Universidade Federal do Rio Grande do Norte

Centro de Tecnologia - CT

Programa de Pós-Graduação em Engenharia Mecatrônica

Detecção e diagnóstico de falhas em rolamentos, sob diferentes cargas e velocidades, utilizando Redes Neurais Convolucionais

Wallisson Fernandes Martins dos Santos

Natal-RN, Brasil

2023

Universidade Federal do Rio Grande do Norte

Centro de Tecnologia - CT

Programa de Pós-Graduação em Engenharia Mecatrônica

Detecção e diagnóstico de falhas em rolamentos, sob diferentes cargas e velocidades, utilizando Redes Neurais Convolucionais

Wallisson Fernandes Martins dos Santos

Orientador: Fábio Meneghetti Ugulino de Araújo

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Mecatrônica da UFRN como parte dos requisitos para obtenção do título de Mestre em Engenharia Mecatrônica.

Natal-RN, Brasil

2023

Universidade Federal do Rio Grande do Norte - UFRN
Sistema de Bibliotecas - SISBI
Catalogação de Publicação na Fonte. UFRN - Biblioteca Central Zila Mamede

Santos, Wallisson Fernandes Martins Dos.
Detecção e diagnóstico de falhas em rolamentos, sob diferentes cargas e velocidades, utilizando Redes Neurais Convolucionais / Wallisson Fernandes Martins Dos Santos. - 2023.
85f.: il.

Dissertação (Mestrado) - Universidade Federal do Rio Grande do Norte, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Mecatrônica, Natal, 2023.
Orientador: Dr. Fábio Meneghetti Ugulino de Araújo.

1. Detecção e diagnóstico de falhas - Dissertação. 2. Rede neural convolucional - Dissertação. 3. Técnicas de inteligência artificial - Dissertação. 4. Falhas em rolamentos - Dissertação. 5. Aprendizado de máquinas - Dissertação. I. Araújo, Fábio Meneghetti Ugulino de. II. Título.

RN/UF/BCZM CDU 004

Elaborado por Raimundo Muniz de Oliveira - CRB-15/429

Detecção e diagnóstico de falhas em rolamentos, sob diferentes cargas e velocidades, utilizando Redes Neurais Convolucionais

Wallisson Fernandes Martins dos Santos

Dissertação de Mestrado aprovada em 26 de julho de 2023 pela banca examinadora composta pelos seguintes membros:

_____
Prof. Dr. Fábio Meneghetti Ugulino de Araújo (orientador) . . . . . .DCA/UFRN

_____
Prof. Dr. Carlos Eduardo Trabuco Dórea . . . . . .DCA/UFRN

_____
Prof. Dr. Pablo Javier Alsina. . . . . .DCA/UFRN

_____
Prof. Dr. Marcelo Roberto Bastos Guerra Vale. . . . . DET/UFERSA

Natal-RN, Brasil

2023

“Se você quer ser bem sucedido, precisa ter dedicação total, buscar seu último limite e dar o melhor de si mesmo.”
Ayrton Senna

Agradecimentos

Primeiramente agradeço a Deus que é minha força, meu refúgio e por sempre me mostrar o caminho certo.

A minha mãe Josefa Martins pela compreensão e apoio incondicional ao longo de minha vida. Também por ser responsável por meu caráter e educação.

A minha esposa Amanda, pelo seu amor, pelo carinho e compreensão em todos os momentos. Por sempre estar presente, por vibrar com cada uma de minhas conquistas e, principalmente, por acreditar, incentivar e apoiar minhas escolhas.

A minha filha e princesinha, Maria Fernanda, que veio dar um novo sentido à minha vida, pelo amor incondicional, por ser exemplo de superação, por sua força de vontade e por demonstrar, em cada olhar, que posso vencer desafios.

Agradeço ao meu orientador, professor Dr. Fábio Meneghetti Ugulino de Araújo, por aceitar conduzir o meu trabalho de pesquisa, pela contribuição e empenho para que este trabalho pudesse ser realizado.

A minha sogra Rosimeire, pelo apoio e por sempre estar disposta a ajudar, cedendo, inúmeras vezes, parte do seu tempo para ajudar minha esposa e filha, permitindo que eu pudesse ter tempo para realizar as atividades do mestrado.

Ao colega de curso Thiago Brito, pelo compartilhamento de conhecimentos e por sempre estar disposto a ajudar.

Aos professores do Programa de Pós-Graduação em Engenharia Mecatrônica, pela elevada qualidade do ensino oferecido.

A todos que de alguma forma contribuíram para a realização deste trabalho, o meu muito obrigado!

Resumo

Com o aumento da complexidade e dos custos dos sistemas industriais, medidas de gestão que visam impedir ou mitigar a perda de confiabilidade, diminuição da produtividade e riscos de segurança, provocados por anormalidades de processo e falhas de componentes, tornam-se cada vez mais importantes. Nesse contexto, a Inteligência Artificial (IA) vem se consolidando como um meio eficaz e desafiador no processo de monitoramento, detecção e diagnóstico de falhas em equipamentos e sistemas industriais. Dentre os equipamentos, que são frequentemente objeto de estudos, destacam-se os rolamentos, que são componentes mecânicos críticos das máquinas rotativas. O monitoramento de vibração é a técnica mais amplamente utilizada para detectar, localizar e distinguir falhas em rolamentos. Diante do desempenho eficiente e crescente das técnicas IA e da importância dos rolamentos nos processos industriais, este trabalho implementa uma Rede Neural Convolucional (CNN) para Detecção e Diagnóstico de Falhas (DDF) em rolamentos, sob diferentes cargas e velocidades no motor e diferentes tipos e profundidade de falhas no rolamento. Para o desenvolvimento da abordagem proposta, foi utilizado o banco de dados de ensaios em rolamentos da Case Western Reserve University (CWRU). Os sinais de vibração brutos foram pré-processados através da Transformada Wavelet Continua (TWC) e convertidos em imagens, as quais foram alimentadas diretamente na estrutura CNN desenvolvida. Quando comparado com outros métodos baseados em CNN que utilizaram o mesmo banco de dados, a abordagem proposta demonstrou superioridade ou foi pelo menos tão bem-sucedido quanto, atingindo uma precisão de 97,7% quando testado com arquivos em condições operacionais diferentes das condições de treinamento.

Palavras chaves: detecção e diagnóstico de falhas, rede neural convolucional, técnicas de inteligência artificial, falhas em rolamentos, aprendizado de máquinas.

Abstract

With the increasing complexity and costs of industrial systems, management measures aimed at preventing or mitigating the loss of reliability, decreased productivity and safety risks, caused by process abnormalities and component failures, become increasingly important. . In this context, Artificial Intelligence (AI) has been consolidating itself as an effective and challenging means in the process of monitoring, detecting and diagnosing failures in equipment and industrial systems. Among the equipment, which are frequently the object of studies, bearings stand out, which are critical mechanical components of rotating machines. Vibration monitoring is the most widely used technique for detecting, locating and distinguishing bearing faults. Faced with the efficient and increasing performance of AI techniques and the importance of bearings in industrial processes, this work implements a Convolutional Neural Network (CNN) for Detection and Diagnosis of Faults (DDF) in bearings, under different loads and speeds in the motor and different types and depth of bearing failures. For the development of the proposed approach, the Case Western Reserve University (CWRU) bearing test database was used. The raw vibration signals were pre-processed through the Continuous Wavelet Transform (TWC) and converted into images, which were fed directly into the developed CNN structure. When compared to other CNN-based methods that used the same database, the proposed approach demonstrated superiority or was at least as successful, achieving an accuracy of 97.7% when tested with files under operating conditions other than operating conditions. training..

Keywords: failure detection and diagnosis, convolutional neural network, artificial intelligence techniques, bearing failures, machine learning.

Lista de ilustrações

Lista de tabelas

Sumário

1 Introdução

Os sistemas industriais estão se tornando cada vez mais sofisticados, complexos e caros, menos tolerantes às falhas, à diminuição da produtividade e aos riscos de segurança às instalações, pessoas e meio ambiente. Conforme Dai e Gao (2013), isso leva a uma exigência cada vez maior de confiabilidade e segurança dos sistemas de controle sujeitos às falhas. Como um meio eficaz para garantir a confiabilidade e segurança dos sistemas industriais e reduzir o risco de paradas não planejadas, a Detecção e Diagnósticos de Falhas (DDF) tem sido objeto de interesse tanto na comunidade acadêmica quanto na industrial e encontra seu sucesso em muitas áreas de engenharia.

A detecção e diagnostico de falhas é um processo complexo, onde, em muitos casos, a investigação adequada de uma falha envolve várias disciplinas e o analista de falhas deve reconhecer e usar todos os conhecimentos relevantes para detectar, isolar e identificar a falha. Esse processo, segundo Huet (2002), deve ser realizado independentemente do tamanho dos danos causados por uma falha, da quantidade de tempo e esforço investido em investigá-la e não importando o tamanho do projeto. A partir das informações coletadas, um analista de falhas tenta descobrir o que foi fundamentalmente responsável pela falha (BHAUMIK, 2009).

1.1. Início da detecção e diagnóstico de falhas

Através de pesquisas em diversos periódicos, foram encontrados estudos sobre análise de falha em equipamentos com data de 1917, no qual Harper (1917) preocupado com o aumento dos custos dos materiais de transmissão elétrica e a crescente demanda do setor, fez uma análise sobre as causas de falhas em cabos elétricos subterrâneos para distribuição de energia. Assumindo que os cabos fossem de boa qualidade e projeto, foram analisadas três possíveis causas fundamentais de falha: problemas nas juntas, lesões mecânicas nas bainhas de chumbo e superaquecimento. Dentre as três causas fundamentais analisadas, o superaquecimento foi o problema mais importante a ser considerado, sendo proposto alternativas de melhorias.

A detecção de falhas também passou a ser embutida nos projetos de equipamentos/sistemas, a exemplo de Meyer e Wehrend (1975), onde em um

relatório da NASA foi descrita a necessidade, no projeto de um sistema integrado de controle de voo, de uma lógica de controle capaz de detectar e identificar, automaticamente, falhas em vários subsistemas e alternar (quando necessário) para a próxima estratégia de controle mais segura. A abordagem algorítmica era aplicável a uma grande classe de aeronaves e esperava-se que testes para avaliação da lógica de controle ocorresse no ano seguinte (1976).

Willsky (1976) examinou uma série de métodos para a detecção de falhas em sistemas dinâmicos estocásticos, onde afirmou que o meio mais confiável de detecção de falhas é a votação direta entre instrumentos semelhantes. Segundo o autor, não se encontram erros introduzidos ao comparar as saídas de dispositivos diferentes. No entanto, paga-se o preço da redundância de *hardware* para implementar um esquema de votação. Além disso, a votação tem suas limitações internas, como instrumentos podem não ser exatamente iguais e pode-se ter que usar outros instrumentos para compensar essas discrepâncias. A redundância de instrumentos pode ser útil para identificar a localização da falha, porém, conforme Sakurada (2001), na maioria dos casos é desejável atuar antes que a falha ocorra, pois segundo Abdul-Nour *et al.* (1998), o custo de uma falha durante a operação do equipamento é muito maior do que o custo para substituição antecipada do componente.

Conforme relatado por Isermann (1997), o crescente interesse no campo de detecção e diagnostico de falhas foi levado em consideração pela *International Federation of Automatic Control - IFAC*, criando em 1991 um Comitê Diretor de SAFEPROCESS (Detecção de falhas, supervisão e segurança para processos técnicos) que se tornou um Comitê Técnico em 1993. O Simpósio SAFEPROCESS é um grande encontro internacional de especialistas líderes da academia e da indústria de todo o mundo, que visa fortalecer o contato entre a academia e a indústria. O encontro é organizado a cada três anos, sendo que a última edição aconteceu em Pafos, Chipre, entre 8 e 10 de junho de 2022.

A literatura sobre detecção e diagnósticos de falhas é enorme e diversificada, principalmente devido a uma grande variedade de sistemas, componentes e peças. Centenas de artigos nesta área, incluindo teorias e aplicações práticas, aparecem todos os anos em revistas acadêmicas, anais de conferências e relatórios técnicos.

1.2. Detecção e diagnóstico de falhas e inteligência artificial

Conforme Pyun *et al.* (2020), no ambiente industrial moderno, com o desenvolvimento da tecnologia de sensoriamento físico, inteligência artificial e sistemas distribuídos, as plantas industriais expandiram-se gradualmente e geraram grandes quantidades de dados de processo, abrindo caminho para o desenvolvimento de métodos inteligentes de detecção e diagnósticos de falhas. Em particular, aplicações de última geração contam com o processamento rápido e eficiente de falhas no equipamento/sistema, resultando em aumento de produção e redução de tempos de parada.

Na literatura, várias ferramentas ou técnicas de IA foram usadas no campo da detecção e diagnósticos de falhas. Segundo Liu *et al.* (2018), as técnicas de IA incluem otimização convexa, otimização matemática, bem como métodos baseados em classificação, aprendizado estatístico e probabilidade. Especificamente, classificadores e métodos de aprendizado estatístico têm sido amplamente utilizados no diagnóstico de falhas de máquinas rotativas, incluindo algoritmos *K-Nearest Neighbor* (KNN), Máquina de Vetor de Suporte (MVS) e Rede Neural Artificial (RNA). Desde 2015, segundo Liu *et al.* (2018), as abordagens de aprendizado profundo, em especial a Rede Neural Convolucional (CNN), também começaram a ser aplicadas no campo do diagnóstico de falhas em máquinas rotativas. Na sequência serão demonstrados alguns trabalhos utilizando IA para DDF.

Aplicando IA, Vicente *et al.* (2001) descreveram um sistema de diagnóstico automático para detecção e classificação de falhas em rolamentos utilizando Lógica *Fuzzy* (LF). O sistema projetado foi desenvolvido para poder classificar três tipos de falhas pré-estabelecidos na pista externa (pit, corrosão e superfície riscada) com os rolamentos operando sob diversas velocidades e condições de carga no eixo. Um rolamento perfeito também foi usado para representar a condição normal. O sistema *Fuzzy* proposto apresentou melhores resultados quando comparado com os resultados de uma Rede *Perceptron* Multicamada (MLP) e uma Rede Neural Probabilística (RNP), com diagnostico de 97% do banco de dados de teste (distinção entre condição normal e com falha) e 95% de acertos na classificação entre as quatro classes de falha (normal, pit, corrosão e superfície riscada).

Soualhi, Medjaher e Zerhouni (2014) apresentaram uma abordagem que combina a Transformada de Hilbert-Huang (THH), a máquina de vetor de suporte (MVS) e a Regressão de Vetor de Suporte (RVS) para o monitoramento em rolamentos de esferas. A validação experimental foi feita em uma plataforma de envelhecimento acelerado, com defeitos simulados nos rolamentos. O método proposto conseguiu identificar mais rapidamente as falhas que apareceram no rolamento testado, com início em 116 min, já para a técnica que utiliza o valor RMS, o primeiro sinal de degradação apareceu somente após 138 min de teste. A implementação desta abordagem depende da disponibilidade de dados históricos sobre a degradação dos rolamentos.

Wang (2020) propôs um método de diagnóstico automático de falhas em isoladores, utilizando segmentação de instâncias e análise de temperatura de imagens infravermelhas, através da rede neural convolucional e aprendizado por transferência. Para desenvolver o método, foram utilizadas 1200 imagens infravermelhas, do banco de dados de inspeção da State Grid Beijing Power Maintenance Company da China. Primeiramente os parâmetros iniciais da CNN foram pré-treinados com um conjunto de dados criado pela Microsoft, que continha mais de 300.000 instâncias de imagens, 20.000.000 instâncias de objetos comuns e 80 categorias de objetos. Os resultados experimentais mostraram que a precisão média foi de 77%. Através do método proposto, as falhas de superaquecimento puderam ser detectadas e diagnosticadas em tempo real (quando aplicado a veículos aéreos não tripulados ou inspeção de robôs), em vez de serem julgadas manualmente após a coleta de dados no local.

Conforme abordado pelas recentes técnicas de diagnósticos e prognósticos de falhas, Qin *et al.* (2022) afirmam que a análise detalhada e sistemática de uma grande quantidade de dados do processo pode maximizar a produtividade e minimizar os problemas de segurança, levando a um foco crescente no diagnóstico inteligente de falhas. Nesse contexto, as técnicas de Inteligência Artificial (IA) apresentam inúmeras vantagens sobre as abordagens convencionais de diagnóstico de falhas, pois além de melhorar o desempenho, essas técnicas são fáceis de estender e modificar. Além disso, segundo Liu *et al.* (2018) e Siddique, Yafava e Singh (2003), não exigem conhecimento físico prévio completo do modelo, que pode ser difícil de obter na prática e podem ser adaptados pela incorporação de novos dados ou informações.

1.3. Justificativa do trabalho

Nos últimos anos, abordagens inteligentes que vão desde abordagens de aprendizado de máquina até abordagens mais avançadas de aprendizado profundo foram desenvolvidas no campo de DDF. Numerosos estudos têm sido realizados nesta área e muitos métodos têm sido propostos e implementados para detectar a existência e determinar o tipo de falha presente em sistemas/equipamentos. No entanto, conforme Sikder *et al.* (2021), este campo de pesquisa ainda está em aberto, pois há espaço para melhorias nos resultados.

Motivado pelo desempenho eficiente e crescente das técnicas IA para detecção e diagnósticos de falhas, este trabalho propõe uma abordagem aplicando rede neural convolucional (ConvNet ou CNN) para detecção e diagnóstico de falhas em rolamentos de elementos rolantes, sob condições operacionais variáveis (velocidade e carga do motor e diâmetro e localização da falha). A escolha pelo rolamento se deu devido a importância que esse componente representa para a indústria e principalmente para as máquinas rotativas, pois segundo Bazan (2020), as falhas em rolamentos são as maiores responsáveis por paradas das máquinas rotativas, representando entre 40% e 70% das causas associadas a estas paradas. Além disso, conforme Atta *et al.* (2021), a maioria dos métodos DDF em rolamentos relatados na literatura são dedicados ao trabalho em velocidade/carga fixa. Logo, conforme Tayyab *et al.* (2022), torna-se importante o diagnóstico de falhas em rolamento em condições de trabalho variáveis, pois devido às complexas condições de trabalho das máquinas rotativas, a velocidade do eixo geralmente é variável, afetando as características dos sinais de falha, o que torna o sinal do rolamento não estacionário e o diagnóstico inteligente e preciso de falhas um processo desafiador para os pesquisadores.

Para o desenvolvimento da abordagem proposta foram utilizados os dados públicos de ensaio de vibração em rolamentos da Case Western Reserve University (CWRU), extraindo as características representativas da integridade dos rolamentos dos dados de vibração brutos e classificando em quatro condições: normal (sem falhas), falha na esfera, falha na pista externa e falha na pista interna.

1.4. Estrutura do trabalho

Para uma melhor compreensão, este trabalho está organizado da seguinte forma: O capítulo 2 fornece uma visão geral sobre detecção e diagnostico de falhas, inteligência artificial e redes neurais convolucionais. O capítulo 3 descreve a metodologia utilizada para implementar a arquitetura CNN proposta. No capitulo 4 são demonstrados os resultados da metodologia aplicada ao conjunto de dados experimentais da CWRU, sendo apresentadas comparações da arquitetura CNN construída com outros trabalhos no mesmo campo de diagnóstico de detecção de falhas em rolamentos. O capítulo 5 conclui o trabalho enfocando seus pontos fortes e oportunidade de melhorias.

2 Fundamentação Teórica

Neste capítulo, aborda-se, de forma sucinta, o embasamento teórico necessário para a compreensão e elaboração da proposta apresentada. São apresentados os principais conceitos e terminologias relacionados ao campo de detecção e diagnóstico de falhas, inteligência artificial e redes neurais convolucionais. Além disso, uma breve fundamentação sobre a detecção e diagnóstico de falhas em rolamentos é fornecida.

2.1. Detecção e diagnóstico de falhas

Com o aumento da complexidade e dos custos dos sistemas industriais, medidas de gestão que visam impedir ou mitigar as consequências da degradação do desempenho, diminuição da produtividade e riscos de segurança, provocados por anormalidades de processo e falhas de componentes, tornam-se cada vez mais importantes. Diante disso, conforme Dai e Gao (2013), a detecção e diagnostico de falhas surge como um meio eficaz para garantir a confiabilidade e segurança dos sistemas.

Segundo Ma e Jiang (2011), detecção e diagnóstico de falhas é o processo para detectar, isolar e identificar falhas em um sistema. Algumas definições inerentes às atividades de DDF serão descritas na subseção 2.1.1.

2.1.1 Conceitos e terminologias

Para implementar um processo efetivo de detecção e diagnósticos de falhas, os conceitos associados a esse processo precisam ser bem conhecidos. Nesse sentido, serão abordadas definições feitas por diversos autores, bem como as definições da norma brasileira ABNT (NBR 5462-1994), os quais estão relacionados com a confiabilidade e a mantenabilidade.

2.1.1.1 Falha

A definição de falha pode variar dependendo do autor, onde, em muitos casos, o termo falha é traduzido abrangendo os significados de defeitos. Além disso, conforme relatado por Vale (2014), em alguns trabalhos brasileiros, os termos falha (*failure*) e falta (*fault*) se confundem quando comparado com a

literatura internacional. Para um melhor entendimento deste trabalho, esses termos, com suas equivalências, são abordados na sequência:

a) falha (failure): segundo a ABNT (NBR 5462-1994) falha é definida com o término da capacidade de um item desempenhar a função requerida. Para Isermann (1997), falha é uma interrupção permanente da capacidade de um sistema de executar uma função requerida com requisitos de desempenho especificados;
b) falta (fault) ou defeito: Isermann e Bale (1997), definem falta como um desvio não permitido de pelo menos uma propriedade característica ou parâmetro do sistema em relação à condição normal. O defeito, segundo a ABNT (NBR 5462-1994), é qualquer desvio de uma característica de um item em relação aos seus requisitos, ou seja, o componente/sistema poderá apresentar um defeito ou falta e continuar desempenhando a função requerida (ausência de falhas).

Diante das diferentes definições, neste trabalho, assim como Vale (2014), o termo falha será tratado como uma sendo situações indesejáveis, independente de ocorrer ou não a parada do funcionamento do equipamento/sistema.

2.1.1.2 Detecção e diagnóstico de falhas

Os termos associados à definição da detecção e diagnostico de falhas serão definidos na sequência:

a) detecção de falhas: segundo Shui *et al.* (2009), o objetivo principal da detecção de falhas é analisar vários sintomas que indicam a diferença entre o status normal e o defeituoso. Ou seja, conforme Isermann (1997) e Jardine, Lin e Banjevic (2006), a detecção de falhas é uma tarefa para indicar se algo está errado no sistema monitorado;
b) diagnóstico de falhas: segundo Ma e Jiang (2011) e diversos outros autores, o diagnóstico de falhas inclui isolamento e identificação de falhas. Ou seja:
    - isolamento de falhas: o principal objetivo do isolamento de falhas é localizar o componente que está com em falha. Conforme Shui *et al*. (2009), o procedimento de isolamento de falhas é baseado nos

sintomas analíticos e heurísticos observados e no conhecimento a priori do processo;

- identificação de falhas: segundo Isermann (1997) e Jardine, Lin e Banjevic (2006) a identificação de falhas é uma tarefa para determinar a natureza da falha quando ela é detectada;

2.1.2 Métodos de detecção e diagnósticos de falhas

Conforme Dai e Gao (2013), a detecção e diagnostico de falhas é feito através de algum tipo de modelagem, processamento de sinal e inteligência computacional. Ma e Jiang (2011), Gao, Cecat e Ding (2015), Xu *et al.* (2017) e diversos outros autores classificam os métodos de detecção e diagnóstico de falhas em métodos baseados em modelos, métodos baseados em sinais, métodos baseados em conhecimento e métodos híbridos (métodos de combinação de pelo menos dois métodos). No Quadro 1 são demonstrados os três métodos de DDF e algumas técnicas pertencentes a esses métodos.

Quadro 1 - Métodos de DDF e exemplo de técnicas

Métodos baseados em modelo	**Métodos baseados em sinais**	**Métodos baseados em conhecimento**
Equações de paridade	Transformada de Fourier de Tempo Curto (TFTC)	Redes Neurais Artificiais (RNA)
Filtros de Kalman (FK)	Análise de Tempo-Frequência (ATF)	Lógica *Fuzzy* (LF)
Observador Proporcional Integral (OPI)	Transformada Wavelet (TW)	Análise de Componentes Principais (ACP)
Estimativa de parâmetros	Modelo de sinal Autorregressivo (AR)	Análise Qualitativa de Tendências (AQT)
Desigualdades Matriciais Lineares (DML)	Transformada de Hilbert-Huang (THH)	Mínimos Quadrados Parciais (MQP)

Fonte: Autoria própria.

O método híbrido não está sendo representado no Quadro 1 pois ele é uma combinação de uma ou mais técnicas dos métodos baseados em modelos, métodos baseados em sinais e/ou métodos baseados em conhecimento. Todos esses métodos são descritos, resumidamente, nas subseções 2.1.2.1 a 2.1.2.4.

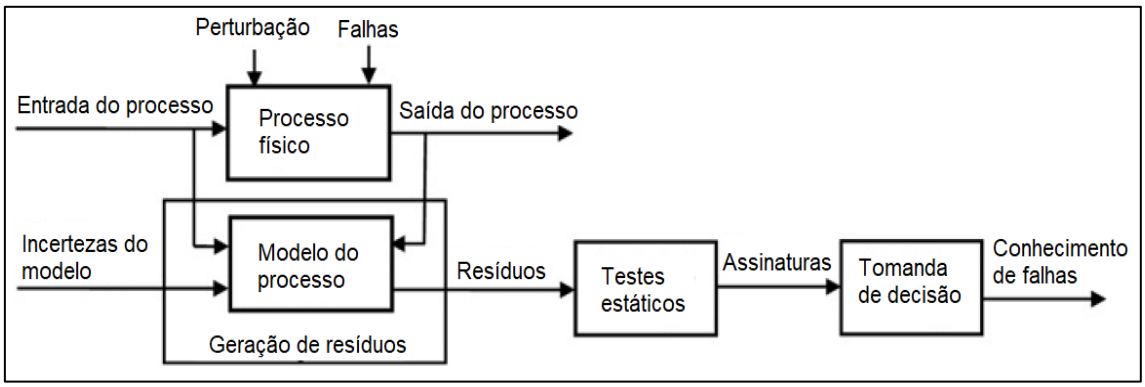
2.1.2.1 Métodos baseados em modelos

Segundo Gao, Cecat e Ding (2015), o diagnóstico de falhas baseado em modelo foi originado por Beard em 1971 para substituir a redundância de hardware por redundância analítica que, segundo Willsky (1976) e Chow e Willsky (1984), é o conceito central da maioria dos DDFs baseados em modelos.

Nos métodos baseados em modelos é necessário que o modelo matemático dos processos industriais ou dos sistemas estejam disponíveis, o que pode ser obtido por meio de princípios físicos ou técnicas de identificação de sistemas. Com base no modelo, algoritmos de diagnóstico de falhas são desenvolvidos para monitorar a consistência entre as saídas medidas dos sistemas observado e as saídas previstas pelo modelo. Conforme Ma e Jiang (2011), as diferenças entre os dados estimados analiticamente e as medidas reais são chamadas de resíduos onde, segundo Gertler (1988), as falhas resultam de alterações das relações normais representadas no modelo, levando a alterações estatisticamente anormais nos resíduos. Portanto, as falhas podem ser detectadas testando esses resíduos estatisticamente.

Para Ma e Jiang (2011) e resumido na Figura 1, os processos de DDF baseados em modelos podem ser divididos nos seguintes subsistemas: geração de resíduos, avaliação de resíduos e tomada de decisão.

Figura 1 - Esquema de DDF baseado em modelo



Fonte: Adaptado de Ma e Jiang (2011).

Jardine, Lin e Banjevic (2006), Ma e Jiang (2011) e Abid, khan e Iqbal (2021) defendem que se um modelo matemático explícito, correto e preciso do sistema for construído, as abordagens baseadas em modelo podem ser mais eficazes do que abordagens sem modelo. No entanto, a modelagem matemática pode não ser viável para sistemas complexos, pois seria muito difícil ou mesmo impossível construir modelos matemáticos para tais sistemas. Além disso, falhas que não foram consideradas na fase de modelagem podem não ser detectadas.

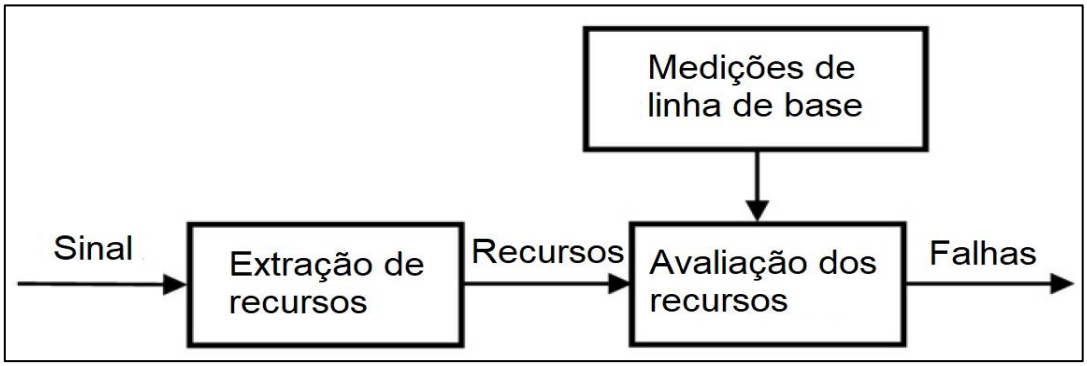
2.1.2.2 Métodos baseado em sinais

Os métodos baseados em sinais tomam decisões de DDF comparando os sinais medidos no processo com sinais de referência conhecidos. Segundo Gao, Cecat e Ding (2015), as falhas no processo são refletidas nos sinais medidos, cujas características são extraídas, e uma decisão diagnóstica é tomada com base na análise de sintomas e no conhecimento prévio dos sintomas dos sistemas saudáveis. Conforme Dai e Gao (2013), na DDF baseado em sinais, a relação entre o sinal de saída e as falhas são construídas a partir da compreensão, a priori, pelo ser humano.

As análises de DDF baseados em sinais são particularmente interessantes para motores e máquinas rotativas e se concentram principalmente em sinais eletrônicos e vibrações. Os sinais característicos a serem extraídos para o processo de detecção e diagnóstico de falhas podem ser no domínio do tempo (por exemplo, média, tendências, desvio padrão, fases, inclinação e magnitudes, como pico e raiz quadrada média) e/ou domínio de frequência (por exemplo, espectro). Além disso, segundo Isermann (2005), modelos de sinais paramétricos (por exemplo, um modelo Auto-Regressivo de Média Móvel) podem ser usados, o que permitem estimar diretamente as frequências principais e suas amplitudes.

Ma e Jiang (2011) descrevem esse método conforme representado na Figura 2.

Figura 2 - Esquema de DDF baseado em sinais



Fonte: Adaptado de Ma e Jiang (2011).

Segundo Dai *et al.* (2019), métodos baseados em sinais têm sido amplamente estudados no diagnóstico inteligente de falhas. Primeiro, ele se beneficia do desenvolvimento de tecnologias de sensores e armazenamento, que permitem que o sistema de monitoramento colete e armazene grandes quantidades de amostras de sinais offline e online. A segunda razão está na constante inovação e desempenho de alto nível alcançado pelos algoritmos de aprendizado de máquina.

2.1.2.3 Métodos baseado em conhecimento

Com o avanço da automação no ambiente industrial moderno, uma grande quantidade de dados e históricos industriais estão disponíveis e, portanto, segundo Abid, khan e Iqbal (2021), métodos baseados em conhecimentos são empregados para fins de monitoramento, detecção e diagnóstico de falhas.

Os métodos baseados em conhecimentos, também conhecidos como métodos baseados nos dados ou histórico de processos, não exigem um modelo ou padrão de sinal conhecido pois, conforme Dai e Gao (2013), a DDF baseada em conhecimento aprende a partir dos dados empíricos para “descobrir” o conhecimento subjacente que representa as características do sistema monitorado. Segundo Gao, Cecat e Ding (2015b), o conhecimento subjacente, que representa implicitamente as características do sistema, pode ser extraído utilizando técnicas de IA ao vasto volume de dados históricos disponíveis. O processo de extração da base de conhecimento pode ser de natureza qualitativa

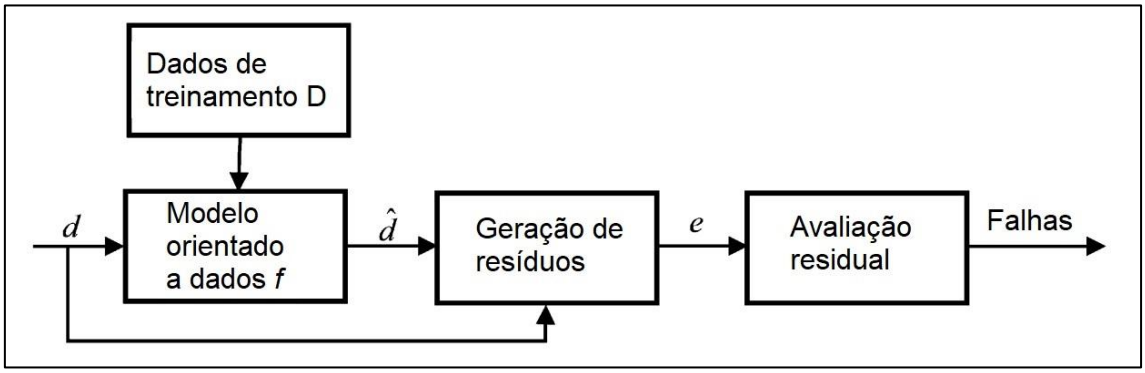
ou quantitativa e a DDF é realizada verificando a consistência do conhecimento subjacente obtido e a funcionalidade do sistema.

Conforme Ma e Jiang (2011), os métodos baseados em conhecimentos também dependem de relacionamentos entre medições correlacionadas dentro de um sistema. No entanto, as relações podem ser formuladas de forma implícita, treinando um modelo empírico através da análise de dados de treinamento sem falhas, obtidos durante operações normais.

Para um melhor entendimento, Ma e Jiang (2011) explicam o método da seguinte forma: suponha que exista uma matriz de dados de treinamento $D \in R^{lxn}$ obtido de um sistema, onde n é o número de variáveis envolvidas, l o número de amostras de dados de treinamento e seja f um modelo empírico treinado usando D. Quando um conjunto de novas medições $d \in R^{lxn}$ torna-se disponível, as estimativas de d podem ser obtidas como $\hat{d} = f(d)$ e os resíduos podem ser gerados como $e = d - \hat{d}$. Quaisquer falhas no sistema causarão mudanças nas relações entre as variáveis em d e isso resulta em mudanças estatisticamente anormais nos resíduos. Consequentemente, as falhas podem ser detectadas e diagnosticadas realizando testes estatísticos nos resíduos.

Ma e Jiang (2011), subdividem os processos de DDF baseados em conhecimentos em: treinamento do modelo, geração de resíduos e avaliação de resíduos. Um esquema de DDF desse método é ilustrado na Figura 3.

Figura 3 - Esquema de DDF baseado em dados



Fonte: Adaptado de Ma e Jiang (2011).

Observe que a Figura 3 mostra o princípio do DDF baseado em conhecimento em um sentido geral. De fato, o modelo f pode ser um mapeamento das variáveis de entrada d_{ent} para as variáveis de d_{sai} que são

subconjuntos de d e consistem em diferentes variáveis. Não é um requisito necessário para f estimar todas as variáveis em d, mas apenas aquelas de interesse para a tarefa de diagnóstico.

Como não são necessários modelos explícitos, os métodos baseados em conhecimentos são mais atraentes para aplicações práticas com sistemas complexos. No entanto, uma limitação importante desses métodos é que o modelo empírico só funciona bem dentro da faixa operacional representada pelos dados de treinamento.

2.1.2.4 Métodos híbridos

Como uma forma de utilizar as melhores característica de cada método (baseado em modelos, orientado por dados ou baseado em sinais) e melhorar o processo de DDF, uma integração ou combinação de dois ou mais métodos de diagnóstico de falhas, chamada de abordagem híbrida, é frequentemente explorada para uma variedade de aplicações de engenharia.

Na sequência serão mostrados alguns exemplos de aplicações utilizando o modelo de DDF híbrido.

Yuan *et al.* (2020), combinaram o método baseado em sinal e o método baseado em conhecimento para monitorar e diagnosticar falhas em rolamentos. O método é baseado em Rede Neural Convolucional (CNN) e Máquina de Vetores de Suporte (MVS). Primeiramente, a Transformada Wavelet Contínua (TWC) é usada para converter sinais de vibração originais unidimensionais em imagens bidimensionais de tempo-frequência. Em segundo lugar, as imagens tempo-frequência obtidas são usadas como entrada para a CNN extrair as características representativas da imagem, usando o método de aprendizagem por transferência. Finalmente, o classificador MVS é treinado usando os recursos extraídos e o diagnóstico da localização e gravidade da falha é concluído.

Em Fang *et al.* (2020), um método híbrido foi proposto para a detecção e diagnóstico de falhas em veículos autônomos. Em primeiro lugar, para detectar se há ocorrência de falhas de estado, a MVS de classe única é usada para treinar a curva limite que separa o domínio seguro e o domínio inseguro. Com base no modelo cinemático do veículo, um observador do Filtro de Kalman foi projetado para prever a posição atual do veículo e, após obter os resíduos entre a previsão e a medição, o teste de Jarque-Bera é aplicado para verificar a normalidade da

distribuição de probabilidade dos resíduos e detectar se há um desvio da trajetória do veículo. Um sistema *Fuzzy* é utilizado para distinguir os tipos de falhas detectadas com base em uma rede neutra modificada, onde a função de pertinência inicial do sistema *Fuzzy* é atualizada, por meio da rede neutra, indicando a probabilidade de cada tipo de falha. Experimentos na plataforma de veículo autônomo `Xinda' e comparação de desempenho com outros detectores de falhas validam a eficácia do método e a usabilidade do sistema de detecção e diagnóstico de falhas.

Ao integrar o processamento de sinais e técnicas baseadas em conhecimentos, um método de detecção e diagnóstico de falhas de curto-circuito entre espiras em motores síncronos de ímã permanente de cinco fases foi abordado por Moosakunju *et al.* (2023). A Transformada Wavelet Discreta (TWD) é usada para extrair os parâmetros de detecção de falhas, a partir da análise das correntes do estator, e o aprendizado de máquina (sistema Lógica *Fuzzy*) é usado para diagnóstico das falhas. Dois ciclos de sinais de corrente do estator são considerados para análise contínua, com a falha sendo detectada e diagnosticada dentro de dois ciclos dessas correntes.

Segundo Dai e Gao (2013), a nova tendência em DDF é integrar várias estratégias para formar uma estrutura hierárquica com uma mistura de vários métodos de DDF homogêneos e/ou heterogêneos. Consequentemente, o estudo do DDF tem sido um campo multidisciplinar envolvendo engenharia de controle, processamento de sinais e inteligência artificial (IA).

A diversidade dos métodos DDF torna difícil para um engenheiro dominar todas as técnicas e tendências em diferentes campos. Em particular, os resultados da IA desempenham e continuarão a desempenhar um papel importante no DDF.

2.2. Inteligência Artificial (IA) na Detecção e Diagnóstico de Falhas

A atual tendência industrial em relação a automatismos e plantas industriais nos leva a sistemas mecatrônicos cada vez mais complexos, trabalhando em um ambiente incerto e evolutivo. Esses sistemas modernos e complexos fazem com que uma quantidade maior de dados industriais e históricos do processo sejam produzidos e isso, segundo Khan *et al.* (2018), resulta em níveis mais altos de incertezas durante os processos de DDF. Para

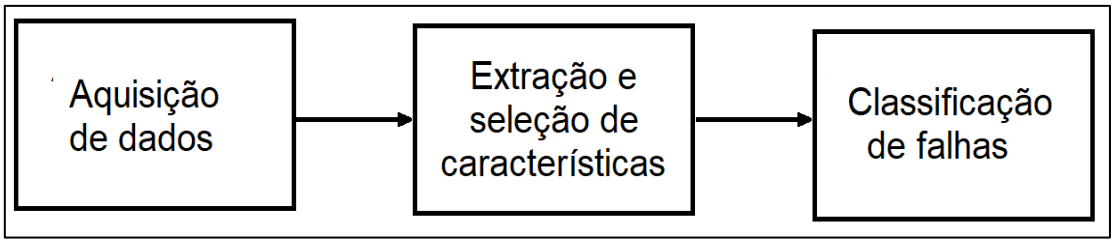
resolver esses desafios impostos pela evolução industrial e aumentar a confiabilidade das análises DDF, técnicas baseadas em Inteligência Artificial (IA) surgem como alternativas.

A Inteligência Artificial é o campo do conhecimento onde se estudam sistemas capazes de reproduzir algumas das atividades mentais humanas (NILSSON, 1986). Ou seja, conforme Nascimento Jr e Yoneyama (2000), a IA busca prover máquinas com a capacidade de realizar algumas atividades mentais do ser humano e, em geral, são máquinas com algum recurso computacional e de variadas arquiteturas. As atividades realizadas por estas máquinas podem envolver a sensopercepção (tato, audição e visão), as capacidades intelectuais (aprendizado de conceitos e de juízos, raciocínio dedutivo e memória), a linguagem (verbais e gráficas) e atenção (decisão no sentido de concentrar as atividades sobre um determinado estímulo).

Segundo Siddique, Yafava e Singh (2003), a Inteligência Artificial, desde o seu surgimento como disciplina em meados da década de 1950, apresenta técnicas com inúmeras vantagens sobre as abordagens convencionais de DDF, pois além de melhorar o desempenho, podem ser adaptadas pela incorporação de novos dados ou informações. Além disso, segundo Lo *et al.* (2019), a IA oferece ferramentas totalmente desacopladas pela estrutura do sistema, não exigindo a modelagem preliminar do sistema.

Para Lei *et al.* (2016), Lei *et al.* (2020) e Nath *et al.* (2021), uma estrutura ideal baseada para diagnostico inteligente de falhas compreende três fases principais: aquisição de dados, extração e seleção de características e classificação de falhas. Essas fases são resumidas na Figura 4 e descritas, por esses autores, na sequência.

Figura 4 - Fases e procedimentos de diagnóstico inteligente de falhas

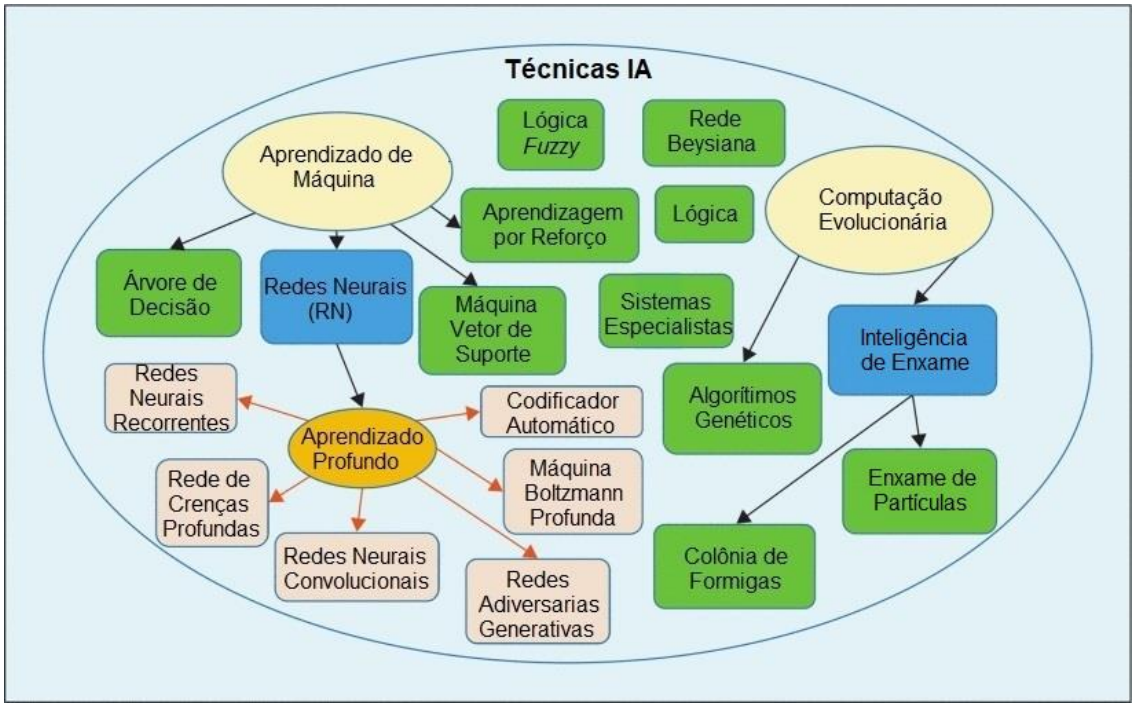


Fonte: Autoria própria.

Na fase de aquisição de dados, os sensores montados em máquinas são a principal fonte de coleta de dados. Em máquinas rotativas, normalmente, são empregados diferentes sensores, como vibração, emissão acústica, temperatura e transformador de corrente. Na segunda fase, a extração de recursos visa extrair características representativas dos sinais coletados com base em técnicas de processamento de sinais, como análise estatística no domínio do tempo, análise espectral de Fourier e Transformada Wavelet. Além disso, para eliminar informações inúteis ou insensíveis, a seleção de características pode ser usada para selecionar características sensíveis por meio de estratégias de redução de dimensão, como Análise de Componentes Principais (ACP), técnica de avaliação de distância e análise discriminante de recursos. Por fim, na fase de classificação de falhas, as características selecionadas são usadas para treinar técnicas de inteligência artificial.

A Figura 5 mostra algumas das técnicas IA mais comumente aplicadas para DDF.

Figura 5 - Exemplo de técnicas IA aplicadas na DDF



Fonte: Adaptado de Khanafer e Shirmohammadi (2020).

Dentre as técnicas IA constantes na Figura 5, a rede neural convolucional (CNN), que faz parte da categoria de aprendizado profundo, foi a técnica utilizada neste estudo. A CNN será abordada na seção 2.3.

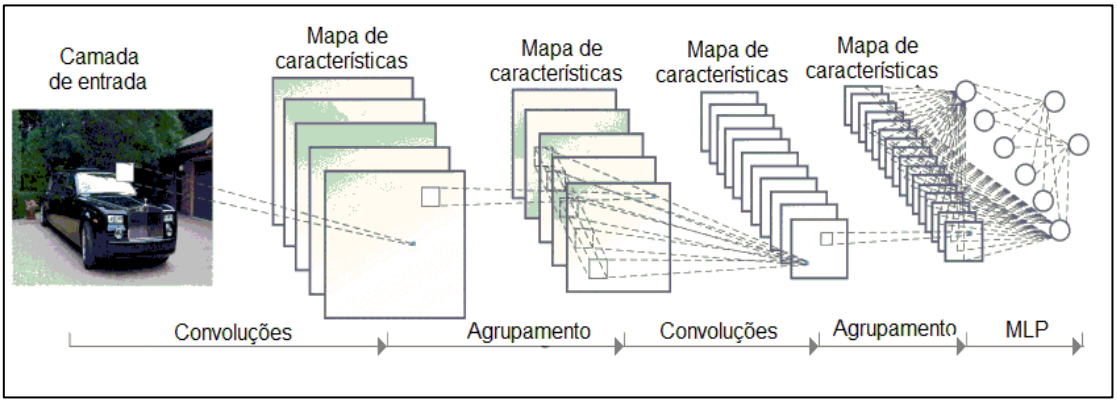
2.3. Redes Neurais Convolucionais (CNN)

Conforme Karn (2016), as redes neurais convolucionais (ConvNets ou CNNs) são uma categoria de redes neurais que apresentam alto desempenho em áreas como reconhecimento e classificação de imagens. A ideia das CNN não é nova, ela foi utilizada por Lecun *et al.* (1998) para reconhecimento de dígitos escritos à mão. Porém, segundo Srinivas *et al.* (2016), devido a restrições computacionais como memória, hardware e a indisponibilidade de grandes quantidades de dados de treinamento, essas redes não eram adequadas para serem utilizadas para imagens muito maiores e, lentamente, caíram em desuso. Com o aumento do poder computacional e a introdução de conjuntos de dados em larga escala, como o ImageNet (Russakovsky *et al.*, 2015) e o conjunto de dados MIT Places (Zhou *et al.*, 2014) foi possível treinar modelos maiores e mais complexos, tornando, conforme Neupane e Seok (2020), as CNNs o modelo mais representativo de aprendizado profundo.

Como qualquer modelo típico de rede neural, as CNNs são baseadas em neurônios que são organizados em camadas que, conforme Kattenborn *et al.* (2021), são projetadas para aprender as características espaciais, por exemplo, arestas, cantos, texturas ou formas mais abstratas, que melhor descrevem a classe ou objeto de destino. O núcleo da rede, para aprender essas características, realizam múltiplas e sucessivas transformações dos dados de entrada (convoluções) em diferentes escalas espaciais.

A arquitetura de uma CNN é construída como uma série de camadas/estágios, onde cada fase tem uma função diferente e cada função é concluída automaticamente dentro do algoritmo. Conforme Tayyab *et al.* (2022), em geral a estrutura de uma CNN é composta por uma camada de entrada, camadas ocultas e uma camada de saída. As camadas ocultas geralmente são camadas convolucionais, camadas de Unidade Linear Retificada (ReLU), camadas de agrupamento ou sub amostragem e camadas totalmente conectadas. A Figura 6 mostra o exemplo de uma arquitetura CNN 2D típica para dados de imagem.

Figura 6 - Diagrama de blocos generalizado de uma CNN



Fonte: Adaptado de Liu *et al.* (2018).

O aprendizado de características ativado por aprendizado profundo tem a vantagem de não exigir uma sequência de construção, pesquisa e seleção de características, pois isso é feito automaticamente dentro da arquitetura da rede, não exigindo conhecimento especializado. Segundo Wang *et al*. (2016), a CNN como um método de aprendizado profundo, pode usar os dados originais como entrada para identificar as características incorporadas e executar o reconhecimento de padrões dentro da estrutura da rede, sem a necessidade de extração explícitas ou artesanais de características e, portanto, evitando a interferência de fatores humanos. Os principais elementos e camadas comumente utilizadas em arquiteturas de CNNs serão descritas de forma sucinta na sequência.

2.3.1 Filtros e passo

O filtro é uma matriz de números inteiros que são usados em um subconjunto dos valores dos dados de entrada, do mesmo tamanho do filtro. Por exemplo, cada pixel na imagem de entrada é multiplicado pelo valor correspondente no filtro e, em seguida, o resultado é somado, representando um único valor, como um pixel, na camada de saída/mapa de características. O filtro é deslocado por n número de pixels, também conhecido como passo. Este processo, também conhecido como convolução, continua até o final da matriz de entrada e a saída de cada produto escalar completa a matriz de saída ou mapa de características.

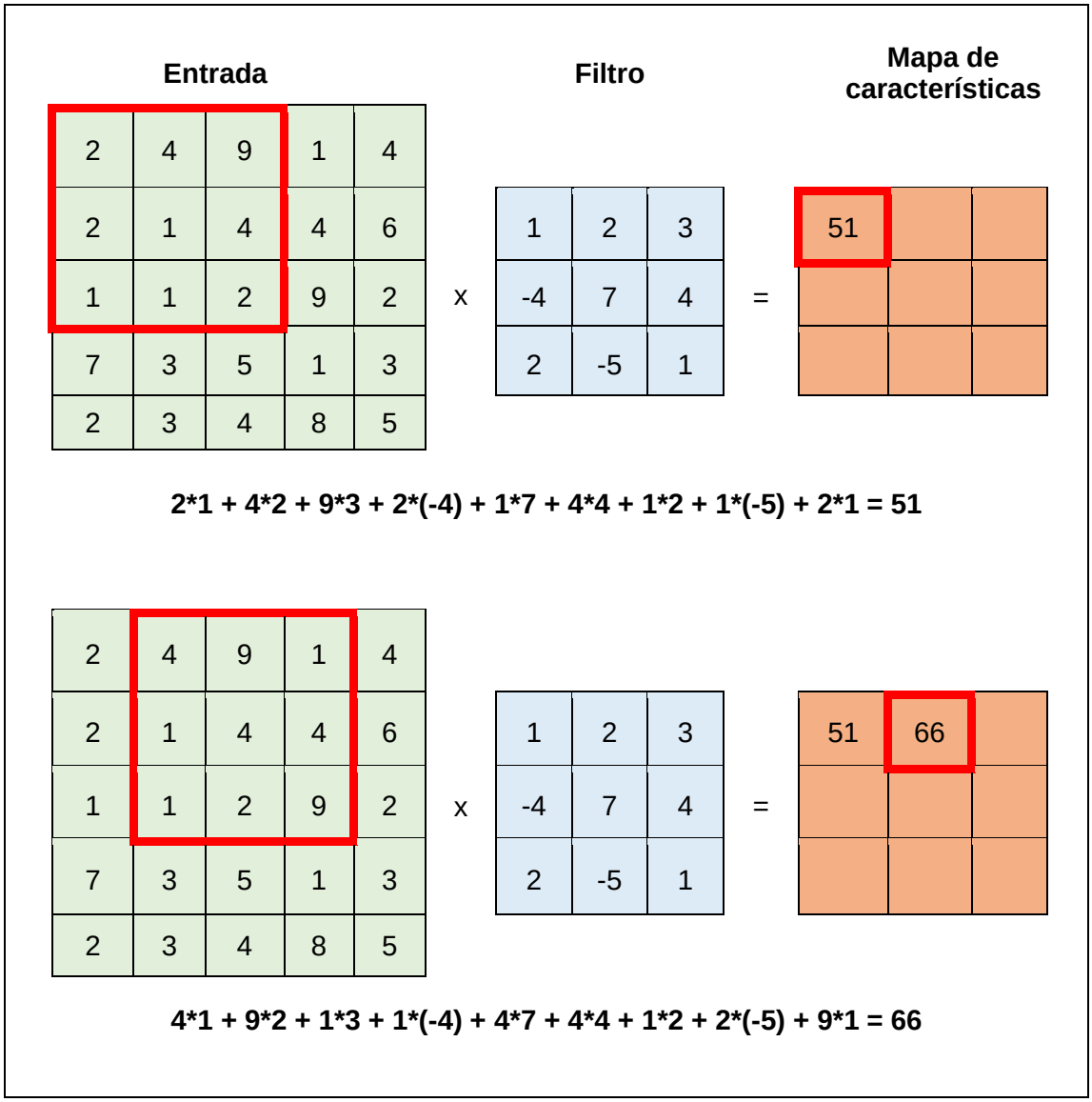
Vale ressaltar que, conforme o tamanho do passo é aumentado, o tamanho da saída é reduzido. Isso pode ser visto na equação 2.1.

$$S = 1 + \frac{E-F}{P} \quad (2.1)$$

sendo S é o tamanho da saída, E o tamanho da entrada, F o tamanho do filtro e P o tamanho do passo.

A Figura 7 mostra o exemplo de um filtro 3x3 de passo 1 varrendo uma entrada 5x5.

Figura 7 – Duas primeiras etapas de uma convolução em uma entrada 5x5 e filtro 3x3



Fonte: Autoria própria.

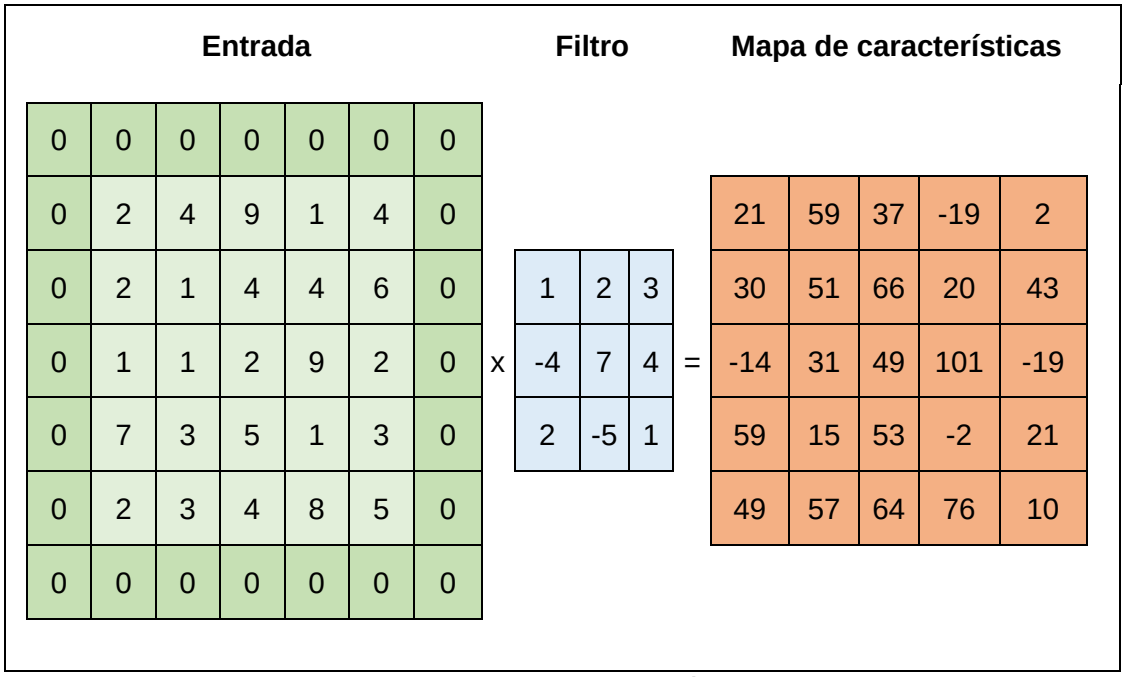
O resultado da operação de convolução da Figura 7, conforme equação 2.1, é um mapa de características 3x3.

2.3.2 Preenchimento

Preenchimento são valores anexados às bordas de uma entrada para aumentar seu tamanho. Segundo Ajit *et al*. (2020), não fazer isso poderá resultar em perda de informações da borda da imagem e redução de dimensão, levando a um baixo desempenho. Conforme Albawi *et al.* (2017), essa perda de informações que possa existir na borda da imagem é uma das desvantagens da etapa de convolução, já que essas características só são capturadas quando o filtro desliza, podendo não serem vistas. Um método muito simples, mas eficiente, para resolver o problema é usar o preenchimento com zeros. O outro benefício do preenchimento com zeros é gerenciar o tamanho da saída.

A Figura 8 mostra o resultado do exemplo de uma convolução de um filtro 3x3, passo 1, varrendo uma entrada 5x5 com preenchimento de tamanho 1. O mapa de características resultante, conforme equação 2.2, terá tamanho 5x5.

Figura 8 - Etapas de uma convolução em uma entrada 5x5, passo 1, preenchimento 1 e filtro 3x3



Fonte: Autoria própria.

A equação 2.1 modificada para incluir o preenchimento com zeros é a equação 2.2.

$$S = 1 + \frac{E+2Z-F}{P} \tag{2.2}$$

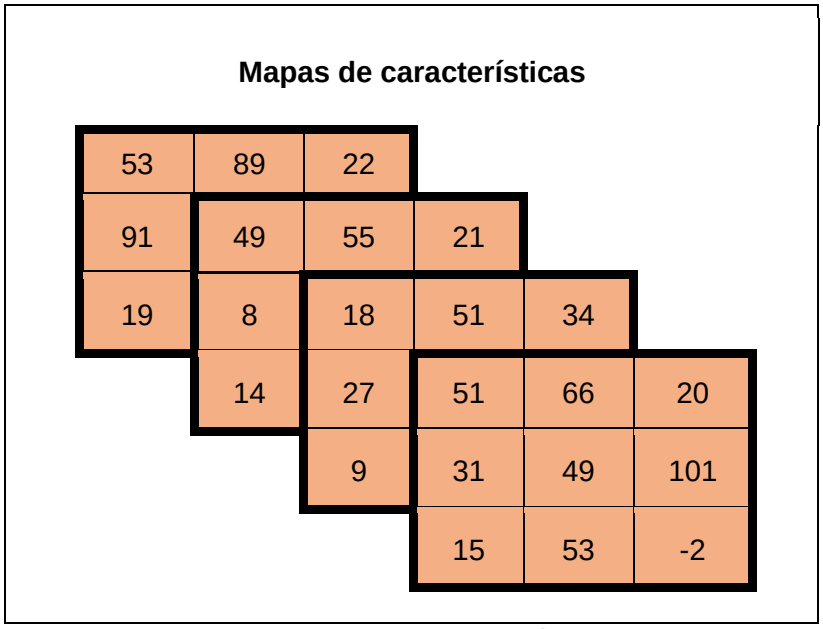
em que Z é o tamanho do preenchimento com zeros na entrada.

2.3.3 Mapas de características

À medida que um filtro se move ao longo da entrada, ele usa o mesmo conjunto de pesos e o mesmo bias para a convolução, formando um mapa de características. Cada mapa de características é o resultado de uma convolução usando um conjunto diferente de pesos e um bias diferente. Portanto, o número de mapas de características é igual ao número de filtros.

A Figura 9 demonstra um exemplo de mapas de características resultantes de uma convolução de uma entrada 5x5, sem preenchimento, passo 1, com 4 filtros 3x3. Esse exemplo pode ser entendido com a convolução resultante da Figura 7, com um total de 4 filtros.

Figura 9 - Mapas de características resultante de uma convolução com 4 filtros 3x3.



Fonte: Autoria própria.

O número total de neurônios (N) em um mapa de características pode ser calculado utilizando a equação 2.3.

$$N = y * x * NF \tag{2.3}$$

em que y e x correspondem, respectivamente, a altura e largura da saída e NF o número de filtros da camada.

Para o exemplo da Figura 7, o número total de neurônios é igual a $N = 3 * 3 * 4 = 36$ neurônios.

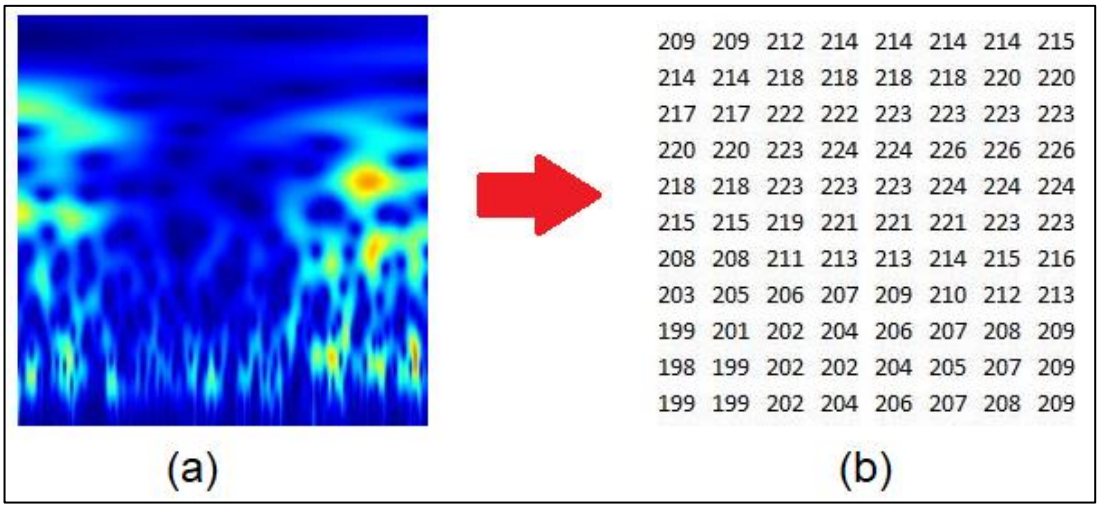
A principal vantagem do mapa de características é que ele armazena todas as características distintivas de uma determinada imagem e, ao mesmo tempo, reduz a quantidade de dados a serem processados.

2.3.4 Camada de entrada

A camada de entrada contém os dados ou imagens a serem processados. Conforme Ajit *et al.* (2020), quando a entrada é uma imagem colorida, a camada é representada em termos de 3 dimensões, ou seja, largura, comprimento e altura, onde comumente é denotado como largura, altura e profundidade, que são pixels para uma imagem na forma de uma matriz. Por exemplo, se a entrada for (227x227x3), então largura: 227px, altura: 227px e profundidade: 3px. A profundidade 3 é usada principalmente para representar imagens coloridas na forma de RGB.

A Figura 10 mostra um exemplo de uma imagem 227x227 (falha na pista interna de um rolamento) e parte de sua matriz de pixels.

Figura 10 - (a) imagem de entrada e (b) parte da matriz de pixel de (a)



Fonte: Autoria própria.

2.3.5 Camada Convolucional

A Camada de convolução é a camada mais básica, mas ao mesmo tempo a mais importante da CNN, pois é onde ocorre a maior parte da computação para extrair informações úteis dos dados de entrada. Os neurônios da camada convolucional se conectam a sub-regiões das imagens de entrada ou saídas da

camada anterior e aprende as características localizadas por essas regiões durante a varredura de uma imagem.

A convolução nessa camada ocorre conforme descrito na subseção 2.3.1, onde a camada envolve a entrada movendo os filtros ao longo da entrada vertical e horizontalmente, calculando o produto escalar dos pesos do filtro e da entrada, adicionando um termo de bias e, finalmente, uma função de ativação é aplicada.

A operação na camada de convolução é definida como na equação 2.4.

$$x_i^j = f\left(\sum_{k=1}^{n} W_{i,k}^j \times x_k^{j-1} + b_i^j\right) \tag{2.4}$$

em que x_i^j denota o i-ésimo mapa de recursos de saída j-ésimo nível, x_k^{j-1} corresponde ao k-ésimo mapa de recursos de entrada do (j – 1)-ésimo nível, $W_{i,k}^j$ é o filtro de convolução entre o i-ésimo mapa de recursos de saída na j-ésima camada e o k-ésimo mapa de recursos de entrada na (j – 1)-ésima camada, n é o número de mapas de recursos de entrada, b_i^j é o bias do i-ésimo mapa de recursos de saída na j-ésima camada e f é a função de ativação. A função de ativação mais comumente utilizada nos problemas de DDF é a ReLU, que será abordada na subseção 2.3.7.

Segundo Verstraete *et al.* (2017), um aspecto importante das camadas convolucionais para análise de imagens é que as unidades dentro do mesmo mapa de características compartilham o mesmo banco de filtros. No entanto, para lidar com a possibilidade de que a localização de um mapa de características não seja a mesma para todas as imagens, diferentes mapas de características usam diferentes bancos de filtros.

2.3.6 Camada de normalização em lote

Introduzida por Ioffe e Szegedy (2015), as camadas de normalização em lote são usadas para normalizar as ativações de um determinado volume de entrada, antes de passá-lo para a próxima camada na rede (ROSEBROCK, 2021). Segundo Kattenborn *et al*. (2021), a normalização em lote normaliza a saída das funções de ativação para média zero e variância unitária e, assim, evita que a rede se torne desequilibrada devido a ativações excessivamente altas ou baixas. Isso suaviza o problema de otimização da função de descida de gradiente e permite faixas maiores de taxas de aprendizado e, portanto, facilita

a convergência da rede. Além disso, ajuda a reduzir o *overfitting* por meio da regularização. Para mais detalhes sobre uso dessa camada, pode-se consultar Ioffe e Szegedy (2015).

2.3.7 Camada ReLU

Conforme Albawi *et al.* (2017), no passado, funções não lineares como tanh e sigmóide foram as mais populares. No entanto, recentemente, a Unidade Linear Retificada (ReLU) tem sido usada com mais frequência pois a rede é capaz de ser treinada muito mais rápido (devido à eficiência computacional) sem fazer uma diferença significativa na precisão. A ReLU também ajuda a aliviar o problema do gradiente de fuga, que é o problema em que as camadas inferiores da rede treinam muito lentamente porque o gradiente diminui exponencialmente pelas camadas.

A camada ReLU é usada para aplicar a função de ativação a todos os valores no volume de entrada (pixel a pixel no caso de imagens) que, conforme equação 2.5, converte todos os valores negativos em zero.

$$ReLU(x) = \max(0, x) \tag{2.5}$$

Segundo Ajit *et al.* (2020) a camada ReLU aumenta as propriedades não lineares do modelo e da rede em geral, sem afetar os campos receptivos da camada ou os hiperparâmetros.

2.3.8 Camada de agrupamento (*Pooling*)

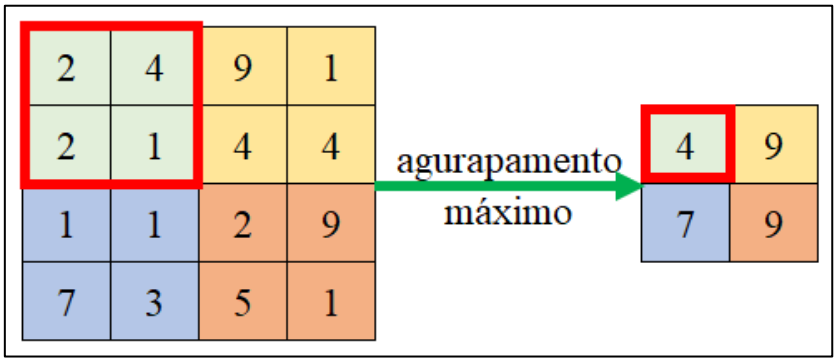
O agrupamento espacial (também chamado de sub amostragem ou redução da resolução), definido por um tamanho de filtro, passo e uma operação de redução, reduz a dimensionalidade de cada mapa de características, retendo as informações mais importantes e reduzindo o número de parâmetros do modelo, portanto, a carga computacional, a chance de *overfitting*, acelerando os cálculos e melhorando a eficiência da rede. Segundo abordado por Ajit *et al.* (2020), o agrupamento permite que a CNN incorpore todas as diferentes dimensões de uma imagem para que ela reconheça com sucesso o objeto dado, mesmo que sua forma esteja distorcida ou presente em ângulos diferentes.

Existem vários tipos de agrupamento, como o agrupamento máximo, o agrupamento médio e o agrupamento global. Porém, a operação de

agrupamento mais comum é o agrupamento máximo, que particiona a imagem em sub-regiões e retorna apenas o valor máximo do interior dessa sub-região, onde a ideia é que as ativações fortes (por exemplo, características de borda ou linha) sejam conservadas dentro da rede.

A configuração mais comumente aplicada no agrupamento máximo, conforme pode ser visto na Figura 11, é tamanho de filtro 2x2 e um passo de 2.

Figura 11 - Exemplo de operação de agrupamento máximo



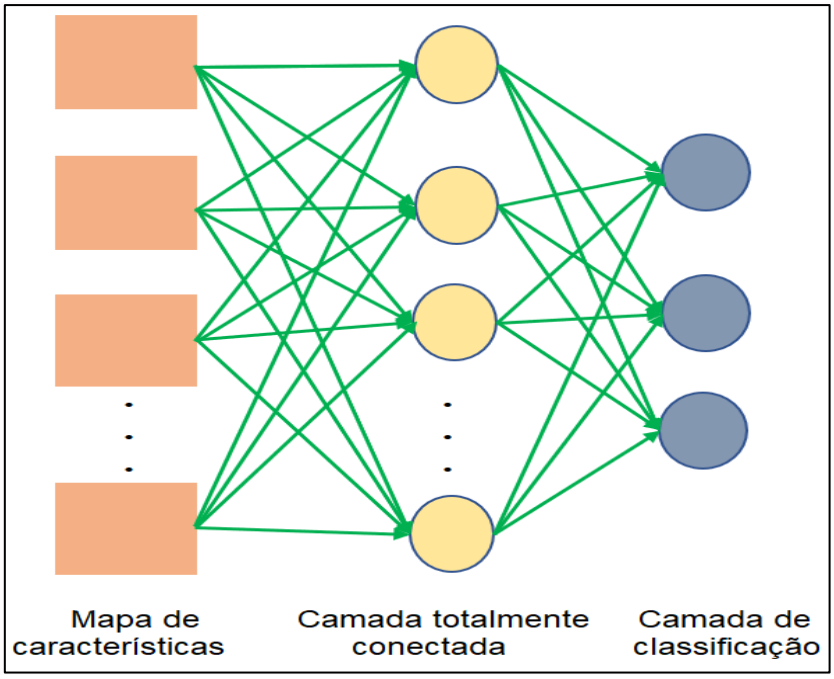
Fonte: Autoria própria.

A operação de agrupamento representada na Figura 11, reduz o tamanho do mapa de características de entrada por um fator de 4, enquanto as células de saída contêm o valor máximo das 4 células de entrada.

2.3.9 Camada Totalmente Conectada

A camada totalmente conectada é semelhante à maneira como os neurônios são organizados em uma rede neural tradicional, um *Perceptron* Multi *Layer* (MLP) tradicional. Portanto, cada nó em uma camada totalmente conectada está diretamente conectado a todos os nós da camada anterior (convolucional ou de agrupamento) e da próxima. Segundo The MathWorks (2023), essa camada combina todas as características (informações locais) aprendidos pelas camadas anteriores para identificar os padrões. Para problemas de classificação, a última camada totalmente conectada combina as características, usando uma função de ativação (softmax, por exemplo) na camada de saída para classificar os dados de entrada, razão pela qual o tamanho da última camada totalmente conectada da rede é igual ao número de classes (saídas) do conjunto de dados. Na Figura 12 e Figura 13(a) são ilustrados exemplos de camadas totalmente conectadas.

Figura 12 - Exemplo de camada totalmente conectada



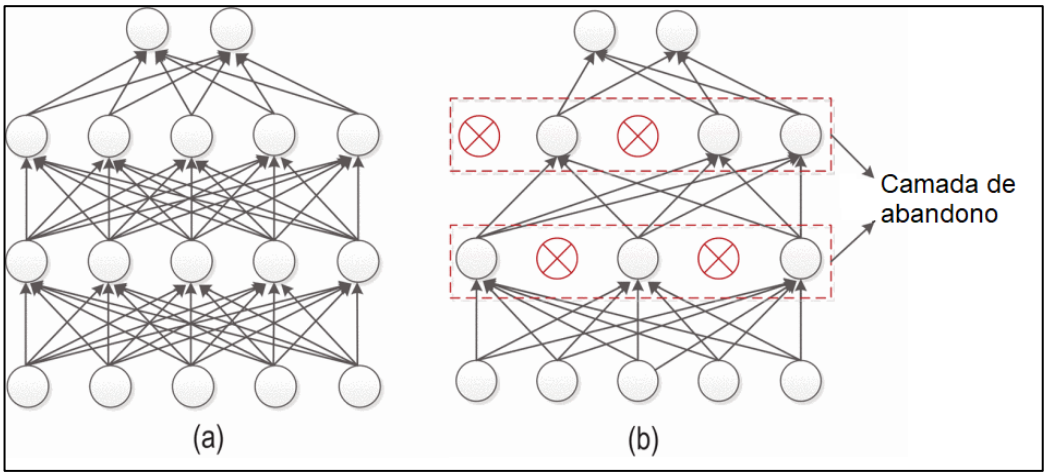
Fonte: Autoria própria.

A principal desvantagem de uma camada totalmente conectada é que ela inclui muitos parâmetros que precisam de cálculos complexos em exemplos de treinamento. Portanto, tentamos eliminar o número de nós e conexões, usando por exemplo, a técnica de abandono.

2.3.10 Camada de abandono (*Dropout*)

Dropout é um método de regularização que zera estocasticamente as ativações das unidades ocultas em cada etapa de treinamento. Para cada conjunto de treinamento, as camadas de abandono, com probabilidade p, descartam aleatoriamente entradas da camada anterior para a próxima camada na arquitetura de rede. Com isso, a rede deve ser capaz de fornecer a classificação ou saída correta para um exemplo específico, mesmo que algumas das ativações sejam descartadas, reduzindo o *overfitting* e, assim, tornando a rede mais generalista. A Figura 13 mostra um exemplo de arquitetura de rede com e sem camadas de abandono.

Figura 13 – Comparação: (a) rede neural convolucional convencional e (b) rede neural convolucional com camada de abandono



Fonte: Adaptado de Wnag (2016).

A Figura 13(a) é uma rede neural convolucional convencional na qual as camadas estão totalmente conectadas, já a Figura 13(b) mostra uma rede com camadas de abandono, onde alguns neurônios foram “banidos” com certa probabilidade, o que significa que a estrutura efetiva da rede muda a cada propagação direta.

Uma observação importante é que essa camada é usada apenas durante o treinamento e não durante o tempo de teste. Para mais detalhes do uso dessa camada, pode-se consultar Srivastava *et al*. (2014).

2.3.11 Camada *Softmax*

Uma função de ativação aplicada à tarefa de classificação multiclasse é uma função *softmax* (outros classificadores como Máquina de Vetor de Suporte também podem ser usados) que normaliza os valores reais de saída da última camada totalmente conectada para probabilidades de classe de destino, onde cada valor varia entre 0 e 1 e todos os valores somam 1. A forma matemática da função *softmax* pode ser definida conforme equação 2.6.

$$S(\vec{Z})_i = \frac{e^{zi}}{\sum_{j=1}^{k} e^{zj}} \tag{2.6}$$

em que Z_i são os elementos do vetor de entrada e K é o número total de elementos no vetor de entrada. O termo no denominador em 2.6 atua como o

termo de normalização e garante que a soma dos valores de saída da função *softmax* seja igual a 1.

Conforme Hussain e Tsai (2021), a função *softmax* é útil tanto no estágio de treinamento quanto no estágio de inferência. No estágio de treinamento, as saídas das funções *softmax* são usadas para calcular a perda do modelo para um lote de imagens. No estágio de inferência, a função *softmax* atua como a camada de saída final, onde o valor mais alto da função é usado como resultado final para classificação do modelo

2.4. Detecção e diagnóstico de falhas em rolamentos

Conforme Neupane e Seok (2020), os rolamentos são os componentes centrais e vulneráveis das máquinas rotativas, cuja condição de saúde afeta diretamente no desempenho, eficiência, estabilidade e vida útil das máquinas. Para Ahmed e Nandi (2018), as falhas em rolamentos estão entre as principais causas de quebra de máquinas rotativas e, para evitar essas avarias, o monitoramento da integridade dos rolamentos é tarefa vital nos processos industriais.

Para o diagnóstico de falhas em rolamentos, um dos conhecimentos de domínio mais importantes é a frequência de características de falha (FCF). De acordo com o mecanismo de falha, quando ocorre uma falha no rolamento, os componentes da frequência características da falha aparecerão nos sinais de vibração. As FCFs para a condição de falha na pista interna, pista externa ou falha na esfera são calculados da seguinte forma:

$$FPI = f_r * \frac{z\left(1-\frac{d}{D}\cos\,\alpha\right)}{2} \tag{2.7}$$

$$FPE = f_r * \frac{z\left(1+\frac{d}{D}\cos\,\alpha\right)}{2} \tag{2.8}$$

$$FE = f_r * \frac{D\left(1-(\frac{d}{D}\cos\alpha)^2\right)}{2d} \tag{2.9}$$

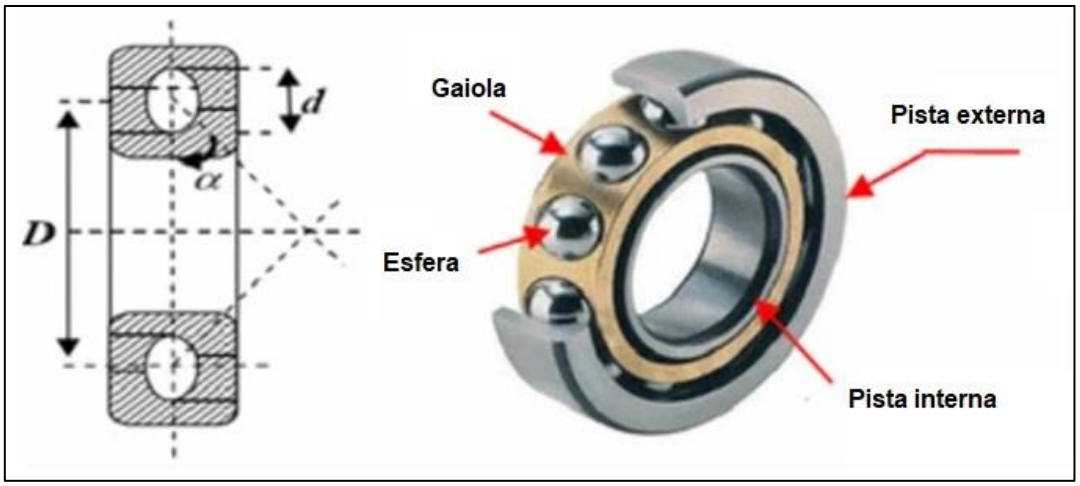
sendo:

- FPI: Frequência característica de falha na pista interna;
- FPE: Frequência característica de falha na pista externa;
- FE: Frequência característica de falha na esfera;
- f_r: Frequência de rotação;

- z: o número de elementos rolantes;
- d: diâmetro da esfera;
- D: diâmetro primitivo;
- α: ângulo de contato do rolamento.

A Figura 14 ilustra os componentes dos rolamentos para o cálculo das FCF e a posição em que as falhas podem ser encontradas.

Figura 14 - Partes estruturais de um rolamento



Fonte: Adaptado de Boudiaf *et al.* (2016).

Em muitos métodos de diagnóstico de falhas de rolamentos existentes, a ideia principal é encontrar FCFs nos sinais de vibração, sendo o monitoramento de vibração a técnica de monitoramento mais amplamente utilizada e econômica para detectar, localizar e distinguir falhas em rolamentos. Conforme Sreejith, Verma e Srividya (2008), as técnicas de análise aplicadas para o processamento dos sinais de vibração para monitoramento da condição dos rolamentos podem ser classificadas em: domínio do tempo, domínio da frequência e métodos de análise de tempo-frequência ou escala de tempo. O método mais popular é a análise no domínio da frequência, que precisa da ajuda de um especialista para interpretar os resultados.

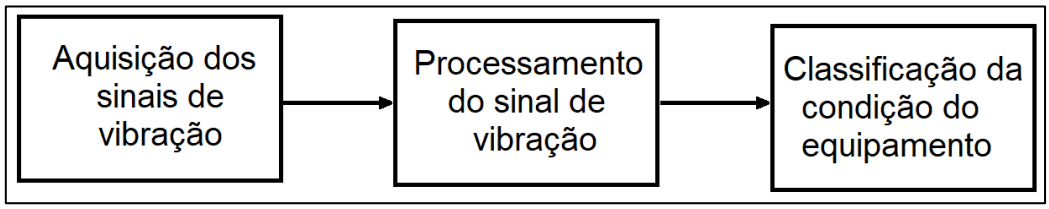
A confiabilidade do monitoramento de condições pode ser aumentada automatizando o processo, o que também proporciona economia de tempo e custo. Além disso, o diagnóstico automático de falhas não depende do julgamento subjetivo humano.

3 Metodologia

A análise de vibração é um método já estabelecido para detecção e diagnóstico de falhas em rolamentos, a qual requer a extração e seleção de características representativas da condição do rolamento. Conforme já discutido nas seções anteriores, a extração e seleção de recursos tradicionais é um processo de trabalho intensivo que requer conhecimento especializado de informações relevantes ao sistema, sendo esse conhecimento um exercício orientado para especialistas. Para resolver esse problema, em vez de extrair recursos empiricamente na abordagem convencional, uma arquitetura CNN foi aplicada para aprender automaticamente as características da saúde do rolamento.

Uma CNN é geralmente projetada para aproveitar a estrutura 2D da entrada. No entanto, o sinal original de falha do rolamento é um sinal 1D no domínio do tempo, a partir do qual é difícil observar qualquer padrão em relação ao defeito do rolamento. Para resolver esse problema, este estudo aplica uma técnica de transformação apropriada para atingir os estados de integridade do rolamento em uma visualização 2D. A metodologia utilizada compreende, basicamente, três etapas principais. Em primeiro lugar, o sinal de vibração bruto é dividido em vários segmentos. Na segunda etapa, a transformada wavelet continua é usada para transformar cada segmento do sinal de vibração unidimensional para um sinal bidimensional, denominado escalogramas, salvos na forma de imagens, as quais contém as informações das condições de saúde do rolamento. Na última etapa, o conjunto de imagens (escalogramas de cada segmento do arquivo original) é fornecido à rede neural convolucional para aprender automaticamente as características representativas dessas imagens e realizar a tarefa detecção e diagnóstico de falhas no rolamento. A estrutura geral da DDF usando a metodologia descrita anteriormente é apresentada na Figura 15.

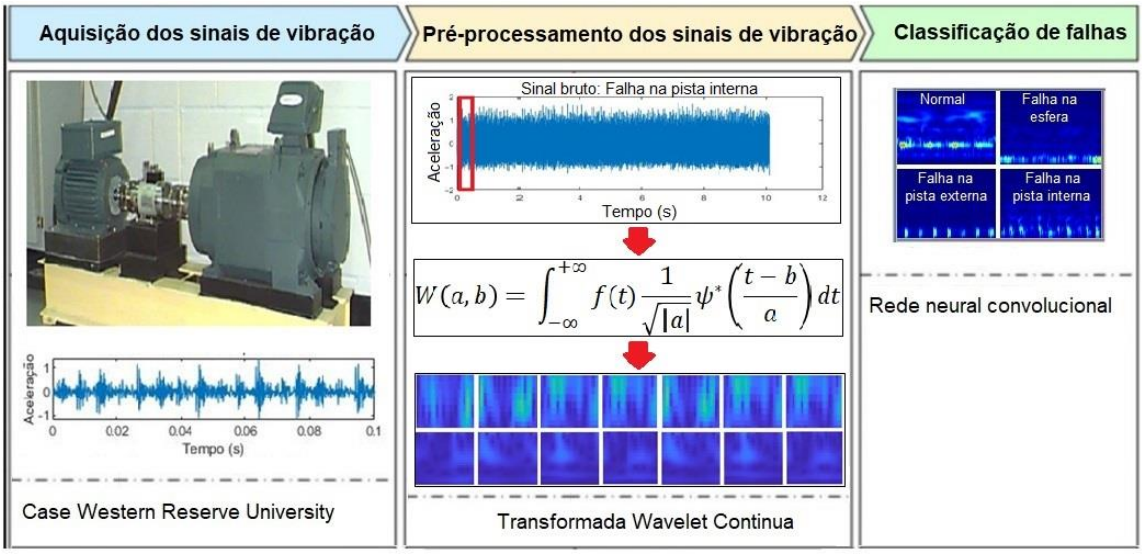
Figura 15 - Estrutura geral DDF baseada em vibração



Fonte: Autoria própria.

A Figura 16, a seguir, ilustra as fases utilizadas para desenvolvimento da abordagem CNN proposta para a tarefa de DDF no rolamento selecionado.

Figura 16 - Fases DDF para desenvolvimento CNN proposta



Fonte: Autoria própria.

As etapas presentes na Figura 16 são detalhadas nas subseções 3.1, 3.2 e 3.3.

3.1. Aquisição dos sinais de vibração

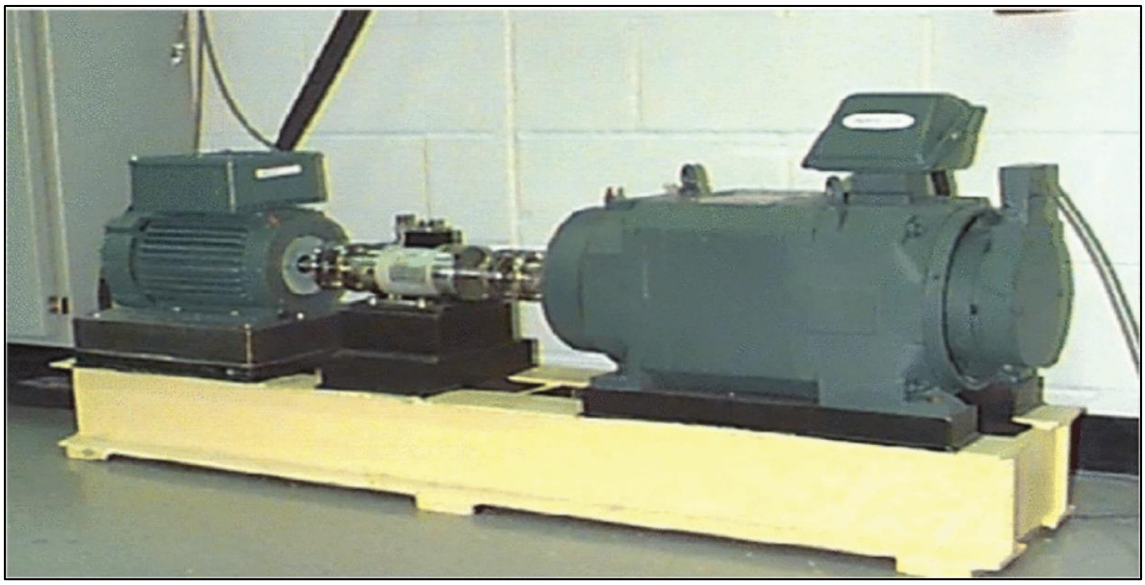
Para o desenvolvimento do algoritmo de detecção e diagnostico de falhas utilizando a CNN, foram utilizados os dados de falhas em rolamentos fornecidos pela Case Western Reserve University (CWRU). Esses dados estão disponíveis gratuitamente no site da instituição e são comumente usados no campo de diagnóstico de falhas em rolamentos.

Segundo Sikder *et al.* (2021), os dados fornecidos pelo CWRU Bearing Data Center tornaram-se uma referência para estudos de falhas em rolamentos de motores. Li *et al.* (2013), Ahmed e Nandi (2018), Guo *et al.* (2019) e Atta *et*

al. (2021) estão entre os diversos pesquisadores em todo o mundo que usaram os dados de rolamentos coletados desse enorme banco de dados para testar e validar seus métodos de diagnóstico de falhas. O repositório se tornou tão popular que, segundo Sikder *et al.* (2021), mais de 40 artigos foram publicados em um único periódico envolvendo seus conjuntos de dados.

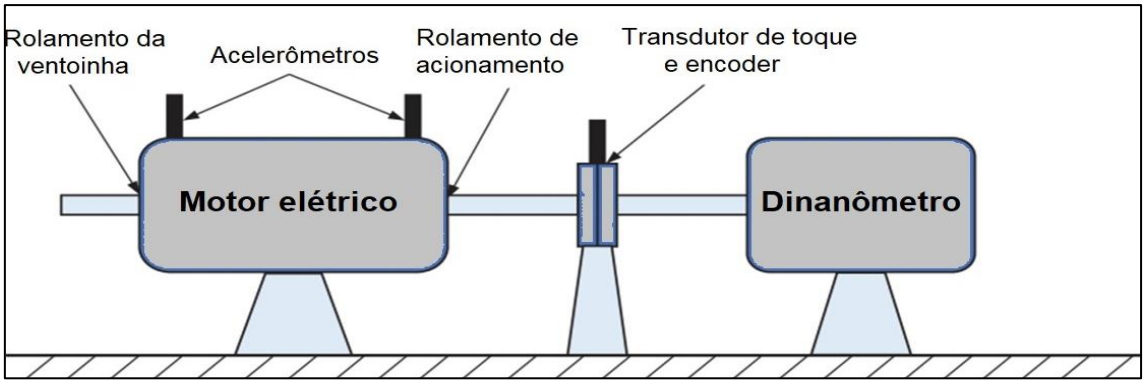
A estrutura da bancada de ensaio em rolamentos da CWRU é mostrada Figura 17 e esquematizada na Figura 18.

Figura 17 - Bancada de testes em rolamentos da CWRU



Fonte: Case School Of Engineering.

Figura 18 - Esquemático da bancada de teste em rolamentos da CWRU



Fonte: Adaptado de Li, Liu e Xiao (2021).

A bancada de teste é composta por um motor de indução de 2 hp conectado a um dinamômetro por meio de um eixo. Os rolamentos do motor foram danificados artificialmente, usando usinagem de eletro-descarga, para representar falhas dos elementos rolantes (esferas), pista interna e pista externa

com diâmetros das falhas de 0.007, 0.014, 0.021 e 0.028 pol. Além dos ensaios com os rolamentos defeituosos, também foram realizados ensaios com um rolamento em condições normais de operação (sem falhas inseridas).

Os sinais de vibração foram coletados usando um acelerômetro, instalado na vertical (12 horas), tanto na extremidade do acionamento quanto na extremidade do ventilador da carcaça do motor (ventoinha), sob condições de carga (0, 1, 2, e 3 hp) e velocidade (1797–1720 rpm) variáveis no motor. Durante alguns experimentos, um acelerômetro também foi conectado à base de suporte do motor.

A Tabela 1 mostra o exemplo do banco de dados com falhas no rolamento de acionamento, na taxa de amostragem de 12kHz.

Tabela 1 – Banco de dados de falha do rolamento de acionamento em 12kHz

Diâmetro da Falha	Carga no motor (hp)	Vel. aprox. do motor (rpm)	Pista interna	Esfera	Posição da pista externa em relação à zona de carga (em horas)		
					Centrado @ 6:00	Ortogonal @ 3:00	Em frente @ 12:00
0,007"	0	1797	IR007_0	B007_0	OU007@6_0	OU007@3_0	OU007@12_0
	1	1772	IR007_1	B007_1	OU007@6_1	OU007@3_1	OU007@12_1
	2	1750	IR007_2	B007_2	OU007@6_2	OU007@3_2	OU007@12_2
	3	1730	IR007_3	B007_3	OU007@6_3	OU007@3_3	OU007@12_3
0,014"	0	1797	IR014_0	B014_0	OU014@6_0	*	*
	1	1772	IR014_1	B014_1	OU014@6_1	*	*
	2	1750	IR014_2	B014_2	OU014@6_2	*	*
	3	1730	IR014_3	B014_3	OU014@6_3	*	*
0,021"	0	1797	IR021_0	B021_0	OU021@6_0	OU021@3_0	OU021@12_0
	1	1772	IR021_1	B021_1	OU021@6_1	OU021@3_1	OU021@12_1
	2	1750	IR021_2	B021_2	OU021@6_2	OU021@3_2	OU021@12_2
	3	1730	IR021_3	B021_3	OU021@6_3	OU021@3_3	OU021@12_3
0,028"	0	1797	IR028_0	B028_0	*	*	*
	1	1772	IR028_1	B028_1	*	*	*
	2	1750	IR028_2	B028_2	*	*	*
	3	1730	IR028_3	B028_3	*	*	*

Fonte: Case School Of Engineering

O banco de dados para o rolamento na condição normal, sem falhas, é demonstrado na Tabela 2.

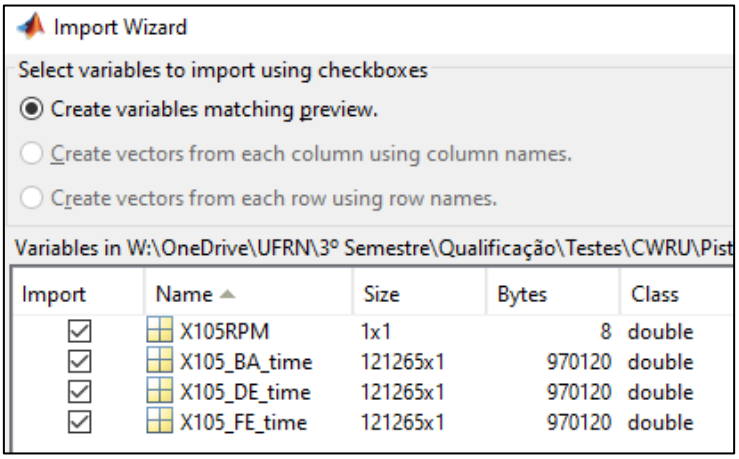
Tabela 2 - Banco de dados do rolamento sem falhas

Carga do motor (hp)	Velocidade aprox. do motor (rpm)	Dados normais
0	1797	Normal_0
1	1772	Normal_1
2	1750	Normal_2
3	1730	Normal_3

Fonte: Case School Of Engineering

Os sinais de vibração foram gravados usando um gravador DAT de 16 canais e pós-processados em um ambiente Matlab. Para o rolamento de acionamento, os dados digitais foram coletados em 12.000 e 48.000 amostras por segundo. Já os dados de velocidade e potência foram coletados usando um transdutor/codificador de torque e registrados manualmente. A Figura 19 mostra um exemplo de um arquivo de dados, em formato Matlab (*.mat), contendo as informações do arquivo de falha na pista interna, em 12kHz, diâmetro de falha 0,007 e sem carga para o rolamento de acionamento (arquivo IR007_0 na Tabela 1) .

Figura 19 – Tela de importação Matlab: dados de falha na pista interna, 12kHz, diâmetro de falha 0,007 e sem carga



Fonte: Autoria própria.

As informações constantes em cada arquivo de dados são:

- RPM – velocidade do eixo (RPM) durante o teste;

- BA - leitura do acelerômetro na base do motor;
- DE - leitura do acelerômetro no rolamento de acionamento;
- FE - leitura do acelerômetro no rolamento da ventoinha.

Para aplicação neste estudo, foram utilizados os dados de vibração do rolamento da extremidade de acionamento, o qual era um rolamento rígido de esferas do tipo SKF 6205-2RSJEM, na frequência de amostragem de 12 kHz. As dimensões desse rolamento são demonstradas na Tabela 3.

Tabela 3 - Dimensões do rolamento (polegadas)

Diâmetro interno	Diâmetro externo	Espessura	Diâmetro da esfera	Diâmetro primitivo
0.9843	2.0472	0.5906	0.3126	1.537

Fonte: Case School Of Engineering.

As frequências característica de falhas (FCF) do rolamento selecionado são demonstrados na Tabela 4.

Tabela 4 - Frequências características de falhas (múltiplo da velocidade de operação em Hz)

Pista interna	Pista externa	Esfera
5.4152	3.5848	4.7135

Fonte: Case School Of Engineering.

Os arquivos dos ensaios de vibração e configurações de teste do rolamento selecionado para estudo foram mostrados na Tabela 1. Vale ressaltar que, conforme pode ser observado na Tabela 1, o conjunto de dados possui lacunas nas configurações do ensaio de vibração com diâmetro de falha de 0,028”, como também na posição da pista externa em relação à zona de carga em @3:00 e @12:00. Devido às lacunas presentes nas configurações do ensaio de vibração do rolamento selecionado, esse estudo excluiu as configurações onde existiam lacunas, ficando apenas com as configurações completas. O motivo para utilizar as configurações sem lacunas foi para manter o equilíbrio das amostras e permitir utilizar todas as configurações possíveis dentro de um banco de dados completo.

A Tabela 5 mostra o conjunto de dados de falhas, sem lacunas, selecionado para estudo.

Tabela 5 - Banco de dados de falhas utilizado no estudo

Diâmetro da Falha	Carga no motor (hp)	Velocidade aprox. do motor (rpm)	Pista interna	Esfera	Pista externa
					Zona de carga centrada às 6:00 horas
0,007"	0	1797	IR007_0	B007_0	OU007@6_0
	1	1772	IR007_1	B007_1	OU007@6_1
	2	1750	IR007_2	B007_2	OU007@6_2
	3	1730	IR007_3	B007_3	OU007@6_3
0,014"	0	1797	IR014_0	B014_0	OU014@6_0
	1	1772	IR014_1	B014_1	OU014@6_1
	2	1750	IR014_2	B014_2	OU014@6_2
	3	1730	IR014_3	B014_3	OU014@6_3
0,021"	0	1797	IR021_0	B021_0	OU021@6_0
	1	1772	IR021_1	B021_1	OU021@6_1
	2	1750	IR021_2	B021_2	OU021@6_2
	3	1730	IR021_3	B021_3	OU021@6_3

Fonte: Autoria própria

O conjunto de dados resultante, utilizado nesse estudo, soma 40 arquivos de leituras de vibração em diferentes condições de operação do motor e integridade do rolamento (Tabela 2 e Tabela 5).

Para desenvolvimento da rede CNN proposta, os arquivos de treinamento foram selecionados em diferentes configurações da bancada de teste (cargas e velocidades no motor e localização e diâmetros de falhas no rolamento) objetivando mostrar a potência e a eficácia do método proposto, independentemente das condições de operação do motor. Vale ressaltar que, para o rolamento com falhas, visando avaliar o poder de aprendizagem e generalização do método proposto, a quantidade de dados de treinamentos selecionados foi a menor possível. Ou seja, para cada diâmetro de falha foram selecionados apenas arquivos em uma única configuração de carga no motor, variando de 0 hp a 2 hp. Os arquivos de vibração na configuração de ensaio com carga de 3 hp no eixo não foram utilizados para treinamento da rede, pois essa configuração será utilizada para avaliar o poder de generalização da CNN desenvolvida (subseção 4.5.2). A estratégia em relação a configuração de carga

no eixo foi a mesma para selecionar os arquivos de treinamento do rolamento sem falhas (normal).

Os arquivos selecionados para treinamento da CNN representam 30% (12) dos arquivos do conjunto de dados para este estudo (Tabela 2 e Tabela 5) e estão resumidos na Tabela 6.

Tabela 6 - Arquivos de dados utilizados para desenvolvimento do algoritmo CNN

Carga no motor (hp)	Velocidade aprox. do motor (rpm)	Rolamento sem falhas	Rolamento com falhas			
			Diâmetro da Falha	Falha na pista interna	Falha na esfera	Falha na pista externa
0	1797	Normal_0	0,007"	IR007_0	B007_0	OU007@6_0
1	1772	Normal_1	0,014"	IR014_1	B014_1	OU014@6_1
2	1750	Normal_2	0,021"	IR021_2	B021_2	OU021@6_2

Fonte: Autoria própria

O conjunto de dados selecionados para treinamento e teste, constantes na Tabela 6, foram pré-processados para representar os dados de vibração em imagens. A seção 3.2 descreve essa conversão.

3.2. Pré processamento dos dados

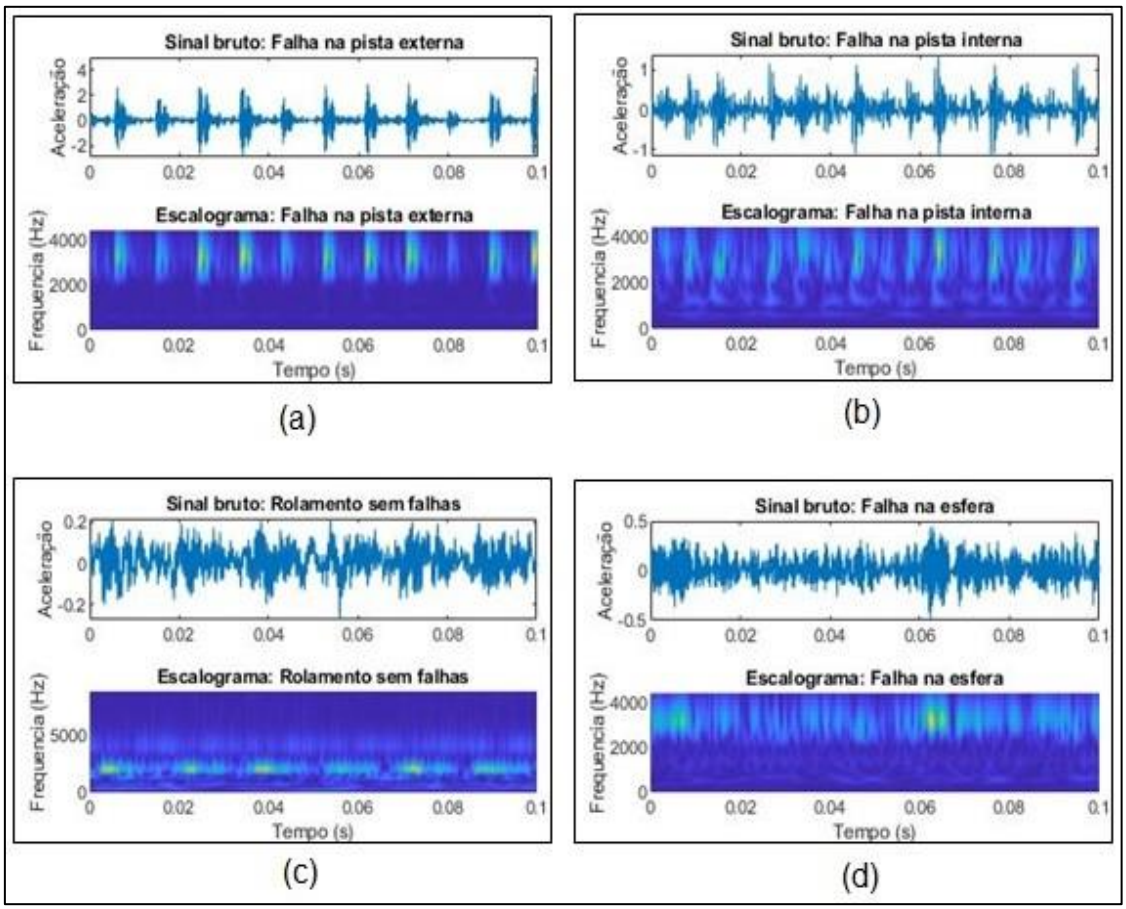
Os sinais de vibração da CWRU são brutos, ruidosos e não estacionários. Para extrair as características representativas da condição do rolamento através dos sinais de vibração, pode-se utilizar a Transformada Wavelet Continua (TWC) que, segundo Guo *et al.* (2019), é um método eficaz para analisar os sinais de vibração não estacionários para detecção de falhas em rolamentos, pois os coeficientes wavelet contínuos obtidos pela TWC contêm as informações completas do domínio do tempo-frequência dos sinais de vibração e evitam a perda de informações dos sinais originais. Além disso, XU *et al.* (2019) afirmam que, dentre as famílias wavelet, a wavelet Morlet provou ser superior em termos de análise de sinal de vibração de rolamentos devido à sua semelhança com os componentes de impulso transiente de falhas de rolamento. Por uma questão de brevidade, os interessados podem consultar Peng e Chu (2004) para uma revisão abrangente do uso da Transformada Wavelet no monitoramento da

condição e diagnóstico de falhas. Logo, para o monitoramento da condição de rolamentos, este estudo utilizou-se da TWC, usando Morlet, para gerar representações de tempo-frequência dos dados brutos e posteriormente a conversão em imagens (escalograma Walvelet), as quais foram alimentadas na arquitetura da CNN para a tarefa de classificação e diagnóstico de falhas no rolamento selecionado.

3.2.1 Transformada Wavelet e Escalogramas

Escalogramas são uma imagem gráfica da Transformada Wavelet que, segundo Verstraete *et al.* (2017), são uma representação linear de tempo-frequência eficaz para sinais não estacionários e transitórios. A Figura 20 mostra exemplos das leituras de vibração, com duração de 0,1 segundos, sendo representadas através de seu sinal de vibração original e através de seus respectivos escalogramas.

Figura 20 - Visualização de sinais no domínio do tempo vs tempo-frequência



Fonte: Autoria própria.

Os arquivos utilizados na Figura 20 são do rolamento de acionamento, na condição de falha (esfera, pista externa e pista interna) de diâmetro de 0,007”, taxa de amostragem de 12kHz e sem carga. O rolamento na condição sem falhas (normal) e sem carga no eixo também foi avaliado.

Na Figura 20, a relação entre o sinal original e seu escalograma demonstram algumas características uteis para o monitoramento das condições da saúde do rolamento. Para entendimento das observações, os cálculos das frequências característica de falhas foram realizados utilizando-se os dados da Tabela 4 e a frequência de rotação de 29,95Hz (1797 rpm). Seguem observações:

a) Figura 20 (a): O sinal bruto resultante do arquivo com falha na pista externa contém 11 impulsos porque a frequência característica de falha do rolamento testado é de 107,36 Hz. Assim, o escalograma mostra 11 picos distintos que se alinham com os impulsos no sinal de vibração;
b) Figura 20 (b): O escalograma da falha da pista interna mostra 16 picos distintos, o que é consistente com as frequências de passagem da esfera (162,18Hz). Como os impulsos no sinal no domínio do tempo não são tão dominantes quanto no caso de falha da pista externa, os picos distintos no escalograma mostram menos contraste.
c) Figura 20 (c): O escalograma da condição normal não mostra picos distintos dominantes;
d) Figura 20 (d): A condição de falha na esfera, tanto para sinal bruto no domínio do tempo quanto no escalograma tempo-frequência, não mostram picos dominante.

O número de picos distintos é uma boa característica para diferenciar entre falhas na pista externa, falhas na pista interna e condições normais. Já para a condição de falha na esfera, conforme Smith e Randall (2015), o conjunto de dados da CWRU necessita de técnicas sofisticadas para diagnosticar essa condição de falha. Logo o escalograma é um bom candidato para classificar falhas em rolamentos, sendo um desafio para a classificação de falhas na esfera, o que agrega valor a abordagem proposta neste trabalho.

3.2.2 Conversão escalogramas em imagens

Para o diagnóstico inteligente de falhas, os classificadores exigem que as amostras de dados sejam treinadas, com os arquivos do sinal original divididos em amostras iguais. Portando cada amostra do sinal de vibração deve conter pontos de amostras em quantidades suficientes para transmitir as características da condição do rolamento, ou seja, se o comprimento for muito curto, a amostra do sinal pode não refletir o estado de integridade do rolamento.

A estratégia utilizada neste estudo para gerar quantidades de imagens com pontos de amostras suficientes para extrair as características da integridade dos rolamentos, foi a divisão do sinal original em vários segmentos e posteriormente a conversão de cada segmento em escalogramas, salvos como imagens.

3.2.2.1 Divisão do arquivo original

Para melhorar a precisão, o arquivo original com as leituras de vibração foi dividido em vários segmentos. A estratégia utilizada foi a de garantir a relação entre a frequência de amostragem dos sinais e a frequência de rotação do eixo constantes e iguais a 2, ou seja, cada segmento de dados irá conter as leituras de vibração resultantes de 2 rotações do eixo. Isso garante que as imagens resultantes contenham as características necessárias para representar a condição do rolamento, pois, para a análise de vibração em rolamentos, uma revolução do eixo é a condição mínima necessária para extrair as características de falhas no rolamento, visto que essas características se repetem a cada revolução do eixo.

A equação 3.1 foi a equação utilizada para a segmentação do arquivo de vibração.

$$A = 2 * \frac{f_a}{(RPM/60)} \tag{3.1}$$

sendo A o tamanho da amostra (sinais de vibração/rotação), f_a a frequência de amostragem dos sinais e RPM a velocidade do eixo.

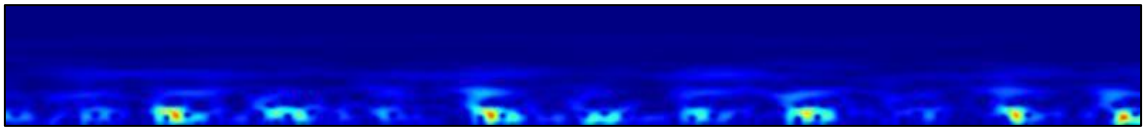
Vale ressaltar que a equação 3.1 permite que a quantidade de leituras de vibração, em cada segmento do arquivo, esteja contida em duas rotações do

eixo, independentemente da velocidade do eixo. Essa estratégia garantiu utilizar arquivos de vibração em várias velocidades e cargas no eixo.

3.2.2.2 Conversão em imagens

Os arquivos de vibração segmentados, conforme descrito na subseção 3.2.2.1, foram convertidos em imagens RGB (3 canais de cores). Se utilizarmos, por exemplo, o arquivo de vibração mostrado na Figura 19, utilizando a equação 3.1, o tamanho de cada segmento do arquivo terá aproximadamente 800 pontos de amostras e o espectro tempo-frequência resultante terá tamanho $800\ X\ f_e$, em que f_e representa o fator de escala e deve ser suficientemente grande para conter as características representativas do sinal de vibração. A Figura 21 mostra o espectro resultante deste exemplo.

Figura 21 - Escalograma das 800 primeiras leituras de vibração do arquivo IR007_0



Fonte: Autoria própria.

Para comprimir a imagem resultante e diminuir a quantidade de parâmetros a serem processados, foi utilizada a interpolação bicúbica, através da função *imresize* do Matlab. O tamanho resultante das imagens a serem utilizadas nesse estudo será aquele que apresentar a melhor relação precisão da rede versus menor custo computacional.

3.2.2.3 Imagens de treinamento e validação

O conjunto de imagens (normal, falha na pista interna, falha na pista externa e falha na esfera) foi dividido, aleatoriamente, em dois grupos, um para treinamento e outro para validação. A proporção será de 80% das imagens para treinamento e os 20% restantes para validação da rede.

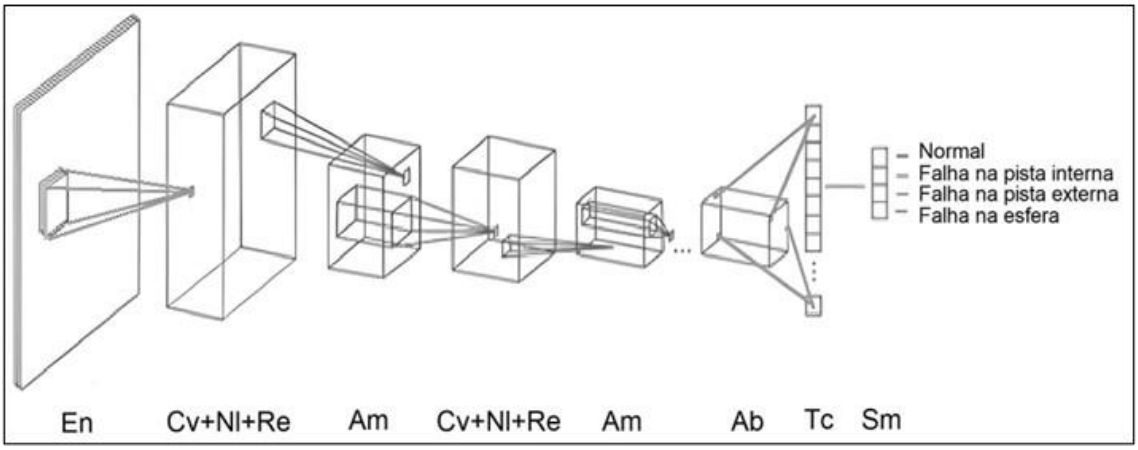
3.3. Rede neural convolucional

A rede neural convolucional, como uma tecnologia representativa de aprendizado profundo, irá receber as imagens originais, resultantes do pré-

processamento (seção 3.2), diretamente como entrada, identificar as características incorporadas e realizar o reconhecimento de padrões, automaticamente, dentro da estrutura da rede.

A arquitetura da rede CNN proposta para a tarefa de DDF no rolamento da CWRU está esquematizada na Figura 22 e será descrita nas próximas subseções.

Figura 22 - Esquemático da arquitetura CNN proposta



Fonte: Autoria própria.

As camadas, esquematizadas na Figura *22*, são as seguintes:

- En: Camada de entrada;
- Cv: Camada convolucional;
- Nl: Camada de normalização em lote;
- Re: Camada ReLU;
- Am: Camada de agrupamento máximo
- Ab: Camada de abandono;
- Tc: Camada totalmente conectada;
- Sm: Camada softmax.

3.3.1 Camada de entrada - En

Nessa camada, as imagens resultantes do pré-processamento serão normalizadas e inseridas diretamente na rede CNN. Conforme XU *et al.* (2019), o tamanho da imagem de entrada deve ser o menor possível, garantindo que a imagem contenha informações suficientes de falha e a normalização de dados deve ser aplicada toda vez que os dados forem propagados pela camada.

As imagens RGB resultantes da etapa de pré-processamento serão entradas com as seguintes configurações:

- Tamanho da imagem de entrada: tamanho definido na etapa de conversão do escalograma em imagens;
- Normalização: aplicado a normalização zerocenter, subtraindo a média dos dados.

3.3.2 Camadas convolucionais - Cv

As camadas convolucionais irão extrair as características da imagem que a última camada com parâmetros ajustáveis e a camada de classificação final irão usar para classificar a imagem de entrada. Nessas camadas, os filtros convolucionais irão se conectar a sub-regiões das imagens de entrada (escalogramas) e extrair, durante a varredura da imagem, as características representativas da condição de saúde do rolamento, gerando os mapas de características conforme descritos na subseção 2.3.3.

A configuração das camadas convolucionais seguiu as premissas descritas na sequência.

3.3.2.1 Tamanho e número de filtros

Assim como o tamanho da imagem de entrada, o tamanho e quantidade de filtros têm influência significativa na quantidade de parâmetros que deverão ser aprendidos pela rede e, como consequência, o custo computacional. Buscando otimizar o desempenho da rede, o tamanho e quantidade de filtros utilizados nas camadas convolucionais foram definidos da seguinte maneira:

- Todos os filtros das camadas convolucionais terão tamanhos 3x3, ou seja, irão varrer (convoluir) os dados de entrada em sub-regiões de tamanho 3x3 pixels;
- O número de filtros corresponde ao número de neurônios na camada que se conectam às sub-regiões, do tamanho do filtro, determinando o número de canais (mapas de características) na saída da camada. A estrutura da rede desenvolvida terá três camadas convolucionais, sendo que a primeira terá 16 filtros, 32 filtros para a segunda e a terceira e última camada convolucional terá 64 filtros.

3.3.2.2 Preenchimento

O preenchimento ao redor das bordas da imagem na camada convolucional foi definido de forma a manter o tamanho da saída igual ao tamanho da entrada. Essa configuração foi garantida utilizando a função *same* no Matlab.

3.3.2.3 Função para inicializar os pesos e bias

Durante o treinamento da rede, o software (Matlab) irá inicializar automaticamente os parâmetros que podem ser aprendidos de acordo com as seguintes propriedades de inicialização da camada:

a) Inicialização dos pesos: Será utilizado o inicializador Glorot (também conhecido como inicializador Xavier) que amostra independentemente de uma distribuição uniforme com média e variância zero. Para mais detalhes deste inicializador, consultar Glorot e Bengio (2010);
b) Inicialização do bias: O bias será inicializado com zeros.

3.3.3 Camada de normalização em lote - Nl

Entre cada camada convolucional e as camadas ReLU será utilizada a normalização em lote, pois conforme Tayyab *et al.* (2022), quando utilizada entre camadas convolucionais e não linearidades, como camadas ReLU, a camada de normalização em lote ajuda a acelerar o treinamento da CNN e a reduzir a sensibilidade à inicialização.

3.3.4 Camada ReLU - Re

A transformação não-linear nos dados de saída das camadas convolucionais será realizada utilizando a função ReLU, conforme descrito na subseção 2.3.7.

3.3.5 Camada de agrupamento (*Pooling*)

Essa camada irá receber as informações extraídas na etapa de convolução, resumir os dados de sub-regiões da imagem em um único valor e repassa-los para a próxima camada. O processo de agrupamento irá utilizar a

função de agrupamento máximo (Am), através da convolução de sub-regiões de tamanho 2x2 pixels e preenchimento de bordas 2x2.

3.3.6 Camada de abandono (*Dropout*) - Ab

Além das camadas de normalização em lote e do uso da função ReLU, também será utilizada uma camada de abandono para ajudar a reduzir, ainda mais, as chances de *overfitting* na rede. A camada de abandono será utilizada para eliminar (definir como zero), aleatoriamente, 20% dos dados antes de entrar na camada totalmente conectada, interrompendo assim a co-adaptação dos detectores de recursos, pois as unidades descartadas não poderão influenciar outras unidades retidas.

3.3.7 Camada totalmente conectada - Tc

Nessa camada é onde é iniciado o processo para classificar a condição de integridade do rolamento, através das informações extraídas pelas camadas anteriores. A configuração dessa camada foi definida da seguinte forma:

a) Tamanho de saída: A camada terá 4 saídas, que corresponde às quantidades de condições de integridade do rolamento (normal, falha na pista interna, falha na pista externa e falha na esfera) para a tarefa de classificação;
b) Inicialização dos pesos: Será utilizado o inicializador Glorot (também conhecido como inicializador Xavier), conforme abordado na subseção 3.3.2.
c) Inicialização do bias: O bias será inicializado com zeros.

3.3.8 Camada *softmax* - Sm

As pontuações de probabilidade serão calculadas para cada rótulo de classe (condições da integridade do rolamento) utilizando a função *softmax*, conforme descrito no item 2.3.11.

3.3.9 Camada de classificação - Cl

Essa será a última camada da estrutura CNN proposta, sendo a camada de saída de classificação. Conforme The MathWorks (2023), a camada de

classificação calcula a perda de entropia cruzada para tarefas de classificação e classificação ponderada com classes mutuamente exclusivas. Quando utilizada no Matlab, *software* utilizado neste estudo, a camada infere o número de classes do tamanho de saída da camada anterior, pega os valores da função *softmax* e atribui cada entrada a uma das classes mutuamente exclusivas usando a função de entropia cruzada para a tarefa de codificação.

A configuração dessa camada será a seguinte:

a) Terá 4 classes, correspondentes a saída da camada totalmente conectada;
b) Será aplicada a perda de entropia cruzada não ponderada;
c) Os nomes das classes serão atribuídos automaticamente durante o treinamento da rede. O *software* irá utilizar os nomes das pastas, onde as imagens de vibração estão salvas, e usar como rótulos para classificação.

4 Resultados e discussões

Este capítulo é dedicado a descrever a implementação da metodologia proposta e avaliar os resultados e eficácia da abordagem proposta. Os dados de vibração foram pré-processados e inseridos diretamente na arquitetura CNN para treinamento e testes. Três experimentos foram conduzidos para verificar a efetividade da abordagem proposta tendo, por fim, os resultados comparados com outros trabalhos similares e recentemente publicados.

Todos os experimentos foram realizados com o *software* Matlab R2020a, em um computador portátil, equipado com um processador Intel core I5, 2.7 GHz e 4 GB de memória. Os resultados serão demonstrados nas subseções seguintes.

4.1. Pré- processamento dos dados

Nesta etapa foi realizada a conversão dos dados de vibração do domínio do tempo para imagens RGB com as características de integridade do rolamento em estudo. Após diversas configurações, o tamanho da imagem de saída foi definido como 256x256x3, sendo essa a configuração que apresentou melhor desempenho em eficiência versus tempo de processamento computacional.

O algoritmo Matlab utilizado para conversão dos sinais de vibração segmentados em escalogramas, salvos como imagem, é demonstrado na Figura 23. Com o algoritmo desenvolvido, as imagens resultantes de cada segmento do arquivo de sinais de vibração (ARQUIVO_X) foram, sequencialmente, salvas em pastas rotuladas com a respectiva condição do rolamento (CONDIÇÃO_ARQUIVO_X).

Para garantir que as imagens armazenadas recebam o rótulo da respectiva pasta, foi utilizada a função *imageDatastore* do Matlab, pois essa função rotula automaticamente as imagens com base nos nomes das pastas. Essa ação garantiu o treinamento supervisionado, pois as etapas e funções posteriores no algoritmo irão processar esses rótulos para a classificação de falhas.

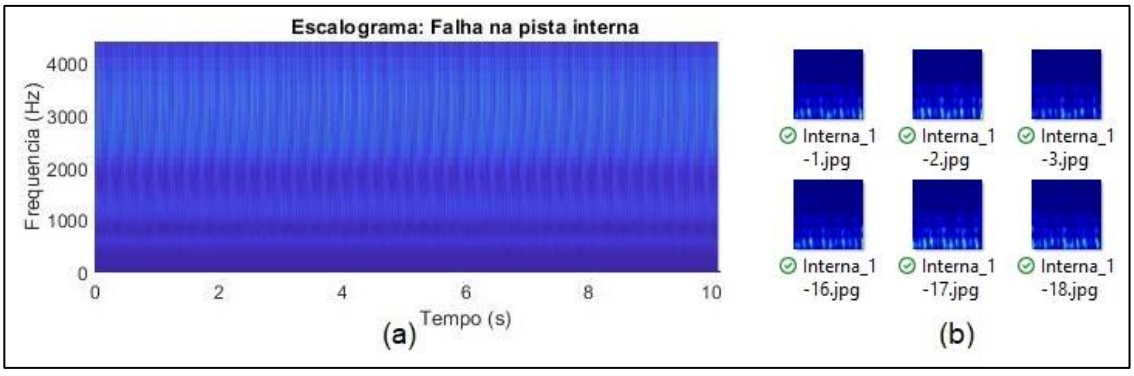
Figura 23 - Algoritmo Matlab do pré-processamento dos arquivos de vibração

```
%Dados de vibração do ARQUIVO_X
Dados_Treinamento = load(fullfile('.','Arquivos',
'Treinamento',ARQUIVO_X));
Dados_Velocidade = Dados_Treinamento.RPM;
Dados_Treinamento = Dados_Treinamento.Sinal;
Freq_Amostra = 12000;  %Frequência de leitura em Hz dos sinais
de vibração
Freq_RPM = 2; Quantidade de voltas do eixo do motor

%Converter os sinais de vibração em escalogramas e salvar as
imagens para treinamento

Intervalo_Amostra =
Freq_RPM*(floor((Freq_Amostra)/(Dados_Velocidade/60)));  %
Intervalo (divisão do sinal) entre cada leitura de vibração.
Razão entre a frequência de amostragem com a de rotação do eixo
Qde_Imagem = floor(numel(Dados_Treinamento)/Intervalo_Amostra);

for i = 1:Qde_Imagem
  Sinal = Dados_Treinamento(Intervalo_Amostra*(i-
1)+1:Intervalo_Amostra*i);
  [twc,freq] = cwt(Sinal,'amor', Freq_Amostra); %amor = Morlet
(Gabor)
  twc = abs(twc);
  Imagem = ind2rgb(round(rescale(flip(twc),0,255)),jet(320));
  Local_Titulo = fullfile('.','Imagens', 'Imagens_Treinamento',
'CONDIÇÃO_ARQUIVO_X',['ARQUIVO_X' '-' num2str(i) '.jpg']);
  imwrite(imresize(Imagem,[256,256]),Local_Titulo);
end
```

Fonte: Autoria própria.

A Figura 24 (a) mostra um exemplo do escalograma do arquivo IR007_0 (falha na pista interna, 12kHz, diâmetro de falha 0,007 e sem carga) completo e a Figura 24 (b) mostra os escalogramas de segmentos do arquivo IR007_0, os quais foram comprimidos e salvos em imagens de tamanho 256x256 utilizando o algoritmo demonstrado na Figura 23.

Figura 24 – Escalogramas do arquivo IR007_0 completo e segmentado



Fonte: Autoria própria.

Percebe-se na Figura 24 que, além de reduzir a quantidade de parâmetros a serem processados, a segmentação proporciona uma melhor identificação da falha presente no rolamento, pois os picos correspondentes às frequências de falhas tornaram-se evidentes nas imagens do arquivo de vibração segmentado.

4.1.1 Imagens resultantes do pré-processamento

A etapa de pré-processamento gerou um total de 1797 imagens. A fim de equilibrar e permitir que as condições de integridade do rolamento (normal, falha na pista interna, falha na pista externa e falha na esfera) contivessem as mesmas quantidades de imagens, tomou-se como referência a condição de integridade com menor número de imagens e feito o ajuste, aleatoriamente, das quantidades de imagens das demais condições de integridade, totalizando 1788 imagens para desenvolvimento da rede CNN. A Tabela 7 mostra o resumo das quantidades de imagens resultantes dos arquivos utilizados (arquivos da Tabela 6).

Tabela 7 - Quantidade de imagens resultantes do pré-processamento

Condições de integridade do rolamento	Quantidade de imagens resultantes	Quantidade ajustada de imagens
Falha na esfera	451	447
Falha pista externa	450	447
Falha pista interna	449	447
Normal	447	447

Fonte: Autoria própria.

Apesar da quantidade de imagens resultantes ser praticamente igual, condição garantida pela equação 3.1 (subseção 3.2.2.1), o ajuste das quantidades visou equilibrar quantidade de imagens em cada condição de integridade do rolamento, pois se uma condição apresentar mais imagens que outra condição, o aprendizado pode ser comprometido.

4.1.2 Imagens de treinamento e imagens de validação

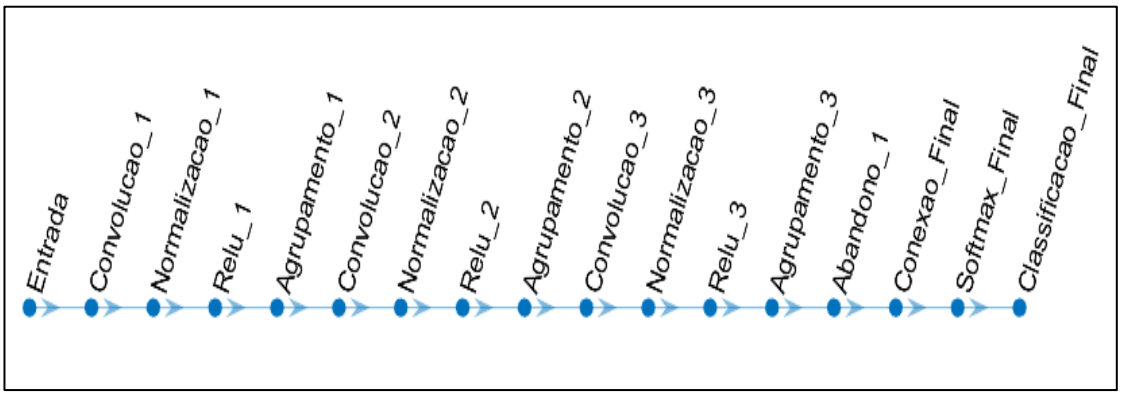
Conforme descrito na metodologia, o conjunto resultante de 1788 imagens foi dividido em 80% para treinamento e 20% para validação.

Vale ressaltar que a divisão entre imagens de treinamento e validação foi realizada de forma aleatória.

4.2. Arquitetura da CNN para DDF em rolamentos

A arquitetura da CNN criada com as configurações definidas na seção 3.3 resultou em uma rede em série, com camadas dispostas uma após a outra, possuindo uma única camada de entrada e uma única camada de saída. A sequência das camadas da rede pode ser visualizada na Figura 25.

Figura 25 – Disposição das camadas na arquitetura CNN criada



Fonte: Autoria própria.

Conforme mostrado na Figura 25, a rede CNN criada para a tarefa de DDF em rolamentos é composta por 17 camadas, sendo:

- 7 camadas com pesos que podem ser ajustados;
- 3 camadas convolucionais;
- 3 camadas de normalização;
- 1 camada totalmente conectada.

Segundo Verstraete *et al.* (2017), é importante otimizar os parâmetros que podem ser ajustáveis, equilibrando o tempo de treinamento versus a precisão da previsão. Após diversos testes, a configuração e os parâmetros que serão computados pele rede são demonstrados na Tabela 8.

Tabela 8 – Configuração e parâmetros da CNN proposta

(continua)

Camada	Nome da camada	Configuração da camada	Ativações	Parâmetros ajustáveis	Total de parâmetros
1	Entrada	256x256x3 images with 'zerocenter' normalization	256×256×3	-	0
2	Convolucao_1	16 3x3x3 convolutions with stride [1 1] and padding 'same'	256×256×16	Weights: 3×3×3×16 Bias: 1×1×16	448
3	Normalizacao_1	Batch normalization with 16 channels	256×256×16	Offset: 1×1×16 Scale: 1×1×16	32
4	Relu_1	ReLU	256×256×16	-	0
5	Agrupamento_1	2x2 max pooling with stride [2 2] and padding [0 0 0 0]	128×128×16	-	0
6	Convolucao_2	32 3x3x16 convolutions with stride [1 1] and padding 'same'	128×128×32	Weights: 3×3×16×32 Bias: 1×1×32	4640
7	Normalizacao_2	Batch normalization with 32 channels	128×128×32	Offset: 1×1×32 Scale: 1×1×32	64
8	Relu_2	ReLU	128×128×32	-	0
9	Agrupamento_2	2x2 max pooling with stride [2 2] and padding [0 0 0 0]	64×64×32	-	0

Tabela 8 - Configuração e parâmetros da CNN proposta

(conclusão)

Camada	Nome da camada	Configuração da camada	Ativações	Parâmetros ajustáveis	Total de parâmetros
10	Convolucao_3	64 3x3x32 convolutions with stride [1 1] and padding 'same'	64×64×64	Weights: 3×3×32×64 Bias: 1×1×64	18496
11	Normalizacao_3	Batch normalization with 64 channels	64×64×64	Offset: 1×1×64 Scale: 1×1×64	128
12	Relu_3	ReLU	64×64×64	-	0
13	Agrupamento_3	2x2 max pooling with stride [2 2] and padding [0 0 0 0]	32×32×64	-	0
14	Abandono_1	20% dropout	32×32×64	-	0
15	Conexao_Final	4 fully connected layer	1×1×4	Weights: 4×65536 Bias: 4×1	262148
16	Softmax_Final	softmax	1×1×4	-	0
17	Classificacao_Final	crossentropyex with 'Falha na esfera' and 3 other classes	-	-	0

Fonte: Autoria própria.

Conforme comentado na subseção 2.3.9, pode-se observar na Tabela 8 que a camada com maior número de parâmetros ajustáveis é a camada totalmente conectada (15ª camada), exigindo assim um maior poder computacional durante o treinamento da rede.

4.3. Parâmetros e opções de treinamento

Depois de definir a arquitetura da rede CNN, a próxima etapa é configurar as opções de treinamento. O treinamento da rede CNN criada foi definido com as opções constantes na Tabela 9.

Tabela 9 - Opções de treinamento da CNN

Argumento Matlab	Valor	Descrição
adam		Otimizador 'adam'(derivado da estimativa de momento adaptativo). KINGMA e BA (2014) descrevem esse solucionador
InitialLearnRate	0.001	Taxa de aprendizado inicial
LearnRateSchedule	piecewise	Opção para diminuir a taxa de aprendizado a cada certo número de épocas
LearnRateDropPeriod	3	Número de épocas para diminuir a taxa de aprendizado
LearnRateDropFactor	0.1	Multiplicador da taxa de aprendizado inicial
MaxEpochs	9	Quantidade máxima de épocas a serem usadas para treinamento
MiniBatchSize	64	Tamanho do subconjunto de dados (minilote) usado para avaliar o gradiente da função de perda e atualizar os parâmetros
ValidationData	Img_Validacao	Conjunto de imagens para validação
ValidationFrequency	10	Frequência (número de iterações) em que os dados serão validados
Shuffle	every-epoch	Embaralhamento dos dados a cada época de treinamento e frequência de validação
Verbose	true	Informações de progresso do treinamento na janela de comando do Matlab
VerboseFrequency	10	Número de iterações entre a impressão das informações de progresso do treinamento
Plots	training-progress	Plotar a evolução do treinamento
ValidationPatience	6	Critério de parada: número de vezes que a perda no conjunto de validação pode ser maior ou igual à menor perda anterior antes que o treinamento da rede

Fonte: Autoria própria.

Após diversos testes realizados no decorrer do desenvolvimento da CNN, a Tabela 9 apresenta a versão final dos argumentos e opções de treinamento, pertencentes ao *Deep Learning Toolbox*™ do MATLAB®, que tiveram seus valores ajustados para obtenção da melhor relação precisão versus tempo computacional no trabalho desenvolvido. As demais opções e argumentos

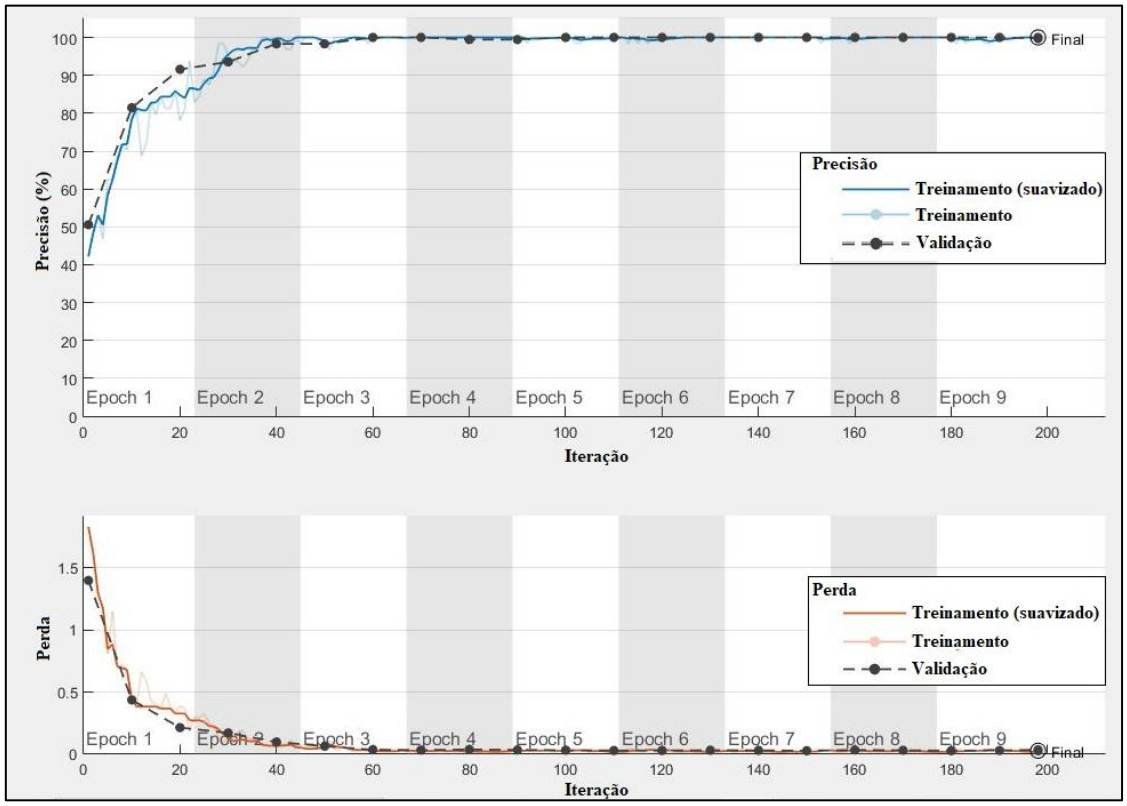
necessários para o treinamento da rede foram mantidos com os valores padrões do MATLAB®, não sendo necessário configurá-los. Para uma leitura mais abrangente, as opções de treinamento podem ser consultadas em The MathWorks (2023).

4.4. Treinamento da rede

O treinamento de uma CNN é um processo iterativo que envolve a minimização de uma função de perda através do algoritmo de descida de gradiente. Em cada iteração, o gradiente da função de perda é avaliado e os pesos do algoritmo de descida são atualizados.

Depois de especificar a arquitetura da rede e os parâmetros de treinamento, a rede foi treinada usando os dados de treinamento (80% do conjunto de imagens pré-processadas). A Figura *26* mostra o gráfico com a evolução do treinamento da rede criada.

Figura 26 – Evolução gráfica do treinamento da CNN proposta



Fonte: Autoria própria.

Para uma análise mais detalhada, os valores numéricos da evolução da rede, durante o treinamento, podem ser observados na Tabela 10.

Tabela 10 - Evolução do treinamento da CNN proposta

Época	Iteração	Tempo decorrido (hh:mm:ss)	Precisão do mini-lote	Precisão de validação	Perda do mini-lote	Perda de validação	Taxa de aprendizagem básica
1	1	00:00:33	42.19%	50.56%	18.295	13.988	1,00E-04
1	10	00:01:57	79.69%	81.46%	0.4436	0.4374	1,00E-04
1	20	00:03:26	78.13%	91.57%	0.3891	0.2147	1,00E-04
2	30	00:04:59	95.31%	93.54%	0.1269	0.1720	1,00E-04
2	40	00:06:29	100.00%	98.31%	0.0671	0.0990	1,00E-04
3	50	00:07:57	98.44%	98.31%	0.0876	0.0650	1,00E-04
3	60	00:09:25	100.00%	100.00%	0.0259	0.0370	1,00E-04
4	70	00:10:59	100.00%	100.00%	0.0333	0.0339	1,00E-05
4	80	00:12:28	100.00%	99.44%	0.0158	0.0387	1,00E-05
5	90	00:13:58	100.00%	99.44%	0.0153	0.0354	1,00E-05
5	100	00:15:32	100.00%	100.00%	0.0287	0.0331	1,00E-05
5	110	00:17:00	100.00%	100.00%	0.0256	0.0303	1,00E-05
6	120	00:18:31	100.00%	100.00%	0.0309	0.0302	1,00E-05
6	130	00:20:02	100.00%	100.00%	0.0358	0.0323	1,00E-05
7	140	00:21:31	100.00%	100.00%	0.0123	0.0315	1,00E-06
7	150	00:22:58	100.00%	100.00%	0.0219	0.0283	1,00E-06
8	160	00:24:25	100.00%	100.00%	0.0217	0.0351	1,00E-06
8	170	00:25:51	100.00%	100.00%	0.0195	0.0323	1,00E-06
9	180	00:27:15	100.00%	100.00%	0.0155	0.0285	1,00E-06
9	190	00:28:43	100.00%	100.00%	0.0241	0.0310	1,00E-06
9	198	00:29:53	100.00%	100.00%	0.0181	0.0268	1,00E-06

Fonte: Autoria própria.

Conforme demonstrados na Figura 26 e Tabela 10, a rede criada alcançou precisão de 100% com o conjunto de dados da Tabela 6, convertidos em 1788 imagens. A precisão de 100% foi alcançada na terceira época de treinamento em 9min e 25s. O critério de parada definido na seção 4.3 não foi alcançado pois a perda no conjunto de validação continuou em decaimento sendo, portando, concluído o treinamento com o número máximo de épocas definido.

4.5. Avaliação da rede

A maneira ideal de avaliar o resultado do treinamento é fazer com que a rede classifique os dados que nunca foram usados durante a etapa de treinamento. Nesse sentido foram realizados três experimentos com os arquivos de vibração em diferentes configurações (velocidade e carga no motor e tipos e diâmetros de falhas no rolamento). Os resultados serão demonstrados nas subseções seguintes.

4.5.1 Arquivos teste e treinamento com as mesmas configurações

Nessa fase foi avaliada a precisão da rede treinada com dados de vibração em diferentes configurações (carga e velocidade do motor e localização e diâmetros de falhas no rolamento) e validada/testada com outros dados, porém com a mesma configuração dos dados de treinamento.

As imagens utilizadas para essa avaliação foram geradas a partir de 30% dos arquivos de vibração com carga no motor em 0, 1 e 2 hp. As imagens resultantes dos 12 arquivos selecionados, foram divididas na proporção de 80% para treinamento e o restante para validação da rede. Essa configuração abrange os arquivos de vibração presentes na Tabela 6, sendo resumidos na Tabela 11.

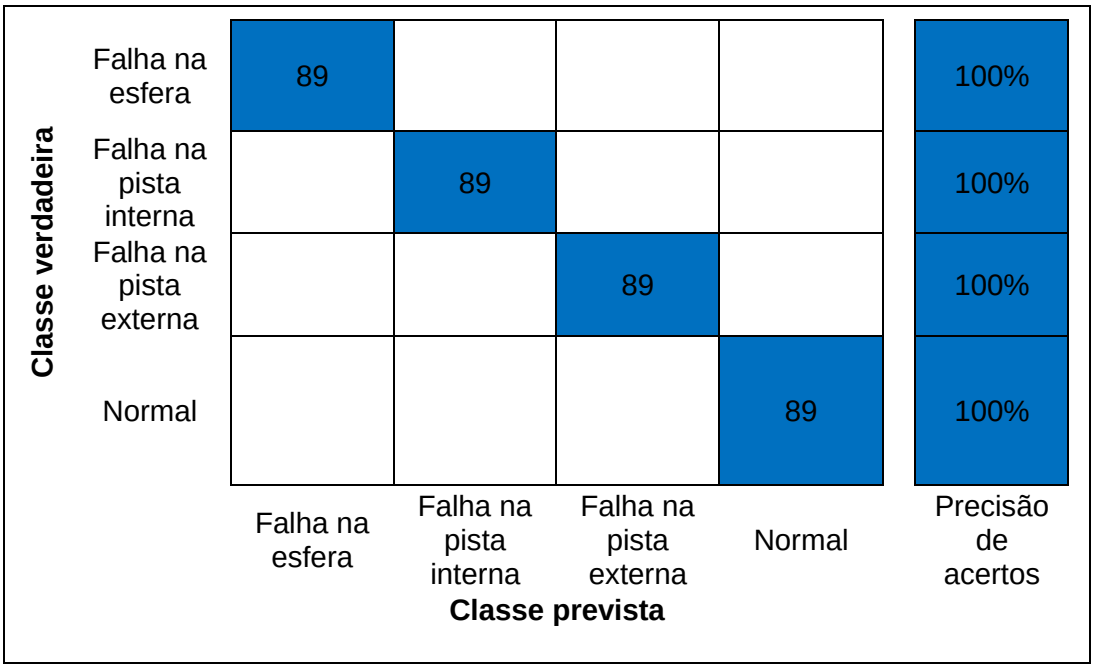
Tabela 11 - Arquivos de treinamento e teste da 1ª avaliação

Configuração dos arquivos	Quantidade de imagens	Velocidade aprox. do motor (rpm)	Carga no motor (hp)
Arquivos de treinamento	1430 (80%)	1750, 1772 e 1797	0; 1 e 2
Arquivos de teste	356 (20%)	1750, 1772 e 1797	0; 1 e 2

Fonte: Autoria própria.

Utilizando o conjunto de 20% das imagens separadas, aleatoriamente, para avaliação da rede, a CNN proposta classificou corretamente as 356 imagens de validação. A Figura 27 resume o resultado da classificação das imagens de validação.

Figura 27 – Matriz de confusão resultantes da 1ª avaliação



Fonte: Autoria própria.

A matriz de confusão da Figura 27 confirma a precisão de 100% alcançada pela rede durante o treinamento e validação.

4.5.2 Arquivos de teste e treinamento em configurações diferentes

Para testar o poder de generalização da arquitetura CNN desenvolvida, foi utilizado um conjunto de dados de vibração com configuração diferente da configuração dos arquivos de treinamento.

Tomando-se como referência a Tabela 5, nessa configuração a rede foi treinada com os arquivos de vibração com carga no motor em 0, 1 e 2 hp e testada com os arquivos de vibração com carga no motor de 3 hp. A Tabela 12 mostra um resumo das diferenças entre as configurações dos arquivos de treinamento e teste.

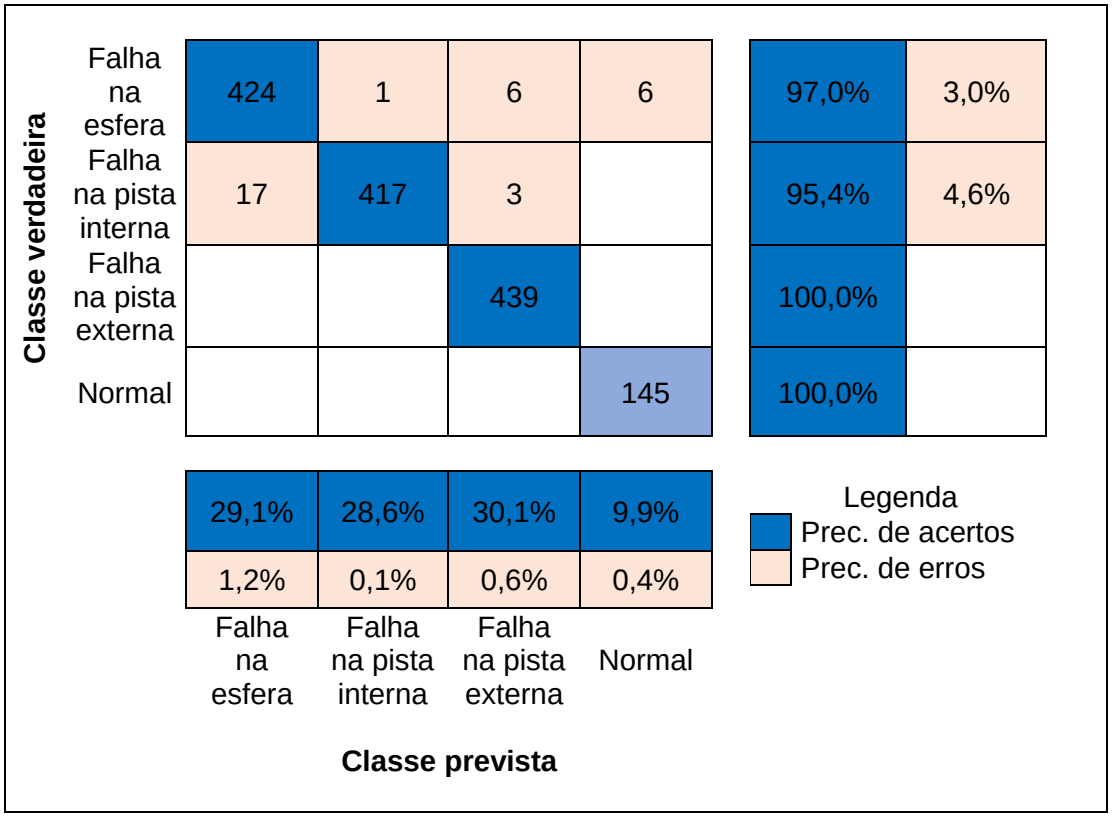
Tabela 12 - Arquivos de treinamento e teste da 2ª avaliação

Configuração dos arquivos	Quantidade de imagens	Velocidade aprox. do motor (rpm)	Carga no motor (hp)
Arquivos de treinamento	1430	1750, 1772 e 1797	0; 1 e 2
Arquivos de teste	1458	1730	3

Fonte: Autoria própria.

A precisão de classificação da rede com os arquivos de teste é demonstrada na Figura 28.

Figura 28 - Matriz de confusão resultantes da 2ª avaliação



Fonte: Autoria própria.

A precisão da rede com esse conjunto de dados foi de 97.7%. Esse resultado demonstra o alto poder de generalização da abordagem proposta, uma vez que o conjunto de imagens analisadas não fazia parte do conjunto de imagens de treinamento. Portanto, o estudo desenvolvido minimiza os requisitos de dados de treinamento, que são críticos para aplicações industriais, pois a maioria dos sistemas práticos não possuem dados defeituosos cobrindo toda a faixa de operação do motor.

4.5.3 Arquivos de teste em todas as configurações (iguais e diferentes da configuração de treinamento)

Nessa etapa foi verificado a precisão da rede sendo avaliada com os conjuntos de imagens de treinamento resultantes dos arquivos na configuração de carga no motor em 0, 1 e 2 hp e o conjunto de imagens de teste resultantes dos arquivos de vibração com 0, 1, 2 e 3HP de carga no motor. A Tabela 13 resume os dados utilizados nessa configuração de teste.

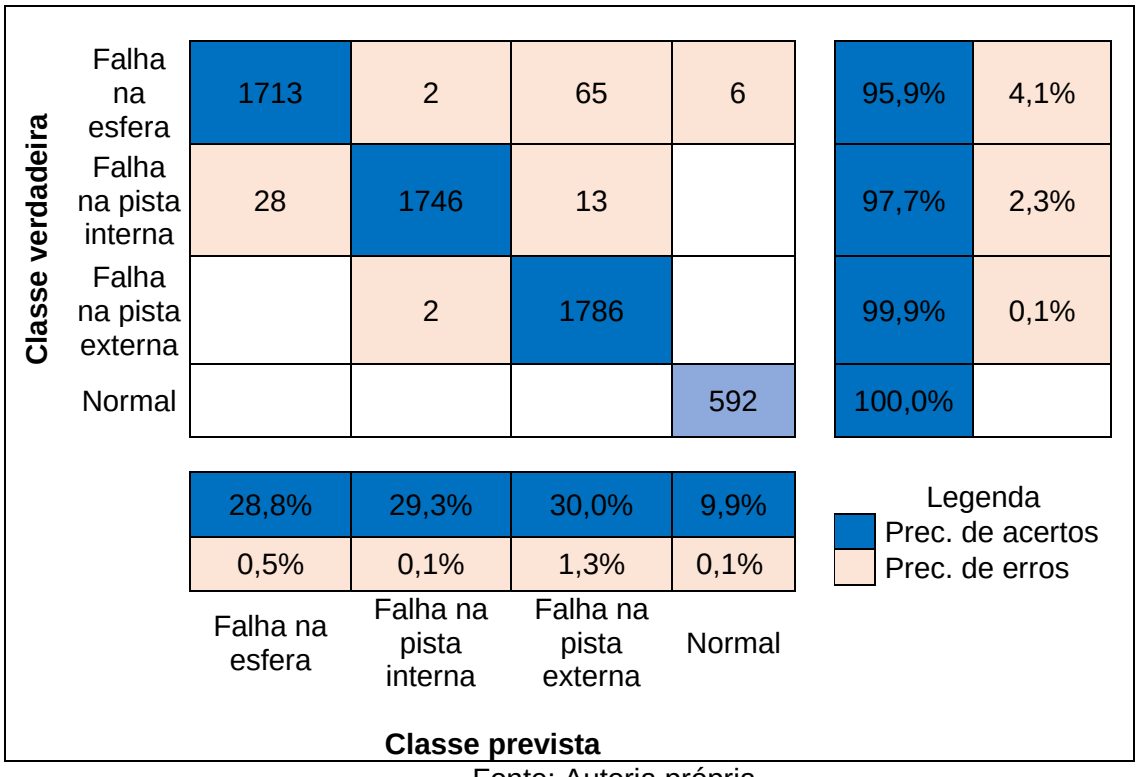
Tabela 13 - Arquivos de treinamento e teste da 3ª avaliação

Configuração dos arquivos	Quantidade de imagens	Velocidade aprox. do motor (rpm)	Carga no motor (hp)
Arquivos de treinamento	1430	1750, 1772 e 1797	0; 1 e 2
Arquivos de teste	5953	1750, 1772, 1797 e 1730	0; 1; 2 e 3

Fonte: Autoria própria.

O resultado dessa avalição, para o total de 5953 imagens, está demonstrado na matriz de confusão da Figura 29.

Figura 29 - Matriz de confusão resultantes da 3ª avaliação



Fonte: Autoria própria.

Percebe-se na Figura 29 que a categoria de falha com menor precisão foi a de falha na esfera. Esse resultado já era esperado, uma vez que Smith e Randall (2015) alertaram sobre a dificuldade, no conjunto de dados de vibração da CWRU, do diagnóstico de falha na esfera. Smith e Randall (2015) encontraram evidencias de que há falhas, não intencionais, na pista externa e interna presentes em muitos conjuntos de dados de falha na esfera, o que dificulta o diagnóstico dessa falha.

Calculando-se as precisões de classificação constantes na Figura 29, a CNN desenvolvida alcançou precisão de 98%. Esse resultado também demonstra o alto poder de aprendizagem e generalização da CNN desenvolvida, pois os dados utilizados nessa avaliação são constituídos de várias configurações de teste, incluindo configurações não utilizadas para o treinamento da rede.

A seção 4.6 irá comparar os resultados obtidos com a CNN desenvolvida e a outros presentes na literatura.

4.6. Comparação com trabalhos similares

Para avaliar a eficácia do modelo CNN desenvolvido, o mesmo foi comparado com técnicas IA recentes que utilizaram a rede neural convolucional para a tarefa de detecção e diagnóstico de falha em rolamento. Os critérios de comparação são baseados em estudos que utilizaram o conjunto de dados CWRU em condições de diferentes cargas e velocidades no motor. Como cada estudo tem suas particularidades (combinação de técnicas utilizadas e arquitetura da CNN), buscou-se estudos que apresentaram configurações semelhantes às abordadas na seção 4.5.

As precisões de cada método comparado estão presentes em seus respectivos artigos, sendo que para chegar a configurações passiveis de comparação com o método proposto, em alguns trabalhos foi preciso calcular a precisão para a configuração mais próxima possível do estudo desenvolvido neste trabalho. Ou seja, as precisões de cada método presentes na Tabela 14 estão calculadas para as seguintes configurações avaliadas:

- Configuração 1: A rede CNN foi treinada com os dados de vibração em diferentes configurações (carga e velocidade do motor e

localização e diâmetros de falhas no rolamento) e validada/testada com outros dados, porém com a mesma configuração dos dados de treinamento;

- Configuração 2: A rede CNN foi treinada com os dados de vibração em diferentes configurações (carga e velocidade do motor e localização e diâmetros de falhas no rolamento) e validada/testada com dados em configuração diferente da configuração dos dados de treinamento;
- Configuração 3: A rede CNN foi treinada com os dados de vibração em diferentes configurações (carga e velocidade do motor e localização e diâmetros de falhas no rolamento) e validada/testada utilizando todas as configurações de ensaio.

Os resultados da comparação com as mais recentes técnicas são apresentados na Tabela 14. Como forma de entender os trabalhos presentes na Tabela 14, a ideia principal dos estudos comparados é resumida a seguir:

a) Ding e He (2017): Abordagem baseada na imagem de Energia de Pacote Wavelet (EPW) e CNN. A transformada de Pacote Wavelet é combinada com a técnica de Reconstrução do Espaço de Fase (REF) para reconstruir uma imagem 2-D. As características identificáveis podem ser aprendidas pela arquitetura CNN proposta;
b) XU *et al*. (2019): Método de diagnóstico de falhas de rolamentos utilizando a Transformada Wavelet Contínua (TWC) em conjunto com a Rede Neural Convolucional (CNN) e vários classificadores Floresta Aleatória (FA). Primeiramente, sinais de vibração no domínio do tempo são convertidos, através da TWC, em imagens bidimensionais (2D) em escala de cinza. Em segundo lugar, um modelo extrai automaticamente características sensíveis à DDF de falhas nas imagens. Finalmente, recursos extraídos em diferentes camadas da CNN são alimentados em vários classificadores de FA para classificar as falhas de forma independente, e as saídas dos vários classificadores são agregadas pela estratégia de conjunto do tipo "o vencedor leva tudo" para fornecer o resultado do diagnóstico;

c) Guo *et al.* (2019): método de DDF em rolamentos baseada em CNN multitarefa e Fusão de Informações (FI). Utiliza a TWC para decompor os sinais de vibração e em seguida adiciona informações com base na estrutura e condições operacionais dos rolamentos. As informações resultantes são fundidas em uma matriz tridimensional (3D), que é servida como entrada para a CNN;
d) Chen *et al.* (2020): método que utiliza o sinal de vibração unidimensional, coletado como dados brutos através de sensores de séries temporais, e em seguida usa como a entrada da rede CNN-1D para a tarefa detecção e diagnostico de falhas em rolamentos;
e) Lee e Le (2021): modelo baseado na imagem do Espectro de Persistência (EP) e na Rede Neural Convolucional com arquitetura ResNet (R-CNN). O espectro de persistência é extraído do envelope do sinal de vibração bruto. Em seguida, a imagem do espectro de persistência é construída com base na Transformada de Fourier de Tempo Curto (TFTC), sendo utilizada como entrada para a R-CNN. Como resultado, o modelo proposto opera de forma eficiente e com alta precisão, não apenas sob diversas cargas de trabalho, mas também sob condições de ruído;
f) Li, Liu e Xiao (2021): método de diagnóstico inteligente de falhas de rolamento baseado em Curtose de Espectro (CE), Espectro de Envelope (EE) e Rede Neural Convolucional (CNN). Neste método, a filtragem CE e filtro passa-banda são usados para melhorar a taxa sinal-ruído de falha dos sinais de vibração originais. Em seguida, as informações das frequências características da falha relacionadas à velocidade de rotação são extraídas pela análise EE, gerando imagens que, posteriormente, são usadas como entrada em um modelo CNN para identificar defeitos no rolamento.

Dentre os trabalhos apresentados na Tabela 14, com exceção da quantidade de dados utilizados para treinamento e avaliação da CNN, apenas Ding e He (2017) realizaram a configuração 2 igual a utilizada no desenvolvimento desde trabalho (seção 4.5.2), não sendo necessário recalcular a precisão, contante no artigo, para essa configuração de ensaio.

Tabela 14 - Comparação entre redes CNNs

Estudo	Autor (es)	Principais técnicas	Precisão na configuração 1 (%)	Precisão na configuração 2 (%)	Precisão na configuração 3 (%)
a)	Ding e He (2017)	WPE+CNN	99,8	96,8	-
b)	Xu *et al.* (2019)	TWC+CNN+FA	99,73	99,08	-
c)	Guo *et al.* (2019)	TWC+FI+CNN	-	96,79	-
d)	Chen *et al.* (2020)	CNN-1D	99,20	98,30	-
e)	Lee e Le (2021)	TFTC+EP+CNN	95,84	91,50	94,75
f)	Li, Liu e Xiao (2021)	CE-EE-CNN	-	-	93,20
Método proposto		TWC+CNN	100,00	97,70	98,00

Fonte: Autoria própria

Analisando os dados constantes na Tabela 14, percebe-se que na configuração 2, quando comparado com a avaliação na configuração 1, a precisão de todos os métodos é reduzida. A causa dessa redução é devido a CNN realizar a tarefa de DDF em dados de vibração em configuração de ensaio (carga de 3 hp no motor) diferente das configurações de ensaios dos dados de treinamento (cargas de 0 hp a 2hp no motor).

Na configuração 2, o método proposto alcançou uma precisão de DDF inferior aos métodos de Xu *et al.* (2019) e Chen *et al.* (2020). Esse fato é resultado, principalmente, da proporção da quantidade de dados de treinamento e avaliação utilizados para desenvolver a CNN. Xu *et al.* (2019), cujo método alcançou a maior precisão entre os métodos avaliados (99,08%), além de utilizar um conjunto dados de treinamento três vezes maior que o conjunto de dados de avaliação, se beneficiou da utilização de diversos classificadores, onde o classificador que alcança a maior precisão é o que representa a precisão do método. Já a CNN com a abordagem proposta, dentre todos os trabalhos constantes na Tabela 14, é a que utilizou a menor proporção na quantidade de arquivos para treinamento e avaliação da rede, sendo utilizado a proporção de apenas 0,98, ou seja, a quantidade de arquivos de treinamento foi menor que a quantidade de arquivos de validação.

Além do alto poder de aprendizagem e generalização demonstrados na Tabela 14, a menor quantidade de arquivos de treinamento utilizados agregou valor a abordagem proposta pois, segundo o estudo de Wang *et al*. (2019), a quantidade de dados de treinamento compromete a precisão da rede, pois com o aumento das amostras de treinamento, a precisão de cada método é aumentada.

5 Conclusão

A análise de vibração é a técnica mais consistente quando se trata de monitoramento de condição de equipamentos rotativos. Neste trabalho, uma combinação da Transformada Wavelet Continua (TWC) e da Rede Neural Convolucional (CNN) foi usada para detecção e diagnostico de falhas de rolamentos, usando dados de falhas em rolamentos do repositório de dados públicos fornecidos pela Case Western Reserve University (CWRU).

O modelo treinado realizou a detecção e diagnóstico de falhas no rolamento em estudo sob uma grande flutuação das condições operacionais (carga e velocidade do motor e localização e diâmetros de falhas no rolamento) atingindo uma precisão de 100% com os arquivos sendo treinados e testados nas mesmas condições operacionais e uma precisão de 97,7% quando testado com arquivos em condições operacionais diferentes das condições de treinamento. Quando comparado com outros métodos baseados em CNN, que utilizaram o mesmo banco de dados, o método proposto demonstrou superioridade ou foi pelo menos tão bem-sucedido quanto. Vale ressaltar que a superioridade foi adquirida mesmo com a CNN treinada com um pequeno conjunto de dados, representando 30% do conjunto de dados utilizado no estudo, e conseguindo generalizar, com alta precisão, para os demais arquivos de dados. No entanto, esse modelo ainda possui algumas limitações que precisam ser estudadas em trabalhos futuros, como por exemplo:

a) A abordagem proposta foi testada apenas em um sistema de avaliação da condição do rolamento do motor da CWRU. Considerando que a tendência de modelos de diagnóstico de falhas baseados em dados se aplica não apenas a máquinas rotativas específicas, mas também a muitos sistemas de máquinas rotativas diferentes, o potencial de aplicabilidade da abordagem proposta precisa ser avaliado em relação a um conjunto diversificado de dados de referência que inclua dados de uma variedade de sistemas de banco de ensaios. Portanto, a capacidade de generalização do modelo proposto ainda precisa ser mais estudada.

Em geral os resultados apresentados indicam que o modelo de diagnóstico de falhas de rolamento proposto, usando o TWC para posterior conversão em imagens, alcançou alta precisão sob diferentes condições de trabalho, podendo competir de forma justa com os modelos existentes de DDF em rolamentos.

Referências

ABDUL-NOUR, G. et al. A reliability based maintenance policy; a case study. Computers & industrial engineering, v. 35, n. 3-4, p. 591-594, 1998.

ABID, Anam; KHAN, Muhammad Tahir; IQBAL, Javaid. A review on fault detection and diagnosis techniques: basics and beyond. Artificial Intelligence Review, v. 54, n. 5, p. 3639-3664, 2021.

AHMED, Hosameldin; NANDI, Asoke K. Compressive sampling and feature ranking framework for bearing fault classification with vibration signals. IEEE Access, v. 6, p. 44731-44746, 2018.

AJIT, Arohan; ACHARYA, Koustav; SAMANTA, Abhishek. A review of convolutional neural networks. In: 2020 international conference on emerging trends in information technology and engineering (ic-ETITE). IEEE, 2020. p. 1-5.

ALBAWI, Saad; MOHAMMED, Tareq Abed; AL-ZAWI, Saad. Understanding of a convolutional neural network. In: 2017 international conference on engineering and technology (ICET). IEEE, 2017. p. 1-6.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 5462: Confiabilidade e Mantenabilidade. Rio de Janeiro, 1994.

ATTA, Mohamed Esam El-Dine; IBRAHIM, Doaa Khalil; GILANY, Mahmoud I. Detection and Diagnosis of Bearing Faults Under Fixed and Time-Varying Speed Conditions Using Persistence Spectrum and Multi-Scale Structural Similarity Index. IEEE Sensors Journal, v. 22, n. 3, p. 2637-2646, 2021.

AWADALLAH, Mohamed A.; MORCOS, Medhat M. Application of AI tools in fault diagnosis of electrical machines and drives-an overview. IEEE Transactions on energy conversion, v. 18, n. 2, p. 245-251, 2003.

BAZAN, Gustavo Henrique. Identificação inteligente de falhas em máquinas elétricas utilizando informação mútua. 2020. Tese (Doutorado em Engenharia Elétrica -Utpfr) - Universidade Tecnológica Federal do Paraná, Cornélio Procópio, 2020.

BEARD, Richard Vernon. Failure accomodation in linear systems through self-reorganization. 1971. Tese de Doutorado. Massachusetts Institute of Technology.

BHAUMIK, S. K. A view on the general practice in engineering failure analysis. Journal of failure analysis and prevention, v. 9, n. 3, p. 185-192, 2009.

BOUDIAF, Adel et al. A comparative study of various methods of bearing faults diagnosis using the case Western Reserve University data. Journal of Failure Analysis and Prevention, v. 16, n. 2, p. 271-284, 2016.

CASE SCHOOL OF ENGINEERING. Case Western Reserve University. Bearing Data Center: Seeded Fault Test Data. Disponível em: <https:// https://engineering.case.edu/bearingdatacenter/apparatus-and-procedures/>. Acesso em: 22 jul. 2022

CHEN, Chih-Cheng et al. An improved fault diagnosis using 1d-convolutional neural network model. Electronics, v. 10, n. 1, p. 59, 2020.

CHOW, E. Y. E. Y.; WILLSKY, Alan. Analytical redundancy and the design of robust failure detection systems. IEEE Transactions on automatic control, v. 29, n. 7, p. 603-614, 1984.

DAI, Juying et al. Signal-based intelligent hydraulic fault diagnosis methods: Review and prospects. Chinese Journal of Mechanical Engineering, v. 32, n. 1, p. 1-22, 2019.

DAI, Xuewu; GAO, Zhiwei. From model, signal to knowledge: A data-driven perspective of fault detection and diagnosis. IEEE Transactions on Industrial Informatics, v. 9, n. 4, p. 2226-2238, 2013.

DING, Xiaoxi; HE, Qingbo. Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis. IEEE Transactions on Instrumentation and Measurement, v. 66, n. 8, p. 1926-1935, 2017.

FANG, Yukun et al. A fault detection and diagnosis system for autonomous vehicles based on hybrid approaches. IEEE Sensors Journal, v. 20, n. 16, p. 9359-9371, 2020.

GAO, Zhiwei; CECATI, Carlo; DING, Steven X. A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches. IEEE transactions on industrial electronics, v. 62, n. 6, p. 3757-3767, 2015.

GAO, Zhiwei; CECATI, Carlo; DING, Steven X. A Survey of Fault Diagnosis and Fault-Tolerant Techniques—Part II: Fault Diagnosis With Knowledge-Based and Hybrid/Active Approaches. IEEE transactions on industrial electronics, v. 62, n. 6, p. 3768-3774, 2015b.

GERTLER, Janos J. Survey of model-based failure detection and isolation in complex plants. IEEE Control systems magazine, v. 8, n. 6, p. 3-11, 1988.

GLOROT, Xavier; BENGIO, Yoshua. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2010. p. 249-256

GUO, Sheng et al. Multitask convolutional neural network with information fusion for bearing fault diagnosis and localization. IEEE Transactions on Industrial Electronics, v. 67, n. 9, p. 8005-8015, 2019.

HARPER, John L. Problems of operation and maintenance of underground cables. Transactions of the American Institute of Electrical Engineers, v. 36, p. 417-422, 1917.

HUET, Roland. The interdisciplinary nature of failure analysis. Practical Failure Analysis, v. 2, n. 3, p. 17-19, 2002.

HUSSAIN, Muhammad Awais; TSAI, Tsung-Han. An efficient and fast softmax hardware architecture (EFSHA) for deep neural networks. In: 2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems (AICAS). IEEE, 2021. p. 1-4.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. IEC-60300-3-11: Guia de Aplicação – Manutenção Centrada em Confiabilidade – RCM. 2ed Switzerland, 2009.

IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pmlr, 2015. p. 448-456.

ISERMANN, Rolf; BALLE, Peter. Trends in the application of model-based fault detection and diagnosis of technical processes. Control engineering practice, v. 5, n. 5, p. 709-719, 1997.

ISERMANN, Rolf. Fault-diagnosis systems: an introduction from fault detection to fault tolerance. Springer Science & Business Media, 2005.

ISERMANN, Rolf. Supervision, fault-detection and fault-diagnosis methods—an introduction. Control engineering practice, v. 5, n. 5, p. 639-652, 1997.

JARDINE, Andrew KS; LIN, Daming; BANJEVIC, Dragan. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical systems and signal processing, v. 20, n. 7, p. 1483-1510, 2006.

KATTENBORN, Teja et al. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. ISPRS journal of photogrammetry and remote sensing, v. 173, p. 24-49, 2021.

KARN, Ujjwal. An Intuitive Explanation of Convolutional Neural Networks. Deep learning, computer vision, nlp, data science. 11 ago. 2016. Disponível em: <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>. Acesso em 25/01/2023.

KHAN, Samir; YAIRI, Takehisa. A review on the application of deep learning in system health management. Mechanical Systems and Signal Processing, v. 107, p. 241-265, 2018.

KHANAFER, Mounib; SHIRMOHAMMADI, Shervin. Applied AI in instrumentation and measurement: The deep learning revolution. IEEE Instrumentation & Measurement Magazine, v. 23, n. 6, p. 10-17, 2020.

KINGMA, Diederik P.; BA, Jimmy. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

LEE, Chun-Yao; LE, Truong-An. Identifying faults of rolling element based on persistence spectrum and convolutional neural network with ResNet structure. IEEE Access, v. 9, p. 78241-78252, 2021.

LECUN, Yann et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, v. 86, n. 11, p. 2278-2324, 1998.

LEI, Yaguo et al. An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data. IEEE Transactions on Industrial Electronics, v. 63, n. 5, p. 3137-3147, 2016.

LEI, Yaguo et al. Applications of machine learning to machine fault diagnosis: A review and roadmap. Mechanical Systems and Signal Processing, v. 138, p. 106587, 2020.

LI, Weihua; ZHANG, Shaohui; HE, Guolin. Semisupervised distance-preserving self-organizing map for machine-defect detection and classification. IEEE Transactions on Instrumentation and Measurement, v. 62, n. 5, p. 869-879, 2013.

LI, Zhengping; LIU, Kaiqiang; XIAO, Lei. Bearing Intelligent Fault Diagnosis Under Complex Working Condition Based on SK-ES-CNN. In: 2021 Global Reliability and Prognostics and Health Management (PHM-Nanjing). IEEE, 2021. p. 1-8.

LIU, Ruonan et al. Artificial intelligence for fault diagnosis of rotating machinery: A review. Mechanical Systems and Signal Processing, v. 108, p. 33-47, 2018.

LO, Ndeye Gueye; FLAUS, Jean-Marie; ADROT, Olivier. Review of machine learning approaches in fault diagnosis applied to IoT systems. In: 2019 International Conference on Control, Automation and Diagnosis (ICCAD). IEEE, 2019. p. 1-6.

MA, Jianping; JIANG, Jin. Applications of fault detection and diagnosis methods in nuclear power plants: A review. Progress in nuclear energy, v. 53, n. 3, p. 255-266, 2011.

MEYER, George; WEHREND, William R. NASA Ames active control aircraft flight experiments (ACA) program. Systems Reliability Issues**---**, p. 21, 1975.

MOOSAKUNJU, Sabna et al. A Hybrid Fault Detection and Diagnosis Algorithm for Five-Phase PMSM Drive. Arabian Journal for Science and Engineering, v. 48, n. 5, p. 6507-6519, 2023.

MOUBRAY, J. Manutenção Centrada em Confiabilidade (Reliability-Centered Maintenance – RCM). Trad. Kleber Siqueira. São Paulo: Aladon, 2000.

NASCIMENTO JR, Cairo Lúcio; YONEYAMA, Takashi. Inteligência artificial em controle e automação. Editora Edgard Blücher Ltda, 2000.

NEUPANE, Dhiraj; SEOK, Jongwon. Bearing fault detection and diagnosis using case western reserve university dataset with deep learning approaches: A review. IEEE Access, v. 8, p. 93155-93178, 2020.

NILSSON, Nils J. Probabilistic logic. Artificial intelligence, v. 28, n. 1, p. 71-87, 1986.

PENG, Zhi Ke; CHU, F. L. Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography. Mechanical systems and signal processing, v. 18, n. 2, p. 199-221, 2004.

PULE, Mushabi; MATSEBE, Oduetse; SAMIKANNU, Ravi. Application of PCA and SVM in Fault Detection and Diagnosis of Bearings with Varying Speed. Mathematical Problems in Engineering. v. 2022, 2022.

PYUN, Hahyung et al. Root causality analysis at early abnormal stage using principal component analysis and multivariate Granger causality. Process Safety and Environmental Protection, v. 135, p. 113-125, 2020.

QIN, Kai et al. Root cause analysis of industrial faults based on binary extreme gradient boosting and temporal causal discovery network. Chemometrics and Intelligent Laboratory Systems. v. 225, p. 104559, 2022.

ROSEBROCK, Adrian. Convolutional Neural Networks (CNNs) and Layer Types. PyImageSearch - You can master Computer Vision, Deep Learning, and OpenCV. 14 may. 2021. Disponível em: < Convolutional Neural Networks (CNNs) and Layer Types - PyImageSearch>. Acesso em 25/01/2023.

RUSSAKOVSKY, Olga et al. Imagenet large scale visual recognition challenge. International journal of computer vision, v. 115, n. 3, p. 211-252, 2015.

SAE-JA1011.Evaluation Criteria for Reliability-Centered Maintenance (RCM) Processes. Society of Automotive Engineers, Agosto1999.

SAKURADA, Eduardo Yuji. As técnicas de Análise do Modos de Falhas e seus Efeitos e Análise da Árvore de Falhas no desenvolvimento e na avaliação de produtos. Florianópolis: Eng. Mecânica/UFSC, (Dissertação de mestrado), 2001.

SHUI, Aishe et al. Review of fault diagnosis in control systems. In: 2009 Chinese Control and Decision Conference. IEEE, 2009. p. 5324-5329.

SIDDIQUE, Arfat; YADAVA, G. S.; SINGH, Bhim. Applications of artificial intelligence techniques for induction machine stator fault diagnostics. In: 4th IEEE

International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, 2003. SDEMPED 2003**.** IEEE, 2003. p. 29-34.

SIQUEIRA, Y. P. D. S. Manutenção centrada na confiabilidade: manual de Implantação. 1ª (Reimpressão). ed. Rio de Janeiro: Qualitymark, 2009.

SIKDER, Niloy et al. Induction motor bearing fault classification using extreme learning machine based on power features. Arabian Journal for Science and Engineering, v. 46, n. 9, p. 8475-8491, 2021.

SLAMANI, Mustapha; KAMINSKA, Bozena. Fault observability analysis of analog circuits in frequency domain. IEEE transactions on circuits and systems II: Analog and digital signal processing, v. 43, n. 2, p. 134-139, 1996.

SMITH, Wade A.; RANDALL, Robert B. Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. Mechanical systems and signal processing, v. 64, p. 100-131, 2015.

SOUALHI, Abdenour; MEDJAHER, Kamal; ZERHOUNI, Noureddine. Bearing health monitoring based on Hilbert–Huang transform, support vector machine, and regression. IEEE Transactions on instrumentation and measurement, v. 64, n. 1, p. 52-62, 2014.

SREEJITH, B.; VERMA, Ajit Kumar; SRIVIDYA, A. Fault diagnosis of rolling element bearing using time-domain features and neural networks. In: 2008 IEEE region 10 and the third international conference on industrial and information systems. IEEE, 2008. p. 1-6.

SRIVASTAVA, Nitish et al. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, v. 15, n. 1, p. 1929-1958, 2014.

SRINIVAS, Suraj et al. A taxonomy of deep convolutional neural nets for computer vision. Frontiers in Robotics and AI, v. 2, p. 36, 2016.

THE MATHWORKS, Inc. Help Center. List of Deep Learning Layers. Disponível em: <https://www.Mathworks.com/help/deeplearning/ug/list-of-deep-learning-layers.html>. Acesso em 02/01/2023.

THE MATHWORKS, Inc. Help Center. TrainingOptions. Disponível em: <https://www.Mathworks.com/help/deeplearning/ref/trainingoptions.html>. Acesso em 02/01/2023.

TAYYAB, Syed Muhammad; CHATTERTON, Steven; PENNACCHI, Paolo. Intelligent Defect Diagnosis of Rolling Element Bearings under Variable Operating Conditions Using Convolutional Neural Network and Order Maps. Sensors, v. 22, n. 5, p. 2026, 2022.

VALE, Marcelo Roberto Bastos Guerra. Sistema híbrido para detecção e diagnóstico de falhas em sistemas dinâmicos. 2014. 86 f. Tese (Doutorado em

Automação e Sistemas; Engenharia de Computação; Telecomunicações) - Universidade Federal do Rio Grande do Norte, Natal, 2014.

VENKATASUBRAMANIAN, Venkat et al. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. Computers & chemical engineering, v. 27, n. 3, p. 293-311, 2003.

VERSTRAETE, David et al. Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings. Shock and Vibration, v. 2017, 2017.

VICENTE, Silmara Alexandra da Silva; FUJIMOTO, Rodrigo Yoshiaki; PADOVESE, Linilson Rodrigues. Rolling bearing fault diagnostic system using fuzzy logic. 10th IEEE International Conference on Fuzzy Systems.(Cat. No. 01CH37297). IEEE, 2001. p. 816-819.

WANG, Bin et al. Automatic fault diagnosis of infrared insulator images based on image instance segmentation and temperature analysis. IEEE Transactions on Instrumentation and Measurement, v. 69, n. 8, p. 5345-5355, 2020.

WANG, Jianyu et al. A deep learning method for bearing fault diagnosis based on time-frequency image. IEEE Access, v. 7, p. 42373-42383, 2019.

WANG, Jinjiang et al. A multi-scale convolution neural network for featureless fault diagnosis. In: 2016 International symposium on flexible automation (ISFA). IEEE, 2016. p. 65-70.

WILLSKY, Alan S. A survey of design methods for failure detection in dynamic systems. Automatica, v. 12, n. 6, p. 601-611, 1976.

XU, Gaowei et al. Bearing fault diagnosis method based on deep convolutional neural network and random forest ensemble learning. Sensors, v. 19, n. 5, p. 1088, 2019

XU, Yan et al. Industrial big data for fault diagnosis: Taxonomy, review, and applications. IEEE Access, v. 5, p. 17368-17380, 2017.

YUAN, Laohu et al. Rolling bearing fault diagnosis based on convolutional neural network and support vector machine. IEEE Access, v. 8, p. 137395-137406, 2020.

ZHOU, Bolei et al. Learning deep features for scene recognition using places database. Advances in neural information processing systems, v. 27, 2014.