

UNIVERSIDADE FEDERAL DO PAMPA (UNIPAMPA)

Luiz Daniel Garay Trindade

**Investigando Técnicas de Processamento de
Imagens com IA na Detecção de Ferrugem
em Folhas de Soja**

Alegrete
2021

Luiz Daniel Garay Trindade

**Investigando Técnicas de Processamento de Imagens
com IA na Detecção de Ferrugem em Folhas de Soja**

Dissertação de Mestrado apresentada ao
Curso de Mestrado Profissional em Engenharia
de Software da Universidade Federal do
Pampa como requisito parcial para a obtenção
do título de Mestre em Engenharia de
Software.

Orientador: Prof. Dr. Fábio Paulo Basso

Coorientador: Prof. Dr. Daniel Welfer

Alegrete
2021

Luiz Daniel Garay Trindade

Investigando Técnicas de Processamento de Imagens com IA na Detecção de Ferrugem em Folhas de Soja

Dissertação/Tese apresentada ao Programa de (Nome do Programa) da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Mestre/Doutor em (Área do mestrado ou doutorado).

Dissertação defendida e aprovada em: 30, Agosto de 2021.

Banca examinadora:

Prof. Dr. Fábio Paulo Basso

Orientador

Unipampa

Prof. Dr. Daniel Welfer

Co-orientador

UFSM

Prof. Dr. Elder de Macedo Rodrigues

Unipampa

Prof. Dr. Maicon Bernardino da Silveira

Unipampa



Assinado eletronicamente por **FABIO PAULO BASSO, PROFESSOR DO MAGISTERIO SUPERIOR**, em 30/08/2021, às 14:01, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **Daniel Welfer, Usuário Externo**, em 30/08/2021, às 17:54, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **MAICON BERNARDINO DA SILVEIRA, PROFESSOR DO MAGISTERIO SUPERIOR**, em 30/08/2021, às 19:45, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **ELDER DE MACEDO RODRIGUES, PROFESSOR DO MAGISTERIO SUPERIOR**, em 01/09/2021, às 14:18, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



A autenticidade deste documento pode ser conferida no site https://sei.unipampa.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0604056** e o código CRC **2734F05D**.

AGRADECIMENTOS

Agradeço a minha família pelo apoio e motivação durante o desenvolvimento deste estudo.

Ao Prof. Dr. Fábio Paulo Basso, pela orientação e dedicação no desenvolvimento desta pesquisa, assim como no apoio ao meu crescimento intelectual, profissional e pessoal.

Ao Prof. Dr. Daniel Welfer, pela coorientação e apoio no desenvolvimento desta pesquisa.

Ao Programa de Pós Graduação em Engenharia de Software (PPGES).

À Banca Examinadora pelo convite aceito e também ao tempo dedicado na avaliação deste estudo.

Ao colega e amigo Antônio de Freitas Vale pelo apoio no desenvolvimento da revisão sistemática da literatura, na aquisição de imagens e na criação do banco de dados.

Ao colega e amigo Dhyonatan Santos de Freitas pelo compartilhamento de conhecimento e apoio no desenvolvimento dos experimentos.

RESUMO

Com o aumento da população mundial, se faz necessário o aumento da produção agrícola. Uma vez que um dos problemas enfrentados pelos agricultores são as pragas e as doenças nas culturas, as quais podem causar grandes danos às lavouras, o desenvolvimento de tecnologias que minimizem as perdas é essencial para o aumento da produtividade. Algumas destas tecnologias zelam pelo meio ambiente, e dentre elas destacam-se as que são construídas para a detecção automática e precoce de doenças por meio de Veículos Aéreos Não Tripulados (VANTs) em conjunto com métodos computacionais de aprendizado profundo. A literatura da área carece de estudos comparativos de tais métodos. O objetivo principal deste estudo foi investigar a acurácia e a eficiência de três métodos computacionais, implementados sob a forma de técnicas de processamento de imagens e arquiteturas de redes neurais, no problema da detecção de ferrugem em folhas de soja. Para mapear os métodos aplicáveis ao problema, executou-se uma revisão sistemática de literatura, caracterizando técnicas de detecção de doenças em plantas e gerando um conjunto de informações importantes que estabeleceram a base inicial da pesquisa pela seleção de três métodos para o treinamento de Redes Neurais Convolucionais (RNCs): Imagens na Íntegra, SLIC e *augmentor*. Os métodos foram testados e comparados, gerando resultados bastante motivadores que comprovam a eficiência do uso de técnicas de aumento de dados de imagens como uma etapa importante de pré-processamento. Como conclusão, os estudos realizados apontam o método *augmentor* como o que apresenta melhores resultados de pré-processamento, sendo, portanto, o algoritmo mais interessante para o treinamento de RNCs quando aplicadas na detecção precoce de ferrugem em folhas do soja. Além disso, também apresenta-se um leque de novas possibilidades de pesquisa que podem ser aplicadas à agricultura inteligente e de precisão.

Palavras-chave: Agricultura de Precisão. Agricultura Inteligente. Aprendizado de Máquina. *Deep Learning*. Processamento de Imagens. Redes Neurais Convolucionais.

ABSTRACT

With the increase in world population, it is necessary to increase agricultural production. Since one of the problems faced to faced by farmers are pests and diseases in crops, which can cause larges damage to crops, the development of technologies that minimize losses is essential to increase productivity. Some of these technologies care for the environment, and among them stand out as they are built for the automatic and early detection of diseases through Unmanned Aerial Vehicles (UAVs) in conjunction with a computational method of deep learning. The literature in the area lacks comparative studies of such methods. The main objective of this study was to investigate the accuracy and efficiency of three computational methods, implemented in the form of image processing techniques and neural network architectures, without rust detection problems in soybean leaves. To map the methods applicable to the problem, we perform a systematic literature review. We characterize techniques for detecting diseases in plants and generate a set of important information that established an initial basis for research by selecting three methods for training Convolutional Neural Networks (CNNs): Original Imagens, SLIC, and Augmentor. The methods were tested and compared, generating quite motivating results that prove the efficiency of using image data augmentation techniques as an essential pre-processing step. In conclusion, the studies carried out point to the Augmentor method as the one with the best pre-processing results, being, therefore, the most interesting algorithm for training CNNs when applied in the early detection of soybean leaf rust. In addition, there is also a range of new research possibilities that can be applied to smart farming and precision agriculture.

Key-words: Precision agriculture. Smart Agriculture. Machine Learning. Detection of Diseases. Deep Learning. Image processing. Convolutional Neural Networks.

LISTA DE FIGURAS

Figura 1 – Fluxograma do cronograma de pesquisa.	22
Figura 2 – Exemplo de aprendizado supervisionado.	30
Figura 3 – Exemplo de aprendizado não-supervisionado.	31
Figura 4 – Neurônio artificial	34
Figura 5 – Funções de ativação	36
Figura 6 – Exemplo de rede <i>feedforward</i> de camada simples	37
Figura 7 – Exemplo de rede <i>feedforward</i> de múltiplas camadas	38
Figura 8 – Exemplo de arquitetura RNC.	40
Figura 9 – Exemplo de convolução.	41
Figura 10 – Exemplos de conectividade global e esparsa.	42
Figura 11 – Exemplo de campo receptivo local nas RNCs.	43
Figura 12 – Exemplo de compartilhamento de parâmetros.	43
Figura 13 – Exemplo de <i>pooling</i> utilizando o operador <i>max</i> , com filtro 2x2 e <i>stride</i> igual a 2.	45
Figura 14 – Exemplo de <i>overfitting</i>	46
Figura 15 – Exemplo de treinamento utilizando técnica de regularização, evitando <i>overfitting</i>	47
Figura 16 – Exemplo de uma camada totalmente conectada de uma RNC	48
Figura 17 – O processo de estudo de mapeamento sistemático.	51
Figura 18 – String de busca genérica	53
Figura 19 – Pesquisa e seleção de estudos	54
Figura 20 – Número de artigos publicados por ano	55
Figura 21 – Amostra de imagens de plantas doentes	59
Figura 22 – Gráfico de dispersão mostrando os resultados obtidos pelo tamanho do conjunto de dados	61
Figura 23 – Gráfico de dispersão mostrando os resultados obtidos em cada estudo pelo número de classes	63
Figura 24 – Imagem da folha da soja com sintomas de ferrugem.	70
Figura 25 – Divisão da base de dados	71
Figura 26 – Modelos criados no primeiro experimento	73
Figura 27 – Resultado dos modelos do primeiro experimento utilizando a base de imagens original	74
Figura 28 – Exemplo da aplicação do algoritmo SLIC em imagem da folha da soja doente.	77
Figura 29 – Divisão da base de dados	78
Figura 30 – Imagens já segmentadas pelo algoritmo SLIC.	79
Figura 31 – Modelos criados no segundo experimento.	80

Figura 32 – Resultado de acurácia dos modelos usando o conjunto de dados aumentado pelo algoritmo SLIC	81
Figura 33 – Resultado da precisão dos modelos usando o conjunto de dados sem o aumento dos dados	81
Figura 34 – Exemplo da utilização do algoritmo <i>augmentor</i>	86
Figura 35 – Divisões de dados utilizadas no terceiro experimento	87
Figura 36 – Modelos de RNCs utilizados no terceiro experimento	89
Figura 37 – Resultado do treinamento e teste do modelo escolhido com imagens originais e imagens aumentadas	90
Figura 38 – Resultados dos melhores modelos dos três experimentos conduzidos	94

LISTA DE TABELAS

Tabela 1 – Esforço na produção científica de materiais de divulgação de resultados	24
Tabela 2 – Material Complementar Produzido ao Longo do Mestrado	25
Tabela 3 – Exemplo de matriz de confusão.	31
Tabela 4 – Comparação dos trabalhos relacionados.	49
Tabela 5 – Descrição do pacote de revisão sistemática da literatura	50
Tabela 6 – Bancos de dados de artigos usados para pesquisar título, resumo e palavras-chave	52
Tabela 7 – Todos os estudos selecionados com sua taxa de sucesso adquirida na detecção de doenças	56
Tabela 8 – Plantas e doenças analisadas nos estudos revisados	58
Tabela 9 – Todos os estudos selecionados com sua configuração de conjunto de dados	62
Tabela 10 – Divisão da base de imagens original utilizada no primeiro experimento	72
Tabela 11 – Resultados dos modelos 1, 2 e 3 utilizando o conjunto de imagens original sem a utilização da técnica de aumento de dados	75
Tabela 12 – Base de imagens após processo de aumento de dados utilizando o algoritmo SLIC.	78
Tabela 13 – Resultados dos modelos 2, 4 e 5 utilizando o conjunto de imagens aumentado pela técnica SLIC	82
Tabela 14 – Resultados dos modelos 2, 4 e 5 utilizando o conjunto de imagens original sem a utilização da técnica de aumento de dados	83
Tabela 15 – Resultados dos modelos 1, 2 e 3 usando o conjunto de imagens aumentado com a técnica SLIC	83
Tabela 16 – Antes e depois da aplicação do algoritmo de aumento de dados <i>augmentor</i>	87
Tabela 17 – Divisão das imagens após o aumento de dados.	88
Tabela 18 – Resultados da validação do modelo com as imagens originais e os resultados dos modelos com as imagens aumentadas e com suas respectivas divisões.	91
Tabela 19 – Resultados da validação dos modelos com o número de épocas diferente.	92
Tabela 20 – Resultados de validação dos melhores modelos de cada um dos experimentos.	95

SUMÁRIO

1	INTRODUÇÃO	19
1.1	Justificativa	20
1.2	Objetivos	21
1.3	Metodologia de Pesquisa	21
1.3.1	Atividades	22
1.3.2	Contribuições	24
1.4	Organização do Trabalho	25
2	EMBASAMENTO TEÓRICO	27
2.1	Agricultura de Precisão	27
2.2	Inteligência Artificial	27
2.2.1	Aprendizado de Máquina	28
2.2.1.1	Aprendizado Supervisionado	29
2.2.1.2	Aprendizado Não-Supervisionado	30
2.2.1.3	Avaliação de Aprendizagem	31
2.2.2	Redes Neurais Artificiais (RNA)	32
2.2.2.1	Neurônio Artificial	33
2.2.2.2	Arquiteturas de RNAs	36
2.2.3	(<i>Deep Learning</i>)	38
2.2.4	Redes Neurais Convolucionais (RNC)	39
3	REVISÃO SISTEMÁTICA DE LITERATURA	49
3.1	Trabalhos Relacionados	49
3.2	Protocolo de Pesquisa	50
3.2.1	Objetivo de Pesquisa	51
3.2.2	Processo de Pesquisa	51
3.2.3	Fase de Planejamento	51
3.2.4	Definição de Protocolo de Pesquisa	52
3.2.5	Questões de Pesquisa	53
3.2.6	Critérios de Inclusão e Exclusão	53
3.2.7	Resultados	54
3.3	Resultados do Estudo de Revisão Sistemática da Literatura	55
3.3.1	Contextos de domínio de estudos aplicados	56
3.3.2	Comparando os estudos	60
3.3.3	Qualidade e número de classes de imagens de conjunto de dados	60
3.4	Discussões	63
3.5	Análise dos Artigos com Foco em Doenças no Soja	64
3.6	Ameaças à validade	65

3.7	Considerações Finais	66
4	EXPERIMENTOS CONTROLADOS	69
4.1	Protocolo e Metodologia dos Experimentos	69
4.1.1	Base de Imagens	69
4.1.2	Ferramentas	70
4.1.3	Metodologia dos Experimentos	70
4.2	Primeiro Experimento	71
4.2.1	Protocolo do Primeiro Experimento	71
4.2.1.1	Etapa 1: Pré-processamento de Imagens do Primeiro Experimento	71
4.2.1.2	Etapa 2: Desenvolvimento dos Modelo de RNCs para o Primeiro Experimento	72
4.2.1.3	Etapa 3: Coleta de Métricas para o Primeiro Experimento	73
4.2.1.4	Resultados do Primeiro Experimento	73
4.2.1.5	Validação do Primeiro Experimento	74
4.2.1.6	Discussões do Primeiro Experimento	75
4.2.2	Considerações Finais do Primeiro Experimento	75
4.3	Segundo Experimento - Avaliação do Algoritmo Superpixel SLIC	76
4.3.1	Protocolo do Segundo Experimento	77
4.3.1.1	Etapa 1: Pré-processamento das Imagens do Segundo Experimento	77
4.3.1.2	Etapa 2: Desenvolvimento dos Modelos de RNCs para o Segundo Experimento	79
4.3.1.3	Etapa 3: Coleta de Métricas para o Segundo Experimento	80
4.3.1.4	Resultados do Segundo Experimento	80
4.3.1.5	Validação do Segundo Experimento	82
4.3.1.6	Discussões do Segundo Experimento	83
4.3.2	Considerações Finais do Segundo Experimento	84
4.4	Terceiro Experimento - Avaliação do Algoritmo <i>Augmentor</i> .	85
4.4.0.1	Etapa 1: Pré-processamento das Imagens do Terceiro Experimento	86
4.4.0.2	Etapa 2: Desenvolvimento do Modelo de RNC para o Terceiro Experimento	88
4.4.0.3	Etapa 3: Coleta de Métricas do Terceiro Experimento	89
4.4.0.4	Resultados do Terceiro Experimento	90
4.4.0.5	Validação do Terceiro Experimento	90
4.4.0.6	Discussões do Terceiro Experimento	92
4.4.1	Considerações Finais do Terceiro Experimento	93
4.5	Comparação dos Três Experimentos	93
4.5.1	Comparação dos Resultados Iniciais dos Experimentos	94
4.5.2	Comparação dos Resultados de Validação dos Três Experimentos	95

5	CONCLUSÃO	97
	REFERÊNCIAS	101

1 INTRODUÇÃO

O aumento contínuo da população mundial, conseqüentemente, resulta na necessidade de uma maior produção de alimentos (FAO, 2009). Assim, avanços tecnológicos na agricultura vem ganhando destaque a fim de oferecer suporte para a tomada de decisão dos agricultores com maior assertividade e, por conseguinte, contribuindo para uma maior produção. Neste contexto, (KAMILARIS; PRENAFETA-BOLDÚ, 2018) informa que a agricultura inteligente contribui diretamente com o suporte para a tomada de decisão e tem como objetivo resolver os problemas encontrados na produção agrícola com base em algumas premissas, tais como: A) produção, B) impacto ambiental, C) segurança alimentar e D) sustentabilidade, visando uma maior produtividade alinhada com a segurança alimentar. A saúde das plantas é extremamente importante para os agricultores e ela está diretamente ligada com a segurança alimentar. Segundo (BOULENT et al., 2019), as pragas e doenças podem levar a perda de até 40% da produção mundial de alimentos, sendo assim, uma ameaça à segurança dos alimentos.

Uma das formas mais comuns para solucionar problemas com pragas e doenças nas lavouras é por meio da utilização de agrotóxicos. Seu uso tem contribuído para o aumento significativo da produção agrícola desde a década de 1950, atendendo assim às necessidades da crescente população mundial (COOPER; DOBSON, 2007). Porém, utilizar agrotóxicos não é totalmente inofensivo ao meio ambiente, podendo afetar negativamente a biodiversidade, incluindo insetos, peixes, pássaros, a qualidade do solo, a qualidade da água, a qualidade do ar e, conseqüentemente, os seres humanos (BOULENT et al., 2019).

Com o advento das tecnologias no campo, a obtenção massiva de dados e informações sobre as condições do campo pode atenuar o uso de agrotóxicos no meio agrícola, essa integração é chamada de agricultura de precisão (BERNARDI et al., 2014). A agricultura de precisão tende à beneficiar o produtor e o meio ambiente, uma vez que permite que o agricultor encontre o local e a hora certa para a aplicação dos defensivos agrícolas. Porém, quando feita manualmente e sem o auxílio de tecnologias de suporte à agricultura inteligente, a tarefa de verificar e analisar a condição das plantas em uma cultura torna-se complexa, resultando em maior trabalho e colaborando para incertezas na tomada de decisão.

Assim, a identificação precoce de doenças em plantas por meio de técnicas computacionais de detecção orientada a dados pode auxiliar e resolver os problemas encontrados pelos agricultores (BOULENT et al., 2019).

As técnicas de aprendizado de máquina, área da inteligência artificial que encontra-se em constante aperfeiçoamento, evoluíram significativamente o processamento de imagens com o uso de Redes Neurais Convolucionais (RNC) (GHOSAL et al., 2018), permitindo a execução de estudos aplicados para a agricultura de precisão (LIBO; TIAN; CHUNYUN, 2019). Segundo (BOULENT et al., 2019), desde 2016, várias aplicações

para a identificação de doenças em culturas foram desenvolvidas (SINGH et al., 2019).

As tecnologias vigentes, aplicadas a agricultura de precisão, são capazes de auxiliar na identificação de problemas na lavoura como o da ferrugem na soja, problema no qual é a motivação desta pesquisa. Nas seções seguintes serão apresentadas as justificativas e os objetivos deste trabalho.

1.1 Justificativa

Segundo (SAVARY et al., 2012), as doenças nas lavouras, incluindo a ferrugem na soja, podem levar à perda da produção agrícola ou aumentar o uso de agrotóxicos. Com o avanço das tecnologias no campo e o fomento de pesquisas nessa área, estudos aplicados para a detecção automática de doenças em plantas são fundamentais para aprimorar as técnicas de extração de conhecimento entre as diferentes áreas de conhecimento, visando a detecção precoce de doenças nas lavouras.

Nesse sentido, torna-se necessário desenvolver e testar alternativas computacionais que possibilitem a detecção precoce de doenças no campo. Em especial, estudos aplicados para as lavouras de soja configuram parte dos interesses do arranjo produtivo dos estados com maior índice de produtividade, por exemplo o Rio Grande do Sul (MACIEL et al., 2015), contexto no qual esta dissertação de mestrado esta inserida.

Na agricultura de precisão, o uso de Veículos Aéreos Não Tripulados (VANTs) tem sido altamente utilizado como forma de monitoramento das lavouras (MADDIKUNTA et al., 2021). Tais veículos oferecem muitos recursos para a captura de imagens e favorecem o uso de técnicas de processamento de imagens (IQBAL et al., 2018). Contudo, um dos problemas enfrentados na maioria das vezes é a escolha de qual técnica/ algoritmo tem o melhor desempenho e, por conseguinte, gerar melhores resultados.

Por mais que a área da agricultura inteligente tenha evoluído em estudos aplicados para a detecção automática de doenças na lavoura (AL-BAYATI; ÜSTÜNDAĞ, 2020), percebe-se que estudos comparativos são escassos na área, assim como estudos utilizando RNCs e VANTs em um ambiente real (PANTAZI et al., 2017). Essa lacuna na literatura foi constatada em uma revisão sistemática conduzida ao longo dessa pesquisa. Nesse sentido, uma das motivações deste estudo é justamente a proposição de evidências comparativas entre diferentes técnicas, abordagens e ferramentas para a identificação de doenças na soja utilizando imagens.

O arranjo produtivo local de Alegrete (RS) caracteriza-se como favorável à adoção de tecnologias deste tipo. Observa-se uma certa mudança de cultura dos produtores rurais, os quais vem gradualmente migrando da pecuária para a agricultura, e.g pastagens de gado para a plantação do soja. Neste sentido, uma vez que diferentes climas, regiões e tratamento precoce afetam bastante o espaço amostral das imagens obtidas pelos VANTs. Com isso, a indisponibilidade das imagens para a realização de estudos aplicados para cenários da região caracteriza-se como um desafio.

Uma vez que a indisponibilidade de imagens apresenta-se como um problema, esta dissertação busca primeiramente formar uma base de dados para o treinamento de RNCs. Para tal, realiza-se a coleta de evidências em relação à viabilidade de algoritmos de enriquecimento de imagens, os quais são importantes para a tomada de decisões de pesquisadores que atuam na área. Portanto, o foco de contribuição é para com a pesquisa no tema, um passo importante para a condução de estudos futuros aplicados para a essa prática.

1.2 Objetivos

O objetivo geral deste trabalho é investigar técnicas de processamento de imagens por meio de inteligência artificial na detecção de ferrugem em folhas de soja. A técnica escolhida para o desenvolvimento desta pesquisa é o aprendizado profundo (*Deep Learning*), mais especificamente as RNCs, aplicadas juntamente com dois diferentes algoritmos de enriquecimento de imagens na etapa de pré-processamento dos dados.

Os objetivos específicos desta pesquisa incluem:

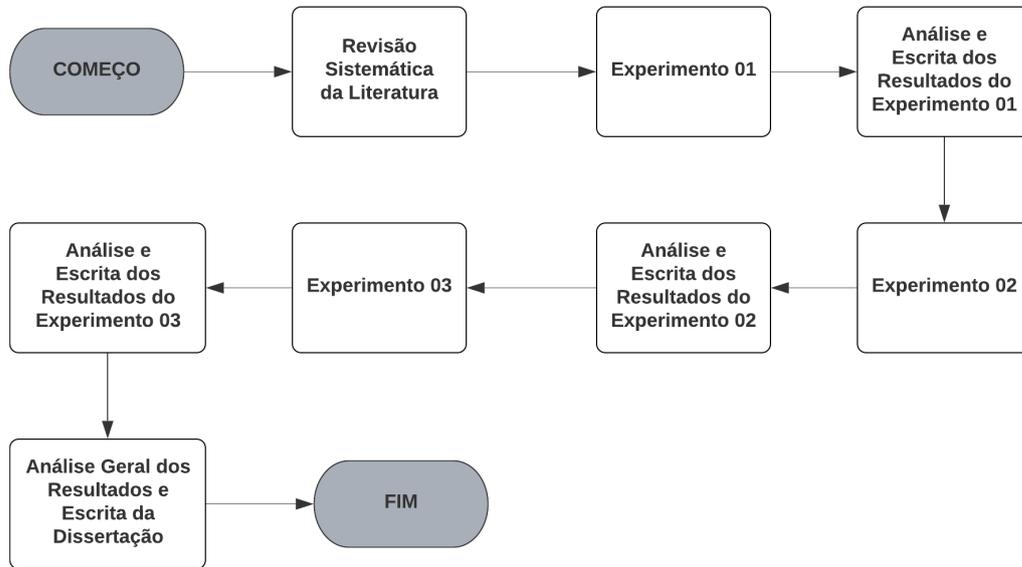
- Revisão sistemática da literatura sobre Detecção de Doenças em Plantas Utilizando Processamento Digital de Imagens e Inteligência Artificial;
- Pré-processamento utilizando os algoritmos SLIC e *augmentor* no enriquecimento das imagens;
- Criação de arquiteturas RNCs com o objetivo em interpretar imagens da folha da soja doente e sadia;
- Realizar o treinamento, teste e validação dos modelos criados utilizando as bases de dados disponíveis;
- Avaliar os modelos com as diferentes bases de dados assim como avaliar as técnicas distintas de aumento de dados utilizadas;
- Avaliar os resultados e o desempenho dos modelos na classificação das amostras não rotuladas;
- Investigar a acurácia e eficiência dos três métodos computacionais utilizados.

1.3 Metodologia de Pesquisa

Na Figura 1 é apresentado o cronograma de pesquisa utilizado neste estudo. As etapas incluem uma revisão sistemática da literatura, desenvolvimento do primeiro experimento, análise e escrita dos resultados do primeiro experimento, desenvolvimento do

segundo experimento, análise e escrita dos resultados do segundo experimento, desenvolvimento do terceiro experimento, análise e escrita dos resultados do terceiro experimento e, por fim, a análise geral dos resultados e escrita da dissertação.

Figura 1 – Fluxograma do cronograma de pesquisa.



Fonte: Autor

1.3.1 Atividades

Atividade 1: Realizar uma Revisão Sistemática da Literatura

Durante a primeira etapa do projeto de pesquisa realizou-se uma revisão sistemática de literatura a fim de obter uma perspectiva ampla em relação aos estudos conduzidos na área de interesse. O objetivo principal da revisão foi obter conhecimento das tecnologias, ferramentas e métodos utilizados pelos estudos na detecção de doenças em plantas com o uso de uma base de imagens. Também procurou-se identificar os *gaps* existentes nos estudos revisados, para então desenvolver um foco de pesquisa.

Atividade 2: Planejar e executar o primeiro experimento Uma vez que foram identificadas algumas lacunas de pesquisa, selecionou-se o problema motivado anteriormente, que possuía uma característica experimental. Assim, o projeto de pesquisa conduzido previu a divisão das possíveis contribuições em três experimentos. O primeiro experimento contou com a criação de modelos e arquiteturas de RNCs para a detecção de ferrugem na folha da soja, utilizando as imagens originais de nosso banco de dados da Embrapa. Identificando que a quantidade das imagens disponíveis seriam um limitador para o treinamento adequado de RNCs, optou-se pelo desenvolvimento e avaliação de duas técnicas de pré-processamento que são recomendadas para o enriquecimento de dados. Tais técnicas forneceram, portanto, três bases de imagens que puderam ser com-

paradas. Para tal, planejou-se desenvolver arquiteturas de RNCs em busca daqueles que apresentassem os melhores resultados para cada base de imagens.

Atividade 3: Análise e relato dos resultados do primeiro experimento

Para o primeiro experimento foi utilizado a base de dados da Embrapa. Os modelos de RNCs criados foram treinados utilizando as imagens originais do banco de dados disponível para o público em geral, mas com a diferença de que as imagens foram pré-processadas minimamente para remover os dados de fundo pela coloração preta. Portanto, é uma base sem a utilização de outra técnica ou algoritmo para o aumento de dados. Os modelos criados passaram pelas etapas de treinamento, teste e validação. Após a execução do experimento, foi realizada uma análise e comparação dos resultados entre os melhores modelos para este arranjo experimental e base de entrada.

Atividade 4: Planejar e Executar o segundo experimento Diferente do primeiro experimento, no segundo foi planejado executar um estudo utilizando além dos modelos e arquiteturas RNCs criadas, um algoritmo de pré-processamento adicional para o enriquecimento de dados. Ou seja, utilizou-se um algoritmo para aumentar o número de imagens em relação ao banco de imagens original.

Atividade 5: Análise e relato dos resultados do segundo experimento O segundo experimento foi relatado do ponto de vista do pesquisador no foco de se utilizar uma técnica para o aumento de dados denominada SLIC. Com isso, após a aplicação de aumento de dados, realizou-se também o treinamento, teste e validações dos modelos criados utilizando esta que veio à ser a segunda base de imagens. Os resultados obtidos não foram muito animadores, como constatado numa análise e comparação com os resultados obtidos no primeiro experimento.

Atividade 6: Planejar e executar o terceiro experimento O terceiro experimento utilizou uma segunda técnica como alternativa de aumento de dados. Por conta dos resultados não motivadores da técnica SLIC do segundo experimento, utilizou-se outra técnica para o enriquecimento de dados denominada *augmentor*. Portanto, o objetivo foi verificar se era possível superar os resultados alcançados no primeiro e segundo experimento com o uso desta nova técnica.

Atividade 7: Análise e relato dos resultados do terceiro experimento No terceiro experimento também foi criada arquiteturas e utilizado o algoritmo *augmentor*, o qual também visa o aumento do número de imagens. Após o aumento, foi realizado o treinamento, teste e validação dos modelos de RNCs criados, e feita uma análise dos resultados obtidos. Com os resultados gerados, foi realizada uma comparação com os resultados encontrados nos dois experimentos anteriores, chegando-se em resultados bastante animadores.

Atividade 8: Escrita da dissertação de mestrado Após as etapas anteriores, foi realizado a escrita da dissertação de mestrado. Nesta etapa foram reunidas todas as informações coletadas desde a revisão sistemática de literatura, incluindo o protocolo

de pesquisa, execução, resultados e análises dos resultados, assim como as informações resultantes dos três experimentos, incluindo protocolo, desenvolvimento, análise de dados, discussões e considerações finais. O escopo do trabalho proposto foi discutido durante o exame de qualificação, que foi reduzido para o foco e conteúdo aqui apresentados. Assim, buscou-se o melhor esclarecimento possível da revisão e experimentos conduzidos durante este mestrado.

Atividade 9: Refinamento da dissertação para trazer um estudo comparativo. Uma vez que executou-se três experimentos ao longo da pesquisa e que se dispõe de uma análise de estudos semelhantes, pretendeu-se acrescentar e incrementar a dissertação num estudo analítico comparativo. Ou seja, foi realizado e detalhado nesta dissertação uma comparação das contribuições derivadas dos três experimentos conduzidos, o que nos permite inferir qual é a melhor alternativa para treinamento e desenvolvimento de uma RNC para processamento de imagens na detecção de ferrugem em folhas de Soja.

Atividade 10: Refinar a apresentação e defender o mestrado Por fim, a última atividade foi o refinamento da apresentação para a defesa do mestrado. A mesma foi refinada e complementada com os resultados obtidos dos três experimentos e também as comparações dos mesmos. Além disso, foi detalhado nas considerações finais, os pontos positivos e negativos encontrados nesta pesquisa, assim como os trabalhos futuros que pretende-se desenvolver na evolução da pesquisa.

1.3.2 Contribuições

A Tabela 1 sumariza todos os artigos já escritos e derivados da pesquisa de mestrado. Complementarmente, a Tabela 2 discute sobre todos os materiais derivados do estudo, como o artigo publicado, algoritmos utilizados, repositório de imagens dos três experimentos, e as arquiteturas e modelos de RNCs criados.

Tabela 1 – Esforço na produção científica de materiais de divulgação de resultados

Contribuição	Tipo	Título	Nome do veículo de publicação	Estágio
Primeiro Autor	Conferência	Analysis of the Superpixel Slic algorithm for increasing data for disease detection using Deep Learning	ISDA	Aceito e Apresentado
Primeiro Autor	Revista	Systematic mapping study on disease detection in plants using image processing techniques and artificial intelligence	Ciência Rural	Em ajuste para o envio
Primeiro Autor	Conferência	Systematic Mapping on Image Processing Architectures for Disease Detection in Plants	Não definido	Em ajuste para o envio

Tabela 2 – Material Complementar Produzido ao Longo do Mestrado

Artigo publicado	O primeiro material resultante desta pesquisa de mestrado é o artigo derivado do primeiro e segundo experimento. Está disponibilizado no link < https://link.springer.com/chapter/10.1007/978-3-030-71187-0_45 >.
Algoritmos utilizados	Todos os algoritmos utilizados para conduzirmos os experimentos estão disponíveis no link < https://doi.org/10.5281/zenodo.5606157 >.
Repositório de Imagens	Esta disponível no repositório o banco de dados separadamente de cada experimento, do primeiro experimento, com as imagens originais, do segundo experimento, com as imagens alteradas pelo algoritmo SLIC, e as imagens aumentadas utilizadas no terceiro experimento. Todas as bases de dados mencionadas anteriormente estão disponíveis no link < https://doi.org/10.5281/zenodo.5598361 >.
Arquiteturas e Modelos	No repositório esta disponível os modelos utilizados nos três experimentos, assim como as arquiteturas. Os modelos e arquiteturas criadas no desenvolvimento deste estudo estão disponíveis no link < https://doi.org/10.5281/zenodo.5602151 >.

1.4 Organização do Trabalho

O restante deste trabalho está organizado da seguinte forma:

- **Capítulo 2:** apresenta o embasamento teórico deste estudo, estruturado sobre os conceitos utilizados nesta pesquisa de mestrado;
- **Capítulo 3** apresenta a revisão sistemática de literatura conduzida durante este trabalho;
- **Capítulo 4:** apresenta os experimentos conduzidos neste estudo, divididos em três etapas. Na primeira etapa, foi conduzido o primeiro experimento, na qual é detalhado na seção 4.2. Na segunda etapa, foi conduzido o segundo experimento, apresentado na seção 4.3. Por fim, na seção 4.4, é apresentado o terceiro experimento;
- **Capítulo 5:** apresenta, por fim, as considerações finais, assim como um relato dos trabalhos futuros.

2 EMBASAMENTO TEÓRICO

Este capítulo é organizado da seguinte forma: (i) conceitos de Agricultura de Precisão na Seção 2.1; (ii) Inteligência Artificial na Seção 2.2; (iii) Redes Neurais Artificiais na Subseção 2.2.2; (iv) *Deep Learning* na Subseção 2.2.3; e por fim, (v) Redes Neurais Convolucionais na Subseção 2.2.4.

2.1 Agricultura de Precisão

Existem vários conceitos de Agricultura de Precisão (AP) existentes na literatura, e alguns desses conceitos serão destacados nesta seção. Segundo (BERNARDI et al., 2014), em 2012 o Ministério da Agricultura, Pecuária e Abastecimento (MAPA), ao dar início a criação a Comissão Brasileira de Agricultura de Precisão (CBAP), veio a definir a agricultura de precisão como um sistema de gerenciamento agrícola baseada na variação espacial e temporal da unidade produtiva e visa o aumento de retorno econômico, à sustentabilidade e à minimização do efeito ao ambiente. A AP também é conceituada como uma abordagem sistêmica que tem por objetivo reorganizar o sistema agrícola a uma agricultura sustentável com a produção baixa e alta eficiência (ZHANG; WANG; WANG, 2002). O surgimento e o aprimoramento de algumas tecnologias, ajudaram a beneficiar a AP, tecnologias como o Sistema de Posicionamento Global (GPS), Sistema de Informações Geográficas (GIS), componentes de computador em tamanhos reduzidos, sensoriamento remoto em campo, controle automático, computação móvel, avanço no processamento de informações e telecomunicações (ZHANG; WANG; WANG, 2002). Com isso, a indústria agrícola é capaz de coletar dados mais precisos sobre a variabilidade da produção no espaço e tempo (ZHANG; WANG; WANG, 2002).

Segundo (BRAMLEY, 2009), a AP pode ser definida como um conjunto de tecnologias que proporciona um melhor gerenciamento da produção agrícola, tendo conhecimento em que a produtividade potencial das terras agrícolas pode variar significativamente, mesmo em distâncias pequenas (alguns metros por exemplo). A AP pode ser considerada como um meio de aumentar a probabilidade de acertos de que as estratégias corretas de manejo de culturas sejam implementadas no lugar correto e na hora certa. Com isso, torna-se notável cada vez mais a necessidade da AP no campo e nas lavouras, visando o maior rendimento possível de uma determinada área, seja ela pequena ou grande, juntamente com os cuidados com o meio ambiente (BERNARDI et al., 2014).

2.2 Inteligência Artificial

A inteligência artificial (IA) vem contribuindo com tecnologias aplicadas aos contextos da agricultura. Segundo (SAGE, 1990), a IA tem por objetivo desenvolver paradigmas ou algoritmos que necessitam de máquinas para realizar tarefas. Segundo (HAYKIN, 2007), um sistema de IA deve ser capacitado para realizar três tipos de tarefas: armaze-

nar conhecimento, aplicar o conhecimento que foi armazenado para resolver problemas, e por fim, adquirir novos conhecimentos por meio de novas experiências e resoluções de problemas. Um sistema de IA deve conter três componentes que são fundamentais: a representação, o raciocínio e a aprendizagem (SAGE, 1990).

A representação é um componente de IA que visa representar o conhecimento genérico sobre um domínio de um problema em específico, assim como o conhecimento específico sobre a solução do problema. O raciocínio, é a habilidade do sistema em resolver problemas. Por fim, na aprendizagem possui um elemento de aprendizagem que recebe informações do ambiente e, posteriormente, utiliza essas informações para aperfeiçoar a base de conhecimento, e por fim é utilizada pelo elemento de desempenho para executar tarefas. As informações fornecidas pelo ambiente normalmente são incompletas, fazendo o elemento de desempenho ficar confuso com os detalhes das informações, com isso, a máquina inicialmente opera por suposição e depois recebe realimentação do elemento de desempenho servindo para a máquina avaliar suas hipóteses e revisá-las se for necessário (SAGE, 1990).

A inteligência artificial está inserida no nosso cotidiano em vários equipamentos que utilizamos (CATH et al., 2018). No nosso caso, a IA está sendo utilizada na agricultura, com o objetivo em detectar anomalias nas plantas da soja. A IA possui diversas sub-áreas, na qual cada uma delas trata-se de um assunto diferente, uma sub-área que já foi apresentada anteriormente e que utilizamos no nosso estudo é o aprendizado de máquina, na qual será apresentado na próxima seção.

2.2.1 Aprendizado de Máquina

Segundo (PRATI, 2006), o aprendizado de máquina pode ser definido como um sistema de aprendizagem (programa, software ou algoritmo) que toma decisões levando em consideração experiências adquiridas por meio da resolução de problemas resolvidos anteriormente. Esses sistemas também possuem características distintas e semelhantes em relação a alguns pontos como linguagem de descrição, forma de aprendizagem utilizada nos mesmos, tarefa, modo e paradigma.

Pode-se dizer também que uma máquina obtém conhecimento sempre que a mesma modifica sua estrutura, programa ou dados levando em consideração suas entradas ou suas respostas a informações externas, de forma que seu desempenho esperado melhore no futuro (NILSSON, 1996). Um algoritmo de aprendizado de máquina também pode ser definido como um processo para adaptar parâmetros de uma Rede Neural Artificial (RNA) com o objetivo da mesma aprender uma função em específico. Não existe apenas um algoritmo de aprendizado de máquina em específico, e sim vários tipos de ferramentas representadas por algoritmos com suas vantagens e desvantagens. Os ajustes de pesos dos algoritmos é o que os tornam diferentes uns dos outros (BRAGA, 2000).

Outra definição de aprendizado de máquina, é que a mesma é o estudo de algo-

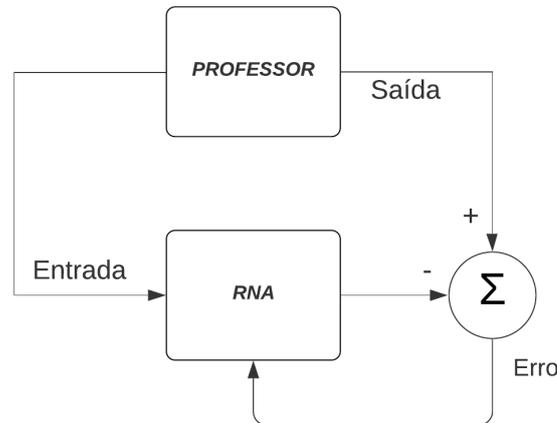
ritmos de computadores que melhoram seus desempenhos automaticamente por meio da experiência, segundo (MICHALSKI; CARBONELL; MITCHELL, 2013). Com isso, um agente é considerado um agente de aprendizado se vir a melhorar seu desempenho em novos dados e também em dados desconhecidos após ter passado por muitos exemplos de treinamento. Quando trata-se de programas (modelos) de aprendizado, deve-se sempre testar a capacidade de generalização do mesmo, utilizando dados que o algoritmo não reconhece (dados de teste). Caso o teste não venha a ser realizado, cria-se a impressão de que o modelo possui um desempenho ideal apenas acessando os dados salvos no treinamento.

Existem vários tipos de métodos de treinamento de redes e, na sua maioria, podem ser classificados em dois principais tipos de paradigmas: o aprendizado supervisionado e o aprendizado não-supervisionado (BRAGA, 2000). No presente estudo utilizamos o método de aprendizado supervisionado. Porém, para um melhor entendimento da diferença entre os dois paradigmas será introduzido os conceitos sobre as duas abordagens nas subseções seguintes.

2.2.1.1 Aprendizado Supervisionado

Neste tipo de aprendizado a sua característica principal é a existência de um supervisor externo que fornece os dados de entrada e saída para a rede que será treinada. O objetivo principal neste tipo de aprendizado é ajustar os parâmetros da rede buscando encontrar uma ligação entre os pares de entrada e saída fornecidos. Na Figura 2 é apresentado um exemplo de aprendizado supervisionado, na qual existe um supervisor (professor) que analisa o andamento do treinamento observando se o mesmo está indo bem ou mal e assim direciona o processo. O supervisor analisa as saídas do treinamento por meio de uma comparação da saída do processo com a saída desejada e em caso de erros, ajusta-se os pesos visando minimizar os erros se possível. A cada etapa são realizados ajustes se necessário, assim a diferença vai diminuindo aos poucos e incrementalmente o treinamento vai chegando a uma solução, se houver (BRAGA, 2000).

Figura 2 – Exemplo de aprendizado supervisionado.



Fonte: modificado de (BRAGA, 2000)

É utilizado como medida de desempenho da rede a soma dos erros quadráticos de todas as saídas de cada etapa, assim como um valor a ser minimizado pelo modelo de treinamento. Uma das desvantagens encontradas no aprendizado supervisionado é que se não houver um supervisor, a rede não irá se atualizar com novas estratégias, utilizando apenas os exemplos já contidos no treinamento da rede. Existem dois tipos de aprendizado supervisionado que podem ser implementados: *off-line* e *on-line*. De forma *off-line*, os dados que fazem parte do banco de dados de treinamento não são modificados, atualizados, ou extraídos, assim o resultado (solução) final do treinamento torna-se fixo sem mudanças futuras. Já na forma *on-line*, os dados que compoem a base de dados de treinamentos podem ser alterados, sendo adicionado ou retirado novos dados, assim o treinamento é feito toda vez que ocorre uma mudança nos dados de treinamento, podendo alterar assim a solução final (BRAGA, 2000).

2.2.1.2 Aprendizado Não-Supervisionado

No aprendizado não-supervisionado, diferente do supervisionado, não existe um supervisor para acompanhar o processo de treinamento e aprendizado. Na Figura 3 é apresentado um exemplo ilustrativo de aprendizado não-supervisionado. Diferente do aprendizado supervisionado que possui no conjunto de treinamento pares de entrada e saída, no aprendizado não-supervisionado apenas os padrões de entrada estão a disposição da rede. Assim que a rede consegue abstrair as regularidades estatísticas da entrada de dados, é desenvolvido uma capacidade em criar representações internas com o objetivo em codificar as características de entrada para criar classes ou grupos novos de forma automática. O aprendizado não-supervisionado só pode ocorrer quando se tem dados redundantes como entrada, caso contrário, a rede não encontra os padrões ou características

dos dados de entrada (BRAGA, 2000).

Figura 3 – Exemplo de aprendizado não-supervisionado.



Fonte: modificado de (BRAGA, 2000)

2.2.1.3 Avaliação de Aprendizagem

As métricas utilizadas em nossos experimentos, resumem-se na utilização da matriz de confusão. A matriz de confusão é uma tabela que mostra os erros e acertos dos modelos, e serve de base para a organização dos resultados. Preenchendo a matriz é possível utilizar diversas fórmulas que resultam em diversas análises e resultados. Um exemplo genérico da matriz de confusão é apresentado na Tabela 3.

Tabela 3 – Exemplo de matriz de confusão.

	Positivo	Negativo
Positivo	Verdadeiros Positivos	Falsos Negativos
Negativo	Falsos Positivos	Verdadeiros Negativos

Como mostra na Tabela 3, temos quatro tipos de resultados: verdadeiro positivo (VP), falso negativo (FN), falso positivo (FP) e verdadeiro negativo (VN), na qual são especificados abaixo:

- **VP**: classificação correta da classe positivo;
- **FN**: erro em que o resultado foi negativo quando o valor esperado era positivo;
- **FP**: erro em que o resultado foi positivo quando o valor esperado era negativo;
- **VN**: classificação correta da classe negativo;

As opções de resultados utilizadas neste trabalho são especificadas abaixo:

- **VP**: a entrada é uma imagem da folha doente e a RNC tem como saída positivo para folha doente;

- **FN**: a entrada é uma imagem da folha doente e a RNC tem como saída negativo para folha doente;
- **FP**: a entrada é uma imagem da folha saudável e a RNC tem como saída positivo para folha doente;
- **VN**: a entrada é uma imagem da folha saudável e a RNC tem como saída negativo para folha doente.

A partir da matriz de confusão foi possível gerar fórmulas para medir a precisão de nossos algoritmos, apresentadas a seguir. Segundo (SOKOLOVA; JAPKOWICZ; SZPAKOWICZ, 2006), a medida empírica mais utilizada para medir a eficácia dos algoritmos de aprendizado de máquina é a acurácia. A acurácia “nos diz” o quanto nosso modelo acertou as previsões possíveis. A fórmula da acurácia é apresentada abaixo:

$$acurácia = \frac{VP + VN}{VP + FP + VN + FN} \quad (2.1)$$

Outra fórmula que foi utilizada é a taxa de erro, que é apresentada a seguir:

$$taxa\ de\ erro = 1 - acurácia \quad (2.2)$$

De acordo com (SOKOLOVA; JAPKOWICZ; SZPAKOWICZ, 2006), usar apenas a acurácia e a taxa de erro para analisar os resultados de um algoritmo de aprendizado de máquina não é totalmente eficaz. Para que a análise seja realizada com maior profundidade e detalhamento, é possível utilizar mais duas fórmulas nas quais também se utilizam os dados contidos na matriz de confusão. As duas fórmulas são:

$$sensibilidade = \frac{VP}{VP + FN} \quad (2.3)$$

$$especificidade = \frac{VN}{VN + FP} \quad (2.4)$$

Por fim, a última fórmula utilizada para obter os resultados finais entre as fórmulas de sensibilidade e especificidade é apresentada a seguir:

$$eficiência = \frac{sensibilidade + especificidade}{2} \quad (2.5)$$

Essas métricas e avaliações de aprendizagem foram utilizadas em todos os experimentos deste estudo, possibilitando uma avaliação segura e eficaz dos nossos modelos de RNCs criados.

2.2.2 Redes Neurais Artificiais (RNA)

As RNAs foram criadas tendo como principal referência a rede neural humana. Segundo (HAYKIN, 2007), uma RNA é uma máquina que foi construída com propósito

em modelar a maneira na qual o cérebro humano resolve seus problemas, uma tarefa em particular ou alguma função de seu interesse. Normalmente as RNAs são projetadas de duas formas: utilizando componentes eletrônicos ou por meio de simulação utilizando linguagens de programação em computadores digitais. Segundo (BRAGA, 2000), as RNAs são sistemas paralelos distribuídos formados por um conjunto de nodos (unidades de processamento simples) com o propósito de calcular funções matemáticas, que na sua maioria, são funções não lineares. Esses nodos são adicionados em uma ou mais camadas e interligados por um grande número de conexões, feitas de modo intercalado. Na maioria das arquiteturas de redes, essas conexões são conhecidas como pesos, que armazenam o conhecimento disponibilizado pelo modelo. Esses pesos também servem para ponderar as informações recebidas pelas entradas dos neurônios. A inspiração para a criação destas RNAs vem da estrutura física conhecida como cérebro humano.

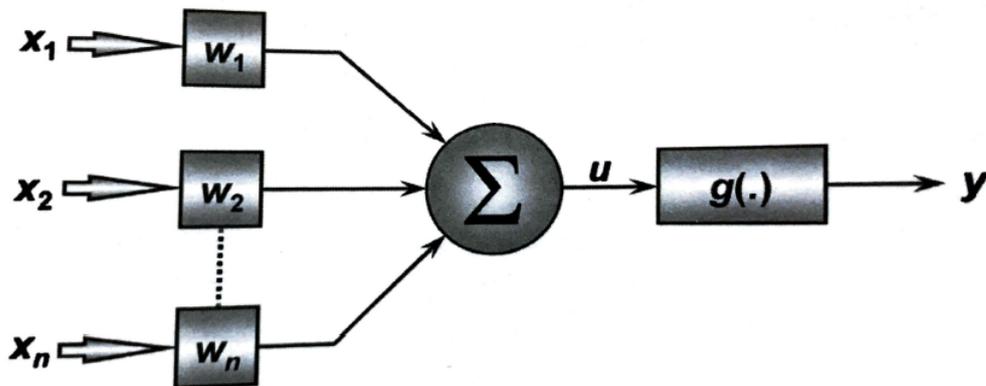
Outra definição apresentada por (HAYKIN, 2007) é que uma rede neural pode ser definida como um processador composto por um grande número de unidades de processamento simples, paralelos e distribuídos. Este processador deve ter a capacidade natural de adquirir conhecimento por meio da experiência e tornar o mesmo disponível para uso, quando necessário. As RNAs possuem dois pontos na qual se assemelham ao cérebro humano. O primeiro está relacionado ao seu conhecimento que é adquirido através de seu ambiente por meio de um processo de aprendizagem. O segundo ponto são os pesos sinápticos (forças de conexão entre neurônios), os quais são utilizados para guardar o conhecimento adquirido (HAYKIN, 2007).

Segundo (SILVA; SPATTI; FLAUZINO, 2010), as RNAs são modelos computacionais com base no sistema nervoso humano, e são capazes de adquirir e manter o conhecimento (com base em informações). As RNAs podem ser descritas como um conjunto de unidades de processamento (chamados de neurônios artificiais) ligados por um grande número de interconexões (chamadas de sinapses artificiais), sendo conhecidas também no estudo de RNAs, como vetores/matrizes de pesos sinápticos (SILVA; SPATTI; FLAUZINO, 2010).

2.2.2.1 Neurônio Artificial

Os neurônios artificiais desenvolvidos para as RNAs são simples comparados aos neurônios biológicos. São modelos não lineares, com saídas ininterruptas, os quais realizam funções basicamente simples. As funções na qual esses modelos realizam incluem captar sinais que estão em suas entradas, abstraí-los conforme a respectiva função operacional, e assim gerar um resultado/resposta, que pode mudar conforme a função de ativação. Um exemplo ilustrativo de um neurônio artificial é apresentado na Figura 4.

Figura 4 – Neurônio artificial



Fonte: Adaptado de (SILVA; SPATTI; FLAUZINO, 2010)

Os sinais de entrada vindos do meio externo são representados pelo conjunto X_1, X_2, \dots, X_n , que no neurônio biológico, são os impulsos elétricos captados pelos dendritos. Esses sinais de entrada X_i são executados através de suas multiplicações pelos pesos sinápticos específicos W_i , ponderando assim todos os dados externos que o neurônio captura como entrada (SILVA; SPATTI; FLAUZINO, 2010). O neurônio artificial possui em sua arquitetura seis elementos básicos, nas quais podem ser apresentados, segundo (SILVA; SPATTI; FLAUZINO, 2010), como:

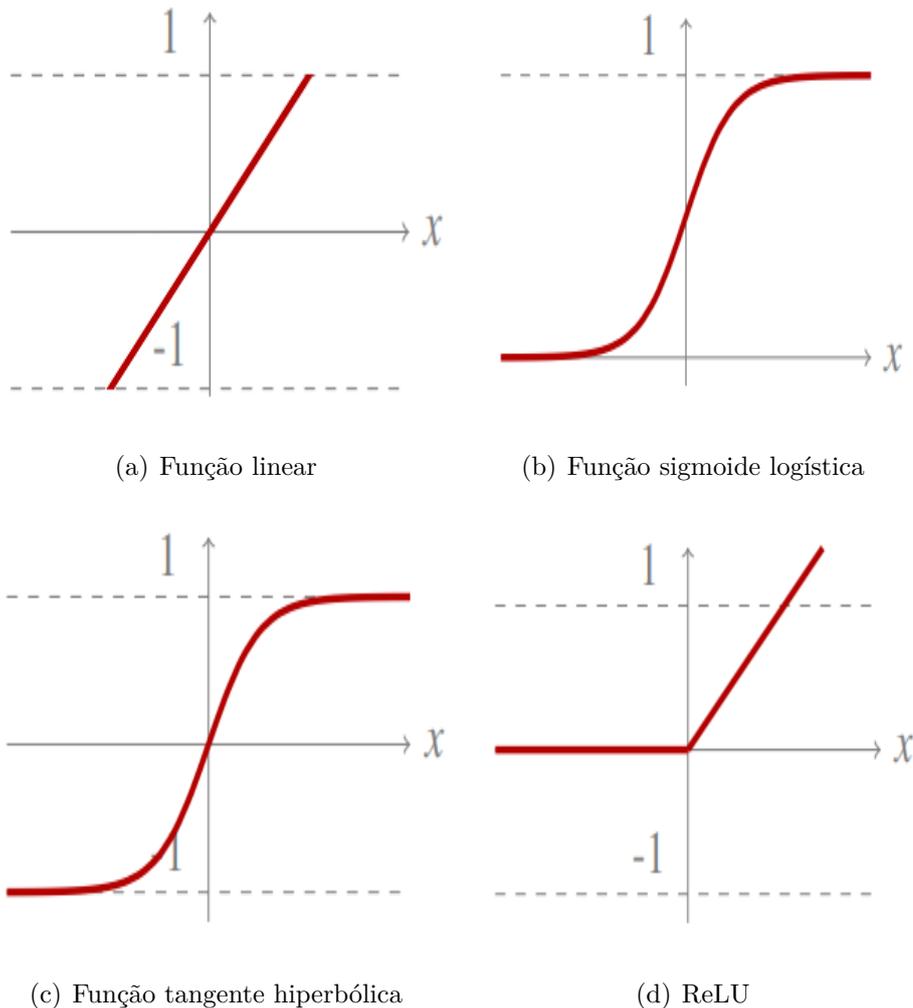
- **Sinais de entrada:** na Figura 4, os sinais de entrada vindas do meio externo são representados pelo conjunto X_1, X_2, \dots, X_n , na qual caracterizam os valores assumidos pelas variáveis de um determinado programa. Geralmente, todos os sinais de entrada são normalizados, para que se possa obter uma melhor eficiência computacional na execução de algoritmos de aprendizagem;
- **Pesos sinápticos:** representados pelo conjunto W_1, W_2, \dots, W_n , conforme mostra a Figura 4. Os pesos sinápticos são responsáveis pela ponderação de cada entrada da rede, quantificando suas relevâncias levando em consideração as funções do neurônio em específico;
- **Combinador linear:** representado pelo símbolo Σ na Figura 4, tem por objetivo receber os sinais de entrada ponderados pelos pesos sinápticos e criar um valor de potencial de ativação;
- **Potencial de ativação:** representado pela letra u na Figura 4, é o resultado do combinador linear. Dependendo do valor produzido, o neurônio pode ou não enviar o resultado para a função de ativação.

- **Função de ativação:** representado por $g(\cdot)$ na Figura 4, tem o objetivo de limitar a saída do neurônio em um intervalo, que pode mudar dependendo do tipo de função de ativação;

- **Sinal de saída:** representado pela letra y na Figura 4, é o resultado final do neurônio, produzido pelo mesmo, que tem como entrada um conjunto de sinais. O sinal de saída pode ser utilizado como entrada em outros neurônios, interligados na sequência.

As funções de ativação, representadas na Figura 4 por $g(\cdot)$ servem como uma ferramenta para o neurônio decidir se o valor (representado por u) resultante do somador (representado por Σ) será ativado ou não. Com isso, a escolha da função de ativação adequada para cada problema em específico é de suma importância. Dentre as várias funções de ativação existentes, na Figura 5 é apresentado as três principais funções de ativação utilizadas: função linear, função sigmoide logística e função tangente hiperbólica. Também é apresentado na Figura 5 a função ReLU. A função Linear, apresentada na Figura 5 (a), tem em sua saída valores entre -1 e 1, mantendo os resultados de saída iguais aos valores do potencial de ativação u . Ilustrada na Figura 5 (b), a função logística, assume valores entre 0 e 1. Já a função tangente hiperbólica, diferente da função logística, terá em sua saída valores entre -1 e 1, representada na Figura 5 (c). Por fim, a função de ativação apresentada na Figura 5 (d) é a ReLU. A função retificadora linear (*rectified linear function*, ReLU), segundo (PONTI; COSTA, 2017), tem sido muito utilizada atualmente em estudos envolvendo *Deep Learning*, e será abordada com mais detalhes na Seção 2.2.4.

Figura 5 – Funções de ativação



2.2.2.2 Arquiteturas de RNAs

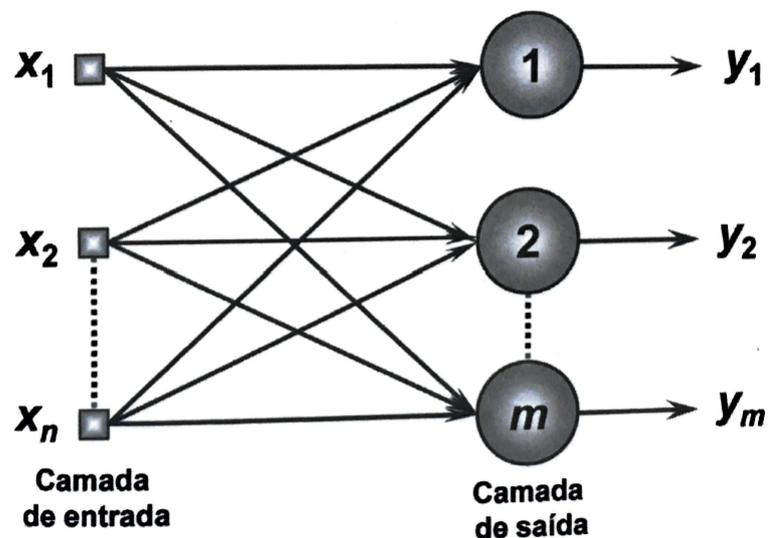
As arquiteturas de redes neurais artificiais definem a forma na qual os neurônios estão arranjados, quais os tipos e quantas camadas possuem e quais conexões a arquitetura adota, assim como a direção dos sinais sinápticos conduzidos por essas conexões. De forma básica, uma arquitetura de RNA pode ser dividida em três camadas. A camada de entrada que recebe os dados e informações do meio externo. A camada intermediária (também conhecida como oculta ou escondida) é uma camada composta por neurônios, os quais tem por objetivo extrair as características do sistema ou processo proposto. Nessa camada a maior parte do processamento interno da rede é executado. Por fim, a camada de saída, também composta por neurônios, tem por objetivo produzir o resultado final da rede, possui como entrada as informações resultantes dos neurônios da(s) camada(s) anterior(es) (SILVA; SPATTI; FLAUZINO, 2010).

As principais arquiteturas de RNAs podem ser classificadas em redes *feedforward* de camada simples, redes *feedforward* de múltiplas camadas, redes recorrentes e redes

reticuladas. Para fins introdutórios, vamos adotar apenas as redes *feedforward* de camada simples e de múltiplas camadas (SILVA; SPATTI; FLAUZINO, 2010).

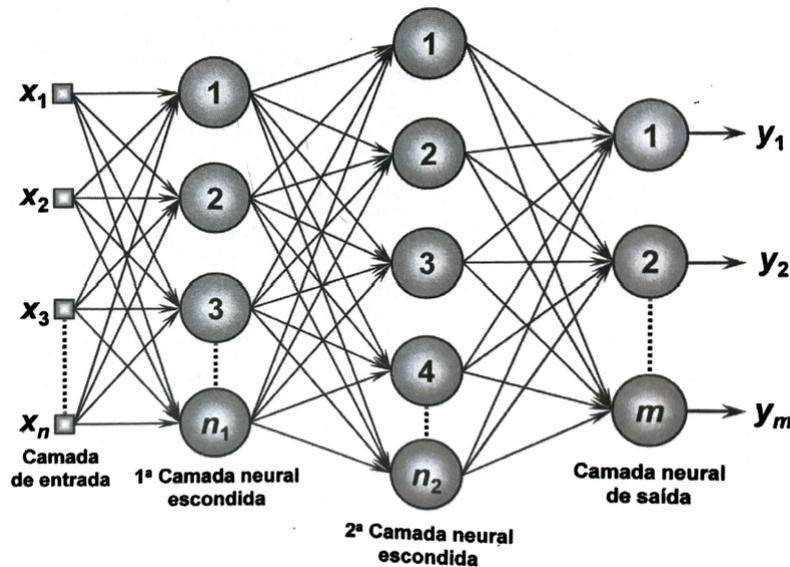
Na Figura 6 é apresentado um exemplo ilustrativo da rede *feedforward* de camada simples. Este tipo de rede é basicamente composta por uma única camada de entrada e apenas uma camada de saída, composta por neurônios. Como mostra a Figura 6, a rede possui n entradas e m saídas.

Figura 6 – Exemplo de rede *feedforward* de camada simples



Fonte: (SILVA; SPATTI; FLAUZINO, 2010)

Na Figura 7 é apresentado um exemplo ilustrativo de rede *feedforward* de camadas múltiplas. Este tipo de rede, diferente da rede de camada única, possui mais camadas. A rede *feedforward* de camadas múltiplas possui uma camada de n sinais de entrada, duas camadas escondidas de neurônios, constituída por $n1$ e $n2$ neurônios. Por fim, uma camada neural de saída, constituída por m neurônios, que representam os valores de saída da aplicação (SILVA; SPATTI; FLAUZINO, 2010).

Figura 7 – Exemplo de rede *feedforward* de múltiplas camadas

Fonte: (SILVA; SPATTI; FLAUZINO, 2010)

2.2.3 (*Deep Learning*)

A Aprendizagem Profunda, também conhecido como *Deep Learning* (DL), do inglês, é considerado como uma extensão ou subconjunto de metodologias de aprendizado de máquina (AM) clássico que utilizam RNAs em específico (GULLI; PAL, 2017). O DL possui mais “profundidade” e complexidade nos seus modelos quando comparados ao AM clássico, transformando os dados utilizando várias funções, tornando assim acessível a representação dos dados de forma hierárquica utilizando para isso vários níveis de abstração (KAMILARIS; PRENAFETA-BOLDÚ, 2018).

Segundo (DENG; YU, 2014), o DL surgiu como uma nova área de pesquisa em AM. Dois pontos importantes foram destacados por vários autores, na qual descreveram o DL de forma semelhante. As duas definições dadas pelos autores destacam que o DL são modelos que são compostos por várias camadas ou estágios de processamento de informações não lineares. O segundo ponto destaca que o DL são métodos para aprendizagem supervisionada e não supervisionada de representação de traços em camadas sucessivamente mais altas e mais abstratas.

Segundo (GOODFELLOW et al., 2016), o DL introduz representações que são expressadas por meio de outras representações mais simples. O DL possibilita que computadores consigam criar conceitos complexos utilizando conceitos mais simples. De uma forma geral, o DL é uma abordagem à inteligência artificial, sendo um tipo de AM que tem por objetivo dar possibilidade de sistemas computacionais serem aprimorados com a experiência e os dados. A AM é a única possibilidade viável no momento de alcançar um sistema de IA que possa operar em ambientes do mundo real. Assim, o DL sendo um

tipo específico de AM, pode atingir grande poder e flexibilidade na sua execução, como aprender e representar informações do mundo como uma hierarquia composta de conceitos nas quais cada um desses conceitos são definidos tendo como parâmetro conceitos mais simples e representações abstratas computadas por outras que são ainda menos abstratas (GOODFELLOW et al., 2016).

Para aplicar o DL existem vários métodos na qual podem ser utilizados. A escolha do método está relacionado ao problema em específico, como também, o tipo de dado, se o aprendizado é supervisionado ou não, assim como o tipo de problema a ser resolvido. Dentre alguns métodos mais conhecidos de DL, temos as redes neurais convolucionais, na qual serão detalhadas com detalhes na seção a seguir.

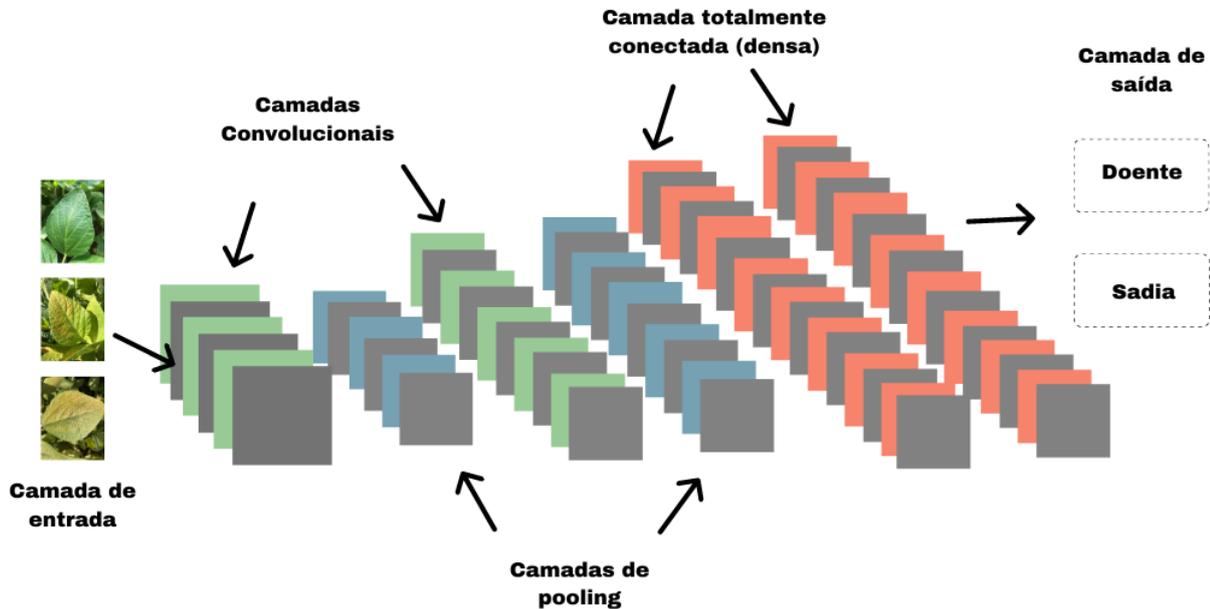
2.2.4 Redes Neurais Convolucionais (RNC)

Segundo (ARAÚJO et al., 2017), as RNCs são consideradas um método de visão computacional, na qual é mais fácil de realizar o treinamento quando se tem uma quantidade consideravelmente grande de amostras rotuladas que representam as diferentes classes de um banco de dados. As RNCs possuem algumas vantagens como a habilidade em extrair características relevantes por meio do aprendizado de transformações, denominados *kernels*. Outro ponto favorável é que as RNCs dependem de um número menor de parâmetros de ajustes do que redes totalmente conectadas com o mesmo número de camadas ocultas. Assim, cada unidade de uma camada não é conectada com todas as unidades da camada subsequente, resultando em um número menor de pesos a serem atualizados, facilitando assim o treinamento (ARAÚJO et al., 2017; KRIZHEVSKY; SUTSKEVER; HINTON, 2012).

Segundo (PONTI; COSTA, 2018), as RNCs são um dos modelos de *Deep Learning* mais utilizados e conhecidos. Uma das principais características que definem este modelo é sua estrutura ser composta por camadas convolucionais, camadas nas quais servem para processar as entradas considerando campos receptivos locais. Outra característica de suma importância das RNCs são as operações chamadas *pooling*, que tem por objetivo reduzir as dimensões espaciais das representações. A principal área de utilização das RNCs são no processamento de informações visuais, especificadamente as imagens. Com a utilização da convolução é possível filtrar as imagens levando em consideração a estrutura espacial (PONTI; COSTA, 2018).

Um modelo RNC é composto por camadas que possuem variáveis que podem ser alteradas, possibilitando a mudança do resultado final dos testes e validações. As camadas de uma RNC são: camada de entrada, camada convolucional, camada de pool, camada densa e camada de saída. Um exemplo de arquitetura RNC com sete camadas, entre convoluções e totalmente conectada (densa) é apresentado na Figura 8.

Figura 8 – Exemplo de arquitetura RNC.



Fonte: adaptado de (ARAÚJO et al., 2017)

O número de camadas de uma arquitetura RNC podem ser alteradas, o modelo apresentado na Figura 8 é apenas um exemplo, na qual pode ser alterado. Muitos modelos na atualidade utilizam várias camadas convolucionais, de *pooling* e densas. A camada convolucional tem por objetivo extrair informações dos dados de entrada. Após a camada convolucional, os resultados passam pela camada de *pooling*, que reduz a dimensionalidade dos dados. A camada totalmente conectada tem por objetivo a propagação do sinal utilizando multiplicação ponto a ponto e também a utilização de uma função de ativação. A saída de uma RNC é igual a probabilidade da imagem de entrada ser de umas das classes na qual a rede está sendo treinada (ARAÚJO et al., 2017).

Uma breve introdução e detalhamento de cada uma das camadas é apresentado a seguir:

i. Camada de entrada:

Como as RNCs normalmente são utilizadas para o processamento de informações visuais, como imagens, a camada de entrada é responsável por receber essas imagens, as quais são representadas por matrizes. Recebendo estes padrões, posteriormente são apresentados para a RNC processar. Este processo é realizado tanto no treinamento

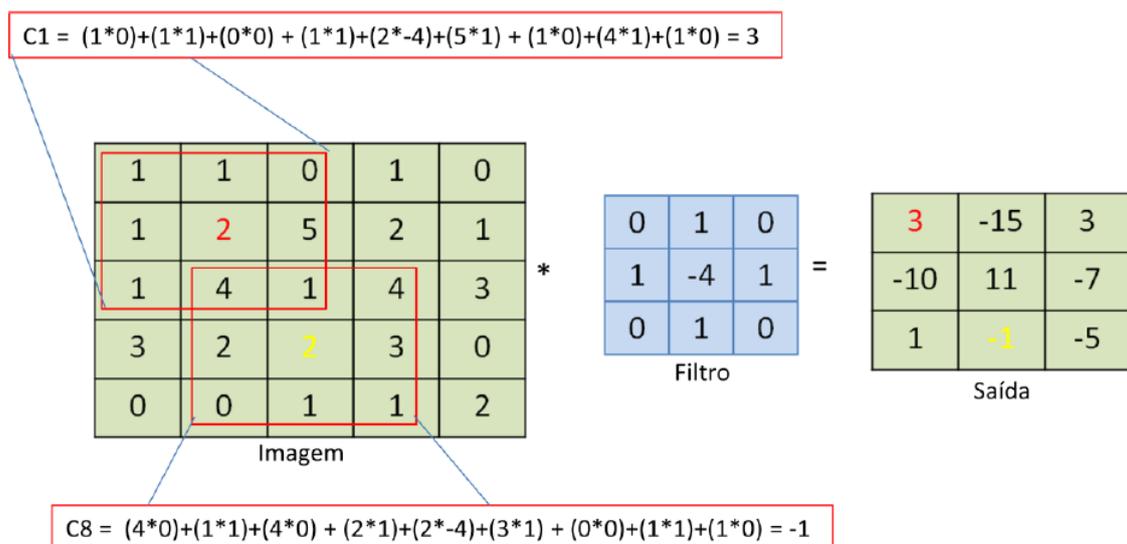
quanto nos testes e validações da RNC.

ii. Camada Convolutacional:

A camada convolutacional recebe os dados de entrada (no nosso exemplo as imagens) e utiliza filtros de aprendizado. Cada neurônio é um desses filtros aplicado a imagem de entrada, e cada filtro representa uma matriz de pesos. Esses filtros correspondem a partes das imagens, ou regiões, e o conjunto desses filtros cobrem toda a imagem. Cada filtro (neurônio) da camada convolutacional, irá processar a imagem resultando em uma transformação desta imagem utilizando uma combinação linear dos pixels vizinhos. Os filtros são gerados por conta de uma região chamada campo receptivo local e o resultado da aplicação dos filtros são chamados de mapas de características (ou *feature maps*).

Um mapa de características é obtido por meio da convolução da matriz de entrada (por exemplo, uma imagem) por um filtro linear. Uma das características principais de uma RNC é possuir ao menos uma camada de convolução, para ser considerada assim, uma RNC. Um exemplo de um tipo de processo de convolução que pode ser utilizado nas camadas convolucionais é apresentado na Figura 9. Neste exemplo apresentado temos como entrada uma matriz de dimensões 5x5 (uma imagem por exemplo) e um filtro específico de dimensões 3x3, que é utilizado para a detecção de bordas das imagens. Os filtros tem por objetivo detectar informações e características dos dados de entrada. É possível ter diferentes filtros que resultarão em diferentes mapas de características para um mesmo tipo de entrada. Na saída, como mostrado na Figura 9, é obtido uma matriz denominada mapa de características.

Figura 9 – Exemplo de convolução.



Fonte: (FARIA, 2018)

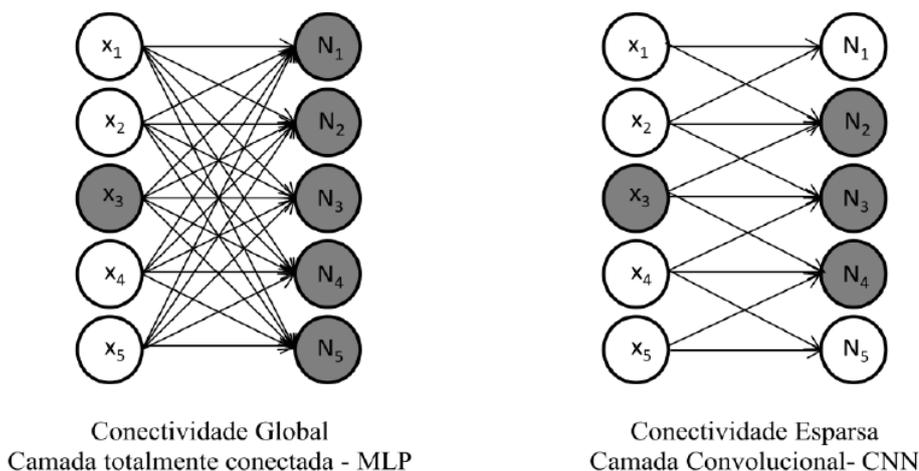
Como mostra a Figura 9, na convolução da imagem de entrada o filtro desliza um pixel por toda a imagem. Para cada janela que desliza sobre a imagem, o produto de cada

elemento do filtro (também conhecido como *kernel*) e o elemento da matriz de entrada na qual a janela sobreposição é calculado e os resultados passam por uma soma para se obter a saída que corresponde aquela localização. Como resultado final desta etapa é obtido o mapa de características.

Segundo (FARIA, 2018), as RNCs possuem características que contrastam com certos paradigmas que fazem parte dos modelos de arquiteturas rasas, como por exemplo as redes neurais de múltiplas camadas (MLPs). São elas: conectividade esparsa, campos receptivos locais e compartilhamento de parâmetro.

- **Conectividade esparsa:** este tipo de conectividade acontece por conta do filtro (*kernel*), que é utilizado na convolução, apresentar dimensões muito menores que a matriz de entrada. Isto possibilita uma correlação espacial local através da aplicação de um padrão de conectividade local entre neurônios de camadas adjacentes. Na Figura 10, é apresentado os dois tipos de conectividade, global e esparsa, para os dois tipos de arquiteturas, MLP e RNC. Nas arquiteturas de rede MLPs, cada neurônio da camada de saída tem uma ligação (interação) com cada neurônio da camada de entrada por meio do processo de multiplicação de matrizes, que gera uma conectividade global entre os parâmetros. Na Figura 10 é possível analisar a entrada x_3 e notar que quando o modelo faz uso da multiplicação de matrizes (conectividade global na MLP) todas as unidades de saída, como N_1 a N_2 , são afetadas pela entrada x_3 . Por outro lado, quando o modelo utiliza camada convolucional (utilizando um *Kernel* de tamanho 3 por exemplo), apenas três unidades da camada de saída (N_2 , N_3 e N_4) possuem correlações com a entrada x_3 .

Figura 10 – Exemplos de conectividade global e esparsa.

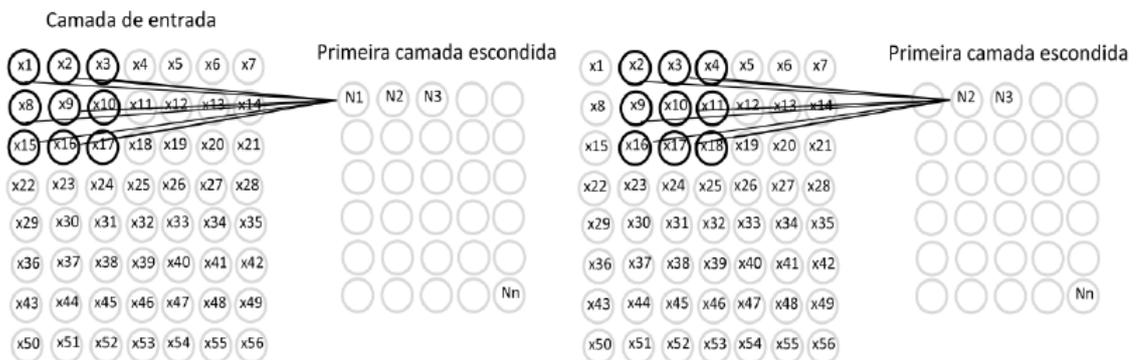


Fonte: (GOODFELLOW et al., 2016)

- **Campo receptivo local:** como já mencionado anteriormente, que um pequeno quadrado em uma imagem pode ser considerado uma região. Cada região na ma-

triz de entrada representa o tamanho do filtro utilizado e corresponde a um único neurônio na camada escondida. Cada um dos deslocamentos da região pela matriz de entrada significa um neurônio na camada escondida. Na Figura 11 é apresentado um exemplo de campo receptivo local em uma RNC. Na Figura 11 temos neurônios de entrada representados de x_1 a x_{17} , destacados em preto, o neurônio N_1 da primeira camada escondida e um filtro de convolução de tamanho 3×3 . Com isso, é possível notar que apenas a região em destaque influencia o neurônio de saída N_1 . Esta região destacada é chamada de campo receptivo local do neurônio de saída N_1 .

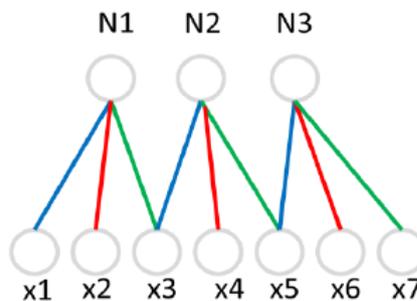
Figura 11 – Exemplo de campo receptivo local nas RNCs.



Fonte: (NIELSEN, 2015)

- **Compartilhamento de parâmetros:** os filtros utilizados na camada convolucional são utilizados em diferentes locais da matriz de entrada. Isto resulta em que os padrões que ocorrem com frequência na matriz de entrada e que estejam localizados em qualquer parte da entrada possam ser aprendidos (HAFEMANN et al., 2015). Na Figura 12 é apresentado um exemplo de compartilhamento de parâmetro, e como o mesmo funciona nas camadas convolucionais. Conexões da mesma cor representam pesos, também conhecidos como elementos do *kernel*, sendo compartilhados.

Figura 12 – Exemplo de compartilhamento de parâmetros.



Fonte: (FARIA, 2018)

Segundo (GOODFELLOW et al., 2016), o uso das conexões esparsas, campos receptivos locais e compartilhamento de parâmetros, permitem as RNCs armazenarem menos parâmetros, resultando assim na redução dos requisitos de memória do modelo e também na melhora da sua eficiência estatística. Outro ponto é que a geração de resultados obtidos necessita de menos operações, possibilitando uma melhoria na eficiência destes modelos de rede.

Após o processo de convolução em uma matriz de entrada em conjunto com a adição de um termo de *bias*, os resultados obtidos são utilizados como entrada para uma função de ativação linear. Segundo (JARRETT et al., 2009; NAIR; HINTON, 2010; GLOROT; BENGIO, 2010; KRIZHEVSKY; SUTSKEVER; HINTON, 2017) a melhor função de ativação para utilizar nos modelos RNCs é a função ReLu apresentada na Equação 2.6. Nesta equação, z representa um valor de entrada de um neurônio, e a saída esperada desta função é o valor máximo entre 0 e a entrada.

$$f(z) = \max(0, z) \quad (2.6)$$

Outro ponto importante que deve ser mencionado são os filtros utilizados nas camadas convolucionais, o *wide* ou *narrow convolution* e o *stride*. O primeiro filtro define a forma na qual deve-se aplicar o filtro, dada uma matriz de dados por exemplo, como aplicar o filtro aos elementos que não possuem vizinhos, elementos na borda da matriz. Uma abordagem que é usada de modo frequente é a utilização de *padding* (preenchimento) que tem por objetivo expandir a matriz de entrada. Em casos em que utiliza-se *padding* (quando tem adição de elementos na borda da matriz de entrada com valor zero), este hiperparâmetro é chamado de *wide convolution*. O mesmo também permite aplicar o filtro em todos os elementos da matriz de entrada. Diferente do primeiro caso, quando não é utilizado o *padding*, o hiperparâmetro é chamado de *narrow convolution*. Já o *stride* (ou passo), é outro hiperparâmetro a ser definido, o qual também pode ser utilizado nas camadas de *pooling*. O mesmo é responsável por definir qual o tamanho do deslocamento do filtro em cada passo. Normalmente é utilizado em reconhecimento de imagens um *stride* de tamanho igual a 1, significando que o filtro é deslocado pela imagem um pixel por vez.

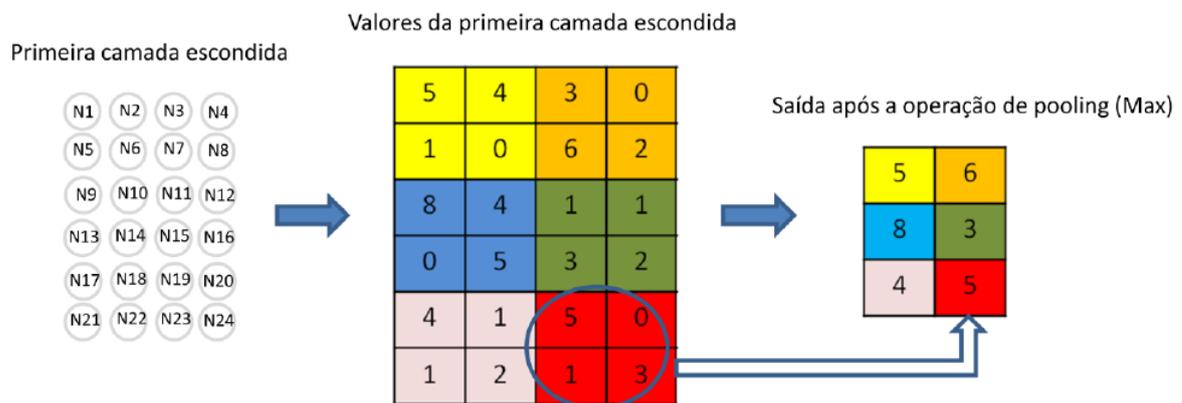
iii. Camada de *Pooling*:

Usualmente, após a camada convolucional, utiliza-se a camada de *pooling*. Camada na qual tem por objetivo extrair as informações mais importantes e significativas obtidas nas camadas convolucionais. Segundo (GOODFELLOW et al., 2016), as camadas de *pooling* alteram a saída das camadas convolucionais em uma determinada região por uma estatística reduzida das saídas mais próximas, podendo assim, reduzir o número de neurônios da camada anterior. Uma das funções de *pooling* que podem ser utilizadas é o operador *max*, que é utilizado para reportar o valor máximo de uma determinada região. Outras funções utilizadas são a média, a norma L2 ou a média ponderada da região local

analisada.

Uma das características importante do *pooling* é que o mesmo fornece uma matriz de saída de tamanho fixo. Por exemplo, uma aplicação que utiliza 100 filtros nas camadas de convolução, utilizando o *pooling* para cada filtro, o resultado será um vetor de saída de dimensão 100, independente do tamanho do filtro ou do tamanho da entrada. Nas aplicações que envolvem imagem, o *pooling* também disponibiliza invariância a translação e rotação, isso significa que, independente da operação realizada na imagem, o operador de *max* selecionará sempre o mesmo valor (valor máximo da região analisada). Na Figura 13 é apresentado o operador máximo (*max*) para uma matriz de saída tamanho 4x6 com uma janela de tamanho 2x2.

Figura 13 – Exemplo de *pooling* utilizando o operador *max*, com filtro 2x2 e *stride* igual a 2.



Fonte: (FARIA, 2018)

Outro ponto importante a ser destacado, que pode ser visto na Figura 13, é que a saída da camada convolucional é composta de 4x6 neurônios, e que após a operação feita pelo *pooling*, restou apenas uma saída de 2x3 neurônios. Uma redução de informações significativas, ficando apenas aqueles dados com as melhores características. Diminuindo o número de características, também se reduz o número de parâmetros necessários para as camadas posteriores.

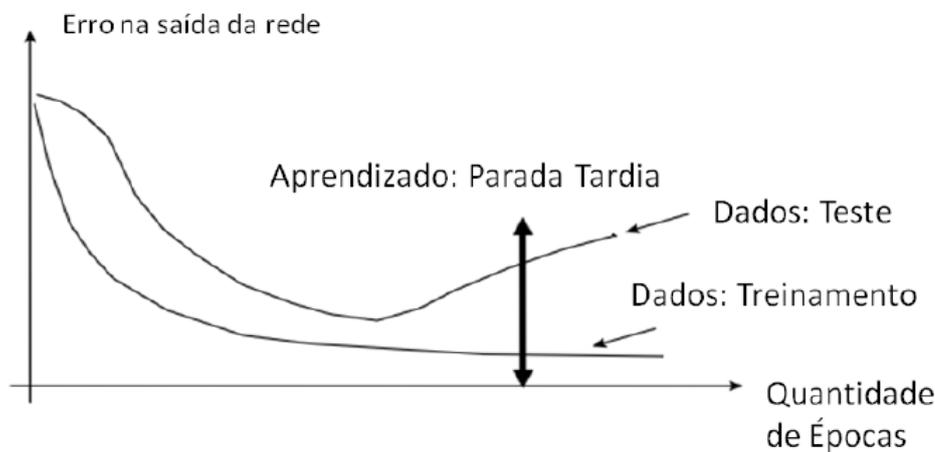
iv. Dropout:

As RCNs são constituídas por camadas alternativas de convolução e camadas de *pooling*, finalizando com camadas totalmente conectadas (*fully connected*). Fazendo uma comparação com as RNCs e as redes MLPs tradicionais (arquiteturas rasas) é correto afirmar que as RCNs possuem um número inferior de conexões e parâmetros por conta da presença de conectividade local e armazenamento de filtros. Porém, uma aplicação que utiliza desta arquitetura pode conter inúmeras camadas de processamento aumentando assim substancialmente a quantidade de parâmetros da rede. Com isso, com maior complexabilidade da RNC, pode ocorrer problemas como o de sobreajuste (*overfitting*).

Isto ocorre quando os modelos se ajustam muito aos dados apresentados no processo de treinamento e acabam não generalizando bem.

Na Figura 14 é apresentado um exemplo de ocorrência de *overfitting* no treinamento de RNAs (como por exemplo, as RNCs). O erro de treinamento onde o critério de parada do algoritmo ocorre de forma tardia, é no caso, uma ocorrência de *overfitting*. É possível notar na Figura 14 que o erro dos dados de treinamento estão diminuindo, porém, os erros nos dados de teste estão aumentando.

Figura 14 – Exemplo de *overfitting*



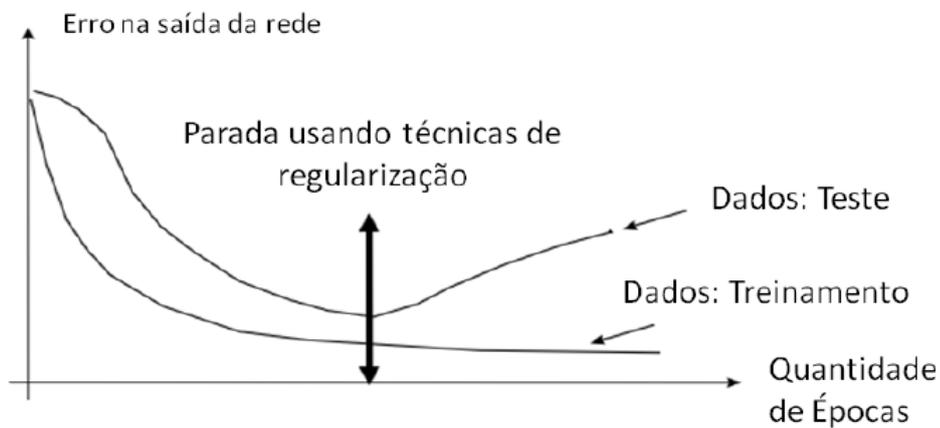
Fonte: (FARIA, 2018)

O problema de *overfitting* pode ser resolvido utilizando diferentes técnicas de regularização. Técnicas na qual são responsáveis por realizar uma modificação no algoritmo de aprendizagem com o objetivo em reduzir o erro de generalização, mas não necessariamente o erro do treinamento. A sua eficácia pode ser medida por um bom *trade-off* entre a redução da variância e o aumento do viés de estimadores (GOODFELLOW et al., 2016).

Uma das técnicas na qual pode ser utilizada e que vem sendo utilizada com sucesso em aplicações RNCs é denominada como *dropout* (HINTON et al., 2012). A proposta da mesma tem por objetivo definir estocasticamente zeros na saída de alguns neurônios de determinadas camadas com probabilidade de $p=0.5$. O restante dos pesos dos neurônios que não foram modificados passam pelo treinamento utilizando o algoritmo *backpropagation*. Este procedimento é realizado para cada exemplo em cada época de treinamento.

Na Figura 15, diferente da figura anterior, é apresentado um exemplo do uso de técnicas de regularização, *dropout* por exemplo. É possível notar que o treinamento é encerrado em um ponto na qual é considerado ótimo de generalização, antes que o erro no conjunto de teste aumentasse.

Figura 15 – Exemplo de treinamento utilizando técnica de regularização, evitando *overfitting*



Fonte: (FARIA, 2018)

Segundo (HINTON et al., 2012), a técnica *dropout* pode diminuir as co-adaptações complexas de neurônios, uma vez que é desativado os neurônios da camada associada, não é possível os mesmos influenciarem os demais neurônios. Com isso, os neurônios acabam aprendendo características mais robustas, aumentando assim a eficiência na generalização do modelo. Autores também relatam que a técnica é eficiente para reduzir o erro no conjunto de testes (generalização), assim como é computacionalmente eficiente, pois é capaz de treinar um grande número de redes diferentes em um tempo razoável.

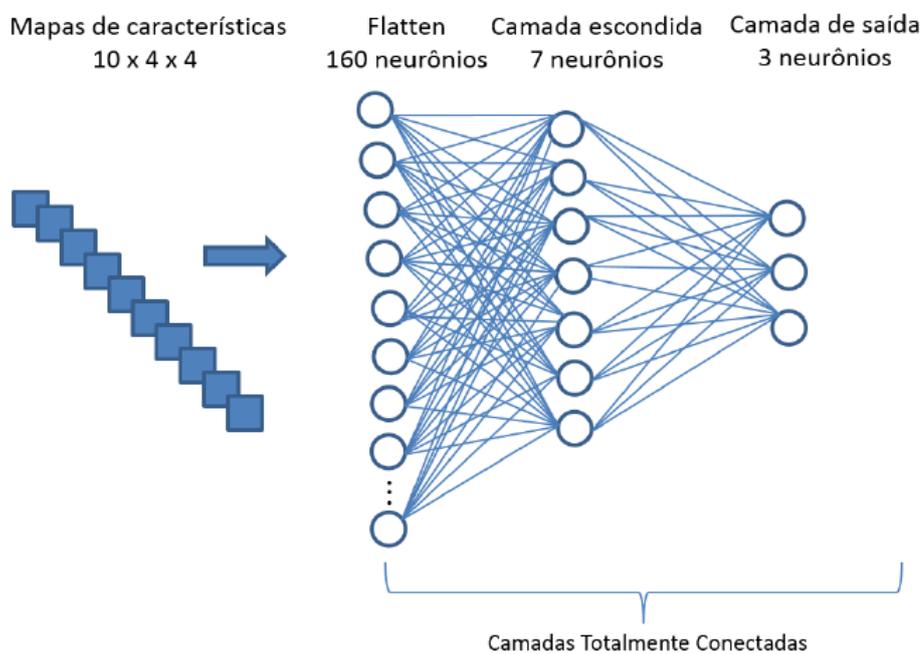
v. Camada totalmente conectada: As camadas totalmente conectadas (ou *Fully Connected* (FC)) são camadas que também estão presentes em redes neurais MLP. Nas RNCs é natural obter uma ou mais camadas FC. A camada FC tem por objetivo utilizar características (ou mapas de ativação) que são gerados por meio das camadas anteriores (camadas convolucionais e *pooling*) para fazer a classificação dos dados de entrada (imagem por exemplo) em múltiplas classes baseada em um conjunto de treinamento (PONTI; COSTA, 2018). Nesta camada cada neurônio possui um peso associado a cada elemento do vetor de entrada, ou seja, possui características de ser completamente conectada, tendo cada neurônio da camada anterior (normalmente a camada de *pooling*) sendo conectado a cada um dos neurônios da próxima camada (camada FC).

Com as características extraídas da imagem de entrada (na camada de convolução e *pooling*) o objetivo da camada FC é utilizar essas características para classificar a imagem em uma classe pré-determinada. A camada FC também é composta de uma última camada (camada de saída), onde cada neurônio representa uma classe específica do modelo. O número de neurônios que a camada de saída possui está diretamente ligado a quantidade de classes que o modelo possui.

Na Figura 16, é apresentado uma ilustração da camada FC (também conhecida como camada densa). Na Figura 16 é possível notar a presença de 10 mapas de carac-

terísticas de tamanho 4x4 (resultado das camadas anteriores, convolucional e *pooling*), na qual cada elemento da matriz de saída é conectado a um determinado neurônio de entrada. Estes resultados são postos de forma linear, resultando assim na primeira camada FC (camada *flatten* constituída por 160 neurônios). Na sequência estas entradas vão alimentar a segunda camada FC (camada escondida composta por 7 neurônios). Por fim, temos a camada de saída composta por 3 neurônios, que correspondes as 3 classes do modelo. Analisando apenas a camada totalmente conectada, é possível notar que a mesma pode ser identificada como uma rede neural tradicional MLP.

Figura 16 – Exemplo de uma camada totalmente conectada de uma RNC



Fonte: (FARIA, 2018)

A função de ativação mais utilizada na camada de saída de uma camada FC, segundo (GOODFELLOW et al., 2016), é a função sigmoide. A função sigmoide é descrita na Equação 2.7, e também pode ser visualizada na seção antecedente na Figura 5. Esta função tem por objetivo limitar a saída a um intervalo entre 0 e 1, tornando-se útil na previsão de probabilidades.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.7)$$

vi. Camada de saída:

A última camada da RNC é a camada de saída, a qual tem por objetivo receber estímulos das camadas intermediárias e criar um padrão que será o resultado final.

3 REVISÃO SISTEMÁTICA DE LITERATURA

Este capítulo apresenta um estudo de Revisão Sistemática da Literatura (RSL), que foi conduzido utilizando o método proposto por (PETERSEN et al., 2008). Apresenta-se uma revisão dos estudos publicados nos últimos dez anos, relacionados à detecção de doenças em plantas, que tratam de detecção automática de doenças por meio de ferramentas de apoio e que utilizam técnicas voltadas para processamento digital de imagens (JOSHI; JADHAV, 2016) e RNAs. Como resultado, selecionamos criteriosamente 35 estudos, mapeando-os em 55 doenças e em 25 técnicas de detecção de doenças. Portanto, nossa contribuição é uma análise e classificação do estado da arte sobre o tema, com uma visão panorâmica de doenças em culturas detectáveis por diferentes abordagens de processamento de imagens.

3.1 Trabalhos Relacionados

Ao longo de nossa pesquisa, encontramos trabalhos relacionados, cujos enfoques e limitações são apresentados na Tabela 4.

Tabela 4 – Comparação dos trabalhos relacionados.

Referência / Ano	Foco de Contribuição	Limitação
(PIERCE; NOWAK, 1999)	Uma contribuição para a fundamentação de conceitos no âmbito da agricultura de precisão. Muito antigo para fornecer evidências para a detecção automática de doenças em plantas.	Nenhum protocolo de pesquisa encontrado. Datado por técnicas modernas.
(RICHARDS; RICHARDS, 1999)	Uma contribuição para a fundamentação de conceitos que abrangem o sensoriamento remoto em conjunto com a análise digital de imagens. Muito antigo para fornecer evidências para a detecção automática de doenças em plantas.	Nenhum protocolo de pesquisa encontrado. Datado para tecnologias de coleta modernas.
(PATRÍCIO; RIEDER, 2018)	Os autores discutem em poucos detalhes as contribuições da visão computacional e da inteligência artificial, não comparando os 25 estudos selecionados.	Nenhum protocolo de pesquisa encontrado. Menor número de estudos.
(FARBER et al., 2019)	Discutem novas oportunidades de pesquisa baseadas em técnicas espectroscópicas para diagnóstico de doenças em plantas.	Nenhum protocolo de pesquisa encontrado. Não no mesmo tópico.
(NGUGI; ABELWAHAB; ABO-ZAHHAD, 2020)	O artigo contribui para a área de pesquisa, por meio da análise de desempenho de 10 arquiteturas RNCs para reconhecimento de doenças foliares.	Nenhum protocolo de pesquisa encontrado. Poucos estudos para comparação.
Nosso RSL 2020	Nossa contribuição compara a acurácia de todos os 35 estudos selecionados, melhorando assim a análise de (NGUGI; ABELWAHAB; ABO-ZAHHAD, 2020) por trazer um melhor poder estatístico. Nosso estudo é RSL, portanto, replicável.	Poucos estudos selecionados em comparação com (NGUGI; ABELWAHAB; ABO-ZAHHAD, 2020).

Claramente, a análise do estado da arte carece de contribuições importantes na área, em especial aquelas que podem se repetir ao longo dos anos para acompanhar com clareza o estado da arte neste tema de pesquisa. Nenhum dos estudos relacionados é repetível uma vez que não há protocolo de pesquisa. Alguns deles alegaram a palavra “sistemática”, mas não seguiram nenhum protocolo de pesquisa como o nosso. Como consequência, são descobertas e afirmações não podem ser refutadas por outras pesquisas.

Nosso estudo complementou os já citados, com um rigoroso protocolo de revisão sistemática da literatura. Nosso objetivo é apresentar uma revisão sistemática real da literatura construída sobre os conceitos de Engenharia de Software Empírica (ESE) (SHULL; SINGER; SJBERG, 2010), tornando nossos achados confiáveis e guardados em algumas condições de contorno discutidas neste estudo.

3.2 Protocolo de Pesquisa

Esta seção é dedicada a caracterizar como planejamos e conduzimos nossa análise de informações de estudos primários, representando assim o protocolo de revisão sistemática da literatura. Os estudos selecionados coletaram, por meio de técnicas automatizadas de detecção de doenças, imagens de formulários de dados de doenças. Para conduzir nossa revisão, seguimos uma diretriz proposta por (PETERSEN et al., 2008), inicialmente planejada como um estudo de mapeamento sistemático e então planejada e reexecutada como Revisão Sistemática da Literatura (RSL) devido ao uso de questões de pesquisa e recursos de comparação. Para chegar a estudos mais recentes, restringimos a revisão aos estudos publicados nos últimos dez anos.

O pacote RSL é descrito na Tabela 5. A primeira linha fornece um link para a pasta *Zenodo* que compartilha o pacote RSL. As outras linhas descrevem cada arquivo.

Tabela 5 – Descrição do pacote de revisão sistemática da literatura

Pacote de revisão sistemática da literatura	
https://doi.org/10.5281/zenodo.5619031	
Artefato	Descrição do conteúdo
"Artigos selecionados" e todos os artigos selecionados em formato PDF	
"Arquivos <i>Bib</i> dos bancos de dados selecionados. Zip"	Todos os <i>bibs</i> dos bancos de dados antes de agrupar
"Arquivos <i>bib</i> agrupados dos bancos de dados selecionados.ods" e entradas de bibliografia agrupadas.	
"Extração de dados.xlsx" e formulários de extração de dados dos 35 estudos selecionados	

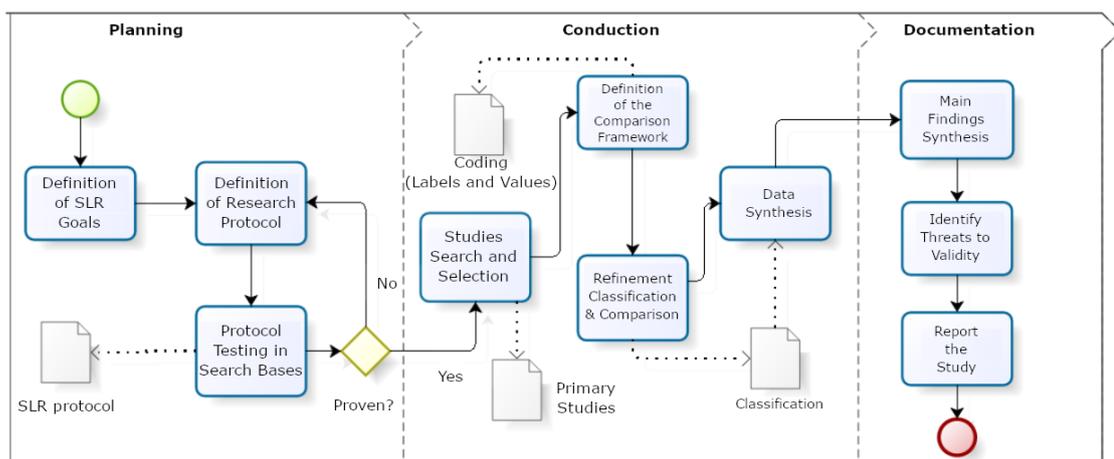
3.2.1 Objetivo de Pesquisa

Nosso objetivo é alcançar quais trabalhos apresentam os melhores resultados de processamento de imagens por culturas e doenças específicas. Procuramos comparar estudos com base em seus relatórios de taxa de utilização, por exemplo, falsos positivos e falsos negativos. Para realizar este estudo, seguimos o protocolo definido por (PETERSEN et al., 2008). A Tabela 5 referencia o protocolo, incluindo um relatório sobre a execução, todos os arquivos derivados e material adicional.

3.2.2 Processo de Pesquisa

As fases e tarefas do método de pesquisa adotado são ilustradas na Figura 17. Essas fases e tarefas foram realizadas por diferentes membros da equipe, sincronizando seus trabalhos desde a concepção dos estudos até o relato.

Figura 17 – O processo de estudo de mapeamento sistemático.



Fonte: Autor

3.2.3 Fase de Planejamento

A fase de planejamento foi realizada em março de 2019, pelos três primeiros pesquisadores. Foi então revisado pelo Revisor 1 e pelo Revisor 2 ainda na fase “Planejamento”, interagindo nas tarefas “Definição de Metas RSL”, “Definição de Protocolo de Pesquis”, “Teste de Protocolo em Bases de Pesquisa”. Pelo menos três execuções piloto de pesquisa de palavras-chave foram realizadas na fase 1, refinando o protocolo de pesquisa repetidamente.

Ao longo da execução, ou “Fase de Condução”, o Revisor 3, nosso revisor técnico, entrou como colaborador para alinhar os elementos técnicos de Processamento de Imagens e Inteligência Artificial a partir dos achados nos estudos selecionados. Ele ingressou no “processo de codificação” (PANDIT, 1996) associado à tarefa “Definição da Estrutura de Comparação”. O processo de codificação fornece entradas na forma de lançamentos de

Planilhas, permitindo a geração de uma “Classificação” como saída da tarefa “Refinamento, Classificação e Comparação”. Por fim, a última tarefa de condução denominada “Síntese de Dados”, libera a classificação final dos estudos selecionados.

Na fase de documentação, os três primeiros executores derivaram os resultados da tarefa “Síntese das principais descobertas” e analisaram as ameaças à validade. Finalmente, o Revisor 4, como membro da equipe técnica de ciência rural, ajudou a alinhar o relatório para um público de ciência rural além de cientistas da computação.

3.2.4 Definição de Protocolo de Pesquisa

Os artigos foram pesquisados em seis bases de dados apresentadas na Tabela 6, que caracteriza as bases de dados mais adotadas para artigos de Ciência da Computação.

Tabela 6 – Bancos de dados de artigos usados para pesquisar título, resumo e palavras-chave

Biblioteca Digital	Link
ACM Digital Library	http://dl.acm.org/
IEEE Xplore	http://ieeexplore.ieee.org/
Science Direct	http://www.sciencedirect.com/
Scopus	http://www.scopus.com/
Springer Link	http://link.springer.com/
Web of Science	http://apps.webofknowledge.com/

A *string* de pesquisa foi planejada de acordo com a diretriz de População, Intervenção, Comparação, Resultado e Contexto (PICOC) (KITCHENHAM; CHARTERS, 2007):

- **População:** estudos propondo abordagens automatizadas de detecção de doenças;
- **Intervenção:** técnicas de apresentação, ou ferramentas, ou dispositivos de coleta de imagens;
- **Comparação:** precisão de estudos publicados entre 2009 e 2019;
- **Resultado:** discutindo o reconhecimento de padrões, aprendizado profundo, processamento de imagem;
- **Contexto:** aplicado a plantas em contextos de agricultura de precisão, automação agrícola ou agricultura inteligente.

Para realizar a busca utilizou-se uma *string* mostrada na Figura 18.

Figura 18 – String de busca genérica

```
“pattern recognition” OR “deep learning” OR  
“image processing”  
  
AND  
  
“precision agriculture” OR “agricultural  
automation” OR “smart farming”
```

3.2.5 Questões de Pesquisa

Estamos motivados a investigar a seguinte lacuna de pesquisa encontrada na literatura da área: a falta de um estudo sistemático comparativo levantando as contribuições no estado da arte dedicadas à pesquisa de detecção automática de doenças, usando técnicas de processamento de imagens e redes neurais como meio para melhorar a agricultura de precisão. Nosso principal objetivo é caracterizar os estudos direcionados a implementações e metodologias para extração de conhecimento em padrões de imagens que apresentam doenças de plantas em lavouras. Para tanto, as seguintes questões de pesquisa são investigadas:

- **QP.1:** Para quais domínios de aplicação/contexto são usados o processamento de imagens e as redes neurais para detecção de doenças em plantações? O objetivo é caracterizar os estudos por domínios de aplicação com um mapa que descreva as novas oportunidades de aplicação do processamento de dados no reconhecimento de doenças de plantas na agricultura de precisão.
- **QP.2:** Quão precisos são os estudos selecionados para a detecção de doenças de plantas? Como a qualidade da detecção da doença é medida pelo mínimo de um atributo, nosso objetivo é identificar os estudos que mostram os melhores resultados para as doenças testadas.
- **QP.3:** Quais são os conjuntos de dados usados pelos estudos? Nosso objetivo é identificar os bancos de imagens disponíveis para a realização de novos estudos, sejam eles abertos ou pagos pela quantidade de imagens disponíveis, e o número de classes utilizadas nos estudos.

3.2.6 Critérios de Inclusão e Exclusão

Critério de inclusão:

- CI 1 - Os estudos devem estar dentro do domínio do problema “Detecção de doenças em plantas”;
- CI 2 - Estudos da área de Ciência da Computação;

- CI 3 - Os estudos não devem ser mapeamentos ou revisões de literatura;

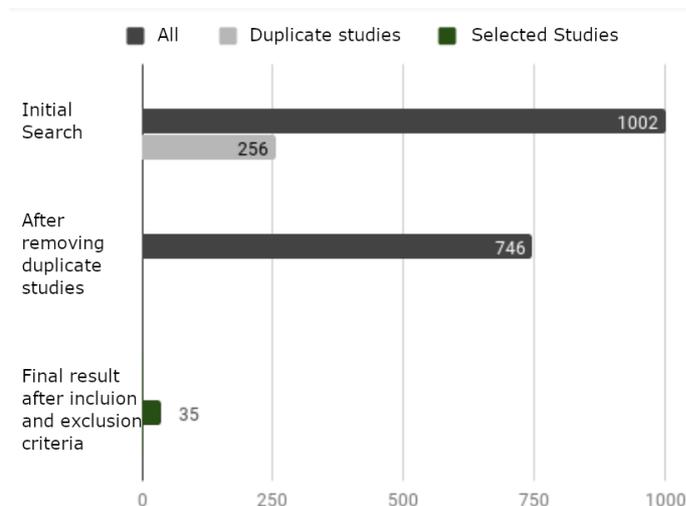
Critério de exclusão:

- CE 1 - Estudos escritos em outro idioma (que não seja o inglês);
- CE 2 - Estudos indisponíveis para download em acesso aberto, ou por meio do Portal Capes, ou apenas parcialmente disponíveis para download sem seu conteúdo completo;
- CE 3 – Estudos que não discutem questões de processamento de imagem ou RNAs;
- CE 5 - Estudos de outras áreas;
- CE 6 – Estudos secundários e terciários não serão incluídos, ou seja, outros mapeamentos e revisões, embora os tenhamos coletado para relatar como trabalhos relacionados;
- CE 7 - Estudos abaixo de cinco páginas; e
- CE 8 - Estudos anteriores a 2009.

3.2.7 Resultados

A Figura 19 apresenta um gráfico com os resultados obtidos na etapa de pesquisa “Pesquisa e Seleção de Estudos”. Nesta etapa, usamos a *string* de pesquisa nos bancos de dados selecionados. Com a busca inicial foram reunidos 1.002 estudos, incluindo 256 duplicatas. Foram excluídos os artigos duplicados, resultando em um total de 746 estudos primários.

Figura 19 – Pesquisa e seleção de estudos



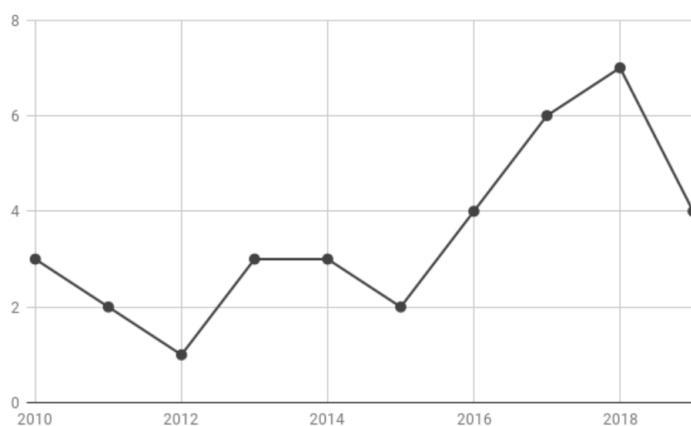
Fonte: Autor

Neste ponto, chegamos à tarefa “Busca e Seleção de Estudos”. Esses estudos foram avaliados quanto aos critérios de inclusão e exclusão. Assim, excluímos contribuições não dedicadas ao reconhecimento de doenças de plantas usando técnicas de processamento de imagens ou redes neurais. O resultado final é um total de 35 estudos selecionados.

Os estudos selecionados também passaram por mais duas etapas. As questões de pesquisa foram utilizadas como base para questões que nos permitiram retirar informações importantes e necessárias dos estudos para que fosse possível concluir a pesquisa. Um dos critérios de inclusão e exclusão descritos no protocolo de pesquisa é que os estudos selecionados devem estar entre 2009 e 2019, não sendo possível incluir estudos anteriores a 2009.

Uma informação importante é que os estudos sobre o tema vêm crescendo nos últimos anos, conforme apresentado na Figura 20. Em 2018 foram concluídos sete estudos, aumento de um estudo em relação a 2017. Em 2012, foi encontrado apenas um estudo. E até março de 2019, data em que as bases de dados foram pesquisadas, surgiram quatro novos estudos. Assim, considerando também as publicações de 2018, isso demonstra que este tema de pesquisa está em ascensão.

Figura 20 – Número de artigos publicados por ano



Fonte: Autor

3.3 Resultados do Estudo de Revisão Sistemática da Literatura

A Tabela 3.3 apresenta os estudos selecionados. Cada um desses estudos é caracterizado na Seção 3.3.1 por seus contextos de aplicação a doenças e plantas específicas, na Seção 3.3.2 por sua precisão para detectar doenças em plantas, Seção 3.3.3 pela qualidade de suas imagens e elementos técnicos.

Tabela 7 – Todos os estudos selecionados com sua taxa de sucesso adquirida na detecção de doenças

ID	Referência	Acurácia
S01	(SINGH et al., 2019)	97,13%
S02	(PICON et al., 2019)	84%
S03	(LEE et al., 2019)	95%
S04	(LIBO; TIAN; CHUNYUN, 2019)	97,62%
S05	(HOSSAM et al., 2018)	95,73 to 98 %
S06	.(APARNA; SUPRIYA, 2018)	Undefined
S07	(LUNA; DADIOS; BANDALA, 2018)	91,6% to 95,75%
S08	(KAUR; PANDEY; GOEL, 2018)	85,65%
S09	(KERKECH; HAFIANE; CANALS, 2018)	95,80%
S10	(WANG et al., 2018)	95%
S11	.(GHOSAL et al., 2018)	94,13%
S12	(SCHOR et al., 2017)	61% to 95%
S13	(HLAING; ZAW, 2017)	78,7%
S14	(KUMAR et al., 2017)	96%
S15	(KNAUER et al., 2017)	99,8%
S16	(LU et al., 2018)	78,8% to 100%
S17	(PANTAZI et al., 2017)	95%
S18	(JOSHI; JADHAV, 2016)	87,02% to 89,23%
S19	(MORIYA et al., 2016)	92,50%
S20	(HOU; LI; HE, 2016)	75% to 94,4%
S21	(CHEMURA; MUTANGA; DUBE, 2017)	76,2% to 82,5%
S22	(AWATE et al., 2015)	Undefined
S23	(BHANGE; HINGOLIWALA, 2015)	82%.
S24	(YANG et al., 2015)	93,0% to 96,5%
S25	(CALDERÓN et al., 2014)	Undefined
S26	(OERKE; MAHLEIN; STEINER, 2014)	Undefined
S27	(JHURIA; KUMAR; BORSE, 2013)	Undefined
S28	(LI et al., 2014)	86%
S29	(YUAN et al., 2012)	71%
S30	(HILLNHÜTTER et al., 2012)	79,4%
S31	(BAUER; KORČ; FÖRSTNER, 2011)	86% and 91%
S32	(SANTOSO et al., 2011)	96% and 84%
S33	(ZHANG et al., 2011)	80%
S34	(POLDER et al., 2010)	71% to 90,82%
S35	(SHIJUAN; YEPING, 2009)	Undefined

3.3.1 Contextos de domínio de estudos aplicados

A caracterização dos estudos aplicados é essencial para entender se os estudos selecionados contribuem para as necessidades práticas do ponto de vista dos produtores. Com isso em mente, esta seção responde ao **QP.1** “Para quais domínios de aplicativo/contexto são usados o processamento de imagens e as redes neurais para detecção de doenças em culturas?”.

O objetivo geral dos trabalhos revisados é desenvolver técnicas ou sistemas para detecção automática de doenças em plantas por meio do processamento de imagens em conjunto com outras técnicas como redes neurais convolucionais, softwares, ferramentas, *frameworks*, entre outros.

A Tabela 8 mostra a relação entre plantas e doenças dos estudos selecionados. Na primeira coluna especifica-se o tipo de planta estudada, e na segunda coluna as doenças relacionadas a cada uma dessas plantas e o estudo relacionado a cada uma dessas plantas e doenças. As plantas que mais apareceram nos artigos revisados foram a uva, o trigo e o tomate. Tendo a uva (ou videira) com escopo definido por um total de oito estudos, seguidos do tomate e o trigo, com escopo definido por quatro estudos.

Tabela 8 – Plantas e doenças analisadas nos estudos revisados

Cultura	Lista de estudos e doenças	Total de estudos
Manga	S01{Anthracnose}	1
Trigo	S02{Septoria, Tan spot, Rust}; S17, S29, and S33 {Yellow Rust}	4
Batata	S03{Bacterial Soft Rot}	1
Tomate	S04{Gray mold, Canker, Oidium, Prague Yersinia Pestis, Leaf miner, whitefly, Oidium}; S07{Phoma Rot, Leaf Miner, Ponto Alvo}; S13{Bacterial stain, Late rust, Mold, Septoria leaf spot, Two-spotted mites, Target spot, Mosaic virus}; and S16{Yellow leaf }	4
Milho	S10{Large leaf spots, Dwarf mosaic, Gray spots}	1
Soja	S08{Mildew, Frog eye, Septoria blight}; S11{Bacterial spot, Bacterial pustule, Sudden death syndrome, Brown Septoria spot, Frogeye leaf spot}	2
Videira e Uva	S09{Esca}; S20{Grapevine leafroll disease (GLD)}; S15, S22, S27{Powdery mildew}; S22, S27{Black Rot}; S22{Downy Mildew}	8
Pimentão	S12{Powdery mildew (PM), Tomato spotted wilt virus (TSWV)}	1
Cana de açúcar	S14{Leaf excoriation, Red stripe}; S14, S19{Mosaic}	3
Arroz	S18{Bacterial rust, Rice explosion, Brown rice spot, Rice husk rot}	1
Café	S21{Coffee leaf rust (CLR)}	1
maçã	S22, S27{Apple Scab, Apple Rot}; S22{Apple Blotch}	3
Romã	S22, S23{Bacterial Blight}; S22{Aspergillus Fruit Rot, Gray Mold.}	3
Algodão	S24{Cotton Root Rot }	1
Ópio	S25{Downy Mildew (DM)}	1
Citrus	S28{Citrus Greening}	1
Beterraba	S30{Heterodera schachtii, Rhizoctonia solani}; S31{Cercospora beticola, Uromyces betae}	2
Palma	S32{Basal Rot Disease}	1
Tulipa	S34{Tulip Breaking Virus (TBV)}	1

Em relação às doenças encontradas nos estudos, cada contribuição tratou da identificação/deteção de uma ou mais doenças em uma ou mais culturas. A Figura 21 mostra algumas amostras dessas doenças encontradas em plantas.

Figura 21 – Amostra de imagens de plantas doentes



3.3.2 Comparando os estudos

Como a qualidade para detecção de doenças é medida por pelo menos um atributo, a acurácia, nosso objetivo é identificar os estudos que apresentam os melhores resultados para as doenças testadas, respondendo assim ao **QP.2** da seguinte forma.

As Figuras 22 e 23 mostram os resultados dos estudos revisados, contendo a acurácia de cada trabalho. A divisão foi realizada em dois gráficos devido aos dados disponibilizados por cada grupo de estudo. A Figura 22 mostra os estudos que apresentaram a divisão do conjunto de dados, que foi utilizado para calibração/treinamento e teste e validação de seus algoritmos. Enquanto a Figura 23 apresenta os resultados dos estudos que não mencionaram uma divisão do conjunto de dados para as fases de treinamento/calibração, teste e validação, na qual mencionaram apenas o número de classes. Em ambos os gráficos da Figura 22 e 23, existem resultados como S18a e S18b, o que significa que dois experimentos são usados pelo estudo S18.

A Figura 22 mostra os resultados de acurácia, com sua relação por uma série de imagens utilizadas. O estudo S05 obteve o melhor resultado 98%, que também possui o maior conjunto de dados (93500). O estudo S29 apresentou como resultado o menor valor de acurácia de 71%, que também possui o menor conjunto de dados (91). Nota-se que na maioria dos casos, o estudo que utilizou o maior número de imagens, tem maior chance em obter o melhor resultado. Porém, também observamos uma discrepância em alguns estudos, obtendo melhores resultados com um menor número de imagens.

Uma possível explicação para tal discrepância é que existem n fatores que interferem no resultado, como o tipo de imagem, se tratada ou não preliminarmente, obtida em um cenário real não controlado ou em algum cenário controlado, pré-processamento, algoritmo, ou arquitetura de classificação, entre outros fatores de confusão não detalhados por cada estudo. Mesmo com todas as variáveis, é possível constatar que um grande número de imagens aumenta as chances de bons resultados.

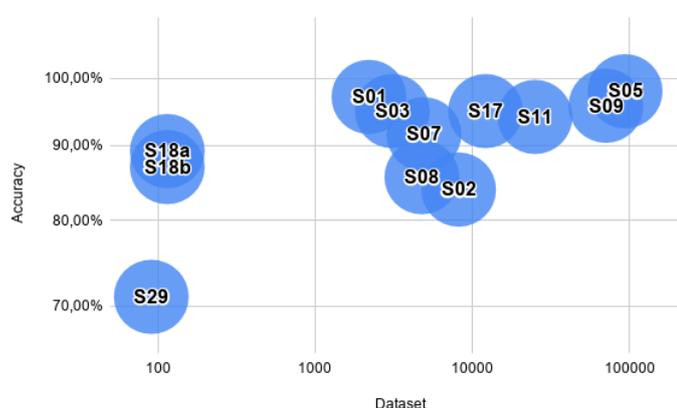
3.3.3 Qualidade e número de classes de imagens de conjunto de dados

Nosso objetivo é identificar os conjuntos de dados de imagens disponíveis gratuitamente para estudos posteriores, ou seja, se são abertos ou a pagar, o número de imagens disponíveis no conjunto de dados e também o número de classes utilizadas nos estudos. Assim, o resultado de **QP.3** é apresentado a seguir.

A Tabela 9 mostra algumas informações sobre a amostra dos estudos selecionados, como o tipo do conjunto de dados, se é próprio ou obtido de terceiros, o tamanho do conjunto de dados utilizado no estudo, e também se o conjunto de dados está livremente acessível ao público em geral ou não. Apenas três estudos forneceram contato para solicitação das bases de dados S04, S11, S15. Os demais estudos não mencionaram a disponibilidade do conjunto de dados, bem como a indisponibilidade de contato para a

solicitação. Quatro estudos utilizaram o banco de dados *Plantvillage*, que está aberto ao público, no entanto, atualmente, os dados não estão mais disponíveis S01, S05, S08, S13. O estudo S18 foi o único a utilizar imagens de terceiros em conjunto com imagens da Internet. Sete estudos não forneceram o tamanho do conjunto de dados S06, S22, S24, S26, S27, S30 e S35. Três estudos não mencionaram um banco de dados ou o número de imagens usadas S26, S27 e S35.

Figura 22 – Gráfico de dispersão mostrando os resultados obtidos pelo tamanho do conjunto de dados



Fonte: Autor

A Figura 23 apresenta os resultados dos estudos, tendo como referência a acurácia e o número de classes encontradas no conjunto de dados dos estudos selecionados. Os melhores resultados são alcançados pelos estudos S15 e S16, obtendo precisão de 99,80% e 100%, respectivamente. Ambos os trabalhos usaram apenas duas classes de imagens, uma classe para imagens de plantas saudáveis e outra para imagens de plantas doentes. Outro destaque é o estudo S04, com uma precisão de 97,62%, utilizando dez classes diferentes para o desenvolvimento do estudo.

Para isolar fatores de confusão na detecção de doenças, pode-se notar que classes adicionais são adicionadas pelos estudos em seu conjunto de dados usado para treinar o computador de aprendizado de máquina. À medida que mais classes são adicionadas ao conjunto de dados, mais esforço é necessário para realizar a calibração ou classificação das doenças, tornando essas abordagens mais confiáveis. Paradoxalmente, esses resultados mostram que o número de classes pode não interferir no resultado final, uma vez que muitos estudos estão no intervalo de duas classes adotadas e também apresentam resultados significativos, comparáveis àqueles estudos com mais de oito classes também apresentando bons resultados.

Tabela 9 – Todos os estudos selecionados com sua configuração de conjunto de dados

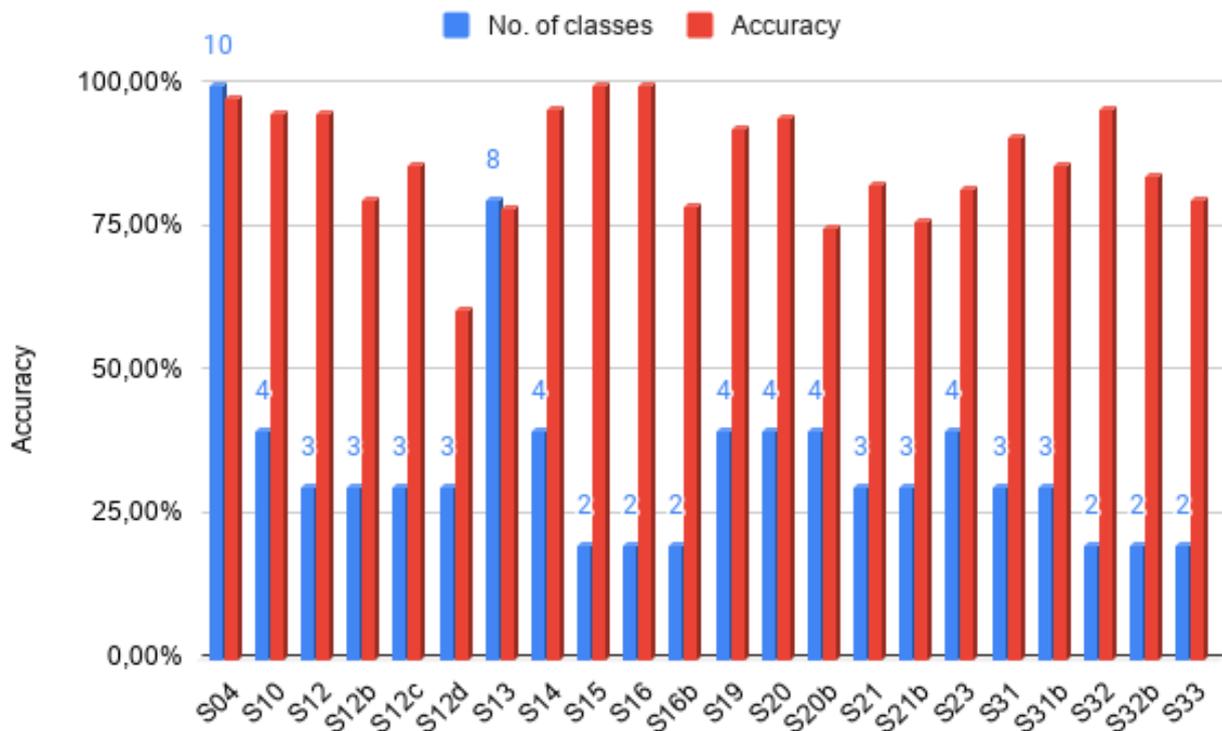
ID	Conjunto de dados	Tamanho	Licença
S01	Próprio / <i>Plantvillage</i>	2200	Indefinido
S02	Próprio	8178	Indefinido
S03	Próprio	3100	Indefinido
S04	Próprio	5000	Gratuitamente. *
S05	<i>Plantvillage</i>	93500	Indefinido
S06	Próprio	Indefinido	Indefinido
S07	Próprio	4923	Indefinido
S08	<i>Plantvillage</i>	4775	Indefinido
S09	Próprio	70560	Indefinido
S10	Próprio	300 to 500	Indefinido
S11	Próprio	25000	Gratuitamente. *
S12	Próprio	250 to 350	Indefinido
S13	<i>Plantvillage</i>	3988	Indefinido
S14	Próprio	200	Indefinido
S15	Próprio	60	Gratuito. *
S16	Próprio	166	Indefinido
S17	Próprio	12120	Indefinido
S18	T	115	Indefinido
S19	Próprio	113	Indefinido
S20	Próprio	49	Indefinido
S21	Próprio	63	Indefinido
S22	Próprio	Indefinido	Indefinido
S23	Próprio	610	Indefinido
S24	Próprio	Indefinido	Indefinido
S25	Próprio	40	Indefinido
S26	Indefinido	Indefinido	Indefinido
S27	Indefinido	Indefinido	Indefinido
S28	Próprio	128	Indefinido
S29	Próprio	91	Indefinido
S30	Próprio	Indefinido	Indefinido
S31	Próprio	700	Indefinido
S32	Próprio	206	Indefinido
S33	Próprio	50	Indefinido
S34	Próprio	257	Indefinido
S35	Indefinido	Indefinido	Indefinido

LEGENDA:

* Fornece contato para solicitação de conjunto de dados.

T Conjunto de dados e imagens da Internet de terceiros.

Figura 23 – Gráfico de dispersão mostrando os resultados obtidos em cada estudo pelo número de classes



Fonte: Autor

Nosso estudo apresenta um problema de agrupamento dos resultados de precisão pela planta. Ou seja, para cada planta, é possível que haja um ponto em que o aumento do número de classes não faça diferença no aumento da acurácia de forma diferente da de outra planta. Entender essas diferenças é muito importante para traçarmos boas práticas em reconhecimento de imagem. Porém, para entender melhor essa relação, são necessários mais dados de novos estudos primários e, como há poucos estudos mapeados para determinadas plantas, não é viável realizar melhores agrupamentos.

3.4 Discussões

Constatamos que algumas lacunas permanecem na área de pesquisa. A indisponibilidade de um banco de dados aberto de imagens, utilizado nos estudos selecionados, foi apontado como um problema, o que impossibilita a replicação científica dos resultados por meio de novos testes. Como consequência, os resultados apresentados pelos pesquisadores não podem ser refutados. Alguns estudos adotaram um conjunto de imagens da *PlantVillage* S01, S05, S08 e S13, que é aberto ao público. No entanto, descobrimos que este conjunto de imagens não está mais disponível.

Muitos estudos não compararam suas técnicas propostas com outras. De todos os artigos revisados, doze técnicas comparadas demonstrando resultados melhorados S05,

S10, S17, S18, S21, S24, S25, S28, S30, S31, S32 e S34, enquanto os 23 restantes avaliaram apenas a precisão da técnica proposta. Doze estudos comparativos são um bom índice de proporção, mostrando que a área de pesquisa está em maturação científica. Porém, notamos que nenhum estudo apresentou discussão sobre ameaça à validade. Assim, para se ter uma ideia clara de como essas técnicas e ferramentas funcionam melhor, uma caracterização clara deve ser fornecida, como o contexto de coleta de imagens com diferentes configurações de imagens, condições climáticas, estações, culturas e outros fatores de confusão presentes em cenários reais. Assim, novos estudos podem considerar a discussão de suas condições de contorno experimentais.

Em relação ao número de classes no conjunto de dados de imagem, alguns dos artigos revisados usam redes neurais convolucionais para classificar e diferenciar folhas doentes de folhas saudáveis S01, S02, S03, S04, S05, S07, S09, S11. Em comparação com outras técnicas utilizadas nos estudos, as RNCs ainda são um número menor. No entanto, como as RNCs ainda são objeto de pesquisas recentes, e considerando que os estudos citados acima são publicados entre 2018 e 2019, é provável que sejam amplamente utilizados.

Os estudos S01, S02 e S04, por exemplo, utilizam em suas pesquisas imagens retiradas de um cenário real, nas quais atendem para caracterizar variabilidade real como poeira, iluminação, sombras, fundos diversos, entre outras informações que, em um ambiente controlado, não existe. Isso mostra que esses resultados possibilitarão o uso de RNCs em um ambiente real não controlado. Os estudos S03, S05 e S07, por exemplo, usam imagens que não fazem parte de um ambiente real. Dentre os estudos citados acima, S09 utiliza imagens adquiridas por meio de um VANT, em ambiente real não controlado, reforçando a hipótese de que pode ser possível utilizar RNCs em ambientes reais para detectar doenças em plantas.

3.5 Análise dos Artigos com Foco em Doenças no Soja

Após uma análise geral dos artigos selecionados para o desenvolvimento do estudo, foi feita a escolha da técnica e a cultura a ser utilizada em nosso estudo. Após escolhermos a soja como a cultura a ser estudada e as RNCs como técnica padrão para a aplicação do estudo, foi realizado uma pesquisa a parte contendo dois estudos que também utilizaram técnicas na detecção de doenças na folha da soja, esses estudos estão entre os estudos analisados na revisão sistemática da literatura.

Os dois estudos que utilizaram a cultura da soja foram os trabalhos S08 e S11. Os dois estudos optaram por detectar doenças na folha da soja, utilizando um banco de imagens para a realização dos experimentos. O estudo S08 utilizou um banco de imagens de terceiros, na qual denomina-se *PlantVillage*, já o estudo S11 fez uso de um banco de imagens próprio, utilizando uma câmera digital para a aquisição das imagens em um cenário real. Em relação ao número de imagens, o estudo S08 utilizou 4775 imagens para

o treinamento e teste dos algoritmos, já o estudo S11 utilizou 65.760 imagens para o treinamento, teste e validação dos algoritmos criados. Isso mostra a variedade no número de imagens utilizadas pelos estudos.

Em relação as técnicas utilizadas para a detecção das doenças, o estudo S08 utilizou a linguagem de programação MATLAB como técnica padrão, juntamente com o algoritmo *k-means*. Já o estudo S11 optou pela utilização das redes neurais convolucionais profundas como técnica padrão na detecção de doenças na folha da soja. Os resultados obtidos pelos estudos são distintos, enquanto o estudo S08 alcançou uma acurácia igual a 85,65% de acurácia, o estudo S11 obteve um resultado igual a 94,13% de acurácia. Com os resultados obtidos por ambos os estudos, é correto afirmar que o estudo S11 alcançou melhores resultados com o treinamento de seus algoritmos quando comparado ao estudo S08.

Existem muitas variáveis entre os dois estudos como o número de imagens, tipo de imagem utilizada, técnicas utilizadas para a detecção das doenças, pré-processamento e divisão do conjunto de imagens para os treinamentos e testes. Levando em consideração que o estudo S11 utiliza imagens próprias e tiradas de um cenário real, o resultado positivo se torna ainda mais importante, assim como a criação de um bando de imagens próprio com um número alto de imagens, na qual favorece o treinamento dos algoritmos criados. Comparando os dois estudos com o proposto nesta dissertação, o mesmo contém um número muito inferior de imagens, com isso, tentar alcançar bons resultados foi um grande desafio. O desenvolvimento e resultados de nossa pesquisa será apresentado nos capítulos seguintes.

3.6 Ameaças à validade

Segundo (AMPATZOGLOU et al., 2019), os estudos de RSL podem sofrer algumas ameaças à validade, divididas em ameaças construtoras, ameaças internas, ameaças externas e ameaças à conclusão.

Quanto aos problemas do construtor, as possíveis ameaças são direcionadas ao planejamento do estudo. Para evitar ameaças como essa, a pesquisa foi previamente planejada, resultando em um protocolo duplamente verificado por dois especialistas em revisões sistemáticas, com revisão piloto realizada ao longo de um curso de mestrado.

Ameaças internas estão relacionadas a quão bem nossas análises respondem às questões de pesquisa, considerando, portanto, atributos da revisão como identificação do tema de pesquisa, seleção de estudos, avaliação da qualidade, extração e síntese de dados. Em relação aos critérios de inclusão e exclusão, a seleção por meio da leitura do título e resumo pode implicar na perda de estudos relevantes ao longo do processo de seleção. Para mitigar essa ameaça, quando dois revisores não chegavam a um consenso sobre a seleção de um estudo, eles recorriam ao líder do projeto para votação de decisão. Quanto aos atributos de qualidade, estes foram utilizados, mas não nas questões de pesquisa relatadas neste artigo. Estão disponíveis na continuidade deste material, estratégia propositalmente

utilizada para relatar apenas parte do estudo geral. O próprio relatório pode sofrer algumas ameaças internas, que foram tratadas após reuniões entre os sete pesquisadores envolvidos para garantir que nenhuma perda afetaria as conclusões do estudo.

Ameaças externas dizem respeito ao poder do estudo de revisar e generalizar os resultados. Nesse sentido, os resultados dos achados não devem ser generalizados para todas as técnicas de extração de informações. Assim, os resultados são observados especificamente no contexto apresentado neste artigo, e com mais detalhes em nosso protocolo de pesquisa vinculado na Seção 3.

As ameaças de conclusão são principalmente direcionadas ao poder estatístico do estudo para apoiar os resultados. Nesse sentido, 35 estudos selecionados podem ser considerados uma ameaça ao poder estatístico. No entanto, é importante mencionar que as revisões sistemáticas são, em sua maioria, comparativas e apresentam questionamentos aprofundados. Assim, devem garantir a seleção de estudos de qualidade e não em quantidade, o que costuma resultar em menos números do que estudos de mapeamento sistemático.

3.7 Considerações Finais

A detecção automática e precoce de doenças está evoluindo ao longo dos anos e, um dia, cada vez mais ajudarão o dia a dia dos agricultores. A seguir apontamos porque eles ainda não são uma realidade observada nos kits de ferramentas dos agricultores. Nesta revisão sistemática, mostramos que há uma variedade de estudos na área adotando diferentes formas de detectar doenças em plantas. A acurácia média alcançada pelos estudos que especificaram em seus trabalhos a divisão do conjunto de dados para realizar o treinamento, calibração, teste ou validação (mostrado na Figura 22) foi de 90,30%. Estudos que não detalhavam a divisão ou utilização do conjunto de dados (mostrado na Figura 23), obtiveram uma acurácia média de 86,02%.

Alguns pontos negativos dos artigos revisados foram destacados. Devido à natureza da pesquisa, voltada para questões experimentais voltadas para o rigor, encontramos dificuldades para entender como as tecnologias propostas são transferidas para a sociedade. Ou seja, exceto pelo estudo S02 que menciona levemente a integração com a indústria, essas tecnologias não são relatadas como adotadas na prática, o que soa estranho devido à alta demanda do setor rural. Trabalhos futuros explorarão essas questões por meio de uma pesquisa de “snowballing”, tentando identificar como foi realizada a continuidade da abordagem proposta para detecção de doenças a partir do acompanhamento dos autores principais.

Em termos de protocolo de pesquisa que pode ser repetido, constatamos que a falta de detalhamento dificulta a execução de um método científico adequado. Não encontramos um estudo com acompanhamento das técnicas respondendo às abordagens propostas. Da mesma forma, outro motivo pelo qual essas tecnologias não são expressamente relatadas

como adotadas na prática é também porque a maioria dos estudos apresenta dificuldades que dificultam ou impedem a replicação de seus trabalhos por outros pesquisadores. Isso é importante para obter viabilidade para adoção.

Outro ponto negativo que dificulta a execução de um método científico ideal é a indisponibilidade do banco de dados de imagens utilizado pelos estudos. Apenas três estudos forneceram contato para a solicitação. A fim de fornecer uma base de conhecimento empírica imparcial, os conjuntos de dados devem ser compartilhados entre a comunidade científica. Finalmente, a comparação de técnicas fornece um *feedback* mais amplo quando confrontada com experimentos que usam apenas uma técnica. No entanto, a falta de estudos considerando cenários reais torna sua adoção na prática muito difícil, portanto, embutida de baixa confiabilidade.

Para superar essas questões, estudos futuros na área devem adotar métodos de comparação com outras propostas semelhantes para detecção de doenças. Assim, esta revisão sistemática da literatura fornece um grande valor para pesquisas futuras, uma vez que: forneceu à literatura da área métodos de comparação; incrementou o estado da arte com um mapa para futuras análises de trabalhos relacionados; serve como um guia para seleção dos estudos que apresentaram melhor desempenho na detecção da doença, sendo útil também para a tomada de decisão.

4 EXPERIMENTOS CONTROLADOS

O foco principal deste estudo foi comparar três abordagens para RNCs desenvolvidas com foco na detecção da ferrugem na folha da soja. Para tal, três experimentos controlados foram necessários. Juntamente com a utilização das RNCs, os estudos variaram em termos de técnicas de aumento de dados, assim como de algoritmos utilizados durante o treinamento das RNCs.

Este capítulo apresenta os três experimentos conduzidos neste estudo. Na Seção 4.2 são discutidos o formato e os resultados do primeiro experimento. Na Seção 4.3 é apresentado uma variação e a utilização de um algoritmo de aumento de dados não utilizada no primeiro experimento, e portanto o desenvolvimento e resultados do segundo experimento. Por fim, o terceiro experimento, com a variação do algoritmo de aumento de dados, é apresentado na Seção 4.4.

Os três experimentos visam avaliar três diferentes conjuntos de arquiteturas e modelos de RNCs e a utilização ou não de um algoritmo de enriquecimento de dados na etapa de pré-processamento de imagens. Para que sejam estudos comparáveis, adotou-se protocolos equivalentes, variando as hipóteses e objetivos principais.

4.1 Protocolo e Metodologia dos Experimentos

Nesta seção serão apresentadas a metodologia e o protocolo de pesquisa utilizado nos três experimentos. O restante do protocolo de cada um dos experimentos é apresentado em suas respectivas seções. Nesta seção estão as informações redundante que não caberiam por repetidamente em cada uma das seções dos experimentos.

Os protocolos utilizados nos três experimentos foram definidos e conduzidos a partir do livro de (WOHLIN et al., 2012), baseando-se em uma norma sólida para um bom desenvolvimento do protocolo. Esta seção resume os principais recursos dos nossos protocolos de pesquisa, enquanto o protocolo completo de cada um dos experimentos é disponibilizado em suas respectivas seções.

4.1.1 Base de Imagens

A base de imagens utilizada nos três experimentos possui um total de 65 imagens da folha de soja infectada com os sintomas da doença da ferrugem e 65 imagens da folha de soja saudável. A base de imagens utilizada nos experimentos é pública e foi retirada do repositório da Empresa Brasileira de Pesquisa Agropecuária (Embrapa) denominado Digipathos (BARBEDO et al., 2018) ¹.

Um exemplo das imagens que compõem a base de dados utilizada nos três experimentos pode ser consultada na Figura 24.

¹ <https://www.digipathos-rep.cnptia.embrapa.br>

Figura 24 – Imagem da folha da soja com sintomas de ferrugem.



Fonte:(BARBEDO et al., 2018)

4.1.2 Ferramentas

Para o desenvolvimento deste trabalho foi utilizado um computador com sistema operacional Ubuntu 18.04, um hardware com 6 GB de RAM, disco com capacidade de 1T, CPU Intel Core i5-3337U com 4 *threads* e 2 núcleos. Também foi utilizado as bibliotecas TensorFlow e Keras, ambas na linguagem de programação Python. A linguagem de programação e as bibliotecas utilizadas foram selecionadas por sua capacidade de desenvolvimento e por serem escritas em código aberto.

A versão das bibliotecas e da linguagem de programação são descritas a seguir:

- *Versão Tensorflow*: 1.5.0
- *Versão Keras*: 2.2.5
- *Versão Python*: 2.7.15

4.1.3 Metodologia dos Experimentos

Os três experimentos utilizaram uma metodologia composta por três etapas:

1. *pré-processamento e organização das imagens*: nesta etapa ambos os experimentos realizaram a organização das imagens utilizadas no estudo. O primeiro experimento fez uso do banco de imagens original (sem a utilização de algoritmo para o aumento de dados). Já os outros dois experimentos utilizaram algoritmos de enriquecimento de dados, aumentando assim o número de imagens.
2. *desenvolvimento do modelo RNC*: nesta etapa nos três experimentos foram desenvolvidos as arquiteturas RNCs utilizando a API *Keras* e a biblioteca *TensorFlow*. A linguagem Python foi utilizada como linguagem de programação padrão;

3. *coleção de métricas*: etapa em que a matriz de confusão e as fórmulas de acurácia, taxa de erro, sensibilidade, especificidade e eficiência foram utilizadas para gerar os resultados obtidos de cada experimento.

4.2 Primeiro Experimento

O primeiro experimento deu início a criação das arquiteturas RNCs e aos testes das mesmas na detecção da ferrugem na folha da soja. Neste experimento foi utilizado o conjunto de dados original (sem alteração ou aumento do conjunto inicial de imagens) nas etapas de treinamento, teste e validação. O objetivo principal foi testar as arquiteturas RNCs criadas utilizando as imagens disponíveis, e encontrar a arquiteturas que alcançaram os melhores resultados.

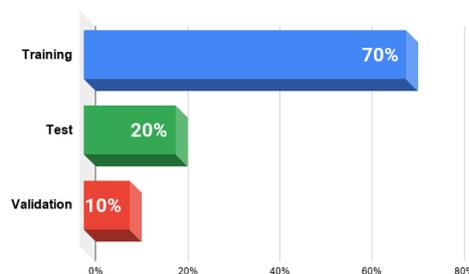
4.2.1 Protocolo do Primeiro Experimento

O protocolo foi definido e conduzido a partir do livro de (WOHLIN et al., 2012), como apresentado na seção 4.1.

4.2.1.1 Etapa 1: Pré-processamento de Imagens do Primeiro Experimento

Nesta etapa foi feita a preparação do banco de dados que posteriormente foi utilizado nas etapas de treinamento, teste e validação. O número total de imagens foi dividido em 70% dos dados para o treinamento, 20% dos dados para os testes e 10% para a validação, como mostra a Figura 29.

Figura 25 – Divisão da base de dados



Fonte: Autor

Na Tabela 10 é apresentado a divisão de dados utilizada no primeiro experimento, na qual foi feita utilizando as imagens originais sem a aplicação de técnicas de aumento de dados.

Tabela 10 – Divisão da base de imagens original utilizada no primeiro experimento

	Divisão das Imagens da Soja com Sintomas da Ferrugem	Divisão das Imagens da Soja Saudável
Treinamento	45	45
Teste	13	13
Validação	7	7
Total	65	65

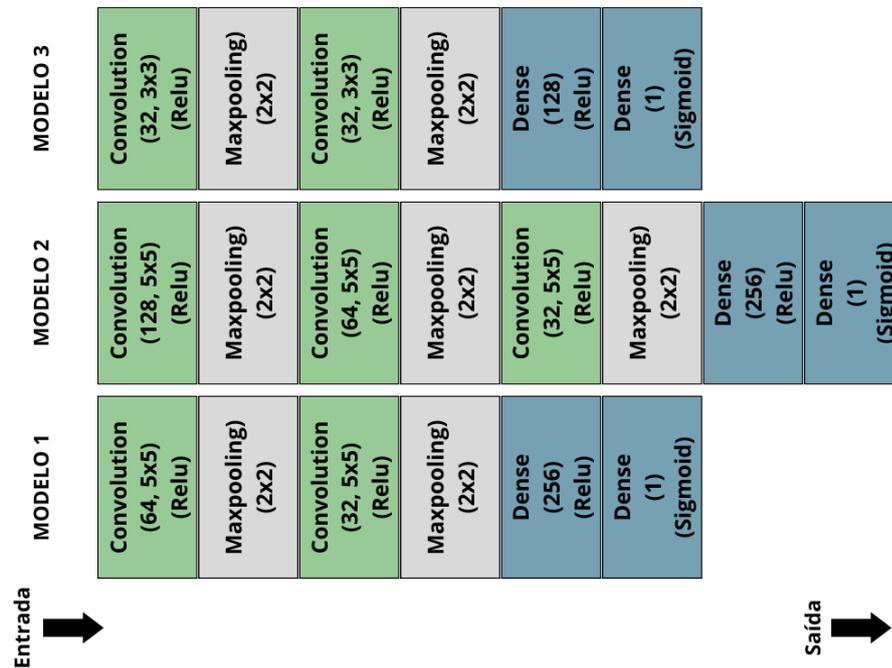
4.2.1.2 Etapa 2: Desenvolvimento dos Modelo de RNCs para o Primeiro Experimento

No desenvolvimento deste experimento, vários modelos foram testados para o treinamento, testes e validação. Ao final, selecionamos os 3 modelos que apresentaram os melhores resultados. Todos os modelos foram treinados e testados utilizando 70% do conjunto total de imagens para treinamento e 20% do conjunto total de imagens para teste. Os 10% restantes foram utilizados na fase de validação.

As três arquiteturas utilizadas nos 3 modelos são apresentadas na Figura 26. Como mostra a figura, os modelos 1, 2 e 3 utilizaram três arquiteturas diferentes, nas quais foram as que alcançaram os melhores resultados. Os três modelos utilizaram um número de épocas igual a 100 como padrão, isso porque a utilização de um número menor ou maior de épocas foi ineficaz, com resultados inferiores aos alcançados utilizando 100 épocas.

Conforme mostrado na Figura 26, os modelos 1 e 3 possuem duas camadas de convolução, duas camadas de *pooling* e duas camadas densas (totalmente conectadas), sendo uma delas a camada de saída. O modelo 2, por outro lado, possui 3 camadas de convolução, 3 camadas de *pooling* e 2 camadas densas. Os demais modelos que não foram selecionados e apresentados na figura tiveram diversas variações desses modelos, como o número de camadas, o tamanho do *kernel*, diferentes funções de ativação, números de épocas, entre outras diferenças importantes, mas que resultou uma eficiência inferior aos apresentados.

Figura 26 – Modelos criados no primeiro experimento



Fonte: Autor

4.2.1.3 Etapa 3: Coleta de Métricas para o Primeiro Experimento

A coleta de métricas e análise dos resultados foi realizada em cima das métricas apresentadas na Seção 2.2.1.3, onde são explicadas as métricas utilizadas no presente estudo.

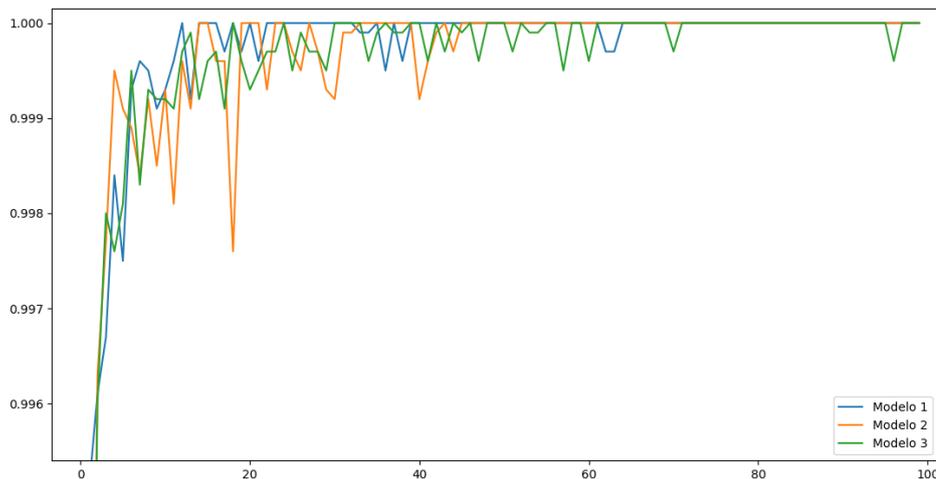
4.2.1.4 Resultados do Primeiro Experimento

Após a etapa de pré-processamento do conjunto de imagens, na qual realizamos a organização das imagens e a divisão das mesmas em treinamento, teste e validação, foi dado início a etapa da criação das arquiteturas e modelos de RNCs. Nesta etapa realizamos o treinamento, testes e validações dos nossos modelos utilizando o banco de imagens disponível. Após as etapas mencionadas anteriormente, foi calculado os dados e realizado a análise dos mesmos, utilizando as métricas de avaliação para identificar os pontos positivos e negativos dos modelos.

A primeira avaliação e análise realizada neste experimento contou com os resultados obtidos nas fases de treinamento e teste. Os resultados são dos três modelos que obtiveram os melhores resultados quando comparados com os demais, por isso, apenas os três estão sendo apresentados. Assim, na Figura 27 é apresentado os resultados obtidos pelos três modelos. No eixo X é possível visualizar a época em que o resultado foi obtido, indo de 0 a 100. Já no eixo Y é apresentado os resultados (acurácia) de 0 a 1 obtidos por cada um dos modelos em sua respectiva época. É possível verificar que nas primeiras

épocas todos os modelos possuem resultados com acurácia menor que nas ultimas épocas, na qual os resultados estão com uma acurácia maior.

Figura 27 – Resultado dos modelos do primeiro experimento utilizando a base de imagens original



Fonte: Autor

Alguns dos demais modelos criados durante o desenvolvimento deste experimento possuem o número de épocas diferentes (menores e maiores que 100 épocas), por conta dos resultados inferiores aos encontrados nos modelos escolhidos, não foram colocados em gráficos para comparação, apenas os 3 melhores.

4.2.1.5 Validação do Primeiro Experimento

Após as etapas de treinamento e testes, os modelos propostos foram validados. A validação, ao contrário do treinamento e dos testes, não tem rótulo nas imagens. No treinamento e nos testes, foi introduzido rótulos nas imagens, indicando quais imagens são da folha de soja saudável e quais são da folha da soja infectada. Na validação, para testar o algoritmo, o mesmo deve informar se as imagens utilizadas para validação fazem parte da classe de plantas sãs ou da classe de plantas doentes.

A etapa de validação conta com 10% do conjunto total de imagens, tanto para o conjunto de imagens da folha da soja doente quanto para as imagens da folha da soja sadia. Na Tabela 14 é apresentado os resultados de validação dos três modelos selecionados. A tabela conta com todos os resultados extraídos conforme as métricas de avaliação e matriz de confusão.

Tabela 11 – Resultados dos modelos 1, 2 e 3 utilizando o conjunto de imagens original sem a utilização da técnica de aumento de dados

Métricas	Modelo 01	Modelo 02	Modelo 03
VP	7	7	7
FP	2	2	4
VN	5	5	3
FN	0	0	0
Acurácia	0,8571	0,8571	0,7142
Taxa de erro	0,1429	0,1429	0,2850
Sensibilidade	1	1	1
Especificidade	0,7142	0,7142	0,4285
Eficiência	0,8571	0,8571	0,7142

4.2.1.6 Discussões do Primeiro Experimento

Os resultados apresentados na Seção 4.2.1.5 nos mostram como há uma variedade de possíveis modelos a serem criados quando se utiliza RNCs. Em nosso estudo foram criados várias arquiteturas e modelos, finalizando assim com os 3 que apresentaram os melhores resultados. Desses três modelos criados, dois apresentaram resultados idênticos e melhores quando comparados ao outro modelo. Os modelos que apresentaram os melhores resultados foram os modelos 01 e 02. O modelo que obteve os piores resultados foi o modelo 03. Os modelos 01 e 02 foram superiores tanto nos valor de acurácia quanto no resultado da eficiência, confirmando assim a eficácia dos modelos.

Os resultados obtidos neste experimento foram obtidos durante uma pesquisa inicial, para testar a eficiência das arquiteturas RNCs no treinamento, teste e validação na detecção da ferrugem na soja com o número de imagens disponível. Os resultados não são resultados de alta porcentagem e também não são resultados ruins. Com isso, os modelos podem ser utilizados na detecção de doenças, porém é possível obter uma melhora nos resultados obtidos, aplicando outras técnicas e algoritmos em conjunto com os modelos de RNCs. Considerando que o número de imagens não é um número muito alto, foi optado a utilização de técnicas de aumento de dados para testar as arquiteturas e verificar se haverá aumento na acurácia e eficiência dos resultados obtidos.

4.2.2 Considerações Finais do Primeiro Experimento

Nesta primeira etapa, realizamos um experimento com o objetivo em criar e testar as arquiteturas e modelos de RNCs criadas para a detecção da ferrugem nas folhas da soja utilizando as bibliotecas *tensorflow* e *keras* e a linguagem de programação *python*. Os modelos que alcançaram maior eficiência foram os modelos 1 e 2, ficando com os resultados inferiores o modelo 3. Os modelos 1 e 2 alcançaram uma acurácia igual a 0,8571 e eficiência de 0,8571. Já os resultados obtidos pelo modelo 03 foram de uma acurácia igual a 0,7142 e eficiência igual a 0,7142.

Conclui-se com os resultados encontrados que as arquiteturas e modelos criados obtiveram resultados parcialmente bons, na qual não foram ótimos ou ruins. Mesmo sem a utilização de algoritmos para o enriquecimento de dados os modelos chegaram a uma acurácia superior a 80% e uma eficiência acima de 80% também. Com os resultados encontrados neste experimento fazendo uso do banco de imagens original, nos próximos experimentos optou-se pela utilização de algoritmos de enriquecimento de dados em busca de resultados superiores aos encontrados no presente experimento. Com isso, nas próximas seções é apresentado o segundo e terceiro experimento.

4.3 Segundo Experimento - Avaliação do Algoritmo Superpixel SLIC

O primeiro experimento foi iniciado com o objetivo em encontrar as arquiteturas de RNCs com os melhores resultados na detecção de ferrugem nas folhas da soja. Já no segundo experimento, o objetivo foi testar o algoritmo denominado *Simple Linear Iterative Clustering* (SLIC) para aumentar a quantidade de imagens do nosso banco de imagens. Foi escolhido utilizar um algoritmo de aumento de dados no segundo experimento para tentar superar os resultados encontrados no primeiro experimento. Outro ponto pela qual foi escolhido utilizar o SLIC para o aumento dos dados é os resultados do estudo de revisão sistemática de literatura, na qual um dos desafios encontrados no uso de técnicas automáticas para o domínio de detecção de doenças em soja foi a falta de dados.

O uso de técnicas de aumento de dados pode ser uma tarefa desafiadora. Uma possível hipótese derivada para este experimento é que o enriquecimento de imagens com técnicas de pré-processamento pode contribuir para o treinamento de redes neurais convolucionais. Assim, como primeiro passo para testar esta hipótese mais geral, derivou-se a seguinte hipótese específica deste experimento.

Hipótese: O algoritmo de aumento de dados conhecido por SLIC apresenta resultados para o treinamento de RNCs, utilizadas na detecção automática da ferrugem em folhas de soja, resultados estes que são próximos, mas ainda inferiores, à um banco de dados de imagens sem pré-processamento e que foram coletadas em cenário real.

Objetivo: Para testar esta hipótese, definiu-se o seguinte objetivo principal deste experimento: verificar a eficiência do algoritmo SLIC como técnica de pré-processamento em conjunto com as RNCs, ou seja, como método de classificação, em comparação com uma base de imagens sem pré-processamento.

O experimento realizado neste estudo segue o protocolo definido por (WOHLIN et al., 2012). O restante deste experimento está estruturado da seguinte forma: na Seção 4.3.1 é apresentado o protocolo utilizado para o desenvolvimento do experimento; na subseção 4.3.1.1 é apresentado o pré-processamento utilizado no segundo experimento, o desenvolvimento dos modelos RNCs na subseção 4.3.1.2 e a coleta das métricas especificadas em subseção 4.3.1.3; e por fim, em subseção 4.3.2 são apresentadas as considerações finais.

4.3.1 Protocolo do Segundo Experimento

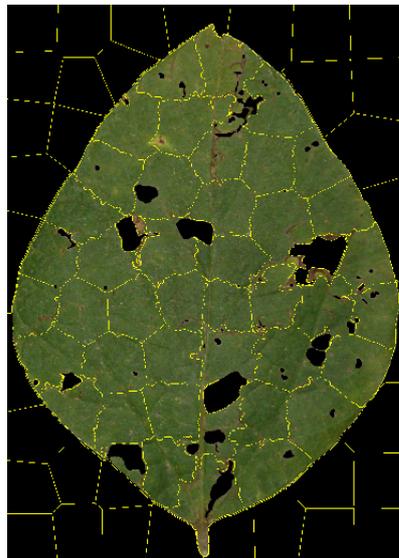
Assim como no primeiro experimento, o protocolo do segundo experimento foi definido e conduzido a partir do livro de (WOHLIN et al., 2012). Esta seção resume os principais recursos de nosso protocolo de pesquisa, enquanto o protocolo completo está disponível no repositório experimental em *link*.

4.3.1.1 Etapa 1: Pré-processamento das Imagens do Segundo Experimento

No pré-processamento das imagens, usamos o algoritmo de segmentação SLIC. De acordo com (ACHANTA et al., 2012), o algoritmo SLIC é uma adaptação do algoritmo *K-means* para gerar regiões semelhantes, chamadas de *superpixels*. O parâmetro K do algoritmo SLIC está relacionado ao tamanho do *superpixel* sobre a imagem, em nosso trabalho definimos um tamanho de *superpixel* adequado para nosso estudo.

Levando em consideração o número de imagens em nosso banco de dados, usamos esta técnica para aumentar o número de imagens e testar a mesma no treinamento dos modelos RNCs criados, visando também superar os resultados obtidos no primeiro experimento. Geralmente, quanto maior o número de imagens, mais benéfico se torna o treinamento das RNCs, pois pode facilitar o aprendizado dos modelos. Um exemplo prático da utilização da técnica de SLIC usada em nosso experimento pode ser visto na Figura 28.

Figura 28 – Exemplo da aplicação do algoritmo SLIC em imagem da folha da soja doente.

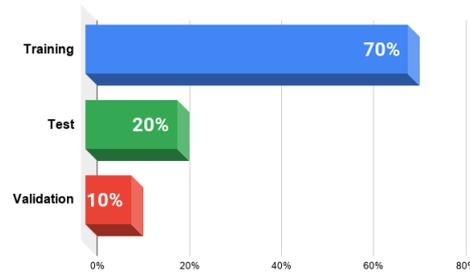


Fonte: Autor

Após a aplicação da técnica de segmentação SLIC, de 65 imagens, foi obtido um total de 1.720 imagens de folhas de soja com sintomas de ferrugem, um aumento que pode ser significativo para o treinamento das RNCs. As imagens da folha de soja sã, após a segmentação, de 65 imagens, obteve-se um total de 1.207 imagens. Após a aplicação do

SLIC, a base de imagens foi dividida utilizando a técnica de validação cruzada. Nesta técnica, foi definido que 70% das imagens das folhas doentes e saudáveis seriam separadas para o treinamento da RNC, 20% das imagens das folhas doentes e saudáveis destinadas aos testes, e por fim, para a validação, 10% das imagens das folhas doentes e sãs foram separadas, como mostra a Figura 29.

Figura 29 – Divisão da base de dados



Fonte: Autor

A ampliação da base de imagens usando SLIC e a divisão das imagens usando validação cruzada podem ser vistas na Tabela 12.

Tabela 12 – Base de imagens após processo de aumento de dados utilizando o algoritmo SLIC.

Agrupamento	Antes do processo de aumento de dados		Após processo de aumento de dados	
	Imagens de folhas infectadas	Imagens de folhas saudáveis	Imagens de folhas infectadas	Imagens de folhas saudáveis
Treinamento	45	45	1.204	845
Teste	13	13	344	241
Validação	7	7	172	121
Total	65	65	1.720	1207

Uma opção importante que o algoritmo SLIC fornece é o ajuste do tamanho dos *superpixels*. Neste trabalho, o tamanho de 60 *superpixels* foi utilizado como padrão, resultando entre 43 e 58 imagens de cada imagem. Um número maior poderia ser colocado, como 100, 200 ou 500, porém, isso resultaria em imagens de baixa qualidade, o que poderia ser prejudicial para o treinamento das RNCs. Portanto, o tamanho de 60 *superpixels* foi adotado como padrão, o que nos dá um aumento significativo de dados e com imagens de boa qualidade.

Os *superpixels* que retornavam apenas com imagens sem dados relevantes (fundo escuro), como mostrado na Figura 28, foram descartados. A diminuição do número de imagens pode auxiliar na redução do uso de poder computacional, focando apenas nas imagens que possuem conteúdo foliar relevante, com ou sem sintomas de ferrugem.

Na Figura 30 o resultado das imagens após a segmentação utilizando o algoritmo SLIC é apresentado.

Figura 30 – Imagens já segmentadas pelo algoritmo SLIC.



Fonte: Autor

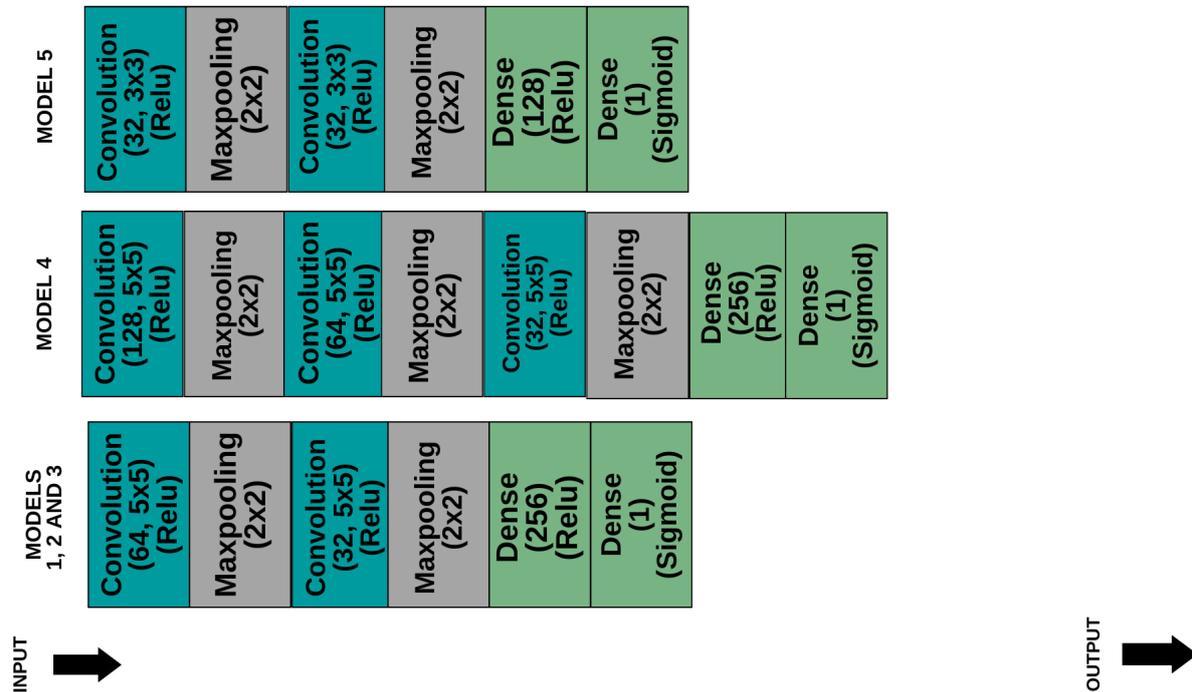
4.3.1.2 Etapa 2: Desenvolvimento dos Modelos de RNCs para o Segundo Experimento

No desenvolvimento deste experimento, vários modelos foram testados para o treinamento, testes e validação. Ao final, selecionamos os 3 modelos que apresentaram os melhores resultados. Todos os modelos foram treinados e testados utilizando 70% do conjunto total de imagens para treinamento e 20% do conjunto total de imagens para teste. Os 10% restantes foram utilizados na fase de validação, que será detalhada na Seção 4.3.1.5.

As duas arquiteturas utilizadas nos 5 modelos são apresentadas em Figura 31. Como mostra a figura, os modelos 1, 2 e 3 utilizaram a mesma arquitetura, mas o que diferencia um modelo do outro é o número de épocas utilizado. No modelo 01 foram utilizadas 60 épocas, no modelo 02 foram utilizadas 100 épocas e, por fim, no modelo 03, 150 épocas. O modelo 4 possui uma arquitetura diferenciada, na qual 100 épocas foram utilizadas para seu treinamento. Finalmente, o modelo 5 também foi treinado e testado usando 100 épocas.

Conforme mostrado na Figura 31, os modelos 1, 2 e 3 possuem duas camadas de convolução, duas camadas de *pooling* e duas camadas densas (totalmente conectadas), uma das quais é a camada de saída. O modelo 4, por outro lado, possui 3 camadas de convolução, 3 camadas de *pooling* e 2 camadas densas. O modelo 5 também tem duas camadas de convolução, duas camadas de *pooling* e duas camadas densas, uma das quais é a camada de saída. Os demais modelos que não foram selecionados tiveram diversas variações desses modelos, como o número de camadas, o tamanho do *kernel*, diferentes funções de ativação, entre outras diferenças importantes, mas que resultaram em números inferiores.

Figura 31 – Modelos criados no segundo experimento.



Fonte: Autor

4.3.1.3 Etapa 3: Coleta de Métricas para o Segundo Experimento

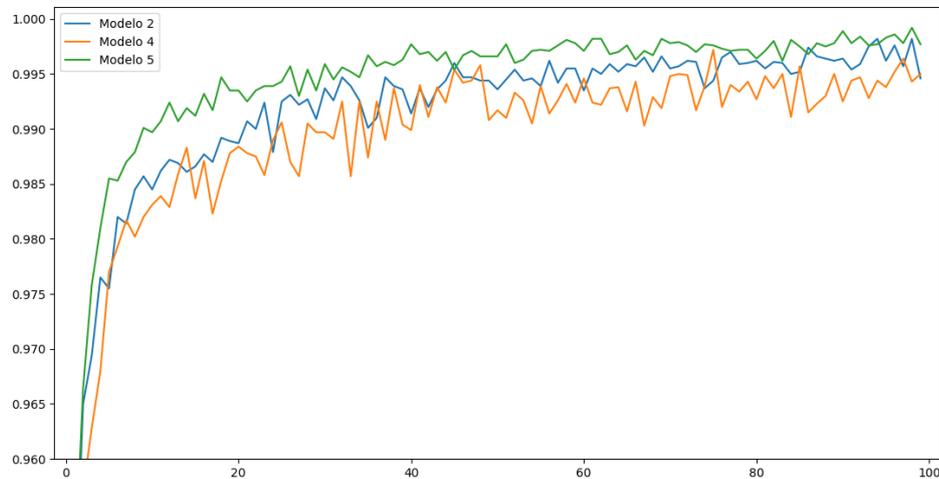
A coleta de métricas e análise dos resultados foi realizada em cima das métricas apresentadas em Seção 2.2.1.3, em que são explicadas as métricas utilizadas no presente experimento.

4.3.1.4 Resultados do Segundo Experimento

Após o pré-processamento do conjunto de imagens, desenvolvimento dos modelos, realizamos o treinamento, os testes e as validações dos nossos modelos criados utilizando nosso banco de imagens. Após o treinamento, teste e validação, calculamos os dados e realizamos uma análise dos mesmos, usando as métricas de avaliação para identificar os pontos positivos e negativos dos modelos.

A primeira etapa desta avaliação contém os resultados após o treinamento e teste dos modelos, na qual analisamos a precisão dos modelos e sua taxa de erro. Na Figura 32 os resultados obtidos pelos modelos 2, 4 e 5 são apresentados usando o conjunto de dados aumentado pelo algoritmo SLIC durante as fases de treinamento e teste.

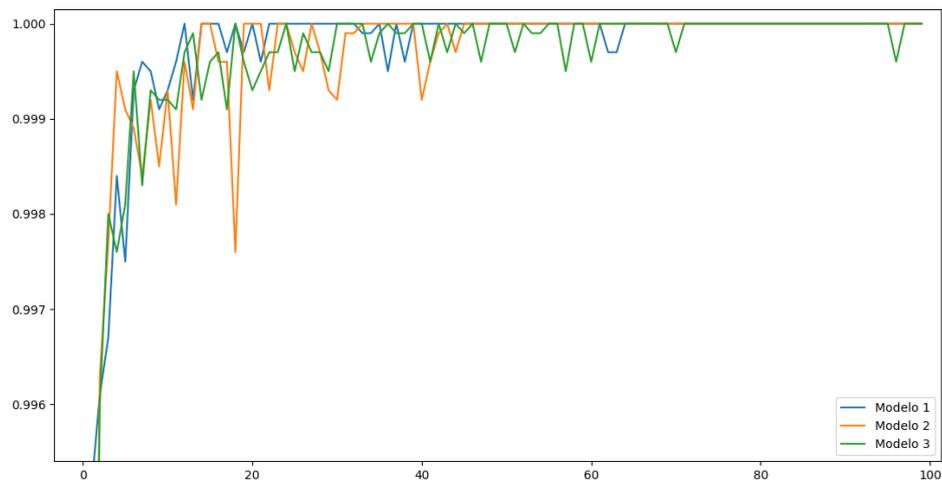
Figura 32 – Resultado de acurácia dos modelos usando o conjunto de dados aumentado pelo algoritmo SLIC



Fonte: Autor

O segundo conjunto de resultados foi obtido utilizando os mesmos modelos, com diferença no conjunto de imagens em que foram utilizadas as imagens originais, sem aumentar os dados e com o número original de imagens. Assim, na Figura 33 os resultados são apresentados.

Figura 33 – Resultado da precisão dos modelos usando o conjunto de dados sem o aumento dos dados



Fonte: Autor

Os modelos 1 e 3 possuem o número de épocas diferentes (60 e 150 épocas conse-

quentemente), por isso não foram colocados em um gráfico para comparação. Apresentaremos os resultados de validação desses modelos na subseção seguinte e o motivo da apresentação dos resultados desses modelos na etapa de validação também será justificado.

4.3.1.5 Validação do Segundo Experimento

Após as etapas de treinamento e testes, os modelos propostos foram validados. A validação, ao contrário do treinamento e dos testes, não tem rótulo nas imagens, assim como no primeiro experimento. No treinamento e nos testes foi utilizado rótulos nas imagens, indicando quais imagens eram da folha da soja saudável e quais eram da folha da soja infectada. Na validação, para testar o algoritmo, as imagens utilizadas não possuem rótulos, portanto o algoritmo deve informar se as imagens utilizadas para validação fazem parte da classe de plantas sãs ou da classe de plantas doentes. Na validação, usamos 10% do conjunto total de imagens.

Na Tabela 13, os resultados dos modelos são apresentados a partir do conjunto de imagens que passaram pelo aumento de dados na etapa de validação. Usando as fórmulas apresentadas na Seção 2.2.1.3 com base na matriz de confusão foi obtido os resultados apresentados na Tabela 13.

Tabela 13 – Resultados dos modelos 2, 4 e 5 utilizando o conjunto de imagens aumentado pela técnica SLIC

Métricas	Modelo 02	Modelo 04	Modelo 05
VP	83	39	26
FP	0	0	0
VN	121	121	121
FN	89	133	146
Acurácia	0,6962	0,5460	0,5017
Taxa de erro	0,3038	0,4540	0,4983
Sensibilidade	0,4825	0,2267	0,1511
Especificidade	1	1	1
Eficiência	0,7412	0,6133	0,5755

A segunda tabela de resultados obtidos pelos modelos utilizando o conjunto de imagens originais na etapa de validação é apresentada na guia Tabela 14.

Tabela 14 – Resultados dos modelos 2, 4 e 5 utilizando o conjunto de imagens original sem a utilização da técnica de aumento de dados

Métricas	Modelo 02	Modelo 04	Modelo 05
VP	7	7	7
FP	2	2	4
VN	5	5	3
FN	0	0	0
Acurácia	0,8571	0,8571	0,7142
Taxa de erro	0,1429	0,1429	0,2850
Sensibilidade	1	1	1
Especificidade	0,7142	0,7142	0,4285
Eficiência	0,8571	0,8571	0,7142

Por fim, a Tabela 15 apresenta os resultados obtidos pelos modelos 1, 2 e 3 em fase de validação. Esses resultados foram obtidos por meio das imagens resultantes do aumento de dados realizado pelo algoritmo SLIC no treinamento, teste e validação.

Tabela 15 – Resultados dos modelos 1, 2 e 3 usando o conjunto de imagens aumentado com a técnica SLIC

Métricas	Modelo 01	Modelo 02	Modelo 03
VP	64	83	61
FP	0	0	0
VN	121	121	121
FN	108	89	111
Acurácia	0,6313	0,6962	0,6211
Taxa de erro	0,3687	0,3038	0,3789
Sensibilidade	0,3720	0,4825	0,3546
Especificidade	1	1	1
Eficiência	0,6860	0,7412	0,6773

4.3.1.6 Discussões do Segundo Experimento

Os resultados obtidos pelos modelos 1 e 3 em conjunto com o modelo 2 na etapa de validação dos modelos na subseção 4.2.1.5 foram apresentados para mostrar o limite no número de épocas utilizadas. O modelo 1 possui 60 épocas, o modelo 2 100 e o modelo 3 150. Todos esses modelos possuem a mesma arquitetura de RNC, sendo diferentes apenas no número de épocas. Quando aumentamos o número de épocas de 60 para 100, foi obtido uma melhora na acurácia e também na eficiência. Quando aumentamos o número de épocas para 150, os resultados de acurácia foram menores do que o modelo com 100 épocas, bem como uma eficiência inferior. Com esses resultados, fica comprovado que somar um número maior que 100 épocas não iria trazer melhores resultados.

Os resultados comparativos entre os modelos usando imagens de diferentes conjuntos de imagens (imagens originais e imagens ampliadas por SLIC) também são apresentados na subseção 4.2.1.5. Esses resultados mostram que os modelos que utilizam as

imagens originais obtiveram melhores resultados quando comparados aos modelos que utilizam as imagens pré-processadas, tanto no valor de acurácia quanto no valor de eficiência. O modelo que obteve melhor resultado com as imagens pré-processadas foi o modelo 2, e os piores resultados foram obtidos pelo modelo 5. Já quando se utilizou as imagens originais, os modelos 2 e 4 obtiveram resultados favoráveis. O modelo 5 obteve os piores resultados com as imagens originais. Outro detalhe é que o modelo 5, que foi o pior, com as imagens originais, obteve uma melhor acurácia em relação ao modelo 2 com as imagens pré-processadas pelo algoritmo SLIC, porém obteve menor eficiência.

Com os resultados apresentados, conclui-se que uma das hipóteses é que algumas imagens decorrentes do aumento dos dados foram rotuladas incorretamente como doentes. Isso porque uma folha inteira doente pode ter partes saudáveis e também partes doentes. Com isso, alguns *superpixels* resultantes das imagens doentias, embora saudáveis, poderiam ter sido adicionados à classe das doentes, confundindo o modelo de RNC. Para que esse problema não ocorresse, seria necessário a avaliação de um técnico da área para rotular imagem por imagem (resultante do pré-processamento) adicionando-as na classe correta. Se a parte de uma imagem (*superpixel*) resultante do pré-processamento for comprovada como saudável pelo técnico, ela será rotulada como saudável e colocada na classe das imagens de folha da soja sã, não na classe de imagens da folha da soja doente.

Os resultados obtidos neste experimento, nos mostram a importância de avaliar todas as hipóteses e resultados, e que o aumento dos dados nem sempre auxilia no treinamento dos modelos, pois podem ocorrer erros durante a execução dos mesmos prejudicando assim a execução. Mesmo com o menor número de imagens no treinamento com as imagens originais, os resultados do primeiro experimento foram melhores.

4.3.2 Considerações Finais do Segundo Experimento

A evolução e o aprimoramento das técnicas de aumento de dados e classificação automática de doenças podem ajudar cada vez mais os agricultores. Os estudos nestas áreas são de suma importância para a evolução da produção agrícola, e pode cada vez mais evitar problemas que possam ser encontrados por investigadores e técnicos. Neste estudo, realizamos um experimento com o objetivo de testar o processo de aumento de dados na etapa de pré-processamento usando o algoritmo SLIC e a classificação da ferrugem da soja usando modelos de RNCs criados com o uso das bibliotecas *tensorflow* e *keras*. O número de imagens ampliadas no pré-processamento foi eficiente, porém os resultados dos modelos que utilizaram as imagens ampliadas não foram motivadores. Os resultados que obtiveram maior eficiência foram os modelos treinados com as imagens originais sem o aumento do conjunto de imagens. O motivo pelo qual os valores não foram eficientes pode ter sido derivado de folhas de soja que apresentam diferentes condições, concluindo assim a necessidade de uma avaliação por um técnico antes de utilizar essas imagens para métodos de detecção automática. O melhor resultado obtido com a utilização das

imagens que passaram pelo processo de pré-processamento foi de acurácia igual a 0,6962 e eficiência de 0,7412. Os resultados obtidos com o uso das imagens originais foram de acurácia igual a 0,8771 e eficiência igual a 0,8571.

Com isso, conclui-se que os resultados obtidos pelo primeiro experimento foram melhores que os resultados obtidos pelo segundo experimento. Após os resultados obtidos nos dois experimentos, foi decidido partir para o terceiro experimento, na qual o objetivo principal foi a utilização de outro algoritmo de enriquecimento do banco de imagens. O terceiro experimento utilizou o algoritmo *augmentor* como técnica padrão para o aumento dos dados e os modelos de RNCs criados. Na seção seguinte será apresentado o terceiro experimento deste trabalho.

4.4 Terceiro Experimento - Avaliação do Algoritmo Augmentor

O terceiro experimento, assim como o primeiro e segundo, também visa investigar as capacidades das RNCs para a detecção da doença ferrugem na folha da soja. A principal diferença entre o terceiro e os outros dois experimentos, é a utilização de um algoritmo diferenciado para o aumento dos dados denominado *augmentor*.

Como observado no segundo experimento, os resultados utilizando o banco de dados original (sem pré-processamento) foram mais motivadores do que os resultados utilizando o banco de imagens aumentado (utilizando o algoritmo SLIC). Portanto, ficou uma dúvida quanto se a escolha pelo SLIC foi uma boa opção para o pré-processamento das imagens, o que abriu caminho para a utilização de outras alternativas.

Já neste experimento foi utilizado o algoritmo *augmentor*, também devotado para o aumento de dados, visando obter melhores resultados comparados aos resultados obtidos pelo primeiro experimento que utilizou imagens sem o pré-processamento de imagens. Assim, derivou-se a seguinte hipótese:

Hipótese: A hipótese investigada neste estudo é a de que o algoritmo *augmentor* apresenta resultados superiores ao SLIC e que os modelos criados neste experimento alcancem resultados superiores aos do primeiro experimento.

Objetivo 1: O objetivo principal deste experimento é verificar a eficiência do algoritmo *augmentor* para o aumento do banco de dados.

Objetivo 2: Encontrar a melhor arquitetura de RNC, com os melhores resultados nas fases de validação.

O experimento realizado neste estudo segue o protocolo definido por (WOHLIN et al., 2012). O restante deste experimento está estruturado da seguinte forma: na subseção 4.4.0.1 é apresentado o pré-processamento do terceiro experimento, o desenvolvimento dos modelos de RNCs é apresentado na subseção 4.4.0.2 e na subseção 4.4.0.3 é especificado a coleta das métricas; e por fim, na subseção 4.4.1 são apresentadas as considerações finais do presente experimento.

4.4.0.1 Etapa 1: Pré-processamento das Imagens do Terceiro Experimento

No pré-processamento das imagens, foi utilizado o algoritmo denominado *augmentor*. *Augmentor* é um pacote *python* projetado para auxiliar no aumento e geração artificial de dados de imagens para tarefas de aprendizado de máquina. É principalmente uma ferramenta de aumento de dados, porém, também incorporara a funcionalidade básica de pré-processamento de imagens.

Levando em consideração o número de imagens em nosso banco de dados, usamos esta técnica para aumentar o número de imagens do mesmo, assim como foi feito no segundo experimento. Porém neste estudo trocamos a técnica, por conta dos resultados ineficazes obtidos no segundo experimento. Um exemplo prático da utilização do algoritmo *augmentor* utilizada em nosso experimento pode ser visto na Figura 34. As três folhas apresentadas na figura são imagens da mesma folha da soja com sintomas de doença. A diferença é que a primeira folha da esquerda para a direita é a imagem original, já as duas imagens a direita são as alterações da primeira imagem após a aplicação da técnica do algoritmo *augmentor*.

Figura 34 – Exemplo da utilização do algoritmo *augmentor*



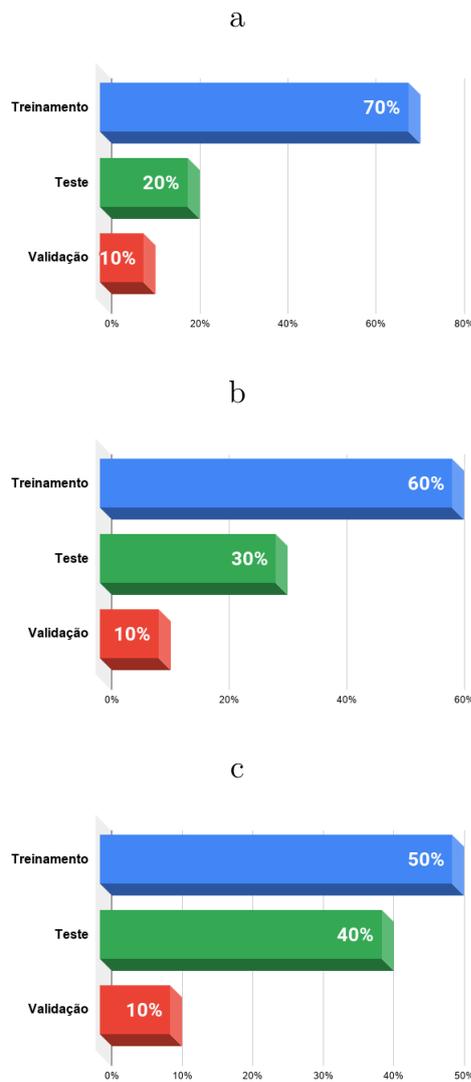
Fonte: Autor

É possível observar como o algoritmo procede, podendo ser alterado a posição do objeto em vários tipos de ângulos. Neste estudo utilizamos várias alterações aleatórias, mudando de imagem para imagem, para se ter imagens variadas e distintas. Alterando desta maneira as imagens, o algoritmo não “nota” que a folha é a mesma, ele realiza o treinamento como se fosse folhas distintas, resultando assim em um aumento eficaz no número de imagens.

Após a aplicação da técnica de segmentação *augmentor*, de 65, foi obtido um total de 1.495 imagens de folhas de soja com sintomas de ferrugem. As imagens da folha de soja sã, após segmentação, de 9 imagens, obteve-se um total de 236 imagens. Após a aplicação do *augmentor*, a base de imagens foi dividida utilizando a técnica de validação cruzada. Nesta técnica, foi definido 3 tipos de divisão de dados: a primeira, apresentada na Figura 35 (a), conta com 70% das imagens das folhas doentes e saudáveis para o treinamento da RNC, 20% das imagens das folhas doentes e saudáveis destinadas aos testes, e por fim, para a validação, 10% das imagens das folhas doentes e sãs foram separadas;

na segunda divisão, apresentada na Figura 35 (b), foram 60% das imagens das folhas doentes e saudáveis para o treinamento; 30% para o teste e 10% para a validação; por fim, a terceira divisão, apresentada na Figura 35 (c), conta com 50% para o treinamento, 40% para os testes e 10% para a validação.

Figura 35 – Divisões de dados utilizadas no terceiro experimento



Fonte: Autor

A ampliação da base de imagens utilizando o algoritmo *augmentor* pode ser visualizada na Tabela 16.

Tabela 16 – Antes e depois da aplicação do algoritmo de aumento de dados *augmentor*

	Antes do processo de aumento de dados		Após o processo de aumento de dados	
	Sadias	Doentes	Sadias	Doentes
Total	09	65	236	1.495

Na Tabela 17 é apresentado os três tipos de divisão de dados utilizados neste estudo. Após encontrar a arquitetura de RNC que apresentou os melhores resultados, optamos por mudar as divisões das imagens de treinamento, teste e validação. Cada uma das etapas recebeu uma porcentagem diferente de imagens, ficando igual apenas o número de imagens na etapa de validação. Na Tabela 17 é possível observar o número de imagens em cada uma dessas etapas para cada uma das divisões.

Tabela 17 – Divisão das imagens após o aumento de dados.

Divisão de imagens após técnica de aumento de dados				
		Treinamento	Teste	Validação
Divisão de dados 01	Folhas doentes	1213	347	172
	Folhas sadias	165	47	24
Divisão de dados 02	Folhas doentes	1040	520	172
	Folhas sadias	141	71	24
Divisão de dados 03	Folhas doentes	866	694	172
	Folhas sadias	118	94	24

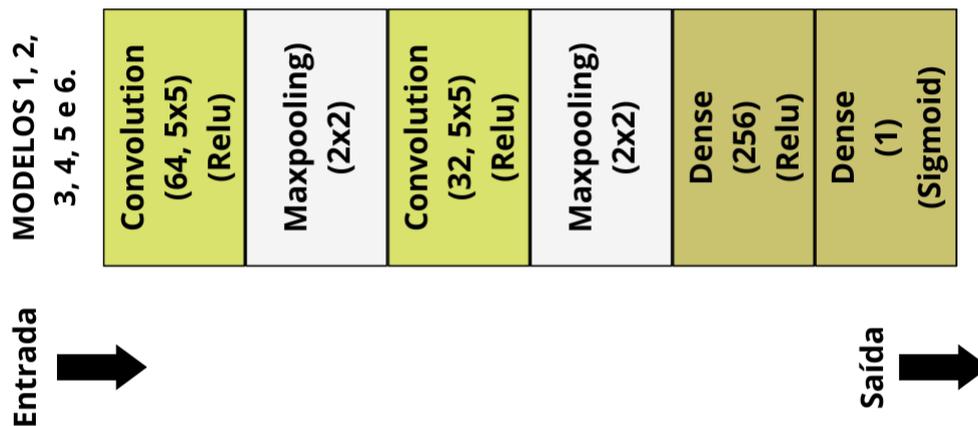
4.4.0.2 Etapa 2: Desenvolvimento do Modelo de RNC para o Terceiro Experimento

Para desenvolver e encontrar o modelo que apresente os melhores resultados, foram testados vários modelos, e o que apresentou os melhores resultados foi selecionado como modelo padrão para ser testado nas diferentes divisões de dados criadas. Estes variados modelos foram testados com diferentes arquiteturas e também com diferentes números de épocas, para assim encontrar a melhor arquitetura e o número correto de épocas. Esse procedimento é feito buscando bons resultados e evitando a ocorrência de *overfitting*, que é um ponto negativo no treinamento das RNCs.

Os modelos criados e selecionados neste experimento são apresentados na Figura 36. Conforme apresentados na figura, os modelos contam com duas camadas convolucionais, duas camadas de *pooling* e duas camadas densas (totalmente conectadas). É possível identificar na Figura 36 que todos os modelos apresentados possuem a mesma arquitetura, porém eles possuem algumas diferenças. O modelo 1, por exemplo, utiliza imagens originais, sem o aumento de dados (antes da utilização do algoritmo *augmentor* para o aumento do banco de imagens). Já os modelos 2, 3 e 4 utilizam divisões do banco de imagens diferenciadas (quantidade de imagens que compõem o treinamento e o teste são diferentes entre os 3 modelos). Já os modelos 5 e 6, possuem o número de épocas diferentes dos demais modelos, tendo um total de 50 épocas o modelo 5 e 120 épocas o modelo 6. A utilização de épocas diferentes se faz necessária para analisar os resultados e

escolher o número que alcança os melhores resultados, definindo assim um número padrão de épocas a ser utilizado.

Figura 36 – Modelos de RNCs utilizados no terceiro experimento



Fonte: Autor

Por fim, vale salientar, que os modelos 1, 2, 3 e 4, fazem uso do mesmo número de épocas (igual a 100). Também é importante mencionar, que os modelos 1, 2, 5 e 6 utilizam a mesma divisão de dados, composta por 70% de imagens para o treinamento, 20% para o teste e 10% para a validação, tanto para as imagens da folha da soja doentes quanto para as imagens da folha da soja saudáveis.

Os demais modelos que não foram selecionados tiveram diversas variações desses modelos, como o número de camadas, o tamanho do *kernel*, diferentes funções de ativação, entre outras diferenças, mas que resultaram em uma menor eficiência. Isso demonstra que o número de arquiteturas que podem ser criadas são grandes, e neste estudo foram criadas mais de 15 arquiteturas diferentes, sendo selecionada as melhores.

4.4.0.3 Etapa 3: Coleta de Métricas do Terceiro Experimento

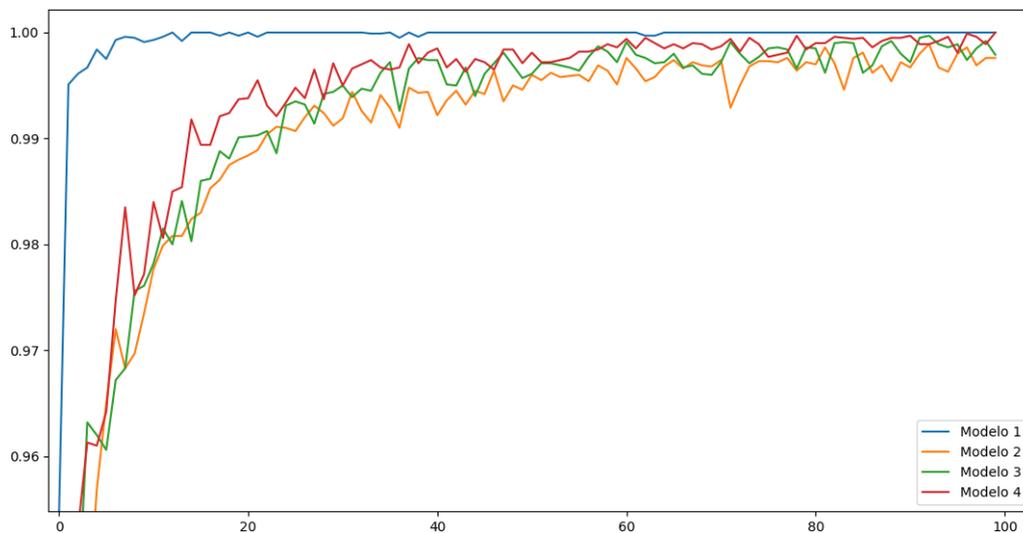
Assim como utilizado no primeiro e segundo experimento, no terceiro experimento, a coleta das métricas e a análise dos resultados também foi realizada em cima das métricas apresentadas na Seção 2.2.1.3, onde são detalhadas as métricas utilizadas no presente estudo.

4.4.0.4 Resultados do Terceiro Experimento

Após as etapas de pré-processamento e desenvolvimento dos modelos RNC, foi realizado o treinamento, teste e validação dos modelos utilizando o banco de imagens aumentado pelo algoritmo *augmentor*. Após as etapas mencionadas anteriormente, foi realizado o cálculo dos dados resultantes e a análise dos mesmos, utilizando as métricas de avaliação para identificar os pontos negativos e positivos dos modelos.

A primeira etapa desta avaliação contém os resultados do treinamento e teste dos modelos propostos. Nestes resultados é apresentado a acurácia dos modelos propostos. Na Figura 37 os resultados obtidos pelos modelos 1, 2, 3 e 4 são apresentados.

Figura 37 – Resultado do treinamento e teste do modelo escolhido com imagens originais e imagens aumentadas



Fonte: Autor

A fim de testar a eficácia dos modelos propostos neste experimentos, utilizamos o modelo 1 como parâmetro de comparação. O modelo 1 foi o modelo que obteve o melhor resultado no primeiro experimento, assim apresentamos os resultados deste modelo juntamente com os resultados do restante dos modelos. O modelo 1 utiliza as imagens originais sem o aumento de dados. Já os modelos 2, 3 e 4 utilizam as imagens com o aumento de dados aplicado (utilizando o algoritmo *augmentor*).

4.4.0.5 Validação do Terceiro Experimento

Após as etapas de treinamento e teste, os modelos de RNCs passaram pela etapa de validação para validar os modelos propostos. Na validação, ao contrário do treinamento e do teste, as imagens não possuem rótulos. Já nas etapas anteriores foi colocado rótulos

nas imagens, indicando quais imagens são da folha da soja saudável e quais são da folha da soja infectada. Na validação, para testar o modelo, as imagens que fazem parte da validação, não possuem rótulos, testando assim a eficácia do algoritmo. Nesta fase, os modelos devem informar quais das imagens fazem parte da classe das folhas doentes e quais imagens fazem parte da classe das folhas sãs. Foi utilizado 10% do conjunto total de imagens na etapa de validação, nos três tipos de divisão de dados, tanto para as imagens da folha da soja doente quanto para as imagens da folha da soja saudável.

Não realizamos neste estudo o treinamento, teste e validação da RNC com as imagens originais (antes do aumento de dados), pois este estudo já foi realizado no primeiro experimento, apresentado na seção 4.2. Apenas utilizamos os resultados do melhor modelo obtidos no primeiro experimento como parâmetro de comparação. Na Tabela 18 é apresentado os resultados obtidos na etapa de validação. Na imagem é possível observar os resultados do modelo 1 (obtidos através das imagens originais) e os modelos 2, 3 e 4 (obtidos através do experimento utilizando as imagens aumentadas pelo algoritmo *augmentor*).

Tabela 18 – Resultados da validação do modelo com as imagens originais e os resultados dos modelos com as imagens aumentadas e com suas respectivas divisões.

Métricas	M01	M02	M03	M04
VP	7	171	170	170
FP	2	13	08	15
VN	5	11	16	09
FN	0	1	02	02
Acurácia	0,8571	0,9285	0,9489	0,9132
Taxa de erro	0,1429	0,0715	0,0515	0,0868
Sensibilidade	1	0,9941	0,9883	0,9883
Especificidade	0,7142	0,4583	0,6666	0,3750
Eficiência	0,8571	0,7262	0,8274	0,6816

Outro resultado importante a ser observado, são as dos modelos que utilizamos como parâmetro para a escolha do número de épocas. Para definir o melhor número de épocas para os modelos, os modelos passam pelo treinamento, teste e validação com número de épocas diferentes, até ser encontrado os melhores resultados. Pois quando o número de épocas é muito grande e os resultados na validação são ruins, é possível que esteja ocorrendo *overfitting*. Na Tabela 19 é apresentado o resultado da validação dos modelos com o número de épocas diferentes. O modelo 2 possui o número de épocas igual a 100 (definido como padrão no experimento). O modelo 5 possui o número de épocas igual a 50. Por fim, o modelo 6 possui o número de épocas igual a 120. Como mostrado na figura, o modelo que obteve os melhores resultados, tanto de acurácia quanto de eficiência, é o modelo 2.

Tabela 19 – Resultados da validação dos modelos com o número de épocas diferente.

Métricas	M05	M02	M06
VP	170	171	172
FP	15	13	22
VN	09	11	02
FN	02	01	0
Acurácia	0,9132	0,9285	0,8877
Taxa de erro	0,0868	0,0715	0,1123
Sensibilidade	0,9189	0,9941	1
Especificidade	0,3750	0,4583	0,0833
Eficiência	0,6469	0,7262	0,5416

4.4.0.6 Discussões do Terceiro Experimento

Nos resultados apresentados anteriormente, na etapa de validação, os modelos 2, 5 e 6 foram apresentados com o objetivo em mostrar o limite no número de épocas utilizado no terceiro experimento. O modelo 5 possui 50 épocas, o modelo 2 possui 100 e o modelo 6 possui 120 épocas. Todos esses três modelos possuem a mesma arquitetura de RNC, diferentes apenas no número de épocas. Quando aumentamos o número de épocas de 50 para 100, foi obtido uma melhora nos resultados tanto na acurácia quanto na eficiência. Quando foi aumentado de 100 para 120 o número de épocas, já tivemos uma piora nos resultados, a acurácia diminuiu assim como a eficiência. Esses resultados confirmaram que aumentando o número de épocas em um número maior que 100, não traria melhoras nos resultados, e sim uma piora.

Os outros resultados apresentados são diretamente ligados ao objetivo principal deste experimento, testar a eficácia dos modelos utilizando o conjunto de imagens originais e também testar os modelos utilizando as imagens aumentadas pelo algoritmo *augmentor*. Os resultados utilizando com as imagens originais foram extraídas do primeiro experimento, na qual já foram calculadas. Neste experimento pegamos a arquitetura de RNC que apresentou os melhores resultados, e aplicamos as divisões de dados utilizadas no treinamento e teste. Assim ficou dividido em 3 três modelos com divisões diferentes (modelos 2, 3 e 4), como apresentado na Figura 35. Analisando os resultados de validação apresentados na Tabela 18, é comprovado que o modelo 3 obteve os melhores resultados, quando comparado aos modelos 1, 2 e 4.

O modelo com os piores resultados (apresentado na Tabela 18 entre os 4 modelos foi do modelo 1, que utilizou as imagens originais sem o aumento dos dados. Com isso, já está confirmado que mesmo o modelo que obteve o pior resultado utilizando as imagens aumentadas, superou o modelo que utilizou as imagens originais. Os modelos 2 e 4 superaram o modelo 1, mas obtiverem resultados inferiores aos resultados obtidos pelo modelo 3, que foi o modelo que alcançou os melhores resultados, analisando tanto a acurácia quanto a eficiência. Mesmo com a acurácia inferior, o modelo 1 obteve o melhor

resultado na eficiência, quando comparado com os demais.

Com esses resultados, conclui-se que a técnica de aumento de dados *augmentor* é eficaz no enriquecimento dos dados, auxiliando assim as arquiteturas de RNCs na detecção da ferrugem na folha da soja. Mesmo todos os modelos tendo a eficiência inferior a do modelo 1, a diferença entre o modelo 3 e 1 é pouca (analisando a eficiência). Já comparando a acurácia, a diferença do modelo 3 para os demais é significativa, tornando-o assim o modelo com os melhores resultados.

4.4.1 Considerações Finais do Terceiro Experimento

O aperfeiçoamento de técnicas de detecção de doenças (como as arquiteturas de RNCs) juntamente com as técnicas de aumento de dados, que complementam as RNCs, podem auxiliar cada vez mais os pesquisadores e produtores agrícolas. Os estudos que envolvem a tecnologia com a agricultura são de uma grande importância, para que cada vez mais possamos evitar problemas como as pragas e as doenças nas lavouras. Neste estudo, realizamos um experimento com o objetivo em encontrar uma arquitetura de RNC (criados com o uso das bibliotecas *tensorflow* e *keras*) que apresentasse resultados motivadores na detecção da doença ferrugem na soja. Outro objetivo foi investigar a utilização do algoritmo *augmentor* na etapa pré-processamento para o enriquecimento da base de dados.

Analisando o número de imagens aumentadas pelo algoritmo *augmentor* pode-se confirmar que foi satisfatório, de 09 imagens de folha da soja saudáveis, alcançamos 236 imagens, e de 65 imagens de folhas da soja doente, alcançamos 1.495 imagens. Os resultados alcançados pelas arquiteturas de RNCs criadas neste experimento também são motivadores. Em comparação ao primeiro experimento na qual foi obtido os melhores resultados utilizando as imagens originais (sem o aumento de dados), o experimento utilizando as imagens aumentadas pelo algoritmo *augmentor* foi mais eficaz. O melhor resultado obtido pelo modelo do experimento 01 (modelo 1) foi de uma acurácia igual a 0,8571 com uma eficiência de 0,8571. Já o modelo que alcançou os melhores resultados utilizando as imagens aumentadas pelo algoritmo *augmentor* (modelo 3) alcançou uma acurácia de 0,9489 e eficiência de 0,8274. Com isso, prova-se que o terceiro experimento obteve resultados superiores aos resultados obtidos nos demais experimentos.

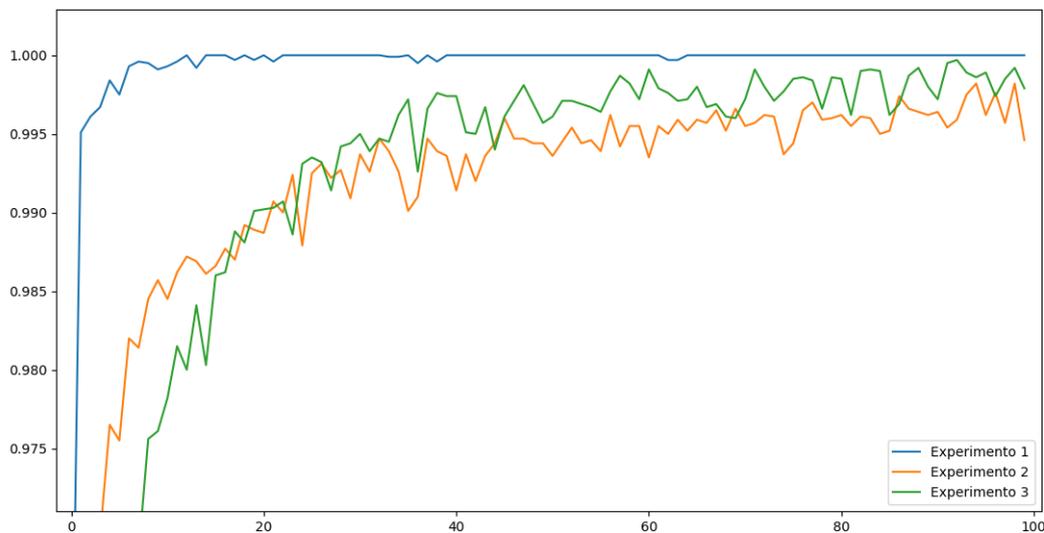
4.5 Comparação dos Três Experimentos

Nesta seção será apresentado uma comparação dos três experimentos conduzidos durante esta dissertação de mestrado a fim de comparar os resultados obtidos pelos três experimentos fazendo uma análise geral destacando os pontos positivos e negativos de cada um dos experimentos.

4.5.1 Comparação dos Resultados Iniciais dos Experimentos

Os modelos criados nos experimentos passaram por algumas etapas até chegarem ao resultado final. Duas das principais etapas do experimento é o treinamento e o teste dos modelos desenvolvidos. As duas etapas são responsáveis por treinar e testar os modelos criados, na qual o próprio algoritmo retorna resultados como a acurácia e a taxa de erro. Na Figura 38 é apresentado os resultados dos melhores modelos de cada um dos experimentos conduzidos. Tendo assim o resultado do modelo que obteve os melhores resultados no primeiro experimento, o modelo com melhores resultados do segundo experimento e o modelo que alcançou os melhores resultados no terceiro experimento.

Figura 38 – Resultados dos melhores modelos dos três experimentos conduzidos



Fonte: Autor

Um detalhe importante na qual não podemos deixar de mencionar é que as imagens nessas etapas são rotuladas, diferente da validação, onde as imagens não são rotuladas. Os melhores resultados entre os experimentos nesta etapa é do modelo utilizado no primeiro experimento, depois o modelo utilizado no terceiro experimento, e por fim, com os piores resultados, o modelo do segundo experimento. O modelo do primeiro experimento apresentou resultados com a porcentagem de acurácia alta, com muitas das épocas alcançando quase 100% de acurácia. O terceiro experimento obteve resultados positivos com valores altos de acurácia, sendo superior ao segundo experimento e inferior ao primeiro experimento. Por fim, o segundo experimento obteve os piores resultados, mesmo sendo resultados ótimos de acurácia.

4.5.2 Comparação dos Resultados de Validação dos Três Experimentos

Outra etapa de suma importância nos experimentos é a etapa de validação, nas quais os modelos são testados utilizando imagens não rotuladas. Nesta etapa os modelos treinados recebem uma imagem de entrada para os modelos gerarem uma saída de verdadeiro ou falso para as folhas da soja doente ou saudável.

Na Tabela 20 é apresentado os modelos que apresentaram os melhores resultados de cada experimento.

Tabela 20 – Resultados de validação dos melhores modelos de cada um dos experimentos.

Métricas	Modelo Experimento 01	Modelo Experimento 02	Modelo Experimento 03
VP	7	83	170
FP	2	0	08
VN	5	121	16
FN	0	89	02
Acurácia	0,8571	0,6962	0,9489
Taxa de erro	0,1429	0,3038	0,0515
Sensibilidade	1	0,4825	0,9883
Especificidade	0,7142	1	0,6666
Eficiência	0,8571	0,7412	0,8274

O modelo que apresentou os piores resultados foi o modelo do segundo experimento. Este modelo utilizou para o treinamento, teste e validação as imagens aumentadas pelo algoritmo SLIC. Os resultados mostram que a utilização deste algoritmo para aumento de dados não é eficaz quando comparado aos resultados obtidos sem a utilização de um algoritmo de aumento de dados ou quando comparados as imagens resultantes do algoritmo de aumento de dados *augmentor*.

O modelo do primeiro experimento obteve o segundo melhor resultado entre os experimentos. Neste experimento utilizamos apenas o banco de imagens original sem qualquer técnica de aumento de dados, contando com um número inferior de imagens da folha da soja sadia e imagens da folha da soja doente.

Por fim, o modelo que obteve os melhores resultados foram as do terceiro experimento. No terceiro experimento foi utilizado o algoritmo *augmentor* para o aumento de dados, na qual foi gerado um aumento significativo no número final de imagens e bons resultados nas etapas de treinamento, teste e validação. Com esses resultados, pode-se concluir que o algoritmo *augmentor* é eficaz no enriquecimento de dados, tornando assim o terceiro experimento o mais eficaz quando comparado aos demais experimentos.

5 CONCLUSÃO

A utilização de técnicas, ferramentas e metodologias na agricultura pode trazer grandes benefícios para os agricultores, pesquisadores e a população em geral. A detecção precoce de doenças pode ser um ponto importante no combate a pragas e doenças nas lavouras de forma rápida e eficaz, dando suporte ao produtor agrícola. Pesquisas aplicadas baseadas em experimentação na detecção de doenças em plantas, em especial as que utilizam técnicas de processamento de imagens e aprendizado de máquina, são de suma importância. Com o objetivo em realizar uma análise ampla de três técnicas promissoras para aplicação na detecção precoce de ferrugem em folha do soja, esta dissertação apresenta como principal contribuição uma comparação experimental.

Antes de selecionar as técnicas e a cultura a ser utilizada nesta dissertação foi conduzida uma revisão ampla por meio de um estudo de Revisão Sistemática de Literatura (RSL). Tal estudo resultou em um mapa dos possíveis métodos e metodologias empregados na detecção de doenças em plantas, assim como os algoritmos e arquiteturas utilizadas. Após conduzir a RSL, foi possível identificar as técnicas que apresentaram os melhores resultados de acurácia e eficiência, resultando portanto na caracterização de um estudo comparativo. Por fim, a RSL também possibilitou apontarmos as limitações e possíveis lacunas de pesquisa na área. Uma vez que o estudo de RSL explorou diversas culturas nas quais as técnicas de processamento de imagens e aprendizado de máquina eram empregadas, identificou-se um foco em potencial: a detecção precoce de ferrugem em folhas da soja. Identificou-se que apenas dois trabalhos eram empregados para este problema, e que nenhum deles foi planejado para comparar diferentes técnicas. Essa lacuna de pesquisa foi então explorada nesta dissertação de mestrado. Uma metodologia para o experimento controlado foi planejado considerando a comparação de três técnicas. Para tal, foi necessário executar três experimentos isolados, cujo protocolo base é o mesmo. Isso permitiu com que os resultados fossem posteriormente comparáveis de modo à determinar qual das três alternativas é mais interessante para o problema foco de pesquisa.

O primeiro experimento teve como objetivo principal a criação e a testagem das arquiteturas e modelos de RNCs. Os testes e validações foram aplicados utilizando o número original de imagens, sem a utilização de técnicas de aumento de dados. Os resultados obtidos com o uso das imagens originais foram de acurácia igual a 0,8771 e eficiência igual a 0,8571. O segundo experimento foi realizado para testar as arquiteturas e modelos de RNCs criadas e o algoritmo de aumento de dados SLIC. O melhor resultado obtido no segundo experimento com a utilização das imagens que passaram pelo processo de pré-processamento utilizando o algoritmo SLIC foi de acurácia igual a 0,6962 e eficiência igual a 0,7412. Comparando o primeiro e o segundo experimento nota-se que a utilização do algoritmo SLIC para o aumento de dados apresentou resultados não motivadores. Já no terceiro experimento, realizamos a criação das arquiteturas RNCs e utilizamos o algoritmo

augmentor para o aumento de dados. Outra característica do terceiro experimento foi a utilização de divisões de dados diferentes. Analisando e comparando os resultados obtidos no primeiro e terceiro experimento, o melhor resultado obtido no primeiro experimento (modelo 1, na qual também foi utilizado no segundo experimento como parâmetro de comparação) foi de uma acurácia igual a 0,8571 e com uma eficiência igual a 0,8571. Já o modelo que alcançou melhores resultados utilizando as imagens aumentadas (modelo 3 do terceiro experimento) alcançou uma acurácia de 0,9489 e eficiência de 0,8274.

Em comparação aos dois estudos retirados da RSL que também realizaram um estudo para detectar doenças na folha da soja, também mencionados na seção 3.5, a acurácia resultante dos dois estudos são inferiores a acurácia alcançada no modelo 3 do terceiro experimento do nosso estudo. A acurácia do estudo que utilizou MATLAB na detecção das doenças nas folhas da soja foi igual a 85,65%, já o estudo que utilizou RNCs para a detecção de doenças na folha da soja foi de uma acurácia igual a 94,13%. Já o modelo que obteve os melhores resultados em nosso estudo (modelo 3 do terceiro experimento) alcançou uma acurácia igual a 94,89%, resultado quase igual ao estudo S11. Não pode-se afirmar que nossos modelos são melhores ou não que os apresentados nos outros estudos, pois existem uma variedade de fatores que podem alterar os resultados de um estudo para outro, mas a comparação é importante para demonstrar que nossos estudos alcançaram bons resultados quando comparados com outros estudos da área. Outro detalhe importante é o número de imagens utilizada, sendo que o número de imagens em nosso banco de imagens foi bem inferior ao banco de imagens dos dois estudos, isso demonstra que mesmo com o número pequeno de imagens no início do estudo, foi possível utilizar uma técnica ótima de aumento de dados possibilitando um bom resultado final de acurácia e eficiência. E também é importante salientar que os resultados podem variar de um estudo para o outro por conta de muitas variáveis, incluindo o tipo de imagem, se foi adquirida de em um ambiente real ou não, se foi aplicado outros algoritmos de segmentação ou pré-processamento de dados, os métodos e técnicas utilizadas para a classificação das imagens, entre outras variáveis que podem mudar o resultado final.

Em relação aos *gaps* encontrados nos estudos selecionados na RSL, incluem a falta da disponibilidade do banco de dados para a replicação do estudo por outros pesquisadores. Problema na qual não será encontrado em nosso estudo, por conta da disponibilidade do banco de dados dos três experimentos conduzidos em nosso estudo, assim como a disponibilidade das arquiteturas e modelos RNCs criados em nossos experimentos. Outro ponto negativo encontrado na maioria dos estudos foi a falta de comparação de técnicas, na qual abordamos nos três tipos de experimentos, comparando os resultados e técnicas utilizadas pelos três experimentos, tendo mais de uma opção utilizada e abordada em um estudo que tem como fonte inicial o mesmo banco de imagens. Nossos experimentos também foram conduzidos utilizando um protocolo bem definido na qual facilita uma futura replicação do estudo ou aprimoramento do mesmo por outros pesquisadores. A utilização

de imagens em cenário real não controlado não foi conduzido no presente estudo, mas faz parte dos trabalhos futuros na qual pretende-se obter imagens em um cenário real utilizando para a aquisição das imagens veículos aéreos não tripulados. Outro ponto é a integração da academia com a indústria, que também faz parte dos projetos e trabalhos futuros, na qual tentaremos obter parceria com empresas no desenvolvimento das pesquisas futuras.

Com isso, comprova-se que os melhores resultados foram obtidos pelo terceiro experimento que utilizou o algoritmo *augmentor* no enriquecimento dos dados. Comprova-se também que o algoritmo *augmentor* é eficaz no aumento de banco de imagens, quando utilizado nas mesmas circunstâncias do presente estudo. Como trabalho futuro, o objetivo é utilizar imagens diretamente da lavoura (ambiente não controlado) com a aquisição feita por um Veículo Aéreo Não Tripulado (VANT). Com isso, o objetivo será encontrar a melhor técnica de aumento de dados (caso o número de imagens não seja suficiente para os experimentos) e a melhor arquitetura de RNC. Também será investigado os pontos positivos e negativos do estudo, assim como as dificuldades encontradas no estudo envolvendo imagens em um ambiente não controlado, onde os desafios podem ser ainda mais desafiadores. Com o estudo feito analisaremos a acurácia dos modelos assim como a eficiência dos mesmos. Outro objetivo futuro é a integração ou parcerias do meio industrial, com empresas que visam e apoiam o desenvolvimento da agricultura de precisão e a inserção da tecnologia no campo para o auxílio dos produtores agrícolas, principalmente na prevenção de pragas e doenças na lavoura de soja.

REFERÊNCIAS

- ACHANTA, R. et al. Slic superpixels compared to state-of-the-art superpixel methods. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 34, n. 11, p. 2274–2282, 2012. Citado na página 77.
- AL-BAYATI, J. S. H.; ÜSTÜNDAĞ, B. B. Artificial intelligence in smart agriculture: Modified evolutionary optimization approach for plant disease identification. In: IEEE. **2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)**. [S.l.], 2020. p. 1–6. Citado na página 20.
- AMPATZOGLOU, A. et al. Identifying, categorizing and mitigating threats to validity in software engineering secondary studies. **Information and Software Technology**, v. 106, p. 201 – 230, 2019. ISSN 0950-5849. Citado na página 65.
- APARNA, K.; SUPRIYA, P. Precision agriculture in maize fields. In: IEEE. **2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)**. [S.l.], 2018. p. 1407–1410. Citado na página 56.
- ARAÚJO, F. H. et al. Redes neurais convolucionais com tensorflow: Teoria e prática. **SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. III Escola Regional de Informática do Piauí. Livro Anais-Artigos e Minicursos**, Sociedade Brasileira de Computação, v. 1, p. 382–406, 2017. Citado 2 vezes nas páginas 39 e 40.
- AWATE, A. et al. Fruit disease detection using color, texture analysis and ann. In: IEEE. **2015 International Conference on Green Computing and Internet of Things (ICGCIoT)**. [S.l.], 2015. p. 970–975. Citado na página 56.
- BARBEDO, J. G. A. et al. Annotated plant pathology databases for image-based detection and recognition of diseases. **IEEE Latin America Transactions**, IEEE, v. 16, n. 6, p. 1749–1757, 2018. Citado 2 vezes nas páginas 69 e 70.
- BAUER, S. D.; KORČ, F.; FÖRSTNER, W. The potential of automatic methods of classification to identify leaf diseases from multispectral images. **Precision Agriculture**, Springer, v. 12, n. 3, p. 361–377, 2011. Citado na página 56.
- BERNARDI, A. d. C. et al. Agricultura de precisão: resultados de um novo olhar. **Embrapa Instrumentação-Livro técnico (INFOTECA-E)**, Brasília, DF: Embrapa, 2014. 596 p., 2014. Citado 2 vezes nas páginas 19 e 27.
- BHANGE, M.; HINGOLIWALA, H. Smart farming: Pomegranate disease detection using image processing. **Procedia Computer Science**, Elsevier, v. 58, p. 280–288, 2015. Citado na página 56.
- BOULENT, J. et al. Convolutional neural networks for the automatic identification of plant diseases. **Frontiers in plant science**, Frontiers Media SA, v. 10, 2019. Citado na página 19.
- BRAGA, A. d. P. **Redes neurais artificiais: teoria e aplicações**. [S.l.]: Livros Técnicos e Científicos, 2000. Citado 5 vezes nas páginas 28, 29, 30, 31 e 33.
- BRAMLEY, R. Lessons from nearly 20 years of precision agriculture research, development, and adoption as a guide to its appropriate application. **Crop and Pasture Science**, CSIRO, v. 60, n. 3, p. 197–217, 2009. Citado na página 27.

- CALDERÓN, R. et al. Detection of downy mildew of opium poppy using high-resolution multi-spectral and thermal imagery acquired with an unmanned aerial vehicle. **Precision agriculture**, Springer, v. 15, n. 6, p. 639–661, 2014. Citado na página 56.
- CATH, C. et al. Artificial intelligence and the ‘good society’: the us, eu, and uk approach. **Science and engineering ethics**, Springer, v. 24, n. 2, p. 505–528, 2018. Citado na página 28.
- CHEMURA, A.; MUTANGA, O.; DUBE, T. Separability of coffee leaf rust infection levels with machine learning methods at sentinel-2 msi spectral resolutions. **Precision agriculture**, Springer, v. 18, n. 5, p. 859–881, 2017. Citado na página 56.
- COOPER, J.; DOBSON, H. The benefits of pesticides to mankind and the environment. **Crop Protection**, Elsevier, v. 26, n. 9, p. 1337–1348, 2007. Citado na página 19.
- DENG, L.; YU, D. Deep learning: methods and applications. **Foundations and trends in signal processing**, Now Publishers Inc. Hanover, MA, USA, v. 7, n. 3–4, p. 197–387, 2014. Citado na página 38.
- FAO. How to feed the world 2050. the special challenge for sub-saharan africa. In: **High level expert forum**. [S.l.: s.n.], 2009. Citado na página 19.
- FARBER, C. et al. Advanced spectroscopic techniques for plant disease diagnostics. a review. **TRAC Trends in Analytical Chemistry**, Elsevier, 2019. Citado na página 49.
- FARIA, E. L. d. Redes neurais convolucionais e máquinas de aprendizado extremo aplicadas ao mercado financeiro brasileiro. Universidade Federal do Rio de Janeiro, 2018. Citado 7 vezes nas páginas 41, 42, 43, 45, 46, 47 e 48.
- GHOSAL, S. et al. An explainable deep machine vision framework for plant stress phenotyping. **Proceedings of the National Academy of Sciences**, National Acad Sciences, v. 115, n. 18, p. 4613–4618, 2018. Citado 2 vezes nas páginas 19 e 56.
- GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: **Proceedings of the thirteenth international conference on artificial intelligence and statistics**. [S.l.: s.n.], 2010. p. 249–256. Citado na página 44.
- GOODFELLOW, I. et al. **Deep learning**. [S.l.]: MIT press Cambridge, 2016. v. 1. Citado 6 vezes nas páginas 38, 39, 42, 44, 46 e 48.
- GULLI, A.; PAL, S. **Deep learning with Keras**. [S.l.]: Packt Publishing Ltd, 2017. Citado na página 38.
- HAFEMANN, L. G. et al. Transfer learning between texture classification tasks using convolutional neural networks. In: IEEE. **2015 International Joint Conference on Neural Networks (IJCNN)**. [S.l.], 2015. p. 1–7. Citado na página 43.
- HAYKIN, S. **Redes neurais: princípios e prática**. [S.l.]: Bookman Editora, 2007. Citado 3 vezes nas páginas 27, 32 e 33.

HILLNHÜTTER, C. et al. Use of imaging spectroscopy to discriminate symptoms caused by heterodera schachtii and rhizoctonia solani on sugar beet. **Precision Agriculture**, Springer, v. 13, n. 1, p. 17–32, 2012. Citado na página 56.

HINTON, G. E. et al. Improving neural networks by preventing co-adaptation of feature detectors. **arXiv preprint arXiv:1207.0580**, 2012. Citado 2 vezes nas páginas 46 e 47.

HLAING, C. S.; ZAW, S. M. M. Plant diseases recognition for smart farming using model-based statistical features. In: IEEE. **2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)**. [S.l.], 2017. p. 1–4. Citado na página 56.

HOSSAM, M. et al. Plantae: an iot-based predictive platform for precision agriculture. In: IEEE. **2018 International Japan-Africa Conference on Electronics, Communications and Computations (JAC-ECC)**. [S.l.], 2018. p. 87–90. Citado na página 56.

HOU, J.; LI, L.; HE, J. Detection of grapevine leafroll disease based on 11-index imagery and ant colony clustering algorithm. **Precision agriculture**, Springer, v. 17, n. 4, p. 488–505, 2016. Citado na página 56.

IQBAL, Z. et al. An automated detection and classification of citrus plant diseases using image processing techniques: A review. **Computers and electronics in agriculture**, Elsevier, v. 153, p. 12–32, 2018. Citado na página 20.

JARRETT, K. et al. What is the best multi-stage architecture for object recognition? In: IEEE. **2009 IEEE 12th international conference on computer vision**. [S.l.], 2009. p. 2146–2153. Citado na página 44.

JHURIA, M.; KUMAR, A.; BORSE, R. Image processing for smart farming: Detection of disease and fruit grading. In: IEEE. **2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)**. [S.l.], 2013. p. 521–526. Citado na página 56.

JOSHI, A. A.; JADHAV, B. Monitoring and controlling rice diseases using image processing techniques. In: IEEE. **2016 International Conference on Computing, Analytics and Security Trends (CAST)**. [S.l.], 2016. p. 471–476. Citado 2 vezes nas páginas 49 e 56.

KAMILARIS, A.; PRENAFETA-BOLDÚ, F. X. Deep learning in agriculture: A survey. **Computers and electronics in agriculture**, Elsevier, v. 147, p. 70–90, 2018. Citado 2 vezes nas páginas 19 e 38.

KAUR, S.; PANDEY, S.; GOEL, S. Semi-automatic leaf disease detection and classification system for soybean culture. **IET Image Processing**, IET, v. 12, n. 6, p. 1038–1048, 2018. Citado na página 56.

KERKECH, M.; HAFIANE, A.; CANALS, R. Deep leaning approach with colorimetric spaces and vegetation indices for vine diseases detection in uav images. **Computers and Electronics in Agriculture**, Elsevier, v. 155, p. 237–243, 2018. Citado na página 56.

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. 2007. Citado na página 52.

- KNAUER, U. et al. Improved classification accuracy of powdery mildew infection levels of wine grapes by spatial-spectral analysis of hyperspectral images. **Plant methods**, BioMed Central, v. 13, n. 1, p. 47, 2017. Citado na página 56.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2012. p. 1097–1105. Citado na página 39.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Communications of the ACM**, ACM New York, NY, USA, v. 60, n. 6, p. 84–90, 2017. Citado na página 44.
- KUMAR, S. et al. Precision sugarcane monitoring using svm classifier. **Procedia Computer Science**, Elsevier, v. 122, p. 881–887, 2017. Citado na página 56.
- LEE, U. et al. Image analysis for measuring disease symptom to bacterial soft rot in potato. **American Journal of Potato Research**, Springer, p. 1–11, 2019. Citado na página 56.
- LI, H. et al. ‘extended spectral angle mapping (esam)’for citrus greening disease detection using airborne hyperspectral imaging. **Precision Agriculture**, Springer, v. 15, n. 2, p. 162–183, 2014. Citado na página 56.
- LIBO, Z.; TIAN, H.; CHUNYUN, G. Wireless multimedia sensor network for rape disease detections. **EURASIP Journal on Wireless Communications and Networking**, Springer, v. 2019, n. 1, p. 159, 2019. Citado 2 vezes nas páginas 19 e 56.
- LU, J. et al. Using hyperspectral imaging to discriminate yellow leaf curl disease in tomato leaves. **Precision agriculture**, Springer, v. 19, n. 3, p. 379–394, 2018. Citado na página 56.
- LUNA, R. G. de; DADIOS, E. P.; BANDALA, A. A. Automated image capturing system for deep learning-based tomato plant leaf disease detection and recognition. In: IEEE. **TENCON 2018-2018 IEEE Region 10 Conference**. [S.l.], 2018. p. 1414–1419. Citado na página 56.
- MACIEL, V. G. et al. Life cycle inventory for the agricultural stages of soybean production in the state of rio grande do sul, brazil. **Journal of Cleaner Production**, Elsevier, v. 93, p. 65–74, 2015. Citado na página 20.
- MADDIKUNTA, P. K. R. et al. Unmanned aerial vehicles in smart agriculture: Applications, requirements, and challenges. **IEEE Sensors Journal**, IEEE, 2021. Citado na página 20.
- MICHALSKI, R. S.; CARBONELL, J. G.; MITCHELL, T. M. **Machine learning: An artificial intelligence approach**. [S.l.]: Springer Science & Business Media, 2013. Citado na página 29.
- MORIYA, E. A. S. et al. Mapping mosaic virus in sugarcane based on hyperspectral images. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, IEEE, v. 10, n. 2, p. 740–748, 2016. Citado na página 56.
- NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: **ICML**. [S.l.: s.n.], 2010. Citado na página 44.

- NGUGI, L. C.; ABELWAHAB, M.; ABO-ZAHHAD, M. Recent advances in image processing techniques for automated leaf pest and disease recognition—a review. **Information Processing in Agriculture**, Elsevier, 2020. Citado na página 49.
- NIELSEN, M. A. **Neural networks and deep learning**. [S.l.]: Determination press San Francisco, CA, 2015. v. 2018. Citado na página 43.
- NILSSON, N. J. **Introduction to machine learning: An early draft of a proposed textbook**. [S.l.]: USA; Stanford University, 1996. Citado na página 28.
- OERKE, E.-C.; MAHLEIN, A.-K.; STEINER, U. Proximal sensing of plant diseases. In: **Detection and Diagnostics of Plant Pathogens**. [S.l.]: Springer, 2014. p. 55–68. Citado na página 56.
- PANDIT, N. R. **The creation of theory: A recent application of the grounded theory method**. [S.l.]: The qualitative report, 2(4), 1-14., 1996. Citado na página 51.
- PANTAZI, X. E. et al. Detection of biotic and abiotic stresses in crops by using hierarchical self organizing classifiers. **Precision agriculture**, Springer, v. 18, n. 3, p. 383–393, 2017. Citado 2 vezes nas páginas 20 e 56.
- PATRÍCIO, D. I.; RIEDER, R. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. **Computers and electronics in agriculture**, Elsevier, v. 153, p. 69–81, 2018. Citado na página 49.
- PETERSEN, K. et al. Systematic mapping studies in software engineering. In: **Ease**. [S.l.: s.n.], 2008. v. 8, p. 68–77. Citado 3 vezes nas páginas 49, 50 e 51.
- PICON, A. et al. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. **Computers and Electronics in Agriculture**, Elsevier, v. 161, p. 280–290, 2019. Citado na página 56.
- PIERCE, F. J.; NOWAK, P. Aspects of precision agriculture. In: **Advances in agronomy**. [S.l.]: Elsevier, 1999. v. 67, p. 1–85. Citado na página 49.
- POLDER, G. et al. Detection of the tulip breaking virus (tbv) in tulips using optical sensors. **Precision agriculture**, Springer, v. 11, n. 4, p. 397–412, 2010. Citado na página 56.
- PONTI, M.; COSTA, B. Tópicos em gerenciamento de dados e informações. **SBC, Uberlândia**, p. 63–88, 2017. Citado na página 35.
- PONTI, M. A.; COSTA, G. B. P. da. Como funciona o deep learning. **arXiv preprint arXiv:1806.07908**, 2018. Citado 2 vezes nas páginas 39 e 47.
- PRATI, R. C. **Novas abordagens em aprendizado de máquina para a geração de regras, classes desbalanceadas e ordenação de casos**. Tese (Doutorado) — Universidade de São Paulo, 2006. Citado na página 28.
- RICHARDS, J. A.; RICHARDS, J. **Remote sensing digital image analysis**. [S.l.]: Springer, 1999. v. 3. Citado na página 49.
- SAGE, A. P. **Concise Encyclopedia of Information Processing in Systems & [and] Organizations**. [S.l.]: Pergamon press New York, NY, 1990. Citado 2 vezes nas páginas 27 e 28.

- SANTOSO, H. et al. Mapping and identifying basal stem rot disease in oil palms in north sumatra with quickbird imagery. **Precision Agriculture**, Springer, v. 12, n. 2, p. 233–248, 2011. Citado na página 56.
- SAVARY, S. et al. **Crop losses due to diseases and their implications for global food production losses and food security**. [S.l.]: Springer, 2012. Citado na página 20.
- SCHOR, N. et al. Development of a robotic detection system for greenhouse pepper plant diseases. **Precision agriculture**, Springer, v. 18, n. 3, p. 394–409, 2017. Citado na página 56.
- SHIJUAN, L.; YEPING, Z. Three-dimension visualization for primary wheat diseases based on simulation model. In: SPRINGER. **International Conference on Computer and Computing Technologies in Agriculture**. [S.l.], 2009. p. 375–381. Citado na página 56.
- SHULL, F.; SINGER, J.; SJBERG, D. I. K. **Guide to Advanced Empirical Software Engineering**. 1st. ed. [S.l.]: Springer Publishing Company, Incorporated, 2010. ISBN 1849967121. Citado na página 50.
- SILVA, I. d.; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais para engenharia e ciências aplicadas. **São Paulo: Artliber**, v. 23, n. 5, p. 33–111, 2010. Citado 5 vezes nas páginas 33, 34, 36, 37 e 38.
- SINGH, U. P. et al. Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. **IEEE Access**, IEEE, v. 7, p. 43721–43729, 2019. Citado 2 vezes nas páginas 20 e 56.
- SOKOLOVA, M.; JAPKOWICZ, N.; SZPAKOWICZ, S. Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In: SPRINGER. **Australasian joint conference on artificial intelligence**. [S.l.], 2006. p. 1015–1021. Citado na página 32.
- WANG, X. et al. Feature extraction based on dimension reduction and clustering for maize leaf spot images. **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 32, n. 12, p. 1854029, 2018. Citado na página 56.
- WOHLIN, C. et al. **Experimentation in software engineering**. [S.l.]: Springer Science & Business Media, 2012. Citado 5 vezes nas páginas 69, 71, 76, 77 e 85.
- YANG, C. et al. Evaluating unsupervised and supervised image classification methods for mapping cotton root rot. **Precision Agriculture**, Springer, v. 16, n. 2, p. 201–215, 2015. Citado na página 56.
- YUAN, L. et al. Selection of leaf orientation insensitive bands for yellow rust detection. In: SPRINGER. **International Conference on Computer and Computing Technologies in Agriculture**. [S.l.], 2012. p. 78–84. Citado na página 56.
- ZHANG, J. et al. Development, evaluation and application of a spectral knowledge base to detect yellow rust in winter wheat. **Precision agriculture**, Springer, v. 12, n. 5, p. 716–731, 2011. Citado na página 56.

ZHANG, N.; WANG, M.; WANG, N. Precision agriculture—a worldwide overview. **Computers and electronics in agriculture**, Elsevier, v. 36, n. 2-3, p. 113–132, 2002. Citado na página 27.