Journal of Engineering and Applied Science Vol. 10, No. 01, pp. 33–42, January 2026

Journal of Engineering and Applied Science

Contents are available at www.jeas.ruet.ac.bd

Vibration-Based Gear Fault Diagnosis and Real-Time Condition Monitoring Using Machine Learning Techniques

Shahajada Mahmudul Hasan[1], Md. Najmul Islam Shawon[1,] Md. Imran Hasan[1], Abdullah Adnan Abir[1,*]

[1]*Department of Mechanical Engineering, Rajshahi University of Engineering & Technology, Bangladesh*

ARTICLE INFORMATION

Received date: 18 Jan 2026
Revised date: 15 Jun 2026
Accepted date: 16 Jun 2026

Keywords

Predictive Maintenance
Random Forest
Support Vector Machine
Artificial Neural Network
Graphical User Interface

ABSTRACT

Modern industrial systems operating under varying speed and load conditions widely employ gear assemblies for smooth power transmission. Unexpected gear failures can lead to costly downtime and safety hazards. With the advancement of Industry 4.0, sensor-based predictive maintenance has become essential for early fault detection. Traditional condition monitoring methods rely heavily on human expertise and specialized instruments, which may fail to identify early-stage faults. This study presents a low-cost intelligent predictive maintenance framework for vibration-based gear fault diagnosis and real-time condition monitoring using machine learning techniques. Vibration signals were acquired from a laboratory-scale gearbox test rig using an ADXL345 accelerometer under three gear conditions: healthy, root crack, and broken tooth, over a speed range of 600–2400 RPM. After preprocessing, approximately 930 vibration samples were used to train and evaluate Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Network (ANN) models. Among the implemented models, the ANN achieved the highest classification accuracy of 99.46% on the held-out test set, with precision, recall, and F1-scores of approximately 0.99 for all three classes and convergence within roughly 53 training epochs. A 5-fold stratified cross-validation of the ANN yielded a mean accuracy of 99.89 ± 0.22%, with the worst-fold accuracy matching the single-split result, confirming that the reported performance is stable and reproducible. In addition, a Python-based graphical user interface (GUI) using the trained ANN model, was developed for real-time condition monitoring, enabling continuous sensor-data acquisition, live fault prediction, vibration visualization, and automatic alarm activation during abnormal operating conditions. The proposed framework demonstrates high diagnostic reliability, low implementation cost, and practical applicability for Industry 4.0–based predictive maintenance systems.

* Corresponding authors: Department of Mechanical Engineering, Rajshahi University of Engineering & Technology, Bangladesh
E-mail addresses: abdullahadnanabir@gmail.com (Abdullah Adnan Abir)

1. Introduction

Gears are an essential, regulating part of power transmission systems in rotating machinery, and demand for operational precision has grown in recent years. Small defects such as tooth cracks, backlash, and misalignment can cause vibrations that eventually lead to failure of supporting parts like shafts and bearings, so monitoring gear health is crucial to prevent machine breakdown [1]. Machine maintenance keeps machine elements running with minimal downtime [2]. Three maintenance types are used for condition monitoring: reactive, performed after equipment has failed and increasing costly downtime; preventive, performed at predetermined intervals to minimise downtime; and predictive, which uses monitoring tools to predict machine condition and determine when maintenance is necessary [3]. Predictive maintenance offers the advantage of accurate scheduling of maintenance activity to avoid critical over other types maintenance. In predictive maintenance, sensor data from the machine and analyses algorithm are utilized to schedule the maintenance activities [4], [5].

Vibration data is widely used for predictive maintenance, vibration being the reactive behaviour of a machine to external or internal forces [6]. Continuous monitoring of vibration data identifies deterioration caused by early faults, which indicates the gear malfunction [7]. Using the vibration data three approaches are used for performing predictive maintenance. These are — statistical approaches, artificial intelligence approaches, and model-based approaches [8]. By comparing these approaches for predicting the condition of machine, it has been found that artificial approaches are more accurate than other approaches [9].

AI techniques such as SVM and ANN have been used to diagnose faults in key components in recent years, with condition monitoring treated as a classification problem based on training patterns from malfunctioning gears. SVM, founded on statistical learning theory, minimises an upper bound on the predicted risk through structural risk minimisation, whereas ANN applies empirical risk minimisation to the training set. ANN can handle highly non-linear functions, making it ideal for identifying complex patterns that traditional models fail to recognise, while SVM offers shorter training time [10].

Real-time condition monitoring using Artificial Intelligence enables continuous tracking of vibration data from mechanical systems to predict machine health, allowing early fault detection that reduces costly downtime and prevents critical failures. Several studies have demonstrated the effectiveness of machine learning and sensor-based approaches for predictive maintenance of rotating machinery.

Carvalho et al. provided a systematic review of ML methods for PdM and identified Support Vector Machines (SVM), Random Forests (RF), and Artificial Neural Networks (ANN) as the most widely adopted classifiers[10] Praveenkumar et al. successfully applied ML techniques to automobile gearbox fault diagnosis using statistical vibration features [1], and Kumar et al. proposed a big-data-driven sustainable manufacturing framework for condition-based maintenance prediction[10]. Mohammed and Abdulateef integrated ML with industrial IoT and MQTT communication for proactive motor maintenance [11], whereas Antonino-Daviu demonstrated electrical-signature monitoring under transient conditions for motor PdM, but reported a strong dependence on signal-processing expertise [12]. Shahapurkar et al. employed FFT-based vibration analysis for effective detection of gearbox defects [13]. The use of statistical vibration features combined with machine learning classifiers has also shown promising results for automatic fault classification in gear systems [1], [14]. These studies confirm the general suitability of ANN, SVM, and RF for fault classification. Hassan et al. presented an in-depth comparison of vibration sensors and established that MEMS-class accelerometers provide acceptable sensitivity for early-stage fault detection at a small fraction of the cost of piezoelectric industrial transducers [15]. Mones et al. demonstrated that an on-rotor microelectromechanical-system (MEMS) accelerometer can effectively diagnose planetary-gearbox faults[16] , confirming that consumer-grade sensors are capable of supporting practical PdM applications. Nevertheless, most published implementations stop at offline classification accuracy: the trained models are seldom embedded in a working real-time interface, and very few studies report a complete, deployable sensor-to-decision pipeline that is reproducible by industries operating with constrained instrumentation budgets.

Based on the foregoing review, three principal research gaps can be identified. *First*, although ANN, SVM, and RF have each been applied individually to gear-fault diagnosis, a direct head-to-head comparison of these three classifiers on an identical, low-cost microelectromechanical-system (MEMS)-accelerometer dataset covering multiple gear conditions and a wide speed range is largely absent in the literature. *Second*, most existing studies focus on offline analysis and do not provide a real-time, sensor-to-decision implementation that an operator can use without specialised signal-processing expertise. *Third*, very few works integrate the trained classifier into a graphical user interface with condition-specific alarm logic — an essential feature for practical shop-floor adoption.

To address these gaps, this study develops and experimentally validates a complete, low-cost, AI-driven predictive-maintenance and real-time condition-monitoring framework for gear systems. Its main contributions are: (i) a laboratory-scale gear rig instrumented with an ADXL345 MEMS accelerometer and Arduino Uno acquires tri-axial vibration signals under three gear conditions (healthy, root-cracked, broken-tooth) across 31 speeds (600–2400 RPM), yielding a curated 930-sample dataset; (ii) a feature-correlation analysis identifies the most informative vibration axes, giving a transparent feature-selection basis; (iii) three classifiers (ANN, SVM, RF) are trained and benchmarked on the same data with an identical 80/20 split and compared on accuracy, precision, recall, F1-score, and learning-curve behaviour; (iv) the best classifier is embedded in a Python–Tkinter GUI that streams live data, classifies the gear condition, visualises signal trends, and triggers audible alarms for root-crack and broken-tooth events; and (v) the framework is evaluated as a deployable, cost-effective alternative to commercial vibration analysers for small- and medium-scale industrial users.

2. Methodology

The proposed framework integrates vibration-based sensing with artificial-intelligence (AI) techniques for predictive maintenance and real-time condition monitoring of gear assemblies. The end-to-end workflow (Figure 1) comprises five stages: (i) acquisition of tri-axial vibration signals using a low-cost MEMS accelerometer; (ii) signal transmission to a microcontroller-based data-acquisition unit; (iii) data preprocessing, including outlier removal, normalisation, and feature selection; (iv) training and benchmarking of three supervised classifiers — ANN, SVM, and RF — on the prepared dataset; and (v) deployment of the best-performing classifier inside a Python-based GUI that receives live sensor data, classifies the gear condition, visualises vibration trends, and activates audible alarms when an abnormal state is detected. Each stage is described in the following sub-sections.

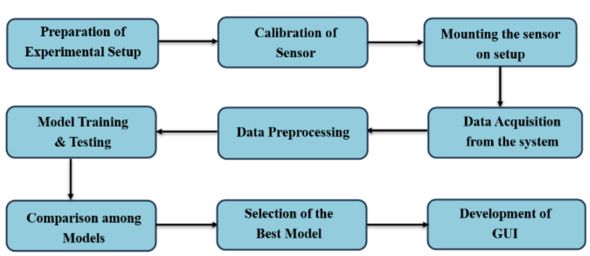

Figure 1. Workflow of Methodology.

2.1. Experimental setup

The complete experimental rig is shown schematically in Figure 2 and in its physical form in Figure 3. The major components are: a three-phase induction motor providing the primary drive; a variable-frequency drive (VFD) that varies the motor speed between 600 and 2400 RPM; a pair of meshing spur gears (a driver pinion on the motor shaft and a driven gear on the output shaft); a counter-weight that maintains rotational balance; a rigid supporting bar carrying the accelerometer close to the driven-gear bearing housing; an ADXL345 tri-axial MEMS accelerometer sensing vibration along the X, Y, and Z directions; and an Arduino Uno microcontroller that digitises the output via its I²C interface and streams it to a host computer over USB. CoolTerm terminal software logs the streamed data, which are then exported to Microsoft Excel for offline analysis and model training.

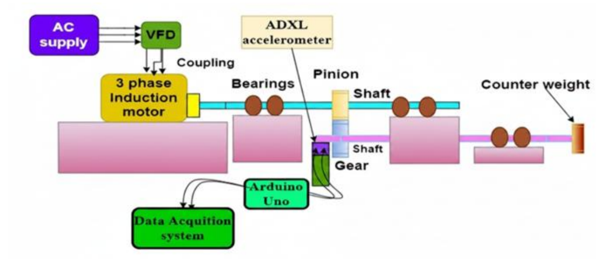

Figure 2. Schematic of Experimental Setup

Figure 3 shows the actual laboratory-scale setup corresponding to the schematic in Fig. 2.

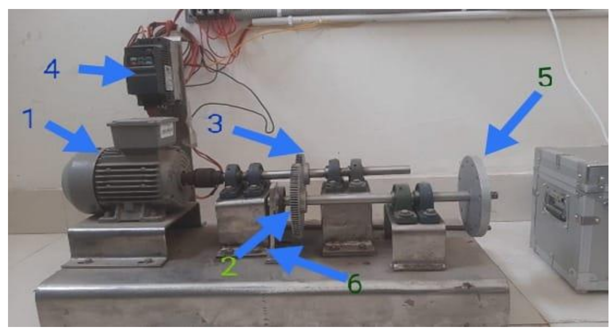

Figure 3. Experimental Setup.

Experimental setup consists of

1. Three phase induction motor.
2. Spur gear.
3. Pinion.
4. Variable frequency drive (VFD)
5. A counter weight.
6. A supporting bar.

2.2. Gear conditions for data collection

Three gear-health conditions were considered: (i) a healthy baseline gear with no observable defect, (ii) a root-crack condition with an artificial crack at the tooth root simulating early fatigue damage, and (iii) a broken-tooth condition with one tooth removed simulating a severe, late-stage failure. The two induced fault states are shown in Figure 4(a), 4(b).

Figure 4. (a) Broken Teeth. (b) Root crack

2.3. Sensor integration and data collection

The ADXL345 was taped to one side of the supporting bar close to the driven gear, with an Arduino Uno reading its output. The sensor's SDA, SCL, GND, and Vcc pins were connected to the A4, A5, GND, and 5 V ports of the Arduino, and the output was viewed in the Arduino IDE serial monitor. CoolTerm software stored the vibration data, which were then transferred to an Excel sheet for AI model training and testing (Figure 5). The sensor was calibrated using a CSI 2140 vibration analyser. Tri-axial (x, y, z) vibration data were collected at 31 motor speeds (600–2400 RPM) for the three gear conditions (healthy, root crack, and broken teeth). Of over 1400 samples initially collected, about 930 were retained after data cleaning for training and testing.

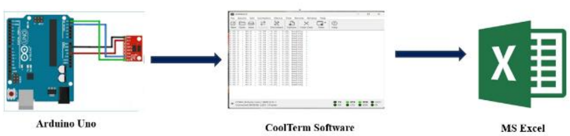

Figure 5. Data Acquisition.

2.4. Data preprocessing, Model training and testing

The acquired dataset was passed through a four-stage preprocessing pipeline implemented in Python using pandas, NumPy, and scikit-learn. Rows with missing values or exact duplicates were discarded; a physics-based filter removed samples exceeding the ADXL345 full-scale range of ±16 g (≈ ±157 m s^{-2}) as saturation or transmission errors; and a within-group Z-score test (|z| > 3, applied independently within each Condition × Speed sub-population to account for the speed-dependent rise in amplitude) eliminated statistical outliers. After cleaning, 930 of the original 1400+ samples were retained, distributed evenly across the three classes (310 each) and across 31 motor-speed levels between 10 and 40 Hz (600–2400 RPM).

Each sample is represented by a four-dimensional feature vector $\mathbf{x} = [\text{Speed}, a_x, a_y, a_z]$. To prevent the higher-magnitude features from dominating gradient-based ANN training and SVM kernel computations, all features were rescaled to the unit interval [0, 1] using Min–Max normalisation (Eq. 1):

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \qquad (1)$$

The scaler parameters $x_{\min}$ and $x_{\max}$ were computed exclusively on the training partition and then applied to the test partition to prevent information leakage. The cleaned dataset was finally partitioned into training (80 %, 744 samples) and test (20 %, 186 samples) sets using stratified random sampling with a fixed seed of 42, ensuring class-balanced partitions and full reproducibility.

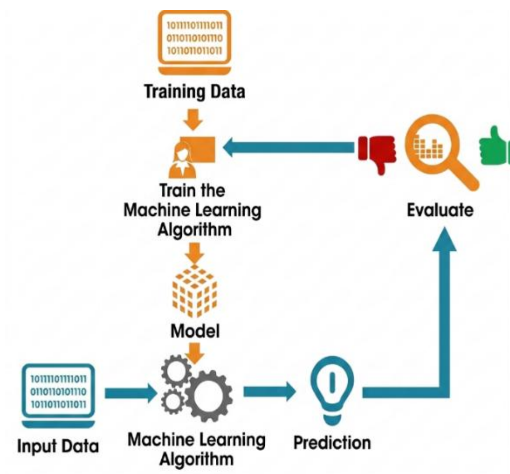

Figure 6. Flow chart of model training & testing

2.4. Making of GUI using Tkinter for Real-Time Condition Monitoring

The ADXL345 is integrated with the Arduino to record tri-axial vibration data, and a Python-based GUI built with Tkinter performs real-time condition monitoring. Real-time visualisation uses Matplotlib, the trained ANN model is imported via scikit-learn, and serial communication is handled through pyserial. Features extracted from the input data are fed to the model loaded inside the GUI, which continuously classifies new input to predict the machine condition.

3. Results and Discussions

3.1. Data visualizing using MATLAB

Vibration signals along the X, Y, and Z axes were recorded at 31 motor speeds (600–2400 RPM) under healthy, root crack, and broken tooth conditions to analyse the variation of vibration intensity with speed and gear condition. The horizontal axis represents motor speed and the vertical axis the vibration amplitude (m/s²). Across all three axes, the amplitude increases with speed.

Under healthy conditions the signals remain smooth and regular, indicating stable meshing. In the root-crack condition, waveform irregularities appear, particularly at higher speeds, from uneven load transmission caused by micro-cracks. In the broken-tooth condition, sharp spikes and sudden oscillations arise from impact forces during tooth engagement and disengagement.

These results (Figure 7) indicate that damaged gears exhibit irregular torque transmission and varying contact stiffness, producing higher dynamic loads and harmonic responses. The X and Z axes show greater variation than the Y-axis, demonstrating that horizontal and tangential vibrations are more sensitive to gear faults. Triaxial vibration behaviour therefore serves as a reliable indicator for fault classification and early diagnosis.

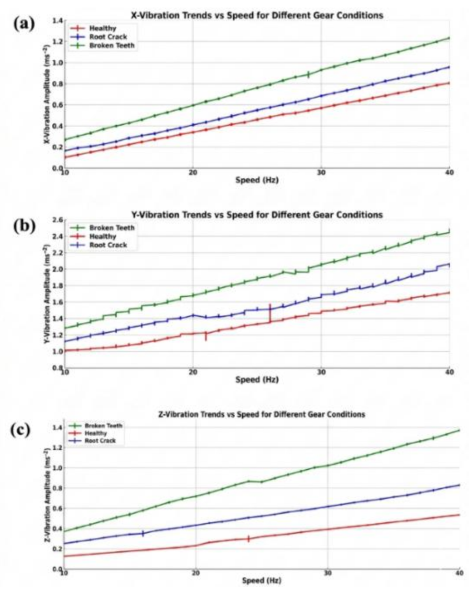

Figure 7. Vibration along (a) X axis, (b) Y axis, (c) Z axis for three conditions vs Speed.

3.2. Correlation Heatmap of numerical features

Figure 8 presents the feature correlation heatmap relating motor speed and vibration along the X, Y, and Z axes. Motor speed correlates strongly with vibration along the X-axis (0.88) and Z-axis (0.63), indicating that amplitude in these directions rises significantly with speed, and a very high correlation (0.92) between the X and Z axes shows they respond similarly to gear-meshing forces. In contrast, the Y-axis shows near-zero correlation with speed and the other components, implying limited sensitivity to gear-related excitation. The X and Z directions are therefore more informative for gear-condition assessment, supporting their selection as dominant predictors for machine learning–based fault classification.

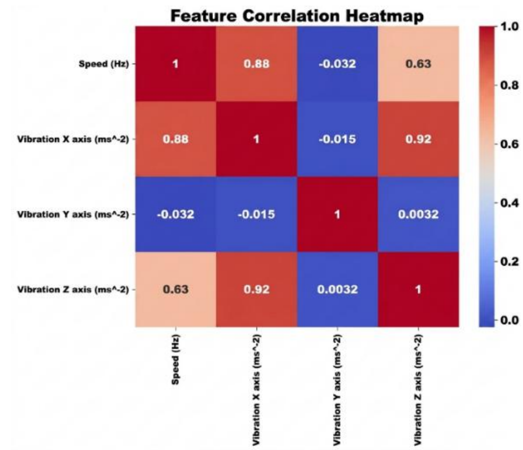

Figure 8. Feature Correlation Heatmap

3.3. Model Performance in RF

Figure 9 shows excellent classification performance for all three classes in the confusion matrix, with the model detecting 60 cases of Root Crack, 61 of Broken, and 60 of Healthy. Precision, recall, and F1 score were 0.984 for the Broken class; 0.984, 0.968, and 0.976 for the Healthy class; and 0.968 for all three metrics for Root Crack. With only 5 misclassifications out of 185 predictions and all measures exceeding 0.96, the model is balanced and accurate.

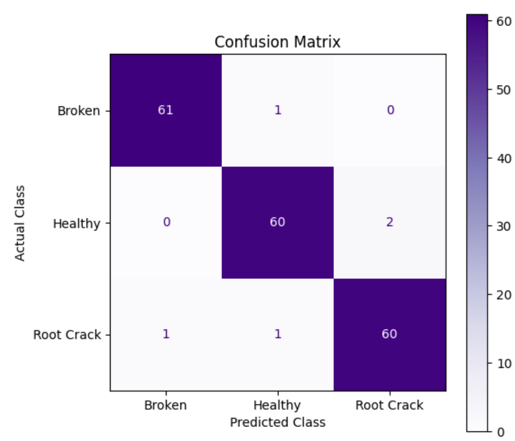

Figure 9. Confusion Matrix for RF

The overall parameters that found from RF

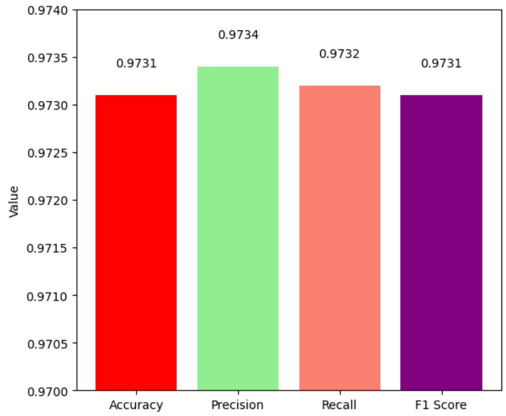

Figure 10. Performance Metrics of RF

3.3.1. Learning curve for RF

Figure 11 represents the learning curve for RF. Following insights are obtained from the learning curve:

- Low initial validation accuracy with limited data indicates underfitting.
- Validation accuracy rises with more training data, showing improved generalization.
- Training accuracy held constant at 100% while unseen-data performance lags, indicating overfitting.
- Validation accuracy plateaus between 275 and 500 samples, indicating diminishing returns from more data.

Figure 11. Learning curve for RF

3.4. Model Performance in SVM

Figure 12 shows excellent classification performance for all three classes in the confusion matrix, where predicted classes lie along the horizontal axis and actual classes along the vertical axis. The model correctly detected 60 cases of Root Crack, 60 of Broken, and 61 of Healthy, with only a few misclassifications. The resulting Precision, Recall, and F1 score were 0.9836, 0.9677, and 0.9756 for the Broken class; 0.9531, 0.9839, and 0.9682 for the Healthy class; and 0.9836, 0.9677, and 0.9756 for the Root Crack class.

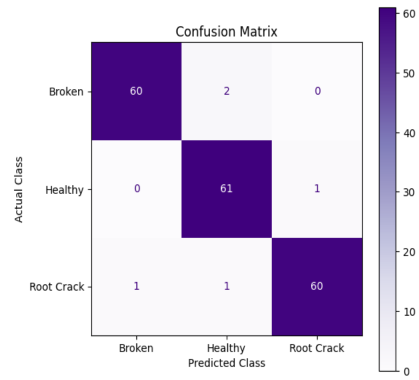

Figure 12. Confusion Matrix for SVM

The overall parameters that found from the SVM

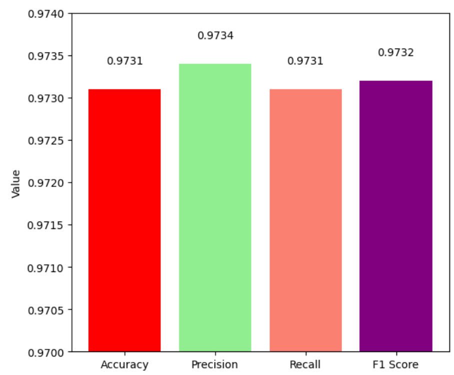

Figure 13. Performance Metrics of SVM

3.4.1. Learning curve for SVM

From Figure 14 following insights were obtained:

- At small training sizes, validation accuracy is much lower than training accuracy, indicating variance and overfitting.
- Validation accuracy rises quickly as the training set grows, showing the model improves with more data.
- Training and validation accuracy converge and stabilise after about 275 samples, indicating generalizable patterns and a good bias–variance balance.

Figure 14. Learning curve for SVM

3.5. Model Performance in ANN

Figure 15 shows excellent classification performance for all three classes in the confusion matrix, where predicted classes lie along the horizontal axis and actual classes along the vertical axis. The model correctly detected 61 cases of Root Crack, 62 of Broken, and 62 of Healthy, with only a single misclassification. The resulting Precision, Recall, and F1 score were 1.00, 1.00, and 1.00 for the Broken class; 0.9841, 1.00, and 0.992 for the Healthy class; and 1.00, 0.9839, and 0.9919 for the Root Crack class.

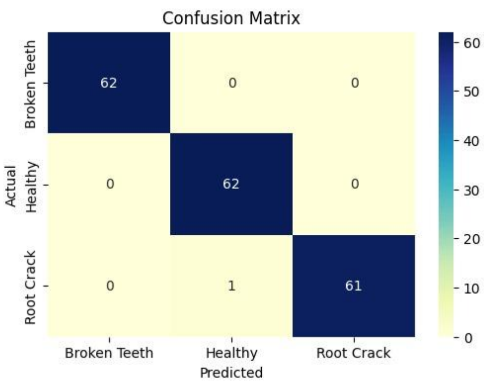

Figure 15. Confusion Matrix for ANN

The overall parameters that found from the ANN

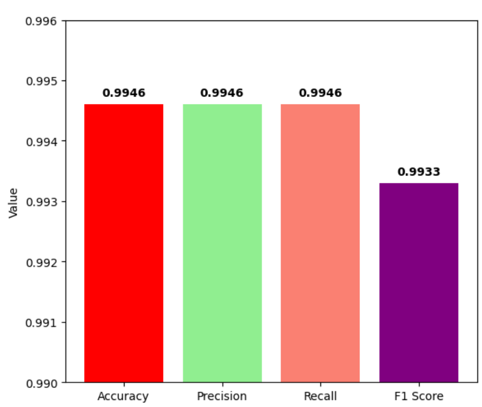

Figure 16. Performance Metrics for ANN

3.5.1. Model Accuracy vs Epoch for ANN

An epoch is one complete pass over the training dataset, during which the model adjusts its weights and biases to reduce the estimated loss. The accuracy vs. epoch curve helps monitor how well the model performs during training and aids early problem detection by comparing training and validation accuracy. Figure 17 shows the model's performance over several epochs. The initial performance is low due to random weight initialization, but after epoch 10 the curve converges and becomes smooth and steady, indicating that the model successfully learned the features of the training data. At about epoch 53 the accuracy reaches a maximum of about 99.46%, with both training and validation accuracy levelling off near 1.00. The small discrepancy between the two confirms strong generalization performance.

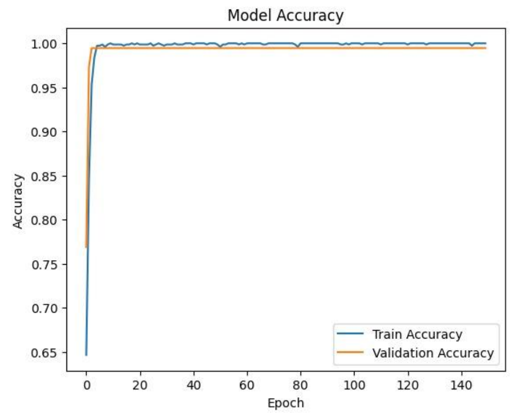

Figure 17. Accuracy vs Epoch curve for ANN

3.5.2. Loss Accuracy vs Epoch for ANN

The loss is the difference between the actual target value and the model's predicted output, which the model minimises by adjusting its weights during training. Training loss and validation loss are two key measures for tracking performance and generalization. Figure 18 shows that both training and validation loss decrease drastically.

The sharp reduction within epochs 0–10 reflects effective parameter adjustment and rapid capture of the underlying data patterns. As training progresses the rate of reduction slows, and after approximately epoch 30 the loss curve flattens and fluctuates slightly around a low value. These minor variations up to 50 epochs indicate mild overfitting; however, the close alignment between training and validation loss confirms that overfitting is not severe and that the model maintains good generalization performance.

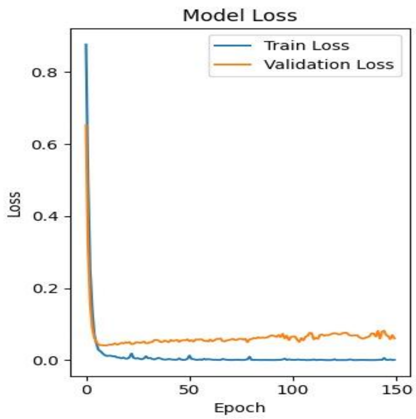

Figure 18. Loss vs Epoch curve for ANN

3.5.3 Cross-Validation Analysis

Because the ANN was selected for real-time deployment in the GUI (Section 3.7), a 5-fold stratified cross-validation was performed to verify reproducibility across data partitions, using the same dataset and pipeline with Min–Max normalisation re-fitted independently per fold. The ANN achieved a mean accuracy of 99.89 ± 0.22 % across the five folds; the single-split result of 99.46 % corresponds to the worst fold, while the other four yielded perfect classification, confirming that the reported performance is stable and reproducible.

3.6. Comparison among AI models

Among the three models (ANN, RF, SVM), ANN gave the highest accuracy. ANN captures context-dependent vibration patterns more accurately than RF or SVM and performs well on continuous, normalised data such as ours. It can also down-weight less informative inputs like a constant speed, whereas RF and SVM are more prone to dimensionality issues.

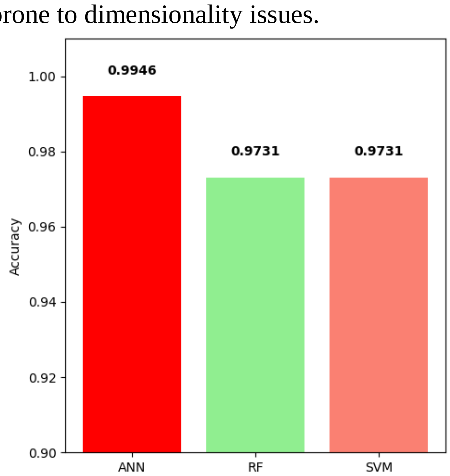

Figure 19. Accuracy vs Model

Deployment choice could not rest on accuracy alone, so the three models were also compared on three practical criteria for real-time operation: training time, inference latency per sample, and memory footprint. The benchmark ran on a CPU-only laptop (the GUI's target platform), averaging training time over five runs and inference time over fifty passes through the test set. Results are summarised in Table 1.

Table 1. Training and inference benchmark for the three classifiers.

Model	Training time (s)	Inference (ms/sample)	Model size (kB)
Artificial Neural Network	17.402 ± 0.873	0.5573 ± 0.0594	2.89
Support Vector Machine	0.005 ± 0.000	0.0055 ± 0.0003	5.32
Random Forest	0.203 ± 0.005	0.0545 ± 0.0029	381.43

3.7. Real-time Monitoring of the gear system in GUI

A Tkinter-based GUI provides real-time condition monitoring, with the trained ANN model integrated into a Python script that continuously reads ADXL345 sensor data via the Arduino and predicts the gear condition.

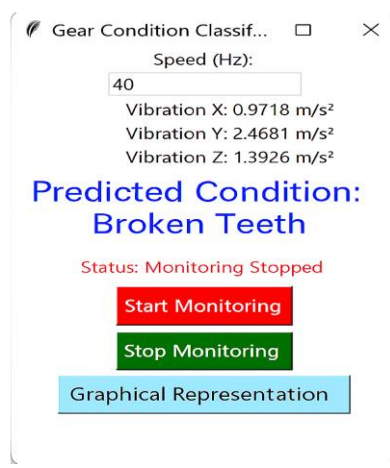

Figure 20. Graphical User Interface

The system takes four inputs—speed and vibration along the x, y, and z axes. Speed is set manually according to operating conditions, and the collected data for a given speed is sent to the model, which displays the predicted condition on the interface. An alarm with distinct sounds for Root Crack and Broken Tooth activates whenever the condition is not healthy and rings continuously until monitoring is stopped.

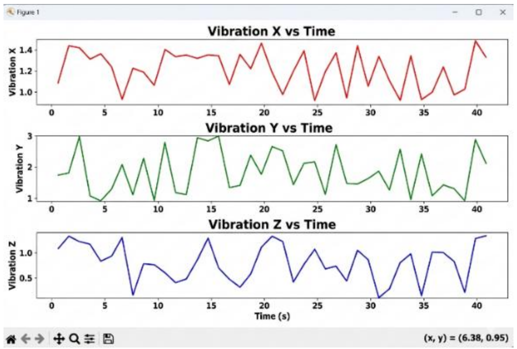

Figure 21. Time dependent Tri-axial vibration data (ms-2) monitoring

Figure 21 shows vibration along the x, y, and z axes, captured via the ADXL345 and Arduino and visualised in the GUI.

Table 2. Graphical User Interface Features

Features	Description
Real-Time Data Feed	Receives acceleration along x, y, z from Arduino for every 1000 ms.
Speed Input	Manual user input.
Graphical Representation	Dynamic graph plotting vibration in three axes over time.
Status Display	Outputs machine condition.
Control of Monitoring	Start/stop monitoring button are used to start/stop monitoring.
Alarm System	Provides alarm for Root crack and Broken teeth.

4. Conclusions

This study proposes a low-cost and reliable AI-based condition monitoring system for gear fault diagnosis using vibration data from an ADXL345 accelerometer. Some key conclusions from this study are:

- A cost-effective vibration-based condition monitoring framework was developed using the ADXL345 accelerometer and Arduino Uno, collecting data along three axes (X, Y, Z) under Healthy, Root Crack, and Broken Tooth gear conditions.
- From over 1400 samples, 930 processed vibration samples (310 per class) were used to train and evaluate three AI models—Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Network (ANN).
- The ANN model achieved the highest performance, with an accuracy of 99.46%, outperforming RF and SVM.
- A Tkinter-based real-time GUI was implemented to continuously classify incoming sensor data and visualize vibration signals.
- An alarm system with distinct alerts for Root Crack and Broken Tooth conditions was integrated to enhance operational safety.

This study has several limitations. Data came from a single laboratory-scale rig under controlled conditions, which may limit generalisability to industrial systems in variable, noisy environments. Only three gear conditions were considered, the consumer-grade ADXL345 has sensitivity constraints relative to industrial transducers, and the framework relies on a manually set speed input.

Future work could explore deeper architectures such as CNN and LSTM networks for more complex faults, and transfer learning for cross-machine generalisation. Expanding the dataset across more fault types, loads, and gearbox configurations would improve robustness, while deployment on edge platforms such as Raspberry Pi or NVIDIA Jetson would enable autonomous, low-latency monitoring in industrial settings.

References

[1] T. Praveenkumar, M. Saimurugan, P. Krishnakumar, and K. I. Ramachandran, “Fault Diagnosis of Automobile Gearbox Based on Machine Learning Techniques,” *Procedia Eng.*, vol. 97, pp. 2092–2098, Jan. 2014, doi: 10.1016/J.PROENG.2014.12.452.

[2] T. Zhu, Y. Ran, X. Zhou, and Y. Wen, “A Survey of Predictive Maintenance: Systems, Purposes and Approaches,” *IEEE Communications Surveys and Tutorials*, Mar. 2024, doi: 10.1109/COMST.2025.3567802.

[3] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, “Machine learning for predictive maintenance: A multiple classifier approach,” *IEEE Trans. Industr. Inform.*, vol. 11, no. 3, pp. 812–820, Jun. 2015, doi: 10.1109/TII.2014.2349359.

[4] S. J. Wu, N. Gebraeel, M. A. Lawley, and Y. Yih, “A neural network integrated decision support system for condition-based optimal predictive maintenance policy,” *IEEE Transactions on Systems, Man, and*

Cybernetics Part A:Systems and Humans, vol. 37, no. 2, pp. 226–236, Mar. 2007, doi: 10.1109/TSMCA.2006.886368.

[5] E. Frontoni, R. Pollini, P. Russo, P. Zingaretti, and G. Cerri, “HDOMO: Smart Sensor Integration for an Active and Independent Longevity of the Elderly,” *Sensors 2017, Vol. 17, Page 2610*, vol. 17, no. 11, p. 2610, Nov. 2017, doi: 10.3390/S17112610.

[6] J. Yin, W. Wang, Z. Man, and S. Khoo, “Statistical modeling of gear vibration signals and its application to detecting and diagnosing gear faults,” *Inf. Sci. (N. Y).*, vol. 259, pp. 295–303, Feb. 2014, doi: 10.1016/J.INS.2013.03.029.

[7] O. D. Mohammed, M. Rantatalo, J. O. Aidanpää, and U. Kumar, “Vibration signal analysis for gear fault diagnosis with various crack progression scenarios,” *Mech. Syst. Signal Process.*, vol. 41, no. 1–2, pp. 176–195, Dec. 2013, doi: 10.1016/J.YMSSP.2013.06.040.

[8] A. K. S. Jardine, D. Lin, and D. Banjevic, “A review on machinery diagnostics and prognostics implementing condition-based maintenance,” *Mech. Syst. Signal Process.*, vol. 20, no. 7, pp. 1483–1510, Oct. 2006, doi: 10.1016/J.YMSSP.2005.09.012.

[9] M. Baptista, S. Sankararaman, I. P. de Medeiros, C. Nascimento, H. Prendinger, and E. M. P. Henriques, “Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling,” *Comput. Ind. Eng.*, vol. 115, pp. 41–53, Jan. 2018, doi: 10.1016/J.CIE.2017.10.033.

[10] T. P. Carvalho, F. A. A. M. N. Soares, R. Vita, R. da P. Francisco, J. P. Basto, and S. G. S. Alcalá, “A systematic literature review of machine learning methods applied to predictive maintenance,” *Comput. Ind. Eng.*, vol. 137, p. 106024, Nov. 2019, doi: 10.1016/J.CIE.2019.106024.

[11] N. A. Mohammed, O. F. Abdulateef, and A. H. Hamad, “An IoT and machine learning-based predictive maintenance system for electrical motors,” *Journal Européen des Systèmes Automatisés*, vol. 56, no. 4, p. 651, 2023.

[12] J. Antonino-Daviu, “Electrical monitoring under transient conditions: A new paradigm in electric motors predictive maintenance,” *Applied Sciences*, vol. 10, no. 17, p. 6137, 2020.

[13] L. Dhamande, … A. P.-I. J. of, and undefined 2014, “Detection of Fault in Gear Box System using Vibration Analysis Method,” *researchgate.net*, vol. 3985, no. 6, 2014, Accessed: Oct. 21, 2025. [Online]. Available: https://www.researchgate.net/profile/Amar-Pawar-4/publication/359106968_Detection_of_Fault_in_Gear_Box_System_using_Vibration_Analysis_Method/links/62283f6a84ce8e5b4d122dc9/Detection-of-Fault-in-Gear-Box-System-using-Vibration-Analysis-Method.pdf

[14] A. Kumar, R. Shankar, and L. S. Thakur, “A big data driven sustainable manufacturing framework for condition-based maintenance prediction,” *J. Comput. Sci.*, vol. 27, pp. 428–439, Jul. 2018, doi: 10.1016/J.JOCS.2017.06.006.

[15] I. U. Hassan, K. Panduru, and J. Walsh, “An in-depth study of vibration sensors for condition monitoring,” *Sensors*, vol. 24, no. 3, p. 740, 2024.

[16] Z. Mones, Q. Zeng, L. Hu, X. Tang, F. Gu, and A. D. Ball, “Planetary gearbox fault diagnosis using an on-rotor MEMS accelerometer,” in *2017 23rd International Conference on Automation and Computing (ICAC)*, IEEE, 2017, pp. 1–6.