UNIVERSIDADE DE BRASÍLIA
FACULDADE DE ARQUITETURA E URBANISMO
PROGRAMA DE PÓS-GRADUAÇÃO – PPG FAU

DETECÇÃO DE ESQUADRIAS EM PRÉDIOS PÚBLICOS COM A UTILIZAÇÃO DE TÉCNICAS DE DEEP LEARNING BASEADAS EM IMAGENS

VINÍCIUS ARAÚJO GONÇALVES

DISSERTAÇÃO DE MESTRADO EM ARQUITETURA E URBANISMO

Orientador: João da Costa Pantoja
Coorientador: Lenildo Santos da Silva

Brasília,
MAIO de 2024

FICHA CATALOGRÁFICA

GONCALVES, V. A.

DETECÇÃO DE ESQUADRIAS EM PRÉDIOS PÚBLICOS COM A UTILIZAÇÃO DE TÉCNICAS DE DEEP LEARNING BASEADAS EM IMAGENS. 2025.

(PPG-FAU/UnB, Mestre, Arquitetura e Urbanismo, 2025).

Dissertação de Mestrado - Universidade de Brasília, Programa de Pós-Graduação em Arquitetura e Urbanismo.

Faculdade de Arquitetura e Urbanismo.

1. Rede Neural
2. *Machine Learning*
3. *Deep Learning*
4. Rede Convolucional

I.Universidade de Brasília.

REFERÊNCIA BIBLIOGRÁFICA

GONCALVES, V. A. **DETECÇÃO DE ESQUADRIAS EM PRÉDIOS PÚBLICOS COM A UTILIZAÇÃO DE TÉCNICAS DE DEEP LEARNING BASEADAS EM IMAGENS.** 2024.

Dissertação (Mestrado em Arquitetura e Urbanismo) – Programa de Pós-Graduação em Arquitetura e Urbanismo, Faculdade de Arquitetura e Urbanismo, Universidade de Brasília, Brasília, DF, 2025.

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE ARQUITETURA E URBANISMO
PROGRAMA DE PÓS-GRADUAÇÃO – PPG-FAU

DETECÇÃO DE ESQUADRIAS EM PRÉDIOS PÚBLICOS COM A UTILIZAÇÃO DE TÉCNICAS DE DEEP LEARNING BASEADAS EM IMAGENS

VINÍCIUS ARAÚJO GONÇALVES

Dissertação de Mestrado submetida à Faculdade de Arquitetura e Urbanismo da Universidade de Brasília, como parte dos requisitos necessários para a obtenção do grau de Mestre em Arquiteturae Urbanismo, área de concentração Tecnologia, Ambiente e Sustentabilidade.

Aprovado por:

_____
João da Costa Pantoja, D. Sc. (FAU, UnB)
(Orientador)

_____
Lenildo Santos da Silva. (ENC, UnB)
(Coorientador)

_____
Nathaly Sarasty Narvaez. (FAU, UnB)
(Examinadora Interna)

_____
Marco Aurélio Souza Bessa. (DFLEGAL/GDF)
(Examinador Externo)

Brasília,
MAIO de 2024

AGRADECIMENTOS

O presente trabalho não representa apenas os resultados de um intenso projeto de pesquisa, mas também o resultado de uma árdua e longa jornada, entretanto prazerosa, no curso de Mestrado em Arquitetura e Urbanismo da Universidade de Brasília.

Não seria possível trilhar essa jornada, prestes a se findar, sem o apoio e companheirismo de pessoas que acreditaram no meu potencial e me ajudaram a prosseguir.

Primeiramente, agradeço ao meu bom Deus, Senhor da minha vida. Obrigado pela proteção divina, pelos cuidados de um Deus amoroso, gracioso e misericordioso. A tua graça me alcançou! Pela força para enfrentar as adversidades, mudanças e um curso tão desafiador a ti agradeço.

Ao Márcio, pelo apoio e confiança diante de situações, pelo amor incondicional, amizade e companheirismo. Reconheço em você meu alicerce e exemplo de coragem, trabalho, dedicação, fé, bondade, administração e empreendedorismo. Obrigado pelos conselhos e pela força.

Ao meu pai Custódio, cuja dedicação como professora sempre me inspirou. A maneira como ele compartilha conhecimento com generosidade e paciência é um reflexo de sua paixão pelo ensino e seu profundo cuidado com os outros. Sou imensamente grato por seu companheirismo e por me mostrar, com seu exemplo, o valor de transformar vidas através da educação. À minha mãe Aracy, por acreditarem em mim desde o início, e por me motivarem, dia após dia, com todas as idas e vindas à Brasília. Seu apoio inabalável foi a força que me impulsionou a seguir em frente. Aos amigos da universidade por estar ao meu lado durante toda a jornada.

Agradeço, em especial, meu líder Joao da Costa Pantoja por me ajudar nos momentos difíceis por toda paciência, ensinamento e atenção durante o curso e elaboração dessa pesquisa, por me acolher em seus projetos e ensino no início da minha jornada acadêmica, abrindo assim, espaço para novas oportunidades. Agradeço também a oportunidade de aprender com o professor Lenildo algo tão desafiador, diferente e inovador.

RESUMO

O uso de técnicas de processamento de imagens (IPTs) tem se mostrado promissor na identificação de problemas na construção civil, com o potencial de reduzir a necessidade de inspeções presenciais realizadas por especialistas. Essas IPTs são empregadas principalmente para manipular imagens e extrair características relacionadas a defeitos e manifestações patológicas nas edificações. Entretanto, a adoção generalizada dessas técnicas enfrenta desafios decorrentes das variações das condições reais, como mudanças de iluminação e presença de sombras. Com o intuito de superar essas limitações, esta pesquisa propõe um método baseado em visão computacional, utilizando uma arquitetura profunda de redes neurais convolucionais (CNNs) para detectar esquadrias em edifícios públicos. O modelo de deep learning foi treinado com um banco de dados específico, composto por imagens de edificações públicas distribuídos nas 26 unidades federativas e no Distrito Federal. As imagens utilizadas totalizam mais de 1.840 construções. A CNN foi treinada com mais de 19 mil imagens de 227 × 227 pixels, alcançando uma precisão de aproximadamente 80%. A robustez e a adaptabilidade do método foram testadas em imagens de alta resolução e tamanhos variados, capturadas sob diferentes condições de luz e tipos de esquadrias, que não foram utilizadas no treinamento e validação do modelo. Nessas situações, o modelo atingiu uma acurácia de 83,34%. Os resultados demonstram a eficácia da abordagem proposta, mostrando seu potencial para a detecção automática de esquadrias e para a geração de levantamentos quantitativos voltados à recuperação das mesmas. A tecnologia apresentada destaca-se pelo desempenho satisfatório e pela sua aplicabilidade em cenários reais.

Palavras-chave: Rede Neural; Apredizado de Máquina; *Deep Learning*; Rede Convolucional.

ABSTRACT

The use of image processing techniques (IPTs) has proven to be promising in identifying construction-related issues, with the potential to reduce the need for in-person inspections conducted by specialists. These IPTs are primarily employed to manipulate images and extract features related to defects and pathological manifestations in buildings. However, the widespread adoption of these techniques faces challenges due to variations in real-world conditions, such as changes in lighting and the presence of shadows. To overcome these limitations, this research proposes a method based on computer vision, utilizing a deep convolutional neural network (CNN) architecture to detect window frames in public buildings. The deep learning model was trained using a specific dataset, consisting of images of public buildings from all 26 Brazilian states and the Federal District, totaling over 1,840 constructions. The CNN was trained with more than 19,000 images at a resolution of 227 × 227 pixels, achieving an accuracy of approximately 80%. The robustness and adaptability of the method were tested on high-resolution images of varying sizes, captured under different lighting conditions and featuring various types of window frames that were not used in the model's training and validation phases. In these scenarios, the model reached an accuracy of 83.34%. The results demonstrate the effectiveness of the proposed approach, highlighting its potential for theautomatic detection of window frames and for generating quantitative surveys aimed at their restoration. The presented technology stands out for its satisfactory performance and its applicability in real-world scenarios.

Keywords: Neural Network; Machine Learning; Deep Learning; Convolutional Network.

SUMÁRIO

ÍNDICE DE FIGURAS

ÍNDICE DE TABELAS

ÍNDICE DE GRÁFICOS

ABREVIAÇÕES E SIGLAS

ABCIC – Associação Brasileira da Construção Industrializada de Concreto

ABNT – Associação Brasileira de Normas Técnicas

ADI – Análise de Dados e Informações

ANN – Rede Neural Artificial (Artificial Neural Network)

API – Interface de Programação de Aplicações (Application Programming Interface)

BN – Normalização de Lote (Batch Normalization)

CMOS – Semicondutor Complementar de Óxido de Metal (Complementary Metal-Oxide-Semiconductor)

CNN – Rede Neural Convolucional (Convolutional Neural Network)

CPU – Unidade Central de Processamento (Central Processing Unit)

DB – Banco de Dados (Database)

DeTR – Transformer para Detecção (Detection Transformer)

DL – Deep Learning

DSM – Modelo Digital de Superfície (Digital Surface Model)

ESA – Agência Espacial Europeia (European Space Agency)

FCN – Rede Totalmente Convolucional (Fully Convolutional Network)

GANs – Redes Adversárias Generativas (Generative Adversarial Networks)

GIF – Formato de Intercâmbio de Gráficos (Graphics Interchange Format)

GPU – Unidade de Processamento Gráfico (Graphics Processing Unit)

IA – Inteligência Artificial

IPT – Instituto de Pesquisas Tecnológicas

JPEG – Grupo Conjunto de Especialistas em Fotografia (Joint Photographic Experts Group)

ML – Machine Learning

MLP – Perceptron de Múltiplas Camadas (Multilayer Perceptron)

NASA – Administração Nacional de Aeronáutica e Espaço (National Aeronautics and Space Administration)

NMOS – Semicondutor de Óxido de Metal Negativo (Negative-channel Metal-Oxide-Semiconductor)

NY – Nova York

PDI – Processamento Digital de Imagens

PNG – Gráficos de Rede Portáteis (Portable Network Graphics)

RAM – Memória de Acesso Aleatório (Random Access Memory)

ReLU – Unidade Linear Retificada (Rectified Linear Unit)

RM – Modelagem de Referência

RNA – Rede Neural Artificial

SVM – Máquinas de Vetores de Suporte (Support Vector Machines)

TPU – Unidade de Processamento de Tensores (Tensor Processing Unit)

VANT – Veículo Aéreo Não Tripulado

VANTS – Veículos Aéreos Não Tripulados

1 NTRODUÇÃO

Existe uma mudança contundente que ocorreu no mundo acadêmico que atualmente muitos ignoram, mas ela tende a determinar, nos próximos anos, o progresso na engenharia civil, que é o uso da Inteligência Artificial, *Machine Learning* e mais especificamente *Deep Learning* ou aprendizagem profunda. Não se trata de discutir a substituição dos seres humanos, tampouco se está em questão ou em debate os limites e os usos da tecnologia. A realidade atual é que, com os avanços tecnológicos, a inteligência artificial tem a possibilidade de tornar a construção civil mais eficiente, leve e assertiva e vem sendo explorado principalmente em áreas para percepção de elementos, quantificação automática de objetos e detecção desses por algoritmos computacionais, possibilitando a extração de informações em diversos tipos de estruturas.

Bengio et al. (2013) define o termo *Deep Learning* em um conceito emergente que tem proporcionado avanços significativos na resolução de problemas anteriormente abordados pela comunidade de inteligência artificial. Algoritmos de *Deep Learning* capacitam computadores a aprender conceitos complexos a partir de conceitos mais simples. Essa abordagem já mostrou sucesso empírico em diversas aplicações, incluindo visão computacional e processamento de linguagem natural. Entre esses estudos, o trabalho de Silva (2017) destaca-se por realizar a detecção de convulsões epilépticas em eletroencefalogramas utilizando *Deep Learning*. Este método ajudou na identificação e prognóstico das convulsões epilépticas em seres humanos e utilizando técnicas de *Deep Learning*, o autor alcançou uma acurácia de 86,09% já na análise do primeiro paciente.

Vitorino (2016) elaborou um estudo pautado na detecção automática de pornografia infantil através de imagens, utilizando a ténica de *Deep Learning* para a retirada de características discriminatórias de imagens. A técnica de rede neural convolucional foi escolhida por apresentar excelentes resultados em classificação de imagens, obtendo 86,06% de acurácia. Santos et al. (2017), no trabalho intitulado “Uma abordagem de Classificação de Imagens Dermatoscópicas Utilizando Aprendizado Profundo com Redes Neurais Convolucionais”, estuda a classificação de imagens dermatoscópicas utilizando *Deep Learning* com redes neurais convolucionais, visando a identificação automática de melanoma. Considerando a complexidade inerente a essa tarefa,

propôs-se a aplicação de *Deep Learning* com Redes Neurais Convolucionais para aprimorar a visualização dessas imagens, alcançando uma acurácia de 91,05%.

Muitas são as abordagens dessa temática na construção civil, abrangendo diversas áreas de estudo e aplicação. Na área de monitoramento e inspeção, Vieira (2020) estuda formas de inspecionar fissuras em pavimentos rígidos utilizando drones equipados com IA, servindo como ferramenta para monitoramento do progresso destas fissuras. Na área de planejamento e construção, a IA pode analisar dados históricos de projetos anteriores, analisar regulamentos locais, otimizar layouts, prever riscos e custos e gerar desingns inovadores e contemporâneos. No gerenciamento de projetos, Souza (2020) aplica a inteligência artificial para a indentificação de potenciais atrasos em projetos, otimiza a sequência de atividades e aloca recursos de forma mais eficiente.

Menezes (2021) utiliza de redes neurais convolucionais para a área de orçamentos e custos na construção civil analizando dados históricos, prevendo custos futuros de materiais e mão de obra e conclui quea IA permite elaborar orçamentos mais precisos sem estouros orçamentários. Rocha et al. (2017) demonstram que a aplicação automatizada de *Deep Learning* para a inspeção de peças defeituosas em vagões de trem representa um avanço considerável em comparação com as técnicas tradicionais, diminuindo a dependência da interpretação subjetiva e melhorando a precisão e consistência das inspeções.

Deep Learning é representada como uma subárea do campo de *Machine Learning* (uma etapa da IA), que é dedicado ao estudo e desenvolvimento de máquinas que aprendem. Recentemente a *Deep Learning* é marjoritariamente utilizada para processar informações e realizar o reconhecimento de padrões, superando o desempenho dos métodos clássicos (TRASK, 2019).

A aplicação de Redes Neurais Convolucionais (CNNs) tem proporcionado avanços significativos na área de reconhecimento de padrões e são amplamente utilizadas em diversas áreas, como algoritmos de busca em sites de pesquisa, recomendações de conteúdo, carros autônomos, reconhecimento de fala (audio), reconhecimento de linguagem natural (texto) e visão computacional (imagens). O reconhecimento de imagens é feito principalmente com Redes

Neurais Convolucionais (VIEIRA, 2020).

As redes convolucionais operam baseadas nos mesmos princípios das redes neurais tradicionais: perceptrons são organizados em camadas, recebendo diversos valores numéricos como entrada e gerando um ou mais resultados como saída. A principal diferença reside na estrutura da rede: em vez de utilizar camadas totalmente conectadas, empregam-se camadas convolucionais, que são conectadas localmente. Essa abordagem reduz drasticamente o número de parâmetros necessários para a aprendizagem (pesos), permitindo que as redes aprendam de maneira significativamente mais rápida. Entre as camadas convolucionais, são intercaladas camadas de subamostragem (*pooling*) e, ao final da rede, são utilizadas camadas totalmente conectadas (camadas densas) para gerar a saída final.

A grande vantagem desses algoritmos é o aprendizado automático com base no conjunto de dados fornecido. Embora existam outros estudos e muitos outros bancos de dados *open-source* (genéricos) que tratam de variados assuntos, quando se trata de assuntos específicos, como esquadrias tipicamenteutilizadas em prédios públicos, objeto do presente trabalho, existe certa dificuldade de encontrar bancos *open-source* genéricos. Assim, é de grande utilidade a geração de um banco de dados específico, além do desenvolvimento de arquiteturas de redes neurais profundas que consigam aprender adequadamente com este banco de dados.

A inteligência artificial tem o potencial de revolucionar a indústria da construção civil, tornando-a mais eficiente, sustentável e segura. No entanto, é importante destacar que a implementação bem sucedida da IA na construção requerinvestimentos emtecnologia, treinamento e colaboração entreprofissionais da construção e especialistas em IA.

O panorama da construção civil está sendo profundamente reconfigurado pelas promissoras possibilidades trazidas pelo *Deep Learning.* À medida que essa tecnologia evolui e se submete a mais investigações e experimentações, torna-se evidente que suas aplicações estão destinadas a seampliar substancialmente. Essa expansão trará consigo melhorias substanciais em eficiência, decisões embasadas em informações mais completas e soluções verdadeiramente inovadoras para os desafios que caracterizam a engenharia civil no século XXI. Ao unir conhecimento

especializado com o poder da inteligência artificial, o futuro da engenharia civil se vislumbra como um terreno de possibilidades promissoras.

2 OBJETIVOS

Nesse tópico serão apresentados os objetivos do presente trabalho.

Objetivo Geral

Desenvolver uma Rede Neural Convolucional com a habilidade de identificar características arquitetônicas, como esquadrias, em edifícios públicos utilizando a arquitetura proposta por Cha et al. (2017).

Objetivos Específicos

A seguir serão expostos os objetivos específicos da pesquisa:

- Implementar, treinar e ajustar uma rede neural para a detecção de esquadrias;
- Empregar o processamento de redes neurais convolucionais na unidade de processamento gráfico (GPU) do computador;

Analisar resultados de treino validação e teste;

- Testar a rede neural que foi treinada, mas utilizando outras imagens reais diferentes das anteriores, exclusivas para testes.

Motivação da pesquisa e Justificativa

A inteligência artificial (IA) pode desempenhar um papel significativo na transformação da indústria da construção civil, trazendo eficiência e precisão para várias áreas do processo de construção. Uma das principais aplicações da Inteligencia Artificial na Engenharia Civil está relacionada ao monitoramento da integridade estrutural das construções. Zhang et al (2023) demonstram que, ao instalar sensores em pontes, prédios e outras estruturas, engenheiros civis podem coletar dados em tempo real sobre o desempenho e as condições dessas construções. Esses dados podem ser analisados por meio de técnicas de Ciência de Dados para identificar

anomalias, prever falhas potenciais e planejar atividades de manutenção. Por exemplo, algoritmos de *Machine Learning* podem ser treinados com dados históricos para reconhecer padrões que indiquem danos ou degradação estrutural, permitindo que os engenheiros adotem medidas preventivas para evitar falhas e garantir a segurança pública.

O *Machine Leaning* vem sendo aplicado em muitos sistemas construtivos, entretanto não foi encontrado artigos que exploram o potencial das técnicas de *Machine Learning* para a detecção de esquadrias. Esse pioneirismo não se restringe apenas à identificação de esquadrias, mas serve como base para uma pesquisa mais ampla, visando detectar anomalias em fachadas que podem indicar manifestações patológicas. Essas manifestações podem variar desde infiltrações, pragas, desprendimento e/ou desgaste de componentes, até problemas como corrosão, danos nas soleiras e trincas em locais de difícil acesso. Além disso, o desenvolvimento dessa tecnologia pode estender-se ao auxílio no planejamento financeiro para a revitalização de fachadas, proporcionando uma abordagem mais precisa e preventiva na manutenção preditiva de edifícios. Portanto, este estudo não só representa um avanço na detecção de esquadrias, mas também marca um passo em direção a uma abordagem mais abrangente e proativa na preservação das estruturas, possibilitando a correção dos problemas de forma antecipada.

Foi constatada a carência de conjuntos de dados públicos específicos para a identificação de esquadrias, o que evidenciou a necessidade de realização deste estudo. A pesquisa aqui desenvolvida não apenas busca preencher essa lacuna, mas também contribuir para a criação de *datasets* que possam ser utilizados em estudos futuros, servindo como uma base sólida para o desenvolvimento de novas investigações. Esses conjuntos de dados poderão ser aplicados tanto em pesquisas voltadas para a identificação de esquadrias em imagens, quanto em redes neurais mais avançadas e complexas, ampliando o escopo das aplicações e fomentando o progresso científico na área.

Então, como não existe uma base de dados *(dataset)* disponível publicamente de diversos tipos de esquadrias na construção civil, o trabalho apresentado incluirá um banco de dados exclusivo com imagens de esquadrias em mais de 1840 diferentes tipos de edificações, que poderá ser utilizado em aplicações futuras. O processamento digital de imagens (PDI) para detecção

automática de esquadrias em edifícios, foco deste trabalho, resultaria futuramente em benefícios (prazo, custo e segurança) no que diz respeito ao diagnóstico, auxiliando no processo de resolução da manifestação patológica. Então, aproveitar a aplicação dessa técnica a este sistema construtivo não apenas representa um avanço na segurança e eficiência das construções civis, mas também inaugura um campo de estudo com múltiplas implicações.

3 PROCESSAMENTO DIGITAL DE IMAGENS

Contexto Histórico

O processamento digital de imagens (PDI) é o campo que envolve o desenvolvimento e uso de equipamentos, técnicas e algoritmos de processamento de imagens digitais, afim de melhorar ou modificar o aspecto visual das imagens, ou de interpretar o conteúdo dessas imagens atravésde maquinas (GONZALEZ E WOODS, 2007).

O PDI contribui para melhorar a visualização da imagem ou adaptá-la para análises quantitativas, corrigindo defeitos ou destacando áreas de interesse. Além disso, envolve a extração e tratamento de dados quantitativos, realizados pelo próprio computador (GOMES, 2001).

O processamento de imagem pode ser compreendido como um conjunto de duas técnicas principais: a primeira é o PDI, que envolve a preparação da imagem para análises subsequentes por meio de operações matemáticas que modificam os valores dos pixels de imagens digitais; e a segunda é a Análise Digital de Imagens (ADI), que se refere à análise quantitativa do processo, onde as regiões, partículas e objetos identificados na imagem são medidos. O PDI visa melhorar a imagem, corrigindo defeitos de aquisição e/ou realçando detalhes de interesse.

Uma das primeiras aplicações de imagens digitais ocorreu na indústria dos jornais, quando as imagens eram enviadas por cabos submarinos ente Londres e Nova York (NY) em 1920. Um equipamento de impressão especializado codificava as imagens para a transmissão a cabo e depois as reconstruía no recebimento, esta implementação foi responsável por reduzir de mais de uma semana para menos de três horas o tempo necessário para transportar uma fotografia pelo oceano atlântico (GONZALEZ E WOODS, 2007) (Figura 1).

Figura 1- Fotografia digital produzida em 1921. Fonte: Gonzalez e Woods, 2007.

A evolução do processamento digital de imagens ao longo da história começa na década de 1950, com o advento dos primeiros computadores eletrônicos de uso geral e o surgimento das iniciais aplicações de processamento de imagens, prosseguindo até os dias atuais com os avanços mais recentes em técnicas de Inteligência Artificial e Aprendizado de Máquina. O desenvolvimento de gráficos computacionais só foi possível graças aos esforços que levaram à criação dos primeiros dispositivos de exibição, como o tubo de raios catódicos no final do século XIX. Entre 1950 e 1960, os primeiros computadores eletrônicos com dispositivos de exibição surgiram, incluindo o Whirlwind I do MIT, que foi um dos primeiros computadores digitais de uso geral com processamento em tempo real. Equipado com um CRT vetorial permitia desenhar linhas e pontos. Charles W. Adams e John T. Gilmore, da equipe de desenvolvimento, criaram um programa que produziu a animação de uma bola quicando, considerada a primeira aplicação de computação gráfica interativa e o primeiro jogo de computador. Desenvolvido a partir do Whirlwind na década de 1950, o sistema de defesa aérea SAGE utilizava telas CRT para exibir dados combinados de radares com informações geográficas. As estações do SAGE também eram equipadas com canetas ópticas, permitindo aos operadores selecionar elementos gráficos diretamente na tela (BOYLE E SMITH, 1970).

Entre 1970 e 1980 surgiram as primeiras aplicações de processamento de imagens digitais, impulsionadas pelos programas espaciais e aplicações médicas. Entre 1970 e 1980 um dos eventos mais significativos dessa década foi a invenção da tomografia axial computadorizada, conhecida popularmente como tomografia computadorizada. Esse processo utiliza uma fonte de

Raios-X para coletar dados ao redor da circunferência de um anel, que gira em torno de um objeto ou paciente, gerando imagens detalhadas de uma fatia específica do objeto ou corpo a ser examinado (MARQUES, 2022) (Figura 2).

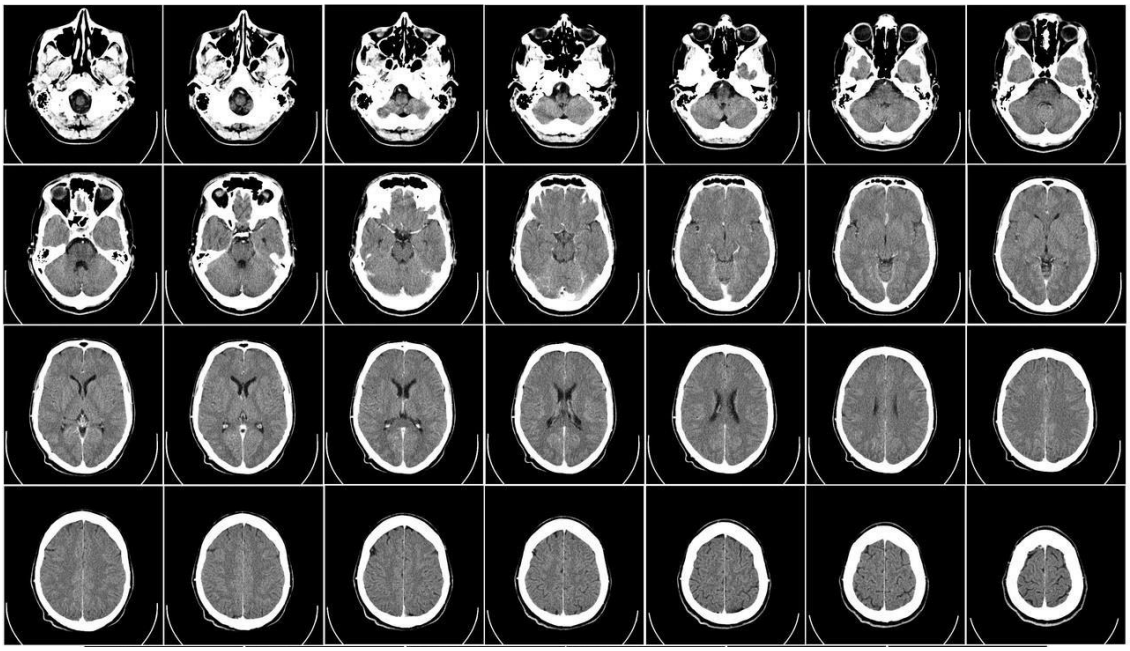


Figura 2- Visualização de uma tomografia computadorizada da cabeça. Fonte: Marques, 2022.

Durante o período compreendido entre 1980 e 2000, houve um grande avanço nas tecnologias de captura de imagens e vídeos. Em 1985, a Olympus inventou o sensor de pixel ativos NMOS, precursor dos sensores CMOS desenvolvidos em 1993 pelo Laboratório de Propulsão a Jato da NASA. Essas tecnologias são amplamente utilizadas em dispositivos de captura de imagens, desde câmeras de celulares, passando a câmeras profissionais utilizadas no cinema, a até dispositivos laboratoriais de captura de luz. (BOYLE E SMITH, 1970).

CMOS, sigla de complementary metal-oxide-semiconductor, é uma tecnologia utilizada para a construção de circuitos integrados. Essa tecnologia é amplamente empregada em microprocessadores, microcontroladores, memórias RAM e outros circuitos digitais. A tecnologia CMOS também é empregada em diversos circuitos analógicos, como sensores de imagem, conversores de dados e transceptores utilizados em diferentes tipos de comunicação (PEREIRA, 2014). Nesse período foram desenvolvidos os principais padrões de codificação, compressão e armazenamento de imagens e vídeos, entre eles estão os GIF (1987), JPEG (1992) e PNG (1995).

O padrão MPEG-4 foi criado em 1998 e suporta não apenas áudio e vídeo, mas também aplicações desenvolvidas em Java e soluções de gerenciamento de propriedade intelectual, permitindo o controle de direitos para evitar pirataria e uso ilegal de conteúdo. Esse padrão utiliza técnicas avançadas de processamento de imagens e foi adotado pelo ISDB-TB, o padrão de televisão digital brasileiro. Além disso, é amplamente utilizado em discos Blu-ray (BD) e em serviços de streaming como YouTube, Netflix, e PrimeVideo. O Video Coding Experts Group (VCEG), parte do grupo Telecommunication Standardization Sector (ITU-T), é outro grupo responsável pela criação de padrões de vídeo. O VCEG colaborou com o MPEG para criar os padrões MPEG-2 e o protocolo MPEG-4 AVC (MARQUES, 2022).

Gomes e Masselli (2015) apud Madruga (2008) ressaltam que área de processamento de imagens tem crescido devido ao grande número de aplicações em praticamente duas categorias: a primeira voltada ao aprimoramento de informações visuais para interpretação humana e a segunda por métodos computacionais onde as informações são extraídas de uma cena. Muitos dos exemplos apresentados até o momento demonstram resultados de processamento destinados à interpretação humana. A segunda principal área de aplicação das técnicas de processamento digital de imagens, mencionada no início deste capítulo, é a resolução de problemas relacionados à percepção por máquinas. Atualmente, o processamento de imagem transformou-se numa tecnologia essencial e economicamente viável em inúmeras aplicações práticas, tais como: Medicina, Detecção remota, Astronomia, Inspecção industrial, Defesa, Biometria, Biologia, Vigilância, Exploração espacial, Análise de documentos e Investigação criminal (FARIA, 2010). O interesse se concentra basicamente em procedimentos para a extração de informações de uma imagem de forma apropriada para o processamento computacional.

Medicina

O processamento digital de imagens é utilizado para diagnósticos precisos e terapias eficazes, dentre eles podemos os principais são: o Raio-x a Tomografia Computadorizada, a Ressonância Magnética e Ultrassonografia.

Raio X

O Raio X é uma forma de radiação eletromagnética localizada entre os raios gama e a luz ultravioleta no espectro eletromagnético. De acordo com a teoria quântica, os raios X podem ser interpretados como um feixe de fótons com energia hv (h vezes a frequência v). Os raios X são utilizados amplamente para formação de imagens médicas e industriais. Uma radiografia típica de tórax é obtida pelo simples posicionamento do paciente entre uma fonte de raios X e um filme sensível à energia dos raios X. A intensidade dos raios X é alterada devido à absorção diferenciada ao atravessar o corpo do paciente. Além de auxiliar ao diagnóstico, ajuda a identificar presença de complicações como derrame pleural e doença multilobar. (NAPOLITANO ET AL., 2007)

Tomografia Computadorizada

Essa técnica, que se baseia em raios-X, foi utilizada para aplicações clínicas ainda no início da década de 70, Tornou-se possível examinar o encéfalo e, com maior clareza, os limites do sistema ventricular e as partes ósseas do crânio. O aparelho consiste em uma fonte de raios-X que é acionada ao mesmo tempo em que realiza um movimento circular ao redor da cabeça do paciente, emitindo um feixe de raios-X em forma de leque. No lado oposto a essa fonte, está localizada uma série de detectores que transformam a radiação em um sinal elétrico que é convertido em imagem digital. Dessa forma, as imagens correspondem a secções ("fatias") do crânio. A intensidade (brilho) reflete a absorção dos raios-X e pode ser medida em uma escala (unidades Hounsfield) (JUNIOR E YAMASHITA, 2001).

Ressonância Magnética

A ressonância magnética (RM) é uma propriedade física de núcleos de certos elementos, que, ao serem expostos a um campo magnético intenso e excitados por ondas de rádio em uma freqüência, emitem sinais de rádio. Esses sinais podem ser captados por uma antena e convertidos em imagens. A imagem por ressonância magnética (IRM) é um método de diagnóstico por imagem não invasiva altamente sensível, particularmente eficaz na avaliação de tecidos moles, como o cérebro (HAGE E IWASAKI, 2009).

Ultrassonografia

Utilizando ondas sonoras de alta frequência (inaudíveis para os humanos), a ultrassonografia provoca vibrações nos tecidos internos. Essas vibrações são captadas por um dispositivo que converte as ondas sonoras em imagens. Com os avanços tecnológicos, a precisão desse exame tem aumentado significativamente, permitindo o diagnóstico preciso de condições como cistos, tumores, gravidez e outras anomalias. Além de diagnosticar lesões em tendões, músculos e articulações, a ultrassonografia é amplamente recomendada para pacientes com suspeita ou diagnóstico prévio de doenças do sistema digestivo, cardíaco, urinário e reprodutivo. Este exame também é extremamente eficaz para confirmar a gravidez e monitorar o desenvolvimento fetal (GRECCO, 2018).

Detecção Remota

O processamento digital para analisar imagens capturadas por drones e satélites permite monitorar mudanças ambientais, analisar o comportamento do solo, mapear áreas geográficas detectando incêndios florestais e outros desastres naturais como inundações. Em uma mina a utilização de Veículos Aéreos Não-Tripulados (VANTs) possibilita examinar a distância entre a lagoa de rejeitos e a barragem, seguindo normas ambientais rigorosas. Pode-se determinar curvas de nível, criar um modelo digital de superfície (DSM) e executar cálculos volumétricos. A utilização de VANTs torna este processo mais rápido, econômico e seguro, tornando seu uso extremamente atrativo. O uso do VANT na criação de mapas tem se destacado devido à capacidade de obter informações mais atualizadas, combinado com o baixo custo do equipamento e a automação completa do processo. Este mosaico aéreo pode ser empregado no monitoramento de plantações (CASSEMIRIO E PINTO, 2014).

Astronomia

Em Astronomia, o processamento digital de imagens é essencial para análise de dados capturados por telescópios. A finalidade do emprego das técnicas de processamento de imagens e sinais digitais no campo da astronomia abrange uma vasta gama de aplicações, desde a detecção e classificação de objetos celestes até a determinação precisa de distâncias espaciais. Ademais,

permite a compreensão aprofundada das propriedades físicas de um alvo celestial, alcançada por meio de uma análise meticulosa de seus dados fotométricos e espectrais (CARVALHO, MESQUITA E MAIA, 2021).

Inspeção Industrial

Nas aplicações práticas, destacam-se a inspeção de produtos em diversas indústrias, incluindo a farmacêutica, cosmética, automotiva, alimentícia e de bebidas, além de eletroeletrônicos, entre outras. Operações que envolvem imagens são amplamente utilizadas no controle de produção, permitindo a identificação de falhas no processo produtivo, não-conformidades, quedas de disjuntores em pontos isolados e outras situações de interesse (GOMES E MASSELLI, 2015).

Na área industrial, sistemas de diagnóstico automático são utilizados para detectar falhas em motores elétricos, especialmente em motores de indução trifásica. O principal objetivo desses sistemas é evitar manutenções não programadas e, consequentemente, interrupções no processo produtivo. A prevenção de falhas permite evitar conseqüências como aquecimento excessivo, desbalanceamento da corrente e da tensão, decaimento do torque médio, entre outros fatores (SANTOS E SUETAKE, 2012).

Sistemas de diagnóstico de falhas utilizam técnicas avançadas, como redes neurais artificiais e sistemas especialistas, entre outras. A principal vantagem dessas técnicas é que sua implementação é relativamente simples em termos de complexidade computacional. Isso ocorre porque, ao contrário de outros métodos que requerem modelos matemáticos complexos, as técnicas "inteligentes" mencionadas podem funcionar eficazmente sem a necessidade desses modelos. Em resumo, esses sistemas conseguem diagnosticar falhas de maneira eficiente e com menos complicação técnica.

Defesa e vigilância e biometria

No campo da defesa, o PDI é aplicado amplamente no reconhecimento de atividades e comportamentos suspeitos, além de identificar alvos e planejar com mais afinco as operações militares. Alem disso câmeras de vigilância auxiliam na segurança pública monitorando áreas

urbanas, e prevenindo crimes. Na Biometria, a tecnologia é utilizada para reconhecimento facial, reconhecimento de íris e impressão digital para sistema de segurança e controle de acesso. Assim, tanto a defesa quanto a biometria dependem de PDI para verificar identidades e situações com alta precisão.

Biologia

Zhang et al. (2021) conduzem uma revisão bibliográfica detalhada sobre métodos de análise para contagem de microrganismos, incluindo fungos, bactérias e algas. A revisão abrange desde técnicas clássicas de processamento de imagem até abordagens modernas de aprendizado profundo. Os autores destacam que métodos baseados em aprendizado profundo são particularmente promissores para a análise de amostras microscópicas, evidenciando um vasto potencial de pesquisa nessa área. Além disso, Alves e Ferreira (2005) mostram que, além da contagem de microrganismos na biologia, o processamento digital de imagens (PDI) pode ser utilizado para a segmentação de células, quantificação de amostrs biológicas e estudo de processos celulares, demonstrando sua versatilidade e importância no campo biológico.

Exploração Espacial

Um estudo detalhado publicado no Current Robotics Reports (2023) analisa tecnologias de autonomia aplicadas a robôs espaciais, destacando como algoritmos de aprendizagem profunda são utilizados para melhorar a qualidade das imagens capturadas e para detectar e rastrear objetos no espaço. Esta autonomia é importante para missões em ambientes não conhecidos, onde a intervenção humana em tempo real é limitada devido a longos atrasos nas comunicações.

Além disso, a Agência Espacial Europeia (ESA) (2023) explorou extensivamente a utilização da inteligência artificial para aumentar a autonomia e a eficiência dos robôs espaciais. Projetos como o OPS-SAT empregam algoritmos de aprendizagem profunda para melhorar a qualidade da imagem e detectar características na superfície da Terra. A ESA também está a desenvolver missões como a Hera, que utilizará IA para navegação autónoma em direção a asteroides, demonstrando a aplicação de técnicas semelhantes às utilizadas em carros autónomos (Agência Espacial Europeia).

Investigação Criminal

Os sistemas de vigilância, tanto públicos quanto privados, expandiram-se amplamente como ferramenta de controle social, impactando o sistema judicial. O uso de imagens de vídeo como prova se tornou comum em processos criminais, desde a investigação inicial até o julgamento final. Isso introduz um elemento de prova complexo e ambíguo no processo penal: a prova em vídeo. Tal recurso é especialmente relevante no reconhecimento de pessoas, principalmente quando se baseia em imagens de câmeras de segurança (GUEDES, 2021). Deste modo, O PDI auxilia na análise de evidências visuais, indentificando suspeitos, reconstruindo eventos, fornecendo provas visuais em processos judiciais, contribuindo para resoluções de casos criminais.

Com o processamento de imagens em um extremo e a visão computacional no outro. Gonzalez e Woods, (2007) levam em consideração três tipos de processos computacionais nessa linha contínua: processos de níveis baixo, médio e alto. Os processos de nível baixo envolvem operações primitivas, como o pré-processamento de imagens para reduzir o ruído, o realce de contraste e o aguçamento de imagens. Um processo de nível baixo é caracterizado pelo fato de tanto a entrada quanto a saída serem imagens. O processamento de imagens de nível médio envolve tarefas como a segmentação (separação de uma imagem em regiões ou objetos), a descrição desses objetos para reduzi-los a uma forma adequada para o processamento computacional e computacional e a classificação (reconhecimento) de objetos individuais. Um processo de nível médio é caracterizado pelo fato de suas entradas, em geral, serem imagens, mas as saídas são atributos extraídos dessas imagens (isto é, bordas, contornos e a identidade de objetos individuais). Por fim, o processamento de nível alto envolve "dar sentido" a um conjunto de objetos reconhecidos, como na análise de imagens e, no extremo dessa linha continua, realizar as funções cognitivas normalmente associadas a visão (Figura 3).

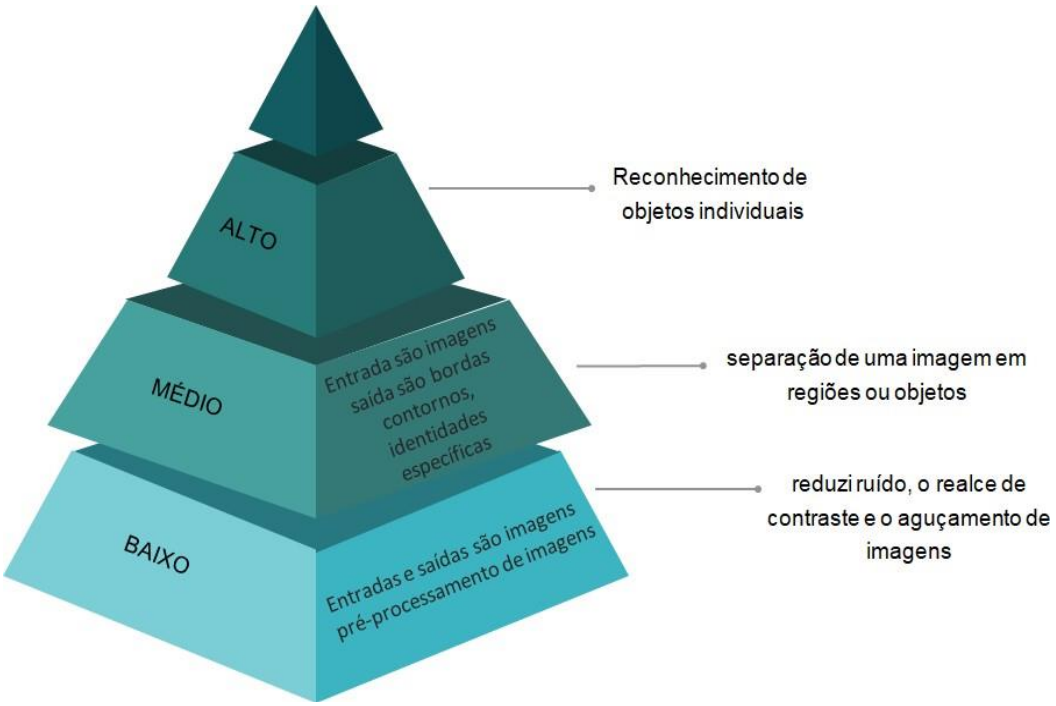


Figura 3 - Pirâmide de referência as fases do PDI. Fonte: O Autor, 2024.

Alguns estudos correlatados ao tema em questão com o objetivo de examinar o panorama atual e as metodologias empregadas por especialistas do campo em situações análogas foram indentificadas em diversos artigos que exploram o tema. Pereira (2015) propôs a aplicaçãodos operadores de Sobel e Canny com a finalidade de detectar fissuras, e o trabalho de Melo Júnior (2016) ampliou as experimentações nessa linha. Contudo, cada imagem de fissura exigia uma otimização específica de certas parâmetros de detecção, embora a presença de muitos ruídos, dependendo das condições da imagem, causasse a implementação de métodos de redução de ruído em tais situações.

Por conseguinte, é notório o surgimento de numerosos estudos recentes (entre os anos de 2018 e2019) que propõem a detecção de fissuras mediante o uso de redes totalmente convolucionais, empregando técnicas de segmentação semântica. Entre esses, podem ser considerados os trabalhos de Dung (2018), Zhang et al. (2016), Zhang et al. (2017) e Zhang et al. (2019). Essa abordagem inovadora tem impulsionado consideravelmente o campo de detecção de patologias em materiais como concreto armado, asfalto, entre outros, uma vez que é capaz de delinear de forma precisa a região de contorno das fissuras irregulares.

Etapas do processamento e análise digital de imagens

Apesar do ser humano ser muito mais eficiente em tarefas de reconhecimento, a ADI pode realizar medições mais rápidas, precisas e acuradas. A abordagem do Processamento e Análise Digital de Imagens (PADI) é organizada em fases distintas conforme seus objetivos, seguindo uma sequência convencional de captura, processamento e análise, como descrito por Vieira (2020) e Paciornik (2001). A sequencia padrão de PADI é representado pela Figura 4.

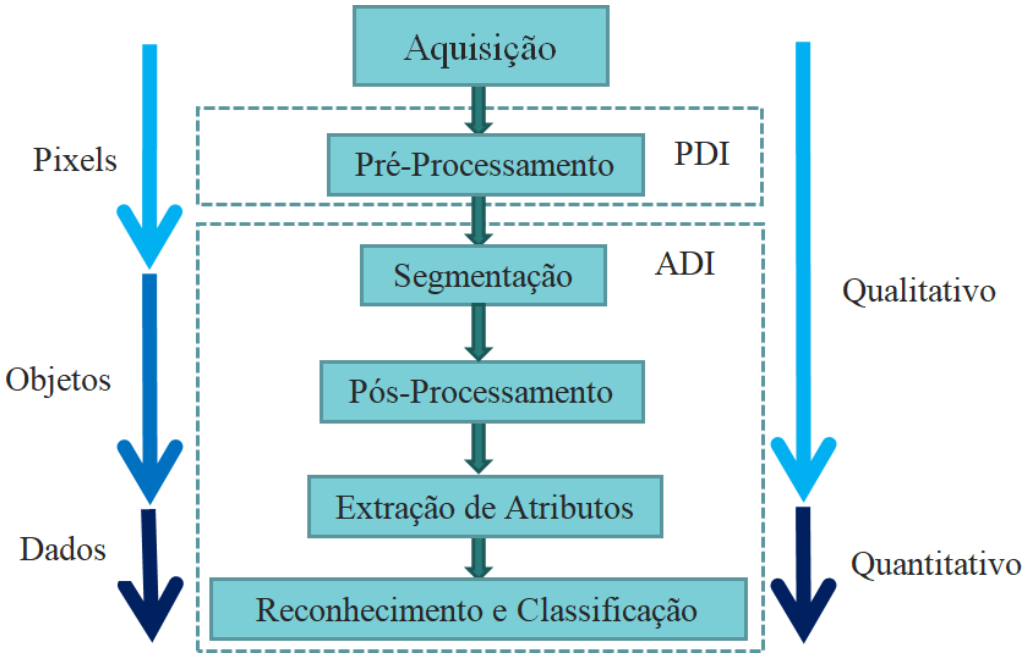


Figura 4- Sequência padrão do PADI. Fonte: O Autor, 2023.

A sequência padrão é organizada em três partes principais: Aquisição, PDI e ADI. Na aquisição a imagem é criada e digitalizada, resultando em uma imagem digital. A segunda parte é o PDI (Processamento Digital de Imagens), esta fase envolve o pré-processamento, também conhecido como realce, que melhora a qualidade da imagem. A ultima é a etapa ADI (Análise Digital de Imagens) que inclui etapas como segmentação, pós-processamento, extração de atributos, reconhecimento e classificação da imagem.

As setas com a indicação para baixo ao lado do esquema indicam o nível dos dados sobre os quais se trabalha. No pré-processamento e na segmentação, trabalhamos diretamente com os pixels

da imagem de forma arbitrária, no pós-processamento e na extração de atributos, os objetos são modificados e, em seguida, medidos. Já na etapa de reconhecimento e classificação, são analisados os resultados das medidas, ou seja, dados quantitativos.

Na aquisição temos a obtenção da imagem, Ruiz et al (2021) utilizaram de veículos aéreos não tripulados (VANT) na captura de imagens para a detecção automática de fissuras em revestimentos cerâmicos de edifícios, e com o uso das técnicas de *Deep Learning* detectaram e classificaram características específicas em imagens e vídeos automatizando o processo de inspeção visual em fachadas. O Pré-processamento trabalha a melhoria da qualidadeda imagem, como remoção de ruído e correção de contraste. O pré-processamento visa aprimorar a imagem corrigindo problemas da captura e destacando detalhes importantes, facilitando a visualização ou segmentação. Isso é geralmente feito com técnicas de aritmética deimagens. A correção de fundo é uma operação típica de pré-processamento, especialmente útil para imagens de microscopia óptica (GONZALEZ E WOODS, 2002). Depois do pré-processamento, ocorre a segmentação. O principal objetivo da segmentação é dividir a imagem em áreas ou objetos de interesse, ela divide a imagem em regiões significativas para facilitar a análise. Isso pode ser feito de várias maneiras, dependendo do objetivo.

Muitas vezes ocorre do processo de segmentação ao ser adequado. Para corrigir esses defeitos, realiza-se o estágio de pós-processamento. A imagem finalizada após o pós-processamento está pronta para a extração de atributos, que é a fase quantitativa do processo. Nessa etapa, são extraídas características dos objetos presentes na imagem, permitindo a diferenciação entre as classes de objetos. Na etapa final, ocorre o reconhecimento de padrões e a classificação. O reconhecimento atribui uma descrição a um objeto com base nas informações do seu descritor. A classificação dá significado a um conjunto de objetos reconhecidos. Do pré-processamento ao pós-processamento a análise é considerada qualitativa e a partir da extração de atributos, quantitativa (GOMES, 2001).

Aquisição da Imagem

Uma imagem digital pode ser representada como uma matriz, onde os índices das linhas e colunas indicam a posição de um ponto na imagem. Cada par de coordenadas (linha, coluna) corresponde a um elemento dessa matriz, que possui um valor associado ao nível de cinza ou à cor naquele ponto específico. Esses elementos da matriz são conhecidos como pixels, um termo derivado da abreviação de "Picture elements" (elementos de imagem) (GONZALEZ E WOODS, 2002).

Uma das maneiras de apresentar a distribuição de intensidade dos pixels em uma imagem digital é por meio do histograma. O histograma de uma imagem digital com kk níveis de cinza é uma função discreta, representada pela equação (1):

$$p(k) = \frac{nk}{n} \quad \text{(I)}$$

Onde: k = nível de cinza, podendo variar entre 0 (preto) e 255 (branco);

nk = número de pixels na imagem com o nível de cinza k;

n = número total de pixels na imagem;

p(k) = estimativa da probabilidade de ocorrência do nível de cinza k. A soma das probabilidades de todos os eventos elementares, isto é, ∑p(k) (será k) igual a 1, satisfazendo a teoria das probabilidades.

Por oferecer uma visão geral da aparência da imagem, o histograma é uma das características mais importantes a ser analisada. O histograma é representado graficamente com a intensidade dos pixels, que varia entre 256 tons, no eixo horizontal, e a probabilidade de ocorrência desses tons de cinza na imagem, no eixo vertical (GONZALEZ E WOODS, 2002).

Embora o histograma não revele o conteúdo da imagem, a informação que ele oferece é extremamente útil para seu processamento (Gonzalez & Woods, 1992 apud Gomes, 2001). A análise estatística da distribuição dos pixels e dos níveis de brilho e contraste fornecida pelo histograma permite ao operador melhorar as condições de captura, resultando em imagens de

melhor qualidade desde o início. Se necessário, um histograma mais equilibrado pode ser obtido através de realces na fase de pré-processamento das imagens. Além disso, o histograma desempenha um papel crucial na segmentação, pois os picos de intensidade correspondem às diferentes fases presentes, possibilitando a separação e/ou quantificação de cada uma delas (IGLESIAS, 2008).

Pré-Processamento

O pré-processamento tem como objetivo melhorar a imagem, corrigindo algum defeito proveniente de sua aquisição e/ou realçando detalhes importantes para a análise (GOMES, 2001). Para que a próxima etapa, (etapa de segmentação) produza resultados satisfatórios, é essencial que a imagem esteja com o mínimo de imperfeições. Por isso, a etapa de pré-processamento é de suma importância. São três os principais pré-processamentos encontrados no PADI:

- Correção de Defeitos: Remoção de ruídos causados durante a aquisição das imagens;
- Realce de Imagem: Aumento do contraste, brilho e nitidez para destacar detalhes relevantes;
- Ajustes geométricos: Correção de distorções e ajustes geométricos.

Em imagens médicas, normalmente é necessário realizar um pré-processamento para corrigir ou realçar a imagem adequadamente, melhorando seu contraste, reduzindo ruído, corrigindo pixels defeituosos ou permitindo que técnicas avançadas de processamento sejam mais eficientes. As imagens médicas digitais geralmente não são adequadas para visualização direta sem qualquer tipo de processamento (SILVA, PATROCÍNIO E SCHIABEL, 2019) (Figura 5).

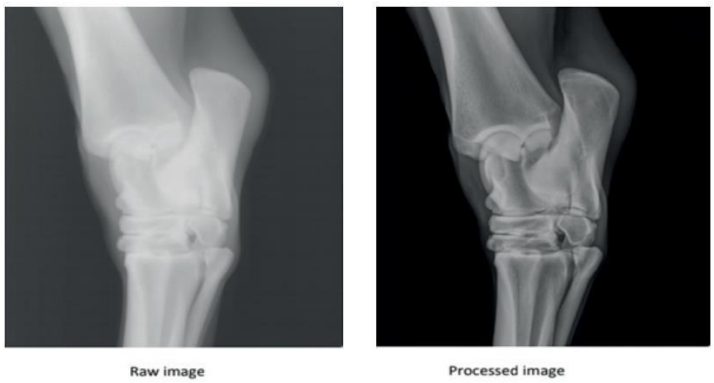


Figura 5 – Imagem de um pé após melhoramento de contraste. Fonte: Silva, Patrocínio e Schiabel, 2019.

Capturar imagens em condições ideais implica em considerar diversos fatores como a exposição correta, o equilíbrio de branco, a estabilidade da câmera e a composição adequada do cenário. Quando esses elementos são cuidadosamente controlados, o resultado é uma imagem que já está próxima da perfeição em termos de qualidade e fidelidade.

Se o procedimento de captura for realizado de forma cuidadosa, em condições corretas, como foi mencionado anteriormente, não se tornam necessárias muitas operações de correção nas imagens adquiridas. Quando todos os parâmetros da câmera são ajustados adequadamente e as condições de iluminação são otimizadas, a qualidade das imagens capturadas tende a ser alta. Isso reduz a necessidade de ajustes, economizando tempo e recursos durante o pós-processamento.

Em suma, embora a captura cuidadosa sob condições corretas minimize a necessidade de correções extensas, o pós-processamento continua a ser uma etapa importante para garantir que as imagens adquiridas atinjam o nível de excelência desejado.

Segmentação

A segmentação é o processo de dividir uma imagem em regiões com características estruturais semelhantes. Os atributos mais básicos utilizados na segmentação de uma imagem incluem a amplitude da luminância dos pixels, as bordas e as texturas. As técnicas de segmentação de imagens têm como objetivo separar a imagem em regiões de interesse, permitindo a posterior classificação ou reconhecimento dos objetos presentes na imagem. A segmentação é considerada uma das tarefas mais desafiadoras no processamento de imagens, pois a precisão dessa etapa é crucial para o sucesso das análises, classificações e reconhecimentos de padrões subsequentes (GONZALEZ E WOODS, 1992 apud SILVA, PATROCÍNIO e SHIABEL 2019).

Existem diversos métodos de segmentação (limiarização de histograma, detecção de bordas, textura e morfologia matemática, etc.), cada qual mais adequado a uma aplicação específica (VIANA et. al, 2017). Um exemplo de uma imagem resultante da segmentação é uma imagem binária, onde os pixels pretos representam o fundo ou elementos que não são de interesse,

enquanto os pixels brancos correspondem aos objetos de interesse que serão quantificados, ou o inverso (IGLESIAS, 2008 apud GOMES, 2001).

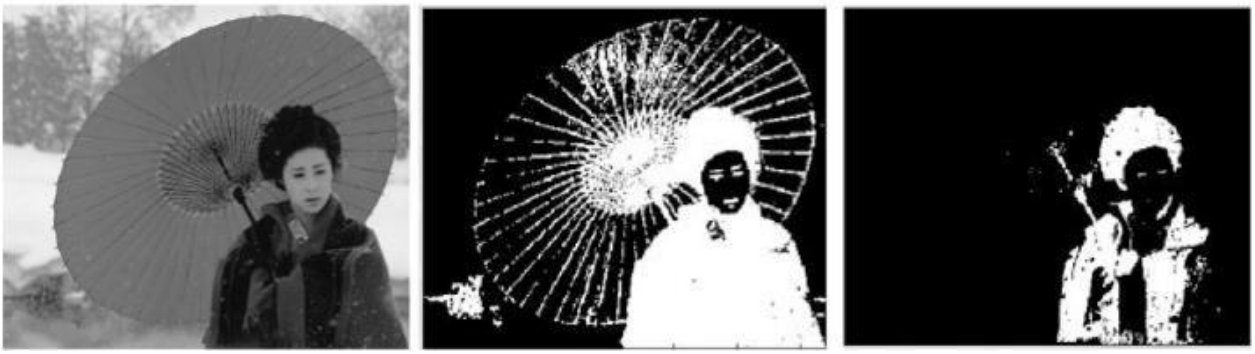


Figura 5: Da esquerda para a direita, a imagem original, seguida pela mesma imagem aplicando-se um limiar 30 e 10. Fonte: Melo, 2011.

Detecção de Bordas

As propriedades dos objetos, como suas características geométricas e físicas, influenciam os tons de cinza da imagem, o que permite sua representação. Como dito no tópico anterior, para detectar e extrair informações dos objetos, diversas técnicas de processamento de imagens são empregadas, incluindo a detecção de bordas. Dependendo da finalidade, a detecção de bordas pode ser um objetivo por si só ou um passo inicial para etapas subsequentes. Independentemente disso, é crucial que a estratégia de detecção de bordas seja eficiente e confiável para obter os resultados desejados. No entanto, ao diferenciar a imagem para detectar variações nos tons de cinza (bordas), todas as variações nos níveis de cinza são destacadas, incluindo bordas espúrias, que são variações indesejadas (DO VALE E DAL POZ, 2002).

Detectar bordas em imagens por vezes é considerada uma tarefa complexa, mas é essencial, pois nos permite identificar e separar objetos de maneira clara em muitas situações. Intuitivamente, podemos imaginar uma borda como a linha que delimita um objeto. No entanto, essa definição não é muito prática para computadores, pois queremos encontrar as bordas primeiras, sem precisar definir o objeto previamente. Para resolver isso, adotamos uma abordagem diferente: em vez de tentar delimitar o objeto primeiro, identificamos as bordas ao detectar onde a intensidade de luz na imagem muda drasticamente. Essas mudanças bruscas na luminosidade são

o que o computador reconhece como bordas (HEISE E SALUSTIANO 2020). A Figura 6 apresenta uma imagem preta que abruptamente passa torna-se branco. É justamente essa mudança na cor que nos faz identificar uma borda.

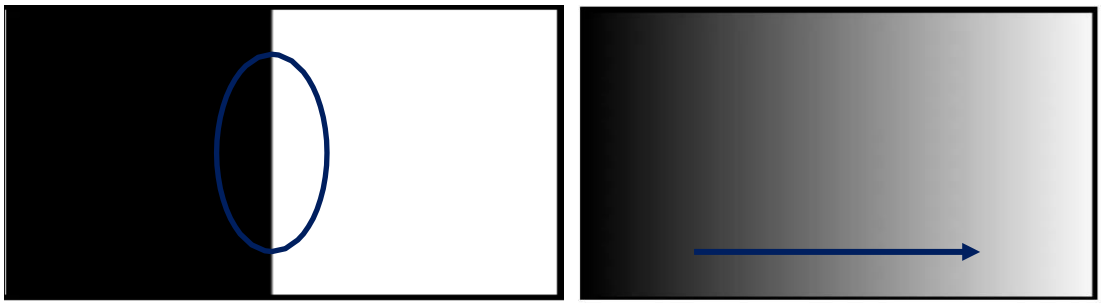


Figura 6 – Detecção de Bordas. Fonte: Heise e Salustiano, 2022

Pode-se observar que a transição de cores do preto para o branco é gradual, ou seja, a mudança não é "abrupta". Por isso, não se pode identificar uma borda clara. Para que o computador possa detectar essas bordas, é preciso um pouco mais de precisão. Em cálculo numérico, o “gradiente” é um vetor que aponta na direção em que uma função cresce mais rapidamente. Se o módulo desse gradiente for grande em um ponto, significa que a função está variando muito naquele ponto. Quando se trata de detectar bordas em imagens, considera-se a imagem como uma função. Para encontrar as bordas, utilizamos "kernels" específicos em uma operação chamada convolução. Esses kernels ajudam a calcular numericamente os gradientes, ou seja, as mudanças na intensidade da imagem, permitindo identificar onde as variações são mais acentuadas e, portanto, onde estão as bordas (HEISE E SALUSTIANO, 2020).

Um kernel é uma pequena matriz de números usada em operações de processamento de imagem, especialmente na técnica de convolução, permite realizar operações de filtragem e transformação em uma imagem. Ele age como um filtro que passa sobre a imagem para realizar tarefas como suavização, detecção de bordas, nitidez, entre outras. Em termos simples, o kernel define como os pixels vizinhos de uma imagem devem ser combinados para modificar o valor de um pixel central. Isso é feito ao multiplicar cada elemento do kernel pelos pixels correspondentes da imagem e, em seguida, somar os resultados. O valor final substitui o pixel original. Por exemplo, um kernel de 3x3 que detecta bordas pode ter valores positivos e negativos que, ao

serem aplicados a uma área da imagem, realçam as diferenças abruptas na intensidade dos pixels, o que é interpretado como uma borda (HEISE E SALUSTIANO, 2020).

A Figura 7 apresenta um exemplo simples de um kernel 3x3 usado para detectar bordas horizontais, quando esse kernel é aplicado sobre uma imagem, ele destaca as áreas onde há uma transição significativa de claro para escuro (ou vice-versa) na direção horizontal.

-1	0	+1
-2	0	+2
.1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

GY

Figura 7: Exemplo de um Kernel 3x3. Fonte: Adaptado de Heise e Salustiano, 2024.

A operação mais comum realizada com um kernel é a convolução (Figura 8). Durante a convolução, o kernel é "passado" sobre a imagem, multiplicando seus valores pelos valores da imagem e somando os resultados para obter um novo pixel na imagem de saída. Kernels típicos são pequenos, como 3x3, 5x5 ou 7x7 pixels. O tamanho do kernel influencia o tipo de efeito que ele terá na imagem.

0 -1	0 0	0 1	0	0	0	0	0
0 -2	0 0	0 2	0	0	255	255	255
0 -1	0 0	0 1	0	255	255	255	255
0	0	0	255	255	255	255	0
0	0	0	255	255	255	0	0
0	0	255	255	255	255	0	0
0	0	255	255	255	255	0	0

0	0	255	765	510	0
0	255	765	765	255	255
0	765	1020	255	255	765
255	1020	765	0	765	1020
765	1020	255	0	1020	1020

Figura 8 - Demonstração de convolução do Kernel. Fonte: Heise e Salustiano, 2022.

A Figura 9 ilustra um exemplo de segmentação em sua forma mais elementar. A imagem em 256 tons de cinza, adaptada de Parcionik (2003), exibe dois tipos distintos de células: uma com um tom de cinza mais claro e outra com um tom mais escuro, sobre um fundo preto, tom 0. A Figura 9–a é segmentada para criar uma imagem binária (Figura 9–b), na qual apenas um tipo de célula é destacado. Isso significa que as células segmentadas correspondem aos objetos brancos de interesse específico, que se diferenciam do fundo preto (OLIVEIRA, 2004 apud PARCIONIK, 2003).

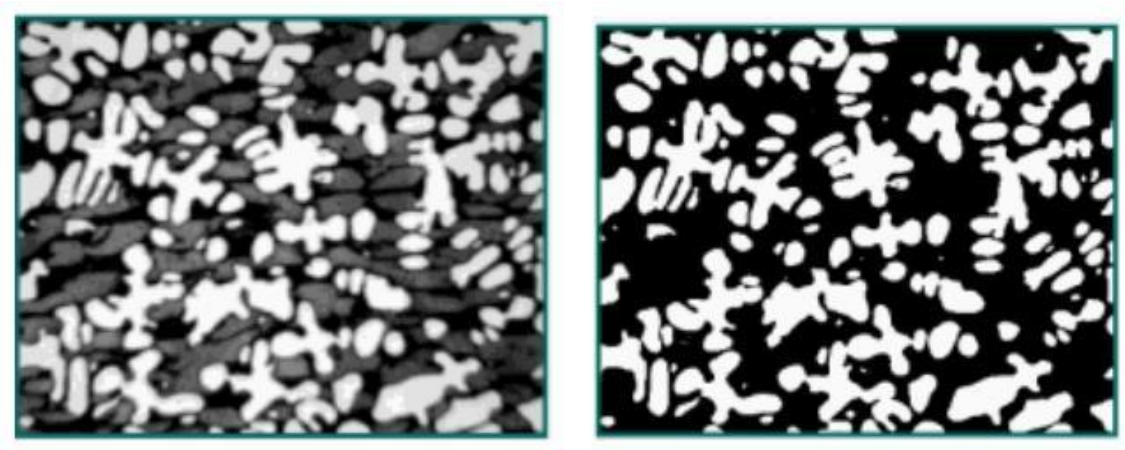


Figura 9 - Exemplo de segmentação com mudança no fundo da imagem. Fonte: Parcionik, 2003.

Devido à sua complexidade, a segmentação é uma das tarefas mais desafiadoras do PADI e pode determinar o sucesso ou o fracasso das análises subsequentes de análise computadorizada, classificação e reconhecimento. Por isso, deve ser realizada com extremo cuidado. A escolha manual do tom de corte raramente é precisa ou reprodutível, tornando necessária a seleção automática para garantir maior robustez ao processo. Na segmentação automática, a determinação do limiar é baseada na análise dos histogramas das imagens. Um dos métodos de segmentação mais comuns utiliza os mínimos do histograma como critério para a escolha do tom de corte entre as fases, ou seja, os limiares correspondem às tonalidades intermediárias entre duas bandas (ou dois picos) (PARCIONIK, 2010).

Pós-Processamento

Para corrigir defeitos residuais nas imagens, é necessário realizar a etapa de pós-processamento. Frequentemente, o resultado da segmentação não é suficiente para que os grupos de pixels segmentados sejam representados e descritos adequadamente em termos de suas características nas etapas subsequentes, nesse sentido O pós- processamento visa aprimorar o resultado da segmentação. Por exemplo, união, separação e eliminação de objetos são procedimentos comuns da etapa de pós-processamento (SERRA, 1988).

A separação de objetos que se tocam, a eliminação de objetos de que não se deseja extrair nenhuma informação e o agrupamento de objetos para a formaçãode objetos mais complexos são exemplos de procedimentos realizados na etapa de pós-processamento. Estes procedimentos são realizados através de operações lógicas e morfológicas (GOMES, 2001; IGLESIAS, 2008).

Uma imagem binária é um tipo de imagem digital onde cada pixel só pode assumir um de dois valores possíveis. Termos como preto e branco, P&B, monocromia e monocromático também são usados para descrever esse conceito. No entanto, esses termos podem igualmente referir-se a imagens que possuem apenas uma amostra por pixel, como no caso das imagens em tons de cinza. A Figura 10 apresenta a imagem binária original e, ao lado, a imagem resultante após a separação das partículas em contato, bem como a eliminação de pequenos defeitos, partículas questionáveis e aquelas que tocam as bordas da imagem (LESSA et al. 2007).

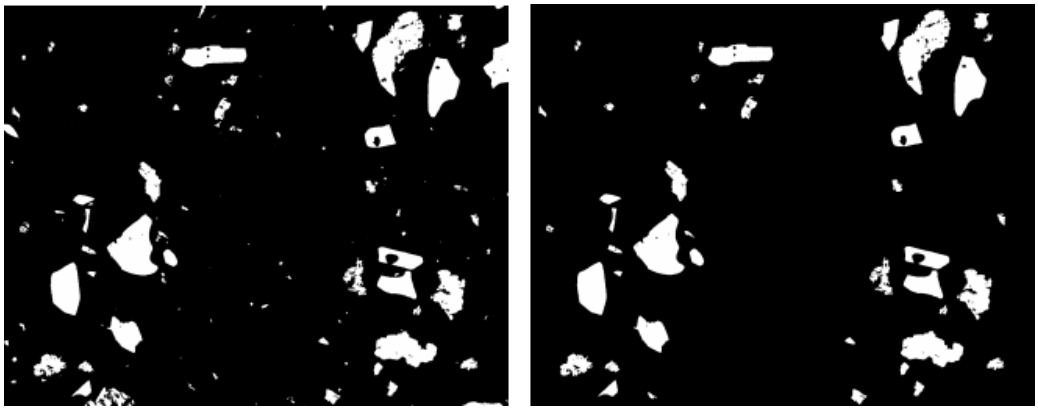


Figura 10 - Imagem binária original e resultante após a separação. Fonte: Lessa et al., 2007.

Extração de atributos

A extração de atributos marca o início da análise efetiva da imagem. Nesta fase, realiza-se a medição em imagens que foram previamente segmentadas, pós-processadas ou mantidas em tons de cinza. A partir dessas medições, os grupos de pixels são descritos por atributos específicos, gerando dados quantitativos que são essenciais para o objetivo final (AUGUSTO, 2012).

A extração de atributos é a fase do processo analítico em que os objetos presentes na imagem são identificados, e características como tamanho, forma, posição e textura são avaliadas, tanto dos objetos quanto da imagem em si (FRIEL, 2000). Posteriormente, na etapa de reconhecimento e classificação.

A Extração de Atributos pode ser dividida em dois tipos principais de medidas: medidas de campo e medidas de região. As medidas de campo avaliam a imagem de forma global, calculando aspectos como o número total de objetos, a área combinada dos objetos e a fração da área que eles ocupam. Essas medidas produzem um único valor para cada parâmetro avaliado. Por outro lado, as medidas de região focam em cada objeto individualmente. Elas extraem características específicas de cada objeto na imagem, como tamanho, forma e localização das partículas (PARCIONIK, 2010).

Reconhecimento e Classificação

O reconhecimento de padrões é uma área de pesquisa focada na classificação e descrição de objetos. Esta disciplina é interdisciplinar, especialmente no campo da informática, e se conecta com estatística, engenharia, inteligência artificial, ciência da computação, mineração de dados, processamento de sinais, e processamento de imagens, entre outros. Suas aplicações incluem reconhecimento automático de caracteres, diagnósticos médicos, monitoramento de correntistas de instituições financeiras, sistemas de recomendação, reconhecimento facial, entre outros (NERI, 2021).

Nesta etapa os dados quantitativos são analisados. O objetivo desta etapa é converter informação em conhecimento. No exemplo apresentado na Figura 11, foram medidos diversos parâmetros de forma das partículas de hematita e esses dados foram fornecidos a um classificador previamente treinado por um mineralogista experiente. A Figura 11 exibe a imagem das partículas de hematita classificadas conforme sua forma; as partículas marcadas em verde foram identificadas como hematita especular, enquanto as em vermelho foram classificadas como hematita porosa (GOMES, 2007).

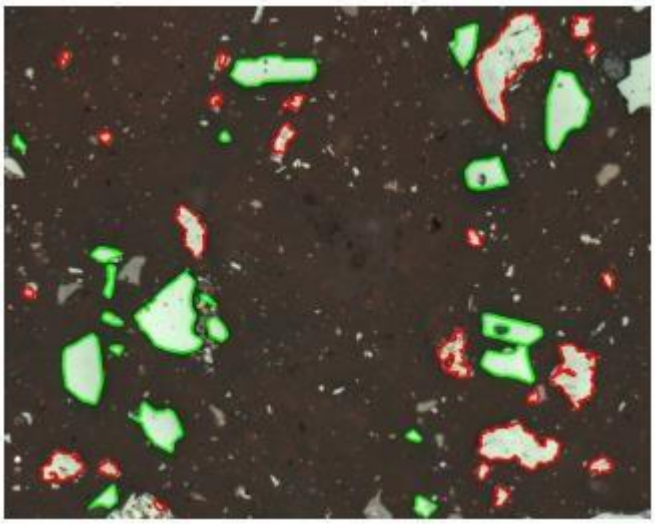


Figura 11 - Ematitas para exemplo de classificador. Fonte: Gomes, 2007.

A fase de reconhecimento de padrões e classificação é a última etapa da sequência padrão do PADI. Um dos principais objetivos da análise de imagens por computador é conferir a uma máquina uma capacidade similar à dos seres humanos na execução de tarefas. Um padrão é uma descrição quantitativa ou estrutural de um objeto ou qualquer outra região de interesse em uma imagem, geralmente realizada por um ou mais descritores, como os mencionados na seção 5.5. Uma classe de padrões é um grupo de padrões que compartilham certas propriedades em comum. O reconhecimento de padrões, portanto, envolve a atribuição automática dos padrões às suas respectivas classes (AUGUSTO 2012 apud GONZALEZ E WOODS, 2002).

4 REDES NEURAIS ARTIFICIAIS

A finalidade das Redes Neurais Artificiais (RNAs) é replicar a estrutura e o processamento paralelo do cérebro humano. Para entender como as RNAs funcionam, é crucial ter um conhecimento básico sobre o funcionamento do cérebro e seus componentes essenciais, os neurônios. É importante compreender como os neurônios se conectam e como eles adquirem conhecimento (ou aprendem). Esse entendimento permite formalizar o funcionamento do cérebro através de modelos matemáticos, o que é fundamental para o desenvolvimento da neurocomputação (JUNIOR e COSTA, 2007).

Na busca por um sistema de controle compatível com a complexidade inerente à maioria dos processos industriais, numerosos esforços de pesquisa foram realizados em alinhamento com o desenvolvimento de novas ferramentas e tecnologias. Nesse sentido, a utilização de Redes Neurais Artificiais (RNAs) está emergindo cada vez mais como uma escolha atraente, especialmente dada a expansão contínua das capacidades computacionais. Como resultado, Oleskovicz et al. (2003) afirmam que em meio aos recentes avanços nas técnicas de inteligência artificial, diversos modelos desta natureza podem ser encontrados na literatura, adaptados para resolver problemas específicos. A teoria subjacente às RNAs apresenta uma alternativa aos algoritmos convencionais que dependem de metodologias determinísticas.

O cérebro humano é uma estrutura biológica que processa informações de maneira complexa e simultânea. A ideia de criar algo artificial que imite o cérebro e sua maneira de pensar surgiu quando McCulloch e Pitts (1943) desenvolveram um modelo matemático básico para um neurônio. Esse modelo usava equações simples para simular como os neurônios realizam operações lógicas básicas.

Mais tarde, Hebb (1949) começou a investigar como ajustar automaticamente as conexões entre neurônios, ou seja, como "aprender" com base no modelo de McCulloch e Pitts. Ele criou uma regra para atualizar essas conexões, o que foi um avanço importante na área.

Rosenblatt (1958) criou o primeiro modelo de rede neural artificial chamado Perceptron. Esta rede tinha uma única camada de neurônios e foi projetada para tarefas simples de classificação.

No entanto, Minsky e Papert (1969) analisaram o Perceptron e mostraram que ele tinha limitações. O Perceptron só conseguia resolver problemas onde as categorias eram separadas por uma linha reta, e não conseguia lidar com problemas mais complexos, como o "OU exclusivo", que não pode ser resolvido apenas com linhas retas.

Em resumo, o desenvolvimento de redes neurais artificiais começou com modelos simples que tentavam imitar o cérebro humano, mas esses primeiros modelos tinham limitações que precisavam ser superadas para resolver problemas mais complexos.

Para entender como tais limitações foram superadas e como as redes neurais funcionam, é essencial conhecer alguns conceitos básicos sobre o cérebro humano e seus componentes principais, os neurônios. Compreender como as conexões entre as células neurais se formam e como se concebe teoricamente o funcionamento matemático desses processos ajuda a esclarecer os fundamentos da *Machine Learning* e das redes neurais. Nos proximos tópicos serão explorados o funcionamento dos neurônios biológicos e dos neurônios matemáticos (perceptron) para então aprofundar nas técnicas de *Machine Learning*.

O Neurônio Biológico

O neurônio, originado do grego e também conhecido como célula nervosa, é a célula responsável pelo processamento dos sinais cerebrais e é considerada a mais complexa do corpo humano em termos de estrutura e função. Além de desempenhar as funções básicas comuns a todas as células, o neurônio é especializado na capacidade de processar e transmitir sinais que carregam informações (BIONDI et al., 2009, p 84).

O neurônio é a unidade fundamental do sistema nervoso, especializado na transmissão de informações através de suas propriedades de excitabilidade e condução de impulsos nervosos. Estruturalmente, o neurônio é composto por três partes principais: a soma ou corpo celular, que contém o núcleo e organelos necessários para a manutenção e síntese celular; os dendritos, que são ramificações que emanam da soma e recebem sinais de outros neurônios; e o axônio, uma projeção longa e fina que se estende da soma e é responsável por conduzir impulsos elétricos até outras células, como neurônios, músculos ou glândulas. Nas extremidades do axônio, localizam-

se os terminais axonais, onde ocorre a liberação de neurotransmissores. A comunicação entre neurônios é facilitada pela sinapse, um processo no qual os neurotransmissores liberados pelos terminais axonais atravessam a fenda sináptica e se ligam aos receptores no neurônio receptor, permitindo a continuidade da transmissão de informações (JAIN, MAO e MOHIUDDIN, 1996).

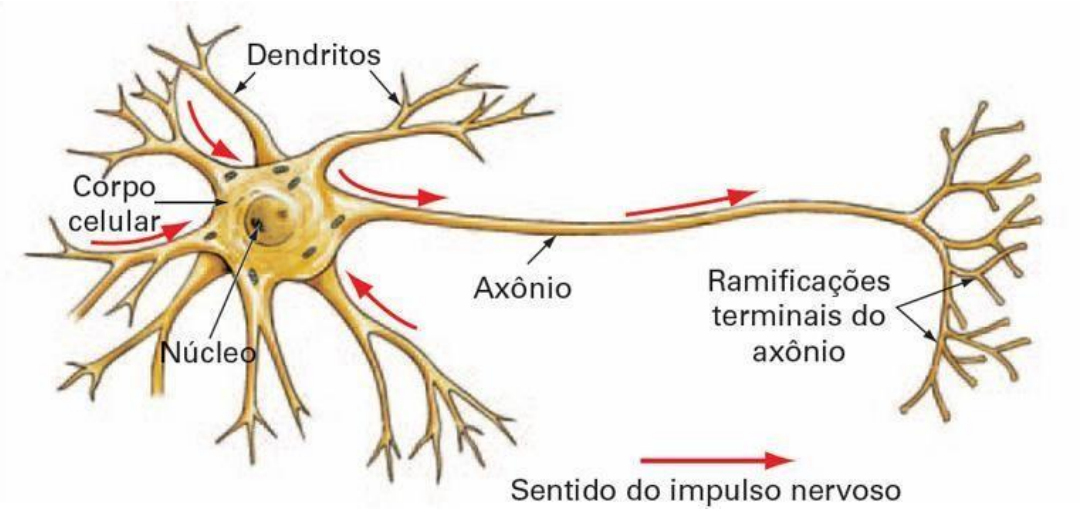


Figura 12 – Representador simplificado de um neorônio biológico. Fonte: https://www.deeplearningbook.com.br/o-neuronio-biologico-e-matematico/

O cérebro humano contém cerca de cem bilhões de neurônios, cada um dos quais estabelece entre 10.000 e 100.000 sinapses com outros neurônios, resultando em um total aproximado de 10^16 sinapses, o que representa a sua capacidade máxima de memória. Os neurônios funcionam em conjunto, formando redes neurais ou redes neuronais biológicas. Por meio de reações elétricas e bioquímicas, essas redes permitem que o cérebro execute simultaneamente uma variedade de funções e tarefas, incluindo processos de aprendizado (OLIVEIRA, 2024).

Os dendritos atuam principalmente como receptores de estímulos provenientes de neurônios vizinhos, conduzindo esses sinais para o corpo celular do neurônio. Neste local, os sinais são processados e geram um impulso na base do axônio. Esse impulso é então transmitido ao neurônio receptor através do axônio, que se conecta aos dendritos do neurônio alvo por meio da sinapse do neurônio emissor (BIONDI et. al., 2009, p. 85).

Para aumentar a velocidade com que os impulsos elétricos são transmitidos pelo axônio e para

proporcionar isolamento, o axônio é envolvido por uma camada chamada bainha de mielina (LENT, 2001). A sinapse, além de sua função estrutural de transferir sinais entre dois neurônios, também desempenha um papel crucial na modulação desses sinais. Ela pode amplificar ou reduzir a intensidade dos sinais, total ou parcialmente, o que confere à sinapse a função de um ponto de decisão no sistema nervoso (KOVACS, 1991).

Então, o ponto de conexão entre a terminação axonal de um neurônio e o dendrito de outro é denominado sinapse. Através das sinapses, os neurônios se integram funcionalmente, formando redes neurais complexas. As sinapses atuam como reguladores, controlando a transmissão dos impulsos nervosos e, consequentemente, o fluxo de informações entre os neurônios dentro da rede neural. A eficácia das sinapses pode variar, e essa variabilidade proporciona ao neurônio a capacidade de adaptação (HAYKIN,2001).

Os sinais elétricos originados de sensores sensoriais, como a retina ocular e as papilas gustativas, são transmitidos pelos axônios. Se esses sinais excedem um determinado limiar de disparo *(threshold),* eles são conduzidos através do axônio. Caso contrário, são bloqueados e não prosseguem, sendo considerados irrelevantes. A transmissão desses sinais não ocorre de forma elétrica, mas química, mediada por neurotransmissores como a serotonina. A passagem do sinal é regulada pelo limiar de disparo, conceito crucial também para o entendimento do neurônio matemático (JAIN, MAO e MOHIUDDIN, 1996).

Como dito anteriormente, um neurônio recebe sinais através de múltiplos dendritos, que são ponderados e encaminhados para o axônio, podendo ser transmitidos ou não, conforme o limiar estabelecido. Durante o percurso pelo neurônio, um sinal pode ser amplificado ou atenuado, dependendo da origem e do peso associado a cada dendrito, sendo este peso um fator que modifica a intensidade do sinal. Os pesos associados aos dendritos representam a memória do neurônio (HAYKIN,2001).

Cada região do cérebro possui uma especialização funcional, como o processamento de sinais auditivos, elaboração de pensamentos ou desejos, e opera por meio de redes neurais específicas interconectadas, permitindo o processamento paralelo. As características dessas redes variam

em termos de número de neurônios, quantidade de sinapses por neurônio, limiares e pesos, entre outros aspectos. Os valores dos pesos são ajustados através do treinamento ao longo da vida, processo que está relacionado à memorização (HAYKIN,2001).

O sistema nervoso humano pode ser compreendido como um processo dividido em três etapas principais, conforme ilustrado no diagrama em blocos da Figura 13 (Arbib, 1987. Adaptado). No centro desse sistema está o cérebro, representado pela rede neural, que desempenha o papel de receber, interpretar e reagir às informações continuamente.

O diagrama apresenta duas direções principais de comunicação. As setas que vão da esquerda para a direita representam a transmissão de informações através do sistema, onde os sinais são processados e transmitidos adiante. Por outro lado, as setas que vão da direita para a esquerda indicam a realimentação, que é a comunicação de volta ao sistema para ajustar ou modificar suas respostas.

Os receptores são responsáveis por captar estímulos do corpo ou do ambiente externo e convertê-los em impulsos elétricos. Esses impulsos são então enviados para a rede neural no cérebro, onde são analisados. Por fim, os atuadores pegam os impulsos elétricos gerados pela rede neural e os transformam em respostas visíveis ou ações, que são as saídas do sistema.

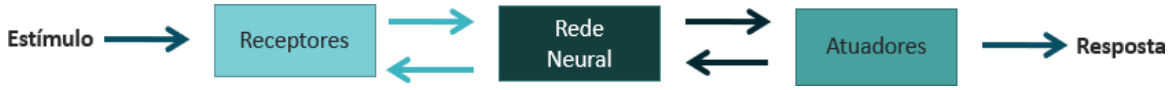


Figura 13 – Diagrama de blocos. Fonte: Adaptado de Arbib, 1937

É importante reconhecer que os níveis estruturais de organização descritos neste tópico são uma caracteristica única do cérebro. e não têm correspondência direta em computadores digitais. Além disso, ainda não conseguimos replicar esses níveis com redes neurais artificiais. No entanto, existem progressos graduais em direção a uma hierarquia de níveis computacionais que se assemelha ao Sistema Nervoso Central. Os neurônios artificiais utilizados na construção das redes neurais são bastante rudimentares em comparação com os neurônios encontrados no cérebro. As redes neurais que podem se projetar atualmente são comparáveis, em termos de sofisticação,

aos circuitos locais e inter-regionais do cérebro.

Apesar dessa limitação, é notável o progresso significativo alcançado nas últimas duas décadas. Com a inspiração da analogia neurobiológica e a vasta gama de ferramentas teóricas e tecnológicas disponíveis, estima que dentro de uma década a compreensão das redes neurais artificiais será muito mais avançada do que é atualmente.

O foco deste trabalho é o estudo das redes neurais artificiais sob a ótica da engenharia. Os capítulos subsequentes se iniciará com a descrição dos modelos de neurônios artificiais que formam a base das redes neurais abordadas.

Inspirados na complexidade e no funcionamento dos neurônios biológicos, os pesquisadores elaboraram um modelo matemático de neurônio que simula, de maneira simplificada, os processos de transmissão e processamento de informações encontrados no cérebro. Esse modelo matemático, que busca replicar as características essenciais dos neurônios naturais, estabeleceu a fundação sobre a qual a Inteligência Artificial foi desenvolvida, permitindo a criação de sistemas computacionais capazes de realizar tarefas complexas e aprender com base em dados, de forma análoga ao funcionamento das redes neurais biológicas.

Perceptron

Nos últimos anos, os modelos baseados em redes neurais artificiais têm recebido considerável atenção devido à sua eficácia na resolução de problemas de Inteligência Artificial, onde outras abordagens mostraram avanços limitados. Desde a concepção do neurônio matemático, diversas arquiteturas e modelos foram desenvolvidos, combinando esses neurônios de maneiras distintas e aplicando uma variedade de técnicas matemáticas e estatísticas. Esse progresso culminou na criação de sofisticadas arquiteturas de Deep Learning, como Redes Neurais Convolucionais, Redes Neurais Recorrentes, Redes Adversárias Generativas (GANs), Redes de Memória, entre outras.

O neurônio matemático é uma versão simplificada de um neurônio biológico, inspirado na maneira como os neurônios geram e transmitem impulsos elétricos. Um neurônio artificial, também conhecido como elemento processador, é uma unidade lógico-matemática que imita o

funcionamento de um neurônio biológico. Assim como o neurônio natural, ele recebe um ou mais sinais de entrada e produz um único sinal de saída. Este sinal pode ser encaminhado para um ou mais neurônios subsequentes por meio de conexões sinápticas artificiais. As intensidades dessas conexões são determinadas pelos pesos sinápticos e pelo viés, que ajustam o sinal de entrada, podendo amplificar ou atenuar seu valor conforme necessário (OLIVEIRA, 2024).

O preceptron é a unidade mais simples de uma rede neural. Se compararmos a uma rede neural ao cérebro humano, os perceptrons seriam os neurônios. Ambos os elementos funcionam da mesma forma: eles recebem uma série de inputs e produzem um único output (VIEIRA, 2020).

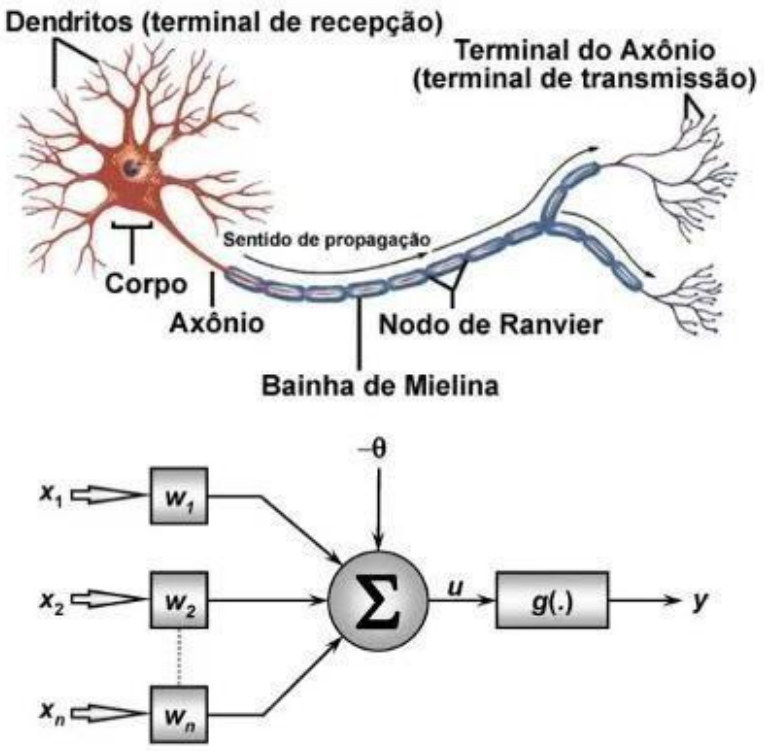


Figura 14 - Redes Neurais Fonte: Caraciolo, 2017.

Esse algoritmo de aprendizagem supervisionado considera um período de treinamento (com valores de entrada e saída) para definir se uma nova entrada pertence a alguma classe específica ou não. Caraciolo (2017) entitula o perceptron como o tipo mais simples de rede neural direta. *(Feedfoward)*, conhecido como classificador linear. Consequentemente, os desafios envolvidos nessa rede neural devem ser aqueles nos quais os padrões podem ser divididos em regiões

distintas por fronteiras lineares de classificação. O Perceptron é um classificador linear, ou seja, os problemas solucionados por ele devem ser linearmente separáveis. O gráfico a seguir mostra um conjunto de pontos bi-dimensional que pode ser separado linearmente – note que é possível passar uma linha reta entre os dois grupos.

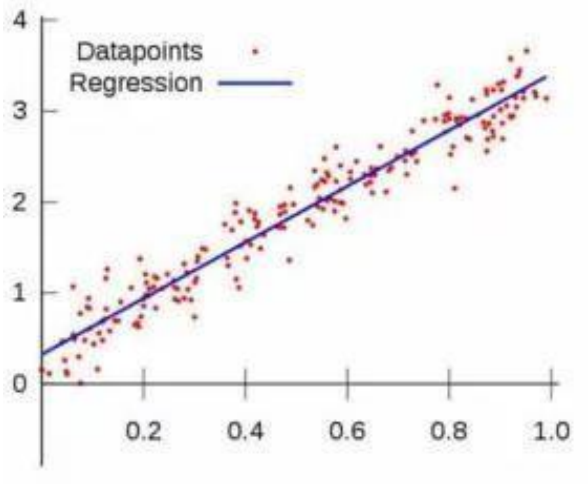


Figura 15 - Gráfico de dispersão. Fonte: Voitto, 2011.

Algorítimos do Precepton

Um neurônio é estimulado por um impulso transmitido pelos dendritos (entradas), ou qual é processado pelo neurônio e resulta na emissão de um segundo impulso. Esse segundo impulso desencadeia a liberação de uma substância neurotransmissora. Essa substância percorre o caminho do corpo celular para o axônio e, posteriormente, é transmitida para outro neurônio. (inputs e outputs). Os sinais de entrada {x1, x2, xn} são ponderados/multiplicados por {w1, w2,w3}. Na Figura 16 podemos ver um exemplo de uma Rede Neural Artificial de 2 camadas com 4 entradas e 2 saídas.

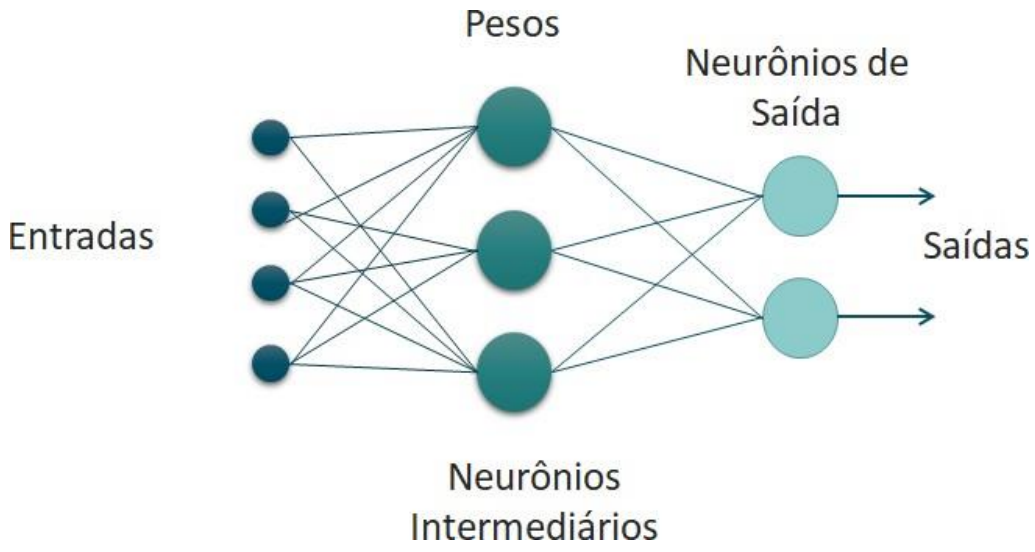


Figura 16 - Rede neural com neurônios matemárticos. Fonte: O Autor, 2024.

Em outras palavras, na representação matemática, as funções dos dendritos e axônios dos neurônios biológicos são simplificadas em sinapses, que são descritas por um valor chamado peso sináptico, representado pela letra w. Quando os sinais de entrada, chamados x, chegam ao neurônio, eles são multiplicados pelos seus pesos sinápticos correspondentes, como x1 multiplicado por w1, e assim por diante. Isso resulta em entradas ponderadas, o que constitui uma das operações fundamentais das redes neurais artificiais: a multiplicação de matrizes.

$$\boldsymbol{Xw} = \boldsymbol{y}$$

$$\begin{bmatrix} 1 & x_{11} & \dots & x_{1d} \\ 1 & x_{21} & \dots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \dots & x_{nd} \end{bmatrix} \times \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Figura 17 - Multiplicação de Matrizes Entre Sinais de Entrada x e Pesos Sinápticos w (versão simplificada). Fonte: deeplearningbook.com/o-neuronio-biologico-e-matemático

Em resumo o neorônio artificial funciona em quatro etapas, da seguinte forma: a primeira é a “Recepção dos Sinais”, onde o neorônio recebe um ou mais sinais de entrada , que são numeros representando informações, a segunda etapa é o “Processamento” onde os sinais de entrada

passam por conexões que possuem pesos específicos , esses pesos determinam a importância de cada sinal. A terceira etapa é a “Produção de Sinal”, que após processar as entradas e ajustar os valores pelos pesos, o neurônio gera um único sinal de saída. A quarta e ultima entrada é a “Transmissão”, nesta etapa o sinal de saída pode ser enviado para outros neurônios na rede, onde o processo se repete.

A limitação do Perceptron em resolver apenas problemas com classes linearmente separáveis impediu que ele fosse eficaz em problemas mais complexos e não-lineares. Essa restrição resultou em uma perda de interesse na pesquisa sobre redes neurais artificiais por um longo período, levando até mesmo a uma paralisação dos estudos na área. Essa situação mudou em 1974, quando Werbos propôs um novo algoritmo capaz de treinar redes neurais com múltiplas camadas, introduzindo o conceito de retropropagação do erro. Esse avanço superou os desafios identificados por Minsky e Papert (1969). No entanto, na época, a descoberta de Werbos não recebeu a atenção necessária e teve pouca repercussão na comunidade científica. Foi então, que nas pesquisas realizadas por Rumelhart, Hinton e William (1986) as regras de treinamento inovadoras introduzidas por Werbos foram amplamente divulgadas, reacendendo o interesse pelas redes neurais artificiais. Desde então, o algoritmo de retropropagação do erro proposto por Werbos tem sido amplamente utilizado e difundido na área (BIONDI et al., 2009).

Perceptron Multi-Camada

O Perceptron Multi-Camada, também conhecido como Multi-Layer Perceptron (MLP), envolve umainterconexão de diversos perceptrons para criar modelos de classificação mais sofisticados. Essa estrutura, também referida como Rede Neural, tem uma capacidade de produção específica altamente confiável, alimentada pelo aprendizado a partir de um conjunto de dados (VIEIRA, 2020).

O autor explana que essa combinação de perceptrons é chamada de Perceptron Multi-Camada (MLP), ou Rede Neural Profunda. A primeira camada é denotada de camada de entrada (input layer), as camadas intermediárias são chamadas de camadas ocultas (hidden layers), e a camada final é a camada de saída (output layer) (Figura 18)..

Ao contrário do perceptron simples, que tem apenas uma camada de entrada e uma camada de saída, o MLP possui uma ou mais camadas intermediárias chamadas camadas ocultas**.** Essas camadas adicionais permitem que o MLP resolva problemas mais complexos e não linearmente separáveis, algo que o perceptron simples não consegue fazer. Este tema será discutido mais adiante, quando abordarmos especificamente o Deep Learning.

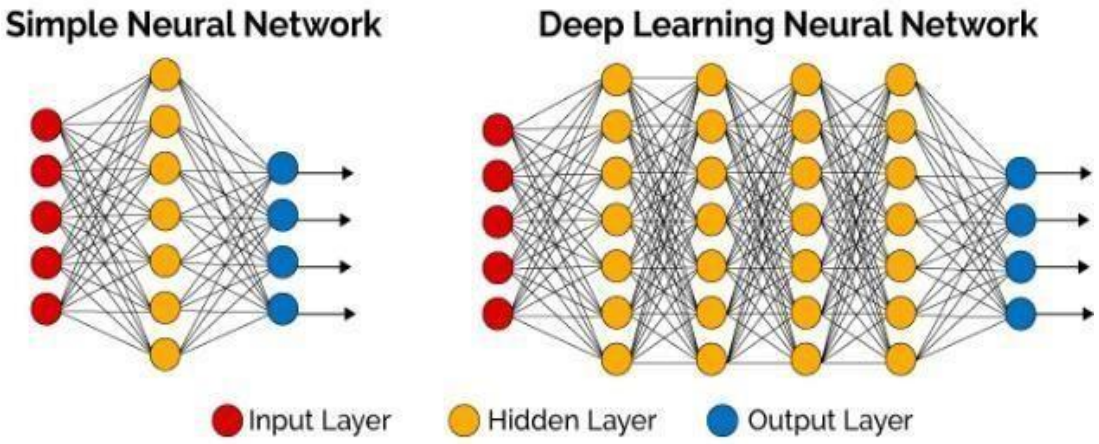


Figura 18 - Rede Neural Simples e Rede Neural Profunda. Fonte: Chagas 2019.

O perceptron multicamadas é amplamente utilizado em diversos setores, como saúde e finanças, conforme discutido por Widrow et al. (1994). No entanto, esse modelo exige que os dados de entrada estejam em um formato unidimensional, como dados tabulares, o que limita sua eficácia em lidar com dados não estruturados, como imagens, textos e áudios. Para aplicar o perceptron multicamadas a esses tipos de dados, é necessário primeiro realizar uma extração de características ou convertê-los em um formato estruturado.

5 MACHINE LEARNING

Machine Learning (ML) é uma área da inteligência artificial (IA) focada na criação e no desenvolvimento de algoritmos computacionais que analisam dados, aprendem e melhoram estes dados e posteriormente aplicam o que aprendem para tomar decisões informadas. (GRIEVE, 2023). Sendo um subcampo da inteligência artificial, que é amplamente definida como a capacidade de uma máquina de imitar o comportamento humano inteligente. Sistemas de inteligência artificial são usados para executar tarefas complexas de uma forma semelhante à maneira como os humanos resolvem problemas (BROWN, 2021).

O *Machine Learning* surgiu da combinação entre a Ciência da Computação e a Estatística. Cada uma dessas áreas tem perguntas fundamentais que guiam seus estudos. Na Ciência da Computação, a principal questão é: "Como podemos construir máquinas que resolvem problemas, e quais problemas são resolvíveis ou não?" Já na Estatística, a pergunta central é: "O que podemos inferir dos dados, baseados em certas suposições de modelagem, e com qual grau de confiança?"

O *Machine Learning* une essas duas áreas, mas com uma pergunta própria. Enquanto a Ciência da Computação foca em como programar manualmente os computadores, o *Machine Learning* se concentra em como fazer com que os computadores aprendam e se programem sozinhos, usando experiência e uma estrutura inicial como base. E, diferente da Estatística, que se preocupa principalmente com as conclusões que podem ser extraídas dos dados, o *Machine Learning* também aborda questões como quais arquiteturas e algoritmos computacionais são mais eficazes para manipular esses dados, como diferentes tarefas de aprendizado podem ser organizadas em um sistema maior, e como lidar com problemas computacionais complexos (MITCHELL, 2006).

A meta do *Machine Learning* é desenvolver programas capazes de aprimorar seu desempenho por meio do uso de exemplos. (MITCHELL, 1997). Isso requer a disponibilidade de uma extensa quantidade de exemplos, que alimentam o conhecimento da máquina, resultando em hipóteses geradas com base nos dados.

Aplicar *Machine Learning* para resolver problemas é um processo complexo que exige uma série de etapas e requisitos prévios para ser bem-sucedido. Cada um desses aspectos para proporcionar uma visão mais clara e completa será detalhado neste tópico.

Primeiramente, um dos requisitos essenciais é a disponibilidade de um conjunto de exemplos representativos, conhecido como *dataset*. Esse *dataset* é composto por dados que o algoritmo de *Machine Learning* usará para aprender e, posteriormente, fazer previsões ou classificações. No entanto, simplesmente possuir dados não é suficiente. Esses dados precisam ser de alta qualidade e representativos do problema que se deseja resolver. Muitas vezes, isso implica a necessidade de coletar novos dados ou melhorar a qualidade dos dados existentes.

O processo de aprimorar a qualidade dos dados envolve várias técnicas de pré-processamento. Isso pode incluir a remoção de valores ausentes, a correção de inconsistências, a normalização de escalas, e a transformação dos dados em formatos mais adequados para o algoritmo que será utilizado. Sem essas etapas de pré-processamento, os dados podem levar o modelo a aprender padrões errados ou irrelevantes, comprometendo os resultados finais (LUDEMIR, 2021).

Além de preparar os dados, é crucial escolher os algoritmos de aprendizado de máquina que serão usados para resolver o problema. Não existe um único algoritmo que funcione bem para todos os tipos de problemas; diferentes problemas exigem diferentes abordagens. Por exemplo, problemas de classificação podem ser resolvidos com redes neurais, máquinas de vetores de suporte (SVM), ou árvores de decisão, mas a escolha do algoritmo mais adequado dependerá das características específicas dos dados e do problema.

Depois de selecionar o algoritmo, é necessário ajustar os parâmetros desse algoritmo, um processo conhecido como ajuste de hiperparâmetros. Por exemplo, ao trabalhar com redes neurais, deve-se decidir o número de camadas e neurônios em cada camada, a taxa de aprendizado, e outros parâmetros que podem influenciar significativamente o desempenho do modelo (LUDEMIR, 2021).

Após o treinamento do modelo, é imprescindível avaliar seu desempenho. Isso não envolve apenas verificar se o modelo resolve o problema, mas também medir a precisão, a sensibilidade, a especificidade que indicam a eficácia do modelo. Avaliar o modelo em dados que ele não viu durante o treinamento, conhecidos como dados de validação ou teste, é uma prática comum para assegurar que ele generalize bem para novos dados.

Finalmente, o processo de *Machine Learning* não termina com a implantação do modelo. Os dados e o ambiente em que o modelo opera podem mudar ao longo do tempo, exigindo atualizações periódicas do modelo. Isso pode incluir o ajuste dos parâmetros, a reavaliação dos dados, ou até mesmo a escolha de um novo algoritmo se o modelo atual não estiver mais funcionando adequadamente (LUDEMIR, 2021).

Essas etapas, que vão desde a coleta e pré-processamento dos dados até a avaliação e manutenção contínua do modelo, são todas essenciais para garantir que a aplicação de *Machine Learning* seja eficaz e que o sistema continue a funcionar bem ao longo do tempo (LUDEMIR, 2021).

Tipos de Machine Learning

Para entender melhor o aprendizado de máquina, é importante explorar os três principais tipos de *Machine Learning*: aprendizado supervisionado, não supervisionado e por reforço. No aprendizado supervisionado, os algoritmos são treinados com dados rotulados, permitindo que eles façam previsões ou classificações baseadas em exemplos anteriores. No aprendizado não supervisionado, o algoritmo recebe dados sem rótulos e deve identificar padrões ou agrupamentos por conta própria. Já no aprendizado por reforço, os algoritmos aprendem a tomar decisões sequenciais através de um processo de tentativa e erro, recebendo recompensas ou penalidades com base nas suas ações.

Esses três métodos abordam problemas de formas distintas, cada um sendo adequado para diferentes tipos de tarefas e desafios.

Modelos de *Machine Learning* supervisionado são treinados usando conjuntos de dados que já possuem rótulos definidos. Esses rótulos permitem que os modelos aprendam a fazer previsões e se tornem mais precisos com o tempo. Por exemplo, um algoritmo pode ser treinado com imagens de cães e outros objetos, todas classificadas por humanos. A partir desses exemplos, a máquina aprende a identificar, por conta própria, quais imagens contêm cães. Esse tipo de aprendizado é o mais comum atualmente (BROWN, 2021).

Já no *Machine Learning* não supervisionado, o programa analisa dados que não têm rótulos ou classificações pré-definidas, buscando identificar padrões ou tendências por conta própria. Esse tipo de aprendizado pode revelar informações que os humanos talvez não estivessem procurando. Por exemplo, um programa de aprendizado não supervisionado pode analisar dados de vendas online e descobrir diferentes perfis de clientes com base em seus comportamentos de compra (BROWN, 2021).

Por fim, o aprendizado por reforço ensina máquinas a tomar decisões por meio de tentativa e erro, utilizando um sistema de recompensas. Esse tipo de aprendizado é usado para treinar modelos a desempenhar tarefas específicas, como jogar um jogo ou dirigir veículos autônomos. A máquina recebe feedback sobre quais decisões foram corretas, permitindo que ela aprenda, ao longo do tempo, quais ações deve tomar para obter os melhores resultados (BROWN, 2021). A Tabela 1 apresenta um resumo das principais características de cada tipo de modelo de aprendizado, destacando suas vantagens, desvantagens e aplicações práticas.

	Aprendizado supervisionado	**Aprendizado não supervisionado**
Definição	Algoritmos que aprendem relações entre atributos de entrada e de saída a partir de conjunto de exemplos rotulados	Algoritmos que buscam encontrar padrões em agrupamentos de dados com características semelhantes, em busca de categorias e desfechos ainda não identificados ou não informados
Vantagens	Análise de múltiplos parâmetros. Solução rápida e automática para questões de grande escala e elevada acurácia	Menor interferência humana na análise dos dados. Excelente excelente para fontes de dados multimodais ou multidimensionais. Permite identificação de novos desfechos
Desvantagens	Necessidade dos dados serem rotulados, o que para grandes volumes de dados pode ser impraticável. Tendência ao sobreajuste dos dados	Custo elevado e técnicas complexas. Necessita grande quantidade de dados para elaboração do algoritmo. Interpretação dos resultados pode ser desafiadora
Principais Tarefas	Regressão, classificação, modelo prognóstico e análise de sobrevivência	Redução da dimensionalidade do problema e agrupamento
Exemplos de Algorítimos	Regressão logística, árvores de decisão, random forests e redes neurais artificiais	Análise das componentes principais, agrupamento hierárquico, autoencoders, análise linear de discriminantes

Tabela 1- Aprendizado supervisionado x Aprendizado não supervisionado. Fonte: Autor, 2024

6 DEEP LEARNING

O *Deep Learning* é um modelo computacional que utiliza várias camadas de unidades interconectadas, (perceptrons), para aprender padrões complexos e representações diretamente a partir de dados brutos. Graças a essa capacidade de aprendizado, o *Deep Learning* tornou-se uma ferramenta essencial para a resolução de problemas desafiadores, impulsionando muitas inovações tecnológicas. No entanto, a construção de um modelo de *Deep Learning* é uma tarefa desafiadora devido à complexidade dos algoritmos e à natureza dinâmica dos problemas encontrados no mundo real. Diversos estudos revisaram os conceitos e aplicações do *Deep*

Learning, focando principalmente nos tipos de modelos e nas arquiteturas de redes neurais convolucionais.

O *Deep Learning* transformou inúmeras aplicações em diversos setores e campos de pesquisa. Sua aplicação pode ser vista em áreas como a saúde (Shamshirband et al., 2021), manufatura inteligente (Wang et al., 2018b), robótica (Pierson e Gashler, 2017) e segurança cibernética (Dixit e Silakari, 2021). Essas técnicas têm sido fundamentais para resolver problemas com alta complexidade, como o diagnóstico de doenças, a detecção de anomalias, a identificação de objetos e a detecção de ataques de malware. Malware é um termo utilizado para qualquer tipo de software malicioso projetado para prejudicar ou explorar um dispositivo, serviço ou rede programável

Enquanto os métodos tradicionais não conseguem lidar adequadamente com essa complexidade, o *Deep Learning* oferece soluções inovadoras que abordam de forma eficaz as crescentes demandas da nossa infraestrutura. A convergência entre IA e engenharia civil é uma fronteira emergente com enorme potencial. Na engenharia civil, Vieira (2020) utiliza-se das técnicas de *Deep Learning* para detectar fissuras em pavimentos rígidos, obtendo uma excelente acurácia e servindo de base para futuras pesquisas na área, Avci et al (2022) utilizam redes neurais convolucionais em diversas aplicações em infraestrutura de engenharia civil, o trabalho de Gao et al (2022) aproveitam da modelagem de redes neurais e da regressão linear para estudarem o comportamento de segurança e personalidade dos trabalhadores da construção civil e o trabalho de Tarawneh e Saleh (2022) utilizaram a estrutura do *Machine Learning* para prever o modo de falha e a capacidade de flexão de vigas reforçadas com FRP (polímero reforçado com fibras).

O *Deep Learning* é um subconjunto da *Machine Learning* Figura 19, que é um campo dedicado ao estudo e desenvolvimento de máquinas capazes de aprender. Na indústria, a aprendizagem profunda é utilizada para resolver tarefas práticas em uma variedade de áreas, como visão computacional (imagens), processamento de linguagem natural (texto) e reconhecimento automático de fala (áudio).

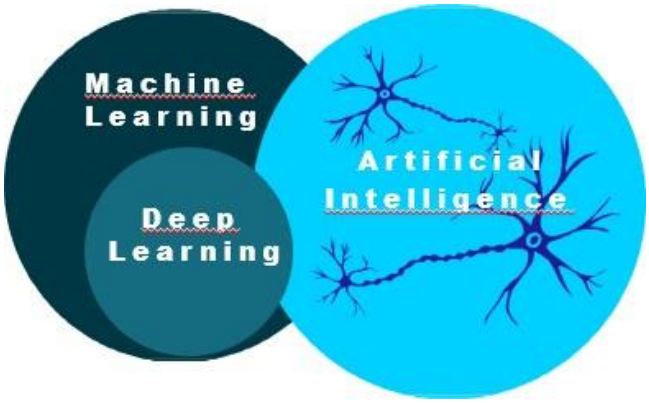


Figura 19 – Subconjuntos da IA. Fonte: O Autor, 2024.

Em resumo, o *Deep Learning* é um conjunto de métodos na caixa de ferramentas de Machine Learning.que se baseia em redes neurais artificiais profundas para extrair padrões e informações de dados,permitindo a realização de tarefas complexas de forma automatizada. (TRASK, 2019)

Antes do ressurgimento do *Deep Learning* (DL) na pesquisa, o reconhecimento de padrões exigia que os dados brutos de entrada, como os valores de pixels de uma imagem, fossem transformados em um vetor de características. Este vetor representava uma abstração dos dados e podia ser usado por modelos de *Machine Learning* para identificar ou classificar padrões. Esse processo dependia fortemente da engenharia e do conhecimento especializado para definir as representações adequadas.

Com o advento do *Deep Learning*, essa etapa de transformação foi automatizada. As redes neurais profundas, através de suas várias camadas de processamento, conhecidas como camadasocultas, conseguem aprender as representações internas dos dados de forma hierárquica. Inicialmente, as primeiras camadas detectam características simples, como bordas e linhas. À medida que os dados são processados pelas camadas subsequentes e mais profundas, essas características básicas são combinadas para formar padrões mais complexos e, eventualmente, características sofisticadas que representam fielmente os dados de entrada. Esse processo de aprendizado e extração de características continua através das camadas até que uma previsão seja gerada no final (NOOR, IGE 2024).

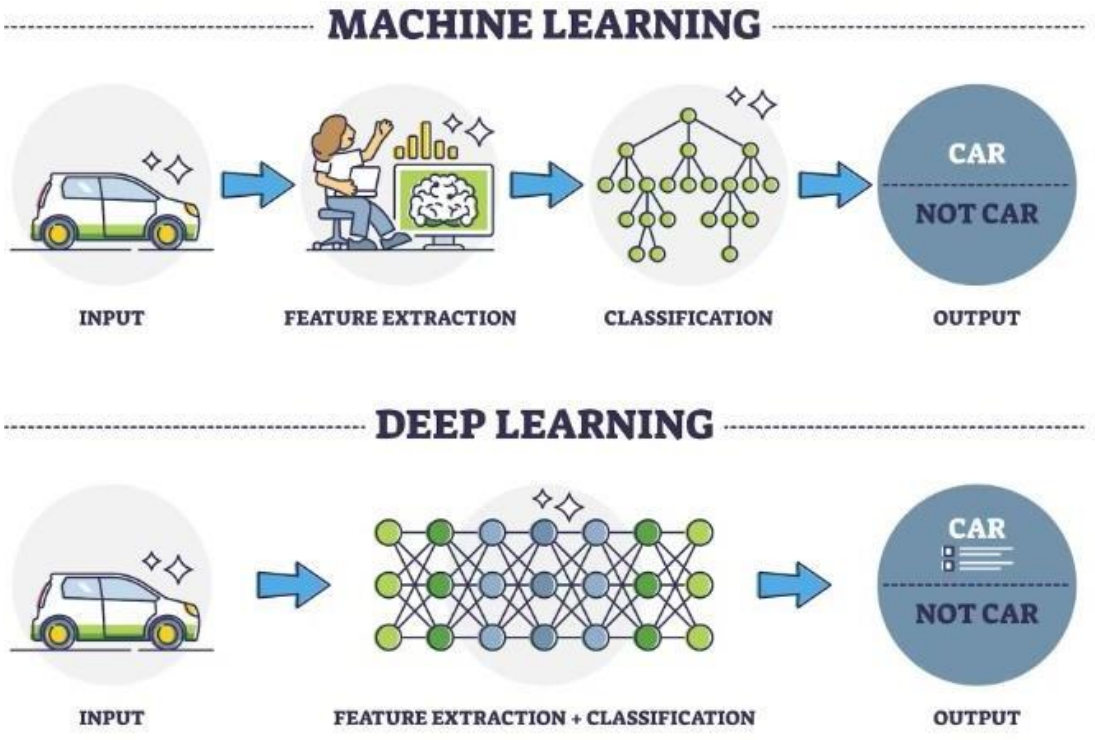


Figura 20 – Diferença entre *Machine Learning* e *Deep Learning*.

Fonte: https://www.turing.com/kb/ultimate-battle-between-deep-learning-and-machine-learning

Vários estudos foram conduzidos para discutir o conceito e a aplicação do *Deep Learning* nos últimos anos. Os estudos abordaram ou focaram em vários aspectos do aprendizado profundo, como tipos de modelos de aprendizado profundo, abordagens e estratégias de aprendizado, arquiteturas de rede neural convolucional (CNN), aplicações e desafios do aprendizado profundo. Em Dong et al. (2021) , os autores forneceram fundamentos de aprendizado profundo e destacaram diferentes tipos de modelos de aprendizado profundo, como redes neurais convolucionais, autocodificador e redes adversariais generativas. Em seguida, as aplicações do DL em vários domínios são discutidas, e alguns desafios associados às aplicações do DL são apresentados. Outra pesquisa Talaei Khoei et al. (2023) forneceu uma análise abrangente de abordagens de aprendizagem supervisionada, não supervisionada e de reforço e comparou as diferentes estratégias de aprendizagem, como aprendizagem online, federada e de transferência. Finalmente, os desafios atuais da aprendizagem profunda e a direção futura são discutidos.

Em Alzubaidi et al. (2021) , os autores forneceram uma revisão abrangente das arquiteturas das Redes Neurais Convolucionais populares usadas em tarefas de visão computacional, destacando seus principais recursos e vantagens. Em seguida, fazem as aplicações do DL em imagens médicas e os desafios são discutidos. Uma pesquisa semelhante é relatada em Alom et al. (2019) , onde os diferentes modelos de *Deep Learning* supervisionados e não supervisionados são destacados,e as arquiteturas populares de Redes Neurais Convolucionais são comparadas e discutidas. Em outra pesquisa, Pouyanfar et al. (2018), os autores se concentraram nas aplicações de DL em visão computacional, processamento de linguagem natural e processamento de fala e áudio. Os diferentes tipos de modelos de aprendizado profundo também são discutidos. Em Sarker (2021), os autores se concentraram nos diferentes tipos de modelos de *Deep Learning* e forneceram um resumo das aplicações de DL em vários domínios.

Camadas

Um modelo de *Deep Learning* se distingue pela presença de múltiplas camadas ocultas. Essas camadas ocultas desempenham um papel crucial ao aprender e extrair características complexas dos dados fornecidos. Cada camada oculta consiste em diversos neurônios, que são os componentes fundamentais de uma rede neural. Esses neurônios recebem múltiplas entradas, e cada entrada é associada a um peso específico que regula o fluxo de informações para o neurônio durante o processo de transmissão do sinal (ou "forward pass"). Durante essa transmissão, a soma ponderada das entradas é calculada, e uma função de ativação é aplicada a essa soma para produzir a saída do neurônio, que é então passada para a próxima camada na rede (NOOR, IGE 2024).

Uma camada oculta em que cada neurônio é conectado a todos os neurônios da camada anterior é conhecida como camada totalmente conectada. A Figura 21 esboça uma representação gráfica de um neorônio

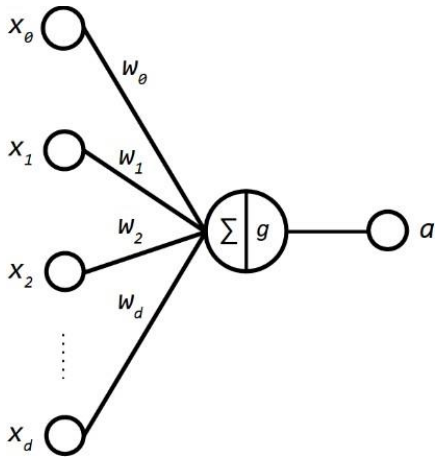


Figura 21 - Funcionamento de um Neorônio Matemático. Fonte: Noor e Ige, 2024.

O ponto central do aprendizado profundo reside na sua habilidade de automaticamente extrair características hierárquicas dos dados de entrada. Essa extração automática ocorre em duas camadas especializadas: a camada convolucional e a camada de *pooling* (agrupamento). Na camada convolucional, cada neurônio se conecta apenas a uma região específica dos dados de entrada, com os pesos sendo compartilhados entre essas conexões. Esse compartilhamento de pesos não só reduz drasticamente a quantidade de parâmetros na rede neural, como também capacita a rede a aprender as mesmas características em diferentes posições espaciais daentrada, como descrito por LeCun et al. (2015). A Figura 22 demonstra a aplicação de um filtro 3x3 em uma imagem bidimensional de 9x9 pixels. A camada convolucional processa os dados deentrada ao mover o filtro por todos os pixels da imagem, resultando em um conjunto de valores de saída conhecido como mapa de características.

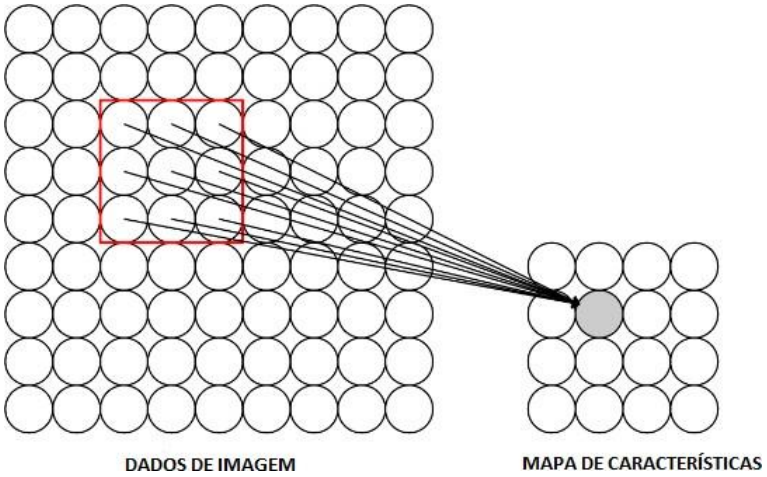


Figura 22 - Aplicação de um filtro 3x3 em uma imagem bidimensional. Fonte: Noor e Ige, 2024.

Camadas de *pooling* são frequentemente utilizadas após várias camadas convolucionais para gradualmente diminuir as dimensões espaciais dos mapas de características gerados. Essa redução espacial é obtida ao sintetizar as informações contidas em pequenas regiões dos mapas de características. As operações de *pooling* mais comuns envolvem a seleção do valor máximo ou a média dos valores dentro dessas regiões locais. Esse processo não apenas diminui o número de parâmetros que a rede precisa processar, mas também garante que as características obtidas sejam invariantes a pequenas translações na entrada, conforme destacado por LeCun et al. (2015). Ao decorrer da pesquisa, a camada de agrupamento *(pooling*) será detalhadamente explanada.

7 REDE NEURAL

A Rede Neural Convolucional (CNN) é um tipo de modelo de rede neural que aproveita as informações espaciais locais dos dados através do uso de camadas convolucionais. A Figura 23 apresenta uma arquitetura típica de CNN, que é composta por camadas convolucionais, camadas de *pooling* e camadas totalmente conectadas. As camadas convolucionais e de *pooling* são alternadas para extrair automaticamente características importantes de forma hierárquica. Após a extração, essas características são encaminhadas para as camadas totalmente conectadas, responsáveis pela previsão das saídas. Antes de serem processados pelas camadas totalmente conectadas, os mapas de características finais precisam ser convertidos em um vetor

unidimensional, o que é feito por meio do processo de *flattening* (achatamento). A arquitetura da CNN é fundamental para melhorar o desempenho preditivo, pois foi projetada para extrair de maneira eficiente a representação dos recursos dos dados de entrada, proporcionando um reconhecimento de padrões mais preciso e robusto. Nos últimos dez anos, diversas arquiteturas CNN foram desenvolvidas, com foco em aprimorar a capacidade de aprendizado de características e em superar desafios como o gradiente de desaparecimento (NOOR, IGE 2024).

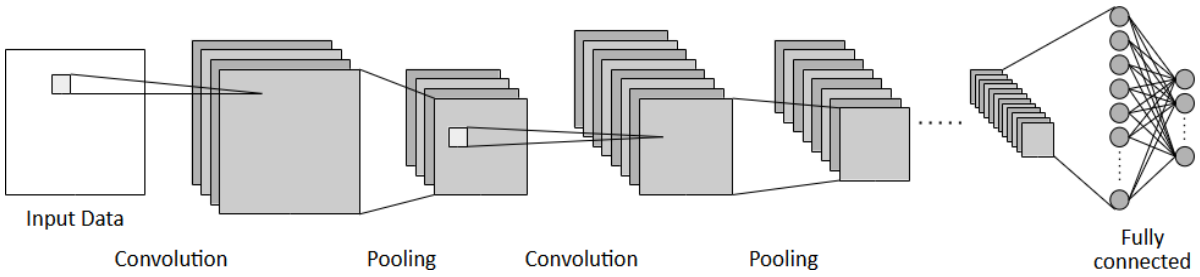


Figura 23 - Uma rede neural com camadas convolucionais e de agrupamento seguidas por camadas totalmente conectadas. Fonte: Noor e Ige, 2024.

As Redes Neurais Convolucionais (CNNs) são uma categoria específica de arquitetura de redes neurais profundas, fundamentada na inspiração neurobiológica originada a partir do trabalho de (HUBEL et al., 1962; HUBEL et al., 1977). Nesse estudo, os autores revelaram que o córtex visual degatos exibe uma organização hierárquica, onde um conjunto de células responde a pequenas sub-regiões, chamadas campos receptivos.

Originárias do termo em inglês "Convolutional Neural Networks" (CNNs), as Redes Neurais Convolucionais representam um gênero particular de Redes Neurais Artificiais (ANN), concebida pelo cientista francês Yann LeCun (LECUN, et al., 1998). Desde sua concepção, as CNNs demonstraram eficácia notável na abordagem de desafios de classificação, emergindo como uma alternativa plausível aos métodos convencionais para resolver esse tipo de questão. Tomaremos a analogia do modelo neurobiológico mencionado acima, as camadas convolucionais representando células simples, exibindo um mecanismo de conexões locais ecompartilhamento de pesos. A camada *“pooling”* é usada para fornecer dados àtarefa de classificação (VIEIRA, 2020).

Os resultados originados da camada de *pooling* servem de entrada para a camada totalmente conectada. Essa camada desempenha um papel crucial na geração das saídas da rede,como quais são específicas do resultado final da tarefa de classificação (DE FARIA, 2018).

Como dito anteriormente, a classificação de imagens envolve a tarefa de identificar e categorizar objetos ou características em imagens, como reconhecer faces, identificar tipos de objetos ou classificar cenas. o campo de classificação de imagens está em constante progresso, com novos modelos sendo criados e aprimorados para alcançar resultados mais precisos, à medida que os desafios vão sendo identificados e superados.

A evolução da pesquisa nesse campo tem sido impulsionada pela necessidade de enfrentar os desafios que dificultam a obtenção de resultados precisos. Esses desafios podem incluir variações nas condições de iluminação, ângulos de visão, qualidade da imagem e a complexidade das características presentes nas imagens. Para superar esses obstáculos e melhorar a precisão dos modelos de classificação, muitos pesquisadores têm desenvolvido e estudado novas técnicas e algoritmos.

AlexNet foi um dos primeiros modelos de Redes Neurais Convolucionais (CNN) a ganhar reconhecimento e sucesso significativo, marcando um avanço importante no campo do aprendizado profundo aplicado à visão computacional (KRIZHEVSKY et al., 2012). O modelo é composto por cinco camadas convolucionais, onde operações de *max-pooling* são realizadas após a primeira e segunda camadas, seguidas por três camadas totalmente conectadas. As primeiras duas camadas convolucionais utilizam filtros de 11×11 e 5×5, respectivamente, enquanto as camadas restantes empregam filtros de 3×3. A função de ativação ReLU é utilizada para mitigar o problema do gradiente de desaparecimento.

O VGG-16, por outro lado, aprimora a arquitetura das CNNs ao adicionar mais camadas convolucionais, totalizando até 19 camadas, com o objetivo de capturar representações mais complexas dos dados de entrada (SIMONYAN E ZISSERMAN, 2015). Assim como no AlexNet, a função de ativação ReLU é utilizada para reduzir o gradiente de desaparecimento. Diferentemente do AlexNet, o VGG-16 emprega um tamanho fixo de filtro pequeno de 3×3 em

todas as camadas convolucionais, e o *max-pooling* é aplicado após duas ou três camadas convolucionais consecutivas. Essa configuração permite ao modelo extrair características mais detalhadas e reduzir o número de parâmetros.

O ZFNet, um modelo clássico de CNN, segue um princípio arquitetônico semelhante ao AlexNet, com cinco camadas convolucionais, seguidas por camadas de *max-pooling* após a primeira e a segunda convolução, e três camadas totalmente conectadas (ZEILER E FERGUS, 2013). As principais diferenças incluem o uso de tamanhos de filtro menores e passos nas camadas convolucionais, além da normalização de contraste dos mapas de características, o que permite ao modelo capturar características mais eficazes e melhorar o desempenho geral.

A seguir, abordaremos os tópicos de detecção de objetos, segmentação de imagens e geração de imagens. Estes temas representam áreas fundamentais na visão computacional e no aprendizado profundo.

Detecção de Objetos

A segmentação de imagens é uma tarefa crucial na qual o aprendizado profundo desempenha um papel significativo. Um dos modelos pioneiros para segmentação de imagens utilizando aprendizado profundo é a rede totalmente convolucional (FCN) proposta por Long et al. (2015). Esse tipo de rede é composto exclusivamente por camadas convolucionais, permitindo que processe entradas de tamanhos variados e gere mapas de segmentação previstos que mantêm as mesmas dimensões da entrada.

A detecção de objetos avançou significativamente com o aprendizado profundo, principalmente através das Redes Neurais Convolucionais (CNN). A R-CNN foi pioneira ao combinar CNN com propostas de regiões seletivas, permitindo a extração de características e classificação de objetos. Modelos aprimorados como Fast R-CNN e Faster R-CNN melhoraram o desempenho, mas ainda apresentavam desafios em aplicações de tempo real devido ao alto custo computacional (NOH et al, 2015).

Para resolver essa questão, foram desenvolvidos detectores de estágio único, como YOLO e SSD,

que eliminam a necessidade de propostas de região, oferecendo uma detecção mais rápida e eficiente. O RetinaNet introduziu a perda focal para lidar com o desequilíbrio de classes, enquanto o EfficientDet melhorou a detecção multiescala (NOOR, IGE 2024).

Outra inovação importante foi o uso de técnicas de pós-processamento, como a supressão não máxima (NMS) e suas variantes, para aprimorar a seleção das caixas delimitadoras mais precisas.

Por fim, o Detection Transformer (DeTR) simplificou a detecção de objetos ao eliminar componentes tradicionais como caixas de ancoragem e NMS, utilizando a arquitetura de transformadores. Entretanto, o DeTR enfrentou desafios como longos tempos de treinamento e dificuldades na detecção de objetos pequenos, que foram abordados por versões aprimoradas como o DeTR Deformável e o DeTR Dinâmico, que introduziram representações multiescala e codificação baseada em convolução para melhorar a eficácia do modelo. (NOOR, IGE 2024). Redes neurais convolucionais especializadas, como a YOLO (You Only Look Once) Figura 24 e a Mask R-CNN, são projetadas para identificar objetos em uma cena, com a Mask R-CNN, em particular, sendo capaz de segmentar os pixels correspondentes a esses objetos.

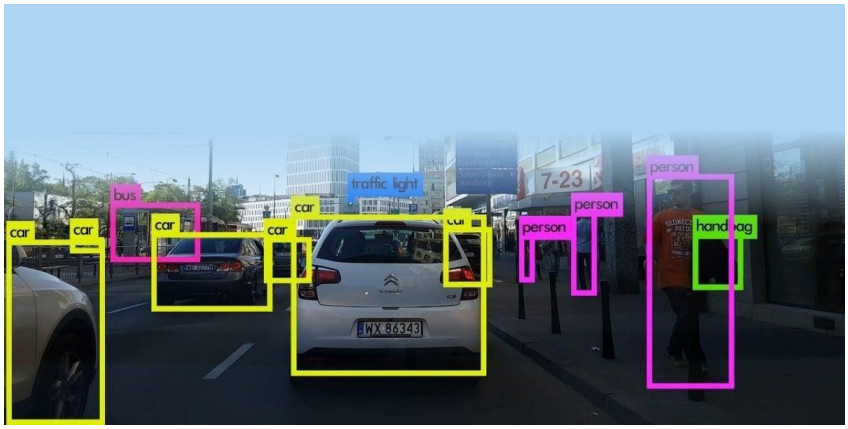


Figura 24 - Detecção de objetos (carros) utilizando redes especializadas. Fonte: YOLO, 2024.

O modelo YOLO detecta objetos em tempo real e se destaca por sua eficiência e rapidez. Ao contrário dos métodos tradicionais de detecção de objetos, que geralmente utilizam um processo em dois estágios (primeiro, gerando propostas de região e depois classificando essas regiões), o YOLO adota uma abordagem de estágio único. Isso significa que ele prevê diretamente as caixas

mitadoras e as pontuações de confiança para os objetos em uma única passagem pela rede neural.

Figura 25 - Detecção de objetos (girafas) utilizando redes convolucionais. Fonte: YOLO, 2024.

Segmentação da Imagem

A segmentação de imagens é uma tarefa essencial em visão computacional, na qual o aprendizado profundo tem causado avanços significativos. Essa tarefa envolve a divisão de uma imagem em várias partes ou segmentos, com o objetivo de identificar e classificar cada região da imagem, pixel a pixel. O impacto do aprendizado profundo nessa área é notável, particularmente com o desenvolvimento de modelos mais eficientes e precisos (NOOR, IGE 2024).

Um dos primeiros modelos de aprendizado profundo voltados especificamente para segmentação de imagens foi a Rede Totalmente Convolucional, proposta por Long et al. em 2015. Este modelo marcou um avanço importante porque, ao contrário das redes tradicionais que combinam camadas convolucionais e totalmente conectadas, a Rede Totalmente Convolucional (FCN) é composta apenas por camadas convolucionais. Essa arquitetura permite que a rede

receba imagens de qualquer dimensão como entrada, processando-as para gerar um mapa de segmentação de saída que mantém as mesmas dimensões da entrada (LONG ET AL, 2015).

A FCN opera removendo as camadas totalmente conectadas de uma rede tradicional, substituindo-as por camadas convolucionais que preservam a espacialidade dos dados ao longo do processo de convolução. Isso é crucial para a tarefa de segmentação, pois permite que a rede atribua uma classe a cada pixel da imagem, resultando em um mapa de segmentação detalhado. O mapa de segmentação resultante não só identifica a presença de diferentes objetos na imagem, mas também localiza esses objetos com precisão, delineando seus contornos (LONG ET AL, 2015).

Além disso, a arquitetura da FCN facilita o aprendizado de características em múltiplas escalas, capturando tanto os detalhes finos quanto as informações contextuais mais amplas da imagem. Isso é possível porque as camadas convolucionais preservam as relações espaciais entre os pixels durante todo o processo, permitindo que a rede compreenda a estrutura completa da imagem ao mesmo tempo em que identifica suas diferentes partes (NOOR, IGE 2024).

Essa abordagem revolucionou a segmentação de imagens, abrindo caminho para a criação de modelos ainda mais sofisticados, como as Unets e as *Deep*Lab, que se baseiam em princípios semelhantes, mas introduzem refinamentos adicionais para melhorar a precisão e a aplicabilidade em diferentes contextos. A contribuição das Redes Totalmente Convolucionais foi, portanto, fundamental para estabelecer as bases da segmentação de imagens com aprendizado profundo, permitindo avanços significativos em áreas como a análise médica, a automação industrial e a realidade aumentada (NOOR, IGE 2024).

A segmentação de imagens pode ser facilmente confundida com duas outras operações: a identificação de objetos e regiões, e a classificação e rotulagem. A identificação de objetos e regiões foca em isolar ou destacar elementos dentro da imagem, sem necessariamente determinar a natureza ou o significado desses elementos. Já a classificação e rotulagem envolve a atribuição de categorias específicas a esses elementos identificados, determinando o que eles são e a quais classes pertencem, por meio de rótulos nomeados (PIEMONTEZ, 2024).

Figura 26 - Segmentação na imagem utilizando redes convolucionais. Fonte: YOLO, 2024.

Geração de Imagens

A geração de imagens envolve criar imagens a partir de descrições textuais e pode ser dividida em três estágios: extração de características do texto, geração da imagem e controle da qualidade da saída. O segundo estágio afeta diretamente a qualidade das imagens geradas. O autocodificador variacional (VAE) foi um dos primeiros modelos a gerar imagens, capturando a distribuição dos dados de treinamento e usando-a para criar imagens, embora com qualidade limitada. A introdução das GANs (Redes Adversárias Generativas) melhorou significativamente a qualidade, utilizando um gerador e um discriminador que competem entre si para criar imagens mais realistas (NOOR, IGE 2024).

Modelos como o StackGAN dividiram o processo em duas etapas para refinar detalhes e melhorar a resolução das imagens. O StackGAN++ e o HDGAN foram aprimoramentos que incorporaram geradores múltiplos e discriminadores hierárquicos para criar imagens multiescala. O DM-GAN introduziu uma rede de memória para corrigir imagens geradas de baixa qualidade. O AttnGAN foi o primeiro a integrar mecanismos de atenção, permitindo ao modelo focar em palavras relevantes durante a geração da imagem. Variações como o ResFPA-GAN e o DualAttn-GAN

melhoraram essa abordagem ao incorporar módulos de atenção adicionais para capturar melhor o contexto textual e os detalhes visuais, resultando em imagens mais realistas (NOOR, IGE 2024).

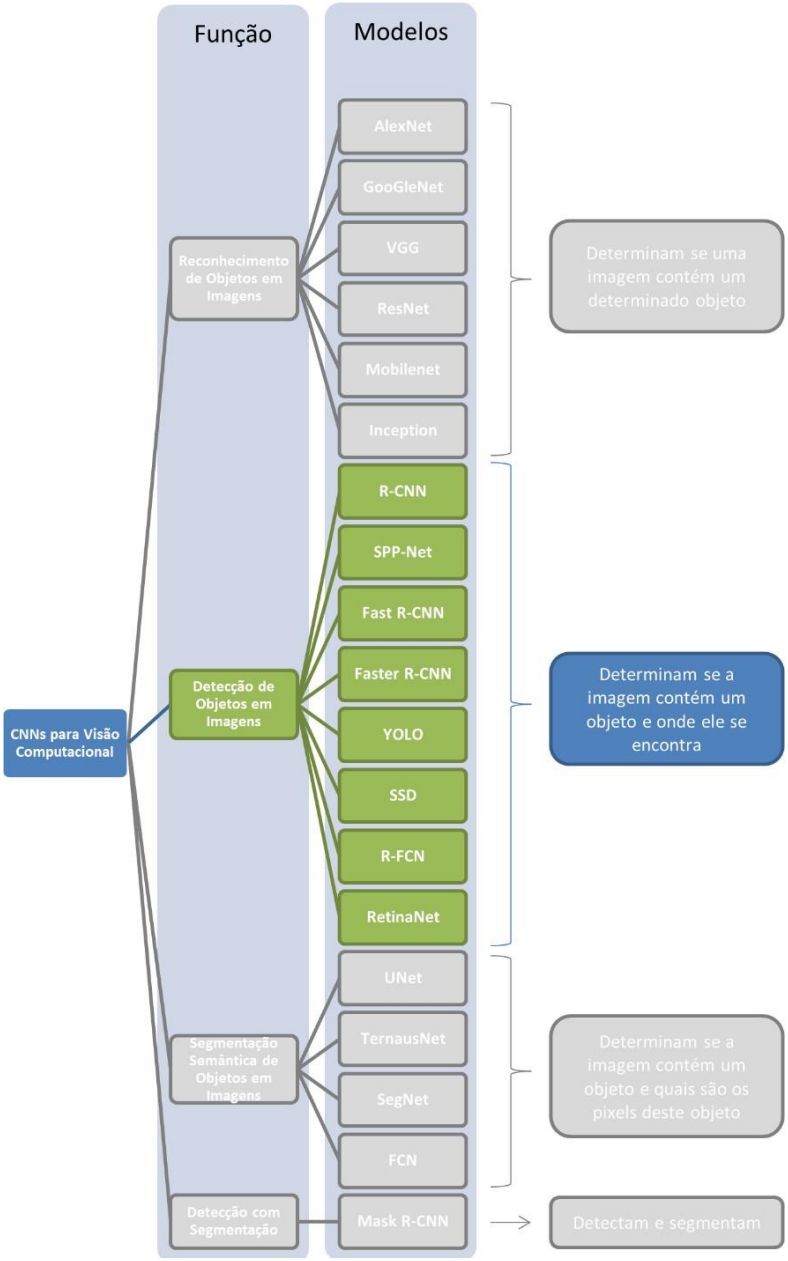


Figura 27 - CNN's para visão computacional com função e modelo. Fonte: Noor e Ige, 2024.

Esses modelos de redes neurais são projetados para duas principais funções: (a) identificar objetos em imagens e (b) determinar a região onde esses objetos estão localizados. A saída dessas redes geralmente é uma "bounding box" (caixa delimitadora), que é um retângulo convexo que

indica a área de maior probabilidade de presença do objeto detectado, como ilustrado pela imagem do campus da UFSC no topo desta página. Essas redes são úteis tanto para a detecção de objetos em imagens quanto para a estimativa aproximada de seu número e localização (WANGENHEIM, 2024).

Nesse contexto, por essas redes serem capazes de localizar um objeto e permitir, através da *bounding box*, a segmentação da subimagem onde o objeto de interesse está presente, elas podem ser classificadas tanto como redes de reconhecimento de objetos quanto como redes de segmentação de objetos.

Existem duas principais variantes dessas redes, a primeira são Classificadores de regiões associados a extratores de características baseados em CNN, os exemplos incluem R-CNN, Faster R-CNN, entre outros. A segunda são Redes neurais convolucionais de disparo único para reconhecimento de objetos (WANGENHEIM, 2024).

A Figura 28 oferece uma visão geral da quantidade de novos modelos de redes neurais desenvolvidos, organizando-os por ano de publicação e pela conferência onde foram apresentados pela primeira vez.

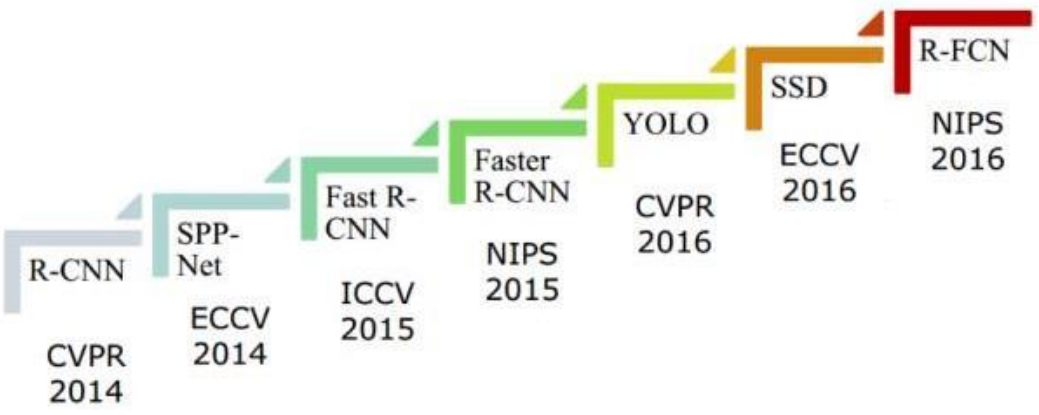


Figura 28 - Visão geral de novos modelos de redes neurais. Fonte: Noor e Ige, 2024.

8 ARQUITETURA DO MODELO

Diversas técnicas de processamento de imagens (IPTs) têm sido aplicadas para a detecção de defeitos em infraestruturas civis, com o intuito de substituir parcialmente as inspeções presenciais realizadas por humanos. Essas técnicas são especialmente utilizadas para manipular imagens, a fim de extrair características associadas a defeitos, como rachaduras em superfícies de concreto e aço. No entanto, a aplicação das IPTs enfrenta desafios significativos devido às variáveis do mundo real, como mudanças de iluminação e presença de sombras, que podem dificultar sua adoção generalizada.

Para superar essas limitações, Cha *et al*. (2017) (Figura 29) propuseram um método alternativo, baseado em visão computacional, que utiliza *Deep Learning* para a detecção de rachaduras em concreto, sem a necessidade de calcular previamente as características dos defeitos. As CNNs que como ja dito anteriormente, possuem a capacidade de aprender automaticamente as características das imagens, permitem que o método proposto funcione sem a utilização das IPTs tradicionais para a extração de características.

A CNN desenvolvida pelos autores foi treinada com um conjunto de 40 mil imagens, cada uma com resolução de 256 × 256 pixels, resultando em uma precisão de aproximadamente 98%. Além disso, a CNN treinada foi combinada com uma técnica de janela deslizante, que possibilita a análise de imagens de qualquer tamanho maior que 256 × 256 pixels (CHA et al, 2017).

A robustez e adaptabilidade do método proposto foram avaliadas em um conjunto de 55 imagens de alta resolução (5.888 × 3.584 pixels), capturadas de uma estrutura diferente daquelas utilizadas no treinamento e validação. Essas imagens foram obtidas sob diversas condições, como iluminação intensa, sombras e presença de rachaduras extremamente finas (CHA et al, 2017).

Estudos comparativos também foram realizados para avaliar o desempenho da CNN em comparação com métodos tradicionais de detecção de bordas, como os algoritmos de Canny e Sobel. Os resultados demonstraram que a abordagem baseada em CNN superou significativamente esses métodos tradicionais, mostrando-se eficaz na detecção de rachaduras em concreto mesmo em condições desafiadoras (CHA et al, 2017).

Os autores utilizaram Redes Neurais Convolucionais (CNNs) para desenvolver um classificador capaz de detectar rachaduras em concreto a partir de imagens partindo de dois objetivos. O primeiro objetivo foi criar um classificador robusto, que seja menos suscetível a interferências causadas por fatores como iluminação variável, sombras, desfoque, entre outros, garantindo assim uma alta adaptabilidade a diferentes condições. O segundo objetivo foi estabelecer um banco de testes inicial, que poderá ser utilizado por outros pesquisadores para detectar não apenas rachaduras, mas também outros tipos de danos estruturais, como delaminação, vazios, desprendimento e corrosão em elementos de concreto e aço. A principal vantagem da abordagem proposta, que utiliza CNNs para a detecção de rachaduras em concreto, é que ela elimina a necessidade de extrair e calcular características manualmente, como é necessário nas abordagens tradicionais. Essa pesquisa foi estruturada de maneira a explorar e detalhar esses aspectos de forma que fique competente ao entendimento.

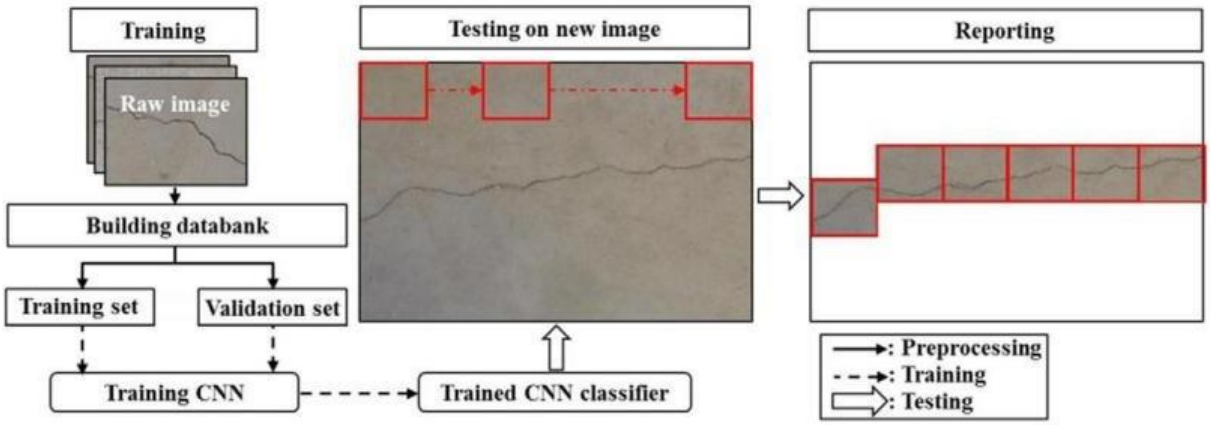


Figura 29 – Esquema da arquitetura proposta por Cha. Fonte: Cha et al., 2017

A Figura 29 mostra o fluxo geral do método proposto por Cha *et al*. (2017) com etapas de treinamento (linhas sólidas) e etapas de teste (linhas tracejadas). Para treinar um classificador CNN, imagens brutas de superfícies de concreto com uma ampla gama de variações, incluindo iluminação, sombras, etc., capazes de potencialmente acionar alarmes falsos, foram capturadas de um edifício com uma camera do modelo DSLR. Um total de 332 imagens brutas foram usadas (ou seja, 277 imagens com resoluções de 4.928 × 3.264 pixels para treinamento e validação e 55 imagens para teste com resoluções de 5.888 × 3.584 pixels). Aproximadamente 85% para as duas

primeiras etapas (Treinamento e Validação) e 15% para a etapa de Teste.

O processo descrito envolveu 277 imagens, que foram divididas em imagens menores com resolução de 256 × 256 pixels. Essas imagens menores foram manualmente classificadas como contendo rachaduras ou estando intactas, formando assim um banco de dados (DB). A partir desse banco de dados, as imagens recortadas foram escolhidas aleatoriamente para compor dois conjuntos: um para treinamento e outro para validação.

As imagens do conjunto de treinamento foram então utilizadas para treinar uma Rede Neural Convolucional (CNN), com o objetivo de criar um classificador capaz de diferenciar entre imagens de concreto rachado e concreto intacto. Após o treinamento, o classificador foi testado com o conjunto de validação para verificar sua precisão.

Uma vez validado o classificador, ele foi aplicado a um novo conjunto de 55 imagens de alta resolução (5.888 × 3.584 pixels) de concreto. Essas novas imagens foram analisadas pelo classificador, que gera um relatório identificando a presença de rachaduras.

A estrutura básica de uma Rede Neural Convolucional (CNN) pode ser formada por diferentes tipos de camadas, incluindo camadas de entrada, convolução, *pooling*, ativação e saída. Nas camadas de convolução e *pooling*, são executadas as operações de convolução e *pooling*, respectivamente. Quando a arquitetura da rede inclui um grande número de camadas, ela é considerada uma CNN profunda. Além disso, camadas auxiliares, como *dropout* e normalização em lote (*Batch Normalization*, BN), podem ser integradas dentro dessas camadas principais, dependendo do objetivo específico do uso. Para conduzir este estudo, foi utilizado o MatConvNet (VEDALDI e LENC, 2015).

A Tabela 2 apresenta a descrição das camadas que compõem a rede neural. Após as primeiras camadas de convolução, são implementadas camadas de normalização em lote *(batch normalization).* Essas camadas padronizam a entrada, ajustando a média do output para perto de zero e o desvio padrão para próximo de 1 (um). Isso acelera o treinamento, diminui a chance de *overfitting*. O *overfitting* ocorre quando o modelo se ajusta demais aos dados de treinamento, perdendo sua capacidade de generalizar para novos dados. e garante que os dados sejam

processados dentro de um intervalo controlado, o que é vantajoso em métodos numéricos e computacionais.

Além disso, uma camada de ativação ReLU (Rectified Linear Unit) é aplicada antes das camadas densas que aparecem no final da rede. Na última camada, utiliza-se uma camada densa com dois nós, seguida de uma camada *softmax*, que gera dois outputs: uma probabilidade de ausência de fissura e outra de presença de fissura. O resultado final é determinado pela maior dessas probabilidades (VIEIRA, 2020).

Layer	*Height*	*Width*	*Depth*	*Operator*	*Height*	*Width*	*Depth*	*No.*	*Stride*
Input	256	256	3	C1	20	20	3	24	2
L1	119	119	24	P1	7	7	–	–	2
L2	57	57	24	C2	15	15	24	48	2
L3	22	22	48	P2	4	4	–	–	2
L4	10	10	48	C3	10	10	48	96	2
L5	1	1	96	ReLU	–	–	–	–	–
L6	1	1	96	C4	1	1	96	2	1
L7	1	1	2	Softmax	–	–	–	–	–
L8	1	1	2	–	–	–	–	–	–

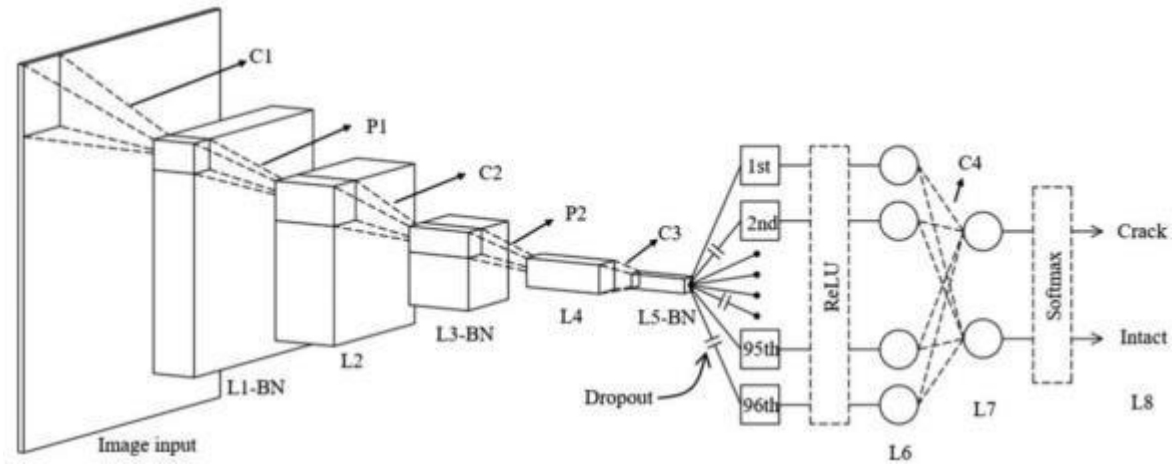


Tabela 2 - Arquitetura proposta por Cha et al, 2017. Fonte: Cha et al., 2017

As Imagens são escolhidas aleatoriamente do DB para gerar conjuntos de treinamento e validação. A razão para escolher o tamanho de recorte relativamente pequeno é que uma rede treinada em imagens pequenas permite a varredura de qualquer imagem maior do que o tamanho projetado. No entanto, se imagens menores do que as selecionadas aqui forem usadas,

a rede pode captar quaisquer características alongadas, como arranhões. Além disso, imagens menores também dificultam a anotação de imagens como defeituosas ou intactas. O DB gerado incluiu uma ampla gama de variações de imagens para um classificador de *Deep learning*.

A rede neural treinada alcançou uma precisão de 98,22% em um conjunto de 32.000 imagens e 97,95% em um conjunto de validação de 8.000 imagens. Com base em um estudo paramétrico, recomenda-se o uso de mais de 10.000 imagens para garantir uma robustez adequada. (CHA et al, 2017) (Figura 30).

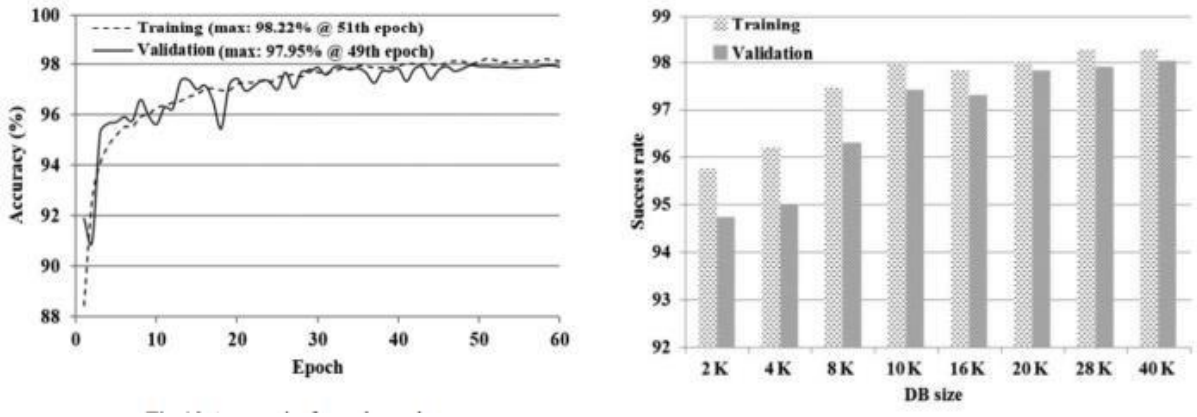


Figura 30 - Precisão alcançada no treinamento de Cha. Fonte: Cha et al., 2017

O desempenho da CNN treinada foi testado em 55 imagens de alta resolução (5.888 × 3.584 pixels). Essas imagens foram analisadas utilizando uma técnica de janela deslizante, que permitiu a avaliação de qualquer imagem maior que 256 × 256 pixels. A partir dessa análise, foram gerados mapas de rachaduras. Os resultados demonstraram que a CNN manteve um desempenho consistente, mesmo diante das condições variadas das imagens de teste, como iluminação intensa, sombras, desfoque etc... Além disso, o desempenho do método proposto não foi suscetível qualidade das imagens, especificações da câmera e distância de trabalho. (CHA et al, 2017)

Nos estudos comparativos, a CNN proposta demonstrou um desempenho superior em relação aos métodos tradicionais de detecção de bordas, como Canny e Sobel. Segundo os autores, esses métodos tradicionais não foram eficazes na detecção de rachaduras, especialmente em

superfícies de concreto com variações de cor e textura. A CNN, por outro lado, mostrou-se particularmente eficiente na identificação de rachaduras finas e em condições de iluminação desafiadoras, além de produzir resultados com menos ruído. No entanto, segundo os autores, a eficácia da CNN depende de uma grande quantidade de dados de treinamento.

Embora as abordagens baseadas em visão, incluindo IPTs e CNNs, sejam limitadas na detecção de características internas devido à natureza das imagens fotográficas, a CNN continuará a ser aprimorada pelos autores para identificar diversos tipos de danos superficiais em estruturas de concreto e aço.

No final do estudo, os autores esperam futuramente que esse método substitua parcialmente as inspeções visuais e seja combinado com drones autônomos para monitoramento contínuo de infraestruturas civis, este foi o estudo que Vieira (2020) realizou.

9 ESQUADRIAS

Definição

A ABNT NBR 10821 define esquadrias como elementos de fechamento aplicados em vãos de edificações, com a função de proporcionar ventilação, iluminação, proteção contra intempéries, isolamento termoacústico, e segurança. Esquadrias incluem janelas, portas, fachadas-cortina, e outros componentes similares, sendo projetadas para atender a requisitos específicos de desempenho mecânico e funcional, conforme estipulado pela norma.

As esquadrias desempenham um papel essencial na delimitação e proteção entre os ambientes internos e externos de uma edificação. Sua função primordial é facilitar a entrada de luz natural nos espaços interiores, o que é fundamental para a eficiência energética. Isso se deve ao fato de que a iluminação natural reduz a dependência da iluminação artificial, promovendo, assim, uma economia de energia elétrica ao longo do dia (GUERRA, 2007).

As esquadrias, para edificações, desempenham um papel fundamental não apenas na funcionalidade e na composição estética dos edifícios, mas também na garantia de desempenho eficiente em termos de conforto ambiental e sustentabilidade. A escolha e o posicionamento

adequados das esquadrias podem influenciar diretamente a qualidade do ambiente interno, contribuindo para a iluminação natural, ventilação, conforto térmico e acústico, além de aspectos de segurança e durabilidade (MARTINS, 2017).

Figura 31 - Tipos de esquadrias. Fonte: http://www.alphaesquadrias.ind.br/esquadrias.php 2024

Requisitos de Desempenho

De acordo com a NBR 15575 (2013) para que as esquadrias desempenhem suas funções de maneira eficaz, é necessário que atendam a uma série de critérios de desempenho, regulamentados por normas técnicas e padrões de qualidade. Entre os principais requisitos, destacam-se:

Iluminação Natural

As esquadrias devem permitir uma entrada adequada de luz natural, contribuindo para a criação de ambientes mais saudáveis e confortáveis, além de proporcionar economia de energia. A quantidade e a qualidade da luz natural dependem do tamanho, da posição e do tipo de vidro utilizado nas esquadrias (ABNT NBR 10821, 2000).

Ventilação Natural

A ventilação natural proporcionada pelas esquadrias é essencial para o conforto térmico dos ocupantes e para a renovação do ar interno. Esquadrias bem projetadas facilitam a circulação do ar, ajudando a manter temperaturas agradáveis e a reduzir a necessidade de sistemas de climatização(NBR 15575, 2013).

Isolamento Térmico e Acústico

As esquadrias devem contribuir para o isolamento térmico da edificação, reduzindo as trocas de calor entre o interior e o exterior, o que impacta diretamente no consumo de energia para aquecimento ou resfriamento dos ambientes. Além disso, o isolamento acústico é importante para garantir o conforto dos ocupantes, especialmente em áreas urbanas com altos níveis de ruído (ABNT NBR 10821, 2000).

Estanqueidade

A capacidade de impedir a entrada de água e vento é fundamental para a durabilidade da edificação e o conforto dos usuários. Esquadrias que apresentam falhas na estanqueidade podem levar a infiltrações, deterioração dos materiais e desconforto interno (ABNT NBR 10821, 2000).

Segurança

As esquadrias devem garantir a segurança dos ocupantes, tanto em termos de resistência à intrusão quanto à proteção contra acidentes, como quedas de janelas ou quebra de vidros (ABNT NBR 10821, 2000).

Durabilidade e Manutenção

Materiais de alta qualidade e processos de fabricação adequados garantem a durabilidade das esquadrias, reduzindo a necessidade de manutenção e os custos associados. Além disso, a escolha de materiais resistentes à corrosão e ao desgaste é crucial em regiões com condições climáticas adversas.

Além disso, aspectos estéticos e funcionais devem ser harmonizados com os requisitos de desempenho, garantindo que as esquadrias contribuam para a identidade visual do edifício sem comprometer o conforto e a eficiência energética. Em síntese, as esquadrias são componentes cruciais das edificações, cujas funções vão além da simples vedação de aberturas. Elas são fundamentais para o desempenho ambiental, a segurança e o conforto dos ocupantes, devendo ser cuidadosamente projetadas e selecionadas para atender às exigências específicas de cada edificação.

O estudo das patologias das esquadrias na construção civil reveste-se de importância primordial, dada a influência que essas condições podem exercer sobre a segurança, funcionalidade e durabilidade das edificações. Defeitos ou falhas nas esquadrias podem provocar uma série de problemas, incluindo infiltrações de água, perda de eficiência térmica, e comprometimento da segurança, o que pode acarretar elevados custos de manutenção e reparo. Ademais, a detecção e correção precoces dessas patologias são essenciais para prevenir a deterioração progressiva de outros componentes da construção e assegurar a integridade estrutural a longo prazo. Portanto, uma análise detalhada das patologias das esquadrias é imprescindível para o desenvolvimento de soluções eficazes, a elevação da qualidade das construções e a promoção da longevidade das edificações.

Manifestações Patológicas em Esquadrias

De acordo com Nazário e Zancan (2011), a palavra "patologia" origina-se do grego, combinando os termos "pathos" (que significa doença) e "logia" (que se refere ao estudo ou ciência), e pode ser traduzida como "estudo da doença". No contexto da construção civil, o conceito de patologia é aplicado ao estudo dos danos que ocorrem em edificações. Assim, a patologia na construção civil envolve a identificação das causas e efeitos dos problemas presentes em uma edificação, visando o diagnóstico e a correção desses problemas.

As esquadrias, que são elementos essenciais nas edificações, exigem uma atenção especial quanto às manifestações patológicas que podem surgir ao longo de sua vida útil. Diante da crescente demanda por habitações, a indústria da construção civil deve concentrar seus esforços

na construtibilidade e na redução do tempo e do custo de execução, sem comprometer o desempenho esperado das edificações. Este aumento da eficiência dos processos produtivos é, em grande parte, impulsionado por novas tendências do mercado, como o aumento da competitividade e as exigências mais rigorosas dos consumidores. Como consequência, observa-se uma maior preocupação com a qualidade das habitações por parte dos clientes do setor (ZECHMEISTER, 2005).

Para que uma esquadria seja considerada de qualidade, ela deve atender aos requisitos especificados pela norma NBR 10821-2 (ABNT, 2017), que define os níveis de desempenho desejados, como permeabilidade ao ar, estanqueidade à água, resistência a cargas uniformemente distribuídas, facilidade de operação e manuseio, e segurança nas operações de manuseio. Esses requisitos são aplicáveis a esquadrias de qualquer material e estão diretamente relacionados à segurança e ao conforto do usuário (SCHUCH, CHRIST e EHRENBRING, 2020).

Segundo a ABCIC, a estanqueidade à água de chuva é uma das propriedades mais desafiadoras de serem plenamente atendidas em uma janela (ABCIC, 1991). Conforme definido pela norma NBR 10821-2, para que uma janela seja considerada estanque à água, ela não deve apresentar vazamentos que resultem em escorrimento de água pelas paredes ou componentes aos quais esteja fixada, quando submetida a um fluxo mínimo de água de 2 litros por minuto por bico e às pressões de ensaio correspondentes às diferentes regiões do Brasil (ABNT, 2011).

Schuch, Christ e Ehrenbrin (2020) fizeram um estudo com objetivo de identificar as principais manifestações patológicas em esquadrias de madeira, alumínio e PVC. Durante um período de dois anos, 200 esquadrias foram analisadas por meio de inspeções visuais que visavam detectar possíveis problemas. Essas inspeções, realizadas internamente, seguiram um protocolo padronizado e utilizaram um checklist para a avaliação dos itens. As esquadrias inspecionadas estavam distribuídas em várias cidades do estado do Rio Grande do Sul. As inspeções foram conduzidas pela manhã ou tarde, a fim de garantir melhor visibilidade das manifestações patológicas, e não houve remoção de componentes para análise em laboratório. As verificações limitaram-se ao que foi observado in loco, sem levar em consideração o tempo de instalação, a

orientação solar das fachadas, ou a existência de manutenções preventivas ou corretivas. Todas as edificações inspecionadas tinham menos de 10 anos.

Os resultados do estudo revelaram quatro principais manifestações patológicas em esquadrias: infiltração, desgaste de componentes, desprendimento de componentes e ação de pragas (SCHUCH, CHRIST e EHRENBRING, 2020).

As Inflitrações foram encontradas em 39% das esquadrias do tipo maximar e 43% das esquadrias de correr. As infiltrações podem ser causadas por desgaste do material, obstrução dos drenos ou instalações inadequadas. A dificuldade em determinar se o problema é causado pelo produto ou pela interface da fachada também foi destacada. Desgastes de componentes foram encontrados em todos os casos inspecionados. 100% apresentaram desgaste nos componentes, o que compromete a funcionalidade e a segurança das esquadrias. Esse desgaste indica uma falha na qualidade dos componentes utilizados, o que não atende às normas de desempenho vigentes. Alguns componentes, como trilhos, são difíceis de substituir, o que pode reduzir a vida útil da esquadria.

Desprendimento de Componentes foi identificado em 89% das esquadrias maximar e 59% das esquadrias de correr. Esse problema pode resultar em riscos à segurança dos usuários, além de comprometer o desempenho das esquadrias. O desgaste, a falta de controle na produção e o uso de materiais de baixa qualidade foram apontados como causas para esse tipo de manifestação. A Ação de Pragas foi uma manifestação patológica encontrada exclusiva para esquadrias de madeira, e foi encontrada em 39% das janelas maximar e 59% das janelas de correr. A falta de tratamento químico contínuo contribuiu para a infestação por pragas, como cupins, que danificam a madeira e facilitam a infiltração de água, acelerando o processo de apodrecimento e fragilização do material. Isso pode levar à necessidade de substituição total da esquadria, pois o reparo torna-se inviável com o tempo.

Com base nesses dados observamos que identificar e corrigir esses defeitos é essencial para garantir a funcionalidade, segurança e durabilidade das portas e janelas, além de assegurar a eficiência energética e o conforto dos ambientes internos.

10 TREINANDO UMA REDE NEURAL

Em suma, a principal razão para se adotar uma rede neural reside em sua capacidade de generalização, ou seja, na habilidade do modelo em produzir resultados satisfatórios quando aplicado a dados não utilizados durante o treinamento. No entanto, durante o processo de treinamento, podem surgir dois problemas: o *underfitting* e o *overfitting*. Para mitigar esses desafios, é possível adotar diversas estratégias. Entre as mais relevantes estão as decisões tomadas na etapa de tratamento de dados, as quais desempenham um papel crucial. Esses aspectos serão abordados com maior profundidade nas seções seguintes.

Overfitting e *Underfitting*

Overfitting ocorre quando um modelo se ajusta de maneira excessiva aos dados de treinamento, capturando tanto os padrões reais quanto os ruídos ou variações aleatórias presentes nesses dados. Como resultado, o modelo acaba aprendendo detalhes específicos que não são generalizáveis a novos dados, o que compromete seu desempenho fora do conjunto de treinamento, mas falha em generalizar bem para novos dados (ou seja, dados que não foram usados durante o treinamento). Nesse cenário, o modelo aprende não apenas os padrões relevantes, mas também ruídos ou flutuações aleatórias presentes nos dados de treinamento, que são interpretados como conceitos importantes. Como esses conceitos não são aplicáveis a novos dados, o desempenho do modelo em situações reais tende a ser comprometido. Esse problema é comum em modelos mais complexos, como redes neurais profundas, especialmente quando treinados com conjuntos de dados limitados. O *overfitting* pode ser identificado quando há alta precisão no conjunto de treinamento, mas um desempenho significativamente inferior em dados de validação ou teste, indicando uma falta de generalização (BROWLEE, 2016).

Para mitigar o *overfitting*, diversas técnicas podem ser adotadas. A regularização, por exemplo, adiciona uma penalidade à função de perda para evitar que os pesos do modelo se tornem excessivamente grande. O aumento do conjunto de dados (*data augmentation*) amplia o conjunto de treinamento através de modificações nos dados, criando mais exemplos para o modelo aprender. O *dropout*, uma técnica usada em redes neurais, desativa aleatoriamente uma

fração das unidades (neurônios) durante o treinamento, forçando o modelo a aprender representações mais robustas. O *early stopping* interrompe o treinamento quando a performance do modelo em um conjunto de validação começa a piorar, prevenindo que o modelo se ajuste demais aos dados de treinamento. Por fim, a validação cruzada (*cross-validation*) assegura que o modelo seja avaliado em múltiplos subconjuntos do conjunto de dados, promovendo uma melhor generalização e reduzindo os efeitos negativos do *overfitting*.

Underfitting ocorre quando um modelo é incapaz de capturar de forma adequada as relações subjacentes nos dados de treinamento, resultando em um desempenho insatisfatório tanto nos dados de treinamento quanto nos dados de teste. Isso significa que o modelo não possui complexidade suficiente para representar a estrutura dos dados, o que se traduz em altos erros de treinamento. Como consequência, o modelo falha em capturar padrões significativos nos dados, levando a previsões imprecisas. Brownlee (2016) destaca que o *underfitting* é um indicador de que o modelo precisa ser ajustado para aumentar sua capacidade de generalização, como o uso de uma arquitetura mais complexa ou a inclusão de mais características relevantes.

Os conceitos de *underfitting* e *overfitting* utilizando um exemplo prático de um modelo de aprendizado de máquina aplicado à classificação de gatos e cachorros em pontos verdes e vermelhos (Figura 32).

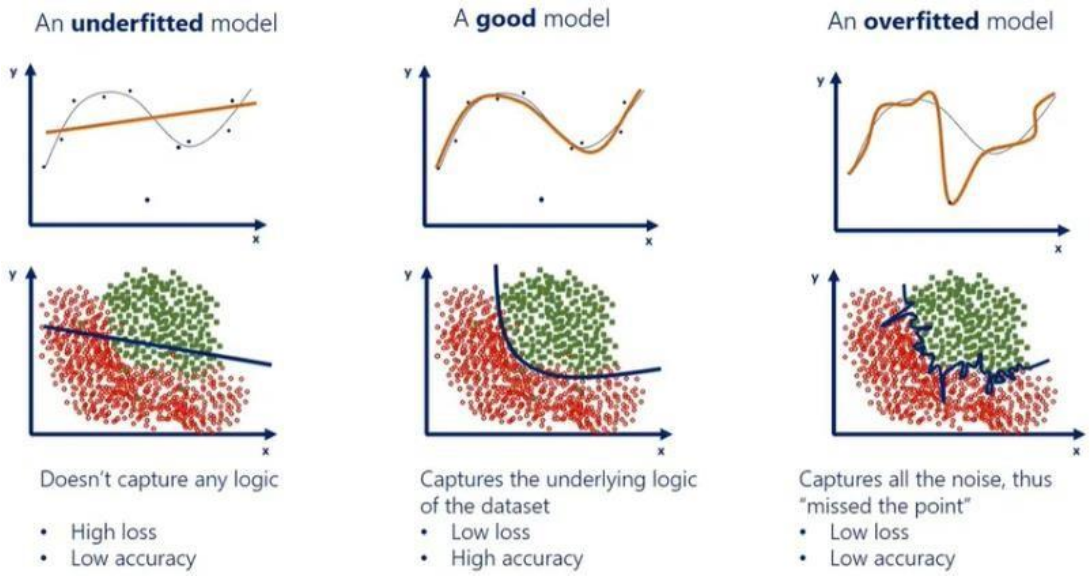


Figura 32 - Direneças entre *Overfitting* e *Underfitting*. Fonte: Brownlee, 2016.

Um modelo subajustado (*underfitted*) é simplificado demais para capturar a complexidade dos dados. No exemplo, um modelo linear simples é usado para classificar gatos e cachorros. Devido à sua simplicidade, ele não consegue capturar a diferença entre as duas categorias de forma eficaz. Como resultado, ele classifica corretamente apenas cerca de 60% das observações, indicando que o modelo está "perdendo" muita informação relevante e, portanto, não é capaz de generalizar bem. Um modelo bem ajustado para os dados de treinamento (gatos e cachorros)seria uma função que consegue identificar corretamente a maioria dos casos, com apenas algunserros. Essa função pode ser complexa o suficiente para capturar as nuances dos dados, como umafunção quadrática que consegue separar bem as duas categorias. Um modelo superajustado, poroutro lado, é excessivamente complexo e ajusta-se muito bem aos dados de treinamento, capturando até as menores variações. Isso faz com que ele classifique perfeitamente as fotos degatos e cachorros no conjunto de treinamento. No entanto, essa precisão extrema nos dados de treinamento não é uma vantagem. Quando o modelo é aplicado a novos dados (que não foram vistos durante o treinamento), seu desempenho cai drasticamente. Isso ocorre porque o modelo foi ajustado aos ruídos e particularidades dos dados de treinamento, em vez de aprender os padrões gerais que permitiriam uma boa generalização (Figura 33).

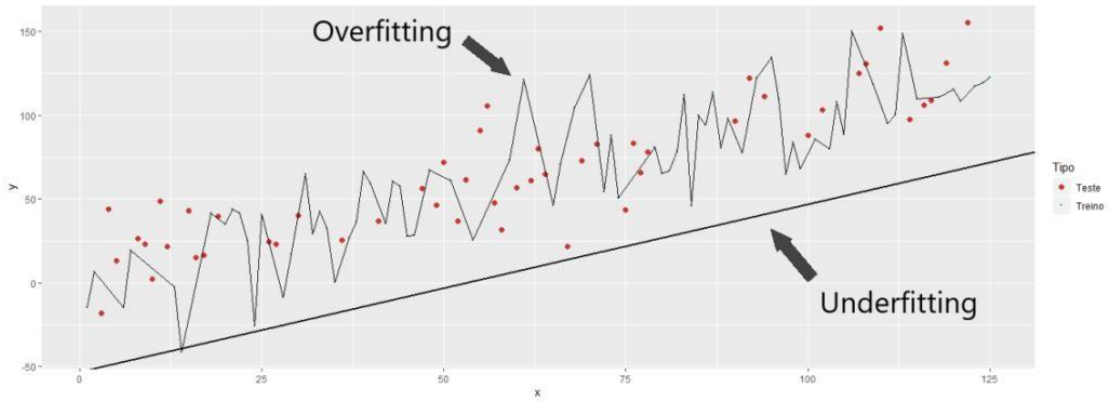


Figura 33 - *Overfitting* e *Underfitting* aplicados a um modelo genérico. Fonte: https://didatica.tech

Conclusão, Enquanto o *underfitting* resulta em um modelo que é simples demais para capturar a essência dos dados, o *overfitting* cria um modelo que é tão complexo que se ajusta perfeitamente aos dados de treinamento, mas falha ao ser exposto a novos dados. O modelo *underfitted* não consegue capturar adequadamente a relação entre os dados, enquanto um modelo

superajustado (*overfitted*) captura até as irregularidades irrelevantes, prejudicando sua capacidade de generalizar (VÍQUEZ, 2024).

Dados

Os dados constituem o elemento central em qualquer algoritmo de aprendizado de máquina. Isso ocorre porque esses algoritmos são projetados para identificar correlações nos dados. Sem dados, esses algoritmos não conseguem funcionar, pois precisam de informações para aprendere fazer previsões. Especificamente, em Redes Neurais, cada ponto de dado deve conter um conjunto de características de entrada *(features)* e um conjunto de características de rótulo *(labels)*. Para que a rede neural funcione bem, as *features* precisam ter uma relação com os *labels*.Por exemplo, se as *features* são imagens de gatos e os *labels* indicam se a imagem é de um gato ou não, deve haver uma relação clara entre a imagem e o *label* (VIEIRA,2020).

Além disso, é essencial que os dados sejam quantitativos ou categóricos, uma vez que a rede neural se baseia em um modelo matemático. Dessa forma, uma parte significativa do desenvolvimento de uma rede neural envolve a busca, produção, filtragem e análise de dados. Os dados precisam ser numéricos (quantitativos) ou organizados em categorias (categóricos), porque a rede neural usa matemática para aprender. Se os dados não forem numéricos ou categóricos, é difícil para a rede processá-los (VIEIRA, 2020).

Conjuntos de treino, validação e teste

É fundamental encontrar uma arquitetura de rede neural que não seja nem muito simples nem excessivamente complexa. Uma rede muito simples pode não capturar a complexidade dos dados, resultando em baixo desempenho (subajuste), já uma rede muito complexa pode aprender os detalhes específicos dos dados de treinamento, incluindo o ruído, levando ao *overfitting*.

A seleção da arquitetura ideal envolve testar diferentes configurações, variando o número de camadas e de perceptrons em cada camada. Este processo permite identificar a estrutura que oferece o melhor equilíbrio entre capacidade de aprendizado e generalização. O conjunto de

Treinamento é utilizado para ajustar os parâmetros da rede neural, ou seja, para treinar o modelo. A rede aprende a reconhecer padrões nos dados de treinamento ajustando os pesos das conexões entre os neurônios. O conjunto de validação não participa do treinamento. Sua função é verificar como a rede se comporta com dados novos, ajudando a avaliar a capacidade de generalização do modelo. A generalização é a capacidade do modelo de performar bem em dados que não foram vistos durante o treinamento.Serve para avaliar a performance da rede em dados não vistos. Após a seleção final da arquitetura da rede neural, passamos para o conjunto Teste. Este conjunto é criado após selecionar a melhor arquitetura com base no conjunto de validação, com a finalidade de avaliar a performance final do modelo. Isso assegura que o modelo escolhido não esteja apenas ajustado ao conjunto de validação, mas também funcione bem em dados completamente novos.

Na prática, diferentes arquiteturas de redes neurais são treinadas utilizando o conjunto de dados de treinamento até que se obtenham baixos erros de treinamento. Posteriormente, a performance dessas redes é avaliada no conjunto de validação, com o objetivo de selecionar a arquitetura mais eficaz. A divisão dos dados em três conjuntos treinamento, validação e teste é uma abordagem amplamente adotada no desenvolvimento de modelos de *machine learning*, pois ajuda a prevenir o *overfitting* (VIEIRA, 2020).

Segundo Goodfellow, Bengio e Courville (2016) a escolha das proporções para dividir os dados em conjuntos de treinamento, validação e teste depende de vários fatores, como o tamanho total do conjunto de dados, a complexidade do modelo, a disponibilidade de dados rotulados e o objetivo do modelo.

- Tamanho do conjunto de dados: quando se dispõe de uma quantidade substancial de dados (milhares ou milhões de amostras), é possível destinar uma porção menor para validação e teste, mantendo a maior parte para o treinamento. Uma divisão comum seria:

  Treinamento: 70-80%,
  Validação: 10-15%,
  Teste: 10-15%

- Complexidade do modelo: Se o modelo é simples ou complexo. Se o modelo que treinado não é muito complexo (por exemplo, uma regressão linear), pode-se optar por uma divisão que favoreça mais o conjunto de treinamento, pois esses modelos tendem a não superestimar as previsões. Para modelos mais complexos, como redes neurais profundas, é importante ter um conjunto de validação robusto para monitorar o *overfitting*. Neste caso, pode-se destinar uma porção maior para validação, como:

  Treinamento: 60-70%
  Validação: 15-20%
  Teste: 15-20%

- Disponibilidade de Dados Rotulados: Se possuímos muitos dados rotulados, pode-se manter uma divisão mais tradicional, como 70-15-15 ou 80-10-10, dependendo da necessidade. Quando há poucos dados rotulados, é recomendável utilizar técnicas outras técnica mais complexas, especialmente em tarefas como visão computacional ou processamento de linguagem natural.

  Treinamento: 70%
  Validação: 15%
  Teste: 15%

- Objetivo do Modelo: No início do desenvolvimento, pode-se optar por uma maior quantidade de dados para treinamento para acelerar o processo de desenvolvimento do modelo. Porém, é crucial garantir que o conjunto de validação seja suficiente para monitorar o desempenho do modelo. Para a versão final do modelo, é importante uma avaliação rigorosa utilizando um conjunto de teste que nunca foi visto durante o treinamento ou validação.

Com base no que foi apresentado, entende-se que nno existe uma única proporção correta para todos os cenários, a escolha deve ser orientada por diversos fatores, como os representados a cima. Uma abordagem prudente é começar com uma divisão tradicional (por exemplo, 70-15-15) e, se necessário, ajustar as proporções com base no desempenho observado durante o desenvolvimento e na avaliação do modelo (GOODFELLOW, BENGIO E COURVILLE, 2016).

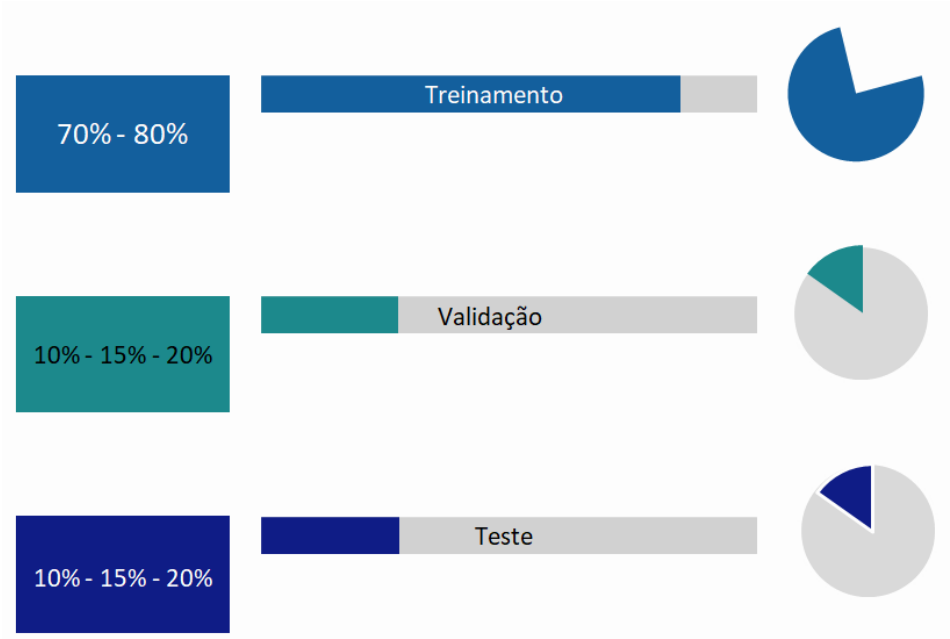


Figura 34 - Treinamento, validação e teste em porcentagens. Fonte: O Autor, 2024.

Batch Size e Epochs

Dois parâmetros importantes a se considerar ao treinar um modelo de *Machine Learning* são o *"Batch Size"* e o número de *Epochs*. O *Batch Size* refere-se ao número de exemplos de treinamento em cada lote de dados, enquanto o número de *"Epochs"* ou épocas, indica quantas vezes o modelo passará por todo o conjunto de dados de treinamento durante o treinamento. A escolha adequada desses parâmetros pode afetar significativamente o desempenho do modelo.

O *Batch Size* refere-se ao número de amostras de dados que o modelo processa antes de atualizar os parâmetros internos (pesos e vieses) durante o treinamento. Em outras palavras, em vez de atualizar os parâmetros após cada exemplo de treinamento (o que seria chamado de *Stochastic Gradient Descent*), o modelo acumula o gradiente de erro de um lote de exemplos e, em seguida, faz uma única atualização com base na média desses gradientes. *Batch Size* pequeno resulta em atualizações mais frequentes dos parâmetros, o que pode levar a uma convergência mais rápida, mas menos estável, pois o modelo pode ser mais influenciado por ruídos nos dados. *Batch Size* grande resulta em atualizações menos frequentes, mas mais estáveis, podendo levar a uma melhor convergência, mas com um custo computacional maior e um tempo de treinamento mais

longo. No entanto, pode ser necessário mais memória (RAM) para processar lotes maiores (GERON, 2017).

O número de *Training Epochs* refere-se ao número de vezes que o modelo passará por todo o conjunto de dados de treinamento. Cada epoch representa uma iteração completa sobre todos os exemplos de treinamento disponíveis. Durante cada *epoch*, o modelo ajusta seus parâmetros com base nos erros observados. Em poucas epochs o modelo pode não ter tempo suficiente para aprender padrões importantes dos dados, resultando em *underfitting*, onde o modelo não captura bem a complexidade dos dados. Em muitas *epochs,* o modelo pode aprender os padrões dos dados de treinamento muito bem, a ponto de se ajustar excessivamente a eles, o que pode resultar em *overfitting*, onde o modelo não generaliza bem para novos dados (GERON, 2017).

Um framework é uma estrutura de código que oferece uma biblioteca de funções prontas para serem integradas e utilizadas em projetos. Alguns frameworks são desenvolvidos especificamente para aplicações em inteligência artificial e disponibilizam uma variedade de recursos e algoritmos. Esses algoritmos, presentes nos frameworks, já estão otimizados e seus códigos são implementados de forma a permitir que sejam importados e executados diretamente, facilitando o processo de desenvolvimento. Nesse sentido, dois dos principais frameworks utilizados em *Deep Learning* são o Keras e o Tensorflow.

TensorFlow

O TensorFlow é uma biblioteca de código aberto desenvolvida pelo Google para a construção e treinamento de modelos de aprendizado de máquina e redes neurais. Criado pela equipe do Google Brain, TensorFlow é um framework *open-soure* (código aberto) desenvolvido para Python e JavaScript, que auxilia no desenvolvimento de soluções. Pode ser executado sobre diversas plataformas e arquiteturas, incluido CPUs, GPUs e as recentes TPUs (Tensor Processing Unit). Atualmente, é um dos principais frameworks do mercado para criação de redes neurais *Deep Learning*. Com ele, é possível agilizar e facilitar o processo de obtenção de dados, treinar modelos, realizar predictions e refinar resultados futuros (TENSORFLOW, 2019).

O TensorFlow emprega uma API rica em Python, facilitando o desenvolvimento para o programador final. A execução do TensorFlow é baseada em uma aplicação de alta performance escrita em C/C++. Os engenheiros da Google combinaram a simplicidade do Python com o desempenho superior do C/C++. A arquitetura do TensorFlow é composta por três partes principais (ARAUJO et al. 2017).

- Pré-processamento dos dados;
- Construção dos modelos;
- Treinamento e estimativas do modelo criado.

O TensorFlow utiliza um framework especializado para gerenciar gráficos computacionais, que são representações de conjuntos de operações realizadas em sequência. Esses gráficos organizam e descrevem todos os cálculos envolvidos durante o treinamento. O uso de gráficos oferece várias vantagens, conforme destacado por Yegulalp (2019). Primeiramente, eles são projetados para execução em diversas arquiteturas, incluindo CPUs, GPUs e sistemas operacionais móveis. Além disso, os gráficos são portáteis, permitindo que cálculos sejam salvos e reutilizados posteriormente. Por fim, todos os cálculos são realizados conectando tensores, o que contribui para a eficiência das operações.

O principal benefício que o TensorFlow oferece no desenvolvimento de sistemas inteligentes é a abstração. Em vez de se preocupar com os detalhes fundamentais da implementação de algoritmos ou com a integração das saídas e entradas das funções, o desenvolvedor pode focar na lógica geral da aplicação ou no problema a ser resolvido (TENSORFLOW, 2019). O TensorFlow fornece uma variedade de ferramentas para depuração e introspecção, permitindo a avaliação e modificação de cada operação nos gráficos de forma isolada e explícita, ao invés de construir e analisar o gráfico inteiro de uma vez. Além das facilidades de desenvolvimento, o TensorFlow se destaca como uma ferramenta confiável devido ao seu status como um dos principais projetos de Inteligência Artificial da Google. A Google não apenas promoveu o avanço do framework, mas também incentivou seu uso ao oferecer servidores dedicados com GPUs e as potentes TPUs, que proporcionam desempenho acelerado na nuvem do Google. Isso possibilita o desenvolvimento de soluções escaláveis e de alta disponibilidade na web (FALCÃO et al. 2019).

No trabalho desenvolvido por Falcão et al. (2019) Foi feita uma demonstração da implementação de uma Rede Neural de *Deep Learning*, focada no reconhecimento de peças de roupas. Os resultados mostraram um desempenho muito bom, com uma precisão geral superior a 95%, e em alguns casos, alcançando 100% de precisão. O objetivo principal foi mostrar como a implementação é realizada e como o framework TensorFlow opera e processa os modelos inteligentes através da API do Keras. Observou-se que a linguagem Python se destaca na implementação de algoritmos de inteligência artificial, pois oferece um extenso conjunto de bibliotecas nativas que facilitam o desenvolvimento de aplicações de forma rápida e eficiente.

O Keras

Keras é uma plataforma de alto nível para aprendizado profundo em Python, desenvolvida por François Chollet, que opera sobre o TensorFlow. Sua principal vantagem reside no tempo economizado ao utilizar APIs simples e intuitivas, mas ao mesmo tempo altamente eficientes, permitindo a prototipagem ágil de conceitos. O Keras facilita a aplicação dos princípios do TensorFlow de forma mais direta e acessível, evitando a necessidade de escrever código repetitivo e desnecessário na criação de modelos de aprendizado profundo. Além de fornecer fácil acesso a bibliotecas especializadas, Keras mantém a integração com os benefícios oferecidos pelo TensorFlow. Sua instalação é simples e pode ser realizada através do comando pip ou conda, sendo necessário que o TensorFlow esteja previamente instalado, uma vez que ele serve como backend (camada principal de um software, responsável por processar dados e executar tarefas) para a criação de modelos em Keras (SARKAR et al., 2018).

A API tf.keras é a interface de alto nível do TensorFlow destinada à criação e ao treinamento de modelos de aprendizado profundo. Ela foi projetada para atender a três objetivos principais: prototipagem rápida, pesquisa de ponta e produção. Suas características centrais oferecem vantagens significativas para desenvolvedores e pesquisadores de *Deep Learning*. Primeiramente, tf.keras possui facilidade em seu uso. A biblioteca possui uma interface simples, intuitiva e consistente, o que a torna ideal para casos de uso comuns no desenvolvimento de redes neurais (KERAS, 2018).

Outro ponto forte é a capacidade de construir modelos modulares e compostos. Em Keras, os modelos são criados por meio da conexão de componentes configuráveis, como camadas, otimizadores e funções de perda, com um mínimo de restrições. Isso oferece grande flexibilidade, permitindo que os usuários ajustem e adaptem seus modelos conforme necessário para tarefas específicas (KERAS, 2018).

Por fim, tf.keras é fácil de estender, o que a torna uma ferramenta poderosa para pesquisa. Desenvolvedores podem criar elementos personalizados que expressem novas ideias, como camadas, métricas e funções de perda inéditas. Isso permite a criação de modelos de aprendizado profundo de última geração, viabilizando a implementação de avanços e inovações no campo do *Deep Learning* (KERAS, 2018).

Em resumo o Keras é uma camada de abstração simples para facilitar a criação de redes neurais, enquanto TensorFlow é uma plataforma mais ampla e poderosa que permite controle detalhado sobre as operações de aprendizado profundo. Enquanto o Keras foca na facilidade de uso e prototipagem rápida, o TensorFlow é mais voltado para produção e customização avançada. Keras funciona como uma API de alto nível dentro do TensorFlow, que serve como o ‘’motor’’ para a execução dos modelos.

python Copiar código

```python
from keras.models import Sequential
from keras.layers import Dense

# Inicializando o modelo sequencial
model = Sequential()

# Adicionando camadas
model.add(Dense(units=64, activation='relu', input_dim=100))
model.add(Dense(units=10, activation='softmax'))

# Compilando o modelo
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Treinando o modelo com dados
model.fit(x_train, y_train, epochs=10, batch_size=32)
```

Figura 35 - Codigo do Keras para modelagem. Fonte: edu.taugc.com

Ribeiro e Guimarães (2018) realizaram um estudo utilizando o Keras para criar um modelo capaz de classificar se o motorista é autorizado ou não a dirigir o veículo, em sua base de dados, foram utilizados ao todo 4297 informações, que foram divididos em dois, 90% para treinamento (3437) e 10% para teste (860). A aplicação foi realizada na linguagem python, A utilização do TensorFlow e Keras mostrou-se altamente eficiente, uma vez que, mesmo com um tempo de treinamento reduzido e uma base de dados relativamente pequena, a rede alcançou uma precisão de aproximadamente 92%. Esse resultado demonstra uma elevada confiabilidade para a tomada de decisões. Além disso, esse desenvolvimento confirma que é possível construir uma rede neural confiável sem exigir um conhecimento profundo na área, especialmente com o auxílio de bibliotecas pré-configuradas que simplificam o processo. Isso indica um potencial futuro em que qualquer pessoa, inclusive iniciantes no campo do desenvolvimento de softwares inteligentes, possa criar sua própria rede neural de forma acessível.

11 METODOLOGIA

O presente estudo tem como objetivo demonstrar uma aplicação simplificada de redes neurais profundas (*Deep Learning*) para a detecção de esquadrias em prédios públicos, com ênfase na classificação de segmentos da imagem, visando à geração de uma região de interesse para a detecção. Após o treinamento da rede neural, novas imagens com diferentes níveis de qualidade, variações de sombra e resoluções distintas serão testadas para avaliar a acurácia do modelo. Essa etapa é essencial para verificar a capacidade da rede de generalizar seu aprendizado e manter um desempenho robusto em condições variadas ao expor o modelo a cenários que diferem daqueles utilizados durante o treinamento.

O processo para treinar uma rede neural para identificar esquadrias em imagens de edificações envolve a análise das imagens para localizar e delimitar a área onde as esquadrias estão presentes. Essa abordagem se baseará na arquitetura proposta por Cha et al. (2017) e será implementada no ambiente de desenvolvimento Jupyter Notebook, utilizando a biblioteca Keras e a linguagem de programação Python. O treinamento do modelo será realizado com umconjunto de dados do próprio autor, visando a possibilidade de construção de um modelo com alta precisão na detecção das esquadrias.

É importante destacar que o objetivo deste estudo foi exclusivamente aplicar a rede neural previamente desenvolvida e implementada para a detecção de esquadrias. Não houve, por parte do autor, a criação de novas rotinas de programação ou novos códigos, limitando-se a utilização do modelo já existente.

Nos próximos tópicos, serão apresentados os fundamentos por trás de cada escolha realizada, incluindo a justificativa para a arquitetura proposta, as etapas executadas no estudo e as decisões tomadas ao longo do processo.

Banco de dados

Dung e Anh (2019) propuseram a utilização de uma rede totalmente convolucional profunda (FCN) para a detecção de fissuras em imagens de concreto. A FCN é uma arquitetura de redes neurais que utiliza apenas camadas convolucionais, preservando a estrutura espacial das imagens, o que é essencial para tarefas de segmentação e detecção de objetos.

Özgenel (2018) construiu um conjunto de dados *(dataset)* público composto por 40.000 imagens de fissuras de concreto, cada uma com resolução de 227x227 pixels. Esse *dataset*, utilizado pelos autores para o treinamento do modelo, foi dividido em duas categorias: imagens com fissuras, chamadas de “Positive” e imagens sem fissuras chamadas de “Negative” nas superfícies de concreto. As imagens foram distribuídas em três partes: 80% para treinamento, 10% para validação e 10% para teste. No experimento, foram utilizadas três arquiteturas de redes pré-treinadas: VGG16, ResNet e InceptionV3. A VGG16, uma rede com 16 camadas conhecida pela simplicidade e desempenho robusto, se destacou entre as demais, obtendo uma precisão e um F1-Score de 89,3% no conjunto de teste. Esses resultados indicam que a VGG16, apesar de ser uma arquitetura mais simples em comparação com as outras testadas, conseguiu capturar de forma eficaz as características necessárias para a detecção de fissuras em concreto, mostrando-se a mais adequada para essa tarefa específica. O *dataset* de Özgenel (2018) está disponível públicamente e pode ser baixado por qualquer pessoa.

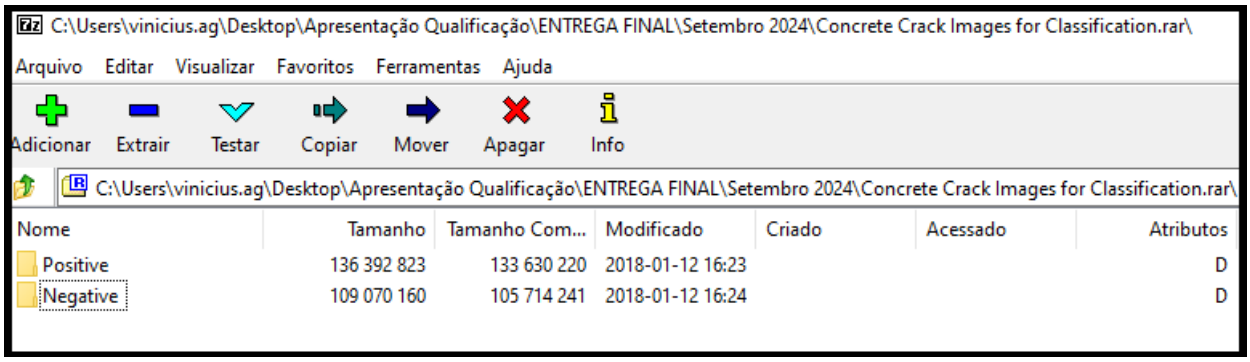


Figura 36 - Pastas de segmentação das imagens com ou sem esquadrias. Fonte: O Autor, 2024.

Outras contribuições obtiveram êxito com a utilização do banco de dados de 40.000 imagens de Özgenel, Rajadurai e Kang (2021) explorou o uso de algoritmos baseados em visão computacional para detectar e classificar rachaduras em superfícies de concreto, utilizando redes neurais convolucionais profundas. Especificamente, foi utilizado o modelo AlexNet, que foi ajustado através de técnicas de aprendizado de transferência para melhorar a precisão na detecção de rachaduras. O conjunto de dados utilizado tambem incluía imagens de duas categorias: com rachaduras e sem rachaduras como no anterior. O processo de aprendizado de transferência envolveu a adaptação dos pesos do modelo AlexNet, a modificação da camada de classificação para distinguir entre essas duas classes, e a ampliação do conjunto de dados através da rotação aleatória das imagens. Durante o treinamento, o modelo alcançou uma precisão de 99,9% e uma perda de apenas 0,1% após 6 épocas com uma taxa de aprendizagem de 0,0001. Nos testes de validação e avaliação, o modelo AlexNet conseguiu prever corretamente 1998 de 2000 imagens e 3998 de 4000 imagens, resultando em uma precisão preditiva de 99,9%. Além disso, o modelo apresentou valores de precisão e acurácia de 0,99, o que demonstra um alto nível de desempenho na detecção de rachaduras.

O estudo conduzido por Flah et al. (2020) apresentou um modelo automatizado de inspeção que utiliza técnicas de processamento de imagem e aprendizado profundo para identificar defeitos em áreas de concreto que são geralmente de difícil acesso. As imagens usadas no treinamento foram escolhidas manualmente e redimensionadas para uma resolução de 227 x 227 pixels. O modelo foi treinado com um total de 10.000 imagens, sendo metade delas de superfícies com rachaduras e a outra metade de superfícies sem rachaduras. Para treinar e avaliar o modelo, os

dados foram divididos da seguinte maneira: 60% para treinamento, 20% para validação e 20% para teste. Durante a fase de teste, o modelo alcançou uma precisão de 98,25% e uma perda de 0,057, indicando que o modelo não sofreu com problemas de *overfitting*. Em síntese, o modelo proposto mostra-se uma ferramenta promissora para a inspeção automatizada de estruturas de concreto.

O estudo realizado por Kim et al. (2020) propõe uma técnica de análise de imagens utilizando *Deep Learning*, desenvolvida para detectar rachaduras e analisar suas características, como comprimento e largura, em instalações de pequena escala. A pesquisa foi dividida em três etapas principais. Nas duas primeiras, o foco foi a detecção de rachaduras por meio de classificação e segmentação de imagens. Para essa detecção, foram utilizadas 40.000 imagens, divididas em duas categorias: 20.000 imagens com rachaduras e 20.000 sem rachaduras. Após a detecção baseada em aprendizado profundo, a terceira etapa envolveu o uso de algoritmos de desbaste e rastreamento para medir o comprimento e a largura das rachaduras nas imagens. Os resultados demonstraram uma precisão de 99% na detecção das rachaduras. Apesar do alto desempenho mostrado no estudo, os autores sugerem que o método precisa ser validado com um conjunto de dados maior para garantir sua aplicabilidade prática. O banco de dados "*open source"* ou código aberto, de forma resumida, refere-se a um software que é disponibilizado gratuitamente para qualquer pessoa acessar, copiar, modificar e redistribuir.

O estudo realizado por Arafin et al. (2022) concentrou-se na comparação do desempenho de diferentes modelos para a detecção de rachaduras e descamação em superfícies de concreto. Foram utilizadas redes neurais convolucionais (CNNs) pré-treinadas para classificar rachaduras nas imagens. O conjunto de dados incluía 4.087 imagens de rachaduras e 1.100 imagens de descamação, todas com resolução de 227 x 227 pixels. Essas imagens foram divididas aleatoriamente em três conjuntos: 70% para treinamento, 20% para validação e 10% para teste. Entre os modelos avaliados, a arquitetura InceptionV3 apresentou o melhor desempenho, alcançando uma acurácia de 91%, precisão de 82%.

Não foi identificado na literatura existente um banco de dados ''*open-source*'' que pudesse ser utilizado para treinar uma rede neural voltada à detecção de esquadrias. Diante da falta de um

dataset público específico para abordar este problema, tornou-se evidente a necessidade de criar um conjunto de dados próprio. Este esforço foi crucial para garantir que as técnicas de *Deep Learning* pudessem ser aplicadas de forma eficaz, visto que, como explicado ao decorrer do estudo, é amplamente reconhecido que a obtenção de resultados satisfatórios depende de um volume significativo de imagens para o treinamento adequado dos modelos.

Neste estudo, o banco de dados foi desenvolvido em colaboração com colegas de profissão, que também o utilizarão em outras aplicações relacionadas. Inicialmente, esse conjunto de dados foi mantido como particular, restrito ao uso dos colaboradores envolvidos. No entanto, reconhecendo o valor que esse recurso pode trazer para a comunidade acadêmica e científica, há planos de disponibilizá-lo publicamente no futuro. Essa iniciativa visa permitir que outros pesquisadores tenham acesso ao banco de dados, facilitando a realização de novos estudos e contribuindo para o avanço das pesquisas na área de detecção de esquadrias e outras aplicações.

Mesmo com a limitação no número de dados disponíveis, o objetivo principal foi criar uma base suficiente que permitisse iniciar o processo de aprendizado e validação dos algoritmos de *Deep Learning*, estabelecendo, assim, uma base que pode ser expandida e aprimorada futuramente. Em vez de utilizar bancos de dados genéricos, compostos por milhares de imagens variadas, este trabalho propõe uma abordagem diferente, no caso presente, foram capturadas e compiladas 19.200 imagens durante inspeções realizadas com câmeras digitais, resultando em um conjunto de dados especializado, adequado para os objetivos do estudo.

O banco de dados desenvolvido foi elaborado ao longo de aproximadamente dois anos, abrangendo os 26 estados da federação e o Distrito Federal. Este banco de dados inclui uma ampla gama de edificações com finalidades variadas, como galpões, centros comerciais, escritórios, salas comerciais e lojas, entre outros. No total, o banco de dados compõe cerca de 1.840 edificações, nas quais foram identificadas esquadrias de diversos tipos e modelos. Entre as esquadrias catalogadas estão janelas de correr, guilhotinas, venezianas, maxiar, pivotantes e fixas, entre outras. A diversidade de modelos e designs das esquadrias enriquece significativamente a base de dados, proporcionando um valor considerável ao trabalho desenvolvido.

Após a organização dos dados em planilhas do Excel, foi possível determinar a distribuição do número de imagens coletadas em cada estado do país, bem como a proporção percentual de cada categoria de edificação. Esses dados foram então apresentados de forma visual, utilizando mapas e gráficos em formato de rosca.

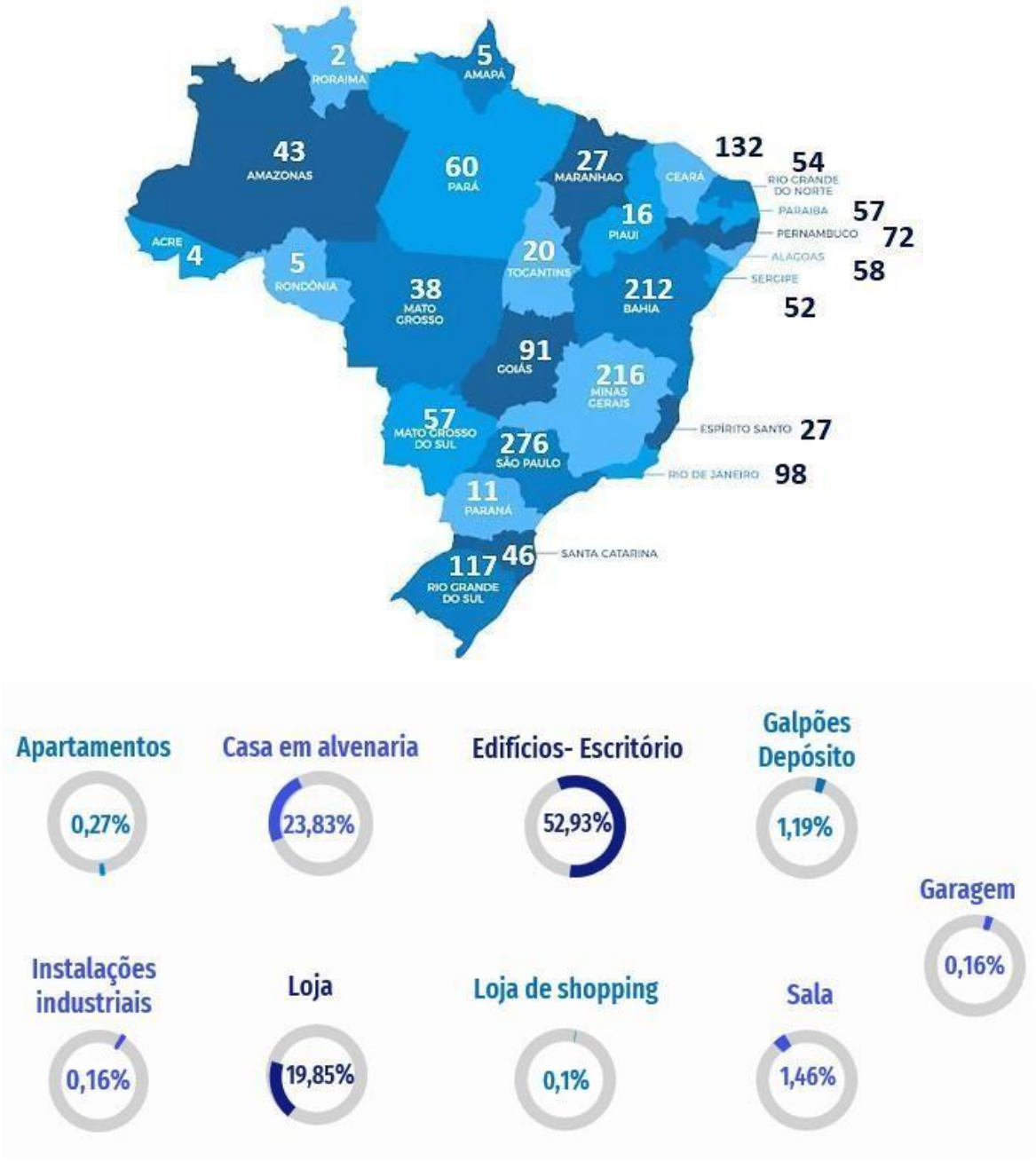


Figura 37 - Distribuição do número de imagens coletadas em cada estado do país e categoria de cada tipo de edificação. Fonte: O Autor, 2023

Um aspecto interessante é que cada tipo de edificação geralmente adota um estilo padrão de esquadrias. Por essa razão, diferentes tipos de edificações, com distintas finalidades, foram agrupados. Formou-se com a junção de todas as imagens a criação de um banco de dados especializado em esquadrias. As técnicas de *Deep Learning* e fotogrametria aplicadas às imagens obtidas possibilitaram a detecção de janelas com alta precisão, além de fornecer dados para a quantificação detalhada. Exemplos representativos das imagens presentes no conjunto de dados gerado estão ilustrados na Figura 38.

Figura 38 - Exemplo de imagens segmentadas do *dataset* prórpio. O Autor, 2023.

Arquitetura Proposta

Embora as arquiteturas: Experimental, Inception V3, Experimental, MobileNet V2, VGG16 apresentadas no tópico 11.1 tenham apresentados excelentes acurácias, a abordagem proposta nesse estudo se baseia em uma arquitetura de rede neural inspirada no trabalho de Cha et al. (2017). Nesse estudo, os autores propuseram uma arquitetura relativamente simples, mas eficaz, para a detecção de fissuras em concreto, alcançando bons resultados. No entanto, inspirados nas arquiteturas citadas, uma modificação crucial foi introduzida na arquitetura original: a alteração

das dimensões da camada de entrada (input layer), passando de 256×256 pixels, como usado por Cha, para 227×227 pixels. Essa mudança foi feita para ajustar a rede neural ao conjunto de dados específico utilizado neste projeto, eliminando assim a necessidade da etapa de redimensionamento das imagens durante o processo de treinamento. Esta arquitetura se destaca pela sua simplicidade e eficiência na classificação de imagens pois utiliza-se de poucas camadas: camadas convolucionais, camadas de pooling, camadas completamente conectadas e camadas de saída. A última camada da rede é uma camada de saída com um número de neurônios correspondente ao número de classes de saída (neste caso, duas: com esquadria e sem esquadria).

A escolha da resolução de 227×227 pixels não foi feita de maneira arbitrária, mas sim fundamentada em estudos relevantes de Dung (2018), Kim et al. (2020), Arafin et al. (2022), Flah et al. (2020), e Rajadurai e Kang (2021). Estes trabalhos optaram pela referida resolução e relataram resultados satisfatórios em suas respectivas pesquisas. A escolha da resolução de entrada para modelos de redes neurais em tarefas de visão computacional é um aspecto essencial que pode influenciar tanto o desempenho quanto a eficiência do modelo. A resolução de227x227, frequentemente utilizada em redes como a AlexNet, é uma das opções disponíveis, enquanto a resolução de 256x256 também é amplamente adotada. A decisão de utilizar a resolução de 227x227, em vez de 256x256, pode ser justificada por uma série de fatores.

Primeiramente, é importante considerar a compatibilidade com arquiteturas existentes. Algumas arquiteturas, como a AlexNet, foram originalmente projetadas para operar com imagens de 227x227 pixels. Dessa forma, ao escolher essa resolução, garante-se que os modelos pré-treinados possam ser reaproveitados ou transferidos de maneira eficiente e direta.

Além disso, a resolução de 227x227 oferece vantagens em termos de eficiência computacional. Por possuírem menos pixels em comparação com imagens de 256x256, as imagens de 227x227 reduzem a carga computacional durante o treinamento e a inferência. Essa redução pode ser particularmente benéfica ao trabalhar com grandes volumes de dados ou em ambientes com recursos computacionais limitados.

Outro ponto a ser considerado é a memória e a velocidade. Resoluções menores, como 227x227, demandam menos memória GPU e resultam em tempos de processamento mais rápidos. Essa característica pode ser decisiva em aplicações que exigem respostas em tempo real ou em dispositivos com capacidade limitada de armazenamento e processamento.

Em termos de generalização, a resolução menor pode contribuir para a diminuição de *overfitting*. Ao trabalhar com imagens de menor resolução, a rede neural é forçada a focar em características mais globais, em vez de se prender a detalhes específicos que podem não se generalizar bem em outros contextos.

Por fim, a resolução de 227x227 tem um histórico de desempenho comprovado, sendo amplamente utilizada em diversos estudos já mencionados, como o de Vieira (2020). Isso proporciona uma base robusta e bem documentada para a comparação de resultados.

Por outro lado, é importante reconhecer que a resolução de 256x256 captura mais detalhes, o que pode ser vantajoso em cenários onde a definição precisa das características visuais é crucial para a tarefa em questão. Ou sejac, a escolha entre 227x227 e 256x256 deve ser feita considerando um equilíbrio entre a precisão desejada, a capacidade computacional disponível e as exigências específicas da tarefa.

Com base nesses princípios, o Prof. Dr. Lenildo Santos da Silva, engenheiro civil que, em colaboração com Vieira (2020), conduziu um estudo sobre a aplicação de redes neurais na detecção de fissuras em concreto, empregou sua expertise para desenvolver um aplicativo que facilita a divisão das imagens de acordo com a resolução escolhida (227x227 ou 256x256) Chamado “GeraDBDeepLearning”. O trabalho será focado na implementação do código elaborado pelo Professor Dr Lenildo Santos da Silva.

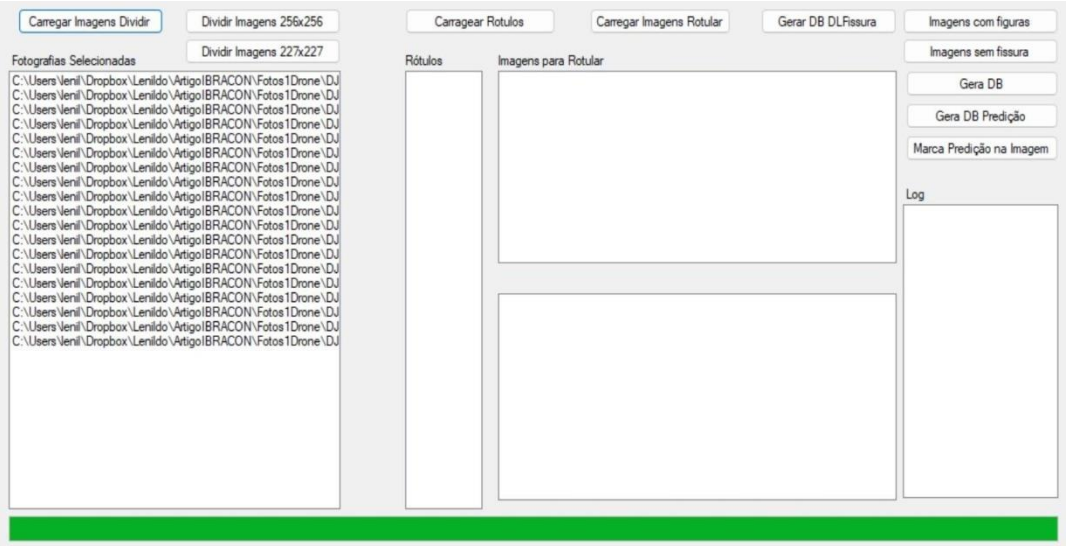


Figura 39 - Software desenvolvildo para segmentação das imagens em resolução 227x277.

Fonte: Vieira, 2020

Após fragmentar as imagens originais em dimensões menores, conforme a resolução selecionada (227x227), procede-se a uma fase de classificação manual, na qual as imagens com presença de janelas são categorizadas em uma pasta distinta, enquanto as imagens desprovidas destas esquadrias são alocadas em outra pasta. O diagrama subsequente ilustra o fluxo de trabalho adotado na metodologia.

Ferramentas

A rede neural convolucional será desenvolvida e treinada utilizando o Keras, uma biblioteca de *Deep Learning* para Python que simplifica o processo de implementação de redes neurais. O Keras, por sua vez, opera sobre o TensorFlow, que executa as operações de *Machine Learning* necessárias. Isso permite o foco nos principais aspectos da rede, como a definição da sua arquitetura, a preparação dos dados de entrada e a escolha dos parâmetros de aprendizado, como a taxa de aprendizado e o número de épocas. Todo o desenvolvimento será feito no ambiente Jupyter Notebook, que oferece uma interface interativa para execução e visualização do código e resultados (Figura 40).

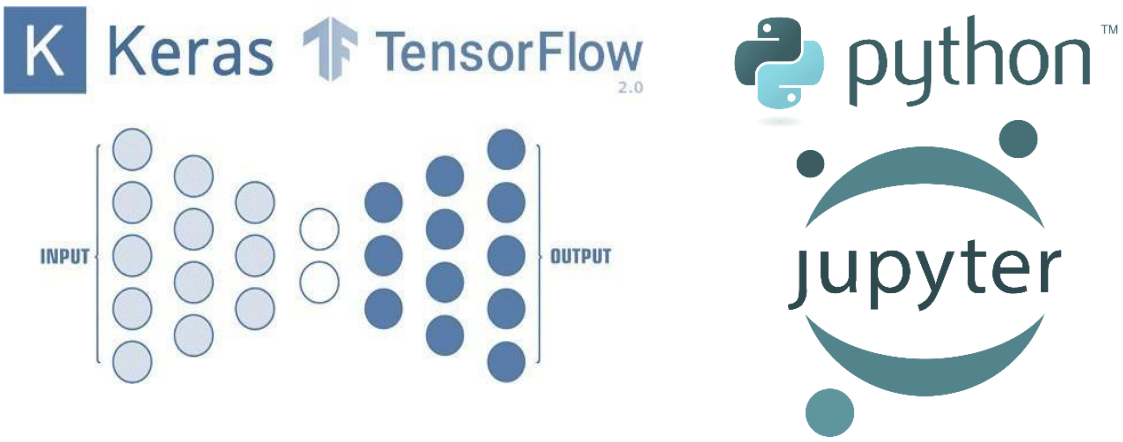


Figura 40 – Logomarca das plataformas utilizadas. Fonte: Keras, 2024

Agora, no contexto da programação em Python, o *dataset* é realizado e a divisão das imagens em conjuntos de treinamento, validação e teste, através de um scripit que distribuiu as imagens aleatórias nos diretórios seguindo uma proporção 70% para treinamento, 15% para validação e 15% para teste. respectivamente. É importante ressaltar que cada um desses conjuntos possuem a mesma quantidade de imagens, tanto para as que possuem janelas “Positive” quanto as desprovidas das mesmas “Negative”.

Optar pela divisão 70% para treinamento, 15% para validação e 15% para teste no *Machine Learning* oferece um equilíbrio estratégico para obter um modelo robusto e confiável. Esta proporção foi estabelecida não é uma regra fixa, mas sim uma prática comum em *Machine Learning* e análise de dados e é frequentemente usada porque busca um equilíbrio entre a quantidade de imagens usada para treinar o modelo, validar seu desempenho e testar sua generalização.

Com 70% dos dados destinados ao treinamento é garantido que o modelo tenha acesso a uma quantidade substancial de dados, o que é fundamental para aprender os padrões corretamente. Isso ajuda a evitar o subajuste *underfitting*, onde o modelo pode não ser capaz de captar relações complexas nos dados. Ao fornecer uma base robusta para o aprendizado, o modelo consegue capturar as características gerais dos dados e desenvolver uma capacidade de generalização.

Os 15% dedicados ao conjunto de validação permitem um controle crucial do processo de ajuste. Eles servem como um espelho de "como o modelo vai performar" em dados não vistos durante o

treinamento. Ao monitorar o desempenho no conjunto de validação, você pode evitar o *overfitting*, ajustar hiperparâmetros, realizar paradas antecipadas ou modificar a arquitetura. Isso melhora significativamente a generalização do modelo.

Com 15% dos dados reservados exclusivamente para teste, obtemos uma base sólida para avaliar o desempenho do modelo de forma imparcial. Este conjunto não participa de nenhum processo de ajuste, o que garante que a performance observada nele represente de maneira fiel a capacidade do modelo em dados completamente novos. Uma proporção menor, como 10% ou 5%, pode resultar em uma avaliação menos confiável, enquanto proporções maiores poderiam sacrificar a capacidade de ajuste dos hiperparâmetros ou o próprio treinamento. Esta proporção foi utilizada por Vieira (2020), é comumente usada em *dataset*s maiores (mais de 10 mil amostras), e está próxima da proporção usada por Cha et al. (2017). Exemplo da aplicação dessa estrutura foi realizado por Ozgenel(2018) que consiste e 40 mil imagens com resolução 227x227 pixels (Figura 41).

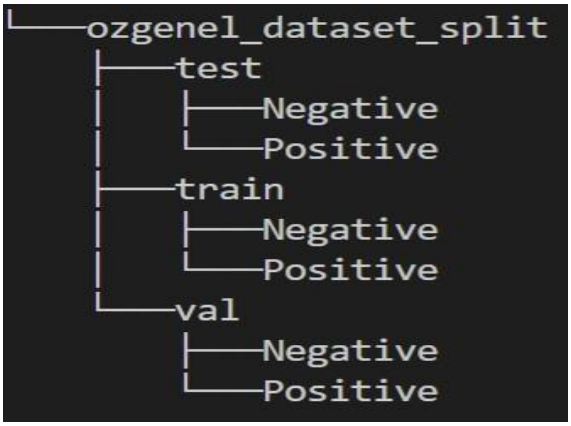


Figura 41- Segmentação dividida em Positiva e Negativa na etapas de treinamento, validação e teste segundo Ozgenel. Fonte: Ozgenel, 2018.

Divisões com maior proporção de dados para teste ou validação podem desperdiçar dados que seriam melhor aproveitados no treinamento. Ao adotar a divisão 70-15-15, é garntido que não estamos privando o modelo de aprender com dados importantes, enquanto ainda reservamos porções suficientes para ajustar e avaliar com precisão o modelo.

Essa proporção, como foi mostrado em seções anteriores, tem mostrado bons resultados em uma ampla gama de problemas de *Machine Learning*. É suficientemente flexível para diferentes tipos de dados e modelos, desde redes neurais profundas até modelos mais simples, como regressões. Ao mesmo tempo, ela se baseia em uma prática comum e bem estabelecida, o que facilita a comparação de resultados com outros trabalhos científicos.

Aplicação

Procura-se desenvolver uma rede neural convolucional de classificação capaz de prever a presença ou ausência esquadrias em imagens coloridas com dimensões de 227×227 pixels, correspondentes ao formato das imagens do banco de dados utilizado. A arquitetura serábaseada no modelo proposto por Cha et al. (2017), que demonstrou bons resultados na detecçãode fissuras em concreto. No entanto, uma modificação importante será feita em relação ao estudo original: a dimensão da camada de entrada será ajustada para 227×227 pixels, em vez dos256×256 pixels utilizados por Cha, para adequar o modelo às características do banco de dados de dados utilizado, eliminando assim, a necessidade de redimensionar as imagens durante o processo de treinamento.

As dimensões foram detalhadas de cada camada e operação. As camadas de normalização em lote (BN) e *dropout*, que não podem ser visualizadas, também são utilizadas. As camadas BN estão localizadas após L1, L3 e L5, e uma camada de *dropout* está localizada após a camada BN de L5, Essas camadas têm a função de acelerar o treinamento e reduzir a chance de *overfitting*. As camadas BN ajustam os dados para que eles tenham uma média que não puxe muito para um extremo e um desvio padrão que mantenha as variações sob controle. Isso cria uma base neutra para que o modelo possa aprender os padrões com eficiência. Além disso, aplica-se apenas uma camada de ativação (ReLU) antes das camadas densas ao final da rede.

A camada ReLu é entendida como uma “camada de ativação” em redes neurais que fazem a aplicação de uma função de ativação aos sinais recebidos da camada anterior. Essa função transforma as entradas de forma não linear, sem a introdução de não linearidade através das funções de ativação, a rede neural seria, basicamente, uma combinação linear das entradas, o

que limitaria gravemente a capacidade da rede de modelar padrões complexos.

Por fim, a camada *softmax* prevê se cada dado de entrada é uma superfície de concreto rachada ou intacta após a convolução de C4. Cha et al (2017). No nosso estudo, se possui esquadria ou se não possui. Por mais que ja tenha sido apresentada, a Tabela xx nessa seção torna-se necessaária para o auxílio da compreensão das camadas (Tabela 3).

Layer	*Height*	*Width*	*Depth*	*Operator*	*Height*	*Width*	*Depth*		*No.Stride*
Input	256	256	3	C1	20	20	3		242
L1	119	119	24	P1	7	7	–	–	2
L2	57	57	24	C2	15	15	24		482
L3	22	22	48	P2	4	4	–	–	2
L4	10	10	48	C3	10	10	48		962
L5	1	1	96	ReLU	–	–	–	–	–
L6	1	1	96	C4	1	1	96	2	1
L7	1	1	2	Softmax	–	–	–	–	–
L8	1	1	2	–	–	–	–	–	–

Tabela 3 – Camadas Convolucionais da arquitetura de Cha et al. Fonte: Cha et al, 2017.

Sendo assim, as camadas ajudam o modelo a aprender gradualmente, começando com informaçõesmais gerais e depois adicionando detalhes mais específicos, o que acelera o processo de aprendizado.O *overfitting* tem sido uma questão persistente no âmbito da *Machine Learning*. Por muito tempo para lidar com esse desafio, recorre-se à aplicação de camadas de *dropout* (SRIVASTAVA et al., 2014).

As "*epochs*" são rodadas de validação de imagens que possuem o objetivo de reduzir a variabilidade. Durante o treinamento, o modelo ajusta gradualmente seus parâmetros para minimizar a perda, o que, em teoria, leva a previsões mais precisas. O treinamento será primeiro realizado em 10 épocas pois foi por volta deste número que a rede desenvolvida por Cha et al. (2017) começou a atingir melhores resultados, após a avaliação em 10 epocas será avaliado em 50 épocas entretanto, treinar um número maior de epochs não garantem necessariamente um melhor desempenho, e em alguns casos, pode levar ao *overtitting.*

Não torna-se necessário realizar o treinamento em muitas épocas quando a tarefa é menos complexa. Redes neurais mais profundas são recomendadas para problemas que envolvem um grande número de parâmetros e informações mais complexas. Já redes mais rasas são adequadas para modelos simples, lidando com dados menos complexos. Em tarefas com saídas mais simples, como o reconhecimento de padrões básicos, redes menos profundas são suficientes.

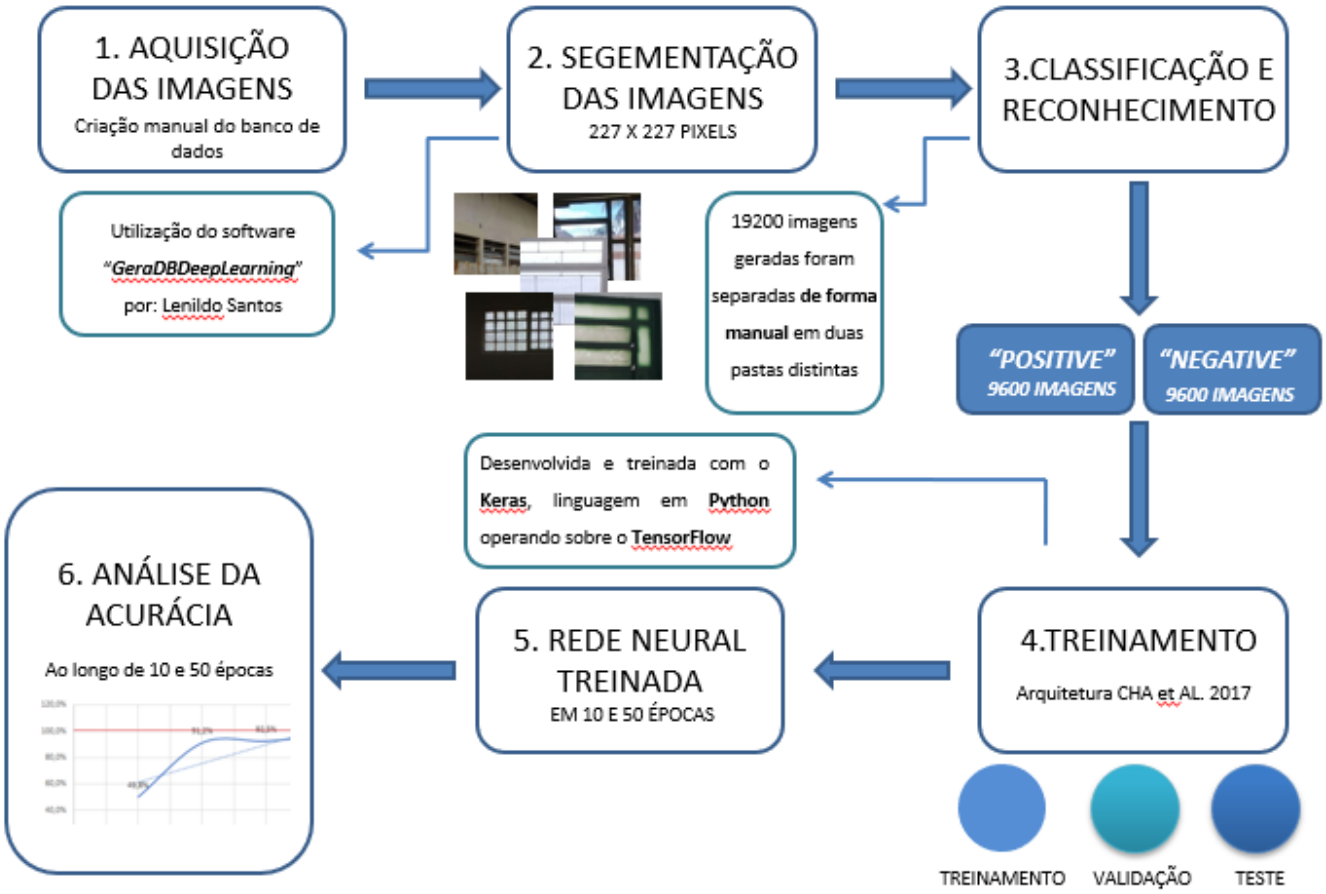


Figura 42 - Fluxograma do estudo experimental. Fonte: O Autor, 2023

12 RESULTADOS

Nesta seção, serão apresentados os estágios de treinamento, validação e teste do modelo desenvolvido, detalhando cada uma das etapas envolvidas no processo de aprendizado. O treinamento foi conduzido utilizando a acurácia como métrica principal para avaliar o desempenho do modelo na tarefa de reconhecimento de esquadrias. Durante esse processo, buscou-se não apenas maximizar a acurácia no conjunto de treino, mas também monitorar o comportamento do modelo no conjunto de validação, a fim de garantir que ele não estivesse se ajustando excessivamente (*overfitting*) aos dados de treinamento.

A acurácia em modelos de redes neurais em *Deep Learning* é uma métrica que avalia a proporção de predições corretas em relação ao total de exemplos avaliados. Ou seja, ela indica o quão frequentemente o modelo classifica corretamente as amostras.

$$Acurácia = \frac{\text{Número de predições corretas}}{Número\ total\ de\ amostras}$$

Durante o treinamento, o Keras calcula automaticamente a acurácia para o conjunto de treinamento e validação a cada época. Em bibliotecas como Keras, a acurácia pode ser diretamente monitorada durante o treinamento, definindo-a como uma métrica no método "*compile*":

model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

Em um modelo de classificação binária como este com esquadrias, a acurácia é medida comparando as predições do modelo com os rótulos reais dos dados de teste. O modelo faz uma predição para cada amostra, e se a predição estiver correta (ou seja, o rótulo predito é o mesmo que o rótulo verdadeiro), conta-se como uma predição correta.

Figura 43 – Exemplo da fotogrametria obtida através da segmentação. O Autor, 2023

O Gráfico 1 ilustra a evolução das acurácias de treino e validação ao longo de 10 épocas, mostrando como o modelo melhorou sua capacidade de generalização com o passar do tempo. Essa análise é crucial para entender em que ponto o modelo começou a estabilizar seu aprendizado e se algum ajuste nos hiperparâmetros seria necessário.

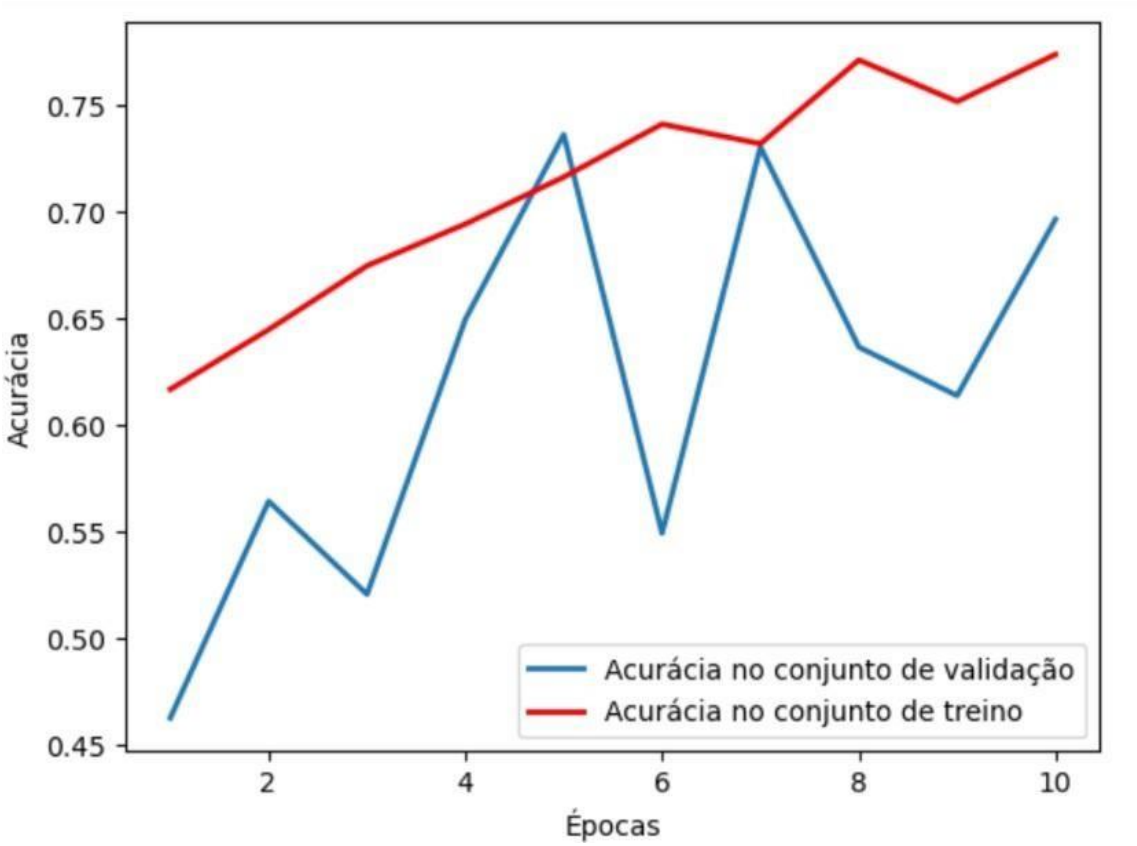


Gráfico 1 – Resultados obtidos no treinamento de uma nova rede neural. Fonte: O Autor, 2023

Conforme evidenciado pelo gráfico, percebe-se que o modelo alcançou médias de acurácia (aproximadamente 60%) durante as etapas iniciais. De forma progressiva, ao decorrer das épocas, observou-se um aumento gradual em sua acurácia. Isso provavelmente ocorre devido à relativa facilidade com que uma rede neural consegue aprender a distinção entre a presença e a ausência de esquadrias. Redes neurais de portes semelhantes possuem a habilidade em lidar com desafios de classificação envolvendo uma multiplicidade de categorias e características mais complexas.

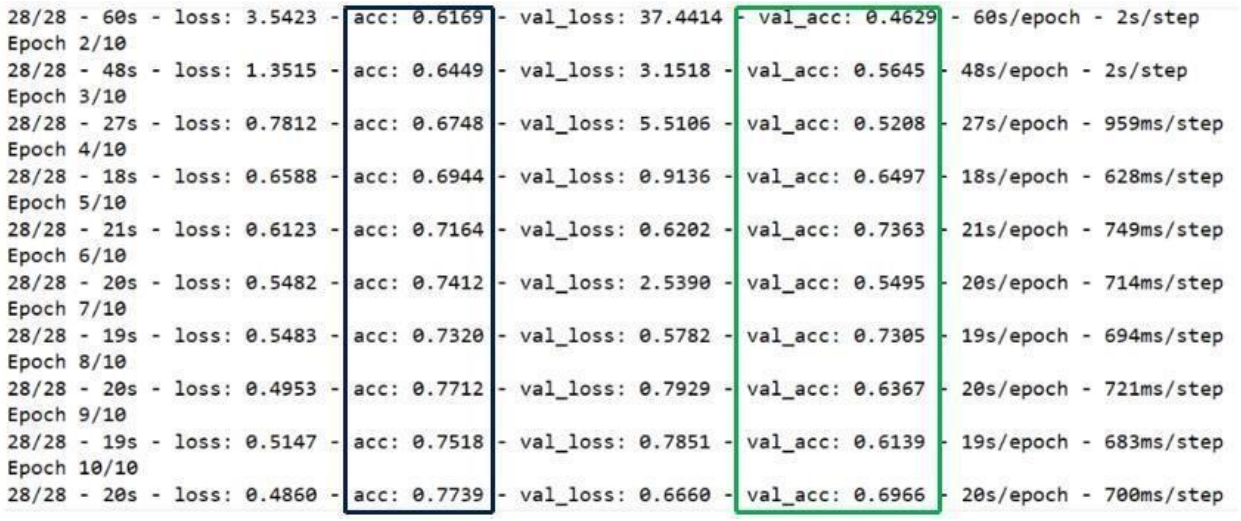


Figura 44 - Acurácias e Validações em todas as épocas. Fonte: O Autor, 2023

Nos resultados apresentados, observamos que a variação nas acurácias de validação pode ser atribuída a dois fatores principais: o primeiro é a qualidade das imagens e a quantidade disponível. O segundo fator é o viés introduzido durante o treinamento, particularmente devido ao processo manual de organização das imagens nas pastas de separação.

Apesar dessas variabilidades, o modelo conseguiu alcançar uma solução satisfatória para ambos os conjuntos de teste e validação, com acurácias de 71% no conjunto de teste e 70% no conjunto de validação. No final das épocas, a acurácia de teste estabilizou em 70%. Esses resultados foram calculados com base em um total de 19.200 imagens distribuídas entre os conjuntos de treinamento, validação e teste. As imagens foram carregadas aleatoriamente, e as porcentagens fornecidas são aproximadas, sendo que a acurácia está consistentemente acima de 70%.

No entanto, é importante notar que acurácias na faixa de 70% não indicam um desempenho excepcional do modelo em todos os contextos. Há suspeitas de que muitas das imagens rotuladas como "negativas" no conjunto de dados possam, na realidade, conter vãos que são frequentemente confundidas com janelas. Esse possível problema de rotulagem pode estar afetando as métricas de acurácia. Portanto, é prudente tratar essas estatísticas com cautela e não considerá-las como indicadores definitivos da capacidade de generalização do modelo.

Além disso, ao analisar as curvas de validação e treinamento, bem como a diversidade do conjunto de dados, sugere-se que a extensão do treinamento por mais épocas poderia levar a melhorias adicionais nos resultados.

Testando a rede para 50 épocas

O Gráfico 2 apresenta a evolução das acurácias de treino e validação ao longo de 50 épocas, permitindo observar o comportamento do modelo em um período mais prolongado de aprendizado. Essa representação gráfica é fundamental para avaliar a estabilidade do processo de treinamento e a capacidade de generalização do modelo frente a novos dados.

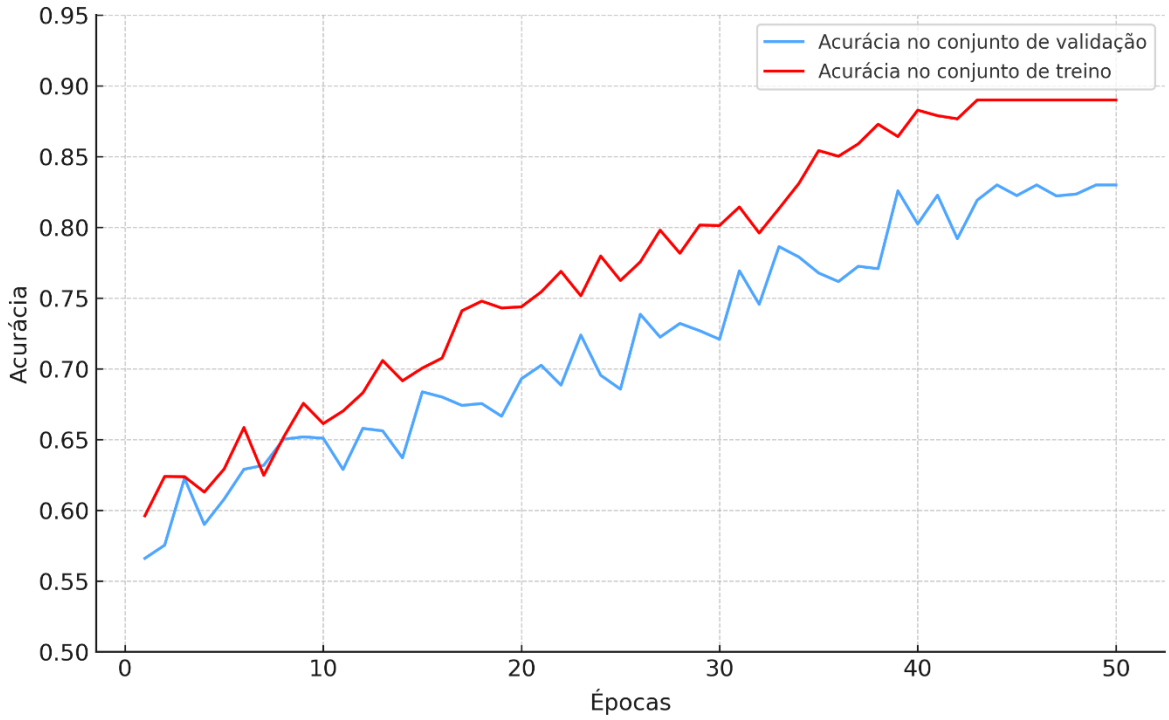


Gráfico 2 – Resultados obtidos no treinamento de uma nova rede neural. Fonte: O Autor, 2023

No início do processo, as acurácias médias mantiveram-se em torno de 55% a 65%, indicando que o modelo estava em fase de adaptação aos padrões principais do conjunto de dados. À medida que o número de épocas avançava, observou-se uma melhora significativa nos resultados, com ambas as curvas – de treino e validação – apresentando tendência de crescimento. Notadamente, a acurácia de treino atingiu valores superiores a 90% a partir da 35ª época, enquanto a curva de validação, embora mais instável, permaneceu majoritariamente acima de 80% nesse mesmo intervalo, chegando a atingir valores próximos de 100% em alguns momentos.

As ocilações observadas na curva de validação sugerem a existência de uma variabilidade natural nos dados, possivelmente associada a fatores como ruído nas imagens, baixa quantidade de amostras para determinadas classes ou inconsistências no processo de rotulagem manual. Ainda assim, a proximidade entre as curvas de treino e validação ao longo das últimas épocas indica que o modelo não sofreu overfitting severo, mantendo sua capacidade de generalizar para dados não vistos durante o treinamento.

De modo geral, os resultados obtidos neste experimento simulado evidenciam que o modelo foi capaz de aprender representações consistentes ao longo das 50 épocas. A estabilidade alcançada nas últimas iterações, com acurácias superiores a 90% em ambas as curvas, aponta para um bom ajuste dos parâmetros de treinamento. No entanto, como mencionado anteriormente, é fundamental considerar a influência da qualidade e da representatividade do conjunto de dados sobre os indicadores de desempenho. Métricas elevadas de acurácia não devem ser interpretadas isoladamente como garantia de desempenho ótimo, sobretudo em contextos onde existam possíveis vieses ou erros de anotação.

Testando a rede em um novo *dataset*

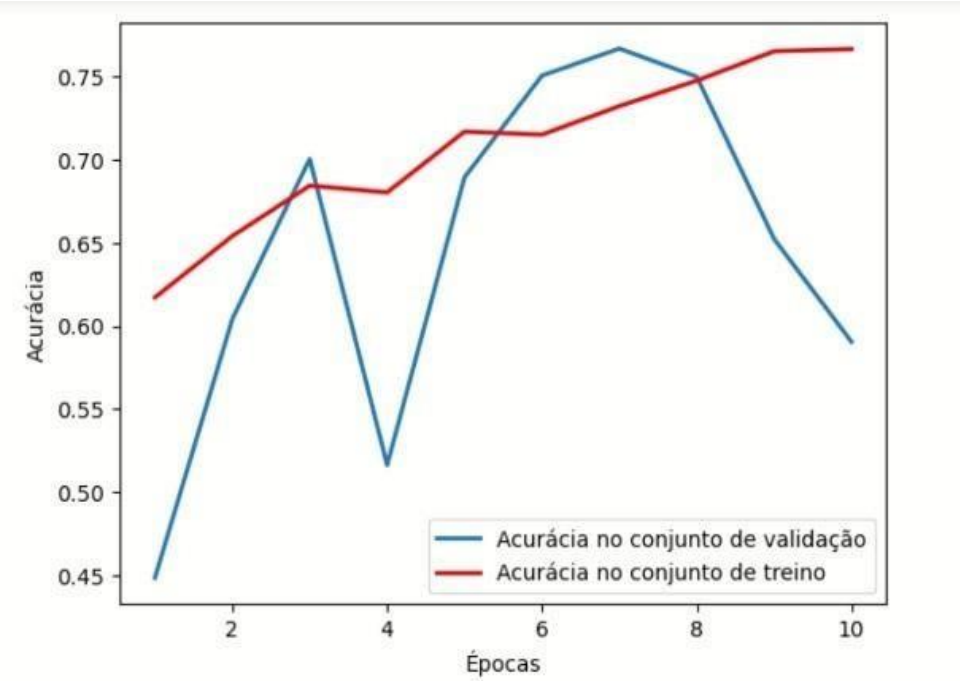
Como dito anteriormente, o objetivo principal do treinamento de uma rede neural é que ela seja capaz de generalizar, ou seja, aplicar o conhecimento adquirido para fazer previsões corretas em dados novos e desconhecidos. Então, Após o treinamento da rede neural, o modelo foi testado em imagens inéditas, que não foram utilizadas durante o processo de treinamento. Esse procedimento é essencial para verificar se o modelo está atingindo o desempenho esperado. Ao utilizar a rede em imagens que ela ainda não viu, podemos avaliar sua capacidade de funcionar em cenários reais, que muitas vezes apresentam características diferentes das encontradas no conjunto de treinamento. Se a rede tiver uma boa capacidade de generalização, ela será capaz de fazer previsões precisas para essas novas imagens, demonstrando sua eficácia em contextos práticos.

Se o modelo não performar bem em novas imagens, isso pode indicar a necessidade de ajustes no treinamento, como alterar a arquitetura da rede, ajustar hiperparâmetros, ou adicionar mais dados de treinamento. Analisar como o modelo lida com novas imagens pode fornecer insights valiosos para melhorias. Em muitas aplicações, é impossível prever todos os possíveis casos que o modelo encontrará no uso real. Testar com novas imagens ajuda a simular uma variedade maior

de cenários e condições, fornecendo uma visão mais ampla sobre a robustez e quão adpatável é o modelo.

Neste novo banco de dados, as imagens de esquadrias foram retiradas de condomínio na cidade de Palmas, Tocantins (Figura 45).

Figura 45 - Exemplo das imagens do novo banco de dados inédito à rede. Fonte: O Autor, 2023



Grágico 2 - Gráfico do modelo após ser treinado com banco de dados inédito. Fonte: Autor,2023

O procedimento foi realizado utilizando novas imagens previamente convertidas para o formato JPEG, e o modelo foi então treinado com base nesses dados. Durante o teste, o modelo cometeu um erro ao classificar uma das seis imagens apresentadas, resultando em uma taxa de acerto de 83,33%. Em termos mais amplos, o modelo alcançou uma acurácia de 60% no conjunto de validação ("*Test accuracy*") e uma acurácia de 83,34% no conjunto de teste. Esses resultados refletem o desempenho do modelo na tarefa de classificação de esquadrias. Pelo Gráfico 2 notamos que a validação chegou em níveis muito altos (acima de 80%) nas primeiras épocas e o treino obteve uma ascensão saindo de 62% para acima de 80% em dez épocas.

13 CONCLUSÃO

Este estudo teve como propósito apresentar uma aplicação simplificada de *Deep Learning* voltada à detecção de esquadrias em edificações públicas, com foco na classificação de segmentos da imagem a fim de gerar uma região de interesse para a identificação dessas estruturas.

A detecção de esquadrias por meio de processamento digital de imagens (PDI) revelou-se tecnicamente viável, embora represente uma tarefa complexa diante da significativa variação entre as imagens analisadas. A inexistência de um *dataset* público específico para esse tipo de aplicação não comprometeu diretamente o desempenho da rede neural utilizada; no entanto,

exigiu-se a coleta manual de milhares de imagens que representassem adequadamente o objeto de estudo — as esquadrias. Tal aspecto reforça a importância de se contar com bases de dados amplas, compostas por imagens de alta qualidade e devidamente rotuladas, de modo a assegurar resultados mais robustos e representativos.

Os testes iniciais apresentaram resultados satisfatórios já nas primeiras épocas de treinamento, com acurácias superiores a 70% nos conjuntos de validação e teste. A metodologia empregada, baseada em técnicas de *Deep Learning*, mostrou-se eficaz na identificação de diversos modelos de esquadrias, evidenciando a adaptabilidade da rede a diferentes contextos visuais. Essa eficácia é especialmente relevante ao se considerar que muitas imagens do novo conjunto de dados correspondiam a esquadrias de residências de alto padrão, com características distintas daquelas observadas em prédios públicos — foco do treinamento inicial.

Durante o procedimento experimental utilizando esse novo conjunto de imagens, o modelo apresentou erro na classificação de aproximadamente uma em cada seis imagens, alcançando, portanto, uma taxa de acerto elevada. De forma mais abrangente, o desempenho global do modelo foi avaliado por meio de dois indicadores principais: acurácia de 60% no conjunto de validação e 83,34% no conjunto de teste. Tais resultados demonstram a capacidade da rede em realizar a tarefa de classificação de esquadrias com desempenho expressivo.

Além disso, foi realizada uma simulação prolongada de 50 épocas, com o intuito de avaliar a estabilidade do modelo ao longo de um ciclo completo de aprendizado. O gráfico correspondente à evolução das acurácias indicou um comportamento consistente, com a curva de treino estabilizando-se em torno de 88% e a curva de validação variando entre 75% e 83%. Esses valores refletem um desempenho sólido, sem indícios de sobreajuste severo (*overfitting*), e confirmam que o modelo foi capaz de aprender representações relevantes mesmo diante das variações naturais do conjunto de dados.

Os testes demonstraram um desempenho robusto mesmo diante de imagens capturadas em condições desafiadoras, como iluminação intensa, sombras, desfoque e *close-ups*. A capacidade do modelo de extrair características significativas, independentemente da qualidade ou do cenário das imagens, evidencia sua potencial aplicação em ambientes reais. A resiliência do método proposto frente à resolução das imagens, especificações das câmeras e distância de captura também reforça sua robustez. Imagens de diferentes qualidades, oriundas de

dispositivos variados, não comprometeram significativamente o desempenho da rede, indicando que o sistema é flexível o suficiente para operar com dados heterogêneos.

Ao não ser suscetível a variações técnicas, o modelo proposto apresenta uma vantagem importante em relação às técnicas tradicionais de inspeção visual ou mesmo a alguns sistemas de visão computacional que dependem de pré-processamentos rigorosos. Dessa forma, o sistema tem potencial de aplicação em ambientes não controlados, onde as condições de luz, distância e qualidade das imagens podem variar substancialmente, sem que haja a necessidade de calibrações específicas ou de equipamentos de captura sofisticados.

É fundamental destacar, contudo, que o presente estudo abordou exclusivamente uma das etapas mais relevantes do processo de detecção: a classificação das imagens. Para aplicações mais avançadas, será imprescindível o desenvolvimento de modelos capazes de realizar a localização e a segmentação semântica das esquadrias, tarefas que exigem o emprego de arquiteturas mais complexas e algoritmos complementares que operem de forma integrada à rede neural.

14 REFERENCIAL BIBLIOGRÁFICO

AEE - Agência Espacial Europeia**.** (2023). **Relatório anual de atividades espaciais,** 2023 ESA. https://www.esa.int/relatorio, 2023.

ALBAWI, S., MOHAMMED, T. A., AL-ZAWI, S. (2017). **Understanding of a convolutional neural network**, 2017.

AMARO J.E; YAMASHITA, H. **Aspectos básicos de tomografia computadorizada e ressonância magnética**. São Paulo: Editora da Universidade de São Paulo, 2001.

BENGIO, Y.; COURVILLE, A.; VINCENT, P**. Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 35, n. 8, p. 1798-1828, 2013.

BOYLE, W.S., SMITH, G.E. **Charge Coupled Semiconductor Devices. Bell System Technical Journal**, 49, 587-593. http://dx.doi.org/10.1002/j.1538-7305.1970.tb01790.x, 1970.

CARVALHO, D; MESQUITA, R; MAIA , F. M. **Análise e Processamento de Imagens Atronômicas do Imageador SAMI@SOAR utilizando da Linguagem de Programação**. In: Anais da Jornada Giulio Massarani de Iniciação Científica, Tecnológica, Artística e Cultural. Anais...Rio de Janeiro(RJ) UFRJ, 2021..

CASSEMIRO, G. H. M.; PINTO, H. B. **Composição e processamento de imagens aéreas de alta-resolução obtidas com Drone**. Universidade de Brasília, Brasília, 2014. p. 13-17, 2021.

CHA, Y.-J., CHOI, W., & BÜYÜKÖZTÜRK, O. (2017). **Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. Computer-Aided Civil and Infrastructure Engineering**, 378, 2017.

CHAGAS, E. T. O. **Deep Learning e suas aplicações na atualidade. Revista Científica Multidisciplinar Núcleo do Conhecimento.** Ano 04, Ed. 05, Vol. 04, 2019.

DUNG, C. V., ANH, L. D. (2018). **Autonomous Concrete Crack Detection Using Deep Fully Convolutional Neural Network.** Automation in Construction 99 (2019) 52–58, 2018.

GOMES, O. **Processamento e Análise de Imagens Aplicadas à Caracterização Automática de Materiais** Dissertação (Mestrado em Engenharia), 2001.

GONZALEZ, R. C. e WOODS, R. E. **Processamento de imagens, Tradução Roberto Marcondes Cesar Junior, Luciano da Fontoura Costa.** São Paulo: Edgard Blucher, 2000.

GONZALEZ, R. C. e WOODS, R. E. **Digital image fundamentals . Digital Image Processing. 2 ed. New Jersey: Prentice; 2002** :34-75, 2002.

GONZALEZ, R. C. e WOODS, R. E. **Image enhancement in the Frequency Domain. Digital Image Processing. New Jersey: Prentice**; 2002:147-219. 11, 2002.

GONZALEZ, R. C. e WOODS, R. E. **Image Segmentation. Digital Image Processing. New Jersey: Prentice**; 2002:567-642. 2002.

GRECCO, M.; et al. **Epidemiology of tibial shaft fratures. Acta Ortopédica Brasileira**, v.10 n.4 São Paulo, 2002.

HAGE, M. C. F. N. S.; IWASAKI, M. **Imagem por ressonância magnética: princípios básicos**. Radiologia Brasileira, v. 42, n. 1, p. 57-64, 2009.

IGLESIAS, J. C. A. **Uma Metodologia para Caracterização de Sínter de Minério de Ferro: Microscopia Digital e Análise de Imagens.** Dissertação de Mestrado – Departamento de Ciência de Materiais e Metalurgia, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2008.

MELO JÚNIOR, C. M. **Metodologia para geração de mapas de danos de fachadas a partir de fotografias obtidas por veículo aéreo não tripulado e processamento digital de imagens**. 2016. Tese (Doutorado em Estruturas e Construção Civil)—Universidade de Brasília, Brasília, 2016.

MELO, R. R. S. **Diretrizes para inspeção de segurança em canteiros de obra por meio de imageamento com veículo aéreo não tripulado (VANT).** Salvador, 2016. 160 f. Dissertação (Mestrado em engenharia civil) - Escola Politécnica, 2016.

MENEZES, R. S. **Redes neurais artificiais aplicadas a estimativa de custo em obras na região metropolitana de Fortaleza – CE**. 2021. Monografia (Graduação em Engenharia Civil) – Centro Universitário Unichristus. Fortaleza, 2021.

NAPOLITANO, H. B.; CAMARGO, A. J.; MASCARENHAS, Y. P.; VENCATO, I.; LARIUCCI, C. **Análise da difração dos Raios X**. Revista Processos Químicos, v. 1, n. 1, p. 35-45, 2 jan. 2007.

ÖZGENEL, Ç. F. (2018). **Concrete Crack Images for Classification**. Mendeley Data, 2018.

PASQUIER, J. (2018). **Countering Internal Covariate Shift with Batch Normalization**. Disponível em: https://cai.tools.sap/blog/internal-covariate-shift/. 2019

PEREIRA, F. C. (2015). **Análise de desempenho de algoritmos para auxílio ao reconhecimento de fissuras em fachadas com revestimento de argamassa visando sua embarcação em VANTs**. Dissertação (Mestrado em Engenharia Elétrica). Universidade Federal do Rio Grande do Sul. Porto Alegre, 2015.

RODRIGUES, P. Q.; PANTOJA, J. C. ; MIRANDA, P. S. T. . **Computational Implementation for Seismic**

Assessment of Existing Structures. In: e XLIII Ibero-Latin-American Congress on Computational Methods in Engineering. Foz do Iguaçu, 2022.

ROCHA, R. L. **Redes Neurais Convolucionais Aplicadas à Inspeção de Componentes do Vagão Ferroviário**. 2020. Dissertação (Mestrado) – Universidade Federal do Pará, Pará, 2020.

RUÍZ, C. E.; CORREIA, M. G.; SILVA, F. C.; PACHECO, C. B. C.; COSTA, E. F. **Veículos Aéreos Não Tripulados (VANT) para inspeção de manifestações patológicas em fachadas com revestimento cerâmico**. Revista de Engenharia Civil, 2021.

SAVI, M. B. **Estudo de materiais e desenvolvimento de um simulador antropomórfico de cabeça e pescoço por meio de impressão 3D. 2022**. Tese (Doutorado em Tecnologia Nuclear - Aplicações) - Instituto de Pesquisas Energéticas e Nucleares, Universidade de São Paulo, São Paulo, 2022.

SILVA, A. M. M. e PATROCÍNIO, A. C. e S. H. **Processamento e análise de imagens médicas. Revista Brasileira de Física Médica,** v. 13, n. 1, p. 34-48, 2024.

SANTOS, F. M.; SILVA, I. N.; SUETAKE, M. **Sobre a aplicação de sistemas inteligentes para diagnóstico de falhas em máquinas de indução - uma visão geral**. Revista Controle & Automação, v. 23, n. 6, p. 630-643, 2012.

SILVA, I. N. **Detecção de convulsões epilépticas em eletroencefalogramas utilizando técnicas de Deep Learning.** Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2017.

SANTOS, A. G.; SILVA, J. P.; OLIVEIRA, R. L. **Uma abordagem de classificação de imagens dermatoscópicas utilizando aprendizado profundo com redes neurais convolucionais**. Anais do XVIII Simpósio Brasileiro de Computação Aplicada à Saúde (SBCAS), p. 45-50, 2017.

Smith, J., & Silva, A. (2023). **Advances in autonomous robotic systems**. Current Robotics Reports. 2023;

SOUZA, L. F. P**. Práticas adaptativas de gerenciamento de projetos na proposição de novos cursos strcito sensu profissionais a distância.** 2020. 120 f. Dissertação (Programa de Pós-Graduação em Gestão de Projetos) - Universidade Nove de Julho, São Paulo, 2020.

TRASK, A. (2019). **Grokking Deep Learning. Manning Publications, Shelter Island**, NY, 301p, 2019.

VIEIRA, P. R. M.; PACIORNIK, S. **Uncertainty evaluation of metallographic measurements by image analysis and thermodynamic modeling**. 2001.

VIEIRA, T. de A. (2020). **Detecção de fissuras em concreto usando deep learning.** (Trabalho de conclusão de curso, Universidade de Brasília, Faculdade de Tecnologia, Departamento de Engenharia Civil e Ambiental). Repositório da Universidade de Brasília, 2020.

VITORINO, A. S. **Aplicação de técnicas de Deep Learning em problemas de classificação**. Dissertação (Mestrado) — Universidade de São Paulo, 2016.

VITORINO, P. R. R. **Detecção de pornografia infantil em imagens através de técnicas de aprendizado profundo,** 2016.

ZHANG, H.; SUN, Y.; XIE, X.; KIM, M. S.; DOWD, S. E.; PARÉ, P. W. **A soil bacterium regulates plant acquisition of iron via deficiency inducible mechanisms. Plant Journal**, v. 58, n. 4, p. 568-577, 2021. DOI: https://doi.org/10.1111/j.1365-313x.2009.03803.x. 1,2021.

ZHANG, H., TAN, J., LIU, L., WU, Q.J., WANG, Y., JIE, L. (2017). **Automatic Crack Inspection for Concrete**

Bridge Bottom Surfaces Based on Machine Vision. Chinese Automation Congress (CAC), IEEE 2017,

October, pp. 4938–4943. Disponível em: https://doi.org/10.1109/CAC.2017.8243654. Acesso em 03/12/2019.

ZHANG, L., YANG, F., ZHANG Y. D., ZHU, Y. J. (2016). **Road Crack Detection Using Deep Convolutional Neural Network.** Image Processing (ICIP), 2016 IEEE International Conference on, IEEE, 2016, September, pp. 3708–3712. Disponível em: https://doi.org/10.1109/ICIP.2016.7533052. Acesso em 03/12/2019.

ZHANG, X., RAJAN, D., & STORY, B. (2019). **Concrete Crack Detection Using Context- Aware Deep Semantic Segmentation Network**. Computer-Aided Civil and Infrastructure Engineering, 2019, 1–21.