

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Augusto Victor Martins Gomides

**Análise do impacto do aumento de dados no
reconhecimento de ferrugem em folhas de café**

Uberlândia, 2024

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Augusto Victor Martins Gomides

Análise do impacto do aumento de dados no reconhecimento de ferrugem em folhas de café

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Bruno Augusto Nassif Travençolo

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Sistemas de Informação

Uberlândia,

2024

Dedico este trabalho ao meu Orientador que compartilhou de seu tempo, esforço e conhecimento para comigo e a todos que buscam seguir o belo caminho do saber.

Agradecimentos

A minha família que me forneceu apoio durante a graduação, aos meus amigos e colegas que compartilharam de seu companheirismo e ao meu orientador Bruno por todo seu apoio.

“Não sei como pareço aos olhos do mundo, mas eu mesmo vejo-me como um pobre garoto que brincava na praia e se divertia em encontrar uma pedrinha mais lisa uma vez por outra, ou uma concha mais bonita do que de costume, enquanto o grande oceano da verdade se estendia totalmente inexplorado diante de mim. (Isaac Newton)”

Resumo

A ferrugem dos cafezais é uma doença muito grave que ataca lavouras em todo o mundo e identificá-la de forma rápida e assertiva pode auxiliar no seu controle. Atualmente existem diversas formas de realizar o combate a doença da ferrugem e a melhor escolha do conjunto de estratégias para identificação afetará diretamente o processo, podendo acelerar o tempo entre o conhecimento do problema e ação de combate ao problema. Este trabalho tem como objetivo comparar o desempenho de seis Redes Neurais Convolucionais (CNN) no contexto da detecção de ferrugem em folhas de café, utilizando-se de um banco de imagens público chamado RoCoLe. Na base encontra-se um conjunto com 6 classes de folhas (saudável, doente com níveis de severidade 1 a 4, e folha com a presença de danos causados pelo ácaro-vermelho). As redes foram treinadas com e sem aumento de dados. O experimentos mostraram que com a estratégia de aumento de dados em algumas situações melhora a tarefa de classificação da rede.

Palavras-chave: Aumento de dados, Visão Computacional, Reconhecimento de Padrões, Folha de Café, Redes Convolucionais.

Sumário

1	INTRODUÇÃO	8
1.1	Objetivos	9
1.2	Hipótese	9
1.3	Justificativa	10
1.4	Organização do Trabalho	10
2	REVISÃO BIBLIOGRÁFICA	11
2.1	Redes Neurais Artificiais	11
2.1.1	Redes Neurais Profundas	11
2.2	Aumento de dados	13
2.2.1	Frameworks para Redes Neurais Profundas	13
2.2.2	Trabalhos Relacionados	14
3	MATERIAIS E MÉTODOS	16
3.1	Bases de Dados	16
3.2	Aumento de Dados	16
3.2.1	Técnicas Básicas	16
3.2.2	Técnicas Avançadas	17
3.2.3	Estratégia de aplicação do aumento de dados	18
3.2.4	Parâmetros de treinamento das redes	19
3.2.5	Ambiente de Desenvolvimento	19
4	EXPERIMENTOS	20
4.1	Parâmetros dos Experimentos	20
4.2	Estratégias de avaliação	21
4.3	Resultados	22
4.3.1	Discussão	24
4.3.1.1	ResNet	24
4.3.1.2	AlexNet	25
4.3.1.3	VGG	25
4.3.1.4	SqueezeNet	26
4.3.1.5	DenseNet	26
4.3.1.6	Inception	27
4.4	Ameaças a validade dos resultados	27
5	CONCLUSÃO	28

5.1	Trabalhos Futuros	29
	REFERÊNCIAS	31

1 Introdução

Agricultura de precisão é o termo que define metodologias no uso da tecnologia e processamento da informação em processos que visam corroborar com a agricultura (EMBRAPA, 2012), com o enfoque em gerenciar a informação, o espaço de cultivo, uso dos processos já estabelecidos na agricultura tradicional, otimizando os resultados alcançados independente de cultura produzida, da região geográfica e período do ano em que é demandada atuação. Quando aplicada corretamente, a expectativa é um ganho em volume de colheita, redução de custos, melhoria da qualidade da safra, o que impacta fortemente o mercado agrícola, uma vez que aquece o mercado devido à melhoria e aumento na quantidade do produto, que segundo levantou o jornal Estadão, tínhamos pouco mais da metade (53%) de produtores agrícolas realizando o uso de alguma metodologia de agricultura de precisão (PAPP, 2014). A agricultura de precisão tem grande impacto ambiental que devido à melhoria e aumento na quantidade do produto e a conseguinte redução de demanda territorial, reduz a necessidade de invasão ao território silvestre.

Atualmente a agricultura de precisão é fortemente dependente de tecnologias de mapeamento por imagens e a classificação de plantas existentes no espaço de atuação, que apesar de aparentar certa simplicidade, possui uma complexidade elevada e um alto custo computacional associado. A técnica não era amplamente utilizada devido a escassez de recursos, porém, o grande avanço em variedade de ferramentas e o poder computacional adquirido nos últimos anos trouxe o foco para a agricultura de precisão. Como exemplo, um importante avanço em tecnologia de dispositivos de captura de imagens (como celulares) permitiu que a visão computacional seja utilizada na agricultura e em diversas áreas com forte impacto na sociedade atual, tais como avaliação das plantas, monitoramento remoto de atividade, monitoramento de animais (CHIACCHIO, 2017). Vale observar a conseguinte redução dos custos operacionais associado diretamente ao amadurecimento e solidificação das novas tecnologias, possibilitando o acesso até mesmo a pequenos e médios produtores, que agora, são capazes de arcar com os custos de tecnologias de captura por imagens.

Vale citar que a agricultura de precisão tem como objetivo também o reconhecimento de pragas e doenças nas culturas, além de sua forte sinergia com inteligência artificial e internet das coisas, visando a automação de processos de ponta a ponta, se aprofundar em suas técnicas permite uma melhora significativa (JAFAR et al., 2024),

O cultivo do café é de grande importância em diversos países, movendo cerca de 20 bilhões de dólares todos os anos. Uma importante atividade é a avaliação da ferrugem na folha de café, uma vez que a doença é muito severa e pode afetar toda uma lavoura.

Atualmente o processo de identificação da ferrugem é feito por um agente especialista humano, considerando que em várias localidades o cultivo é realizado por pequenos produtores, o custo de contratar um especialista pode ser uma barreira, desta forma o uso da tecnologia pode tornar possível ultrapassar esta dificuldade (MACHADO, 2023).

Neste contexto, nesta pesquisa são exploradas várias estratégias para identificação da ferrugem nas folhas de café. Serão usadas seis redes neurais profundas para essa tarefa: Resnet18, AlexNet, VGG11-bn, SqueezeNet1-0, DenseNet121 e Inception v3. Foi realizado um estudo do impacto da aplicação de aumento de dados no treinamento das redes neurais usadas em comparação com o treinamento utilizando apenas as imagens originais. São apresentadas comparações entre os resultados, que permitem demonstrar qual estratégia alcança melhores resultados na classificação de folhas de café como saudáveis ou com a doença da ferrugem, sendo possível classificar os estágios diferentes da doença. A classificação correta da ferrugem e do seu estágio levará a decisões com maior efetividade, além de tornar possível uma predição em qualquer momento do ano. Isso é relevante, uma vez que mudanças climáticas demonstram afetar o desenvolvimento e propagação da doença (THAMADA; RODRIGUES; MEIRA, 2015; MEIRA; RODRIGUES, 2015).

1.1 Objetivos

Este trabalho tem como objetivo comparar o desempenho de Redes Neurais Convolucionais (CNN) no contexto da detecção de ferrugem em folhas de café, utilizando-se de um banco de imagens disponibilizado publicamente RoCoLe. Como objetivos específicos são elencados:

- Treinar diferentes conjuntos de redes neurais convolucionais utilizando a base sem aumento de dados e com aumento de dados e combinações de suas técnicas, de forma a efetuar o processo de classificação de um mesmo conjunto de imagens de teste;
- Avaliar como o uso de técnicas de aumentos de dados interfere na acurácia das CNNs.

1.2 Hipótese

Este trabalho tem a seguinte hipótese: é possível obter um melhor desempenho na acurácia da classificação da doença da ferrugem na folha do café por meio do uso de aumento de dados durante o treinamento de redes convolucionais.

1.3 Justificativa

Identificar o estágio da ferrugem é essencial para que o agricultor tome decisões eficientes e otimize a produção, reduzindo custos e aumentando a confiabilidade dos resultados. No entanto, desafios como o acesso limitado à tecnologia e a pouca familiaridade com smartphones complicam o uso da inteligência artificial no campo, além das dificuldades na coleta de dados de qualidade, prejudicada por variações climáticas e limitações dos dispositivos. Assim, é fundamental que a solução seja acessível, intuitiva e integrada com sistemas de apoio locais, garantindo usabilidade e eficácia. O uso de técnicas variadas de aumento de dados, aliado a uma abordagem planejada, pode oferecer uma ferramenta eficiente para monitorar e combater a ferrugem, superando as barreiras do ambiente rural e beneficiando agricultores de diferentes perfis.

1.4 Organização do Trabalho

Neste primeiro capítulo foi apresentada a contextualização do trabalho, bem como seus objetivos e hipótese.

Seguem elencados no Capítulo 2 uma revisão dos tópicos pertinentes e de trabalhos com conteúdo relevante à monografia, possuindo estes, as aplicações de diversas técnicas essenciais e são descritas as técnicas fundamentais para desenvolvimento da metodologia e desenvolvimento do corpo do modelo proposto.

O Capítulo 3 refere-se aos materiais e métodos, que envolvem a metodologia de desenvolvimento e pesquisa abordada na implementação do projeto.

Na sequência, o Capítulo 4 registra os experimentos realizados pertinentes ao modelo, sendo a base de dados utilizados, experimentos executados e resultados obtidos.

Finalmente o Capítulo 5 se encerra demonstrando as conclusões alcançadas pelo modelo e trabalhos desenvolvidos. Além disso, os trabalhos futuros que foram desencadeados pelas indagações e resultados.

2 Revisão Bibliográfica

Este trabalho conduz a aplicação de conhecimento em duas áreas distintas: O Processamento de Digital de Imagens e a Inteligência Artificial – estas possuem técnicas que funcionam em parcimônia, como é o caso da classificação de imagens digitais em conjunto com Redes Neurais Artificiais. A principal tarefa que será realizada é a de Classificação, que basicamente determina se um objetivo pertence a uma categoria (RUSSELL; NORVIG, 2022). Em particular, serão classificadas se as folhas pertencem à categoria saudável ou a categoria que possui ferrugem (e qual o grau da ferrugem). Para essa tarefa serão utilizadas redes neurais artificiais. E, para o treinamento dessas redes, será utilizada a estratégia de aumento de dados. Neste capítulo serão detalhados esses conceitos necessários para o desenvolvimento da pesquisa.

2.1 Redes Neurais Artificiais

Rede Neural Artificial consiste da conexão de diversos neurônios artificiais, que por sua vez, é uma unidade processadora, que resumidamente pode ser considerada como um modelo simplificado da estrutura neural biológica (YEGNANARAYANA, 2009). O modelo geral de uma Rede Neural utiliza suas unidades processadoras executando a somatória das entradas multiplicadas pelos pesos e gera uma saída em um intervalo desejado. São consideradas como inibidas entradas que possuem seus pesos negativos e excitatória nos casos em que o pesos são positivas.

2.1.1 Redes Neurais Profundas

Redes Neurais profundas foram definidas na década de 1970, mas seu uso foi intensificado na última década. O termo aprendizado profundo refere-se ao aprendizado dessas redes por meio de várias camadas de elementos de computação simples e configuráveis (RUSSELL; NORVIG, 2022). Neste trabalho foram utilizadas seis tipos dessas redes: Resnet18, AlexNet, VGG11-bn, SqueezeNet1-0, DenseNet121 e Inception v3.

- ResNet18: Rede neural profunda foi selecionada por se tratar de uma variante com estrutura simplificada, possuindo 18 camadas e que tem como objetivo mitigar ¹, um problema presente quando usamos redes mais profundas. Além disso, faz uso

¹ O problema do gradiente desaparecido ocorre durante o treinamento de redes neurais muito profundas, quando os gradientes das camadas iniciais se tornam muito pequenos à medida que são propagados para trás. Isso impede que essas camadas aprendam de maneira efetiva, levando a dificuldades no treinamento e em alcançar boas performances.

do conceito de conexões residuais, permitindo que os gradientes fluam melhor durante o treinamento, possuindo um bom desempenho no uso em problemas de visão computacional (HE et al., 2016).

- AlexNet: CNN com cinco camadas convolucionais seguidas de três camadas completamente conectadas, ganhando popularidade após vencer um desafio de reconhecimento de imagens em 2012. Foi a primeira rede neural moderna a tornar viável trabalhar em grandes conjuntos de dados com redes profundas e possui relativa simplicidade se comparada a outros modelos modernos, mas possui limitações de performance quando comparada com modelos mais recentes, além de exigir maior cuidado para regularização com o intuito de evitar *overfitting*² (KRIZHEVSKY; SUTSKEVER; HINTON, 2012).
- VGG11-bn: A VGG é uma rede que ganhou visibilidade por simplificar a complexidade das CNNs (SIMONYAN; ZISSERMAN, 2014). A VGG11-bn é uma versão que possui 11 camadas convolucionais e utiliza *batch normalization* para estabilizar e acelerar o treinamento. Apesar de possuir uma estrutura relativamente simples e uniforme, seu desempenho de uso de recursos e performance é inferior a modelos mais modernos. Entretanto, sua estrutura modular a torna atraente para várias aplicações de visão computacional.
- SqueezeNet1-0: Modelo projetado para ser pequeno e eficiente em termos de recursos computacionais, mantendo uma precisão comparável a redes maiores (IANDOLA et al., 2016). SqueezeNet utiliza “fire modules”, que combinam diferentes configurações de camadas convolucionais 1×1 e 3×3 , reduzindo o número de parâmetros sem perda significativa de precisão. Esse modelo é ideal para aplicações móveis e embarcadas, além de ser eficaz em implementações em tempo real, devido ao seu baixo consumo de recursos. Contudo, perde desempenho em tarefas que exigem maior complexidade, quando comparado a outras CNNs mais profundas.
- DenseNet121: A arquitetura DenseNet121 conecta cada nova camada a todas as anteriores, facilitando o fluxo de informações e de gradientes, o que auxilia no treinamento de redes muito profundas (HUANG et al., 2017). Essa configuração reduz o problema do gradiente desaparecido³ e requer menos parâmetros de entrada, quando comparada a outras CNNs. Entretanto, a reutilização dessa arquitetura para tarefas diferentes das quais foi inicialmente treinada pode ser desafiadora, e seu custo computacional é elevado.

² Overfitting é um problema que ocorre quando um modelo de aprendizado de máquina se ajusta excessivamente aos dados de treinamento, aprendendo ruídos e detalhes irrelevantes, o que resulta em um desempenho ruim em novos dados.

³ O problema do gradiente desaparecido ocorre quando, durante o treinamento de redes neurais profundas, os gradientes das camadas iniciais se tornam muito pequenos à medida que são propagados para trás, dificultando o aprendizado adequado das camadas iniciais.

- Inception v3: O Inception v3 é uma versão avançada da arquitetura Inception, que utiliza convoluções em múltiplas escalas e uma combinação complexa de blocos convolucionais (SZEGEDY et al., 2016). O modelo equilibra profundidade e largura de maneira eficiente, proporcionando um excelente desempenho em tarefas de classificação de imagens. Porém, a complexidade da sua implementação e a necessidade de entradas de tamanho específico impactam negativamente sua flexibilidade em resoluções de problemas diversos.

2.2 Aumento de dados

O aumento de dados em redes neurais e visão computacional é uma técnica utilizada com o propósito de incrementar a quantidade e a diversidade de exemplos disponíveis, sem a necessidade de produzir novos dados manualmente (SHORTEN; KHOSHGOFTAAR, 2019). Essa técnica aplica transformações e manipulações nos dados já existentes, gerando variações que aumentam a complexidade e a representatividade do conjunto de dados (PEREZ; WANG, 2017).

O aumento de dados é essencial para melhorar o desempenho dos modelos e aumentar sua capacidade de generalização, atendendo à necessidade de grandes volumes de amostras para alcançar resultados significativos e reduzir o risco de *overfitting* (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Assim, o uso de aumento de dados é particularmente relevante em aplicações onde é difícil ou custoso coletar um grande número de exemplos, como em imagens médicas, segurança automotiva e inspeção de qualidade (SIMONYAN; ZISSERMAN, 2014).

Vale destacar que existem técnicas consideradas como “aumentos de dados simples”, que podem manipular a imagem por meio de rotação, espelhamento (*flip*), translação, corte (*cropping*) e escalonamento, e “aumentos de dados avançados” que podem gerar manipulações mais robustas, como ajuste de iluminação e cor, adição de ruído, distorção de perspectiva, corte com zoom aleatório e *mixup*, sendo essas técnicas frequentemente utilizadas de forma combinada (HE et al., 2016; SZEGEDY et al., 2016).

2.2.1 Frameworks para Redes Neurais Profundas

Existem vários frameworks para o desenvolvimento e treinamento de redes neurais profundas, sendo dois dos mais populares: PyTorch e TensorFlow. O PyTorch, desenvolvido pelo Facebook AI Research, usa programação imperativa ou procedural, onde o código é executado linha por linha, permitindo que os resultados sejam “visíveis” em tempo real. Esse modelo é intuitivo para desenvolvedores com experiência em linguagens como Java e C, facilitando a depuração e possibilitando ajustes dinâmicos durante o treinamento.

Além disso, sua integração natural com Python e ferramentas como Torchvision o tornam uma boa escolha para desafios de visão computacional. No entanto, como foi projetado com foco em pesquisa e prototipagem rápida ([PyTorch Contributors, 2024](#)), ele não possui um ecossistema tão robusto para *deployment* em produção como o TensorFlow, que oferece ferramentas específicas para essa finalidade, como TensorFlow Serving e TensorFlow Lite.

Por outro lado, o TensorFlow, desenvolvido pelo Google, é um framework consolidado, ideal para *deployment* em larga escala e suporte a diferentes plataformas, incluindo TensorFlow Lite e TensorFlow.js. Ele também oferece o TensorBoard para visualização do treinamento e tem forte suporte a treinamento distribuído. No entanto, apresenta algumas desvantagens: sua curva de aprendizado é mais íngreme, e a configuração inicial pode ser mais complexa, especialmente para prototipagem rápida, devido à necessidade de estruturação rigorosa e uso de grafos computacionais ([TensorFlow Contributors, 2024](#)).

Além disso, a ampla gama de ferramentas pode ser excessiva para projetos menores, dificultando a adoção eficiente, e o estilo declarativo de seu código torna a depuração menos intuitiva em comparação com o fluxo dinâmico do PyTorch. Embora o TensorFlow ofereça ferramentas avançadas e seja mais adequado para *deployment* em escala, o PyTorch se destaca pela flexibilidade e facilidade de uso. Sua API imperativa e a integração com Torchvision facilitam a prototipagem rápida e permitem ajustes dinâmicos durante a pesquisa. Essas características são especialmente importantes para este estudo, uma vez que o foco é explorar e testar diferentes modelos de CNN e técnicas de aumento de dados. Além disso, o PyTorch foi utilizado no trabalho de referência sobre o dataset RoCoLe, fortalecendo a decisão de empregá-lo para garantir consistência metodológica e comparabilidade dos resultados obtidos.

2.2.2 Trabalhos Relacionados

Este projeto teve como inspiração um Trabalho de Conclusão de Curso desenvolvido na Universidade Federal de Uberlândia ([MACHADO, 2023](#)). Esse trabalho teve como foco o mesmo problema e base de dados descrito na presente pesquisa. No entanto, para esta pesquisa é realizada a tarefa adicional de avaliar o impacto do aumento de dados na acurácia do treinamento das redes.

No estudo de ([GHEORGHIU et al., 2024](#)), foi realizada uma análise sobre o uso de técnicas de aumento de dados para maximizar o potencial do conjunto de dados Robusta Coffee Leaf (RoCoLe) que é caracterizado por um volume reduzido de imagens e um alto desequilíbrio de classes, com predominância de amostras de folhas saudáveis e poucas imagens de folhas afetadas por doenças, o que representa um desafio no treinamento de modelos robustos de aprendizado de máquina. Para superar essas limitações e aprimorar a precisão da classificação, o objetivo foi identificar as combinações mais eficazes de

estratégias para balancear o conjunto de dados e enriquecer a diversidade das amostras, garantindo que os modelos de classificação, como os baseados em redes convolucionais (CNN) e *Transformers*, pudessem obter melhores resultados e generalização em cenários desafiadores de dados escassos e desbalanceados.

No trabalho de (NAWAZ et al., 2024), foi realizada a análise de doenças em folhas de café utilizando aprendizado profundo. Eles propõem o uso de mecanismos de atenção espaciais e de canais no modelo ResNet-50. Conseguiram com isso uma acurácia de 98,54% na tarefa de classificação.

3 Materiais e Métodos

Neste capítulo serão descritas as imagens usadas, as estratégias de aumento de dados e os parâmetros usados no treinamento das redes.

3.1 Bases de Dados

Para os experimentos foi utilizada a base RoCoLe (PARRAGA-ALAVA, 2019). Essa base, publicada em 2019, contém 1560 imagens, sendo estas um conjunto de 4 imagens para cada planta de café diferente, totalizando 390 unidades da cultura de café robusta, também conhecido como *Coffea canephora* que possui resistência natural a doença da ferrugem. A base de imagens foi utilizada no treinamento e validação dos algoritmos de classificação, podendo ser subdivida em um conjunto binário (folha doente e folha saudável) ou em uma conjunto com 6 classes (folha saudável, folha doente com níveis de severidade 1 a 4 e folha com a presença de danos causados pelo ácaro-vermelho). Na Figura 1 é mostrado um exemplar de cada tipo de folha de café que passou pelo processo de classificação.

3.2 Aumento de Dados

Foram selecionadas técnicas de aumento de dados para variar a iluminação e adicionar sombras sintéticas, considerando que, por se tratar de um local exposto ao ambiente natural, há impactos gerados pela luz solar e pelo posicionamento da câmera (como celulares) em relação às folhas. Além disso, foram aplicadas manipulações de cor e transformações no espaço de cor, dado que existem outras doenças que apresentam padrões visuais semelhantes na folhagem. Por fim, foram utilizadas técnicas de rotação e translação para aumentar a robustez do modelo.

3.2.1 Técnicas Básicas

- **Rotação:** Gira a imagem em ângulos variados para aumentar a variabilidade angular dos dados.
- **Translação:** Move a imagem em diferentes direções para simular deslocamentos.
- **Flip Horizontal e Vertical:** Espelha a imagem nos eixos horizontal e vertical.
- **Crop:** Realiza cortes aleatórios para simular diferentes enquadramentos.



Figura 1 – Exemplos das folhas de café presentes na base RoCoLe. Da direita para esquerda e de cima para baixo: folha de café saudável, com ácaro vermelho e com ferrugem dos níveis de 1 ao 4.

- **Escalonamento (*Zoom in/out*):** Ajusta a escala da imagem para aumentar a diversidade de perspectivas.

3.2.2 Técnicas Avançadas

- **Alteração de Brilho, Saturação e Matiz:** Utiliza ajustes parciais de matiz e saturação para simular diferentes condições de cor e iluminação. O impacto dessas modificações são moderadas, mas combinadas com outras transformações, podem alterar significativamente a aparência das doenças, dificultando a discriminação en-

tre as classes.

- **Adição de Sombra:** Insere sombras sintéticas na imagem para emular variações de luz natural. O impacto dessa adição é que ela pode mascarar detalhes importantes, como manchas ou lesões, dificultando a identificação precisa, especialmente em níveis leves de ferrugem.
- **Iluminação Direcional:** Aplica luz de forma parcial para simular condições de iluminação ambiente com diferentes direções. Simula iluminação realista, mas pode introduzir artefatos que dificultam a identificação de pequenos detalhes.
- **Desfoque Gaussiano:** Aplica a suavização Gaussiana (`GaussianBlur`) com diferentes tamanhos de máscara e sigma, aumentando a robustez do modelo. Simula borramentos na imagem. Desfoques muito fortes podem ocultar detalhes importantes, prejudicando a classificação das doenças e a identificação de ácaros.
- **Transformação do Espaço de Cor:** Ajusta a imagem para espaços de cor alternativos. É útil para treinar o modelo em diferentes condições de iluminação, mas alterações intensas podem confundir o modelo.
- **Redimensionamento:** As imagens são redimensionadas para 224×224 pixels, garantindo consistência no treinamento.

3.2.3 Estratégia de aplicação do aumento de dados

Durante o treinamento, utilizou-se aumento de dados com probabilidade de 50%, aplicando distorções como rotação, contraste e brilho. Isso visa tornar o modelo mais robusto a variações reais, como iluminação e ruídos visuais.

A justificativa para a escolha da probabilidade de 50% foi feita com a intenção de gerar um conjunto de dados mais diversos e realizar os aumentos de maneira mais “agressiva”, ou seja, com maior frequência. A seguir são apresentados três exemplos em que o aumento de dados pode ser justificado.

- **Exemplo 1:** Uma folha saudável capturada sob iluminação forte pode aparecer ligeiramente diferente se a foto for tirada em um ambiente mais escuro. O aumento de dados, ao alterar o brilho, ensina o modelo a reconhecer que ambos representam uma folha saudável.
- **Exemplo 2:** A ferrugem em plantas pode ser capturada de diferentes ângulos. Aplicar rotações aleatórias nas imagens de treinamento ajuda o modelo a identificar padrões da doença independentemente da orientação da foto.

- **Exemplo 3:** Pequenos ruídos, como desfoque ou sujeira na lente da câmera, podem ser simulados com transformações de contraste. Assim, o modelo aprende a ignorar interferências e focar nas características principais da planta.

3.2.4 Parâmetros de treinamento das redes

Para treinamento das redes foram usados os seguintes parâmetros:

- **Batch Size:** 32 - As imagens serão processadas em lotes de 32 por vez.
- **Critério de Perda:** Entropia cruzada (*CrossEntropyLoss*) – Função de perda fornecida pelo módulo `torch`.
- **Número de Épocas:** 5.

3.2.5 Ambiente de Desenvolvimento

Todos os experimentos foram executados ambiente descrito a seguir. O código utilizado está disponível no github.

- **Sistema Operacional:** Windows 10, 64 bits
- **Processador:** 12th Gen Intel(R) Core(TM) i5-12450H 2.50 GHz
- **Memória RAM:** 24 GB (23,7 GB utilizáveis)
- GPU:
 - GPU: NVIDIA GeForce RTX 3050
 - Memória Total: 4096 MiB
 - **Utilização da GPU:** 100% (Sob carga máxima)
 - **Driver:** Versão 546.12 **CUDA:** Versão 12.3
- **Código, Frameworks e Ferramentas:**
 - Python: 3.10.15
 - PyTorch: 2.3.0 (CPU) e Torchvision: 0.18.1
 - TensorBoard: Integrado para visualização
 - Link para o git: [Repositório Git](#)

4 Experimentos

Este capítulo apresenta os experimentos realizados, os resultados obtidos e finaliza com a discussão desses resultados.

4.1 Parâmetros dos Experimentos

Todas as redes foram treinadas com as seguintes configurações:

- **Redes avaliadas:** ResNet18, AlexNet, VGG11-bn, SqueezeNet1-0, DenseNet121, Inception v3.
- **Número de Épocas:** 5
- **K-folds:** 2
- **Classes:** 6 (1 Saudável + 5 Estágios de Ferrugem)

Cada rede é avaliada em 2 *folds*, e cada *fold* envolve 5 épocas de treinamento. Como são 6 redes avaliadas, foram realizados 12 experimentos. Cada *fold* a base foi separada em Treino e Validação.

1. Treinamento do Modelo

- Cada *fold* é treinado com 780 imagens, abrangendo 6 classes.
- O treinamento é realizado por 5 épocas, com otimização dos pesos para minimizar a função de perda.
- Técnicas de Aumento de dados são aplicadas para aumentar a robustez do modelo (nossa hipótese).
- O desempenho é monitorado durante o treinamento usando a perda (*loss*) e a acurácia em cada época.

2. Validação do Modelo

- Cada *fold* é avaliado em 780 imagens, abrangendo 6 classes.
- A métrica de acurácia foi utilizada para medir o desempenho.
- Os resultados obtidos em ambos os *folds* são combinados para fornecer uma avaliação final do modelo.

4.2 Estratégias de avaliação

A Tabela 1 apresentação seis estratégias que foram utilizadas para validação da hipótese desse trabalho. Cada estratégia recebeu um nome do tipo *Aug* <número>.

Tabela 1 – Estratégia de avaliação de aumento de dados. **Nome** indica um rótulo para a estratégia. **Aumento de Dados** indica como foi realizado o aumento de dados. **Transformações Aplicadas** mostram as principais funções em *python* para implementação das estratégias.

Nome	Aumento de Dados	Transformações Aplicadas - funções <i>python</i> usadas e seus parâmetros
Aug 0	Nenhum aumento de dados, apenas normalização	Redimensionamento, CenterCrop, Normalização com média = [0.485, 0.456, 0.406] e desvio = [0.229, 0.224, 0.225]
Aug 1	<i>Aug 0</i> + Aumento básico com 50% de probabilidade	RandomResizedCrop, RandomHorizontalFlip, RandomVerticalFlip, RandomAffine (translação), RandomRotation
Aug 2	<i>Aug 0</i> + Aumento avançado com 50% de probabilidade	AdjustColorSpace(l_adjust=10, a_adjust=10, b_adjust=10, hist_eq=True), AdjustHueSaturation(hue_shift=10, saturation_scale=1.2, brightness_shift=20), ApplyDirectionalLight(light_angle=120, light_intensity=0.9), transforms.GaussianBlur(kernel_size=(5, 9), sigma=(0.1, 5)), AddShadow(vertex_range=(0.2, 0.6), intensity_range=(0.5, 0.7), shadow_color=(0, 0, 0))
Aug 3	Aumento básico 100%	RandomResizedCrop, RandomHorizontalFlip, RandomVerticalFlip, RandomAffine, RandomRotation
Aug 4	Aumento avançado 100%	AdjustColorSpace(l_adjust=10, a_adjust=10, b_adjust=10, hist_eq=True), AdjustHueSaturation(hue_shift=10, saturation_scale=1.2, brightness_shift=20), ApplyDirectionalLight(light_angle=120, light_intensity=0.9), transforms.GaussianBlur(kernel_size=(5, 9), sigma=(0.1, 5)), AddShadow(vertex_range=(0.2, 0.6), intensity_range=(0.5, 0.7), shadow_color=(0, 0, 0))
Aug 5	Aumento básico + aumento avançado, cada um com 50% de probabilidade	Combinação de todas as transformações dos aumentos básico e avançado.

4.3 Resultados

Os resultados obtidos em cada análise especificados na Seção 4.1 são mostrados nas Tabelas 2 a 7.

Tabela 2 – Resultados ResNet – 2 *folds* – 5 Épocas

Experimento	Melhor <i>Val Loss</i>	Val Acc (%)	Train Loss (Última Época)	Train Acc (%)
Fold 0 - Aug 0	0,8350	70,51	0,3615	90,90
Fold 0 - Aug 1	0,6944	73,21	0,4789	82,18
Fold 0 - Aug 2	0,8836	67,82	0,7611	73,21
Fold 0 - Aug 3	0,7213	73,85	0,6718	74,10
Fold 0 - Aug 4	1,4080	51,67	1,4198	48,59
Fold 0 - Aug 5	0,9811	65,26	0,9487	65,26
Fold 1 - Aug 0	0,2788	95,90	0,2005	96,54
Fold 1 - Aug 1	0,4587	85,13	0,4902	83,97
Fold 1 - Aug 2	0,7295	75,13	0,7532	75,00
Fold 1 - Aug 3	0,6228	76,03	0,6709	73,85
Fold 1 - Aug 4	1,3982	49,74	1,3894	50,26
Fold 1 - Aug 5	0,9014	67,18	0,9938	65,13

Tabela 3 – Resultados AlexNet – 2 *folds* – 5 Épocas

Experimento	Melhor <i>Val Loss</i>	Val Acc (%)	Train Loss (Última Época)	Train Acc (%)
Fold 0 - Aug 0	0,8937	65,90	0,5794	78,72
Fold 0 - Aug 1	0,7210	74,10	0,7018	74,87
Fold 0 - Aug 2	0,8368	69,23	0,8826	68,08
Fold 0 - Aug 3	0,7556	72,95	0,7886	70,26
Fold 0 - Aug 4	1,3547	51,67	1,3781	49,74
Fold 0 - Aug 5	0,9779	64,74	0,9521	64,74
Fold 1 - Aug 0	0,5893	77,18	0,3995	84,74
Fold 1 - Aug 1	0,7105	71,28	0,6190	77,44
Fold 1 - Aug 2	0,8159	72,56	0,8447	68,85
Fold 1 - Aug 3	0,7327	71,54	0,7437	73,59
Fold 1 - Aug 4	1,3704	49,74	1,3748	51,67
Fold 1 - Aug 5	0,9254	65,26	0,9548	66,15

Tabela 4 – Resultados VGG – 2 folds – 5 Épocas

Experimento	Melhor <i>Val Loss</i>	Val Acc (%)	Train Loss (Última Época)	Train Acc (%)
Fold 0 - Aug 0	0,7407	72,05	0,4631	83,85
Fold 0 - Aug 1	0,5458	78,59	0,5067	81,54
Fold 0 - Aug 2	0,8948	67,31	0,8498	69,23
Fold 0 - Aug 3	0,6565	75,00	0,6637	74,23
Fold 0 - Aug 4	1,3607	51,67	1,3903	50,00
Fold 0 - Aug 5	0,9243	66,79	0,9286	65,38
Fold 1 - Aug 0	0,3687	86,79	0,2405	91,54
Fold 1 - Aug 1	0,5325	78,33	0,4811	81,92
Fold 1 - Aug 2	0,8488	68,21	0,7860	72,95
Fold 1 - Aug 3	0,6328	77,31	0,6095	77,05
Fold 1 - Aug 4	1,3732	49,74	1,3642	51,67
Fold 1 - Aug 5	0,9051	67,69	0,9168	66,15

Tabela 5 – Resultados SqueezeNet – 2 folds – 5 Épocas

Experimento	Melhor <i>Val Loss</i>	Val Acc (%)	Train Loss (Última Época)	Train Acc (%)
Fold 0 - Aug 0	1,0224	61,92	1,0438	61,67
Fold 0 - Aug 1	0,7042	74,10	0,7198	74,23
Fold 0 - Aug 2	1,0096	63,97	1,0012	64,23
Fold 0 - Aug 3	0,7101	73,59	0,7498	73,59
Fold 0 - Aug 5	0,9844	64,23	1,0459	60,77
Fold 1 - Aug 0	0,6789	74,36	0,6678	75,64
Fold 1 - Aug 1	0,6773	75,13	0,6291	78,21
Fold 1 - Aug 2	0,9640	64,62	0,9678	66,67
Fold 1 - Aug 3	0,7199	72,95	0,6865	73,85
Fold 1 - Aug 4	1,3768	49,74	1,3631	51,67
Fold 1 - Aug 5	1,0203	63,59	0,9447	66,79

Tabela 6 – Resultados DenseNet – 2 *folds* – 5 Épocas

Experimento	Melhor <i>Val Loss</i>	Val Acc (%)	Train Loss (Última Época)	Train Acc (%)
Fold 0 - Aug 0	0,7845	68,33	0,3791	90,38
Fold 0 - Aug 1	0,4896	82,44	0,4513	85,13
Fold 0 - Aug 2	0,8391	69,36	0,8532	69,87
Fold 0 - Aug 3	0,6347	75,26	0,5588	79,36
Fold 0 - Aug 4	1,3804	51,41	1,4289	47,31
Fold 0 - Aug 5	0,9618	65,13	1,0077	60,51
Fold 1 - Aug 0	0,2874	91,79	0,1837	96,67
Fold 1 - Aug 1	0,4929	85,00	0,3877	87,31
Fold 1 - Aug 2	0,8216	69,49	0,8100	71,67
Fold 1 - Aug 3	0,6319	74,87	0,5722	79,23
Fold 1 - Aug 4	1,3953	49,74	1,3852	51,54
Fold 1 - Aug 5	0,9648	64,74	0,9548	65,26

Tabela 7 – Resultados Inception – 2 *folds* – 5 Épocas

Experimento	Melhor <i>Val Loss</i>	Val Acc (%)	Train Loss (Última Época)	Train Acc (%)
Fold 0 - Aug 0	0,8239	67,95	0,8561	75,77
Fold 0 - Aug 1	0,5256	80,00	0,7133	81,41
Fold 0 - Aug 2	0,8629	68,85	1,1352	69,23
Fold 0 - Aug 3	0,6346	75,38	0,8219	78,59
Fold 0 - Aug 4	1,3872	51,79	1,9690	47,82
Fold 0 - Aug 5	0,9334	66,67	1,3190	64,62
Fold 1 - Aug 0	0,4647	82,69	0,5494	86,67
Fold 1 - Aug 1	0,4446	84,62	0,6502	82,95
Fold 1 - Aug 2	0,7786	72,82	1,0722	73,33
Fold 1 - Aug 3	0,5115	80,64	0,8145	77,05
Fold 1 - Aug 4	1,3822	49,74	1,9314	50,77
Fold 1 - Aug 5	0,8870	67,18	1,3593	63,72

4.3.1 Discussão

Nesta seção, para cada rede estuda é feita uma análise dos resultados obtidos, trazendo os principais destaques que podem ser observados nas tabelas. Vale observar que essa é apenas uma análise inicial, pois o número de épocas usado e o número de *folds* foi muito baixo e isso pode ter interferido no resultado.

4.3.1.1 ResNet

- **Fold 1 (Validação):**

- Aug 0: 95.90%
- Aug 4 (Aumento Avançado 100%): 49.74%
- Aug 5 (Aumento Básico + Avançado): 67.18%

- **Fold 1 (Treinamento):**

- Aug 0: 96.54%
- Aug 4: 50.26%
- Aug 5: 65.13%

Análise: A aplicação de aumentos avançados reduziu consideravelmente o desempenho tanto em treino quanto em validação. Isso sugere que aumentos muito complexos podem introduzir variações irrelevantes para a tarefa.

4.3.1.2 AlexNet

- **Fold 0 (Validação):**

- Aug 0 (Sem Aumento): 65.90%
- Aug 1 (Aumento Básico com 50%): 74.10%
- Aug 3 (Aumento Básico Completo): 72.95%

- **Fold 0 (Treinamento):**

- Aug 0: 78.72%
- Aug 1: 74.87%
- Aug 3: 70.26%

Análise: O aumento básico (tanto parcial quanto completo) trouxe ganhos significativos na *Val Acc*. Embora a acurácia de treinamento tenha diminuído com aumentos básicos completos (Aug 3), a generalização do modelo melhorou na validação.

4.3.1.3 VGG

- **Fold 0 (Validação):**

- Aug 0: 72.05%
- Aug 1: 78.59%
- Aug 3: 75.00%

- **Fold 0 (Treinamento):**

- Aug 0: 83.85%
- Aug 1: 81.54%
- Aug 3: 74.23%

Análise: O aumento básico ajudou a melhorar a acurácia de validação (*Val Acc*), especialmente em Aug 1. No entanto, a menor acurácia de treino com Aug 3 indica uma melhor generalização.

4.3.1.4 SqueezeNet

- **Fold 0 (Validação):**

- Aug 0: 61.92%
- Aug 1: 74.10%
- Aug 3: 73.59%

- **Fold 0 (Treinamento):**

- Aug 0: 61.67%
- Aug 1: 74.23%
- Aug 3: 73.59%

Análise: O aumento básico (Aug 1) trouxe melhorias significativas na validação em comparação com a ausência de aumento (Aug 0). Os resultados de Aug 3 mostram um equilíbrio entre treino e validação.

4.3.1.5 DenseNet

- **Fold 0 (Validação):**

- Aug 0: 68.33%
- Aug 1: 82.44%
- Aug 3: 75.26%

- **Fold 0 (Treinamento):**

- Aug 0: 90.38%
- Aug 1: 85.13%
- Aug 3: 79.36%

Análise: O Aumento Básico com 50% (Aug 1) teve o maior impacto na *Val Acc*, alcançando 82.44%. A discrepância entre a acurácia de treino e validação em Aug 0 sugere *overfitting* sem aumento.

4.3.1.6 Inception

- **Fold 1 (Validação):**
 - Aug 0: 82.69%
 - Aug 1: 84.62%
 - Aug 2 (Aumento Avançado com 50%): 72.82%
- **Fold 1 (Treinamento):**
 - Aug 0: 86.67%
 - Aug 1: 82.95%
 - Aug 2: 73.33%

Análise: A Inception alcançou valores elevados de acurácia com aumentos básicos (Aug 1), confirmando o benefício do aumento leve. Já o aumento avançado (Aug 2) teve um impacto negativo, sugerindo que transformações mais complexas podem introduzir ruído.

4.4 Ameaças a validade dos resultados

Conforme já discutido anteriormente, o número baixo de épocas executados nos experimentos não permite realizar afirmações assertivas sobre os resultados obtidos. No trabalho de (MACHADO, 2023) foram usadas mais de 50 épocas, sendo que para a ResNet pode-se observar o início da convergência com 5 épocas. Outro ponto de destaque é que não foram realizados experimentos que levam em consideração o desbalanceamento das classes – e esse fator também é uma variável importante para a realização de conclusões a respeito dos experimentos. Além disso, também foram usados vários filtros em sequência, o que pode gerar algumas “contradições” e operações repetidas a depender da probabilidade escolhida.

5 Conclusão

Neste trabalho foi apresentado o desempenho de seis diferentes redes neurais convolucionais na tarefa de classificação do tipo de ferrugem presentes em folhas de café. Foi avaliado o desempenho dessas redes com diferentes tipos de treinamento – com e sem as estratégias de aumento de dados.

Os resultados mostram o impacto do Aumento de Dados na Acurácia de Validação (*Val Acc*). Comparando a acurácia entre Aug 0 (sem aumento) e as outras variações com diferentes estratégias de aumento para avaliar seu efeito. A análise foi dividida entre as fases de treinamento e validação, focando nos ganhos significativos em *Val Acc* e na relação entre a acurácia de treinamento e validação. Esse comparativo ajuda a identificar possíveis *overfitting* ou ruídos indesejados gerados pelas distorções aplicadas durante os aumentos.

Foi observado que uma probabilidade de 50% para aplicar os aumentos de dados pode ter introduzido um nível excessivo de ruído, impactando negativamente os resultados. Por isso, a redução da intensidade das transformações e da frequência de aumentos (para algo entre 10% e 30%) pode gerar variações mais adequadas que melhoram o treinamento, mitigam o *overfitting*, e reduzem o tempo de processamento por época, otimizando o aprendizado do modelo.

Além disso, a variação no número de épocas e a estratégia de validação cruzada (k-fold) podem trazer impactos positivos no desempenho, aumentando a robustez e a generalização do modelo. Esses ajustes são importantes para equilibrar a variedade gerada pelos aumentos e o desempenho consistente em dados não vistos. Também devem ser consideradas questões relacionadas ao desbalanceamento das classes. Dado o vasto número de variáveis envolvidas para uma análise precisa, não foi possível validar a hipótese do trabalho, pois para isso seriam necessários diversos outros experimentos e, pela limitação de tempo, tais experimentos foram elencados como trabalhos futuros.

Este trabalho é bastante desafiador, visto que o curso de Sistemas de Informação não tem em sua base curricular obrigatória disciplinas de Processamento de Imagens, Inteligência Artificial, Reconhecimento de Padrões, entre outras. Pode-se citar também que muitos conceitos matemáticos também não são abordados no curso. Assim, realizar uma pesquisa em uma área tão atual e complexa é uma ótima experiência para complementação da formação acadêmica. Neste trabalho foi realizada uma base metodológica para análise de folhas de café. Apesar das várias melhorias que podem ser realizadas, a estrutura para experimentos está pronta, bastando a execução de novos experimentos que podem ajudar na validação da hipótese apresentada.

5.1 Trabalhos Futuros

Além das sugestões já indicadas neste conclusão (aumento das épocas e dos *folds*), como trabalhos futuros podem ser feitas investigações que de utilização de diferentes parâmetros e diferentes estratégias de aumento de dados.

Ajuste das Probabilidades e intensidades:

- **Aplicável a todas as arquiteturas**, pois o controle sobre a intensidade das transformações (como `GaussianBlur` e `ApplyDirectionalLight`) ajuda a evitar ruído excessivo.
- No entanto, **redes mais profundas** (como **ResNet** e **DenseNet**) podem ser mais sensíveis a variações irrelevantes, exigindo ajustes mais cuidadosos nas probabilidades.

Ajuste Adaptativo por Arquitetura:

- **AlexNet e SqueezeNet**: Por serem redes menos profundas, toleram melhor variações maiores e podem se beneficiar de **transformações mais intensas**.
- **ResNet e DenseNet**: Precisam de aumentos sutis (como leves desfoques) para evitar a introdução de ruído que dificulte o aprendizado.
- **Inception**: Esta arquitetura é adequada para dados variados, mas o uso excessivo de transformações complexas pode prejudicar sua generalização.

Validação e Monitoramento Contínuos:

- **Essencial para todas as arquiteturas**, pois permite a detecção de *overfitting* e ajuste das transformações conforme necessário.
- As redes mais profundas (ex.: **DenseNet**) podem necessitar de um monitoramento mais frequente, pois tendem a memorizar padrões espúrios em dados excessivamente transformados.

Planos Específicos para Treinamento e Validação:

- **Treinamento**: Aplicar aumentos diversificados (como sombras e saturação) é útil para redes menos profundas (ex.: **AlexNet**, **VGG**).
- **Validação**: Todas as redes devem usar transformações leves durante a validação para manter a consistência com os dados reais.

Por fim, é importante mencionar que os planos descritos anteriormente têm como objetivo alcançar resultados próximos ao ótimo. Para isso, foram organizados em etapas distintas, e este trabalho representa o primeiro passo na direção desse objetivo: identificar o “melhor” conjunto de estratégias para maximizar a eficácia do treinamento e o reconhecimento dos itens desejados.

Assim que esse conjunto de estratégias estiver definido, a próxima fase será “aprimorar” essas abordagens, propondo ajustes que explorem seus limites e, eventualmente, nos levem a resultados superiores. Na etapa seguinte, o foco será desenvolver novas soluções que tirem o máximo proveito do poder computacional disponível, impulsionando ainda mais a eficiência e a precisão dos processos.

Referências

- CHIACCHIO, S. S. R. *Veículo aéreo não tripulado de asa rotativa na atividade de mapeamento e coleta de imagem na agricultura de precisão e no monitoramento de animais*. Dissertação (M.Sc. Thesis) — Faculdade de Zootecnia e Engenharia de Alimentos, February 2017. Citado na página 8.
- EMBRAPA. *Planejamento e gerenciamento de todos os processos da produção*. [S.l.], 2012. Disponível em: <<https://www.macroprograma1.cnptia.embrapa.br/redeap2/o-que-e-agricultura-de-precisao>>. Citado na página 8.
- GHEORGHIU, A. et al. Evaluating data augmentation techniques for coffee leaf disease classification. *Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART)*, Springer, p. TBD, 2024. Disponível em: <<https://arxiv.org/abs/2401.05768>>. Citado na página 14.
- HE, K. et al. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, p. 770–778, 2016. Citado 2 vezes nas páginas 12 e 13.
- HUANG, G. et al. Densely connected convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, p. 2261–2269, 2017. Citado na página 12.
- IANDOLA, F. N. et al. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size. *arXiv preprint arXiv:1602.07360*, 2016. Disponível em: <<https://arxiv.org/abs/1602.07360>>. Citado na página 12.
- JAFAR, A. et al. Revolutionizing agriculture with artificial intelligence: plant disease detection methods, applications, and their limitations. *Frontiers in Plant Science*, v. 15, p. 1356260, 2024. Disponível em: <<https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2024.1356260/full>>. Citado na página 8.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, ACM, v. 60, n. 6, p. 84–90, 2012. Citado 2 vezes nas páginas 12 e 13.
- MACHADO, T. V. *Aplicação de técnicas de Visão Computacional para identificação de ferrugem em folhas de café*. 2023. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal de Uberlândia. Disponível em: <<https://repositorio.ufu.br/handle/123456789/38294>>. Citado 3 vezes nas páginas 9, 14 e 27.
- MEIRA, C. A. A.; RODRIGUES, L. H. A. *Árvore de decisão na análise de epidemias da ferrugem do cafeeiro*. [S.l.], 2015. 3-5 p. Citado na página 9.
- NAWAZ, M. et al. Coffeenet: A deep learning approach for coffee plant leaves diseases recognition. *Expert Systems with Applications*, v. 237, p. 121481, 2024. ISSN 0957-4174. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0957417423019838>>. Citado na página 15.

- PAPP, A. C. *Agricultura de 'precisão' ganha espaço no País*. [S.l.], 2014. Disponível em: <<https://economia.estadao.com.br/noticias/geral,agricultura-de-precisao-ganha-espaco-no-pais-imp-,1166826>>. Citado na página 8.
- PARRAGA-ALAVA, J. *RoCoLe: A robusta coffee leaf images dataset*. Mendeley, 2019. Disponível em: <<https://dx.doi.org/10.17632/C5YVN32DZG.2>>. Citado na página 16.
- PEREZ, L.; WANG, J. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017. Disponível em: <<https://arxiv.org/abs/1712.04621>>. Citado na página 13.
- PyTorch Contributors. *PyTorch Documentation*. [S.l.], 2024. Disponível em: <<https://pytorch.org/docs/stable/index.html>>. Citado na página 14.
- RUSSELL, S. J.; NORVIG, P. *Inteligência Artificial: Uma Abordagem Moderna*. [S.l.]: Grupo GEN LTC, 2022. ISBN 9788595159495. Citado na página 11.
- SHORTEN, C.; KHOSHGOFTAAR, T. M. A survey on image data augmentation for deep learning. *Journal of Big Data*, Springer, v. 6, p. 1–48, 2019. Citado na página 13.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. Disponível em: <<https://arxiv.org/abs/1409.1556>>. Citado 2 vezes nas páginas 12 e 13.
- SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, p. 2818–2826, 2016. Citado na página 13.
- TensorFlow Contributors. *TensorFlow Documentation: Tensor Guide*. [S.l.], 2024. Disponível em: <<https://www.tensorflow.org/guide/tensor?hl=pt-br>>. Citado na página 14.
- THAMADA, T. T.; RODRIGUES, L. H. A.; MEIRA, C. A. A. *Predição da taxa de progresso da ferrugem do cafeeiro por meio de ensembles*. [S.l.], 2015. Disponível em: <http://www.sbicafe.ufv.br/bitstream/handle/123456789/4134/8_IX-SPCB-2015.pdf>. Citado na página 9.
- YEGNANARAYANA, B. *ARTIFICIAL NEURAL NETWORKS*. PHI Learning, 2009. ISBN 9788120312531. Disponível em: <https://books.google.com.br/books?id=RTtvUVU_xL4C>. Citado na página 11.