

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA “PAULA SOUZA”**

**FACULDADE DE TECNOLOGIA DE BEBEDOURO**

**TECNOLOGIA EM BIG DATA NO AGRONEGÓCIO**

**CLASSIFICAÇÃO DE DOENÇAS AGRÍCOLAS POR  
IMAGEM**

**AUTOR: JOSIELI CRISTINA SILVA**

**ORIENTADOR: prof. Me. RENAN GUILHERME NESPOLO**

**BEBEDOURO**

**2024**

JOSIELI CRISTINA SILVA

**CLASSIFICAÇÃO DE DOENÇAS AGRÍCOLAS POR  
IMAGEM**

Monografia apresentada à Faculdade de Tecnologia  
de Bebedouro, como parte dos requisitos para a  
obtenção do título de Tecnólogo em Big Data no  
Agronegócio

Orientador: **prof. Me. Renan Guilherme Nespolo**

**BEBEDOURO**

2024

*Renda-se, como eu me rendi. Mergulhe no que você não conhece  
como eu mergulhei. Não se preocupe em entender, viver ultrapassa  
qualquer entendimento.*

CLARICE LISPECTOR

## **Agradecimentos**

Primeiramente, agradeço a Deus, pela força, sabedoria e discernimento concedidos ao longo da minha jornada. Sua presença constante iluminou meus caminhos, permitindo que eu superasse cada desafio e alcançasse os meus objetivos.

Aos meus filhos, Daniel e Giulia, meu amor eterno. Vocês são a razão do meu esforço e dedicação. O sorriso, as palavras de incentivo, os gestos de carinho de vocês me motivaram a continuar, mesmo nas horas mais difíceis. Este trabalho é nosso, e espero que ele inspire em vocês o valor da perseverança e da busca pelo conhecimento.

Aos meus amigos e professores, sou imensamente grata pelos três anos compartilhados de amizade e aprendizado mútuo. Vocês estiveram ao meu lado nos momentos de incerteza, e juntos enfrentamos cada desafio. Essa parceria foi fundamental para a realização deste trabalho.

Por fim, não posso deixar de mencionar o momento de dor pela perda do meu esposo, José Luiz. Sua memória e o amor que compartilhamos me deram a força necessária para continuar, mesmo quando a dor parecia insuportável. Este trabalho é também uma homenagem à sua memória, e carrego comigo a certeza de que estaria feliz e orgulhoso.

**SILVA, J. C. Classificação de Doenças Agrícolas por Imagem.** Trabalho de Graduação (Monografia). Centro Estadual de Educação Tecnológica “Paula Souza”. Faculdade de Tecnologia de Bebedouro. n° p. 2024.

## RESUMO

O trabalho teve o objetivo de desenvolver e avaliar modelos de machine learning. Nos modelos de classificação de imagens para identificação de doenças em laranjeiras foi aplicado redes neurais convolucionais (CNNs). Para o desenvolvimento dos modelos foi utilizado o *Dataset* com imagens de frutos de laranjas saudáveis e frutos acometidos por câncer cítrico, *greening* e mancha preta. O projeto foi pensando em contribuir para uma solução prática do campo devido à carência de profissional para fazer a identificação correta de doenças agrícolas, de modo a detectar precoce e precisamente essas doenças, para melhorar a produção e reduzir perdas. Diferentes modelos foram executados na máquina local. Estes modelos apresentam configurações diferentes, sendo três modelos personalizados a partir do Estado da Arte e modelos pré-treinados, deste modo permitiu a análise do seu impacto em termos de acurácia e tempo de execução. Entre todos os testes os que apresentaram melhores resultados foram o Modelo III com 90.35% de acurácia e o Modelo VI com 90.15% de acurácia. O Modelo III utilizou um conjunto de dados reduzido com 1090 imagens e configuração moderada, que equilibrou o custo operacional e a eficiência, enquanto o Modelo VI utilizou-se de modelo pré-treinado conhecido como *MobileNet*, com um conjunto de dados de 9.637 imagens. Técnicas de validação cruzada, aumento de dados, permitiu uma melhor generalização dos modelos, diminuindo o risco de *Overfitting*. Contudo, o presente estudo reforça a importância da customização de parâmetros para maximizar o desempenho, portanto sugere aprimoramentos futuros, que vão desde a ampliação do conjunto de dados e o uso de técnicas mais avançadas de aprendizado por transferência, o que poderá tornar o modelo mais robusto e adaptável a diferentes condições.

**Palavras-chave:** Agronegócio. Pragas. Machine Learning. Redes neurais convolucionais. Classificação de imagens.

**SILVA, J. C. Classificação de Doenças Agrícolas por Imagem.** Trabalho de Graduação (Monografia). Centro Estadual de Educação Tecnológica “Paula Souza”. Faculdade de Tecnologia de Bebedouro. n° p. 2024.

## **ABSTRACT**

*The objective of this study was to develop and evaluate machine learning models. Convolutional neural networks (CNNs) were applied in image classification models for the identification of diseases in orange trees. The models were developed using a Dataset containing images of healthy orange fruits and fruits affected by citrus canker, greening, and black spot. The project aimed to contribute to a practical solution for the field, addressing the lack of professionals available to correctly identify agricultural diseases, enabling early and accurate detection to improve production and reduce losses. Different models were executed on a local machine. These models featured various configurations, including three models customized from the state-of-the-art and pre-trained models, allowing the analysis of their impact in terms of accuracy and execution time. Among all the tests, the best results were achieved by Model III with 90.35% accuracy and Model VI with 90.15% accuracy. Model III used a reduced Dataset of 1,090 images and a moderate configuration that balanced operational cost and efficiency, while Model VI utilized a pre-trained model known as MobileNet, with a Dataset of 9,637 images. Cross-validation techniques and data augmentation enabled better generalization of the models, reducing the risk of Overfitting. However, this study emphasizes the importance of parameter customization to maximize performance, suggesting future improvements such as expanding the Dataset and employing more advanced transfer learning techniques, which could make the model more robust and adaptable to different conditions.*

**Keywords:** *Agribusiness. Pests. Machine Learning. Convolutional neural networks. Image classification.*

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>12</b>
<b>1.1</b>	<b>OBJETIVOS.....</b>	<b>13</b>
<b>1.2</b>	<b>ESTRUTURA DO TRABALHO .....</b>	<b>13</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>14</b>
<b>2.1</b>	<b>MANEJO AGRÍCOLA .....</b>	<b>14</b>
<b>2.2</b>	<b>AGRICULTURA E AS TECNOLOGIAS.....</b>	<b>16</b>
<b>2.3</b>	<b>VISÃO COMPUTACIONAL.....</b>	<b>17</b>
<b>2.3.1</b>	<b>PROCESSAMENTO DE IMAGENS .....</b>	<b>17</b>
<b>2.4</b>	<b>APRENDIZADO DE MÁQUINAS.....</b>	<b>19</b>
<b>2.4.1</b>	<b>APRENDIZADO SUPERVISIONADO .....</b>	<b>20</b>
<b>2.4.2</b>	<b>REDES NEURAIS ARTIFICIAIS.....</b>	<b>21</b>
<b>2.4.3</b>	<b>APRENDIZADO PROFUNDO.....</b>	<b>22</b>
<b>2.4.4</b>	<b>REDES NEURAIS CONVOLUCIONAIS .....</b>	<b>26</b>
<b>2.4.5</b>	<b>TREINAMENTO .....</b>	<b>32</b>
<b>3</b>	<b>TRABALHOS RELACIONADOS .....</b>	<b>34</b>
<b>3.1</b>	<b>PREPARAÇÃO DE EXPERIMENTOS BASEADOS NO ESTADO DA ARTE.....</b>	<b>35</b>
<b>4</b>	<b>METODOLOGIA .....</b>	<b>37</b>
<b>4.1</b>	<b><i>DATASET</i> .....</b>	<b>37</b>
<b>4.2</b>	<b>PRÉ-PROCESSAMENTOS.....</b>	<b>39</b>
<b>5</b>	<b>RESULTADOS.....</b>	<b>42</b>
<b>6</b>	<b>CONCLUSÃO .....</b>	<b>50</b>
	<b>REFERÊNCIAS.....</b>	<b>51</b>

## LISTA DE FIGURAS

<b>Figura 1:</b>	<b>Drone na agricultura.</b> .....	<b>15</b>
<b>Figura 2:</b>	<b>Hierarquia de Processamento</b> .....	<b>19</b>
<b>Figura 3:</b>	<b>Modelos de classificação e regressão.</b> .....	<b>21</b>
<b>Figura 4:</b>	<b>Único neurônio.</b> .....	<b>23</b>
<b>Figura 5:</b>	<b>Exemplos de Redes Neurais Rasas.</b> .....	<b>24</b>
<b>Figura 6:</b>	<b>Exemplos de Redes Neurais Profundas</b> .....	<b>25</b>
<b>Figura 7:</b>	<b>Exemplo de redução de dimensão com <i>Max-Pooling</i> (24×24 para 12×12 neurônios) ..</b> .....	<b>27</b>
<b>Figura 8:</b>	<b>Mapa de recurso com camadas convolucionais e <i>Max- Pooling</i></b> .....	<b>28</b>
<b>Figura 9:</b>	<b>Mapa de recurso completo</b> .....	<b>29</b>
<b>Figura 10:</b>	<b>Técnica de regularização com <i>Dropout</i>.</b> .....	<b>30</b>
<b>Figura 11:</b>	<b>Exemplo de operação de achatamento - <i>Flatten</i></b> .....	<b>31</b>
<b>Figura 12:</b>	<b>Exemplo de laranjas com mancha-preta no <i>Dataset</i>.</b> .....	<b>37</b>
<b>Figura 13:</b>	<b>Exemplo de laranjas com cancro cítrico no <i>Dataset</i>.</b> .....	<b>38</b>
<b>Figura 14:</b>	<b>Exemplo de laranjas com <i>greening</i> no <i>Dataset</i>.</b> .....	<b>38</b>
<b>Figura 15:</b>	<b>Exemplo de laranjas saudáveis no <i>Dataset</i>.</b> .....	<b>39</b>



## LISTA DE TABELAS

<b>TABELA 1: DESCRIÇÃO DAS CONFIGURAÇÕES DOS MODELOS CUSTOMIZADOS....</b>	<b>40</b>
<b>TABELA 2: DESCRIÇÃO DAS CONFIGURAÇÕES DOS MODELOS PRÉ-TREINADOS....</b>	<b>40</b>
<b>TABELA 3: RELEVÂNCIA DA COR PARA IDENTIFICAÇÃO DOS ASPECTOS DAS ..... LESÕES. ....</b>	<b>41</b>
<b>TABELA 4: CONFIGURAÇÕES DOS TESTES REALIZADOS.....</b>	<b>41</b>
<b>TABELA 5: CONFIGURAÇÕES DOS TESTES REALIZADOS E RESULTADO DA MÉDIA... DA ACURÁCIA DE CADA MODELO.....</b>	<b>42</b>
<b>TABELA 6: COMPARAÇÃO GERAL DOS MODELOS CUSTOMIZADOS. ....</b>	<b>45</b>
<b>TABELA 7: COMPARAÇÃO GERAL DOS MODELOS PRÉ-TREINADOS.....</b>	<b>47</b>

## LISTA DE GRÁFICOS

<b>Gráfico 1:</b>	<b>Função de ativação ReLU.</b>	<b>27</b>
<b>Gráfico 2:</b>	<b>Resultados de Acurácia Modelo I (Estado da arte).</b>	<b>43</b>
<b>Gráfico 3:</b>	<b>Resultados de Acurácia Modelo II (<i>Dataset</i> 1090 imagens, 32x32, lote 256 100 épocas) média de 90,23%.</b>	<b>43</b>
<b>Gráfico 4:</b>	<b>Resultados de Acurácia Modelo III (<i>Dataset</i> 1090 imagens, 64x64, lote 32 50 épocas) média de 90,35%.</b>	<b>44</b>
<b>Gráfico 5:</b>	<b>Resultados de Acurácia Modelo V (ResNet).</b>	<b>46</b>
<b>Gráfico 6:</b>	<b>Resultados de Acurácia Modelo VI (<i>MobileNet</i> 1).</b>	<b>46</b>
<b>Gráfico 7:</b>	<b>Resultados de Acurácia Modelo IV (VGG16).</b>	<b>47</b>
<b>Gráfico 8:</b>	<b>Acurácia dos treinamentos.</b>	<b>48</b>

## Lista de Siglas

AM – Aprendizado de Máquina  
CV – Visão Computacional  
GANs – Redes Geradoras Adversariais  
GPS – Sistema de Posicionamento Global  
IA – Inteligência Artificial  
IoT – Internet das Coisas  
 $k$ -NN –  $k$ -Vizinhos mais próximos  
LDA – Análise Discriminante Linear  
PDI – Processamento Digital  
CNN – Redes Neurais  
RGB – Vermelho, Verde, Azul  
SVM – Máquinas de Vetores de Suporte  
TICs – Tecnologias da Informação e Comunicação  
VGG – Grupo de Geometria Visual  
ReLU – *Rectified Linear Unit*  
Tanh – Tangente Hiperbólica

## 1 INTRODUÇÃO

Na agricultura moderna, a identificação precoce e precisa de doenças nas culturas é um desafio crítico para garantir a produtividade e a qualidade das colheitas. Tradicionalmente, a detecção de doenças agrícolas depende do conhecimento especializado e da observação manual, métodos que podem ser demorados e suscetíveis a erros humanos. Nesse cenário destaca-se a necessidade de soluções mais eficientes e precisas, e é justamente neste contexto que a aplicação de aprendizado de máquinas é uma abordagem promissora para automatizar e aprimorar o processo de identificação de doenças agrícolas (Correa *et al.*, 2002).

O problema principal desta área é dificuldade de identificar corretamente as doenças em estágios iniciais, especialmente quando diferentes enfermidades apresentam sintomas visuais semelhantes. Essa imprecisão pode levar a tratamentos inadequados, resultando em perdas significativas na produção e no aumento dos custos operacionais. Além disso, as mudanças climáticas e a intensificação das práticas agrícolas têm contribuído para a emergência de novas pragas e doenças, tornando o monitoramento contínuo e eficiente ainda mais desafiador (Silva & Pereira, 2021).

Nesse contexto, a aplicação de aprendizado de máquinas se justifica pela sua capacidade de processar grandes volumes de dados e identificar padrões complexos que seriam difíceis de reconhecer por métodos tradicionais. Ao treinar algoritmos com dados provenientes de diversas fontes, como imagens de alta resolução, é possível criar modelos que não apenas detectam, mas também classificam doenças com maior rapidez e precisão. Isso permite intervenções mais eficazes, minimizando perdas e contribuindo para a sustentabilidade da produção agrícola (Sabrol & Satish, 2016).

Dessa forma, o uso do aprendizado de máquinas na classificação de doenças agrícolas não só aborda a problemática da detecção imprecisa, como também promove inovações tecnológicas que podem transformar a agricultura. Ao integrar inteligência artificial com práticas agrícolas, essa abordagem oferece uma solução robusta para os desafios contemporâneos, garantindo uma produção mais segura, eficiente e sustentável (Medeiros *et al.*, 2011).

## 1.1 OBJETIVOS

Os objetivos do presente trabalho são:

- Classificar doenças agrícolas utilizando imagens.
- Propor um modelo de Rede Neural Convolutacional para identificar doenças provocadas por pragas agrícolas na cultura de citrus;
- Validar o modelo proposto de identificação de pragas por meio de avaliação quantitativa.

## 1.2 ESTRUTURA DO TRABALHO

O presente trabalho está estruturado em cinco partes principais. Com início na Introdução, onde é realizado a revisão teórica dos conceitos utilizados, a fim de analisar o problema, verificar a viabilidade de execução e fundamentar o desenvolvimento do projeto. Em seguida, a Fundamentação Teórica, onde são discutidas as metodologias já existentes que abordam o problema, tanto com técnicas de aprendizado de máquina quanto com modelos de visão computacional.

Na seção Materiais e Métodos são detalhados a metodologias aplicadas neste trabalho, onde todos os aspectos são abordados, desde a definição do conjunto de dados até as estratégias de avaliação utilizadas. Em Resultados e Discussão, os resultados obtidos são analisados, comparando-os com os métodos existentes, para chegar a uma conclusão sobre a eficácia das abordagens adotadas. Por fim, na seção de Conclusão, as considerações finais são apresentadas referente aos resultados alcançados, com discussão de possíveis melhorias da metodologia empregada e sugerindo direções para trabalhos futuros sobre o tema. As Referências utilizadas ao longo do trabalho são apresentadas ao final.

## 2 FUNDAMENTAÇÃO TEÓRICA

O trabalho foi desenvolvido em duas etapas, sendo a revisão de literatura e a experimentação prática utilizando um *Dataset* de imagens de frutos de laranjas. A revisão de literatura teve como bases fontes de dados confiáveis, como sites, livros e artigos publicados em revistas científicas. As plataformas utilizadas para a coleta dos dados foram Google Acadêmico, Science Direct, Embrapa e o Ministério da Agricultura, Pecuária e Abastecimento (Mapa). Esta revisão forneceu o embasamento teórico necessário para entender as principais tecnologias e inovações no setor agrícola, com foco na aplicação de soluções tecnológicas para otimizar a produção. Num segundo momento, foi conduzida uma análise prática utilizando um *Dataset* com imagens de frutos de laranjas com cancro cítrico, *greening* e mancha preta, três das principais doenças que afetam as laranjeiras e por fim imagens de frutos saudáveis. Com estas imagens foi realizado a classificação e comparação para identificar os padrões visuais que distinguem os danos provocados por esta três principais doenças dos frutos saudáveis, com o objetivo de auxiliar na identificação correta e apoiar na tomada de decisão sobre o manejo agrícola.

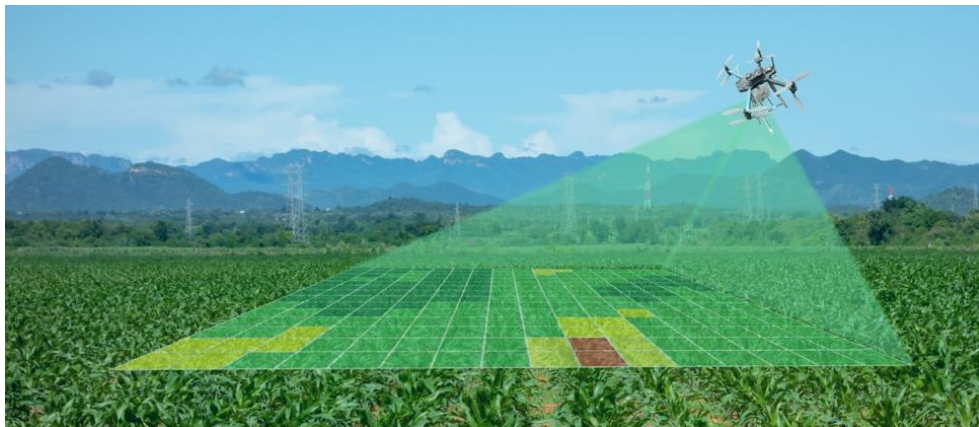
### 2.1 MANEJO AGRÍCOLA

A agricultura tem passado ao longo dos anos por muitas mudanças. Os avanços tecnológicos está cada vez mais presente no setor agrícola, e não só na administração, mas também nas operações de campo, permitindo a integração dos processos dos sistemas de produção em escala de forma mais prática e ágil (Mantovani *et al.*, 2020). O aumento populacional impulsionou um aumento na produção de alimentos, aliado aos impactos das mudanças climáticas e aos custos crescentes de produção são alguns dos fatores que levam os agricultores a buscarem inovações tecnológicas para manter a competitividade no setor agrícola (Olesen, 2016). A relação entre clima e produção agrícola é complexa, pois a variação nas temperaturas e precipitações pode afetar diretamente o crescimento das culturas, a disponibilidade de recursos hídricos e o rendimento das lavouras. Por isso, o desenvolvimento de tecnologias para adaptação e mitigação dos efeitos climáticos se torna essencial para garantir a sustentabilidade e eficiência da produção agrícola.

Essa necessidade de mercado, o interesse por inovações tecnológicas e o aumento da rentabilidade no setor agrícola, tem nos permitindo passar por uma evolução tecnológica neste

setor. A agricultura iniciou sua evolução pela agricultura 1.0, antes de 1950, onde um terço da população se dedicava a produção agrícola. Após passou pela agricultura 2.0 conhecida como revolução verde, iniciada da década de 50, onde prevaleceu melhorias genéticas, a utilização de defensivos e fertilizantes agrícolas sintéticos e o uso de máquinas agrícolas para incrementar a produção. Em seguida, tivemos a agricultura 3.0, conhecida como agricultura de precisão e iniciada na década de 90, utilizando biotecnologia e softwares para as aplicações agrícolas. O uso destas tecnologias permitiu o rastreamento com precisão da produtividade do campo em tempo real, medindo fatores como quantidade de grãos colhidos, a velocidade da colheita e a eficiência do processo em diferentes áreas, gerando mapas de produtividade, mostrando rendimento da colheita dentro de uma mesma lavoura. E por fim, as agriculturas 4.0 e 5.0, baseadas na Internet das coisas, Big Data, Computação em Nuvem, Sensoriamento Remoto, Robótica e Inteligência artificial (Melgar, 2018). Um dos exemplos dessas duas revoluções é o uso de drones na agricultura com o objetivo de sensoriar áreas em tempo real para prevenir pragas, como apresentado na Figura 1, entre outras aplicações.

**Figura 1: Drone na agricultura.**



Fonte: Brasil, Ministério da Agricultura (2020).

A importância agrícola é justificável por vários aspectos, inclusive no que tange uma disponibilidade maior de alimentos para a humanidade. Contudo, algumas práticas como as relacionadas a substituição da cobertura vegetal nativa, com várias espécies vegetais, por poucas espécies plantadas e cultivadas, tem provocado desequilíbrio ao meio ambiente. Este desequilíbrio favorece o crescimento populacional de insetos considerados pragas, que procuram na cultura alimento, causando danos, devido a reprodução intensa (Medeiros *et al.*, 2011).

Os insetos podem ser classificados como pragas ou inimigos naturais. São pragas quando se alimentam de lavouras em cultivo, e transmitem doenças, causam prejuízos. Os considerados inimigos naturais são benéficos, auxiliando no desenvolvimento das plantas e controlando os insetos considerados praga. A diversidade vegetal diminui a necessidade de intervenção, em virtude da entrada pelas bordaduras das propriedades, pois observa-se uma maior eficiência do controle biológico em regiões próximas à vegetação natural (Medeiros *et al.*, 2011).

## **2.2 AGRICULTURA E AS TECNOLOGIAS**

A partir de 2010 houve uma maior interação entre tecnologia da informação e comunicação (TICs), como GPSs e sensores e a agricultura. Atualmente essa interação é maior com a presença da internet das coisas (IoT), big data e inteligência artificial, tornando-as mais conectada e automatizada. Exemplos desta interação são vistos em irrigações automatizadas, drones de monitoramento e aplicação de defensivos agrícolas e análise de dados para tomada de decisões. A expressão “agricultura digital” indica o uso frequente nas tecnologias no campo, e evidencia já o uso de automação dos modelos de produção e de dados preditivos (Ahmad e Nabi, 2021).

No campo, tradicionalmente o monitoramento é realizado por meio de inspeções em campo, onde os técnicos se utilizam de observação visual e coletas de amostras de folhas para posterior análise. Isto caracteriza uma abordagem onerosa. O uso de dados de sensoriamento remoto, tanto imagem quanto curvas espectrais têm possibilitado o mapeamento do campo, isto em escala de talhões, planta e a nível foliar (Furlanetto *et al.* 2017).

Atualmente a interação entre sensoriamento remoto e aprendizagem de máquinas tem permitido a extração de padrões de dados. A aprendizagem de máquina faz parte da inteligência artificial e utiliza algoritmos, que podem ser paramétricos ou não, para detectar padrões em dados. Esses algoritmos têm a capacidade de garantir tais padrões por meio de uma análise hierárquica e não linear do conjunto de amostras. No caso da aprendizagem supervisionada, o algoritmo obtém exemplos específicos dos padrões que deve identificar, enquanto na aprendizagem não supervisionada, ele é capaz de aprender de forma independente (Guzmán *et al.*, 2018; Feng *et al.*, 2019).



## 2.3 VISÃO COMPUTACIONAL

A Visão Computacional (CV) é uma área da ciência que busca desenvolver tecnologias capazes de fazer com que máquinas "vejam" e extraiam informações de imagens digitais obtidas por sensores e dispositivos variados. A partir dessas informações, é possível identificar, analisar e processar dados relacionados aos objetos presentes nas imagens. Com o objetivo principal de aprendizado semelhante a capacidade humana na interpretação das imagens, são utilizadas técnicas de Processamento Digital de Imagens (PDI) e algoritmos de Aprendizado de Máquina (AM) (Ballard e Brown, 1982).

### 2.3.1 PROCESSAMENTO DE IMAGENS

Uma imagem pode ser definida como uma função bidimensional,  $f(x, y)$ , no qual  $x$  e  $y$  são coordenadas espaciais (plano), e a amplitude de  $f$  em qualquer par de coordenadas  $(x, y)$  é chamada de intensidade ou nível de cinza da imagem naquele ponto. As variáveis  $x$ ,  $y$  e os valores de intensidade de  $f$  são quantidades finitas e discretas, chamamos isso de imagem digital. O processamento digital de imagens é o campo que lida com o processamento de imagens digitais por meio de um computador digital. Vale destacar que uma imagem digital é composta por um número finito de elementos, cada um com uma localização e um valor específicos. Esses elementos são chamados de elementos pictóricos, elementos de imagem, pels e pixels, sendo pixel o termo mais popular para representar os elementos de uma imagem digital (Gonzalez, 2009).

Os humanos dependem da visão como um dos principais sentidos, tornando as imagens fundamentais para a percepção do mundo. No entanto, o campo de visão humano é restrito, ao contrário dos computadores, que podem processar várias informações, ampliando sua capacidade de análise visual. A resolução ideal depende do tipo de informação que precisamos extrair da imagem. Quando possível, é recomendado adquirir e armazenar as imagens com a maior resolução disponível. No entanto, em alguns casos, reduzir a resolução pode ser vantajoso, pois além de aumentar a eficiência e reduzir os custos de armazenamento e processamento, também pode facilitar a análise, evitando o excesso de detalhes desnecessários (Faria, 2010).

Há várias restrições quanto aos dados caracterizados por imagens. Isto se deve à capacidade de capturar de forma efetiva, as características das imagens em análise (variáveis de entrada).

Segundo Steffens *et al.* (2019), estas restrições ficam em evidência em situações como iluminação inadequada, superfícies reflexivas e altos contrastes. É comum encontrar imagens digitais nas quais partes do conteúdo não estão visíveis, o que pode dificultar a análise e interpretação dos dados.

As redes neurais convolucionais (CNN) é uma metodologia mais utilizada para o treinamento de modelos de predição, restauração e validação de imagens. O uso de CNNs requer amostras bem delineadas, nas quais cada entrada está associada a uma saída esperada, permitindo que o modelo aprenda a reconhecer padrões e características das imagens (Steffens *et al.*,2019).

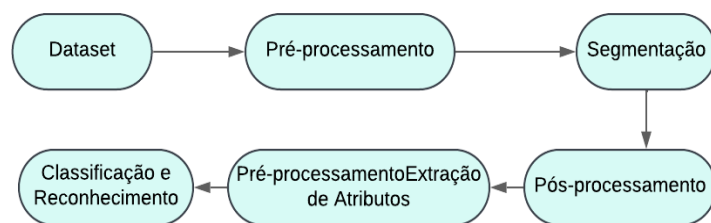
Entre as abordagens inovadoras na análise de dados de imagens, destacam-se as Redes Geradoras Adversariais (GANs), conforme descrito por Goodfellow *et al.* (2014). As GANs utilizam duas redes neurais em um arranjo gerador-discriminador: a rede discriminadora aprende a classificar dados como reais ou falsos, enquanto a rede geradora se concentra em criar dados que pareçam reais. Essa interação entre as duas redes permite um aprendizado conjunto, mas o processo é computacionalmente intensivo, já que requer o treinamento simultâneo de ambas as redes.

Compreender conceitos estatísticos é essencial para a análise exploratória de imagens. Esses conceitos não apenas ajudam a aprofundar nosso entendimento sobre a estrutura do sistema visual humano, mas também promovem o desenvolvimento de técnicas de representação e codificação mais eficientes. Isso possibilita uma verificação mais precisa e uma análise mais rica do que as imagens podem revelar, mesmo quando partes do conteúdo não estão claramente visíveis. A aplicação dessas técnicas permite melhorar a qualidade das análises, aumentando a capacidade de extrair informações relevantes das imagens, mesmo em condições adversas (Steffens *et al.*,2019).

Na Figura 2 é apresentado um fluxo de trabalho típico para processamento e análise de dados em projetos de aprendizado de máquina ou visão computacional envolve várias etapas interconectadas: inicia-se com o *Dataset*, que compreende o conjunto de dados coletados, como imagens ou vídeos; segue-se o *pré-processamento*, onde os dados são preparados por meio de

técnicas como normalização, remoção de ruído e transformações para melhorar sua qualidade; a etapa de *segmentação* divide as imagens em partes significativas, separando objetos de interesse do fundo; no pós-processamento, os resultados da segmentação são refinados, ajustando bordas ou eliminando artefatos; na *extração de atributos*, características específicas, como padrões, texturas ou cores, são identificadas para uso no próximo estágio; finalmente, ocorre a classificação e reconhecimento, onde um modelo de aprendizado de máquina ou algoritmo analisa os atributos para identificar ou classificar os padrões ou objetos.

**Figura 2: Hierarquia de Processamento**



Fonte: Elaborado pelo Autor

## 2.4 APRENDIZADO DE MÁQUINAS

A tecnologia moderna oferece uma vasta quantidade de dados, tanto estruturados quanto não estruturados. O aprendizado de máquina se desenvolveu como um subcampo de inteligência artificial, com o desenvolvimento de algoritmos capazes de aprender a partir de dados. Os modelos identificam padrões e fazem previsões ou decisões com base nos dados. Há na machine learning vários tipos de aprendizagem, uma delas é a aprendizagem supervisionada, não supervisionada e aprendizagem por reforço (Burkov, 2019; Raschka, 2015).

No aprendizado supervisionado, o objetivo é treinar o modelo com dados rotulados, ou seja, o conjunto de dados de treinamento inclui entradas e as respectivas saídas esperadas. Desta forma fazemos com que a máquina aprenda a mapear a entradas para as saídas corretas para generalizar bem para novos dados. Regressão linear, máquinas de vetores de suporte (SVM) e redes neurais, são exemplos de algoritmos supervisionados. No caso de aprendizado de máquinas aplicado ao reconhecimento de imagens, CNNs são eficazes devido à sua capacidade de extrair características visuais complexas. No aprendizado não supervisionado, os dados não têm rótulos ou classificações pré-definidas. O objetivo é que o modelo explore os dados e encontre padrões ou estruturas escondidas por conta própria. Técnicas como agrupamento (*clustering*), que separa os dados em grupos com características semelhante e redução de

dimensionalidade, que simplifica os dados mantendo as informações mais importantes. Essas técnicas são comumente usadas para descobrir estruturas escondidas. No aprendizado por reforço, temos um agente (um programa ou sistema) que aprende a agir dentro de um ambiente específico. Esse agente toma decisões e, com base nessas decisões, recebe recompensas (se as decisões forem boas) ou penalidades (se as decisões forem ruins). O objetivo do agente é aprender com essas experiências e melhorar suas decisões ao longo do tempo, sempre buscando maximizar a quantidade total de recompensas. Esse tipo de aprendizado é especialmente útil em situações em que as decisões precisam ser tomadas em sequência, como em robótica, onde um robô aprende a se mover, ou em jogos, onde o objetivo é vencer o adversário (Burkov, 2019; Raschka, 2015).

A aprendizagem de máquina tornou-se mais útil à medida que a quantidade de informação aumentou.

#### **2.4.1 APRENDIZADO SUPERVISIONADO**

A aprendizagem de máquina supervisionada é o tipo de aprendizado mais usada na prática. O processo começa com a escolha de um conjunto de dados rotulado, que é uma coleção de pares formados por uma entrada e sua respectiva saída. A entrada pode ser qualquer tipo de dado, como um e-mail, uma imagem ou medições de sensores. As saídas podem ser números reais, etiquetas ou, em alguns casos, até mesmo vetores. Uma característica importante desse tipo de aprendizado é que alguns algoritmos requerem a transformação de etiquetas em números, como 0 e 1, para representar os rótulos de maneira adequada (Korbut, 2017).

Treinar um modelo que aprenda a mapear as entradas e saídas corretas é o desafio do aprendizado supervisionado. A partir deste processo, o modelo pode generalizar bem para dados não vistos, fazendo previsões precisas. Neste tipo de aprendizado são utilizados algoritmos como regressão linear, árvores de decisão, máquinas de vetores de suporte (SVM) e redes neurais, sendo estes os mais comuns neste tipo de aprendizado (Korbut, 2017).

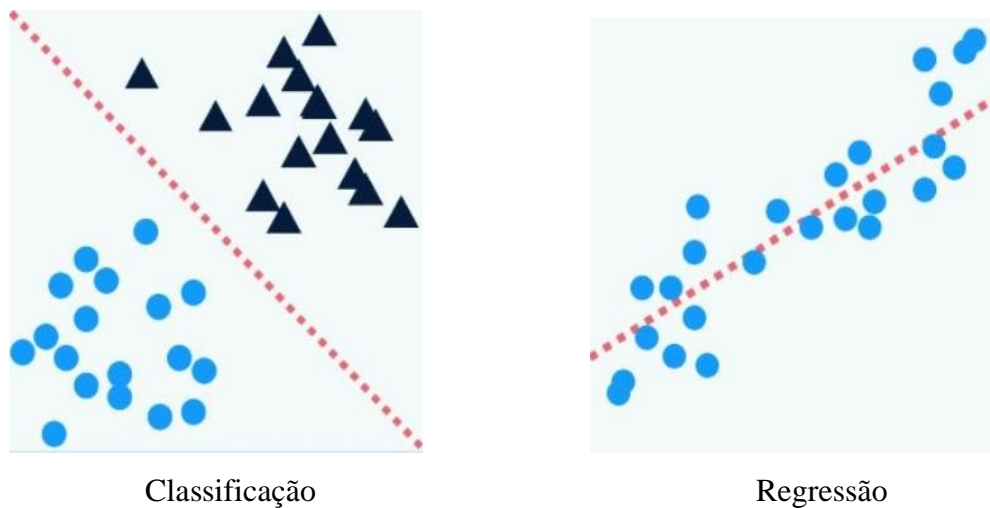
As CNNs, frequentemente usadas para reconhecimento de imagens, o modelo aprende a extrair automaticamente as características mais importantes de uma imagem e faz uma previsão sobre qual classe ela pertence (Korbut, 2017).

O *Transfer Learning* é uma abordagem eficiente no aprendizado profundo que utiliza modelos já treinados em grandes bases de dados, como o ImageNet, para executar novas tarefas de forma eficaz. Modelos que utilizam VGG16, ResNet e MobileNet, usam o *transfer learning*. Estas três redes neurais citadas anteriormente são importantes para o aprendizado profundo.

Usando a técnica de transferência, cada um com suas respectivas arquiteturas, previamente treinadas em *Datasets* extensos como o *ImageNet*, são ferramentas valiosas para treinar modelos complexos do zero. Deste modo maximizam o aproveitamento dos modelos, mas permitem que aplicações sejam desenvolvidas de forma eficiente, econômica e acessível (He *et al*, 2016).

A Figura 3, temos a representação de modelos de classificação e regressão.

**Figura 3: Modelos de classificação e regressão.**



Fonte: Adaptado de (KORBUT, 2017)

## 2.4.2 REDES NEURAIS ARTIFICIAIS

Uma área importante do aprendizado de máquina são as redes neurais artificiais, que se inspiram no funcionamento do cérebro humano e em seus neurônios. Essas redes são capazes de aprender e reconhecer padrões de forma eficiente, adquirindo conhecimento por meio da experiência. As redes neurais são compostas por unidades de processamento conectadas, que recebem e enviam sinais, simulando o funcionamento dos neurônios. Elas incluem camadas de entrada e saída, além de pelo menos uma camada intermediária. Em geral, mais camadas melhoram a capacidade de aprendizado, pois permitem representar de forma mais eficaz as relações entre os dados de entrada e saída. Com poucas camadas, o algoritmo pode falhar em capturar relações não-lineares. A presença de várias camadas intermediárias amplia a capacidade de representação do modelo, permitindo que ele aprenda qualquer função. Redes com múltiplas camadas são conhecidas como redes neurais profundas (*Deep Learning*). Elas

são ferramentas poderosas para identificar padrões complexos que seriam difíceis para um programador humano ensinar à máquina (Rosa, 2019).

### 2.4.3 APRENDIZADO PROFUNDO

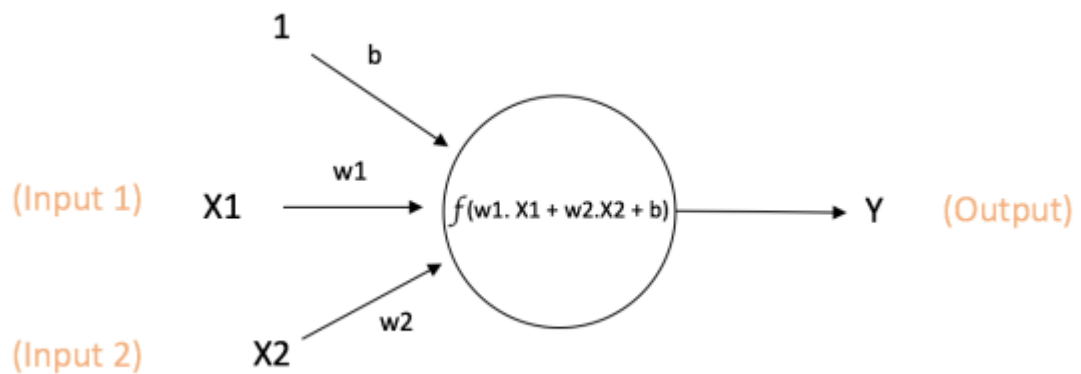
O aprendizado profundo é um tipo de aprendizado de máquina que faz uso de redes neurais artificiais. O termo em inglês é *Deep Learning*. Neste tipo de aprendizado o computador aprende a partir de suas experiências, com compreensão de níveis diferentes de complexidade. O aprendizado é organizado em camadas, de forma que conceitos mais simples possam ser usados para formar ideias mais complexas (Goodfellow *et al.*, 2016). Simplificadamente, imitam o funcionamento do cérebro humano. Assim como os neurônios, que se conectam e trocam sinais, as redes neurais consistem em “nós” que recebem e enviam informações. Desta forma os computadores reconhecem padrões e melhoram sua capacidade de tomar decisões com o tempo, baseados na experiência. O aprendizado é gradativo e o conhecimento acumulado a partir dos conceitos mais básicos até entender os conceitos mais complicados, de forma parecida com o cérebro humano (Goodfellow *et al.*, 2016).

O neurônio artificial é a unidade básica de processamento de uma Rede Neural Artificial. Os sinais de entrada para ele, por convenção, são chamados de entradas de um neurônio. Um único neurônio também é conhecido como nó ou unidade. Ele recebe uma ou mais entradas de outros nós ou de uma fonte externa e calcula uma saída. Cada entrada possui pesos, determinados pela importância relativa dessa entrada em relação às outras. Após serem multiplicados pelos pesos de conexão, esses valores são somados e então passam por uma função de ativação, que gera a saída desse neurônio. A função de ativação permite ao modelo aprender padrões complexos ao introduzir não-linearidade. A função de transferência mais comum é a sigmoide, mas outras incluem ReLU (*Rectified Linear Unit*) e tanh (Tangente Hiperbólica) (Walkarn, 2016).

A Figura 4 representa um modelo simples de neurônio artificial descrita inclusive por Walkarn, (2019). O neurônio recebe duas entradas,  $x_1$  e  $x_2$ , valores fornecidos externamente ao neurônio para processamento. Cada entrada está associada a um peso ( $w_1$  e  $w_2$ ), que indica a importância relativa dessa entrada, onde pesos maiores tornam a entrada correspondente mais influente no resultado. O termo  $b$  representa o viés, que permite ao neurônio ajustar sua função de ativação e controlar sua saída independentemente das entradas. Dentro do neurônio, ocorre uma combinação linear: cada entrada é multiplicada por seu peso correspondente, somada e acrescida do viés ( $w_1x_1 + w_2x_2 + b$ ).

Esse valor resultante passa então pela função de ativação  $f$ , que gera a resposta — a saída do neurônio,  $Y$ . A saída pode ser passada para a próxima camada da rede ou considerada como o resultado, dependendo da arquitetura do modelo.

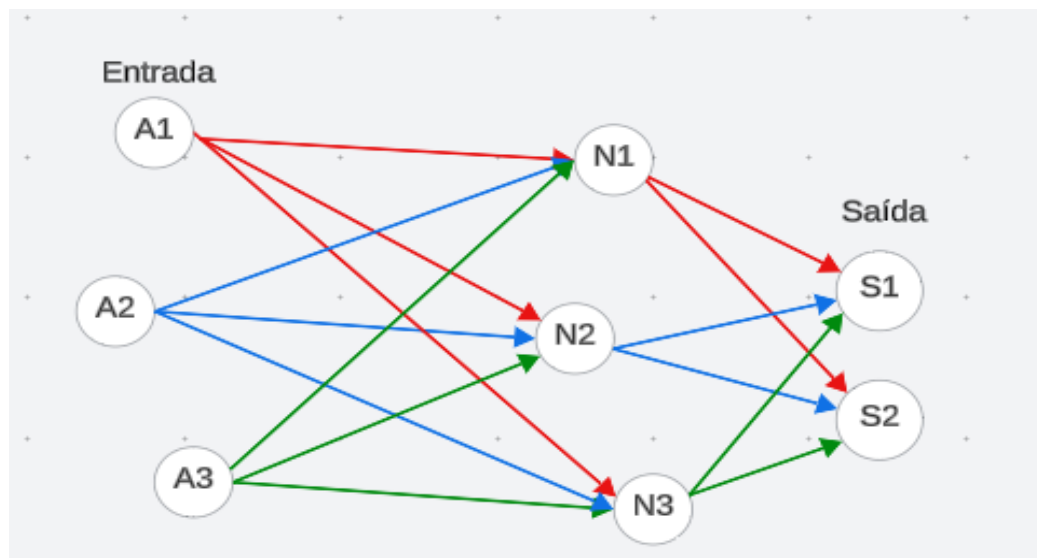
Figura 4: Único neurônio.



Fonte: [ujjwalkarn.me](http://ujjwalkarn.me) (2016)

Na figura 5, tem-se uma representação de Rede Neural Artificial (RNA), mais especificamente uma Rede Neural Multicamadas ou Perceptron Multicamadas (MLP). Nesta representação temos algumas entradas, sendo que cada entrada representa uma característica (ou feature) dos dados de entrada. No caso de classificação de imagens, estes neurônios podem representar pixels ou características extraídas da imagem. Eles apenas recebem os dados e repassam para a próxima camada, porém não realizam nenhuma transformação.

**Figura 5: Exemplos de Redes Neurais Rasas.**

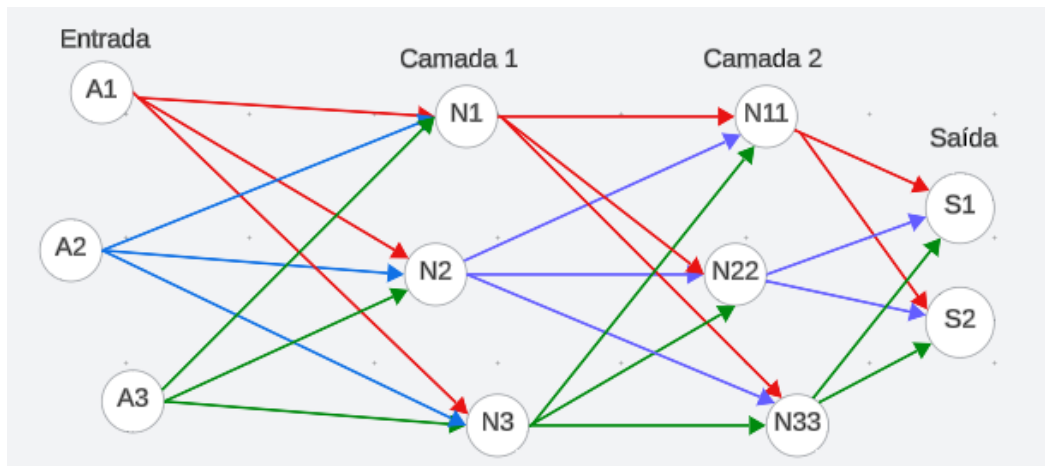


Fonte: Elaborado pelo Autor

A figura 6, a estrutura geral da rede neural apresentada inclui neurônios de entrada ( $A1, A2$  e  $A3$ ), que recebem os dados do sistema; uma primeira camada oculta (Camada 1), composta pelos neurônios  $N1, N2$  e  $N3$ , que processam as informações recebidas da entrada; uma segunda camada oculta (Camada 2), formada pelos neurônios  $N11, N22$  e  $N33$ , que processam os dados da Camada 1 e os encaminham para a camada de saída; e a camada de saída, representada pelos neurônios  $S1$  e  $S2$ , responsáveis por gerar os resultados finais da rede. As conexões entre camadas consecutivas são representadas por setas com pesos ajustáveis, que podem variar conforme o treinamento, e as cores das setas (vermelho, azul e verde) podem simbolizar diferentes valores de peso ou tipos de informação. O fluxo de informações ocorre da esquerda para a direita: os dados entram pela camada de entrada, passam pelas camadas ocultas e chegam à saída. Esta rede, em comparação à anterior, apresenta uma camada oculta adicional, tornando-a mais profunda e capaz de aprender padrões mais complexos, sendo ideal para aplicações como reconhecimento de imagens, processamento de linguagem natural e previsão de séries temporais, embora requiera mais dados para evitar *Overfitting*.



**Figura 6: Exemplos de Redes Neurais Profundas**



Fonte: Elaborado pelo Autor

A camada central da rede, ou seja, a camada intermediária ou ainda chamada camada oculta, realiza as operações de dados. Estes neurônios combinam as entradas da camada anterior com pesos específicos e *biases* ou vieses (valores adicionais usados para ajustar saídas das operações de ativação), e aplicam uma função de ativação (ReLU). O maior objetivo desta camada é extrair padrões ou representações úteis dos dados de entrada, sendo essencial para resolver problemas complexos que não poderiam ser resolvidos com uma camada e sem camadas ocultas (Goodfellow *et al.*, 2016; Haykin, 2001).

A camada de saída é a última camada da rede onde são gerados os resultados. O número de neurônios nesta camada dependerá do tipo de problema. Em classificação binária, um único neurônio é o suficiente, com função de ativação sigmoide. Porém em classificação multi-classe, o número de neurônios é igual ao número de classes e é comum utilizar a função de ativação *softmax*, sendo que este tipo de ativação transforma os valores dos neurônios de saída em uma distribuição de probabilidade, somando a 1, facilitando deste modo a interpretação dos resultados como probabilidades para cada classe (Goodfellow *et al.*, 2016; Haykin, 2001).

É importante salientar que a função de ativação ReLU é usada nas camadas intermediárias (ocultas), pois transforma os valores negativos em zero e deixa os positivos inalterados, possibilitando a rede aprender representações não lineares e a manter o gradiente estável durante o treinamento. No caso da camada de saída para classificação, o ReLU não oferece probabilidades, nem garante que a soma das saídas será 1. Também não oferece uma interpretação clara das saídas como pertencentes a classes específicas. Os valores caso o ReLU seja aplicado nas camadas de saída seriam arbitrários, não relacionados a probabilidades, dificultando a interpretação como previsões de classe. Ele é amplamente usado em

representações complexas, em especial redes profundas, permitindo uma propagação eficiente do gradiente, ajudando a evitar o desvanecimento do gradiente (Goodfellow *et al.*, 2016; Haykin, 2001).

#### 2.4.4 REDES NEURAIAS CONVOLUCIONAIS

As Redes Neurais Convolucionais (CNNs, do inglês *Convolutional Neural Network*) surgiram como uma solução eficaz para lidar com os desafios associados ao treinamento de modelos de aprendizado profundo, especialmente quando se trata de dados de alta dimensionalidade, como imagens. Entre os algoritmos mais utilizados na literatura para reconhecimento e classificação de imagens, as CNNs se destacam como a escolha mais comum. Elas pertencem à subclasse das Redes Neurais Profundas, e sua principal diferença em relação a redes neurais tradicionais é a utilização de pelo menos uma camada convolucional. Essas redes são capazes de identificar objetos detectando inicialmente características de baixo nível, como bordas, linhas e curvas, e, em seguida, constroem características mais abstratas por meio de várias camadas convolucionais (Rosa, 2019).

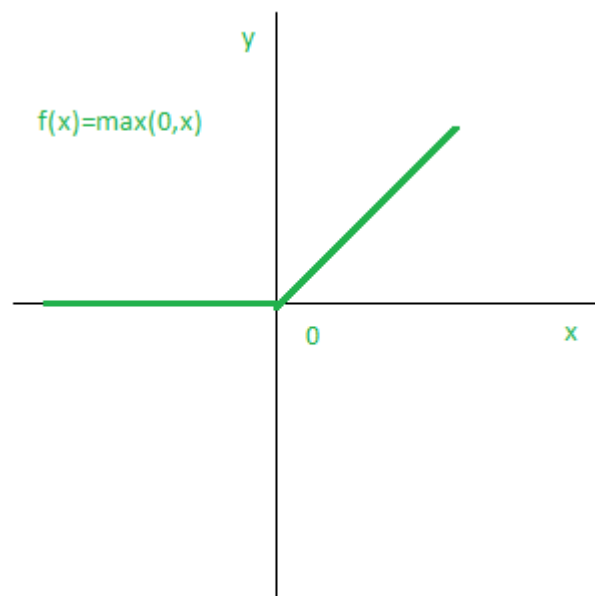
Uma das principais inovações das CNNs é a aplicação de operações de convolução, que permitem que a rede aprenda automaticamente a detectar características relevantes, como bordas, texturas e padrões complexos, sem a necessidade de engenharia manual de recursos. As camadas convolucionais aplicam filtros que são aprendidos durante o treinamento, o que reduz drasticamente a dimensionalidade dos dados e, conseqüentemente, o número de parâmetros necessários em comparação com redes neurais tradicionais, como as MLPs (*Multilayer Perceptrons*). Devido à sua eficácia em tarefas como classificação de imagens, detecção de objetos e segmentação, e por serem projetadas para lidar com a complexidade dos dados visuais, as CNNs têm revolucionado a abordagem de problemas de visão computacional (Burkov, 2019). Essas redes conseguem reconhecer padrões altamente complexos e adaptam-se a distorções e variações nas imagens, tornando-as uma ferramenta poderosa para o campo.

A topologia de uma Rede Neural Convolucional (CNN) segue uma estrutura em camadas. Neste trabalho a CNN foi projetada para processar dados visuais, as imagens de frutos cítricos. A imagem entra pela camada de entrada, sendo processada em várias camadas convolucionais e de *Pooling*, onde as características são extraídas. Após, as informações passam pelas camadas totalmente conectadas, que fazem a classificação final, o que resulta na saída de rede, que representa a classe à qual a imagem pertence. Desta forma as CNNs podem visualizar diferentes níveis de complexidade, desde características simples até representações

abstratas, o que torna essa arquitetura ideal para reconhecimento e classificação de imagens (Pereira, 2021).

O Gráfico 1 tem-se a representação da função de transferência ReLU que significa Unidade Linear Retificada. Ele pega uma entrada de valor real e a limita em zero (substitui valores negativos por zero)

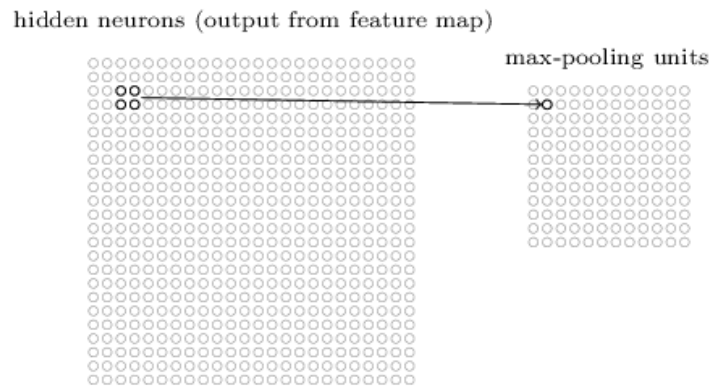
**Gráfico 1: Função de ativação ReLU.**



Fonte: Acervo Lima (2024)

Além das camadas convolucionais, as camadas de agrupamento (*Pooling*), são geralmente usadas após as camadas convolucionais. Uma camada de *Pooling* recebe sua entrada da camada convolucional e é usada para preparar um mapa de características comprimido. Por exemplo, em um mapa de características de uma camada convolucional, cada unidade da camada de *Pooling* pode resumir uma pequena região de 2x2 neurônios vizinhos. Uma forma popular de realizar o *Pooling* é utilizando o valor máximo na região de *Pooling*; isso é conhecido como *Max-Pooling*. Essa camada, a unidade de *Pooling* simplesmente gera o valor máximo das ativações na região de entrada 2x2 (Goodfellow *et al*, 2016).

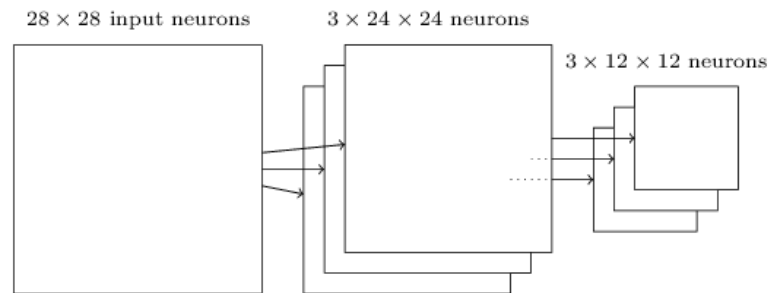
**Figura 7: Exemplo de redução de dimensão com *Max-Pooling* (24x24 para 12x12 neurônios)**



Fonte: Data Science Academy (2022)

Na Figura 7 tem-se a representação de 24x24 neurônios emitidos da camada convolucional, e que após agrupamento, temos 12x12 neurônios. Importante salientar, que a camada convolucional envolve mais que um único mapa de características, com a aplicação do *Max-Pooling* para cada mapa de recursos separadamente. A figura 8 apresenta um exemplo de três mapas de recursos, com camadas combinadas, convolucional e *Max-Pooling* para exemplificar.

**Figura 8: Mapa de recurso com camadas convolucionais e *Max-Pooling***

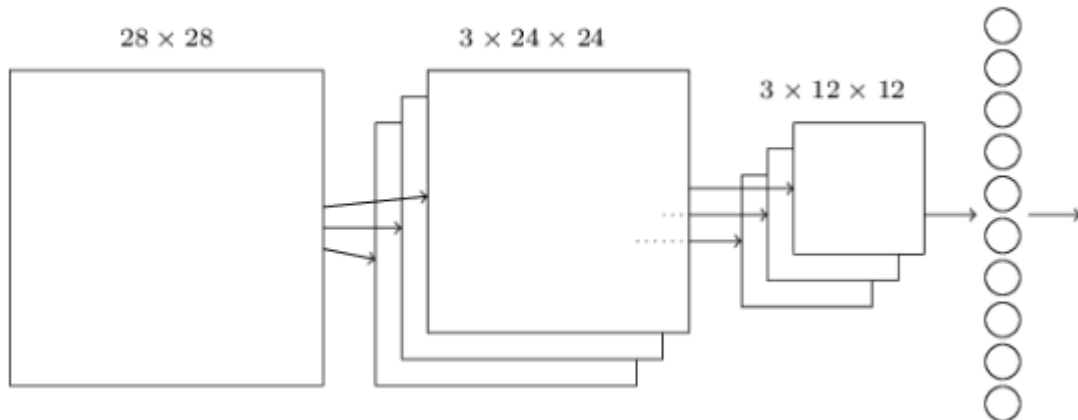


Fonte: Data Science Academy (2022).

Na CNN completa, assim como representada na Figura 8 e 9, começamos com 28x28 neurônios de entrada (pixels), seguida por uma camada convolucional, com um campo receptivo local de 5x5 e três mapas de característica, que resulta numa camada de 3x24x24 neurônios ocultos. Após utilizamos uma camada *Max-Pooling*, aplicadas a região 2x2, em cada um dos três mapas de recursos., resultando numa camada de 3x12x12 neurônios ocultos. A última camada é a camada totalmente conectada com todos os neurônios da camada de *Max-Pooling* e a cada um dos neurônios de saída. uma rede composta por muitas unidades simples, com comportamentos determinados por seus pesos e viés, com objeto de usar dados de

treinamento para treinar os pesos e viés da rede para que ocorra um trabalho eficiente de classificação dos dígitos de entrada (Goodfellow *et al*, 2016).

**Figura 9: Mapa de recurso completo**



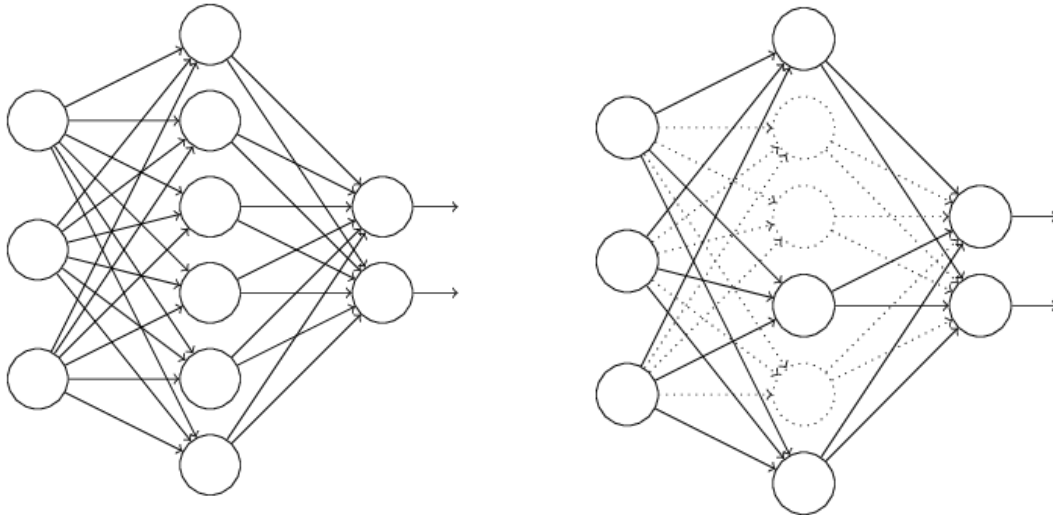
Fonte: Data Science Academy (2022)

O *Dropout* é uma técnica de regularização, que independe da modificação da função de custos, isto porque ele modifica a própria rede. Ele melhora o desempenho das redes neurais, principalmente em redes grandes e profundas, com problemas de *Overfitting* agudo. Em uma rede neural temos uma entrada  $x$  e a saída correspondente  $y$ , normalmente treinada pela propagação direta de  $x$  através da rede e depois retrocede (retropropagação) para determinar a contribuição do erro para o gradiente. Com o *Dropout* isto é modificado. Ele desativa aleatoriamente e temporariamente um número específico de neurônios em cada interação, de acordo com a taxa indicada. Na fase de teste nenhum neurônio é desativado, porém os pesos são ajustados. O processo é repetido, restaurando os neurônios removidos, e escolhendo um novo subconjunto aleatório de neurônios ocultos para excluir, e utilizando pesos e viés diferente na rede (Arivastava *et al*, 2014).

Na Figura 10, o funcionamento do Dropout é representado com conexões pontilhadas, indicando os neurônios desativados durante o processo de desativação temporária em uma rede neural. Esse mecanismo consiste em excluir, de forma aleatória e temporária, determinados neurônios durante as iterações do treinamento, sendo este processo repetido diversas vezes. Após cada iteração, os neurônios previamente desativados são restaurados, e novos subconjuntos de neurônios ocultos são escolhidos para exclusão. Para garantir a robustez e a

eficácia do modelo, pesos e vieses são ajustados ao longo do treinamento, conforme descrito por Arivastava *et al*, (2014).

**Figura 10:** Técnica de regularização com *Dropout*.

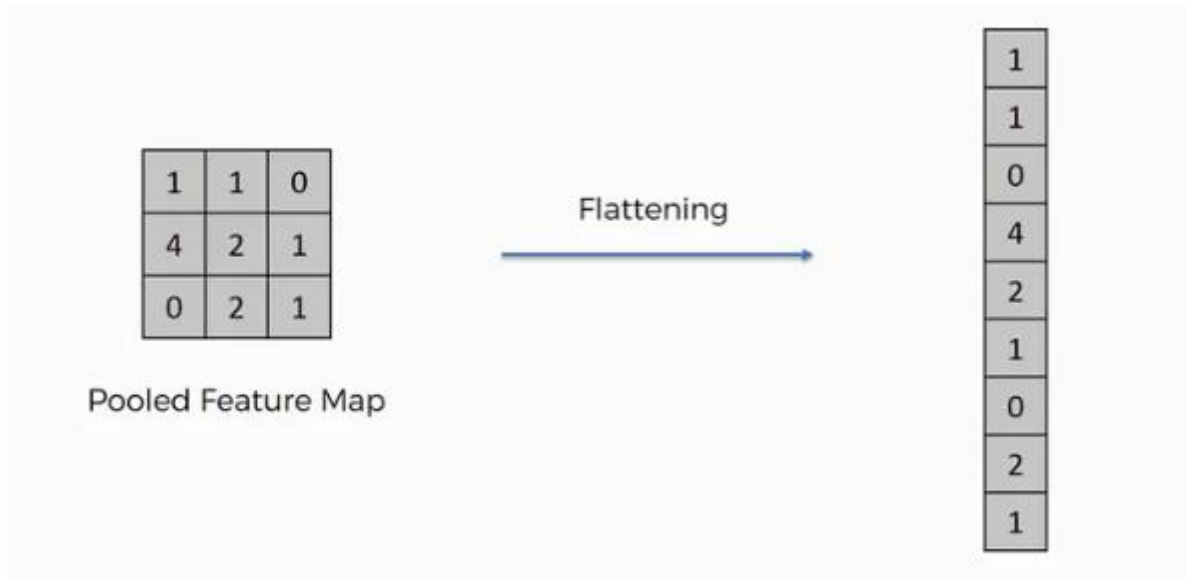


Fonte: Data Science Academy (2022)

Nas redes neurais convolucionais também usamos o *Flatten* ou operação de achatamento, é usada para transformar uma matriz multidimensional em um vetor unidimensional, muito útil quando temos camadas densas. Como as camadas convolucionais trabalham em formato de tensor, normalmente 3D, camadas densas, por outro lado, precisam de entradas em formato de vetor, 1D. Deste modo, a camada *Flatten* reduz a dimensionalidade do tensor para ser usada na camada densa, onde o vetor corresponde cada vetor corresponde a um neurônio nesta camada densa. É importante salientar que o *Flatten* não realiza nenhuma operação de aprendizado. Não reduz a quantidade de informações do modelo, mas muda a organização do tensor (Goodfellow *et al*, 2016).

Na Figura 11, é apresentado um exemplo do processo de *flattening*, onde, à esquerda, encontra-se um mapa de características (Pooled Feature Map) organizado como uma matriz bidimensional  $3 \times 3$ . Durante o processo de *flattening*, os elementos dessa matriz são rearranjados linearmente em um vetor unidimensional, conforme mostrado à direita. Esse vetor reúne todos os valores do mapa de características em uma sequência única, tornando-os prontos para serem processados pelas camadas subsequentes da rede neural, geralmente camadas totalmente conectadas.

**Figura 11:** Exemplo de operação de achatamento - *Flatten*



Fonte: médium.com (2022)

Há também uma arquitetura de rede neural convolucional que tem sido amplamente usada. Desenvolvida pela equipe do *Visual Geometry Group* da Universidade de Oxford, a VGG16 é simples e tem um desempenho excelente em tarefas de classificação de imagens. É uma arquitetura que consiste em 16 camadas treináveis, que incluem 13 camadas convolucionais e 3 camadas totalmente conectadas, além de camadas de *Pooling*. Os filtros convolucionais são pequenos 3x3, aplicados sequencialmente, com funções de ativação ReLU após cada camada. Esta arquitetura é relativamente pesada. Apresenta em torno de 138 milhões de parâmetros, exigindo alto poder computacional e memória para treinar. Por ser treinado no *Dataset ImageNet*, com milhões de imagens e milhares de classes, o VGG16 é frequentemente usado como base em aplicações de aprendizado por transferência, adaptando-se facilmente a outros domínios. Embora seja uma rede que demanda alto poder computacional, e deste modo computacionalmente cara, possui um design simples, com estrutura bem padronizada. Além disso tem facilidade de reutilização em novos problemas usando *Transfer Learning* (Simonyan, *et al*, 2015).

O ResNet (Residual Network) também é uma arquitetura de rede neural convolucional (CNN), amplamente conhecida por revolucionar a forma como redes profundas são treinadas. Usada na classificação de imagens e detecção de objetos. Essa arquitetura implementa blocos

residuais, que utilizam conexões de atalho para facilitar o fluxo de gradientes, mitigando o problema da degradação de desempenho em redes muito profundas. Quando comparado à arquitetura VGG, o ResNet utiliza significativamente menos parâmetros, tornando-o uma escolha mais eficiente e robusta para *Transfer Learning* e outras aplicações em aprendizado profundo (He *et al*, 2015).

Outra arquitetura de rede neural utilizada é a *MobileNet*, desenvolvida pela Google. É uma rede neural convolucional eficiente e compacta, primeiramente pensada para dispositivos móveis e sistemas com recursos computacionais limitados. Esta rede introduz camadas convolucionais separadas em profundidade o que faz reduzir significativamente o número de parâmetros e cálculos, deste modo mantém um bom desempenho na classificação de imagens. Possui variantes como *MobileNetV1*, *V2* e *V3*, sendo que cada uma dela possui progressivas melhorias. *MobileNetV1* é uma rede neural que se baseia em convoluções separáveis em profundidade, reduzindo custos computacionais sem comprometer excessivamente a precisão, enquanto o *MobileNetV2*, inclui os *Inverted Residuals* blocos de convolução que combinam expansão, uma camada de projeção e conexões de atalho para aumentar a eficiência e as capacidades de aprendizado, e por fim, o *MobileNetV3* que mescla técnicas de busca automatizada de arquitetura neural (NAS) com refinamentos manuais, como o uso de ativadores avançados, incluindo o h-swish, alcançando um equilíbrio ideal entre precisão e eficiência (Sawdler *et al*, 2018; Howard *et al*, 2019).

#### 2.4.5 TREINAMENTO

O treinamento no aprendizado profundo é um processo de ajustar uma rede neural que melhora suas previsões, utilizando a função de perda como um guia e o otimizador como uma ferramenta para reduzir os erros ao longo do tempo. O modelo aprende a fazer previsões mais precisas, baseado nos dados fornecidos. Com a definição de um problema, precisamos fazer ajuste nos parâmetros (pesos) para que suas previsões se aproximem das respostas corretas. Para averiguar como os dados fornecidos estão sendo modelados usamos a função de perda. Desta forma averígua-se o quão longe as previsões do modelo estão das respostas corretas. Para reduzir o erro ou a perda usa-se um otimizador, que permite fazer ajuste dos parâmetros. O otimizador modifica os pesos das conexões entre os “neurônios” da rede, com base no valor da função de perda. Cada ciclo do aprendizado, denominado época, ele faz pequenos ajustes nos pesos, com o objetivo de minimizar os erros. O treino acaba quando o modelo chega a um ponto em que não consegue mais reduzir significativamente a perda, ou seja, quando ele atinge uma



convergência. Nesse ponto, o modelo fez o melhor ajuste possível para os dados fornecidos. Após o treinamento, é comum avaliar o desempenho da rede usando dados que antes não foram vistos, garantindo nesta forma que o modelo tenha aprendido a generalizar e não apenas memorizar os exemplos do conjunto de treino. O treinamento é considerado bem-sucedido se o modelo tiver uma boa performance (Rosa, 2019; Goodfellow et al., 2016).

A acurácia é uma métrica amplamente utilizada para avaliação de desempenho, muito utilizada em tarefas de classificação. A acurácia pode ser enganosa, embora seja fácil sua interpretação. Isto se deve ao desbalanceamento de classes, onde a proporção das classes pode mascarar erros do modelo (FERRI *et al*, 2009). Outras métricas são utilizadas em conjunto com a acurácia, fornecendo uma visão mais abrangente do desempenho.

No desenvolvimento dos modelos, a técnica de validação cruzada *k-Fold*, avalia a capacidade de generalização de um modelo. O *Dataset* é dividido em  $k$  partes de tamanhos aproximadamente iguais (*folds*) e o modelo treinado em  $k - 1$  *folds* enquanto o *fold* restante é usado para validação. Este processo é repetido  $k$  vezes, com alternância dos *folds* de validação, sendo as métricas finais obtidas como a média dos resultados de cada interação. Oferece uma robustez e um melhor desempenho do modelo (ARLOT *et al*, 2010).

A utilização de validação cruzada no modelo é para garantir uma avaliação mais robusta. A técnica de validação cruzada usada é com 10 dobras (*10-fold Cross-Validation*). Esta técnica é confiável, minimizando os efeitos da variabilidade nos dados de treinamento e teste. A validação consiste em dividir o conjunto de dados em 10 subconjuntos de tamanho aproximadamente igual. O modelo é treinado 10 vezes, cada vez utilizando um subconjunto diferente como dados de teste e os outros 9 como dados de treinamento. A média dos resultados dessas 10 iterações fornece uma estimativa mais robusta do desempenho do modelo em dados não vistos.

### 3 TRABALHOS RELACIONADOS

Foram encontrados trabalhos durante a pesquisa de fundamentação teórica que se assemelham em alguns requisitos ao trabalho proposto neste TCC. Na seção 3.1 está descrito o trabalho de Rosa (2019), que utilizou aprendizado de máquina para fazer classificação de imagens de várias frutas, como maçãs vermelhas, banana, melão, manga e mamão. Na seção 3.2, o trabalho de Santiago *et al.* (2019) que focou na classificação de imagens de laranjas. Por fim, na seção 3.3 está descrito o trabalho de Pereira (2021) que fez uso de visão computacional para reconhecer imagens de frutas em imagens RGB.

Rosa (2019) desenvolveu seu trabalho utilizando aprendizado de máquinas para classificar imagens de frutas utilizando o algoritmo SVM. Num primeiro momento foram extraídos manualmente as características como cor e textura. Após foi utilizado neste trabalho CNNs para classificar as imagens de frutas, usando VGGNet e Inception, para realizar esta tarefa de classificação. O trabalho apresentou também técnicas de pré-processamento de imagens para melhorar a robutez do modelo. Para tal usou data augmentation para aumentar os dados. O objetivo do trabalho foi atingir uma acurácia de 90%. O conjunto de dados utilizado foi o *Fruit Dataset*, que possui 20 classes de frutas. Os dados foram divididos em treino e teste. Dois experimentos foram realizados, sendo que no primeiro foi utilizado o *Dataset* original, treinado com 50 épocas com 1.244 imagens e neste primeiro experimento a acurácia foi de 96,66%. Num segundo momento o experimento 2 teve o *Dataset* ampliado com a função *augmentation*, o que gerou 29.603 imagens para o treinamento por 150 épocas e a acurácia final foi de 98,06%. O trabalho teve resultados satisfatórios nos dois experimentos e demonstrou que as CNNs são algoritmos eficazes para a classificação de frutas por imagens RGB. Importante salientar que o aumento do número de imagens foi um fator determinante para o aumento da acurácia do segundo experimento. Neste trabalho ainda foi proposto que houvesse a ampliação do *Dataset*, com a inclusão de imagens de outras frutas e vegetais.

Santiago (2019) em seu trabalho, desenvolveu e testou um algoritmo baseado em técnicas de processamento de imagens e aprendizado de máquina para classificar laranjas baseado no seu calibre. Testou 204 imagens de laranjas da variedade Pêra Rio. Estas imagens foram classificadas em pequena, média e grande e o calibre foi aferido a partir do diâmetro equatorial da fruta, sendo que obtidos através do paquímetro. Os ruídos foram removidos e as áreas segmentadas, sendo o método utilizado a limiarização de Otsu para destacar a fruta na imagem (pré-processamento). Neste trabalho Santiago testou dois algoritmos, um de Análise

Discriminante Linear (LDA) e  $k$ -vizinhos mais próximos ( $k$ -NN). Nos testes a acurácia atingida foi de 82% usando o LDA e a acurácia usando o  $k$ -NN foi de 72,13%. Deste modo ficou evidente que o algoritmo LDA obteve melhores resultados, principalmente nas classes onde tinha imagens de laranjas classificadas como pequenas e grandes. Contudo, houve dificuldade de o algoritmo classificar as laranjas de tamanho médio. Embora precise de aprimoramento devido a pouca quantidade de imagens entre outras melhorias como de iluminação controlada, especialmente para identificar as laranjas de tamanho médio, a metodologia tem viabilidade técnica para ser aplicada em projetos de máquinas automatizadas para classificar laranjas por tamanho.

Neste trabalho Pereira (2021), teve como objetivo principal classificar frutas como bananas, maçãs vermelhas e laranjas em imagens RGB. Para isto desenvolveu um modelo baseado em CNNs,  $z$  para gerar um modelo de classificação. Além disso foi criada uma base de dados com estas imagens e o modelo teve seu desempenho avaliado com base nos acertos e erros. O modelo escolhido foi devido a sua alta capacidade de lidar com problemas de reconhecimento de padrões em imagens. O *Dataset* utilizado foi baseado no trabalho de Kuang (2015), contendo imagens de frutas (bananas, laranjas e maçãs), com um total de 1.244 imagens no primeiro experimento e 29.603 imagens no segundo. Os dados foram divididos em treino, teste e validação. O modelo construído teve várias camadas convolucionais e de *Pooling*, no qual foram necessárias para extrair características importantes das imagens. Inclusive foi utilizado na arquitetura do modelo as camadas de *Dropout* para evitar *Overfitting* e camadas totalmente desconectadas para realizar a classificação.

### **3.1 PREPARAÇÃO DE EXPERIMENTOS BASEADOS NO ESTADO DA ARTE**

Foram realizadas comparações com um trabalho referência (Pereira, 2021) com proposta semelhante do uso de CNNs para o reconhecimento de imagens. Várias técnicas de pré-processamento e modelagem de treino de modelos foram aplicadas no estudo atual. A estrutura da rede neural convolucional consistiu em três camadas de convolução, todas ativadas pela função ReLU para destacar características relevantes, inter-relacionadas por três camadas de *Pooling*, que encolhem a imagem e com isso reduzem sua dimensionalidade. Para cortar o sobre ajuste, foi acrescentada uma camada de *Dropout*, e uma camada de *Flatten* foi usada para converter uma matriz de imagem de 2D para 1D. O modelo possuía ainda duas camadas densas:

uma com função de ativação ReLU e outra com *Softmax*, que servia para a classificação final.

Na compilação e otimização, a rede foi configurada com a função de perda *Sparse Categorical Crossentropy* e o otimizador Adam, reconhecido por sua eficiência em ajustes adaptativos. A métrica de avaliação utilizada foi a acurácia.

Pereira (2021) testou dois cenários principais em seu experimento. No primeiro momento o modelo treinado tinha um tamanho de lote de 256, ao longo de 50 épocas, utilizando um conjunto de dados composto por 1.244 imagens. No segundo treinamento o tamanho de lote maior, de 512, com 150 épocas de treinamento, empregando um conjunto de dados com 29.603 imagens.

## 4 METODOLOGIA

Este capítulo apresenta as etapas realizadas no desenvolvimento desta pesquisa. A estrutura foi dividida em duas seções principais. A Seção 4.1 aborda o *Dataset* e como ele foi dividido para o treinamento. A Seção 4.2 pré-processamento, onde foram descritos a configuração dos modelos treinados, assim como a avaliação do modelo.

### 4.1 *Dataset*

O conjunto de dados utilizado nesta pesquisa é o conjunto de dados de *Orange Diseases Dataset*. O conjunto de dados foi criado para aplicação de modelos de redes neurais convolucionais no aprendizado automático da classificação de doenças utilizando a rede neural convolucional. O conjunto de dados compreende imagens de laranja com diferentes condições de saúde. As imagens são particionadas em três divisões distintas que são, treino, teste e validação que compreendem 70%, 15% e 15%, respectivamente. As laranjas nos dados de imagem estão infectadas com mancha preta, apresentados na figura 12; câncer cítrico, apresentados na figura 13; *greening*, apresentados na figura 14; e laranjas saudáveis, apresentadas na figura 15.

**Figura 12:** Exemplo de laranjas com mancha-preta no *Dataset*.



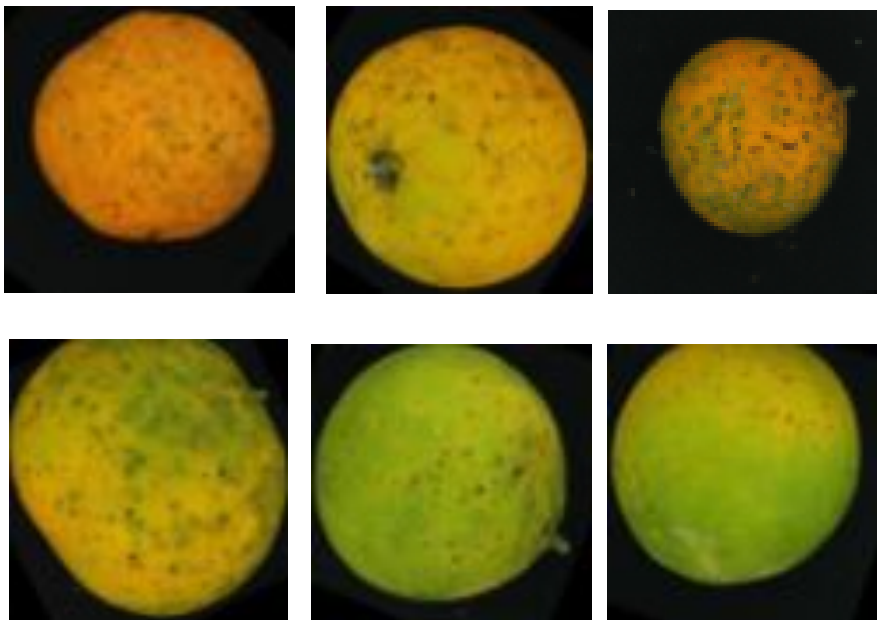
Fonte: (Cristovão, 2024).

**Figura 13:** Exemplo de laranjas com cancro cítrico no *Dataset*.



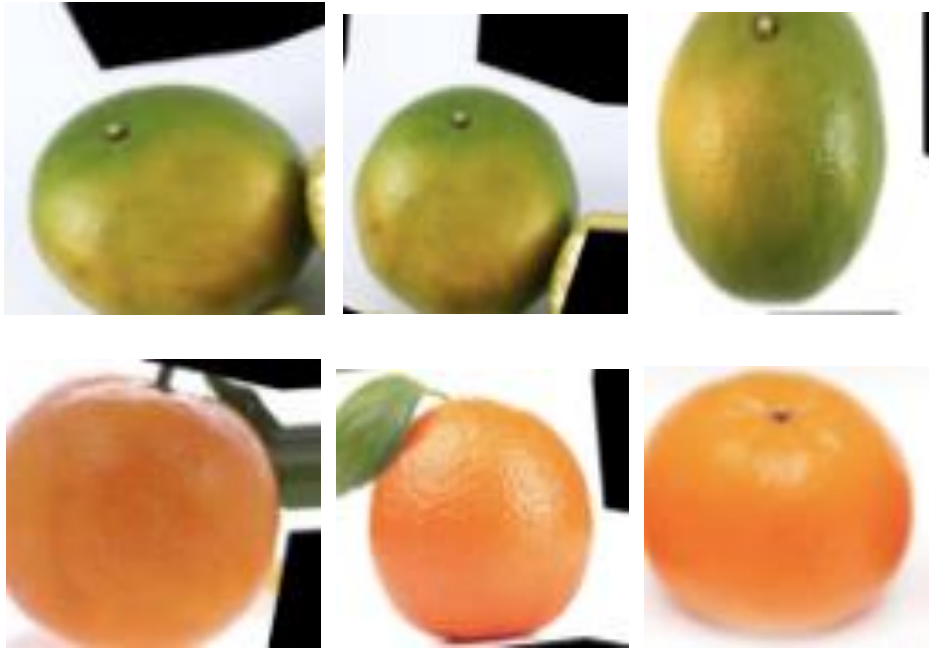
Fonte: (Cristovão, 2024).

**Figura 14:** Exemplo de laranjas com *greening* no *Dataset*.



Fonte: (Cristovão, 2024).

**Figura 15:** Exemplo de laranjas saudáveis no *Dataset*.



Fonte: (Cristovão, 2024).

O conjunto de dados estão divididos em três subconjuntos: Treinamento (70%) Validação (15%) e Teste (15%), porém todos foram anexados em um único *Dataset* pois o método de inferência de modelo adotado foi o *k-fold Validation* que elimina essa necessidade de separação de arquivos.

Pelo fato de o conjunto de dados apresentar um número limitado de exemplos para o treinamento e validação, o *Dataset* utilizou de técnicas de enriquecimento de base, de forma artificial, ampliado de 1090 imagens para 9637 imagens, garantindo um treinamento robusto e uma avaliação precisa dos modelos.

## 4.2 Pré-processamentos

No presente trabalho diferentes configurações de pré-processamento foram utilizadas. Eles mudaram de acordo com os modelos testados, isto devido ao objetivo de explorar e analisar o impacto de diferentes tamanhos de imagens, assim com o tamanho dos lotes (*batch sizes*) e números de épocas no desempenho das redes treinadas. Em função de uma maior capacidade de generalização dos modelos, utilizou-se o *ImageDataGenerator* da biblioteca *Keras*. Como técnica de aumento dos dados de maneira virtual em função da quantidade limitada de imagens no treinamento, aplicou-se o *data augmentation*. Os tamanhos das imagens também variaram de acordo com as características e demandas específicas de cada modelo.

Os modelos tiveram sua performance avaliada com o processo de validação cruzada K-fold, que também auxiliou na redução do risco de *Overfitting*. O *Dataset* foi dividido em 10 subconjuntos, e a cada treinamento dos subconjuntos foi impressa a acurácia e ao final a média foi calculada e impressa.

Nos modelos customizados, adotou-se o *Dataset* original da *Kaggle* com 1090 imagens. Três modelos foram treinados e estão descritos na Tabela 1, apresentada abaixo:

**Tabela 1: Descrição das configurações dos modelos customizados.**

<i>Dataset</i>	Tamanho das imagens	Lote ( <i>Batch size</i> )	Épocas
<b>1.090</b>	150x150	256	150
<b>1.090</b>	32x32	256	100
<b>1.090</b>	64x64	32	50

Fonte: Elaborado pelo Autor.

Nestes modelos customizados tiveram várias camadas de convolução e *Max-Polling*, seguida de camadas totalmente conectadas. Foi aplicada a função ReLU nas camadas ocultas, com função *Softmax* na camada de saída. Nestes modelos não houve a implementação dos *callbacks*.

Os modelos com arquiteturas pré-treinadas estão descritos na Tabela 2, que apresenta:

**Tabela 2: Descrição das configurações dos modelos pré-treinados.**

<i>Dataset</i>	Modelos	Tamanho das imagens	Lote ( <i>Batch size</i> )	Épocas
<b>9637</b>	VGG16	224x224	32	40
<b>9637</b>	ResNet	224x224	32	40
<b>9637</b>	<i>MobileNet</i>	128x128	64	40

Fonte: Elaborado pelo Autor.

Durante o treinamento utilizou-se dois *callbacks* principais para controlar o processo de otimização. O *EarlyStopping*, configurado para monitorar a acurácia de validação, com possibilidade de interrupção se o modelo não apresentasse melhoras após um número específico de épocas, evitando o *Overfitting*. E o *ReduceLROnPlateau* para ajudar na taxa de aprendizado quando a acurácia de validação não melhora após um número específicos de épocas, auxiliando a convergir para um melhor ponto de solução.



Um outro aspecto importante foi pela definição de todos os modelos serão treinados em RGB, visto a complexidade da identificação das lesões nos frutos de laranjas. Na Tabela 3 tem-se uma descrição da relevância da cor para a identificação correta das lesões.

**Tabela 3: Relevância da cor para identificação dos aspectos das lesões.**

<b>Doença</b>	<b>Aspectos das lesões nos frutos</b>	<b>Cor predominante das lesões</b>
<b>Cancro Cítrico</b>	Lesões em formato circular com centro elevado e textura de cortiça. Bordas amareladas	Amarelo nas bordas com marrom/cortiça no centro.
<b>Greening (HLB)</b>	Frutos deformados, com amadurecimento irregular. Parte do fruto pode permanecer verde, enquanto outra amadurece.	Verde intenso nas áreas não maduras e amarelo nas maduras
<b>Mancha Preta</b>	Pontos pretos pequenos e irregulares que podem coalescer, formando manchas maiores com bordas bem definidas	Preto com bordas possivelmente amareladas em frutas maduras

Fonte: Elaborado pelo Autor.

Na Tabela 4 estão todos os modelos utilizados com os parâmetros relevantes para o presente estudo.

**Tabela 4: Configurações dos testes realizados.**

<b>Parâmetros</b>	<b>Dimensões de Entrada</b>	<b>Batch size</b>	<b>Épocas</b>	<b>Imagens</b>
<b>Modelo I (Estado da Arte)</b>	150x150	256	150	1.090
<b>Modelo II</b>	32x32	256	100	1.090
<b>Modelo III</b>	64x64	32	50	1.090
<b>Modelo IV - VGG16</b>	224x224	32	40	9.637
<b>Modelo V – ResNet</b>	224x224	32	40	9.637
<b>Modelo VI – MobileNet</b>	128x128	64	40	9.637

Fonte: Elaborado pelo Autor.

## 5. RESULTADOS

Neste capítulo serão discutidos os resultados encontrados nos seis modelos treinados. Faz-se necessário observar que em cada experimento ocorreram variações com relação aos parâmetros como números de épocas, quantidade de camadas densas e tamanho do lote (*Batch Size*). Neste modo, houve uma análise detalhada dos efeitos de cada ajuste no tempo de treinamento, precisão e otimização para uma implementação prática.

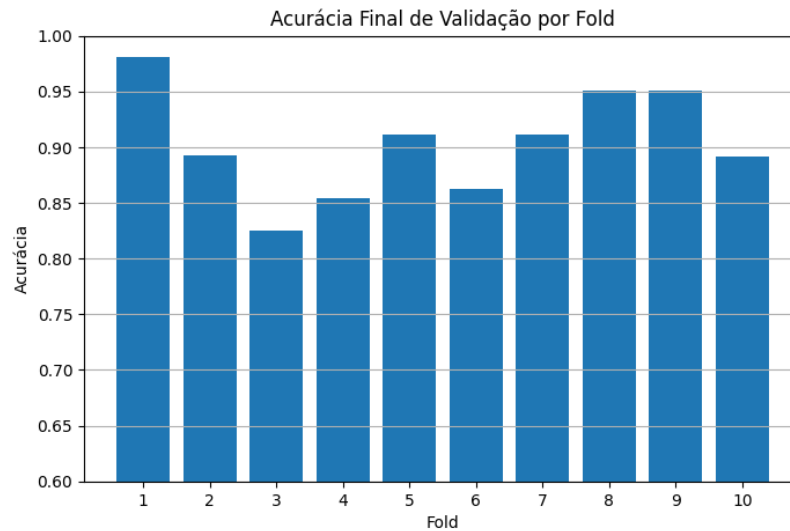
Na Tabela 5, são apresentados os modelos treinados com suas respectivas acurácias.

**Tabela 5: Configurações dos testes realizados e resultado da média da acurácia de cada modelo.**

Parâmetros	Dimensões de Entrada	Batch size	Épocas	Imagens	Acurácia
<b>Modelo I (Estado da Arte)</b>	150x150	256	150	1.090	86.71%
<b>Modelo II</b>	32x32	256	100	1.090	90.23%
<b>Modelo III</b>	64x64	32	50	1.090	90.35%
<b>Modelo IV - VGG16</b>	224x224	32	40	9.637	90.07%
<b>Modelo V - ResNet</b>	224x224	32	40	9.637	90.07%
<b>Modelo VI - MobileNet</b>	128x128	64	40	9.637	90.15%

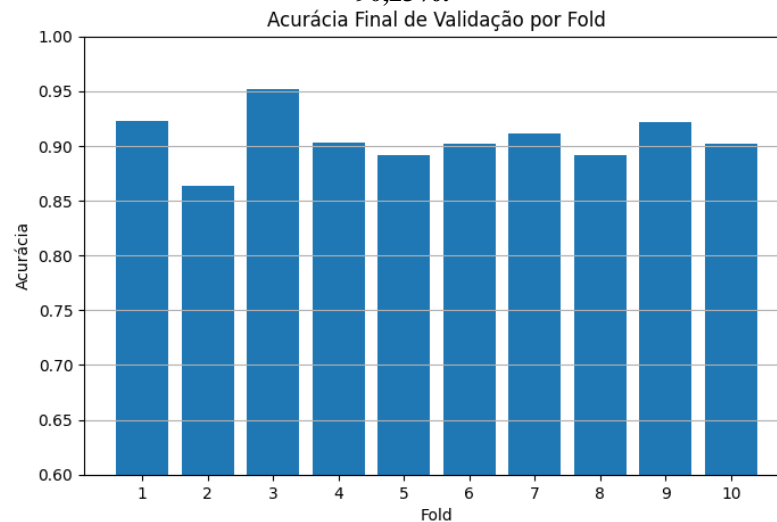
Fonte: Elaborado pelo Autor.

De acordo com os resultados dos treinamentos, pode-se observar que o Modelo I foi reproduzido dimensionalidade das imagens de entrada, número de épocas e quantidade de camadas densas descritas no trabalho de Pereira,2021 e que apesar do modelo ter um número elevado de épocas e de uma resolução intermediária das imagens obteve uma acurácia inferior quando comparada a outros modelos que utilizam o mesmo *Dataset*. Isto indica que o modelo sofreu *Overfitting* ou a otimização não foi adequada para os recursos. No Gráfico 2 pode-se observar as acurácias de cada *fold* deste modelo treinado.

**Gráfico 2: Resultados de Acurácia Modelo I (Estado da arte).**

Fonte: Elaborado pelo Autor.

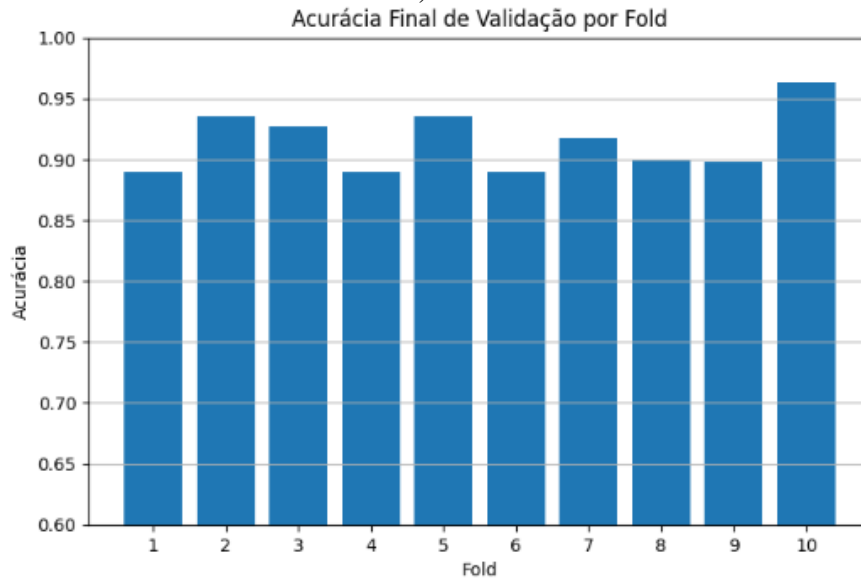
A partir da referência do Modelo I, a opção foi ajustar as dimensões das imagens para 32x32, mantendo o número de lotes e treinar o modelo com 100 épocas. Mesmo reduzindo a dimensionalidade da imagem, a acurácia não ficou comprometida e o tempo de treinamento foi reduzido, ou seja, otimizou custos computacionais. Os ajustes realizados permitiram que o modelo capturasse as características relevantes mesmo com um *Dataset* pequeno e permitiu que a acurácia aumentasse num percentual de 3.52% a acurácia quando comparado com o Modelo I (Estado da Arte). Os resultados da acurácia dos *k-folds* do Modelo II estão descritos no Gráfico 3.

**Gráfico 3: Resultados de Acurácia Modelo II (*Dataset* 1090 imagens, 32x32, lote 256 100 épocas) média de 90,23%.**

Fonte: Elaborado pelo Autor.

O Gráfico 4 apresenta os resultados da acurácia dos *k-folds* do Modelo III. Deste modelo as configurações apresentaram dimensionalidade de entrada de 64x64, um tamanho de lote de 32 e o modelo foi treinado com 50 épocas.

**Gráfico 4: Resultados de Acurácia Modelo III (Dataset 1090 imagens, 64x64, lote 32 50 épocas) média de 90,35%.**



Fonte: Elaborado pelo Autor.

Entre os 3 primeiros modelos com configurações personalizadas, o modelo III foi o que apresentou maior acurácia, levando em consideração que os três modelos foram treinados com o *Dataset* original. Neste caso em específico, a redução do custo operacional foi excelente, além da precisão do modelo III, tudo isto sem comprometer a acurácia.

A tabela 6, mostra a comparação geral dos modelos customizados, como forma de entender melhor os treinamentos realizados.

Após foram realizados treinamentos utilizando um *Dataset* maior, com 9.637 imagens. Três configurações foram testadas de modelos pré-treinados: VGG16, ResNet e *MobileNet*. Com a utilização de aplicações de *Transfer-Learning* o treinamento foi mais rápido do que treinar com o mesmo código ao qual foram treinados o Modelo I, II e III. A escolha foi devido a exigência de cálculos matemáticos como multiplicações de matrizes e convoluções que crescem proporcionalmente ao tamanho do *Dataset*.

Tabela 6: Comparação Geral dos modelos customizados.

Modelo	Resolução	BachSize	Épocas	Acurácia	Vantagens	Desvantagens
<b>Modelo I</b> (Estado da Arte)	150x150	256	150	86.71%	Boa capacidade de aprendizado com maior número de épocas.	Acurácia inferior aos demais modelos. Tempo de treinamento elevado devido a 150 épocas.
<b>Modelo II</b>	32x32	256	100	90.23%	Treinamento mais rápido devido à menor resolução de entrada. - Acurácia significativamente maior que o Modelo I.	Resolução de imagem baixa pode limitar a identificação de padrões mais detalhados
<b>Modelo III</b>	64x64	32	50	90.35%	Excelente acurácia com apenas 50 épocas. Compromisso ideal entre resolução e eficiência computacional.	Tamanho do lote menor pode resultar em maior número de passos por época, aumentando o tempo por treinamento.

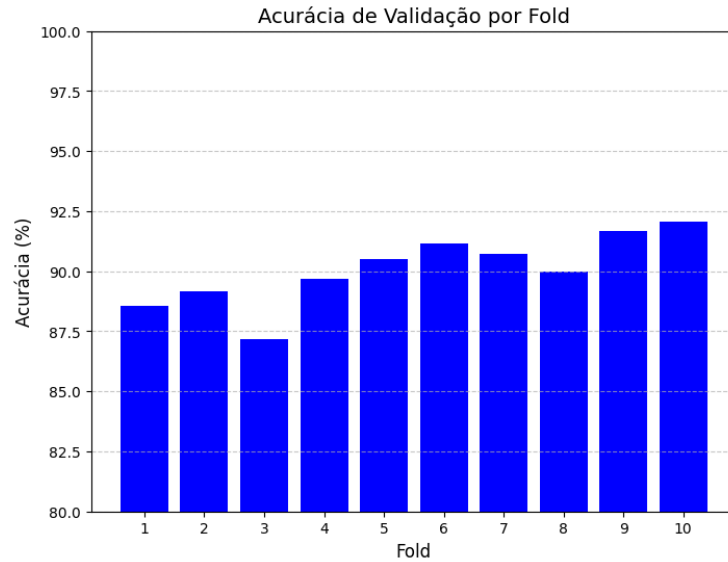
Fonte: Elaborado pelo Autor.

Os Gráficos 5 e 6 mostram as acurácias respectivamente de Modelo IV e Modelo V. ambos os treinamentos apresentaram as mesmas configurações de entrada e acabaram por apresentar a mesma acurácia. É importante salientar que embora a eficácia do treinamento no que tange o desempenho do aprendizado pela métrica de acurácia tenha sido a mesma, o VGG16 é uma arquitetura profunda e densa, que apresenta aproximadamente 138 milhões de parâmetros. Neste caso a sua estrutura fixa de camadas convolucionais seguidas de *Max-Pooling* pode ter sido excessiva devido ao *Dataset* ser menor.

Portanto o uso de um modelo menos complexo como ResNet do modelo V e até mesmo do modelo VI, indica que o modelo IV pode estar superdimensionado, ou seja, o poder computacional extra e o tempo de treinamento investido em VGG16 não trouxeram benefícios significativos em termos de precisão. Este treinamento demandou mais tempo, memória e poder de processamento quando comparado com os modelos V e VI, em função do alto número de parâmetros e imagens de alta resolução.

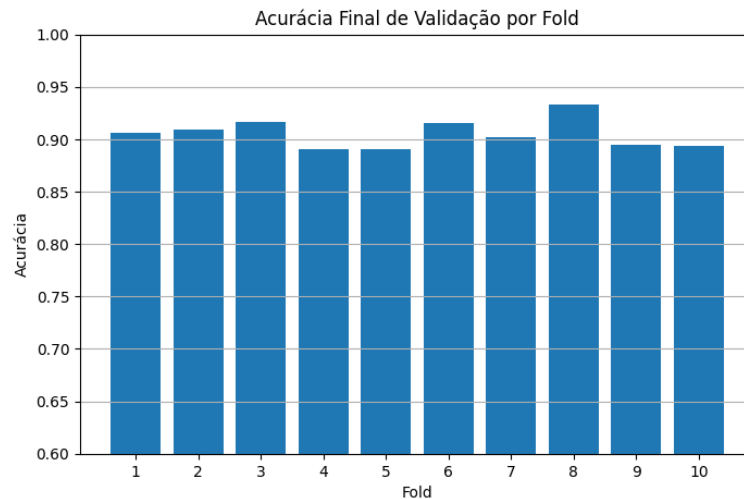
O modelo V embora tenha imagens grandes (224x224) e tenha muitas camadas, a ResNet é mais eficiente que o VGG16, isto porque o esforço computacional durante o treinamento é reduzido o que facilita o aprendizado. Este treinamento apresentou menor risco de problemas durante o treinamento.

**Gráfico 5: Resultados de Acurácia Modelo V (ResNet).**



Fonte: Elaborado pelo Autor.

**Gráfico 6: Resultados de Acurácia Modelo VI (MobileNet 1).**



Fonte: Elaborado pelo Autor.

O modelo VI fez uso dos recursos do *MobileNet*, embora a dimensionalidade das imagens também para este modelo era de uma resolução maior, ou seja, 224x224, as referências consultadas sugeriam que poderia ser usada uma resolução de 128x128. Para compensar a

diminuição da resolução, a opção foi aumentar o número de lotes, mas manter a quantidade de épocas. De modo mais eficiente o *MobileNet* reduziu significativamente o número de parâmetros e cálculos, e ainda assim alcançou um desempenho competitivo utilizando de menos recurso computacional e com menor tempo de treinamento, mostrando sua adequação a necessidade de rápida implementação e de restrições de hardware. As acurácias de cada *k-fold* deste modelo podem ser observadas no gráfico 7.

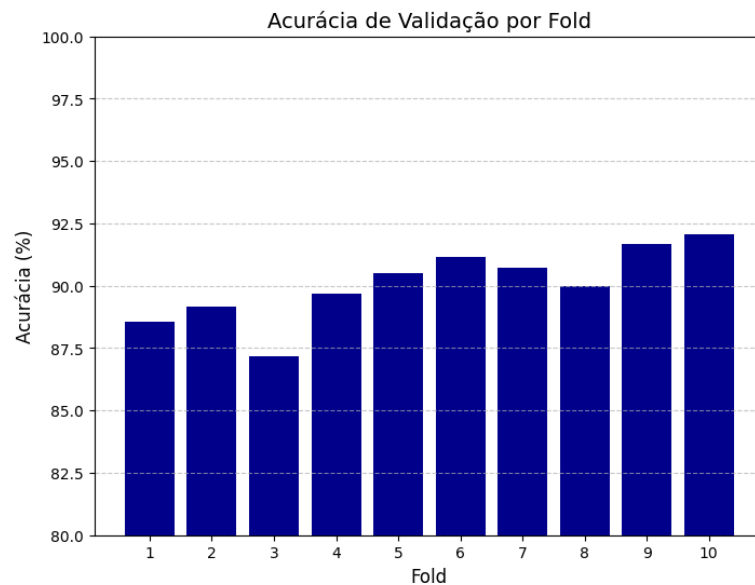
A Tabela 7 descreve as vantagens e desvantagens sobre os modelos pré-treinados. Tem-se a comparação geral dos modelos pré-treinados, com as vantagens e desvantagens de cada modelo usado neste trabalho.

**Tabela 7: Comparação Geral dos modelos pré-treinados**

Modelo	Resolução	<i>BachSize</i>	Épocas	Acurácia	Vantagens	Desvantagens
VGG16	224x224	32	40	90.07%	Robusto e consolidado	Alto custo computacional e tempo
ResNet	224x224	32	40	90.07%	Eficiente e resolve problemas de gradiente	Complexidade maior que a necessária
<i>MobileNet</i>	128x128	64	40	90.15%	Leve, rápido e custo-benefício ideal	Não maximiza desempenho em hardware potente

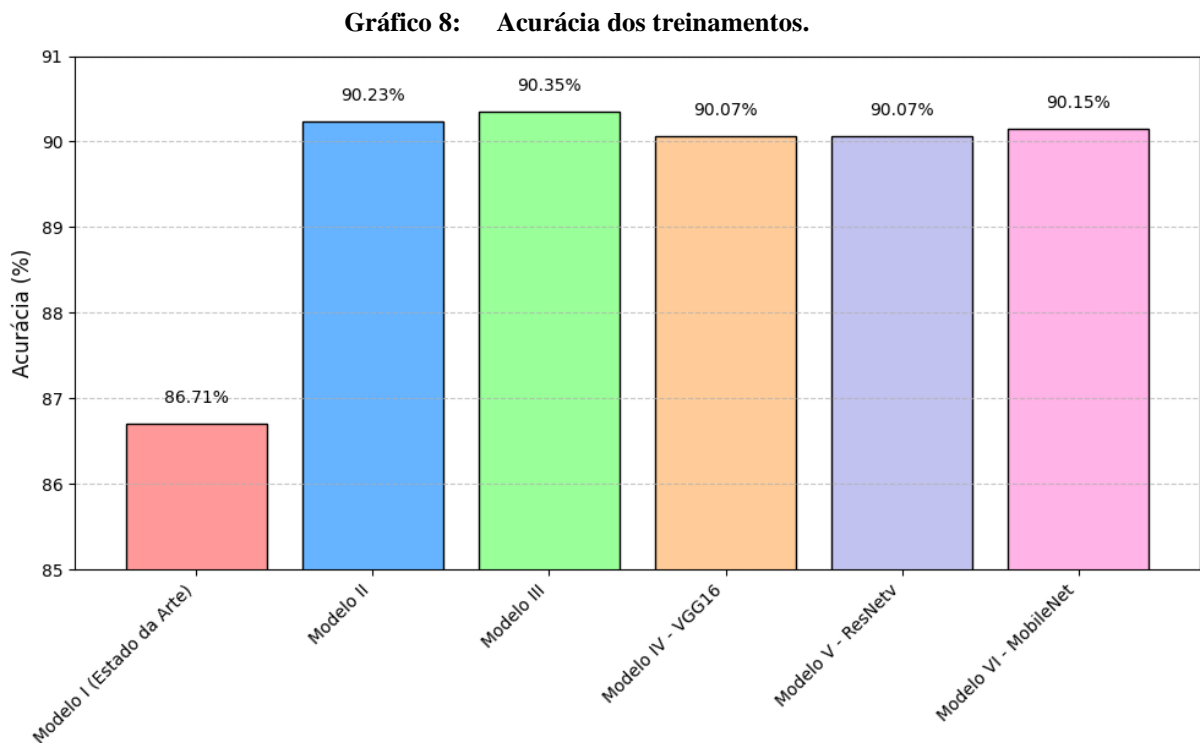
Fonte: Elaborado pelo Autor.

**Gráfico 7: Resultados de Acurácia Modelo IV (VGG16).**



Fonte: Elaborado pelo Autor.

Por fim, para uma melhor visualização das acurácias dos seis Modelos treinados, tem-se o Gráfico 8 onde pode-se ter uma visualização comparativa das acurácias.



Fonte: Elaborado pelo Autor.

Dentre os modelos personalizados, o Modelo III, com resolução de entrada 64x64, com apenas 50 épocas apresentou melhor precisão, 90.35%, indicando um equilíbrio entre a eficiência computacional e desempenho. Embora, o tamanho do lote seja menor (*batch size*), o que ocasionou um aumento do número de passos por época, a eficiência no treinamento compensou essa desvantagem, possibilitando seu uso em aplicações que exigem agilidade e bom desempenho com recursos computacionais moderados.

O Modelo II embora o treinamento seja mais rápido, as limitações na identificação de padrão em função de uma resolução de entrada menor (32x32) podem comprometer sua aplicabilidade em situações mais complexas. Nos modelos baseados em arquiteturas pré treinadas, o Modelo com *MobileNet* se destaca pela leveza, rapidez e precisão de 90.15%, indicando um excelente custo-benefício, em especial quando há recursos computacionais limitados. Deve-se considerar que o VGG16 e a ResNet são modelos mais poderosos, o custo computacional adicional não trouxe ganhos significativos em desempenho, o que torna o *MobileNet* uma melhor escolha para este problema.



No entanto, deve-se levar em consideração as aplicações práticas, e neste aspecto o Modelo III é a melhor escolha. Por apresentar uma resolução intermediária oferece um compromisso entre detalhamento e custo computacional, somado a uma precisão superior e um tempo de treinamento reduzido. Quando se analisa o contexto acadêmico e a implementação prática, o modelo III atendeu às demandas de eficiência e desempenho de projetos de classificação de imagens.

O sistema operacional utilizado para treinar o modelo é Microsoft Windows 11 Home *Single Language*, Versão 10.0.22631, em um computador com processador CPU Intel(R) Core (TM) i5-8265U @ 1.60GHz, com 4 núcleos e 8 threads, frequência de 1.8 GHz e memória RAM disponível de 8 GB.

## 6. CONCLUSÃO

Neste trabalho, foram investigados modelos de classificação de configurações diferentes para diagnóstico de doenças agrícolas por imagem de frutos afetados no cenário das CNNs e redes pré-treinadas. Para além da análise de precisão de classificação, também foram levados em conta o tempo de treinamento e aplicabilidade na prática, o que permitiria determinar a configuração mais eficiente na área agrícola.

O estudo demonstrou o quanto é importante ajustar as configurações de treinamento para que o resultado satisfaça a demanda específica de cada aplicação. Os resultados apontados mostram que o uso de modelos personalizados ou pré-treinados pode ser equilibrado entre eficiência computacional e desempenho, mas que deve ser feito levando em conta os recursos disponíveis e a demanda dos níveis de complexidade do problema, ou seja, alta ou baixa complexidade.

Recomendo que futuras pesquisas verifiquem os mesmos testes, mas com resoluções intermediárias entre as utilizadas, a fim de analisar a influência gerada no treinamento e precisão. Além disso, a experimentação com técnicas mais avançadas de data *augmentation* ajudam a potencializar o aumento de diversidade do *Dataset* e, por consequência, a melhorar a generalização dos modelos. O uso de *transfer learning* testado em modelos robustos, como *EfficientNet* e *Inception*, pode ajudar a explorar melhor como suas arquiteturas podem colaborar com a melhoria da classificação. E o teste em equipamentos mais poderosos, como GPUs mais rápidas ou TPUs, pode garantir o tempo de treinamento menor e o uso de *batch size* mais alto, favorecendo comparações mais abrangentes. Com todas essas melhorias, a aplicação pode se tornar ainda mais robusta, sendo um instrumento essencial para a agricultura sustentável e produtiva.

## REFERÊNCIAS

AHMAD, Latief; NABI, Firasath. *Agriculture 5.0: Artificial Intelligence, IoT and Machine Learning*. Boca Raton: CRC Press, 2021. Disponível em: <https://doi.org/10.1201/9781003125433>. Acesso em: 16 nov. 2024.

ARLOT, Sylvain; CELISSE, Alain. A survey of cross-validation procedures for model selection. *Statistics Surveys*, v. 4, p. 40–79, 2010. Disponível em: <https://www.projecteuclid.org/journals/statistics-surveys/volume-4/issue-none/A-survey-of-cross-validation-procedures-for-model-selection/10.1214/09-SS054.full>. Acesso em: 16 nov. 2024.

BALLARD, D. H.; BROWN, C. M. *Computer Vision*. New York: Prentice Hall Professional Technical Reference, 1982. ISBN 978-0-13-165316-0.

BRASIL. Ministério da Agricultura, Pecuária e Abastecimento. *Imagens - Tecnologia Agropecuária*. Brasília: MAPA, 2020. Disponível em: <https://www.gov.br/agricultura/pt-br/assuntos/sustentabilidade/tecnologia-agropecuaria/imagens-tecnologia-agropecuaria/tecnologia-agropecuaria/view>. Acesso em: 6 set. 2024.

BURKOV, A. *The Hundred-Page Machine Learning*. Quebec City, QC, Canada: Andriy Burkov, 2019.v.1

CRISTOVÃO, J. Orange diseases dataset. 2024. Disponível em: <https://www.kaggle.com/datasets/jonathansilva2020/orange-diseases-dataset/data>.

DATA SCIENCE ACADEMY. *Data Science Academy*. Disponível em: <https://www.datascienceacademy.com.br>. Acesso em: 16 nov. 2024.

FARIA, DIOGO. *Trabalhos Práticos Análise e Processamento de Imagem*. Artigo de Periódico. 2010. Disponível em: [https://web.fe.up.pt/~tavares/downloads/publications/relatorios/MEB\\_Diogo\\_Faria\\_TrabPraticos.pdf](https://web.fe.up.pt/~tavares/downloads/publications/relatorios/MEB_Diogo_Faria_TrabPraticos.pdf). Acesso em: 19. out. 2024.

FERRI, Cèsar; HERNÁNDEZ-ORALLO, José; MODROIU, Ramon. An experimental comparison of performance measures for classification. *Pattern Recognition Letters*, v. 30, n. 1, p. 27–38, 2009. Disponível em: <https://www.math.ucdavis.edu/~saito/data/roc/ferri-class-perf-metrics.pdf>. Acesso em: 16.nov.2024.

FURLANETTO, R.H.1; NANNI, M.R.2; CRUSIOL, L.G.T.1; SILVA, G.F.C.3; OLIVEIRA JUNIOR, A. de4. Potencial de utilização de sensores multiespectral e hiperespectral no estudo de diferentes alvos agrícolas In: *JORNADA ACADÊMICA DA EMBRAPA SOJA*, 12., 2017,

Londrina. Anais [...] Londrina: Embrapa Soja, 2017. p. 146-154., 2017. Disponível em: <https://ainfo.cnptia.embrapa.br/digital/bitstream/item/161864/1/146.pdf>. Acesso em: 23. set. 2024.

GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S. Generative adversarial nets. In: *Advances in neural information processing systems*; 2014. p. 2672–2680.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep Learning. *Deep Learning Book*: MIT Press, 2016. Disponível em: <http://www.deeplearningbook.org>. Acesso em: 21. out. 2024.

GUZMÁN, S. M., PAZ, J. O., TAGERT, M. L. M., MERCER, A. E., POTE, J. W. 2018. An integrated SVR and crop model to estimate the impacts of irrigation on daily groundwater levels. *Agricultural Systems*, 159, 248–259. Disponível em: <https://doi.org/10.1016/j.agry.2017.01.017>. Acesso em: 23. set. 2024.

HAYKIN, S. *Redes neurais: princípios e prática*. 2. ed. Bookman Editora, 2001.

HE, KAIMING; ZHANG, XIANGYU; REN, SHAOQING; SUN, JIAN. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 770-778, 2016. Disponível em: <https://doi.org/10.1109/CVPR.2016.90>. Acesso em 16.nov.2024.

Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., Wang, W., et al. (2019). *Searching for MobileNetV3*. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1314-1324. Disponível em <https://arxiv.org/abs/1905.02244>. Acesso em: 16.nov.2024.

KAGGLE. *Conjunto de dados de doenças de laranja*. Kaggle, 2020. Disponível em: <https://www.kaggle.com/Datasets/jonathansilva2020/orange-diseases-Dataset/code>. Acesso em: 16. jul. 2024.

KORBUT, D. *Machine Learning Algorithms: Which One to Choose for Your Problem*. 2017. Disponível em: <https://dzone.com/articles/machine-learning-algorithms-which-one-to-choose-fo>. Acesso em: 19. out. 2024.

MANTOVANI, E., MIRANDA, R., LANDAU, E., PASSOS, A. Agricultura de precisão no contexto do sistema de produção: lucratividade e sustentabilidade. *Boletim de pesquisa e desenvolvimento Embrapa*, 2020.

MEDEIROS, M. A.; HARTERREITEN-SOUZA, É. S.; TOGNI, P. H. B.; MILANE, P. V. G.

N.; PIRES, C. S. S.; CARNEIRO, R. G.; SUJII, E. R. Princípios e práticas ecológicas para o manejo de insetos-praga na agricultura. Projeto Biodiversidade e Transição Agroecológica de Agricultores Familiares. 1. ed. Brasília, DF: Embrapa Recursos Genéticos e Biotecnologia, 2011. p. 8 - 23.

MELGAR, M. Agricultura Digital o Agricultura 4.0. En *CENGICAÑA: Memoria Presentación de resultados de investigación Zafra 2017–2018* (p. 13). CENGICAÑA.

OLESEN, J. E. Socio-economic Impacts—Agricultural Systems. In: North Sea Region Climate Change Assessment. Regional Climate Studies; Springer: Berlin/Heidelberg, Germany, 2016 397-407. Disponível em: [https://doi.org/10.1007/978-3-319-39745-0\\_13](https://doi.org/10.1007/978-3-319-39745-0_13). Acesso em: 6. set. 2024.

PEREIRA, J. K. O. Uso de Visão Computacional para Reconhecimento de Imagens de Frutas em Imagens RGB. Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, 3, Russas, 2021. Orientação: Profa. Dra. Tatiane Fernandes Figueiredo, 2021. Disponível em: [https://repositorio.ufc.br/bitstream/riufc/64490/1/2022\\_tcc\\_jkopereira.pdf](https://repositorio.ufc.br/bitstream/riufc/64490/1/2022_tcc_jkopereira.pdf). Acesso em: 19. out. 2024.

RASCHKA, S. Python Machine Learning: Unlock deeper insights into machine learning with this vital guide to cutting-edge predictive analytics. Birmingham: Packt Publishing, 2015.

ROSA, A. L. d. Classificação de imagens de frutas utilizando aprendizado de máquina. 2019. Disponível em: <https://repositorio.ufsc.br/bitstream/handle/123456789/197598/TCC-Ayrton-Lima.pdf?sequence=1&isAllowed=y>. Acesso em: 19. out. 2024.

SABROL, H. & SATISH, K. Classificação de doenças de plantas de tomateiro em imagens digitais usando árvore de classificação. In: Conferência Internacional sobre Comunicação e Processamento de Sinais, 6-8 abril 2016, Melmaruvathur. Anais... Melmaruvathur: IEEE, 2016. p. 1242-1246. Disponível em: <https://doi.org/10.1109/ICCSP.2016.7754351>. Acesso em: 06.Set.2024.

SAMPAIO, R. M., FREDO, C. E., COSTA, A. R., & BORTOLOTTI, G. Tecnologias fitossanitárias: um estudo do Manejo Integrado de Pragas (MIP) na produção paulista de laranja, LUPA 2016/2017. Revista de Economia e Sociologia Rural, 61(3), e258289, 2023. Disponível em: <https://doi.org/10.1590/1806-9479.2022.258289>. Acesso em: 06. Set.2024.

SANDLER, M., HOWARD, A., ZHU, M., ZHMOGINOV, A., & CHEN, L. C. (2018). *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4510-4520. Disponível em: <https://arxiv.org/abs/1801.04381>. Acesso em: 16.nov.2024.

SIMONYAN, KAREN; ZISSERMAN, ANDREW. Very Deep Convolutional Networks for Large-Scale Image Recognition. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS (ICLR), 2015, San Diego. Anais [...]. San Diego: ICLR, 2015. Disponível em: <https://arxiv.org/abs/1409.1556>. Acesso em: 16 nov. 2024.

SILVA, J. R. & PEREIRA, A. L. Técnicas avançadas de monitoramento agrícola e sua importância para a sustentabilidade no agronegócio. *Revista de Agricultura Sustentável*, 2021, 12(3), 45-60.

SRIVASTAVA, Nitish; HINTON, Geoffrey; KRIZHEVSKY, Alex; SUTSKEVER, Ilya; SALAKHUTDINOV, Ruslan. *Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research*, v. 15, n. 56, p. 1929-1958, 2014. Disponível em: <http://jmlr.org/papers/v15/srivastava14a.html>. Acesso em: 16.nov.2024.

STEFFENS, D.R., DREWS JR, P.L.J., BOTELHO, S.S., MATTOS, V.L.D. Análise exploratória de dados de imagens digitais noturnas. 2019. *Scientia Plena*, v. 15, n. 4. file:///C:/Users/Usuario/Downloads/Analise\_Exploratoria\_de\_Dados\_de\_Imagens\_Digitais\_.pdf. Acesso em: 06.Set.2024.

VIEIRA, G. Analista de Automação de Processos | Em transição para Especialista em IA e Machine Learning. Disponível em: <https://www.linkedin.com/pulse/o-problema-de-Overfitting-guilherme-vieira/>. Acesso em: 21. out.2024.

WALKARN, Ujj. “A Quick Introduction to Neural Networks”. Disponível em: <https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/>. Acesso em: 12. out. 2024.