Universidade Estadual Paulista Júlio de Mesquita Filho
Faculdade de Engenharia de Ilha Solteira

VINÍCIUS DE ARAÚJO SALMAZO

Redes Neurais Convolucionais na Análise de Vibrações para Identificação de Falhas de Ignição em Motores Veiculares a Combustão

Ilha Solteira, SP
2022

VINÍCIUS DE ARAÚJO SALMAZO

Redes Neurais Convolucionais na Análise de Vibrações para Identificação de Falhas de Ignição em Motores Veiculares a Combustão

Dissertação apresentada à Faculdade de Engenharia de Ilha Solteira - UNESP, programa de Pós-Graduação em Engenharia Mecânica, como requisito para obtenção do título de Mestre em Engenharia Mecânica.

Orientador: Prof. Dr. Amarildo Tabone Paschoalini

Ilha Solteira, SP
2022

FICHA CATALOGRÁFICA
Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

Salmazo, Vinícius de Araújo.
S171r Redes neurais convolucionais na análise de vibrações para identificação de falhas de ignição em motores veiculares a combustão / Vinícius de Araújo Salmazo. -- Ilha Solteira: [s.n.], 2022
67 f. : il.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Mecânica dos Sólidos, 2022

Orientador: Amarildo Tabone Paschoalini
Inclui bibliografia

1. Inteligência artificial. 2. Redes neurais convolucionais. 3. Motor de combustão interna. 4. Falha de ignição.

João Josué Barbosa
Serviço Técnico de Biblioteca e Documentação
Diretor Técnico
CRB 8-5642

UNIVERSIDADE ESTADUAL PAULISTA

Câmpus de Ilha Solteira

CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: Redes Neurais Convolucionais na análise de vibrações para identificação de falhas de ignição em motores veiculares a combustão

AUTOR: VINÍCIUS DE ARAÚJO SALMAZO
ORIENTADOR: AMARILDO TABONE PASCHOALINI

Aprovado como parte das exigências para obtenção do Título de Mestre em Engenharia Mecânica, área: Mecânica dos Sólidos pela Comissão Examinadora:

Prof. Dr. AMARILDO TABONE PASCHOALINI (Participaçao Virtual)
Departamento de Engenharia Mecânica / Faculdade de Engenharia de Ilha Solteira - UNESP

Prof. Dr. EMANUEL ROCHA WOISKI (Participaçao Virtual)
Departamento de Engenharia Mecânica / Faculdade de Engenharia de Ilha Solteira - UNESP

Prof. Dr. AIRTON NABARRETE (Participaçao Virtual)
Departamento de Engenharia Aeroespacial e Aeronáutica / Instituto Tecnológico de Aeronáutica

Ilha Solteira, 01 de setembro de 2022

Dedico este trabalho aos meus pais, à minha irmã e à minha namorada.

AGRADECIMENTOS

Primeiramente, agradeço a meus pais, que sempre me apoiaram e auxiliaram em todas as etapas até aqui, proporcionando as melhores condições possíveis para minha formação. Agradeço muito por todo amor, respeito, confiança, carinho e atenção dedicados a mim.

Agradeço a minha irmã, que sempre me escutou em momentos de anseio e dúvidas, me deu conselhos e me apoiou em diversas decisões. Sua alegria e amizade sempre me deram força e energia para continuar.

A minha namorada, agradeço por toda compreensão, apoio e carinho dados. Obrigado por estar sempre presente nos momentos alegres e difíceis. Não tenha dúvida de que seu auxílio foi fundamental para que eu pudesse superar todos os obstáculos encontrados.

Ao meu orientador, por todo apoio e acolhimento prestado desde a época da graduação, com o estágio acadêmico. Sou extremamente grato pelas oportunidades e aprendizados que tem me proporcionado ao longo desses anos.

Aos professores e técnicos do Programa de Pós-Graduação em Engenharia Mecânica da UNESP de Ilha Solteira pelos ensinamentos passados.

Aos companheiros e amigos de curso e morada que tive ao longo de todos esses anos que, de alguma forma, sempre me auxiliaram nessa caminhada.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 88887.481000/2020-0.

"Obstáculos são aquelas coisas assustadoras que vemos quando desviamos o foco do nosso objetivo."

Henry Ford

RESUMO

A manutenção é uma atividade que sempre existiu e que com o passar do tempo adquiriu novas dimensões, estando em constante processo de desenvolvimento. No atual cenário, as tecnologias têm evoluído rapidamente, viabilizando estudos e aplicações de novas abordagens para identificação e predição de falhas em sistemas mecânicos. Dentre as diversas invenções tecnológicas desenvolvidas pelo homem, o motor de combustão interna (MCI) se destaca pelos avanços que trouxe para a sociedade moderna. Uma de suas possíveis falhas é a falha de ignição, a qual possui diversas causas e pode ser crítica em relação ao aumento nas emissões de poluentes e perda de desempenho do motor. Trabalhos recentes têm abordado diferentes métodos para identificação da ocorrência desse tipo de falha, alguns empregando técnicas de Inteligência Artificial e Aprendizado de Máquina. Este trabalho tem como objetivo avaliar a aplicação de Redes Neurais Convolucionais (RNC) na identificação de falhas de ignição em MCIs. Para isso, um conjunto de aquisição de baixo custo foi confeccionado empregando um acelerômetro MEMS capacitivo. Sinais de vibração foram coletados a partir de dois MCIs similares de quatro cursos e ignição por centelha operando em quatro condições distintas. Os dados foram preparados no domínio da frequência em duas formas de entrada: utilizando gráficos e em formato de matrizes compactas das amplitudes. Hiperparâmetros (HPs) foram definidos para a RNC utilizando gráficos como entrada enquanto que para a RNC com dados em forma de matrizes alguns HPs foram selecionados por *gridsearch*. Treinando as RNCs até a convergência com dados do primeiro motor, acurácias superiores a 99% foram alcançadas para as duas redes classificando dados do banco de teste do mesmo motor. Ao classificar dados do segundo motor, o desempenho geral ficou entre 68% e 72%. Em relação ao tempo de classificação, a rede utilizando dados em formato compacto apresentou desempenho sete vezes superior, rotulando dados individuas em um tempo médio inferior a $418\mu s$. Os resultados demonstram grande potencial de aplicação das RNCs em conjunto com o sistema desenvolvido para a tarefa proposta.

Palavras-chave: Inteligência Artificial, Redes Neurais Convolucionais, Motor de Combustão Interna, Falha de Ignição.

ABSTRACT

Maintenance is an activity that has always existed and that over the time it has acquired new dimensions, being in a constant process of development. Currently, technologies have been evolving quickly, enabling studies and applications of new approaches to identify and predict failures in mechanical systems. Among various technological inventions developed by man, the internal combustion engine (ICE) stands out due to the advances that it has brought to modern society. One of its possible failures is the misfire, which has several causes and can be critical in relation to the increase in pollutant emissions and loss of engine performance. Recent works have addressed different methods to identify the occurrence of this type of failure, some of them applying Artificial Intelligence and Machine Learning techniques. This work aims to evaluate the application of a Convolutional Neural Network (CNN) to identify the occurrence of misfires in ICEs. For this, a low cost acquisition set was developed using a capacitive MEMS accelerometer. Vibration signals were collected from two similar four-stroke spark-ignition ICEs operating under four different conditions. The data were prepared in the frequency domain for two types of input: using graphs and in a compact matrix of amplitudes format. The hyperparameters (HPs) were manually defined for the CNN using graphs as input while for the CNN with matrix data some HPs were selected by gridsearch. By training the CNNs till the convergence with data from the first engine, accuracies greater than 99% were achieved for both networks classifying data from the test dataset of the same engine. When sorting data from the second engine, the overall performance was between 68% and 72%. Regarding the classification time, the network using data in the compact format presented a seven times better performance, labeling individual data in an average time of less than $418\mu s$. The results demonstrate great potential for the application of CNNs with the developed system for the proposed task.

Keywords: Artificial Intelligence, Convolutional Neural Networks, Internal Combustion Engine, Misfire.

LISTA DE FIGURAS

Figura 1 – Motor de combustão interna e alguns de seus componentes. 21
Figura 2 – Diagrama esquemático de um acelerômetro MEMS uniaxial capacitivo. 24
Figura 3 – Relação entre os termos subgrupos da Inteligência Artificial. 25
Figura 4 – Rede Neural Artificial completamente conectada com duas camadas ocultas. 27
Figura 5 – Convolução de uma matriz 5x5 com um *kernel* 3x3. 31
Figura 6 – Aplicação de três filtros 3x3 em uma camada de convolução. 32
Figura 7 – Módulo MPU6050 com acelerômetro e giroscópio tipo MEMS. . . . 34
Figura 8 – Diagrama esquemático das conexões entre os componentes do circuito eletrônico. 36
Figura 9 – Posicionamento do acelerômetro em relação ao motor. 38
Figura 10 – Posicionamento do acelerômetro no motor do segundo veículo. . . . 40
Figura 11 – Sinal janelado no domínio do tempo e da frequência. 41
Figura 12 – Ilustração de um dado de entrada em formato de imagem, originalmente em preto e branco, para treinamento da rede. 42
Figura 13 – Ilustração da topologia da rede utilizada. 44
Figura 14 – Espectrograma do sinal de aceleração utilizando o acelerômetro MEMS para o motor do primeiro veículo. 48
Figura 15 – Comparação do espectrograma com a rotação real do motor obtida pela ECU do primeiro veículo. 49
Figura 16 – Espectrograma do sinal de aceleração utilizando o acelerômetro MEMS para o motor do segundo veículo. 49
Figura 17 – Comparação do espectrograma com a rotação real do motor obtida pela ECU do segundo veículo. 49
Figura 18 – Curva de Perda durante treinamento da RNC com entradas em formato de imagens. 50
Figura 19 – Curva de Acurácia durante treinamento da RNC com entradas em formato de imagens. 51
Figura 20 – Visualização dos Mapas de Características na saída da primeira camada de convolução da RNC com entrada de imagens. 52
Figura 21 – Matriz de confusão da RNC treinada com imagens avaliando os dados de teste. 53
Figura 22 – Curva de Perda durante treinamento final da RNC com entradas em formato compacto. 55
Figura 23 – Curva de Acurácia durante treinamento final da RNC com entradas em formato compacto. 55

Figura 24 – Matriz de confusão da RNC treinada com matrizes avaliando os dados de teste. 57
Figura 25 – Matriz de confusão da RNC treinada com imagens avaliando os dados do segundo motor. 58
Figura 26 – Matriz de confusão da RNC treinada com matrizes avaliando os dados do segundo motor. 59

LISTA DE TABELAS

Tabela 1 – Características da placa GY-521 contendo o MPU6050. 34
Tabela 2 – Características da plataforma de desenvolvimento *ESP32* da fabricante *Espressif Systems*. 35
Tabela 3 – Características dos motores veiculares utilizados. 37
Tabela 4 – Condições possíveis para simulação das falhas desejadas. 39
Tabela 5 – Estrutura da RNC com entradas em formato de imagem. 45
Tabela 6 – Espaço de Hiperparâmetros Testados. 46
Tabela 7 – Melhores Hiperparâmetros definidos para a rede por *Gridsearch*. . . 54
Tabela 8 – Estrutura final da RNC com entradas em formato compacto. 54

LISTA DE ABREVIATURAS E SIGLAS

CAN	*Controller Area Network*
DOF	*Degrees of Freedom*
ECU	*Engine Control Unit*
FFT	*Fast Fourier Transform*
GPIO	*General Purpose Input/Output*
HP	Hiperparâmetro
IA	Inteligência Artificial
MEMS	*Micro-Electro-Mechanical Systems*
MCI	Motor de Combustão Interna
MCU	*Microcontroller Unit*
OBD	*On Board Diagnostics*
ReLU	*Rectified Linear Unit*
RNA	Rede Neural Artificial
RNC	Rede Neural Convolucional
SOHC	*Single Over Head Camshaft*

SUMÁRIO

1 INTRODUÇÃO 15
1.1 Motivação . . . 17
1.2 Objetivos . . . 19
1.3 Organização do Trabalho . . . 19

2 REVISÃO BIBLIOGRÁFICA 21
2.1 Motor de Combustão Interna . . . 21
2.1.1 Falhas de Ignição . . . 22
2.2 Acelerômetros MEMS . . . 23
2.3 Inteligência Artificial . . . 25
2.3.1 Redes Neurais Artificiais . . . 26
2.3.2 Redes Neurais Convolucionais . . . 30

3 MATERIAIS E MÉTODOS 33
3.1 Conjunto Experimental . . . 33
3.1.1 Critérios de Seleção . . . 33
3.1.2 Acelerômetro MEMS . . . 34
3.1.3 Microcontrolador . . . 35
3.1.4 Circuito Eletrônico . . . 35
3.1.5 Motor de Combustão Interna . . . 36
3.1.6 Adaptador OBD-II . . . 37
3.2 Coleta dos Dados Experimentais . . . 37
3.3 Preparação dos Dados . . . 40
3.3.1 Entrada em Forma de Imagens . . . 42
3.3.2 Entrada em Forma de Matriz de Amplitudes . . . 43
3.4 Estrutura das Redes Neurais Convolucionais e Treinamento . . . 43
3.4.1 RNC Classificando Imagens . . . 44
3.4.2 RNC Classificando Dados Compactos . . . 46
3.4.3 Ambiente de Desenvolvimento . . . 47

4 RESULTADOS E DISCUSSÕES 48
4.1 Correspondência dos Dados Coletados . . . 48
4.2 Treinamento das RNCs e Classificação . . . 50
4.2.1 RNC com Entrada de Gráficos . . . 50
4.2.2 RNC com Entrada por Matrizes . . . 54
4.3 Classificação de Falhas do Segundo Motor . . . 58

5 CONCLUSÕES 61

5.1 Sugestões para Trabalhos Futuros 62

REFERÊNCIAS **63**

1 INTRODUÇÃO

Por volta do século XVI, o termo "manutenção" começou a ser conhecido e utilizado, mesmo sendo uma atividade que sempre existiu, embora despercebida (MORO; AURAS, 2007). Contudo, a evolução da manutenção começa a acontecer quando a própria indústria passa por mudanças significativas e tem a necessidade de uma melhor organização e planejamento. Dessa forma, a partir de 1930, classificam-se quatro diferentes gerações da evolução da manutenção.

A Primeira Geração é caracterizada por ser, fundamentalmente, corretiva e não planejada, devido a seu contexto histórico que antecede a Segunda Guerra Mundial, em que a indústria era simples e pouco mecanizada, em que a produtividade não era prioritária. A Segunda Geração abrange, principalmente, as décadas de 1950 a 1970, período em que houve um aumento da mecanização e da complexidade das instalações industriais devido ao aumento da demanda por todos os tipos de produtos após a Segunda Guerra. Dessa forma, a produtividade torna-se importante, surgindo a manutenção preventiva com a finalidade de se manter o bom funcionamento das máquinas, evitando falhas, além de realizar controles de manutenção e sistemas de planejamento (KARDEC; NASCIF, 2001).

A partir da década de 1970, tem-se a Terceira Geração, quando houve o crescimento da automação e mecanização, reforçou-se o conceito e utilização da manutenção preventiva e, em virtude do avanço da informática, sucedeu-se o desenvolvimento de melhores softwares para o planejamento, controle e acompanhamento dos serviços de manutenção. Posteriormente, a Quarta Geração se inicia em meados da década de 1990, e tem como desafio a minimização de "falhas prematuras" ou "falhas de mortalidade infantil" (MOUBRAY, 1996). A prática de análise de falhas torna-se, então, uma metodologia capaz de melhorar a performance dos equipamentos e, consequentemente, da própria empresa, sendo a manutenção preditiva e o monitoramento de condição de equipamentos e de processos cada vez mais empregados na indústria (KARDEC; NASCIF, 2001).

Como resultado de toda evolução mencionada, a manutenção, atualmente, não é mais um problema puramente técnico como no início do século XX, mas sim uma questão de gestão estratégica. Além disso, com os avanços tecnológico como novos materiais, sensores e sistemas de monitoramento, coleta de dados e análises, a manutenção adquiriu novas dimensões, as quais continuam se desenvolvendo constantemente (KOBBACY; MURTHY et al., 2008).

Em 2011, surge o termo Indústria 4.0 na Alemanha, o qual se refere a um novo conceito de desenvolvimento de alta tecnologia para manufatura. Também conhecida por "Internet das Coisas (IoT – Internet of Things)" ou "Internet Industrial", ela fornece uma maneira de obter melhor visibilidade sobre as operações da empresa, por meio da integração de sensores de máquina, softwares e sistemas de armazenamento, trabalhando geralmente com grande volume de dados, mais conhecido como *Big Data* (GILCHRIST,

2016).

Tal aumento na coleta de dados só foi possível graças ao avanço das tecnologias de sensoriamento, estando diretamente relacionado à redução de custos e das dimensões dos sensores, proporcionados pelo desenvolvimento da microeletrônica (GILCHRIST, 2016). Conforme mencionado por Albarbar et al. (2008), o preço médio dos acelerômetros MEMS (*Micro-Electro-Mechanical Systems*) diminuiu, de uma média de US$ 2,50 em 2004 para menos de US$ 1,90 em 2009, possibilitando ainda mais sua aplicação em projetos, além de viabilizar o aumento dos estudos sobre sua construção, princípios de medição e avaliações de desempenho.

Mesmo com o avanço das tecnologias que proporcionam uma maior facilidade na aquisição de dados, tem-se que essa coleta de maneira isolada, de modo geral, não é o suficiente para o desenvolvimento de análises e resoluções de problemas. Em consequência disso, a participação de Inteligência Artificial (IA), técnicas de aprendizado de máquina (*machine learning*) e suas subáreas, têm se mostrado bastante interessantes e eficazes em relação à análise e aplicação desses dados.

Adentrando em assuntos referentes à Inteligência Artificial e sua contextualização, em meados de 1950, seus pioneiros determinaram uma missão extremamente ambiciosa, a qual tinha como principal objetivo a recriação da inteligência humana em uma máquina. No entanto, desde sua criação, a IA tem enfrentado diversos períodos de expansão e retrocesso, devido principalmente aos contextos de cada época e seus principais interesses (LEE, 2019).

Inicialmente, o campo da Inteligência Artificial se dividia em duas abordagens diferentes, a "Baseada em Regras" (também conhecida como "Sistemas Simbólicos" ou "Sistemas de Especialistas") e, a de "Redes Neurais". A primeira abordagem tentava ensinar os computadores a pensar por meio de codificações de regras lógicas, enquanto a segunda realizava o ensinamento tentando reconstruir o próprio cérebro humano, imitando sua arquitetura, construindo camadas de neurônios artificias que recebem e transmitem informações de maneira semelhante à rede biológica (LEE, 2019).

Embora ao longo das décadas de 1950 a 1960, as primeiras versões das redes neurais artificiais tiveram resultados promissores, no decorrer da década de 1970 a abordagem perdeu força e se estagnou, pois esse método acabou se revelando pouco confiável e limitado. Para que o campo das redes neurais artificiais pudesse emergir novamente, seriam necessárias mudanças em seus dois principais componentes: tratamento de dados e aumento significativo do poder computacional. Foi então que nos anos 2000, o pesquisador Geoffrey Hinton produziu o avanço técnico necessário com a descoberta de uma maneira eficiente de se treinarem as camadas nas redes neurais. Com isso, as redes neurais artificiais ganharam repentinamente um grande enfoque, pois pesquisadores e empresas perceberam o potencial que essa inovação apresentava, a qual utiliza o Aprendizado de Máquinas (*Machine Learning*) na resolução de problemas do mundo real, abrangendo desde a interpretação

da linguagem natural e tradução de documentos, a até mesmo sua utilização dentro da indústria como ferramenta de detecção e previsão de falhas (LEE, 2019).

Após uma maior consolidação da Aprendizagem de Máquinas, e mais especificamente das Redes Neurais, observou-se um grande aumento de novas arquiteturas, as quais foram se aprofundando conforme os avanços de hardware computacional também evoluíam. Além disso, surgiram novas maneiras de organizar camadas sucessivas e avanços incrementais nos algoritmos de treinamento. Algumas dessas novas arquiteturas hoje são bem consolidadas na literatura, como as Redes Densas, as Redes Neurais Profundas, Redes Recorrentes, Redes Adversariais, além das Redes Convolucionais (LEIJNEN; VEEN, 2020).

As Redes Neurais Convolucionais (RNCs) são um tipo de arquitetura normalmente profunda, de grande destaque por produzir ótimos resultados em problemas de reconhecimento de padrões e classificação de imagens (PANG et al., 2018; HAYKIN, 2009). Elas exploram correlações espaciais locais, impondo padrões de conectividade local entre neurônios de camadas adjacentes. Além disso, possuem importantes propriedades como, alto grau de invariância à translação, à escala, às inclinações e diversos outros tipos de distorções, as quais são essenciais para o reconhecimento de padrões (HAYKIN, 2009).

Muitos são os trabalhos que envolvem RNCs como ferramenta de desenvolvimento de modelos preditivos (BARINO; BESSA, 2020), detecção de peças defeituosas (ROCHA et al., 2019), monitoramento rápido e preciso (INCE et al., 2016), detecção de falhas em motores e turbinas (INCE et al., 2016; LIU et al., 2018), além de muitas outras aplicações, indicando ser uma abordagem bastante versátil e eficaz.

1.1 Motivação

Dentre as diversas áreas possíveis de aplicação das técnicas de IA para detecção de falhas utilizando análise de vibrações, um cenário muito vasto e interessante é a área veicular. Veículos automotores são desenvolvidos com a integração de diversos sistemas complexos, como conjunto de suspensão, direção, motor, transmissão e freios. Para correta operação, tais sistemas dependem de uma quantidade significativa de variáveis, tanto internas quanto externas, sendo suscetíveis a condições de operação adversas.

Os motores de combustão interna (MCI) utilizam-se da queima de combustível para produção de trabalho mecânico, o qual pode ser utilizado em inúmeras atividades, sendo uma das mais importantes delas, a locomoção de veículos (MARTINS, 2011). Perante, então, às necessidades diárias de deslocamento da população de todas as faixas econômicas e sociais, relaciona-se o uso cada vez mais intenso de veículos automotores com a finalidade de suprir essa carência de maneira confortável e segura (DUARTE; LIBARDI; SÁNCHEZ, 2012). Diante disso, observa-se uma frota de aproximadamente 107,9 milhões de veículos emplacados no Brasil no ano de 2020 (IBGE, 2020), em comparação com uma população de cerca de 211,7 milhões de habitantes, segundo estimativas do IBGE realizadas no

mesmo período.

No Brasil, de acordo com a ANFAVEA (Associação Nacional dos Fabricantes de Veículos Automotores), em 2018, cerca de 13% das emissões de dióxido de carbono eram originárias do transporte, sendo 91% desses, referentes ao rodoviário. Com isso, a regulação brasileira segue referências internacionais e, torna-se cada vez mais restritiva para que haja uma redução progressiva dos limites desses poluentes (ROBERTS; BROOKS; SHIPWAY, 2014; ABEDIN et al., 2016; ANFAVEA, 2021). Dessa forma, para atender os limites impostos pela legislação, algumas tecnologias são introduzidas a fim de reduzir a toxidade de emissão de gases, dentre elas tem-se a aplicação de catalisadores, os quais atualmente são indispensáveis nos automóveis, além de dispositivos para medir e controlar a relação ar/combustível, injeções eletrônicas mais avançadas, válvulas de recirculação dos gases de escape (EGR), entre outros (JUNIOR; SOUZA, 2018).

Dentre inúmeras possíveis irregularidades de um motor a combustão, tem-se a falha de ignição, ou em inglês, *misfire*, definida como a ocorrência de uma combustão anormal ou inexistente dentro dos cilindros do motor. Essa falha é complexa, pois pode ocorrer devido a diversas causas como, ausência de centelhamento, mistura ar-combustível inadequada, compressão deficiente, entre outras. A falha de ignição pode acarretar não apenas em elevada perda de potência do motor e, consequentemente, baixo rendimento com aumento do consumo de combustível, mas também agravamento na emissões de gases poluentes.

Como mencionado anteriormente, a falha de ignição possui diversas causas e, sendo assim, diversos trabalhos encontrados na literatura tentam realizar sua detecção e classificação empregando diferentes métodos. Dentre as abordagens já realizadas, pode-se citar alguns exemplos, como a análise de velocidade angular instantânea do virabrequim (SHAO; WANG; WANG, 2020), análise de pressão instantânea dos gases de escape do motor (CECCARANI; REBOTTINI; BETTINI, 1998) e ainda, a medição e análise do sinal de ionização dentro da câmara de combustão (FAN et al., 2012). Entretanto, a maioria desses procedimentos são considerados pouco viáveis em determinadas situações, devido a seus custos e suas características invasivas.

Em contrapartida, outros artigos na mesma linha, utilizam dados de vibração, coletados a partir de acelerômetros piezoelétricos, para tentar identificar a ocorrência de *misfire*, podendo destacar os trabalhos de Babu, I e Sugumaran (2010) e Mulay, Sugumaran e Devasenapati (2018), os quais utilizam diferentes técnicas de *machine learning* para a classificação de falhas individuais nos cilindros. Ainda utilizando dados de vibração, Bhadane et al. (2020) realizaram uma comparação entre técnicas de aprendizados de máquina, concluindo que a rede neural convolucional empregada obteve um melhor desempenho em relação às outras avaliadas. Dessa forma, abordagens baseadas em aquisição de dados, principalmente de vibrações detectadas por acelerômetros e análises envolvendo aprendizagem de máquinas, estão cada vez mais presentes na área de manutenção.

Assim, como motivação para a realização do presente trabalho, vê-se a oportunidade

de implementação de uma Rede Neural Convolucional como ferramenta para identificação de falhas de ignição em motores veiculares de combustão interna, avaliando-se sua robustez na classificação dessas falhas mesmo com a aplicação de uma abordagem experimental simplificada contando com sensores de baixo custo.

1.2 Objetivos

O objetivo principal do presente trabalho consiste no estudo da aplicação de Redes Neurais Convolucionais (RNC) para a detecção de falhas de ignição em motores de combustão interna com ignição por centelha, em condição de marcha lenta, utilizando dados de aceleração obtidos a partir de acelerômetro MEMS capacitivo de baixo custo.

Como objetivos específicos, têm-se:

- Seleção de componentes e confecção de um conjunto de aquisição de baixo custo para coleta dos dados de aceleração;
- Verificação da viabilidade do sensor MEMS selecionado operando em um ambiente agressivo como o cofre do motor, bem como sua capacidade de monitoramento da velocidade de rotação do motor sem a necessidade de acesso à rede CAN do veículo;
- Comparação do desempenho das RNCs, utilizando duas formas de inserção de dados: convencionalmente, com arquivos de imagens; ou em formato compacto, passando apenas matrizes de amplitudes;
- Classificação do tipo de falha de ignição ocorrida, treinando a RNC, em três diferentes categorias: Falha em um único cilindro; Falha em dois cilindros com combustão sequencial; Falha em dois cilindros com combustão alternada;
- Avaliação do desempenho das RNCs treinadas ao classificar dados provenientes de outro motor veicular semelhante ao utilizado para treinamento e validação dos resultados.

1.3 Organização do Trabalho

Para uma melhor organização do trabalho, este será dividido em quatro capítulos após o capítulo introdutório. No capítulo 2 uma breve revisão bibliográfica será apresentada sobre os temas e conceitos necessários utilizados durante a pesquisa e seu desenvolvimento, incluindo definições e tópicos relacionados a motores de combustão interna, falhas relacionas a estes, acelerômetros MEMS capacitivos, Inteligência Artificial, Redes Neurais Artificiais e Redes Neurais Convolucionais.

No capítulo 3, será apresentada a metodologia utilizada, descrevendo as condições de projeto e considerações feitas para realização do trabalho, incluindo a seleção dos componentes eletrônicos utilizados para instrumentação do conjunto de aquisição, etapas de

avaliação e preparação dos dados nas duas formas de entradas propostas para treinamento das redes, bem como os hiperparâmetros definidos para estruturação de cada rede avaliada.

Já no capítulo 4, inicialmente os resultados obtidos com a aplicação do conjunto confeccionado serão apresentados, sendo avaliadas a robustez do sensor MEMS empregado nesse tipo de aplicação e a sua capacidade de descrever a rotação dos motores. Em seguida, os resultados obtidos com o treinamento das redes neurais propostas são apresentados, avaliando tempos de treinamento e desempenho na classificação dos diferentes tipos de falhas de ignição gerados.

Por fim, no capítulo 5 referente às considerações finais, uma retomada dos objetivos será feita, verificando-se a efetividade do trabalho em relação aos objetivos propostos, listando-se, ainda, algumas sugestões para realização de trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

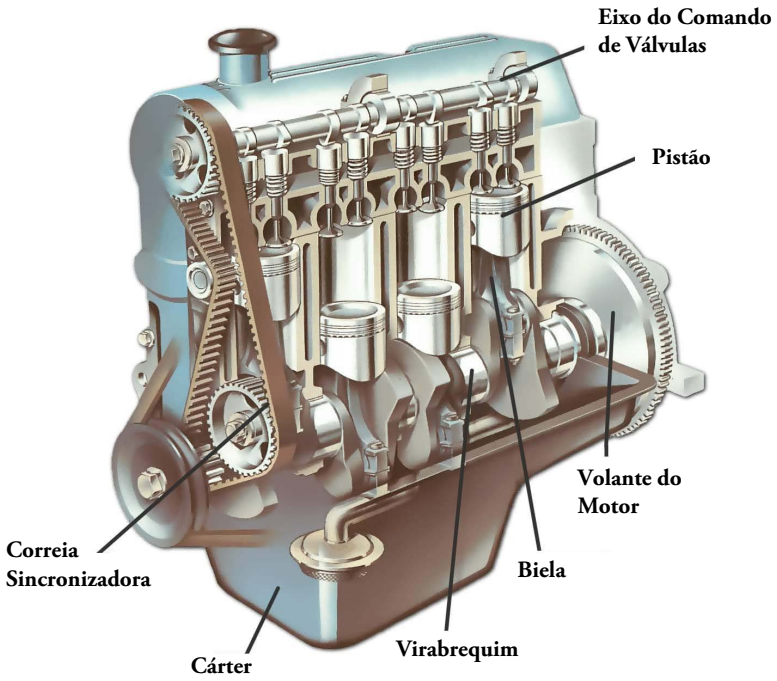
O presente capítulo tem como objetivo geral realizar uma revisão bibliográfica sobre alguns dos conceitos e tópicos utilizados ao longo da realização do projeto.

2.1 Motor de Combustão Interna

Os motores de combustão interna (MCI) foram criados e tiveram sucesso em seu desenvolvimento ao final da década de 1860. Sua criação é considerada como sendo uma das invenções mais significativas do século passado, impactando significativamente a humanidade, principalmente no quesito de mobilidade, sendo responsável pelo desenvolvimento de diversas outras áreas comerciais e tecnológicas da indústria (FERGUSON, 2016).

Diferente dos motores de combustão externa, que contam com câmaras de combustão separadas para queima de combustível, os motores de combustão interna são definidos como aqueles que realizam a liberação da energia química dos combustíveis diretamente em seu interior, utilizando-a imediatamente para realização de trabalho mecânico (FERGUSON, 2016). A Figura 1 ilustra uma estrutura exemplo de um MCI.

Figura 1 – Motor de combustão interna e alguns de seus componentes.



Fonte: Modificado de Mui (2016).

Conforme descrito por Ferguson (2016), os dois tipos de MCIs que se destacam em relação à aplicação são os de ignição por centelha e os de ignição por compressão, sendo estes os mais utilizados em motores comerciais. De forma simplificada, os motores de

ignição por centelha utilizam velas de ignição para dar início à combustão da mistura ar/combustível, buscando trabalhar próximo da estequiometria. Já os de ignição por compressão não contam com tais dispositivos, sendo o início da queima ocasionado pela elevada pressão no interior da câmara de combustão, apresentando geralmente estruturas mais robustas para suportar as elevadas razões de compressão (FERGUSON, 2016).

Além do modo de operação, diversas são as características que podem distinguir a estrutura de motores, como: geometria e curso dos pistões, quantidade de cursos por ciclo, tipo de refrigeração utilizada, número de cilindros e sua disposição espacial, quantidade de válvulas e sua forma de acionamento, dentre outras.

Por se tratar de um sistema composto por um elevado número de componentes móveis que dependem de diversos fatores interdependentes para uma correta operação, diferentes tipos de falhas podem comprometer de formas distintas o funcionamento de um motor de combustão interna. Na subseção seguinte, descreve-se brevemente um tipo de falha geralmente observado em motores de combustão interna, denominada como falha de ignição (*misfire*).

2.1.1 Falhas de Ignição

Segundo Carley (2019), a falha de ignição em motores de combustão interna é um problema relativamente comum que pode ou não ser de fácil diagnóstico. Quando ocorre, a performance veicular é comprometida, dificultando o controle de rotação do motor pela central eletrônica, gerando aumento no consumo de combustível e agravando emissões devido ao despejo de hidrocarbonetos não queimados diretamente no coletor de escape.

Dentre as possíveis causas para ocorrência de não queima em um cilindro, três podem ser listadas, sendo algumas de suas sub-causas descritas conforme listado nos itens a seguir (CARLEY, 2019):

- Falha de centelhamento: Inclui qualquer causa relacionada ao impedimento da alta tensão gerada pela bobina de chegar até o eletrodo da vela de ignição, não efetivando o centelhamento; Velas sujas, gastas ou danificadas; Cabos de velas rompidos ou com fuga; Defeitos na bobina de ignição;

- Mistura ar/combustível muito desbalanceada: Relação fora dos padrões que possibilitem a ignição da mistura para o tipo de motor, combustível e razão de compressão do conjunto; Bicos injetores sujos, obstruídos ou inoperantes; Vazamentos na válvula EGR; Filtro de combustível obstruído; Medição de ar inadequada; Baixa pressão na linha de combustível;

- Perda de compressão do cilindro: Baixa pressão e quantidade de ar/combustível no interior do cilindro na etapa de centelhamento; Válvulas com vazamento e irregularidades no assentamento; Juntas de cabeçote rompidas/queimadas.

Os sintomas de ocorrência de *misfire* podem ser semelhantes mesmo tendo diferentes tipos de causa, como já listado anteriormente. Entretanto, obter-se o conhecimento se a falha ocorre em um único cilindro ou em um conjunto de cilindros a cada ciclo de combustão pode auxiliar na identificação de possíveis causas do problema (CARLEY, 2008).

Falhas repetitivas em um só cilindro podem ser sinal de falha em um único componente referente ao cilindro em que a falha ocorre, como por exemplo: Bico injetor obstruído, inoperante ou danificado; Cabo de vela rompido ou com fuga; Vela suja, gasta ou danificada; Falha no sinal gerado pela central responsável pelo controle de combustível ou centelhamento do cilindro; Válvulas com mau assentamento ou vazamento, dentre outros motivos.

Já falhas repetidas em pares de cilindros podem indicar mal funcionamento em componentes gerais e comuns a esses cilindros, como: Bomba de combustível; Bobina de ignição (no caso de bobina comum aos cilindros); Combustível de baixa qualidade; Queima da junta do cabeçote conectando cilindros adjacentes, dentre outros motivos. Embora o motivo possa continuar de difícil identificação, esse tipo de indicador pode reduzir a lista de possibilidades geradoras da falha identificada.

Assim como já mencionado na seção 1.1, diversos trabalhos encontrados na literatura tentam realizar a detecção e classificação de falhas de ignição, utilizando diferentes métodos. Alguns exemplos de técnicas já empregadas podem ser mencionados, como a análise de velocidade angular instantânea do virabrequim (SHAO; WANG; WANG, 2020), análise de pressão instantânea dos gases de escape do motor (CECCARANI; REBOTTINI; BETTINI, 1998) e ainda, a medição e análise do sinal de ionização dentro da câmara de combustão (FAN et al., 2012).

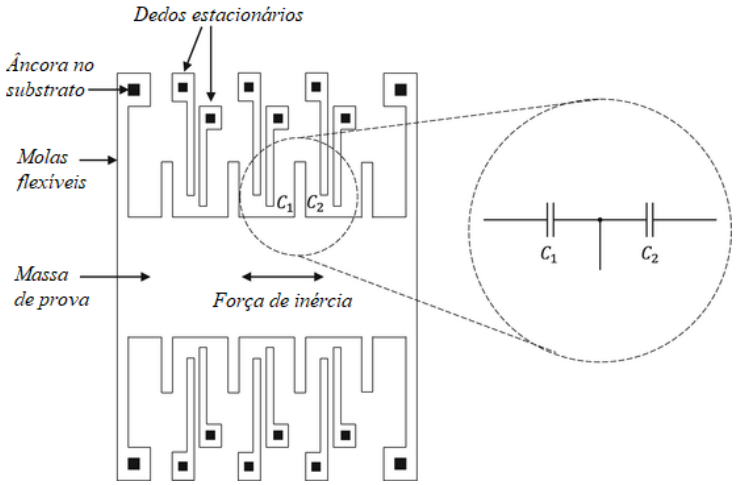
Outros artigos na mesma linha utilizam dados de vibração, coletados a partir de acelerômetros piezoelétricos, como tentativa de identificar a ocorrência da falha de ignição. Técnicas de *machine learning* são também aplicadas para análise dos dados e classificação de falhas individuais nos cilindros (BABU; I; SUGUMARAN, 2010; MULAY; SUGUMARAN; DEVASENAPATI, 2018).

2.2 Acelerômetros MEMS

O acelerômetro MEMS é um transdutor que transforma energia mecânica em energia elétrica, sendo a aceleração do sistema sua fonte de informação. Este é constituído por três estruturas fundamentais, a massa de prova, uma região de molas e estruturas ou dedos estacionários, conforme mostrado na Figura 2. Os dedos estacionários se encontram em ambos os lados da massa de prova, tendo o mesmo valor de capacitância na posição de equilíbrio mecânico do conjunto. Sendo submetida a uma aceleração, a massa de prova que é conectada fisicamente com as placas dos condensadores desloca-se, alterando

a distância ou a área entre os dedos capacitivos, alterando com isso a capacitância do conjunto condensador. Esse valor de capacitância é proporcional à aceleração aplicada no sistema (TEZ; AKIN, 2013).

Figura 2 – Diagrama esquemático de um acelerômetro MEMS uniaxial capacitivo.



Fonte: Modificado de (ALBARBAR; TEAY, 2016).

A crescente demanda por sensores aplicáveis em situações de monitoramento e análise de vibrações tem aumentado a necessidade e a procura por alternativas aos acelerômetros convencionais de alto custo (ALBARBAR; TEAY, 2016). De acordo com Korvink e Paul (2006) os dispositivos MEMS são considerados como sendo as menores máquinas criadas pelo homem, podendo o custo de tais dispositivos chegar geralmente a no máximo décimos dos valores praticados nos acelerômetros convencionais mais baratos disponíveis no mercado (ALBARBAR et al., 2008).

Trabalhos da última década têm investigado a substituição dos acelerômetros piezoelétricos por acelerômetros do tipo MEMS capacitivo, onde estes sensores são aplicados em casos que vão desde a detecção de falhas em veículos (REBELLO; CLEGHORN; MILLS, 2010) a até mesmo o acompanhamento e análise dinâmica da marcha humana (LIMA et al., 2013).

Comparando o desempenho de acelerômetros MEMS com piezoelétricos para monitoramento do estado de funcionamento de MCIs, Bismor (2019) constatou a aplicação de tais dispositivos como viável na detecção de falhas de ignição em motores veiculares. O dispositivo de baixo custo apresentou boa correspondência com o sinal amostrado utilizando um acelerômetro de referência, principalmente em frequências abaixo de 200Hz, sendo capaz de identificar um aumento no número de harmônicas geradas ao inserir a condição de falha de ignição no motor avaliado durante os ensaios.

2.3 Inteligência Artificial

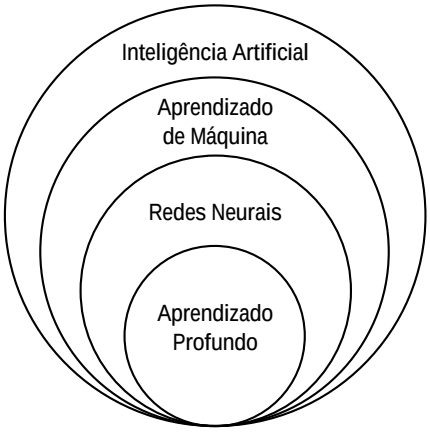
No ano de 1956, em uma conferência em Dartmouth College, com participação de muitos pioneiros da área, John McCarthy cunhou o termo Inteligência Artificial (COPPIN, 2004; BRUNETTE; FLEMMER; FLEMMER, 2009), o qual se relacionava com a ambição de pesquisadores em recriar a inteligência humana em máquinas, e que, posteriormente, tornou-se uma abordagem de grande potencial em diversas aplicações (LEE, 2019).

Conforme descrito por Santos (2018), o termo Inteligência Artificial (IA) não é de fácil definição, entretanto pode ser entendido como um termo amplo usado para classificar algoritmos que tentam mimetizar a inteligência humana, estando estes inseridos dentro do grande grupo definido como IA (KAVLAKOGLU, 2020). Atualmente, essa área inovadora é uma das mais exploradas dentro da ciência, devido ao seu grande potencial nas mais diversas aplicações, estando presente em situações que vão desde simples entretenimento a até mesmo grandes sistemas industriais realizando detecção e previsão de falhas (SANTOS, 2018; RISI; TOGELIUS, 2017).

Já o termo Redes Neurais (RNs) é considerado um subgrupo de uma subárea chamada Aprendizado de Máquina, mais conhecida como *Machine Learning*, a qual é considerada também um subgrupo da área mais abrangente, Inteligência Artificial. As RNs são arquiteturas que permitem às máquinas resolverem problemas sem necessariamente serem programadas para essa tarefa, utilizando camadas de neurônios artificias, recebendo e transmitindo informações de maneira semelhante à uma rede biológica (RASCHKA, 2015; LEE, 2019).

Quando as RNs apresentam mais de três camadas, contando com as camadas de entrada e de saída, o termo Aprendizado Profundo, ou do inglês, "*Deep Learning*", pode ser utilizado para defini-las nesse novo subgrupo (KAVLAKOGLU, 2020). Um melhor entendimento das áreas e nomenclaturas dadas aos subgrupos descritos pertencentes à grande área de IA pode ser obtido analisando-se a Figura 3.

Figura 3 – Relação entre os termos subgrupos da Inteligência Artificial.



Fonte: Próprio autor

Diversas são as arquiteturas de RNs que foram estruturadas e consolidadas ao longo dos anos em busca de melhor desempenho nas mais variadas tarefas, como as Máquinas de Boltzmann, as Redes Neurais Recorrentes, Redes Neurais Convolucionais, dentre diversas outras (LEIJNEN; VEEN, 2020). Nas próximas subseções, é feita uma breve descrição das redes empregadas no presente trabalho com suas características.

2.3.1 Redes Neurais Artificiais

Dentro da área de IA, as Redes Neurais Artificiais (RNAs) possuem um papel primordial e de grande destaque, sendo sua arquitetura tomada como base para diversas outras, contando com aplicação nas mais variadas áreas comerciais que empregam aprendizado de máquina (GOODFELLOW; BENGIO; COURVILLE, 2016).

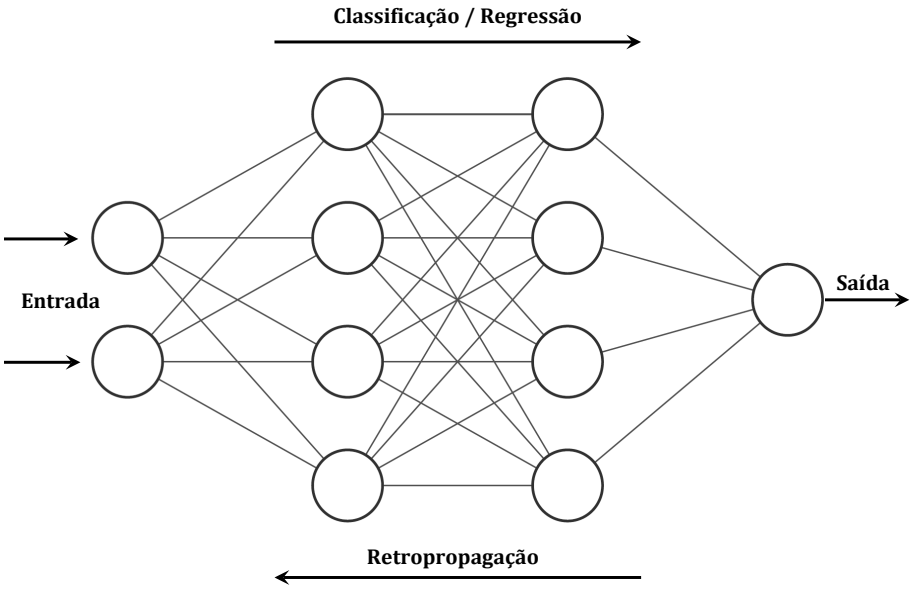
As RNAs são estruturas de aprendizado de máquina que tentam reproduzir a maneira de funcionamento de neurônios biológicos. A base de uma RNA consiste em uma combinação de pequenas estruturas, denominadas neurônios, interconectadas em camadas. Essas camadas também são conectadas entre si, formando uma rede.

Para aplicar uma RNA a um determinado problema, primeiro é preciso definir um conjunto de parâmetros e configurações que controlam o comportamento da rede. Essas configurações recebem o nome de Hiperparâmetros (HP). Tais parâmetros não são atualizados durante o treinamento da rede, a menos que um outro algoritmo seja aplicado para avaliar o efeito da variação dos HP sobre o desempenho da rede, realizando uma busca externa por melhores HPs (GOODFELLOW; BENGIO; COURVILLE, 2016).

A metodologia mais simples utilizada para treinamento de uma RNA é a do tipo supervisionada com aplicação do algoritmo de Retropropagação (*Backpropagation*). De forma simplificada, essa metodologia consiste na procura dos melhores pesos e parâmetros treináveis da rede, fornecendo exemplos de entradas e saídas. A rede então calcula a saída utilizando os pesos disponíveis e determina um valor de erro de acordo com a similaridade ou divergência em relação a saída esperada, modificando os parâmetros treináveis de forma regressiva, sempre buscando minimizar o erro de saída (HAYKIN, 2009).

A Figura 4 apresenta a topologia de uma RNA completamente conectada, também conhecida como densa. Na figura é mostrado ainda a direção de escoamento dos dados na rede durante etapas de classificação e regressão, fluindo das entradas para saída (*feed-forward*), e na retropropagação, da saída para a entrada, atualizando os pesos na etapa de treinamento.

Figura 4 – Rede Neural Artificial completamente conectada com duas camadas ocultas.



Fonte: Próprio autor

Para alcançar um determinado nível desejado de desempenho, deve-se selecionar HPs adequados e efetuar o treinamento da rede com dados representativos. Uma RNA é descrita como boa generalizadora quando o mapeamento das entradas e saídas calculadas pela rede apresentam bom desempenho analisando dados não vistos anteriormente durante seu treinamento (HAYKIN, 2009).

A Equação 1 apresenta uma expressão que descreve o funcionamento dos neurônios na camada de entrada da rede. O vetor $\boldsymbol{x}^{(1)}$ corresponde aos dados de entrada da rede com dimensão i, enquanto que o vetor $\boldsymbol{h}^{(1)}$ corresponde às saídas dos neurônios da primeira camada, com j elementos. A matriz $\boldsymbol{W}^{(1)}$ por sua vez terá dimensão $i \times j$, sendo chamada de matriz de pesos. A matriz é multiplicada pelo vetor de entrada da rede sendo o resultado da multiplicação somado ao vetor $\boldsymbol{b}^{(1)}$, chamado de *bias*, também de tamanho j. O resultado da operação é então inserido na função $g^{(1)}$, que é a função de ativação da camada de entrada, gerando os dados que são atribuídos à $\boldsymbol{h}^{(1)}$, podendo receber então o nome de estado de ativação da camada. A matriz $\boldsymbol{W}^{(1)}$ e o vetor $\boldsymbol{b}^{(1)}$ tem seus valores ajustados durante o treinamento da rede, enquanto que a função $g^{(1)}$ é um HP, definida para a camada em questão.

$$\boldsymbol{h}^{(1)} = g^{(1)}\left(\boldsymbol{W}^{(1)\top}\boldsymbol{x} + \boldsymbol{b}^{(1)}\right) \tag{1}$$

Já para as l camadas posteriores, o vetor de entrada se torna a saída da camada anterior, conforme mostrado na Equação 2 para a segunda camada da rede.

$$\boldsymbol{h}^{(l)} = g^{(l)}\left(\boldsymbol{W}^{(l)\top}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}\right) \tag{2}$$

Como já mencionado, a função de ativação é um HP da rede, devendo ser escolhida adequadamente para cada aplicação. Conforme descrito por Goodfellow, Bengio e Courville (2016), em redes neurais modernas, principalmente do tipo *feedforward*, há uma forte recomendação pela escolha na aplicação da função de ativação Unidade Linear Retificada, conhecida por ReLU (*Rectified Linear Unit*), sendo apresentada pela Equação 3.

$$g(z) = max\{0,z\} \tag{3}$$

Analisando essa função de ativação, é possível perceber que se trata de uma função rampa, apresentando como saída valores iguais aos de entrada, contudo zerando a saída para valores de z negativos.

Como vantagens da utilização dessa função de ativação nas camadas ocultas da rede neural, tem-se a característica dela ser não linear, ser de cálculo simples e zerar boa parte das saídas dos neurônios nas camadas que a usam como função de ativação. Isso atribui à rede a característica de rede esparsa, o que poupa uso de recursos computacionais e acelera a realização dos cálculos envolvidos nas etapas de treinamento e aplicação da rede neural (GLOROT; BORDES; BENGIO, 2010).

Já para a camada de saída de uma rede neural classificadora, funções como a *Sigmóide* e a *Softmax* podem ser aplicadas, contudo somente a segunda pode ser usada em problemas de classificação multi-classe (GOODFELLOW; BENGIO; COURVILLE, 2016). A função de ativação *Softmax* transforma as saídas para cada uma das classes em valores de 0 à 1, dividindo pela soma das saídas, dando assim a probabilidade da entrada corresponder a cada uma das classes. A função descrita é apresentada na Equação 4.

$$softmax(\boldsymbol{x})_i = \frac{exp(x)_i}{\sum_{j=1}^{n} exp(x)_j} \tag{4}$$

Conforme apresentado na Figura 4, os valores calculados por cada neurônio de cada camada, usando as Equações 1 e 2, são passados para a camada seguinte, até se obter os valores de ativação da última camada da rede. Tendo os estados de saída definidos, um cálculo é feito para quantificar o erro entre o que foi obtido com o que era esperado. Essa diferença é calculada utilizando uma função de perda $\mathcal{L}$ (*Loss Function*).

Novos valores são então calculados para $\boldsymbol{W}$ e $\boldsymbol{b}$ buscando minimizar o valor obtido pela função de perda. Esse método de otimização recebe o nome de Descida do Gradiente (*Gradient Descent*). Os pesos utilizados em uma dada iteração t são ajustados proporcionalmente ao erro obtido na nessa mesma iteração, sendo aplicados na próxima iteração $t+1$, conforme mostrado pela Equação 5 (LECUN et al., 2012).

$$W^{t+1} = W^t - \alpha \nabla \mathcal{L} \tag{5}$$

O termo α define a rigorosidade dada à perda calculada, sendo chamado de taxa de aprendizado ou *learning rate*, do inglês. Já o gradiente $\nabla \mathcal{L}$ tem seus elementos calculados com a aplicação da derivada parcial com relação aos pesos levando em conta a regra da cadeia (GOODFELLOW; BENGIO; COURVILLE, 2016). Assim, a Equação 6 mostra como são calculados os pesos $w_{i,j}$, que compõem $W^{(l)}$, de uma dada cama l da rede.

$$\frac{\partial \mathcal{L}}{\partial w_{i,j}^{(l)}} = \frac{\partial z_j^{(l)}}{\partial w_{i,j}^{(l)}} \frac{\partial y_j^{(l)}}{\partial z_j^{(l)}} \frac{\partial \mathcal{L}}{\partial y_j^{(l)}} \tag{6}$$

O mesmo cálculo é feito para os *bias*, pesos b_j, bastando substituir $w_{i,j}$ por b_j na Equação 6.

Esse processo de atualização dos pesos da rede recebe o nome de Retropropagação devido à forma como são calculadas as derivadas apresentadas, indo das camadas finais da rede para seu início sempre em relação aos pesos $w_{i,j}$ ou *bias* b_j (LECUN et al., 2012).

A atualização das matrizes de pesos e *bias* pode ser realizada após o processamento individual de cada vetor de entrada ou utilizando uma média das derivadas parciais da função de perda, depois de realizar a passagem de um determinado número de vetores pela rede. A quantidade de entradas levadas em consideração no cálculo dessa média das derivadas parciais é um HP denominado por *batch size*. Para esse caso, o processo de otimização da rede recebe o nome de Descida do Gradiente Estocástica (*Stochastic Gradient Descent*) (GOODFELLOW; BENGIO; COURVILLE, 2016).

A seleção do *batch size* pode influenciar diretamente no tempo de treinamento da rede, devido à quantidade de atualizações impostas para rede, tempo para passagem de todos os dados através da rede, limitações de memória, dentre outros fatores. Goodfellow, Bengio e Courville (2016) comentam que, especialmente em casos de treinamento usando GPUs, é recomendado o uso de *batch sizes* de potencias de dois, geralmente iniciando com um tamanho de 16 para modelos grandes, ficando normalmente entre 32 e 256 para modelos menores.

Trabalhos recentes desenvolveram melhorias em cima do algoritmo de Descida do Gradiente, apresentando otimizadores de fácil implementação como o caso do algoritmo *Adam*, introduzido e explicado por Kingma e Ba (2014), que requer apenas gradientes de primeira ordem e utiliza pouca quantidade de memória. Este algoritmo foi desenvolvido para combinar as vantagens de outros métodos, como *AdaGrad* e *RMSProp*, sendo indicado como boa alternativa para testes iniciais na grande maioria das aplicações de classificação, apresentando bom desempenho em problemas com muitos dados e com gradientes esparsos e ruidosos (KINGMA; BA, 2014).

2.3.2 Redes Neurais Convolucionais

As Redes Neurais Convolucionais (RNCs) são um tipo de arquitetura, geralmente profunda, de grande destaque por produzir ótimos resultados em problemas de reconhecimento de padrões e classificação de imagens (PANG et al., 2018; HAYKIN, 2009). Graças ao avanço no poder computacional, sua aplicação em problemas complexos fazendo uso de dados de entrada com elevadas dimensões tem ganhado destaque em diversas áreas (KARUPPASAMY et al., 2022).

Elas exploram correlações espaciais locais, impondo padrões de conectividade entre neurônios de camadas adjacentes. Além disso, possuem propriedades como, alto grau de invariância à translação, escala, inclinações e outras distorções, as quais são essenciais para o reconhecimento de padrões (HAYKIN, 2009). Embora inicialmente desenvolvidas e relacionadas à área de Visão Computacional, essa classe de redes não se limita apenas a tarefas dessa área (BARINO; BESSA, 2020).

Algumas de suas vantagens e características são listadas nos seguintes itens:

- Apresentam elevada habilidade na extração automática de características importantes dos dados de entrada sem demandar estudos exaustivos sobre os dados ou extração manual prévia de parâmetros;
- Amenizam problemas com a alta dimensionalidade nos vetores de entrada em redes com apenas camadas densas;
- Contam com elevado grau de invariância à translação, inclinações, escalas e outras distorções no padrão a ser classificado;
- São flexíveis, permitindo sua aplicação em diferentes áreas e acoplamento com outras técnicas.

Goodfellow, Bengio e Courville (2016) descrevem que as RNCs podem ser basicamente definidas como sendo Redes Neurais convencionais que fazem uso da operação linear de convolução em ao menos uma das camadas da rede. A função da convolução é extrair características relevantes dos dados de entrada da rede.

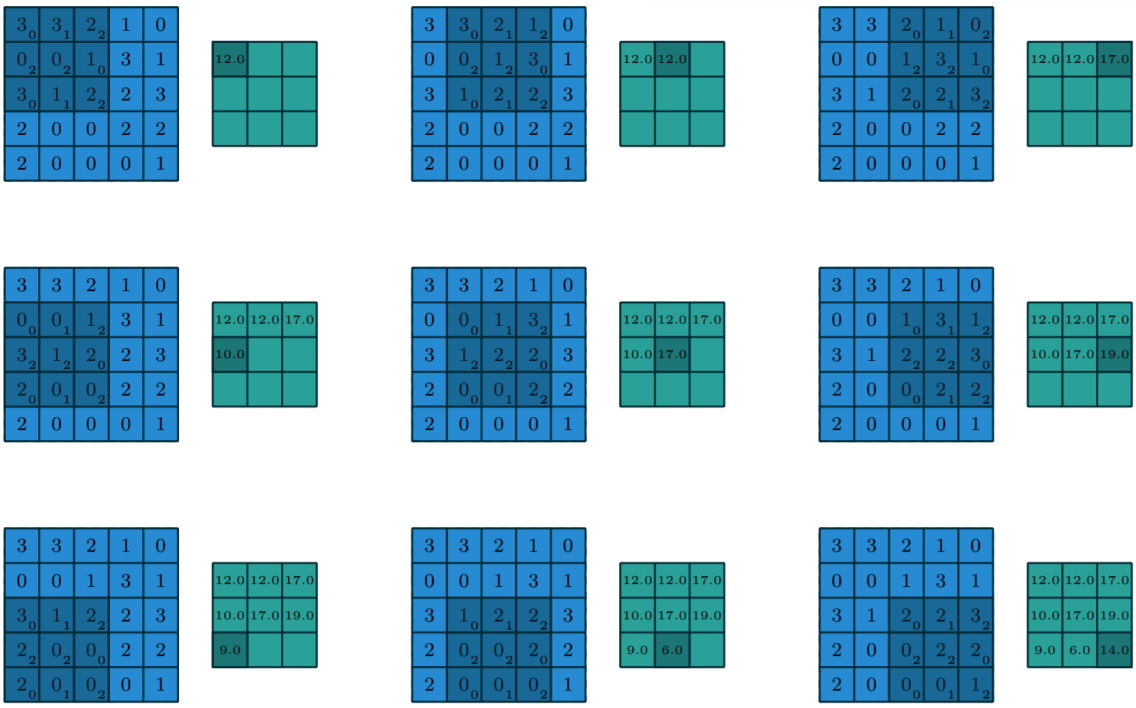
Supondo uma matriz X com dimensões $h \times w$ como entrada de uma camada l de uma dada RNC, a rede deve ser capaz de extrair as características importantes dessa entrada e disponibilizá-las em sua saída. Tais saídas da camada de convolução recebem o nome de Mapas de Características, ou *Feature Maps* do inglês, definidas aqui como $Y^{(l)}$. Para obtenção de tais mapas, é realizado o procedimento de convolução na entrada $X^{(l)}$ aplicando f *kernels* K_f, também chamados de filtros, com dimensões $k_1 \times k_2$ menores do que a entrada. Os filtros K_f recebem esse nome, pois a sua utilização é semelhante ao processo de aplicação de filtros em imagem digitais. A operação de convolução realizada em uma dada camada é descrita pela Equação 7.

$$y_{i,j}^{(l)} = a\left(\sum_{m=1}^{k_1}\sum_{n=1}^{k_2} K_{m,n}.X_{i+m,j+n}^{(l)} + b_{i,j}^{(l)}\right) \tag{7}$$

$$\text{para } i = 1,\ldots,(w-k_1+1) \text{ e } j = 1,\ldots,(h-k_2+1)$$

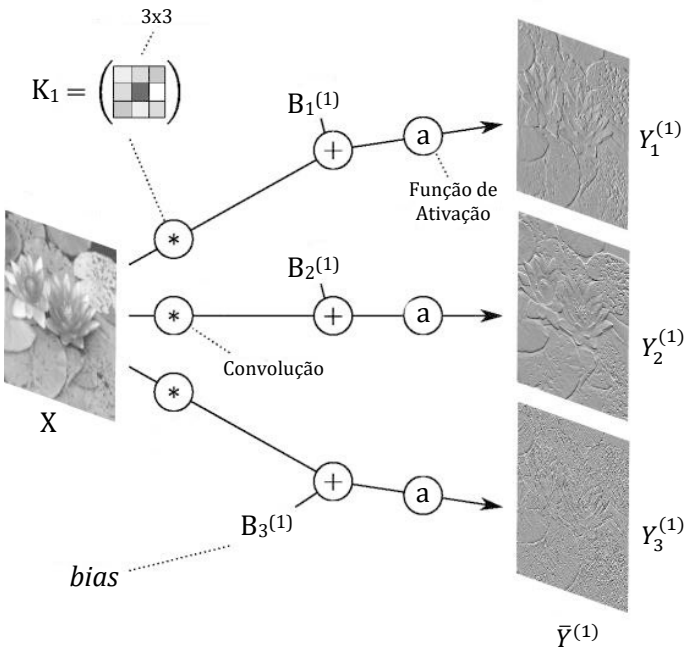
A Figura 5 ilustra de forma simplificada o processo de convolução propriamente dito, enquanto que a Figura 6 apresenta sua aplicação em uma camada de convolução com três filtros de dimensões 3x3.

Figura 5 – Convolução de uma matriz 5x5 com um *kernel* 3x3.



Fonte: Modificado de Dumoulin e Visin (2018)

Figura 6 – Aplicação de três filtros 3x3 em uma camada de convolução.



Fonte: Modificado de Bianco et al. (2019)

Conforme apresentado na Figura 6, para cara filtro aplicado sobre a entrada X teremos um mapa de característica Y na saída. O conjunto de mapas de características $\overline{Y}$ irá por sua vez servir como entrada para a próxima camada convolucional ou será redimensionado para a forma de vetor para adentrar uma camada densamente conectada. A quantidade de filtros f, seus tamanhos e funções de ativação são HPs da rede, diferentemente dos *bias* e pesos dos *kernels*, que sofrerão ajustes durante as etapas de treinamento. A função de ativação a utilizada comumente é a função $ReLU$ (GOODFELLOW; BENGIO; COURVILLE, 2016).

Além das camadas de convolução, outra característica comumente encontrada em RNCs são as camadas de *Pooling*. Esse tipo de camada realiza a operação de passar uma janela por pequenas regiões da matriz de saída da camada anterior realizando a extração de características tomadas como importantes, extraindo normalmente o valor máximo ou médio do intervalo avaliado pela janela, normalmente de dimensão 2x2, recebendo respectivamente o nome de *Max Pooling* e *Average Pooling*. Além da vantagem de extrair os parâmetros mais relevantes e aumentar a invariância translacional local da RNC, o procedimento ajuda ainda com a diminuição de dimensionalidade dos dados passados para a próxima camada de convolução, o que reduz parâmetros treináveis e pode auxiliar significativamente no desempenho da rede (DUMOULIN; VISIN, 2018; GOODFELLOW; BENGIO; COURVILLE, 2016; O'SHEA; NASH, 2015).

3 MATERIAIS E MÉTODOS

Nas subseções do presente capítulo, são apresentados os materiais e procedimentos empregados na etapa experimental, a metodologia utilizada para coleta, avaliação e preparo dos dados para treinamento das redes neurais, bem como suas estruturas e hiperparâmetros utilizados em seus treinamentos.

3.1 Conjunto Experimental

Nessa subseção serão apresentados os componentes eletrônicos selecionados para instrumentação do conjunto de aquisição, com as respectivas características e justificativas de escolha.

3.1.1 Critérios de Seleção

Para o desenvolvimento da pesquisa, alguns critérios foram estabelecidos como delimitadores para a seleção dos componentes empregados em sua etapa experimental. As condições de projeto delimitadas são listadas nos itens a seguir:

- O acelerômetro empregado na coleta dos dados de vibração deve ser de baixo custo do tipo MEMS capacitivo;
- Todo o conjunto de aquisição deve ser composto exclusivamente por componentes de baixo custo, incluindo o microcontrolador utilizado para amostragem e armazenamento dos dados de aceleração provenientes do acelerômetro selecionado;
- A instalação do conjunto de aquisição não deve necessitar de qualquer adaptação da estrutura original do motor ou veículo utilizados para realização dos ensaios, não devendo inclusive realizar a remoção de capas protetoras plásticas ou qualquer outro componente original dos mesmos.

Os critérios relacionados à utilização de acelerômetros MEMS e redução de custos foram definidos a fim de viabilizar uma futura implementação do conjunto desenvolvido em um equipamento comercial, possivelmente aplicável em veículos de frota. A utilização de acelerômetros piezoelétricos ou sistemas de aquisição com elevado valor agregado inviabilizariam a inserção do conjunto em veículos populares, onde muitas vezes o valor dos equipamentos utilizados em laboratório superam o valor de certos veículos.

Já o último critério listado, referente à fácil instalação sem necessidade de modificações do veículo, visa não limitar a aplicação do conjunto a ser desenvolvido em casos que a alteração de componentes do veículo possam gerar quebra de garantias ou situações relacionadas.

Além disso, a utilização do conjunto de baixo custo e a sua instalação simplificada são pontos interessantes para a pesquisa, já que podem ser tomadas como condições adicionais para avaliar a robustez das RNCs a serem treinadas. Assim, tendo os critérios de desenvolvimento traçados, deu-se continuidade ao projeto com a definição dos componentes para composição do conjunto experimental.

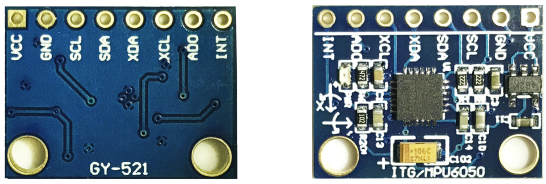
3.1.2 Acelerômetro MEMS

Para fazer parte do conjunto de aquisição, o módulo de acelerômetro tipo MEMS capacitivo selecionado foi o MPU6050, da fabricante *InvenSense*. Conforme mostrado no trabalho de Ribeiro e Lameiras (2019), boa parte dos diferentes modelos de acelerômetros do tipo MEMS avaliados apresentaram em geral performance satisfatória, incluindo o acelerômetro digital MPU6050, comumente encontrado na *breakout board* GY-521.

Dentre os motivos de sua seleção para integrar o projeto está seu baixo custo, sua grande aplicação em projetos científicos na área de engenharia, e sua disponibilidade no inventário. Este modelo conta com um conjunto acoplado de acelerômetro e giroscópio do tipo MEMS de três eixos cada em um único chip, além de um sensor de temperatura, tendo como protocolo de comunicação o padrão I2C (*Inter-Integrated Circuit*).

Além disso, a *breakout board* opera com uma tensão de 3V à 5V, contando com um conversor analógico-digital de 16 bits. A Figura 7 ilustra o módulo GY-521 contendo o MPU6050 de 6 DOF (*Degrees of Freedom*), sendo que algumas de suas características gerais são apresentadas na Tabela 1. Mais detalhes sobre os modos de operação, calibração e outras características sobre o MEMS podem ser vistas em Invensense (2013).

Figura 7 – Módulo MPU6050 com acelerômetro e giroscópio tipo MEMS.



Fonte: Próprio autor

Tabela 1 – Características da placa GY-521 contendo o MPU6050.

Frequência de Amostragem (Fs) [kHz]	**Sensibilidade Transversal do Acelerômetro**	**Desempenho em Ruído [$\mu g/\sqrt{Hz}$]**	**Faixas de Medida**	
			Giros [$^o/s$]	**Acel. [g]**
Acel.: 1 Giros: 8	+2%	400(1)	±250, ±500, ±1000, ±2000	±2, ±4, ±8, ±16

(1) Utilizando a faixa de ±2g e Fs = 1kHz para uma largura de banda de 10Hz.

Font: (INVENSENSE, 2013)

3.1.3 Microcontrolador

Com o acelerômetro MEMS capacitivo definido, uma plataforma capaz de comunicar com o mesmo para aquisição dos dados precisou ser selecionada. Com intuito de atender ao critério de projeto de que todo o sistema deve ser composto por componentes de baixo custo, diversas placas de prototipagem desse segmento foram listadas, avaliando suas características e capacidade de atender ou não à demanda do projeto.

Assim, para o projeto em questão, selecionou-se como hardware base para aquisição dos dados a plataforma de desenvolvimento da família *Espressif Systems*, denominada como ESP32. Essa escolha foi feita baseada no custo-benefício do conjunto, tendo em vista o suficiente número de portas GPIO (*General Purpose Input/Output*) disponíveis, uso de microcontrolador com dois núcleos, além de desempenho muito superior em processamento, quando comparado a dispositivos com propostas similares, como as plataformas da família Arduino disponíveis no mercado. Algumas características básicas do MCU (*Microcontroller Unit*) fornecidas por Espressif (2021) são descritas na Tabela 2.

Tabela 2 – Características da plataforma de desenvolvimento *ESP32* da fabricante *Espressif Systems*.

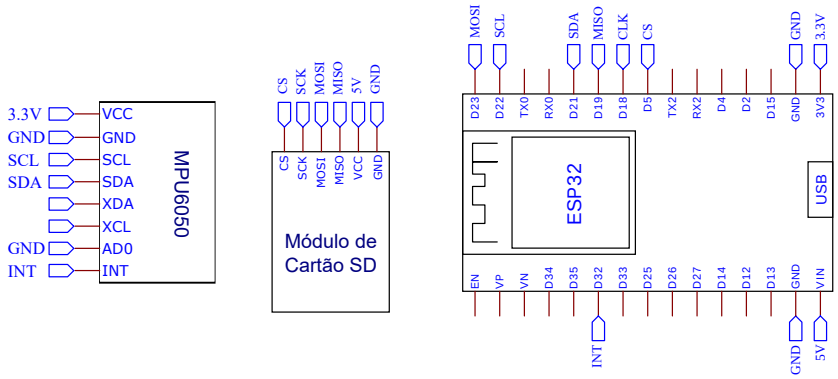
Parâmetro	**Característica**
MCU	Xtensa Dual-Core 32-bit LX6
Frequência Máxima	240 MHz
SRAM	512 kBytes
Flash	16 MBytes
GPIOs (portas)	36
PWM (canais)	Software: 16 | Físico: 1
SPI / I2C / 12S / UART	4 / 2 / 2 / 3
Tensão de Operação	3,3V
Tensão de Entrada do módulo	4,5V $\sim$ 9,0V
Wi-Fi | Bluetooth	802.11 b/g/n | 4.2
Temperatura de Operação	-40^{o}C $\sim$ 125^{o}C

Fonte: espressif.com/en/products/devkits

3.1.4 Circuito Eletrônico

Os componentes eletrônicos selecionados para a composição do conjunto de aquisição foram conectados entre si de forma a atender as configurações dos módulos, utilizando GPIOs adequadas para a comunicação I2C e SPI, respectivamente para o módulo do acelerômetro e do cartão de memória. O esquema de ligação final entre os componentes pode ser visto na Figura 8.

Figura 8 – Diagrama esquemático das conexões entre os componentes do circuito eletrônico.



Fonte: Próprio autor

O circuito foi alimentado utilizando a conexão USB de um notebook desligado da rede elétrica sempre que utilizado para a realização dos ensaios, a fim de evitar possíveis interferências da rede externa de energia CA. O conjunto foi inicialmente montado em uma placa de prototipagem rápida para verificar suas ligações e funcionamento adequado do código, sendo posteriormente transferido para uma placa de circuito impresso, a fim de garantir a correta conexão entre os componentes quando soldados.

Foi desenvolvido um código em *C++* utilizando os dois núcleos disponíveis no microcontrolador selecionado de forma a garantir uma frequência de amostragem constante conjuntamente ao armazenamento dos dados no cartão de memória. Para isso, o sistema FreeRTOS foi aplicado para administrar as tarefas de aquisição dos dados provenientes do acelerômetro MEMS e de transferência dos dados armazenados da memória volátil para o cartão SD.

3.1.5 Motor de Combustão Interna

Para realização dos ensaios propostos, fez-se uso de dois veículos da marca FIAT, sendo eles um Punto HLX 1.8 8v e um Palio Weekend 1.8 8v, ambos fabricados no ano de 2008, período no qual eram produzidos com motores modelo Y2 e 93313090-7U, respectivamente, compartilhados da linha GM. Em relação às condições de uso, o odômetro do primeiro marcava aproximadamente 150 mil quilômetros rodados, enquanto que o segundo, aproximadamente 50 mil quilômetros a mais, apresentando ambos condições de manutenção e uso próximas por pertencerem ao mesmo proprietário.

A respeito da motorização, trata-se de um motor transversal de ignição por centelha, quatro cursos, com comando do tipo SOHC (*Single Overhead Camshaft*), injeção de combustível na porta e volume de deslocamento total de 1796cm^3, sendo composto por quatro cilindros em linha. O sistema de injeção de combustível e ignição é integrado, sendo controlado por uma unidade de controle central Delphi multiponto sequencial. Apenas uma

central eletrônica controla ambas as funções, determinando conjuntamente a duração do tempo de injeção, para a dosagem de combustível, e o ângulo de avanço da ignição.

Algumas outras características dos motores presentes nos veículos selecionados como bancadas de testes são apresentadas na Tabela 3.

Tabela 3 – Características dos motores veiculares utilizados.

Sequência de Ignição	**Diâmetro x Curso [mm]**	**Razão de Compressão**	**Potência Máxima [cv] (rpm)**	**Torque Máximo [kgf.m] (rpm)**	**Marcha Lenta [rpm]**
1-3-4-2	80,5 x 88,2	10,5±0,3:1	115 (5500)	18,5 (2800)	900±50

Font: (FIAT, 2008)

Embora não mencionado no manual do usuário, os veículos contam com uma porta de comunicação com a rede CAN (*Controller Area Network*) (CARLEY, 2020) no padrão OBD-II (*On Board Diagnostics*) junto ao compartimento de fusíveis internos, compatível com o padrão aceito pelo adaptador descrito na subseção seguinte.

3.1.6 Adaptador OBD-II

Como parâmetro de comparação dos dados experimentais obtidos utilizando o sistema composto pelo microcontrolador e o acelerômetro MEMS, um adaptador OBDII modelo ELM327 V1.5 foi utilizado para a comunicação com a rede CAN dos veículos. Este adaptador conta com um microcontrolador PIC18F25K80 integrado e comunicação serial via Bluetooth 2.1. Sua escolha foi feita baseada em sua disponibilidade e aplicação em diversos veículos da linha FIAT, atendendo aos modelos selecionados para realização dos testes.

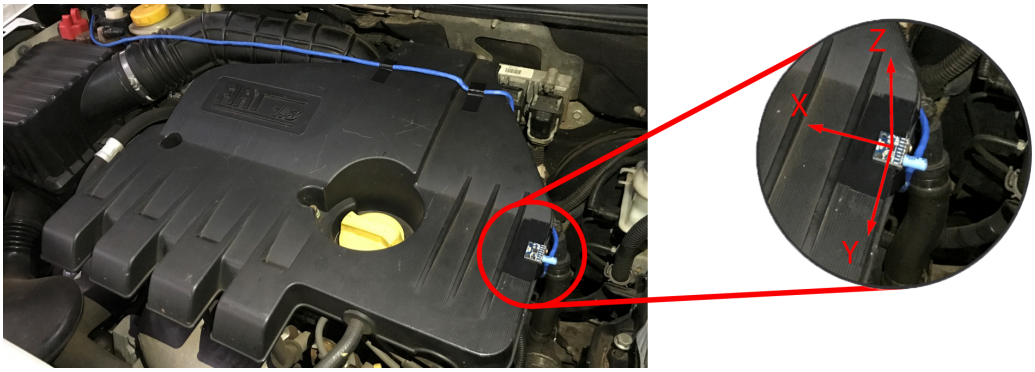
3.2 Coleta dos Dados Experimentais

Com todo o aparato experimental selecionado e confeccionado, deu-se início à etapa de coleta dos dados. Primeiramente realizou-se a instalação do conjunto de aquisição no cofre do motor do primeiro veículo utilizado como bancada de testes. Conforme descrito na subseção 3.1.1 do presente capítulo, um dos critérios estipulados foi a não violação de qualquer sistema ou componente original do veículo. Dessa forma, a capa de proteção do motor foi definida como sendo o melhor local disponível para fixação do acelerômetro.

A fixação foi feita utilizando uma camada adesiva entre o componente do motor e a superfície da placa do sensor. Adicionalmente um grampo de pressão foi aplicado para garantir uma melhor adesão entre os componentes e evitar um possível desprendimento durante a realização dos ensaios.

Em relação à orientação dos eixos do acelerômetro MEMS, este foi posicionado de forma com que seu eixo X fosse alinhado ao eixo do virabrequim do motor. O eixo Z foi montado normal à superfície da capa protetora, alinhado ao eixo de deslocamento dos pistões. A Figura 9 ilustra o posicionamento do acelerômetro com o direcionamento de seus eixos em relação ao motor.

Figura 9 – Posicionamento do acelerômetro em relação ao motor.



Fonte: Próprio autor

Antes de descrever o roteiro de coleta dos dados, cabe relembrar, conforme descrito nos objetivos do trabalho, as quatro condições de operação do motor que foram avaliadas, sendo elas:

- **(S) Motor saudável:** Motor operando em marcha lenta com condições adequadas de funcionamento em todos os seus quatro cilindros;
- **(F1) Falha em um cilindro:** Simulação de falha em apenas um cilindro de forma contínua;
- **(F2) Falha em dois cilindros com combustão sequencial:** Simulação de falha contínua em dois cilindros adjacentes em relação à sequência de combustão do motor;
- **(F3) Falha em dois cilindros com combustão alternada:** Simulação de falha contínua em dois cilindros alternados em relação à sequência de combustão do motor;

Para simular as três condições de falha de ignição descritas, realizou-se o desligamento do cabo de vela nos cilindros em que as falhas eram desejadas, impedindo que o centelhamento ocorresse em seu interior. Conhecendo a sequência de combustão do motor utilizado (1-3-4-2), a Tabela 4 apresenta as combinações de desligamento dos cilindros para geração de cada uma das falhas descritas.

Tabela 4 – Condições possíveis para simulação das falhas desejadas.

Falha	**Cilindros Desligados**
F1	1 ; 2 ; 3 ; 4
F2	(1 e 3) ; (3 e 4) ; (4 e 2) ; (2 e 1)
F3	(1 e 4) ; (3 e 2)

Fonte: Próprio autor

Assim, após a devida instalação do sistema de aquisição, e definida a forma de ocasionar as falhas intencionais, a etapa de coleta dos dados de vibração do motor foi iniciada. O procedimento aplicado durante a realização dos ensaios experimentais está descrito a seguir:

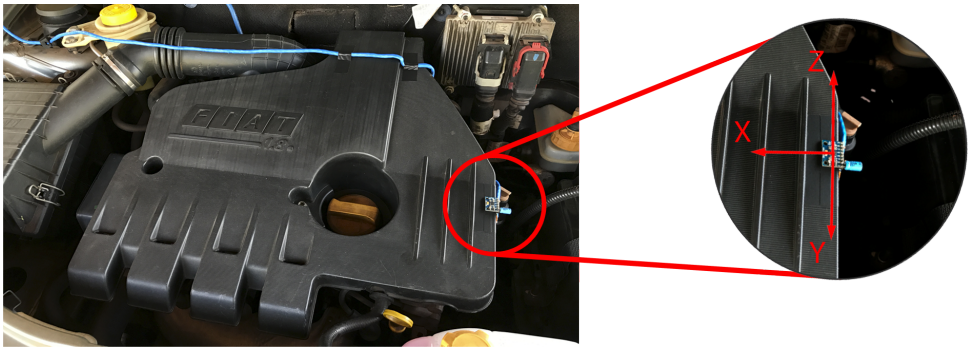
- Checagem das conexões elétricas do conjunto de aquisição de dados e verificação da correta fixação do acelerômetro;
- Remoção e isolamento dos cabos de vela para simulação da falha desejada no sistema (apenas para coletas dos tipos F1, F2 e F3);
- Partida no motor do veículo, aguardando estabilização da rotação e alcance da temperatura de trabalho;
- Início da aquisição dos dados de aceleração;
- Finalização da gravação ao atingir o volume de dados necessários para a classe de funcionamento;
- Desligamento do motor.

As etapas listadas foram repetidas ao longo da coleta dos dados de todas as situações de operação propostas, iniciando pela condição de motor saudável e finalizando com os dois casos da falha F3. Aproximadamente cinco minutos de dados foram gravados para cada condição de operação da Tabela 4. Para todas as coletas realizadas foi utilizada a máxima frequência de amostragem suportada pelo acelerômetro, de 1000Hz. A escala máxima de aceleração setada foi de $\pm$8g.

Uma coleta de dados extra na situação de motor saudável foi realizada conjuntamente com o adaptador OBD-II plugado à rede CAN do veículo em modo de gravação. Exclusivamente para esse caso, após a partida e estabilização da rotação e temperatura, o motor foi acelerado e desacelerado algumas vezes. Os dados coletados nessa etapa exclusiva do procedimento experimental foram utilizados para avaliar a correspondência dos dados obtidos utilizando o conjunto de aquisição desenvolvido com acelerômetro MEMS capacitivo em relação a real frequência de rotação do eixo virabrequim.

Após realizar todas as etapas de coleta descritas utilizando o motor do primeiro veículo, o conjunto de aquisição foi então transferido para o motor do segundo veículo. A fixação e orientação do sensor MEMS no segundo motor foram feitas tentando reproduzir a mesma configuração utilizada para o primeiro caso. A Figura 10 mostra o posicionamento do acelerômetro e a orientação de seus eixos em relação ao MCI do segundo veículo.

Figura 10 – Posicionamento do acelerômetro no motor do segundo veículo.



Fonte: Próprio autor

Em seguida, todas as etapas descritas para a coleta de dados no primeiro motor foram repetidas para o segundo motor. Todas as configurações do conjunto de aquisição foram mantidas, coletando-se aproximadamente dois minutos de dados para cada condição de operação da Tabela 4.

O procedimento de gravação dos dados, em conjunto com o adaptador OBD-II monitorando a velocidade angular do virabrequim, foi novamente realizado, para que a verificação da correspondência do sinal obtido pelo sensor MEMS pudesse ser feita também para esse caso.

Após a etapa de coleta dos dados, deu-se início à preparação destes para posterior utilização como entradas das RNCs. Cabe aqui mencionar que as metodologias de preparação das entradas que serão descritas a seguir foram aplicadas sobre os dados coletados de ambos os motores. No entanto, os dados do segundo motor foram usados apenas para a etapa pós treinamento das redes, com o objetivo de avaliar o comportamento das redes tratando dados de um novo conjunto.

3.3 Preparação dos Dados

Para preparação dos dados coletados, fez-se uso da linguagem *Python*, utilizando adicionalmente algumas de suas bibliotecas geralmente empregadas para tarefas de tratamento e apresentação de dados, como *Pandas* (REBACK et al., 2022), *Numpy* (WALT; COLBERT; VAROQUAUX, 2011) e *MatplotLib* (HUNTER, 2007).

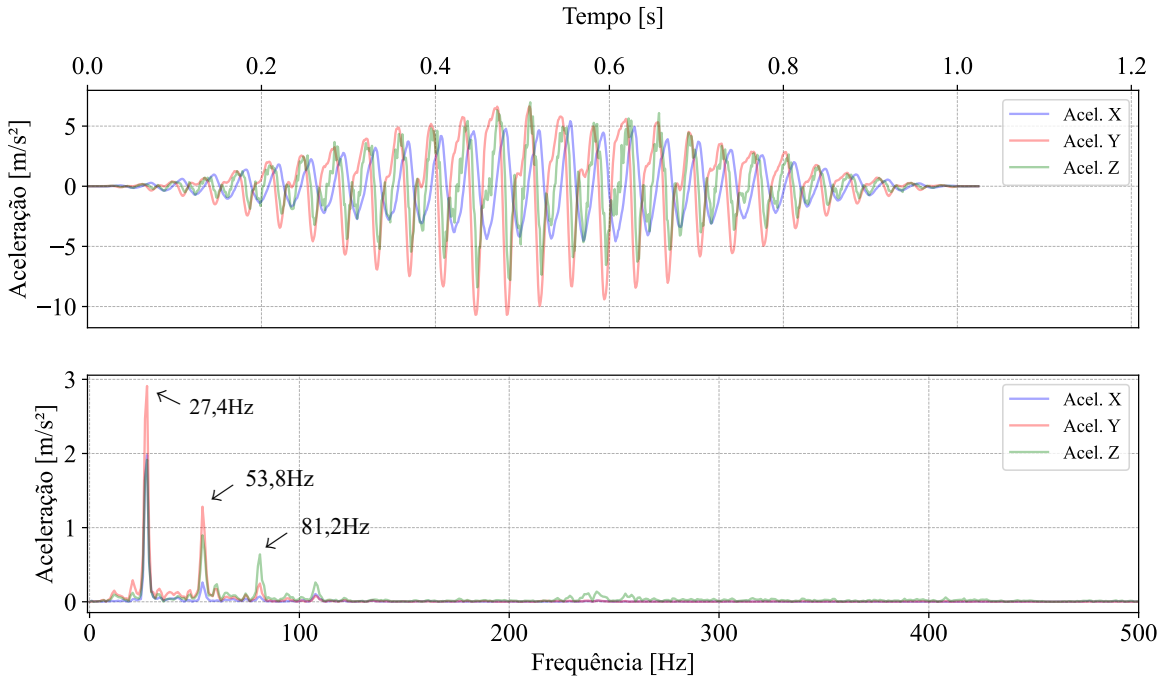
Inicialmente, os arquivos em formato ".csv" armazenados no cartão de memória contendo os sinais amostrados foram transferidos para um computador, sendo organizados e

renomeados de acordo com a classe pertencente. Os dados de aceleração foram importados para variáveis na memória, removendo-se preliminarmente a componente DC dos sinais.

Para cada um dos vetores de aceleração, o sinal completo foi percorrido por uma janela do tipo *Hanning* (OPPENHEIM; SCHAFER, 1975) com comprimento de 1024 pontos, fatiando o sinal em intervalos também de tamanho 1024. O deslocamento da janela sobre o sinal original ocorreu com um passo de 512 pontos, ou seja, gerando uma sobreposição de 50% em seu deslocamento sobre o vetor original. Tal janelamento, além de reduzir o espalhamento espectral durante a posterior análise dos dados no domínio da frequência, proporcionou também um aumento da quantidade de dados disponíveis para treino das redes neurais.

Considerando o sinal amostrado quase estacionário, por conta do sinal ser coletado em condição de motor operando em marcha lenta constante, calculou-se a FFT dos sinais janelados. Dessa forma, conhecendo a frequência de amostragem do conjunto, e o número de pontos utilizados para cálculo das FFTs, de acordo com o teorema de amostragem de Nyquist (OPPENHEIM; SCHAFER, 1975) o conjunto possibilitou uma avaliação no domínio da frequência de um espectro de 500Hz, com resolução de aproximadamente 1Hz. A Figura 11 mostra um exemplo com a aplicação das etapas de preparação dos dados descritas até agora para um intervalo janelado do sinal de motor saudável.

Figura 11 – Sinal janelado no domínio do tempo e da frequência.



Fonte: Próprio autor

Em seguida, as amplitudes do sinal no domínio da frequência foram normalizadas entre 0 e 1 em relação à máxima amplitude de cada eixo de aceleração. A partir desse ponto da preparação, os dados foram submetidos a duas diferentes etapas de processamento antes

de serem armazenados e posteriormente utilizados como entradas das RNCs. Tais etapas são descritas nas seções 3.3.1 e 3.3.2.

3.3.1 Entrada em Forma de Imagens

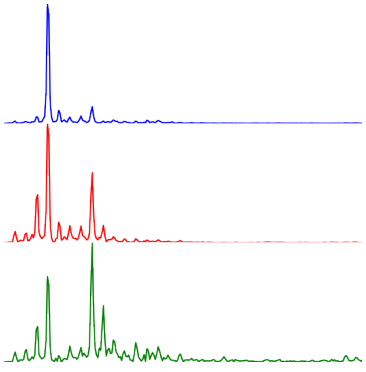
Para o primeiro método de entrada, em forma de imagens, buscou-se avaliar não somente o desempenho da rede ao interpretar os dados coletados, mas também sua capacidade de ler tais entradas em formato de gráficos, possibilitando sua aplicação em situações em que apenas entradas visuais são fornecidas.

Assim, para este método de entrada, os espectros normalizados no domínio da frequência dos três eixos de aceleração foram acoplados plotando-os em figuras com proporções quadradas, com dimensões de 512x512 pixels e profundidade unitária. O espectro do sinal de aceleração do eixo X foi posicionado no topo da figura, enquanto que o de Y foi inserido ao centro e o de Z na região inferior da imagem. Também foram removidos todos os elementos gráficos adicionais da imagem, como contornos, eixos, legendas e grades.

As figuras geradas foram rotuladas de acordo com o tipo de classe pertencente (S, F1, F2 ou F3), sendo armazenadas em diretórios contendo apenas as respectivas classes. Ao todo, a partir dos dados coletados do primeiro motor, foram geradas 8800 imagens para o banco de treinamento, e 2400 para o banco de teste, apresentando quantidades iguais para cada uma das quatro classes. Já com os dados do segundo motor, foram geradas 1600 imagens voltadas à um banco de teste isolado, utilizado apenas ao final de todos treinamentos. Não houve coleta e preparação de um banco de validação visto que, conforme será descrito na seção 3.4.1, não ocorreu uma etapa de ajuste dos hiperparâmetros para a rede utilizada nessa condição de dados de entrada.

A Figura 12 apresenta um exemplo de um dado de entrada preparado em forma de imagem originalmente em preto e branco, mas nesse caso em escala RGB exclusivamente para ilustrar o posicionamento dos dados referentes a aceleração nos eixos X, Y e Z respectivamente pelas cores azul, vermelho e verde.

Figura 12 – Ilustração de um dado de entrada em formato de imagem, originalmente em preto e branco, para treinamento da rede.



Fonte: Próprio autor

3.3.2 Entrada em Forma de Matriz de Amplitudes

Diferente das entradas armazenadas em formato de gráficos, a segunda forma de entrada preparada utilizou apenas as amplitudes normalizadas das FFTs obtidas com os dados de aceleração dos três eixos, sem qualquer geração de imagens. Dessa maneira, esperava-se reduzir significativamente as dimensões de entrada, buscando minimizar o uso de recursos, como memória e processamento.

Nesse caso, os três vetores de amplitude dos espectros de frequência foram acoplados originando matrizes com dimensões de 3x512. As amplitudes da aceleração do eixo X foram posicionadas na primeira linha, enquanto que as do eixo Z foram inseridas na terceira linha da matriz.

As matrizes foram então armazenadas em outro diretório contendo suas respectivas classes, novamente sendo os dados rotulados de acordo com o tipo de classe pertencente (S, F1, F2 ou F3). A quantidade de dados e sua organização foram as mesmas mencionadas para os dados armazenados em forma de imagens, já que toda a etapa anteriormente descrita a respeito do janelamento e normalização foi comum para ambos os conjuntos de arquivos gerados.

3.4 Estrutura das Redes Neurais Convolucionais e Treinamento

Conforme já descrito na seção 2, a estrutura de uma rede neural convolucional pode variar significativamente, dependendo da aplicação para a qual ela for direcionada. A quantidade de camadas convolucionais, filtros, funções de ativação, camadas densamente conectadas, quantidade de neurônios, dentre outros hiperparâmetros, são normalmente selecionados de acordo com o tipo de problema em que as redes serão aplicadas.

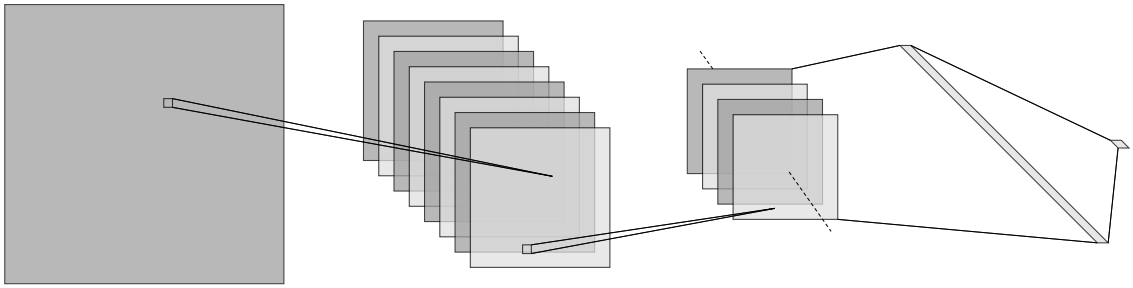
Frequentemente esses hiperparâmetros são escolhidos por meio de experimentação, tentativa e erro, avaliando-se o desempenho de diferentes configurações, utilizando-se geralmente mecanismos de busca como *Gridsearch*, *Randomsearch* e otimização Bayesiana. Opcionalmente, técnicas manuais para seleção de HPs também podem ser utilizadas, empregando estruturas que já apresentaram bom desempenho na realização de tarefas similares (SHAHUL, 2022; HUTTER; KOTTHOFF; VANSCHOREN, 2019).

Para atender os objetivos propostos para o trabalho, duas diferentes abordagens foram utilizadas na determinação dos hiperparâmetros das redes. Na seção 3.4.1 são apresentadas as características da RNC treinada para classificação dos dados gerados em forma de gráficos, tomando como base estruturas utilizadas na classificação de imagens, enquanto que na seção 3.4.2 é descrita a forma como a busca por hiperparâmetros foi feita para estruturar a segunda RNC, responsável pela classificação dos dados em forma de matrizes de amplitude.

3.4.1 RNC Classificando Imagens

Para a primeira RNC, direcionada a classificação dos gráficos em formato de imagem, optou-se pela implementação de uma estrutura composta por apenas duas camadas de convolução e uma densamente conectada. Essa escolha foi feita tomando-se como base a topologia utilizada em trabalhos com propostas similares de classificação, apresentando redes com bom desempenho ao utilizarem duas ou quatro camadas de convolução seguidas de apenas uma camada completamente conectada (LIU et al., 2018; ROCHA et al., 2019). A Figura 13 apresenta de forma simplificada a topologia selecionada para a primeira rede.

Figura 13 – Ilustração da topologia da rede utilizada.



Fonte: Próprio autor

A camada de entrada da rede recebe as imagens preparadas seguindo o procedimento descrito na seção 3.3.1. Os dados de entrada fluem a partir do banco de dados de imagens previamente gerado onde estão armazenados e organizados de acordo com sua classe.

Após a entrada na rede, os dados passam por uma primeira camada de convolução, onde são aplicados oito filtros com dimensões 3x3, seguido de uma camada com operação de *maxpooling* de tamanho 2x2, reduzindo dessa maneira pela metade a proporção dos dados e buscando suas características mais significativas.

Em seguida, os dados passam por uma segunda camada de convolução, utilizando filtros de mesmas dimensões aplicadas na primeira camada, contudo agora composta apenas por quatro *kernels*. Como saída dessa segunda camada são obtidos 32 mapas de características, sendo esses submetidos a outra operação de *maxpooling* de dimensão 2x2.

Antes de entrar na camada densamente conectada, os dados passam então por uma operação de achatamento (*flattening*), onde os mapas de características obtidos após o segundo *maxpooling* são planificados e concatenados, sendo convertidos em um vetor 1D.

A camada densa, composta por 64 neurônios, recebe então como entrada os dados planificados. Por fim, os dados fluem para a última camada da rede, composta por apenas quatro neurônios, visto que se trata de um problema de classificação com quatro classes distintas. Após as duas camadas de convolução e a camada densa, é aplicada a técnica de *dropout* com uma taxa de desligamento de 20% dos neurônios, a fim de reduzir-se as

chances de um possível sobreajuste (*overfitting*) da rede com os dados de treinamento.

Para todas as camadas, tanto de convolução, quanto densamente conectada, a função de ativação empregada foi a *ReLU*, com exceção da camada de saída. Por se tratar de um problema de classificação com mais de duas classes, a função selecionada para a última camada foi a *Softmax*. A Tabela 5 apresenta a estrutura e as características descritas para cada camada da rede.

Tabela 5 – Estrutura da RNC com entradas em formato de imagem.

Camada	**Tipo**	**Características**
1	Convolução	8 filtros de dimensão 3x3 - *ReLU*
2	*Maxpooling*	Dimensão 2x2
3	*Dropout*	20% de desligamento
4	Convolução	4 filtros de dimensão 3x3 - *ReLU*
5	*Maxpooling*	Dimensão 2x2
6	*Dropout*	20% de desligamento
7	*Flattening*	Achatamento para a camada densa
8	Densa	64 neurônios - *ReLU*
9	*Dropout*	20% de desligamento
10	Saída	4 neurônios - *Softmax*

Fonte: Próprio autor

A rede foi compilada utilizando o *Adam* como algoritmo de otimização, empregando como critério para cálculo de perda a função *Categorical Crossentropy*, tomando como métrica a acuracidade de classificação dos resultados. Os parâmetros relacionados ao otimizador, como a taxa de aprendizado, foram mantidos com seus valores padrões recomendados pela biblioteca.

Foi empregada a técnica de *augmentation* nos dados utilizados para treinamento da rede a fim de tentar aumentar a capacidade de predição da rede e reduzir as chances de ocorrência de *overfitting*, além claro, de aumentar o banco de dados de treinamento. Para isso, foi aplicado um deslocamento horizontal randômico nos dados de entrada em uma faixa de $\pm 10\%$ da largura das imagens, preservando suas características verticais. Essa translação horizontal simula um deslocamento nas bandas de frequência dos gráficos de entrada, contudo sem deformação dos mesmos.

A quantidade máxima de épocas estipulada para o treinamento da rede foi de 50, sendo definido um *batch size* igual à 16. Algumas funções de *callbacks* foram utilizadas a fim de monitorar e salvar os melhores pesos durante a evolução do treinamento. A técnica de *Early Stopping* foi utilizada buscando a minimização da função de perda no conjunto de validação, definindo-se uma paciência de 10 épocas.

3.4.2 RNC Classificando Dados Compactos

Em relação à segunda RNC a ser implementada, fazendo uso dos dados com dimensões 3x512 em sua entrada, alguns trabalhos descreveram bons resultados ao utilizar entradas com características similares aos propostos no presente trabalho, tanto para objetivos de classificação quanto regressão (BOAVENTURA, 2021; HA; YUN; CHOI, 2015; GHOLAMREZAII; ALMODARRESI, 2019).

Outros trabalhos fazem uso de dados com dimensões similares, porém concatenando vetores de aceleração a fim de manter um único vetor de entrada, sendo tais estruturas definidas como redes convolucionais unidimensionais. Bons resultados também foram obtidos com essa forma de entrada, utilizando tanto dados de amplitude no domínio do tempo, quanto no domínio da frequência (LEE; YOON; CHO, 2017; DJAMAL; FURI; NUGRAHA, 2019).

Conforme observado em tais trabalhos, as maiores variações no desempenho das redes empregadas em tarefas similares à proposta estão relacionadas aos hiperparâmetros ligados às camadas convolucionais. Redes com apenas uma camada densamente conectada após as etapas de convolução demonstraram bom desempenho, tanto em tarefas de regressão, quanto classificação.

Dessa forma, optou-se por manter a estrutura de saída da rede utilizada para a primeira RNC, voltada à classificação de imagens gráficas, e realizar-se uma busca por melhores HPs apenas nas camadas de convolução. Para isso, foi utilizada a técnica de *Gridsearch*, varrendo um espaço de possíveis hiperparâmetros para as camadas iniciais da rede. Os HPs e as faixas avaliadas são apresentados na Tabela 6.

Tabela 6 – Espaço de Hiperparâmetros Testados.

Hiperparâmetro	**Faixa**
Camadas de Convolução	1 ; 2
Quantidade de Filtros	4 ; 8 ; 16
Tamanho dos Filtros	(2,1) ; (3,2) ; (4,1) ; (8,2)
Batch Size	16 ; 32

Fonte: Próprio autor

Os hiperparâmetros selecionados para a varredura foram escolhidos, tomando-se como base aqueles aplicados nos trabalhos já mencionados relacionados na área, possibilitando assim uma redução significativa na quantidade de configurações testadas, ao selecionar-se os que apresentaram melhor desempenho em suas estruturas finais.

Em relação aos demais HPs, estes foram fixados com características similares à RNC classificadora de imagens. O otimizador definido para todas as redes foi o *Adam*, e a função de todas as camadas convolucionais foi a *ReLU*. Um *Dropout* de 20% foi aplicado após cada camada de convolução e densa. A operação de *flattening* foi aplicada antes

de adentrar a camada densa, sendo esta última composta por 64 neurônios ativados pela função *ReLU*. Novamente, para a camada de saída, a função de ativação *Softmax* foi utilizada, por se tratar do mesmo problema de classificação com quatro classes.

Para este caso também foi empregada a técnica de *augmentation* nos dados de treinamento da rede a fim de se tentar aumentar sua capacidade de predição e reduzir as chances de ocorrência de *overfitting*. A mesma faixa de $\pm 10\%$ de deslocamento horizontal randômico foi aplicada.

O limite máximo de épocas estipulado para treinamento durante a varredura realizada pelo espaço de hiperparâmetros foi de 80 épocas. Ao término das combinações possíveis, foi selecionada a estrutura que apresentou melhor acurácia na classificação dos dados provenientes do banco de teste.

A estrutura final foi então submetida a um novo treinamento até que o limite de 200 épocas fosse atingido ou alguma função de *callback* fosse acionada. Para esse caso, a função de *Early Stopping* também foi utilizada, sendo programada para interromper o treinamento, quando a função de perda não fosse reduzida por mais de 50 épocas consecutivas.

3.4.3 Ambiente de Desenvolvimento

Assim como na etapa de preparação dos dados, todo o processamento computacional relacionado às redes neurais foi desenvolvido na linguagem de programação *Python*, em sua versão 3.9. Novamente sua seleção foi guiada levando em consideração as facilidades de implementação proporcionadas por ela, bem como sua grande aplicação na área de *Machine Learning*.

Além das bibliotecas mencionadas anteriormente, destaca-se aqui a biblioteca *TensorFlow* (ABADI et al., 2015) em sua versão 2.7.0, utilizada para estruturação e treinamento das redes. Essa biblioteca foi escolhida devido à sua extensa documentação, contando com apoio e desenvolvimento da empresa Google, suporte para computação utilizando GPUs, sendo atualmente uma das bibliotecas mais populares para desenvolvimento de redes neurais.

Toda etapa de estruturação e treinamento da rede foi desenvolvida em um notebook rodando o sistema operacional Windows 11. O hardware utilizado conta com um processador i7 11800H com frequência base de 2,30GHz, 32GB de memória RAM, e uma placa de vídeo dedicada GeForce RTX 3060 da fabricante NVIDIA com 6GB de VRAM.

4 RESULTADOS E DISCUSSÕES

Neste capítulo serão apresentados os resultados em relação ao conjunto de aquisição desenvolvido e ao treinamento das estruturas propostas para as RNCs com seus dois diferentes métodos de entrada, avaliando-se, ainda, o desempenho das duas redes ao classificarem dados anteriormente não vistos.

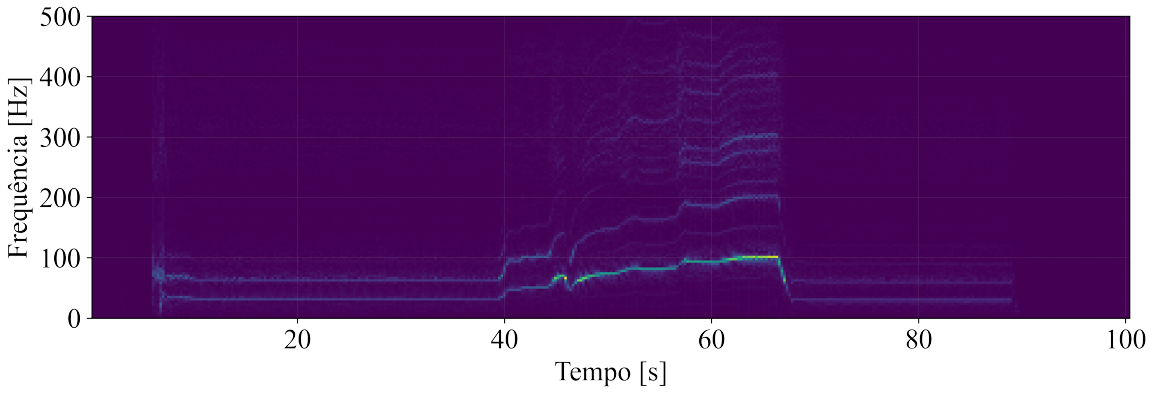
4.1 Correspondência dos Dados Coletados

Conforme já mencionado, as coletas dos sinais de vibração do motor para treinamento e avaliação da ocorrência de falha de ignição foram realizadas em condições estáticas, ou seja, com o veículo em repouso e o motor operando em marcha lenta. Entretanto, para avaliar a correspondência do sinal lido pelo acelerômetro com a real rotação do motor, foram gerados espectrogramas em condição de variação da rotação, realizando a aceleração e desaceleração dos motores conforme mencionado no Capítulo 3.

Uma vez que o motor de combustão interna de quatro cilindros e quatro cursos opera de forma com que a cada volta completa do virabrequim são gerados dois pulsos de pressão por par de cilindros defasados entre si, é sabido que em condições normais de operação a frequência predominante do motor deva ser o dobro da rotação de seu eixo do virabrequim. Assim, para a comparação realizada entre os sinais provenientes do acelerômetro e os parâmetros lidos da rede CAN do veículo, os valores de rotação fornecidos pela ECU foram multiplicados por dois, ajustando as unidades de medida.

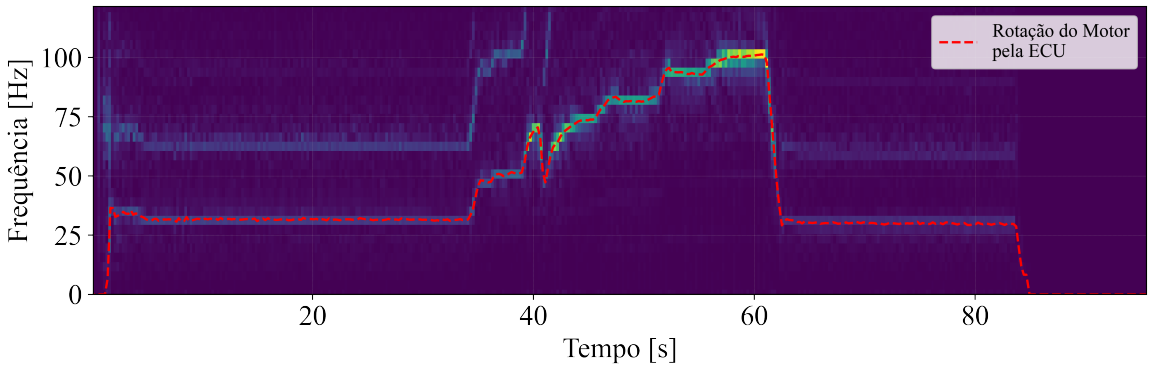
Sabendo isso, um espectrograma do sinal de vibração amostrado na direção X foi gerado para o primeiro motor, mostrado na Figura 14, sendo feita uma sobreposição com os dados da ECU como apresentado na Figura 15. O eixo da frequência na Figura 15 foi restringido para uma melhor visualização da área de interesse do espectro, referente a frequência característica do motor.

Figura 14 – Espectrograma do sinal de aceleração utilizando o acelerômetro MEMS para o motor do primeiro veículo.



Fonte: Próprio autor

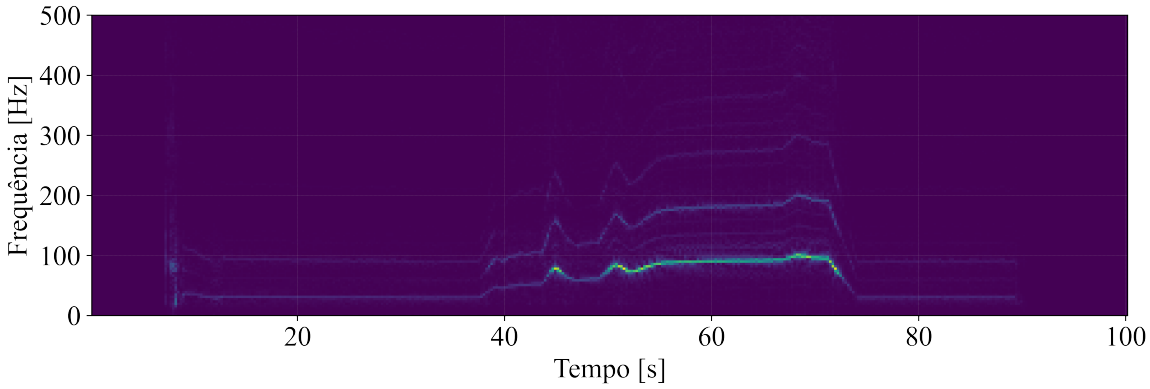
Figura 15 – Comparação do espectrograma com a rotação real do motor obtida pela ECU do primeiro veículo.



Fonte: Próprio autor

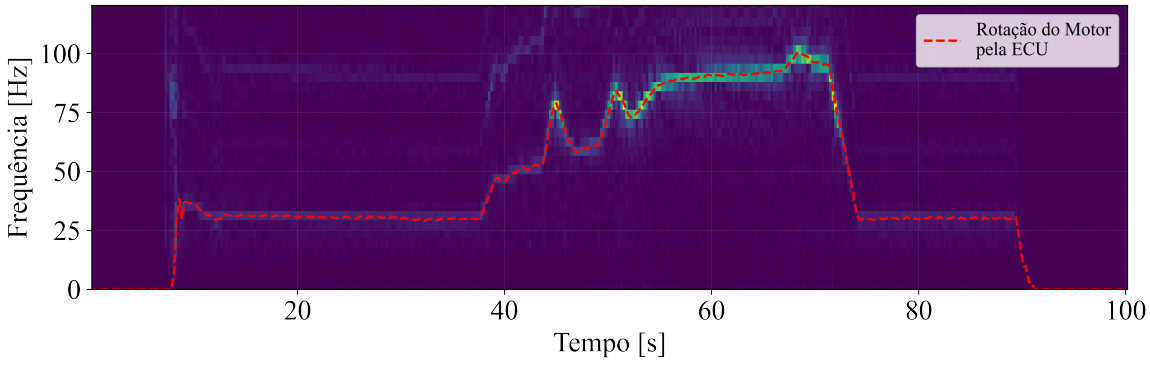
Os mesmos gráficos foram gerados com os dados obtidos para o segundo motor, sendo apresentado na Figura 16 o espectrograma do sinal amostrado com o acelerômetro MEMS, e na Figura 17 a sobreposição com os dados da ECU do segundo veículo.

Figura 16 – Espectrograma do sinal de aceleração utilizando o acelerômetro MEMS para o motor do segundo veículo.



Fonte: Próprio autor

Figura 17 – Comparação do espectrograma com a rotação real do motor obtida pela ECU do segundo veículo.



Fonte: Próprio autor

Conforme esperado, tomando como base os trabalhos de Ribeiro e Lameiras (2019) e Bismor (2019), o acelerômetro MEMS capacitivo selecionado foi capaz de suportar as condições de operação impostas, trabalhando sem problemas no interior do cofre dos dois motores. Além disso, sua máxima frequência de amostragem, o limite de ±8g configurado, e a forma de fixação aplicada, mostraram-se suficientes para descrever a frequência característica dos motores ao longo dos ensaios. Os dados coletados da ECU se sobrepuseram exatamente sobre as bandas de frequência com maiores amplitudes ao longo to tempo por todo o sinal amostrado para ambos os motores.

São perceptíveis algumas harmônicas da frequência característica nos espectrogramas gerados. Tais componentes também foram notadas por Bismor (2019), tanto para o acelerômetro MEMS ADXL345 quanto para o PCB-622B01, piezoelétrico utilizado como referência, sendo possivelmente originadas devido aos demais componentes periféricos dos motores, como comando de válvulas e acessórios conectados ao virabrequim. Assim, foi possível concluir-se que o conjunto confeccionado estava apto para coletar dados de entrada representativos para as etapas seguintes relacionadas ao treinamento das RNCs.

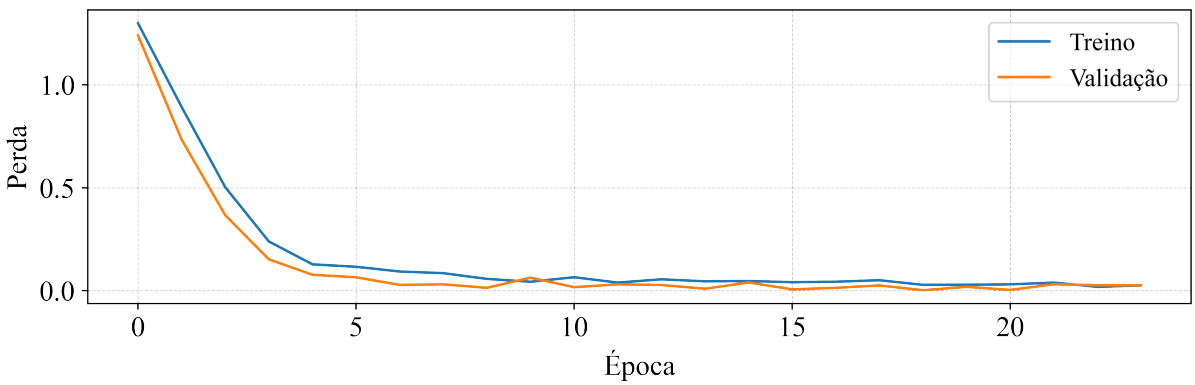
4.2 Treinamento das RNCs e Classificação

Com o conjunto de aquisição validado, pôde-se dar início às etapas de treino das redes neurais convolucionais.

4.2.1 RNC com Entrada de Gráficos

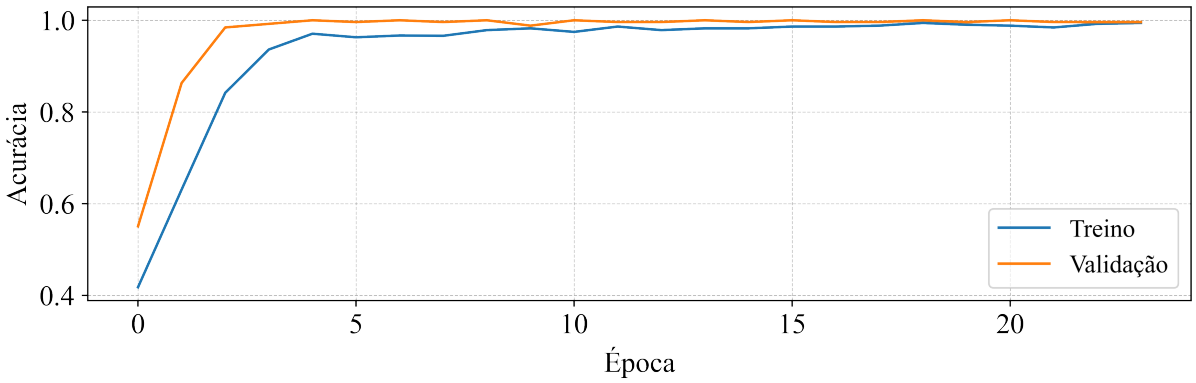
Para visualizar o desempenho da rede durante seu treinamento, os valores da função de perda e acurácia ao longo das épocas foram armazenados, possibilitando gerar seus gráficos ao término do processo, sendo estes mostrados respectivamente pelas Figuras 18 e 19.

Figura 18 – Curva de Perda durante treinamento da RNC com entradas em formato de imagens.



Fonte: Próprio autor

Figura 19 – Curva de Acurácia durante treinamento da RNC com entradas em formato de imagens.



Fonte: Próprio autor

Avaliando os gráficos anteriores de perda e acurácia é possível avaliar o bom desempenho da estrutura proposta para a rede neural convolucional classificadora. Embora não tenha sido feita uma busca por melhores hiperparâmetros para sua composição, a estrutura proposta tomando como base HPs comumente utilizados em trabalhos com propostas similares de classificação demonstrou bom resultado em convergência.

Conforme pode ser visto no gráfico da Figura 18, a função de perda atingiu um valor mínimo para os dados de validação na época 13 e, devido ao critério de paciência definido como 10 épocas, a função de *Early Stopping* foi acionada, interrompendo o treinamento antes de atingir valor máximo estipulado como 50 épocas. De forma similar, o modelo apresentou acurácia elevada antes também da ocorrência das dez primeiras épocas, sendo os pesos da rede armazenados sempre que uma melhor acurácia no banco de validação fosse encontrada.

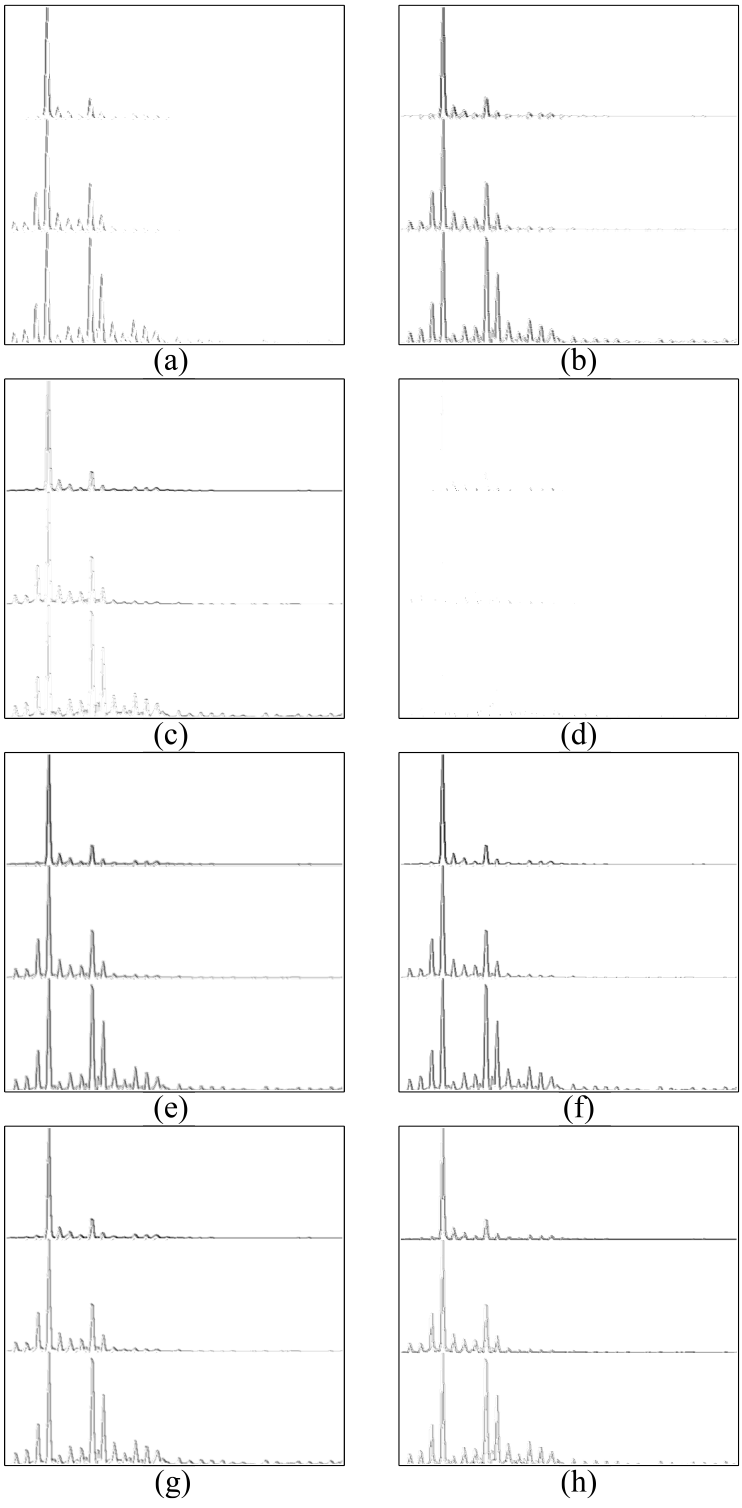
Assim como descrito na seção 3.4.1, às técnicas de *Dropout*, *Augumentation* e *Early Stopping* foram utilizadas a fim de tentar evitar a ocorrência de *overfitting*. Analisando as curvas obtidas ao longo do treinamento da rede, pode-se afirmar que não é notável qualquer enviesamento para os dados de treino, as curvas dos dados de validação acompanharam as características das curvas de treinamento. A rede apresentou excelente desempenho na classificação de dados anteriormente não vistos.

Em relação ao tempo demandado para treino, o tempo médio empregado para conclusão de cada uma das épocas foi de 7,2 segundos. Tal velocidade pode ser entendida ao analisarem-se os dados de entrada, pois, embora as imagens apresentem dimensões de 512x512, boa parte da matriz de entrada é composta por zeros, o que facilita as multiplicações de matrizes realizadas ao longo do processo de treinamento e classificação.

A RNC, assim como outros algorítimos de *Machine Learning*, são consideradas caixas pretas. Para tentar entender um pouco o processo de aprendizado tomado, foram geradas imagens com a saída dos dados após a primeira camada de convolução treinada aplicando

um dado de entrada da classe *F1*. A visualização dos mapas de características gerados pela primeira camada de convolução são mostrados na Figura 20.

Figura 20 – Visualização dos Mapas de Características na saída da primeira camada de convolução da RNC com entrada de imagens.



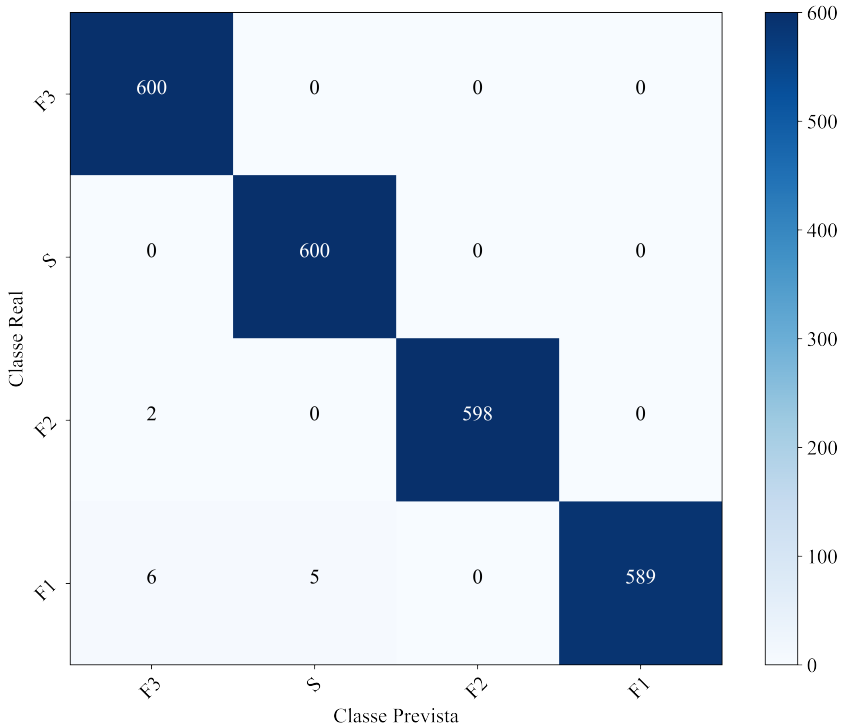
Fonte: Próprio autor

Avaliando-se os Mapas de Características, é possível visualizar que alguns dos filtros selecionados durante o processo de treinamento da rede realizam pouca alteração em relação à imagem original (casos (e), (f), (g) e (h)), enquanto que outros realizam o destaque de linhas horizontais, chegando até à supressão de boa parte da figura, mantendo apenas pontos de espaçamento das frequência sobre os eixos X (caso (d)). Outros, por sua vez, tentam destacar as linhas verticais da imagem (casos (a), (b)), caracterizando os picos de frequência com maiores amplitudes.

No entanto, os mapas de características obtidos a partir da segunda camada de convolução já são mais difíceis de serem apresentados, já que os *kernels* da segunda camada são aplicados sobre cada um dos oito mapas de características gerados pela primeira camada.

Por fim, mas não menos importante, foi confeccionada a Matriz de Confusão aplicando os dados do banco de teste na rede treinada. Os resultados de saída da rede são apresentados na Figura 21. O tempo médio demandado para classificação dos 2400 dados do banco de teste foi de 6,5s, resultando em um tempo médio inferior à 3ms para a classificação de dados individuais após a inicialização da rede.

Figura 21 – Matriz de confusão da RNC treinada com imagens avaliando os dados de teste.



Fonte: Próprio autor

Analisando a matriz de confusão obtida, nota-se um excelente desempenho da RNC classificando dados novos, não vistos na etapa de seu treinamento. A acurácia da classifi-

cação foi superior à 99,4%, classificando corretamente 100% dos dados para as condições de motor saudável S e falha do tipo F3. A rede apresentou certa dificuldade na classificação de falhas do tipo F2, em que dois dos 600 dados foram classificados erroneamente como pertencentes à classe F3, e também do tipo F1, classificando seis dados como do tipo F3 e cinco como do tipo S equivocadamente.

Embora a rede não tenha alcançado uma acurácia de 100%, o resultado pode ser considerado extremamente satisfatório para a abordagem empregada, já que não houve qualquer processo de teste ou seleção de outros possíveis hiperparâmetros melhores para o problema deste caso. A porcentagem de classificações erradas das classes de falhas F1 e F2 ficou abaixo de 0,6%.

4.2.2 RNC com Entrada por Matrizes

Para a segunda RNC, com entradas em formato de matrizes compactas, seu treinamento iniciou com a busca pelos melhores HPs conforme descrito na seção 3.4.2. A técnica de *GridSearch* avaliou todas as combinações apresentadas na Tabela 6. Dentre as possíveis configuração de HPs, a que apresentou maior acurácia no banco de dados de validação após 80 épocas de treinamento é apresentada na Tabela 7.

Tabela 7 – Melhores Hiperparâmetros definidos para a rede por *Gridsearch*.

Hiperparâmetro	**Selecionado**
Camadas de Convolução	1
Quantidade de Filtros	4
Tamanho dos Filtros	(3,2)
Batch Size	32

Fonte: Próprio autor

A estrutura final da RNC combinando os HPs definidos pelo *Gridsearch* com os demais parâmetros pré-definidos é apresentada na Tabela 8.

Tabela 8 – Estrutura final da RNC com entradas em formato compacto.

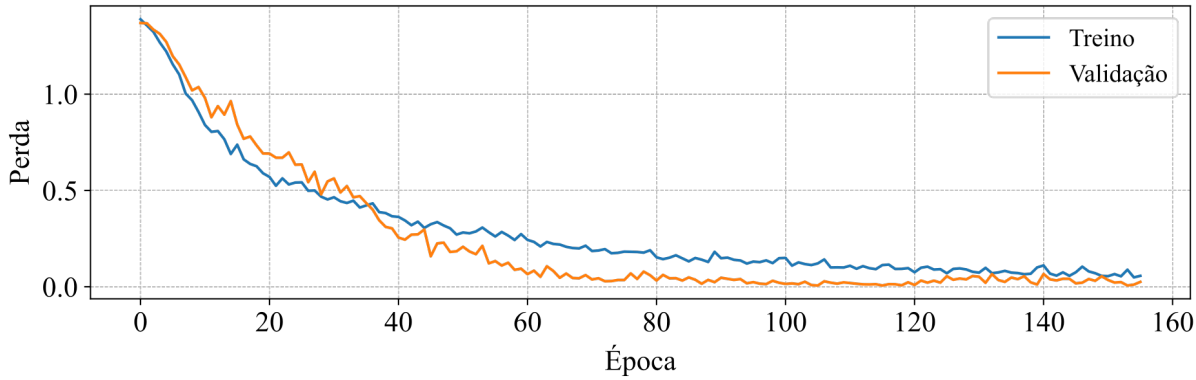
Camada	**Tipo**	**Características**
1	Convolução	4 filtros de dimensão 3x2 - *ReLU*
2	*Dropout*	20% de desligamento
3	*Flattening*	Achatamento para a camada densa
4	Densa	64 neurônios - *ReLU*
5	*Dropout*	20% de desligamento
6	Saída	4 neurônios - *Softmax*

Fonte: Próprio autor

Conforme pode ser observado da Tabela 8, a estrutura final da rede foi composta por apenas uma camada convolucional com 4 filtros de tamanho 3x2 e função de ativação *ReLU*, sendo aplicado um *Dropout* de 20% em sua saída, seguido pelo processo de *Flattening*. O restante da estrutura se manteve o mesmo da primeira RNC classificando imagens, ou seja, com apenas uma camada densa de 64 neurônios com função de ativação *ReLU* e um *Dropout* de 20% antes da camada de saída, sendo esta última composta por quatro neurônios com função de ativação *Softmax*.

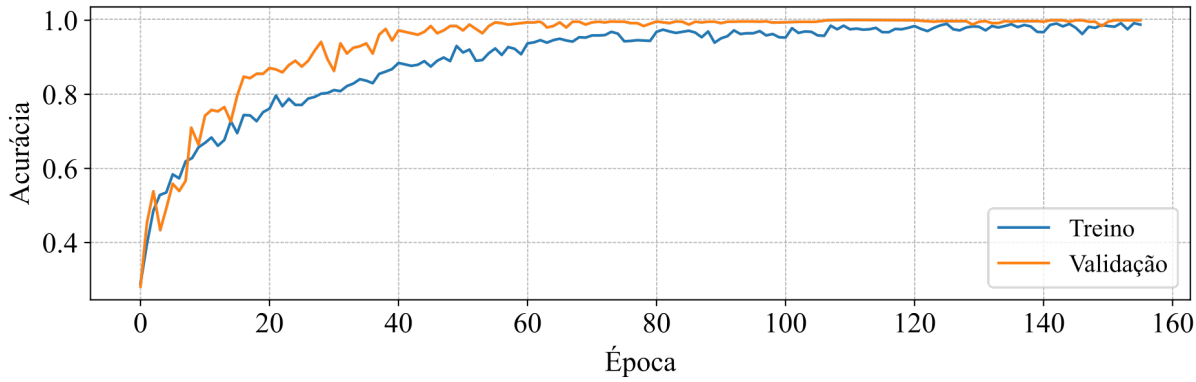
Com a estrutura final da rede definida, fez-se o treinamento até a convergência conforme descrito na seção 3.4.2. Para visualizar o desempenho da RNC durante seu treinamento final, os valores da função de perda e acurácia ao longo das épocas foram armazenados, gerando os gráficos mostrados respectivamente pelas Figuras 22 e 23 após interrupção do processo de treino.

Figura 22 – Curva de Perda durante treinamento final da RNC com entradas em formato compacto.



Fonte: Próprio autor

Figura 23 – Curva de Acurácia durante treinamento final da RNC com entradas em formato compacto.



Fonte: Próprio autor

Conforme pode ser observado nos gráficos, o treinamento foi interrompido antes de atingir o número máximo de 200 épocas definidas como limite de treino. A atualização dos pesos foi finalizada na época 155 com o último *checkpoint* registrado para a função de *Early Stopping* ocorrendo 50 épocas antes.

As curvas obtidas mostram um treinamento ligeiramente mais ruidoso do que o apresentado para a RNC utilizando dados de entrada em forma de imagem, possivelmente devido a forma compacta adotada para a entrada. Contudo, a convergência também foi alcançada de forma satisfatória, reduzindo a função perda para valores próximos de zero e atingindo uma acurácia próxima de um para os dados de validação, assim como no caso anterior. Os HPs selecionados para a rede neural convolucional classificadora demonstraram serem adequados para a nova forma de entrada, apresentando bom desempenho ao classificar dados do mesmo motor não vistos durante o processo de treino.

A ligeira acurácia superior observada para o banco de validação em relação ao banco de treino pode ser justificada devido à taxa de *Dropout* aplicada, já que apenas os dados proveniente do banco de treinamento são submetidos ao desligamento aleatório dos neurônios. Outro fator influente para essa ocorrência é a técnica de *Augumentation* que também atua somente sobre os dados de treino da RNC.

Assim como no caso anterior, é plausível dizer que não houve a ocorrência de *overfitting* durante o treinamento da rede, não observando um enviesamento para os dados do banco de treino, possivelmente graças às técnicas de *Dropout*, *Augumentation* e *Early Stopping*. A rede apresentou excelente desempenho na classificação de dados anteriormente não vistos.

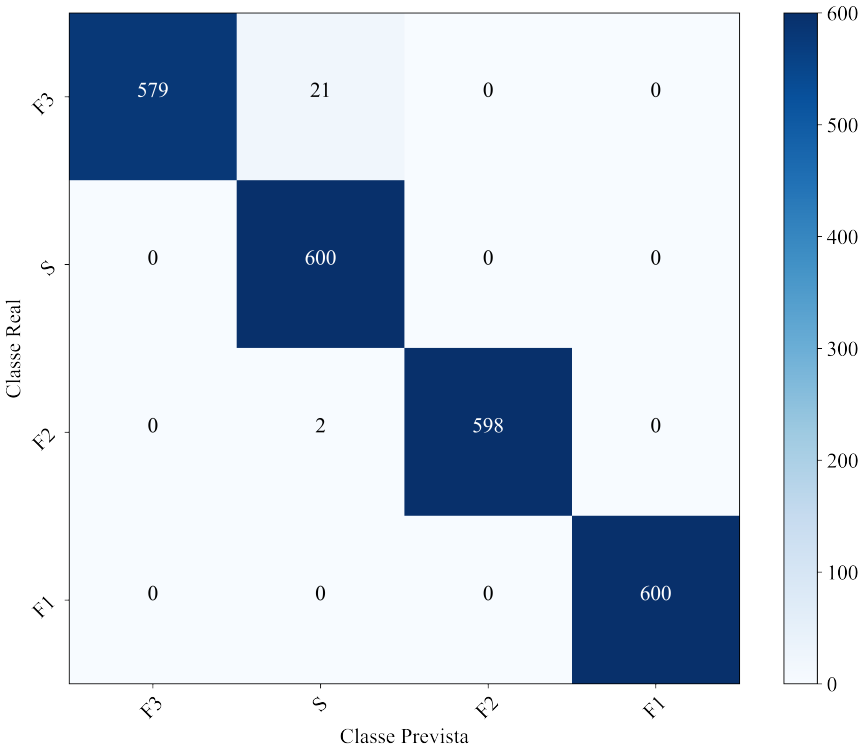
Já em relação ao tempo demandado por época para treinamento da estrutura final da rede, este foi significativamente menor do que o obtido para a RNC com entradas em formato de gráficos. O tempo médio necessário para conclusão de cada uma das épocas foi inferior a 1,02 segundos, representando uma melhora de 85% no tempo de treino por época em comparação com o caso anterior.

Essa vantagem em relação ao tempo de treinamento era esperada, já que houve uma significativa diminuição na dimensão dos dados para o segundo método de entrada. Os dados em formato de matriz apresentam uma redução de 99% em relação à dimensão total dos dados em formato de gráficos. Entretanto, como o número de épocas demandadas para atingir o critério de convergência foi superior para esse caso utilizando dados em formato de matriz compacta, a redução no tempo total de treinamento da rede foi inferior a 10 segundos.

Por fim, após o término da etapa de treinamento da segunda RNC, foi confeccionada a Matriz de Confusão, aplicando para isso os dados armazenados no banco de teste. Os resultados obtidos com a classificação feita pela rede são apresentados na Figura 24. O tempo médio demandado para classificar os 2400 dados do banco de teste foi de aproximadamente $1s$, resultando em um tempo médio inferior a $418\mu s$ para classificação de

cada um dos dados individualmente após a inicialização da rede. Esse resultado descreve uma melhora de sete vezes no tempo individual de classificação dos dados em relação ao tempo demandado para o caso anterior, utilizando gráficos como forma de entrada.

Figura 24 – Matriz de confusão da RNC treinada com matrizes avaliando os dados de teste.



Fonte: Próprio autor

Avaliando a Matriz de Confusão é possível notar desempenho equiparável ao apresentado pela RNC da seção 3.4.1. Mesmo utilizando dados em forma compacta na sua entrada, a nova estrutura apresentou elevada acurácia, sendo capaz de classificar corretamente 99% dos dados de teste.

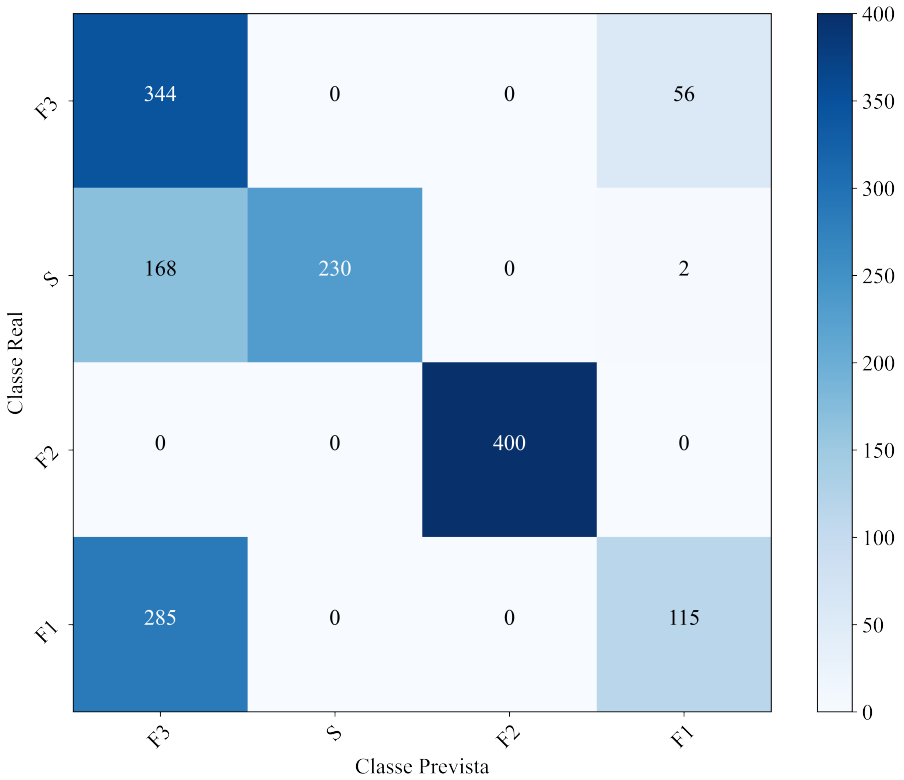
Em relação ao desempenho individual por classe, a rede classificou corretamente 100% dos dados para as condições de motor saudável S e falhas do tipo F1. Já para as outras duas classes, a rede apresentou certa dificuldade classificando falhas do tipo F3, rotulando equivocadamente 21 dados como pertencentes a classe S, enquanto que para as falhas do tipo F2, dois dos 600 dados também foram definidos como saudáveis erroneamente.

De forma geral, o resultado obtido pode ser avaliado como extremamente satisfatório, já que foi mantida uma acurácia acima de 99% na classificação de novos dados apresentando ainda um desempenho superior em relação ao tempo de classificação e uso de recursos computacionais. A porcentagem de classificações erradas das classes de falhas F2 e F3 ficou abaixo de 1%.

4.3 Classificação de Falhas do Segundo Motor

Por fim, para a última etapa de avaliação das RNCs treinadas, empregou-se os dados coletados com o motor do segundo veículo utilizado como plataforma de testes, conforme descrito na seção 3.2. Vale ressaltar que nenhuma das duas redes foram treinadas com dados provenientes do segundo motor. A Matriz de Confusão obtida a partir da primeira RNC, utilizando dados de entrada coletados do segundo motor em forma de gráficos, é apresentada pela Figura 25.

Figura 25 – Matriz de confusão da RNC treinada com imagens avaliando os dados do segundo motor.



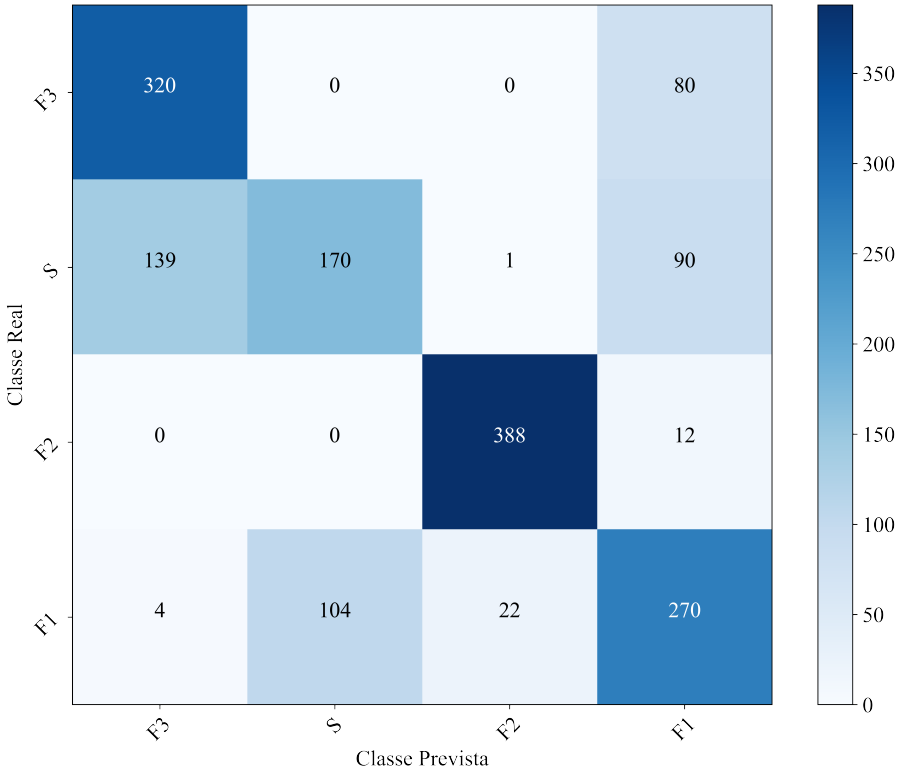
Fonte: Próprio autor

Avaliando a Matriz de Confusão gerada é possível observar uma maior dificuldade da rede em classificar dados obtidos a partir do sinal de vibração do segundo motor. A acuracidade alcançada para esse caso foi de 68,1%, desempenho aproximadamente 31% inferior àquele obtido classificando dados não vistos provenientes do mesmo motor com o qual a rede foi treinada.

A RNC alcançou acurácia elevada apenas para a classe F2, referente aos dados de motor com falha contínua em cilindros adjacentes na sequência de combustão, em que classificou 100% dos dados corretamente. Já para as demais classes F3, S e F1 as porcentagens de acerto foram 86,0%, 57,7% e 28,8%, respectivamente.

Já para a segunda RNC, os mesmos dados do segundo motor foram utilizados para avaliação, porém em formato de matrizes compactas, sendo as classificações de saída apresentadas pela Matriz de Confusão da Figura 26.

Figura 26 – Matriz de confusão da RNC treinada com matrizes avaliando os dados do segundo motor.



Fonte: Próprio autor

Conforme pode ser observado, o desempenho obtido com a RNC classificando dados do segundo motor em forma compacta foi ligeiramente superior ao da primeira rede. A acurácia geral alcançada foi de 71,8%.

Novamente a classe que apresentou maior acurácia foi a F2, identificando corretamente 97% dos dados referentes a ela. Já para as classes F3, F1 e S as porcentagens de acerto foram 80,0%, 67,5% e 42,5%, respectivamente.

Já que todos os conjuntos de dados utilizados contaram com a mesma quantidade para cada uma das quatro classes, a maior porcentagem de acerto na classe F2 não pode ser justificada por um desbalanceio entre as classes do conjunto de treinamento. Dessa forma, é plausível dizer que a classe com maior semelhança entre os sinais dos dois motores seja a de falha contínua em cilindros adjacentes na sequência de combustão, visto que ambas as redes apresentaram boa capacidade de classificar corretamente essa condição de operação dos motores.

De forma geral, o desempenho de ambas as redes podem ser considerados satisfatórios

ao classificar dados do segundo motor, ressaltando os seguintes pontos:

- A forma de acoplamento do acelerômetro sobre a capa de proteção do motor, aplicada para atender os critérios definidos na seção 3.1.1, dificulta a generalização dos sinais de vibração do motor, já que pode modificar significativamente o espectro de frequência de acordo com o estado de fixação da capa;
- Embora o motor do segundo veículo tenha exatamente as mesmas configurações, compartilhe peças e seja do mesmo ano de fabricação do utilizado no primeiro veículo, as condições de uso e manutenção diferem;
- A fixação do motor ao chassis do segundo veículo, embora feita de forma similar ao primeiro, conta com coxins diferentes em relação à geometria e posicionamento do primeiro veículo;

Cabe aqui mencionar que algumas ações poderiam ser tomadas como tentativa de aumentar a acuracidade na classificação de dados do segundo motor, buscando contornar os pontos listados anteriormente, como por exemplo:

- Alterar a forma de fixação do acelerômetro, caso haja a possibilidade, acoplando-o diretamente sobre o cabeçote dos motores;
- Realizar um novo treino das redes, utilizando certa quantidade de dados do segundo motor, buscando inserir características exclusivas ou significativas ao novo aprendizado das redes;
- Continuar o treinamento das RNCs com dados referentes as condições de operação do segundo motor, evitando assim uma nova inicialização randômica dos pesos e aproveitando parte do aprendizado já desenvolvido para o primeiro motor.

5 CONCLUSÕES

Com o desenvolvimento da pesquisa e considerando os objetivos propostos para o trabalho, pode-se concluir que:

- Quanto ao conjunto de aquisição confeccionado:
  - O acelerômetro MEMS capacitivo selecionado se mostrou robusto, conseguindo operar sem qualquer ocorrência de falhas nas condições em que foi submetido;
  - O microcontrolador ESP32 escolhido para trabalhar como placa de aquisição foi adequado para captura e armazenamento de dados de forma contínua;
  - O conjunto de aquisição foi capaz de capturar a frequência característica dos MCIs. Os dados cruzados com a rede CAN dos veículos demonstraram direta relação e correspondência entre os sinais.
- Quanto ao treinamento e desempenho das RNCs propostas:
  - Tanto os HPs diretamente selecionados para a primeira estrutura de RNC quanto aqueles definidos por *Gridsearch* para a segunda estrutura demonstraram ser bons parâmetros para o problema de classificação proposto;
  - A RNC com entradas em formato de imagens apresentou acuracidade superior a 99,4% ao classificar gráficos, convergindo em apenas 13 épocas, enquanto que a RNC com entradas em formato compacto também apresentou acuracidade superior a 99%, porém demandando 105 épocas para convergência;
  - Embora o tempo total de treinamento tenha sido próximo para ambas as estruturas de redes, o tempo médio para classificação de dados individuais foi sete vezes menor para a rede com dados em forma de matriz compacta;
  - A acurácia das redes classificando dados do segundo motor ficou abaixo de 72%. A atualização dos pesos utilizando uma parcela de dados do segundo motor poderia ser tomada como alternativa para aumentar o desempenho classificatório das redes nesse caso;
  - Mesmo com o sensor fixado de forma não convencional, os dados de entrada coletados com o uso do acelerômetro MEMS mostraram-se suficientes para a classificação dos tipos de falhas propostos relacionados à ocorrência de *misfire*, verificando a robustez das RNCs na identificação de padrões nessa aplicação.

5.1 Sugestões para Trabalhos Futuros

Como sugestões para trabalhos futuros na área, alguns itens podem ser listados:

- Alterar o posicionamento do sensor MEMS, fixando-o diretamente sobre o cabeçote dos dois motores; Avaliar o desempenho das mesmas estruturas de redes sendo treinadas e testadas com os novos dados gerados a partir dos sinais coletados com o novo posicionamento;
- Ajustar os pesos das redes já treinadas com dados provenientes do segundo motor objetivando uma melhor acurácia ao classificar dados originados desse;
- Avaliar o desempenho de RNCs utilizando dados de entrada ainda em formato gráfico, porém com dimensões reduzidas;
- Comparar o desempenho da RNC utilizando dados em forma de matriz compacta em relação a estruturas convolucionais 1D, empregando dados tanto no domínio do tempo quanto no domínio da frequência.

REFERÊNCIAS

ABADI, M. et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. Software available from tensorflow.org. Disponível em: https://www.tensorflow.org/.

ABEDIN, M. et al. An overview on comparative engine performance and emission characteristics of different techniques involved in diesel engine as dual-fuel engine operation. *Renewable and Sustainable Energy Reviews*, Oxford, v. 60, p. 306–316, 2016. ISSN 1364-0321. Dis-ponível em: https://www.sciencedirect.com/science/article/pii/S1364032116001489.

ALBARBAR, A. et al. Suitability of mems accelerometers for condition monitoring: An experimental study. *Sensors*, Molecular Diversity Preservation International, Basel, v. 8, n. 2, p. 784–799, 2008.

ALBARBAR, A.; TEAY, S. H. MEMS accelerometers: Testing and practical approach for smart sensing and machinery diagnostics. In: *Advanced Mechatronics and MEMS Devices II*. Springer International Publishing, 2016. p. 19–40. Disponível em: https: // doi.org/10.1007/978-3-319-32180-6 2.

ANFAVEA. *O caminho da descarbonização do setor automotivo no Brasil*. São Paulo, 2021. Disponível em: https://anfavea.com.br/docs/apresentacoes/APRESENTA%C3%87%C3%83O-ANFAVEA-E-BCG.pdf. Acesso em: 8 de nov. de 2021.

BABU, D.; I, R. K.; SUGUMARAN, V. Misfire detection in spark ignition engine using support vector machines. *International Journal of Computer Applications*, New York, v. 5, 08 2010.

BARINO, F.; BESSA, A. Rede neural convolucional 1d aplicada à previsão da vazão no rio madeira. In: . [*S.l.: s.n.*], 2020.

BHADANE, G. et al. Misfire detection of automotive engines with convolutional neural network. In: *Techno-Societal 2018*. [*S. l.*]: Springer, 2020. p. 121–131.

BIANCO, M. J. et al. Machine learning in acoustics: Theory and applications. *The Journal of the Acoustical Society of America*, v. 146, n. 5, p. 3590–3628, 2019. Disponível em: https://doi.org/10.1121/1.5133944.

BISMOR, D. Analysis and comparison of vibration signals from internal combustion engine acquired using piezoelectric and mems accelerometers. *Vibrations in Physical Systems*, Politechnika Poznańska. Instytut Mechaniki Stosowanej, v. 30, n. 1, p. 1–8, 2019.

BOAVENTURA, O. D. Z. *Localização de vazamentos em tubulações enterradas usando uma rede Nneural de convolução*. 66 p. Dissertação (Mestrado) — Universidade Esta-dual Paulista (Unesp), Ilha Solteira, 2021. Disponível em: http://hdl.handle.net/11449/217311. Acesso em: 24 de nov. de 2021.

BRUNETTE, E.; FLEMMER, R.; FLEMMER, C. A review of artificial intelligence. In: . [*S. l.: s. n.*], 2009. p. 385–392.

CARLEY, L. *OBD II Engine Diagnostics: Not as Simple as it Seems*. 2008. ImportCAR https://www.import-car.com/obd-ii-engine-diagnostics-not-as-simple-as-it-seems/. Acesso em: 12 nov. 2021.

CARLEY, L. *How To Diagnose & Repair Engine Misfires*. 2019. AA1Car https: //aa1car.com/library/misfire.htm. Acesso em: 12 de nov. de 2021.

CARLEY, L. *How to Diagnose a Controller Area Network (CAN)*. 2020. AA1Car https: //www.aa1car.com/library/can systems.htm. Acesso em: 12 de nov. de 2021.

CECCARANI, M.; REBOTTINI, C.; BETTINI, R. *Engine misfire monitoring for a V12 engine by exhaust pressure analysis*. [*S. l.*], 1998.

COPPIN, B. *Artificial Intelligence Illuminated*. Jones and Bartlett Publishers, 2004.(Jones and Bartlett illuminated series). ISBN 9780763732301. Disponível em: https: // books.google.com.br/books?id=LcOLqodW28EC.

DJAMAL, E. C.; FURI, W. I.; NUGRAHA, F. Detection of eeg signal post-stroke using fft and convolutional neural network. In: *2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*. [*s. n.*], 2019. p. 18–23. Disponível em: https://ieeexplore.ieee.org/abstract/document/8977126.

DUARTE, F.; LIBARDI, R.; SÁNCHEZ, K. *Introdução à mobilidade urbana*. [*S. l.*]: Juará Editora, 2012. 108 p. ISBN 978-8536216737.

DUMOULIN, V.; VISIN, F. *A guide to convolution arithmetic for deep learning*. 2018.

ESPRESSIF. *ESP32 Series Datasheet*. 2021. Disponível em: https://espressif.com/sites/default/files/documentation/esp32 datasheet en.pdf. Acesso em: 28 de maio de 2021.

FAN, Q. et al. Misfire detection and re-ignition control by ion current signal feedback during cold start in two-stage direct-injection engines. *International Journal of Engine Research*, v. 15, p. 37–47, 01 2012.

FERGUSON, C. *Internal combustion engines : applied thermosciences*. 3. ed. Chichester, West Sussex, United Kingdom: John Wiley & Sons, Inc, 2016. ISBN 978-1118533314.

FIAT. *FIAT Punto Handbook*. 2008. Disponível em: https://servicos.fiat.com.br/content/dam/fiat/products/handbooks/118/2008/handbook-2008-punto.pdf. Acesso em: 28 de maio de 2021.

GHOLAMREZAII, M.; ALMODARRESI, S. M. T. Human activity recognition using 2d convolutional neural networks. In: *2019 27th Iranian Conference on Electrical Engineering (ICEE)*. [s.n.], 2019. p. 1682–1686. Disponível em: https://ieeexplore.ieee.org/abstract/document/8786578.

GILCHRIST, A. *Industry 4.0 : the industrial internet of things*. Berkeley, CA New York: Springer, 2016. ISBN 978-1-4842-2047-4.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. *In*: . [*S. l.: s. n.*], 2010. v. 15.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. http://www.deeplearningbook.org.

HA, S.; YUN, J.-M.; CHOI, S. Multi-modal convolutional neural networks for activity recognition. In: *2015 IEEE International Conference on Systems, Man, and Cyberne-tics*. [s.n.], 2015. p. 3017–3022. Disponível em: https://ieeexplore.ieee.org/abstract/document/7379657.

HAYKIN, S. S. *Neural networks and learning machines*. Upper Saddle River, N.J: Pearson, 2009. ISBN 9780131293762.

HUNTER, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007.

HUTTER, F.; KOTTHOFF, L.; VANSCHOREN, J. *Automated Machine Learning*. 1. ed. asel, Switzerland: Springer International Publishing, 2019. (The Springer Series on Challenges in Machine Learning). ISBN 9783030053185. Disponível em: https://link.springer.com/book/10.1007/978-3-030-05318-5.

IBGE. *Frota de veículos - Brasil*. Rio de Janeiro: [s.n.], 2020. Disponível em: https: // cidades.ibge.gov.br/brasil/pesquisa/22/28120. Acesso em: 8 nov. 2021.

INCE, T. et al. Real-time motor fault detection by 1d convolutional neural networks. v. 63, n. 11, p. 7067–7075, 11 2016. Disponível em: https://ieeexplore.ieee.org/abstract/document/7501527.

INVENSENSE. *MPU-6000 and MPU-6050 Product Specification Rev.3.4*. 2013. Online Handbooks, invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf. Acesso em: 28 de maio de 2021. Rev. 3.4.

JUNIOR, O. d. S.; SOUZA, M. T. S. d. Regulation as inductor of environmental technologies for the reduction of toxic emissions from light vehicles in brazil. *Cadernos EBAPE. BR*, SciELO Brasil, v. 16, p. 748–760, 2018.

KARDEC, A.; NASCIF, J. *Manutenção-função estratégica*. [*S. l.*]: Qualitymark Editora Ltda, 2001.

KARUPPASAMY, A. et al. Recent cnn-based techniques for breast cancer histology image classification. *The Journal of Engineering Research [TJER]*, Seoul, v. 19, n. 1, p. 41–53, 2022. Disponível em: https://journals.squ.edu.om/index.php/tjer/article/view/4087.

KAVLAKOGLU, E. *AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the Difference?* [*S. l.*]: IBM, 2020. https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks. Acesso em: 11 de dez. de 2021.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. Disponível em: https://arxiv.org/abs/1412.6980.

KOBBACY, K. A.; MURTHY, D. P. et al. *Complex system maintenance handbook*. [*S.l.*]: Springer, 2008.

KORVINK, J. G.; PAUL, O. (Ed.). *MEMS: A Practical Guide to Design, Analysis, and Applications*. [S.l.]: Springer Berlin Heidelberg, 2006.

LECUN, Y. A. *et al.* Efficient backprop. *In*: . *Neural Networks: Tricks of the Trade: Second Edition*. Berlin, Heidelberg: Springer, 2012. p. 9–48. ISBN 978-3-642-35289-8. Disponível em: https://doi.org/10.1007/978-3-642-35289-8\3.

LEE, K.-F. *Inteligência artificial*. [*S. l.*]: Globo Livros, 2019. ISBN 978-6580634323.

LEE, S.-M.; YOON, S. M.; CHO, H. Human activity recognition from accelerometer data using convolutional neural network. In: *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*. [s.n.], 2017. p. 131–134. Disponível em: https: // ieeexplore.ieee.org/abstract/document/7881728.

LEIJNEN, S.; VEEN, F. The neural network zoo. v. 47, p. 9, 05 2020. Disponível em: https://www.mdpi.com/2504-3900/47/1/9.

LIMA, J. J. et al. Dispositivo para análise dinâmica da marcha humana utilizando sensores inerciais mems. *Revista de Engenharia e Tecnologia*, Ponta Grossa, v. 5, n. 3, p. Páginas–122, 2013. Disponível em: https://revistas.uepg.br/index.php/ret/article/view/11501.

LIU, J. et al. Fault detection for gas turbine hot components based on a convolutional neural network. *Energies*, Basel, v. 11, n. 8, 2018. ISSN 1996-1073. Disponível em: https: // www.mdpi.com/1996-1073/11/8/2149.

MARTINS, J. *Motores Combustão Interna*. 3. ed. [*S. l.*]: Pearson, 2011. ISBN 978-972-8953-85-0.

MORO, N.; AURAS, A. P. *Introdução à Gestão da Manutenção*. Florianópolis, Santa Catarina, 2007. Apostila.

MOUBRAY, J. *Introdução à manutenção centrada na confiabilidade*. [*S. l.: s. n.*], 1996. 129 p.

MUI, A. *How a Car Works - The engine*. 2016. Disponível em: https://www. howacarworks.com/basics/the-engine. Acesso em: 11 de nov. de 2021.

MULAY, S.; SUGUMARAN, V.; DEVASENAPATI, B. Misfire detection in i.c. engine using machine learning approach. *Pakistan Journal of Biotechnology*, Pakistan, v. 15, p. 102–106, 01 2018.

OPPENHEIM, A. V.; SCHAFER, R. W. *Digital signal processing*. Englewood Cliffs, N.J: Prentice-Hall, 1975. 585 p. ISBN 0132141078.

O'SHEA, K.; NASH, R. *An Introduction to Convolutional Neural Networks*. 2015. Dispo-n ível em: https://arxiv.org/abs/1511.08458.

PANG, Y. et al. Convolution in convolution for network in network. *IEEE Transactions on Neural Networks and Learning Systems*, v. 29, n. 5, p. 1587–1597, 2018.

RASCHKA, S. *Python machine learning : unlock deeper insights into machine learning with this vital guide to cutting-edge predictive analytics*. Birmingham, UK: Packt Pu-blishing, 2015. 454 p. ISBN 9781783555130.

REBACK, J. et al. *pandas-dev/pandas: Pandas 1.4.0*. Zenodo, 2022. Disponível em: https://zenodo.org/record/5893288.

REBELLO, J.; CLEGHORN, W. L.; MILLS, J. K. Design and analysis of a low frequency mems vibration sensor for automotive fault detection. *International Journal of Vehicle Design*, Geneva, v. 54, n. 2, p. 93–110, 2010.

RIBEIRO, R. R.; LAMEIRAS, R. d. M. Evaluation of low-cost mems accelerometers for shm: frequency and damping identification of civil structures. *Latin American Journal of Solids and Structures*, São Paulo, v. 16, n. 7, 2019.

RISI, S.; TOGELIUS, J. Neuroevolution in games: State of the art and open challenges. *IEEE Transactions on Computational Intelligence and AI in Games*, IEEE Computer Society, Los Alamitos, CA, USA, v. 9, n. 01, p. 25–41, jan 2017. ISSN 1943-0698.

ROBERTS, A.; BROOKS, R.; SHIPWAY, P. Internal combustion engine cold-start efficiency: A review of the problem, causes and potential solutions. *Energy Conversion and Management*, London, v. 82, p. 327–350, 2014. ISSN 0196-8904. Disponível em: https: // www.sciencedirect.com/science/article/pii/S0196890414001939.

ROCHA, R. L. et al. Image inspection of railcar structural components: An approach through deep learning and discrete fourier transform. In: *Anais do VII Symposium on Knowledge Discovery, Mining and Learning*. Porto Alegre, RS, Brasil: SBC, 2019. p. 33–40. ISSN 2763-8944. Disponível em: https://sol.sbc.org.br/index.php/kdmile/article/view/8786.

SANTOS, C. H. T. *Redes Neurais Aplicadas à Modelagem de Instrumentos. Acústicos para Síntese Sonora em Tempo Real*. 2018. 88 f. Dissertação (Mestrado) — Universidade Federal Fluminense, Niterói, 2018. Disponível em: https://app.uff.br/riuff/handle/1/7613. Acesso em: 11 de dez. de 2021.

SHAHUL, A. B. E. S. *Hyperparameter Tuning in Python: a Complete Guide - neptune.ai*. 2022. Disponível em: https://neptune.ai/blog/hyperparameter-tuning-in-python-complete-guide. Acesso em: 21 de jul. de 2022.

SHAO, M.; WANG, J.; WANG, S. The intelligent fault diagnosis of diesel engine based on the ensemble learning. *Journal of Physics: Conference Series*, Bristol, v. 1549, p. 042106, 06 2020.

TEZ, S.; AKIN, T. Fabrication of a sandwich type three axis capacitive mems accelerometer. In: *SENSORS, 2013 IEEE*. [s.n.], 2013. p. 1–4. ISSN 1930-0395. Disponível em: https://ieeexplore.ieee.org/abstract/document/6688598.

WALT, S. van der; COLBERT, S. C.; VAROQUAUX, G. The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, Piscataway, v. 13, n. 2, p. 22–30, mar. 2011. Disponível em: https://doi.org/10.1109/mcse.2011.37.