



UNIVERSIDADE FEDERAL DE ITAJUBÁ
INSTITUTO DE ENGENHARIA MECÂNICA
Engenharia Mecânica Aeronáutica

TRABALHO DE CONCLUSÃO DE CURSO
- MONOGRAFIA FINAL -

CLASSIFICAÇÃO DE FALHAS EM MOTORES DIESEL A PARTIR DE
RESPOSTAS NO DOMÍNIO DO TEMPO POR MEIO DE
PROCESSAMENTO DE SINAIS E REDE NEURAL CONVOLUCIONAL

Aluno: Gabriel Hasmann Freire Moraes

Orientador: Prof. Dr. Guilherme Gomes Ferreira

Coorientador: M. Sc. Ronny Francis Ribeiro Junior

Itajubá – MG

2023

Agradecimentos

Este trabalho não poderia ser terminado sem a ajuda de diversas pessoas às quais presto minha homenagem:

Aos meus pais, Alexandre e Alyne, a minha tia Cibele, a todos os meus avós, e a toda minha família, os quais sempre me apoiaram e incentivaram-me durante toda minha jornada acadêmica.

Ao meu orientador Prof. Dr. Guilherme Gomes Ferreira, e ao meu coorientador, M.Sc. Ronny Junior, os quais dedicaram tempo e atenção ao meu trabalho, proporcionando muito aprendizado.

A todos os professores de Engenharia Mecânica Aeronáutica da Universidade Federal de Itajubá, que se dedicam e somam cada vez mais excelência ao curso.

A minha namorada e amiga Mikaelle, aos meus amigos de curso com os quais compartilhei bons momentos durante o ensino superior, aos meus grandes amigos de pedal Leo e Lucas, aos meus inúmeros amigos de Eng. Elétrica e agregados, e aos meus amigos de república.

*“Quanto mais aumenta nosso conhecimento,
mais evidente fica nossa ignorância”.*

John F. Kennedy

Resumo

Em um mundo cada vez mais interconectado e em busca da maior eficiência, é vital que em uma indústria moderna seja possível ter maior previsibilidade sobre o estado de funcionamento dos equipamentos. Nesse contexto, a detecção precoce de falhas em motores a diesel, muito usados em incontáveis ambientes industriais, é essencial como uma forma de reduzir custos operacionais, manter a segurança do ambiente de trabalho, evitar paradas não programadas e prolongar a vida útil dos equipamentos. Existem, hoje, inúmeras formas de detecção de anomalias baseadas em diversos aspectos, como termometria, indicadores de desgaste, análise de fluidos lubrificantes, ensaios por ultrassom, radiografia, entre outros. Esses métodos em geral funcionam bem, mas normalmente necessitam de grande capacitação para serem feitos corretamente, exigem uma parada completa da máquina (gerando perda de produtividade) e alguns possuem alto custo de uso e de acessibilidade. Diante disso, este estudo propõe uma abordagem exploratória na detecção de falhas nesses tipos de motores utilizando dados no domínio do tempo de vibração torcional. Esses dados, obtidos por simulações computacionais, são tratados pelas técnicas de Transformada de Fourier de Curto Termo e Transformadas Wavelet e em seguida incorporados em uma rede neural convolucional (em inglês, CNN). Esta então realiza a extração de padrões ocultos e realiza a identificação das condições de operação do motor. Utilizando estas técnicas, foi possível chegar a uma acurácia de 96,5% utilizando a STFT e a 92,2% utilizando a CWT. Para avaliar a robustez da rede e simular condições reais de operação, aplica-se diferentes níveis de ruído aos dados, alcançando-se, nesses casos, acurácias acima de 70% em ambos os casos para níveis de ruído de até 40%.

Palavras-chave: detecção de falhas; STFT; wavelet; rede neural convolucional; motores a diesel.

Abstract

In an increasingly interconnected world and in the pursuit of greater efficiency, it is vital that in a modern industry, there is the ability to have greater predictability regarding the operational status of equipment. In this context, the early detection of faults in diesel engines, widely used in countless industrial environments, is essential as a means to reduce operational costs, maintain workplace safety, avoid unplanned downtime, and extend the lifespan of equipment. Today, there are numerous methods for anomaly detection based on various aspects such as thermometry, wear indicators, lubricating fluid analysis, ultrasound testing, radiography, among others. While these methods generally work well, they often require significant expertise to be performed correctly, demand a complete machine shutdown (resulting in productivity loss), and some have high usage and accessibility costs. In response to these challenges, this study proposes an exploratory approach to detecting faults in these types of engines using data in the time domain of torsional vibration. The data, obtained through computer simulations, are processed using Short-Term Fourier Transform and Continuous Wavelet Transform techniques and then incorporated into a Convolutional Neural Network (CNN). This network extracts hidden patterns and identifies the operating conditions of the engine. Using these techniques, it was possible to achieve an accuracy of 96.5% using STFT and 92.2% using CWT. To assess the robustness of the network and simulate real operating conditions, distinct levels of noise are applied to the data, resulting in accuracies above 70% in both cases for noise levels up to 40%.

Key words: Fault detection; STFT; wavelet; Convolutional Neural Network; diesel engines.

Lista de Figuras

Figura 2.1 Ilustração da translação da janela do tipo Gaussiana utilizada na transformada STFT.....	9
Figura 2.2 – Estrutura típica de uma rede neural convolucional	11
Figura 2.3 - Exemplo de <i>kernel</i> (matriz ao meio) aplicado à extração das características de uma imagem de um ônibus em uma rede 2D. Nota-se na saída o destaque para as regiões mais claras e contrastantes.	12
Figura 2.4 - Funções ReLU e LeakyReLU (com $\alpha = 0.1$ para fins de ilustração) (Fonte: autoria própria).	14
Figura 2.5 - Exemplo de <i>max pooling</i> . Nota-se a grande redução de dimensionalidade entre a entrada e a saída.....	15
Figura 2.6 - Arquitetura de uma rede neural convolucional. Em destaque, a camada FC e a camada de saída.	16
Figura 3.1 - Volume de controle considerado no modelo termodinâmico.	19
Figura 3.2 – Pressão em um dos cilindros do modelo termodinâmico simulado versus dados experimentais para diferentes RPM).	21
Figura 3.3 - Dados de torque e potência reais e simulados (calculados) para diferentes RPM.	22
Figura 3.4 - Curvas de pressão para 2500 RPM.	22
Figura 3.5 - Curvas de torque para 2500 RPM.....	23
Figura 3.6 - Sinais de vibração para cada condição de operação.	25
Figura 3.7 - STFT: (a) Condição normal, (b) redução de pressão no <i>manifold</i> , (c) redução da razão de compressão nos cilindros e (d) redução na quantidade de combustível nos cilindros.	26
Figura 3.8 - CWT: (a) Condição normal, (b) redução de pressão no manifold, (c) redução da razão de compressão nos cilindros e (d) redução na quantidade de combustível nos cilindros	26
Figura 3.9 - Sinais originais e sinais acrescentados de ruídos.....	28
Figura 3.10 - Exemplos de espectrograma com ruído, condição normal: (a) 0%, (b) 10%, (c) 20%, (d) 40%, (e) 60% e (f) 80%.	28
Figura 3.11 - Exemplos de escalograma com ruído, condição normal: (a) 0%, (b) 10%, (c) 20%, (d) 40%, (e) 60% e (f) 80%.	29

Figura 3.12 - Ilustração da estrutura e dimensões da rede neural utilizada.....	31
Figura 4.1 – Matriz com imagens para transformada STFT para cada condição e cada nível de ruído.....	33
Figura 4.2 – Conjunto de matriz de confusão para STFT para cada caso de ruído.....	34
Figura 4.3 - Matriz com imagens para transformada Wavelet (CWT) para cada condição e cada nível de ruído.....	36
Figura 4.4 - Matrizes de confusão para STFT para cada caso de ruído	37
Figura 4.5 – Condição normal; STFT acima, CWT abaixo.....	37
Figura 4.6 – Redução de pressão no <i>manifold</i> ; STFT acima, CWT abaixo.....	38
Figura 4.7 - Redução mássica de combustível; STFT acima, CWT abaixo.....	38
Figura 4.8 – Redução da razão de compressão em um cilindro; STFT acima, CWT abaixo...	39

Lista de Tabelas

Tabela 1 - Tabela de especificações do motor usado como base para simulação dos dados. ..	18
Tabela 2 - Acurácia nos testes para cada transformada nos níveis de ruído estudados.....	39

Lista de Siglas e Abreviaturas

AI – *Artificial Intelligence*

ANN – *Artificial Neural Network*

CNN – *Convolutional Neural Network*

CWT – *Continuous Wavelet Transform*

DL – *Deep Learning*

ENN – *Extension neural network*

FC – *Fully connected*

FFT – *Fast Fourier Transform*

ICMS – *Ionic Current Monitoring System*

MLP – *Multi-Layer Perceptron*

RPM – *Rotações por Minuto*

STFT – *Short-time Fourier Transform*

SVC – *Spark Voltage Characterisation*

SVM – *Support Vector Machines*

WPD – *Wavelet Packet Decomposition*

SUMÁRIO

CAPÍTULO 1 – INTRODUÇÃO	1
1.1 Introdução	1
1.2 Justificativa	2
1.3 Objetivos	3
1.3.1 Objetivo geral	3
1.3.2 Objetivos específicos	3
1.4 Estrutura do Trabalho	4
1.5 Revisão da Literatura	4
CAPÍTULO 2 – EMBASAMENTO TEÓRICO	8
2.1 STFT	8
2.2 Transformada Wavelet Contínua (CWT)	9
2.3 Rede Neural Convolucional	11
2.3.1 Camada de entrada	11
2.3.2 Camada de convolução	12
2.3.3 Funções de ativação	13
2.3.4 Pooling	15
2.3.5 <i>Fully-connected</i> (FC) e classificação	15
CAPÍTULO 3 – METODOLOGIA	17
3.1 Modelagem do problema	17
3.1.1 Especificações do motor	17
3.1.2 Descrição do modelo simulado do motor	18
3.1.3 Descrição dos dados	23
3.2 Modelagem numérica	24
3.3 Formulação da Rede Neural Convolucional	29
CAPÍTULO 4 – RESULTADOS E DISCUSSÃO	32
4.1 Rede CNN-STFT	32
4.2 Rede CNN-CWT	35
4.3 Comparação dos resultados	37
CAPÍTULO 5 - CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS	41
5.1 Conclusões	41

5.2 Sugestões para Trabalhos Futuros	42
--	----

CAPÍTULO 1 – INTRODUÇÃO

1.1 Introdução

Mesmo com o advento da cultura de eletrificação, os motores à combustão possuem, até hoje, uma importância imensurável no desenvolvimento industrial, econômico e social mundial. São encontrados em diversos níveis, como transporte público, máquinas industriais, geradores elétricos em hospitais como backup em quedas de energia etc.

Dada sua importância, seu correto funcionamento é essencial e deve-se garantir a disponibilidade e segurança do sistema onde este está instalado. Nesse contexto, é essencial que qualquer condição anormal de funcionamento seja identificada o quanto antes para evitar que se torne uma falha grave e gere outros problemas maiores, como má performance ou quebra do equipamento, o que significa um risco tanto humano quanto operacional. Vários métodos já existem para identificar tais condições, desde procedimentos simples de ensaios não destrutivos, como líquido penetrante e partículas magnéticas, até métodos mais complexos, como ultrassom, radiografia e termografia (STERKENBURG, 2019). Entretanto, esses métodos em geral precisam de alto conhecimento prévio em leitura de sinais e compreensão técnica profunda, tornando-os métodos caros e as vezes de difícil acesso.

Assim, novas formas utilizando inteligência artificial têm surgido e demonstrando ótima performance em diversas aplicações. Em geral, até alguns anos atrás, métodos como SVM (*support vector machines*) e redes bayesianas (*bayesian networks*, BN) surgiram como classificadores com boa performance e baixo poder computacional (KIRANYAZ *et al.*, 2021).

Foi somente com a introdução da rede AlexNET que métodos como redes neurais convolucionais (*convolutional neural networks*, CNN) passaram a ser mais usados, caminhando paralelamente aos avanços em poder computacional. Esse tipo de *Deep Learning* (DL) se tornou excelente para realizar a detecção de padrões em inúmeros campos, incluindo classificação e detecção de falhas em motores. Esse tipo de aplicação foi demonstrada, por exemplo, em motores elétricos (RIBEIRO JUNIOR *et al.*, 2022), em motores aeronáuticos

(ZHU *et al.*, 2019) e também motores a combustão (PINEDO-SÁNCHEZ; MERCADO-RAVELL; CARBALLO-MONSIVAIS, 2020).

Juntamente com o uso de CNN, frequentemente utiliza-se métodos de pré-processamento de sinais utilizando ferramentas como a Transformada de Fourier de Curto Período (STFT) ou a Transformada Wavelet Contínua (CWT) (AL-BADOUR; SUNAR; CHEDED, 2011; JING *et al.*, 2017; WU; CHEN, 2006). Eles são usados para realizar melhor a extração de características as quais se deseja enaltecer para os problemas estudados. Em geral, essas técnicas se demonstraram eficientes em classificação de falhas, inclusive em motores a combustão.

Desse modo, o presente trabalho apresenta uma metodologia para identificar e classificar alguns tipos de falhas de um motor a diesel marítimo a partir de dados simulados de vibração torcional no virabrequim. Para isso estes dados são inicialmente processados a partir de técnicas de análise de sinais no domínio do tempo-frequência, com o uso das transformadas de Fourier de Curto Tempo e Wavelet Contínua. Em seguida, os dados processados são inseridos em uma Rede Neural de Convolução, que ao final tem como saída o tipo de falha do caso avaliado. Analisa-se também o efeito da adição de ruído no sinal sobre o desempenho de classificação da rede proposta. Espera-se que a metodologia apresentada demonstre boa acurácia na classificação de falhas e permita abrir caminho para o teste e uso em situações reais, sendo ao mesmo tempo uma solução com baixa demanda de poder computacional e boa velocidade de execução para rápida análise dos dados quando comparada aos métodos utilizados atualmente.

1.2 Justificativa

Num panorama industrial caracterizado pela crescente interconexão e busca por eficiência, a antecipação de falhas em motores a diesel torna-se um componente crítico para garantir a continuidade operacional e a eficácia dos processos. A utilização generalizada desses motores em diversos setores industriais ressalta a importância estratégica de desenvolver métodos inovadores para a detecção precoce de anomalias. A proposta deste trabalho de empregar dados simulados de aceleração rotacional, transformados por meio de técnicas como STFT e Wavelet, para alimentar uma rede neural convolucional (CNN) surge como uma resposta a essa demanda.

A escolha de dados simulados se fundamenta na dificuldade de acesso a conjuntos de dados reais abrangentes e na necessidade de criar ambientes controlados para validação. Parte

dessa dificuldade se reside no fato de que simulações de falhas em motores a diesel podem ser mais custosas e colocam em risco a integridade do motor. O uso de dados simulados e validados com o fabricante permite investigar a viabilidade do modelo sem precisar realizar grandes investimentos. A aplicação dessas transformações e a utilização de uma CNN visam superar as limitações dos métodos tradicionais de diagnóstico, oferecendo uma abordagem mais rápida, simples, economicamente viável e eficaz na identificação de falhas em motores a diesel.

Além disso, dada a grande variedade e complexidade de modelos de motores existentes hoje, o procedimento descrito visa desenvolver um método de reconhecimento de falhas global, independentemente da tecnologia embarcada e proprietária de cada equipamento. Dessa forma, esta pesquisa se justifica pela contribuição para a eficiência operacional, segurança e sustentabilidade das indústrias, propondo uma solução inovadora que integra avanços em processamento de sinais e aprendizado de máquina para aprimorar a confiabilidade dos motores em ambientes industriais críticos.

1.3 Objetivos

1.3.1 Objetivo geral

Este trabalho tem como objetivo utilizar redes CNN em conjunto com as transformadas de sinais STFT e wavelet para identificar condições anormais de funcionamento (severidades) de motores a combustão por meio de sua resposta de vibração.

1.3.2 Objetivos específicos

Para atingir o objetivo geral deste trabalho, será necessário:

- Avaliar o desempenho de duas técnicas de processamento de sinal, sendo elas: transformada wavelet contínua (CWT) e transformada de Fourier de curto tempo (STFT);
- Desenvolver um *pipeline* (leia-se metodologia) de análise dos dados em função do modelo de IA desenvolvido para uso prático com diferentes configurações;
- Avaliar a performance dos modelos aplicando-se ruído em diferentes níveis nos dados.

1.4 Estrutura do Trabalho

No Capítulo 1 estão apresentados a introdução do trabalho, a justificativa e os objetivos dele, além de apresentar como a monografia foi estruturada e uma breve revisão da literatura apresentando os principais trabalhos que serviram de base. Já no Capítulo 2, é exposto o embasamento teórico que rege o trabalho, como explicações das transformadas abordadas e dos conceitos bases para redes neurais convolucionais. O Capítulo 3 apresenta a metodologia utilizada para desenvolver o algoritmo de identificação de falhas em motores a diesel. Os resultados alcançados utilizando essa metodologia são exibidos e discutidos no Capítulo 4. Por fim, o Capítulo 5 apresenta conclusões sobre o trabalho e apresenta sugestões para trabalhos futuros.

1.5 Revisão da Literatura

Dada a extrema importância dos motores à diesel em diversas indústrias, diversos estudos foram desenvolvidos nos últimos anos a fim de se investigar métodos de diagnóstico de falhas em motores a combustão e máquinas relacionadas utilizando diversas abordagens.

Howlett et al. (1999) propõem um método de análise de falhas em motores à gasolina utilizando a vela de ignição como o sensor de combustão. Essa técnica, chamada *Spark Voltage Characterisation* (SVC), é baseada em outra chamada *Ionic Current Monitoring System* (ICMS) porém elimina fontes de tensão externas e outros componentes extras. Nesse método, analisa-se o formato de onda da tensão necessária aplicada na vela durante o processo de ignição e investiga-se a eficiência de redes neurais multicamadas de perceptrons (*Multi-Layer Perceptron*, MLP) em classificar com uma precisão de cerca de 95% os tipos de falhas e anomalias no processo de combustão.

Al-Badour, Sunar e Cheded (2011) estudam a eficiência do uso de transformadas *wavelets*, em particular a transformada *wavelet* contínua e a transformada por *wavelet packet*, na detecção de falhas em regimes transientes em máquinas rotativas e compara-se com a análise por FFT e STFT. As falhas, sendo fricção periódica de rotores na carcaça causada por desbalanceamento e desalinhamento dos eixos, são analisadas em condições de aceleração/inicialização rápida e em desaceleração (sem carga aplicada pelo motor). Os sinais vibracionais principais são obtidos utilizando sensores de proximidade e extensômetros, mas utiliza-se acelerômetros para comparação. Com esses métodos, mostra-se que a análise por

wavelet se mostra muito superior à por Fourier em regimes transitórios. Especificamente, a WPT consegue gerar uma análise espectral melhor que a CWT.

Utilizando dados sonoros captados de um motor a combustão interna, Shatnawi e Yousef (2014) utilizam uma rede neural de extensão (*Extension neural network*, ENN) para detectar e classificar falhas em motores a combustão, como vazamento de ar na entrada do motor, falha na combustão em um ou em dois dos cilindros, falha no sensor de temperatura do líquido de arrefecimento e falha no sensor de fluxo de massa. Dada natureza do experimento, em que os dados sonoros sofrem flutuações devida às características dinâmicas e imperfeições desse tipo de motor, os autores utilizam a análise por modulação fractal por *wavelet* (ou em inglês, *wavelet packet decomposition*, ou WPD) para realizar a extração de características que serão utilizadas para alimentar a rede neural. Esta, por sua vez, é do tipo ENN, sendo composta somente por uma camada de entrada e uma camada de saída proposta inicialmente por Wang e Hung (2003), então mais simples e rápida que uma rede multicamadas. Os resultados desse método atingem taxas acima de 85% de acurácia para a classificação das falhas.

Jing et al. (2017) avaliam a aplicação de redes CNN para detectar sete condições diferentes (uma normal e seis falhas) e extrair características de (a) caixas de redução do tipo planetárias por meio de experimento e (b) dados de vibração de uma caixa de câmbio de bases de dados públicas. Em ambos os conjuntos de dados brutos, aplica-se a transformada rápida de Fourier (*Fast Fourier Transform*, FFT). No caso de (b) os dados são obtidos de acelerômetros acoplados a uma caixa de redução a qual passa por diversas falhas para coleta das informações. Nesse trabalho, compara-se diversas configurações de redes, destacando-se com maior acurácia a rede com uma camada convolucional, uma camada de *pooling*, e uma camada completamente conectada com a função de ativação *softmax*. Em ambos os casos, a rede proposta foi capaz de atingir acurácia acima de 98% na detecção das falhas propostas.

Zhu et al. (2019) apresenta um método de classificação de modos de combustão de motores do tipo *scramjet* baseado em redes neurais convolucionais. Os dados são da pressão estática dentro da câmara de combustão dos motores e são obtidos por experimentação. São avaliados por CNN de duas formas: com sinais brutos e com base de dados de características estatísticas no domínio do tempo e da frequência (tratadas por FFT) geradas a partir dos dados brutos. Os resultados mostram que o método baseado em CNN é capaz de revelar informações características e consegue classificar os quatro modos de combustão sugeridos do motor *scramjet* com uma acurácia acima de 90%, destacando-se em relação aos outros tipos de redes

neurais comumente utilizados, como *perceptron* de várias camadas (*multilayer perceptron*, ou MLP), *k-nearest neighbors* (k-NN), rede neural *feedforward* de uma camada escondida (SLFN) e *support vector machine* (SVM).

Pinedo-Sánchez, Mercado-Ravell e Carballo-Monsivais (2020) propõem um método baseado em CNN para estimar o nível de desgaste em rolamentos em cada um dos elementos, sendo pista interna, pista externa e esfera. Destaca-se aqui o uso de dados vibracionais experimentais rotulados automaticamente por meio da combinação de raiz quadrada média (*root-mean-square*, RMS) e entropia de Shannon, junto com clusterização por *k-means* para agrupamento de dados. Estes foram convertidos em imagem no domínio do tempo para identificação em duas dimensões. A rede CNN, com 3 camadas convolucionais, foi capaz de obter acuracidade acima de 98% na identificação das falhas.

Utilizando duas técnicas com diferentes dados, Firmino *et al.* (2021) avalia a detecção de falhas no evento de má combustão do combustível em um motor a gasolina com quatro cilindros de quatro tempos. A primeira técnica utiliza dados de vibração adquiridos a partir de acelerômetros instalados no meio do bloco do motor. A segunda utiliza sinais sonoros obtidos por uso de microfone posicionados a 30 cm do equipamento. A coleta dos dados foi feita simultaneamente. Nas duas técnicas, para simular as falhas, foram removidas as velas de ignição de cada um dos cilindros, uma por vez, além dos dados de funcionamento normal, somando cinco casos diferentes. Os sinais passam por um tratamento por FFT e alimentam uma rede neural artificial (ANN). O estudo mostra 99,30% de acurácia para os sinais de vibração e 98,7% de acurácia para os sinais sonoros na identificação das classes propostas.

Ribeiro *et al.* (2022) busca expandir a análise de motores elétricos para outros tipos de falhas que podem ocorrer nesses equipamentos, como folga na base do motor, desalinhamento, barras e eixos fraturados. Os sinais de aceleração obtidos por acelerômetro uniaxial são convertidos para o domínio do tempo-frequência por meio de STFT e assim geradas as imagens que a CNN utiliza. A rede, com três camadas convolucionais e três camadas de *pooling*, registra uma acurácia de 100% na identificação das 7 condições estudadas, demonstrando uma ótima robustez do método proposto.

Shahid, Ko e Kwon (2022), por sua vez, aplica uma ideia semelhante aplicada a motores a combustão, utilizando sensor magnético (*magnetic pick-up sensor*, MPU) para capturar a velocidade rotacional do virabrequim. Sugere a aplicação de uma rede CNN 1D para análise em tempo real de falhas relativas à queima irregular de combustível e mudanças de condições

de carga durante a operação do motor. A extração das características e classificação de falhas é feita por uma rede CNN com uma camada oculta, a qual atinge 99% de acurácia na detecção das condições anormais de operação com baixo erro de generalização, mostrando boa eficiência mesmo para uma rede simples.

Viana *et al.* (2023) realiza a análise de sinais de vibração torcional em um motor a diesel para quatro diferentes condições de funcionamento, sendo uma normal e três em modos de falha. Utilizando os métodos de rede neural artificial e floresta aleatória (*Random Forest*), ele realiza a classificação e regressão dos dados a partir de uma base de dados simulados com 3500 pontos de testes divididos entre as quatro condições. Entre os dois métodos, o RF obteve o menor valor da raiz do erro quadrático médio (RMSE) e a menor dispersão entre os valores.

Percebe-se que há poucas referências na literatura que tratem de falhas mais complexas e fora do espectro estrutural do motor. Portanto, o presente trabalho tem como objetivo principal o desenvolvimento de uma metodologia robusta e eficaz para classificar diferentes condições de operação (falhas) além das já extensivamente estudadas (majoritariamente estruturais) em motores a diesel a partir de sinal de vibração torcional do virabrequim. Para isto, técnicas de pré-processamento de sinal serão aplicadas (STFT e CWT) para gerar imagens nos domínios do tempo e da frequência que serão utilizadas em um algoritmo de inteligência artificial (CNN) de modo a reconhecer novos padrões de falhas.

CAPÍTULO 2 – EMBASAMENTO TEÓRICO

Neste capítulo é apresentada a base teórica necessária para o desenvolvimento do presente trabalho.

2.1 STFT

A Transformada de Fourier é uma técnica matemática fundamental que desempenha um papel crucial na análise de sinais e sistemas em diversos campos, como engenharia, física, matemática aplicada, processamento de sinais, entre outros. A ideia principal por trás dela é decompor uma função ou sinal complexo no domínio do tempo em suas componentes de frequência. Em outras palavras, ela permite representar um sinal como uma soma (ou integral, no caso da Transformada de Fourier contínua) de senos e cossenos de diferentes frequências. Isso é especialmente útil porque muitas vezes é mais fácil analisar e entender o comportamento de um sinal no domínio da frequência do que no domínio do tempo.

Essa ferramenta matemática converte uma função no domínio do tempo $f(t)$ em uma função no domínio da frequência $F(\omega)$, onde ω representa a frequência angular da onda. A fórmula dessa transformada para alguma função no domínio do tempo é dada pela equação 1 (BRUNTON; KUTZ, 2019).

$$\hat{f}(\omega) = \mathcal{F}(f(x)) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x} dx \quad (1)$$

A sua inversa, permitindo obter a função $f(t)$ dada sua transformada, é dada pela equação 2.

$$f(x) = \mathcal{F}^{-1}(\hat{f}(\omega)) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega)e^{i\omega x} d\omega \quad (2)$$

A Transformada de Fourier de Curto Período Tempo (em inglês, *Short-Time Fourier Transform*, ou STFT, ou ainda Transformada de Gabor), é uma extensão da Transformada de Fourier que consegue fornecer informações sobre a evolução temporal das frequências em um sinal, ou seja, como os sinais estão posicionados em relação ao tempo. Enquanto a Transformada de Fourier tradicional fornece uma representação global de frequências em todo o sinal, a STFT divide o sinal em segmentos menores e aplica a Transformada de Fourier a

cada segmento individual. Essa abordagem fornece uma representação no tempo e na frequência dada por um espectrograma, permitindo a análise de como as características espectrais de um sinal variam ao longo do tempo.

A fórmula para a STFT é dada pela equação 3.

$$\mathcal{G}(f)(t, \omega) = \hat{f}_g^{-1}(t, \omega) = \int_{-\infty}^{\infty} f(\tau) e^{-i\omega\tau} \hat{g}(\tau - t) d\tau \quad (3)$$

Onde $g_{t,\omega}(\tau)$ é definido como:

$$g_{t,\omega}(\tau) = e^{i\omega\tau} g(\tau - t) \quad (4)$$

A função $g(t)$ é o kernel e é em geral uma função Gaussiana dada por:

$$g(t) = e^{-\frac{(t-\tau)^2}{a^2}} \quad (5)$$

Na equação acima, a é metade do comprimento da janela ou segmento do qual se analisa o sinal e aplica-se a transformada de Fourier tradicional. A variável τ é a posição no tempo da metade dessa janela. Uma ilustração desse conceito pode ser vista na figura 2.1.

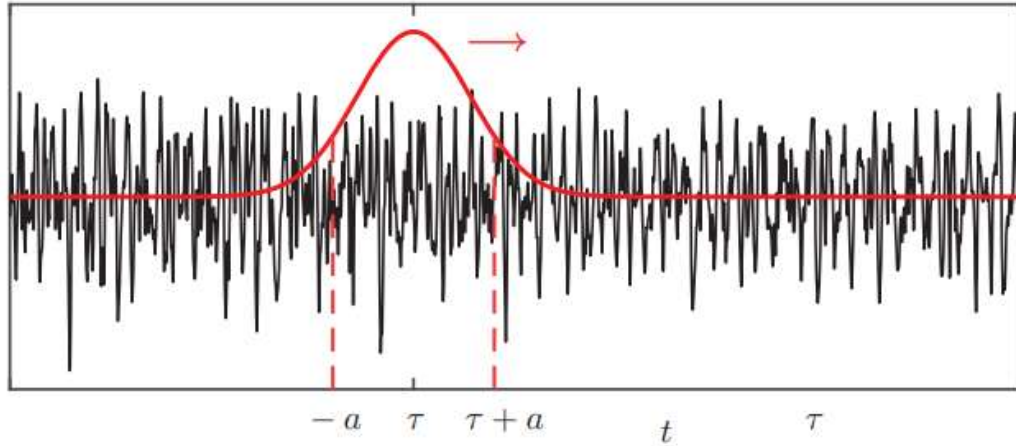


Figura 2.1 Ilustração da translação da janela do tipo Gaussiana utilizada na transformada STFT.

(Fonte: BRUNTON; KUTZ, 2019)

A inversa da STFT é dada pela equação 6 abaixo.

$$f(t) = \mathcal{G}^{-1}(\hat{f}_g(t, \omega)) = \frac{1}{2\pi\|g\|^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{f}_g(\tau, \omega) g(t - \tau) e^{i\omega\tau} d\omega d\tau \quad (6)$$

2.2 Transformada Wavelet Contínua (CWT)

A transformada wavelet estende o conceito da transformada de Fourier para uma base mais geral, podendo ser usada para analisar sinais estacionários (distribuição de frequência

constante ao longo do tempo) e não estacionários (distribuição de frequência dependente do tempo). Assim, ela permite uma análise multi-resolução mais poderosa no campo tempo-frequência especialmente de períodos transitórios em sinais não estacionários (WU; CHEN, 2006). Isso pode ser usado para análise de sinais de um motor em regime transitório, por exemplo, durante a ignição ou aumento da rotação, quando está sujeito à variação de vibração.

No caso da transformada *wavelet* contínua (CWT), é possível analisar sinais usando funções *wavelets* em diferentes escalas. Em outras palavras, é uma forma de decompor o sinal em *wavelets* menores e aplicar uma análise em cada uma usando escalas diferentes. Assim, fica possível observar como as frequências do sinal analisado distribuem-se em função do tempo. Isso é importante, por exemplo, em eletroencefalografia (EEG), permitindo que médicos observem mudanças na distribuição de frequência de sinais cerebrais e identifiquem anomalias relacionadas com transtornos neurológicos.

A ideia básica de uma análise *wavelet* se inicia com uma função chamada “mãe *wavelet*”, dada pela equação 7. A partir dela, gera-se uma família de funções escaladas e transladadas (BRUNTON; KUTZ, 2019).

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi \left(\frac{t-b}{a} \right) \quad (7)$$

Os parâmetros a e b são responsáveis por escalar e transladar a função ψ , respectivamente. A CWT é dada em função de $\psi_{a,b}(t)$, mostrada na equação 8 (BRUNTON; KUTZ, 2019).

$$\mathcal{W}_\psi(f)(a,b) = \int_{-\infty}^{\infty} f(t) \bar{\psi}_{a,b}(t) dt \quad (8)$$

onde $\bar{\psi}_{a,b}$ representa o complexo conjugado de $\psi_{a,b}$. Isso somente é válido para funções $\psi(t)$ cuja *wavelet* mãe satisfaz a condição dada pela inequação 9. A função inversa é dada pela equação 10 (BRUNTON; KUTZ, 2019).

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty \quad (9)$$

$$f(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathcal{W}_\psi(f)(a,b) \psi_{a,b}(t) \frac{1}{a^2} da db \quad (10)$$

2.3 Rede Neural Convolucional

Uma rede neural convolucional (*convolutional neural network*, ou CNN) é uma categoria especializada de rede neural profunda projetada para processar dados com estrutura de grade, como imagens. Elas se destacam na extração e aprendizado de características hierárquicas por meio de camadas convolucionais, que realizam diversas operações para detectar padrões locais. Elas são notoriamente conhecidas pela boa capacidade de extração automática de características e com ótima generalização para diversos problemas (ZHU *et al.*, 2019)

Uma rede neural típica é composta basicamente pelos seguintes elementos: camadas convolucionais, camadas de *pooling*, uma camada totalmente conectada com um modelo de classificação padrão e uma função de ativação ao final para concluir a classificação. Um exemplo típico desse tipo de rede pode ser visto na imagem 2.2. Essa arquitetura é eficaz na classificação de imagens, reconhecimento de objetos e tarefas relacionadas à visão computacional (PINEDO-SÁNCHEZ; MERCADO-RAVELL; CARBALLO-MONSIVAIS, 2020; RIBEIRO JUNIOR *et al.*, 2022).

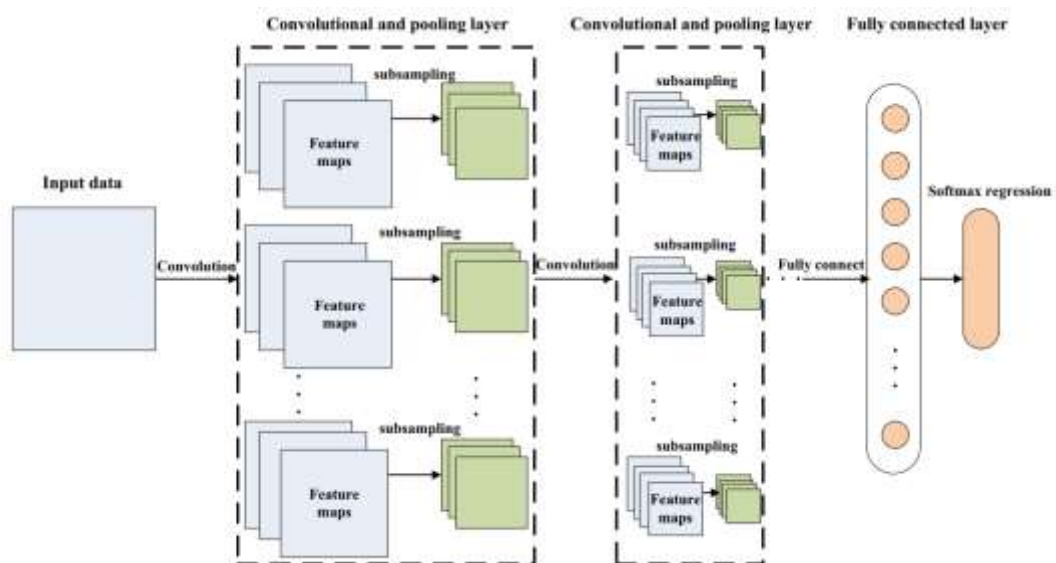


Figura 2.2 – Estrutura típica de uma rede neural convolucional (Fonte: JING *et al.*, 2017)

2.3.1 Camada de entrada

Na camada de entrada são inseridos os dados a serem avaliados pela rede neural. Estes dados podem ser tanto imagens 2D como vetores 1D. Dado que o método CNN é considerado supervisionado, exige-se que os dados sejam categorizados com suas respectivas classes para que a rede possa calcular o erro da rede.

2.3.2 Camada de convolução

As camadas de convolução são as partes principais das redes CNN. São elas que contém os *kernels*, ou filtros, que são matrizes que contém representações de características específicas que se desejam extrair. O resultado da aplicação dos *kernels* são os *feature maps*, os quais possuem dimensão menor que a da camada de entrada.

O funcionamento dos *kernels* baseia-se em percorrer a matriz de entrada e extraem as características mais relevantes da imagem. Em outras palavras, esses elementos são matrizes quadradas que representam “pesos” que realçam o que for mais importante (como bordas, regiões contrastantes e formas definidas) (RIBEIRO JUNIOR *et al.*, 2022). Um exemplo de como os *kernels* funcionam pode ser visto na imagem 2.3.



Figura 2.3 - Exemplo de *kernel* (matriz ao meio) aplicado à extração das características de uma imagem de um ônibus em uma rede 2D. Nota-se na saída o destaque para as regiões mais claras e contrastantes. (Fonte: WANG, Jay *et al.*, 2023)

As características que os *kernels* dão mais enfoque podem ser determinados manualmente, mas bibliotecas e métodos programáticos mais recentes permitem efetuar a extração de características automaticamente. Entretanto, outros hiper parâmetros como tamanho da matriz quadrada, *padding* e *stride* devem ser otimizados para cada problema (WANG, Jay *et al.*, 2023).

Em aplicações mais simples, o tamanho do *kernel* fica entre 3x3 e 6x6. Um kernel maior extrai menos informação, o que leva a uma redução mais rápida da dimensão na camada, mas com piora da precisão da rede. Em contrapartida, um kernel menor consegue extrair mais características levando a uma rede mais precisa, mas exige mais poder computacional. O *padding* permite conservar dados nas bordas dos mapas de ativação, gerando redes mais

profundas, mas exigindo muito poder computacional. Já o *stride* é o quanto o *kernel* se movimenta a cada instante. Em geral, o kernel se move item a item, e seu impacto segue da mesma forma que o tamanho do *kernel* (quanto maior o valor, menos características são aprendidas e perde-se performance, mas a rede se torna mais rápida). (KIRANYAZ *et al.*, 2021; WANG, Jay *et al.*, 2023)

Na ponta final de cada camada convolucional, cada *feature map* gerado é passado por uma função de ativação não linear que permite representações complexas que seriam impossíveis caso o modelo fosse composto em funções lineares, em outras palavras, resumido em um modelo linear (SUTTON; BARTO, 2014). Quanto mais camadas escondidas (intermediárias) com funções não lineares, maiores as abstrações da rede. Existem diversas funções, como a função de unidade linear retificada (do inglês, *rectified linear unit*, ou ReLU), *LeakyReLU* e *softmax*.

A representação matemática da camada de convolução pode ser vista na equação 11.

$$x_j^l = f \left(\sum_i x_i^{l-1} * k_{ij}^l + b_j^l \right) \quad (11)$$

onde x_j^l é o j-ésimo *feature map* produzido pela l-ésima camada, x_i^{l-1} é o i-ésimo *feature map*, k_{ij}^l é o j-ésimo kernel que está ligado ao i-ésimo *feature map*, b_j^l é o *bias* associado com o j-ésimo kernel, * representa o operador de convolução em duas dimensões (visto que o problema trata imagem), e f é a função de ativação.

2.3.3 Funções de ativação

2.3.3.1 ReLU

Uma função de ativação comumente usada é a ReLU nas camadas intermediárias por ter boa eficiência computacional na prática com problemas de classificação (SZANDAŁA, 2021). Ela e sua derivada são dadas pelas equações 12 e 13 abaixo.

$$f(x) = x^+ = \max(0, x) = \begin{cases} x & \text{se } x > 0, \\ 0 & \text{se não} \end{cases}, \quad (12)$$

$$f'(x) = \begin{cases} 1 & \text{se } x > 0 \\ 0 & \text{se } x < 0 \end{cases} \quad (13)$$

Nota-se que ela zera valores negativos e retorna o próprio valor para valores positivos. Apesar de ser muito usada, um grande ponto negativo dessa função é o problema do neurônio morto. Nessa situação, ao longo do treinamento os valores para o neurônio se tornam zero e tornam-se insensíveis ao treinamento, ou seja, perdem a capacidade de aprender.

Essa situação faz parte do problema da dissipação do gradiente em redes com métodos de aprendizado por gradiente e por *backpropagation*. Ele ocorre quando o gradiente se torna tão pequeno que perde-se a capacidade de alterar os valores dos pesos no neurônio, efetivamente desativando-o (SZANDAŁA, 2021).

2.3.3.2 LeakyReLU

Pensando no problema da função ReLU, existe a variante LeakyReLU. Nessa função, a parte negativa deixa de ser igual a zero e torna-se uma função linear com baixa inclinação positiva. Desse modo, diminui-se a chance de se retornar valores nulos e diminui-se o problema do neurônio morto. Ela é dada pela equação XX abaixo e pode ser visualizada na figura 14.

$$f(x) = \begin{cases} x & \text{se } x > 0, \\ \alpha * x & \text{se } x \leq 0 \end{cases} \quad (14)$$

Abaixo, uma visualização das funções ReLU e LeakyReLU.

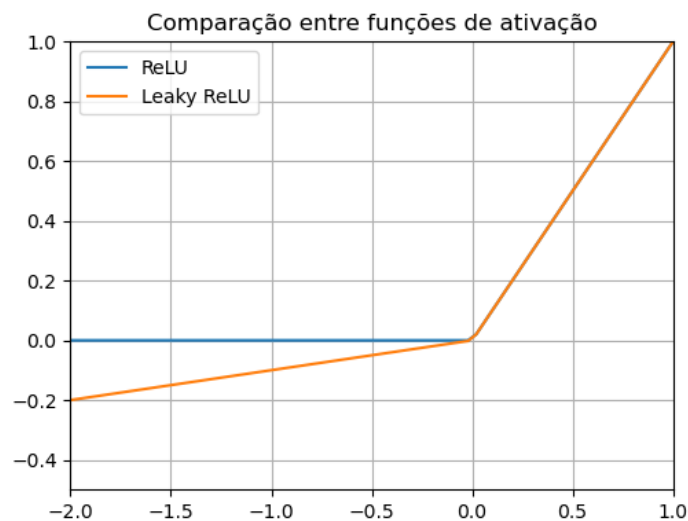


Figura 2.4 - Funções ReLU e LeakyReLU (com $\alpha = 0.1$ para fins de ilustração) (Fonte: autoria própria).

2.3.3.3 Softmax

A função *softmax* é bastante usada na camada de saída em redes neurais. Ela é dada pela equação 15 abaixo.

$$\text{Softmax}(x_i) = S(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \quad (15)$$

Onde i é o índice do neurônio de saída e j representa os índices de todos os neurônios de um nível. Essa função garante que, na saída da rede, a somatória dos valores de saída seja 1.

Assim, essa função é usada como indicador de probabilidade para cada item em problemas com mais de duas saídas possíveis. Por exemplo, em um problema com 4 possíveis saídas A, B, C e D, um caso possível é $S(x_A) = 0.20$, $S(x_B) = 0.10$, $S(x_C) = 0.63$ e $S(x_D) = 0.07$. Somando as probabilidades, temos o total de 1,0.

2.3.4 Pooling

Seguindo após as camadas de convolução, as camadas de *pooling* reduzem a dimensionalidade ao mesmo tempo que preservam características essenciais e tornam a rede invariante para pequenas transformações ou distorções, o que acaba melhorando a eficiência computacional do modelo. Semelhante à camada convolucional, a camada de *pooling* possui *kernels* que, por meio de alguma estratégia definida, realizam essa operação nos dados de entrada. As estratégias mais comuns de *pooling* são *max pooling* (pega o maior valor do *kernel*), *average-pooling* (pega o valor médio do *kernel*) e *stochastic pooling* (ZHU *et al.*, 2019).

A estrutura de uma camada com *max pooling* pode ser vista na figura 2.5. Neste exemplo o maior valor no kernel para o ponto analisado é 1.3, portanto é o valor escolhido. Nota-se, também, que, apesar de ter reduzido pela metade em X e em Y, a saída ainda consegue preservar as principais características da imagem original.

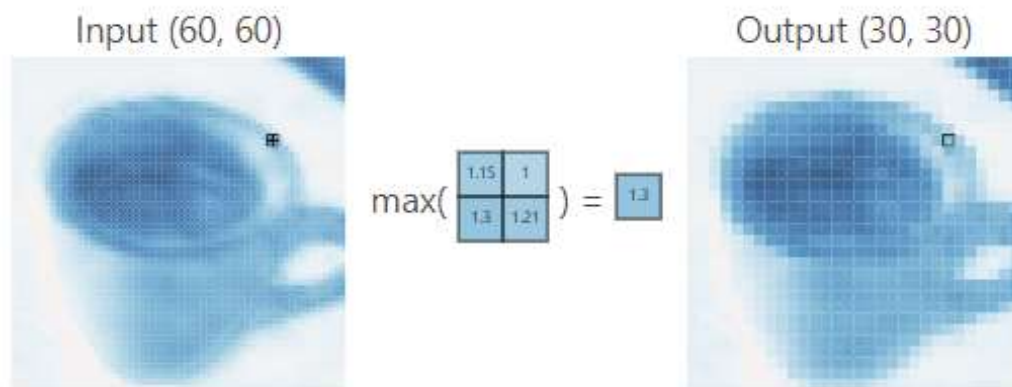


Figura 2.5 - Exemplo de *max pooling*. Nota-se a grande redução de dimensionalidade entre a entrada e a saída (Fonte: WANG, Jay *et al.*, 2023).

2.3.5 Fully-connected (FC) e classificação

Após passar por uma ou mais conjuntos de camadas de convolução e de *pooling*, finaliza-se uma CNN com uma rede completamente conectada (FC) e por uma função de classificação. A rede FC conecta cada neurônio em uma camada a cada neurônio na camada

adjacente. A função principal da camada FC é converter os mapas de características 2D em um vetor de características 1D e sintetizar a informação local nos recursos simples extraídos pelas camadas de convolução e *pooling*. Por fim, o vetor de características processadas é inserido na camada *softmax* dada pela equação 15 para produzir a pontuação de probabilidade de cada classe, e a classe com a pontuação mais alta será o resultado final da classificação. Na imagem 2.6 abaixo, é possível observar a camada FC e a camada de saída à direita.

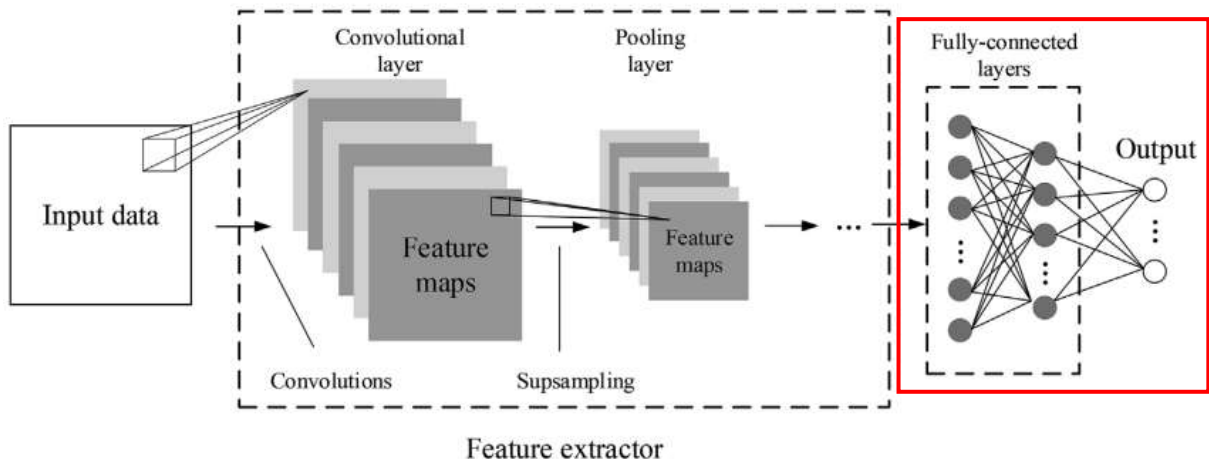


Figura 2.6 - Arquitetura de uma rede neural convolucional. Em destaque, a camada FC e a camada de saída (Fonte: ZHU *et al.*, 2019).

CAPÍTULO 3 – METODOLOGIA

3.1 Modelagem do problema

Neste trabalho utiliza-se dados de vibração torcional de um motor a diesel. Essa vibração pode ser caracterizada como uma flutuação da velocidade rotacional do virabrequim ocasionada pela natureza periódica da combustão e do próprio movimento inercial dos elementos móveis. Esse tipo de vibração é detectável principalmente no virabrequim do motor (WANG, P. *et al.*, 1992; XUE; HOWARD, 2018). Diversas formas de medição podem ser aplicadas, como utilizando dois acelerômetros lineares posicionados diametralmente opostos na superfície do eixo (eliminando movimento translacional), acelerômetros angulares (posicionados na face da seção transversal do eixo), extensômetros (ou *strain gauge*), sensores magnéticos, ou ainda interferômetros a laser (mais caros e de uso em ambiente controlado) (TORSIONAL VIBRATION: WHAT IS IT?, 2014).

Os dados de vibração utilizados neste trabalho são obtidos por meio de simulações computacionais de falhas de um motor a diesel de 6 cilindros em linha real e posteriormente validados com a fabricante. Eles foram extraídos a partir da base de dados 3500-DEfault, disponibilizadas publicamente dentre os *datasets* da plataforma Mendeley (PESTANA, 2020). O motor foi avaliado em condição normal de operação e em mais três outras condições anormais de operação (falhas). Estão descritas abaixo:

- Condição normal;
- Redução de pressão na entrada de ar do *manifold*;
- Redução na pressão de compressão em cada um dos cilindros;
- Redução na quantidade de combustível injetada nos cilindros.

3.1.1 Especificações do motor

Os dados correspondem ao motor a diesel marítimo sobrealimentado Acteon 6.12TCE fabricado pela empresa MWM Diesel Motors (GUTIÉRREZ *et al.*, 2018). Na tabela 1 abaixo, pode-se observar as características físicas e operacionais do motor usado de base para a geração dos dados simulados de falhas.

Tabela 1 - Tabela de especificações do motor usado como base para simulação dos dados.

Descrição	Especificação
Tipo de ciclo	4 tempos
Número de cilindros	6 cilindros em linha
Controle de válvula	no cabeçote do cilindro
Número de válvulas/cilindro	2 válvulas
Diâmetro do cilindro	105 mm
Curso do pistão	137 mm
Comprimento da biela	207 mm
Cilindrada total	7118 L
Razão de compressão	16, 8:1
Ângulo de fechamento da válvula de entrada	203°
Ângulo de abertura da válvula de escape	507°
Torque máximo	900 Nm em 1600 RPM
Potência máxima	191 kW em 2500 RPM
Ordem de ignição	1-5-3-6-2-4
Direção de rotação	Anti-horário
Pressão na linha de injeção	350 a 1400 bar
Temperatura da água de refrigeração	80-100 °C

Fonte: VIANA *et al.*, 2023.

3.1.2 Descrição do modelo simulado do motor

Para realizar a simulação, foram considerados os seguintes modelos (VIANA *et al.*, 2023):

- Modelo termodinâmico de dimensão zero (0D).
- Modelo de massa concentrada para a vibração torcional no virabrequim.
- Modelo de simulação de falhas.

O modelo termodinâmico foi validado relacionando as curvas de pressão dentro do cilindro, enquanto o modelo computacional de vibração torcional usou a curva de torque nominal de saída do motor, ambos fornecidos pelo fabricante do motor citado anteriormente.

O modelo termodinâmico matemático adotado é o zero dimensional ou termodinâmico *single-zone* e é baseado na Primeira Lei da Termodinâmica e na equação de estado, com a finalidade de obter os dados de pressão dos gases contidos no interior do cilindro, em função do ângulo da árvore do virabrequim. Considera-se que a mistura de gases dentro do cilindro durante o processo de admissão e de exaustão ocorre a pressão constante, então o momento da compressão pode ser considerado como um processo politrópico, visto pela equação XX.

Observa-se que esses valores dependem da posição angular do virabrequim, dado por θ (GUTIÉRREZ *et al.*, 2018).

$$\frac{dP}{d\theta} = -n \frac{P}{V} \frac{dV}{d\theta} \quad (16)$$

O fluxo mássico de ar durante o processo de admissão e de exaustão é estacionário, e o volume de controle considerado nesse modelo zero dimensional é dado pela figura 3.1.

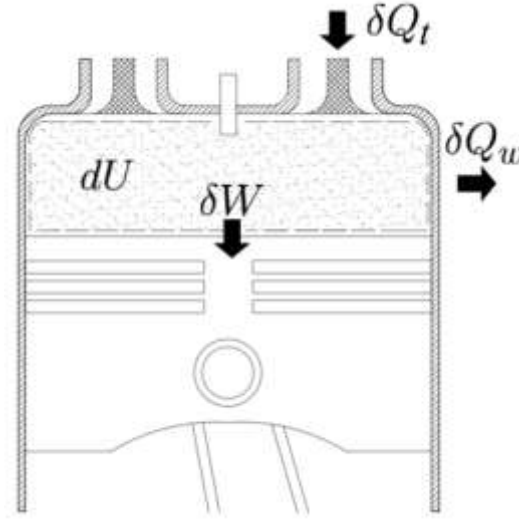


Figura 3.1 - Volume de controle considerado no modelo termodinâmico (VIANA *et al.*, 2023).

A partir da primeira lei da termodinâmica e considerando perdas por convecção e por radiação, pode-se chegar na equação 17 que representa a taxa da temperatura da mistura de gases no cilindro:

$$\frac{dT}{d\theta} = \left(\frac{\delta Q_t}{d\theta} - \frac{\delta Q_w}{d\theta} - P \frac{dV}{d\theta} \right) \frac{1}{mc_v} \quad (17)$$

Onde $\frac{\delta Q_t}{d\theta}$ é calor fornecido ao sistema devido à queima do combustível, $\frac{\delta Q_w}{d\theta}$ é a perda de calor para as paredes do cilindro, $\frac{dV}{d\theta}$ é o volume da câmara de combustão. Todos esses itens são dependentes da posição angular θ do virabrequim. P , m e c_v representam, respectivamente, a pressão instantânea dos gases, a massa da mistura gasosa e o calor específico à volume constante da mistura.

Como foi considerado um modelo zero-dimensional, pode-se afirmar que o gás dentro do cilindro se comporta de maneira ideal. Portanto, a variação da pressão em função da posição do virabrequim $\frac{dP}{d\theta}$ pode ser dada pela equação 18.

$$\frac{dP}{d\theta} = \left(mR \frac{dT}{d\theta} - P \frac{dV}{d\theta} \right) \frac{1}{V} \quad (18)$$

onde P é a pressão interna ao cilindro (em pascal), m é a massa em kg de ar no cilindro, R é a constante universal dos gases, V é o volume do cilindro na posição θ e T é a temperatura da mistura dentro do cilindro (VIANA *et al.*, 2023).

Além do modelo termodinâmico, deve-se incluir o modelo de massa concentrada para representar a dinâmica das partes móveis do motor. Esse modelo, então, representa a dinâmica do virabrequim, considerando dois amortecedores torcionais, a polia do virabrequim, um conjunto de engrenagens para transmissão de potência para outras partes do motor, os seis cilindros e o volante de inércia. Considerando a segunda lei de Newton, tem-se a equação 19:

$$[J]\{\ddot{\theta}(t)\} + [C]\{\dot{\theta}(t)\} + [K]\{\theta(t)\} = \{M(t)\} \quad (19)$$

onde $[J]$ é a matriz torcional, $[C]$ a matriz de amortecimento e $[K]$ a matriz de rigidez do sistema. $\{\ddot{\theta}(t)\}$, $\{\dot{\theta}(t)\}$ e $\{\theta(t)\}$ são, respectivamente, a aceleração angular, a velocidade angular e a posição angular do virabrequim em forma de vetores. Por fim, $\{M(t)\}$ é o vetor de momentos sobre o eixo.

A partir desse modelo teórico, é possível definir um modelo de vibração torcional. Este, na prática, é intrínseco ao funcionamento de um motor a combustão dado que sobre o virabrequim ocorrem a aplicação de torques inerciais, devido ao movimento alternado dos componentes móveis (pistão, biela, anéis etc.), e torques de combustão, devido à queima do combustível na câmara. Em ambos os casos, a aplicação de torque não é constante, mas sim oscilatória (MENDES; MEIRELLES; ZAMPIERI, 2008; VIANA *et al.*, 2023). Em outras palavras, dependem da posição angular do virabrequim. A carga devido à inercia é dada abaixo pela equação 20 e a carga devido à combustão pela 21.

$$F_r = m_r r \Omega^2 (\cos \theta + l \cos(2\theta) - \frac{l^3}{4} \cos 4\theta + \frac{9l^5}{128} \cos 6\theta) \quad (20)$$

$$F_c = P \frac{\pi D^2}{4} \quad (21)$$

O momento então é dado por:

$$M = r(F_r + F_c)(\sin \theta + \cos \theta \tan \gamma) \quad (22)$$

As variáveis m_r é a massa alternada, r é o raio do virabrequim, Ω é a velocidade de rotação do virabrequim em rad/s e l é a razão entre o raio do virabrequim e o comprimento da biela. A variável gama é dependente de θ e é dada por:

$$\gamma = \frac{l \sin \theta}{1 - \frac{l^2}{4} + \frac{l^2}{4} \cos 2\theta} \quad (23)$$

Os resultados da simulação são comparados com as curvas de pressão e de torque experimentais fornecidos pelo fabricante. Na figura 3.2, observa-se as curvas simuladas de pressão dentro do cilindro comparadas com as curvas reais para 1000, 1900, 2300 RPM e 2500 RPM. De acordo com Viana *et al.* (2023) e (GUTIÉRREZ *et al.*, 2018), os menores erros de pressão máxima e pressão média são de 0% e 5%, respectivamente. Na figura 3.3, compara-se valores de potência e de torque entre dados simulados e dados experimentais para o mesmo intervalo de rotações. Comparando com dados do fabricante, o modelo termodinâmico se mostra fisicamente válido com baixo erro.

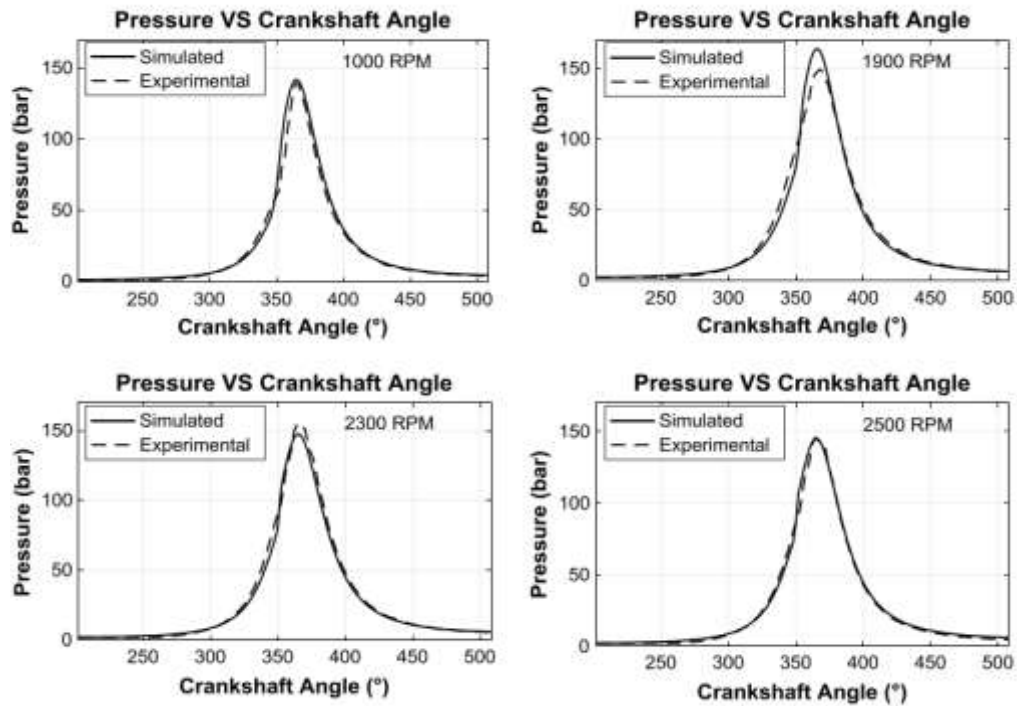


Figura 3.2 – Pressão em um dos cilindros do modelo termodinâmico simulado versus dados experimentais para diferentes RPM (Fonte: GUTIÉRREZ *et al.*, 2018).

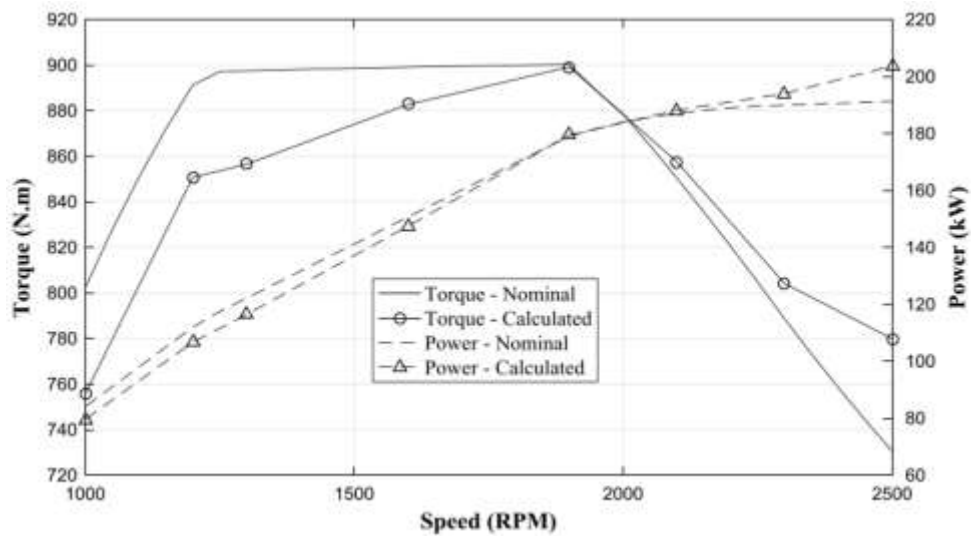


Figura 3.3 - Dados de torque e potência reais e simulados (calculados) para diferentes RPM (Fonte: GUTIÉRREZ *et al.*, 2018).

O modelo zero dimensional pode ser aplicado para mais de um cilindro visto que os comportamentos são idênticos entre si. Na figura 3.4 observa-se as curvas de pressão simulada de cada cilindro em função da posição angular do virabrequim. Já na figura 3.5 estão expostas as curvas de torque, também para cada cilindro. Nota-se que é respeitada a ordem de ignição dos cilindros, de acordo com a tabela 1.

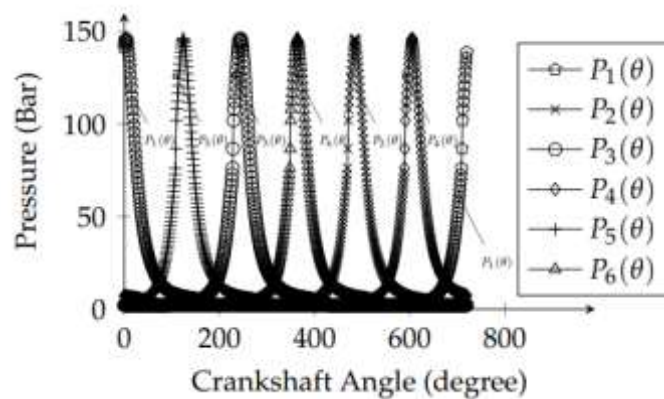


Figura 3.4 - Curvas de pressão para 2500 RPM (Fonte: VIANA *et al.*, 2023).

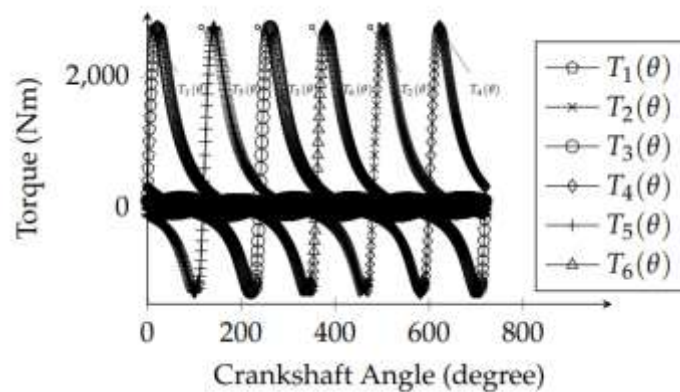


Figura 3.5 - Curvas de torque para 2500 RPM (VIANA *et al.*, 2023)

3.1.3 Descrição dos dados

A base de dados utilizada possui no total 3500 sinais diferentes relativos à vibração torcional do virabrequim. Esses dados foram construídos utilizando simulação numérica com base no modelo de motor detalhado anteriormente e geradas considerando o motor com rotação de 2500 RPM, dado que nesse RPM a simulação demonstrou o menor erro conjunto da pressão média e máxima do ciclo de combustão durante a etapa de validação dos modelos termodinâmicos dentre todas as rotações observadas. A frequência de amostragem dos dados de é 15kHz (VIANA *et al.*, 2023).

A base de dados disponibilizada por Pestana (2020) e usada neste trabalho possui dados com diversos níveis de ruído acrescentados, sendo 0dB (sem ruído), 15dB, 30dB e 60dB. Entretanto, como este trabalho avalia um novo método, escolheu-se trabalhar com dados com 0db. Posteriormente, será acrescentado valores de ruído em outras intensidades manualmente para avaliar detalhadamente a performance do modelo para além dos valores originalmente publicados.

Das diversas falhas possíveis que podem ocorrer em um motor a diesel, escolheu-se falhas de natureza termodinâmica. Os sinais estão divididos em:

- 1) 250 pontos para condição normal;
- 2) 250 pontos para condição de redução de pressão, ΔP_i , na entrada de ar no *manifold*, causada, por exemplo, por mal funcionamento do *turbocharger* ou corrosão da válvula de admissão;
- 3) 1500 pontos para condição de redução da razão de compressão, Δr , nos cilindros, devido à corrosão ou folga do pistão;

- 4) E 1500 pontos para condição de redução da quantidade de combustível, Δm , injetada nos cilindros.

O segundo caso é um problema geral que afeta todos os cilindros. O terceiro e o quarto caso são falhas locais de cada cilindro. Assim, pode-se definir o vetor de falhas pela equação 24:

$$\Delta f = \left\{ \Delta P_i, \Delta m_{c_j}, \Delta r_j \right\}, j = 1, 2, 3, 4, 5, 6 \quad (24)$$

onde j é o número do cilindro. Cada item desse vetor é uma falha dada em porcentagem do valor sob condição normal de operação.

Viana *et al.* (2023) desenvolve os dados e a aplicação de severidade conforme descrito a seguir. Como são dados simulados, aplica uma pequena porcentagem de severidade no caso normal. Essa severidade, contida em um intervalo de 0 a 0.1%, foi distribuída com uma probabilidade uniforme entre as variáveis físicas utilizadas na simulação. Com isso, deseja-se simular as condições reais de funcionamento em que o motor funciona com pequenas variações ao redor do funcionamento ótimo.

Já na segunda condição acima, acrescentaram valores de severidades de pressão no *manifold* em incrementos de 0,2 pp, ou seja, $\Delta P_r \in \{1.0, 1.2, 1.4, \dots, 49.8, 50\}$ em porcentagem, totalizando 250 pontos, onde ΔP_r representa a variação de pressão aplicada na pressão em operação normal no *manifold*.

Nos dois últimos casos, ambos com 1500 pontos, as severidades foram aplicadas para cada cilindro. Assim, considerando o cilindro i , onde $i \in \{1, 2, 3, 4, 5, 6\}$, vários cenários com severidades $\Delta r_i \in \{1.0, 1.2, 1.4, \dots, 49.8, 50\}$ e $\Delta m_{c_i} \in \{1.0, 1.2, 1.4, \dots, 49.8, 50\}$, onde Δr_i é o incremento aplicado sobre a taxa de compressão em operação normal r no cilindro i e Δm_{c_i} é o incremento na vazão mássica de combustível na vazão mássica em operação normal m_c no cilindro i . Portanto, como em cada intervalo existem 250 pontos, onde cada um é aplicado para cada um dos seis cilindros, totaliza-se 1500 pontos para cada uma das severidades 3 e 4.

3.2 Modelagem numérica

Os dados de vibração torcional obtidos a partir do processo de simulação computacional explicado na seção 3.1 são exibidos na Figura 3.6, onde pode-se observar o sinal referente ao primeiro sinal (ponto) de cada condição operacional. Verifica-se que com base apenas nos sinais de vibração no tempo, não é possível distinguir e classificar os sinais para cada tipo de

condição, assim faz-se necessário o emprego de técnicas de processamento de sinais para se extrair mais informações do sinal.

Assim, estes sinais são pré-processados a partir das transformadas STFT e CWT, dando origem aos espectrogramas e escalogramas, respectivamente. Para cada um dos 3500 pontos, aplicam-se ambas as transformadas, obtendo-se um total de 3500 espectrogramas e 3500 escalogramas.

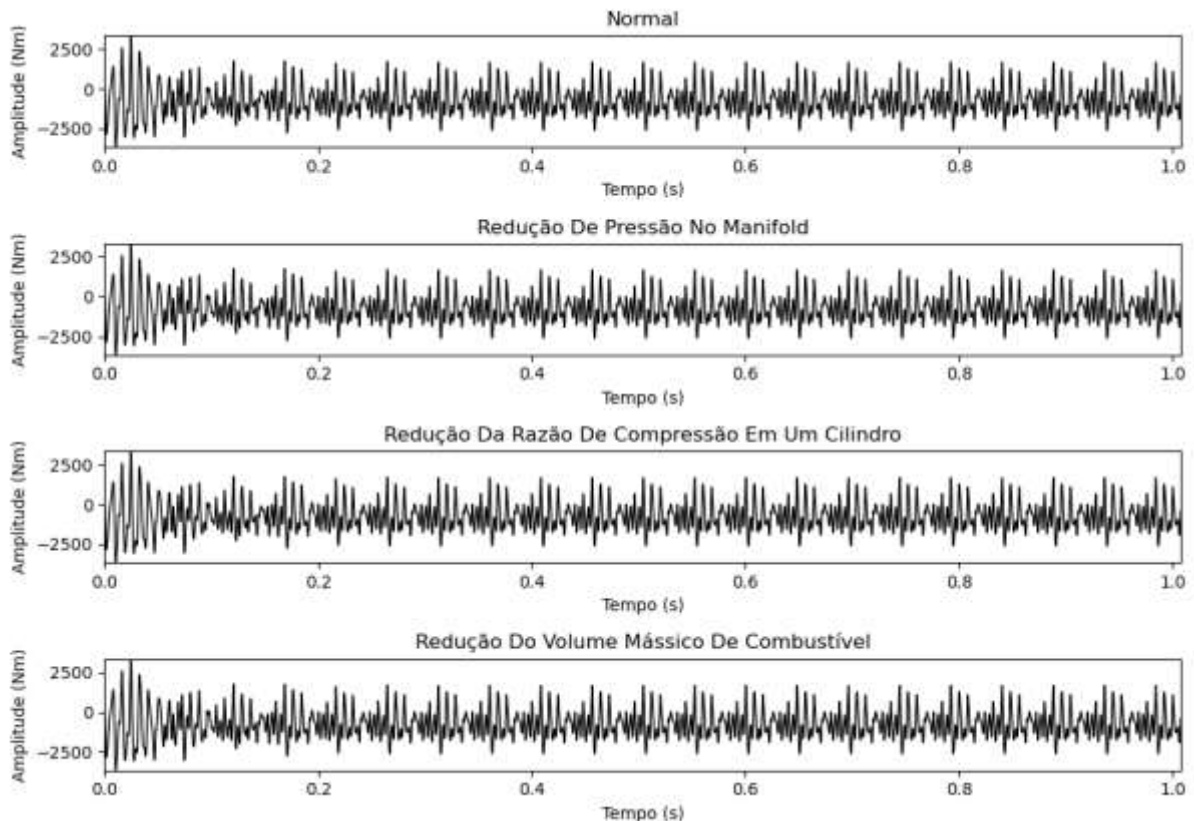


Figura 3.6 - Sinais de vibração para cada condição de operação.

Para representar a transformada STFT e gerar os espectrogramas, utilizou-se a função *spectrogram* da biblioteca SciPy em Python, tendo como parâmetros a função Blackman (OPPENHEIM; SCHAFER, 2010) para a janela $w(\tau)$, com comprimento de 256 segmentos. Observou-se que os valores de frequência ficavam baixos, portanto, limitou-se os gráficos com um limite superior de 1000 Hz.

Já para gerar os escalogramas para representar a saída da transformada CWT, utilizou-se a função CWT da biblioteca ObsPy. Por padrão da biblioteca, a wavelet mãe é representada pela função Morlet (KRISTEKOVA *et al.*, 2006), a frequência máxima inicial foi de 2000 Hz e a frequência mínima inicial de 1 Hz. Essas frequências foram escolhidas por meio de experimentação e por observar a ocorrência de baixas frequências na STFT. Deixou-se uma

margem superior para análise dos dados, entretanto notou-se que os valores, nesse caso, também ficavam abaixo de 1000 Hz, portanto posteriormente adotou-se o mesmo limite superior que na STFT. Essa observação é válida dado que a rotação com que os dados foram gerados é de 2500 RPM ou 46,67 Hz.

Um exemplo de espectrograma e escalograma para cada condição operacional é apresentado nas Figuras 3.7 e 3.8, respectivamente. No eixo y está distribuída a frequência e no eixo x, o tempo. No eixo z está representado a amplitude do sinal.

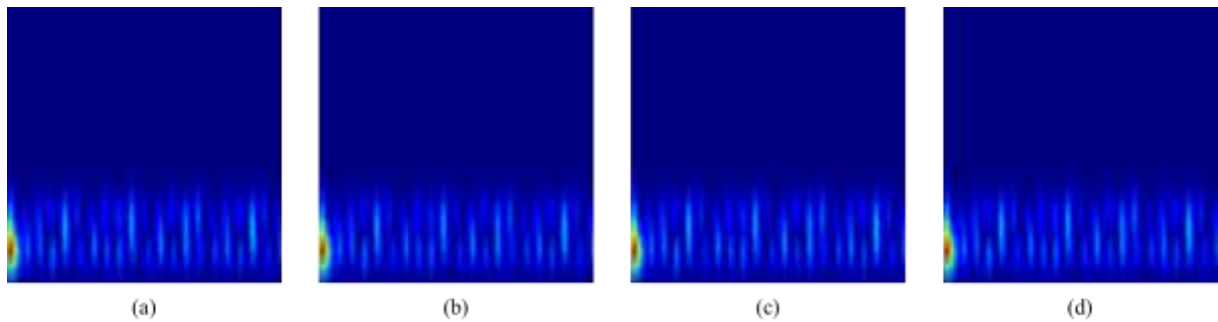


Figura 3.7 - STFT: (a) Condição normal, (b) redução de pressão no *manifold*, (c) redução da razão de compressão nos cilindros e (d) redução na quantidade de combustível nos cilindros.

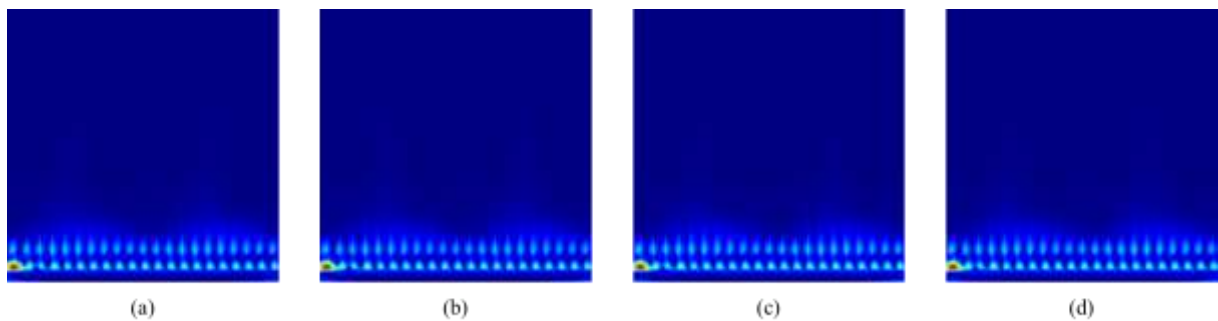


Figura 3.8 - CWT: (a) Condição normal, (b) redução de pressão no manifold, (c) redução da razão de compressão nos cilindros e (d) redução na quantidade de combustível nos cilindros

É possível notar que as diferenças entre as condições são praticamente imperceptíveis à olho nu, o que demonstra que, mesmo que haja um treinamento de análise de sinais, a detecção de falhas auxiliado por redes neurais se mostra como uma ótima alternativa para esta situação.

Para avaliar a robustez do modelo desenvolvido e aproximar as condições aqui descritas da realidade simulando dados medidos em uma condição de experimentação real, aplicou-se ruído branco nos dados do tipo Gaussiano (em inglês, *White Gaussian Noise*, ou WGN). Ele é gerado e acrescentado a cada um dos sinais de vibração analisados. Além de 0dB, aos dados foram aplicados 5 diferentes níveis de ruído, obtidos pelo inverso da razão de sinal sobre

potência de ruído (*signal-to-noise ratio*, ou SNR) (JONDRAL, 2018), sendo eles: 10%, 20%, 40%, 60% e 80%. A equação para a SNR é dada por 25.

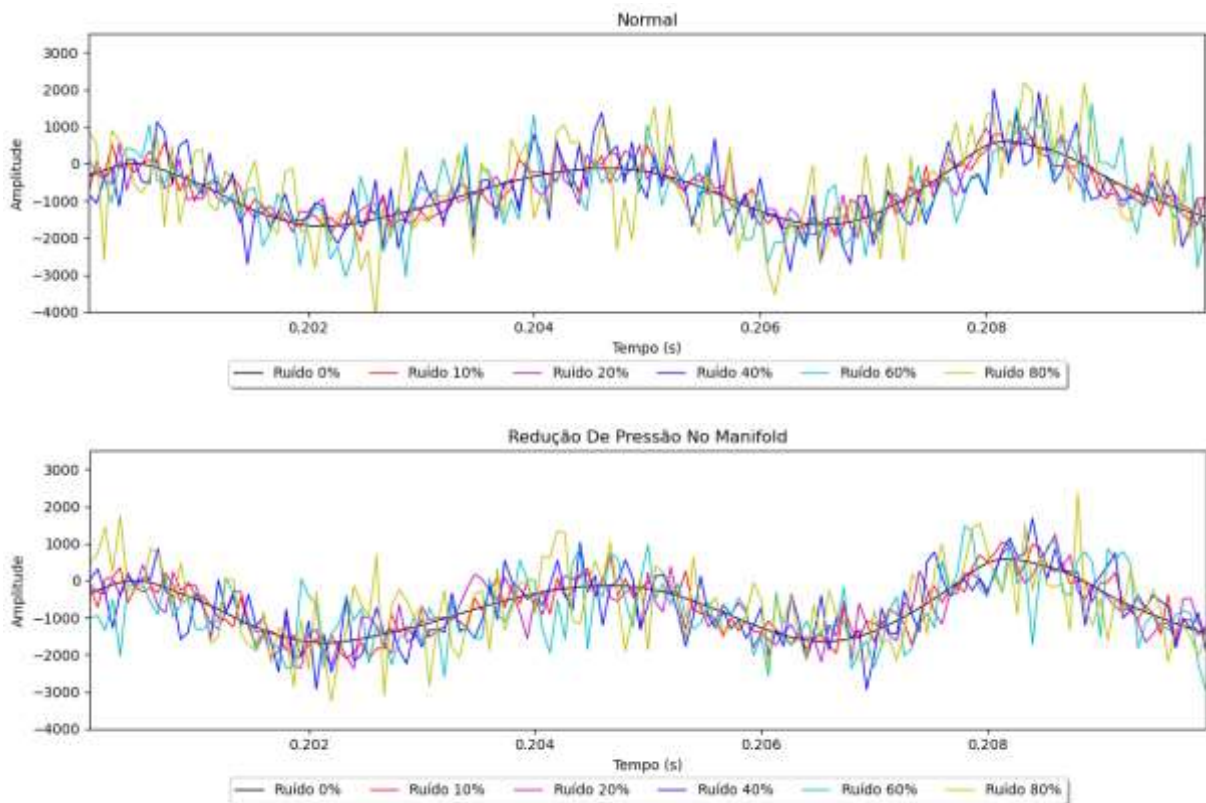
$$SNR = \frac{\text{Potência do sinal de entrada}}{\text{Potência do ruído}} = \frac{S}{R} = \frac{100}{\% \text{ de ruído}} \quad (25)$$

$$SNR_{dB} = S_{dB} - R_{dB} \quad (26)$$

Deve-se converter o valor de potências para decibéis. Portanto aplica-se a equação 27, onde j indica se é sinal original ou ruído.

$$SNR_{j,dB} = 10 \log_{10}(SNR_j) \quad (27)$$

Com o sinal em *watts*, extrai-se a média e a converte para decibéis utilizando a equação 27 (acima). Por fim para gerar um ruído aleatório, utiliza-se uma função aleatória de distribuição normal com média zero e variância igual a R_{dB} . Os valores gerados aleatoriamente são então somados ao sinal original, dando origem ao sinal ruidoso. Na imagem 3.9 é possível ver amostras dos dados com ruído.



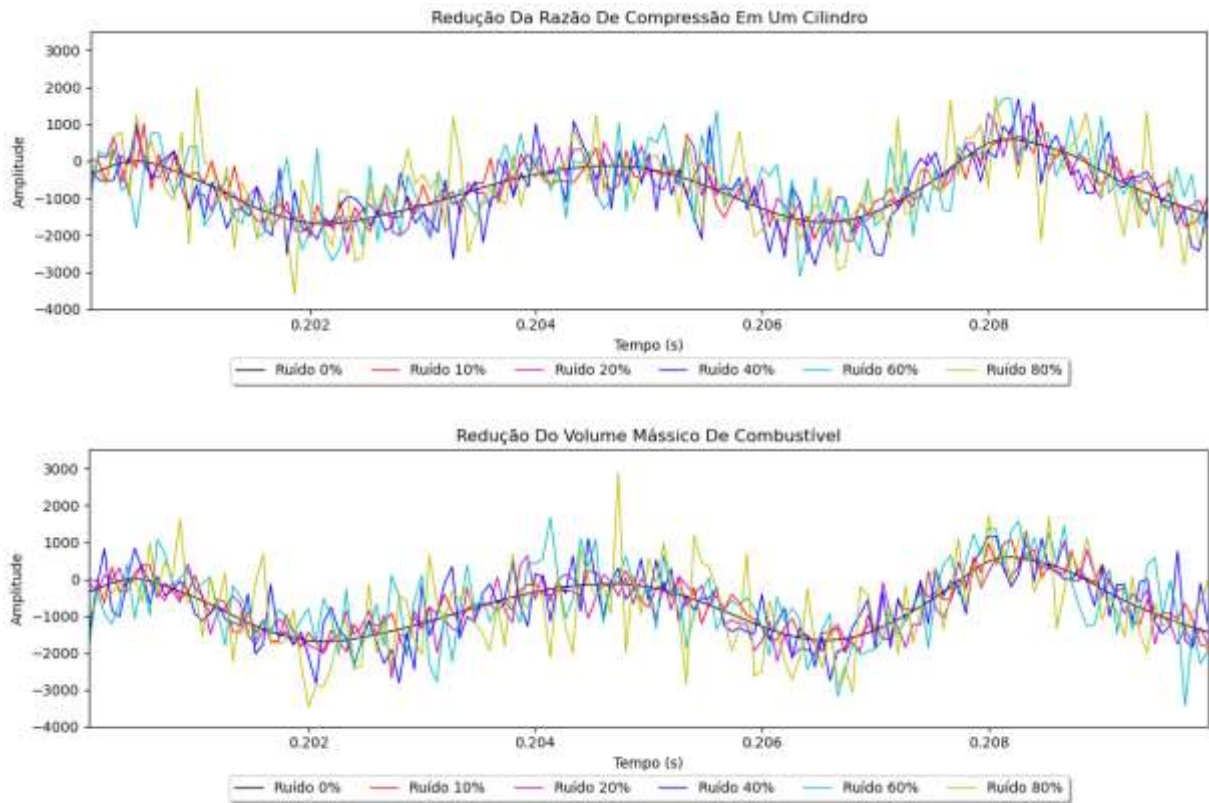


Figura 3.9 - Sinais originais e sinais acrescentados de ruídos.

Da mesma forma que os sinais sem ruído (0dB), foram geradas 3500 imagens para cada nível de ruído citado anteriormente. Abaixo, estão exemplos para o caso normal.

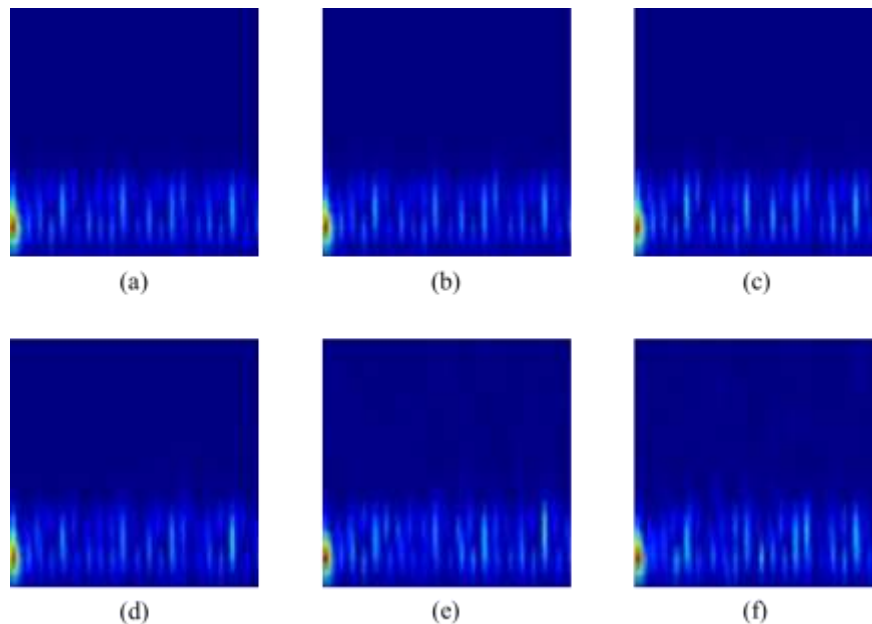


Figura 3.10 - Exemplos de espectrograma com ruído, condição normal: (a) 0%, (b) 10%, (c) 20%, (d) 40%, (e) 60% e (f) 80%.

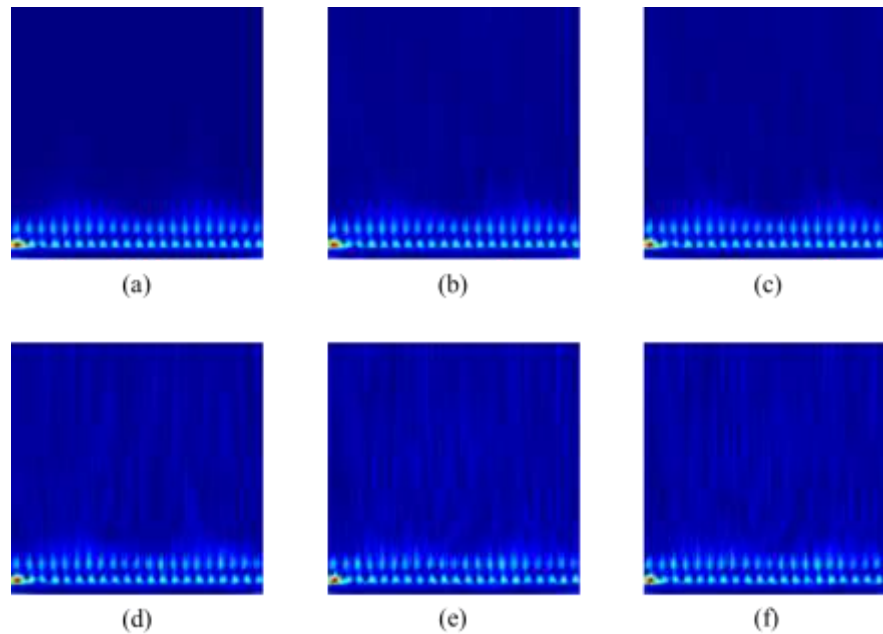


Figura 3.11 - Exemplos de escalograma com ruído, condição normal: (a) 0%, (b) 10%, (c) 20%, (d) 40%, (e) 60% e (f) 80%.

Percebe-se que a transformada STFT se demonstrou pouco sensível ao ruído, com traços de ruído aparecendo à olho nu somente a partir de 60%. Já com a CWT, logo em 10% já é possível notar uma clara presença de ruído, com o sinal se tornando muito ruidoso a partir de 40%. Isso pode ser explicado pelo fato da CWT se adaptar melhor a sinais que sofrem alterações de frequência ao longo do tempo, e, neste caso, o sinal se mantém em regime permanente na maior parte do gráfico. Vale notar, entretanto, que no CWT o sinal se comporta melhor logo no começo do tempo à esquerda do gráfico, onde há um pico de amplitude do sinal. Esse pico, causado pela partida do motor, causa um breve efeito transiente que faz o sinal ser menos sensível ao ruído.

Os *scripts* e processos de otimização das redes neurais foram feitos em um computador com a seguinte configuração: processador AMD Ryzen 5 5600H, com 16 GB de memória RAM e placa de vídeo NVIDIA GeForce RTX 3060. Vale ressaltar que o processamento foi feito majoritariamente pelo processador, entretanto há outros métodos de aceleração de software que utilizam a placa de vídeo, tornando o processo bem mais rápido.

3.3 Formulação da Rede Neural Convolutacional

Dentre as 3500 imagens de espectrogramas (STFT) e escalogramas (CWT), utilizou-se 75% para treino e os outros 25% para testes. Os dados foram carregados e separados automaticamente entre cada um destes grupos utilizando a função *train_test_split* da

biblioteca *scikit-learn*. Já a rede neural convolucional foi criada utilizando a biblioteca Keras. Foram utilizadas 3 camadas de convolução, cada uma com *kernels* com tamanho 5x5 e *stride* de 2x2. À cada camada convolucional foi atribuída uma função de ativação do tipo LeakyReLU com um alfa de 0,3 e um regularizador de filtros do tipo L2. Este regularizador ajuda a evitar *overfitting*, em outras palavras, ajuda a deixar a rede com capacidade generalizada e identificar novos exemplos além dos que foram usados para treinamento. Após à cada camada convolucional, segue-se com uma camada de *max pooling* com filtro de tamanho 2x2 e uma camada de *dropout*. O *dropout* funciona desligando as ligações anteriores e posteriores de um neurônio temporariamente, criando, portanto, uma nova rede. Essa seleção de neurônios é dada pela probabilidade p de *dropout*. Esta ferramenta também evita *overfitting* (ZHU *et al.*, 2019). Para a camada final *fully-connected* (FC), adotou-se a função de ativação *softmax* e o algoritmo ADAM como otimizador.

Para configurar o treino da rede neural, é necessário definir alguns parâmetros, como taxa de aprendizado (*learning rate*), *batch size* e número de épocas (*epochs*). O primeiro tem bastante influência na determinação dos pesos que serão definidos na rede neural. Já *batch size* permite separar os dados de treino em pequenos grupos para evitar sobrecarga computacional. Já o número de épocas controla quantas vezes o algoritmo irá rodar durante o treino com cada bloco de dados. Esses itens foram escolhidos por tentativa (RIBEIRO JUNIOR *et al.*, 2022).

Primeiramente, manteve-se o número de *batch size* e *epochs* constantes em 64 e 500, respectivamente. O *learning rate* foi variado utilizando os valores em {0,0001, 0,0005, 0,001, 0,005, 0,01, 0,05}. A maior acurácia obtida foi com 0,001. Com esse novo valor, variou-se o *batch size* entre 8 a 64 com valores intermediários múltiplos de 8, chegando na melhor acurácia com valor de 16. Por fim, variou-se o número de épocas da mesma forma, variando de 0 a 500, com passo 100. Aqui, não foi percebida muitas mudanças no valor de acurácia, portanto definiu-se 500 épocas. Essa configuração foi usada em todos os procedimentos.

Para determinação dos hiperparâmetros, utilizou-se a biblioteca *keras_tuner* que permite inserir intervalos ou listas de possibilidade aos campos numéricos, não-numéricos e binários contidos na construção da rede. Por exemplo, para os campos de valor de regularizadores L2 foi adotado um intervalo de testes de 10^{-4} a 0,1. Assim, pode-se avaliar vários pontos e combinações de hiperparâmetros durante o processo de treinamento. Outros campos que foram testados foram: valores de *dropout* variando entre 0,015 e 0,15 e número de *dense units* na última camada FC, variando de valores inteiros entre 16 e 512 com um passo de 16

unidades. O procedimento de otimização dos hiperparâmetros é feito de maneira automática utilizando o algoritmo de otimização bayesiano com o objetivo estipulado de obter a maior acurácia. Um esquema visual da estrutura da rede final pode ser visto na figura 3.12.

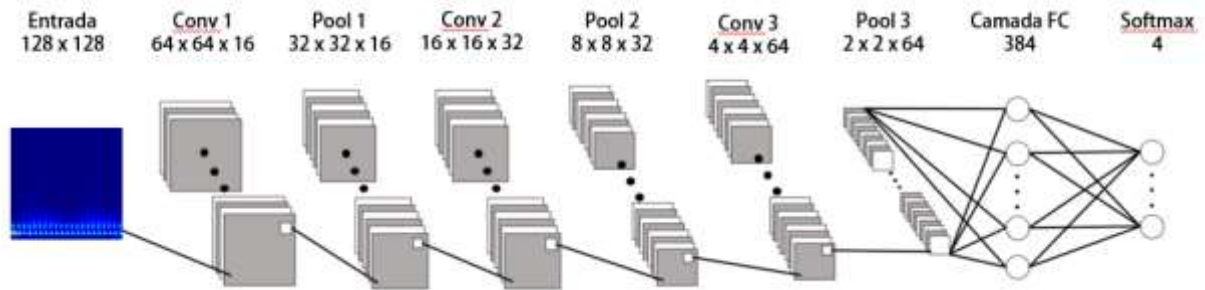


Figura 3.12 - Ilustração da estrutura e dimensões da rede neural utilizada (Adaptado de: RIBEIRO JUNIOR *et al.*, 2022)

Todos os códigos utilizados podem ser encontrados em um repositório Git no seguinte link: https://github.com/gfreire57/ic_engines_CNN_fault_detection.

CAPÍTULO 4 – RESULTADOS E DISCUSSÃO

4.1 Rede CNN-STFT

O script para geração das imagens com sinais transformados por STFT obteve um tempo de execução relativamente rápido, gerando as 3500 imagens em cerca de 10 minutos. O treino também foi executado de maneira relativamente rápida, demorando em média de 4 a 5 minutos.

Para avaliar os resultados obtidos a partir da metodologia descrita, são apresentados na figura 4.1 os sinais de vibração transformados pela STFT para cada tipo de condição estudado e para cada nível de ruído. Nota-se nessa figura a pouca sensibilidade desta transformada ao ruído, sendo possível notar alguma diferença somente após o nível de 40%. Além disso, é possível notar que houve pouca definição nos espectrogramas no eixo de frequência mesmo variando o valor da janela de busca. Isso é uma característica dessa transformada, a qual precisa fazer uma troca de resolução entre tempo e frequência, as vezes não definindo tão bem ambos (AL-BADOUR; SUNAR; CHEDED, 2011).

Na figura 4.2, destaca-se o conjunto de matriz confusão para cada nível de ruído. Aqui, aplicou-se 25% dos dados para avaliação da rede, totalizando 875 figuras (divididas aproximadamente de acordo com a proporção da quantidade de imagens entre cada classe). É possível notar uma boa performance no geral, principalmente até os 20%, obtendo uma acurácia de 96,5% para sinais sem ruído, 81,1% para 10% de ruído e 75,1% para 20% de ruído. Entretanto, vale observar uma dificuldade pontual da rede em classificar as figuras entre redução de combustível e normal, havendo muitas previsões falsas entre essas duas classes. Com alto ruído essas dificuldades se tornaram muito mais evidentes, com acurácias menores que 72%, chegando a 57% com 80% de ruído.

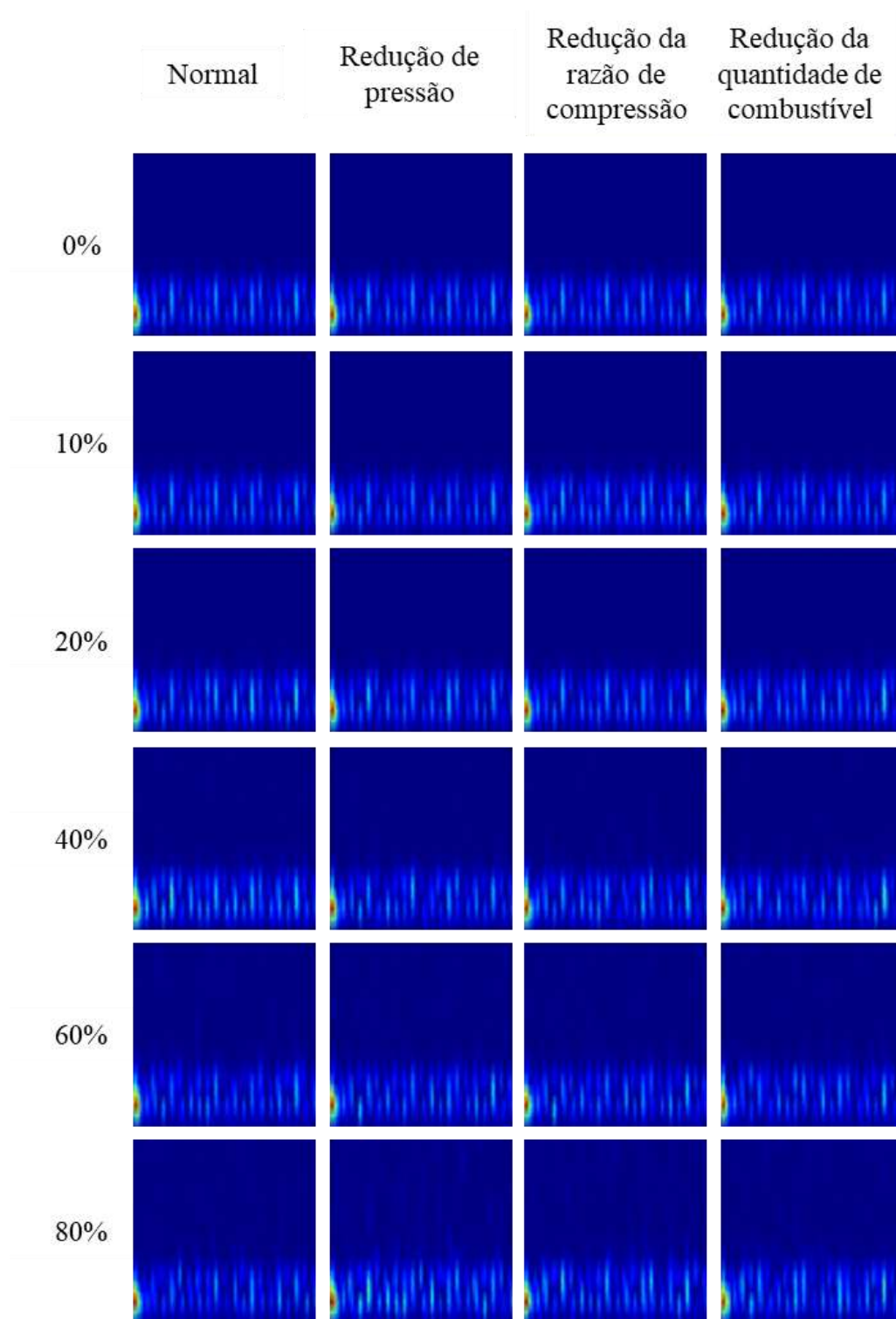


Figura 4.1 – Matriz com imagens para transformada STFT para cada condição e cada nível de ruído.

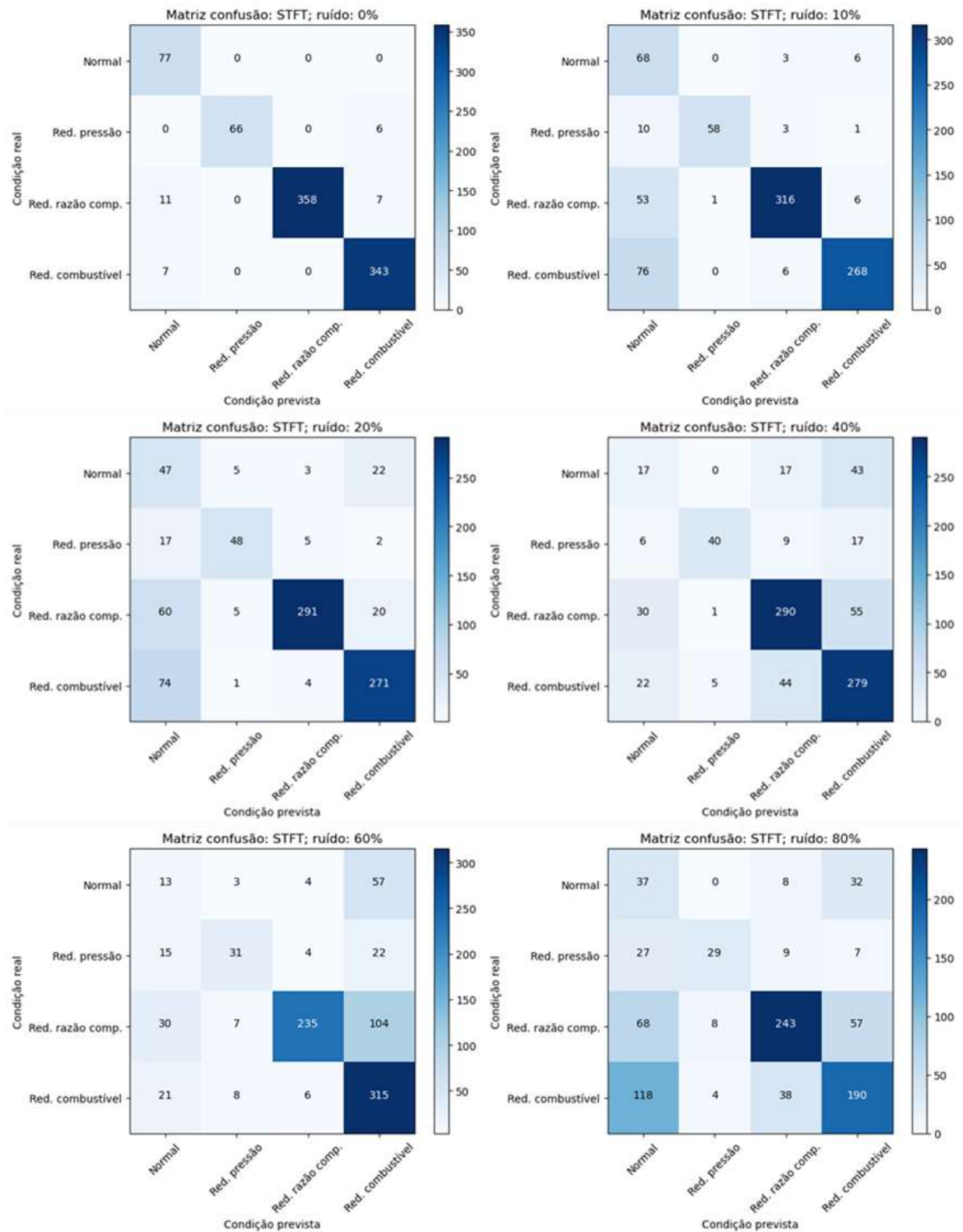


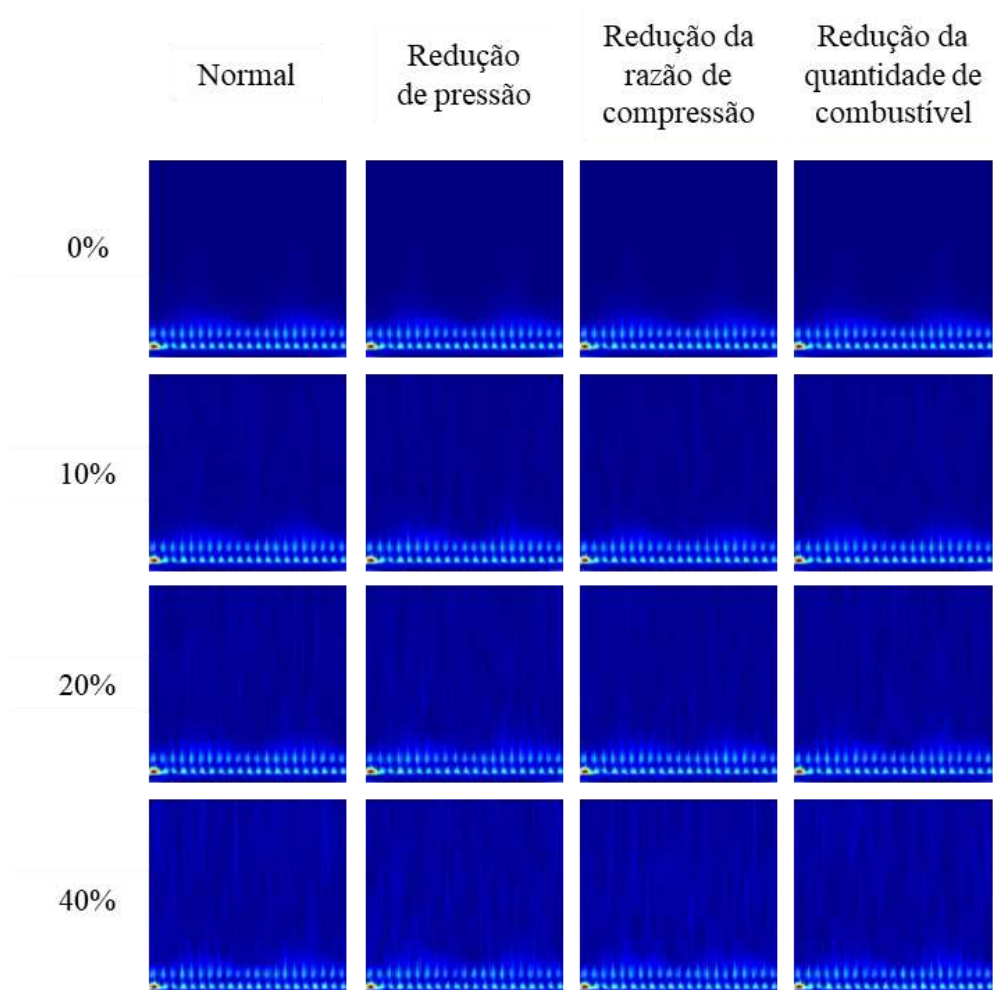
Figura 4.2 – Conjunto de matriz de confusão para STFT para cada caso de ruído.

4.2 Rede CNN-CWT

Ao contrário das imagens geradas por STFT, utilizando a transformada CWT com os sinais o tempo de execução foi bem mais demorado, demorando várias horas para gerar as 3500 imagens. O treino também foi mais demorado, levando cerca de 30 minutos.

Para avaliar os resultados obtidos a partir da metodologia descrita, são apresentados na figura 4.3 os sinais de vibração transformados pela CWT para cada tipo de condição estudado e para cada nível de ruído. Neste caso, já é perceptível o efeito a aplicação de ruído já em 10%. Em níveis maiores, o efeito se torna bastante intenso. Isso faz sentido, visto que esse método é mais sensível a variações de frequência ao longo do tempo.

Na figura 4.4, da mesma forma ao que foi apresentado anteriormente, destaca-se o conjunto de matriz confusão para cada nível de ruído. Foi aplicado para o mesmo número de imagens de teste. Fica clara uma performance geral melhor que a aplicação de STFT em todos os casos. Entretanto, percebeu-se novamente uma certa dificuldade de separação entre as classes normal e redução de combustível.



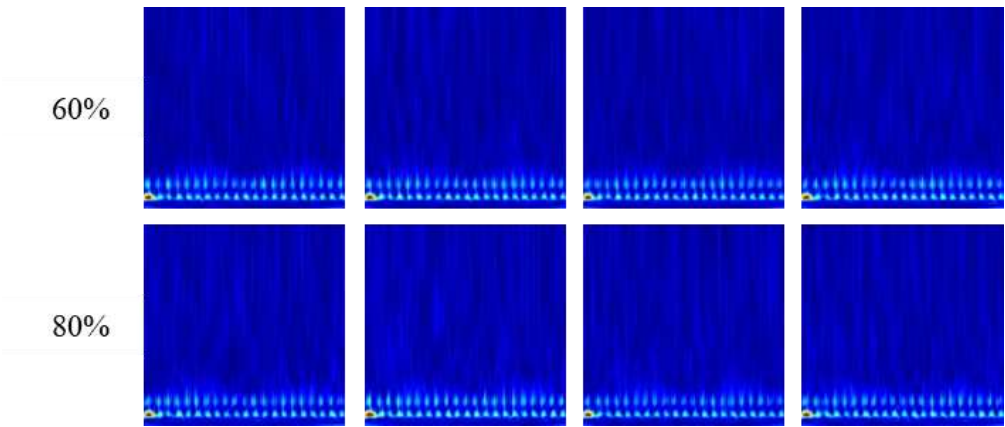
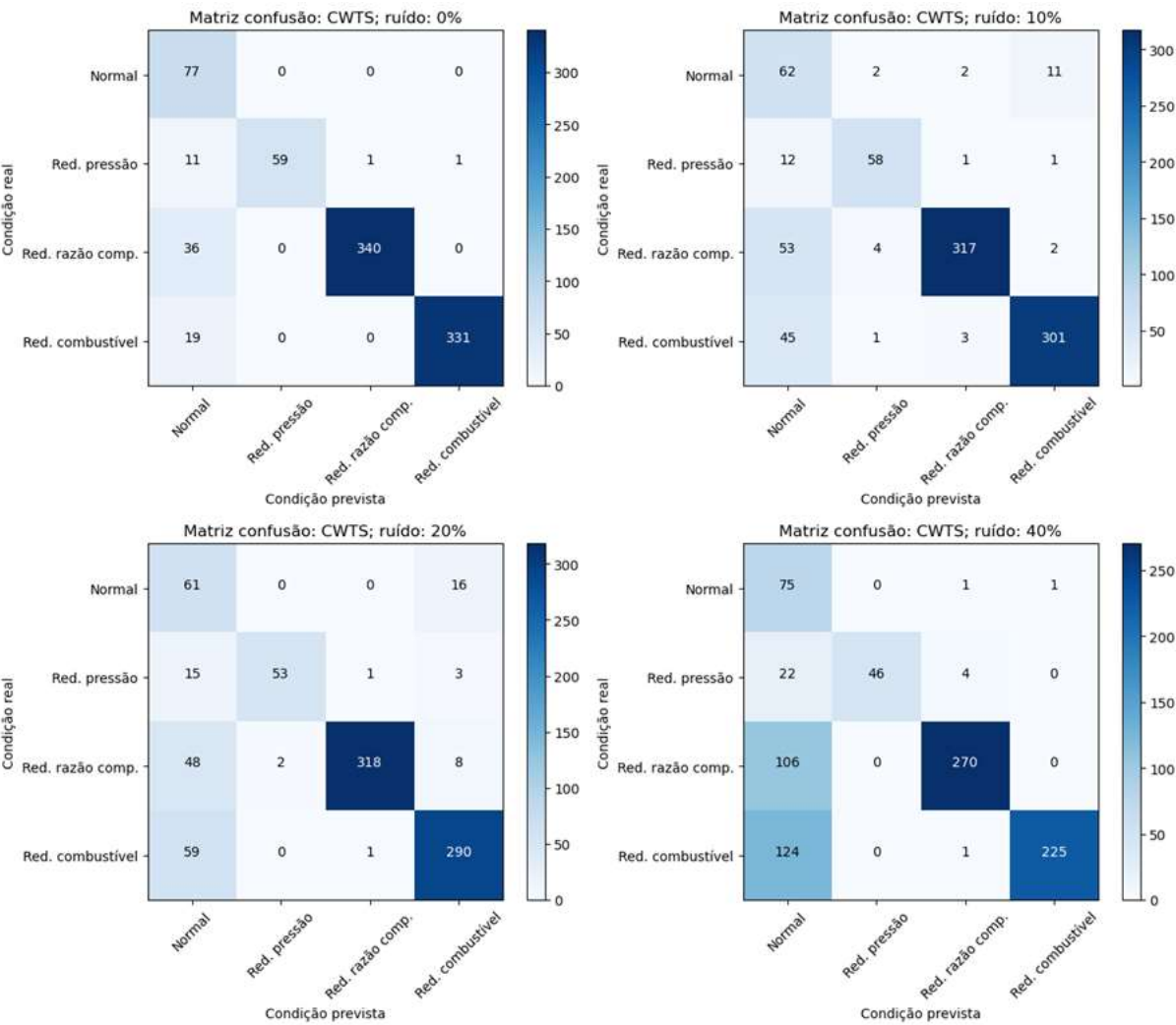


Figura 4.3 - Matriz com imagens para transformada Wavelet (CWT) para cada condição e cada nível de ruído.



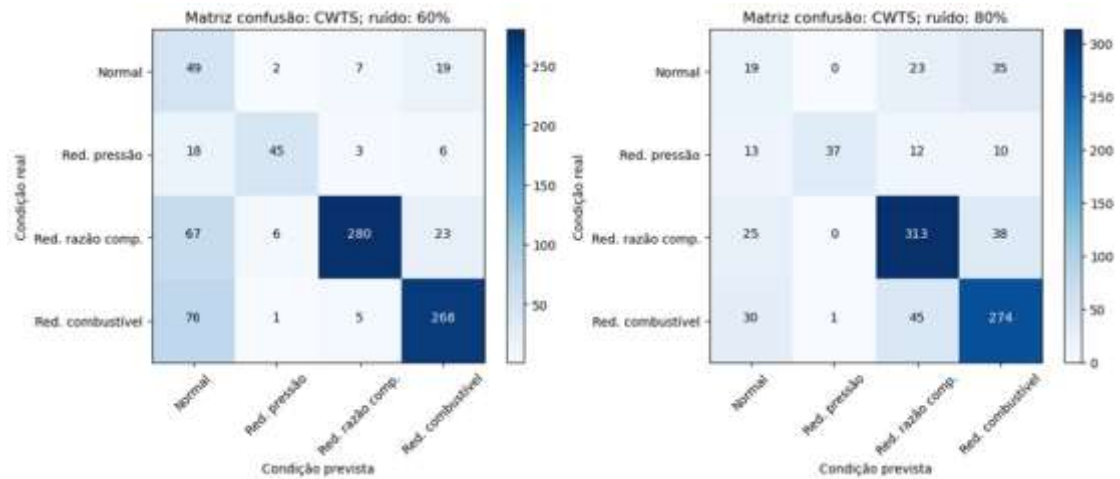


Figura 4.4 - Matrizes de confusão para STFT para cada caso de ruído

4.3 Comparação dos resultados

Das figuras 4.5 a 4.8, é possível ver o output das primeiras camadas convolucional, de *pooling* e com o *dropout* tanto para a rede com STFT quanto com CWT para cada caso dentre as classes com ruído de 20% (para exemplificar). Observa-se a filtragem das características e a escolha daquelas mais relevantes, destacadas nas figuras pelas cores mais intensas. Além disso, vale notar a saída da camada de *max pooling*, a qual seleciona as características mais intensas.

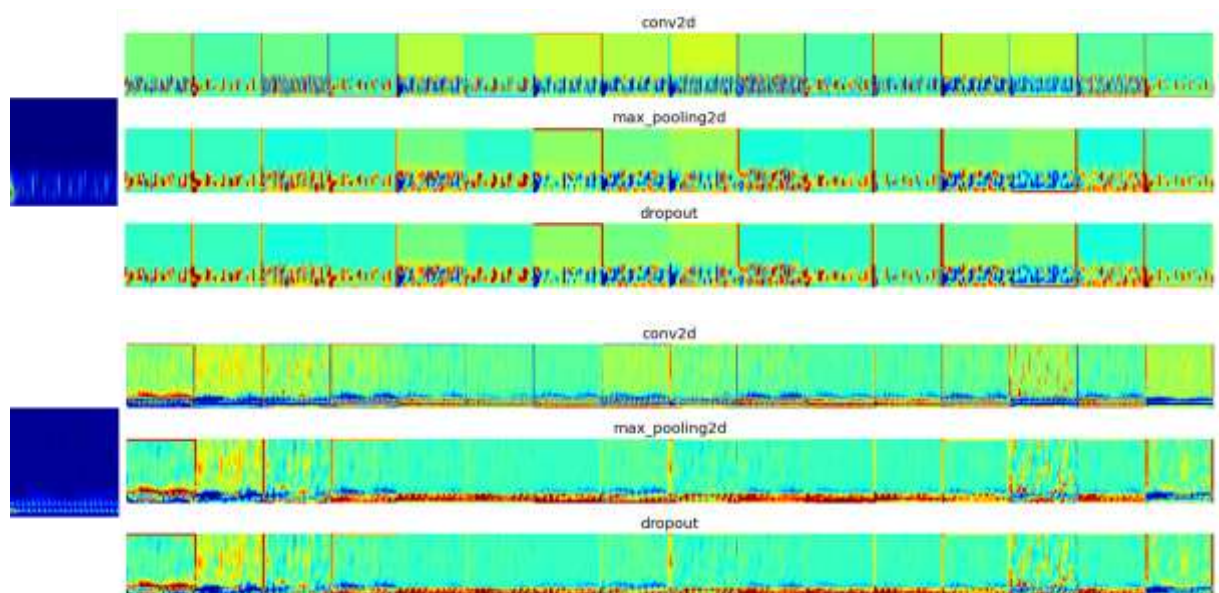


Figura 4.5 – Condição normal; STFT acima, CWT abaixo.

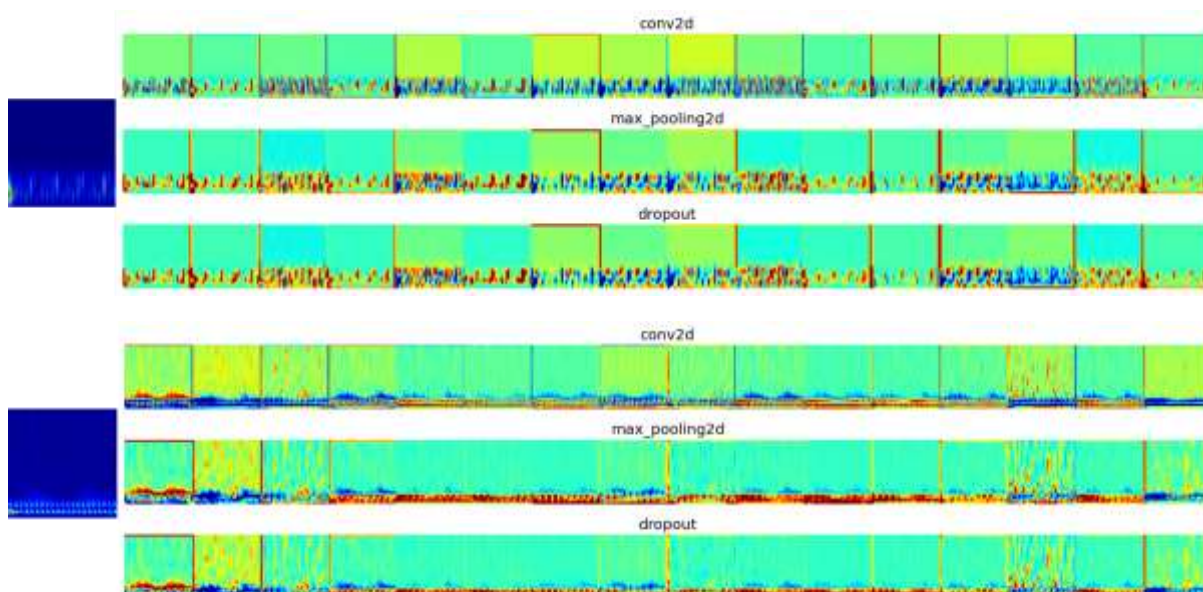


Figura 4.6 – Redução de pressão no *manifold*; STFT acima, CWT abaixo.

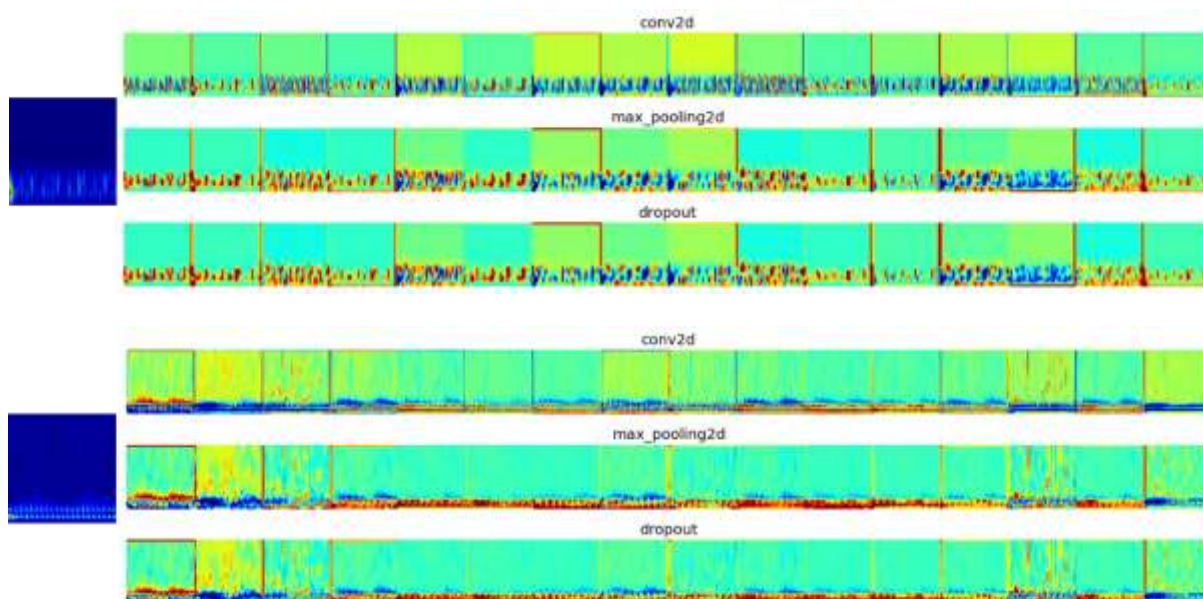


Figura 4.7 - Redução mássica de combustível; STFT acima, CWT abaixo.

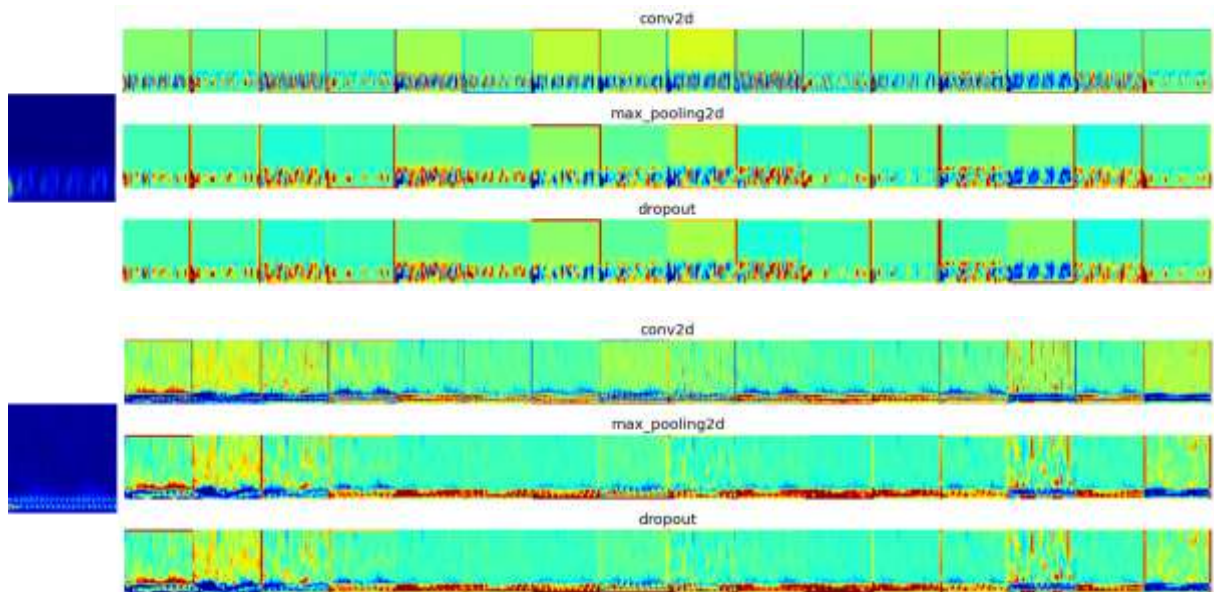


Figura 4.8 – Redução da razão de compressão em um cilindro; STFT acima, CWT abaixo.

Na tabela 2, é possível observar os valores detalhados de acurácia para cada transformada separados para cada nível de ruído. Nota-se que em ambos os casos utilizando cada transformada os valores seguiram aproximadamente lado a lado até 40% de ruído, havendo em seguida uma brusca queda ao usar a STFT e uma constância ao usar a CWT.

Tabela 2 - Acurácia nos testes para cada transformada nos níveis de ruído estudados

Sinal	CNN STFT	CNN CWT
Base	96,5%	92,2%
10% ruído	81,1%	84,3%
20% ruído	75,1%	82,5%
40% ruído	71,5%	70,4%
60% ruído	67,9%	73,4%
80% ruído	57,0%	73,5%

Como comentado anteriormente, o uso da transformada Wavelet permitiu atingir maiores níveis de acuracidade se comparada à STFT, entretanto exige um tempo muito maior de execução. Portanto, o uso de cada pode depender da forma como se deseja aplicar na prática: por exemplo, a STFT pode ser aplicada em sistemas embarcados para análise em tempo real de um motor a diesel. Já a CWT pode ser utilizada para avaliações periódicas dos equipamentos utilizando computadores mais potentes.

Vale ressaltar, entretanto, que existe uma forma rápida da transformada Wavelet explicada por Beykin, Coifman e Rokhlin (1991) que pode ser explorada como uma alternativa à CWT, podendo portanto resolver o problema de alto custo computacional que

esta transformada exige. Além disso, outro caminho viável é combinar mais de uma transformada a fim de mesclar as melhores características de ambos os métodos.

CAPÍTULO 5 - CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

5.1 Conclusões

A partir dos resultados obtidos, pode-se dizer que a metodologia proposta para identificação e classificação de falhas em motores a diesel apresenta um bom desempenho geral, alcançando valores de acurácia superiores a 70% para valores médios de ruído. A transformada CWT se destacou em robustez, mantendo boas acurácias mesmo para altos valores de ruído, demonstrando que a rede foi mais bem adaptada para o uso com essa transformada. Entretanto, vale ressaltar que, mesmo utilizando uma função de *tunning* automático de alguns hiperparâmetros, pode ser necessário ampliar os parâmetros avaliados automaticamente para gerar redes mais bem ajustadas para cada situação.

Outro possível motivo para a melhor performance do uso de transformada por *wavelet* pode ser explicado pelo fato de haver mais variações da frequência do sinal principalmente no começo, representado na prática pela partida do motor. Essas variações na frequência ao longo do tempo são mais bem captadas por essa transformada, como já foi descrito pela literatura e demonstrado na prática.

Entretanto, em contrapartida é muito importante ressaltar que o uso da transformada CWT exige um poder computacional muito mais extenso, o que reflete também em um tempo muitas vezes maior para realizar o processamento de sinal. Isso pode refletir em seu uso prático, visto que muitas vezes não é possível ter acesso a um supercomputador em campo e, portanto, o seu uso fica totalmente inviável. Assim, nesses casos, a STFT tem grande vantagem, podendo ser utilizada em dispositivos mais simples e com menor custo. Dessa forma, caso haja acesso a bom poder computacional, a CWT pode ser a melhor opção; no caso oposto, caso o acesso for mais restrito, a STFT pode ainda ser utilizada com um bom nível de confiabilidade.

Portanto, é válido dizer que há potencial em aplicar o método de identificação proposto neste trabalho em problemas reais, visto que há diversas formas de se captar a vibração

torcional de um motor, inclusive várias delas com baixo custo (como uso de extensômetros). Além disso, com um mesmo setup é possível expandir o número de classes de falhas estudados visto que a vibração torcional pode captar outros problemas, principalmente estruturais, como observado neste trabalho e em outras fontes na literatura com propostas no mesmo campo.

5.2 Sugestões para Trabalhos Futuros

Como sugestões para próximos trabalhos, melhorias e orientações para validação dos métodos e resultados apresentados, sugere-se alguns pontos, como:

- Utilizar valores experimentais de vibração torcional para as falhas e rede usados nesse trabalho;
- Testar outras configurações de rede mais simples para analisar em tempo real as condições do motor;
- Explorar mais as funções de *tunning* de hiperparâmetros de maneira mais complexa para buscar redes com melhores *fits*;
- Ampliar os tipos de falhas que podem ocorrer com motores a combustão e avaliá-las com o método proposto.
- Testar a rede e as transformadas propostas com outros tipos de sinais, como térmicos e acústicos, simultaneamente com os dados utilizados neste trabalho;

REFERÊNCIAS BIBLIOGRÁFICAS

AL-BADOUR, F.; SUNAR, M.; CHEDED, L. Vibration analysis of rotating machinery using time–frequency analysis and wavelet techniques. **Mechanical Systems and Signal Processing**, v. 25, n. 6, p. 2083–2101, ago. 2011. <https://doi.org/10.1016/j.ymssp.2011.01.017>.

BEYLKIN, G.; COIFMAN, R.; ROKHLIN, V. Fast wavelet transforms and numerical algorithms I. **Communications on Pure and Applied Mathematics**, v. 44, n. 2, p. 141–183, mar. 1991. <https://doi.org/10.1002/cpa.3160440202>.

BRUNTON, Steven L.; KUTZ, J. Nathan. **Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control**. 1. ed. [S. l.]: Cambridge University Press, 2019. DOI 10.1017/9781108380690. Disponível em: <https://www.cambridge.org/core/product/identifier/9781108380690/type/book>. Acesso em: 18 nov. 2023.

FIRMINO, João L.; NETO, João M.; OLIVEIRA, Andersson G.; SILVA, José C.; MISHINA, Koje V.; RODRIGUES, Marcelo C. Misfire detection of an internal combustion engine based on vibration and acoustic analysis. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, v. 43, n. 7, p. 336, jul. 2021. <https://doi.org/10.1007/s40430-021-03052-y>.

GUTIÉRREZ, Ricardo H. R.; BELCHIOR, Carlos R. P.; VAZ, Luiz A.; MONTEIRO, Ulisses A. Diagnostic methodology in four-stroke marine diesel engine by identifying operational parameters. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, v. 40, n. 10, p. 500, out. 2018. <https://doi.org/10.1007/s40430-018-1416-x>.

HOWLETT, Robert; DE ZOYSA, Malindi; WALTERS, Simon; HOWSON, Peter. **Neural Networks Techniques for Monitoring and Control of Internal Combustion Engines**. [S. l.: s. n.], 1999.

JING, Luyang; ZHAO, Ming; LI, Pin; XU, Xiaoqiang. A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox. **Measurement**, v. 111, p. 1–10, dez. 2017. <https://doi.org/10.1016/j.measurement.2017.07.017>.

JONDRALE, Friedrich K. White Gaussian Noise – Models for Engineers. **Frequenz**, v. 72, n. 5–6, p. 293–299, 25 abr. 2018. <https://doi.org/10.1515/freq-2017-0064>.

KIRANYAZ, Serkan; AVCI, Onur; ABDELJABER, Osama; INCE, Turker; GABBOUJ, Moncef; INMAN, Daniel J. 1D convolutional neural networks and applications: A survey. **Mechanical Systems and Signal Processing**, v. 151, p. 107398, abr. 2021. <https://doi.org/10.1016/j.ymssp.2020.107398>.

KRISTEKOVA, M.; KRISTEK, J.; MOCZO, P.; DAY, S. M. Misfit Criteria for Quantitative Comparison of Seismograms. **Bulletin of the Seismological Society of America**, v. 96, n. 5, p. 1836–1850, 1 out. 2006. <https://doi.org/10.1785/0120060012>.

MENDES, A S; MEIRELLES, P S; ZAMPIERI, D E. Analysis of torsional vibration in internal combustion engines: Modelling and experimental validation. **Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics**, v. 222, n. 2, p. 155–178, 1 jun. 2008. <https://doi.org/10.1243/14644193JMBD126>.

OPPENHEIM, Alan V.; SCHAFER, Ronald W. **Discrete-time signal processing**. 3. ed., internat. ed., [Nachdr.]. Upper Saddle River Munich: Pearson Education, 2010.

PESTANA, Denys. **Diesel Engine Faults Features Dataset (3500-DEFault)**. [S. l.]: Mendeley, 29 abr. 2020. DOI 10.17632/K22ZXZ29KR.1. Disponível em: <https://data.mendeley.com/datasets/k22zxz29kr/1>. Acesso em: 18 nov. 2023.

PINEDO-SÁNCHEZ, Luis A.; MERCADO-RAVELL, Diego A.; CARBALLO-MONSIVAIS, Carlos A. Vibration analysis in bearings for failure prevention using CNN. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, v. 42, n. 12, p. 628, nov. 2020. <https://doi.org/10.1007/s40430-020-02711-w>.

RIBEIRO JUNIOR, Ronny Francis; DOS SANTOS AREIAS, Isac Antônio; CAMPOS, Mateus Mendes; TEIXEIRA, Carlos Eduardo; DA SILVA, Luiz Eduardo Borges; GOMES, Guilherme Ferreira. Fault Detection and Diagnosis in Electric Motors Using Convolution Neural Network and Short-Time Fourier Transform. **Journal of Vibration Engineering & Technologies**, v. 10, n. 7, p. 2531–2542, out. 2022. <https://doi.org/10.1007/s42417-022-00501-3>.

SHAHID, Syed Maaz; KO, Sunghoon; KWON, Sungoh. Real-time abnormality detection and classification in diesel engine operations with convolutional neural network. **Expert Systems with Applications**, v. 192, p. 116233, abr. 2022. <https://doi.org/10.1016/j.eswa.2021.116233>.

SHATNAWI, Yousef; AL-KHASSAWENEH, Mahmood. Fault Diagnosis in Internal Combustion Engines Using Extension Neural Network. **IEEE Transactions on Industrial Electronics**, v. 61, n. 3, p. 1434–1443, mar. 2014. <https://doi.org/10.1109/TIE.2013.2261033>.

STERKENBURG, Ronald. **Aircraft Maintenance & Repair**. 8th Edition. New York: McGraw-Hill Education, 2019. Disponível em: <https://www.accessengineeringlibrary.com/content/book/9781260441055>.

SUTTON, Richard S.; BARTO, Andrew. **Reinforcement learning: an introduction**. Nachdruck. Cambridge, Massachusetts: The MIT Press, 2014(Adaptive computation and machine learning).

SZANDAŁA, Tomasz. Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks. In: BHOI, Akash Kumar; MALLICK, Pradeep Kumar; LIU, Chuan-Ming; BALAS, Valentina E. (orgs.). **Bio-inspired Neurocomputing**. Studies in Computational Intelligence. Singapore: Springer Singapore, 2021. v. 903, p. 203–224. DOI 10.1007/978-981-15-5495-7_11. Disponível em: http://link.springer.com/10.1007/978-981-15-5495-7_11. Acesso em: 18 nov. 2023.

TORSIONAL VIBRATION: WHAT IS IT? 2014. Disponível em: <https://community.sw.siemens.com/s/article/torsional-vibration-what-is-it>. Acesso em: 19 nov. 2023.

VIANA, Denys P.; DE SÁ SÓ MARTINS, Dionísio H. C.; DE LIMA, Amaro A.; SILVA, Fabrício; PINTO, Milena F.; GUTIÉRREZ, Ricardo H. R.; MONTEIRO, Ulisses A.; VAZ, Luiz A.; PREGO, Thiago; ANDRADE, Fabio A. A.; TARRATACA, Luís; HADDAD, Diego B. Diesel Engine Fault Prediction Using Artificial Intelligence Regression Methods. **Machines**, v. 11, n. 5, p. 530, 5 maio 2023. <https://doi.org/10.3390/machines11050530>.

WANG, Jay; TURKO, Robert; SHAIKH, Omar; PARK, Haekyu; DAS, Nilaksh; HOHMAN, Fred; KAHNG, Minsuk; CHAU, Polo. CNN Explainer - Learn Convolutional Neural Network (CNN) in your browser! 30 out. 2023. **CNN Explainer - Learn Convolutional Neural Network (CNN) in your browser!** Disponível em: <https://poloclub.github.io/cnn-explainer/>. Acesso em: 30 out. 2023.

WANG, M.H.; HUNG, C.P. Extension neural network and its applications. **Neural Networks**, v. 16, n. 5–6, p. 779–784, jun. 2003. [https://doi.org/10.1016/S0893-6080\(03\)00104-7](https://doi.org/10.1016/S0893-6080(03)00104-7).

WANG, P.; DAVIES, P.; STARKEY, J.M.; ROUTSON, R.L. A torsional vibration measurement system. **IEEE Transactions on Instrumentation and Measurement**, v. 41, n. 6, p. 803–807, dez. 1992. <https://doi.org/10.1109/19.199411>.

WU, Jian-Da; CHEN, Jien-Chen. Continuous wavelet transform technique for fault signal diagnosis of internal combustion engines. **NDT & E International**, v. 39, n. 4, p. 304–311, jun. 2006. <https://doi.org/10.1016/j.ndteint.2005.09.002>.

XUE, Song; HOWARD, Ian. Torsional vibration signal analysis as a diagnostic tool for planetary gear fault detection. **Mechanical Systems and Signal Processing**, v. 100, p. 706–728, fev. 2018. <https://doi.org/10.1016/j.ymssp.2017.07.038>.

ZHU, Xiaobin; CAI, Zun; WU, Jianjun; CHENG, Yuqiang; HUANG, Qiang. Convolutional neural network based combustion mode classification for condition monitoring in the supersonic combustor. **Acta Astronautica**, v. 159, p. 349–357, jun. 2019. <https://doi.org/10.1016/j.actaastro.2019.03.072>.