MPU6050-Based Motion Data Logger and Anomaly Detection Using Nano Edge AI Studio and STM32

MACHINE LEARNING FOR EMBEDDED SYSTEM PROJECT REPORT

Submitted by

BL.EN.P2EBS25010 LOGESH L

Guided by

Dr. Rashmi M.R.

ELECTRICAL AND ELECTRONICS ENGINEERING

AMRITA SCHOOL OF ENGINEERING,

BENGALURU AMRITA VISHWA VIDYAPEETHAM

BENGALURU

560035

November 2025

Table of Contents

CHAPTER 1: INTRODUCTION

Sensors are a crucial part of embedded systems as they measure physical parameters and allow the system to make intelligent decisions. The MPU6050 is a 6-axis MEMS unit that offers 3-axis accelerometer and 3-axis gyroscope data. Here, the MPU6050 sensor is connected to an STM32 Nucleo microcontroller, and a data logger is made to record the movement data in real time.

The data that is gathered serves as the basis for creating datasets for a Machine Learning (ML) anomaly detection model training with the use of Nano Edge AI Studio. Afterward, the trained model is sent back to the STM32 board to help it identify the changes in the vibration pattern of a rotating fan. This essentially enables the microcontroller to perform ML inference locally ("edge AI").

The presented work is a proof of the concepts of hardware interfacing, embedded C programming, serial communication, data acquisition, ML model training, and deployment on an embedded platform.

1.1 Problem Statement

Rotating machines as well as electromechanical systems like ceiling fans, motors, pumps, and industrial equipment usually have typical vibration patterns when they are operating normally. When mechanical wear, imbalance, loose components, misalignment, or external disturbances cause the faults, the vibration signatures change drastically. Detecting such abnormal behavior has been done by periodic manual inspection, which is slow, inconsistent, and susceptible to human error.

Most of the existing automated vibration monitoring methods can only detect through fixed threshold-based systems or require expensive industrial-grade sensors and cloud-based processing.

Threshold systems do not have the ability to adjust to machine behavior, noise, or environmental changes. In the same way, cloud-based analytics increase the complexity, cost, and latency of the system, thus making them unsuitable for small-scale or embedded applications.

The problem to be solved is the invention of a small, cheap, and smart device for detecting anomalies that can be locally performed on an embedded microcontroller with limited computational resources.

The device should be able to:

- Continuously get 6-axis motion data from the MPU6050 sensor
- Correctly record and store normal and abnormal vibration patterns

- Without intervention, the device learns the normal operational behavior of a machine
- Identifying the deviations or anomalies instantaneously
- Working without the need for cloud connectivity or high-end processors

Moreover, the motion data collected need to be in a certain format and processed to be able to train a machine-learning model that can tell apart normal and abnormal vibration signatures. A trained model is then transferred to a resource-constrained STM32 microcontroller, which is capable of anomaly detection locally ("on the edge") with a short response time.

1.2 Objective

The primary goals of this project are:

- Implementing the connection of an MPU6050 sensor with STM32 microcontroller through the I2C protocol.
- Creating a real-time motion data logger that can be accelerometer and gyroscope data transmitted over UART.
- Recording the vibrational patterns of a machine during both its normal and abnormal operations.
- Formulating an Anomaly Detection ML model with the help of Nano Edge AI Studio.
- Uploading the ML library created by STM32 for on-device anomaly detection.
- Using similarity scores and vibration behaviour to assess the performance of the system.

CHAPTER 2: SYSTEM DESCRIPTION

The system integrates hardware sensing, embedded firmware, and machine learning deployment to enable real-time motion anomaly detection. It is composed of three major parts: the hardware architecture, the software architecture, and the machine learning workflow. Each part works together to capture motion data, process it, and identify abnormal vibration patterns

2.1 Hardware Architecture

The hardware setup includes an MPU6050 motion sensor connected to an STM32 Nucleo microcontroller board. The MPU6050 is a 6 DOF IMU that outputs 3-axis accelerometer, which are required for vibration and motion analysis. I2C is used for communication between the STM32 and the sensor, where PB8 (SCL) and PB9 (SDA) are generally used pins. The STM32 board is powered by a USB cable, can be programmed through the IDE, and is connected to the PC by a serial communication channel for data logging.

The MPU6050 sensor should be attached to a rotating fan or any vibrating surface to record the correct movement patterns for both the normal and abnormal situations. The right positioning of the sensor not only guarantees stable but also that interesting data acquisition.

2.2 Software Architecture

Firmware is developed by the help of STM32CubeIDE which besides other things is also a tool for project configuration and peripheral initialization. The I2C peripheral is set up to interact with the MPU6050, and the UART peripheral is enabled for serial data transmission.

The working cycle first includes the configuration of the MPU6050 registers, sensor activation and then setting the measurement ranges. The microcontroller is reading raw data from the accelerometer and gyroscope and storing them in the fixed sampling rate. After formatting the data, the microcontroller sends these values to the PC in a space-separated format suitable for Nano Edge AI Studio.

After that, a machine-learning library is created, then the library is combined with the STM32 firmware. The ML inference function which receives the real-time sensor data evaluates the data and returns a similarity score, which gives an idea of whether the motion is normal or abnormal. The result is then printed through UART for the user's convenience.

CHAPTER 3: ALGORITHM DESCRIPTION

3.2 Machine Learning Workflow

Nano Edge AI Studio is the tool to create a very compact anomaly-detection model. The tool gets the motion data from the STM32 microcontroller and that data is categorized into two groups: Normal Vibration and Abnormal Vibration. The platform runs a Benchmark process to test multiple algorithms and selects the best one based on accuracy, memory, and processing capability.

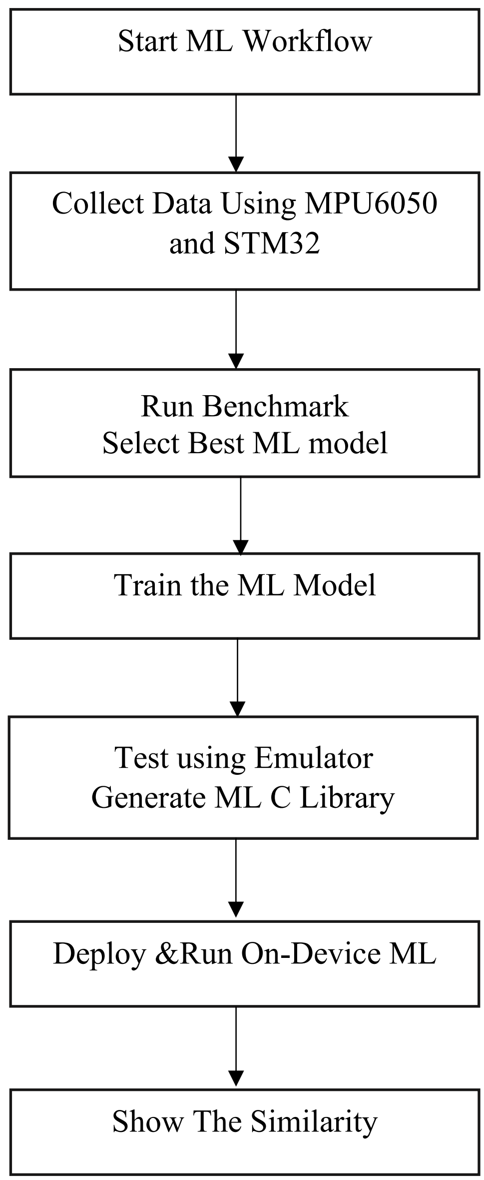

Fig 1: Machine Learning Flowchart

After the perfect model is chosen, Nano Edge AI Studio creates a compact C library that can be easily integrated into the STM32 microcontroller for learning and inference tasks. The ML engine initially gets acquainted as shown in Figure 1with the normal vibration features and later it compares the new samples with it to find the deviations. The higher the similarity score, the more normal the behavior, while a lower score shows that an anomaly has occurred.

3.2 System Operation

During operation, the STM32 continuously reads motion data from the MPU6050 and processes it through the embedded ML model. Each new sample is analyzed in real time, producing a similarity score that reflects the current state of motion.
The system outputs these results via UART, allowing real-time observation of normal and abnormal vibration behavior.

The fully integrated process—ranging from sensor acquisition to ML inference—runs locally on the microcontroller, enabling fast and efficient edge-based anomaly detection without relying on cloud services or external computation.

CHAPTER 4: RESULT AND INFERENCE

Various aspects of the motion data logging process and the anomaly detection model, such as accuracy, reliability, and performance, were tested through the implemented system. The team made observations during data collection, model testing in the Nano Edge AI Studio, as well as real-time deployment on the STM32 microcontroller.

4.1 Data Logger Result

The MPU6050 sensor successfully delivered continuous 6-axis motion data to the STM32 microcontroller through the I2C interface. The UART transmission produced stable and clean numerical outputs in the required space-separated format:

The sensor values clearly changed based on the fan's rotational behavior and reacted accurately to disturbances or external vibrations. The data was recorded without missing samples, ensuring a reliable dataset for machine learning.

4.2 Machine Learning Model Result

4.2.1 Benchmark Accuracy

Benchmark Accuracy is the performance level that is shown when Nano Edge AI Studio runs a variety of machine-learning algorithms automatically to pick the best anomaly-detection model from the given vibration data.

In short, after importing the normal and abnormal samples, the tool automatically tests hundreds of algorithm combinations, each with different internal parameters, preprocessing techniques, and similarity measurement methods.

As part of this process, each algorithm is trained on normal vibration samples and then tested with both normal and abnormal datasets. The system evaluates how good the model is in differentiating the two categories by using the similarity score. A score that is very close to 100% means that the input pattern is very similar to the learned normal behavior, while a lower score means a deviation or anomaly.

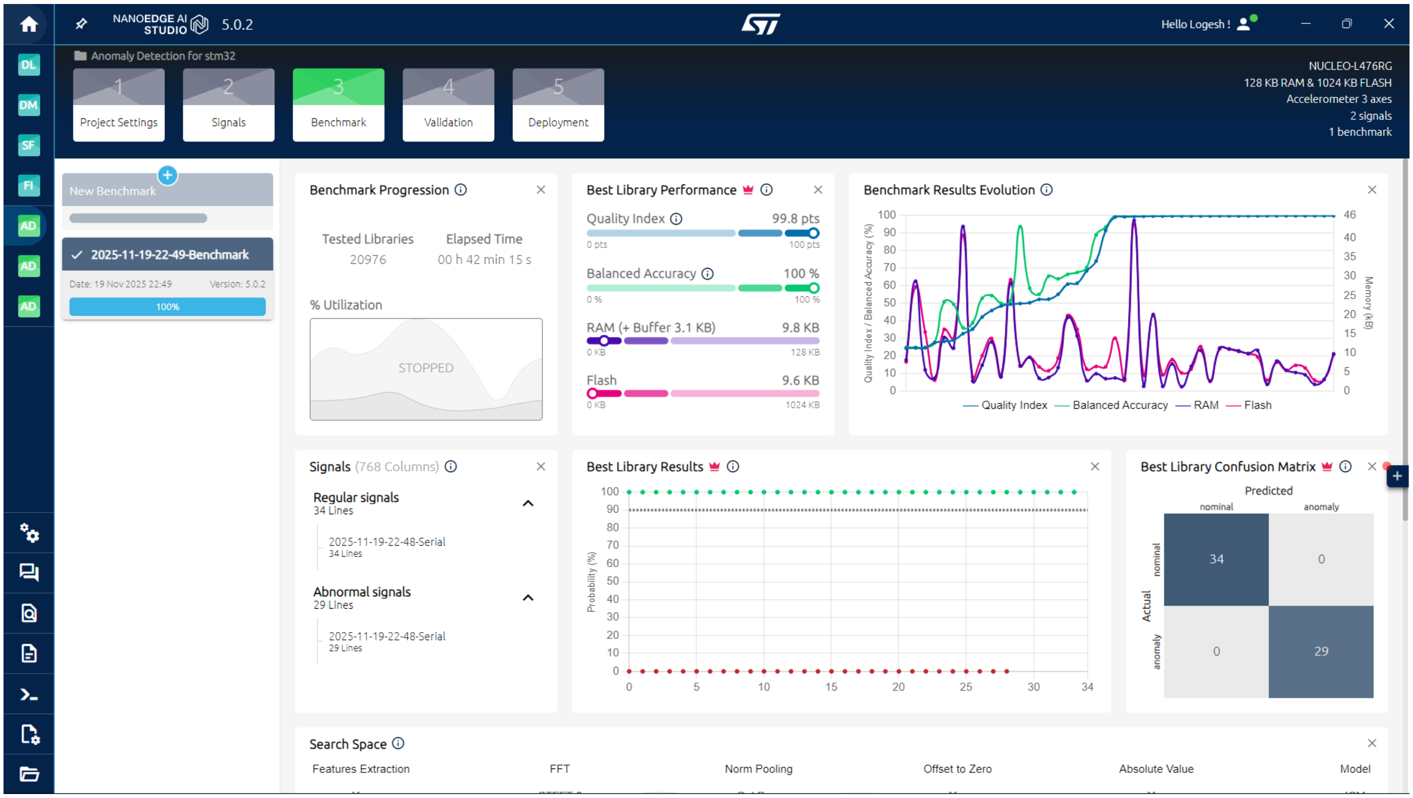

Fig 2 :Benchmark Result for the data in NANO EDGE AI

In this system, the Benchmark operation led to an accuracy of 100% as shown in Figure 2 which means that the selected model had strong discriminative capability and was able to identify abnormal vibration patterns accurately and reliably. Such a high level of accuracy is an indication that the dataset was stable and the algorithm chosen is the right one for real-time anomaly detection when running on the STM32 microcontroller.

4.2.2 Emulator Testing

During emulator testing:

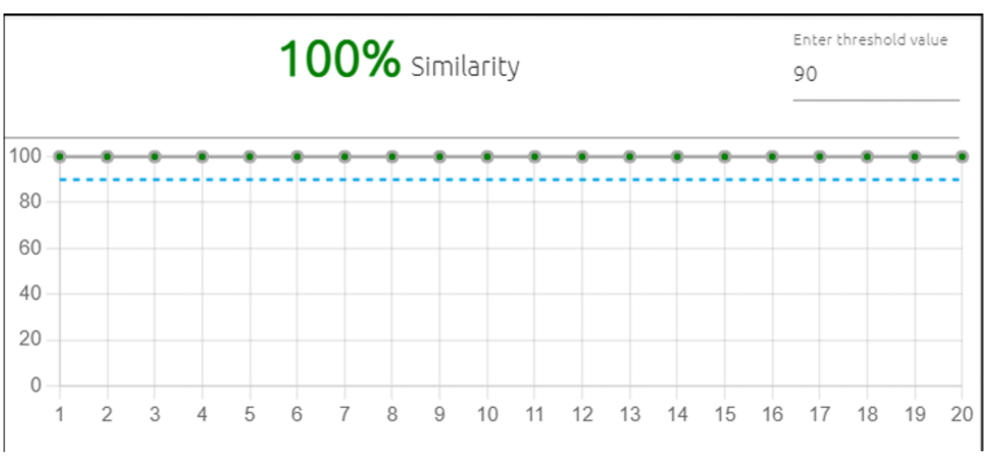

Fig 3 : Normal vibration data in Emulator Similarity Result

- Normal vibration data produced similarity scores close to 95–100%.

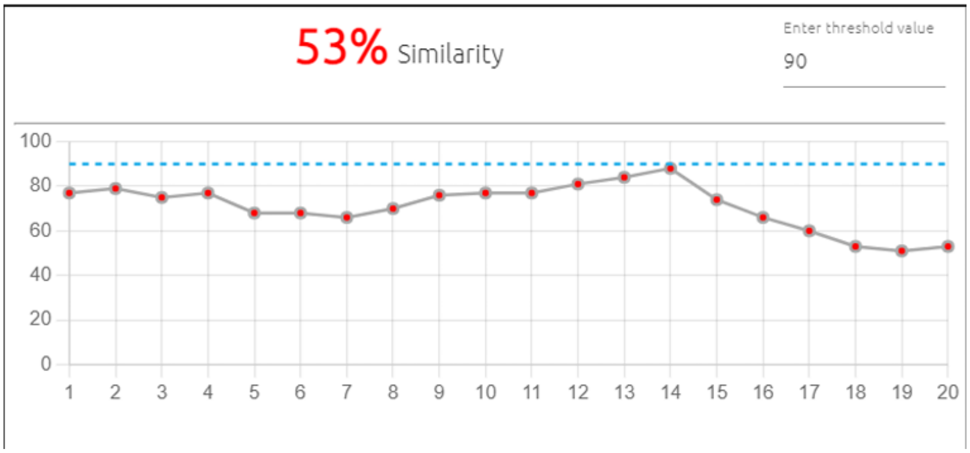

Fig 4 : Abnormal vibration data in Emulator Similarity Result

- Disturbed or abnormal vibration data resulted in a noticeable drop in similarity scores, typically falling between 0–60%, depending on severity.

This confirmed that the selected anomaly detection model could reliably differentiate between normal and abnormal mechanical conditions.

4.3 Real-Time Detection Results on STM32

After integrating the trained ML library into the STM32 firmware, real-time inference was performed directly on the microcontroller:

- Under normal fan operation, the similarity output remained consistently high and stable.
- When slight disturbances such as touching or blocking the fan blade were introduced, the similarity value dropped sharply, triggering an anomaly indication instantly.
- The inference response time was extremely low (milliseconds), confirming the efficiency of the edge-based ML approach.

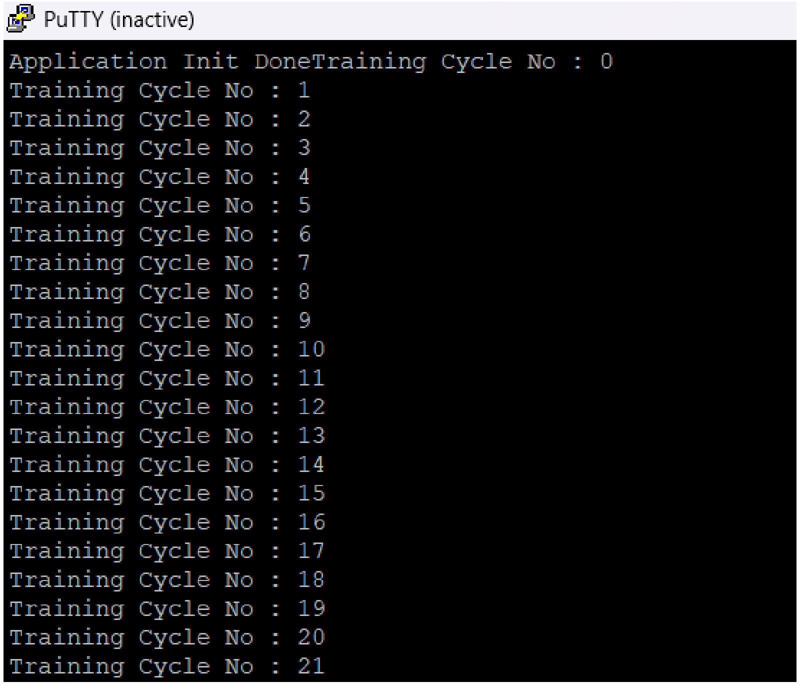

Fig 5 : Training Cycle of the STM32 Controller

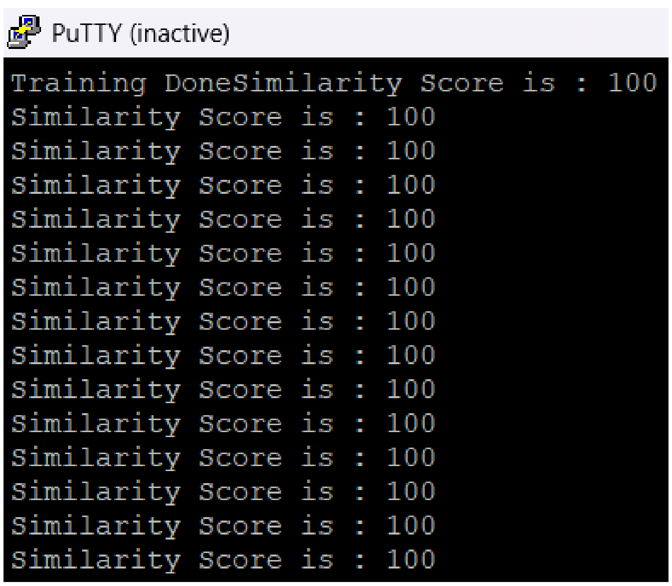

Fig 6 : Similarity Score for Normal input

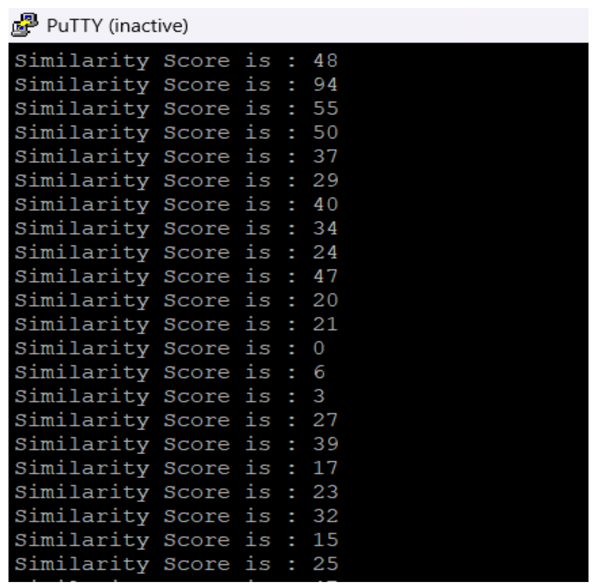

Fig 7: Similarity Score for Abnormal input

The real-time anomaly detection behavior closely matched the emulator's performance, validating the accuracy of the deployed model.

4.4 Inference

From the results that have been observed, we can infer the following:

- The MPU6050 is a source of good quality and consistent vibration data that can be used in machine learning.
- Nano Edge AI model has efficiently acquired the normal pattern of the rotating system.
- The similarity score can be used as a powerful tool to tell apart normal movement from abnormal ones.
- STM32 edge inference is quick, minimal in terms of resources, and does not need cloud support.
- It is possible to detect anomalies in real-time with a high degree of accuracy using a cheap sensor and microcontroller.
- The entire hardware-software solution is a great tool for predictive maintenance, vibration monitoring, and fault detection in the case of rotating machinery.

CONCLUSION

The integration of motion sensing, embedded programming, and machine learning to local real-time detection of vibration anomalies on a resource-constrained microcontroller was effectively demonstrated by the developed system. The MPU6050 sensor delivered precise 6-axis motion data, and the STM32 microcontroller was able to perform data acquisition, transmission, and ML inference very efficiently.

The recording of data representing normal and abnormal conditions of the machine made it possible to establish a trustworthy model for anomaly detection in Nano Edge AI Studio. The Benchmark operation resulted in a very high accuracy score which means that a strong separation of normal and disturbed vibration patterns was achieved. After deployment, the machine-learning library was performing the real-time analysis directly on the STM32 thus, it was very fast and the similarity scores were stable which is a correct identification of the abnormal behaviour.

Overall, the system serves as evidence that machine learning at the edge is possible with low-cost hardware. Such a device that is capable of learning and recognizing anomalies even without cloud connection is an ideal tool for predictive maintenance, equipment monitoring, and a vast number of other on-demand real-time embedded applications. Next, one might think of implementing features like wireless data transmission, SD card logging, FreeRTOS integration, or multi-sensor environments extensions for the model.

REFERENCES

[1] MPU6050 Datasheet – InvenSense

[2] STM32 Nucleo Board User Manual – STMicroelectronics

[3] Nano Edge AI Studio Documentation – STMicroelectronics

[4] STM32CubeIDE Reference Guide