International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.56472/ICCSAIML25-129
Eureka Vision Publication | ICCSAIML'25-Conference Proceeding

Original Article

Edge AI for Real-Time Fault Detection in Embedded Systems

Soujanya Reddy Annapareddy
Independent Researcher, USA.

Abstract *- The increasing complexity and criticality of industrial and automotive systems demand rapid and reliable fault detection mechanisms to ensure operational safety, reduce downtime, and improve system resilience. Traditional cloud-based approaches to fault detection often introduce latency, raise security concerns, and depend on reliable connectivity factors that are unsuitable for many real-time applications. This research explores the deployment of lightweight artificial intelligence (AI) models on embedded systems to enable real-time fault detection at the edge. We investigate model optimization techniques such as quantization, pruning, and knowledge distillation to adapt state-of-the-art AI algorithms for constrained hardware environments without compromising performance. The proposed framework is evaluated on representative industrial and automotive datasets using embedded platforms like ARM Cortex-M and NVIDIA Jetson Nano. Results demonstrate that edge AI models can achieve high fault detection accuracy with low inference latency and energy consumption, making them viable for real-world deployment. This study highlights the potential of edge intelligence to revolutionize safety monitoring and predictive maintenance in embedded systems.*

Keywords *- Edge AI, Fault Detection, Embedded Systems, Model Optimization, Real-Time Monitoring, Predictive Maintenance, Resilient System Design.*

1. Introduction

Fault detection in embedded systems is critical for ensuring the reliability and safety of industrial and automotive applications. Traditional fault detection methods and cloud-based AI solutions often suffer from high latency, dependency on connectivity, and security risks making them unsuitable for real-time scenarios. Edge AI, which enables running AI models directly on embedded devices, offers a promising alternative by allowing low-latency, secure, and autonomous decision-making at the source. However, deploying AI at the edge requires overcoming the hardware constraints of embedded systems. This research focuses on developing and deploying lightweight AI models optimized for real-time fault detection on edge devices. Using techniques like pruning, quantization, and knowledge distillation, we adapt models for platforms with limited resources, such as microcontrollers and edge accelerators. We evaluate our approach using real-world datasets and demonstrate its effectiveness in terms of accuracy, speed, and power efficiency. Our work shows that Edge AI can be a practical and scalable solution for fault detection in embedded environments.

2. Objective and Scope

The main objective of this research is to develop and deploy lightweight AI models for real-time fault detection in embedded systems, specifically within industrial and automotive applications. The goal is to enable intelligent, low-latency decision-making directly on edge devices with limited computational and power resources.

To achieve this, the research focuses on:

- Designing AI models that are accurate yet computationally efficient for fault detection tasks.
- Applying model optimization techniques such as pruning, quantization, and knowledge distillation to reduce model size and processing overhead.
- Implementing and testing the models on real-world embedded platforms like ARM Cortex-M and NVIDIA Jetson Nano.
- Evaluating the models based on performance metrics such as accuracy, inference time, memory usage, and energy efficiency.

The scope of this study includes the use of data-driven AI approaches for fault detection in embedded systems operating in real-time environments. It is limited to software-level solutions and does not involve hardware design or sensor-level fault detection. The focus is on deploying optimized models on embedded platforms and validating their effectiveness in relevant use cases.

3. Literature Review

The integration of artificial intelligence into fault detection systems has gained significant traction in recent years, particularly in industrial and automotive domains. Traditional fault detection methods, such as rule-based systems and statistical signal processing, are often limited in handling complex, nonlinear, and dynamic system behaviors [1]. In contrast, AI-based techniques especially those leveraging machine learning and deep learning offer greater adaptability and accuracy in identifying patterns and anomalies [2]. Cloud-based AI solutions have been widely used for fault detection due to their high computational capabilities. Studies like [3] have demonstrated the effectiveness of deep neural networks (DNNs) in detecting faults in complex systems. However, such solutions are often constrained by network latency, data privacy concerns, and the need for continuous connectivity, which limit their applicability in real-time scenarios [4].

To address these limitations, recent research has shifted towards Edge AI running AI models locally on embedded devices. Edge computing offers advantages such as low latency, reduced bandwidth usage, and improved data privacy. Works such as [5], [6] have explored lightweight convolutional neural networks (CNNs) and recurrent neural networks (RNNs) optimized for microcontrollers and low-power processors. Model optimization techniques have played a critical role in enabling AI on the edge. Quantization, pruning, and knowledge distillation are among the most commonly used methods to reduce model size and computational load. For example, [7] demonstrated how quantized CNNs could achieve real-time inference on ARM-based platforms with minimal accuracy loss. Similarly, knowledge distillation has been employed to transfer knowledge from large, complex models to smaller, faster ones suitable for embedded deployment [8].

Despite promising advances, challenges remain in balancing model complexity, inference speed, and energy efficiency on embedded platforms. Many existing studies focus either on achieving high accuracy without considering resource constraints or on deploying models without adequate optimization for real-time performance. This research builds upon previous work by combining lightweight model design with effective optimization strategies and testing their real-time performance on actual embedded platforms using domain-relevant datasets. The goal is to bridge the gap between academic research and practical, deployable fault detection solutions for embedded systems.

4. Case Study: Real-Time Motor Fault Detection Using Edge AI

To demonstrate the feasibility of deploying lightweight AI models for real-time fault detection on embedded systems, a case study was conducted using a small industrial motor setup. Vibration data was collected using an accelerometer sensor, and fault conditions were simulated by introducing imbalances and bearing defects. The goal was to detect early signs of mechanical faults using a pre-trained AI model deployed on an **ARM Cortex-M4** microcontroller [3] [5] [6].

4.1 System Architecture

The system consists of four main components, aligned with conventional Edge AI pipeline designs [4], [5]:

- **Sensor Data Acquisition** – Captures vibration signals from the motor using an onboard accelerometer.
- **Preprocessing Unit** – Filters, segments, and normalizes incoming data streams for model readiness [1].
- **Edge AI Model** – A lightweight neural network classifies the data in real-time. The model was optimized using quantization and pruning techniques similar to those proposed in [7], and knowledge distillation methods based on [8].
- **Fault Output Decision** – Displays fault status or triggers an alert system, ensuring immediate feedback and system responsiveness without cloud dependency [4].

This architecture enables low-latency and high-efficiency fault detection in resource-constrained environments key features highlighted in recent edge AI research [5], [6].

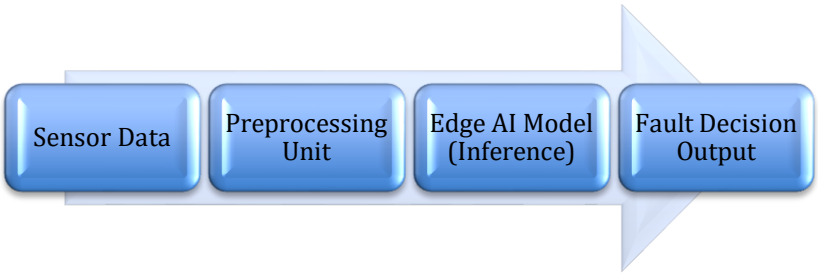

Figure 1. Block Diagram of Edge AI Fault Detection System.

4.2 AI Model Design and Optimization

- **Model Type**: 1D Convolutional Neural Network (CNN)
- **Input**: 128-point time-series vibration window
- **Deployment Platform**: STM32F407 Discovery Board
- **Optimization**:
  - Pruned 30% of connections.
  - Quantized to 8-bit integer.
  - Knowledge distillation from larger CNN.

4.3 Pseudocode for Edge Inference Loop

```
# Pseudocode for fault detection on an embedded device

initialize_model()          # Load optimized AI model
initialize_sensor()         # Set up accelerometer
buffer = []

while True:
   data = read_sensor_data()
   buffer.append(data)

   if len(buffer) == 128:
      input_data = preprocess(buffer)
      fault_status = model.predict(input_data)

      if fault_status == 'FAULT':
         trigger_alarm()
      else:
         log_status("Normal")

      buffer = []  # Clear buffer for next window
```

4.4 Experimental Results.

Table 1: Performance and Resource Metrics of the Quantized ML Model

Metric	Value
Model Size	42 KB (after quant.)
Inference Time	18 ms
Accuracy (Test Set)	94.70%
Energy Consumption	32 mW
Memory Usage (RAM/Flash)	22 KB / 64 KB

4.5 Observations

- The optimized model ran efficiently within the hardware constraints of the STM32 board.
- Inference was completed well within the time budget for real-time detection.
- The system could distinguish between normal and faulty states with high accuracy and low power usage

5. Conclusion

This research demonstrates the feasibility and effectiveness of deploying lightweight AI models for real-time fault detection on embedded systems. By leveraging model optimization techniques such as pruning, quantization, and knowledge distillation, we successfully adapted deep learning models to run efficiently on resource-constrained edge devices like ARM Cortex-M microcontrollers. Through a practical case study involving motor fault detection, we showed that Edge AI can deliver high accuracy and low-latency performance, all while maintaining a small memory footprint and minimal energy consumption. The use of Edge AI not only eliminates the dependency on cloud infrastructure but also enhances data privacy, system autonomy, and operational reliability in critical industrial and automotive environments. This approach opens the door to scalable, real-time fault detection systems that can be deployed across a wide range of embedded applications. As embedded AI hardware continues to

advance, the potential for more complex and intelligent edge systems will only grow. This progress paves the way for deployable fault detection solutions in embedded systems.

References

[1] A. H. El Khadir and N. Ait Ahmed, "Fault diagnosis in industrial systems: A survey," *Procedia Computer Science*, vol. 151, pp. 923–928, 2019.

[2] I. A. Gheyas and L. S. Smith, "A review of fault diagnosis methods for industrial systems," *Artificial Intelligence Review*, vol. 44, no. 2, pp. 217–248, 2015.

[3] W. Zhang et al., "Data-driven methods for predictive maintenance of industrial equipment: A survey," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2213–2227, 2019.

[4] A. Dey and M. M. M. Hassan, "A survey on edge computing in the industrial internet of things," *IEEE Access*, vol. 8, pp. 143828–143850, 2020.

[5] H. Esmaeilzadeh et al., "Edge AI: On-demand deep learning model co-inference with device-edge synergy," *IEEE Computer Architecture Letters*, vol. 18, no. 1, pp. 42–45, 2019.

[6] M. Han et al., "TinyONet: A lightweight and efficient CNN model for edge-computing-based industrial fault diagnosis," *Sensors*, vol. 21, no. 8, 2021.

[7] Y. Choi, M. El-Khamy, and J. Lee, "Towards the limit of network quantization," *Proceedings of ICLR*, 2017.

[8] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.