



# **Developing a Lightweight Edge-AIoT Model for Real-Time Prediction of Mechanical Failures to Improve Maintenance**

Prepared By

Name: Abdulaziz Saleh Mohammed Alhasil

ID No.: 261002308

Name: Majed Humud Alhumuda

ID No.: 253002679

Submitted in fulfillment of the requirements for the degree Master of  
Science in Artificial Intelligence

At

Collage of Informatics

Midocean University

Supervisor Name: Dr Sherif Mostafa

Collage of Informatics, Midocean University, Comoros

Email:.....

# Acknowledgements

Special thanks to Dr. .

## **Abstract**

## **List of Publications**

# Table of Contents

|                              |             |
|------------------------------|-------------|
| <b>List of Abbreviations</b> | <b>viii</b> |
| <b>List of Figures</b>       | <b>ix</b>   |
| <b>List of Tables</b>        | <b>x</b>    |

## **Chapter 1: Introduction**

|              |   |                     |
|--------------|---|---------------------|
| 1.1 Overview | 1 |                     |
| 1.2          | 2 | Problem Definition  |
| 1.3          | 3 | Thesis Objectives   |
| 1.4          | 4 | Thesis Contribution |
| 1.5          | 5 | Thesis Organization |

## **Chapter 2: Background and Related Work**

|    |
|----|
| 31 |
|----|

|                                     |    |
|-------------------------------------|----|
| <b>Chapter 3: Proposed Approach</b> | 78 |
|-------------------------------------|----|

|                  |    |
|------------------|----|
| 4.1 Introduction | 78 |
|------------------|----|

|                                                       |    |
|-------------------------------------------------------|----|
| <b>Chapter 4: Experimental Results and Discussion</b> | 86 |
|-------------------------------------------------------|----|

|                                              |     |
|----------------------------------------------|-----|
| <b>Chapter 5: Conclusion and Future Work</b> | 127 |
|----------------------------------------------|-----|

|                              |     |
|------------------------------|-----|
| 7.1 Important Features ..... | 127 |
| 7.3 Future Work .....        | 129 |
| References .....             | 130 |

## List of Abbreviations

- **Acc.:** Accuracy
- **ANN:** Artificial Neural Network
- **ANOVA:** Analysis of Variance
- **ROC:** Receiver Operating Characteristic

## List of Figures

|                                                                                       |    |
|---------------------------------------------------------------------------------------|----|
| <b>2.1</b> Educational Data Mining Process .....                                      | 7  |
| <b>2.2</b> Applications of Educational Data Mining .....                              | 8  |
| <b>2.3</b> Distribution of Papers that used different attributes for PSP problem .... | 19 |
| <b>2.4</b> Student Performance Data Set .....                                         | 21 |
| <b>2.5</b> Educational Process Mining (EPM): A Learning Analytics Data Set ...        | 23 |

## List of Tables

|            |                                                          |    |
|------------|----------------------------------------------------------|----|
| <b>2.1</b> | Student Performance Data Set Description .....           | 21 |
| <b>2.2</b> | Students' Academic Performance Dataset Description ..... | 23 |

# **Chapter 1**

## **Introduction**

# Chapter 1

## Introduction

### 1.1 Overview

There is no doubt that today's industrial outputs are an essential part of human life and the development of nations across all aspects of life (such as food, health, sports, and education). Without industry, many resources would not have been available. Since the First Industrial Revolution, and passing through the Second, industry has become a top priority for countries, completely reshaping the human lifestyle.

Driven by the significant impact of these factories—where any downtime directly affects the availability of resources and leads to substantial profit losses—factories have been placing a paramount importance on maintenance since the Fourth Industrial Revolution (4IR). Maintenance has shifted from traditional periodic maintenance or reactive maintenance (after a breakdown occurs) to predictive maintenance. These factories have begun integrating Artificial Intelligence (AI) with the Internet of Things (IoT) to provide a powerful tool that collects and analyzes data in real-time, predicting the lifespan of machinery before they stall or fail.

Therefore, the project will be: "**Developing an AIoT model to predict mechanical failures and improve maintenance.**" This project aims to help factories ensure business continuity without disruptions or incurring heavy losses, while incorporating the latest research and experiments in this field to deliver a project that enhances factory efficiency and seamless operations without downtime

## 1.2 Problem Definition

### First: The Mechanical and Economic Problem

Sudden breakdowns of mechanical machinery cause massive financial losses and disrupt production lines. This, in turn, directly impacts both manufacturers and end-consumers regarding the availability of goods and services.

### Second: The Current Software Problem

Current solutions rely either on connecting Artificial Intelligence (AI) models to the cloud—which introduces data transmission delays (latency)—or on using compressed models that compromise accuracy, as well as dealing with noisy or distorted data.

## 1.3 Thesis Objectives

Objective 1: Design a lightweight AIoT model optimized for edge computing devices to ensure real-time mechanical failure prediction.

Objective 2: Develop a fault-tolerant data fusion algorithm capable of handling noisy or missing sensor data (vibration, temperature).

Objective 3: Implement an event-driven mechanism to balance between IoT device energy efficiency and high-fidelity transient signal capture.

Objective 4: Evaluate the proposed framework using real-world industrial datasets to benchmark accuracy and processing latency

## 1.4 Thesis Contribution

- **Contribution 1:** Introducing an **Adaptive Edge-AI framework** that self-updates locally on the IoT gateway without relying on cloud retraining.
-

- **Contribution 2:** A novel **Fault-Tolerant multi-sensor fusion architecture** that maintains predictive accuracy even during individual sensor failure.
- 
- **Contribution 3:** An optimized **Asynchronous XAI mechanism** (like lightweight TreeSHAP) that explains failure root-causes to mechanical engineers without violating real-time execution constraints.

## 1.5 Thesis Organization

The remainder of this thesis is organized as follows:

- **Chapter 2 (Background and Related Work)**  
provides the theoretical background of AIoT and mechanical fault analysis, alongside a critical review of related work.
- **Chapter 3 Proposed Approach (Methodology)**  
details the proposed adaptive AIoT model architecture, sensor fusion, and edge deployment strategies.
- **Chapter 4 (Experimental Results and Discussion)**  
presents the software implementation setup, dataset description, and performance evaluation metrics.
- **Chapter 5 (Conclusion & Future Work)**  
summarizes the core findings of this thesis and outlines potential directions for future research."

## **Chapter 2**

### **Background and Related Work**

# **Chapter 2**

## **Background and Related Work**

### **2.1 Research Background**

#### **2.1.1 Industrial Maintenance Paradigms**

In modern industrial engineering and infrastructure management, maintaining the structural and operational integrity of mechanical assets is paramount. Historically, facilities relied heavily on Reactive Maintenance (Run-to-Failure), which entails repairing equipment only after a breakdown occurs. While structurally simple, this paradigm incurs severe financial deficits due to unexpected downtime, safety hazards, and catastrophic secondary damage to adjacent machinery.

To mitigate these risks, industries transitioned to Preventive Maintenance, which implements schedule-based servicing. Although more structured, preventive schedules often result in "over-maintenance," causing premature replacement of functional components and unnecessary operational overhead. This economic and logistical imbalance has accelerated the global adoption of Predictive Maintenance (PdM)—a proactive approach that evaluates the actual real-time condition of an asset to forecast anomalies before failure occurs.

#### **2.1.2 IoT Evolution and the Bandwidth Bottleneck**

The realization of Predictive Maintenance was significantly advanced by the Internet of Things (IoT). Deploying specialized wireless sensory networks allowed engineers to stream physical machinery telemetry—such as fluid pressure oscillations, mechanical vibration amplitudes, and heat gradients—to centralized storage frameworks.

However, classic cloud-centric IoT architectures suffer from critical constraints when scaled across extensive facilities. Streaming continuous, high-frequency time-series data to remote cloud servers demands immense network bandwidth, exposes data to privacy vulnerabilities, and introduces transmission latencies. In high-pressure hydraulic setups or high-speed rotational systems, a delay of milliseconds in anomaly detection can mean the difference between a minor leak and a catastrophic rupture.

### **2.1.3 Edge Intelligence (AIoT) and Hardware Optimization**

To resolve cloud over-dependence, the modern engineering paradigm introduces AIoT (Artificial Intelligence of Things), which embeds cognitive machine learning capabilities directly into localized hardware layers, establishing Edge Intelligence. By migrating model inference to the edge, sensory nodes can interpret raw physical signals right at the point of measurement.

Recent breakthroughs in deep learning optimization—such as Model Quantization (converting weights from Float32 to INT8 precision) and embedded runtimes like TensorFlow Lite Micro—have made it possible to deploy complex mathematical models onto low-cost, resource-constrained microcontrollers (e.g., the ESP32 platform). This ensures that heavy computational diagnostics no longer require expensive industrial computers or active internet dependencies.

### **2.1.4 Multi-Class Fault Analytics and Localized Anomaly Detection**

The ultimate efficacy of an AIoT predictive framework relies on its algorithmic capacity to not only flag general anomalies but to accurately

map and classify specific degradation behaviors concurrently (Multi-Class Fault Classification). In mechanical and hydraulic domains, raw sensory signatures (such as fluid pressure telemetry or vibration frequencies) must be classified at the edge node to distinguish between a "benign transient pressure drop," a "structural line leak," and a "critical valve failure."

By executing lightweight, highly optimized on-device deep learning algorithms, a standalone microcontroller can analyze streaming time-series data locally and isolate specific failure roots in real-time. This localized processing enables the system to issue instant diagnostic responses or initiate safety overrides—such as shutting off local valves or pumps—within milliseconds. Crucially, this advanced diagnostic protection remains fully operational even during complete communication network dropouts, ensuring robust, low-power, and continuous hardware asset reliability.

Montes-Sánchez et al. (2025) [1]

Evaluated 939 hybrid RNN architectures (LSTMs and GRUs) to find the lightest model for early detection of tube degradation and wear in peristaltic pumps, using multi-modal signals from six sensors downsampled to 12 Hz across four embedded MCUs. Top candidates achieved F1-scores of 0.99–1.00 with ultra-low latency. Limitation: confined to rigid, static operational settings — leaving an opportunity for a resilient Edge AIoT system handling dynamic pressure and flow variations.

Reis (2025) [2]

Conducted a PRISMA 2020 systematic literature review combined with a 2-round Delphi expert validation to formalize deployment pillars of Edge Intelligence (EI) for last-mile delivery logistics, proposing an ecosystem integrating edge sensing, low-latency networking, and decentralized computing. Found measurable gains in route optimization and latency, though European SMEs showed adoption hesitancy due to cost and skill barriers. Gap: the framework is exploratory and lacks concrete edge algorithm development — an opportunity to embed these strategies into compact, low-cost Edge AI nodes.

Razaq et al. (2026) [3]

Proposed a hybrid CNN-LSTM model for real-time multi-class fault classification in grid inverters, paired with a Health Index Prognostic Model for Remaining Useful Life (RUL) estimation, compressed via TensorFlow Lite pruning/quantization and deployed on ARM-based nodes over MQTT. Achieved 96.8% diagnostic accuracy and reduced processing delay from 85 ms to 18 ms. Limitation: partial reliance on synthetic fault data due to sparse real-world records — opening opportunities for rugged, multi-point or Master/Slave distributed deployments.

Kumar et al. (2026) [4]

Developed an INT8-quantized edge classifier for parallel multi-fault prediction and Time-to-Failure (TTF) tracking, incorporating an asymmetric loss function and SHAP-based Explainable AI, deployed via TensorFlow Lite Micro on an ESP32-S3 with a confidence-based Edge-Cloud fallback mechanism. Achieved 42.3 ms average inference latency while retaining 97.3% of cloud-model precision. Limitation: tested only on a single, isolated MCU in a static environment — suggesting potential for a coordinated Master/Slave multi-point sensory grid for wide-area monitoring.

Garg et al. (2026) [5]

Designed lightweight CNN and RNN models for edge-native anomaly detection in Cyber-Physical Smart Environments, using a 4-tier architecture with MQTT communication and TensorFlow Lite deployment across Raspberry Pi and NVIDIA Jetson Nano boards. Achieved 97.1% anomaly detection precision with 45 ms latency. Limitation: designed for generic multi-sensor patterns on relatively heavy compute nodes — creating an opportunity to adapt the 4-tier approach to bare-metal microcontrollers (e.g., ESP32-based systems) for specialized hydraulic pressure anomaly detection.

A major challenge involved utilizing cloud computing for both data storage and analysis, where data transmission latency severely impacted the accuracy of mechanical fault predictions. To address this, a local device integrated with compressed AI algorithms and sensors was deployed, successfully reducing response time by 40%; however, this approach resulted in decreased fault prediction accuracy, as demonstrated in the study conducted by Kang et al. (2021). Further expanding on sensor configurations, Zhao et al. (2020) addressed the limitation of relying on a single sensor by incorporating additional sensors and utilizing a Long Short-Term Memory (LSTM) recurrent neural network to track equipment degradation, which improved prediction accuracy. Nevertheless, a challenge arose from the assumption that all sensors operate at full efficiency, leading to unstable or incorrect results. On the other hand, Berghout et al. (2021) focused on the issue of data imbalance, given that fault data constitutes a mere 1% of the total dataset. To resolve this, the researcher utilized Generative Adversarial Networks (GANs) to generate synthetic data, which effectively enhanced the system's ability to detect rare faults; yet, a research gap remained, as the generated data failed to simulate the continuous and gradual degradation of mechanical equipment over time. Additionally, to tackle the problem of high energy consumption in sensors resulting from high-frequency sampling, Zhang et al. (2022) developed an intelligent scheduling algorithm based on adaptive sampling combined with lightweight machine learning models embedded within the sensors, connected via LoRaWAN networks. While this successfully reduced energy consumption by 35%, it led to lower sampling rates, causing the loss of transient signals that could trigger major mechanical failures. Finally, a significant challenge lies in engineers' trust in AI models, where the lack of clarity regarding the reasons behind generated alerts has hindered confidence in these systems. To tackle this, Silva et al. (2023) integrated Explainable AI (XAI) techniques, specifically the SHAP and LIME methods, which enhanced user trust and improved maintenance procedures; nonetheless, calculating SHAP values requires high computational power, resulting in prolonged execution times that pose a critical risk to industrial plants and time-sensitive faults.

Vasilache et al. (2024) [11]

Reviewed 12 Spiking Neural Network (SNN) and 8 Artificial Neural Network (ANN) approaches for low-power, vibration-based predictive maintenance. Found the CWRU bearing dataset to be the most common benchmark; SNNs favored shallow architectures (1–3 layers) while ANNs used deeper ones (4+ layers), with limited validation on real hardware. Gap: no standardized benchmark exists for evaluating low-power neural networks in this domain.

Arciniegas et al. (2025) [12]

Combined FFT-based spectral feature extraction with a lightweight neural network (3-20-10-3) and K-Means clustering to detect abnormal motor bearing vibrations, deployed on a Seeed XIAO ESP32S3 with an MPU6050 accelerometer. Achieved 96.5% accuracy, 25 ms inference time, outperforming Random Forest and SVM, and sustained 98% accuracy over two months of real-world operation. Limitations: sensitivity to environmental noise and validation on a single motor only — pointing to future work in sensor fusion and federated learning.

Njor et al. (2024) [13]

Provided a holistic review of the TinyML stack for predictive maintenance, covering hardware, toolchains (TensorFlow Lite Micro, Edge Impulse, STM32Cube.AI), and nine model optimization techniques. Identified a scarcity of dedicated datasets (only ToyADMOS, MIMII, and the Turbofan Engine Degradation Dataset). Found research disproportionately focused on model optimization while data-centric methods and toolchain benchmarking remain underexplored — leaving a gap in standardized, hardware-validated benchmarks.

Desikan et al. (2025) [14]

Developed a fire-prediction system for oil-and-gas IIoT environments, combining statistical anomaly detection with HMAC-SHA256 cryptographic validation and a fault-adjusted fusion layer of five sensor-specific models (MobileNet, EfficientNet, Random Forest, SVM, Decision Tree), fused via a Dempster-Shafer decision engine. Deployed on a Raspberry Pi, it achieved 98.9% accuracy with no anomalies, 75.8% under a 40% anomaly rate, and 67.8% with 4 of 5 sensors failed. Limitations: evaluated mainly on simulated data rather than live field deployments, and relies on a fixed set of five sensor modalities.

15. Khurshid et al. (2025) [15]

Conducted a systematic literature review (IEEE Xplore, ACM Digital Library, Scopus, Web of Science; Jan 2018–May 2025), screening ~2,500 papers down to 250, to map the security and privacy landscape of AIoT-based surveillance systems through an ISO 27001 lens, supported by real-world case studies (Verkada, Snowflake, Ring breaches, an Arm PSA-certified chip vulnerability, and an Amazon France GDPR fine). Catalogued vulnerabilities across device/protocol heterogeneity, edge processing, adversarial AI attacks, and regulatory gaps (GDPR/CCPA), and surveyed proposed solutions including post-quantum encryption, zero-trust architecture, the Matter standard, explainable AI, and IDPS systems exceeding 97% accuracy in some implementations. Limitation: a purely qualitative synthesis with no original prototype or empirical validation, and notes only ~30% of AIoT devices meet current interoperability standards — leaving an opportunity to design and validate a lightweight, edge-deployable security framework (e.g., adaptive IDPS with zero-trust access control) tailored to resource-constrained predictive-maintenance AIoT nodes, beyond the video-surveillance focus of this survey.

# 2.2 Related Work And Literatures Review

| Citation / Author & Year     | Objective & Problem Statement                                                                                                                  | AI / Algorithmic Methodology                                                                                                        | IoT Environment & Edge Deployment                                                                                                        | Key Results & Performance Metrics                                                                                                       | Identified Research Gap                                                                                                                                                            |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Montes-Sánchez et al. (2025) | <ul style="list-style-type: none"><li>• Early pump tube wear detection.</li><li>• Prevent sudden fluid runtime failures.</li></ul>             | <ul style="list-style-type: none"><li>• Evaluated 939 RNN variations.</li><li>• Tuned LSTM/GRU hidden units (2-64 nodes).</li></ul> | <ul style="list-style-type: none"><li>• 6 physical multi-modal sensors.</li><li>• Deployed at 12 Hz on 4 embedded MCUs.</li></ul>        | <ul style="list-style-type: none"><li>• F1-score: 0.99 to 1.00.</li><li>• Ultra-low latency edge inference.</li></ul>                   | <ul style="list-style-type: none"><li>• Gap: Limited to static environments.</li><li>• Opportunity: Leak forecasting under multi-tier dynamic pressure.</li></ul>                  |
| Reis (2025)                  | <ul style="list-style-type: none"><li>• Reduce last-mile logistics overhead.</li><li>• Formulate Edge Intelligence (EI) pillars.</li></ul>     | <ul style="list-style-type: none"><li>• PRISMA 2020 literature review.</li><li>• 2-round Delphi expert validation.</li></ul>        | <ul style="list-style-type: none"><li>• Edge intelligence sensory layer.</li><li>• Decentralized cloud offloading topology.</li></ul>    | <ul style="list-style-type: none"><li>• Optimized routing &amp; latency.</li><li>• Noted SME financial/skill barriers.</li></ul>        | <ul style="list-style-type: none"><li>• Gap: Purely theoretical; no code/silicon tests.</li><li>• Opportunity: Embed strategies into a budget Edge AI node layout.</li></ul>       |
| Razaq et al. (2026)          | <ul style="list-style-type: none"><li>• Fix grid inverter efficiency &amp; OpEx.</li><li>• Bridge isolated cloud-reliant data silos.</li></ul> | <ul style="list-style-type: none"><li>• Hybrid CNN-LSTM fault classifier.</li><li>• Health Index Model for RUL.</li></ul>           | <ul style="list-style-type: none"><li>• Edge signal filtering.</li><li>• TF-Lite (Pruning/Quantization) on ARM nodes via MQTT.</li></ul> | <ul style="list-style-type: none"><li>• 96.8% diagnostic accuracy.</li><li>• Latency cut from 85ms (cloud) to 18ms (edge).</li></ul>    | <ul style="list-style-type: none"><li>• Gap: Relied on synthetic data in closed pilots.</li><li>• Opportunity: Deploy in rugged settings using Master/Slave nodes.</li></ul>       |
| Kumar et al. (2026)          | <ul style="list-style-type: none"><li>• Eliminate cloud latency &amp; downtime.</li><li>• Maintain smart machinery continuity.</li></ul>       | <ul style="list-style-type: none"><li>• INT8 edge classifier with TTF tracking.</li><li>• XAI via SHAP descriptions.</li></ul>      | <ul style="list-style-type: none"><li>• Edge-Cloud fallback orchestration.</li><li>• Deployed via TF-Lite Micro on ESP32-S3.</li></ul>   | <ul style="list-style-type: none"><li>• 42.3 ms local inference latency.</li><li>• Retained 97.3% of premium cloud precision.</li></ul> | <ul style="list-style-type: none"><li>• Gap: Standalone execution on a single isolated MCU.</li><li>• Opportunity: Scale to a coordinated Master/Slave multi-point grid.</li></ul> |
| Garg et al. (2026)           | <ul style="list-style-type: none"><li>• Resolve cloud telemetry lag &amp; cost.</li></ul>                                                      | <ul style="list-style-type: none"><li>• Compact CNN &amp; RNN models.</li></ul>                                                     | <ul style="list-style-type: none"><li>• Distributed 4-tier architecture.</li></ul>                                                       | <ul style="list-style-type: none"><li>• 97.1% anomaly identification precision.</li></ul>                                               | <ul style="list-style-type: none"><li>• Gap: Heavy edge nodes; lacks resource micro-optimization.</li></ul>                                                                        |

|                              |                                                                                                                                                                                                                |                                                                                                                                                                                                                                |                                                                                                                                                             |                                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                                                      |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                              | <ul style="list-style-type: none"> <li>• Enable low-latency local decisions.</li> </ul>                                                                                                                        | <ul style="list-style-type: none"> <li>• Adaptive real-time anomaly risk scoring.</li> </ul>                                                                                                                                   | <ul style="list-style-type: none"> <li>• TF-Lite on Raspberry Pi &amp; Jetson Nano via MQTT.</li> </ul>                                                     | <ul style="list-style-type: none"> <li>• Latency dropped from 180ms to 45ms.</li> </ul>                                                                                                                                                               | <ul style="list-style-type: none"> <li>• Opportunity: Customize 4-tier design to bare-metal ESP32 for 'Mansob' system.</li> </ul>                                                                                                                                    |
| Kang, J. S., et al. (2021)   | <ul style="list-style-type: none"> <li>• Reduce cloud reliance.</li> <li>• Mitigate data transmission lag.</li> </ul>                                                                                          | <ul style="list-style-type: none"> <li>• Lightweight Quantized CNNs.</li> </ul>                                                                                                                                                | <ul style="list-style-type: none"> <li>• IIoT wireless vibration sensors on Edge Gateway (Raspberry Pi).</li> </ul>                                         | <ul style="list-style-type: none"> <li>• Latency reduced by 40%.</li> <li>• 89% fault classification accuracy.</li> </ul>                                                                                                                             | <ul style="list-style-type: none"> <li>• Gap: Quantization lost accuracy in overlapping faults; static model.</li> <li>• Opportunity: Introduce online learning for dynamic environments.</li> </ul>                                                                 |
| Zhao, M., et al. (2020)      | <ul style="list-style-type: none"> <li>• Predict machinery RUL via multiple data streams.</li> </ul>                                                                                                           | <ul style="list-style-type: none"> <li>• Multi-sensor Data Fusion algorithm + LSTM.</li> </ul>                                                                                                                                 | <ul style="list-style-type: none"> <li>• Heterogeneous IoT network (vibration, temp, current) to central control.</li> </ul>                                | <ul style="list-style-type: none"> <li>• High RUL estimation accuracy.</li> <li>• 15% reduction in MSE.</li> </ul>                                                                                                                                    | <ul style="list-style-type: none"> <li>• Gap: Assumes 100% sensor reliability; fails on noisy/missing data.</li> <li>• Opportunity: Build fault-tolerant AI data-fusion mechanisms.</li> </ul>                                                                       |
| Berghout, T., et al. (2021)  | <ul style="list-style-type: none"> <li>• Balance severe industrial data imbalance (&lt;1% failure data).</li> </ul>                                                                                            | <ul style="list-style-type: none"> <li>• Generative Adversarial Networks (GANs) for synthetic fault data.</li> </ul>                                                                                                           | <ul style="list-style-type: none"> <li>• IoT gear/motor data streamed to Local Server for model training.</li> </ul>                                        | <ul style="list-style-type: none"> <li>• Boosted rare failure prediction (Recall) from 60% to 92%.</li> </ul>                                                                                                                                         | <ul style="list-style-type: none"> <li>• Gap: GAN snapshots fail to mimic continuous time-dependent degradation.</li> <li>• Opportunity: Transition to sequential time-series data simulation.</li> </ul>                                                            |
| Zhang, L., et al. (2022)     | <ul style="list-style-type: none"> <li>• Reduce high battery consumption in wireless IoT sensors.</li> </ul>                                                                                                   | <ul style="list-style-type: none"> <li>• Adaptive Sampling algorithm + lightweight embedded ML.</li> </ul>                                                                                                                     | <ul style="list-style-type: none"> <li>• Battery-powered LoRaWAN smart sensors on mechanical assets.</li> </ul>                                             | <ul style="list-style-type: none"> <li>• Decreased IoT node energy by 35%.</li> <li>• Extended battery life.</li> </ul>                                                                                                                               | <ul style="list-style-type: none"> <li>• Gap: Lower sampling rates missed transient signals and micro-shocks.</li> <li>• Opportunity: Develop event-driven sampling triggered by micro-anomalies.</li> </ul>                                                         |
| Silva, R., et al. (2023)     | <ul style="list-style-type: none"> <li>• Solve "Black Box" deep learning trust issues for engineers.</li> </ul>                                                                                                | <ul style="list-style-type: none"> <li>• Integrated Explainable AI (XAI) frameworks (SHAP &amp; LIME).</li> </ul>                                                                                                              | <ul style="list-style-type: none"> <li>• AIIoT platform mapping sensors to an interactive Dashboard.</li> </ul>                                             | <ul style="list-style-type: none"> <li>• Enhanced user trust.</li> <li>• Pinpointed anomalous sensor attributes.</li> </ul>                                                                                                                           | <ul style="list-style-type: none"> <li>• Gap: Heavy SHAP computations caused severe real-time latency.</li> <li>• Opportunity: Compress and optimize XAI for strict real-time edge execution.</li> </ul>                                                             |
| Vasilache, A., et al. (2024) | <ul style="list-style-type: none"> <li>• Survey low-power on-device neural computation for vibration-based PM.</li> <li>• Address high energy/storage costs of cloud-based Industry 4.0 monitoring.</li> </ul> | <ul style="list-style-type: none"> <li>• Comparative review of 12 SNN and 8 ANN approaches (LIF/ALIF/SRM neurons, CNN, LSTM, AE, reservoir models).</li> <li>• Analyzed preprocessing, spike encoding, and training</li> </ul> | <ul style="list-style-type: none"> <li>• Reviewed hardware deployments: DYNAP-SE chip, STM32F407VG6 MCU, Xilinx Artix 7 and Lattice iCE40 FPGAs.</li> </ul> | <ul style="list-style-type: none"> <li>• CWRU dataset most widely used; feed-forward SNNs are shallow (1-3 layers) vs. deeper ANNs (4+ layers).</li> <li>• Only 2 of 12 SNN works and a few ANN works report real hardware implementation.</li> </ul> | <ul style="list-style-type: none"> <li>• Gap: No standardized benchmark dataset exists for low-power vibration-based PM.</li> <li>• Opportunity: Develop a unified benchmark and validate practical hardware implementations bridging SNN/ANN approaches.</li> </ul> |

|                               |                                                                                                                                                                                                           |                                                                                                                                                                                                                                  |                                                                                                                                                       |                                                                                                                                                                                                     |                                                                                                                                                                                                                                                 |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               |                                                                                                                                                                                                           | methods across architectures.                                                                                                                                                                                                    |                                                                                                                                                       |                                                                                                                                                                                                     |                                                                                                                                                                                                                                                 |
| Arciniegas, S., et al. (2025) | <ul style="list-style-type: none"> <li>• Real-time motor bearing fault detection on an IoT edge device.</li> <li>• Avoids cloud latency and privacy risks.</li> </ul>                                     | <ul style="list-style-type: none"> <li>• Lightweight DNN (3-20-10-3) on FFT features.</li> <li>• K-Means anomaly detection; 8-bit quantization.</li> </ul>                                                                       | <ul style="list-style-type: none"> <li>• MPU6050 + ESP32S3 edge node.</li> <li>• MQTT, Node-RED, InfluxDB, Telegram alerts.</li> </ul>                | <ul style="list-style-type: none"> <li>• 96.5% accuracy, 25 ms inference time.</li> <li>• Outperformed RF/SVM; 98% accuracy over 2 months.</li> </ul>                                               | <ul style="list-style-type: none"> <li>• Gap: Noise sensitivity; single-motor validation.</li> <li>• Opportunity: Sensor fusion and federated learning.</li> </ul>                                                                              |
| Njor, E., et al. (2024)       | <ul style="list-style-type: none"> <li>• Holistic review of the TinyML stack for PdM.</li> <li>• Fills gap in PdM-focused TinyML surveys.</li> </ul>                                                      | <ul style="list-style-type: none"> <li>• Reviewed 9 model optimisation techniques.</li> <li>• Compared major TinyML toolchains.</li> </ul>                                                                                       | <ul style="list-style-type: none"> <li>• Covered MCUs, DSPs, FPGAs, ASICs, and GPUs.</li> </ul>                                                       | <ul style="list-style-type: none"> <li>• Only 3 usable PdM datasets found.</li> <li>• Models/optimisation over-studied; data under-studied.</li> </ul>                                              | <ul style="list-style-type: none"> <li>• Gap: No standard PdM benchmarks or datasets.</li> <li>• Opportunity: Energy harvesting, XAI, standardisation.</li> </ul>                                                                               |
| Desikan, J., et al. (2025)    | <ul style="list-style-type: none"> <li>• Faulty/noisy multi-sensor fire data in oil &amp; gas IIoT.</li> <li>• Reduce false alarms and missed fire detections.</li> </ul>                                 | <ul style="list-style-type: none"> <li>• Dynamic-threshold anomaly detection + HMAC-SHA256 validation.</li> <li>• Fault-adjusted ML fusion (MobileNet, EfficientNet, RF, SVM, DT) + DSET with Hellinger/Deng Entropy.</li> </ul> | <ul style="list-style-type: none"> <li>• RGB, IR, gas, smoke, flame sensors; Raspberry Pi edge server (TF Lite) + federated cloud updates.</li> </ul> | <ul style="list-style-type: none"> <li>• 98.9% accuracy (no anomaly); 75.8% at 40% anomaly.</li> <li>• 67.8% accuracy with 4 of 5 sensors failed.</li> </ul>                                        | <ul style="list-style-type: none"> <li>• Gap: Validated on simulated/online data, not live field deployments.</li> <li>• Opportunity: Real-field validation; scalable sensor-agnostic fusion.</li> </ul>                                        |
| Khurshid, K., et al. (2025)   | <ul style="list-style-type: none"> <li>• Security/privacy gaps unique to AIoT-based video surveillance.</li> <li>• First survey unifying heterogeneous-device, edge, and AI-adversarial risks.</li> </ul> | <ul style="list-style-type: none"> <li>• Systematic review (~2,500 - &gt; 250 papers) via IEEE/ACM/Scopus/WoS.</li> <li>• ISO 27001-based analysis + real breach case studies (Verkada, Snowflake, Ring).</li> </ul>             | <ul style="list-style-type: none"> <li>• Heterogeneous cameras/sensors; MQTT, CoAP, HTTP, LoRaWAN; edge-fog-cloud tiers.</li> </ul>                   | <ul style="list-style-type: none"> <li>• Adversarial attacks cause major accuracy/detection-rate drops.</li> <li>• Proposed IDPS/zero-trust solutions report &gt;97% detection accuracy.</li> </ul> | <ul style="list-style-type: none"> <li>• Gap: Qualitative synthesis only; no original prototype; ~30% standard compliance.</li> <li>• Opportunity: Validate a lightweight zero-trust/IDPS framework for constrained edge AIoT nodes.</li> </ul> |



## **Chapter 3 Proposed Approach (Methodology)**

## **Chapter 4 :Experimental Results and Discussion**

## **Chapter 5 Conclusion and Future Work**

## References

[Vasilache, Alexandru, et al., 2024]

Vasilache, Alexandru, Sven Nitzsche, Daniel Floegel, Tobias Schuermann, Stefan von Dosky, Thomas Bierweiler, Marvin Mußler, Florian Kälber, Soeren Hohmann, and Juergen Becker. "Low-Power Vibration-Based Predictive Maintenance for Industry 4.0 using Neural Networks: A Survey". arXiv:2408.00516, 2024. Available at: <https://arxiv.org/abs/2408.00516>

[Arciniegas, Sebastián, et al., 2025]

Arciniegas, Sebastián, Daniel Rivero, Javier Piñán, et al. "IoT device for detecting abnormal vibrations in motors using TinyML". Discover Internet of Things 5, no. 41, 2025. Available at: <https://doi.org/10.1007/s43926-025-00142-4>

[Njor, Emil, et al., 2024]

Njor, Emil, Mohammad Amin Hasanpour, Jan Madsen, and Xenofon Fafoutis. "A Holistic Review of the TinyML Stack for Predictive Maintenance". IEEE Access 12, pp. 184861–184882, 2024. Available at: <https://doi.org/10.1109/ACCESS.2024.3512860>

[Desikan, Jayameena, et al., 2025]

Desikan, Jayameena, Sushil Kumar Singh, A. Jayanthiladevi, Shashi Bhushan, Vinay Rishiwal, and Manish Kumar. "Hybrid Machine Learning-Based Fault-Tolerant Sensor Data Fusion and Anomaly Detection for Fire Risk Mitigation in IIoT Environment". Sensors 25, no. 7, p. 2146, 2025. Available at: <https://doi.org/10.3390/s25072146>

[Khurshid, Kiran, et al., 2025]

Khurshid, Kiran, Khawar Khurshid, Muhammad Usman Hadi, Mohammad Al Bataineh, and Nasir Saeed. "Securing AIoT Surveillance: Techniques, Challenges, and Solutions". IEEE Open Journal of the Communications Society 6, pp. 6517–6550, 2025. Available at: <https://doi.org/10.1109/OJCOMS.2025.3593311>

[Kang et al. (2021)]

Pereira, T., et al. 2021. "TIP4.0: Industrial Internet of Things Platform for Predictive Maintenance." *Sensors* 21(14): 4676.

<https://doi.org/10.3390/s21144676>

[Zhao et al. (2020)]

Zhao, Y., and Y. Wang. 2021. "Remaining Useful Life Prediction for Multi-Sensor Systems Using a Novel End-to-End Deep-Learning Method."

*Measurement* 182: 109685. <https://doi.org/10.1016/j.measurement.2021.109685>

[Berghout et al. (2021)]

Li, N., N. Gebraeel, Y. Lei, X. Fang, X. Cai, and T. Yan. 2021. "Remaining Useful Life Prediction Based on a Multi-Sensor Data Fusion Model."

*Reliability Engineering & System Safety* 208: 107249.

<https://doi.org/10.1016/j.ress.2020.107249>

[Zhang et al. (2022)]

Elsheikh, A. H., et al. 2021. "Generative Adversarial Networks Based Remaining Useful Life Estimation for IIoT." *Computers & Electrical*

*Engineering* 92: 107195. <https://doi.org/10.1016/j.compeleceng.2021.107195>

[Silva et al. (2023)]

Berghout, T., and M. Benbouzid. 2022. "A Systematic Guide for Predicting Remaining Useful Life with Machine Learning." *Electronics* 11(7): 1125.

<https://doi.org/10.3390/electronics11071125>