UNIVERSIDADE FEDERAL DO PARÁ
NÚCLEO DE DESENVOLVIMENTO AMAZÔNICO EM ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

RAFAEL DE LIMA ROCHA

REDES NEURAIS CONVOLUCIONAIS APLICADAS À INSPEÇÃO DE COMPONENTES DO VAGÃO FERROVIÁRIO

Tucuruí
2020

UNIVERSIDADE FEDERAL DO PARÁ
NÚCLEO DE DESENVOLVIMENTO AMAZÔNICO EM ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

RAFAEL DE LIMA ROCHA

REDES NEURAIS CONVOLUCIONAIS APLICADAS À INSPEÇÃO DE COMPONENTES DO VAGÃO FERROVIÁRIO

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Núcleo de Desenvolvimento Amazônico em Engenharia, da Universidade Federal do Pará, como requisito para a obtenção do título de Mestre em Computação Aplicada.

Orientador: Prof. Dr. Cleison Daniel Silva
Coorientadora Dra. Ana Claudia da Silva Gomes

Tucuruí
2020

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD
Sistema de Bibliotecas da Universidade Federal do Pará
Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a) autor(a)

D278r de Lima Rocha, Rafael
Redes neurais convolucionais aplicadas à inspeção de componentes do vagão ferroviário / Rafael de Lima Rocha. — 2020.
78 f. : il. color.

Orientador(a): Prof. Dr. Cleison Daniel Silva
Coorientação: Profª. Dra. Ana Claudia da Silva Gomes
Dissertação (Mestrado) - Mestrado Profissional em Computação Aplicada, Núcleo de Desenvolvimento Amazônico em Engenharia, Universidade Federal do Pará, Tucuruí, 2020.

1. Inspeção de vagão ferroviário. 2. Aprendizado profundo. 3. Rede neural convolucional. 4. Classificação de imagem. 5. Transforma discreta de Fourier. I. Título.

CDD 006.3

RAFAEL DE LIMA ROCHA

REDES NEURAIS CONVOLUCIONAIS APLICADAS À INSPEÇÃO DE COMPONENTES DO VAGÃO FERROVIÁRIO

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Núcleo de Desenvolvimento Amazônico em Engenharia, da Universidade Federal do Pará, como requisito para a obtenção do título de Mestre em Computação Aplicada.

Data da Defesa: 03 de Fevereiro de 2020
Conceito: Aprovado

Banca Examinadora

Prof. Dr. Cleison Daniel Silva
Faculdade de Engenharia Elétrica - UFPA
Orientador

Dra. Ana Claudia da Silva Gomes
Instituto SENAI de Inovação em Tecnologias Minerais - ISI
Coorientadora

Prof. Dr. Rafael Suzuki Bayma
Núcleo de Desenvolvimento Amazônico em Engenharia - UFPA
Membro interno

Prof. Dr. Alexandre Trofino Neto
Departamento de Automação e Sistemas - UFSC
Membro externo

Este trabalho é dedicado à minha família e amigos.
Em especial a minha amada mãe, meu questionador sobrinho e minha querida namorada,
que com todo apoio e carinho me ajudam a chegar a essa etapa de minha vida.

AGRADECIMENTOS

Agradeço ao Instituto SENAI de Inovação em Tecnologias Minerais (ISI-TM) pelo apoio e compreensão, em especial minha coorientadora Ana Claudia Gomes pelos sempre bons conselhos dados e pela ajuda imprescindível na realização deste trabalho.

Agradeço ao Instituto Tecnológico Vale (ITV) e a Vale S.A. pelo apoio logístico para realização deste trabalho, em especial ao meu ex-gerente Cleidson de Souza pelo apoio e incentivo a minha jornada acadêmica.

Agradeço ao Programa de Pós-Graduação em Computação Aplicada (PPCA) da Universidade Federal do Pará (UFPA) campus Tucuruí, em especial ao meu orientador Cleison Silva pelo apoio e ajuda essencial para a realização deste trabalho, e a todos os professores que contribuíram de alguma maneira para meu desenvolvimento acadêmico.

Agradeço por fim, e não menos importante, aos meus amigos Viviane Santos e Gerson Serejo por todo apoio e incentivo durante essa jornada que foi a Pós-Graduação em Tucuruí, onde sem a ajuda de vocês a realização deste trabalho seria uma tarefa ainda mais árdua.

"Quando atingimos o nosso ponto mais baixo,
é quando estamos abertos para a maior mudança."
(Aang, A Lenda de Korra)

RESUMO

O vagão ferroviário é um dos patrimônios mais importantes em uma empresa mineradora, onde toneladas de minério são transportados por este diariamente, além disso, o vagão ferroviário pode ser utilizado para o transporte de pessoas. Por isso, a inspeção de defeitos em componentes estruturais do vagão ferroviário é uma atividade de suma importância, possibilitando evitar problemas na logística da ferrovia, assim como prevenir acidentes. A tarefa de inspeção é realizada visualmente por um técnico operacional que está exposto a acidentes no local em que a inspeção é realizada, além da possibilidade de erro humano devido ao estresse, fadiga e outros. O *pad* é componente ferroviário analisado neste trabalho, onde este é responsável pela suspensão primária, papel que é importante na dinâmica dos vagões. Assim, o intuito deste trabalho é utilizar técnicas de aprendizado profundo, especificamente redes neurais convolucionais (CNN) para a realização da inspeção do componente. A CNN classifica a imagem do componente estrutural analisado em relação aos possíveis estados em que ele se encontra na ferrovia, *pad* ausente, *pad* não danificado e *pad* danificado. Além disso, pretende-se investigar a contribuição da imagem do componente no domínio da frequência obtida através da magnitude e fase da transformada discreta de Fourier (DFT) da imagem original (domínio espacial) no processo de classificação da CNN. As técnicas de equalização de histograma e o aumento do número de imagens através do *data augmentation* também são examinadas, de modo a avaliar suas colaborações na melhoria no desempenho de classificação. Os resultados da inspeção do *pad* por CNN demonstram-se bastante inspiradores, em especial quando é utilizada a imagem espacial do componente em conjunto da imagem da magnitude da DFT da imagem de origem como entradas da CNN, que se demonstram superiores quando é utilizada somente a imagem original (espacial) do componente, atingindo uma acurácia de classificação de $95,65\%$. Em especial, o método que utiliza o aumento do número de imagens de treinamento pelo *data augmentation* e as imagens do domínio espacial e da frequência (magnitude) é o que alcança a maior acurácia, com $97,47\%$, que representa aproximadamente $385,5$ imagens classificadas corretamente de um total de $395,2$ imagens.

Palavras-chave: inspeção de vagão ferroviário. aprendizado profundo. rede neural convolucional. classificação de imagem. transformada discreta de Fourier.

ABSTRACT

The railcar is one of the most important assets in a mining company, where tons of ore are transported daily by it, besides, the railcar can be used to transport people. Therefore, the inspection of defects in structural components of the railcar is a very important activity, making it possible to avoid problems in railway logistics, as well as to prevent accidents. The inspection task is performed visually by an operating technician who is exposed to accidents where the inspection is performed, in addition to the possibility of human error due to stress, fatigue, and others. The pad is a rail component analyzed in this work, where it is responsible for the primary suspension, a role that is important in the railcar dynamics. Thus, the purpose of this work is to use deep learning techniques, specifically convolutional neural networks (CNN) for the component inspection. CNN classifies the image of the structural component analyzed concerning the possible state it is in the railway, absent pad, undamaged pad, and damaged pad. Also, it intends to investigate the contribution of the component image in the frequency domain obtained through the magnitude and phase of the discrete Fourier transform (DFT) of the original image (spatial domain) in the CNN classification process. Histogram equalization and increasing the number of images through data augmentation techniques are also examined to evaluate their collaborations in improving classification performance. The results of CNN inspection of the pad prove to be quite inspiring, especially when the spatial component image is used together with the DFT magnitude image of the original image as CNN inputs, which are superior when only the original (spatial) image of the component is used, achieving a classification accuracy of 95.65%. In particular, the method that uses the increase in the number of training images by the data augmentation and the spatial domain and frequency (magnitude) images achieves the highest accuracy, with 97.47%, which represents approximately 385.5 correctly classified images from a total of 395.2 images.

Keywords: railcar inspection. deep learning. convolutional neural network. image classification. discrete Fourier transform.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ANN	Artificial Neural Network
BP	Backpropagation
CNN	Convolutional Neural Network
DFT	Discrete Fourier Transform
DT	Decision Tree
EMT	Eletromagnetic Tomography
GANs	Generative Adversarial Networks
GCCM	Gradient Coded Co-ocurrence Matrix
GLCM	Gray Level Co-ocurrence Matix
HOG	Histogram of Oriented Gradients
IDFT	Inverse Discrete Fourier Transform
kNN	k Nearest Neighboor
LBP	Local Binary Pattern
LDA	Linear Discriminant Analysis
MLP	Multilayer Perceptron
PCA	Principal Component Analysis
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SIFT	Scale Invariant Feature Transform
SVM	Support Vector Machine

SUMÁRIO

1 INTRODUÇÃO

1.1 Problema de pesquisa

O vagão ferroviário é um dos patrimônios mais importantes de uma empresa mineradora e pode ser utilizado tanto para o transporte de cargas (por exemplo, minério de ferro) como para o transporte de passageiros. Na extensão da ferrovia um dos tipos de acidentes mais grave que pode ocorrer é o descarrilamento. Este geralmente ocorre quando há falhas nas rodas ou eixos, trilhos danificados ou objetos nas ferrovias (MACUCCI et al., 2016). Devido a sua gravidade, o descarrilamento pode gerar causalidades e fatalidades, resultando também em danos aos ativos ferroviários (ZHAO; CHAN; STIRLING, 2006). Além disso, também geram implicações ambientais e financeiras como o custo de manutenção e o efeito sobre a logística ferroviária. Uma ação particularmente importante para evitar os descarrilamentos é a inspeção de componentes de vagões, trilhos e rodeiros.

Neste âmbito, a Vale S.A. é a segunda maior empresa de mineração no mundo, e o transporte ferroviário desempenha um papel fundamental nas suas operações. Somente no Brasil, a empresa opera aproximadamente 2.000 quilômetros de trilhos. Uma de suas ferrovias opera o segundo maior trem do mundo, composto por quatro locomotivas e 330 vagões.

Apesar disso, seus vagões ainda são inspecionados de maneira visual (a olho humano) por um técnico. A inspeção visual tem várias desvantagens incluindo a lentidão e a falta de objetividade, propensão ao erro devido à distração, estresse ou fadiga e divergência entre diferentes técnicos (PARK et al., 1996). O problema se torna mais crítico quando alguns dos componentes a serem inspecionados estão localizados abaixo dos vagões, o que significa que é necessário mover estes itens para um local especialmente equipado para realizar a inspeção. Isso requer um tempo adicional, equipamentos específicos e trabalho intenso para realizá-la em tempo aceitável. Além disso, a inspeção visual pode implicar em riscos de segurança para o funcionário (HART et al., 2008).

1.2 Justificativa

Nos últimos anos, técnicas de aprendizado de máquina (*machine learning*) têm sido amplamente utilizadas visando a inspeção de diversos itens nas mais diversas áreas, como, na construção civil, segurança em aeroportos, monitoramento de veículo marinho não tripulado, defeitos em tecidos, iluminação automática (CHA; CHOI; BÜYÜKÖZTÜRK, 2017; LI; ZHAO; PAN, 2017).

As aplicações citadas no parágrafo anterior utilizam um ramo do aprendizado de máquina que é largamente utilizado e tem demonstrado ser extremamente eficaz quando se utiliza imagens, que é conhecido como aprendizado profundo (do inglês, *deep learning*) (LECUN et al., 1998).

O uso de técnicas de aprendizado de máquina também aumentou no âmbito ferroviário com o objetivo de inspecionar automaticamente diversos componentes que contemplam o escopo ferroviário, alguns exemplos são: inspeções da chave de contenção e do crescimento da vegetação ao redor do trilho, além da inspeção dos fixadores do trilho (LIU; ZHOU; HE, 2016a; NYBERG, 2016; GIBERT; PATEL; CHELLAPPA, 2017). Há também outras inspeções feitas na ferrovia, porém sem a utilização de técnicas de aprendizado de máquina, que são as tarefas que envolvem a inspeção do desgaste das rodas através de equipamentos lasers e eletromagnéticos (LIU et al., 2015; CAVUTO et al., 2016).

As abordagens típicas dos sistemas automatizados de inspeção de componentes envolvem aquisição de imagens, pré-processamento, extração de características e classificação. Entre os métodos de classificação aqueles que envolvem aprendizado de máquina, e em particular os métodos de aprendizado profundo, vem se estabelecendo, especialmente em problemas onde as imagens são complexas devido a condições ambientais, reflexão ou distorção da lente (PARK et al., 2016; RAVIKUMAR; RAMACHANDRAN; SUGUMARAN, 2011).

Neste trabalho o componente do vagão do trem que é explorado é o *pad*. Este componente é responsável por suavizar o atrito da roda do vagão durante sua movimentação. Logo o *pad* tem papel fundamental na ferrovia, cujo defeito pode gerar consideráveis problemas, entre os mais graves, o descarrilamento.

1.3 Objetivos

1.3.1 Objetivo geral

O objetivo geral do trabalho é a aplicação de técnicas de aprendizado profundo para realizar a inspeção por imagens de componentes do vagão ferroviário, em particular o *pad*. O algoritmo avaliará a imagem o qual o componente está presente como entrada e classificará está de acordo com os possíveis estados o qual o *pad* pode ser encontrado na ferrovia, entre estes, o estado de danificado. O *pad* é um componente que se encontra próximo de outros no vagão ferroviário, logo, a imagem analisada não possui somente o componente investigado, mas partes de outros componentes, bem como o vagão o qual o *pad* está inserido, desse modo, classificar a imagem quanto ao componente é uma tarefa complexa.

A classificação será avaliada por métricas comumente empregadas na classificação, a saber: acurácia, precisão, *recall* e f1-*score*. Além disso, é investigado o gasto computacional de determinada tarefa, para investigar suas vantagens e desvantagens em relação ao tempo utilizado na inspeção em um ambiente real do componente.

1.3.2 Objetivos específicos

Além da imagem original que possui o componente, pertencente ao domínio espacial, será feita uma investigação da imagem no domínio da frequência e sua contribuição na classificação realizada, onde espera-se encontrar diferenças significativas entre os padrões da imagem no domínio da frequência para a classificação realizada pela técnica de aprendizado profundo. Para análise da frequência, será examinada a transformada de Fourier.

Neste trabalho, são investigadas técnicas de processamento digital de imagens para a aplicação de pré-processamento às imagens do domínio do espaço e a expansão artificial do número de imagens utilizadas na inspeção do componente. O pré-processamento tem como intuito tornar o modelo classificador do *pad* insensível a variações de contraste e iluminação, possibilitando desta maneira uma melhoria na classificação realizada. Por outro lado, a expansão tem como propósito aumentar a quantidade de imagens disponíveis e deixar a inspeção do componente robusta às condições adversas que uma aplicação real requer, possibilitando um melhor desempenho na inspeção feita do *pad*.

Os objetivos específicos deste trabalho são dados como segue:

- Investigar se o aumento no número de imagens espaciais pode gerar uma melhoria no desempenho de classificação;
- Investigar a expansão artificial de imagens do domínio espacial, até quanto essa expansão pode gerar uma melhoria no desempenho de classificação e o custo computacional inerente;
- Investigar individualmente as imagens dos domínios espacial e da frequência para a inspeção do *pad*;
- Investigar a combinação das imagens dos domínios espacial e da frequência;
- Investigar a aplicação de pré-processamento e/ou a expansão artificial de imagens nas melhores combinações de imagens espacial e da frequência.

2 REVISÃO DA LITERATURA

Neste capítulo são apresentados e discutidos os resultados da revisão sistemática da literatura orientada com o propósito de analisar técnicas de *deep learning* e/ou *machine learning* para inspecionar e classificar imagens de componentes do vagão do trem (e/ou de outra aplicação real na indústria e afins) com o intuito de descobrir possíveis danos ou características de um determinado componente.

2.1 Introdução

A inspeção visual de componentes do vagão do trem têm diversas desvantagens, como lentidão, propensão ao erro devido à distração, estresse ou fadiga, além do perigo que o inspetor está exposto na ferrovia (PARK et al., 1996). Logo, o contexto da pesquisa baseia-se em encontrar técnicas para automatizar a inspeção visual existente.

Para efetuar a automatização da inspeção visual dos componentes do vagão do trem, optou-se pela utilização de técnicas que envolvam imagens, especificamente a classificação delas. Essa escolha se deve ao fato da necessidade de se utilizar equipamentos específicos para a execução de inspeção, por exemplo, lasers (DENG et al., 2005; CAVUTO et al., 2016) e sensores eletromagnéticos (LIU et al., 2015).

No contexto da revisão, espera-se encontrar técnicas utilizadas na inspeção de componentes no âmbito ferroviário, na indústria e afins para que sejam aplicadas na inspeção automática de componentes do vagão do trem.

2.2 Planejamento

Para a condução da revisão da literatura espera-se responder a seguinte questão da pesquisa: **"É possível e aplicável utilizar técnicas de *deep learning* e/ou *machine learning* para automatizar a inspeção visual no âmbito ferroviário e afins?"**. A partir desta questão é possível subdividi-la em outras, como segue:

- **(Q1)** Quais as técnicas utilizadas nos estudos para a realização da inspeção?
- **(Q2)** Quais as vantagens das técnicas que tem como base *Machine Learning*?
- **(Q3)** Quais as vantagens das técnicas que tem como base *Deep Learning*?

O objetivo de subdividir a questão da pesquisa é tentar cobrir a maior parte de técnicas utilizadas na inspeção nas áreas ferroviária e correlacionadas, para que se possa extrair as vantagens e desvantagens delas e assim concluir a respeito da viabilidade da utilização das mesmas para a inspeção.

Para o desenvolvimento da revisão utilizou-se uma string de busca que combina palavras-chaves e sinônimos com o intuito de obter, pelas ferramentas de busca, uma quantidade de estudos relevantes, a string de busca final é como segue:

- ((*inspection*) AND (*deep learning*)) OR ((*train inspection*) OR (*wagon inspection*))

Os critérios utilizados na formação da *string* de busca tem como finalidade verificar publicações relacionadas a inspeções de um modo geral (âmbito ferroviário e afins), assim como obter informações sobre as abordagens utilizadas estritamente relacionadas à inspeção de componentes do trem e do vagão.

Diversas fontes foram utilizadas para a realização da revisão. A lista de fontes final as quais a revisão da literatura foi executada é mostrada na Tabela 1. O Portal de Periódicos da Capes e *Networked Digital Library of Theses and Dissertations* (NDLTD) foram as principais fontes de busca utilizadas com a *string* de busca escolhida onde, os resultados gerados pela busca foram em seguida encaminhados para as outras fontes presentes na Tabela 1.

Tabela 1 – Fontes que foram utilizadas na revisão da literatura.

#	Fontes
1	arXiv
2	Cambridge Core
3	DiVA
4	IEEE Xplore Digital Library
5	IEICE
6	IET Digital Library
7	IOP Science
8	IOS Press
9	ISPRS
10	MIT Library
11	Oxford Academic
12	Repositório Institucional Unesp
13	ScienceDirect
14	SpringerLink
15	SpringerOpen
16	Wiley Online Library

Para a inclusão dos estudos primários na revisão da literatura, foram definidos os critérios de avaliação para decidir se eles deveriam ou não ser selecionados no contexto da revisão sistemática. Os critérios de inclusão e exclusão dos estudos são mostrados a seguir:

- Descrever a classificação de imagens utilizando técnicas de *deep learning*.
- Descrever a inspeção de objetos, produtos ou componentes na área industrial e afins, entre outras.

- Descrever a inspeção de componentes do trem, do vagão do trem ou do ambiente em geral que envolva a logística da via ferroviária.
- Descrever um procedimento com as técnicas de *deep learning* utilizadas em uma aplicação real que utilize imagens reais ou próximas ao real.

A escolha dos critérios se deu pela necessidade de inserir na revisão estudos que possuíssem a descrição da inspeção feita em ambientes envolvendo a ferrovia ou áreas afins, como a industrial, além disso, os estudos poderiam descrever a classificação de imagens utilizando técnicas de *deep learning*, assim como, utilizar imagens reais ou próximas do item a ser inspecionado.

A leitura do resumo, leitura de uma ou mais secções ou leitura de todo o trabalho são os procedimentos realizados para avaliação dos estudos de acordo com os critérios de inclusão e exclusão. Em uma execução inicial foi gerado um total de 45 estudos, porém entre os estudos preliminares obtidos, somente 26 dos estudos satisfazem todos os critérios de inclusão e exclusão e estes foram selecionados para serem utilizados como estudos primários na revisão da literatura.

Tabela 2 – Identificação (ID) e respectiva referência dos estudos presentes na revisão da literatura.

ID - Referência	
E01 - (STENTOUMIS et al., 2016)	E14 - (CHEN; JAHANSHAHI, 2018)
E02 - (JACCARD et al., 2016)	E15 - (POUND et al., 2017)
E03 - (LIU; ZHOU; HE, 2016a)	E16 - (GIBERT; PATEL; CHELLAPPA, 2017)
E04 - (FENG et al., 2014)	E17 - (LI; ZHAO; PAN, 2017)
E05 - (WANG et al., 2018)	E18 - (JACCARD et al., 2017)
E06 - (SOME, 2016)	E19 - (LIU et al., 2015)
E07 - (NYBERG, 2016)	E20 - (VIEIRA, 2016)
E08 - (LI; ZHANG; LIN, 2016)	E21 - (SOUKUP; HUBER-MÖRK, 2017)
E09 - (GHOSAL et al., 2017)	E22 - (CARO; CABRERA et al., 2016)
E10 - (AFFONSO et al., 2017)	E23 - (MENTZELOS, 2016)
E11 - (MÜHLING et al., 2017)	E24 - (DENG et al., 2005)
E12 - (LUCKOW et al., 2016)	E25 - (CAVUTO et al., 2016)
E13 - (CHA; CHOI; BÜYÜKÖZTÜRK, 2017)	E26 - (LIU; ZHOU; HE, 2016b)

Dentro do grupo de estudos obtidos, aqueles que foram excluídos da revisão da literatura deveu-se ao fato de seu conteúdo não preencher pelo menos um dos critérios de inclusão e exclusão, citados anteriormente. Muitos dos quais, foram eliminados pois não explicavam de forma clara a inspeção de vagões ou trem; ou não utilizavam as técnicas de *deep learning* na realização da classificação e sim na detecção de imagens, em séries temporais, em textos ou outros.

A identificação (ID) de cada um dos estudos utilizados na revisão da literatura e sua respectiva referência estão listados na Tabela 2. Cada um dos estudos presentes na Tabela 2 é identificado com E (de estudo) seguido de sua numeração (valores entre 1 e 26). Assim, quando os estudos que compõem a revisão da literatura forem citados no restante do Capítulo 2, os IDs serão utilizados.

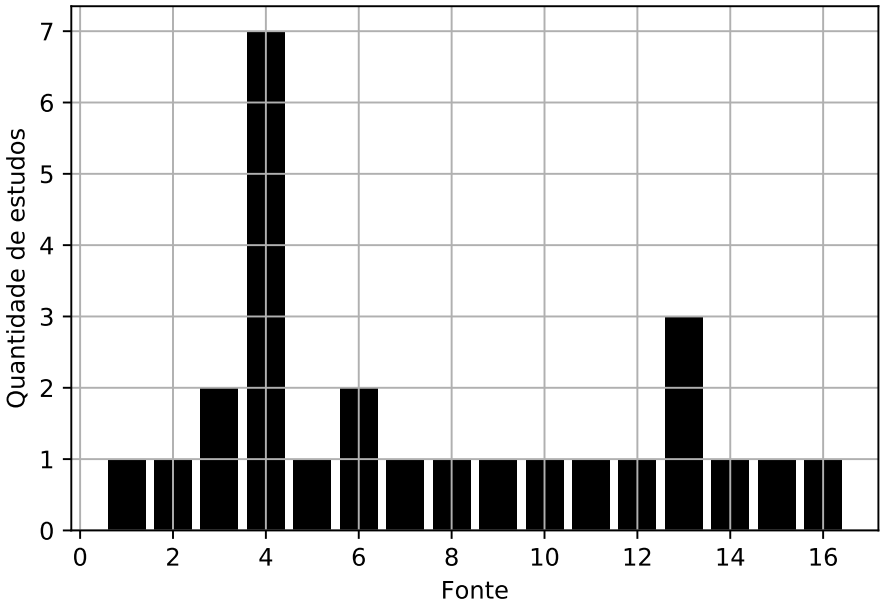


Figura 1 – Quantidade de estudos obtidos por cada uma das dezesseis fontes utilizadas a revisão da literatura.

A Figura 1 mostra a quantidade de estudos selecionados por cada uma das fontes presentes na Tabela 1. Observa-se que as fontes com mais estudos selecionados são as fontes de número 4 e 13, com respectivamente 7 e 3 estudos cada. Com a quantidade de estudos da fonte IEEE Xplore Digital Library (fonte de número 4) representa, aproximadamente, 27% do total de estudos selecionados para a revisão da literatura.

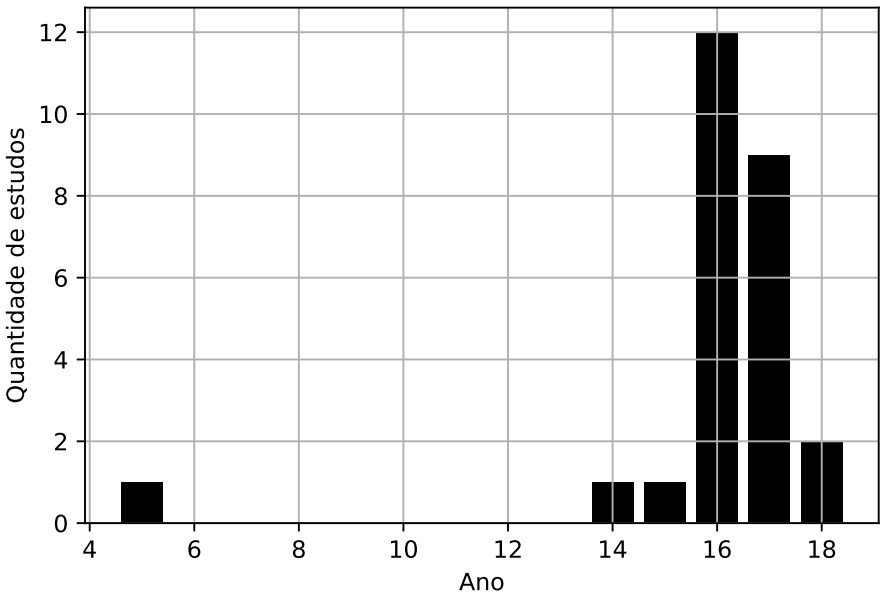


Figura 2 – Quantidade de estudo obtidos na revisão da literatura, distribuídos por ano, entre o período de 2005 (05) e 2018 (18).

A quantidade de estudos selecionados para a revisão de acordo com o ano de publicação é exemplificada na Figura 2. As informações que se destacam são os dos anos de 2016 e 2017

com 12 e 9 estudos respectivamente, somados os dois são mais que 80% dos estudos utilizados na revisão.

2.3 Execução

Com o intuito de cobrir os 26 estudos de forma otimizada, esse grupo foi separado em dois, como mostra a Tabela 3. O grupo 1 possui os estudos que contém inspeções envolvidas em aplicações reais que utilizam *deep learning* e o grupo 2 descreve inspeções na área ferroviária que utilizam técnicas de *machine learning*, *deep learning* ou outras. O grupo 1 se aplica a 18 dos estudos (70%) e o grupo 2 a 8 dos estudos (30%) presentes na revisão da literatura.

Antes de analisar cada um dos grupos faz-se necessária a apresentação de dois conceitos: aprendizado supervisionado e não supervisionado. O aprendizado supervisionado é comumente relacionado como sinônimo de classificação. Esse tipo de aprendizado ocorre quando a base de dados de treinamento está rotulada. Já no aprendizado não supervisionado (conhecido também como clusterização) possui uma base de dados que não está rotulada (HAN; KAMBER; PEI, 2011).

Tabela 3 – Divisão dos estudos utilizados na revisão da literatura.

Grupos	Tipo de aplicação	Estudos	Quantidade
1	*Deep learning*	E01-02, E05-06, E08-15, E17-18, E20-23	18
2	Ferrovia	E03-04, E07, E16, E19, E24-26	8

No grupo 1 a maior parte dos estudos abordam a utilização de redes neurais convolucionais (*convolutional neural network*, CNN) para a inspeção via classificação de determinado item de interesse. Esses itens de interesse abrangem diversas áreas da indústria como construção civil (E01), segurança em aeroporto (E02), detecção de rachaduras na rodovia (E06), monitoramento de veículo marinho não tripulado (E23), entre outros.

Um número menor de resultados no grupo 1 tem como abordagem o uso de *deep learning* no aprendizado não supervisionado. Essa inspeção é feita como o uso de *autoencoder* (AE) para aprender a representação do conjunto de dados. Os estudos primários que utilizam AE são: o estudo cinco (E05), que demonstra um sistema de inspeção para iluminação automática; e os estudos oito e dezessete (E08 e E17) que abordam inspeção para a detecção de defeitos de fábrica em tecidos.

Em relação ao estudo E08, ele se destaca pela utilização do critério de *fisher* com *deep learning* (AE) para o reconhecimento de padrões de tecido, mostrando-se superior ao estado da arte. Porém, a principal crítica a este estudo recai sobre os padrões dos tecidos utilizados serem relativamente simples. Em contrapartida, o estudo E17 faz uma inspeção eficiente em ambos os padrões de tecido (simples e complexos), apesar do número insuficiente de amostras negativas presentes na base de dados. Já o outro estudo que utiliza AE (E05), se destaca principalmente

no que diz respeito a sua viabilidade e eficiência. Contudo, o método proposto pelo estudo funciona somente quando a distância de observação está correta, além de necessitar de uma imagem de extrema qualidade para a realização da inspeção com o objetivo fazer uma iluminação automática.

Ainda no grupo 1, parte das abordagens propostas são comparadas com técnicas comumente utilizadas para o aprendizado supervisionado, como por exemplo, redes neurais artificiais (*artificial neural network*, ANN), Árvore de decisão (*decision tree*, DT), e k vizinhos mais próximos (*k nearest neighboor*, kNN), *random forest* (RF), máquinas de vetores de suporte (*support vector machine*, SVM). Porém, como o que deve ser classificado (inspecionado) é uma imagem, é necessário a utilização de métodos para extração de características da imagem que servirão entrada para algum dos classificadores supervisionados.

No estudo E01, por exemplo, é feita a troca na saída da CNN por outros classificadores, como ANN, kNN e SVM para o propósito de comparação para a abordagem proposta pelo estudo.

Por outro lado, no grupo 2, destacam diversos estudos que abordam a inspeção no âmbito ferroviário, como: inspeção da chave de contenção em trens de carga (E03, E16), ausência, desgaste e classificação dos fixadores do trilho (E04), inspeção do crescimento da vegetação ao redor do trilho (E07), inspeção do desgaste da roda (E19, E25), entre outros.

Dentro do grupo 2, a maioria dos estudos são focados em extração de características das imagens combinados a um classificador supervisionado. Os classificadores são semelhantes aos observados no grupo 1. Já as técnicas de extração de características das imagens destacam-se o *bag-of-words* (BOF), a matriz de co-ocorrência de níveis de cinza (*gray level co-ocurrence matrix*, GLCM) e algumas variações, *garbor wavelet* (GW), histograma de gradientes orientados (*histogram of oriented gradients*, HOG), o descritor de textura Haralick's (*haralick's texture descriptor*, HTD), Padrão binário local (*local binary pattern*, LBP) e SIFT (*scale invariant feature transform*).

Uma parte pequena deste grupo realiza a inspeção na ferrovia por meio de diversos equipamentos e técnicas robustas para medições de componentes relacionados a área. O estudo E19 se destaca pelo uso da técnica de tomografia eletromagnética (*electromagnetic tomography*, EMT), onde é possível obter o formato do defeito e sua respectiva posição no trilho. Além da robustez do equipamento necessário neste estudo, surge a dificuldade de implementação do mesmo, uma vez que todos os testes feitos foram realizados em laboratório.

Já os estudos E24 e E25 se destacam pela inspeção do perfil da roda ferroviária, que está relacionado ao nível de desgaste da mesma. O primeiro dos estudos utiliza o laser para medir o perfil da roda no início e no final de seu ciclo de vida, porém a principal dificuldade no estudo se dá em função da diversidade de tipos de rodas a serem analisadas, fator de suma importância para ser fazer a medição do perfil. Por outro lado, em E25, a inspeção é feita por

laser ultrassônico onde a medida é obtida sem qualquer tipo de contato com a roda, tornando-se assim, uma aplicação mais ágil e flexível.

Em comparação a ambos os grupos, a abordagem apresentada no estudo E16, que pertence ao grupo 2, é o único dos estudos presentes na revisão literatura que apresenta uma abordagem que utiliza técnica de *deep learning* (CNN) para a inspeção (classificação) em um ambiente que contempla o escopo ferroviário. Neste estudo, o componente a ser inspecionado é o fixador dos trilhos, que pode ser classificado em três possíveis estados: ausência de fixador, fixador quebrado e fixador bom. Dessas classes o estudo ainda subdivide as classes quebrado e bom em outras. O resultado obtido deste estudo demonstra uma vantagem superior da abordagem CNN comparada ao classificador SVM, que necessita de um passo anterior para a extração de características HOG.

Ainda sobre E16, vale ressaltar que o principal obstáculo encontrado no estudo se dá pelo número reduzido de amostras em algumas das classes. Apesar disso, o estudo consegue contornar o pequeno número de amostras com o aumento de dados (*data augmentation*) inserindo espelhamento vertical e horizontal nas imagens, além de fazer recortes pseudoaleatórios nestes dados.

Dois outros estudos do grupo 2 destacam-se: E03 e E26. Ambos abordam a inspeção para verificar a presença da chave de contenção presente em trens de carga. O classificador SVM e a extração de características são utilizadas por ambos. A principal diferença entre eles se deve à técnica de extração de características utilizada, no estudo E03 é utilizado o GLCM e variações para a extração, porém no estudo E26 as características são extraídas com a técnica HOG.

A principal colaboração dos estudos E03 e E26 se dá pela automatização de inspeção visual da chave de contenção garantindo a segurança do tráfego na ferrovia. A técnica proposta em E03, GCCM (*gradient coded co-ocurrence matrix*), que é uma variação da GLCM, se destaca principalmente devido a sua robustez no que diz respeito a mudanças de iluminação em ambientes ao ar livre. Por outro lado, o estudo E26, destaca-se pela excelente performance no que diz respeito a uma resposta real do sistema, apesar da acurácia dos resultados serem semelhantes ao da inspeção humana.

2.4 Resultados

2.4.1 Contextualização

Nesta seção são realçadas as perspectivas mais exploradas pelos estudos primários. Assim como serão respondidas as perguntas iniciais da revisão da literatura. Por fim será concluída a revisão da literatura para que a inspeção de componentes do vagão do trem possa ser realizada.

Dentre os assuntos abordados nos estudos presentes na revisão da literatura, o que se encontra frequentemente explanado é o tópico de extração de características para a posterior

classificação (inspeção) do objeto de interesse. A extração de característica é definida como uma forma de redução da dimensionalidade dos dados em reconhecimento de padrões e processamento de imagens, ou seja, a obtenção de informações relevantes dos dados e representá-las em um espaço dimensional menor (KUMAR; BHATIA, 2014).

Caso as características extraídas sejam escolhidas de maneira cuidadosa, serão extraídas informações valiosas dos dados de entrada para a realização da tarefa em questão ao invés da utilização do dado completo, como uma imagem por exemplo. Ao se escolher erroneamente as características extraídas, o classificador não será capaz de realizar de maneira hábil sua tarefa (KUMAR; BHATIA, 2014).

É possível inferir que aprender padrões de imagens é uma tarefa complexa, que pode ser realizada por métodos de extração de características. Além disso, a escolha dessa característica a ser extraída também é uma tarefa árdua.

Deep learning é uma abordagem cada vez mais utilizada para processar dados complexos, e em especial as CNNs (LECUN et al., 1989) têm sido empregadas de maneira eficiente com o intuito de aprender diretamente de dados bidimensionais (2-D). Esta topologia em grade 2-D não requer nenhum tipo pré-processamento e extração de características comumente empregados, como os artigos da revisão da literatura mostram. Além disso, as CNNs têm como vantagem a flexibilidade em relação a variação de translação e distorções locais (LECUN et al., 1998; LECUN; BENGIO; HINTON, 2015).

2.4.2 Respostas das questões

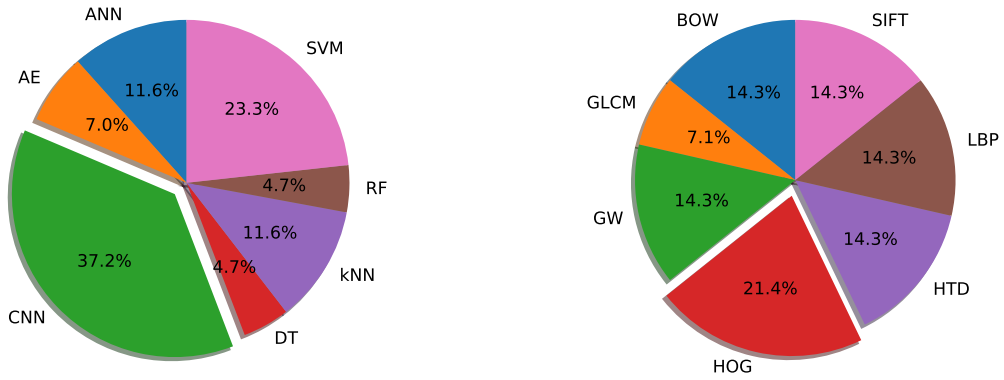
Com base na revisão sistemática da literatura é possível responder cada umas das questões presentes inicialmente, as questões e suas respectivas respostas são mostradas a seguir.

(Q1) Quais as técnicas utilizadas para a realização da inspeção?

Diversas técnicas são utilizadas para que a inspeção possa ser feita. Em relação as técnicas de classificação têm-se AE, ANN, CNN, DT, kNN, RF e SVM como algumas das técnicas encontradas na revisão. Como grande parte destas técnicas não aceitam uma imagem como entrada, se tem a necessidade de extrair as características das imagens via algum método de extração, como, BOW, GLCM, GW, HOG, HTD, LBP e SIFT.

A Figura 3 mostra os gráficos de pizza das técnicas de classificação e de extração de características presentes nos estudos primários da revisão da literatura, respectivamente. As frequências desses gráficos foram obtidas das técnicas utilizadas nos estudos da revisão da literatura, sejam as técnicas do método proposto pelo estudo ou que servem de comparação do mesmo.

Da Figura 3a é possível notar a porcentagem referente a técnica CNN, com 37,2% do total, mostrando que a técnica é a mais frequente nos estudos na soma (44) das frequências de



(a) Técnicas de classificação **(b) Técnicas de extração**

Figura 3 – Gráfico em pizza das técnicas de classificação (a) e extração de características (b) encontradas nos trabalhos da revisão da literatura.

todas as técnicas. Em comparação a técnica RF, que possui a menor frequência (2,3%), ou seja, presente em apenas um dos estudos. Ainda é possível evidenciar a soma das frequências das técnicas de *deep learning*, CNN e AE, que é pouco mais de 44% da frequência total.

A frequência (Figura 3b) das técnicas de extração de características encontradas nos estudos primários é bem equilibrada. Porém, vale destacar a técnica HOG, que possui 21,4% de frequência do total de 12 na soma das frequências das técnicas de extração. Essa porcentagem é equivalente 3, ou seja, a técnica HOG aparece em 3 dos estudos primários da revisão da literatura. Em contrapartida ao GLCM e suas variações, que possuem somente $7,1\%$ de frequência, o menor valor entre as técnicas de extração de características.

(Q2) Quais as vantagens das técnicas que tem como base *machine learning*?

A vantagem da utilização das técnicas de *machine learning* se dá principalmente pelo uso de estratégias de extração de características. Como o objetivo da extração é obter as informações mais relevantes dos dados originais (imagem) em uma dimensão menor. Por exemplo, quando se tem um grande quantidade de dados, mas não se possui muita informação, se faz necessária a transformação dos dados de entrada para uma representação reduzida do mesmo, chamada de vetor de características (KUMAR; BHATIA, 2014).

Além disso, tem-se a vantagem do uso de técnicas comumente empregadas em tarefas de classificação. Como ANN, SVM, e outras, como foram mostradas ao longo da revisão da sistemática da literatura.

(Q3) Quais as vantagens das técnicas que tem como base *deep learning*?

Ao contrário da resposta da Q2, aqui (CNN) não se tem a necessidade da extração de características das imagens. Pois como dito anteriormente, essa extração ocorre de maneira inerente a camada da arquitetura da rede convolucional utilizada (LECUN; BENGIO; HINTON, 2015).

2.4.3 Conclusão

Com as questões da revisão da literatura respondidas, é possível observar a grande variedade de aplicações que utilizam técnicas de *deep learning* e *machine learning* para automatizar certa inspeção visual existente.

Como dito no início da revisão espera-se responder a seguinte questão principal: **"É possível e aplicável utilizar técnicas de *deep learning* e/ou *machine learning* para automatizar a inspeção visual no âmbito ferroviário e afins?"**. A resposta para essa questão é: sim, é possível a utilização destas técnicas, em especial as redes neurais convolucionais.

No que diz respeito a alguns estudos do grupo 2, se tem a necessidade de equipamentos específicos que façam as medidas do desgaste da roda, sem nenhum tipo de inteligência presente, ou seja, é necessário um sistema robusto para que se realize a inspeção. Por outro lado, muitos dos estudos primários que envolvem a inspeção no âmbito ferroviário utilizam técnicas de *machine learning* para realizar essa tarefa, porém é requerido um passo adicional para a extração das características das imagens a serem inspecionadas (classificadas), além do cuidado na escolha destas características.

No entanto, as redes neurais convolucionais (técnica de *deep learning*) se destacam, entre outras coisas, devido a falta de necessidade deste passo anterior, pois, o processo de extração de característica ocorre de forma interna, além de resultados superiores aos de *machine learning* como mostram os estudos primários da revisão da literatura.

Assim, conclui-se que redes neurais convolucionais são aplicáveis na automatização da inspeção visual existente na área ferroviária, já que está é uma técnica de *deep learnig* robusta o suficiente para se fazer a extração de características seguida da classificação de um possível dano do componente que será analisado, sem necessitar de uma etapa de extração de características anterior à classificação.

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados aos conceitos básicos de análise de dados, especificamente, a classificação de dados, a conceituação atrelada às redes neurais artificiais, indo desde de rede neural de uma camada até a de múltiplas camadas. Aqui também será abordada a fundamentação teórica de redes neurais convolucionais e suas principais propriedades, cujos conceitos são de suma importância no decorrer do trabalho. Além disso, será apresentado o conceito por trás da transformada discreta de Fourier, a qual também servirá de entrada para a rede neural convolucional.

3.1 Análise de dados

3.1.1 Conceito de classificação

Classificação é uma forma de análise de dados que extrai modelos descrevendo classes de dados importantes (HAN; KAMBER; PEI, 2011). Cada modelo ou classificador, estima rótulos de dados categóricos. Muitos métodos de classificação têm sido propostos por pesquisadores no aprendizado de máquina, reconhecimento de padrões e estatística. No aprendizado de máquina, diversas abordagens são utilizadas na construção de um modelo classificador de dados, como redes neurais (Seção 3.2), naive bayes, máquinas de vetores de suporte ou SVM e k vizinhos mais próximos (kNN).

Classificação está inserida em diversas aplicações, como detecção do estado apresentado por determinada planta, autenticidade de notas de euro, e inspeção do estado apresentado pelo fixador do trilho ferroviário, entre outros (GHOSAL et al., 2017; SOUKUP; HUBER-MÖRK, 2017; GIBERT; PATEL; CHELLAPPA, 2017).

Em diversas tarefas que empregam classificação, o classificador ou modelo é construído para estimar rótulos de classes categóricos, como "saudável"ou "deficiência em ferro"em aplicações que investigam o estado apresentado por determinada planta; "genuína"ou "falsificada"ao analisar a autenticidade de notas de euros; e "ausente", "bom"ou "quebrado"ao realizar a inspeção por imagens do estado do fixador do trilho. Estas categorias ou classes apresentadas podem ser representadas por valores discretos, como 1, 2 ou 3 (ou A, B ou C) para as categorias "ausente", "bom"ou "quebrado", por exemplo.

3.1.2 Classificação de dados

Duas etapas compõem a classificação de dados, etapa de treinamento ou aprendizagem, onde o modelo de classificação de fato é construído e a etapa de classificação, onde o modelo é usado para estimar os rótulos das classes para um determinado dado.

Na etapa de aprendizagem, o algoritmo de classificação constrói o classificador analisando o conjunto de dados de treinamento e suas respectivas classes. Como o conjunto de treinamento e suas classes são fornecidos, esse processo de aprendizagem é compreendido como aprendizado supervisionado. Em contrapartida, no aprendizado não supervisionado, o rótulo das classes de cada dado de treinamento não é fornecido (HAN; KAMBER; PEI, 2011).

Enquanto a etapa de classificação, o conjunto de dados de validação (dados e rótulos das classes) avalia o modelo gerado pelo algoritmo de classificação através de uma determinada métrica de classificação. Vale ressaltar que o conjunto de dados de validação são independentes do conjunto de treinamento, deste modo, não é utilizado durante a construção do classificador.

Em relação aos valores discretos que representam as classes, é possível descrever variáveis discretas dadas por um de K possíveis estados mutualmente exclusivos (BISHOP, 2006), onde o estado indica a classe do respectivo dado. O esquema 1-de-K é comumente utilizado quando se trabalha com tais variáveis, o qual as variáveis são representadas por um vetor x de dimensão K, onde um dos elementos de x_k é igual a 1, e os elementos restantes são iguais a 0.

$$x = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

A Equação 3.1 representa uma variável com $K = 3$ estados, indicando os possíveis estados do fixador do trilho por exemplo, que são "ausente", "bom"ou "quebrado". Na Equação 3.1, o vetor x com um estado correspondente em $x_3 = 1$, mostra que esse dado é estimado como pertencente a classe 3, representando o estado "quebrado"do fixador do trilho.

3.2 Redes neurais artificiais

3.2.1 Introdução

Partindo do pressuposto que o cérebro humano e um computador processam informações de maneira diferente tem-se o interesse no estudo de redes neurais artificiais (*artificial neural network*, ANN). O cérebro é bastante complexo e capaz de realizar certos tipos de processamento, como o reconhecimento de padrões muito mais rápido que um computador, devido à capacidade de organizar seus neurônios. Diariamente o ser humano é capaz de realizar diversos reconhecimentos de padrões, dentre os mais comuns destacam-se o reconhecimento de faces e voz de pessoas conhecidas. Reconhecimento de padrões é definido como o processo que recebe um padrão ou sinal de entrada e consegue atribui-lo a uma de suas classes predefinidas (HAYKIN, 2009).

A rede neural é uma máquina biologicamente inspirada planejada para simular o modo como o cérebro humano realiza determinada tarefa (BISHOP, 1995). Na rede neural através de um processo de aprendizagem, o conhecimento é adquirido e esse conhecimento é armazenado nos

pesos sinápticos (força de conexão entre neurônios). Ambos os aspectos citados se assemelham ao cérebro.

Rede neural artificial é uma técnica de aprendizado de máquina ou *machine learning* definida por (HAYKIN, 2007) como um processador paralelamente distribuído que possui unidades de processamento simples que tem a habilidade de armazenar conhecimento e torna-lo disponível para o uso. O ato de "aprender"(conhecimento) de uma rede neural diz respeito a sua capacidade de generalizar. Generalização é a propensão da rede neural produzir saídas corretas para as entradas que não se encontravam durante a aprendizagem (treinamento) (BISHOP, 2006).

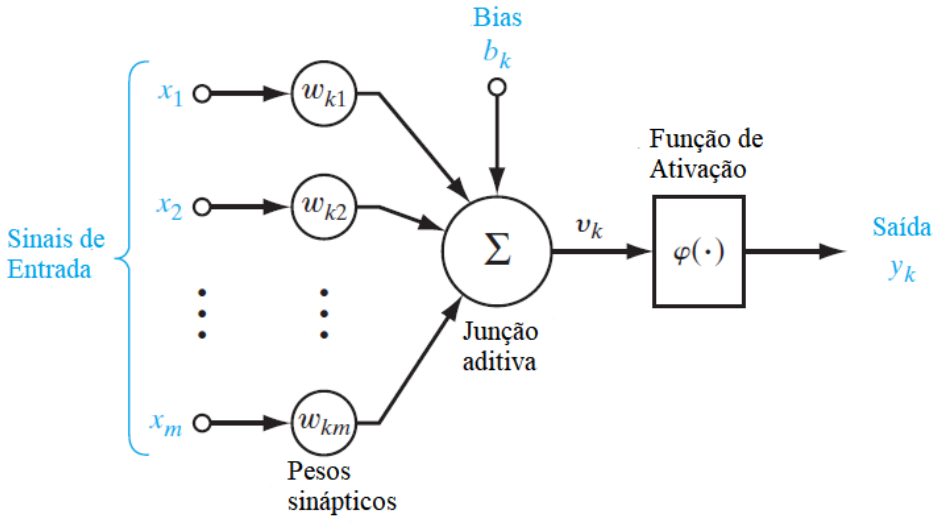


Figura 4 – Modelo que apresenta as conexões em um neurônio simples utilizado como base para uma rede neural. Essas conexões com o neurônio são formadas por sinapses, pelo somador e pela função de ativação. Fonte: Adaptada de (HAYKIN, 2009).

3.2.2 Perceptron de uma camada

Uma rede neural possui uma unidade que processa informações, chamado de neurônio artificial, que é essencial para o funcionamento da rede. O exemplo de um neurônio e suas correspondentes conexões é mostrado na Figura 4. Esse modelo apresentado é idêntico ao modelo probabilístico utilizado para o aprendizado supervisionado apresentado por (ROSENBLATT, 1958), cuja denominação dada ao modelo é perceptron. O perceptron, que representa uma rede neural de uma única camada, é um combinador linear (caso uma função linear seja utilizada) que gera um de dois valores discretos como resultado de saída, quando as entradas do perceptron são representadas por duas classes linearmente separadas (SHYNK, 1990).

No perceptron da Figura 4 é possível verificar os sinais de entrada associados a cada um dos pesos sinápticos, onde este é diretamente associado a um certo neurônio. É possível também notar o somador presente no neurônio que tem a função de associar as entradas às sinapses. Por fim, têm-se a função de ativação, cujo o intuito é limitar os valores da saída em certo intervalo e

o bias que pode aumentar ou reduzir o valor de entrada na função de ativação.

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \tag{3.2}$$

$$y_k = \varphi(v_k) \tag{3.3}$$

$$v_k = u_k + b_k \tag{3.4}$$

O neurônio e conexões presentes na Figura 4 podem também serem explanados de uma maneira matemática, representados nas Equações 3.2-3.4. O valor de u_k representa o resultado do somatório da multiplicação dos pesos w_{kj} referente ao k-ésimo neurônio e dos sinais de entrada x_j. A saída da função de ativação do neurônio k é dada por y_k, a função de ativação φ é aplicada a combinação linear u_k somado ao valor do bias b_k o qual o neurônio é aplicado. Para uma melhor simplificação do que será exposto adiante optou-se por simplificar a entrada na função de ativação para a variável v_k, que é representado adição de u_k e b_k.

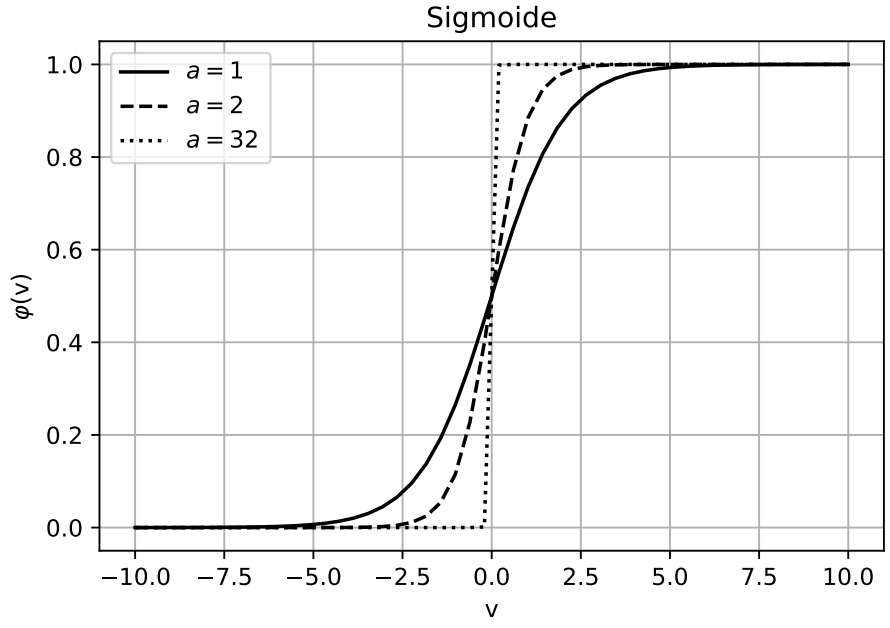


Figura 5 – Função logística sigmoide. Três diferentes valores de a são exibidos, a curva sólida representa $a = 1$, o valor $a = 2$ é exemplificado na curva tracejada e a curva pontilhada mostra o parâmetro a recebendo o valor 32.

Diversas funções de ativação podem ser utilizadas, porém uma das mais comuns utilizadas na estrutura de construção de redes neurais é a função sigmoide, que é uma função de ativação que apresenta certo equilíbrio entre os comportamentos linear e não linear. Um comum exemplo deste tipo de função é a logística mostrada na Equação 3.5 e a Figura 5 exemplifica o funcionamento da função logística. O eixo horizontal representa os valores de entrada v cujos valores neste

caso variam entre -10 e 10, enquanto que o eixo vertical reflete os valores associados a saída da função de ativação φ entre os valores 0 e 1.

$$\varphi(v) = \frac{1}{1 + e^{-av}} \tag{3.5}$$

A entrada v da função é mostrada na Equação 3.4 e com o parâmetro a é possível ajustar a inclinação da curva da função logística. Alguns exemplos de inclinações da função logística sigmoide são mostradas na Figura 5, são exemplificados os valores 1, 2 e 32 nas curvas sólida, tracejada e pontilhada, respectivamente. Ambas as curvas sólida e tracejada apresentam um comportamento não linear (mais de duas classes não linearmente separáveis). No entanto, quando o valor de a aumenta consideravelmente a curva pontilhada se comporta como uma função linearmente separável (também conhecida como função limiar), como mostra o comportamento próximo a origem dessa curva na Figura 5, onde os valores superiores e inferiores a 0 (eixo horizontal) pertencem a uma classe de saída cada.

3.2.3 Perceptron de múltiplas camadas

Apesar de um perceptron de uma camada ser utilizada em várias aplicações que necessitam somente de duas classes de saída, existem outras aplicações que requerem um número maior de saídas que uma saída binária. Partindo deste requisito, tem-se a utilização da rede neural que utiliza um perceptron de múltiplas camadas (*multilayer perceptron*, MLP), que entre outras características, emprega a classificação com mais de duas classes.

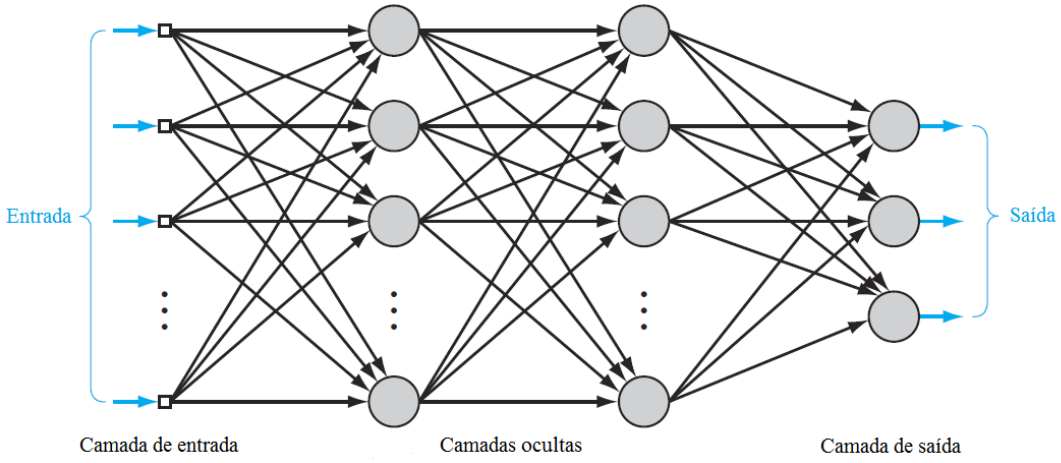


Figura 6 – Modelo de um perceptron de múltiplas camadas. Essa rede neural é formada pela camada de entrada, duas camadas ocultas e a camada de saída. Fonte: Adaptada de (HAYKIN, 2009).

O perceptron multicamadas tem algumas peculiaridades em relação ao perceptron com uma única camada. A primeira é relacionada ao fato de que cada neurônio na rede está atrelado a uma função de ativação não-linear, o qual permite que a classificação de mais de duas classes de

saída. Além disso, essa rede tem, pelo menos, uma camada oculta de ambas camadas de entrada e saída (HAYKIN, 2009).

As camadas ocultas, no que diz respeito ao processo de aprendizagem do MLP, são responsáveis por descobrir progressivamente, dentro dos dados utilizados para aprendizagem (ou dados de treinamento), características proeminentes por meio das funções de ativação não lineares presentes nesses neurônios. Desse modo é possível separar os dados de entrada para que a classificação supervisionada de padrões seja realizada. A classificação supervisionada é estabelecida por (HAYKIN, 2009) como um processo que detém conhecimento do ambiente em si, que pode ser representado como o conhecimento das entradas e suas respectivas classes de saída. Assim, os neurônios das camadas ocultas atuam como detectores de características, como é mostrado em (HAYKIN, 2009).

A Figura 6 apresenta um modelo de um perceptron multicamadas, a arquitetura desta rede é composta por uma camada de entrada, duas camadas ocultas e uma camada de saída de três neurônios. Sobre a Figura 6, vale observar que há conexões entre todos os neurônios de qualquer camada com quaisquer neurônios da camada antecedente a essa, para esta peculiaridade a rede é denominada totalmente conectada (HAYKIN, 2009).

3.2.4 Aprendizagem

Existem alguns algoritmos de aprendizagem utilizados em redes neurais artificiais, em especial o modelo de perceptron multicamadas, porém o algoritmo de retropropagação (*backpropagation*, BP) do erro vem sendo utilizado extensamente, e sua popularização se deu por (RUMELHART et al., 1987).

Para o algoritmo ser utilizado é necessário que as entradas da rede estejam identificadas de maneira correta, ou seja, cada uma das entradas da rede neural esteja associada a um rótulo ou classe. No que diz respeito as entradas, é necessário um entendimento sólido do domínio o qual ela representa, onde por muitas vezes é inevitável o conhecimento de especialistas da área em questão (GOH, 1995). Caso haja necessidade da redução da quantidade de informações presentes em cada entrada da rede neural, há investigações estatísticas usadas para distinguir as informações mais expressivas nas entradas da rede, como o estudo em (STEIN, 1993).

O algoritmo *backpropagation* tem uma premissa fundamental, para o treinamento (ou aprendizagem) que é a atualização ou ajuste dos pesos sinápticos, que armazenam o conhecimento obtido na tarefa, aplicados aos neurônios. Para exemplificar de maneira adequada esse funcionamento, o algoritmo é dividido em duas etapas.

Na etapa inicial, com pesos fixos, o fluxo de execução ocorre da esquerda para direita do MLP (Figura 6), passando da camada de entrada, seguindo de camada em camada oculta da rede e sempre computando o resultado da função de ativação que serve de entrada para a camada oculta seguinte até que as saídas da rede sejam obtidas. A essa etapa se dá a denominação adiante

(*forward*).

Na próxima e última etapa, que é denominada para trás (*backward*), é calculado o erro em relação a saída esperada (classe correta) e a saída da rede. Este erro é propagado por toda a rede, de maneira oposta a etapa adiante. O processo acaba quando os pesos sinápticos são atualizados refletindo o erro entre as saídas esperadas e as saídas obtidas pela rede neural.

3.3 Aprendizado profundo

3.3.1 Introdução

O principal objetivo da inteligência artificial é a resolução de problemas de extrema complexidade para seres humanos, os quais se tornam triviais para computadores ou máquinas. Por exemplo, na resolução de complexos problemas matemáticos que envolvam certas regras ou estratégias para sua resolução. Quando a situação é inversa, ou seja, óbvia para humanos (reconhecimento de faces, por exemplo), porém penosas para uma máquina, onde no que diz respeito as máquinas a principal dificuldade está inerente a criação de uma descrição formal da tarefa que se pretende realizar.

O aprendizado profundo (*deep learning*) vem com o intuito de contornar as situações, onde existe certa complexidade na realização da tarefa por uma máquina. Um algoritmo de *deep learning* visa gerar o aprendizado por meio da compreensão da hierarquia de conceitos (GOODFELLOW; BENGIO; COURVILLE, 2016).

Na hierarquia de conceitos a experiência do conhecimento é adquirida através do aprendizado de conceitos mais simples, para posteriormente aprender conceitos mais complicados (GOODFELLOW; BENGIO; COURVILLE, 2016). A hierarquia de conceitos pode ser entendida como um grafo profundo, ou seja, possui várias camadas que representam o processo de aprendizado. Essas diversas camadas são pontos fundamentais para os algoritmos de *deep learning*.

Um dos principais problemas encontrados em algoritmos de *machine learning* está na representação dos dados de entrada. Essa representação nada mais é do que a escolha de uma porção da informação (entrada do algoritmo), ou seja, algumas características (*features*) dos dados de entrada. Caso a escolha dessas características seja feita de uma maneira desatenta, pode gerar uma consequência negativa no algoritmo de *machine learning*. A representação dos dados e a escolha das características do mesmo é denominada aprendizado de representação (*representation learning*) (GOODFELLOW; BENGIO; COURVILLE, 2016).

Porém, o problema do aprendizado de representação é resolvido por *deep learning*, onde o algoritmo é capaz de traçar conceitos complexos (por exemplo, identificação de um objeto) através de conceitos simples (bordas, cantos, contornos, partes do objeto).

3.3.2 Rede neural convolucional

Em uma rede neural artificial (por exemplo, uma rede MLP) os dados de entrada estão em um formato de vetor de características unidimensional (1D). Essas variáveis de entrada podem ser representadas em qualquer ordem (desde que essa ordem seja respeitada por todas as entradas) com vetores de tamanho fixo sem afetar o processo de treinamento.

No que diz respeito a dados bidimensionais (imagens, por exemplo) as variáveis presentes na matriz espacial estão altamente correlacionadas, assim, qualquer tipo alteração nesses valores locais (pixels em imagens) gera um efeito significativo no treinamento da rede neural. Essa correlação local é utilizada para extrair e combinar as características locais para a realização da tarefa, como o reconhecimento. A extração de características é umas das propriedades fundamentais de uma rede neural convolucional (LECUN; BENGIO, 1995).

Uma rede neural convolucional (*convolutional neural network*, CNN) é um perceptron multicamadas cuja função é realizar o reconhecimento ou classificação de dados bidimensionais (também conhecida como topologia em grades), além disso esse tipo de rede possui alto grau de invariância à translação, escala, inclinações e outros tipos de distorções (HAYKIN, 2009). CNNs, de um modo mais específico, são redes neurais que utilizam a operação linear convolução no lugar da multiplicação de matrizes em ao menos uma de suas camadas e são largamente aplicadas a imagens.

Em algumas aplicações de classificação de imagens, é utilizada a transformação da imagem bidimensional em um vetor de uma única dimensão como entrada para a rede neural totalmente conectada. Porém, este mecanismo ignora uma propriedade fundamental de imagens, que diz que os pixels adjacentes são fortemente correlacionados que pixels mais distantes (BISHOP, 2006). Rede neurais convolucionais investigam de uma maneira mais adequada essa propriedade específica, através da extração de características locais que dependem de regiões menores da imagem.

Uma rede neural convolucional segue uma estrutura na qual ela lida com os dados de entrada, no caso de imagens, de maneira bastante específica. Essa estrutura principal de uma CNN é mostrada na Figura 7.

A primeira camada tem como entrada na rede neural uma imagem, que compreende dados bidimensionais formados por um conjunto de valores de intensidade de pixels, usualmente esses valores variam entre 0 e 255 níveis de cinza. A imagem pode ter uma única camada monocromática (escala de cinza) ou pode ser formada por uma composição de três imagens monocromáticas, que são as bandas vermelha, verde e azul (GONZALEZ; WOODS, 2008).

O núcleo de uma rede neural convolucional está na camada convolucional. Nela, ocorre a operação linear denominada convolução, que pode ser definida pela Equação 3.6. Nessa equação, a imagem de entrada como uma função espacial é representada por I, onde os índices (i, j) indicam a localização e I representa a intensidade dada na escala de cinza, que passa pela

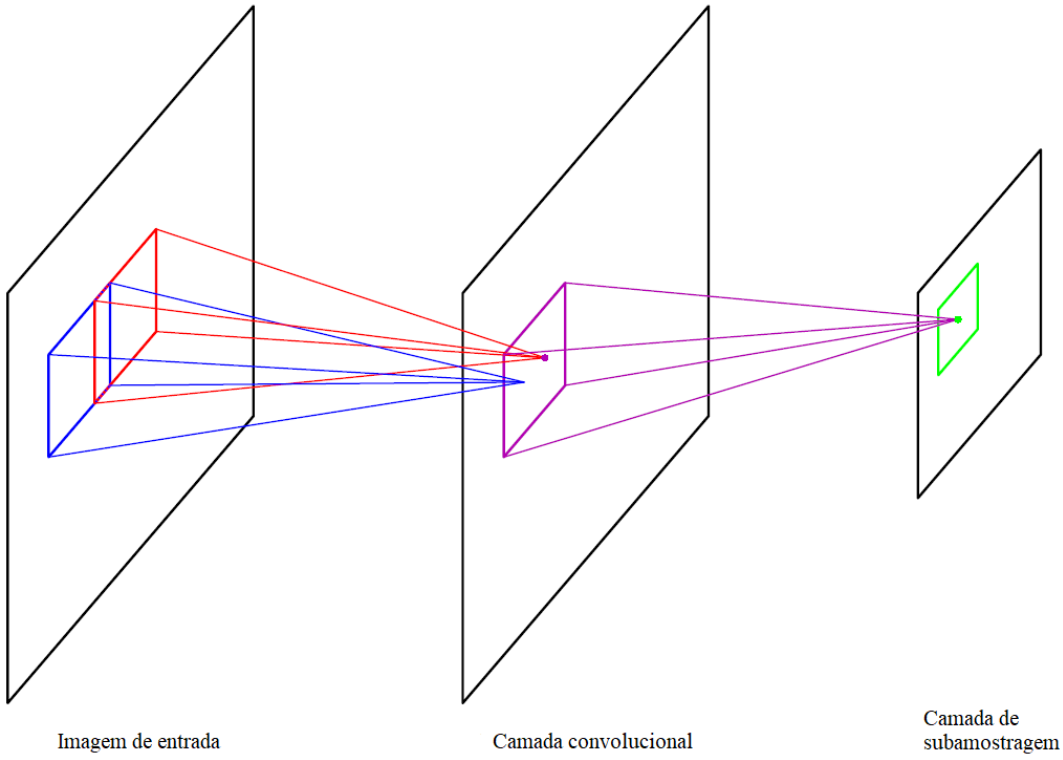


Figura 7 – Estrutura básica de uma rede neural convolucional. O elemento inicial da estrutura é composto pela camada de entrada, que corresponde a imagem de entrada a ser apresentada na camada seguinte que opera a convolução sobre a imagem, e por fim o resultante da convolução passa por um processo de subamostragem, representado pela última camada. Fonte: Adaptada de (BISHOP, 2006).

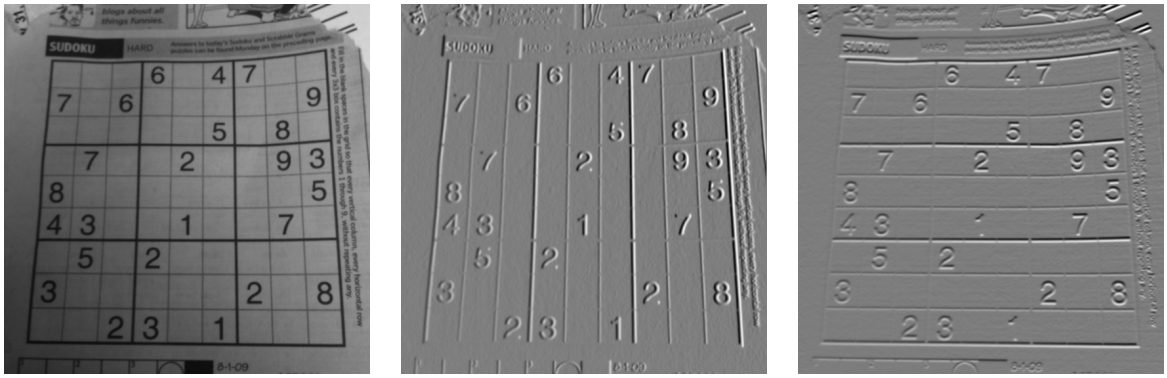
operação convolução com K que é o filtro ou *kernel* que é uma matriz de parâmetros ajustáveis pelo algoritmo de aprendizado. A saída da operação S é dada por unidades que são organizadas em planos, os quais são chamados de mapas de característica (*features maps*).

$$S(m,n) = \sum_i \sum_j I(i,j)K(m-i,n-j) \tag{3.6}$$

O *kernel* tem uma função fundamental na arquitetura de uma rede neural convolucional, que é extrair características visuais locais fundamentais da imagem de entrada, como bordas e contornos por exemplo. Desse modo, o algoritmo de aprendizagem é responsável por ajustar essa matriz de parâmetros para que este extraia as características que posteriormente serão utilizadas na classificação de imagens.

Durante a convolução, as unidades no mapa de características ou imagens de entrada são induzidas a realizarem a mesma operação com o *kernel* em cada porção da imagem através do compartilhamento dos pesos sinápticos, ou seja, a matriz de pesos (para determinado *kernel*) é fixa ou compartilhada em toda a imagem. Esta é uma das principais características de uma CNN, conhecida como compartilhamento de pesos (LECUN et al., 1998). Esse compartilhamento de pesos possui duas principais vantagens em sua utilização, que são: a invariância a translação pela

utilização da Equação 3.6 seguida de uma função de ativação; além da redução do número de parâmetros utilizados (HAYKIN, 2009).



(a) Imagem original **(b) Derivada x** **(c) Derivada y**

Figura 8 – Operador Sobel. A convolução da imagem original (a) com *kernels* de tamanho 3×3 para calcular a aproximação das derivadas para detecção de bordas considerando mudanças na horizontal (b) e vertical (c).

No processamento digital de imagens, há *kernels* comumente utilizados para detecção de contornos em imagens, como o operador ou filtro Sobel, que é utilizado como detector de contornos baseado em gradiente. O operador Sobel calcula as derivadas de primeira ordem dos eixos x e y, que representam as mudanças horizontais e verticais na imagem de origem. As Equações 3.7 e 3.8 mostram os *kernels* utilizados pelo operador Sobel para detecção de contornos, considerando mudanças nos eixos x e y, respectivamente.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{3.7}$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{3.8}$$

A Figura 8 demonstra a utilização do operador Sobel, onde a imagem original (Figura 8a) é aplicada aos *kernels* para detecção de bordas considerando as derivadas de primeira ordem em relação a x (Figura 8b) e y (Figura 8c), respectivamente. Quando se considera o contexto de redes neurais convolucionais, os *kernels* (Equações 3.7-3.8) correspondem a duas matrizes de parâmetros ajustáveis durante a aprendizagem e as imagens resultantes (Figuras 8b-8c) dessa operação representam dois diferentes mapas de características obtidos.

Ainda na camada convolucional da Figura 7, uma das funções de ativação não linear mais utilizadas em uma CNN é a função unidade linear retificada (*rectified linear unit*, ReLU) que é dada pela Equação 3.9. Onde x é a entrada e $f(x)$ é a saída da função de ativação. Na função ReLU todo valor menor que zero passará a ser igual a zero e todo valor maior ou igual

a zero será igual ao valor de entrada, como é mostrado na curva sólida da Figura 9. A função ReLU também pode ser denominada de função rampa.

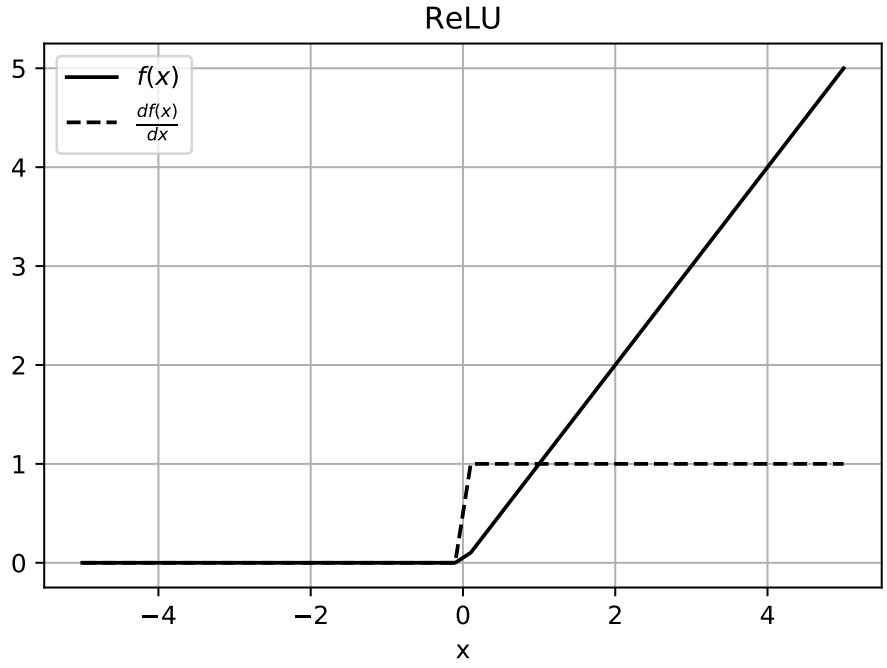


Figura 9 – Função de ativação não linear ReLU. As variáveis x e y são as entradas e saídas da função, respectivamente. Os valores das entradas variam entre -5 e 5 para melhor exemplificar a função.

Uma das diferenças essenciais da função de ativação ReLU se dá pela velocidade no treinamento da rede neural que faz uso da ReLU quando comparado às outras funções de ativação, como a logística (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Essa característica fundamental se dá pois a função ReLU possui uma derivada consideravelmente simples, o degrau, onde a saída da função $\frac{df(x)}{dx}$ é igual a 1 quando a entrada é maior ou igual a 0, e saída 0 caso contrário, como mostram a Equação 3.10 e a curva tracejada da Figura 9. Essa derivada da função de ativação torna a utilização de um algoritmo de aprendizagem (como, o *backpropagation*) combinado a um otimizador (abordado na subseção 3.3.3) rápida e ao mesmo tempo robusta durante o processo de treinamento para se obter a atualização dos pesos sinápticos da rede neural convolucional.

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3.9}$$

$$\frac{df(x)}{dx} = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3.10}$$

No que diz respeito as dimensões de saída dos mapas de características na camada convolucional, segundo o estudo da aritmética da convolução aplicada ao *deep learning* mos-

trado em (DUMOULIN; VISIN, 2018) as dimensões resultantes podem ser calculadas pela Equação 3.11.

As dimensões de entrada e a saída são denominados de n_{input} e n_{output}, respectivamente. A variável p controla a adição ou preenchimento (*padding*) de zeros nas bordas dos mapas de características, a dimensão do *kernel* é representada por k e por fim s contém o valor do passo (ou *stride*), que é o número de pixels descolados na matriz de entrada durante a convolução com o *kernel*. O *stride* é comumente definido por um número inteiro, porém é também encontrado na literatura através da representação de uma matriz, como por exemplo um *stride* de tamanho 2 pode ser representado por uma matriz de tamanho 2×2, indicando o deslocamento matricial realizado através da matriz de entrada.

$$n_{output} = \left[\frac{n_{input}+2p-k}{s}\right] + 1 \tag{3.11}$$

Considerando que uma imagem é formada por uma matriz de M linhas (altura) por N colunas (largura) e que M e N sejam diferentes, as dimensões de entrada e saída serão individuais para cada dimensão, essa premissa também é válida quando as dimensões do *kernel* e *stride* forem diferentes. Por exemplo, uma imagem com resolução de entrada 30×62, sem a utilização de *padding* ($p = 0$), com um *kernel* e um *stride* de tamanho 2×2 tem uma saída com as dimensões 15×31 ($15 = \left[\frac{30+2\cdot0-2}{2}\right] + 1$ e $31 = \left[\frac{62+2\cdot0-2}{2}\right] + 1$).

O último dos principais componentes presentes em uma CNN, mostrado na Figura 7, é camada de subamostragem (*subsampling*) ou *pooling*. Esta camada é responsável pela redução da resolução espacial do mapa de características através de uma determinada função (LECUN et al., 1998), reduzindo assim a sensibilidade da saída a deslocamento e distorções na entrada da camada.

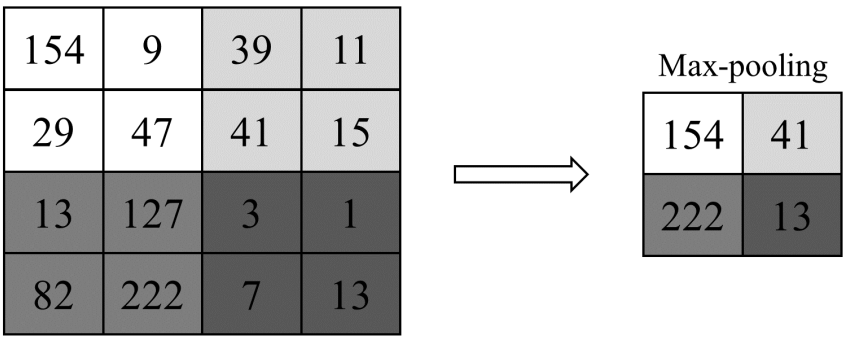
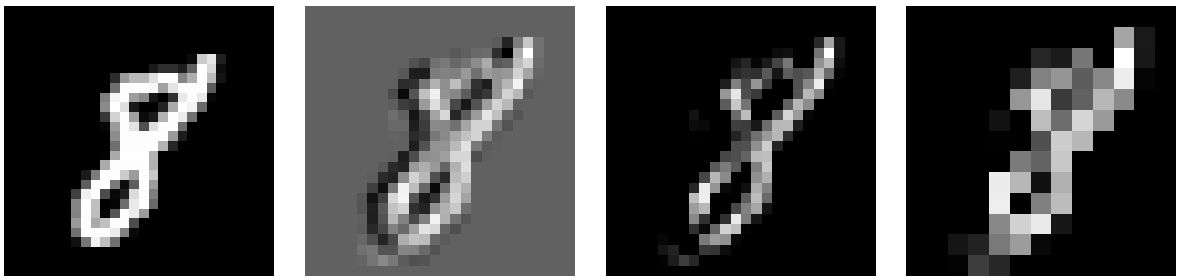


Figura 10 – Exemplo de utilização do max-pooling. A entrada é uma mapa de características de resolução 4×4, a janela ou pool possui o tamanho 2×2 e o stride de descolamento de tamanho 2×2.

Uma das funções empregadas na camada de *pooling* mais utilizadas em rede convolucionais é a função máximo, comumente denominada de *pooling* máximo ou *max-pooling*. Ela reduz a dimensionalidade dos mapas de características da camada anterior passando o máximo local dentro do tamanho da janela (que pode ser denominada também por *pool*) utilizada. A

Figura 10 exemplifica o *max-pooling*, onde o mapa de características (à esquerda) de tamanho 4×4 é subamostrado por um *pool* de tamanho 2×2 com um passo (*stride*) de deslocamento de tamanho 2×2. O resultante da operação é mostrado à direita da Figura 10, onde cada elemento deste representa o máximo local de cada um dos deslocamentos realizados pelo *pool*, que são representados pelos níveis de cinza na Figura 10. Outras funções podem ser utilizadas na camada de *pooling*, como a função média (*average pooling*), que realiza a redução de dimensionalidade através da média dos valores dentro do *pool*.



(a) Entrada **(b) Convolução** **(c) ReLU** **(d) Max-pooling**

Figura 11 – Mapa de característica aprendido por uma rede neural convolucional que utiliza a base de dados de reconhecimento de dígitos escritos a mão. As Figuras (b), (c) e (d) apresentam, respectivamente, o mesmo mapa de características da entrada representada pela imagem do dígito oito (a) após convolução, função de ativação ReLU e max-pooling.

A Figura 11 demonstra as operações feitas nas camadas convolucional e *max-pooling* de uma imagem da base de dados[1] utilizada no reconhecimento de dígitos escritos à mão feito em (LECUN et al., 1998). A imagem de entrada (Figura 11a) possui uma resolução 28×28, a Figura 11b mostra a saída (mapa de característica) da camada convolucional com um *kernel* de tamanho 3×3, *stride* 1 e sem preenchimento de zeros que produz uma imagem de resolução 26×26 resultante da Equação 3.11, enquanto que a Figura 11c representa a imagem pós convolução (Figura 11b) aplicada a função de ativação ReLU e por fim a Figura 11d representa a Figura 11c após a redução de dimensionalidade da camada *max-pooling* (*kernel* e *stride* de 2×2), com uma resolução de 13×13.

3.3.3 Treinamento

Ao se utilizar uma arquitetura com diversas camadas convolucionais, seguidas de outras camadas de subamostragem, os mapas de características obtidos por estas operações são usualmente transformados em um vetor de características, que são conectados a um perceptron de múltiplas camadas até os resultados obtidos pela rede serem aplicados a função que tem como resultado a distribuição de probabilidade das classes a serem avaliadas pela CNN.

A Figura 12 mostra a transformação de um mapa de característica de tamanho 2×2 resultante das operações de subsecivas camadas convolucionais e de *pooling* em um vetor de

[1] Base de dados disponível em: <http://yann.lecun.com/exdb/mnist/>

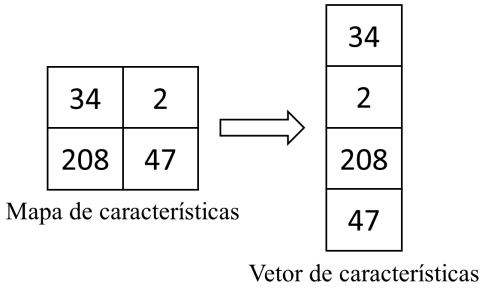


Figura 12 – Transformação de um mapa de características de tamanho 2×2 em um vetor de uma dimensão de tamanho 4.

características unidimensional de tamanho 4 utilizado como entrada de um MLP. Em uma arquitetura de CNN existem diversos (a Figura 12 apresenta apenas um) mapas de características, desse modo, o vetor de características de entrada do MLP é composto pela junção dos vetores de características dos mapas de características após as camadas convolucionais e de *pooling*.

A função de ativação comumente utilizada no final da arquitetura de uma rede neural convolucional é a *softmax*, que é representada na Equação 3.12, onde a função *softmax* $\sigma(y_i)$ representa a distribuição de probabilidades sobre as J diferentes possibilidades de classe de saída y_i da rede MLP. Desse modo, o função *softmax* limita as saídas da rede entre os valores 0 e 1, onde a soma dos valores de saída (probabilidade de pertencer a uma determinada classe) após *softmax* é igual a 1.

$$\sigma(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{J} e^{y_j}} \tag{3.12}$$

Para comparação entre saída obtida e saída esperada da rede neural, é utilizada a entropia cruzada (*cross-entropy*) dada pela Equação 3.13. Os rótulos das classes (saída esperada) são transformados para uma codificação onde um vetor de tamanho igual ao tamanho J que possui valores de zeros em quase todas as posições (indicada pelo índice i), com exceção da posição referente a classe específica, que é denominada por R. O logaritmo na Equação 3.13 é o logaritmo natural, que é usualmente chamado de logaritmo neperiano.

$$C(\sigma(y), R) = -\sum_{i} R_i log(\sigma(y_i)) \tag{3.13}$$

Para exemplificar o funcionamento das Equações 3.12 e 3.13, tem-se a saída da rede (composta por três neurônios), relacionada a uma única entrada, posta como um vetor linha $y = \begin{bmatrix} 3 & 4 & 5 \end{bmatrix}$, onde esta possui tamanho $J = 3$ e cada elemento no vetor é representado por i e a um único neurônio. A saída y é a entrada da função *softmax* $\sigma(y)$, cujo resultado com a utilização da Equação 3.12 é dado por $\sigma(y) = \begin{bmatrix} 0.0900 & 0.2447 & 0.6653 \end{bmatrix}$ que representa a distribuição de probabilidade das três possíveis classes de saída. Na posição $i = 3$ de $\sigma(y)$, nota-se o maior valor presente neste vetor o que indica uma probabilidade relativamente alta desta determinada entrada da rede neural convolucional ser referente a classe 3. Para determinar entropia cruzada

(Equação 3.13), é necessário que se tenha a saída esperada da rede determinada pelo rótulo $R = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$, cujo elemento $i = 3$ monstra que essa entrada pertence a classe 3. O valor de $C(\sigma(y), R)$ é obtido através do somatório da operação da Equação 3.13 elemento a elemento dos vetores σ e R, cujo resultado é $C(\sigma(y), R) = 0.4075$ $(-0 \cdot log(0.0900) - 0 \cdot log(0.2447) - 1 \cdot log(0.6653))$.

$$L = \frac{1}{N} \sum_{j} C(\sigma(y)_j, R_j) \tag{3.14}$$

Com a comparação entre a saída obtida pela rede e a saída desejada, através da entropia cruzada da Equação 3.13, é possível obter a função de custo que será minimizada pelo algoritmo de aprendizagem. Aqui essa função de custo é chamada de função de perda (*loss function*) que representa a média da entropia cruzada de todos os dados de treinamento utilizados, como mostra a Equação 3.14. A letra j no somatório representa cada um dos dados utilizados na aprendizagem.

A otimização por descida do gradiente (*gradient descent*) é um dos métodos largamente utilizados para minimizar a função de custo (perda) da Equação 3.14. Considerando que tanto a saída y da rede neural quanto a função de custo dependem dos valores dos pesos sinápticos w, o intuito da minimização é obter o menor valor de entropia cruzada entre as saídas com base em determinados valores para os conjuntos de pesos.

A descida do gradiente é um método que utiliza a informação do gradiente (função cujo componente são derivadas parciais de determinadas variáveis) da função de perda para realizar a atualização dos pesos sinápticos executando pequenos passos na direção negativa do gradiente (BISHOP, 2006), como mostra a Equação 3.15.

$$w^{(\tau+1)} = w^{(\tau)} - \eta \bigtriangledown L(w^{(\tau)}) \tag{3.15}$$

O valor de τ diz respeito ao passo atual da atualização dos pesos sinápticos w, onde cada passo processará todo o conjunto de dados de treinamento para estimar $\bigtriangledown loss$. A taxa de aprendizagem (*learning rate*) a qual essa atualização é aplicada é denominada de η, cujos valores tendem a ser maiores que 0. Os métodos que todo o conjunto de treinamento é utilizado para o aprendizado do modelo são denominados métodos em lote (*batch*).

Por outro lado, existem algoritmos que são aplicados de maneira sequencial no que diz respeito a aprendizagem. Este algoritmo é uma variação da descida do gradiente, que é chamado de descida do gradiente estocástica (*stochastic gradient descent*, SGD) que também é conhecido como atualização *on-line* e se demonstra bastante útil na utilização de grandes conjuntos de dados (LECUN et al., 1998).

No SGD os pesos sinápticos são atualizados utilizando um dado ou lotes (*batchs*) pequenos de dados aleatórios do conjunto por vez. Além disso, a função de perda (Equação 3.14)

é calculada somando-se cada uma das perdas obtidas pelas amostras utilizadas. A otimização por SGD tem duas principais vantagens, a primeira é em relação a presença de dados redundantes no conjunto de treinamento, contornado devido a utilização de uma pequena quantidade dados aleatórios. A outra vantagem é em relação aos mínimos locais, que são evitados de uma maneira superior em comparação à descida do gradiente padrão (BISHOP, 2006).

3.4 Transformada discreta de Fourier

3.4.1 Introdução

Quando o assunto é o processamento digital de imagens, logo se pensa na transformada de Fourier e seu modo de se trabalhar em outro domínio (frequência) que não seja o domínio da espacial. O domínio da frequência da transformada de Fourier pode ser utilizado para consideráveis fins, como no projeto e desenvolvimento de filtros em áreas que visam o realce e restauração de imagens (GONZALEZ; WOODS, 2008).

O matemático francês Jean Baptiste Joseph Fourier tem como uma de suas principais contribuições a comprovação que qualquer função periódica, não importa o quão complexa seja a função, pode ser expressa como somas de senos e/ou cossenos de diferentes frequências que são multiplicadas por diferentes coeficientes, cuja soma hoje é denominada como série de Fourier.

Também podem ser representadas no domínio de frequência as funções não periódicas, desde que sua área sobre a curva seja finita. A representação de funções não periódicas podem ser externadas como uma integral de senos e/ou cossenos multiplicados por uma função de pesos. A formulação dessa integral é definida como transformada de Fourier, o qual opera sobre uma função contínua.

Tanto a série quanto a transformada de Fourier possuem uma peculiaridade fundamental, que é a capacidade de recuperar (ou reconstruir) completamente a função original através de um processo inverso sem nenhum tipo de perda de informação (GONZALEZ; WOODS, 2008). Ou seja, é possível trabalhar no domínio da frequência (ou domínio de Fourier) e então retornar para o domínio original da função.

3.4.2 Domínio da frequência

Considerando que neste trabalho lida-se com imagens digitais, é necessário investigar a transformada discreta de Fourier (*discrete Fourier transform*, DFT) em duas dimensões (2-D). A transformada discreta de Fourier de uma imagem dada por uma função $f(x,y)$ é definida pela Equação 3.16, a qual representa a imagem $f(x,y)$ do domínio espacial no domínio da frequência $F(u,v)$.

Os valores de x e y representam as coordenadas espaciais da imagem, cujos valores variam, respectivamente, entre 0 e M-1 e 0 e $N-1$, onde $M \times N$ remete a resolução da imagem.

Os valores associados a x e y representam a posição de determinado nível de intensidade de cinza em $f(x,y)$. As frequências u e v são definidas como variáveis (ou coordenadas) de frequência que variam entre os valores 0 a $M-1$ e 0 a $N-1$, respectivamente. Por fim, $F(u,v)$ caracteriza a imagem no domínio da frequência.

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(\frac{ux}{M}+\frac{vy}{N})} \tag{3.16}$$

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(\frac{ux}{M}+\frac{vy}{N})} \tag{3.17}$$

É possível reverter o sistema de coordenadas para o domínio de origem sem nenhum tipo de perda de informações, como explicado anteriormente. Esse retorno é factível através da inversa da transformada discreta de Fourier (*inverse discrete Fourier transform*, IDFT), como mostra a Equação 3.17. Usualmente a IDFT é utilizada quando se aplica uma filtragem no domínio da frequência, visando por exemplo o realce da imagem quando a reversão de um sistema para o outro (domínio da frequência para o espacial) é aplicada.

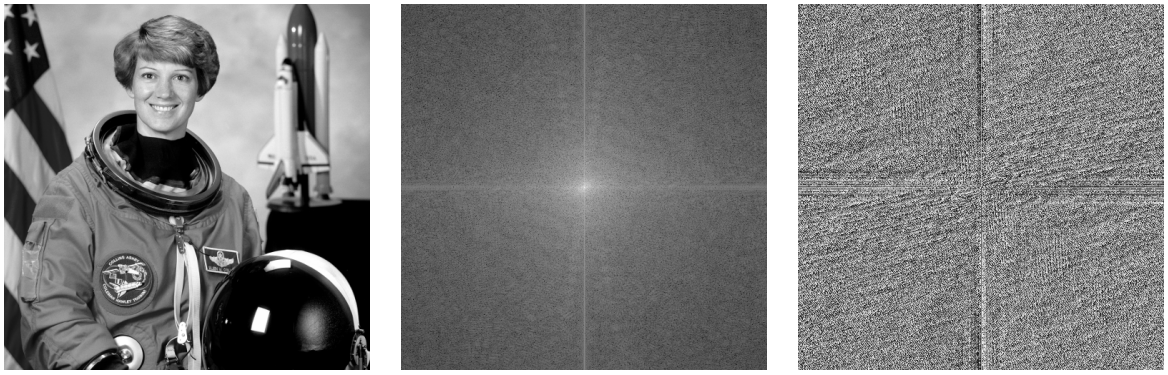
Os componentes da transformada de Fourier (Equação 3.16) são quantidades complexas, e por vezes, na investigação de números complexos, é apropriado trabalhar $F(u,v)$ em termos de coordenadas polares, que são obtidas a partir de suas partes real e imaginária. Assim, é possível trabalhar a DFT de dois aspectos: através da magnitude ou espectro (Equação 3.18) e fase ou espectro de fase (Equação 3.19), onde $R(u,v)$ é componente real e $I(u,v)$ é o componente imaginário da transformada discreta de Fourier.

$$|F(u,v)| = \left[R^2(u,v) + I^2(u,v)\right]^{\frac{1}{2}} \tag{3.18}$$

$$\phi(u,v) = tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right] \tag{3.19}$$

Para demonstração da transformada discreta de Fourier utilizou-se a imagem convertida para a escala de cinzas, com resolução de 512×512 da astronauta norte-americana Eileen Collins (retirada do domínio público Flickr[2]), como mostra a Figura 13a. A imagem da Figura 13a representa a função $f(x,y)$ no domínio espacial da Equação 3.16, onde M e N são iguais a 512. Por outro lado, as Figuras 13b e 13c apresentam a imagem da astronauta no domínio de Fourier (frequência) através de sua magnitude $|F(u,v)|$ e fase $\phi(u,v)$, respectivamente, onde as imagens no domínio da frequência mantém a resolução de 512×512 da imagem do domínio de origem.

[2] <https://www.flickr.com/photos/nasacommons/16504233985/>



(a) Imagem em escala de cinza **(b) Magnitude da imagem** **(c) Fase da imagem**

Figura 13 – Demonstração da transformada discreta de Fourier. A imagem em escala de cinzas (a) de resolução 512×512 no domínio espacial e sua magnitude (b) e fase (c) da DFT de resolução 512×512 no domínio da frequência.

4 METODOLOGIA

A metodologia de pesquisa utilizada no decorrer desse trabalho é apresentada neste capítulo. O problema da inspeção do componente do vagão do trem é especificado para o melhor entendimento. A base de dados e a arquitetura de rede utilizada são especificadas para a criação dos modelos para os experimentos realizados no trabalho de classificação quanto ao defeito do pad. As métricas utilizadas para validar a qualidade dos modelos criados são introduzidas. E por fim o cronograma de atividades a serem realizadas é apresentado.

4.1 Elementos da metodologia de pesquisa

Com o objetivo de desenvolver a pesquisa e verificar os benefícios da utilização de *deep learning* no contexto da ferrovia definiu-se a seguinte questão que conduzirá o trabalho no decorrer da pesquisa: "No escopo da inspeção de componentes do vagão do trem, especificamente o pad, quais os principais benefícios da abordagem proposta que tem como premissa a utilização de técnicas de *deep learning*?".

Segundo (POLIT; HUNGLER, 1999), pesquisa exploratória é um estudo preliminar que tem como propósito desenvolver ou criar hipóteses ou testar e definir métodos de coleta de dados. Assim, o tipo da pesquisa utilizada neste trabalho é do tipo exploratória, pois, nela pretende-se explorar a inspeção de componentes presentes no escopo ferroviário através de técnicas de *deep learning*, assunto pouco comum ou encontrado de maneira não frequente na literatura, como mostra o Capítulo 2, que apresenta a revisão sistemática da literatura feita neste trabalho.

Para orientar a pesquisa do trabalho à geração de conhecimentos científicos (ou válidos) faz-se necessário delimitar o método científico utilizado no decorrer do trabalho, cuja definição é: “Método científico pode ser definido como um conjunto de regras básicas para realizar uma experiência, de modo a produzir um novo conhecimento, bem como corrigir e integrar conhecimentos preexistentes” (VIANA, 2001). Dentre os métodos científicos, optou-se pelo uso do método experimental. Essa escolha se dá pela necessidade de realizar experimentos com técnicas de *deep learning*, especificamente redes neurais convolucionais, para a verificação da existência de defeitos no pad. Porém, como o trabalho tratará de medir estatisticamente cada uma das técnicas utilizadas, este também possui traços do método científico estatístico.

Quanto à abordagem de pesquisa usada no decorrer do trabalho ficou decidido a utilização da abordagem quantitativa. Essa escolha está no fato de que os resultados obtidos desta abordagem de pesquisa podem ser quantificados, além disso, esta abordagem tem como foco a objetividade. Neste caso, os resultados quantificados serão medidas estatísticas de acurácia, precisão, *recall*, entre outras, para aferir cada um dos modelos ou classificadores a serem testados.

Partindo do fato da utilização da abordagem quantitativa, os procedimentos realizados são de uma pesquisa experimental, na qual, segundo (GIL, 2007), é fundada em determinar

um objeto de estudo, escolher as variáveis que o influenciam, além de delimitar as formas de controle e de observação dos efeitos que a variável gera no objeto. No contexto deste trabalho, o objeto seria o modelo ou classificador que detectará o possível defeito no componente pad do vagão do trem.

No que diz respeito a base de dados utilizada no trabalho, que são imagens que demonstram os três possíveis estados do componente do trem a ser avaliado, escolheu-se a técnica observação estruturada para a coleta de dados. Nesta técnica, faz-se a especificação do que deve ser observado, além de como deve ser feito seu registro. No trabalho, especificamente, o que deve ser observado é o pad presente do truque do vagão do trem, cujo registro é feito através de uma câmera. Desse modo, coleta de dados que compõe a base de dados utilizada no desenvolvimento do trabalho é realizada.

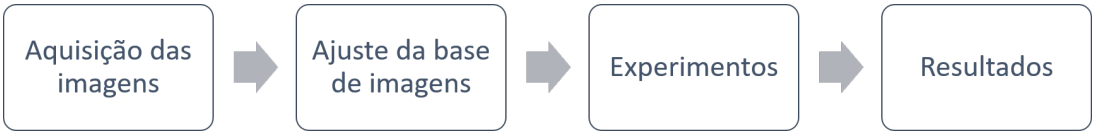


Figura 14 – Ciclo de operação da pesquisa, formado pela aquisição das imagens, ajuste da base de dados gerada, os experimentos realizados e por fim os resultados obtidos por estes.

Após a geração da base de dados de imagens do pad, será realizada sua análise para a posterior separação (rotulação das imagens do pad) da base a fim de refletir as três classes as quais o pad poderá ser classificado e que consideram as possíveis situações de ocorrência do componente. A saber, as classes são: pad ausente, pad normal, e pad danificado

Em seguida essas imagens são preparadas ou ajustadas, como por exemplo, o aprimoramento das imagens que servirão de entrada para a rede neural. Por fim, serão feitos testes com a técnica de *deep learning*, rede neural convolucional, com a base de dados já rotulada e ajustada para avaliá-las.

A Figura 14 mostra o ciclo da pesquisa, que é formado pelos elementos aquisição das imagens (coleta das imagens), ajuste da base de dados composto em duas etapas, a primeira onde as imagens são separadas nas três classes que o componente pode ser encontrado e a segunda que realiza algum tipo de pré-processamento, como o aprimoramento da imagem. A etapa de experimentos, que será expandida no decorrer do trabalho, é onde são feitos os testes para a classificação de *pads* com variações dos parâmetros da rede neural e pré-processamentos utilizados. E por fim os resultados encontrados em cada um dos experimentos.

4.2 Inspeção de componentes do vagão do trem

4.2.1 Identificação do problema

O vagão ferroviário ou vagão do trem é um veículo utilizado para o transporte de cargas ou passageiros em um sistema de transporte ferroviário o qual é impulsionado por uma

locomotiva, que tem como função realizar a força necessária para pôr o trem em movimento.

O vagão ferroviário é um dos bens ativos (patrimônio da empresa) mais representativos na logística da operação ferroviária, o qual necessita de uma manutenção cuidadosa, que é feita por meio da inspeção deste ativo. O termo inspeção diz respeito ao ato de examinar ou verificar o item de interesse ou componente com o objetivo de detectar possíveis problemas ou irregularidades. E em muitas companhias ferroviárias as inspeções são realizadas visualmente, isto é, o inspetor técnico deve avaliar, em um curto tempo, diversos itens.

Por exemplo, na Vale S.A o processo de inspeção visual ocorre na área que está localizado virador de vagões. O virador é um sistema no qual ocorre o processo de descarga de um bloco de 110 vagões contendo toneladas de minério. Essa descarga é realizada contendo dois vagões por vez em cerca de 80 segundos. Nesse tempo a inspeção visual é realizada por dois inspetores (um do lado direito e outro do lado esquerdo de cada vagão) que necessitam avaliar pouco mais de 60 itens (adaptador da roda, mola do truque ferroviário e parafusos do rolamento), onde eles verificam possíveis defeitos nesses componentes, e em caso afirmativo, os vagões serão destinados para a oficina de manutenção para a correção.

No entanto, a realização desta atividade tem demonstrado não ser eficiente, devido à dificuldade de identificar visualmente uma considerável quantidade de itens em um período reduzido. Além disso, o ambiente próximo aos vagões representa um risco ao inspetor, uma vez que para realização da inspeção o inspetor necessita estar próximo aos trilhos os quais os trens são transportados de modo a estarem suscetíveis a acidentes de diferentes gravidades.

4.2.2 Contextualização

Inspecionar coisas ou objetos de interesse é uma tarefa importante em várias áreas, como, diagnóstico médico, fabricação e vigilância automática, diagnóstico técnico, veículos autônomos e orientação de robôs (BROSNAN; SUN, 2002).

A inspeção automática é frequentemente usada na indústria a fim de garantir a qualidade do produto, permitindo a correção de problemas e o descarte de produtos danificados. Esses sistemas fornecem uma avaliação rápida, econômica, consistente e objetiva dos itens a serem avaliados (SUN, 2000).

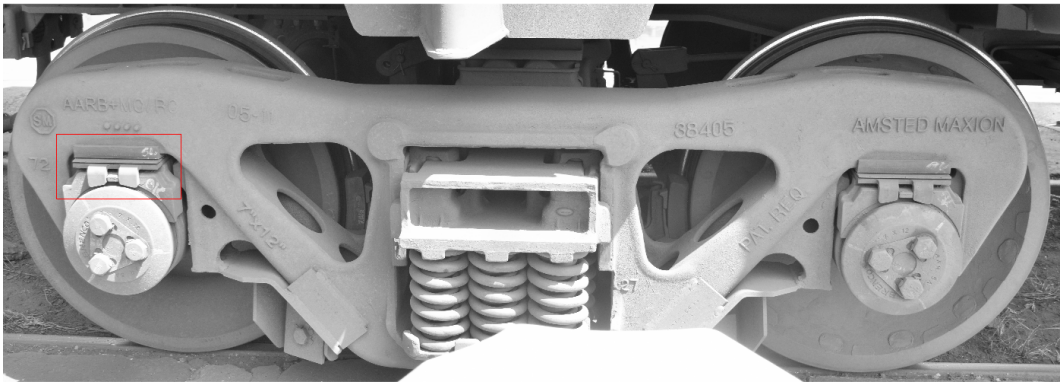
A inspeção é usualmente feita a olho humano, porém, possui acurácia não garantida, consumo de tempo excessivo e trabalho intenso. Além disso, a inspeção visual tem outras desvantagens, incluindo a falta de objetividade, propensão ao erro devido à distração, estresse ou fadiga e divergência entre diferentes técnicos que realizam essa função (PARK et al., 1996).

No que diz respeito a inspeções visuais em ferrovias, há ainda outros agravantes nesta tarefa. Isto se dá pela necessidade da locomoção do vagão para um local especialmente equipado para a realização da inspeção, devido ao componente a ser inspecionada estar situado abaixo do vagão. Este trabalho requer um tempo adicional, um trabalho complexo para sua realização e

além dos riscos aos quais os inspetores estão expostos (HART et al., 2008).

4.2.3 Especificação do problema

O truque ferroviário (Figura 15a) é uma estrutura localizada abaixo do vagão do trem ao qual os eixos (por consequência, as rodas) são fixados através dos rolamentos. Dois truques são usualmente equipados por vagão. Tipicamente cada vagão é composto por truques de 4 rodas que fornece suporte ao corpo do veículo e que é usado para fornecer sua tração e frenagem (IWNICKI, 2006).



(a) Truque ferroviário

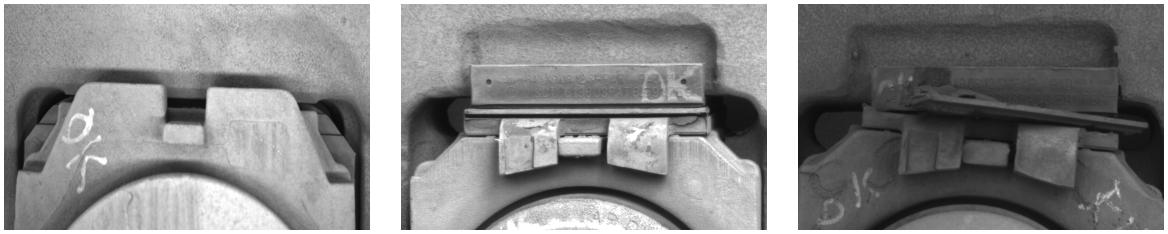
(b) Componente estrutural pad

Figura 15 – Visão de um dos dois truques presentes em um vagão ferroviário e o componente analisado neste trabalho. O retângulo vermelho ao lado esquerdo de (a) destaca o componente de interesse, que é ampliado e melhor detalhado em (b). O pad (b) é formado por três estruturas, os encaixes que fixam o componente a roda, a estrutura central de borracha e as estruturas de metal que fixam a estrutura de borracha.

Este trabalho propõe a utilização de técnicas de aprendizado profundo visando a inspeção automática de componentes do vagão ferroviário através de imagens. Dado o variado número de componentes do vagão e o tipo defeito que possuem, o componente de interesse será o *pad*, que é usado para dar suporte aos quadros laterais do truque no conjunto de rodas. O *pad* consiste em um polímero inserido sobre o adaptador do rolamento e desempenha um papel importante na dinâmica do vagão ferroviário como suspensão primaria (IWNICKI, 2006). O *pad* é composto por metal e borracha e funciona similar a um amortecedor. A Figura 15b mostra o recorte do *pad* presente no truque ferroviário da Figura 15a.

A inspeção do *pad* tem como foco os danos relacionados a quebra ou ruptura do componente, em que se observa o deslocamento de parte ou partes dele em relação à sua posição

esperada. Para este trabalho utiliza-se as três situações em que o *pad* pode se encontrar na ferrovia, que são: *pad* ausente (Figura 16a), *pad* não danificado (Figura 16b) e *pad* danificado (Figura 16c). Em particular, a situação que o *pad* se encontra ausente decorre da diferença entre projetos de vagões de diferentes fabricantes.



(a) Pad ausente **(b) Pad não danificado** **(c) Pad danificado**

Figura 16 – Três situações que o componente estrutural pad pode ser encontrado. Essas situações caracterizam as três classes de imagens utilizadas neste trabalho, que são: ausente (a), não danificado (b) e danificado (c).

4.3 Base de dados

As imagens que compõem a base de dados é formada por imagens do componente analisado (*pad*) de diferentes níveis de intensidade, iluminação e contraste, já que es imagens foram adquiridas em diferentes condições climáticas e períodos do dia, além da imagem estar sujeita a poeira e lama do local em que é capturada, dificultando assim a qualidade da imagem obtida que forma a base de dados.

As imagens adquiridas para a formação da base de dados estão em escala de cinza. Os valores de níveis de cinza nas imagens variam entre 0 a 255, onde 0 representa o tom de cinza mais escuro e 255 o mais claro. Optou-se por trabalhar com imagens em escala de cinza (monocromática) para reduzir o trabalho computacional realizado pela rede convolucional, o qual seria mais custoso caso fosse utilizada uma composição de bandas monocromáticas.

As imagens dos *pads* adquiridos variam entre resoluções 40 a 60 pixels de altura e 80 a 90 pixels de largura por isso as imagens foram redimensionadas para a resolução de 32×64 (altura x largura). Essas dimensões são definidas para que as imagens não percam elementos significativos, o que ocorreria se uma resolução menor fosse escolhida, e não sejam acrescentados elementos (pixels) que não existiam anteriormente nas imagens, o que aconteceria se fosse utilizada uma resolução maior.

Tabela 4 – Número de imagens por classe da base de dados utilizada, além da porcentagem que estas ocupam na base de dados total.

	Classe 1	Classe 2	Classe 3	Total
Número de imagens	651	644	681	1976
Porcentagem	32,94%	32,60%	34,46%	100%

A base de dados é formando por um total de 1976 imagens, que são divididas em três classes que representam as condições que o *pad* pode ser encontrado na operação, a saber: classe 1 (Figura 16a), que é formada por imagens que representam o *pad* ausente com um número de 651 imagens; classe 2 (Figura 16b), que é composta por imagens do *pad* não danificado, com 644 imagens; e a classe 3 (Figura 16c) é referente as imagens do *pad* danificado, com um total de 681 imagens relativas a esta classe.

A Tabela 4 apresenta a quantidade de imagens por classes presentes na base de dados. Além disso, é apresentada as porções que estás imagens ocupam na base de dados utilizada, mostrando uma porcentagem de $32,94\%$, $32,60\%$ e $34,46\%$ de imagens das classes 1, 2 e 3, respectivamente. Dessa maneira, é possível notar um considerável nível de balanceamento da quantidade de imagens por classe.

4.4 Técnicas de processamento de imagens

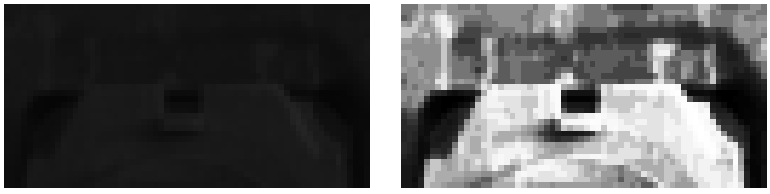
4.4.1 Equalização de histogramas

A equalização de histograma vem sendo utilizada na literatura para melhoria de performance em trabalhos de reconhecimento de imagens de sinais de trânsito. O trabalho feito em (CIREŞAN et al., 2012) redes neurais profundas (*Deep Neural Network*, DNN) e equalização de histogramas como pré-processamento dos dados de entrada para aumentar a performance de reconhecimento, além de tornar o reconhecimento de sinais de trânsito insensível a variações de contraste e iluminação. Esta abordagem alcança uma taxa de acerto de $99,46\%$, o que representa uma taxa de acerto maior que a taxa de reconhecimento humano de $93,31\%$.

Para tornar a classificação do *pad* insensível ao contraste, iluminação e a outros ruídos gerados pelo ambiente de captura das imagens (como mostrado na Seção 4.3) é utilizada a equalização de histograma. No processamento de digital de imagens, as imagens com intensidade escura possuem os pixels com valores de nível de cinza próximos ao 0, imagens de intensidades mais claras têm pixels com valores ao redor de 255 e imagens com baixo contraste que possuem valores de níveis de cinza em torno da metade do valor máximo (255), o qual a maioria dos pixels ficam ao redor do valor 128 na escala de cinza.

$$E_k = \sum_{j=0}^{k} \frac{n_j}{n} \tag{4.1}$$

Devido à diferença de intensidades e contraste entres as imagens da base de dados, é possível utilizar a equalização de histograma para aumentar o contraste das imagens para que os valores de níveis de cinza compreendam a maior parte da faixa de valores, que neste caso são valores entre 0 e 255 (GONZALEZ; WOODS, 2008).



(a) Imagem original **(b) Imagem equalizada**

Figura 17 – Equalização de histograma de uma imagem do componente capturada durante a noite. A imagem (a) de níveis de cinza mais baixos (intensidade escura) e sua respectiva imagem equalizada (b).

A Equação 4.1 apresenta a equalização de histograma, na qual a imagem de saída é obtida pelo mapeamento de cada pixel de determinado nível de cinza em um nível de cinza correspondente E_k, onde k varia entre 0 e 255, n é o número total de pixels na imagem, e n_k é o número de pixels que tem determinado nível de cinza.

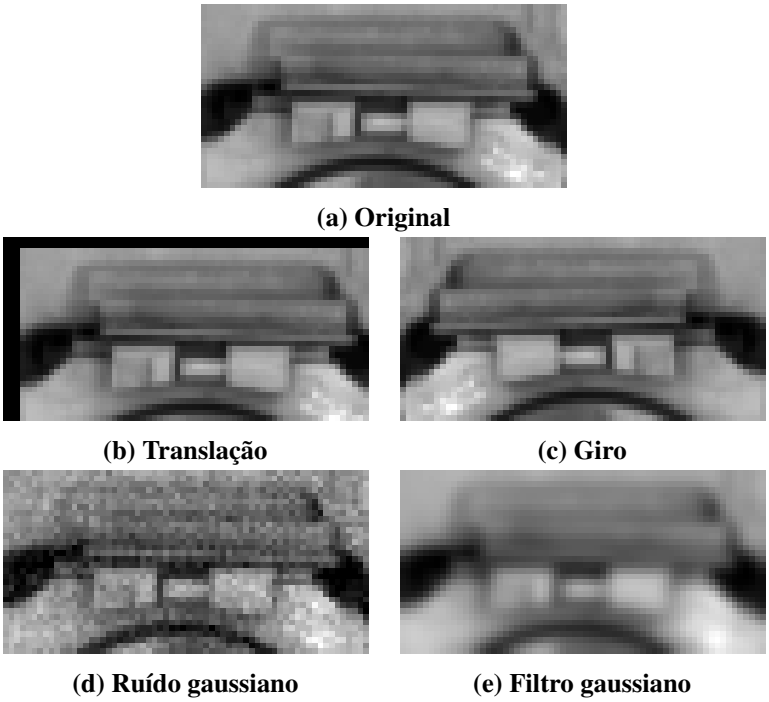
A Figura 17 apresenta a utilização da equalização de histograma, onde a Figura 17a representa a imagem referente a classe 1 com os valores de níveis de cinza variando entre 6 e 37 e a Figura 17b descreve a equalização de histograma aplicada à imagem original com níveis de cinza variando entre 0 a 255.

4.4.2 Data augmentation

Com o intuito de avaliar se o número imagens de treinamento do modelo é suficiente para alcançar uma boa performance de classificação da rede neural, utiliza-se o *data augmentation*, que expande artificialmente o número de amostras (imagens) da porção da base de dados destinadas ao treinamento através de transformações, perturbações ou ruídos nas imagens preservando a classe a qual ela pertence (KRIZHEVSKY; SUTSKEVER; HINTON, 2012; CHATFIELD et al., 2014).

Além de adicionar amostras à base de dados, o *data augmentation* é um dos métodos mais fáceis e comum para reduzir o sobre-ajuste (*over-fitting*) apresentado durante o treinamento do modelo. O *over-fitting* ocorre quando as amostras de treinamento se ajustam corretamente à aprendizagem do modelo, porém quando outras amostras (que não participaram do treinamento) são testadas no modelo construído, elas demonstram resultados inferiores ao do treinamento, ou seja, o modelo se mostra ineficiente ao generalizar e classificar novas amostras.

Na literatura, o *data augmentation* vem sendo utilizado para aumentar o número de imagens de treinamento, além de aumentar a performance de CNNs em aplicações que envolvam a inspeção por imagens de componentes ou objetos. Em (CHA et al., 2018), a abordagem é utilizada para aumentar o número de imagens para a inspeção de rachaduras em estruturas de concreto, através de transformações nas imagens por giros horizontais. Transformações por espelhamentos verticais e horizontais, assim como recortes aleatórios são realizados para aumentar o número de imagens de treinamento para a classificação do fixador do trilho ferroviário



(a) Original

(b) Translação **(c) Giro**

(d) Ruído gaussiano **(e) Filtro gaussiano**

Figura 18 – Transformações realizadas pelo data augmentation. A imagem do componente (a) é transformada por translação (b), giro (c), ruído (d) e filtro(e).

através de CNN, como mostra (GIBERT; PATEL; CHELLAPPA, 2017).

A Figura 18 apresenta as transformações através do *data augmentation* utilizada nas imagens do componente analisado neste trabalho. A Figura 18a mostra a imagem original presente na base de dados, que reflete a imagem de um *pad* danificado (classe 3). As Figuras 18b-18e apresentam as transformações realizadas na imagem original (Figura 18a), que são: translação de 3 pixels no eixo horizontal positivo e 2 pixels no eixo vertical positivo é apresentada na Figura 18b; giro vertical (Figura 18c); ruído gaussiano de média 0 e desvio padrão de $0,05$ para gerar pertubações à imagem é demonstrado na Figura 18d; e a Figura 18e evidencia a utilização de filtro gaussiano com valor 1 de desvio padrão do *kernel* gaussiano para suavizar ou borrar a imagem.

Em relação a translação feita pelo *data augmentation*, as transformações são feitas aleatoriamente entre -3 e 3 pixels nos eixos horizontal e vertical. Já o ruído gaussiano possui uma média fixa de 0 e desvio padrão variando entre $0,001$ e $0,05$, valores que são utilizados na distribuição normal (gaussiana) que gera o ruído aplicado as imagens. Por fim, o *kernel* gaussiano utilizado na convolução com a imagem que resultará na imagem borrada é gerado por um desvio padrão variando entre $0,3$ e 1.

4.5 Descrição dos experimentos

4.5.1 Arquitetura

Os experimentos para a classificação do *pad* com a utilização de redes neurais convolucionais são realizados com a arquitetura de rede convolucional ilustrada na Figura 19. Essa arquitetura é composta por 6 camadas, sendo duas camadas visíveis e quatro camadas ocultas. As visíveis são as camadas de entrada e saída. A camada de entrada é representada por imagens de resoluções de 32×64 e a camada de saída da arquitetura mostra a distribuição de probabilidades da função *softmax* (Equação 3.12) das três classes as quais o *pad* pode assumir.

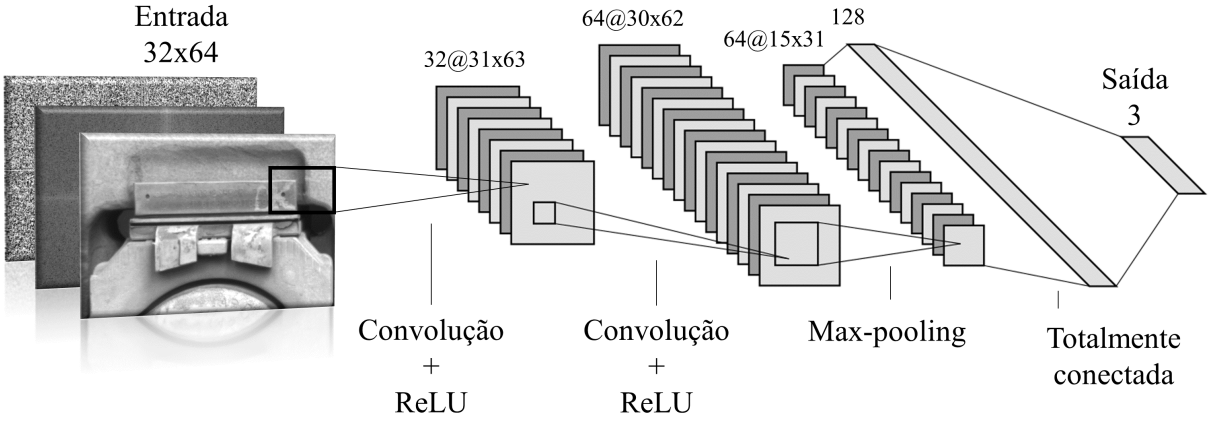


Figura 19 – Arquitetura da rede neural convolucional utilizada nos experimentos para a classificação do pad. A arquitetura possui duas camadas visíveis e quatro camadas ocultas.

As camadas ocultas são formadas por duas camadas convolucionais, uma camada de subamostragem que utiliza *max-pooling* e, por fim, uma totalmente conectada (MLP) com 128 neurônios. A primeira e a segunda camada convolucional possuem 32 e 64 mapas de características, respectivamente. Ambas as camadas convolucionais têm *kernels* e *strides* de tamanho 2×2. Já a camada *max-pooling* possui tanto o *kernel* como *stride* (o qual a operação é realizada) de tamanho 2×2.

As dimensões de saída tanto das camadas convolucionais como as de *max-pooling* são obtidas através da Equação 3.11, quando se considera a não utilização de *padding*, ou seja, $p = 0$. A notação no formato $C@AxB$ presente na Figura 19 diz respeito ao número de mapas de características presente e a resolução dos mesmos após a operação realizada por determinada camada, que são denominados por C e AxB respectivamente.

Das quatro camadas ocultas, três delas (duas convolucionais e a totalmente conectada) utilizam a função de ativação ReLU (Equação 3.9) limitar seus respectivos valores de saída como mostra a Figura 19. A escolha da função de ativação se dá pela simplicidade e velocidade em relação a outras funções de ativação não lineares durante a aprendizagem do algoritmo.

A Tabela 5 resume as informações sobre a arquitetura utilizada nos experimentos. Na camada 1, o intervalo entre $1-3$ presente no campo mapa é referente a quantidade de entradas

Tabela 5 – Descrição da arquitetura da rede neural convolucional de seis camadas utilizadas neste trabalho. São descritos a quantidade de mapas de características (saídas), tamanho da saída, tamanho do kernel, o stride e a função de ativação utilizada na camada.

Camada	Tipo	Tamanho	Mapa	Kernel	Stride	Ativação
1	Entrada	32×64	1-3	-	-	-
2	Convolucional	31×63	32	2×2	2×2	ReLU
3	Convolucional	30×62	64	2×2	2×2	ReLU
4	Max-pooling	15×31	64	2×2	2×2	-
5	MLP	128	-	-	-	ReLU
6	Saída	3	-	-	-	Softmax

da rede neural, que serão descritas na Subseção 4.5.3. Na camada de *max-pooling*, 4, o *kernel* representa o tamanho do *pool*. Enquanto que nas camadas 5 e 6, o campo tamanho denota a quantidade de neurônios nestas camadas.

4.5.2 Configurações de treinamento e classificação

Para o treinamento do modelo utiliza o algoritmo de aprendizagem *backpropagation* em conjunto como otimizador SGD, onde os pesos sinápticos são atualizados a uma determinada taxa de aprendizagem e o um certo tamanho do *batch*. Tanto a taxa de aprendizagem como o tamanho do *batch* correspondem aos hiperparâmetros do modelo, os quais influenciam na qualidade do modelo obtido após o processo de treinamento. Os hiperparâmetros são investigados nos resultados apresentados no Capítulo 5. O treinamento da rede é realizado durante 50 épocas de treinamento, onde uma única época é definida como uma ida e volta do algoritmo BP durante a aprendizagem, utilizando-se todos os dados destinados a esta tarefa.

A validação cruzada é utilizada para avaliar a performance do modelo gerado por cada um dos experimentos realizados. Especificamente, a validação cruzada por *K-fold* que divide o conjunto de dados em K subconjuntos e parte destes, que formam o conjunto de treinamento, são utilizados para o treinamento da rede neural convolucional. No *K-fold* o modelo é ajustado por $K-1$ subconjuntos da base dados e o subconjunto restante (conjunto de validação) é utilizado para validar o modelo que é gerado durante o treinamento, além disso, todos os K subconjuntos são utilizados para validação de modo que os outros subconjunto são utilizados para o ajuste do modelo.

Neste trabalho os experimentos são realizados utilizado-se a validação cruzada por 5-*fold* e a Figura 20 apresenta o *K-fold*, onde $K = 5$ subconjuntos são apresentados, a validação do modelo gerado é feita pelo subconjunto 2 (itálico e sublinhado na Figura 20) e outros subconjunto são utilizados para o treinamento. Desse modo os resultados obtidos dos experimentos representam a média dos $K = 5$ subconjuntos utilizados para a validação do modelo.

Embora a base de dados tenha um certo nível de balanceamento entre o número de imagens por classe, a base de dados possui um número diferente de amostras em cada uma

Figura 20 – Subconjuntos da base dados na validação cruzada por *K-fold*, onde existem $K = 5$ subconjuntos, 4 subconjuntos (1, 3, 4 e 5) são utilizados para o treinamento (ajuste) do modelo e apenas um (2, destacado em vermelho) é utilizado para validação do modelo gerado.

das classes como mostra a Tabela 4. Logo, o 5-*fold* possui, em média, aproximadamente $395,2$ imagens por subconjunto, indicando que as métricas de classificação apresentadas adiante (Seção 4.5.4) são obtidas da média de $395,2$ (quantidade de imagens na validação do modelo), onde essa quantidade é dividida em média $130,2$, $128,8$ e $136,2$ imagens por classes 1, 2 e 3, respectivamente.

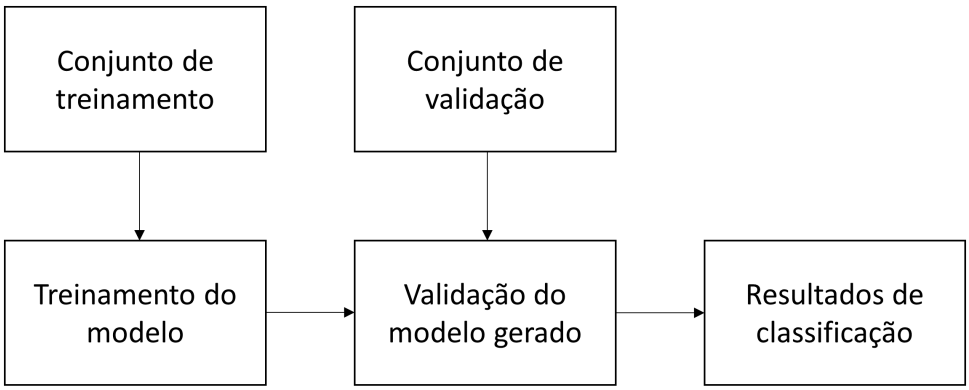


Figura 21 – Fluxograma que demonstra as etapas de treinamento e validação do modelo para classificação de imagens do componente ferroviário pad.

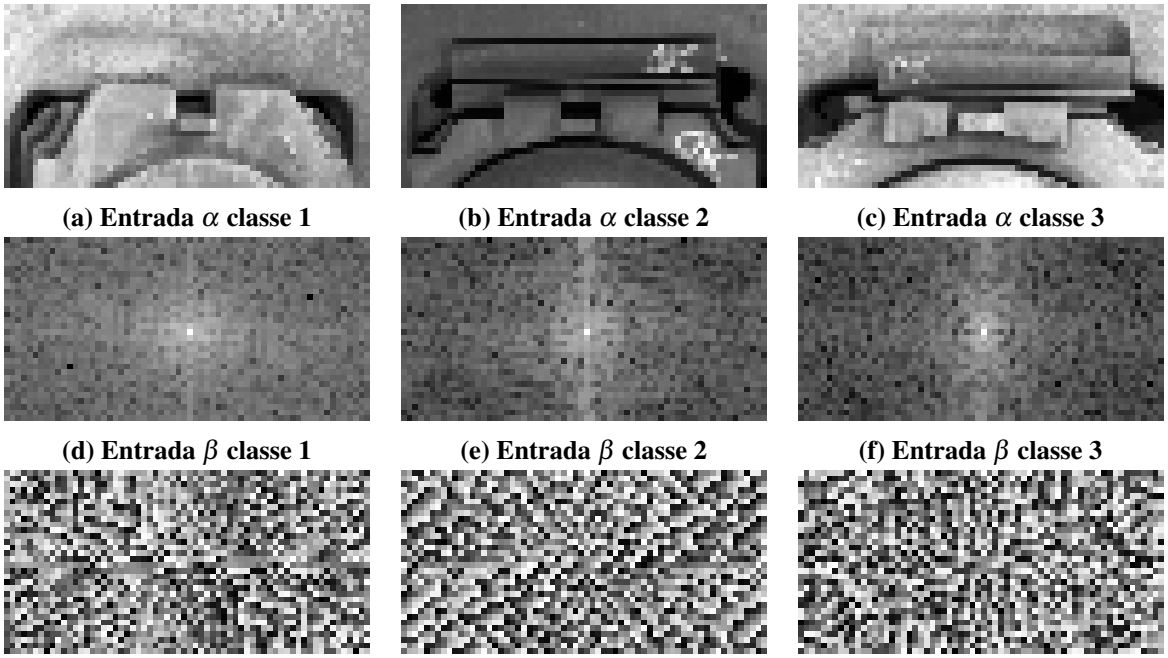
A Figura 21 mostra o fluxograma para o treinamento do modelo gerado pelo treinamento da rede neural convolucional e a validação deste modelo. O treinamento do modelo é feito pelo conjunto de treinamento, e o modelo gerado é validado pelo conjunto de validação o qual são extraídos os resultados de classificação dado pelas métricas apresentadas na Seção 4.5.4.

4.5.3 Metodologia dos experimentos

Imagens coloridas são formadas por composições de bandas monocromáticas, que são imagens na escala de cinza. No sistema de cores RGB (red, blue e green), a imagem é formada por três bandas que representam as cores vermelho, azul e verde, onde a imagem é representada pela dimensão $A \times L \times B$, sendo A, L e B a altura, largura e banda respectivamente (GONZALEZ; WOODS, 2008). Assim, para avaliar a inserção da DTF da imagem do componente na classificação será considerada a lógica similar as bandas em imagens RGB.

Para realização dos experimentos, a imagem do componente em escala de cinza no domínio do espaço é representada por α, a imagem da magnitude (domínio da frequência) da DFT é representada β e a fase, também no domínio da frequência, da DFT é representada por γ. Tanto β quanto γ são obtidos a partir da imagem original do domínio espacial α. Desse modo, um experimento que utiliza α e β como entradas da CNN será representado pela notação $\alpha\beta$, cuja dimensão será dada por $32 \times 64 \times 2$ e o valor 2 representa o número de bandas, por outro lado o experimento que é feito com a combinação dos três tipos de entradas será $\alpha\beta\gamma$ com dimensão $32 \times 64 \times 3$, como mostra a entrada da arquitetura da Figura 19.

A Figura 22 demonstra as entradas da CNN. Na Figura 22 são mostradas três imagens de entrada que representam as três classes analisadas durante a classificação do componente. As Figuras 22a, 22b e 22c são imagens do domínio espacial, por outro lado as Figuras 22d, 22e e 22f são imagens do domínio da frequência (magnitude da DFT), e as Figuras 22g, 22h e 22i representam a fase da DFT, cuja ordem denota as classes que cada uma pertence respectivamente.



(a) Entrada α classe 1 **(b) Entrada α classe 2** **(c) Entrada α classe 3**

(d) Entrada β classe 1 **(e) Entrada β classe 2** **(f) Entrada β classe 3**

(g) Entrada γ classe 1 **(h) Entrada γ classe 2** **(i) Entrada γ classe 3**

Figura 22 – Entradas da rede neural convolucional de dimensões 32×64. As imagens (a), (b) e (c) representam imagens espaciais de três classes distintas, enquanto (d), (e), (f) e (g), (h) e (i) são imagens do domínio da frequência obtidas a partir da DFT (magnitude e fase, respectivamente) de (a), (b) e (c) respectivamente.

Para denotar as técnicas de processamento utilizadas (descritas na Seção 4.4) nos experimentos, é necessário representá-las para uma melhor compreensão, como feito com as entradas. A equalização de histograma é presentada por $\boldsymbol{H}$ e o *data augmentation* é representado por $\boldsymbol{D}$. Por exemplo, um experimento realizado enunciado por $\boldsymbol{H}\alpha$ utiliza a equalização de histograma na imagem espacial de entrada definida por α. É importante salientar que tanto $\boldsymbol{H}$ como $\boldsymbol{D}$ são aplicados as imagens espaciais (original), desse modo um experimento $\boldsymbol{D}\beta$ que utiliza o *data*

augmentation na imagem espacial para depois calcular a magnitude da DFT que será utilizada como entrada β.

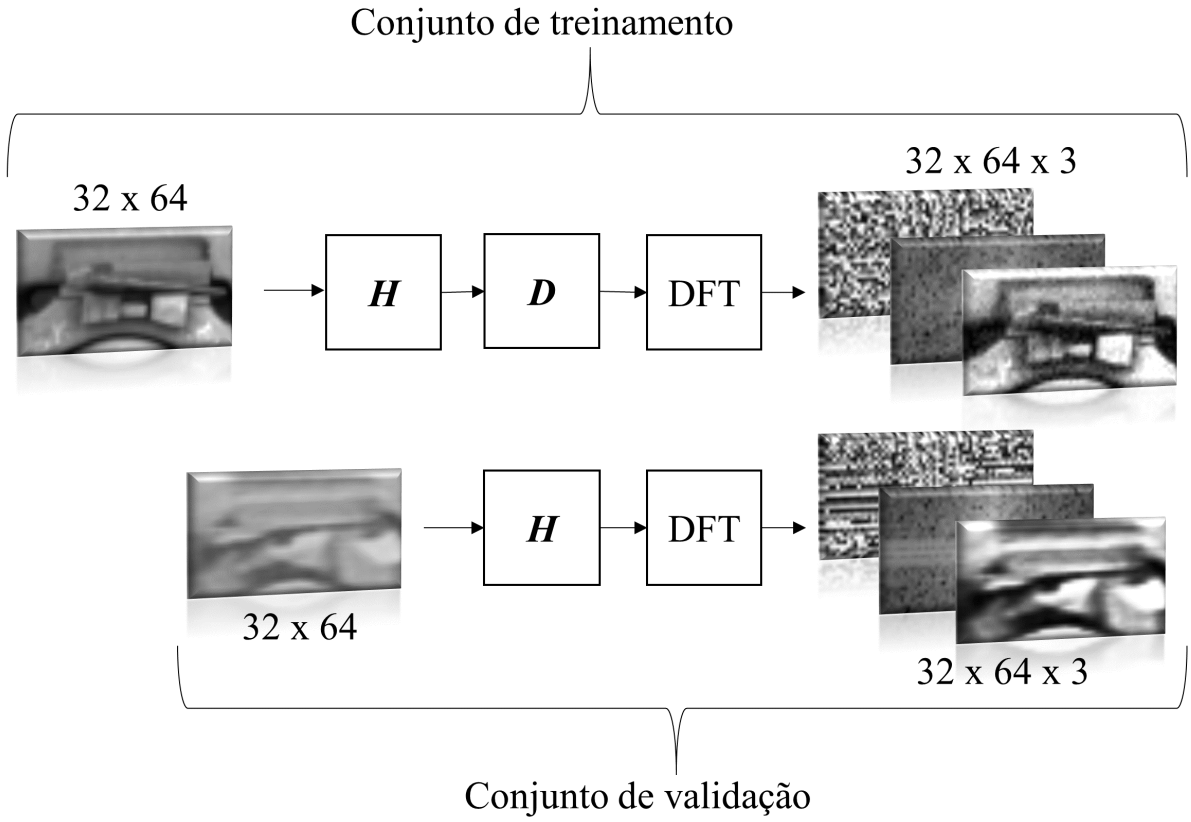


Figura 23 – Fluxograma da abordagem empregada na utilização das técnicas equalização de histograma *H*, *data augmentation D* e na obtenção da DFT nos conjuntos de treinamento e validação. O *Data augmentation* é somente utilizado no conjunto de treinamento.

Além disso, considerando que alguns experimentos serão realizados com a combinação de $\boldsymbol{H}$ e $\boldsymbol{D}$, é necessário especificar a ordem, em qual porção da base de dados (conjuntos de treinamento e validação) serão utilizados e o número de imagens após sua utilização. A primeira técnica utilizada em ordem é $\boldsymbol{H}$, que é aplicada em ambos os conjuntos de treinamento e validação sem alterar seus números de imagens. Em seguida, é utilizado o *data augmentation* $\boldsymbol{D}$ somente no conjunto de treinamento, que representa quatro dos cinco subconjuntos do 5-*fold* realizado. Após $\boldsymbol{D}$, o conjunto de treinamento é expandido (através das transformações mostradas na Seção 4.4.2).

A Figura 23 apresenta o fluxograma da abordagem empregada nos conjuntos de treinamento e validação quando se utilizam $\boldsymbol{H}$ e $\boldsymbol{D}$, bem como são obtidas as imagens β e γ do domínio da frequência. As imagens espaciais α do componente em escala de cinzas de entrada, mostradas pelos conjuntos de treinamento e validação, possuem resolução 32×64, e as saídas do fluxograma tem o tamanho $32 \times 64 \times 3$, ou seja, α, β e γ, onde as entradas e as saídas pertencem a classe 3.

Em relação ao conjunto de treinamento (parte superior da Figura 23), a imagem espacial do *pad* tem seu histograma equalizado pelo bloco $\boldsymbol{H}$. Em seguida, são aplicadas transformações

na imagem já equalizada, via *data augmentation* (bloco $\boldsymbol{D}$), para aumentar quantidade de imagens de treinamento, neste caso, é utilizado o ruído gaussiano com média 0 e desvio padrão igual a $0,05$. Por fim, é calculada a magnitude e fase (bloco DFT) da imagem espacial gerada após $\boldsymbol{H}$ e $\boldsymbol{D}$, e como resultado se tem uma amostra do conjunto de treinamento, formada, neste caso, por α, β e γ.

Vale ressaltar que a imagem de entrada no fluxograma empregado no conjunto de treinamento da Figura 23 está presente no conjunto de treinamento, equalizada, porém sem nenhuma transformação, ou seja, entre os dados de treinamento, existem a imagem equalizada (quando $\boldsymbol{H}$ é utilizada) e suas respectivas β e γ, do mesmo modo que se tem a imagem espacial α após $\boldsymbol{H}$ e $\boldsymbol{D}$, além de suas β e γ, conforme mostra a Figura 23. Desse modo, o número de imagens após expansão pelo *data augmentation*, é dado pela imagem original acrescido pela quantidade de transformações aplicadas, por exemplo, são aplicadas 4 transformações a 20 imagens do conjunto de treinamento, logo a quantidade de imagens após as transformações é de $(1+4)\cdot 20 = 100$.

Do conjunto de validação da Figura 23, é possível notar a ausência do bloco $\boldsymbol{D}$, pois este é somente utilizado no conjunto de treinamento. Desse modo, a imagem espacial α é equalizada (bloco $\boldsymbol{H}$), e logo em seguida são obtidas a magnitude e fase da sua DFT, formando uma amostra presente no conjunto de validação do modelo. No Capítulo 5 é investigada a quantidade de imagens após *data augmentation* e sua influência no desempenho de classificação das imagens do *pad*.

4.5.4 Avaliação estatística dos experimentos

Para quantificar estatisticamente os experimentos tem-se como base a matriz de confusão, que contém o número de classes classificadas corretamente que também pode ser denominada de verdadeiros positivos (VP), os verdadeiros negativos (VN) que representam a quantidade de exemplos que não pertencem a classe porém foram classificados corretamente, os exemplos que foram associados a classe erroneamente conhecidos como falsos positivos (FP) e por fim os falsos negativos (FN) que dizem respeito ao número de exemplos que não foram classificados como a classe específica (SOKOLOVA; LAPALME, 2009).

A relação entre os números de exemplos da classe correta e da classe estimada por um classificador binário são mostrados na Tabela 6. Os termos VP, VN, FP, FN apresentados anteriormente estão contidos na Tabela 6. Para o classificador binário ser ideal é necessário que os valores associados a FN e FP sejam ambos iguais a zero, deste modo todas as amostras das classes 1 e 2 são classificados corretamente.

A Tabela 6 também serve de base para o entendimento do classificador multi-classes (mais de duas classes), que no caso deste trabalho é um classificador de três classes que representam os estados em que o *pad* se encontra. Na matriz de confusão com mais de uma classe, a quantidade

Tabela 6 – Matriz de confusão que apresenta a relação entre o número de exemplos corretos e estimados pelo classificador. Aqui é presentado um classificador binário (apenas duas classes).

		Classe estimada	
		Classe 1	Classe 2
Classe correta	Classe 1	VP	FN
	Classe 2	FP	VN

de FN, FP e VN são aumentadas para refletirem a quantidade de classes presentes, assim como a posição dos valores de VP e FP são movimentados para representar a classe de interesse em determinado momento. A Equação 4.2 apresenta três matrizes de confusão de um classificador que representam três classes de interesse em momentos distintos de análise, respectivamente. Por exemplo, a segunda matriz da Equação 4.2 diz respeito a análise da classe 2, já que o VP está localizado na linha 2 e coluna 2, enquanto os FN representam as outras duas classes.

$$\begin{bmatrix} VP & FN & FN \\ FP & VN & FN \\ FP & FN & VN \end{bmatrix}, \begin{bmatrix} VN & FP & FN \\ FN & VP & FN \\ FN & FP & VN \end{bmatrix} e \begin{bmatrix} VN & FN & FP \\ FN & VN & FP \\ FN & FN & VP \end{bmatrix} \tag{4.2}$$

Devido à dificuldade de se obter um classificador ideal, é necessário a utilização de algumas métricas que possam medir a qualidade geral do classificador, além da qualidade do classificador por classe. A acurácia é uma métrica comumente utilizada para medir a eficácia geral do classificador e que é obtida pela relação entre exemplos classificados corretamente pelo número total de observações. A Equação 4.3 mostra como se obter o valor da acurácia A com base nas matrizes de confusão da Equação 4.2.

$$A = \frac{VP + VN}{VP + FN + FP + VN} \tag{4.3}$$

Outras duas métricas utilizadas para medir a qualidade do classificador são a precisão e a sensibilidade (mais conhecida como *recall*) as quais têm como foco a relação entre os exemplos classificados corretamente pelo total de itens classificados de determinada classe e a eficácia do classificador para identificar exemplos corretamente de uma classe específica, respectivamente. As Equações 4.4 e 4.5 mostram calcular os valores da precisão P e do *recall* R.

$$P = \frac{VP}{VP + FP} \tag{4.4}$$

$$R = \frac{VP}{VP + FN} \tag{4.5}$$

$$F = \frac{2 \cdot P \cdot R}{P + R} \tag{4.6}$$

Por fim, f1-*score* é a métrica que une tanto o *recall* como a precisão através da média harmônica de ambas. A partir da Equação 4.6 é possível calcular o f1-*score*, dado por F, do modelo gerado durante o treinamento. Logo com as métricas das Equações 4.3-4.6 é possível avaliar a qualidade do modelo obtido pelo classificador do *pad* utilizando a arquitetura da rede neural convolucional da Figura 19.

Ainda em relação a avaliação dos experimentos, os tempos gastos durante o treinamento e a validação do modelo são investigados. Em relação aos tempos de treinamento destacam-se os tempos gastos durante a equalização de histogramas das imagens espaciais do componente, expansão artificial das imagens espaciais por meio das transformações realizadas pelo *data augmentation*, tempo gasto na utilização da DFT na imagem espacial, além do tempo de aprendizado (treinamento em si) do modelo. A soma dos tempos gastos durante o treinamento denomina-se de tempo de ajuste (TA). Por outro lado, os tempos de validação são representados por equalização de histograma, uso da DFT e o tempo de validação em si do modelo, e a soma desses tempos denomina-se tempo de resposta (TR).

4.5.5 Configurações da máquina e ferramentas de programação

Os experimentos foram realizados em uma máquina de sistema operacional Windows 10 64 bits, processador Intel Core i7 com $1,8-2$ GHz, memória RAM de 16 GB com velocidade de 2400 MHz, GPU GeForce MX150 com 8 GB de memória.

A linguagem de programação utilizada é o Python (versão 3.7). A biblioteca Numpy (versão 1.16.4) foi utilizada para computação científica com Python, além das funcionalidades da transformada de Fourier. Para o processamento digital de imagens, a biblioteca Opencv (versão 3.4.2) foi utilizada. Os gráficos gerados para este trabalho foram criados a partir da biblioteca Matplotlib (versão 3.1.0). Já para as funcionalidades mais gerais aplicadas no aprendizado de máquina, utilizou-se o Scikit-learn (PEDREGOSA et al., 2011). A biblioteca de alto nível empregada para o aprendizado profundo, o Keras (CHOLLET et al., 2015) versão 2.2.4, é utilizada nos experimentos feitos com CNNs, além disso, o Tensorflow (versão 1.13.1) é executado em baixo nível para o treinamento de modelos de aprendizado de máquina.

5 RESULTADOS E DISCUSSÕES

O experimento inicial feito neste trabalho visa averiguar a necessidade de expandir de maneira artificial (*datau augmentation*) a quantidade de imagens do conjunto de treinamento através da curva de aprendizado, onde de acordo com o comportamento que a curva apresenta é possível indicar se a inserção de novos dados (imagens) ao treinamento resultará em uma melhoria de performance do classificador do componente ferroviário analisado.

No experimento com a curva de aprendizado, a base de dados (imagens do domínio espacial α) é dividida em 80% para o treinamento e 20% para validação do modelo, porcentagens que equivalem a 1580 e 396 imagens respectivamente. O conjunto de treinamento é dividido em quatro partes contendo 158, 631, 1104 e 1580 imagens cada, os quais representam o aumento do número de imagens utilizadas no treinamento do modelo, enquanto o conjunto de validação permanece fixo e comum a cada umas das partes. Para esse experimento, os hiperparâmetros são fixos, com taxa de aprendizagem igual a $0,001$ e tamanho do lote (*batch*) de 16.

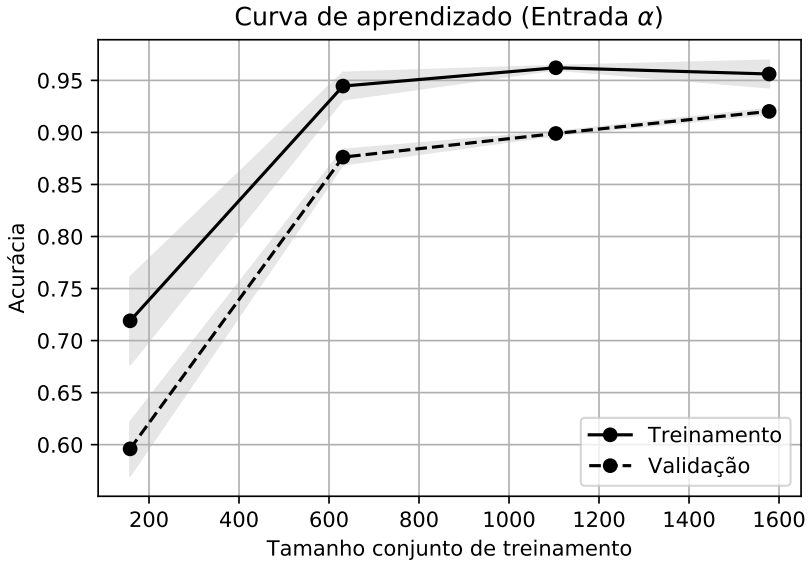


Figura 24 – Curva de aprendizado de diferentes tamanhos de conjuntos de treinamento. Cada resultado de acurácia representa uma porção do conjunto de treinamento completo com o conjunto de validação fixo.

A Figura 24 apresenta o comportamento da curva de aprendizado em relação a acurácia das partes de diferentes tamanhos da base da dados com imagens do domínio espacial α. A curva sólida representa a acurácia de treinamento do modelo e a curva tracejada mostra os resultados de acurácia de validação. O eixo horizontal possui as partes de distintos tamanhos do conjunto de treinamento e o eixo vertical a acurácia média (cinco repetições de treinamento de cada parte da base de dados e seus respectivos resultados de validação). As curvas sólida e tracejada representam as médias das repetições (denotadas pelos pontos nas curvas), já as curvas sombreadas representam o desvio padrão dos resultados obtidos durante o treinamento e

validação, respectivamente.

Ao analisar a curva de aprendizado de treinamento (sólida) da Figura 24, nota-se que a partir de 631 imagens a acurácia de treinamento gira em torno de $0,95$. Porém, ao analisar os resultados de validação, percebe-se o comportamento crescente da curva (tracejada) de acordo com o aumento do número de imagens de treinamento. Dessa maneira, o comportamento da curva de aprendizado de validação indica que a adição de novas imagens (número ainda maior que o total de imagens da Figura 24) ao conjunto de treinamento resulta em uma melhor acurácia de validação, onde o pior resultado obtêm somente $0,59$ e o melhor atinge $0,92$. Assim, é possível inferir que a expansão do número de imagens de treinamento através do *data augmentation* é uma solução viável para melhorar a performance de classificação do componente ferroviário analisado.

O experimento seguinte visa analisar o hiperparâmetro tamanho do *batch* através dos *boxplots* dos resultados de acurácia dos respectivos tamanhos: 16, 32, 64 e 128, a uma taxa de aprendizagem fixa de $0,001$. A Figura 25 apresenta os *boxplots* de acurácia de cada um dos tamanhos de *batch*, onde os resultados dos *boxplots* representam a distribuição das acurácias dos 5-*folds*. Da Figura 25, é possível notar um comportamento decrescente conforme o tamanho do *batch* é aumentado, o que é corroborado pelos centros das distribuições de acurácia dos *boxplots* determinado pela linha tracejada, que representa a mediana (ou segundo quartil) das distribuições das acurácias. Também é possível observar que nenhum dos *boxplots* apresentam *outliers*, ou seja, valores de acurácias que estão abaixo e acima dos limites (caldas dos *boxplots*) inferiores e superiores, respectivamente.

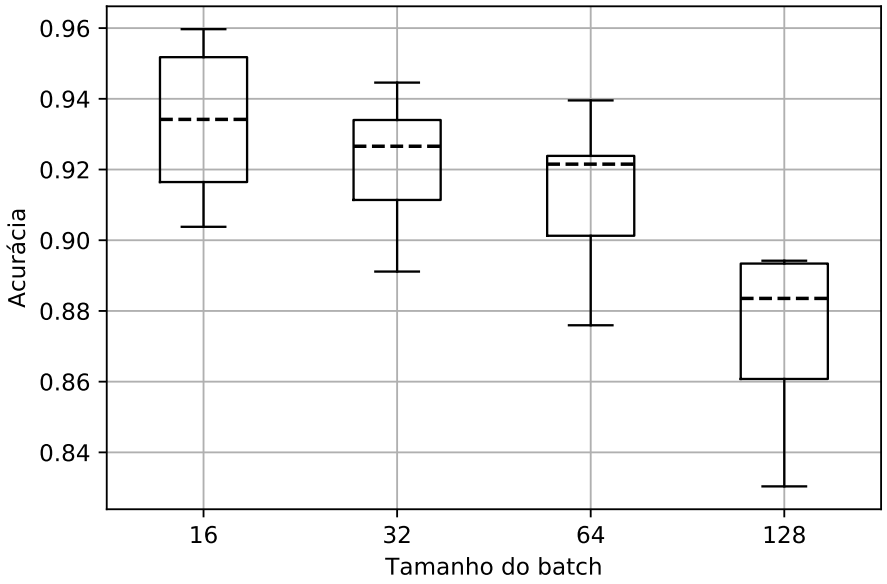


Figura 25 – *Boxplots* das distribuições de acurácias da validação cruzada por 5-fold do hiperparâmetro tamanho do *batch*. Os tamanhos de *batch* investigados são 16, 32, 64 e 128. As caldas, ou limites inferiores e superiores, dos retângulos, representam a faixa de acurácia aceita, e a linha laranja representa a mediana das acurácias obtidas.

O pior resultado dos *boxplots* da Figura 25 é apresentado pelo *batch* de tamanho 128, onde pelo menos metade das acurácias estão entre $0,8607$ e $0,8942$, indicado pelo retângulo do *boxplot* analisado. Vale salientar também, que o *boxplot* do *batch* de tamanho 128 é o que possui o menor limite inferior, indicando que esse *boxplot* aceita valores de acurácias a partir de $0,8106$. O fraco desempenho apresentado pelo *batch* de tamanho 128 é constatado também, pela proximidade entre o terceiro quartil (parte superior do retângulo) e o limite superior, mostrando que a distribuição de acurácias não possibilita valores maiores que $0,8942$. O desempenho deste *batch* é corroborado pelo seu tamanho, já que o otimizador utilizado, SGD, apresenta um melhor funcionamento ao trabalhar com pequenos *batchs* de dados aleatórios por vez (BISHOP, 2006).

Ainda da Figura 25, vale também destacar o *boxplot* do *batch* de tamanho 64, onde através de sua mediana (linha tracejada), é possível analisar a simetria da distribuição das acurácias obtidas. Considerando que a mediana está mais próxima do terceiro quartil que o primeiro quartil (parte inferior do retângulo), esta distribuição é dita assimétrica negativa. É possível observar também, que com exceção do *boxplot* do *batch* de tamanho 16, todos os outros apresentam um comportamento de distribuição de acurácias de simetria negativa, embora de diferentes níveis.

Dos *boxplos* da Figura 25, o de melhor desempenho é apresentado pelo experimento com tamanho do *batch* 16. Da distribuição das acurácias, a maior parte se encontra entre os valores de $0,9164$ e $0,9596$, e com mediana situada na acurácia de $0,9341$. Dentre os *boxplots*, este é o que possui uma distribuição simétrica de acurácias, indicada pela mediana próxima ao centro do retângulo.

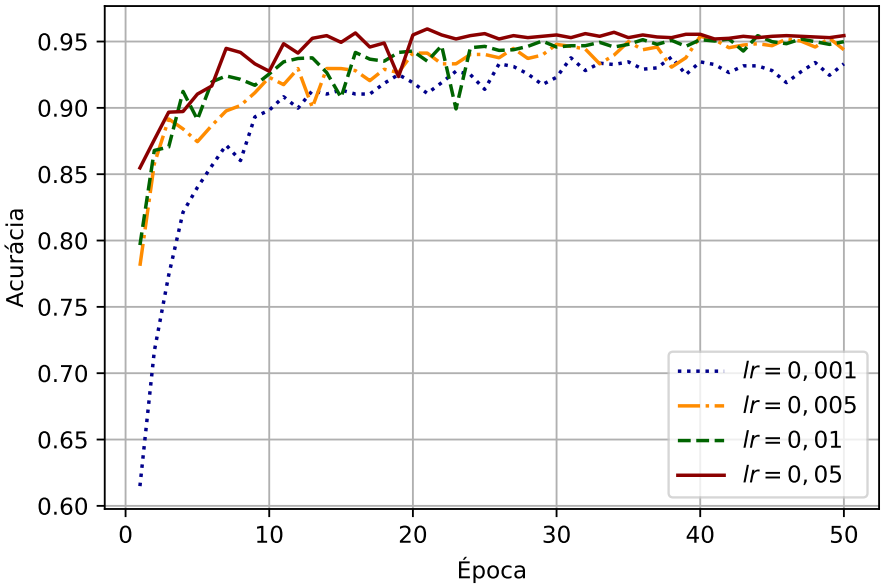


Figura 26 – Curvas de acurácia de diferentes taxas de aprendizagem lr durante 50 épocas de treinamento. O valor de acurácia de cada época representa a média dos resultados da validação cruzada por 5-*folds* utilizada neste experimento.

Após a investigação do hiperparâmetro tamanho do *batch*, é investigado a taxa de

aprendizagem ou *learning rate* (LR). Quatro taxas de aprendizagem são avaliadas, a saber: $0,001$, $0,005$, $0,05$ e $0,01$. As taxas são testadas considerando o tamanho de *batch* que obteve o melhor desempenho, onde conforme os *boxplots* da Figura 25, é o de tamanho 16. A Figura 26 apresenta as curvas de acurácia de cada um dos experimentos realizados com as taxas de aprendizagem, onde cada valor de acurácia (eixo vertical) representa a média dos 5-*folds* na época (eixo horizontal) determinada.

Da Figura 26, as curvas azul pontilhada, amarelo traço-ponto, verde tracejada e vermelha sólida são os resultados de acurácia das taxas de aprendizagem $0,001$, $0,005$, $0,05$ e $0,01$, respectivamente. É possível notar um comportamento crescente conforme a taxa é aumentada. A pior acurácia obtida após 50 épocas é apresentada pela curva azul pontilhada ($lr = 0,001$), obtendo somente $0,9332$ de acurácia. As curvas de $lr = 0,005$ e $lr = 0,01$ tem resultados bem aproximados com $0,9438$ e $0,9499$, apesar da acurácia após 50 épocas da curva em verde tracejada ser ligeiramente superior. Por fim, o melhor desempenho é apresentado pela curva vermelha sólida, $lr = 0,05$, obtendo ao fim de 50 épocas uma acurácia de $0,9544$.

A partir dos resultados obtidos pelos experimentos realizados para obtenção dos hiperparametros (tamanho do *batch* e taxa de aprendizagem) apresentados nas Figuras 25 e 26, é possível concluir que o melhor tamanho de *batch* e a melhor taxa da aprendizagem são os de valores 16 e $0,05$, respectivamente. Desse modo, esses valores de hiperparâmetros são utilizados nos seguintes experimentos feitos neste trabalho.

Tabela 7 – Resultados de acurácia de classificação do *pad* e *recall* da classe 3 de imagens do domínio espacial α e após utilização da equalização de histograma na mesma como entradas da CNN.

Método	*A* (%)	*R* (%)
α	95,44 ($\pm$ 0,98)	92,80
$\boldsymbol{H\alpha}$	95,50 ($\pm$ 0,34)	94,41

Os resultados da Tabela 7 tem como objetivo comparar os desempenho da abordagem padrão, método α, e a aplicação da equalização de histograma realizada na imagem espacial, o método $\boldsymbol{H\alpha}$. Em relação a acurácias dos métodos na Tabela 7, é possível notar um ligeiro aumento na acurácia (média dos 5-*folds*) de classificação do modelo quando a equalização de histograma é empregada, onde se nota também uma redução do desvio padrão entres as acurácias obtidas, indicando uma menor dispersão entres os resultados dos 5-*folds*. O *recall* da classe 3, apresentada na Tabela 7, indica um aumento de $1,73\%$ nesta métrica do método α para $\boldsymbol{H\alpha}$, o que gera um aumento de, aproximadamente, $2,35$ imagens do *pad* defeituoso (classe 3) a mais classificadas corretamente.

A Tabela 8 apresenta os resultados de acurácia (média e desvio padrão dos 5-*fold*) dos experimentos feitos com as entradas individuais tanto referentes as imagens do componente do domínio espacial α quanto as do domínio da frequência β e γ, assim como a combinação de magnitude e fase da DFT como entradas da rede convolucional.

Tabela 8 – Resultados de acurácia de classificação do *pad* obtidos por imagens individuais do domínio espacial α e do domínio da frequência β e γ, assim como a combinação de β e γ como entradas da CNN.

Método	*A* (%)
α	**95,44 (± 0,98)**
β	89,22 (± 1,32)
γ	83,35 (± 1,35)
$\beta\gamma$	90,94 (± 1,47)

Ao analisar somente as informações frequenciais (β e/ou γ) do componente ferroviário, é possível notar que o método γ que utilizada a fase da DFT obtém um resultado de acurácia inferior aos outros métodos, alcançando somente $83,35\%$ de acurácia. O resultado de γ é corroborado pela semelhança entre os padrões de baixa e alta frequência entre as três classes abordadas neste trabalho, conforme é mostrado na Figuras 22g-22i. Já no que diz respeito a influência da magnitude da DFT (β) é possível perceber certa melhora nos resultados, seja no método β ou $\beta\gamma$, já que os padrões de baixa e alta frequências são melhor diferenciados, de acordo como mostram as Figuras 22d-22f. Vale ressaltar o método $\beta\gamma$, que apesar de utilizar informações da fase (γ), apresenta um resultado superior quando é combinado a magnitude (β), obtendo uma acurácia de $90,94\%$.

Quando a análise da Tabela 8 recai sobre todos os métodos, nota-se a superioridade do método que tem como entrada a imagem espacial do componente, dado por α. Dentre todos os métodos da Tabela 8, α é o único que atinge uma acurácia média superior a 91%, além de ser o método que possui o menor desvio padrão ($0,98\%$) entre os resultados obtidos. Esse desempenho é reforçado devido à alta correlação que existe em imagens do domínio espacial, ao contrário do que acontece no domínio da frequência, as quais as informações espaciais estão espalhadas no domínio da frequência.

Tabela 9 – Resultados de acurácia e f1-*score* de classificação do pad obtidos pelas combinações das imagens do domínio espacial α e do domínio da frequência β e γ como entradas da CNN.

Método	*A* (%)	*F* (%)
$\alpha\beta$	**95,65 (± 0,34)**	**95,66**
$\alpha\gamma$	94,58 (± 0,91)	94,60
$\alpha\beta\gamma$	94,79 (± 1,59)	94,81

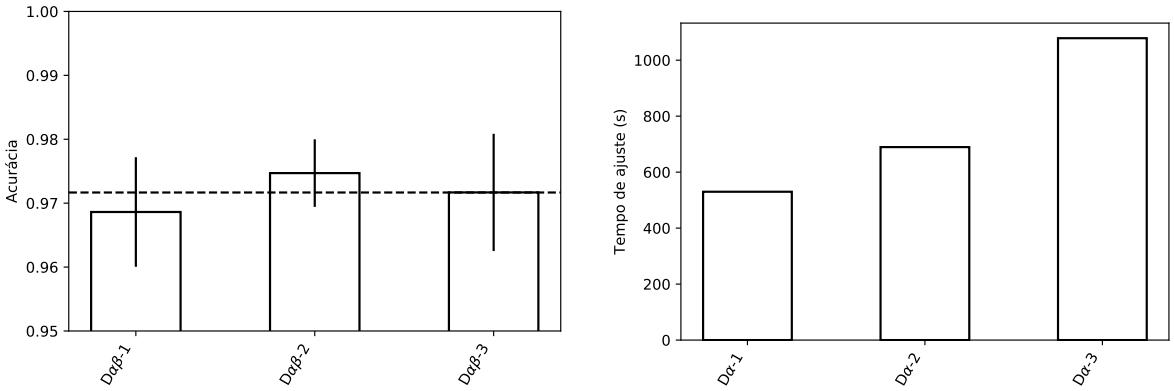
Os próximos experimentos realizados são referentes a combinação das imagens do domínio espacial α com as do domínio da frequência β e/ou γ como entrada da rede convolucional. A Tabela 9 apresenta os resultados de acurácia e f1-*score* desses experimentos.

Da Tabela 9, ao analisar os experimentos que possuem a fase da DFT (γ), nota-se que o melhor desempenho é apresentado pelo método $\alpha\beta\gamma$, atingindo uma acurácia de $93,82\%$ e um f1-*score* de $94,79\%$. Apesar disso, o método $\alpha\beta\gamma$ possui o maior desvio padrão entres os

resultados de acurácia, com $1,59\%$.

O experimento que apresentou o melhor desempenho na Tabela 9 é o método representado pelas entradas $\alpha\beta$, sendo o único experimento a obter tanto a acurácia quanto f1-*score* acima de 95%. Ao comprar o método α (Tabela 8) com o método $\alpha\beta$ (Tabela 9), é possível deduzir a contribuição significativa da informação de magnitude da DFT aliada a imagem espacial no desempenho da classificação realizada. Desse modo, é feita a seguir uma análise do método $\alpha\beta$ combinado a equalização de histograma e o *data augmentation* para avaliar a melhoria no desempenho de classificação do componente *pad*.

Antes analisar a aplicação da equalização de histograma e o *data augmentation* ao método $\alpha\beta$, faz necessário avaliar a quantidade de imagens espaciais α após *data augmentation*, o desempenho na classificação das imagens do *pad*, e o custo computacional inerente a quantidade de transformações aplicadas. A Figura 27 apresenta os resultados da investigação feita sobre o número de imagens do *data augmentation* no método $\alpha\beta$, onde as abordagens $\boldsymbol{D\alpha\beta}$**-1**, $\boldsymbol{D\alpha\beta}$**-2** e $\boldsymbol{D\alpha\beta}$**-3** exemplificam a expansão feita através das transformações (Seção 4.4.2), gerando cinco, sete e onze vezes a quantidade de imagens no conjunto de treinamento, respectivamente. Considerando que em média (5-*folds*) 1580.8 imagens são utilizadas no treinamento da CNN, após a expansão artificial das imagens, o conjunto de treinamento possui, aproximadamente: 7904, $11065,6$ e $17388,8$, quantidades que são empregadas pelas abordagens $\boldsymbol{D\alpha\beta}$**-1**, $\boldsymbol{D\alpha\beta}$**-2** e $\boldsymbol{D\alpha\beta}$**-3**, respectivamente.



(a) Acurácia **(b) Tempo de ajuste**

Figura 27 – Investigação da quantidade de imagens geradas após a utilização do *data augmentation*. As abordagens $\boldsymbol{D\alpha\beta}$-1, $\boldsymbol{D\alpha\beta}$-2, $\boldsymbol{D\alpha\beta}$-3 expandem o conjunto de imagens de treinamento em 5, 7 e 11 vezes, respectivamente. A média (barras verdes) e o desvio padrão (linha roxa) das acurácias do 5-*folds* das três abordagens, além da acurácia média das abordagens (linha vermelha tracejada) são apresentadas em (a) e os tempos de ajuste das abordagens são apresentados em (b).

A Figura 27a apresenta o gráfico de barras das abordagens do *data augmentation*, onde as barras verdes representam a médias e as linhas roxas o desvio padrão das acurácias obtidas pelas abordagens, enquanto a linha vermelha tracejada indica a acurácia média entre as

abordagens. Enquanto a Figura 27b apresenta os resultados em relação ao custo computacional das abordagens, neste caso, o tempo de ajuste (*TA*) em milissegundos (ms).

O pior desempenho é apresentado por $\boldsymbol{D}\alpha\beta$-1, sendo o único experimento abaixo da média (linha vermelha tracejada) entre os experimentos, alcançando uma acurácia de somente $0,9686$ e obtendo o maior desvio padrão entre as abordagens, com $0,86\%$ (linha roxa). Apesar do desempenho em relação a acurácia, a abordagem $\boldsymbol{D}\alpha\beta$-1 é que possui o menor tempo de ajuste, com somente $529,96$s.

Os experimentos $\boldsymbol{D}\alpha\beta$-2 e $\boldsymbol{D}\alpha\beta$-3 são os que obtiveram as acurácias acima da média entre as abordagens, como mostra a Figura 27a, de valor $0,9716$. Apesar disso, $\boldsymbol{D}\alpha\beta$-2 atinge uma acurácia ligeiramente superior a $\boldsymbol{D}\alpha\beta$-3, com $0,9747$ a $0,9717$, respectivamente. Quando *TA* é investigado (Figura 27b), observa-se uma certa diferença as os tempos das abordagens, onde $\boldsymbol{D}\alpha\beta$-2 obtém $689,44$s e $\boldsymbol{D}\alpha\beta$-3 obtém $1078,57$s. Desse modo, conclui-se que apesar da abordagem $\boldsymbol{D}\alpha\beta$-3 expandir o conjunto de treinamento para $17388,8$ imagens, e consequentemente possuir o maior tempo de ajuste, este possui o pior desempenho em termos de acurácia, quando comparada a abordagem $\boldsymbol{D}\alpha\beta$-2, que possui uma menor quantidade de imagens de treinamento, logo $\boldsymbol{D}\alpha\beta$-2 será a abordagem empregada nos experimentos seguintes neste trabalho.

A Tabela 10 apresenta a aplicação das técnicas equalização de histograma e *data augmentation* ao método $\alpha\beta$ que apresentou o melhor desempenho na Tabela 9, além do método que utiliza somente a imagem espacial (α) como entrada da CNN para comparação, conforme foi exibido na Tabela 8. As métricas acurácia, f1-*score*, e os tempos de ajuste (segundos) e resposta (milissegundos) são apresentados na Tabela 10.

Ao analisar os métodos $\alpha\beta$ e $\boldsymbol{H}\alpha\beta$ da Tabela 10, percebe-se certa melhoria nas taxas de acurácia e f1-*score* quando a equalização de histograma é utilizada, embora o desvio padrão das acurácias dos 5-*folds* do método $\boldsymbol{H}\alpha\beta$ seja levemente maior, com $0,44\%$. Em relação aos tempos de ajuste e resposta, nota-se certa similaridade entre os tempos de ajuste, apesar disso, tanto $\alpha\beta$ quanto $\boldsymbol{H}\alpha\beta$ possuem tempos de resposta, em média de ambos, $2,7$ vezes maiores que o método que utiliza somente a imagem espacial (α) como entrada da CNN.

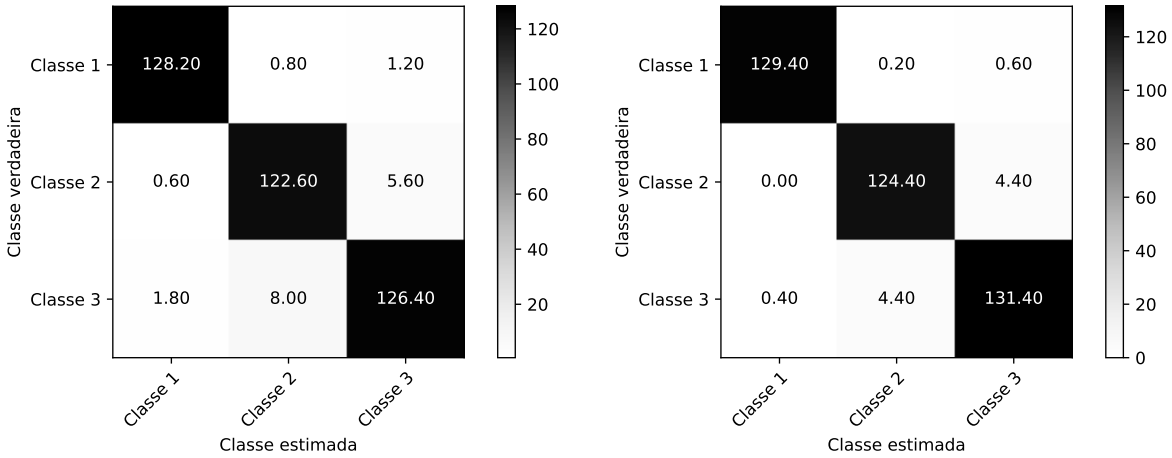
Tabela 10 – Resultados de acurácia e f1-*score*, e os tempos de ajuste e resposta de classificação do pad obtidos pelas imagens do domínio espacial α, da combinação das imagens α e do domínio da frequência β como entradas da CNN, associadas as técnicas equalização de histograma *H* e/ou *data augmentation D*.

Método	*A* (%)	*F* (%)	*TA* (s)	*TR* (ms)
α	95,44 ($\pm$ 0,98)	95,46	**120,39**	**118,93**
$\alpha\beta$	95,65 ($\pm$ 0,34)	95,66	122,71	318,63
$\boldsymbol{H}\alpha\beta$	95,75 ($\pm$ 0,44)	95,76	116,65	324,14
$\boldsymbol{D}\alpha\beta$	**97,47 ($\pm$ 0,53)**	**97,48**	689,44	332,96
$\boldsymbol{HD}\alpha\beta$	97,07 ($\pm$ 0,71)	97,08	691,10	335,94

Os métodos que utilizam *data augmentation* são os que atingem acurácia e f1-*score*

superiores a 97%, apesar do considerável aumento no tempo de treinamento (consequentemente o TA) feito pela expansão do número de imagens do conjunto de treinamento. Especificamente, o método $\boldsymbol{D\alpha\beta}$ é o que obtém a maior acurácia e f1-*score*, com $97,47\%$ e $97,48\%$ respectivamente, assim como o menor desvio padrão em comparação ao segundo melhor método ($\boldsymbol{HD\alpha\beta}$). O resultado F do método $\boldsymbol{D\alpha\beta}$, representa o quão satisfatório está o resultado de classificação em termos de precisão P e *recall* R em relação a todas as classes, já que os resultados de F da Tabela 10 representam f1-*score* médio das três classes do componente ferroviário.

Ao comparar o método padrão α e o método que alcançou o melhor desempenho de classificação do *pad*, $\boldsymbol{D\alpha\beta}$, nota-se um aumento de $2,12\%$ na acurácia média dos 5-*folds*, que representa aproximadamente $8,4$ imagens classificadas corretamente a mais. Apesar do melhor desempenho apresentado por $\boldsymbol{D\alpha\beta}$, o método α apresenta um dos menores tempos de ajuste (TA) e o menor tempo de resposta (TR), com somente $120,39$s e $118,93$ms respectivamente, mostrando a velocidade deste método, onde TA e TR de α são aproximadamente $5,74$ e $2,79$ vezes mais rápidos que os tempos obtidos pelo método $\boldsymbol{D\alpha\beta}$.



(a) Método α **(b) Método $D\alpha\beta$**

Figura 28 – Matrizes de confusão do método padrão α e do método que obteve o melhor desempenho na inspeção do *pad*, $D\alpha\beta$. Os valores das matrizes de confusão representam a média dos 5-*folds* da validação cruzada realizada na base de dados.

A Figura 28 evidencia as matrizes de confusão do método padrão, α, que utiliza somente a imagem espacial do *pad* como entrada da CNN e o método de melhor desempenho ($\boldsymbol{D\alpha\beta}$), que realiza a equalização de histograma e utiliza as imagens dos domínios espacial e da frequência (magnitude da DFT) como entradas da rede. As linhas das matrizes de confusão das Figuras 28a e 28b representam as quantidades de imagens por sua respectiva classe (verdadeira) e as colunas denotam as quantidades de imagens classificadas (estimadas pelo modelo) como determinada classe.

Ao comparar as matrizes de confusão das Figuras 28a e 28b, observa-se um aumento médio de $1,2$ imagens classificadas corretamente a mais da classe 1, além de reduzir para menos de uma imagem ($0,8$, em média) classificada erroneamente como outra classe. Analisando a

classe 2, é observado um aumento de $1,8$ do método α para o método $\boldsymbol{D}\alpha\beta$, vale ressaltar a classificação feita erroneamente das imagens como classe 1, que foi da média $0,6$ para 0. Quando a classe 3 é investigado, nota-se uma diminuição considerável no número de imagens do *pad* danificado classificadas/estimadas como *pad* normal (não danificado), especificamente de 8 para $4,4$, sendo assim é um dos fatores que contribui para o acréscimo de imagens classificadas corretamente de $126,4$ para $131,4$ em relação aos métodos α e $\boldsymbol{D}\alpha\beta$, respectivamente.

6 CONCLUSÃO

A inspeção de componentes ferroviários é uma das tarefas mais importantes em uma empresa mineradora, dada a possibilidade de um componente defeituoso causar diversos problemas a operação na ferrovia, entre os mais graves o descarrilamento de vagões do trem. Automatizar a tarefa de inspeção é uma das opções para contornar os problemas inerentes a inspeção visual feita por técnicos operacionais na ferrovia, dentre os problemas, destacam-se o erro devido a fadiga, distração e divergência entre técnicos, e até mesmo risco de segurança o qual o técnico está suscetível no ambiente ferroviário. O componente investigado neste trabalho é o *pad*, que é utilizado como suspensão primária desempenhando um papel importante na dinâmica dos vagões ferroviários, e pode ser encontrado em três situações na ferrovia, a saber: *pad* ausente, *pad* não danificado e *pad* danificado.

Este trabalho propôs a automatização da inspeção visual do *pad* através de técnicas de aprendizado profundo, especificamente CNNs. A CNN classifica a imagem que possui o componente em relação as três estados que o *pad* pode ser encontrado em operação. É apresentada, também, uma investigação sobre a inserção da imagem do componente do domínio da frequência (através da magnitude e fase da DFT da imagem original) como entradas da CNN, desse modo são investigadas individualmente e em conjunto as imagens do domínio espacial (imagem original) e as imagens do domínio da frequência do componente analisado. Além disso, verifica-se a utilização da equalização de histograma com objetivo de tornar a classificação do *pad* insensível a contraste e iluminação, assim como a aplicação do *data augmentation* para tornar a classificação robusta à condições adversas requerida por uma aplicação real.

Os resultados apresentados no Capítulo 5 mostram um ligeiro aumento ($0,06\%$) no desempenho da acurácia de classificação do *pad*, porém um aumento maior é apresentado na métrica *recall* da classe 3, chegando a um valor de $94,41\%$, quando se utiliza a equalização de histograma na imagem espacial do componente. Quando são avaliadas as combinações das imagens espaciais e da frequência do componente, o melhor desempenho é aprestando pela combinação imagem espacial e magnitude da DFT da imagem original, alcançando uma acurácia de $95,65\%$. A expansão artificial de imagens realizadas pelo *data augmentation* provou melhorar significativamente o desempenho de classificação, em especial as transformações realizadas que multiplicam por sete a quantidade de imagens espaciais de treinamento. O melhor resultado entre todos os experimentos realizados é apresentado pelo método que emprega o *data augmentation* nas imagens do domínio espacial do *pad* utilizadas no conjunto de validação, e a combinação as imagens dos domínios espacial e da frequência (magnitude da DFT) são empregadas como entrada da CNN, esse método alcança uma acurácia de $97,47\%$, representando $385,2$ imagens classificadas corretamente de um total de $395,2$ imagens do *pad*. Desse modo, é possível concluir que a arquitetura da CNN utilizada e os parâmetros empregados, são essenciais na extração de características e classificação das imagens dos domínios espacial e da frequência do

componente ferroviário analisado neste trabalho, o *pad*, ao contrário do que foi encontrado nos estudos da revisão da literatura que comumente empregam um método extrator de características separadamente de um classificador.

Para trabalhos futuros pretende-se explorar a transformada discreta Wavelet da imagem do componente e avaliar as informações obtidas desta transformação em uma CNN, para posterior comparação com a abordagem apresentada neste trabalho. Com o intuito reduzir o número de atributos do vetor de características obtido pela CNN, se cogita explorar métodos de redução da dimensionalidade das características como a Análise de Componentes Principais (*principal component analysis*, PCA) e o Análise de Discriminantes Lineares (*linear discriminant analysis*, LDA). Como alternativa ao *data augmentation*, utilizar Redes Adversárias Generativas (*generative adversarial networks*, GANs) para a geração de novas imagens do *pad* e avaliar seu desempenho na classificação do mesmo.

TRABALHOS REALIZADOS

- Avaliação de técnicas de deep learning aplicadas à identificação de peças defeituosas em vagões de trem. **Workshop of Industry Applications (WIA) in the 30th Conference on Graphics, Patterns and Images (SIBGRAPI'17)**. Niterói, RJ, Brazil, 2017. Disponível em: <http://sibgrapi2017.ic.uff.br/>.

- A deep-learning-based approach for automated wagon component inspection. **33rd Annual ACM Symposium on Applied Computing (SAC 2018)**. Pau, France, 2018. DOI: <https://doi.org/10.1145/3167132.3167157>.

- An ensemble of convolutional neural networks for unbalanced datasets: a case study with wagon component inspection. **International Joint Conference on Neural Networks (IJCNN 2018)**. Rio de Janeiro, Brazil, 2018. DOI: <https://doi.org/10.1109/IJCNN.2018.8489423>.

- Utilização da transformada discreta de Fourier em conjunto com redes neurais convolucionais para inspeção de componentes do vagão ferroviário. **II Congresso de Tecnologias e Desenvolvimento na Amazônia (CTDA 2018)**. Tucuruí, PA, Brazil, 2018. Disponível em: <http://projetoslabex.com.br/e-conference/repository/anais/Anais_II_CTDA.pdf>.

- Image inspection of railcar structural components: an approach through deep learning and discrete Fourier transform. **Symposium on Knowledge Discovery, Mining and Learning (KDMile 2019)**. Fortaleza, Brazil, 2019. DOI: <https://doi.org/10.5753/kdmile.2019.8786>.

REFERÊNCIAS

AFFONSO, C. et al. Deep learning for biological image classification. **Expert Systems with Applications**, Elsevier, v. 85, p. 114–122, 2017.

BISHOP, C. M. **Neural Networks for Pattern Recognition**. [S.l.]: Oxford university press, 1995.

BISHOP, C. M. **Pattern Recognition and Machine Learning**. 5. ed. [S.l.]: Springer, 2006. (Information science and statistics). ISBN 978-0-38-731073-2.

BROSNAN, T.; SUN, D.-W. Inspection and grading of agricultural and food products by computer vision systems—a review. **Computers and electronics in agriculture**, Elsevier, v. 36, n. 2, p. 193–213, 2002.

CARO, F.; CABRERA, G. et al. Morphology and interaction of galaxies using deep learning. **Proceedings of the International Astronomical Union**, Cambridge University Press, v. 12, n. S325, p. 205–208, 2016.

CAVUTO, A. et al. Train wheel diagnostics by laser ultrasonics. **Measurement**, Elsevier, v. 80, p. 99–107, 2016.

CHA, Y.-J.; CHOI, W.; BÜYÜKÖZTÜRK, O. Deep learning-based crack damage detection using convolutional neural networks. **Computer-Aided Civil and Infrastructure Engineering**, Wiley Online Library, v. 32, n. 5, p. 361–378, 2017.

CHA, Y.-J. et al. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. **Computer-Aided Civil and Infrastructure Engineering**, Wiley Online Library, v. 33, n. 9, p. 731–747, 2018.

CHATFIELD, K. et al. Return of the devil in the details: Delving deep into convolutional nets. **arXiv preprint arXiv:1405.3531**, 2014.

CHEN, F.-C.; JAHANSHAHI, M. R. Nb-cnn: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion. **IEEE Transactions on Industrial Electronics**, IEEE, v. 65, n. 5, p. 4392–4400, 2018.

CHOLLET, F. et al. **Keras**. 2015. <https://keras.io>.

CIREŞAN, D. et al. Multi-column deep neural network for traffic sign classification. **Neural networks**, Elsevier, v. 32, p. 333–338, 2012.

DENG, S. et al. On-line inspection system for train wheel dimensions. In: IOP PUBLISHING. **Journal of Physics: Conference Series**. [S.l.], 2005. v. 13, n. 1, p. 171.

DUMOULIN, V.; VISIN, F. A guide to convolution arithmetic for deep learning. **arXiv preprint arXiv:1603.07285v2**, 2018.

FENG, H. et al. Automatic fastener classification and defect detection in vision-based railway inspection systems. **IEEE transactions on instrumentation and measurement**, IEEE, v. 63, n. 4, p. 877–888, 2014.

GHOSAL, S. et al. Interpretable deep learning applied to plant stress phenotyping. **arXiv preprint arXiv:1710.08619**, 2017.

GIBERT, X.; PATEL, V. M.; CHELLAPPA, R. Deep multitask learning for railway track inspection. **IEEE Transactions on Intelligent Transportation Systems**, IEEE, v. 18, n. 1, p. 153–164, 2017.

GIL, A. C. Como elaborar projetos de pesquisa. **São Paulo**, 2007.

GOH, A. T. Back-propagation neural networks for modeling complex systems. **Artificial Intelligence in Engineering**, Elsevier, v. 9, n. 3, p. 143–151, 1995.

GONZALEZ, R. C.; WOODS, R. E. **Digital image processing**. Upper Saddle River, N.J. 07458: Pearson Prentice Hall, 2008. ISBN 978-0-13-168728-8.

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

HAN, J.; KAMBER, M.; PEI, J. **Data mining: concepts and techniques**. 3. ed. 225 Wyman Street,Waltham, MA 02451, USA: Morgan Kaufmann - Elsevier, 2011. ISBN 978-0-12-381479-1.

HART, J. et al. Machine vision using multi-spectral imaging for undercarriage inspection of railroad equipment. In: **Proceedings of the 8th World Congress on Railway Research, Seoul, Korea**. [S.l.: s.n.], 2008.

HAYKIN, S. **Redes neurais: princípios e prática**. 2. ed. São Paulo: Bookman Editora, 2007. ISBN 978-85-7307-718-6.

HAYKIN, S. **Neural networks and learning machines**. 3. ed. Upper Saddle River, N.J. 07458: Pearson Prentice Hall, 2009. ISBN 978-0-13-147139-9.

IWNICKI, S. **Handbook of Railway Vehicle Dynamics**. [S.l.]: CRC Press, 2006. 548 p. ISBN 978-0-84-933321-7.

JACCARD, N. et al. Automated detection of smuggled high-risk security threats using deep learning. IET, 2016.

JACCARD, N. et al. Detection of concealed cars in complex cargo x-ray imagery using deep learning. **Journal of X-ray Science and Technology**, IOS Press, v. 25, n. 3, p. 323–339, 2017.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2012. p. 1097–1105.

KUMAR, G.; BHATIA, P. K. A detailed review of feature extraction in image processing systems. In: IEEE. **Advanced Computing & Communication Technologies (ACCT), 2014 Fourth International Conference on**. [S.l.], 2014. p. 5–12.

LECUN, Y.; BENGIO, Y. Convolutional networks for images, speech, and time series. **The handbook of brain theory and neural networks**, v. 3361, n. 10, 1995.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015.

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

LECUN, Y. et al. Generalization and network design strategies. **Connectionism in perspective**, Citeseer, p. 143–155, 1989.

LI, Y.; ZHANG, J.; LIN, Y. Combining fisher criterion and deep learning for patterned fabric defect inspection. **IEICE TRANSACTIONS on Information and Systems**, The Institute of Electronics, Information and Communication Engineers, v. 99, n. 11, p. 2840–2842, 2016.

LI, Y.; ZHAO, W.; PAN, J. Deformable patterned fabric defect detection with fisher criterion-based deep learning. **IEEE Transactions on Automation Science and Engineering**, IEEE, v. 14, n. 2, p. 1256–1264, 2017.

LIU, L.; ZHOU, F.; HE, Y. Automated visual inspection system for bogie block key under complex freight train environment. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 65, n. 1, p. 2–14, 2016.

LIU, L.; ZHOU, F.; HE, Y. Vision-based fault inspection of small mechanical components for train safety. **IET Intelligent Transport Systems**, IET, v. 10, n. 2, p. 130–139, 2016.

LIU, Z. et al. Electromagnetic tomography rail defect inspection. **IEEE Transactions on Magnetics**, IEEE, v. 51, n. 10, p. 1–7, 2015.

LUCKOW, A. et al. Deep learning in the automotive industry: Applications and tools. In: IEEE. **Big Data (Big Data), 2016 IEEE International Conference on**. [S.l.], 2016. p. 3759–3768.

MACUCCI, M. et al. Derailment detection and data collection in freight trains, based on a wireless sensor network. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 65, n. 9, p. 1977–1987, 2016. Disponível em: <https://doi.org/10.1109/TIM.2016.2556925>.

MENTZELOS, K. **Object localization and identification for autonomous operation of surface marine vehicles**. Tese (Doutorado) — Massachusetts Institute of Technology, 2016.

MÜHLING, M. et al. Deep learning for content-based video retrieval in film and television production. **Multimedia Tools and Applications**, Springer, v. 76, n. 21, p. 22169–22194, 2017.

NYBERG, R. G. **Automating condition monitoring of vegetation on railway trackbeds and embankments**. Tese (Doutorado) — Edinburgh University Press, 2016.

PARK, B. et al. Characterizing multispectral images of tumorous, bruised, skin-torn, and wholesome poultry carcasses. **Transactions of the ASAE**, American Society of Agricultural and Biological Engineers, v. 39, n. 5, p. 1933–1941, 1996.

PARK, J.-K. et al. Machine learning-based imaging system for surface defect inspection. **International Journal of Precision Engineering and Manufacturing-Green Technology**, v. 3, n. 3, p. 303–310, Jul 2016. ISSN 2198-0810. Disponível em: <https://doi.org/10.1007/s40684-016-0039-x>.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

POLIT, D. F.; HUNGLER, B. P. **Nursing research principles and methods**. [S.l.]: Phyladelphia: Lippincott Williams and Wilkins, 1999.

POUND, M. P. et al. Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. **GigaScience**, 2017.

RAVIKUMAR, S.; RAMACHANDRAN, K. I.; SUGUMARAN, V. Machine learning approach for automated visual inspection of machine components. **Expert Syst. Appl.**, Pergamon Press, Inc., Tarrytown, NY, USA, v. 38, n. 4, p. 3260–3266, abr. 2011. ISSN 0957-4174. Disponível em: <http://dx.doi.org/10.1016/j.eswa.2010.09.012>.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, American Psychological Association, v. 65, n. 6, p. 386, 1958.

RUMELHART, D. E. et al. **Parallel distributed processing**. [S.l.]: MIT press Cambridge, MA, 1987. v. 1.

SHYNK, J. J. Performance surfaces of a single-layer perceptron. **IEEE Transactions on Neural Networks**, IEEE, v. 1, n. 3, p. 268–274, 1990.

SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. **Information Processing & Management**, Elsevier, v. 45, n. 4, p. 427–437, 2009.

SOME, L. **Automatic image-based road crack detection methods.** Dissertação (Mestrado) — KTH Royal Institute of Technology, Stokholmo, Sweden, 2016.

SOUKUP, D.; HUBER-MÖRK, R. Mobile hologram verification with deep learning. **IPSJ Transactions on Computer Vision and Applications**, SpringerOpen, v. 9, n. 1, p. 9, 2017.

STEIN, R. Selecting data for neural networks. **AI expert**, MILLER FREEMAN INC, v. 8, p. 42–42, 1993.

STENTOUMIS, C. et al. A holistic approach for inspection of civil infrastructures based on computer vision techniques. **International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences**, v. 41, 2016.

SUN, D.-W. Inspecting pizza topping percentage and distribution by a computer vision method. **Journal of food engineering**, Elsevier, v. 44, n. 4, p. 245–249, 2000.

VIANA, I. O. Metodologia do trabalho científico. **São Paulo: Editora EPU**, 2001.

VIEIRA, F. H. A. Image processing through machine learning for wood quality classification. Universidade Estadual Paulista (UNESP), 2016.

WANG, H. et al. Automatic illumination planning for robot vision inspection system. **Neurocomputing**, Elsevier, v. 275, p. 19–28, 2018.

ZHAO, J.; CHAN, A.; STIRLING, A. Risk analysis of derailment induced by rail breaks-a probabilistic approach. In: IEEE. **Reliability and Maintainability Symposium, 2006. RAMS'06. Annual**. [S.l.], 2006. p. 486–491.