined the piezoelectric transducer-based electromechanical impedance (EMI) method with machine learning modal probabilistic neural networks (PNN) to classify the torque loss data. Getting over 90% of classification accuracy demonstrated potential applications for NDT in real structures. [8] Oybek used the vibration-based loosening detection using Machine Learning Algorithms in a multi-bolt structure by classifying motor vibration induced bolt loosening. The signal features were extracted using the Short Time Fourier Transform (STFT) method, and several classifiers were compared to predict bolt looseness, where the Random Forest Classifier outperformed others with an accuracy of 95.8%. [9]

These sensor-based methods are effective but are arduous to implement in real-life structures due to the limitations with sensor placement and increased complexity of data acquisition. To overcome that and make the bolt loosening detection practically accessible, Kong put forward the landmark tapping and listening methodology where percussion induced audio signals were captured by a standard microphone, which contained sufficient information to access bolt loosening. The features obtained from the frequency and temporal spectrum could differentiate between various preload levels of a single bolted structure. [10] Zhang advanced this method by using Support Vector Machines (SVMs) as classifiers, where high classification accuracy was achieved across various torque levels. [11] Wang translated speech recognition technology using Mel Frequency Cepstrum Coefficients (MFCC) and classified the data using Least Squares SVMs to improve the

practicability of percussion methods based on cyber physics systems. [19] Yuan introduced intrinsic multiscale entropy analysis in combination with back-propagation neural networks (BPNNs), focusing on signal decomposition while processing percussion signals to achieve enhanced sensitivity in looseness level differentiation. [12] Wang and Song employed multifractal analysis for feature extraction from impact signals and combined them with a gradient boosting decision tree classifier, introducing a novel method which extracted richer discriminative information rather than features alone. They further explored this idea by using a 1D-MACLSTM network algorithm for loosening detection, which outperformed the existing techniques. [13] In the same year, Wang and Song proposed another novel method of training interference capsule neural networks (1D-TICapsNet) to classify percussion-induced signals and achieve early bolt looseness detection. 1D-TICpasNet outperformed several state-of-the-art deep learning techniques in terms of accuracy, computational costs and denoising capacity. [21] To circumvent the environmental noise captured by the microphone during percussion, Zhou used laser Doppler vibrometry to simultaneously capture the bolt's vibrational response during percussion. They created spectrograms and then processed the data using a 2D convolutional neural network to achieve superior identification accuracy. [14]

The percussion method has also been adapted for challenging environmental conditions. Undersea pipes and flanges ensure the transfer of oils and other substances over long distances via water. Vibrations induced during the flow, as well as the external factors, can cause the bolts to lose preload. Operational failure here can hamper aquatic life as well as cause unprecedented loss. Chen thus proposed the use of percussion-based method in an underwater environment where audio signals from a submerged bolted flange were combined with a Feature Reduced Multiple Random Convolution Kernel Transform (FM-ROCKET) machine learning model by integrating deep and shallow learning to classify looseness states in conditions where conventional sensing approaches are difficult to use. [15] Wang proposed the Acoustic Emission technique using sensor data to identify the multi-bolt looseness by using dual-shapelet network classifiers to discriminate the AE waves. They addressed the imbalanced classification task, i.e., the minority categories. Compared to the baseline strategies, the AE method boasted significant superiority for the classification of multi-bolt looseness states. [18] Interference of noise can be a big challenge while employing percussion-based methods. A monitoring method based on the All-Pole Group Delay Function (APGDF) and a prototypical network was integrated within the percussion method by Liu for cases

of bolt loosening where there are insufficient samples and noise interference. This method demonstrated some major accuracy advantages with only 50 training samples per class and -6dB signal-to-noise ratio (SNR). [20] Du addressed the challenge of simultaneous loosening across multi-bolt structures by using percussion with the DenseNet-CBAM deep convolutional architecture. The proposed method achieved the detection accuracy of more than 97% in several structures with multiple threaded fasteners tapped in sequence. [16] This is the only work on percussion based multi bolt loosening detection as far as the knowledge of the author.

Feblil Huda applied a method for loose bolt detection by applying a statistical evaluation of the Recognition-Taguchi (RT) method to a six-bolt joint cantilever with a loose bolt. Frequency responses obtained from the finite element analysis and the experiments using the laser excitation were able to detect and identify loose bolt positions [17]

Following upon the recent advancements, State Space Sequence Models (SSM) have been used in signal processing and dynamical systems. The state space models make predictions about dynamic systems by modelling how the internal state evolves using differential equations. In earlier SSM models, the state update was a fixed linear transformation. Gu and Dao in 2024 developed a linear time sequence modelling with the Selective State Spaces called Mamba, which used input tokens to influence how states update themselves rather than using fixed parameters for state update, making them effective for handling long-range dependencies. [24] Haiming Yi in 2025 initiated the use of SSM models (VibrMamba) in the field of fault diagnosis of rotating machinery by using a one-dimensional vibration signal as input. This model showed clear advantages over mainstream CNN and other fault diagnosis models for industrial applications. [25] Further, Peng Wang used the mamba model with Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEDMAN) and bi-directional feature fusion labelled as MD-BiMamba in the fault diagnosis of aero-engine inter-shaft bearing. This method achieved 99.88% and 95.81% accuracy in the standard dataset and -10dB noise environments, respectively. [26] The state space models, however, haven't been used to classify and detect the bolt loosening conditions and their preload states yet.

Recently, percussion-based methods integrated with machine learning algorithms have provided a combination of ease for data acquisition and classificatory advantages of machine learning. Many studies have successfully employed this tapping and listening while training ML models to classify different preload levels in bolted structures. [10,11,12,13,14,15,16,18,19,20,21] Furthermore, the

usage of smartphones to record the impact sound, as seen in the Figure 2.1, makes the data acquisition part simpler and practical. The basic idea behind this methodology is that, when the bolt is tight, the structure is unified and behaves in a stiff manner. When the bolt loses its preload, the joint is not as stiff, and the parts move independently. This changes the dynamic behavior of the system when subjected to impact/loading and thus alters the acoustic signature of percussion data. When the contact is loose, the damping behavior of the system changes drastically, with even the microslips at the structural interface resulting in some motion in the system. This micro slip loses some energy, and the audio signals are dull and short as the energy is absorbed at the loose interface.

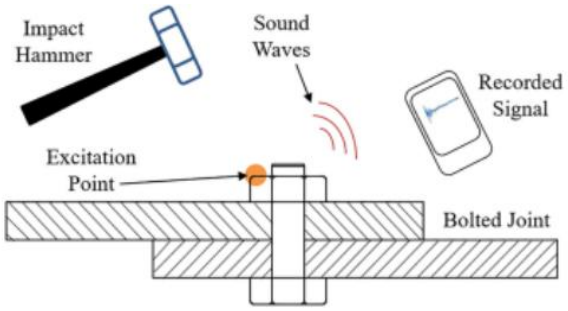

Figure 2.1: Schematic diagram of tapping and listening method

For multi-bolted structures, the acoustic pattern should be different as the energy loss and system damping are different for all the potential combinations. This change in pitch might be insensitive to the human ear, but when the audio features are extracted efficiently, the machine learning algorithms can discriminate between even the minor alterations, thus classifying the system state.

2.1 Signal Processing and Feature Extraction

A vast amount of information can be extracted from an audio signal by processing it in several ways. Initially, the raw signal is in the time domain, where statistical and wave-based features are dominant. Performing a Fourier analysis on this signal reveals frequency-dominated features where the resonance, natural frequencies and other critical information reside. Furthermore, time-frequency-based processing reveals the nonlinearity of signal properties and is crucial for studying the impact-based implications on audio signals. These are the usual spectrum analyses. Cepstrum

analysis, used in speech recognition, extracts texture from the signal and can be essential for distinguishing minor variations in signal features.

Following up the recent advances, different machine learning based feature extraction methods are being used that reduce the high-dimensional features into trainable parameters. Some methods even process the sequential data to capture decay and temporal dependencies in the data.

2.2 Machine Learning Methods

The integration of machine learning into percussion-based structural health monitoring has significantly improved the ability to distinguish bolt looseness conditions from audio and vibrational signals. ML-based frameworks map extracted signal features to discrete preload categories, ensuring a practical, accurate and scalable procedure. With time, the traditional classifiers have been replaced by deep learning architectures which overcame the challenges of accuracy, noise sensitivity, limited data and multi-bolt applications.

Support Vector Machines (SVMs) are the widely used classical method that integrates well with MFCC under small sample conditions. Combined with multifractal features, the tree methods (Random Forest, Gradient Boosted Decision Tree) also boast good performance with decent interpretation. Back Propagating Neural Networks (BPNN) was used as an early deep learning baseline but with the integration of Convolution Neural Network (CNN) based algorithms and their ability to work with spectrograms, scalograms, as well as raw 1D waveforms, CNNs have been quite dominant. For small datasets, hybrid CNN and Long Short-Term Memory (LSTM) models have been used to capture time domain characteristics that carry preload-based information while performing well in noisy conditions. Since CNN is a scalar input-based architecture, its vector counterpart, Capsule Networks (CapsNet), offers end-to-end raw waveform classification with significant sensitivity to loosening, providing the best alternative in small dataset regimes.

CHAPTER THREE: RESEARCH METHODOLOGY

The present study started with formulating the problem of multi-bolt preload classification using vibration-based techniques. A thorough literature review was conducted to examine existing machine learning and deep learning approaches for bolt loosening detection, leading to the identification of key research gaps involving multi-bolt studies.

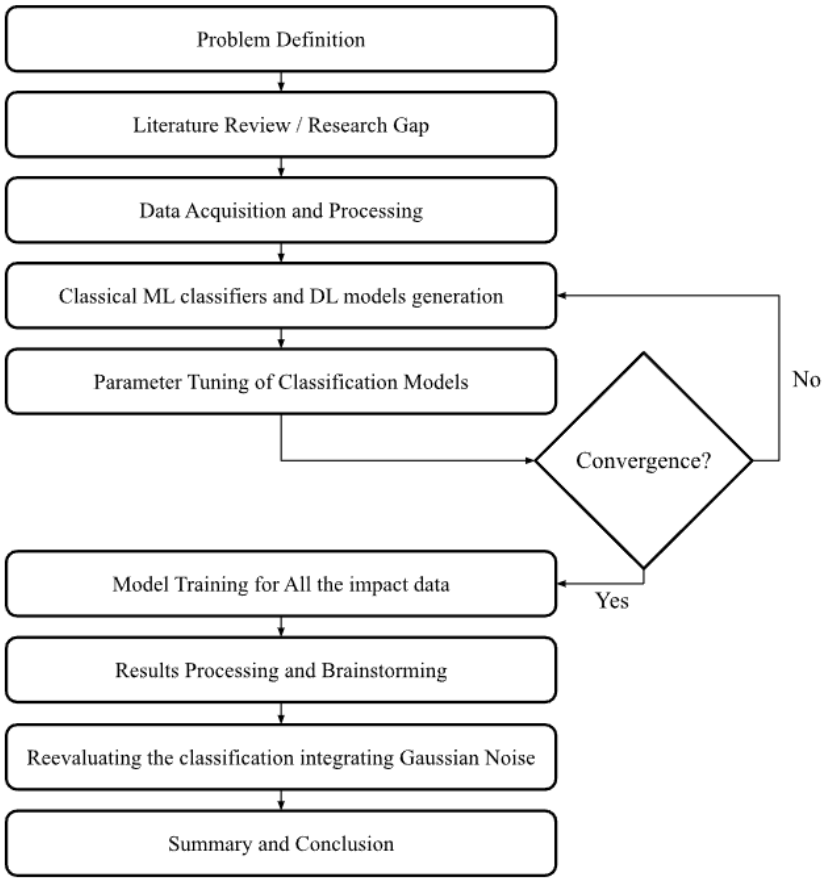

Figure 3.1: Methodology Flowchart

An experimental setup based on a cantilever structure was designed, and percussion-based excitation was applied at multiple impact locations to acquire vibration data for different loosening cases. The recorded signals were then preprocessed and segmented into individual impact events. Three classification models were then selected, and their architectures and hyperparameters were systematically defined and optimized. The models were trained using the processed datasets

corresponding to different impact locations and configurations. Finally, classification results were obtained and analyzed to evaluate the effectiveness of each approach in accurately identifying multi-bolt preload conditions. To quantify the influence of noise in the classification performance, a gaussian white noise of different SNR is integrated to the test signals after the model is trained with noise free acoustic signals. The accuracy of classification of this noisy test dataset is used to evaluate the noise robustness of the employed methodologies.

3.1 Experimental Setup

To train an ML/DL model, large amount of data is required. An experimental setup was built using aluminum extrusions, which was cantilevered in nature, had a base plate on top supported using six hex bolts, as seen in the Figure 3.1.

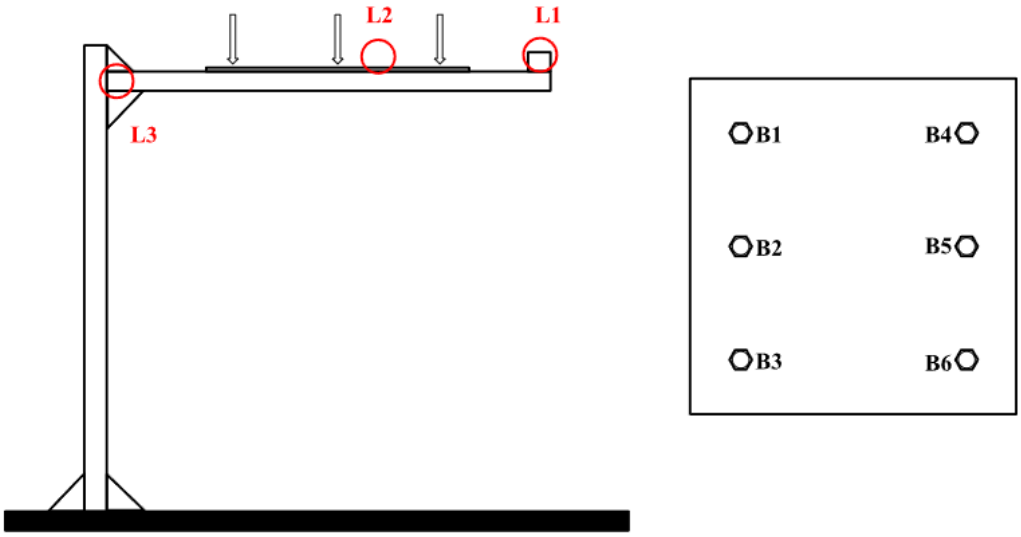

Figure 3.1: Experimental setup with the bolt locations indicated by arrow and impact locations indicated by red circles

Six M10 bolts were used to mount a 365mm*300mm steel plate to the aluminum structure. A total of three preload conditions, completely loose (0Nm preload), medium loose (20Nm preload) and tight (40Nm preload) were defined for each of the bolts. Three impact locations, one at the root of the cantilever, one at the tip and one at the base of the mounted plate were defined for percussion using an impact hammer, as seen by red circles in Figure 3.1. With six bolts with three potential preload conditions, there could be 729 potential states of these bolts. Acquiring data for each of

these states at three impact locations would be tiresome. So, for simplicity, the Latin Hypercube Sampling (LHS) method to search the design space was used to obtain 27 potential states, which covered the space with utmost diversity. So, a total of 81 sets of data were obtained for the given structure.

3.2 Latin Hypercube Sampling for bolt preload states

The 27 states that were provided by LHS and used in this study are tabulated in Table 3.1. Here, 0 is equivalent to loose state, 1 is medium loose, and 2 is tight.

Table 3.1: Preload states across the six bolts generated using Latin Hypercube Sampling

State	Bolt 1	Bolt 2	Bolt 3	Bolt 4	Bolt 5	Bolt 6
1	1	0	1	1	1	0
2	0	0	2	0	0	1
3	2	1	1	2	0	2
4	0	2	2	0	1	0
5	0	1	2	0	1	1
6	2	2	0	2	0	1
7	2	2	2	1	1	2
8	2	2	0	1	1	2
9	1	1	2	2	0	1
10	2	2	0	2	2	1
11	1	0	1	2	1	1
12	1	1	1	0	0	1
13	1	0	2	2	0	0
14	0	0	1	0	2	1
15	2	2	2	1	2	2
16	2	1	0	0	0	2
17	0	0	2	1	0	0
18	1	1	1	1	2	2
19	0	1	1	0	2	0
20	0	2	0	1	2	1
21	0	0	0	0	1	0
22	0	1	0	2	1	2
23	2	2	1	1	2	0
24	1	0	1	0	2	2
25	1	0	0	2	1	0
26	2	2	0	1	2	0
27	1	1	2	2	0	2

3.3 Data collection and processing

After the setup was completed, 100 samples of percussion data were collected for each state at each impact location, generating a total of 8100 data samples. iPhone 14 was used to record the audio data in .m4a format, which had a standard sampling rate of 48kHz.

This data was converted into a waveform format (.wav) using Fast Forward Moving Picture Express Group (FFmpeg), which is usually read by the Python libraries and existing machine learning models. FFmpeg is a free open-source software for handling video, audio and other multimedia files. Each of the obtained data had 100 impacts, which had to be segmented into individual signals, which was done by creating a rectangular window of 10ms over each region having high energy, i.e., the impacts. If the impact decay window is less than 10ms, it's discarded as a spike or noise, and the rest of the impact data is segmented using 50ms onset before and 150ms onset after the impact. Each of these segments is then classified under the same class.

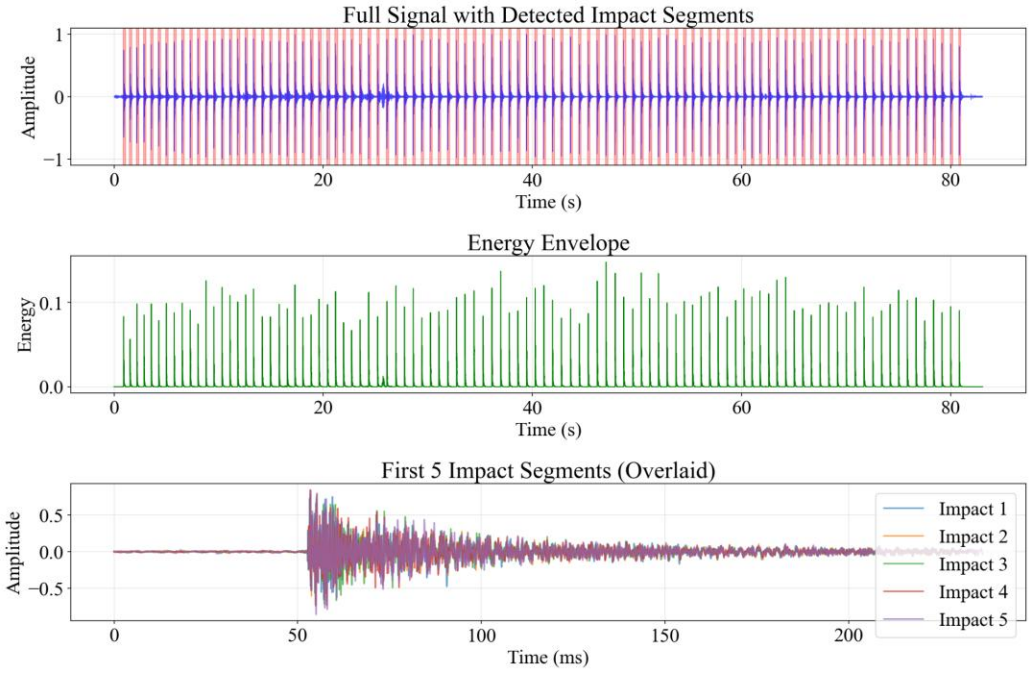

Figure 3.2: Data Segmentation using the energy envelope

As seen in the figure, the instances that have energy below the threshold are disregarded as noise and actual impacts are segmented. The nature of the individual segment can also be seen in the figure 3.2.

3.4 Machine Learning models for classification

For training models, some major classification algorithms like PSD+GBDT, MFCC+SVM, along with deep learning architectures 1DCNN and hybrid 1DCNN-Mamba were used. To compare the performance of these models, equivalent models were built with a similar number of layers and systems. Furthermore, the number of samples per class was set to 40, 60, and 80 to quantify the performance of each of these models on a limited dataset. A white noise of three separate energy levels was also used in the data to observe the noise-resistant characteristics of each of the models. A completely reproducible framework was developed on the same split of percussion dataset and comparable parameters to ensure that any accuracy difference comes from the model architecture and not from the data splits, preprocessing or any hyperparameters.

Model parameters of each of the said models are tabulated as:

Table 3.2: Machine learning model architecture

Model Architecture	**Parameters**
PSD + DT (Power Spectral Density + Gradient Boosting Decision Tree)	No of FFTs for PSD = 2048 No of bands = 40 No. of estimators=100 Max Depth =5 Learning Rate = 0.05 Sub Sample = 0.8 Minimum Sample Per Leaf = 5
MFCC + SVM (Mel Frequency Cepstrum Coefficients + Support Vector Machine)	No. of MFCCs = 20 Mel Filter Banks = 128 STFT Window Size = 2048 STFT Hop Size = 512 Kernel = RBF C = 1.0 Gamma = Scale

1D-CNN (One-Dimensional Convolutional Neural Network)	Optimizer = Adam Learning Rate = 1e-3 Loss = Sparse Categorical Cross entropy Activation Function = ReLU Layers: 1. 1D Conv: 32 filters, kernel size = 80, stride = 4 2. 1D Max Pooling: Size = 4 3. 1D Conv: 64 filters, kernel size = 3 4. 1D Max Pooling: Size = 4 5. 1D Conv: 128 filters, kernel size = 3 6. 1D Max Pooling: Size = 4 7. Flattening Layer 8. Fully Connected Dense Layer: 128 Neurons, ReLU 9. Dropout Layer: 0.5 10. SoftMax Dense Later Epochs = 50 Batch Size = 32 Patience = 10
Mamba + CNN	Optimizer = Adam Learning Rate = 1e-3 Loss = Sparse Categorical Cross entropy Activation Function = ReLU Layers: 1. 1D Conv: 32 filters, kernel size = 80, stride = 4 2. 1D Max Pooling: Size = 4 3. 1D Conv: 64 filters, kernel size = 3 4. 1D Max Pooling: Size = 4

	5. 1D Conv: 128 filters, kernel size = 3 6. 1D Max Pooling: Size = 4 7. Layer Normalization 8. Mamba Block a. I/O dimension = 64 b. Expended Inner dimension = 128 c. SSM latent state size = 16 d. Kernel Width = 8 e. Inner expansion factor = 2 9. Layer Normalization 10. Global Average Pooling 1D 11. Fully Connected Dense: 128, ReLU 12. Dropout: 0.4 13. SoftMax Dense Later

3.5 Hybrid Mamba-CNN architecture

The Mamba-CNN couples a 1D-CNN front-end with a Mamba selective state space block. The CNN layers extract local features, producing a feature sequence X ∈ ℝ^{T×D} that is processed by the Mamba block for long-range temporal modelling, a capability particularly relevant for signals from cantilever structures where global vibration modes extend beyond the receptive field of convolutional layers.[24]

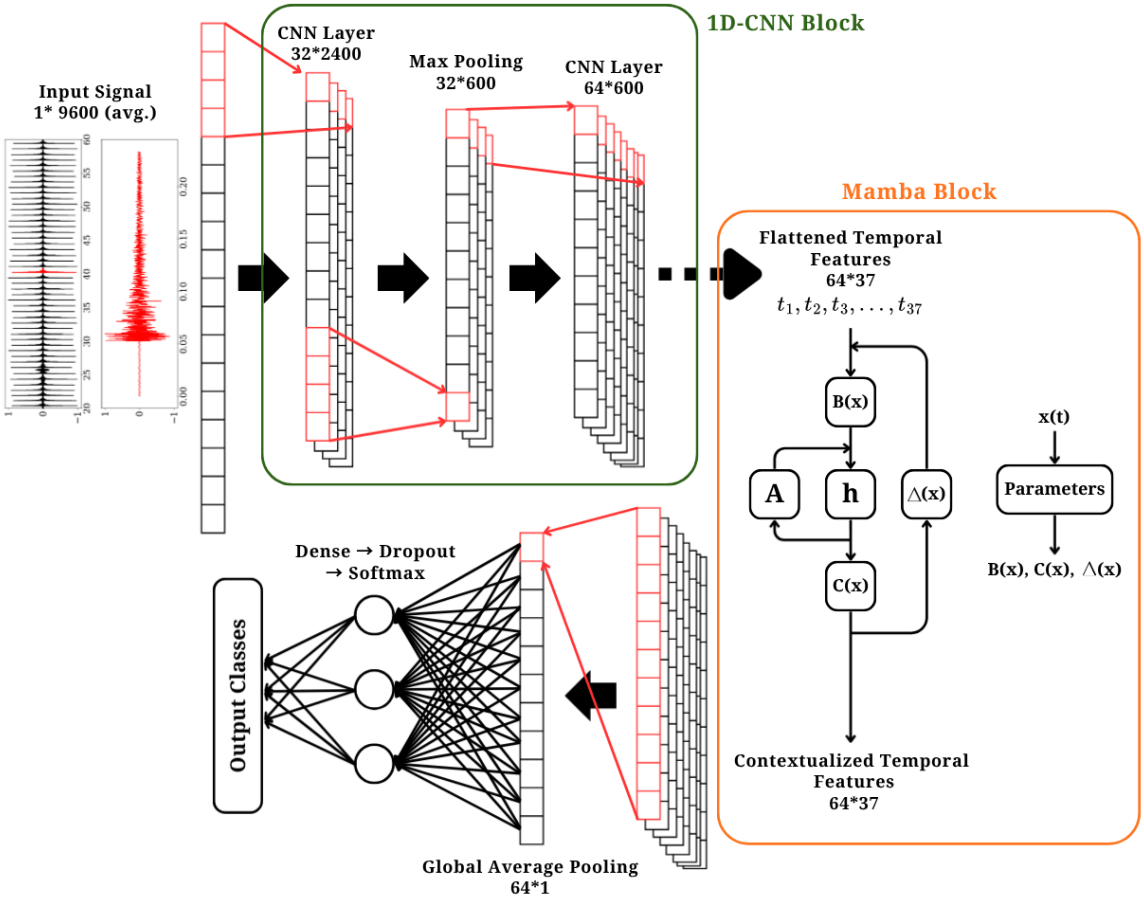

Figure 3.3: Hybrid 1DCNN-Mamba combined architecture

The classical linear time-invariant (LTI) state space model maps a scalar input sequence $x(t)$ to an output $y(t)$ through a state vector $h(t) \in R^N$.

$$\dot{h}(t) = A\,h(t) + B\,x(t)$$

$$y(t) = C\,h(t)$$

Where $A \in R^{N\times N}$ is the state transition matrix, $B \in R^{N\times 1}$ is the input projection, and $C \in R^{1\times N}$ is the output projection.

For discrete percussion signal sequences, the continuous matrices are converted into their discrete equivalents using zero-order hold (ZOH) discretization with the learnable step size Δ:

$$\bar{A} = e^{\Delta A}$$

$$\bar{B} = \Delta A^{-1} e^{\frac{\Delta A - I}{B}}$$

Their discrete recurrence is then given by:

$$h(t) \; = \; \bar{A}\, h(t-1) + \bar{B}\, x(t)$$

$$y(t) \; = \; C\, h(t)$$

In LTI state space models, A, B, C and Δ are fixed across all the time steps, preventing the model from selectively retaining or discarding information based on the current input. To address this, Gu et al. introduced a state space model (S6) where the step size $\Delta(t)$ and projection matrices $B(t)$ and $C(t)$ are computed as direct functions of the current input token $x(t)$:

$$B(t) \; = \; Linear_B(x(t))$$

$$C(t) \; = \; Linear_C(x(t))$$

$$\Delta(t) \; = \; Softplus(Linear_\Delta(x(t)))$$

The time varying discrete recurrence of the S6 model is therefore:

$$H(t) = A(\bar{t})\, h(t-1) + B(\bar{t})\, x(t)$$

$$y(t) \; = \; C(t)\, h(t)$$

This input-selective mechanism enables the model to dynamically retain or suppress information at each time step based on signal content. In the context of the present study, this property allows the model to attend to bolt-state-dependent impact features while suppressing the superimposed structural vibration from the cantilever, which is the primary source of noise-like interference degrading classical model performance. The Mamba block output is then passed to a fully connected SoftMax layer to produce the 27-class joint state prediction.

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1 Hyperparameter tuning for classical models

Before the main performance evaluation, the sensitivity of the two classical model pipelines to their key hyperparameters was evaluated to establish the final configurations used in all following experiments.

For the MFCC+SVM pipeline, classification accuracy was evaluated as a function of the number of MFCC coefficients L. Accuracy increased sharply from approximately 60% at L = 2 to above 95% at L = 10, then plateaued at approximately 99% beyond L = 20, as shown in Figure 4.1. This saturation behavior indicates that the preload-related acoustic information content of the percussion signal is concentrated in the low-order cepstral coefficients, which encapsulate a rough shape of the spectral envelope. Higher-order coefficients which are generally characterized with finer spectral detail, add noise to the feature representation rather than the discriminative signal. Based on this analysis, L = 20 was selected for all the subsequent MFCC+SVM experiments.

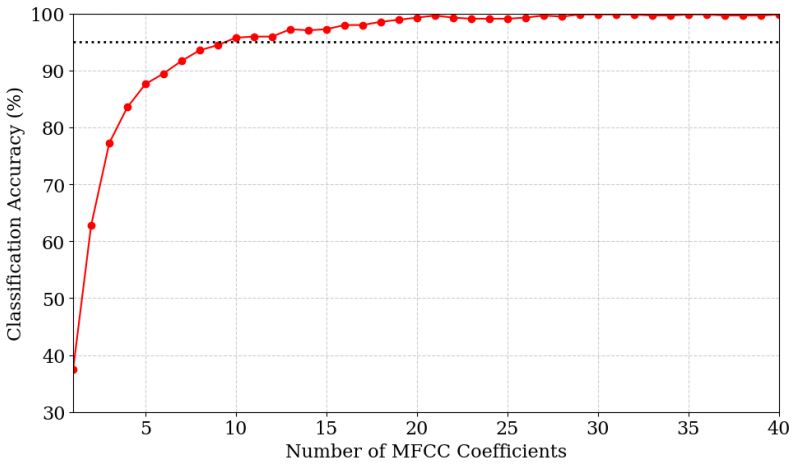

Figure 4.1: MFCC+SVM model independence with respect to mel coefficients

For the PSD+GBDT pipeline, accuracy was evaluated as a function of both the number of estimator trees M and the maximum tree depth. Accuracy saturated at approximately 94% beyond M = 100 trees with negligible marginal gain per additional tree, and depth saturation occurred at a maximum depth of 5 (Figure 4.2 and Figure 4.3). The saturation at shallow depth indicates that the discriminative structure of the 40-band PSD feature space is relatively simple, and deeper trees

primarily overfit rather than extract additional information. Final configurations of M = 100 trees and a maximum depth of 5 were adopted.

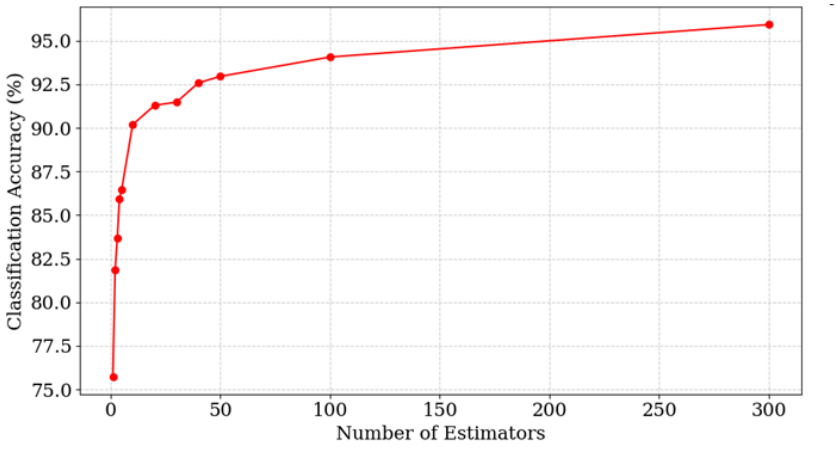

Figure 4.2: PSD+GBDT model independence with respect to the number of estimators count

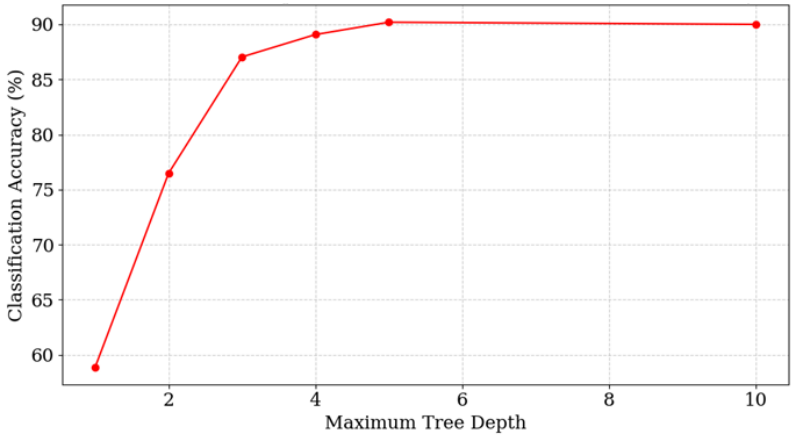

Figure 4.3: PSD+GBDT model independence with respect to the maximum decision tree depth

4.2 Classifiability of dataset analysis

The MFCC+SVM classification operates in an 80-dimensional feature space (20 coefficients × 4 statistical data), which cannot be visualized directly. Thus, the classification structure was

examined using t-distributed Stochastic Neighbor Embedding (t-SNE) to project the 80-dimensional feature vectors to two dimensions while preserving local neighborhood relationships, as shown in Figure 4.4. The 27 joint states provide largely well-separated clusters in the 2D projection, confirming that the MFCC feature representation provides sufficient information to distinguish the 27 preload combinations. A partial overlap can be seen in several clusters which correspond to the joint states that differ in preload condition on only one or two bolts. These states produce similar global stiffness and therefore similar acoustic response, which is reflected in their closeness in the feature space. This observation is also consistent with the confusion matrix patterns.

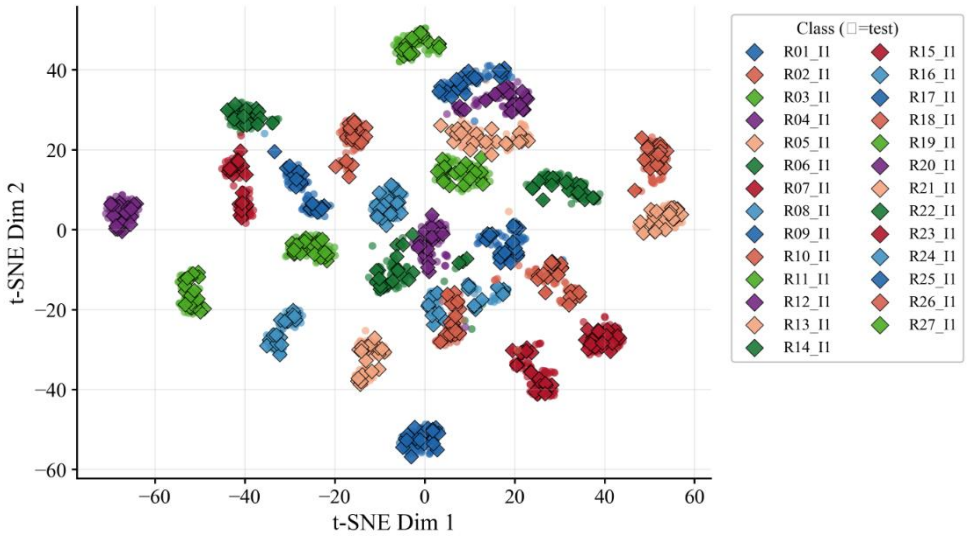

Figure 4.4: t-SNE-based visualization of MFCC+SVM-based hyperplane classification in 80-D space

The PSD feature importance analysis in Figure 4.5 reveals that frequency bands 14 and 19, corresponding to the mid-frequency range of the percussion response have the highest Gini importance scores among the 40 bands. This is physically interpretable as the impact response of the bolted cantilever structure in this frequency range is dominated by the second and third bending modes, which are most sensitive to changes in joint boundary conditions. Lower-frequency bands represent the fundamental cantilever mode, which is relatively insensitive to bolt preload changes at the individual bolt level, while very high-frequency bands are attenuated by structural damping before reaching the microphone.

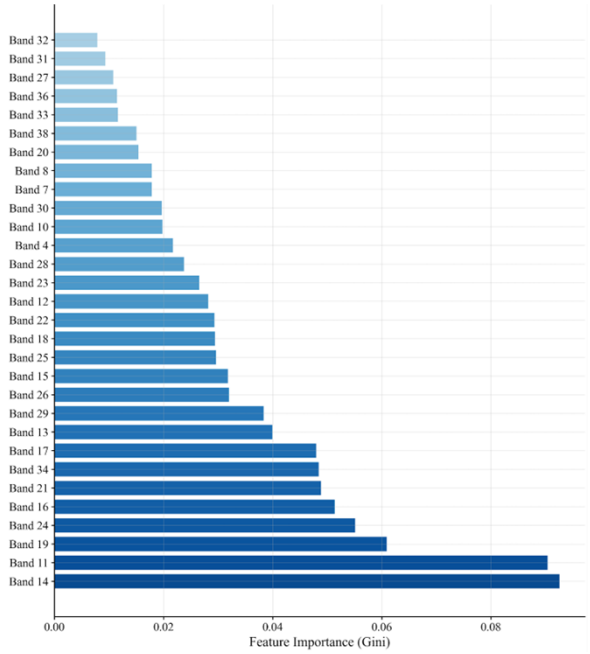

Figure 4.5: Top 30 features out of 40 for PSD+GBDT classification

4.3 Classification accuracy in noise-free analysis

Table 4.1 presents classification accuracy for all four models across the three impact locations and three dataset sizes under noise-free conditions. All models have achieved accuracies above 90% across all configurations indicating that the percussion-based approach is usable for multi-bolt discrete preload levels classification.

Table 4.1: Classification accuracy (%) across models, impact locations, and dataset sizes under noise-free conditions. L1- cantilever end; L2- bolt region; L3- cantilever root.

Model	L1: 80-20	L1: 40-60	L2: 80-20	L2: 40-60	L3: 80-20	L3: 40-60
MFCC+SVM	99.63	99.20	99.72	99.50	99.80	99.32
PSD+GBDT	97.78	93.76	96.67	94.88	94.45	90.57
1D-CNN	98.88	98.70	99.26	99.14	97.23	98.15
Mamba-CNN	97.77	95.37	97.97	96.49	98.51	97.17

The MFCC+SVM model achieved the highest accuracy across all conditions of 99.8% at Location 3 with the 80-20 split and maintained the accuracy higher than 99.2% across all configurations. Given the 27-class complexity of the problem, this result is reliable. It indicates that the MFCC representation of the 200 ms percussion segment, when combined with the maximum-margin classification of SVM, is sufficient to classify the full combined state space of the six-bolt joint under controlled conditions. The negligible accuracy variation across dataset sizes (less than 0.5 percentage points between the 80-20 and 40-60 splits) reflects the strong feature identity as observed in the t-SNE analysis, where each class occupies a discrete, well-separated region in the MFCC feature space which requires relatively few training samples to define reliable class boundaries.

The PSD+GBDT model showed a higher sensitivity to the data than the MFCC+SVM pipeline, with accuracies ranging from 90.6% to 97.8% across splits and locations. The lower performance peak as compared to the MFCC+SVM is consistent with the feature importance analysis. PSD captures the full spectral resolution without the perceptual compression of the Mel scale and it treats the frequency bands that carry little discriminative information in similar weight to the bands which separate the classes. The GBDT classifier compensates this partially through its tree-based feature selection, but this generalization diminishes under limited data due to the higher dimensionality of the raw PSD representation relative to the optimized MFCC vector.

The 1D-CNN and Mamba-CNN models achieved accuracy somewhere between the two classical methods, with 1D-CNN reaching up to 99.5% and Mamba-CNN up to 98.5%. The slight underperformance of deep learning models relative to MFCC+SVM in clean conditions is consistent with the general observation that end-to-end models require larger datasets to fully utilize their representational capacity. With only 80 samples per class in the most data-rich condition, the convolutional feature learning is constrained relative to what would be achievable with larger datasets. The Mamba-CNN's slower convergence compared to the 1D-CNN reflects the additional parameters in the Mamba block, which require more gradient steps to optimize effectively.

4.3.1 Influence of Impact Location

Across all four models, the impact at Location 2 (bolt region) consistently achieved the highest or near-highest accuracy, closely followed by Location 3 (cantilever root), with Location 1 (cantilever end) performing lowest under data-limited conditions. This pattern clearly indicates that the

percussion at the bolt region excites a wave that travels directly through the bolted joint, maximizing the proportion of the captured signal that has the joint stiffness information. Percussion at the cantilever tip must propagate along the full beam length before reaching the joint, during which it accumulates energy in the cantilever's global bending modes. These modes are relatively insensitive to bolt-level preload variations, diluting the preload-related information content of the received signal. An intermediate position is occupied by the cantilever root excitation which drives the signal into the joint through the beam, but from the clamped end, where bending mode participation is lower. This finding has directly indicated that in the percussion based SHM, impact location selection should priorities proximity to the fastener region to maximize the reliability of classification.

4.4 Classification performance at noisy conditions

The confusion matrices were generated for all four models at 40 dB SNR (Figures 4.6 a–d). These matrices indicate a strong diagonal alignment at this noise level for all four models adhering to a reliable distinction across 27 joint states across 6 bolts. A close examination of the off-diagonal elements reveals a consistent misclassification pattern across models. The states that differ only in the preload condition of a single bolt are the most frequently confused pairs. It is physically expected that the aggregate stiffness of all connections governs the global acoustic response of the multi-bolt joint, and a single-bolt preload change produces a smaller change in the overall dynamic signature than a multi-bolt change. This observation implies a practical limit of single-impact percussion-based classification in a six-bolt system that lies in resolving states that are adjacent in the preload configuration space, and future work can address this through multi-impact averaging or targeted local excitation near individual bolts.

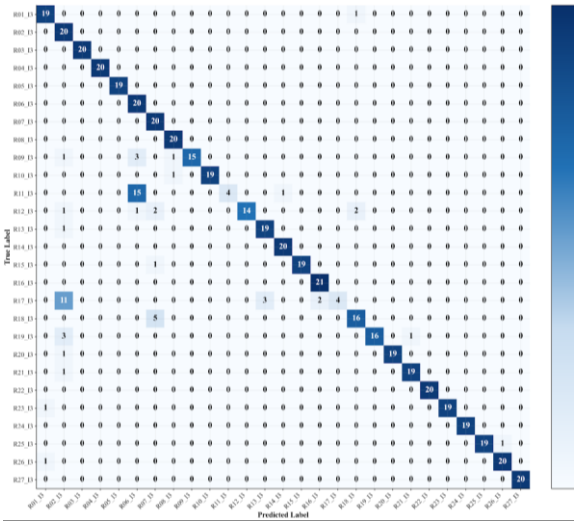

Figure 4.6 (a): Confusion Matrix of MFCC+SVM classification at 40dB SNR

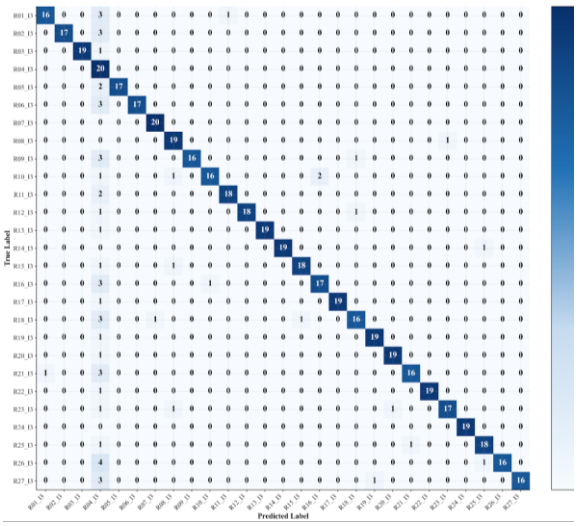

Figure 4.6 (b): Confusion Matrix of PSD+GBDT classification at 40dB SNR

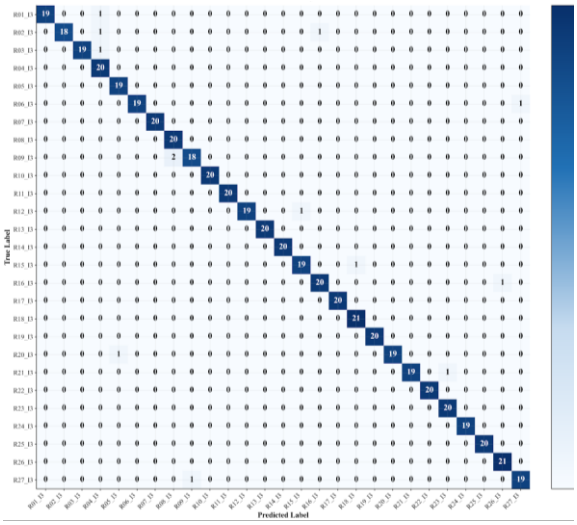

Figure 4.6 (c): Confusion Matrix of 1D-CNN classification at 40dB SNR

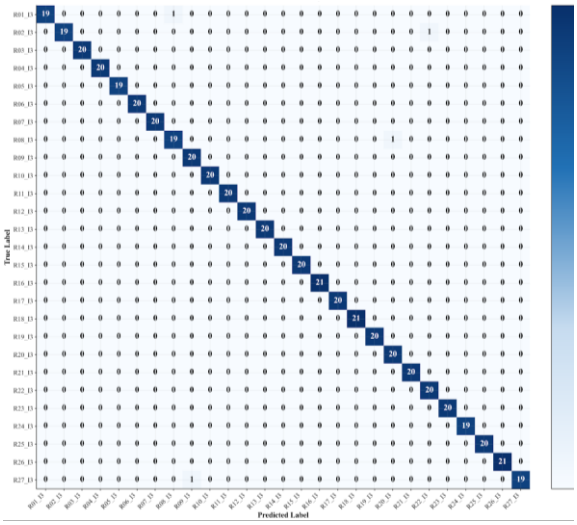

Figure 4.6 (d): Confusion Matrix of Hybrid Mamba-CNN classification at 40dB SNR

The deep learning models (1D-CNN and Mamba-CNN) have fewer off-diagonal classifications than the classical models at the same SNR level representing their higher robustness to noise.

The results in Table 4.2 summarize the classification accuracy for all four models across all five SNR levels. This central result reveals a fundamental divergence between the two model families under realistic noise conditions.

Table 4.2: Classification accuracy (%) as a function of SNR for all four models (Location 2, 80-20 split). Models were trained on clean data; noise was injected post-training.

Model	20 dB	30 dB	40 dB	50 dB	60 dB
MFCC+SVM	9.1	22.6	88.9	99.5	99.6
PSD+GBDT	76.3	83.0	87.6	92.0	92.4
1D-CNN	97.2	98.1	98.5	98.6	98.9
Mamba-CNN	98.9	98.9	98.9	98.9	98.9

The MFCC+SVM pipeline indicates a catastrophic reduction of accuracy at low SNR, decreasing from 99.8% under noise free conditions to 9.1% at 20 dB SNR barely above the chance level of 1/27 = 3.7%. This collapse is interpretable from the t-SNE analysis. The 27-class clusters in MFCC feature space are well-separated under clean conditions, but additive noise corrupts the higher-frequency content of the signal that contributes to the short-time spectral estimates, shifting the MFCC feature vectors substantially from their clean positions. At 20 dB SNR, the noise-shifted clusters overlap aggressively, resulting in the SVM hyperplanes learned from clean training data ineffective. The recovery begins at approximately 40 dB SNR (88.9%) and it is near-complete by 60 dB (99.6%) showing that the noise floor at which MFCC features remain reliably distinguishable is approximately 50 dB SNR for this application.

The PSD+GBDT model has a substantially greater noise tolerance than MFCC+SVM, maintaining 76.3% at 20 dB SNR and recovering to 92.4% by 60 dB. This relative robustness likely reflects the band-averaging inherent in the 40-band PSD representation where the averaging energy across broad frequency bands reduces the sensitivity of individual features to narrow-band noise realizations. The GBDT classifier's ensemble character, in which the final prediction is determined by majority vote across 150 trees, each examining a random feature subset, additionally provides implicit noise averaging that single-classifier methods lack.

Both deep learning models maintained substantially higher accuracy across all tested SNR levels. The 1D-CNN achieved 97.2% at 20 dB SNR with only marginal variation up to 60 dB (98.9%),

demonstrating that raw waveform classification by learned convolutional features is inherently more noise-tolerant than handcrafted spectral features. The learned filters of CNNs which are optimized end-to-end for classification, effectively weight the time-frequency regions that are both most discriminative and most noise-robust instead of treating all spectral components equally as done by the MFCC pipeline.

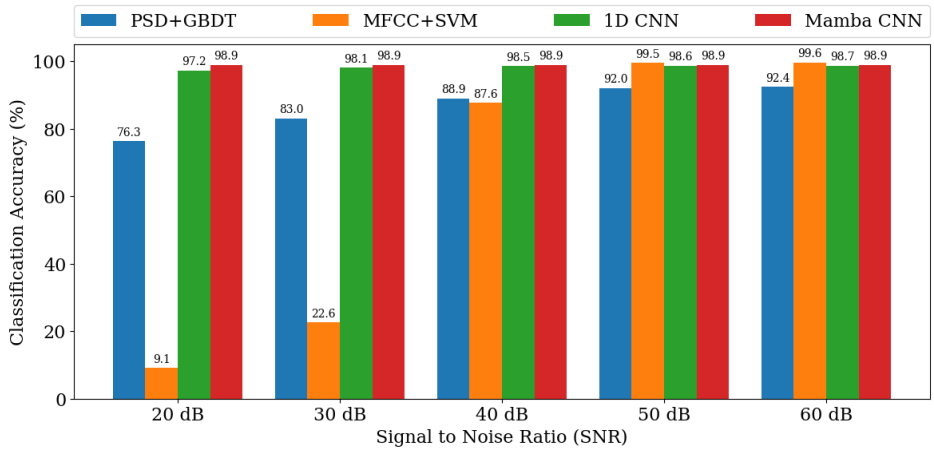

Figure 4.7: Performance of different classification models at discrete noise levels

The Mamba-CNN achieved 98.9% at 20 dB SNR and maintained this accuracy with negligible variation across all SNR levels from 20 to 60 dB, as shown in Figure 5.7. This near-complete noise invariance is a qualitatively distinct behavior from all other models evaluated. It is attributed to the selective state update mechanism of the S6 architecture: at each time step, the matrices B_t, C_t, and step size Δ_t are computed as functions of the current input, enabling the model to dynamically suppress contributions from time steps dominated by stochastic noise while retaining contributions from the structurally informative impact transient and its reflections. This is fundamentally different from static filtering or frequency-domain averaging; the selection is content-based and adapts to the specific noise-signal structure at each point in the sequence. The training curves (Figure 4.8 and 4.9) provide additional evidence for the robustness of the learned representation: the Mamba-CNN exhibits a smaller train-validation generalization gap than the 1D-CNN (visible in the closer tracking of validation loss to training loss), suggesting that the Mamba block's sequential state modelling provides implicit regularization against overfitting to specific noise-free signal realizations.

A clear practical recommendation can be established using these results. For controlled laboratory or industrial environments where SNR exceeds approximately 50 dB, the MFCC+SVM pipeline offers excellent accuracy with minimal computational cost and requires no GPU infrastructure. For realistic industrial usage where SNR may fall below 40 dB due to operational machinery noise, electrical interference, or wind loading, the Mamba-CNN is the recommended architecture, providing reliable classification with accuracy above 98.9% regardless of noise level.

4.5 Training convergence of DL models

The training accuracy and loss curves for both deep learning models are presented in Figures 4.8 and 4.9. A rapid convergence is observed in the 1D-CNN, achieving 80% validation accuracy within the first 10 epochs and reaching its best validation performance at epoch 19 of 50. This fast convergence indicates the efficiency of convolutional feature learning on structured 1D signals. The local temporal patterns in the percussion waveform are consistent with each other such that even a small number of gradient updates is enough to orient the classifier toward distinct frequency-time regions.

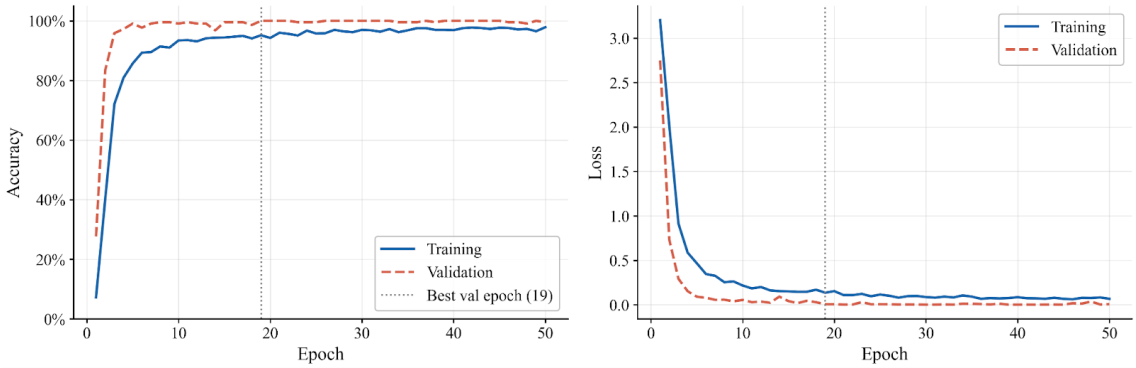

Figure 4.8: Training accuracy and loss curves for 1D-CNN

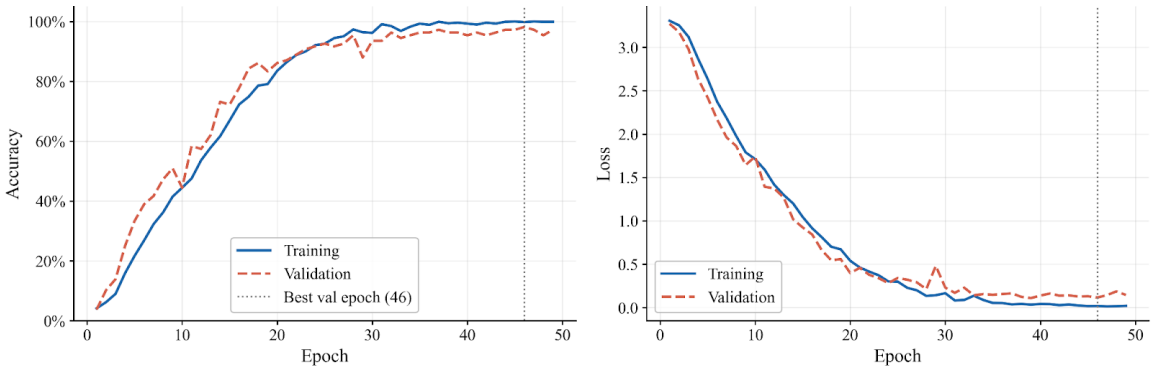

Figure 4.9: Training accuracy and loss curves for Hybrid Mamba-CNN

The Mamba-CNN exhibited a slower, more gradual convergence trajectory, surpassing 80% validation accuracy by epoch 20 and reaching its best validation performance at epoch 46 of 50. Both training and validation accuracy increased in parallelly throughout training, with the validation loss tracking the training loss closely and showing no evidence of overfitting within the 50 epoch of model training. This behavior is consistent with the Mamba block functioning as a form of implicit regularization. The recurrent state integration distributes the gradient signal across the sequence length while preventing individual layers from overfitting to specific local signal patterns. The Mamba-CNN model maintained a good accuracy above 95% across all the impact locations even with the smallest dataset (40-60 split) indicating that the learned representation is both data-efficient and location generalized.

CHAPTER FIVE: CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This study demonstrated the possibility of simultaneous classification of discrete bolt preload levels across a multi-bolt joint with high accuracy using percussion-based acoustic signal analysis. Twenty-seven distinct joint states, defined by three preload levels across six bolts and selected via Latin Hypercube Sampling, were classified using four model architectures evaluated on an aluminum cantilever test structure.

All four models, MFCC+SVM, PSD+GBDT, 1D-CNN, and Mamba-CNN, achieved classification accuracies above 90% across all impact locations and dataset sizes under the noise-free conditions. MFCC+SVM reached a peak accuracy of 99.8%. This establishes that a multi-state multi-bolt preload classification is feasible using a single smartphone microphone and a percussion hammer.

Impact location also had a measurable and physically interpretable influence on classification accuracy. Percussion at the bolt region (Location 2) consistently provided the highest accuracy, while under data-limited conditions, the excitation at the cantilever tip (Location 1) degraded the most. This pattern is explained by the dilution of preload-related acoustic information through cantilever bending modes during propagation and has direct use for impact point selection in real-life testing scenarios.

Under additive noise, the classical and deep learning architectures diverged. The accuracy of MFCC+SVM dropped to 9.1% at 20 dB SNR, while PSD+GBDT was reduced to 76.3%. But the 1D-CNN maintained 97.2%, and the Mamba-CNN maintained 98.9% with near-constant accuracy across all tested SNR levels from 20 to 60 dB. The noise invariance of the Mamba-CNN is due to its input-selective state update mechanism, which adapts the state transition at each time step based on signal content, effectively suppressing noise contributions without requiring explicit filtering.

This study presents the first application of the Mamba selective state space architecture to acoustic structural health monitoring, and the architecture has provided a qualitatively distinct noise robustness advantage over all other evaluated methods. These findings support the adoption of Mamba-CNN as the preferred architecture for percussion-based bolt preload monitoring in realistic industrial environments where SNR control is limited.

5.2 Future work

The framework can be scaled to larger bolt arrays and more complex structural configurations, which can investigate the effects of different structural boundary conditions on the classification performance. Furthermore, evaluation of the framework under non-white noise profiles that are representations of specific industrial noise environments can be carried out. There is also the potential of integrating continuous impact monitoring with the Mamba-CNN classifier into a real-time health index for bolted joint assemblies. These prospects are promising and indicate the future of percussion-based Structural Health Monitoring.

REFERENCES

1. Kakirde, A., & Dravid, S. (2017). Study of vibration loosening of bolted joints: A review. International Journal for Advance Research in Applied Science and Engineering, 6(Special), 988–997.
2. Junker, G. H. (1969). New criteria for self-loosening of fasteners under vibration. In 1969 International Automotive Engineering Congress and Exposition. SAE Technical Paper.
3. Wang, G. P., Hong, Y., Lee, J. J., Hong, D. P., Kim, Y. M., & Kim, J. Y. (2007). Quantitative estimation of the fastening condition of a bolt using piezoceramic (PZT) sensors. Key Engineering Materials, 353, 2436–2440.
4. Sun, W., & Zhang, Y. (2013). Numerical simulation of bolt looseness monitoring based on impedance method. Applied Mechanics and Materials, 351, 1264–1268.
5. Shao, J., Wang, T., Yin, H., Yang, D., & Li, Y. (2016). Bolt looseness detection based on piezoelectric impedance frequency shift. Applied Sciences, 6(10), 298.
6. Song, C., Xie, L., & Xue, S. (2016). Experimental study of bolt loosening detection based on piezoelectric impedance technology. Journal of Highway and Transportation Research and Development, 33(4), 113–119.
7. Huo, L., Wang, B., Chen, D., & Song, G. (2017). Monitoring of pre-load on rock bolt using piezoceramic-transducer enabled time reversal method. Sensors, 17(11), 2467.
8. Na, W. S. (2021). Bolt loosening detection using impedance-based non-destructive method and probabilistic neural network technique with minimal training data. Engineering Structures, 226, 111228.
9. Eraliev, O., Lee, K. H., & Lee, C. H. (2022). Vibration-based loosening detection of a multi-bolt structure using machine learning algorithms. Sensors, 22(3), 1210.
10. Kong, Q., Zhu, J., Ho, S. C. M., & Song, G. (2018). Tapping and listening: A new approach to bolt looseness monitoring. Smart Materials and Structures, 27(7), 07LT02.
11. Zhang, Y., Zhao, X., Sun, X., Su, W., & Xue, Z. (2019). Bolt loosening detection based on audio classification. Advances in Structural Engineering, 22(13), 2882–2891.
12. Yuan, R., Lv, Y., Kong, Q., & Song, G. (2019). Percussion-based bolt looseness monitoring using intrinsic multiscale entropy analysis and BP neural network. Smart Materials and Structures, 28(12), 125001.

13. Wang, F., & Song, G. (2020). Bolt-looseness detection by a new percussion-based method using multifractal analysis and gradient boosting decision tree. Structural Health Monitoring, 19(6), 2023–2032.
14. Zhou, Y., Wang, S., Zhou, M., Chen, H., Yuan, C., & Kong, Q. (2022). Percussion-based bolt looseness identification using vibration-guided sound reconstruction. Structural Control and Health Monitoring, 29(2), e2876.
15. Chen, J., Chen, Z., Zhu, W., & Song, G. (2024). Underwater bolted flange looseness detection using percussion-induced sound and feature-reduced multi-ROCKET model. Structural Health Monitoring, 23(1), 495–511.
16. Du, C., Liu, J., Gong, H., Huang, J., & Zhang, W. (2024). Percussion-based loosening detection method for multi-bolt structure using convolutional neural network DenseNet-CBAM. Structural Health Monitoring, 23(4), 2183–2199.
17. Huda, F., Kajiwara, I., Hosoya, N., & Kawamura, S. (2013). Bolt loosening analysis and diagnosis by non-contact laser excitation vibration tests. Mechanical Systems and Signal Processing, 40(2), 589–604.
18. Wang, F. (2023). Multi-bolt looseness detection using a new acoustic emission strategy. Structural Health Monitoring, 22(3), 1543–1553.
19. Wang, F., Chen, X., & Song, G. (2020). Percussion-based detection of bolt looseness using speech recognition technology and least square support vector machine. In 2020 IEEE International Conference on Networking, Sensing and Control (ICNSC) (pp. 1–3). IEEE.
20. Liu, P., Wang, X., Wang, Y., Zhu, J., & Ji, X. (2024). Research on percussion-based bolt looseness monitoring under noise interference and insufficient samples. Mechanical Systems and Signal Processing, 208, 111013.
21. Wang, F., & Song, G. (2021). 1D-TICapsNet: An audio signal processing algorithm for bolt early looseness detection. Structural Health Monitoring, 20(5), 2828–2839.
22. Wang, T., Song, G., Liu, S., Li, Y., & Xiao, H. (2013). Review of bolted connection monitoring. International Journal of Distributed Sensor Networks, 9(12), 871213.
23. Chelimilla, N., Chinthapenta, V., Kali, N., & Korla, S. (2023). Review on recent advances in structural health monitoring paradigm for looseness detection in bolted assemblies. Structural Health Monitoring, 22(6), 4264–4304.
24. Gu, A., & Dao, T. (2024). Mamba: Linear-time sequence modeling with selective state spaces. In First Conference on Language Modeling.

25. Yi, H., Li, D., Lu, Z., Jin, Y., Duan, H., Hou, L., Duraihem, F. Z., Awwad, E. M., & Saeed, N. A. (2025). VibrMamba: A lightweight Mamba-based fault diagnosis of rotating machinery using vibration signal. Measurement, 249, 116881.
26. Wang, P., Song, Y., Wang, X., & Xiang, Q. (2025). MD-BiMamba: An aero-engine inter-shaft bearing fault diagnosis method based on Mamba with modal decomposition and bidirectional features fusion strategy. Measurement, 242, 115870.

ANNEX I

Measurement Device Calibration

Measured Frequency: 500.00 Hz
Frequency Error: 0.0000 %
Transmitted Amplitude: 0.5
Received Amplitude: 0.2591
Attenuation: -5.71 dB
SNR: 91.88 dB

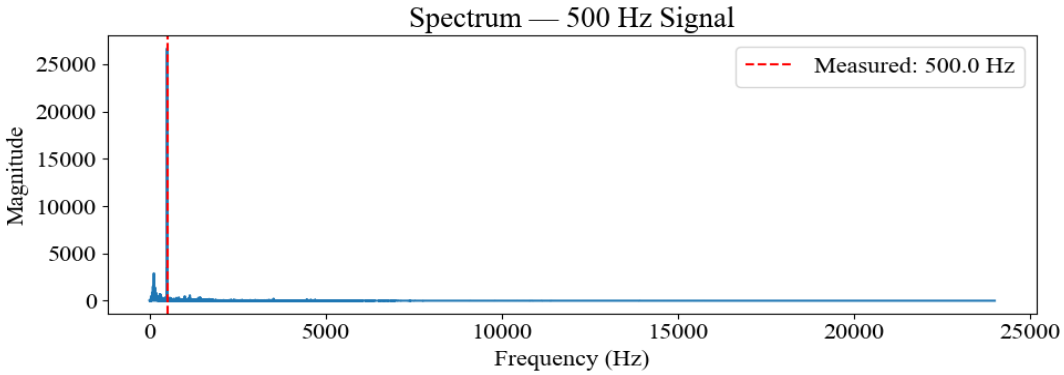

Figure I: Audio device calibration for 500Hz known audio signal

Measured Frequency: 1000.00 Hz
Frequency Error: 0.0000 %
Transmitted Amplitude: 0.5
Received Amplitude: 0.4295
Attenuation: -1.32 dB
SNR: 98.62 dB

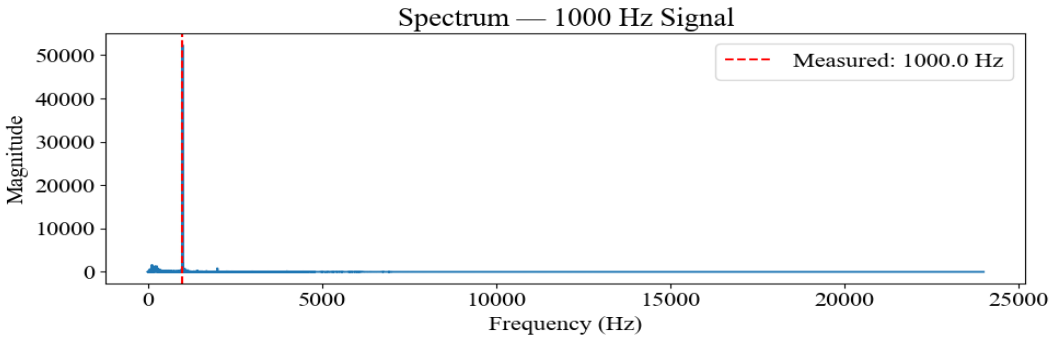

Figure II: Audio device calibration for 1000Hz known audio signal

Measured Frequency: 2000.00 Hz
Frequency Error: 0.0000 %
Transmitted Amplitude: 0.5
Received Amplitude: 0.3123
Attenuation: -4.09 dB
SNR: 112.88 dB

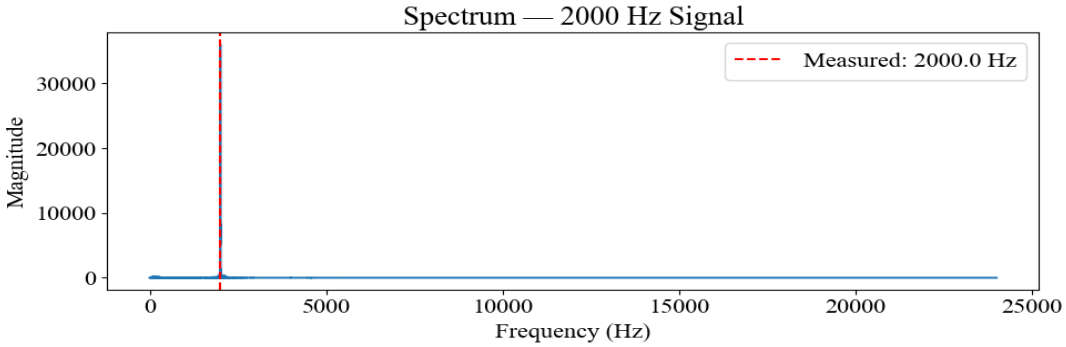

Figure III: Audio device calibration for 2000Hz known audio signal

Measured Frequency: 5000.00 Hz
Frequency Error: 0.0000 %
Transmitted Amplitude: 0.5
Received Amplitude: 0.5369
Attenuation: 0.62 dB
SNR: 104.16 dB

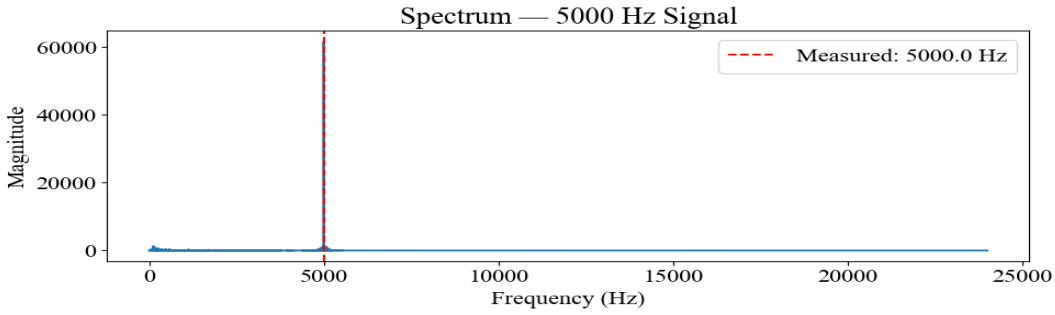

Figure IV: Audio device calibration for 2000Hz known audio signal

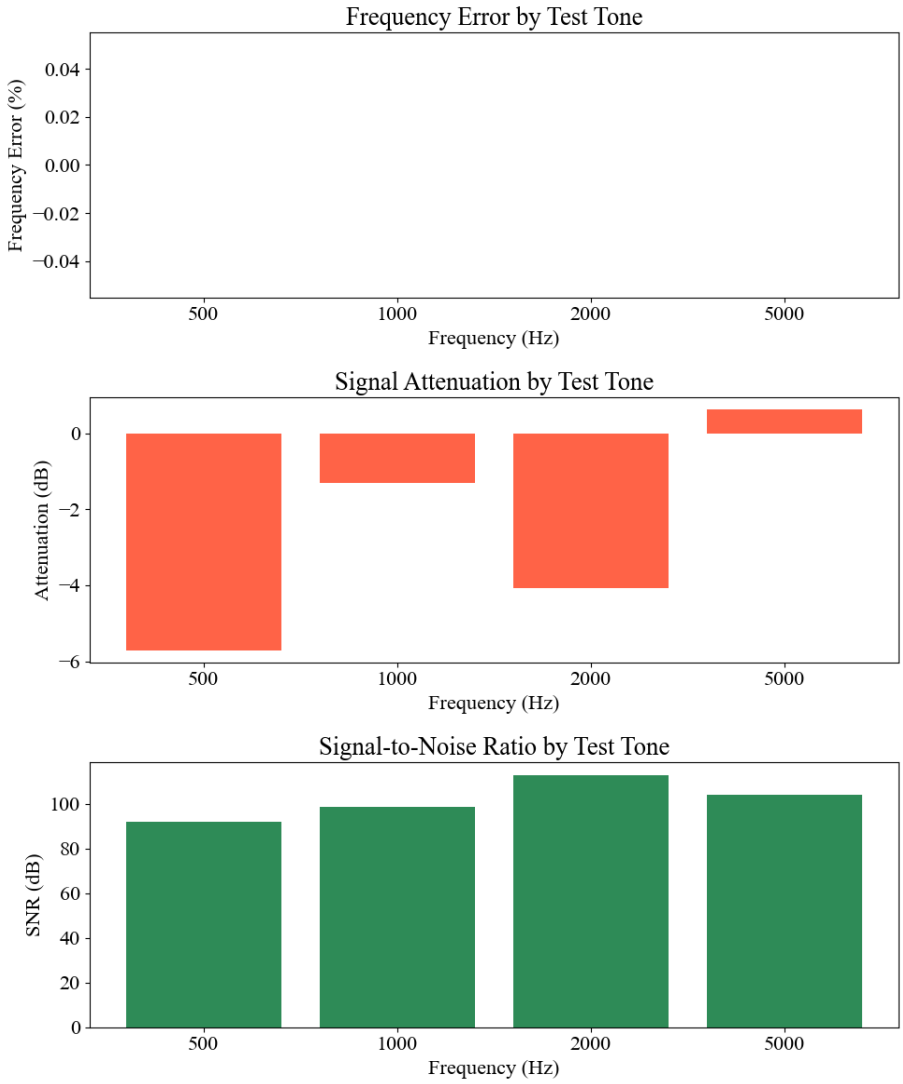

Figure V: Frequency error and integrated noise SNR in audio device calibration

Torque Wrench Calibration

To maintain the preload levels, a torque wrench with the operating range of 12Nm - 60Nm was used, which was suitable for this application. Before using, the Torque Wrench was calibrated using a known load suspended at a known distance from the pivot center. For a definite weight, the distance was increased, increasing the torque until a click was heard from the device, then the distance was measured to determine the loading torque.

The calculated/measured torque and the torque value indicated by the device were calibrated at three different torque levels.

After calibrating the wrench for one value, it accurately predicted the torque levels for other true values, thus validating our measurement device.

Figure VI (a): Torque wrench

Figure VI (b): Torque wrench calibration setup with known weights

ANNEX 2

Database Repository: https://github.com/iammanu57/Segmented_Audio_Cantilver_Data.git

Codes for Classifier Models

```
SAMPLE_RATE = 48000
def load_data_from_folders(data_dir, sr=SAMPLE_RATE):
    data_dir      = Path(data_dir)
    audio_segments, labels, class_names = [], [], []
    class_folders = sorted([f for f in data_dir.iterdir() if
f.is_dir()])
    for class_idx, folder in enumerate(class_folders):
        print(f"  Loading '{folder.name}' ...")
        class_names.append(folder.name)
        for audio_file in sorted(
                list(folder.glob("*.wav")) +
list(folder.glob("*.mp3"))):
            audio, _ = librosa.load(audio_file, sr=sr)
            audio_segments.append(audio)
            labels.append(class_idx)
    return audio_segments, labels, class_names

def prepare_raw_splits(audio_segments, labels, max_len=None):
    """
    Pad/truncate waveforms → numpy array (N, T, 1).
    Applies the GLOBAL test-split with RANDOM_STATE and stratify=y.
    Returns X_train_raw, X_test_raw, y_train, y_test.
    """
    if max_len is None:
        max_len = max(len(a) for a in audio_segments)
    X = []
    for audio in audio_segments:
        if len(audio) < max_len:
            audio = np.pad(audio, (0, max_len - len(audio)),
mode="constant")
        X.append(audio[:max_len])
    X = np.array(X, dtype=np.float32)[..., np.newaxis]   # (N, T, 1)
    y = np.array(labels)
    return train_test_split(X, y,
                            test_size=TEST_SIZE,
                            random_state=RANDOM_STATE,
                            stratify=y)

def extract_psd_features(audio_segment, sr=SAMPLE_RATE,
                         n_fft=2048, n_bands=40):
    S     = np.abs(librosa.stft(audio_segment, n_fft=n_fft)) ** 2
    freqs = librosa.fft_frequencies(sr=sr, n_fft=n_fft)
    band_edges = np.logspace(
        np.log10(max(freqs[1], 1.0)), np.log10(freqs[-1]), n_bands + 1)
    feats = []
```

```
    for lo, hi in zip(band_edges[:-1], band_edges[1:]):
        mask = (freqs >= lo) & (freqs < hi)
        p    = float(np.mean(S[mask])) if mask.any() else 1e-12
        feats.append(10.0 * np.log10(max(p, 1e-12)))   # dB
    return np.array(feats, dtype=np.float32)

def extract_mfcc_features(audio_segment, sr=SAMPLE_RATE,
                          n_mfcc=N_MFCC,
hop_length=HOP_LENGTH,n_fft=N_FFT):
    mfcc = librosa.feature.mfcc(y=audio_segment, sr=sr,
                                n_mfcc=n_mfcc,
                                hop_length=hop_length,
                                n_fft=n_fft)   # (n_mfcc, T)
    feat = np.concatenate([
        mfcc.mean(axis=1),
        mfcc.std(axis=1),
        mfcc.min(axis=1),
        mfcc.max(axis=1),
    ], axis=0)
    return feat.astype(np.float32)

def add_white_noise(audio_segment, snr_db):
    sig     = audio_segment.astype(np.float64)
    p_sig   = np.mean(sig ** 2)
    p_sig   = max(p_sig, 1e-12)          # guard against all-zero
segments
    p_noise = p_sig / (10.0 ** (snr_db / 10.0))
    # Per-segment seed: avoids all samples getting identical noise
pattern
    seg_seed = int(RANDOM_STATE) ^ (hash(sig.tobytes()) & 0x7FFFFFFF)
    rng      = np.random.default_rng(seg_seed)
    noise    = rng.normal(0.0, np.sqrt(p_noise), size=len(sig))
    return (sig + noise).astype(np.float32)

_COMPILE_KWARGS = dict(
    optimizer=keras.optimizers.Adam(learning_rate=1e-3),
    loss="sparse_categorical_crossentropy",
    metrics=["accuracy"],
)

def build_1d_cnn(input_shape, num_classes):
    model = keras.Sequential([
        layers.Conv1D(32,  80, strides=4, activation="relu",
input_shape=input_shape),
        layers.MaxPooling1D(4),
        layers.Conv1D(64,   3, activation="relu"),
        layers.MaxPooling1D(4),
        layers.Conv1D(128,  3, activation="relu"),
        layers.MaxPooling1D(4),
        layers.Flatten(),
        layers.Dense(128, activation="relu"),
        layers.Dropout(0.5),
        layers.Dense(num_classes, activation="softmax"),
```

```
    ], name="1D_CNN")
    model.compile(**_COMPILE_KWARGS)
    return model

class MambaBlock(layers.Layer):
    def __init__(self, d_model, d_state=16, d_conv=4, expand=2, **kw):
        super().__init__(**kw)
        d_inner         = expand * d_model
        self.d_inner    = d_inner
        self.d_state    = d_state
        self.d_conv     = d_conv
        self.in_proj    = layers.Dense(d_inner * 2, use_bias=False)
        self.conv1d     = layers.DepthwiseConv1D(d_conv,
padding="valid",
                                                 activation="linear")
        self.x_proj     = layers.Dense(d_state * 2 + 1, use_bias=False)
        self.dt_proj    = layers.Dense(d_inner, use_bias=True)
        self.A_log      = None   # built in build()
        self.D          = None
        self.out_proj   = layers.Dense(d_model, use_bias=False)

    def build(self, input_shape):
        self.A_log = self.add_weight(
            name="A_log",
            shape=(self.d_inner, self.d_state),
            initializer=tf.initializers.Constant(
                np.log(np.tile(np.arange(1, self.d_state + 1),
                               (self.d_inner, 1)).astype("float32"))),
            trainable=True,
        )
        self.D = self.add_weight(
            name="D",
            shape=(self.d_inner,),
            initializer="ones",
            trainable=True,
        )

    def call(self, x, training=False):
        xz  = self.in_proj(x)
        x_b, z = tf.split(xz, 2, axis=-1)
        x_b = tf.pad(x_b, [[0, 0], [self.d_conv - 1, 0], [0, 0]])
        x_b = tf.nn.silu(self.conv1d(x_b))
        A  = -tf.exp(self.A_log)
        D  = self.D
        xBC = self.x_proj(x_b)
        dt_raw, B_raw, C_raw = tf.split(
            xBC, [1, self.d_state, self.d_state], axis=-1)
        dt = tf.nn.softplus(self.dt_proj(dt_raw))
        dA = tf.exp(tf.einsum("bld,dn->bldn", dt, A))
        dB = tf.einsum("bld,bln->bldn", dt, B_raw)
        u_exp = tf.expand_dims(x_b, -1)
        Bu = dB * u_exp
        log_dA = tf.math.log(tf.maximum(dA, 1e-8))
```

```
        log_cum_decay = tf.cumsum(log_dA, axis=1)
        cum_decay = tf.exp(log_cum_decay)
        Bu_norm = Bu / tf.maximum(cum_decay, 1e-8)
        h_approx = cum_decay * tf.cumsum(Bu_norm, axis=1)
        y_out = tf.einsum("bldn,bln->bld", h_approx, C_raw)
        y_out = y_out + x_b * D

        y_out = y_out * tf.nn.silu(z)
        return self.out_proj(y_out)

def build_mamba(input_shape, num_classes, d_model=64, n_layers=4):
    inp = keras.Input(shape=input_shape)
    x = layers.Conv1D(d_model, 80, strides=4, padding="causal",
                      activation="relu")(inp)
    x = layers.MaxPooling1D(4)(x)
    for _ in range(n_layers):
        residual = x
        x = layers.LayerNormalization()(x)
        x = MambaBlock(d_model)(x)
        x = x + residual  # residual connection
    x = layers.LayerNormalization()(x)
    x = layers.GlobalAveragePooling1D()(x)
    x = layers.Dense(128, activation="relu")(x)
    x = layers.Dropout(0.5)(x)
    out = layers.Dense(num_classes, activation="softmax")(x)
    model = keras.Model(inp, out, name="Mamba")
    model.compile(**_COMPILE_KWARGS)
    return model

def build_cnn_mamba(input_shape, num_classes, d_model=64, n_layers=2):
    inp = keras.Input(shape=input_shape)
    x = layers.Conv1D(32, 80, strides=4, padding="same")(inp)
    x = layers.BatchNormalization()(x)
    x = layers.Activation("relu")(x)
    x = layers.MaxPooling1D(4)(x)
    x = layers.Conv1D(64, 3, padding="same")(x)
    x = layers.BatchNormalization()(x)
    x = layers.Activation("relu")(x)
    x = layers.MaxPooling1D(4)(x)
    x = layers.Conv1D(d_model, 3, padding="same")(x)
    x = layers.BatchNormalization()(x)
    x = layers.Activation("relu")(x)
    x = layers.MaxPooling1D(4)(x)
    for _ in range(n_layers):
        residual = x
        x = layers.LayerNormalization()(x)
        x = MambaBlock(d_model)(x)
        x = x + residual
    x = layers.LayerNormalization()(x)
    x = layers.GlobalAveragePooling1D()(x)
    x = layers.Dense(128, activation="relu")(x)
    x = layers.Dropout(0.4)(x)
    out = layers.Dense(num_classes, activation="softmax")(x)
```

```
    model = keras.Model(inp, out, name="CNN_Mamba")
    model.compile(**_COMPILE_KWARGS)
    return model

def build_psd_gbdt():
    clf = Pipeline([
        ("scaler", StandardScaler()),
        ("gbdt",   GradientBoostingClassifier(
            n_estimators=300,
            max_depth=5,
            learning_rate=0.05,
            subsample=0.8,
            min_samples_leaf=5,
            random_state=RANDOM_STATE,
            verbose=1,
        )),
    ])
    return clf

def build_mfcc_svm():
    clf = Pipeline([
        ("scaler", StandardScaler()),
        ("svm",    SVC(
            kernel="rbf",
            C=1.0,
            gamma="scale",
            probability=True,
            random_state=RANDOM_STATE,
            verbose=True,
        )),
    ])
    return clf
```

Comparative Analysis of MFCC and PSD Features for Multi-Bolt Preload Classification under Noise Conditions

Manu Aryal [a], *Laxman Motra* [a], *Mahesh Chandra Luintel* [a]

Abstract:

This study presents a comparative analysis of Mel-Frequency Cepstral Coefficients (MFCC) and Power Spectral Density (PSD) features for multi-bolt preload levels classification using impact-induced acoustic signals. The experimental data were collected under controlled conditions, and two classical machine learning models, support vector machines (SVM) for MFCCs and gradient boosting decision trees (GBDTs) for PSDs, were evaluated. After the test-train split, the test signal is integrated with white noise (20-60 dB). The results indicate that in controlled conditions, MFCC+SVM has greater prediction accuracy, but when noise is integrated with the acoustic signals, the classification accuracy of MFCC+SVM plummets, whereas PSD+GBDT performs comparitively robust to noise, achieving up to $\sim 76\%$ accuracy at high noise levels. The findings highlight that the time-frequency domain features represented by MFCC perform significantly well in controlled conditions, while in the presence of noise, the time domain data becomes corrupted, and frequency domain data from PSD performs well with the GBDT classifier. This shows the importance of feature selection in acoustic-based structural health monitoring systems.

Keywords:

Percussion, Machine Learning, Mel-Frequency Cepstrum Coefficients, Power Spectral Density, Support Vector Machines, Gradient Boosting Decision Tree

[a] *Department of Mechanical and Aerospace Engineering, IOE Pulchowk Campus, Tribhuvan University, Nepal*

[a] 080msree011.manu@pcampus.edu.np

Pulchowk Campus
Institute of Engineering
Tribhuvan University

MANU ARYAL <080msree011.manu@pcampus.edu.np>

[JIEE] Editor Decision

1 message

postmaster@nepjol.info <postmaster@nepjol.info> Thu, Apr 30, 2026 at 7:25 PM
Reply-To: Raj Kumar Chaulagain <rajkrc12@tcioe.edu.np>
To: Manu Aryal <080msree011.manu@pcampus.edu.np>, Mahesh Chandra Luintel <mcluintel@ioe.edu.np>, Laxman Motra <laxman.motra@pcampus.edu.np>

Manu Aryal, Mahesh Chandra Luintel, Laxman Motra:

We have reached a decision regarding your submission to Journal of Innovations in Engineering Education, "A Comparative Study on Acoustic Fault Detection in Bolted Structures".

Our decision is to: Accept the Submission

Journal of Innovations in Engineering Education

B-Manuscript_Updated.docx
1149K

7% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

53 Not Cited or Quoted 7%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
5% Publications
0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **53** Not Cited or Quoted 7%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
5% Publications
0% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	
eprints.lib.hokudai.ac.jp		<1%
2	Publication	
Leininger, Dustin Allen. "Algebraic Topology Gets on My Nerves: TDA with Some D...		<1%
3	Internet	
dissertations.mak.ac.ug		<1%
4	Internet	
semiwiki.com		<1%
5	Internet	
www.mdpi.com		<1%
6	Publication	
Chenfei Du, Jianhua Liu, Hao Gong, Jiayu Huang, Wentao Zhang. "Percussion-base...		<1%
7	Publication	
Haiming Yi, Danyu Li, Zhenyong Lu, Yuhong Jin, Hao Duan, Lei Hou, Faisal Z. Durai...		<1%
8	Publication	
Manoj Kumar, Tanweer Ali, Jaume Anguera, Suman Lata Tripathi. "Emerging Tech...		<1%
9	Internet	
etd.aau.edu.et		<1%
10	Internet	
www.geeksforgeeks.org		<1%

11 Internet
kclpure.kcl.ac.uk <1%

12 Publication
Bal S. Virdee, Tanweer Ali, Jaume Anguera, Suman Lata Tripathy. "Connecting Int... <1%

13 Publication
Huitao Zhang, Diwei Zhu, Yunxiang Gan, Shuguang Xiong. "End-to-End Learning-... <1%

14 Publication
Pengtao Liu, Xiaopeng Wang, Yongquan Wang, Jian Zhu, Xinyu Ji. "Research on pe... <1%

15 Internet
digiresearch.vut.ac.za <1%

16 Internet
elibrary.tucl.edu.np <1%

17 Publication
Renqing Gou, Osama Sohaib. "RiskMamba", Journal of Organizational and End Us... <1%

18 Internet
www.slideshare.net <1%

19 Internet
nhuang37.github.io <1%

20 Internet
shotstack.io <1%

21 Publication
Nannan Xu, Guang Yang, Linlin Ming, Jiefei Dai, Kun Zhu. "PSOA-LSTM: a hybrid at... <1%

22 Publication
Jiehao Li, Hongkai Chen, Shan Zeng, Jinrong Cui, Xiwen Luo, C. L Philip Chen, Chen... <1%

23 Internet
journals.zeuspress.org <1%

24 Internet
www.ijraset.com <1%

25 Publication
Saurav Mallik, Sandeep Kumar Mathivanan, Basu Dev Shivahare, S.K.B. Sangeeth... <1%

26 Internet
ibis-project.org <1%

27 Publication
Jian Chen, Zheng Chen, Weihang Zhu, Gangbing Song. "Underwater bolted flange... <1%

28 Publication
Parisa Eslambolchilar, Roderick Murray-Smith. "Control centric approach in desig... <1%

29 Publication
Yixuan Chen, Hongyi Chen, Jingyi Wei, Jianchao Wu. "Automatic detection of loos... <1%

30 Internet
anritsu.com.hk <1%

31 Internet
deepai.org <1%

32 Internet
www.frontiersin.org <1%

33 Internet
raw.githubusercontent.com <1%

34 Internet
www.diva-portal.org <1%

35 Internet
dspace.nm-aist.ac.tz <1%

36 Internet
sportdocbox.com <1%

37 Internet
www.scotland.gov.uk <1%