

# Projeto FanGuardian

## Monitoramento Inteligente de Ventiladores com TinyML

Artur Macedo Costa - 2024012996

### 1 Introdução

Este projeto apresenta o desenvolvimento do sistema **FanGuardian**, voltado ao monitoramento inteligente de ventiladores por meio de técnicas de TinyML embarcado. O objetivo é detectar comportamentos anômalos e indicar estados operacionais como: funcionamento normal, parada ou sobrevelocidade. Embora tenham sido utilizados ventiladores DC de 120mm apenas para fins experimentais, o projeto visa aplicações mais amplas, como ambientes industriais e datacenters.

### 2 Aquisição e Preparação dos Dados

Foram utilizadas três classes principais: **Parado**, **Normal** (velocidade média), e **Overspeed** (velocidade máxima). Os dados foram capturados com um sensor eletromagnético integrado ao Arduino Nano 33 BLE Sense, incluindo acelerações (accX, accY, accZ) e componentes magnéticas (magX, magY, magZ).

Cada classe foi gravada por aproximadamente 2 minutos e 30 segundos no conjunto de treinamento, e 40 segundos no conjunto de teste. Para resolver um problema na detecção da classe **Parado**, em que o campo magnético residual do eixo gerava falsos positivos como anomalia, foi realizada uma coleta adicional de 1 minuto nesta condição. Assim, o total de dados coletados foi de 14 minutos e 1 segundo, conforme apresentados na Figura 1.

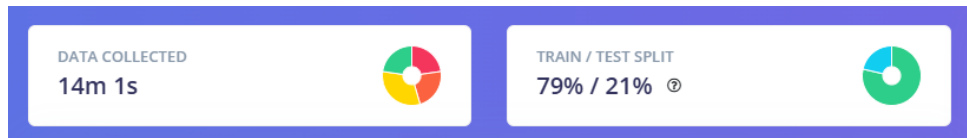


Figura 1: Distribuição das classes no dataset (14 minutos no total)

### Pré-processamento e Extração de Características

Foi aplicada uma filtragem do tipo low-pass com frequência de corte de 43.75Hz e ordem 6, além de análise via Wavelet (bior3.1) com decomposição de nível 1. As features extraídas foram baseadas no valor RMS dos sinais capturados, parâmetros que foram obtidos a partir da utilização do autotune (Figura 2).

Parameters

Autotune parameters

Filter

Scale axes ?

0.015050523608304243

Input decimation ratio ?

1

▼

Type ?

low

▼

Cut-off frequency ?

43.75

Order ?

6

Analysis

Type ?

Wavelet

▼

Wavelet Decomposition Level ?

1

Wavelet ?

bior3.1

▼

Save parameters

▼

Figura 2: Parâmetros de pré-processamento e análise espectral

A separação entre as classes também pôde ser observada de forma visual por meio do *Feature Explorer*, que revela uma organização consistente das amostras no espaço de características. Nota-se, por exemplo, a presença de três grupos distintos dentro da classe **Parado**, devido à coleta adicional realizada para corrigir os falsos positivos mencionados anteriormente.



Figura 3: Exploração de características extraídas por classe

### 3 Treinamento e Classificação

O modelo foi treinado com uma arquitetura de rede neural com duas camadas densas (20 e 10 neurônios) e uma camada de saída com 3 classes. Utilizou-se um total de 30 ciclos de treinamento e taxa de aprendizado de 0.0005. O classificador atingiu **100% de acurácia**, **0.01 de perda**, e F1-score igual a 1.0 em todas as classes (Figura 3).



Figura 4: Resultados do treinamento do modelo

## 4 Detecção de Anomalias

Foi utilizado o modelo K-means para detecção de anomalias baseado nos sinais L0-RMS accX, accY, accZ, magX, magY e magZ. As regiões de densidade formadas no Anomaly Explorer indicam claramente a concentração de operação normal, sendo quaisquer desvios apontados como anomalias (Figura 5).

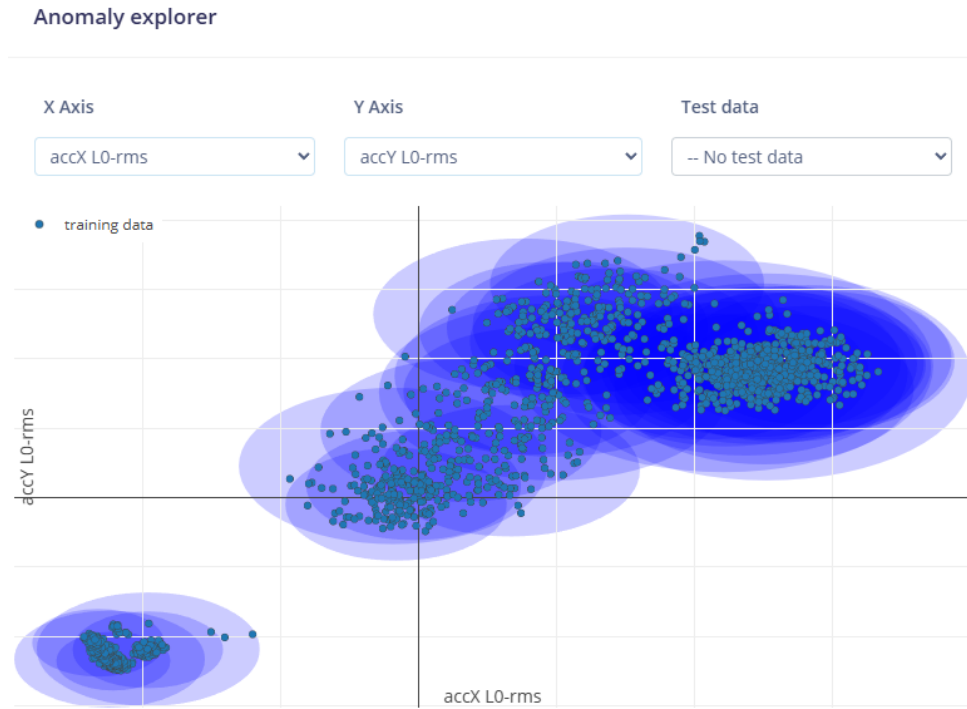


Figura 5: Distribuição de densidade para detecção de anomalias

## 5 Validação do Modelo e Testes

Após o treinamento, o modelo foi testado com sucesso em dados novos. A matriz de confusão mostrou 100% de acurácia para todas as classes, sem indícios de overfitting. A execução em tempo real foi analisada com a versão não quantizada (float32), mantendo a precisão, conforme apresentado na Figura 6.



Figura 6: Resultados do teste do modelo

## 6 Funcionamento Prático da Experiência

O sistema foi implementado em um Arduino Nano 33 BLE Sense, com saída serial monitorada via Arduino IDE. A cada inferência (realizada a cada 2 segundos), o estado atual do ventilador é exibido no terminal: **NORMAL**, **OVERSPEED**, **PARADO**. Além disso, um mecanismo de detecção de anomalias baseado no *anomaly score* também é incorporado: caso o valor ultrapasse 6.5, um LED vermelho na placa é acionado automaticamente, indicando a presença de uma possível falha.

Para simular anomalias reais, foi realizado um teste tocando com o dedo a hélice do ventilador em movimento, provocando uma interferência no campo eletromagnético e também no acelerômetro, resultando em pontuação de anomalia acima do limite.

## 7 Conclusão

O projeto FanGuardian demonstrou eficácia na classificação de estados operacionais e detecção de anomalias em ventiladores. Embora a aplicação tenha sido experimental com fans DC de 120mm, o sistema é pensado para uso mais amplo, como em ambientes industriais ou datacenters, necessitando apenas de uma nova captura de dados do dispositivo a ser utilizado. A utilização de ventiladores AC pode ainda favorecer a análise, devido à maior variação no campo magnético gerado.

### Acesso ao Projeto no Edge Impulse

O projeto completo, incluindo dados, configurações de pré-processamento, arquitetura do modelo e resultados de treinamento/teste, está disponível publicamente na plataforma Edge Impulse. Pode ser acessado através do seguinte link:

<https://studio.edgeimpulse.com/public/732802/live>