

Classificação de Defeitos em Rolamento de Motores Elétricos Utilizando Redes Convolucionais Compactas para Sistemas Embarcados

Caio Marcos Malheiros Alves*, Luiz Alberto Pinto*, Marco Antônio de Souza Leite Cuadros*, Gustavo Maia de Almeida*

*Programa de Pós-graduação em Engenharia de Controle e Automação – Propecaut, Instituto Federal do Espírito Santo, Serra - Brasil; e-mail: (caiomarcos@gmail.com, pintoluizalberto@gmail.com, marcoantonio@ifes.edu.br, gmaia@ifes.edu.br)

Abstract - Electrical motors are ubiquitous in industrial process and in daily activities, therefore, fault detection is especially important. Recent fault detection methods rely on CNNs with great success, but those are run on off-site computers. In this work we will analyze two proposed CNNs that are designed to be run on-site, in a constrained embedded system. Those CNNs have been successfully validated on a proprietary dataset by its developers. Now, we will validate the proposed CNNs using the CWRU data set, a dataset often used as a benchmark for bearing fault detection. We will show that the proposed CNNs are also successful at classifying the CWRU, proving to be versatile and therefore a good bearing fault detection option to be used in different set-ups.

Keywords: Convolutional neural networks; embedded systems; bearing fault detection; vibration analysis, electrical motors.

Resumo – Motores elétricos são onipresentes em processos industriais e no dia a dia e assim o diagnóstico de defeitos em tais máquinas é de suma importância. As técnicas mais recentes de diagnóstico utilizam *CNN* com grande sucesso, porém são implementadas para rodar em servidores *off-site*. Este trabalho analisa duas propostas de *CNNs* desenvolvidas para serem executadas em um sistema embarcado on-site. As *CNNs* foram testadas com sucesso por seus autores utilizando um conjunto de dados proprietário. Os testes realizados aqui utilizam um conjunto de dados diferente, muito utilizado em benchmarks na área, o conjunto de dados da *CWRU*. Os testes realizados demonstram que as *CNNs* propostas também obtêm sucesso ao classificar defeitos do conjunto de dados *CWRU*, validando suas estruturas e demonstrando uma boa generalização para ser aplicada em montagens e sistemas diferentes do utilizado em seu desenvolvimento.

Palavras-chaves: Redes neurais convolucionais; sistemas embarcados; detecção de falhas em rolamentos; análise de vibração, motores elétricos.

1. INTRODUÇÃO

Motores elétricos são responsáveis por cerca de 40% de todo o consumo de energia elétrica no mundo (APEC Energy Working Group 2008). Suas boas condições de funcionamento dependem, grandemente, das boas condições de conservação dos mancais de rolamento. Os mancais de rolamento têm como função dar suporte mecânico às partes rotativas, reduzir o coeficiente de atrito do movimento e garantir a precisão da rotação. Portanto, uma vez que exista um defeito no rolamento, a estabilidade de todo o conjunto pode ser afetada, provocando danos ao equipamento e condições inseguras. Estima-se que defeitos em rolamentos sejam responsáveis por cerca de 69% do total de falhas em motores elétricos. (Bonnett et al. 2008)

O monitoramento contínuo do funcionamento de motores é uma das formas de garantir a continuidade dos processos de produção, reduzir custos provocados por interrupções não programadas e evitar acidentes. Com esse propósito, as indústrias têm implantado sistemas de detecção e diagnósticos

de falhas em rolamentos de motores elétricos. As abordagens atuais mais comumente usadas para detecção e o diagnóstico de falhas em rolamentos, consistem na análise dos sinais de vibração, e na classificação desses por ferramentas de *Machine Learning (ML)*. Pesquisas mais recentes propõem a predição de falhas iminentes, estimativa do estado de conservação e por último, a estimativa da vida útil de rolamentos (*RUL – Rest of Useful Life*).

Uma variedade de abordagens para elaboração de diagnósticos inteligentes de falhas em rolamentos de motores elétricos tem sido proposta na literatura. Por exemplo, Redes Neurais Artificiais (RNA), Redes Neurais Convolucionais (*CNN* do inglês *Convolutional Neural Network*), processamento de sinais de vibração utilizando a Transformada *Wavelet*, e aplicação de outras arquiteturas de redes profundas. Esses métodos têm atingido altos níveis de precisão na identificação de defeitos e na previsão da *RUL*. Porém, trazem também custos computacionais cada vez mais elevados, para ganhos cada vez mais marginais em precisão.

As principais etapas para elaboração de diagnósticos de falhas são: 1. aquisição dos sinais do processo; 2. processamento dos sinais e 3. classificação (elaboração do diagnóstico). Tradicionalmente estas etapas são realizadas em modo *off-line* e remoto. Nesse caso, a etapa de aquisição dos dados não está conectada com as outras duas. A aquisição é realizada *on-site*, no motor, e posteriormente em um computador *off-site* as etapas de processamento e diagnóstico são executadas. Diferentemente dos sistemas *off-line* remotos, nos sistemas *on-site* as etapas de aquisição, processamento e classificação ocorrem de forma contínua, no equipamento, permitindo a elaboração de diagnósticos em tempo real, o que possibilita intervenções imediatas para mitigar os efeitos das falhas.

Com a difusão do conceito de *IIoT (Industrial Internet of Things)* e de Indústria 4.0, pode-se especular sobre o aumento da utilização de sistemas embarcados microcontrolados e *SBC (Single Board Computers)*. Além disso, paralelamente observou-se significantes avanços em conectividade, eficiência energética e capacidade de processamento e de memória dos hardwares para sistemas embarcados disponíveis atualmente. Embora sistemas embarcados já sejam uma realidade em várias áreas da tecnologia, são raros os trabalhos sobre diagnóstico de falhas desenvolvidos para tais sistemas operando *on-site*, devido a ainda recente convergência da evolução de classificadores e melhorias da capacidade computacional de microcontroladores.

Nesse contexto, esse trabalho avaliou duas propostas de *CNNs* desenvolvidas para serem utilizadas em dispositivos com limitação de recursos (Lu et al. 2019), como *SBCs* ou sistemas microcontrolados. A avaliação foi feita utilizando o conjunto de dados da *Case Western Reserve University Bearing Data Center (CWRU)*, amplamente utilizado como *benchmark* em aplicações semelhantes (Smith et al. 2015). Esse estudo demonstrou a viabilidade das arquiteturas das *CNNs* como classificadores de falhas em rolamentos, confirmando suas capacidades de aprendizado e portabilidade para funcionamento em diferentes sistemas, a fim de utilizá-las como classificadores em sistemas microcontrolados.

O restante deste trabalho contém as seguintes seções: a Seção 2 apresenta uma breve revisão da literatura sobre o tema. Na Seção 3 mostra-se a teoria das *CNNs*. Em seguida, na Seção 4, se apresenta o conjunto de dados *CWRU* utilizado neste trabalho. Na Seção 5 descreve-se o método “Espectro Logarítmico de Base 2” que será utilizado em seções posteriores. Na Seção 6 apresentam-se os dois modelos de *CNN* que serão avaliados. Na Seção 7 os processos de treinamento e classificação são descritos. Finalmente em 8 temos as conclusões acerca do trabalho feito.

2. TRABALHOS RELACIONADOS

2.1 Detecção de Falhas

Detecção de falhas em rolamentos de motores utilizando dados de vibração é uma prática estabelecida e estudada na literatura há quase 60 anos. Já no começo da década de 60, Gustafsson et al. (1962) apresentam um método não-destrutivo capaz de detectar danos na superfície dos elementos danificados utilizando a característica “quase periódicas” das vibrações

produzidas ao rotacionar um rolamento danificado. Desde então, ferramentas estatísticas e de processamento de sinais, juntamente com modelagem matemática, são amplamente estudadas e utilizadas para identificação destes defeitos. Esses métodos de diagnóstico consistem em identificar frequências fundamentais e as harmônicas geradas pelos impactos causados entre os elementos defeituosos do rolamento. Randall et al. (2011) apresentam um extenso comparativo entre diferentes métodos utilizando análises do sinal de vibração no tempo, na frequência, e no tempo-frequência como, mas não somente, curtose, RMS, modulação, envelope, transformada de Fourier, transformada de Fourier janelada, e transformada *wavelet*.

Para superar a dificuldade que é modelar um sistema de rolamento, seus defeitos e todas variáveis que podem influenciar seus padrões de vibração, já em 1991 foi proposta (Chow et al. 1991) a utilização de redes neurais artificiais para a detecção destas falhas, pois dessa forma não seria mais necessário modelar equações complexas não-lineares. Toda a modelagem matemática rigorosa e os erros de modelagem que a acompanhavam poderiam ser evitadas. A não-linearidade desejada nesses modelos já seria intrínseca às redes neurais. Naturalmente, diversas técnicas e métodos de aprendizado de máquina foram sendo desenvolvidos e aplicados na detecção de falhas em rolamentos.

Em (Malhi et al. 2004), *PCA (Principal Component Analysis)* foi utilizada com sucesso, melhorando o desempenho das redes neurais classificadoras. Em (Pandarakone et al. 2018) *SVM (Support Vector Machine)* foi utilizada para classificar tipos de arranhões presentes nos elementos do rolamento, usando como parâmetro as componentes de frequência da corrente elétrica. Em (Tian et al. 2016), *PCA* e curtose espectral do sinal de vibração são utilizadas em conjunto com vizinho mais próximo (*KNN*) para desenvolver um método de detecção e diagnóstico de falhas em rolamentos de motores elétricos.

Mais recentemente houve uma explosão de trabalhos utilizando aprendizado profundo para detectar e classificar falhas em motores e seus rolamentos. O primeiro trabalho utilizando *CNN* foi em 2016 (Janssens et al. 2016), tendo os autores alcançado acurácia e precisão >93% utilizando uma *CNN* com uma camada convolucional e uma camada totalmente conectada. Outros trabalhos pioneiros, como (Wen et al. 2018) utilizam redes baseadas na *LeNet-5* para classificar com sucesso o sinal original de vibração.

2.1 Sistemas Embarcados

Em paralelo ao desenvolvimento de técnicas de inteligência artificial, a capacidade computacional de sistemas embarcados microcontrolados tem avançado rapidamente nos últimos anos. Assim, começam a surgir alguns trabalhos nesta interseção entre detecção de anomalias e sistemas embarcados, porém ainda são poucos em comparação com o desenvolvimento de soluções propostas para desktops e servidores. Em (Tzeng et al. 2018) se apresenta um sistema embarcado para captura e pré-processamento de dados de vibração, que então são enviados por uma rede sem fio e monitorados remotamente. Em (Hmida et al. 2016) é implementado um algoritmo de monitoramento de motores,

através de sinais de vibração e transformadas *wavelet*, embarcado em um microcontrolador.

Recentemente surgem trabalhos apresentando métodos de *ML* que são implementados em sistemas microcontrolados para classificação de falhas em motores elétricos. Nestes casos, o sistema embarcado realiza a coleta do sinal, pré-processamento dos dados e classificação das amostras, utilizando um classificador treinado *off-site* em um computador e então embarcando no sistema microcontrolado.

Em (Gongora et al. 2018), uma RNA embarcada é desenvolvida para detectar falhas no rolamento através do sinal da corrente elétrica. Em (Eren et al. 2019) os autores apresentam redes neurais convolucionais compactas capazes de classificar falhas em rolamentos através de sinais de vibração, se utilizando de conjuntos de dados públicos (como o *CWRU*) para validar o trabalho. Em (Markiewicz et al. 2019) os autores apresentam uma solução embarcada completa de baixo consumo para classificação de falhas utilizando redes *Long Short-Term Memory*.

Em (Lu et al. 2019) os autores classificam 7 tipos de falha, além do funcionamento normal, em motores de indução. Para tal, se utilizam de 8 motores, cada um em uma condição de falha ou funcionamento normal. Dados de vibração são adquiridos a uma taxa de 20 *ksps* e três testes de classificação são realizados em um sistema microcontrolado. No primeiro teste, imagens 28 x 28 são geradas a partir dos sinais de vibração e uma *CNN* com duas camadas convolucionais é usada para a classificação. No segundo os autores usam as mesmas imagens 28 x 28 e uma *CNN* com uma camada convolucional para classificação, e no terceiro teste geram imagens 14 x 14 e as classificam utilizando a *CNN* com duas camadas convolucionais. As acurácias alcançadas são 100%, 97,5% e 94,13% para cada teste, respectivamente.

3. REFERENCIAL TEÓRICO

Constituídas por camadas com funções diferentes, as Redes Neurais Convolucionais (*Convolutional Neural Networks – CNN*) são variações das redes *perceptron*, que utilizam filtros convolucionais para a extração de características das imagens em diferentes etapas de processamento. Além das camadas de entrada e de saída, a arquitetura de uma rede neural convolucional é constituída por camadas convolucionais, camadas de *pooling*, camada totalmente conectada e camada de normalização em lote. A Fig. 1 mostra a arquitetura básica de uma *CNN* com as diferentes camadas que a constituem.

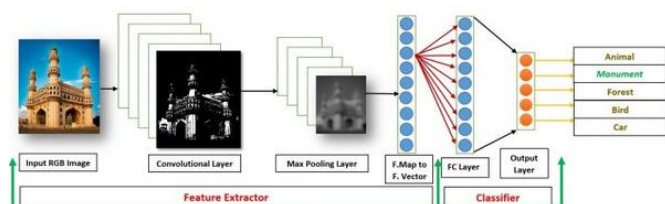


Fig. 1: Arquitetura Básica de uma Rede Neural Convolucional. Disponível em: <https://www.serpro.gov.br/>. Acesso em jun. 2020.

Camada Convolucional: constituída por um conjunto de filtros, a camada convolucional extrai as características da imagem de entrada. Uma *CNN* possui várias camadas convolucionais que são capazes de extrair informações em níveis, sucessivamente, mais baixos de detalhamento. A Fig. 2 apresenta uma camada de convolução com um filtro (kernel) 3 x 3. A configuração dos parâmetros dessas camadas requer a definição da quantidade de filtros, do *stride*, que é a quantidade de saltos do filtro ao se deslocar, e do *padding* que define como será o tratamento das bordas da imagem na realização da convolução. Além desses parâmetros, para a implementação de uma *CNN* é necessário definir uma função de ativação. Pela facilidade de treinamento e pelos bons resultados obtidos pelas redes, a *ReLU* tem sido a função de ativação mais utilizada em tarefas que utilizam as *CNNs*.

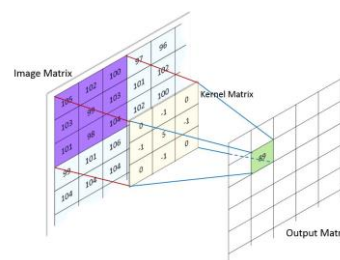


Fig. 2: Camada convolucional de uma *CNN* com um filtro 3x3. Disponível em: <https://www.serpro.gov.br/>. Acesso em jun. 2020.

Camada de *Pooling*: geralmente localizada após uma camada de convolução, a principal função da camada de *pooling* é a redução da dimensão espacial da imagem de entrada para a camada convolucional seguinte. Na prática, essa camada realiza uma subamostragem dividindo a entrada em regiões retangulares menores, e calculando o valor correspondente de cada região. Esta operação reduz o tamanho do mapa de características removendo informações de redundância espacial. Uma das principais vantagens da utilização da camada de *pooling* é a possibilidade de aumentar o número de filtros de convolução nas camadas mais profundas sem que ocorra um aumento considerável do esforço computacional. A estratégia mais comumente utilizada para a implementação do *pooling* é o *Max Pooling* (Fig. 3). Nesse caso, no processo de subamostragem será considerado o maior valor da região.

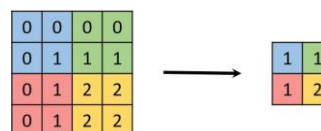


Fig. 3: A realização do *Max Pooling* da camada *pooling*. Disponível em: <https://www.serpro.gov.br/>. Acesso em jun. 2020.

Camada de Normalização em Lote: a camada de normalização em lote é utilizada para reduzir a faixa de variação dos valores das unidades das camadas ocultas, o que resulta na aceleração do treinamento da rede, na redução da sua sensibilidade aos pesos iniciais, além de possibilitar que cada camada da rede aprenda de forma mais independente das demais.

Camada Totalmente Conectada: a camada totalmente conectada é colocada antes da camada de saída da rede. Como todos os neurônios dessa camada são conectados a todos os neurônios da camada anterior, todas as características aprendidas das camadas anteriores são transferidas para o mapa de características. Em problemas de classificação, a última camada totalmente conectada combina as características para a classificação, dessa forma, o tamanho da saída da última camada totalmente conectada é igual ao número de classes.

Camada de Classificação: a camada de classificação é a camada de saída da rede, e deve vir após a última camada totalmente conectada. Em classificação a função de ativação *Softmax* é geralmente utilizada nessa camada, normalizando a saída da camada anterior. Sua saída consiste em números positivos cuja soma é igual a 1, que correspondem às probabilidades das classes.

4. O CONJUNTO DE DADOS CWRU

Este conjunto de dados elaborado pela *Case Western Reserve University* é um dos mais utilizados para verificar métodos de classificação de falhas em rolamentos de motores. A montagem utilizada para sua aquisição é mostrada na Fig. 4. Nela, um motor de potência nominal de 2 hp aciona um dinamômetro acoplados por um transdutor/encoder de torque.

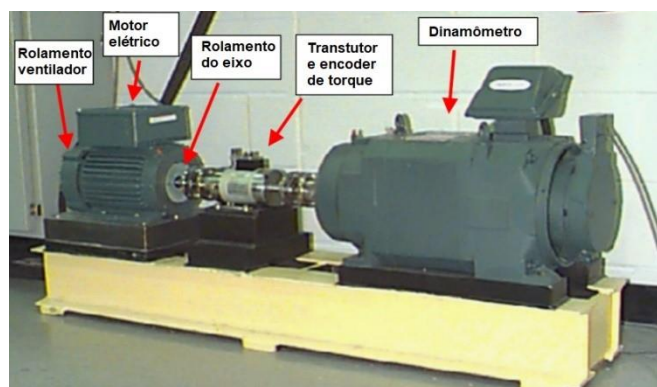


Fig. 4: Montagem para aquisição do conjunto de dados CWRU. Disponível em <https://csegroups.case.edu/bearingdatacenter>. Acesso em jun. 2020.

Falhas pontuais provocadas por descargas elétricas foram artificialmente introduzidas nos elementos dos rolamentos que suportam o eixo. Ao todo foram introduzidas cinco falhas com diâmetros de 7 mils (0,18 mm), 14 mils (0,36 mm), 21 mils (0,53 mm) e 28 mils (0,71 mm), que foram localizadas na esfera, na parte interna do rolamento, e em três posições diferentes do rolamento externo, pois são estacionárias. Os sinais de vibração foram coletados para 4 velocidades de rotação – 1797, 1772, 1750 (nominal) e 1730 rpm. Para tal, acelerômetros foram posicionados no lado acoplado (lado do eixo) do motor e no lado do ventilador do motor. As amostras foram coletadas em 12 *ksp*s (*kilo samples per second*). O conjunto de dados é público e está disponível na internet no site da CWRU. Para este estudo, foi escolhido analisar o defeito com tamanho de 21 mils (0,53 mm), nas 4 condições

de carga, com dados coletados a 12 *ksp*s no lado do eixo do motor.

Como existe um sinal de, aproximadamente, 10 segundos para cada velocidade do motor (1797, 1772, 1750 e 1730 rpm), para cada posição do defeito (normal, esfera, rolamento interno, rolamento externo oposto, rolamento externo ortogonal, e rolamento externo centralizado), temos um total de 24 sinais que podem ser agrupados de diversas maneiras. Para este trabalho, os sinais foram agrupados de acordo com seus defeitos, independente da velocidade de rotação do motor. Assim, após o agrupamento dos sinais obteve-se o total de seis classes, que são, RE oposto, RE centralizado, RE ortogonal, RI, Esfera e Normal correspondentes a C1, C2, C3, C4, C5 e C6 respectivamente.

A Fig. 5 mostra os gráficos dos sinais de aceleração dos conjuntos de dados utilizados, por um período de aproximadamente 0,5 segundo, ou 6000 amostras em uma frequência de amostragem de 12 *ksp*s.

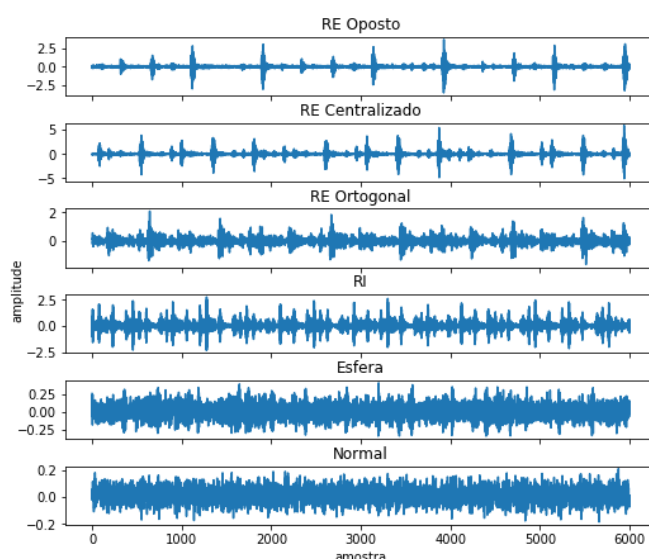


Fig. 5: Exemplos de sinais de vibração do CWRU – 1750 rpm, 12 *ksp*s. Fonte: autor.

5. ESPECTRO LOGARÍTIMICO DE BASE 2

Redes neurais convolucionais, em geral, têm como entrada imagens em duas dimensões, porém, sinais de vibração são unidimensionais, como é o caso dos sinais do conjunto de dados CWRU. Dessa forma, é necessário tratar estes sinais antes de utilizá-los no treinamento e classificação com a CNN.

O espectro logarítmico de base dois (B2LS) do sinal de vibração é utilizado para construir a imagem de entrada da CNN. Se o sinal original fosse utilizado, a imagem seria dominada pelo componente de frequência de maior potência, dificultando o reconhecimento das componentes de frequência de menor potência. Assim, o B2LS diminui as diferenças de amplitude entre os diferentes componentes de frequência, melhorando o contraste e acurácia da classificação (Lu et al, 2019).

Sendo assim, supondo que a imagem de entrada da *CNN* tenha uma resolução de $M \times N$, o sinal de vibração correspondente é dado por

$$V[n], \quad n = 1, 2, \dots, 2MN \quad (1)$$

O B2LS de $V[n]$ é obtido por

$$S[i] = \log_2 \left(\frac{FFT(V[n])}{MN} \right), \quad i = 1, 2, \dots, 2MN \quad (2)$$

onde $FFT(\cdot)$ é a transformada rápida de Fourier. Considerando que $S[i]$ é simétrico, a imagem é construída da forma

$$I = \frac{\begin{pmatrix} S[1] & \dots & S[N] \\ \vdots & \ddots & \vdots \\ S[N(M-1)+1] & \dots & S[MN] \end{pmatrix}}{\max(S[i])} \quad (3)$$

onde $\max(S[i])$ é o maior valor entre todos componentes da B2LS.

6. CNN CLASSIFICADORAS

Para classificar os sinais apresentados na Seção 4, dois modelos de *CNN* foram testados. Ambos foram propostos como modelos apropriados para serem utilizados em sistemas restritos, portanto são relativamente simples, com poucas camadas e parâmetros. Os dois modelos foram validados com sucesso utilizando um conjunto de dados proprietário dos autores que propuseram tais redes convolucionais (Lu et al. 2019). As duas foram testadas aqui utilizando o conjunto de dados *CWRU*, com o objetivo de demonstrar a capacidade de generalização e sua validade ao classificar dados similares, porém não exatamente os mesmos para os quais foram projetadas inicialmente.

Tabela 1. CNN Modelo 1

Parâmetros CNN modelo 1	
Camada	Parâmetros
1a convolução	Kernel: 3x3; Qtd. Kernel: 16
1a max-pooling	Kernel: 3x3; Stride: 2x2
2a convolução	Kernel: 3x3; Qtd. Kernel: 32
2a max-pooling	Kernel: 3x3; Stride: 2x2
Densa	128
Saída	6
Parâmetros totais	71.238

Tabela 2. CNN Modelo 2

Parâmetros CNN modelo 2	
Camada	Parâmetros
1a convolução	Kernel: 2x2; Qtd. Kernel: 2
1a max-pooling	Kernel: 3x3; Stride: 2x2
Densa	8
Saída	6
Parâmetros totais	2.776

7. PRÉ-PROCESSAMENTO, TREINAMENTO E CLASSIFICAÇÃO

7.1 Pré-processamento

Os sinais originais do *CWRU* são compostos por cerca de 120000 pontos (10 segundos em 12 *ksp/s*), porém, não exatamente. Como exemplo, o experimento com defeito de 53 mm na esfera com motor a 1772 rpm gerou um sinal com 121701 pontos, e o experimento com o mesmo defeito com o motor a 1750 rpm gerou um sinal com 122136 pontos. Portanto o primeiro tratamento feito sobre o sinal foi igualar todos com o mesmo número de pontos em 120000. Assim, cada classe (C1 a C6), terá conjunto de 4 sinais (as 4 velocidades do motor, de 1792 a 1730 rpm), cada um com 120000 pontos.

Em seguida, foram aplicados nesses sinais os passos apresentados na Seção 5. Para cada sinal de 120000 pontos, pegamos amostras de acordo com (1), com $M=N=28$, sem sobreposição. Portanto, cada sinal nos permitiu retirar 76 amostras de 1568 pontos. Nessas amostras aplicamos (2), obtendo um conjunto de $M \times N$ pontos representando a *B2LS* da amostra. Por fim aplicamos (3) para obter a amostra em formato de imagem 28×28 pixels que foi utilizada para treinamento e inferência nas *CNNs*. A Fig. 6 ilustra as transformações do sinal original até obtermos a imagem. A Fig. 7 mostra um exemplo de amostra para cada classe. Este procedimento nos dá ao final um total de 304 amostras para cada classe de C1 a C6.

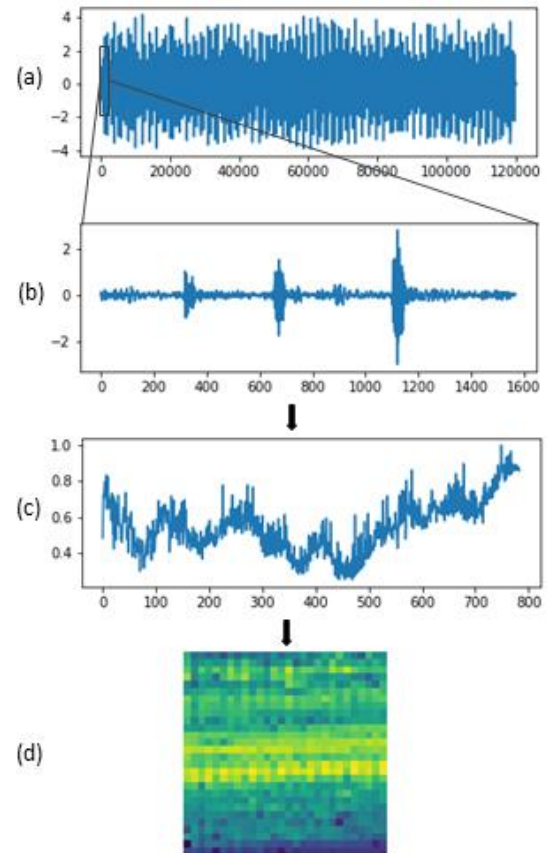


Fig. 6: Transformação do sinal original em amostras para classificação. (a) sinal original do *CWRU*, (b) amostra de 1568 pontos, (c) *B2LS* da amostra e (d) imagem de 28×28 pixels para treinamento e classificação. Fonte: autor.

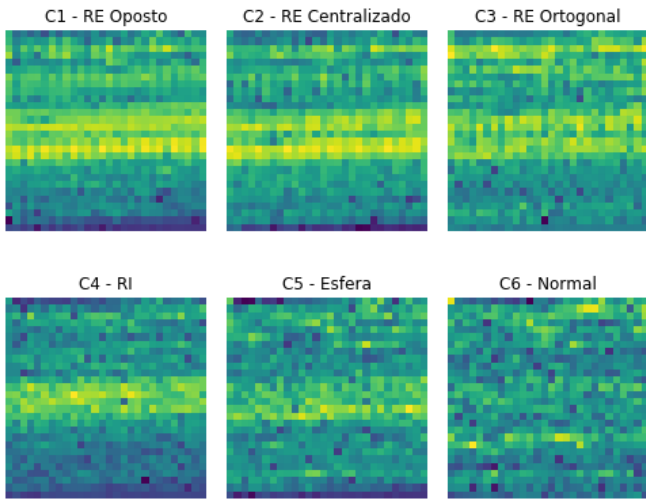


Fig. 7: Exemplos de imagens utilizadas para treinamento e classificação de suas respectivas classes. Fonte: autor.

7.2 Treinamento

Com as 304 amostras de cada classe, as *CNNs* propostas na Seção 6 foram treinadas. Foram executadas 10 rodadas de treinamento/teste, cada uma delas utilizando método *hold-out* onde as 1824 amostras do conjunto foram particionadas aleatoriamente em conjuntos de treino, com 70% das amostras totais (1276), e teste, com as 30% restantes. Foi utilizado batches de 10 amostras, durante 100 épocas, para cada rodada de treinamento.

A Fig. 8 mostra o gráfico de acurácia de treinamento da *CNN* Modelo 1, durante as 100 épocas em das 10 rodadas de treino (rodada #2). O mesmo comportamento foi observado nas outras 9 rodadas, isto é, a acurácia rapidamente se aproxima de 1 e se estabiliza.

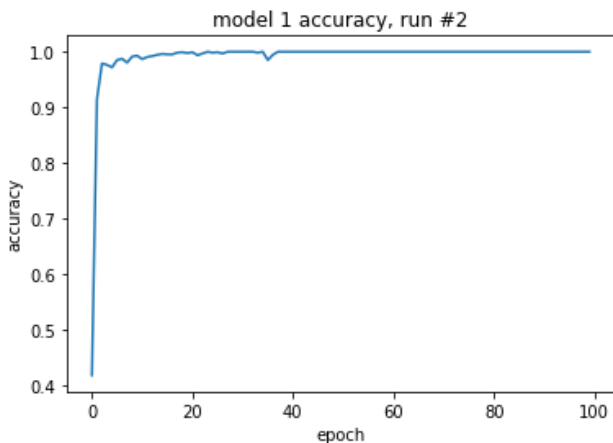


Fig. 8 Acurácia: evolução da acurácia da *CNN* Modelo 1 durante treinamento. Fonte: autor.

O mesmo não ocorreu com a *CNN* Modelo 2. Devido a sua arquitetura mais simples, o treinamento se mostrou instável, em alguns casos obtendo resultados não melhor que uma classificação aleatória de aproximadamente 17%. É possível que seja um caso de *vanishing gradient* e, devido ao baixo número de neurônios e filtros, uma incapacidade de progredir

o aprendizado. Nos outros casos a acurácia de treinamento se estabilizava próximo de 1, como no Modelo 1. A Fig. 9 ilustra três exemplos de curvas de acurácias durante o treinamento da *CNN* Modelo 2 (rodada #6, #7 e #8). Fica claro as diferentes evoluções de aprendizado da mesma rede para valores de inicialização e *hold-out* diferentes.

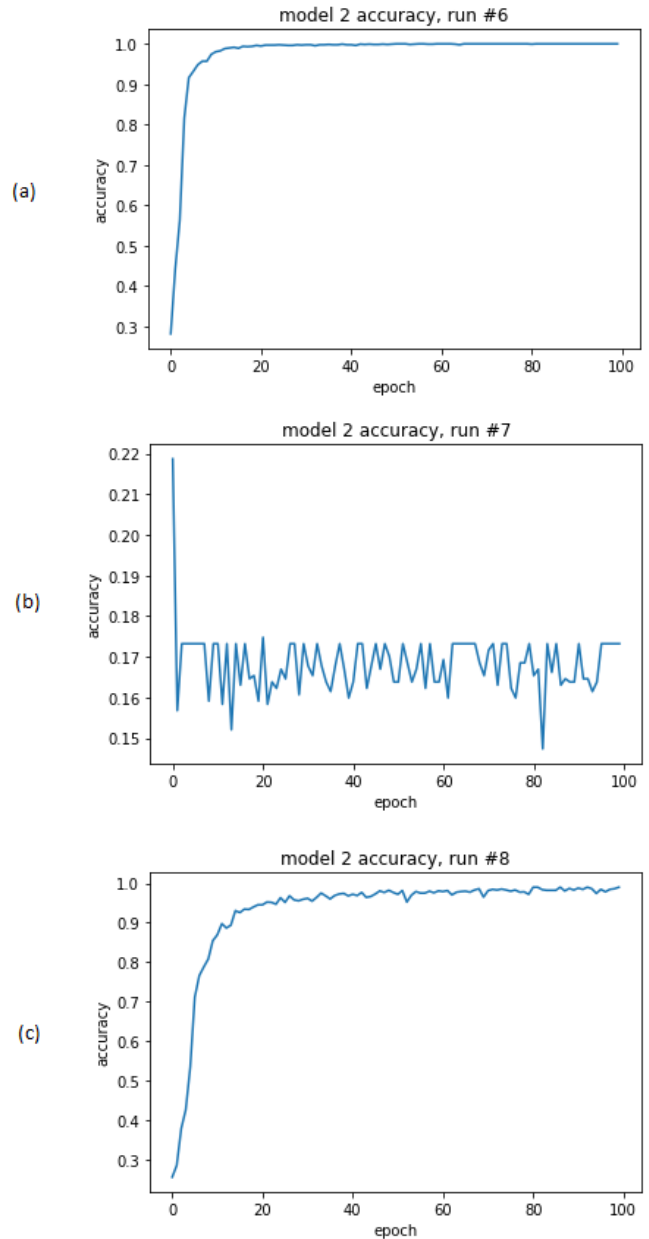


Fig. 9: Acurácia de treinamento da *CNN* Modelo 2, para 3 casos diferentes. (a) convergiu rapidamente, (b) não convergiu e (c) convergiu lentamente. Fonte: autor.

7.3 Classificação

A *CNN* Modelo 1 obteve ótimo resultado em classificar as 6 classes de condição do rolamento. A Tabela 3 mostra a matriz confusão combinada dos 10 testes realizados. A Tabela 4 traz as métricas de classificação alcançadas pelo Modelo 1 nos 10 testes, onde a menor acurácia foi de 99,27% e média de 99,7%.

Tabela 3. Matriz confusão combinada – CNN modelo 1

		Condição prevista					
		C1	C2	C3	C4	C5	C6
Condição real	C1	923	0	0	0	0	0
	C2	0	880	0	0	0	0
	C3	0	0	952	0	0	0
	C4	0	0	0	931	0	0
	C5	0	0	0	0	889	6
	C6	0	0	0	0	9	889

Tabela 4. Métricas modelo 1 (%)

	Média	Máxima	Mínima
Acurácia	99,70	100	99,27
Precisão	99,73	100	99,45
Recall	99,70	100	99,27
F1	99,72	100	99,36

A CNN Modelo 2 não obteve resultados satisfatórios quando se observa o conjunto de 10 treinamentos, mostrados na Tabela 6. Este resultado médio baixo se deve a alguns casos de não convergência da CNN observados durante o processo de treinamento, talvez indicando um limiar inferior de complexidade da CNN para que ela seja capaz de classificar os tipos de amostras aqui estudadas. Porém, analisando casos isolados entre todos as rodadas de treinamento realizadas, notamos que quando o treinamento obtém sucesso, os resultados são tão bons quanto do modelo 1, como se vê por exemplo na matriz de confusão do treinamento/teste #6, na Tabela 5, e nos valores máximos das métricas na Tabela 6.

Tabela 5. Matriz confusão Modelo 1, treinamento #6

		Condição prevista					
		C1	C2	C3	C4	C5	C6
Condição real	C1	92	0	0	0	0	0
	C2	0	98	0	0	0	0
	C3	0	0	95	0	0	0
	C4	0	0	0	88	0	0
	C5	0	0	0	0	81	0
	C6	0	0	0	0	3	90

Tabela 6. Métricas modelo 2 (%)

	Média	Máxima	Mínima
Acurácia	58,649	99,817	0
Precisão	58,912	99,819	0
Recall	58,649	99,817	0
F1	58,757	99,817	0

Aa tabela 7 traz a acurácia obtida por outros quatro modelos de CNN classificadoras do conjunto de dados CWRU, juntamente com os resultados obtidos utilizando a CNN modelo 1 apresentada neste trabalho, que são equivalentes aos melhores resultados na literatura. Porém, as redes aqui apresentadas têm como objetivo tamanho e complexidade computacional reduzidos, o que não é uma preocupação de outros modelos, que são desenvolvidos para desktops.

Tabela 7. Comparação de acurácia na classificação do conjunto de dados CWRU (%)

Modelo	Acurácia
<i>CNN Modelo 1</i>	99,70
Lu et al (2017)	92,60
Shao et al (2018)	99,15
Jia et al (2018)	99,92
Wen et al (2018)	99,79

8. CONCLUSÕES E TRABALHOS FUTUROS

O resultado mais importante e relevante que se pode tirar dos testes realizados neste trabalho é que é possível classificar sinais de vibração relativos a falhas no rolamento de motores elétricos utilizando redes neurais convolucionais simples. As matrizes confusão mostram que é possível classificar com alta acurácia tais sinais, mesmo com as variações em velocidade, o que é um dificultador. Mais ainda, as CNN utilizadas para os testes neste trabalho foram inicialmente desenvolvidas utilizando um conjunto de dados completamente diferente do utilizado aqui, o que demonstra também que além de simples, estas redes não estão limitadas a serem utilizadas em montagens específicas e podem ser generalizadas para diferentes equipamentos, o que é de grande relevância para a viabilidade deste tipo de estratégia de monitoramento, uma vez que diminui a necessidade de desenvolvimentos de arquiteturas específicas para cada sistema a ser monitorado, o que é um problema já levantado em (Randall et al. 2011), que de acordo com o autor inviabilizaria modelos de aprendizado como solução de detecção de falhas em rolamento de motores.

Utilizando ferramentas de portabilidade disponibilizadas por fabricantes de microcontroladores, as redes Modelo 1 e Modelo 2, já treinadas, necessitam de 284 KB e 11 KB de memória, respectivamente, para serem implementadas em microcontroladores, sem qualquer tipo de compressão. Ou seja, além de serem classificadores eficazes, a estrutura das redes aqui analisadas permite que elas sejam embarcadas em microcontroladores que hoje são presença comum em sistemas eletrônicos de baixo custo e baixo consumo, tendo em média de 64 KB a 1MB de memória embarcada disponível.

Uma vez que classificar o CWRU é considerado relativamente simples e um problema já resolvido (Zhang et al. 2020), para o futuro deve-se trabalhar outras características além da eficácia, como custo computacional do classificador, possibilidade de transferência de aprendizado e aplicação em conjuntos de dados mais complexos, em ambientes reais.

REFERÊNCIAS

- APEC Energy Working Group. (2008). Electric Motors – Alignment of Standards and Best Practice Programmes within APEC - Final Report
- Bonnett, A., Yung, C. (2008). Increased Efficiency Versus Increased Reliability. *IEEE Industry Applications Magazine*, 14(1), 29–36.
- Lu, S., Qian, G., He, Q., Liu, F., Liu, Y., Wang, Q. (2019). In situ Motor Fault Diagnosis Using Enhanced Convolutional Neural Network in an Embedded System. *IEEE Sensors Journal*, 1–1.
- Smith, W. A., Randall, R. B. (2015). Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. *Mechanical Systems and Signal Processing*, 64-65, 100–131.
- Gustafsson, O. G., Tallian, T. (1962) Detection of Damage in Assembled Rolling Element Bearings, *ASLE Transactions*.
- Randall, R. B., Antoni, J. (2011). Rolling element bearing diagnostics—A tutorial. *Mechanical Systems and Signal Processing*, 25(2), 485–520.
- Chow, M. -y., Mangum, P. M., Yee, S. O. (1991). A neural network approach to real-time condition monitoring of induction motors. *IEEE Transactions on Industrial Electronics*, 38(6), 448–453.
- Malhi, A., Gao, R. X. (2004). PCA-Based Feature Selection Scheme for Machine Defect Classification. *IEEE Transactions on Instrumentation and Measurement*, 53(6), 1517–1525.
- Esakimuthu Pandarakone, S., Mizuno, Y., Nakamura, H. (2018). Evaluating the Progression and Orientation of Scratches on Outer-Raceway Bearing Using Pattern Recognition Method. *IEEE Transactions on Industrial Electronics*, 1–1.
- Tian, J., Morillo, C., Azarian, M. H., Pecht, M. (2016). Motor Bearing Fault Detection Using Spectral Kurtosis-Based Feature Extraction Coupled With K-Nearest Neighbor Distance Analysis. *IEEE Transactions on Industrial Electronics*, 63(3), 1793–1803.
- Janssens, O., Slavkovikj, V., Vervisch, B., Stockman, K., Loccufier, M., Verstockt, S., ... Van Hoecke, S. (2016). Convolutional Neural Network Based Fault Detection for Rotating Machinery. *Journal of Sound and Vibration*, 377, 331–345.
- Janssens, O., Slavkovikj, V., Vervisch, B., Stockman, K., Loccufier, M., Verstockt, S., ... Van Hoecke, S. (2016). Convolutional Neural Network Based Fault Detection for Rotating Machinery. *Journal of Sound and Vibration*, 377, 331–345.
- Wen, L., Li, X., Gao, L., Zhang, Y. (2018). A New Convolutional Neural Network-Based Data-Driven Fault Diagnosis Method. *IEEE Transactions on Industrial Electronics*, 65(7), 5990–5998.
- Tzeng, C.-B. (2018). Vibration detection and analysis of wind turbine based on a wireless embedded microcontroller system. *2018 IEEE International Conference on Applied System Invention (ICASI)*.
- Hmida, M. A., Braham, A. (2016). ARM based RSWPT implementation for embedded condition monitoring of induction motor. *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*.
- Gongora, W. S., Goedel, A., Castoldi, M. F., Oliveira da Silva, S. A., Da Silva, I. N. (2018). Embedded System to Detect Bearing Faults in Line-Connected Induction Motors. *2018 XIII International Conference on Electrical Machines (ICEM)*.
- Eren, L., Ince, T., Kiranyaz, S. (2018). A Generic Intelligent Bearing Fault Diagnosis System Using Compact Adaptive 1D CNN Classifier. *Journal of Signal Processing Systems*.
- Markiewicz, M., Wielgosz, M., Bochenski, M., Tabaczynski, W., Konieczny, T., Kowalczyk, L. (2019). Predictive Maintenance of Induction Motors Using Ultra-Low Power Wireless Sensors and Compressed Recurrent Neural Networks. *IEEE Access*, 7, 178891–178902.
- Case Western Reserve University Bearing Data Center. <http://csegroups.case.edu/bearingdatacenter>. Acessado em 21 jun. 2020.
- Lu, C., Wang, Z., & Zhou, B. (2017). Intelligent Fault Diagnosis of Rolling Bearing Using Hierarchical Convolutional Network Based Health State Classification. *Advanced Engineering Informatics*, 32, 139–151.
- Shao, H., Jiang, H., Lin, Y., & Li, X. (2018). A Novel Method for Intelligent Fault Diagnosis of Rolling Bearings Using Ensemble Deep Auto-encoders. *Mechanical Systems and Signal Processing*, 102, 278–297.
- Jia, F., Lei, Y., Guo, L., Lin, J., & Xing, S. (2018). A Neural Network Constructed by Deep Learning Technique and its Application to Intelligent Fault Diagnosis of Machines. *Neurocomputing*, 272, 619–628.
- Zhang, S., Zhang, S., Wang, B., Habetler, T. G. (2020). Deep Learning Algorithms for Bearing Fault Diagnostics – A Comprehensive Review. *IEEE Access*, 1–1.