



**ESTUDO E ANÁLISE DE
REDES NEURAIS CONVOLUCIONAIS PROFUNDAS
NA IDENTIFICAÇÃO DE DOENÇAS EM
PLANTAS POR IMAGENS**

ANDRÉ DA SILVA ABADE

**TESE DE DOUTORADO EM SISTEMAS MECATRÔNICOS
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS MECATRÔNICOS (PPMEC)
DEPARTAMENTO DE ENGENHARIA MECÂNICA (ENM)**

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**ESTUDO E ANÁLISE DE
REDES NEURAIS CONVOLUCIONAIS PROFUNDAS
NA IDENTIFICAÇÃO DE DOENÇAS EM
PLANTAS POR IMAGENS**

ANDRÉ DA SILVA ABADE

Orientador: PROF. DR. FLÁVIO DE BARROS VIDAL, CIC/UNB

TESE DE DOUTORADO EM SISTEMAS MECATRÔNICOS

BRASÍLIA-DF, 04 DE JANEIRO DE 2022.

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**ESTUDO E ANÁLISE DE
REDES NEURAIIS CONVOLUCIONAIS PROFUNDAS
NA IDENTIFICAÇÃO DE DOENÇAS EM
PLANTAS POR IMAGENS**

ANDRÉ DA SILVA ABADE

TESE DE DOUTORADO ACADÊMICO SUBMETIDA AO PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS MECATRÔNICOS DO DEPARTAMENTO DE ENGENHARIA MECÂNICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM SISTEMAS MECATRÔNICOS.

APROVADA POR:

Prof. Dr. Flávio de Barros Vidal, CIC/UnB
Orientador

Prof. Dr. Roberto A. Lotufo, DECAI/FEEC/Unicamp
Membro externo

Prof. Dr. Anderson Soares, INF/UFG
Membro externo

Prof. Dr. Luis Paulo Faina Garcia, CIC/UnB
Membro interno

BRASÍLIA, 04 DE JANEIRO DE 2022.

FICHA CATALOGRÁFICA

ANDRÉ DA SILVA ABADE

Estudo e análise de Redes Neurais Convolucionais Profundas na identificação de doenças em plantas por imagens

2022xv, 166p., 201x297 mm

(PPMEC/FT/UnB, Doutor, Sistemas Mecatrônicos, 2022)

Tese de Doutorado - Universidade de Brasília

Faculdade de Tecnologia - Departamento de Engenharia Mecânica

REFERÊNCIA BIBLIOGRÁFICA

ANDRÉ DA SILVA ABADE (2022) Estudo e análise de Redes Neurais Convolucionais Profundas na identificação de doenças em plantas por imagens. Tese de Doutorado em Sistemas Mecatrônicos, Publicação 2022, Departamento de Engenharia Mecânica, Universidade de Brasília, Brasília, DF, 166p.

CESSÃO DE DIREITOS

AUTOR: André da Silva Abade

TÍTULO: Estudo e análise de Redes Neurais Convolucionais Profundas na identificação de doenças em plantas por imagens.

GRAU: Doutor ANO: 2022

É concedida à Universidade de Brasília permissão para reproduzir cópias desta tese de Doutorado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor se reserva a outros direitos de publicação e nenhuma parte desta tese de Doutorado pode ser reproduzida sem a autorização por escrito do autor.

André da Silva Abade

Rua B, Qd. 03, Nº 16 - Jd. Araguaia - Barra do Garças - MT

Dedico esta pesquisa a Deus, que me suporta e sustenta com graça e misericórdia. À minha amada Mãe, eterna intercessora. À minha esposa e companheira Josilene. À minha irmã Andrea e às minhas sempre crianças Júlia Abade e Gabriela Abade Doneda.

André da Silva Abade

Agradecimentos

Agradeço ao meu Orientador, Flávio de Barros Vidal, pois absolutamente nada disso seria possível se não fosse pela oportunidade, disponibilidade, paciência e confiança investidos em minha pessoa. Obrigado por conciliar sua linha de pesquisa com a proposta deste projeto. Obrigado por ser sensível e humano, externando uma compreensão acalentadora em um momento delicado da minha trajetória.

Agradeço às Instituições provedoras desta realização, ao Instituto Federal de Educação, Ciência e Tecnologia do Estado de Mato Grosso, por viabilizar minha qualificação, e ao Programa de Pós-Graduação em Sistemas Mecatrônicos (PPMEC) do Departamento de Engenharia Mecânica (ENM), Faculdade de Tecnologia (FT), da Universidade de Brasília (UnB).

Agradeço aos Funcionários do Secretaria do PPMEC, em nome do servidor Rafael Lima Meireles de Queiroz. A competência, seriedade e préstimos de vocês são determinantes para qualquer aluno deste Departamento.

Agradeço ao Professor Dr. José Nogueira de Moraes, minha fonte de exemplos, meu poço de sabedoria, meu bastião de caráter e profissionalismo. Como sou orgulhoso de poder tê-lo como amigo, lembrar e ser lembrado e ter eternamente comigo a sua presença replicada por seus ensinamentos, exemplos e boas conversas. À minha sempre e eterna mestra Professora Msc. Rosilene Lopes Rocha Cardozo; ainda que a docência tenha uma semântica complexa, a senhora orientou como interpretar a essência do que é ser Professor. Gostaria que esse agradecimento aos ícones também rememorasse a todos os meus Professores, da Tia Edna da pré-escola a todos que contribuíram com minha formação.

Ao meu amigo, Professor Dr. Marco Donisete de Campos, por sua parceria, companheirismo, por sua preocupação e esforço por mim. Da sua essencial ajuda para importarmos equipamentos para esta pesquisa até a revisão final deste texto, você sempre esteve comigo e não há obrigado que seja suficiente.

Ao Professor Dr. Paulo Afonso Ferreira, um amigo que a pesquisa gentilmente me presenteou. Um presente inestimável, de valor incalculável e fundamental para meu aprendizado e, conseqüentemente, para a realização deste trabalho.

Agradeço também ao meu amigo e hoje Doutor, Lucas Faria Porto, você foi o pontapé desta jornada, foi quem me apresentou, quem motivou, quem disse "sim é possível e você é capaz, vai lá".

Agradeço à minha família, aos que, na intimidade do dia a dia, compartilham comigo as angústias, o esforço, as privações, as ausências, as orações, a felicidade e as conquistas.

À minha irmã, Andrea C. Abade Doneda; somos duas partes de um todo geograficamente distantes, mas amorosamente partes um do outro. À minha amada sobrinha, Gabriela Abade Doneda, você tem o dom de me encantar! Sua inteligência, seu humor, seu jeito imprevisível de ser.

Mãe, não há como agradecer a Senhora! Não há espaço aqui, tampouco há papel suficiente no mundo que possa conter sinônimos de agradecimentos que qualifiquem a minha dívida eterna de gratidão, Dona Maria da Guia Abade. Das vigílias constantes às orações incessantes, minha Mãe de joelhos por seu filho de pé.

Agradeço à minha amada e desejada filha Júlia A. Abade, te desejo vida, longa vida! Eu te desejo muito mais que mil amigos, te desejo a poesia que todo poeta esperou, coração de menina cheio de esperança. Eu te desejo voz de pai amigo e olhar de avô.

Agradeço especialmente à minha esposa Josilene Dália Alves, começamos o doutorado juntos e separados batalhamos nossas conquistas, compartilhamos cada momento, cada lágrima, cada sorriso. Enfrentamos a tempestade, experimentamos o milagre e a providência da graça de Deus. Tenho que confessar, como sou vaidoso por ter você ao meu lado, por ter seus cuidados e seu amor.

Por fim, tudo posso naquele que me fortalece! A paz de Deus, que excede todo o entendimento, guardou meu coração e a minha mente em abrigo seguro. Toda honra e toda glória a Deus, que é digno de louvor.

Resumo

Rede Neurais Convolucionais (CNNs), demonstram um potencial para tarefas relacionadas à Visão Computacional. A característica de maior destaque das CNNs é sua capacidade de explorar a correlação espacial ou temporal nos dados. Assim, várias melhorias na metodologia e arquitetura de aprendizagem das redes foram realizadas para tornar as CNNs escaláveis para problemas grandes, heterogêneos, complexos e multiclassificados. A agricultura delimita um escopo de problemas desafiadores, que carecem de tecnologias para proporcionar maior incremento na produção agrícola, principalmente em relação ao enfrentamento de doenças. As doenças de plantas são consideradas um dos principais fatores que influenciam a produção de alimentos, e a sua identificação é primordialmente realizada por técnicas manuais ou por microscopia, o que aumenta o tempo de diagnóstico e as possibilidades de erro. Soluções automatizadas de identificação de doenças de plantas, usando imagens e aprendizado de máquina, em especial as CNNs, têm proporcionado avanços significativos. Entretanto, a maioria das abordagens possui baixa capacidade de classificação, tendo como agravante as infestações simultâneas por diferentes patógenos e as confusões sintomáticas causadas por fatores abióticos. Assim, o objetivo deste trabalho é analisar e avaliar as arquiteturas CNNs, explorando potencialidades e prospectando novos arranjos de arquitetura para classificar doenças de plantas e identificar patógenos. A abordagem fez uso de uma estratégia de customização, na qual redes operativas independentes ou blocos convolucionais são integradas em um único modelo para capturar um conjunto mais variado de características. A *NEMANet* é um resultado relevante desta abordagem de customização de CNNs para classificação de fitonematoides em imagens microscópicas. O modelo alcançou a melhor taxa de acurácia atingindo 99,35%, possibilitando melhorias gerais de precisão superiores a 6,83% e 4,1%, para treinamento com inicialização dos pesos e para transferência de aprendizagem, em comparação com outras arquiteturas avaliadas. Os resultados demonstraram que a customização de arquiteturas CNNs é uma abordagem promissora para o aumento de ganhos em termo de acurácia, convergência das redes e tamanho dos modelos.

Palavras-Chaves: *Doenças de Plantas, Nematoides, CNNs, Imagens Microscópicas.*

Abstract

Convolutional Neural Networks (CNNs) demonstrate a potential for computer vision tasks. The most prominent feature of CNNs is their ability to explore spatial or temporal correlation in the data. Thus, several improvements in the methodology and architecture of learning of the networks were made to make the CNNs scalable for large, heterogeneous, complex, and multi-class problems. Agriculture delimits a scope of challenging problems, which lack technologies to increase agricultural production, especially about coping with diseases. Plant diseases are considered one of the main factors that influence food production, and their identification is primarily performed by manual techniques or microscopy, which increases the time of diagnosis and the possibility of errors. Using imaging and machine learning, especially CNNs, automated plant disease identification solutions have provided significant advances. However, most approaches have low classification capacity, with simultaneous infestations by different pathogens and symptomatic confusion caused by abiotic factors as an aggravating factor. Thus, this work aims to analyze and evaluate CNN architectures, exploring potentialities and prospecting new architectural arrangements to classify plant diseases and identify pathogens. The approach used a customization strategy, in which independent operative networks or convolutional blocks are integrated into a single model to capture a more varied set of characteristics. The *NEMANet* is a relevant result of this CNN customization approach for the classification of phytonematodes in microscopic images. The model achieved the best accuracy rate reaching 99.35%, enabling overall accuracy improvements greater than 6.83% and 4.1%, for weight initialization training and learning transfer, compared to other evaluated architectures. The results showed that the customization of CNN architectures is a promising approach to increase gains in terms of accuracy, the convergence of networks, and the size of the models.

Keywords: *Plant Diseases, Nematodes, CNN, Microscopic Images.*

Sumário

RESUMO	IV
ABSTRACT	V
LISTA DE SÍMBOLOS	XIII
1 INTRODUÇÃO	1
1.1 CONTEXTUALIZAÇÃO E MOTIVAÇÃO	2
1.2 HIPÓTESES	4
1.3 OBJETIVOS	6
1.3.1 OBJETIVOS ESPECÍFICOS	6
1.4 CONTRIBUIÇÕES	6
1.5 ORGANIZAÇÃO DO DOCUMENTO	7
2 FUNDAMENTOS TEÓRICOS	8
2.1 CONCEITO DE DOENÇA DE PLANTAS	8
2.2 TIPOS DE DOENÇAS DE PLANTAS	9
2.3 A IDENTIFICAÇÃO DE NEMATOIDES PARASITOS DE PLANTAS	10
2.4 ANÁLISE DE IMAGENS MICROSCÓPICAS	12
2.5 CONCEITO DE APRENDIZAGEM DE MÁQUINA	14
2.5.1 APRENDIZADO SUPERVISIONADO	15
2.5.2 PERCEPTRON	16
2.5.3 MULTI-LAYER PERCEPTRON	17
2.6 REDES NEURAIS CONVOLUCIONAIS	19
2.6.1 CONVOLUÇÃO	21
2.6.2 NÃO-LINEARIDADE	21
2.6.3 REGULARIZAÇÃO	23
2.6.4 GRADIENTE DESCENDENTE ESTOCÁSTICO	25
2.7 EVOLUÇÃO DAS ARQUITETURAS	26
2.8 ARQUITETURAS COM TOPOLOGIA DE MÚLTIPLOS CAMINHOS	30
2.9 TRANSFERÊNCIA DE APRENDIZAGEM	32
2.10 AJUSTE FINO E HIPERPARÂMETROS	34
2.11 AUMENTO DE DADOS	36

2.12	CONSIDERAÇÕES FINAIS	38
3	METODOLOGIA	39
3.1	DESCRIÇÃO INICIAL DO PROBLEMA ABORDADO	39
3.2	ESTRUTURA DA METODOLOGIA PROPOSTA	40
3.3	ESTRATÉGIAS DE CONCEPÇÃO, AVALIAÇÃO E VALIDAÇÃO DAS ARQUITETURAS CNNs CUSTOMIZADAS	41
3.3.1	ARQUITETURA CNN CUSTOMIZADA PARA CLASSIFICAÇÃO DE DOENÇAS DE PLANTAS	41
3.3.2	ARQUITETURA CNN CUSTOMIZADA PARA IDENTIFICAÇÃO DE PATÓGENOS COM IMAGENS MICROSCÓPICAS	43
3.4	FASES DA PESQUISA	46
3.5	CONSIDERAÇÕES FINAIS	50
4	REVISÃO SISTEMÁTICA SOBRE O RECONHECIMENTO DE DOENÇAS DE PLANTAS USANDO REDES NEURAS CONVOLUCIONAIS	51
4.1	INTRODUÇÃO	51
4.2	PLANEJAMENTO DA REVISÃO	53
4.2.1	CONDUÇÃO DA REVISÃO	56
4.2.2	RESULTADOS E EVIDÊNCIAS DA RSL	56
4.3	ANÁLISE E DISCUSSÃO	66
4.4	CONSIDERAÇÕES FINAIS	79
5	RECONHECIMENTO DE DOENÇAS DE PLANTAS A PARTIR DE IMAGENS DIGITAIS USANDO REDES NEURAS CONVOLUCIONAIS MULTICANAIS	81
5.1	MÉTODO PROPOSTO	82
5.2	RESULTADOS	84
5.3	DISCUSSÃO	84
5.4	CONSIDERAÇÕES FINAIS	86
6	<i>NEMANet</i>: UM MODELO DE REDE NEURAL CONVOLUCIONAL PARA IDENTIFICAÇÃO DE FITONEMATOIDES EM IMAGENS MICROSCÓPICAS	87
6.1	MÉTODO PROPOSTO	89
6.1.1	CONJUNTO DE IMAGENS	89
6.1.2	PRE-PROCESSAMENTO DO NEMADATASET	91
6.1.3	ARQUITETURAS CNNs AVALIADAS	92
6.1.4	MODELO PROPOSTO <i>NEMANet</i>	93
6.1.5	TREINAMENTO E VALIDAÇÃO	93
6.2	RESULTADOS	97
6.3	DISCUSSÃO	98
6.4	CONSIDERAÇÕES FINAIS	104
7	CONCLUSÕES E TRABALHOS FUTUROS	105

7.1	VALIDAÇÃO DAS HIPÓTESES	106
7.2	TRABALHOS FUTUROS	107
REFERÊNCIAS BIBLIOGRÁFICAS		109
A TABELAS DE SUMARIZAÇÃO DE RESULTADOS		133
A.1	CAPÍTULO 04 - REVISÃO SISTEMÁTICA	133
A.2	CAPÍTULO 06 - <i>NEMANet</i>	154
B ARTIGOS PUBLICADOS		157
B.1	ARTIGOS PUBLICADOS EM PERIÓDICOS INTERNACIONAIS	157
B.2	ARTIGOS PUBLICADOS EM CONFERÊNCIAS INTERNACIONAIS	157
B.3	PRIMEIRA PÁGINA DOS ARTIGOS EM ORDEM DE PUBLICAÇÃO	158

Lista de Figuras

1.1	Ranking dos 10 maiores países exportadores de alimentos do mundo.....	3
2.1	Ilustração esquemática da classificação dos tipos de doenças em plantas, agrupadas em duas categorias principais: biótica e abiótica. Adaptada de (GHOSAL <i>et al.</i> , 2018)	9
2.2	Morfologia geral de macho e fêmea de Fitonematoides (A,B) Fonte: (FERRAZ;BROWN, 2016)	11
2.3	Exemplo de uma imagem gerada por microscopia de luz com espécies de fitonematoides.	11
2.4	Exemplos de imagens geradas por microscopia de luz com a composição celular da planta até os patógenos causadores de doenças. Fonte: (EMBRAPA, 2021) (UFV, 2016) (ABADE <i>et al.</i> , 2021b)	13
2.5	Arquitetura MLP composta por uma camada de entrada, duas camadas ocultas e uma camada de saída. Fonte:Aggarwal (2014)	18
2.6	Arquitetura CNN. Adaptada de Lecun <i>et al.</i> (1998)	20
2.7	Ilustração esquemática de uma operação de convolução 2D sem inversão do <i>kernel</i> (3x3) para calcular o valor de saída no pixel (2,2). Adaptada de Kiswani e Verner (2014)	22
2.8	Exemplo de operações com a função MaxPooling. Adaptada de Zajačko <i>et al.</i> (2018)	24
2.9	Estratégias de Regularização: a) Sem Dropout; b) Dropout e c)DropConnect. Adaptado de Gu <i>et al.</i> (2018).....	25
2.10	Comparativo entre os Algoritmos de otimização. Adaptado de Goodfellow <i>et al.</i> (2016)	26
2.11	Taxonomia evolutiva das arquiteturas CNNs com sete categorias diferentes. Adaptado de Khan <i>et al.</i> (2020).....	27
2.12	Arquitetura LeNet-5. Adaptado de Karim (2019)	27
2.13	Arquitetura AlexNet. - Adaptado de Karim (2019)	27
2.14	Arquitetura VGGNet. Adaptado de Karim (2019)	28
2.15	Arquitetura GoogLeNet - InceptionV1. Esta CNN possui duas redes auxiliares com classificadores que são descartadas no momento da inferência. Adaptado de Karim (2019)	29
2.16	Arquitetura ResNet. Adaptado de Karim (2019)	30

2.17	Arquitetura CNNs com topologia de múltiplos caminhos. O diagrama A) representa uma topologia paralela CNN de múltiplos caminhos: diferentes CNNs executam paralelamente a extração de características, concatenando esse resultado para prever o rótulo. Já o diagrama B) corresponde a uma topologia em cascata (série), na qual uma rede é fornecida como entrada para outra tendo como pressuposto a lapidação ou refinamento do resultado.....	32
2.18	Transferência de Aprendizagem. Adaptado de Pan e Yang (2010).....	33
3.1	Fluxograma da Estrutura da Pesquisa.....	40
3.2	Processo de Customização da Arquitetura CNN com Junção das Categorias propostas por KHAN <i>et al.</i> (2020) para Classificação de Doenças de Plantas.....	42
3.3	Algoritmo da Pesquisa de Grade para definição do conjunto de hiperparâmetros. Adaptado de (SNOEK <i>et al.</i> , 2012a).....	43
3.4	a)Bloco Inception básico; b) Bloco Inception implementado na GoogLeNet (SZEGEDY <i>et al.</i> , 2015a); c) Bloco Inception aprimorado na InceptionV3 (SZEGEDY <i>et al.</i> , 2015b); d) Bloco Inception-ResNet (SZEGEDY <i>et al.</i> , 2017a).....	43
3.5	Processo de Customização da Arquitetura CNN com Junção das Categorias propostas por KHAN <i>et al.</i> (2020) para Identificação de Patógenos com Imagens Microscópicas.	44
3.6	Algoritmo Taxa de Aprendizagem Cíclica para Ajuste da Taxa de Aprendizagem usando estratégia Triangulo. Adaptado de (SMITH, 2017).....	45
3.7	Ilustração da Curva ROC-AUC - Adaptada de Prates (2020).	47
3.8	Descrição do processo replicável de Pesquisa.	49
4.1	Fases e Atividades do Processo de Revisão Sistemática da Literatura. Fonte:Nakagawa <i>et al.</i> (2017)	52
4.2	Crerios de Inclusão e Exclusão da RSL.	55
4.3	Fluxograma das fases e seleções dos estudos da RSL.	57
4.4	Gráfico com a evolução temporal da quantidade de estudos publicados por ano ...	58
4.5	Distribuição temporal dos estudos por Base de Dados.	58
4.6	(A) Distribuição geográfica dos estudos avaliados para elegibilidade e (B) Distribuição geográfica dos estudos inclusos na seleção final.....	59
4.7	Número de estudos agrupados pela abordagem predominante.	60
4.8	Número de estudos agrupados por Conjunto de Dados	61
4.9	Número de estudos agrupados pelo tipo de sensor e perspectiva de aquisição das imagens.	62
4.10	Número de estudos agrupados pelo tipo de cultura.....	63
4.11	Número de estudos agrupados por Frameworks.	64
4.12	Número de estudos agrupados por Algoritmos.....	65
4.13	Número de estudos agrupados por tarefa principal realizada pela CNN.	66

4.14	Relação do tipo de cultura X grupos de patógenos X ocorrências de doenças investigadas por agentes causadores.	67
5.1	Exemplos do Conjunto de Dados PlantVillage com imagens de Doenças de Plantas em três versões (Colorida, Escala de Cinza e Segmentada): (A) Sarna da macieira. (B) Uma amostra da podridão negra nas folhas da uva. (C) Queima das folhas do morango	82
5.2	Uma ilustração da arquitetura genérica de uma rede convolucional multicanal. O modelo generaliza uma estrutura com dois canais de entrada e identifica os segmentos relevantes da arquitetura.....	83
6.1	O fluxo de trabalho do método para a identificação de nematoides da cultura da soja usando a Rede Neural Convolucional NEMANet.	89
6.2	Ilustração do protocolo utilizado no método de extração de nematoides.	90
6.3	Exemplos de imagens no estado original de fitonematoides no NemaDataset: (A) <i>Helicotylenchus dihystera</i> (B) <i>Heterodera glycines</i> (J2). (C) <i>Meloydogine incognita</i> (J2). (D) <i>Pratylenchus brachyurus</i> . (E) <i>Rotylenchulus reniformis</i>	91
6.4	Visão geral das etapas de pré-processamento - NemaDataset.	92
6.5	(Esquerda) Arquitetura DenseNet121. (Direita) Dense block, Conv block and Transition block. Adaptado (HUANG <i>et al.</i> , 2017).....	94
6.6	Blocos Inception implementados na arquitetura InceptionV3.	95
6.7	Arquitetura NemaNet customizada com a estrutura de blocos DenseNet121 e Inception.....	96
6.8	Curvas de perda e precisão do processo de treinamento FS sem aumento de dados.	99
6.9	Curvas de perda e precisão do processo de treinamento FS com aumento de dados	99
6.10	Curvas de perda e precisão do processo de treinamento TL sem aumento de dados.	100
6.11	Curvas de perda e precisão do processo de treinamento TL com aumento de dados.	100
6.12	Matrix de Confusão Normalizada - Inicialização de Pesos: (A) NemaNet; (B) InceptionV3; (C) Xception; (D) InceptinResNetV2 e (E) DenseNet169	101
6.13	Matrix de Confusão Normalizada - Transferência de Aprendizagem: (A) NemaNet; (B) DenseNet201; (C) DenseNet169; (D) DenseNet121 e (E) InceptionV3	101
6.14	Curva Característica de Operação do Receptor(ROC) Multiclasse com Inicialização dos Pesos: (A) NemaNet; (B) DenseNet201; (C) DenseNet169; (D) DenseNet121 e (E) InceptionResNetV2	102
6.15	Curva Característica de Operação do Receptor(ROC) Multiclasse com Transferência de Aprendizagem: (A) NemaNet; (B) DenseNet201; (C) DenseNet169; (D) DenseNet121 e (E) InceptionV3	102

Lista de Tabelas

2.1	Ações e Reações do Ajuste dos Hiperparâmetros quanto a Capacidade do Modelo.	35
3.1	Métricas usadas para avaliar o desempenho das arquiteturas CNNs investigadas.	45
4.1	Questões de qualidade para o computo do Índice de Qualidade dos Estudos.....	55
5.1	Hiperparâmetros de Treinamento.....	84
5.2	Comparação das Pontuações $F_1 score$ das arquiteturas investigadas.....	85
6.1	Especificações técnicas do microscópio e da câmera	90
6.2	Composição do conjunto de dados de imagens com as espécies de fitonematoides.	91
6.3	Arquiteturas Populares Avaliadas	92
6.4	Camadas e parâmetros de aumento de dados de imagem.	96
6.5	Sumarização dos Resultados das Métricas por Modelo. Os cinco principais modelos foram destacados de acordo com o tipo de treinamento (FS/TL), com ou sem aumento de dados.	98
A.1	Características dos Estudos Primários da Seleção Final.	133
A.2	Características dos Estudos Primários da Seleção Final.	144
A.3	Métricas para avaliação dos diferentes modelos implementados neste estudo.....	154
A.3	Métricas para avaliação dos diferentes modelos implementados neste estudo.....	155
A.3	Métricas para avaliação dos diferentes modelos implementados neste estudo.....	156

Lista de Símbolos

Letras latinas

A	Espaço de entrada
B	Espaço de saída
C	Número de classes
D	Conjunto de elementos de treinamento
F	Espaço de Funções
I	Entrada bidimensional
K	Kernel da Convolução
T	Conjunto de itens
W	Matriz de Pesos
R	Redução de peso
S	Função de Convolução
c	classe
f	função
h	função de pooling
\mathbf{w}	Vetor de d dimensões
r	Vetor binário de d dimensões
x_i	Vetor de características do i – <i>ésimo</i> elemento
y_i	Rótulo da classe
s_0	Estados iniciais
u_i	Vetor real tridimensional

v_i	Pontos rotulados
w	Pesos
t_i	Saídas desejadas
z_{lk}^{uk}	Valor de entrada no mapa de características

Letras Gregas

α	Inclinação da função linear negativa
θ	Termo de regularização
η	Taxa de aprendizagem
γ	Fator de desconto
λ	Força de regularização adicionada
ϕ	Parâmetros da função de interesse
π	Mapeamento de estados para uma distribuição de probabilidades
ρ	Conjunto finito de parâmetros

Outros símbolos

\mathcal{A}	Conjunto de ações
\mathbb{R}	Números reais
\mathbb{N}	Números naturais
\mathcal{I}	Conjunto de atributos binários
\mathcal{L}	Função de perda da regularização
\mathcal{G}	Função de interesse do SGD
\mathcal{X}	Conjunto de alguns itens em \mathcal{I}
\mathcal{E}_p	Critério de perceptron

Lista de Acrônimos

CCF	Comprehensive Color Feature
CLR	Cyclical Learning Rates
CNN	Convolutional Neural Network
DNN	Deep Neural Network
GAN	Generative Adversarial Networks
GPDCNN	Global Pooling Dilated Convolutional Neural Network
LBPH	Local Binary Patterns Histograms
LIDAR	Light Detection and Ranging
ML	Machine Learning
MLP	Multi-Layer Perceptron
MP	Morphological Processing
NN	Neural Network
PRISMA	Preferred Reporting Items for Systematic and Meta-Analyses
RSL	Revisão Sistemática da Literatura
SGD	Stochastic Gradient Descent
SLP	Single Layer Perceptron
SSD	Single Shot Multibox Detector
SVM	Support Vector Machines
VANT	Veículo Aéreo Não Tripulado
VGG	Visual Geometry Group

Capítulo 1

Introdução

Os algoritmos de aprendizagem de máquina (*Machine Learning*) são conhecidos por aprenderem relações subjacentes nos dados, e assim apoiarem processos de tomada de decisão ou até mesmo executarem determinadas decisões sem a exigência de instruções explícitas (MURPHY, 2012; SUTHAHARAN, 2015). Em 1989, uma nova classe de Redes Neurais (*Neural Network* – NN), denominada Rede Neural Convolucional (*Convolutional Neural Network* – CNN) foi apresentada por LeCun *et al.* (1989), demonstrando um enorme potencial para tarefas relacionadas à Visão Computacional.

As CNNs são um dos melhores algoritmos de aprendizagem para a compreensão de imagens e têm demonstrado desempenho exemplar em tarefas relacionadas à segmentação, classificação, detecção e recuperação de imagens (GU *et al.*, 2018). Uma das características de maior destaque das CNNs é sua capacidade de explorar a correlação espacial ou temporal nos dados. A topologia CNN é dividida em vários estágios de aprendizagem compostos por uma combinação de camadas convolucionais, unidades de processamento não lineares e camadas de subamostragem (KHAN *et al.*, 2020).

Do final da década de 1990 até 2021, várias melhorias na metodologia e arquitetura de aprendizagem das redes foram realizadas para tornar as CNNs escaláveis para problemas grandes, heterogêneos, complexos e multiclases (KHAN *et al.*, 2020). As inovações propostas incluem diferentes aspectos, tais como modificação de unidades de processamento, estratégias de otimização de parâmetros e hiperparâmetros, padrões de “design” e conectividade de camadas, etc (GU *et al.*, 2018; SUN *et al.*, 2020; ALZUBAIDI *et al.*, 2021).

Neste contexto, a agricultura delimita um escopo de problemas desafiadores, que carecem de tecnologias para proporcionar maior incremento na produção agrícola, otimizando produtividade e reduzindo custos. As CNNs surgem como uma dessas tecnologias, com a possibilidade de promover soluções automatizadas de diagnósticos nas mais diversas aplicações agrícolas.

A identificação de doenças de plantas é uma das mais básicas e mais importantes atividades da agricultura (AGRIOS, 2005). Na maioria dos casos, a identificação é realizada manualmente por inspeção visual ou por microscopia (BOS; PARLEVLIET, 1995; AGRIOS, 2005).

O problema da avaliação visual para identificação de doenças é que o avaliador assume uma tarefa subjetiva, propensa a fenômenos psicológicos e cognitivos que podem levar a preconceitos, ilusões de ótica e, finalmente, ao erro (BARBEDO, 2016). As soluções automatizadas de identificação de doenças de plantas, usando imagens e aprendizado de máquina, em especial as redes neurais convolucionais, têm proporcionado significativos avanços, intuindo maximizar a acurácia de acerto nos diagnósticos (KAUR *et al.*, 2019; THYAGHARAJAN; RAJI, 2019).

Por outro lado, análises laboratoriais, como abordagens baseadas em técnicas moleculares e imunológicas de patógenos, geralmente consomem tempo, deixando de fornecer respostas em tempo hábil (KAUR *et al.*, 2019). Nesse contexto, é convincente o desenvolvimento de métodos automáticos capazes de identificar doenças de maneira rápida e confiável. Os métodos automáticos, como as técnicas baseadas em segmentação (PANG *et al.*, 2011), extração de características (ELNEMR *et al.*, 2016), seleção de características (U.R.; PAUL, 2016) e as técnicas baseadas em classificadores (RUMPF *et al.*, 2010; ABADE *et al.*, 2019) são propostas que dependem de imagens digitais, possibilitando o uso de técnicas de aprendizagem de máquina capazes de mitigar o tempo e fornecer agilidade nos diagnósticos.

Embora os avanços sejam significativos, conforme apresentado anteriormente, quanto à utilização das redes neurais convolucionais nos problemas de classificação de doenças de plantas, a maioria das abordagens possui baixa capacidade discricionária, ou seja, conseguem classificar um número pequeno de doenças limitado a um único tipo de cultura (LIN *et al.*, 2019a; DAWEI *et al.*, 2019). Além disso, vários patógenos podem infectar simultaneamente uma planta e causar sintomas que fogem aos padrões obtidos por uma infestação única (BOS; PARLEVLIE, 1995). É preciso destacar ainda as possíveis confusões sintomáticas causadas por fatores abióticos que imitam muito de perto as características de algumas doenças (AGRIOS, 2005).

Outro fator destacado por Barbedo (2018) é que ainda que o primeiro diagnóstico da doença seja por amostra visual da planta, é necessária uma análise laboratorial para confirmar qual patógeno é o verdadeiro causador da doença. Nesta perspectiva, esta pesquisa analisa e avalia as arquiteturas CNNs, explorando potencialidades e prospectando novos arranjos arquitetônicos para classificar doenças de plantas e identificar patógenos.

1.1 Contextualização e Motivação

A Organização das Nações Unidas para a Alimentação e a Agricultura (*Food and Agriculture Organization of the United Nations - FAO*), estima que a população mundial, ao final da década de 2050, pode variar entre 9,3 e 10,6 bilhões de pessoas, 29% maior do que a população atualmente. Conseqüentemente, a produção de alimentos no mundo deverá aumentar em pelos menos 70% para que todos possam ter acesso aos alimentos (FAO, 2021).

Nesta perspectiva, a expansão da produção agrícola deveria superar desafios importantes como, por exemplo, aumentar a produção, preservando florestas e recursos hídricos. Depois da mecanização agrícola, estima-se que as tecnologias da informação e a digitalização de processos

e reajustes de conexões para projetos arquitetônicos e aprimoramento das redes (KHAN *et al.*, 2020). Essa mudança resultou em muitas novas ideias arquitetônicas, como reforço de canal, exploração espacial e inteligente do mapa de características e processamento de informações com base em atenção, etc (WANG *et al.*, 2017a; WOO *et al.*, 2018).

A maioria das arquiteturas de detecção e segmentação de objetos com destaque na literatura, como *Single Shot Multibox Detector* (SSD), *Region-based CNN* (R-CNN), *Faster R-CNN*, *Mask R-CNN* e *Fully Convolutional Neural Network* (FCN) são construídas usando como base os projetos arquitetônicos da ResNet, VGG, Inception, etc. Muitos outros algoritmos de detecção interessantes, como *Feature Pyramid Networks*, *Cascade R-CNN*, *Libra R-CNN*, etc, modificaram evolutivamente essas arquiteturas para melhorar o seu desempenho (LIN *et al.*, 2017; PANG *et al.*, 2019; CAI; VASCONCELOS, 2021).

Da mesma forma, foi observado que o empilhamento de múltiplas transformações, não apenas em profundidade, mas também em paralelo, demonstrou ganhos significativos quanto ao incremento da aprendizagem para problemas complexos (KARPATHY *et al.*, 2014; ZAGORUYKO; KOMODAKIS, 2016; HAN *et al.*, 2017). Diferentes pesquisadores usaram um híbrido das arquiteturas já propostas para melhorar o desempenho de CNNs baseadas na exploração espacial ou de profundidade (HUANG *et al.*, 2016; TARG *et al.*, 2017; KUEN *et al.*, 2017; LV *et al.*, 2019; ABADE *et al.*, 2019).

Entretanto, algumas preocupações são latentes quanto à reestruturação arquitetônica das CNNs. Questões como o alto custo computacional, requisitos de memória e tempo de resposta são sempre desafiadores para implantação de modelos abrangentes de última geração em ambientes com recursos limitados de processamento (SHAKEEL *et al.*, 2019). Na agricultura, muitos aplicativos executam tarefas que necessitam ser realizadas sob plataformas limitadas computacionalmente em tempo hábil (ALZUBAIDI *et al.*, 2021). Deste modo, customizar essas CNNs e torná-las apropriadas a estes ambientes com recursos limitados é um constante desafio.

Assim, apesar de as arquiteturas e modelos tradicionais de redes convolucionais já terem demonstrado seu efetivo desempenho, este estudo prospecta elaborar avaliações, otimizações e customizações em arquiteturas CNNs, com intuito de estabelecer um modelo passível de ser aplicado em conjuntos de imagens com baixo número de amostras rotuladas e adaptável à diversidade de problemas e especificidades dos cenários do mundo real.

1.2 Hipóteses

Hipóteses são suposições colocadas como respostas plausíveis e provisórias para o problema de pesquisa. Formulado o problema, é proposta uma resposta suposta, provável e provisória, plausível como solução (LAKATOS *et al.*, 2007).

Hipótese 1 (H1): *A primeira hipótese proposta neste trabalho é quanto à viabilidade de novos*

arranjos arquitetônicos, compostos pela adição de blocos convolucionais, que conciliem os conceitos de exploração espacial, profundidade, multiconexão com base em largura e múltiplos canais de entrada para compor uma solução apropriada, capaz de amenizar a degradação de desempenho, os problemas de overfitting e underfitting, aumentando a acurácia e diminuindo o tamanho dos modelos.

Hipótese 2 (H2): *A segunda hipótese é formulada sobre a possibilidade de ganhos significativos entre a concepção de uma arquitetura customizada de acordo com o contexto das imagens microscópicas e o desempenho das arquiteturas tradicionais da literatura. Esse novo arranjo arquitetônico customizado é capaz de superar as arquiteturas tradicionais quanto a acurácia, a capacidade de convergência e o número de parâmetros?*

Hipótese 3 (H3): *Já a terceira hipótese é formulada sobre a possibilidade de superação do modelo proposto sobre a técnica de transferência de aprendizagem, comumente utilizadas nas arquiteturas tradicionais. Os modelos customizados e adequados à realidade do conjunto de imagens conseguem alcançar melhores resultados sem a utilização das técnicas de transferência de aprendizagem e aumento de dados?*

As abordagens dos problemas sobre a identificação de doenças de plantas usando redes neurais convolucionais, em sua grande maioria, utilizam arquiteturas tradicionais com técnicas de transferência de aprendizagem e ajuste fino (BARBEDO, 2016; JIN *et al.*, 2018; THYAGHARAJAN; RAJI, 2019; KAUR *et al.*, 2019; ABADE *et al.*, 2019). Este fato reforça a necessidade de validação da primeira hipótese proposta.

Por exemplo, na abordagem proposta por Singh *et al.* (2019), uma arquitetura com fluxos distintos de processamento é apresentada, entretanto, os autores não customizam os modelos. Esta abordagem usufrui apenas do enriquecimento do mapa de características fornecido por meio do processamento paralelo de duas estruturas AlexNet e do emprego das técnicas de transferência de aprendizagem. Não há ajustes no arranjo das camadas e adaptação dos filtros e regularização, tampouco há uma adequação dos hiperparâmetros para um modelo com este tipo de topologia.

Outro problema reportado nos estudos de Barbedo (2016) e Zhang *et al.* (2019a) é quanto à dificuldade de distinção de sintomas bióticos e abióticos. Algumas doenças de plantas causadas por patógenos geram padrões nas imagens similares aos efeitos de fitotoxicidade. Deste modo, a confirmação do diagnóstico seria por meio da identificação do patógeno. Porém, esses patógenos são micro-organismos como, por exemplo, vírus, bactérias, fungos, algas e nematoides visualizados por microscopia. As imagens microscópicas possuem um retrospecto de adversidade quanto ao processamento de imagens, sendo recorrentes os relatos sobre ruídos e distorções (ROEDER, 2019). Adicionalmente, de acordo com Abade *et al.* (2021a) na literatura são escassos estudos que abordem a concepção de modelos CNNs para identificação de patógenos de plantas usando imagens microscópicas.

1.3 Objetivos

Com objetivo de aplicar e difundir os conhecimentos e recursos tecnológicos pautados em Visão Computacional e Aprendizagem de Máquina, como um processo viável e consoante com as boas práticas de manejo das culturas no processo de classificação de doenças de plantas e identificação de patógenos, este estudo prospecta explorar as redes neurais convolucionais compondo novos arranjos arquitetônicos customizados adaptáveis à diversidade de problemas às especificidades do mundo real. Neste contexto, serão efetuadas proposições de otimizações, customizações e adição de blocos convolucionais nos modelos de arquiteturas tradicionais já utilizados na solução de problemas.

1.3.1 Objetivos Específicos

- Conhecer os desafios e tendências do processo de classificação de doenças de plantas e identificação de patógenos utilizando as redes neurais convolucionais;
- Analisar e avaliar as diferentes técnicas e métodos de identificação automatizados de doenças de plantas utilizando as redes neurais convolucionais;
- Customizar, otimizar e propor modelos e arquiteturas de redes neurais convolucionais adequados ao domínio do problema;
- Construir um banco de dados de imagens catalogadas e rotuladas, passível de aplicação de técnicas de visão computacional e aprendizagem de máquina no processo de identificação e classificação automatizadas de fitonematoides.

1.4 Contribuições

Como contribuições desta pesquisa, foram desenvolvidas as seguintes soluções:

- A Caracterização do Estado da Arte sobre o Reconhecimento de Doenças de Plantas com Imagens usando Redes Neurais Convolucionais (ABADE *et al.*, 2021a).
- Reconhecimento de Doenças de Plantas a partir de Imagens Digitais usando Redes Neurais Convolucionais Multicanal (ABADE *et al.*, 2019): uma solução com dois canais de entradas para classificação de doenças de plantas, pautada em imagens foliares utilizando o conjunto de dados PlantVillage (HUGHES; SALATHÉ, 2016).
- Construção do Conjunto de Dados NemaDataset (ABADE *et al.*, 2021b): a criação e disponibilização de um conjunto de dados público contendo 3.063 imagens das cinco espécies de fitonematoides com maior relevância de dano para cultura de soja.

- *NEMANet* (ABADE *et al.*, 2021b): a proposta de um modelo de rede neural convolucional para identificação de fitonematoides em imagens microscópicas usando arquitetura customizada.

1.5 Organização do Documento

Este trabalho está organizado em sete capítulos e suas produções científicas devidamente formalizadas foram organizadas no apêndice. Neste capítulo, em particular, foi apresentado o contexto no qual a pesquisa está inserida, com a motivação, hipóteses, os objetivos pretendidos e as consequentes contribuições.

O Capítulo 2 dos Fundamentos Teóricos apresenta os principais conceitos utilizados durante a elaboração desta pesquisa. O roteiro do capítulo começa com os principais conceitos sobre doenças de plantas, tipos de doenças, a identificação de nematoides parasitos de plantas e análise das imagens microscópicas. O restante do capítulo estrutura o conhecimento face às redes neurais convolucionais.

O Capítulo 3 é a Metodologia desta pesquisa, detalhando o conjunto de métodos envolvidos para concepção de cada solução. A estrutura da metodologia é apresentada e sua execução organizada em fases.

As contribuições desta pesquisa são apresentadas nos Capítulos 4, 5 e 6. No Capítulo 4 elaborou-se uma revisão sistemática da literatura com intuito de caracterizar o estado da arte sobre a identificação e classificação de doenças de plantas usando redes neurais convolucionais. Já no Capítulo 5 é apresentada uma abordagem utilizando arquitetura paralela de múltiplos caminhos para classificação de doenças de plantas, pautada em imagens foliares utilizando o conjunto de dados PlantVillage. Por fim, o Capítulo 6.4 apresenta uma proposta de um modelo de rede neural convolucional para identificação de fitonematoides em imagens microscópicas usando arquitetura paralela de múltiplos caminhos.

Finalmente, no Capítulo 7 apresenta-se a conclusão, a validação das hipóteses levantadas, as limitações e os trabalhos futuros a serem desenvolvidos. Ao final do documento, o Apêndice inclui a produção científica e a primeira página dos artigos publicados ao longo deste trabalho.

Capítulo 2

Fundamentos Teóricos

A concepção deste Capítulo foi motivada para atender ao fato de que a Engenharia Mecatrônica é uma subárea multidisciplinar da engenharia elétrica e mecânica voltada ao projeto de sistemas em geral. Neste contexto, este manuscrito contempla a tópicos das ciências agrárias com ênfase na fitopatologia em conjunto com a ciência da computação com a utilização das Redes Neurais Convolucionais.

Assim, o presente Capítulo apresenta os principais fundamentos teóricos, conceitos e tecnologias relacionados à aprendizagem de máquina e sua escalada evolutiva até as redes neurais convolucionais. A estrutura arquitetônica que compõe as CNNs é detalhada, com ênfase nas camadas de convolução, camadas de ativação e camadas de subamostragem. Adicionalmente, uma evolução temporal sobre as arquiteturas e suas topologias é apresentada. As técnicas de otimização e aumento de ganho também são discutidas. A seguir, são apresentados os conceitos fundamentais de doenças de plantas e como são classificados os tipos de doenças. Posteriormente, a fundamentação teórica essencial sobre a identificação de fitonematoides e a análise de imagens microscópicas.

2.1 Conceito de Doença de Plantas

Conforme descrito por Agrios (2005) e AMORIM *et al.* (2011), a doença consiste em um mau funcionamento de células e tecidos das plantas que resulta da sua contínua irritação, seja por um agente patogênico ou fator ambiental, e conduz ao desenvolvimento de sintomas. Os sintomas, que se constituem na exteriorização da doença devem, portanto, ser de magnitude suficiente para serem perceptíveis pelos sentidos humanos e, assim, caracterizarem o estado de anormalidade ou de doença.

Deste modo, deve-se entender a doença como uma interferência no processo fisiológico da planta, proporcionando um desempenho anormal em suas funções vitais, como na absorção e transporte de água e elementos minerais, na síntese do seu alimento ou na sua utilização. Outra característica da doença que aparece em quase todas as definições é o seu caráter de

processo contínuo, não momentâneo. Embora esta continuidade seja uma característica relativa, em função do tempo de duração do fenômeno, é ela que permite a separação da doença de outros fatores de danos à planta, sendo comumente referidos na literatura pelo nome genérico de injúria (WHEELER, 1975; AGRIOS, 2005; AMORIM *et al.*, 2011).

De acordo com Bos e Parlevliet (1995) e Agrios (2005), a injúria caracteriza-se pela ação momentânea, passageira, de um fator físico-mecânico ou químico sobre a planta. A atuação destes fatores ao nível celular tem uma duração curta, limitada, de fração de segundos, como a mordida de uma lagarta, ou até horas, como a ocorrência de uma geadas, levando à destruição da célula vegetal.

2.2 Tipos de Doenças de Plantas

Basicamente, as doenças de plantas são classificadas conforme o tipo do seu agente causador. Doenças originárias de organismos vivos são denominadas bióticas, ao contrário das doenças produzidas por circunstâncias ecológicas não vivas, denominadas abióticas (HUSIN *et al.*, 2012). As bactérias, fungos, nematoides e vírus são as principais causas de diferentes formas de doenças bióticas. Já as doenças abióticas são causadas por geadas, granizo, condições climáticas adversas, queima por produtos químicos, dentre outros (AGRIOS, 2005). As doenças abióticas são consideradas menos perigosas, por não serem transmissíveis e tampouco infecciosas. Na Figura 2.1 é apresentada uma ilustração esquemática que classifica os tipos de doenças de acordo com as duas principais categorias de agentes causadores.

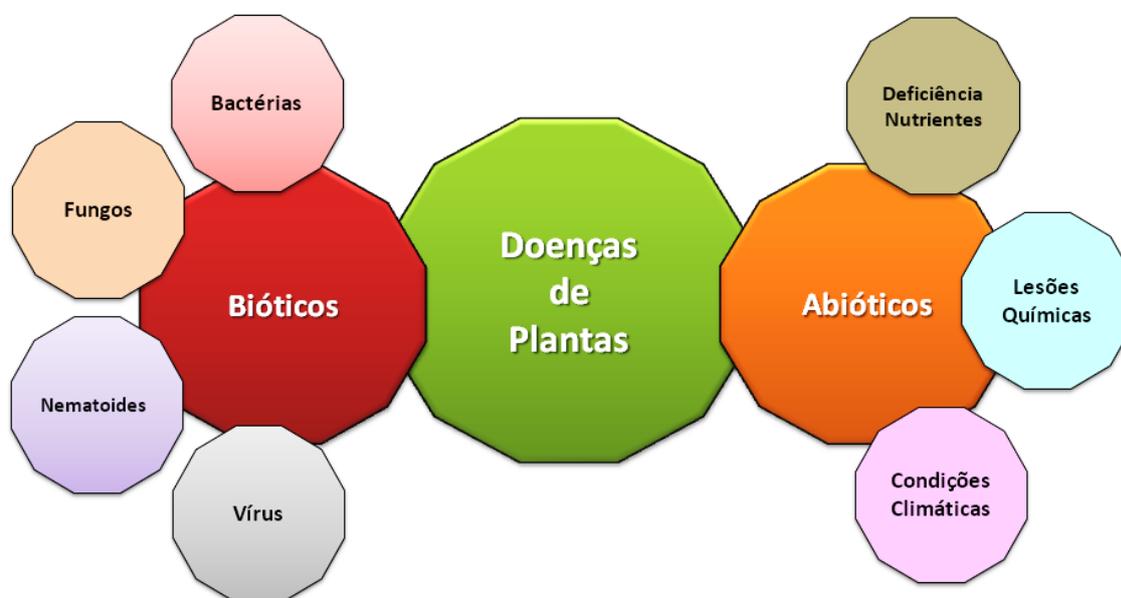


Figura 2.1: Ilustração esquemática da classificação dos tipos de doenças em plantas, agrupadas em duas categorias principais: biótica e abiótica. Adaptada de (GHOSAL *et al.*, 2018)

Tradicionalmente, para diagnosticar uma doença de planta é necessário determinar o tipo do agente causador. Em muitos casos, os sintomas típicos da doença são externalizados pelas

folhas, caules ou frutos da planta, possibilitando, por meio de uma observação criteriosa de uma pessoa experiente, determinar se a doença é causada por um patógeno ou fator ambiental. Entretanto, na maioria dos casos, um exame detalhado dos sintomas e um inquérito das características sintomáticas são necessários para um diagnóstico correto (AGRIOS, 2005).

Nesta perspectiva, os sistemas de detecção de doenças de plantas automatizados propõem a identificação e classificação das doenças de plantas conciliando a *know-how* dos especialistas em fitopatologia e a capacidade de extração de características sintomáticas, por meio dos algoritmos de visão computacional e aprendizado de máquina (KAUR *et al.*, 2019).

Os sistemas de identificação e classificação que recorrem às técnicas de visão computacional possuem um processo organizado em 5 fases: aquisição de imagens, pré processamento, segmentação, extração de características, classificação e reconhecimento (BARBEDO, 2016; THYAGHARAJAN; RAJI, 2019).

2.3 A identificação de Nematoides Parasitos de Plantas

Entre os animais multicelulares, os nematoides são os mais abundantes e diversos; apenas os insetos excedem sua diversidade. Constituem grupo altamente diversificado, pertencendo a um filo próprio (Nematoda) e incluindo formas com diversos hábitos alimentares e diferentes papéis ecológicos no solo. A maioria dos nematoides é de vida livre e se alimenta de bactérias, fungos, protozoários e outros nematoides (40% das espécies descritas); muitos são parasitas de animais invertebrados e vertebrados (45% das espécies descritas) e plantas (15% das espécies descritas) (CARES; HUANG, 2000; LAMBERT; BEKAL, 2002).

Os nematoides são considerados um dos organismos mais difíceis de serem identificados, seja pelo tamanho diminuto, seja pela dificuldade de observação de características chaves para o diagnóstico em microscopia de luz convencional (COOMANS; HEYNS, 1983). As diferenças morfológicas e morfométricas são relativamente pequenas e exigem considerável conhecimento em taxonomia para a segura determinação das espécies. Na Figura 2.2 é apresentada a morfologia geral desses patógenos.

Os nematoides parasitas de plantas, conhecidos como fitonematoides, ocorrem em todos os tamanhos e formas. Sua forma típica é de um animal longo e esguio, parecido com um verme, embora os animais adultos costumam estar inchados e não se parecem mais com vermes. Os fitonematoides são pequenos, entre 300 a 1.000 micrômetros, com alguns chegando até 4 milímetros de comprimento por 15–35 micrômetros de largura (LAMBERT; BEKAL, 2002). Na figura 2.3 é apresentada uma imagem capturada por meio de microscópio de luz com algumas espécies de fitonematoides.

A identificação de nematoides ao nível de espécie frequentemente requer análise morfológica detalhada, crescimento do nematoide em diferentes plantas hospedeiras e análises usando DNA, eletroforese de isoenzimas ou métodos moleculares (DROPKIN, 1989; LAMBERT; BE-

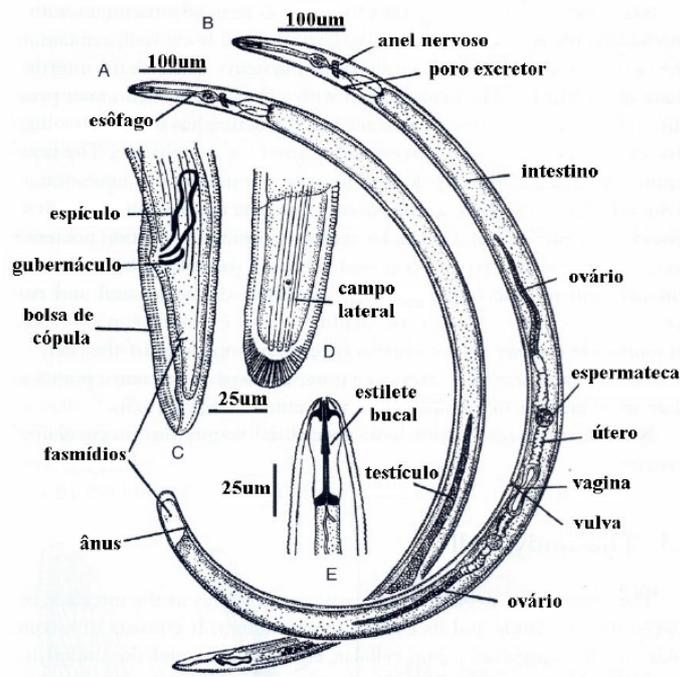


Figura 2.2: Morfologia geral de macho e fêmea de Fitonematoides (A,B) Fonte: (FERRAZ;BROWN, 2016)



Figura 2.3: Exemplo de uma imagem gerada por microscopia de luz com espécies de fitonematoides.

KAL, 2002). As características morfológicas comuns usadas na identificação do nematoide incluem a cavidade bucal (presença ou ausência e formato do estilete), a forma e a sobreposição das glândulas esofagianas com o intestino, o tamanho e a forma do corpo do nematoide na fase adulta, o tamanho da cabeça, cauda, número e posição dos ovários na fêmea. Caracteres mais sutis podem incluir o número de linhas na cutícula do nematoide ou a presença ou ausência de órgãos sensoriais semelhantes a poros (LAMBERT; BEKAL, 2002; PERRY; MOENS, 2011).

O microscópio de luz é a ferramenta mais utilizada para a identificação de fitonematoides, valendo-se da observação tanto da anatomia externa como da interna do patógeno. A anatomia dos fitonematoides está diretamente relacionada às suas características funcionais. O formato do corpo, incluindo seu comprimento, o grau de desenvolvimento da armadura cefálica e o tamanho do estilete são características fortemente relacionadas ao modo de parasitismo (OLIVEIRA *et al.*, 2011).

Comparando-se os diferentes métodos de identificação de fitonematoides, provavelmente aquele que recebeu menos benefícios com o desenvolvimento de novos equipamentos e técnicas foi a microscopia de luz, razão pela qual sua importância relativa tem indiscutivelmente decaído. É inegável que outros métodos de identificação têm se tornado de uso mais fácil, frequente e preciso. Porém, a microscopia de luz ainda mantém seu lugar como importante ferramenta para a identificação dos fitonematoides, principalmente em função das seguintes razões: (i) permite o estabelecimento claro entre os aspectos funcionais e morfológicos dos organismos examinados; (ii) dos métodos disponíveis, é aquele que permite o contato direto entre o observador e o organismo examinado; (iii) como consequência do item anterior, é o método menos sujeito a resultar em erros grosseiros em função de descuidos nos procedimentos; (iv) é ainda o método que proporciona resultados mais rápidos; (v) é o método mais adequado para avaliações quantitativas (DROPKIN, 1989; OLIVEIRA *et al.*, 2011).

Em função das razões apontadas, a microscopia de luz é a mais utilizada para levantamentos populacionais de fitonematoides, com objetivo de estabelecimento de recomendações de manejo.

Entretanto, a literatura enumera desvantagens e limitações significativas quanto ao método de microscopia de luz em relação aos outros métodos de identificação. A principal desvantagem é que o exame microscópico não é suficiente para a identificação segura de várias espécies de nematoides, cujas características morfológicas diferenciais são de observação extremamente difícil ou ainda espécies cuja descrição contemplou já de início características que somente podem ser obtidas por outros métodos (OLIVEIRA *et al.*, 2011).

Por outro lado, não se pode desprezar a microscopia de luz na diagnose de fitonematoides, visto que é o primeiro método que deve ser utilizado pois, em várias situações, ela basta para a identificação segura dos patógenos. No mínimo, a microscopia de luz fornecerá a identificação inicial, até nível de família ou gênero, ou ainda grupo de espécies, informação que, se não for suficiente, ao menos fornecerá importantes subsídios para guiar ou orientar a escolha do método mais adequado para o prosseguimento do trabalho de identificação (OLIVEIRA *et al.*, 2011).

2.4 Análise de Imagens Microscópicas

Os microscópios têm sido usados há muito tempo para capturar, observar, medir e analisar imagens e estruturas de vários organismos vivos em escalas muito abaixo da percepção visual humana. Com o advento das tecnologias computacionais e sensores de imagem com alto de-

sempenho, a imagem digital ganhou destaque e está substituindo a fotomicrografia tradicional baseada em filme como o método de aquisição e armazenamento de imagem de microscópio mais amplamente utilizado. O processamento digital de imagens não é apenas uma extensão natural, mas está provando ser essencial para o sucesso da análise e interpretação de dados subsequentes da nova geração de imagens microscópicas (WU *et al.*, 2008).

As limitações dos instrumentos de imagem ótica e o ruído inerente à imagem ótica tornam o processo de aprimoramento de imagem desejável para muitas aplicações de processamento de imagem microscópica. O aprimoramento de imagem é o processo de aprimorar a aparência de uma imagem ou um subconjunto da imagem para melhor contraste, ou visualização de certos recursos e, subsequentemente, facilitar uma análise de imagem mais precisa. Com este processo, a visibilidade dos recursos selecionados em uma imagem pode ser melhorada, mas o conteúdo de informação inerente não pode ser aumentado. Assim, o desafio não está apenas em processar imagens, mas em processá-las corretamente e efetivamente (WU *et al.*, 2008). Na Figura 2.4 são apresentadas algumas imagens microscópicas desde a composição celular até os patógenos causadores de doenças em plantas.

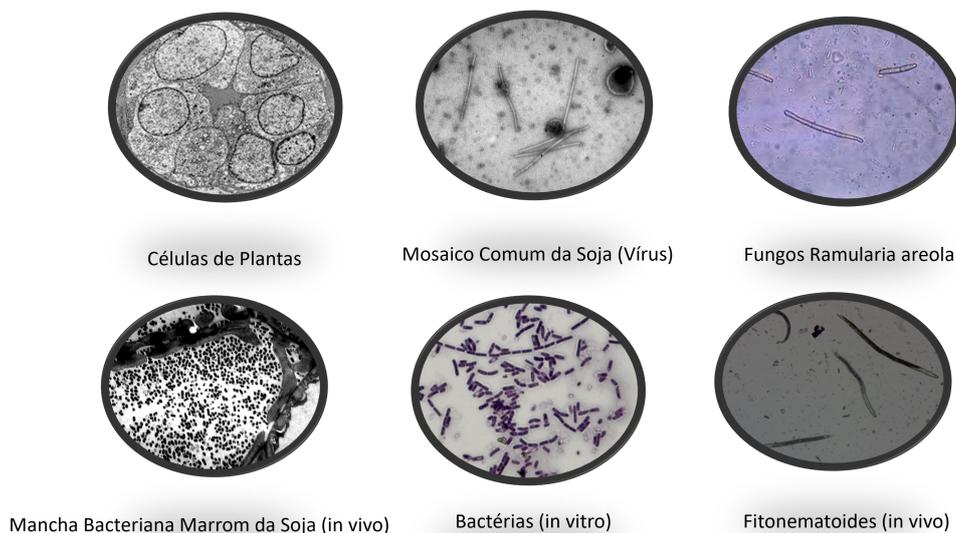


Figura 2.4: Exemplos de imagens geradas por microscopia de luz com a composição celular da planta até os patógenos causadores de doenças. Fonte: (EMBRAPA, 2021) (UFV, 2016) (ABADE *et al.*, 2021b)

O processamento digital pode ser usado para extrair informações quantitativas sobre o espécime de uma imagem de microscópio e pode transformar uma imagem para que uma versão exibida seja muito mais informativa do que seria (ROEDER, 2019). Nesta perspectiva, o Processamento Morfológico de Imagem (MP) baseia-se em sondar uma imagem com um elemento estruturante e filtrar ou quantificar a imagem de acordo com a forma como o elemento estruturante se encaixa (ou não) na imagem. Uma imagem binária é composta de pixels de primeiro e segundo plano, e conjuntos conectados de pixels de primeiro plano constituem os objetos na imagem. O processamento morfológico tem aplicações em diversas áreas do processamento

de imagens, como filtragem, segmentação e reconhecimento de padrões, tanto para imagens binárias quanto em tons de cinza. Uma das vantagens do MP é ser adequado para processamento discreto de imagens, pois seus operadores podem ser implementados em computadores digitais com total fidelidade às suas definições matemáticas. Outra vantagem do MP é sua estrutura de bloco de construção inerente, no qual operadores complexos podem ser criados pela composição de alguns operadores primitivos (WU *et al.*, 2008).

No entanto, os profissionais que recorrem à microscopia são inundados com informações que eles têm que normalizar, filtrar, diminuir o ruído, ajustar foco, reconstruir, registrar, segmentar, classificar, etc. Um dos problemas inerentes à microscopia de luz é a geração de ruído nas imagens. Ruídos são flutuações aleatórias na intensidade em pixel da imagem que obscurecem o sinal real gerado ao longo da etapa de amostra. O ruído está sempre presente nas imagens devido às condições de baixa luminosidade, coletando um número limitado de fótons e o circuito elétrico do microscópio (ROEDER, 2019).

Imagens microscópicas e processamento de imagens são de interesse crescente para as comunidades científicas e de engenharia. Desenvolvimentos recentes em tecnologias celulares, moleculares e nanométricas levaram a descobertas rápidas. Eles avançaram significativamente nas fronteiras do conhecimento humano em Biologia, Medicina, Química, Farmacologia e muitos campos relacionados.

2.5 Conceito de Aprendizagem de Máquina

Ao decompor semanticamente o termo ‘Aprendizado de Máquina’ (*Machine Learning*), de acordo com *Oxford English Dictionary*, encontra-se que o Computador é uma máquina eletrônica capaz de criar, armazenar, organizar, localizar arquivos com uma diversidade de conteúdo e essencialmente realizar cálculos. Já a Aprendizagem é definida como um processo pelo qual as habilidades, competências e conhecimentos são adquiridos ou modificados por meio da experiência, exercício e prática.

Deste modo, uma definição completa para aprendizagem de máquina deve englobar o processo de aquisição do conhecimento com base no computador e indicar onde as habilidades, competências ou conhecimentos que serão desenvolvidos podem ser obtidos.

Nesta perspectiva, Alpaydin (2014) define o aprendizado de máquina como a capacidade de um programa de computador adquirir ou desenvolver novos conhecimentos ou habilidades, por meio de exemplos existentes ou não, visando a otimização do desempenho.

O aprendizado de máquina lida com o universo de incertezas, tais como, por exemplo: qual é a melhor previsão sobre uma condição futura, sob a perspectiva histórica dos dados passados? Qual o melhor modelo para explicar alguns dados? Assim, a teoria das probabilidades pode ser aplicada como uma ferramenta capaz de tratar problemas que envolvam incertezas.

Com uma abordagem probabilística do aprendizado de máquina, Murphy (2012) define o

aprendizado como um conjunto de métodos que podem detectar automaticamente padrões nos dados e, posteriormente, utilizar esses padrões para prever dados futuros ou executar outros tipos de tomada de decisão sob uma incerteza.

Qualquer método que incorpora informações de amostras de treinamento no projeto de um classificador emprega aprendizado. Esta etapa de aprendizado deve ser considerada, pois, a maioria dos problemas práticos de reconhecimento de padrões são tão complexos que não se pode prever a melhor decisão de classificação. De modo geral, a aprendizagem de máquina utiliza três métodos de aprendizado, sendo: Aprendizagem Supervisionada, Aprendizagem Não Supervisionada e Aprendizado por Reforço. O método utilizado neste trabalho é o aprendizado supervisionado, que será explicitado a seguir.

2.5.1 Aprendizado Supervisionado

Os problemas que se enquadram na categoria supervisionada utilizam algoritmos para induzir modelos preditivos, por meio da observação de um conjunto de objetos rotulados, tipicamente referenciado como conjunto de treinamento (LUXBURG; SCHÖLKOPF, 2011). Os rótulos contidos em tal conjunto correspondem às classes ou valores obtidos por alguma função desconhecida. Desse modo, um algoritmo de classificação buscará produzir um classificador capaz de generalizar as informações contidas no conjunto de treinamento, com a finalidade de classificar, posteriormente, objetos cujo rótulo seja desconhecido.

Formalmente, um conjunto de dados de treinamento pode ser definido como uma coleção de tuplas:

$$D = \{(x_i, y_i)\}_i^N = 1 \quad (2.1)$$

sendo que D representa o conjunto de elementos de treinamento e $\{i, \dots, N\}$ é o número de exemplos, com $i \in \mathbb{N}$. Em cada tupla, x_i indica um objeto descrito por m características e y_i indica o rótulo correspondente a x_i . Quando os valores de y_i são definidos por uma quantidade limitada de valores discretos, tem-se um problema de classificação. Quando tais valores são contínuos, tem-se um problema de regressão.

Assim, é possível descrever genericamente o funcionamento dos algoritmos de aprendizado supervisionado, por meio do seguinte exemplo: um conjunto de N elementos de treinamento $\{(x_1, y_1), \dots, (x_N, y_N)\}$ de tal modo que x_i é o vetor de características do i -ésimo elemento e y_i é seu rótulo (classe), um algoritmo de aprendizado supervisionado procura uma função $f : A \rightarrow B$ sendo que A é o espaço de entrada e B é o espaço de saída. A função f é um elemento de algum espaço de possíveis funções F , geralmente denominado espaço de hipótese.

A aprendizagem, então, seria uma busca por meio de hipóteses possíveis por aquele algoritmo que poderá retornar um bom desempenho, mesmo utilizando novos exemplos além do conjunto de treinamento e teste.

Os problemas de aprendizado supervisionado podem ser agrupados como sendo problemas de regressão e classificação, sendo que um problema de classificação ocorre quando a variável de saída é uma categoria como, por exemplo, ‘planta saudável’ ou ‘planta doente’. Já um problema de regressão ocorre quando a variável de saída corresponde a um valor real.

Classificação

O objetivo de um problema de classificação é aprender um mapeamento das entradas às saídas, sendo que $y \in \{1, \dots, C\}$, e C representa o número de classes (categorias). Quando $C = 2$, a classificação é denominada binária, neste caso $y \in \{0, 1\}$. Entretanto, caso $C > 2$, a classificação é denominada multi classe. Um exemplo de tarefa de classificação é o reconhecimento de objetos, no qual a entrada é uma imagem e a saída é um código que identifica o objeto na imagem (MOHRI *et al.*, 2018).

Regressão

A regressão gera uma função que descreve a relação entre uma ou mais variáveis preditoras e a variável de resposta. Nesse caso, para um conjunto de treinamento da forma $\{(x_1, y_1), \dots, (x_N, y_N)\} / x_i \in \mathbb{R}^d, y_i \in \mathbb{R}, i = 1, \dots, N\}$, a função denominada regressora é da forma $r : \mathbb{R}^d \rightarrow \mathbb{R}$

A principal diferença entre a classificação e a regressão é a medida de erro da regressão baseada na magnitude da diferença entre o rótulo com valor real previsto, o verdadeiro, e não baseado na igualdade ou desigualdade desses dois valores (MOHRI *et al.*, 2018).

As técnicas dos algoritmos de aprendizado de máquina supervisionado incluem Regressão Linear e Logística, Árvores de Decisão, *Naive Bayes*, K-Vizinhos mais Próximos, Floresta Aleatória, Máquinas de Vetores de Suporte (SVM) e Redes Neurais Artificiais.

2.5.2 Perceptron

O perceptron de camada única (*Single Layer Perceptron* - SLP), introduzido por Rosenblatt (1958), é um classificador binário de aprendizagem supervisionada. Sua arquitetura é precursora das Redes Neurais Artificiais.

O algoritmo do Perceptron atua como um classificador linear que faz previsões com base em uma função preditora linear, combinada com um conjunto de pesos e um vetor de características. A camada de saída é composta por uma única unidade de limiar linear com pesos w que recebem a entrada x de um conjunto de neurônios de entrada.

A unidade de saída usa uma função de sinal como uma função de ativação, implementando uma tarefa de classificação entre duas classes no espaço $\{-1, 1\}$, sendo a decisão da classe pautada no sinal do produto escalar $w^t x$.

Como exemplo, dado o conjunto de treinamento $X = \{x_1, \dots, x_N\}$ com as saídas desejadas associadas $\{t_1, \dots, t_N\}$ ($t_n \in \{-1, 1\}$) onde -1 representa a classe C_1 e 1 representa a classe C_2 . Para cada entrada x_n , o perceptron gera uma saída $o_n \in \{-1, 1\}$. O treinamento

minimiza o critério de perceptron E_p (DUDA; HART, 1974; BISHOP, 1995) definido sobre o conjunto de treinamento como:

$$E_p = - \sum_{n, o_n \neq t_n} t_n \mathbf{w}^t \mathbf{x}_n \quad (2.2)$$

onde a soma é realizada sobre os exemplos classificados incorretamente ($o_n \neq t_n$). Este critério tenta encontrar o hiperplano linear no espaço x que separa as duas classes.

Nota-se que o produto escalar $-t_n \mathbf{w}^t \mathbf{x}_n$ é sempre positivo para um \mathbf{x}_n classificado incorretamente. Portanto, minimizar o critério de perceptron é equivalente a resolver a desigualdade linear $t_n \mathbf{w}^t \mathbf{x}_n > 0$ para \mathbf{x}_n , que pode ser feito de modo incremental, uma amostra de cada vez ou em lotes.

É preciso destacar que, como um classificador linear, o Perceptron nunca chegará ao estado com todos os valores de entrada classificados corretamente caso o conjunto e treinamento X não seja linearmente separável, ou seja, os exemplos positivos não puderem ser separados dos exemplos negativos, por meio do hiperplano.

2.5.3 Multi-Layer Perceptron

O perceptron de multicamadas (*Multi-Layer Perceptron* - MLP), também conhecido como rede neural *feed-forward* de multicamadas, foi proposto como um aprimoramento do modelo perceptron, implementando as seguintes características:

- O uso de pelo menos uma camada oculta na rede, interposta entre as camadas de entrada e saída, possibilitando a representação interna e adição de mais recursos de modelagem;
- O uso de funções sigmóides nas camadas ocultas e na camada de saída.
- A implementação do algoritmo *Backpropagation* (CHAUVIN; RUMELHART, 1995), que possibilita o treinamento de toda a rede hierarquicamente.
- As múltiplas camadas ocultas e funções de ativação não lineares permitem a MLP distinguir dados que não são linearmente separáveis.

Essas características, juntamente com uma sólida estrutura teórica apoiada em resultados concretos, fez das MLPs a arquitetura de rede neural mais usada para tarefas de classificação.

Os estudos de Cybenko (1989) provaram que uma MLP com uma única camada oculta com unidades sigmóides pode aproximar qualquer mapeamento com a precisão desejada, estabelecendo as MLPs como um aproximador universal (HORNIK *et al.*, 1989). Entretanto, na prática, um desempenho de classificação sub ótimo pode surgir devido aos fatores relacionados ao treinamento inadequado, ao número insuficiente de camadas ocultas ou um conjunto de dados com número pouco representativo de amostras.

Na Figura 2.5 é apresentado um exemplo da arquitetura de uma MLP. As unidades dispostas em camadas que compreendem uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída.

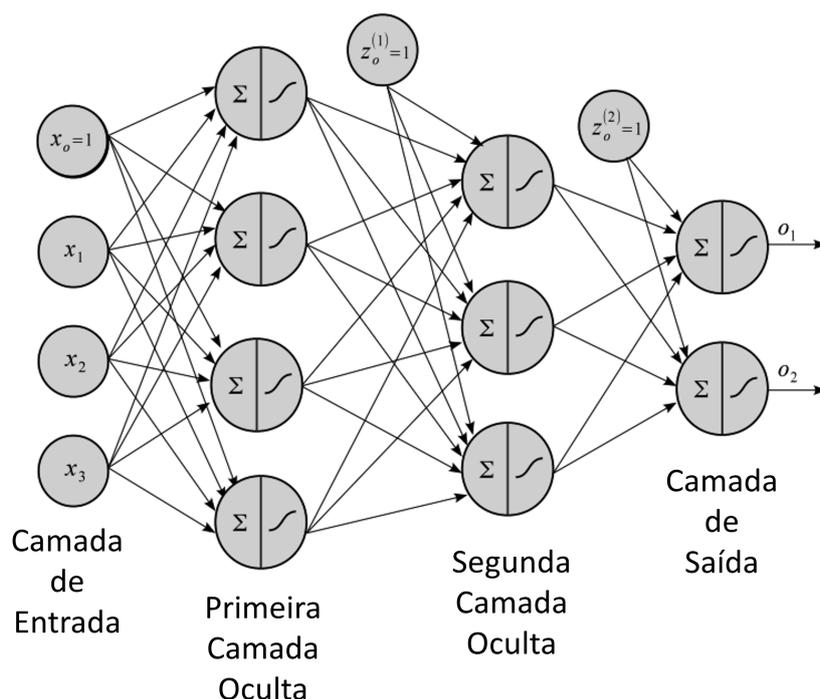


Figura 2.5: Arquitetura MLP composta por uma camada de entrada, duas camadas ocultas e uma camada de saída. Fonte: Aggarwal (2014)

As camadas são organizadas sequencialmente seguindo o fluxo de dados da camada de entrada, passando pelas camadas ocultas até a camada de saída. Os dados de entrada são coletados e processados pelas unidades da camada de entrada e posteriormente propagados para as unidades das camadas ocultas para a camada de saída. Cada unidade, em uma camada, recebe dados das unidades da camada anterior e as unidades da camada de saída fornecem a decisão final de classificação. Esse processo é denominado propagação direta da rede. Nas MLPs as unidades nas camadas ocultas e de saída são tipicamente unidades com funções sigmóides. A escolha do número e tipo de unidades da camada de saída é adaptada de acordo com a tarefa de classificação da rede.

Para problemas de classificação com duas classes $\{C_1, C_2\}$, o MLP contém uma ou mais camadas ocultas e uma única unidade de saída sigmoideal que produz um único valor de saída o . Supondo que o valor alvo seja 0 para a classe C_1 e 1 para classe C_2 , a regra de classificação é descrita como (AGGARWAL, 2014):

$$\begin{cases} x \in C_1 & \text{if } o > 0.5 \\ x \in C_2 & \text{senão} \end{cases} \quad (2.3)$$

com a saída o no intervalo de $[0, 1]$ ou $[-1, 1]$ caso a função \tanh seja usada, onde o limiar de decisão passar a ser zero). Claramente, a unidade de saída implementa uma regressão logística

no espaço de recursos gerados pela última camada oculta. Uma MLP para problemas de duas classes é muito semelhante ao perceptron no uso de uma única unidade de saída. Entretanto, as habilidades de uma MLP são aprimoradas pelo uso das camadas ocultas.

Para problemas envolvendo várias classes ($k > 2$), o MLP possui uma ou mais camadas ocultas e K unidades de saída, cada uma correspondente a uma classe específica do conjunto de treinamento. A função de destino é codificada com um esquema $1 - of - k$ com valores entre 0, 1 e a saída o_k é vista como uma função discriminante da classe C_k , permitindo a decisão de classificação como (AGGARWAL, 2014):

$$x \in C_k \quad \text{if } k = \arg \max_j o_j \quad (2.4)$$

para $j \in \{1, \dots, k\}$ com $o_k \in [0, 1]$ quando a sigmoide é utilizada como função de ativação. O uso de um esquema de codificação $1 - of - k$ para a função de destino é padrão.

2.6 Redes Neurais Convolucionais

Uma rede neural profunda (*Deep Neural Network - DNN*) é uma rede neural artificial com várias camadas ocultas entre as camadas de entrada e saída. Semelhante às MLPs, as DNN podem modelar relacionamentos não lineares e o termo "profundo" é decorrente do grande número de camadas ocultas presentes nestas arquiteturas. As camadas extras (ocultas) permitem a composição de recursos das camadas inferiores, fornecendo um potencial de modelagem de dados complexos com menos unidades do que em uma rede 'rasa' com desempenho semelhante (BENGIO, 2009).

Entretanto, o treinamento de redes neurais profundas com um grande número de camadas e unidades ocultas é uma tarefa difícil. Com vários parâmetros na ordem de milhões, a superfície do erro é complexa e possui vários mínimos locais, na qual muitos deles levam a um baixo desempenho (AUER *et al.*, 1995; FUKUMIZU; AMARI, 2000). Além disso, durante a retropropagação do sinal, há a questão do gradiente de fuga, no qual a norma do gradiente fica menor à medida que a regra da cadeia do cálculo do gradiente se expande por várias camadas (BENGIO *et al.*, 1994; HOCHREITER, 1998).

Por fim, um DNN com um grande número de pesos requer alto poder computacional, o que vem motivando os pesquisadores a desenvolverem técnicas de aprendizado profundo com várias estratégias e heurísticas capazes de superar esses problemas. Algumas dessas estratégias incluem o uso do conhecimento prévio sobre o domínio da aplicação, a aplicação de uma técnica de treinamento seletivo em camadas e o uso de ferramentas de alto desempenho para otimização.

Primordialmente, as arquiteturas DNN foram projetadas para lidar com sinais 1D (texto, voz). Posteriormente, grande parte dos trabalhos em redes neurais profundas foram motivados pelo reconhecimento visual de padrões (LECUN *et al.*, 1998) lidando com sinais 2D (imagens), dando origem a um tipo especializado de rede neural denominado redes neurais convolucionais.

As redes neurais convolucionais (*Convolutional Neural Network* - CNN) são estruturas de aprendizado profundo que ganharam notoriedade pelos resultados expressivos quanto à classificação de objetos em imagens. Uma CNN é uma topologia ou arquitetura que se aproveita de relações espaciais para reduzir o número de parâmetros que devem ser aprendidos utilizando os métodos de *feedforward* e *backpropagation* (SAINATH *et al.*, 2013). O nome ‘rede neural convolucional’ indica que a rede emprega uma operação matemática chamada convolução.

Para os matemáticos, uma convolução é um tipo especializado de operação linear que, a partir de duas funções, resulta em uma terceira função que mede a soma do produto das funções origens ao longo da região subentendida pela superposição delas em função do deslocamento (WIDDER, 1971). Redes convolucionais são simplesmente redes neurais que usam convolução no lugar da multiplicação geral da matriz em pelo menos uma de suas camadas. Geralmente, a operação usada em uma rede neural convolucional não corresponde exatamente à definição de convolução usada em outros campos, como na engenharia ou matemática pura (GODFELLOW *et al.*, 2016).

Basicamente, uma CNN é composta por diversos tipos de camadas: camadas de convolução, camadas de *pooling/subsampling* (agrupamento/subamostragem), camadas de normalização e camadas completamente conectadas. A arquitetura mais comum para uma CNN é aquela organizada em estágios, na qual cada etapa é composta por uma ou mais camadas de convolução em sequência, seguidas por uma camada de pooling que, por sua vez, pode ser seguida por uma camada de normalização. Após o estágio final da rede, são adicionadas uma ou mais camadas completamente conectadas (WANG *et al.*, 2016). Na Figura 2.6 é possível acompanhar a arquitetura da LeNet proposta por Lecun *et al.* (1998).

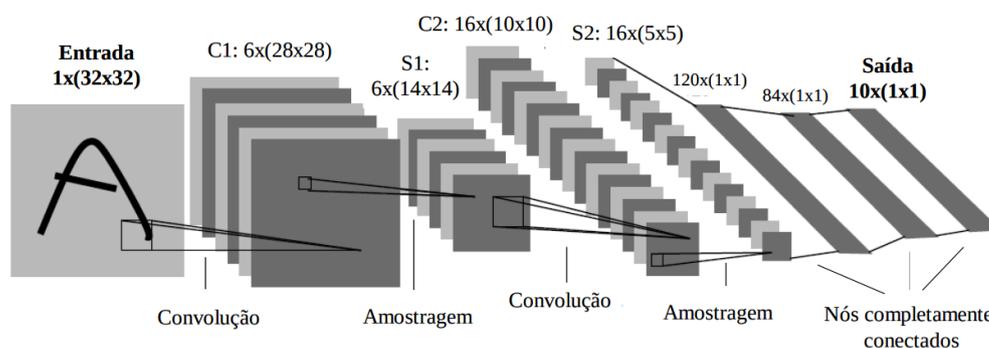


Figura 2.6: Arquitetura CNN. Adaptada de Lecun *et al.* (1998)

De modo geral, em contraste com as redes neurais completamente conectadas, de complexa construção e uso, as CNNs possuem uma topologia simplificada, visto que as camadas de convolução e *pooling* são intercaladas ao longo da arquitetura, corroborando para o aprendizado de combinações não lineares das entradas originais, o que de fato é denominado como processo automático de extração de características (*features extraction*).

A entrada para uma camada convolucional pode ser uma imagem ($m \times m \times c$), em que m representa a largura e altura da imagem, sendo c o número de canais da imagem RGB (3 canais

de cores). A camada convolucional possui k filtros (*kernels*) de tamanho $(n \times n \times q)$, em que n é a menor dimensão da imagem, sendo $q \leq c$ podendo variar de acordo com o *kernel*.

O tamanho dos filtros dão origem à estrutura conectada localmente, cada qual convoluida com a imagem para produzir k mapas de características de tamanho $(m - n + 1)$. Cada mapa é subamostrado, tipicamente com *pool* médio ou máximo sobre regiões contíguas $(p \times p)$, onde p varia entre 2 para imagens pequenas e geralmente não é superior a 5 para imagens maiores. Antes ou depois de cada camada de subamostragem, um ‘bias’ é adicionado e uma função de ativação sigmoide é aplicada em cada mapa de características.

Os próximos tópicos abordarão os conceitos e detalhes destas duas camadas determinantes para as arquiteturas CNNs.

2.6.1 Convolução

Na terminologia das redes neurais convolucionais, a entrada geralmente é uma matriz multidimensional de dados e o *kernel*, conseqüentemente, uma matriz multidimensional de parâmetros que são reajustados pelo algoritmo de aprendizagem. Como cada elemento de entrada e do *kernel* devem ser explicitamente armazenados separadamente, assume-se que essas funções são zero em qualquer lugar, exceto no conjunto finito de pontos para os quais armazenamos valores. Na prática, isso possibilita implementar o somatório infinito como um somatório sobre um número finito de elementos da matriz (GOODFELLOW *et al.*, 2016).

Deste modo, quando uma entrada bidimensional I e um *kernel* bidimensional K são utilizados, realizam-se convoluções em mais de um eixo de cada vez. Algumas bibliotecas de redes neurais implementam uma função chamada de ‘*cross-correlation*’, que é a mesma convolução, porém sem inverter o *kernel*. No aprendizado de máquina, as convoluções são utilizadas simultaneamente com outras funções, e a combinação dessas funções não ocorrem independente das operações de convolução alterar ou não o *kernel* (GOODFELLOW *et al.*, 2016).

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i + m, j + n)K(m, n) \quad (2.5)$$

A convolução aproveita três ideias importantes que auxiliam na melhoria dos sistemas de aprendizagem de máquina: interações esparsas, compartilhamento de parâmetros e representações equivalentes. Adicionalmente, a convolução fornece um meio para trabalhar com entradas de tamanho variável. Na Figura 2.7 é ilustrada uma operação de convolução sem inversão de Kernel.

2.6.2 Não-Linearidade

A camada convolucional é geralmente seguida por operações não lineares. A não linearidade é alcançada usando-se uma família específica de funções chamadas funções de ativação.

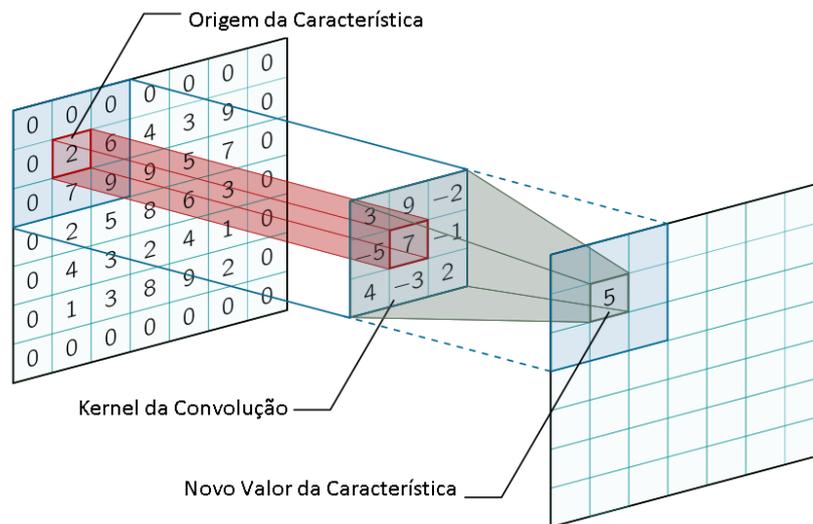


Figura 2.7: Ilustração esquemática de uma operação de convolução 2D sem inversão do *kernel* (3x3) para calcular o valor de saída no pixel (2,2). Adaptada de Kiswani e Verner (2014)

Estas garantem que a representação no espaço de entrada seja mapeada para uma representação esparsa, portanto, alcançando uma certa invariância para variabilidade de dados e uma representação computacionalmente eficiente (GLOROT *et al.*, 2011). Entretanto, representações esparsas são mais resilientes às pequenas modificações que às representações densas.

Primordialmente, as funções de ativação sigmoide e hiperbólica tangente eram as mais comumente utilizadas. Porém, como a escala e o volume de imagens aumentaram significativamente, as CNNs começam a prospectar novos desafios quanto ao reconhecimento e detecção de objetos em grande volume de imagens e vídeos. Aliadas a essa crescente demanda, novas funções de ativação estão sendo propostas continuamente.

ReLU é a função de ativação amplamente utilizada nos projetos de redes neurais atualmente. As Unidades Lineares Retificadas (ReLUs) e variantes, de modo geral, podem ser expressas usando a seguinte fórmula:

$$f(z_{lk}^{uk}) = \begin{cases} \max(z_{lk}^{uk}, 0) & \text{if } z_{lk}^{uk} > 0, \\ \min(\alpha \cdot z_{lk}^{uk}, 0) & \text{if } z_{lk}^{uk} \leq 0, \end{cases} \quad (2.6)$$

no qual z_{lk}^{uk} é o valor de entrada na posição (u, k) no mapa de características k na 1.^a camada e α é a inclinação da função linear negativa.

Uma camada típica de uma CNN consiste em três estágios. No primeiro estágio, a camada realiza várias convoluções em paralelo para produzir um conjunto de ativações lineares, enquanto no segundo estágio cada ativação linear é executada por meio de uma função de ativação não linear como, por exemplo, a função de ativação linear retificada (ReLU) (NAIR; HINTON, 2010). Esse estágio por vezes é denominado estágio detector. No último estágio, é utilizado uma função de 'pooling' para modificar ainda mais a saída da camada (GOODFELLOW *et al.*, 2016).

Uma função de ‘pooling’ substitui a saída da rede em um determinado local pela estatística usual das saídas próximas. Por exemplo, a operação denominada ‘Max-Pooling’ (ZHOU *et al.*, 1988; NAGI *et al.*, 2011) expressa a saída máxima dentro de uma vizinhança retangular. Outras funções populares de agrupamento incluem o ‘Average-Pooling’ para média de uma vizinhança retangular, a L^2 norm de uma vizinhança retangular ou uma média ponderada com base na distância do pixel central. As expressões abaixo representam cada uma dessas funções, respectivamente:

$$h_j^n(x, y) = \max_{\bar{x} \in N(x), \bar{y} \in N(y)} h_j^{n-1}(\bar{x}, \bar{y}) \quad (2.7)$$

$$h_j^n(x, y) = 1/K \sum_{\bar{x} \in N(x), \bar{y} \in N(y)} h_j^{n-1}(\bar{x}, \bar{y}) \quad (2.8)$$

$$h_j^n(x, y) = \sqrt{\sum_{\bar{x} \in N(x), \bar{y} \in N(y)} h_j^{n-1}(\bar{x}, \bar{y})^2} \quad (2.9)$$

Na Figura 2.8 são ilustradas operações com a função ‘MaxPooling’ no espaço de pixel 4x4. A função ‘MaxPooling’ é um processo de discretização baseado em amostra, no qual o objetivo é produzir uma amostragem reduzida de uma representação de entrada, reduzindo sua dimensionalidade e permitindo suposições sobre as características contidas nas sub-regiões classificadas.

Em todos os casos, a função de ‘pooling’ tem por objetivo reduzir o número de parâmetros a serem aprendidos pela rede. Adicionalmente, reduz o ‘overfitting’ e, conseqüentemente, aumenta o desempenho geral e a precisão da rede.

2.6.3 Regularização

O *Overfitting* é um problema imperceptível em CNNs profundas, que pode ser efetivamente reduzido pela regularização. Nesta subseção serão apresentadas algumas técnicas de regularização comumente utilizadas, sendo elas: ℓ_p -norm, Dropout e DropConnect (GU *et al.*, 2018).

- *ℓ_p -norm Regularization*: a regularização modifica a função ao adicionar termos que penalizam a complexidade do modelo. Formalmente, se a função de perda for $\mathcal{L}(\theta, x, y)$, então a função de perda regularizada será:

$$E(\theta, x, y) = \mathcal{L}(\theta, x, y) + \lambda R(\theta) \quad (2.10)$$

onde $R(\theta)$ é o termo de regularização e λ é a força de regularização adicionados.

A função de regularização ℓ_p -norm é normalmente aplicada como $R(\theta) = \sum_j \|\theta_j\|_p^p$. Quando $p \geq 1$, a ℓ_p -norm é conexa, o que facilita a otimização e torna esta função atrativa (HINTON *et al.*, 2012). Quando $p = 2$, a função de regularização ℓ_2 -norm é comumente referida como sendo uma redução de peso (*weight decay*).

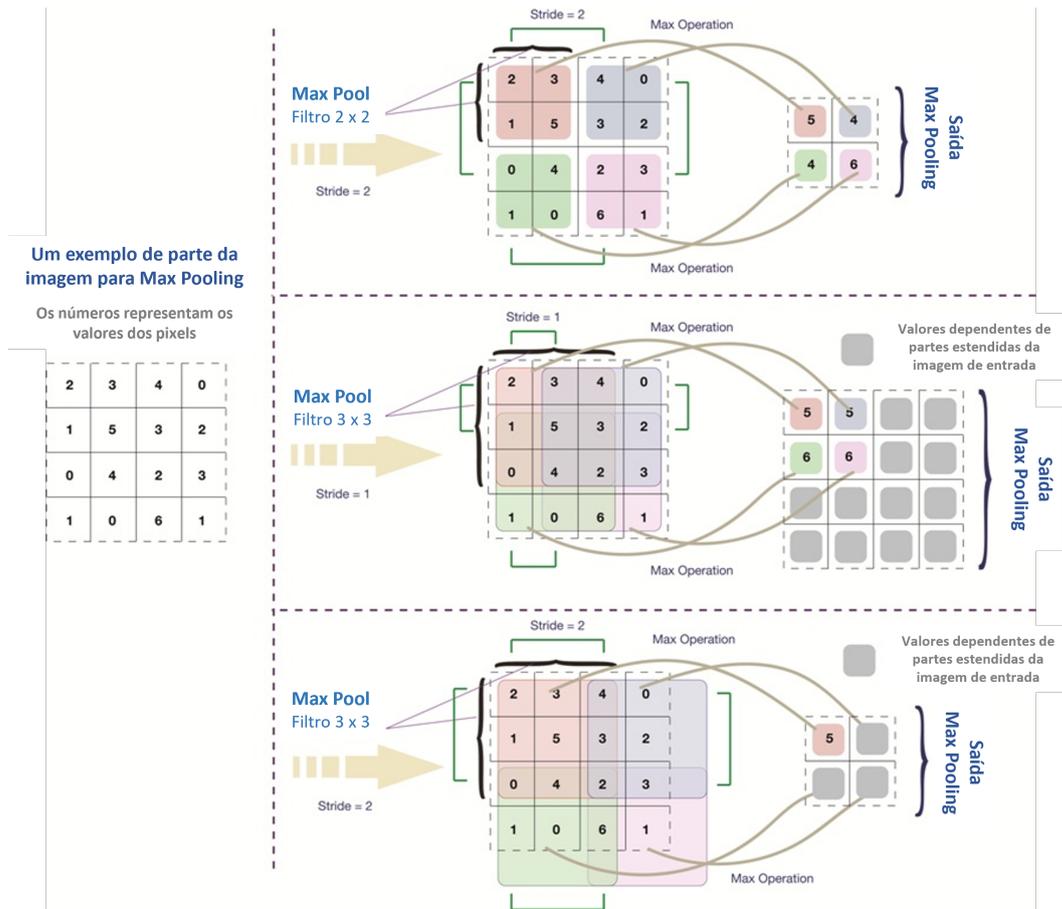


Figura 2.8: Exemplo de operações com a função MaxPooling. Adaptada de Zajačko *et al.* (2018)

- *Dropout*: foi definido por Hinton *et al.* (2012) e tem se mostrado muito eficaz na redução do *overfitting*. A saída do *Dropout* é expressa por $\mathbf{y} = \mathbf{r} \times a(\mathbf{W}^T \mathbf{x})$, onde $\mathbf{x} = \{x_1, x_2, \dots, x_n\}^T$ é a entrada para a camada totalmente conectada, $\mathbf{W} \in \mathbb{R}^{n \times d}$ é uma matriz de peso e \mathbf{r} é um vetor binário de tamanho d cujos elementos são desenhados independentemente de uma distribuição Bernoulli com parâmetros p , *i.e.* $r_i \sim \text{Bernoulli}(p)$. O *Dropout* pode evitar que a rede se torne muito dependente de qualquer um (ou qualquer pequena combinação) de neurônios, e pode forçar a precisão da rede, mesmo na ausência de certas informações. Ao contrário da regularização ℓ_p -norm e ℓ_2 -norm, o *Dropout* não depende da modificação da função de custo.
- *DropConnect*: é uma evolução do *Dropout* proposta por Wan *et al.* (2013). Em vez de definir aleatoriamente as saídas dos neurônios para zero, o *DropConnect* define aleatoriamente os elementos da matriz de peso \mathbf{W} para zero. A saída do *DropConnect* é expressa por $y = a((\mathbf{R} \times \mathbf{W})\mathbf{x})$, onde $R_{ij} \sim \text{Bernoulli}(p)$. Além disso, os vieses também são mascarados durante o processo de treinamento.

Na Figura 2.9 são ilustradas as diferenças em camadas totalmente conectadas quando não se utiliza o *Dropout*, com a utilização do *Dropout* e com a utilização do *DropConnect*.

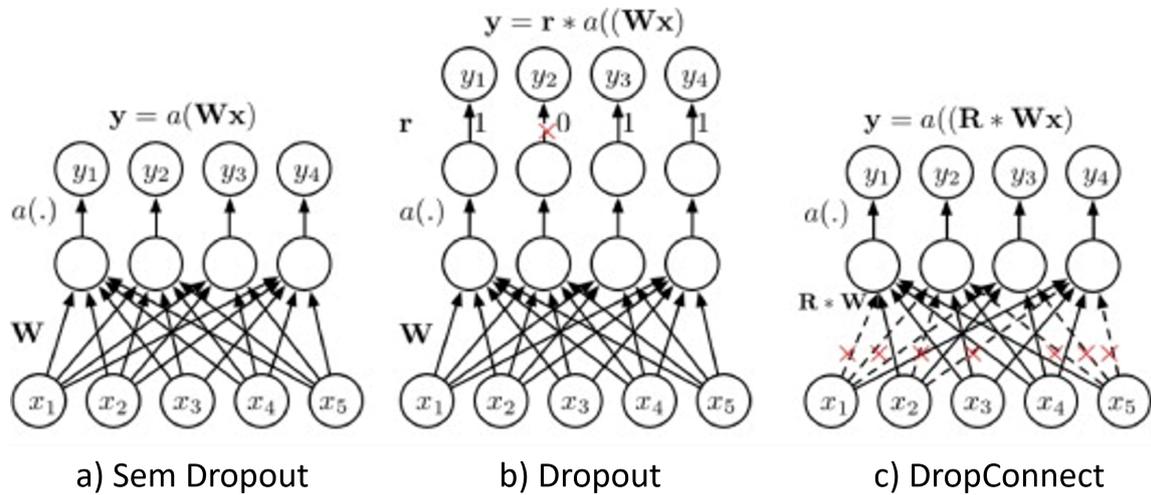


Figura 2.9: Estratégias de Regularização: a) Sem Dropout; b) Dropout e c) DropConnect. Adaptado de Gu *et al.* (2018)

2.6.4 Gradiente Descendente Estocástico

O Gradiente Descendente é um algoritmo de otimização que realiza o ajuste de parâmetros de forma iterativa com o objetivo de encontrar valores que minimizam a função de interesse (WIJNHOVEN; WITH, 2010). O algoritmo Gradiente Descendente padrão atualiza os parâmetros ϕ da função de interesse $\mathcal{G}(\phi)$ como $\phi_{t+1} = \phi_t - \eta \nabla_{\phi} E[\mathcal{G}(\phi_t)]$, onde $E[\mathcal{G}(\phi_t)]$ é a expectativa de $\mathcal{G}(\phi)$ ao longo de todo o conjunto de treinamento e η é a taxa de aprendizagem. Já o algoritmo Gradiente Descendente Estocástico (*Stochastic Gradient Descent - SGD*) proposto por Wijnhoven e With (2010), em vez de calcular $E[\mathcal{G}(\phi_t)]$, como seu antecessor, estima os gradientes com base em um único exemplo escolhido aleatoriamente $(x^{(t)}, y^{(t)})$ do conjunto de treinamento, que pode ser expresso como:

$$\phi_{t+1} = \phi_t - \eta_t \nabla_{\phi} \mathcal{G}(\phi_t, x^{(t)}, y^{(t)}) \quad (2.11)$$

Na prática, cada atualização de parâmetro no SGD é calculada em relação a um mini-lote em oposição a um único exemplo. Isso pode ajudar a reduzir a variação na atualização do parâmetro e levar a uma convergência mais estável. A velocidade de convergência é controlada pela taxa de aprendizagem η_t (HINTON *et al.*, 2012). Entretanto, o mini-lote SGD traz com seus benefícios desafios a serem superados. Um dos mais relevantes é quanto à escolha de uma taxa de aprendizagem adequada. Um método comum é usar uma taxa de aprendizado constante, que dá convergência estável no estágio inicial e, então, reduzi-la à medida que se diminui a velocidade de convergência do treinamento (WIJNHOVEN; WITH, 2010). Na Figura 2.10 é apresentado um comparativo de funcionamento desses dois algoritmos.

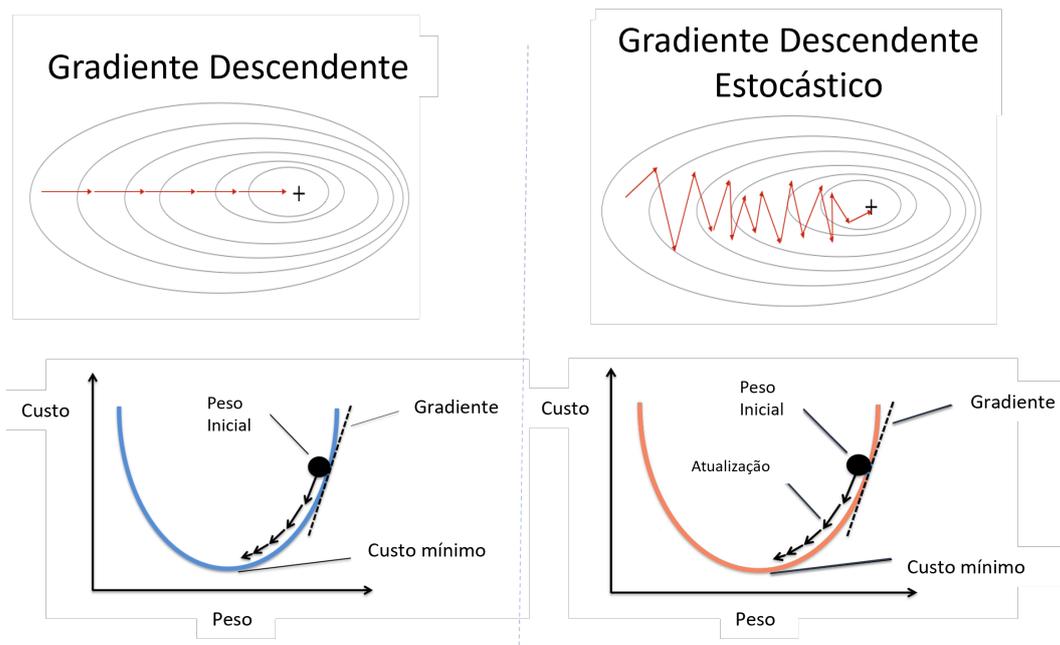


Figura 2.10: Comparativo entre os Algoritmos de otimização. Adaptado de Goodfellow *et al.* (2016)

2.7 Evolução das Arquiteturas

Diferentes inovações nas arquiteturas CNNs foram propostas desde 1998 com a apresentação da LeNet-5 (LECUN *et al.*, 1998) e podem ser categorizadas como otimização de parâmetros, regularização, reformulação estrutural, etc. Entretanto, observa-se que a principal melhoria no desempenho das CNNs é motivada pela reformulação estrutural e projeto de novos blocos (KHAN *et al.*, 2020). De acordo com Khan *et al.* (2020), dependendo do tipo de modificações arquitetônicas, as CNNs podem ser categorizadas em sete classes diferentes, sendo elas: exploração espacial, profundidade, múltiplos caminhos, exploração do mapa de características, aumento de canais de entrada e as CNNs baseadas em atenção. Na Figura 2.11 é apresentada uma taxonomia dessas categorias conforme a perspectiva de evolução dessas arquiteturas.

Seguindo a taxonomia das categorias propostas por Khan *et al.* (2020), este estudo descreve as arquiteturas de maior relevância, tendo em face o contexto de aplicação e o domínio de problema investigado. Na Figura 2.11 é apresentado a taxonomia das categorias de arquiteturas CNNs.

LeNet-5

A LeNet-5 proposta por Lecun *et al.* (1998) é uma das arquiteturas precursoras mais simples das redes convolucionais. Foi projetada para o reconhecimento de caracteres manuscritos e impressos à máquina e era estruturada com 2 camadas convolucionais e 3 camadas totalmente conectadas. Sua arquitetura possui cerca de 60.000 parâmetros.

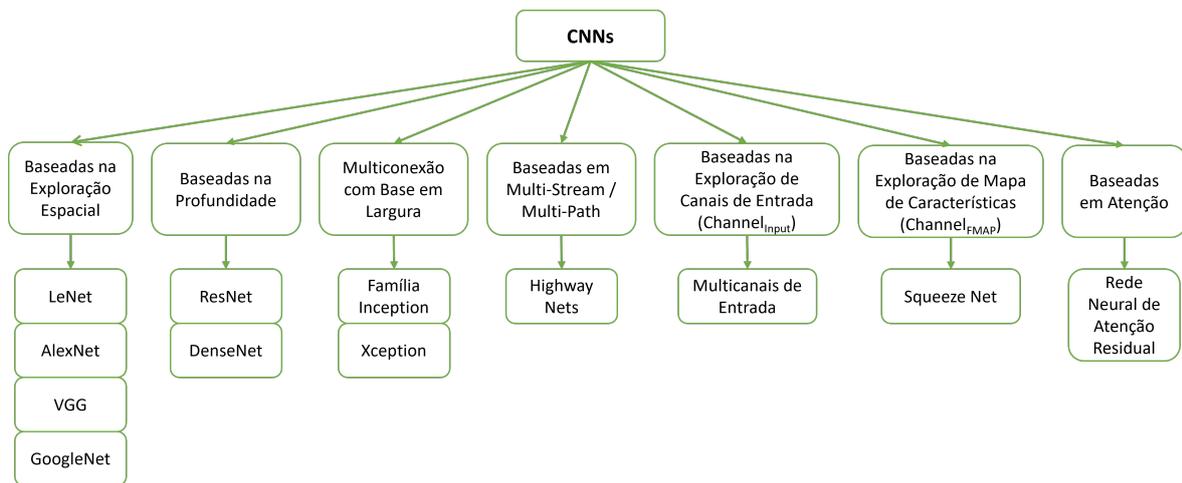


Figura 2.11: Taxonomia evolutiva das arquiteturas CNNs com sete categorias diferentes. Adaptado de Khan *et al.* (2020)

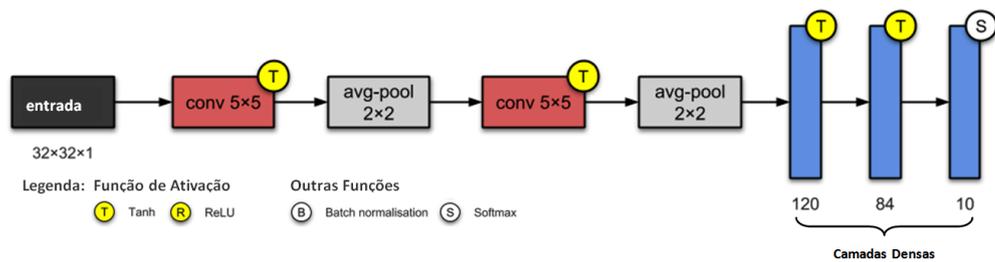


Figura 2.12: Arquitetura LeNet-5. Adaptado de Karim (2019)

A LeNet-5 tornou-se um modelo de referência para o encadeamento das camadas de convolução e pooling, juntamente com a finalização dos modelos com uma ou mais camadas totalmente conectadas.

AlexNet

A AlexNet, proposta por Krizhevsky *et al.* (2012), foi uma arquitetura de grande destaque no cenário de reconhecimento de imagens utilizando o conjunto de dados da ImageNet (DENG *et al.*, 2009).

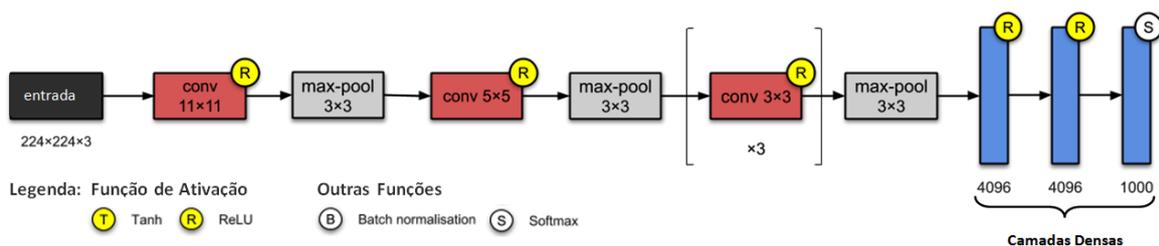


Figura 2.13: Arquitetura AlexNet. - Adaptado de Karim (2019)

A arquitetura é composta por 8 camadas, sendo 5 camadas convolucionais e 3 camadas totalmente conectadas. Na Figura 2.13 são apresentados os detalhes relevantes da arquitetura.

A AlexNet tornou evidente à comunidade científica os ganhos em desempenho das implementações de CNN usando unidades gráficas de processamento (*Graphical Processing Unit - GPU*). Adicionalmente, a arquitetura implementa o método ‘*Max-Pooling*’ (ZHOU *et al.*, 1988; NAGI *et al.*, 2011) após cada camada convolucional e utiliza a *ReLU* (NAIR; HINTON, 2010) como função de ativação. Essas inovações demonstraram melhor desempenho no processo de classificação de imagens.

VGGNet

A introdução do conceito de visualização das características por camadas de convolução apresentado por Zeiler e Fergus (2014) foi um novo aporte na evolução das arquiteturas CNNs, sendo determinante para extração de características em baixa resolução espacial.

A VGGNet (*Visual Geometry Group*) proposta por Simonyan e Zisserman (2014) explora essa evolução e implementa uma arquitetura com 13 camadas convolucionais com pequenos *kernels* de (3×3) , filtros (2×2) nas camadas de subamostragem com *MaxPooling* e 3 camadas totalmente conectadas. A ReLU, a exemplo da AlexNet, é utilizada como função de ativação.

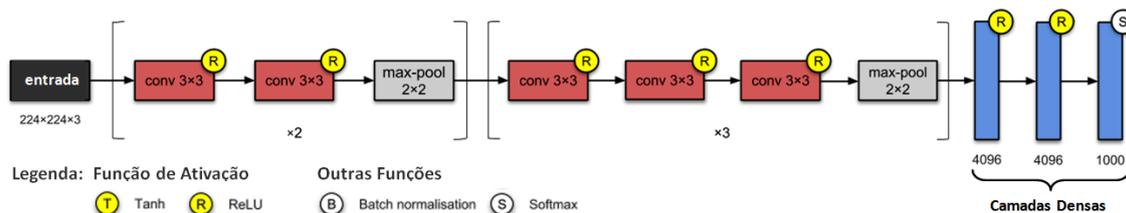


Figura 2.14: Arquitetura VGGNet. Adaptado de Karim (2019)

A arquitetura da VGG procura explorar a extração de características em baixa resolução espacial aumentando o número de camadas (aproximadamente duas vezes mais profunda que a AlexNet). Entretanto, a solução impõe um alto custo computacional devido aos seus 138 milhões de parâmetros, com um modelo com cerca de 500 Megabytes de espaço de armazenamento.

GoogLeNet - InceptionV3

Posteriormente, a Google introduziu uma nova abordagem com a ideia de dividir, transformar e mesclar, denominada *Inception block*. A arquitetura proposta pela Google consolidou o conceito de ramificação dentro de uma camada, permitindo a abstração de características em diferentes escalas espaciais (SZEGEDY *et al.*, 2015a).

Na Figura 2.15 é apresentada a primeira versão da Arquitetura denominada GoogLeNet ou Inception V1. Essa CNN possui 9 blocos denominados *Inception* empilhados linearmente, perfazendo um total de 22 camadas. Nota-se também que ao final a arquitetura utiliza um ‘*Average-Pooling*’ antes das camadas totalmente conectadas.

Com um total de 6.797 milhões de parâmetros, a principal característica dessa arquitetura é otimizar a utilização dos recursos computacionais dentro da rede. Uma das estratégias com

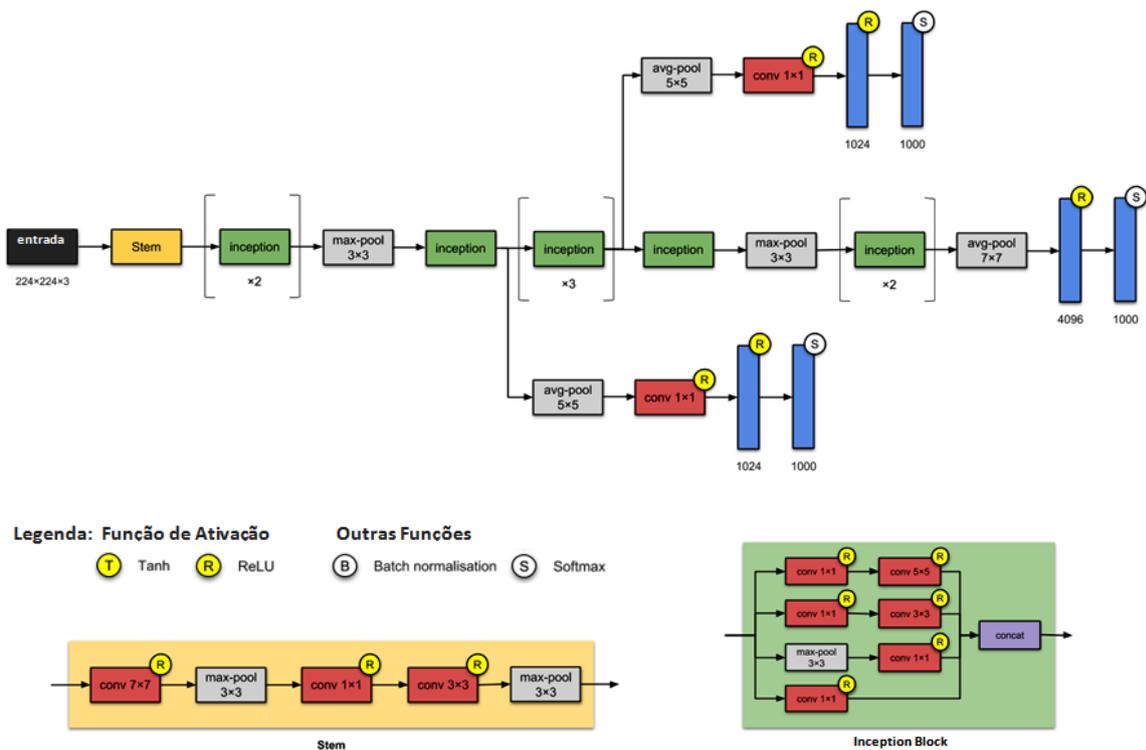


Figura 2.15: Arquitetura GoogLeNet - InceptionV1. Esta CNN possui duas redes auxiliares com classificadores que são descartadas no momento da inferência. Adaptado de Karim (2019)

destaque para esse propósito é a utilização de convoluções (1×1) , utilizadas para reduzir a dimensionalidade e remover gargalos computacionais.

Os exitosos resultados desta proposta deram origem a uma versão amplamente melhorada denominada *InceptionV3* proposta por Szegedy *et al.* (2015b). Nesta nova versão da arquitetura, com 24 milhões de parâmetros, a intuição era que o melhor desempenho fosse alcançado quando as convoluções não alterassem drasticamente as dimensões da entrada. Ao reduzir muito essas dimensões, era inevitável a perda de informações, denominado ‘gargalo representacional’.

Deste modo, a implementação de métodos de fatoração inteligente possibilitou que as convoluções fossem decompostas com *kernels* menores e maior eficiência em termos de complexidade computacional.

ResNet

Ainda em 2015 surge a abordagem de aprendizagem residual proposto pela Microsoft e implementado por meio da ResNet (HE *et al.*, 2016), tornando-se o conceito amplamente utilizado por diferentes projetos arquitetônicos como, por exemplo, a Inception-ResNet (SZEGEDY *et al.*, 2017b).

Criar arquiteturas cada vez mais profundas em busca de um melhor desempenho mostrou-se uma estratégia desafiadora para os projetistas. Aumentando a profundidade da rede, seu desempenho fica saturado ou começa a degradar rapidamente, causando a perda da capacidade de generalização do modelo (HE *et al.*, 2016).

Um dos problemas que a ResNet propõe resolver é o desaparecimento (dissipação) do gradiente ‘*Vanish Gradient*’ (KOLEN; KREMER, 2001). Em arquiteturas muito profundas, os gradientes de onde a função de perda é calculada diminuem drasticamente para zero; isso ocorre após várias aplicações da regra de cadeia. Com esses resultados nos pesos não há atualização de valores e, conseqüentemente, nenhum aprendizado.

A ideia central da abordagem proposta pela arquitetura ResNet é a introdução da chamada ‘conexão de atalho de identidade’, que pula uma ou mais camadas da arquitetura como ilustrado por meio da Figura 2.16. Esse mapeamento de identidade não possui parâmetros e serve apenas para adicionar a saída da camada anterior à camada da frente. A ResNet mudou significativamente a visão de como parametrizar as funções em redes convolucionais profundas (HE *et al.*, 2016).

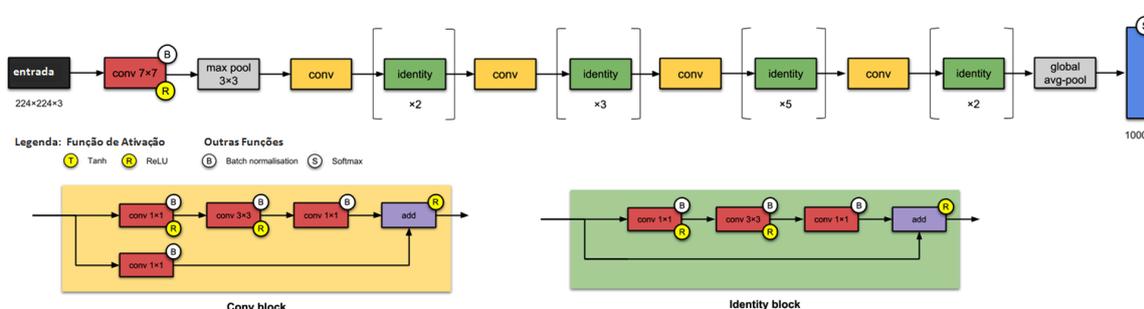


Figura 2.16: Arquitetura ResNet. Adaptado de Karim (2019)

A DenseNet (HUANG *et al.*, 2017), até certo ponto, é a extensão lógica dessa evolução. A principal diferença entre a ResNet e a DenseNet é que as saídas de cada camada na arquitetura DenseNet são concatenadas, em vez de adicionadas, como na ResNet.

Outras arquiteturas, como a SqueezeNet, proposta por Iandola *et al.* (2016), e a MobileNet, proposta por Howard *et al.* (2017), buscam a redução do número de parâmetros e a otimização do tamanho dos modelos visando a adequação e usabilidade aos dispositivos móveis.

2.8 Arquiteturas com Topologia de Múltiplos Caminhos

As arquiteturas de caminho único correspondem aos casos nos quais há um fluxo único de informações, ou seja, os dados de entrada são processados por camadas convolucionais, *pooling* e retificadoras não lineares. Em seguida, os mapas de características são extraídos nas camadas totalmente conectadas e, posteriormente, usados para prever o rótulo na camada de saída.

Ao contrário das redes de caminho único, as arquiteturas com topologia de múltiplos caminhos são um tipo de CNN em que redes operativas independentes são integradas em um único modelo para capturar um conjunto mais variado de características (KHAN *et al.*, 2018). Os módulos podem ser dispostos em paralelo ou em série (cascata), tornando essa arquitetura uma topologia passível de novos arranjos (KARPATY *et al.*, 2014; SRIVASTAVA *et al.*, 2015; LARSSON *et al.*, 2016; KUEN *et al.*, 2018). Esse conceito foi explorado anteriormente por Bromley *et al.* (1993) e Taigman *et al.* (2014) denominando os modelos com estas características como redes siamesas, cujo objetivo era medir a similaridade ou dissimilaridade entre imagens. Karpathy *et al.* (2014) também apresenta uma proposta denominada multiresolução (*multiresolution*), elaborando uma avaliação empírica das CNNs sobre a classificação de vídeos, tendo dois fluxos separados de processamento em duas resoluções espaciais (*multistream*).

Atualmente, estudos como de Khan *et al.* (2018), Chen *et al.* (2019a), Thomas *et al.* (2021), Lin *et al.* (2021) e Wang *et al.* (2021) definem as arquiteturas CNNs que utilizam essa topologia como redes convolucionais de múltiplos caminhos (*multipath*).

Entretanto, ainda que exista uma possibilidade razoável de variações topológicas para esse tipo de abordagem, pode haver apenas duas variantes possíveis quanto ao tipo de arquitetura. A primeira variante agrega as características extraídas de redes separadas antes de fazer as previsões, o que pode ser entendido como um processo de fusão de informações. Esta fusão pode ocorrer em dois níveis: i) no primeiro nível de dados, onde os mapas de características são mescladas para formar um mapa de características maior; ii) no segundo nível de características, onde as características finais extraídas são concatenadas antes de fazer a predição (BERNAL *et al.*, 2019).

Na Figura 2.17 é apresentada uma ilustração de arquiteturas de múltiplos caminhos com topologia em paralelo e em série, na qual diferentes recursos são processados usando redes separadas que são concatenadas antes das previsões. Essa concatenação também pode ser realizada nas camadas convolucionais, onde os mapas de características de diferentes modelos ou blocos formam um mapa de características maior.

A segunda variante de arquiteturas de múltiplos caminhos consiste em aprender conjuntos de mais de duas redes para melhorar a previsão. As previsões feitas por cada módulo para um determinado *voxel* de destino são mesclados probabilisticamente (BERNAL *et al.*, 2019).

O treinamento de CNNs sofre com degradação de desempenho, desaparecimento de gradiente ou problemas de explosão, que não são causados por *overfitting*, e sim pelo aumento da profundidade das arquiteturas (HOCHREITER, 1998; DONG *et al.*, 2016). As arquiteturas com topologia de múltiplos caminhos podem conectar sistematicamente uma camada a outra, pulando algumas camadas intermediárias para permitir o fluxo especializado de informações entre camadas (TONG *et al.*, 2017). Essa topologia também tenta resolver o problema do desaparecimento do gradiente, tornando-o acessível às camadas inferiores.

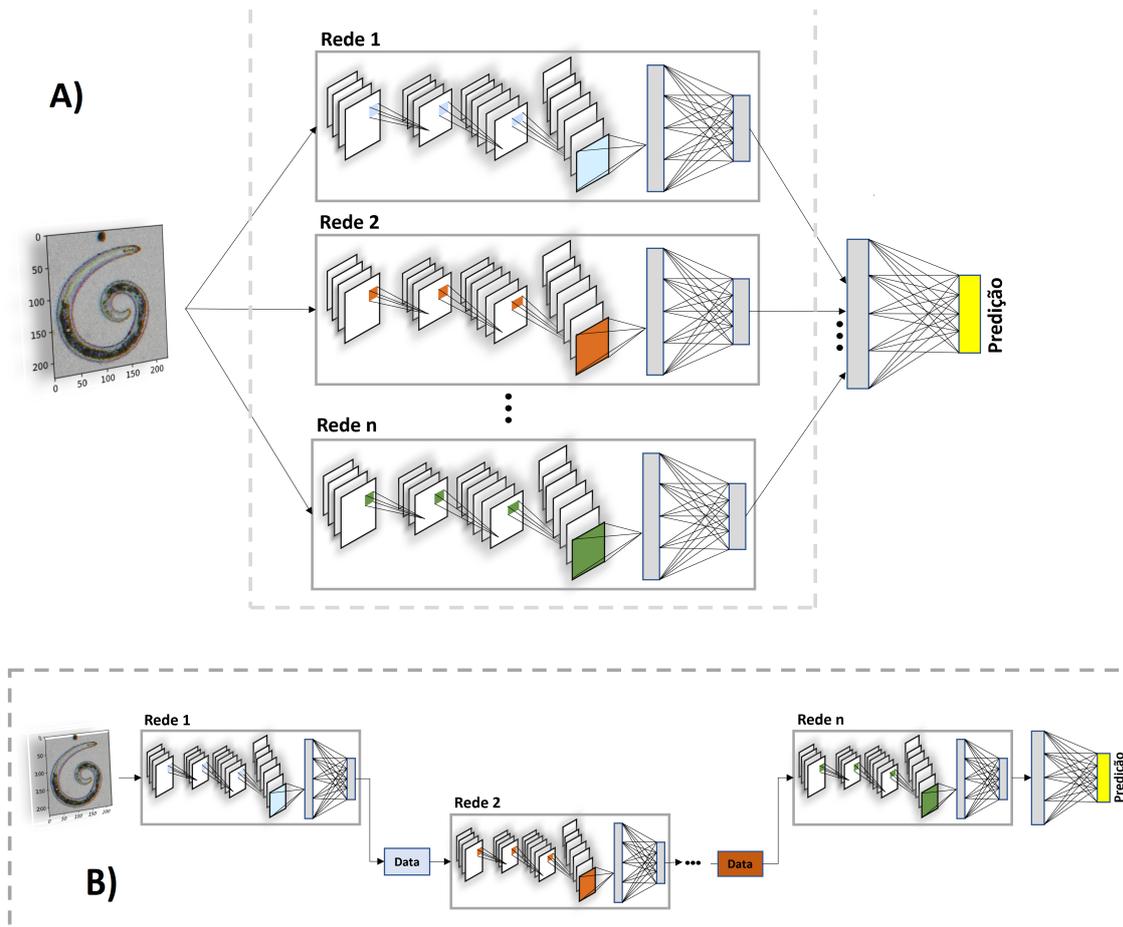


Figura 2.17: Arquitetura CNNs com topologia de múltiplos caminhos. O diagrama A) representa uma topologia paralela CNN de múltiplos caminhos: diferentes CNNs executam paralelamente a extração de características, concatenando esse resultado para prever o rótulo. Já o diagrama B) corresponde a uma topologia em cascata (série), na qual uma rede é fornecida como entrada para outra tendo como pressuposto a lapidação ou refinamento do resultado.

2.9 Transferência de Aprendizagem

Sob a perspectiva do aprendizado supervisionado, as CNNs vêm sendo amplamente estudadas e aplicadas como solução para vários problemas do mundo real. Entretanto, o desempenho dessas redes depende estritamente da qualidade do conjunto de dados de treinamento, capaz de compor um modelo apto a fazer previsões sobre os dados futuros (HASTIE *et al.*, 2009b; MITCHELL, 1997; AGGARWAL, 2014; SUTHAHARAN, 2015).

A diversidade de problemas e as especificidades dos cenários do mundo real dificultam a catalogação semântica dos dados em conjuntos representativos com uma quantidade suficiente de amostras rotuladas. Esse problema torna-se um gargalo relevante e desafiador para tornar os métodos de aprendizado de máquina mais aplicáveis na prática (AGGARWAL, 2014).

Estudos propostos por Zhu (2005), Chapelle *et al.* (2010), Engelen e Hoos (2019) reportam propostas para resolver o problema da escassez de dados rotulados, por meio das técnicas de

aprendizado semi-supervisionado. As abordagens propõem a utilização de grandes conjuntos de dados não rotulados para descobrir uma estrutura de dados intrínseca para propagar efetivamente as informações do rótulo. Entretanto, os bons resultados alcançados são limitados ao domínio de interesse, ou seja, representam um mesmo espaço de características discriminantes.

Em situações em que os dados não rotulados são abundantes, face o alto custo de rotulação manual, técnicas de aprendizado supervisionado iterativo denominada *aprendizado ativo* constituem uma alternativa (TUIA *et al.*, 2011). Essas abordagens fazem uso de um algoritmo capaz de consultar interativamente um oráculo para obter os resultados desejados, prospectando maior precisão com menos rótulos de treinamento. Entretanto, a limitação do oráculo pode gerar um conjunto rotulado não discriminante a outros domínios de interesse (TONG; CHANG, 2001; SETTLES, 2009).

A Transferência de Aprendizagem indutiva empregada nas redes CNNs, por outro lado, é um método que reutiliza modelos aplicados a determinadas tarefas como ponto de partida para um modelo relacionado a um novo domínio de interesse. Assim, o objetivo é emprestar dados rotulados ou extrair conhecimento de alguns domínios relacionados para obter o maior desempenho possível no domínio de interesse (PAN; YANG, 2010; SHAO *et al.*, 2015).

Nesta perspectiva, Pan e Yang (2010) definiram que para um domínio de origem (D_S) e uma tarefa de aprendizagem (T_S), um domínio de destino (D_T) e uma tarefa de aprendizagem (T_T), a transferência de aprendizado visa ajudar a melhorar o aprendizado da função preditiva de destino $f_T(\cdot)$ no D_T usando o conhecimento em D_S e T_S , $D_S \neq D_T$ ou $T_S \neq T_T$. Adicionalmente, destaca-se que em algumas situações nas quais o domínio de origem e o domínio de destino não estejam de alguma forma relacionados, a transferência de aprendizagem não é exitosa e certamente causará prejuízos ao desempenho no domínio alvo. Na Figura 2.18 é apresentada uma ilustração conceitual da transferência de aprendizagem.

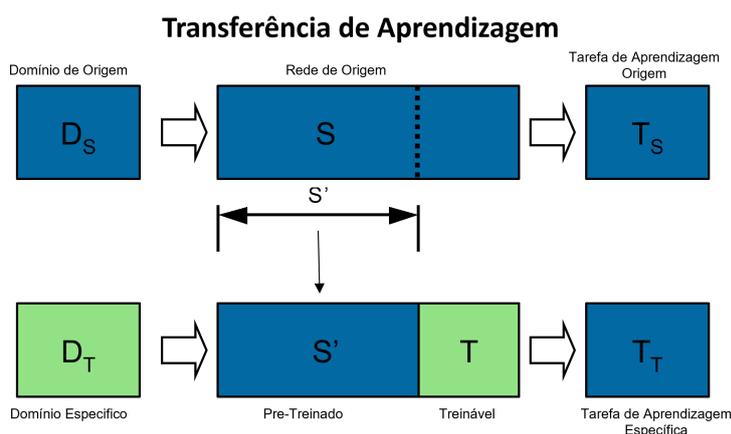


Figura 2.18: Transferência de Aprendizagem. Adaptado de Pan e Yang (2010)

2.10 Ajuste Fino e Hiperparâmetros

Um dos desafios ao projetar e modelar uma CNN é determinar a melhor estrutura e o melhor conjunto de parâmetros essenciais à aprendizagem de acordo com o contexto de aplicação. Geralmente, esses parâmetros são escolhidos com base em regras heurísticas e são ajustados para produzir o melhor desempenho preditor com relação ao conjunto de dados disponíveis. Pode-se exemplificar, como parâmetros, o número de camadas e nós da rede, as taxas de aprendizado (*learning rate*) e as taxas de desistência (*dropout rate*) (TOO *et al.*, 2018).

O ajuste fino (*Fine-tuning*) é um método decorrente do conceito da transferência de aprendizagem. Quando são utilizados modelos de CNNs pré-treinados em grandes conjuntos de dados, como a ImageNet (DENG *et al.*, 2009), alguns ajustes são necessários para adequação de contexto entre o domínio de problemas mais amplo e o mais específico.

Deste modo, algumas técnicas são implementadas como diretrizes deste ajuste fino. Entre elas, a adequação do classificador *softmax* na última camada relevante ao domínio do problema específico, a diminuição da taxa de aprendizagem para evitar distorções na capacidade de generalização do modelo e o congelamento dos pesos das primeiras camadas da rede pré-treinada que capturam recursos universais.

Já um hiperparâmetro pode ser qualificado como um parâmetro cujo valor é definido antes do início do processo de aprendizado e não pode ser treinado diretamente a partir dos dados. Enquanto os parâmetros do modelo são aprendidos durante o treinamento quando otimizamos uma função de perda usando algo como a descida de gradiente. Deste modo, enquanto os parâmetros do modelo especificam como transformar os dados de entrada na saída desejada, os hiperparâmetros definem como nosso modelo é realmente estruturado (DIAZ *et al.*, 2017). Os hiperparâmetros comuns incluem o número de camadas ocultas, o número de iterações de treinamento (iterações e épocas), esquemas de inicialização de pesos, regularização e as funções de ativação.

A maioria dos algoritmos de aprendizado profundo possui hiperparâmetros que controlam muitos aspectos do comportamento do algoritmo. Alguns desses hiperparâmetros afetam o tempo de execução e o consumo de memória do algoritmo. Consequentemente, esses hiperparâmetros afetam a qualidade do modelo recuperado pelo processo de treinamento e sua capacidade de inferir resultados corretos quando implantados com novas entradas.

Existem duas abordagens básicas para ajustar os hiperparâmetros, seja por meio do ajuste manual ou automático. Ajustar os hiperparâmetros manualmente implica em entender minuciosamente o que os hiperparâmetros fazem e como os modelos de aprendizado de máquina alcançam uma boa generalização. Enquanto os algoritmos com seleção automática de hiperparâmetros reduzem muito a necessidade de compreensão e consequência de cada ajuste; entretanto, essas abordagens são as mais dispendiosas computacionalmente falando.

Ajuste Manual dos Hiperparâmetros

Para ajustar os hiperparâmetros manualmente é preciso compreender a relação entre os hi-

parâmetros, o erro de treinamento, o erro de generalização e os recursos computacionais disponíveis.

O objetivo principal do ajuste manual de hiperparâmetros é adequar a capacidade efetiva do modelo para corresponder à complexidade da tarefa. Essa capacidade efetiva é restrita por três fatores: i) a capacidade representacional do modelo; ii) a capacidade do algoritmo de aprendizado de minimizar com êxito a função de custo utilizada para treinar o modelo; e iii) o grau em que a função de custo e o procedimento de treinamento regularizam o modelo (GOODFELLOW *et al.*, 2016).

A abordagem de ajuste manual requer o monitoramento, tanto do treinamento quanto do erro de teste, para diagnosticar se o modelo está com excesso ou não de ajuste. O conhecimento sobre o impacto de cada hiperparâmetro no desempenho do modelo é adquirido empiricamente, ou seja, as experiências pautadas nas observações e monitoramento são determinantes ao correto ajuste.

Na Tabela 2.1 são apresentados os hiperparâmetros de maior relevância quando adotada a abordagem de ajuste manual.

Tabela 2.1: Ações e Reações do Ajuste dos Hiperparâmetros quanto a Capacidade do Modelo.

Hiperparâmetro	Vantagens	Desvantagens
Número de Camadas Ocultas	Aumentar o número de unidades ocultas incrementa a capacidade representacional do modelo	Consequentemente o aumento das unidades ocultas incrementa o tempo de execução e o custo da memória de praticamente todas as operações do modelo
Taxa de Aprendizagem	A taxa de aprendizado certamente é um dos hiperparâmetros mais importante. Ela controla a capacidade efetiva do modelo e deve prover um valor que satisfaça o problema de otimização quanto ao aprendizado do algoritmo.	Uma taxa de aprendizado inadequada, seja muito alta ou muito baixa, resulta em um modelo com baixa capacidade devido as falhas de otimização.
Largura do Kernel da Convolução	Aumentar a largura do kernel incrementa o número de parâmetros no modelo	Um kernel maior resulta em uma dimensão de saída mais estreita, reduzindo a capacidade do modelo. Kernel maiores exigem mais memória para armazenamento de parâmetros e aumentam o tempo de execução do algoritmo.
Preenchimento Zero Implícito (Zero Padding)	A adição de zeros implícitos antes da convolução mantém o tamanho da representação	Aumenta o tempo e custo de memória da maioria das operações.
Coefficiente de redução de Peso	A adição de uma penalidade de tamanho de peso ou regularização de peso a uma CNN tem o efeito de reduzir o erro de generalização e de permitir que o modelo preste menos atenção a variáveis de entrada com menor relevância.	Um modelo com pesos grandes é mais complexo que um modelo com pesos menores. Pesos grandes é um sinal de uma rede que pode ser excessivamente especializada em dados de treinamento com baixa capacidade de generalização.
Taxa de Abandono (Dropout rate)	O descarte de unidades com menor frequência oferece às unidades mais oportunidades de planejar umas com as outras o devido ajuste ao conjunto de treinamento.	Entretanto, essa estratégia de regularização deve atentar para sensibilidade da rede aos pesos específicos das unidades. Quanto menos sensível for a rede a estes pesos específicos, melhor a capacidade de generalização com menor probabilidade de superestimar os dados de treinamento.

Ajuste Automático dos Hiperparâmetros

A abordagem de ajuste manual dos hiperparâmetros pode ser exitosa; entretanto, a dependência do *know-how* do usuário, o tipo do problema a ser otimizado, a especificidade do conjunto de dados e as dificuldades de formalização do conhecimento empírico sobre o correto ajuste podem causar sérios transtornos à eficiência e eficácia do modelo (CETINIC *et al.*, 2018).

Os algoritmos de otimização que executam os ajustes automaticamente nos modelos de aprendizagem, adequando os hiperparâmetros de acordo com o conjunto de dados e as demais peculiaridades do modelo, constituem uma das abordagens de grande contribuição ao sucesso das CNNs. Basicamente, esta abordagem implementa 3 métodos de ajuste automatizado, sendo eles:

- *Pesquisa em Grade - Grid Search* → Para cada hiperparâmetro é selecionado um conjunto finito de valores que podem ser explorados. O algoritmo treina um modelo para cada especificação conjunta de valores de hiperparâmetro, dentro do produto cartesiano

do conjunto de valores para cada hiperparâmetro individual. O experimento que gerar o menor erro do conjunto de validação é escolhido como melhor arranjo de hiperparâmetros (BERGSTRA; BENGIO, 2012).

Um problema óbvio com este método é o custo computacional, pois, quanto maior o conjunto de valores que compõem a grade pesquisa, maior o número de ensaios de treinamento. Assim, o custo de execução do método é exponencial $O(n^m)$.

- *Pesquisa Aleatória - Random Search* → Neste método, é definida uma distribuição marginal, ou seja, uma distribuição uniforme em escala logarítmica para os hiperparâmetros com valores positivos. Diferente da pesquisa em grade, os valores não são discretizados, visando explorar um conjunto maior de possibilidades, evitando custos computacionais adicionais.

Os estudos de Bergstra e Bengio (2012) demonstram que a pesquisa aleatória reduz o erro do conjunto de validação muito mais rapidamente que a pesquisa em grade, pois alcança um arranjo de hiperparâmetros otimizado com número bem menor de ensaios executados.

- *Otimização de Hiperparâmetro com base em Modelos* → A busca por bons hiperparâmetros pode ser considerada um problema de otimização. As variáveis de decisão são os hiperparâmetros. O custo a ser otimizado é o erro do conjunto de validação resultante do treinamento usando esses hiperparâmetros. A maioria dos algoritmos baseados em modelo para pesquisa de hiperparâmetro usa um modelo de regressão bayesiano para estimar o valor esperado do erro do conjunto de validação para cada hiperparâmetro e há uma incerteza em torno dessa expectativa (SNOEK *et al.*, 2012b).

Entretanto, não é possível recomendar a otimização de hiperparâmetros Bayesiana como uma ferramenta plenamente estabelecida. Seus resultados por vezes são animadores em determinadas circunstâncias, mas podem falhar drasticamente em outros tipos de cenários (SNOEK *et al.*, 2012b).

2.11 Aumento de Dados

Quando o termo Generalização é utilizado no aprendizado de máquina, o conceito é definido pela diferença de desempenho de um modelo quando avaliado com dados vistos anteriormente (dados de treinamento) versus dados nunca vistos (dados de teste). Modelos com baixa capacidade de generalização se ajustam muito bem ao conjunto de dados de treinamento, entretanto, são ineficazes quando submetidos à classificação de conjuntos de dados de teste (KOBAYASHI *et al.*, 2018). O problema de super adaptação dos modelos ao conjunto de treinamento é conhecido como *Overfitting* (HAWKINS, 2004).

Muitas abordagens com diferentes estratégias buscam aumentar o desempenho dos modelos prospectando arquiteturas capazes de aumentar a capacidade de generalização em detrimento ao

tamanho do conjunto de dados. Essa iniciativa deu origem a uma sequência de arquiteturas progressivamente mais complexas, sendo elas: AlexNet (KRIZHEVSKY *et al.*, 2012), VGG (SIMONYAN; ZISSERMAN, 2014), GoogLeNet (SZEGEDY *et al.*, 2015a), InceptionV3 (SZEGEDY *et al.*, 2015b), ResNet (HE *et al.*, 2016) e a DenseNet (HUANG *et al.*, 2017).

Aliadas à evolução das arquiteturas CNN, soluções funcionais como regularização por abandono (*dropout regularization*), normalização de lotes (*batch normalization*), transferência de aprendizado (*transfer learning*) e pré-treinamento (*pretraining*) foram implementadas, buscando viabilizar o uso do aprendizado profundo em conjuntos de dados com baixo número de amostras.

Um dos métodos para constatar o *overfitting* é traçar a precisão de treinamento e validação em cada época durante o treinamento. Para consolidar um modelo como útil, a auditoria deve constatar que o erro de validação diminui juntamente com o erro de treinamento.

Uma das abordagens exitosas que buscam mitigar o problema da capacidade de generalização dos modelos é o aumento de dados (*Data Augmentation*). Nessa abordagem, o conjunto de dados original é aumentado por diversas técnicas, buscando uma representação mais abrangente possível do domínio do problema, minimizando a distância entre o conjunto de treinamento e validação (SHORTEN; KHOSHGOFTAAR, 2019).

A abordagem de aumento de dados é uma estratégia cujo objetivo é prover uma quantidade proporcional à qualidade de amostras presentes no conjunto de dados. Assim, sob a perspectiva de um conjunto com número de amostras condizentes às necessidades de representação e o devido balanceamento das categorias (classes) é possível que mais características discriminantes sejam extraídas do conjunto de dados após os devidos aprimoramentos (MIKOAJCZYK; GROCHOWSKI, 2018).

Basicamente, o aumento de dados utiliza os métodos de distorção (*Data Warping*) ou super-amostragem (*Oversampling*), que incrementam artificialmente o tamanho do conjunto de dados de treinamento. A distorção de dados transforma as imagens existentes de modo que o seu rótulo seja preservado, utilizando técnicas como transformações geométricas, espaço de cores, cortes de imagens, rotação, translação, adição de ruído, filtros do kernel e apagamento aleatório (MIKOAJCZYK; GROCHOWSKI, 2018; SHORTEN; KHOSHGOFTAAR, 2019).

O método de super-amostragem cria instâncias sintéticas das imagens originais e as adiciona ao conjunto de treinamento. Para isso, são utilizadas técnicas para embaralhamento e sobreposição de imagens, aumento do espaço de características e redes geradoras adversárias (*Generative Adversarial Networks - GANs*). Os métodos de (*Data Warping*) e (*Oversampling*) não formam uma dicotomia mutuamente exclusiva, assim podem ser utilizados de modo complementar.

2.12 Considerações Finais

Neste capítulo foram apresentados os principais conceitos relativos ao contexto no qual o trabalho está inserido. O recorte conceitual elaborado começa conceituando doenças de plantas na Seção 2.1 e tipos de doenças com seus principais agentes causadores na Seção 2.2. Na Seção 2.4 são discutidos os principais fundamentos sobre a análise de imagens microscópicas sobre a perspectiva de processamento automatizado dessas imagens.

Na Seção 2.5 são apresentados os fundamentos teóricos sobre o aprendizado de máquina e da Seção 2.6 até a Seção 2.11 são abordados os conceitos e técnicas inerentes às redes neurais convolucionais.

No próximo capítulo, denominado Metodologia, serão apresentados os procedimentos realizados durante esta pesquisa.

Capítulo 3

Metodologia

Neste Capítulo serão apresentados os procedimentos e métodos de pesquisa utilizados durante o processo de investigação, para analisar e avaliar as diferentes técnicas e abordagens utilizando CNNs para classificação de doenças de plantas e identificação de patógenos. As observações realizadas possibilitarão customizar, otimizar e propor modelos e arquiteturas CNNs adequados ao domínio do problema investigado.

3.1 Descrição Inicial do Problema Abordado

Trata-se de uma pesquisa de natureza aplicada, de caráter multidisciplinar, com objetivos exploratórios incrementais quanto à formulação e validação de hipóteses, prospectando soluções quanto à classificação de doenças de plantas e a identificação de patógenos utilizando as redes neurais convolucionais.

De acordo com Lakatos *et al.* (2007), a pesquisa exploratória é uma investigação empírica, cujo objetivo é a formulação de questões ou de um problema, com tripla finalidade: 1) desenvolver e validar hipóteses; 2) aumentar a familiaridade com um ambiente, fato ou fenômeno investigado; 3) modificar e clarificar conceitos. Empregam-se geralmente procedimentos sistemáticos ou para obtenção de observações empíricas ou para as análises de dados (ou ambas, simultaneamente). Deste modo, obtêm-se frequentemente descrições tanto quantitativas quanto qualitativas do objeto de estudo, possibilitando ao investigador estabelecer padrões e conceituar as inter-relações entre as propriedades do fenômeno, fato ou ambiente observado.

Nesta perspectiva, foram utilizados critérios quantitativos para mensurar, classificar e interpretar procedimentos técnicos que envolvem desde a seleção de modelos, arquiteturas, topologias, características do conjunto de imagens e demais variáveis capazes de influenciar a eficiência do método. Adicionalmente, buscou-se prospectar customizações e otimizações, por meio das observações realizadas junto ao efeito deste conjunto de fatores nos resultados obtidos no processo de classificação de doenças de plantas e identificação de patógenos usando CNNs.

3.2 Estrutura da Metodologia Proposta

A abordagem deste trabalho, que perpassa desde as observações empíricas, a caracterização do problema até as contribuições de pesquisa, foi estruturada de modo a atender cada objetivo específico e subsidiar afirmativamente as hipóteses levantadas. Na Figura 3.1 a estrutura é ilustrada com detalhes, seguindo um fluxo lógico da execução da pesquisa.

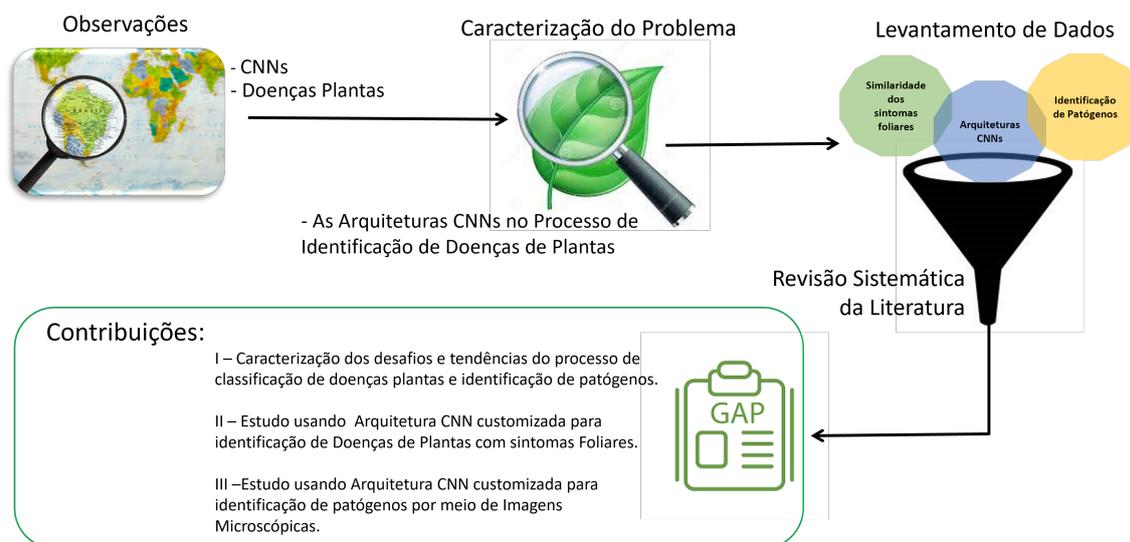


Figura 3.1: Fluxograma da Estrutura da Pesquisa.

Basicamente, o fluxograma é pautado cronologicamente nas observações realizadas por meio das soluções existentes na literatura, evidenciadas pelos resultados das arquiteturas CNNs tradicionais. As evidências desses resultados permitem caracterizar os problemas e desafios reportados e inferidos por meio de análises comparativas por reprodução de resultados.

Deste modo, foram prospectadas três contribuições para esta investigação científica, cada qual com um método científico específico e resultados inerentes aos objetivos desta pesquisa, sendo elas:

- **Revisão Sistemática da Literatura:** São consideradas estudos observacionais de recuperação e análise crítica da literatura. Testam hipóteses e visam levantar, reunir, avaliar criticamente o método de pesquisa e sintetizar os resultados de diversos estudos primários. São consideradas evidências científicas de maior grandeza para identificação de lacunas nas áreas de estudos. Assim, a elaboração deste estudo possibilitou caracterizar o estado da arte do uso CNNs no processo de identificação de doenças de plantas e determinar duas oportunidades de investigações científicas. Conseqüentemente, essas oportunidades foram exploradas, resultando em duas novas contribuições, descritas a seguir.

- Uma Arquitetura CNN com topologia customizada para Identificação de Doenças de Plantas com sintomas Foliares: Esta contribuição foi concebida como uma pesquisa exploratória registrando, analisando, interpretando e identificando resultados. Foram utilizados métodos de modelagem e simulação para comparar resultados obtidos, por meio de métricas quantitativas com estudos realizados em cenários equivalentes.
- Identificação de Fitonematoides usando modelos CNNs por meio de Imagens Microscópicas: Esta última lacuna tem um caráter de pesquisa exploratória, primordialmente pela ausência de estudos correlatos. Deste modo, foi preciso construir um conjunto de dados de imagens de fitonematoides avaliando as arquiteturas tradicionais que representam o estado da arte no processo de classificação de objetos na literatura. Assim, partindo das análises e interpretações iniciais, foi possível prospectar uma Arquitetura de topologia customizada capaz de superar o estado da arte em desempenho, tamanho de modelo, número de parâmetros e tempo de convergência de treinamento.

3.3 Estratégias de Concepção, Avaliação e Validação das Arquiteturas CNNs Customizadas

A literatura demonstrou que a otimização de parâmetros, regularização e a reformulação estrutural das CNNs trazem melhorias para os resultados. Entretanto, de acordo com (KHAN *et al.*, 2020), o principal impulso no desempenho das CNNs advém da reestruturação das unidades de processamento e do projeto de novos blocos. A maioria das inovações nas arquiteturas CNNs foram em relação à profundidade e à exploração espacial, principalmente das abordagens que lidam com a classificação de doenças de plantas e identificação de patógenos (ABADE *et al.*, 2021a). Neste estudo, denominou-se customização de arquitetura ao processo de reestruturação das unidades de processamento (blocos) CNNs juntamente com o arranjo de otimizações de parâmetros e regularização realizados sob uma arquitetura.

Nesta perspectiva, foram elaboradas duas estratégias a partir das observações realizadas e sistematizadas por meio da literatura. A primeira versa sobre o processo de reestruturação das unidades de processamento de arquiteturas CNNs baseadas na exploração espacial, na exploração de canais de entrada (*multichannel*) e em múltiplos caminhos (*multi-path*) para classificação de doenças de plantas. Já a segunda customiza uma arquitetura utilizando categorias baseadas na profundidade, multiconexão com base em largura e em múltiplos caminhos (*multi-path*).

3.3.1 Arquitetura CNN Customizada para Classificação de Doenças de Plantas

A partir do referencial teórico sistematizado foram observadas arquiteturas baseadas na exploração espacial de maior destaque nas abordagens para classificação de doenças de plan-

tas. Adicionalmente, priorizaram-se estudos com um conjunto de dados representativos e com acesso público. Partindo destas premissas, as arquiteturas AlexNet (KRIZHEVSKY *et al.*, 2012) e GoogLeNet (SZEGEDY *et al.*, 2015a) juntamente com o conjunto de dados Plant-Village (HUGHES; SALATHÉ, 2016) foram selecionados.



Figura 3.2: Processo de Customização da Arquitetura CNN com Junção das Categorias propostas por KHAN *et al.* (2020) para Classificação de Doenças de Plantas.

Para a etapa de avaliação e validação, os estudos de Mohanty *et al.* (2016) e Ferentinos (2018) nortearam todo processo. As estratégias utilizadas e os resultados alcançados nestas abordagens foram reproduzidos.

Na Figura 3.2 é apresentado o processo de customização da arquitetura CNN com a junção das categorias selecionadas. A escolha destas categorias foi fundamentada nas afirmações de Karpathy *et al.* (2014) e Khan *et al.* (2020) sobre os benefícios de estruturas arquitetônicas CNNs que fazem uso de multicanais de entrada com um processamento com múltiplos caminhos.

Quanto à otimização dos parâmetros, utilizou-se o algoritmo denominado pesquisa de grade (*Grid Search*) (SNOEK *et al.*, 2012a; LIASHCHYNSKYI; LIASHCHYNSKYI, 2019) para examinar a combinação de hiperparâmetros delimitada dentro de um conjunto pré estabelecido de valores. Esses valores foram obtidos por meio da reprodução dos experimentos realizados nas abordagens utilizadas como referência e devida ajustados às características da arquitetura customizada.

Na Figura 3.3 é apresentada uma ilustração do pseudocódigo desse algoritmo com a representação de seu funcionamento. Primordialmente, o conjunto de hiperparâmetros são definidos para alimentar a pesquisa em grade. Posteriormente, esse mesmo algoritmo examina automaticamente o espaço de busca dos hiperparâmetros e tenta encontrar os valores ideais que maximizam a precisão do modelo. Os hiperparâmetros associados a maior precisão são, então, retornados. Cabe salientar que a escolha por este algoritmo é pautada no conhecimento prévio do tamanho do espaço de busca dos hiperparâmetros, fazendo com que esse algoritmo fosse adequado ao contexto da investigação.

Grid Search

Pseudocódigo

Hiperparâmetro_Um = {a, b, c}
 Hiperparâmetro_Dois = {x, y, z}

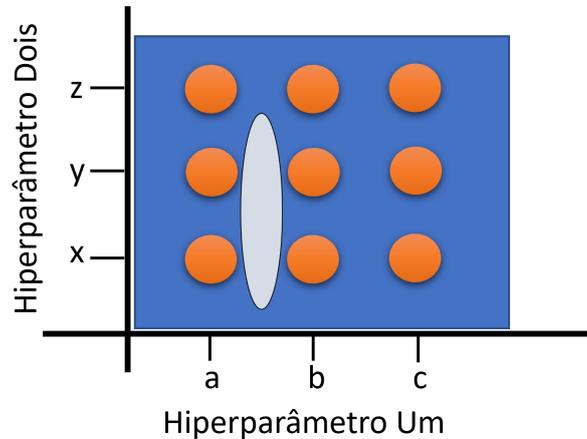


Figura 3.3: Algoritmo da Pesquisa de Grade para definição do conjunto de hiperparâmetros. Adaptado de (SNOEK *et al.*, 2012a)

3.3.2 Arquitetura CNN Customizada para Identificação de Patógenos com Imagens Microscópicas

Diferente da primeira estratégia, poucos estudos abordam a identificação de patógenos causadores de doenças em plantas usando imagens microscópicas. Assim, as observações foram conduzidas para sumarizar qual arquitetura obtinha a melhor acurácia utilizando imagens microscópicas. Os estudos de Modarres *et al.* (2017), Khosravi *et al.* (2018), Golatkar *et al.* (2018), Kensert *et al.* (2019) e Dong *et al.* (2020) evidenciaram uma predominância de arquiteturas da família Inception, com destaque para InceptionV3 (SZEGEDY *et al.*, 2015b). Nesta perspectiva, Khosravi *et al.* (2018) afirma que os blocos inception demonstram melhor capacidade de classificação com imagens microscópicas do que outras arquiteturas avaliadas. Na Figura 3.4 é possível acompanhar a evolução dos blocos inception.

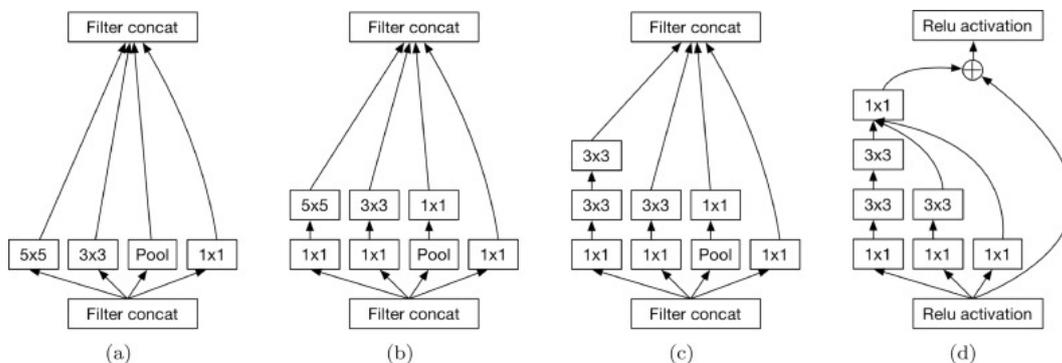


Figura 3.4: a) Bloco Inception básico; b) Bloco Inception implementado na GoogLeNet (SZEGEDY *et al.*, 2015a); c) Bloco Inception aprimorado na InceptionV3 (SZEGEDY *et al.*, 2015b); d) Bloco Inception-ResNet (SZEGEDY *et al.*, 2017a)

Quanto ao conjunto de dados, face à escassez de abordagens havia também uma dificuldade de encontrar conjuntos de dados com imagens catalogadas. Assim, dentro do planejamento de atividades da pesquisa, optou-se por criar um conjunto de dados, sendo o patógeno escolhido os fitonematoides, pelo seu alto nível de potencial de dano às culturas e pelo inexpressivo (FERRAZ; BROWN, 2016) número de abordagens encontradas na literatura (ABADE *et al.*, 2021a).

Após as primeiras evidências observadas quanto a uma arquitetura de referência e a construção do conjunto de dados, foram selecionados treze modelos tradicionais da literatura, sendo eles: VGG16, Xception, InceptionV3, ResNet50, ResNet101, ResNet152, InceptionResNetV2, DenseNet121, DenseNet169, DenseNet201, EfficientNetB0, EfficientNetB3 e a NASNetLarge. Estes modelos representam o estado da arte no processo de classificação de objetos e nortearam as etapas de avaliação, customização e validação. Cabe destacar que todos os modelos arquitetônicos CNNs selecionados utilizavam categorias baseadas em profundidade (ResNet, DenseNet), multiconexão com base em largura (Xception e InceptionV3) e múltiplos caminhos (NASNetLarge). As arquiteturas InceptionResNet e EfficientNet mesclam as propriedades de profundidade e multiconexão com base em largura em suas abordagens.

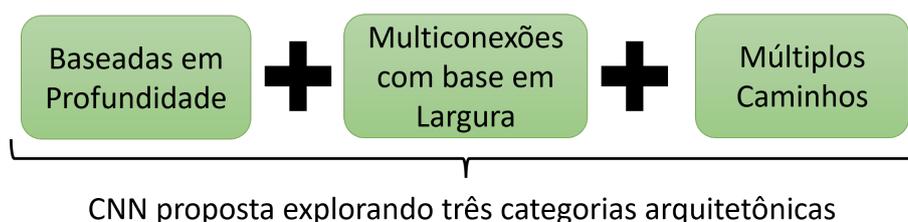


Figura 3.5: Processo de Customização da Arquitetura CNN com Junção das Categorias propostas por KHAN *et al.* (2020) para Identificação de Patógenos com Imagens Microscópicas.

Na Figura 3.5 é apresentado o processo de customização da arquitetura CNN com a junção das categorias para identificação de fitonematoides usando imagens microscópicas. As categorias selecionadas para compor este processo de customização foram fundamentadas nas características arquitetônicas dos modelos CNNs selecionados. Assim, serão selecionadas as estruturas com melhor desempenho para compor a base da solução CNN customizada.

Para efetuar a otimização dos parâmetros, utilizou-se o algoritmo denominado Taxas de aprendizagem cíclica (*Cyclical Learning Rates* - CLR) (SMITH, 2017) para ajustar e encontrar uma taxa de aprendizado ideal para cada modelo. Essa estratégia permite que os modelos alcancem uma convergência mais rápida e com menos experimentos e consequentemente atualizações de hiperparâmetros.

As taxas de aprendizado cíclico oscilam para frente e para trás entre dois limites durante o treinamento, aumentando lentamente a taxa de aprendizado após cada atualização em lote. A opção por este algoritmo foi pela grande variedade de modelos analisados e pela impossibilidade de conhecimento prévio do tamanho do espaço de busca dos hiperparâmetros adequados.

Para etapa de avaliação e validação dos resultados, detalhou-se um conjunto de critérios,

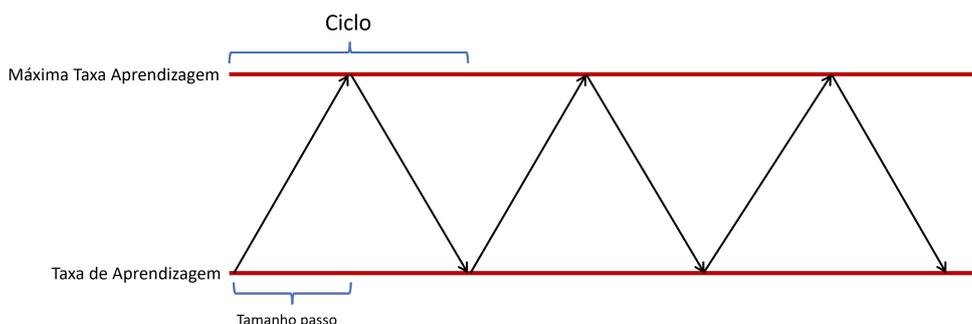


Figura 3.6: Algoritmo Taxa de Aprendizagem Cíclica para Ajuste da Taxa de Aprendizagem usando estratégia Triângulo. Adaptado de (SMITH, 2017)

métricas e indicadores de performance tradicionalmente utilizados por abordagens CNNs para aferir qualidade aos modelos.

3.3.2.1 Métricas e Performance

Para avaliar o desempenho de classificação das arquiteturas propostas, a perda (*loss*), precisão geral (*overall accuracy*), F_1 -scores, precisão (*precision*), recuperação (*recall*) e especificidade (*specificity*) foram selecionadas como as métricas de desempenho de precisão. No estágio de treinamento, os pesos internos do modelo são atualizados durante várias interações. Foi monitorada cada iteração por período de treinamento, salvando os pesos com a melhor capacidade preditiva do modelo determinado pela métrica de precisão geral de validação. A Tabela 3.1 abaixo mostra como as métricas são calculadas.

Tabela 3.1: Métricas usadas para avaliar o desempenho das arquiteturas CNNs investigadas.

Métrica	Fórmula	Foco de Avaliação
Loss	$L(\hat{y}_i, y_i) = - \sum_{i=1}^k y_i \cdot \log(\hat{y}_i)$	Uma função de perda é um método de avaliação da qualidade do modelo do conjunto de dados. A função de perda produzirá um número mais alto se as previsões estiverem fora dos valores alvos reais, ao passo que, caso contrário, ela produzirá um número menor. Como o problema investigado é do tipo classificação multiclasse, utilizou-se a entropia cruzada para a função de perda.
Accuracy	$\sum_{i=1}^k \frac{tp_i + tn_i}{tp_i + tn_i + fp_i + fn_i}$	A precisão de um algoritmo de classificação de aprendizado de máquina é uma maneira de medir com que frequência o algoritmo classifica um ponto de dados corretamente. Número de itens identificados corretamente como verdadeiramente positivos ou negativos em relação ao número total de itens.
F_1 -score	$2 * \frac{Precision * Recall}{Precision + Recall}$	A média harmônica da precisão e da recuperação, mede a eficácia da identificação quando tanta importância é dada a recuperação quanto à precisão.
Precision	$\frac{\sum_{i=1}^k tp_i}{\sum_{i=1}^k (tp_i + fp_i)}$	Concordância dos verdadeiros rótulos da classe com os do classificador, calculada pela soma de todos os TP's e FP's do sistema, em todas as classes.
Recall	$\frac{\sum_{i=1}^k tp_i}{\sum_{i=1}^k (tp_i + fn_i)}$	Eficácia de um classificador para identificar rótulos de classe, calculada pela soma de todos os TP's e FN's no sistema, em todas as classes.
Specificity	$\frac{\sum_{i=1}^k tn_i}{\sum_{i=1}^k (tn_i + fp_i)}$	A especificidade é conhecida como Taxa Negativa Verdadeira. Esta função calcula a proporção de casos reais negativos que foram previstos como negativos pelo modelo.

K = número total de classes; tp = verdadeiros positivos; fp = falsos positivos; tn = verdadeiros negativos; fn = falsos negativos

3.3.2.2 Matrix de Confusão

Uma matriz de confusão (VISA *et al.*, 2011) contém informações sobre as classificações reais e previstas feitas por um sistema de classificação. O desempenho das arquiteturas implementadas foi avaliado a partir dos dados matriciais, auxiliando na localização e eliminação de problemas de vieses e variâncias e possibilitando ajustes capazes de produzir resultados mais precisos. A matriz de confusão (CM) pode ser denotada como na Equação 3.1:

$$CM = \begin{bmatrix} RR_{1,1} & RR_{1,2} & \cdots & RR_{1,N} \\ \vdots & \vdots & & \vdots \\ RR_{2,1} & RR_{2,2} & \cdots & RR_{2,N} \\ \vdots & \vdots & & \vdots \\ RR_{N,1} & RR_{N,2} & \cdots & RR_{N,N} \end{bmatrix} \quad (3.1)$$

sendo $RR_{i,j}$ correspondente ao número total de entidades na classe C_i que foram classificadas na classe C_j . Portanto, os principais elementos diagonais indicam o número total de amostras na classe C_i corretamente reconhecidos pelo sistema.

3.3.2.3 Curva Característica de Operação do Receptor - ROC-AUC

O método *Receiving Operator Characteristics - Area Under Curve (ROC-AUC)* (POWERS, 2011) ajuda a analisar o desempenho da classificação em várias configurações de limite. Altas taxas de verdadeiro positivo (TVP / Sensibilidade) de uma classe indicam que o modelo teve um bom desempenho na classificação dessa classe específica. As curvas ROC-AUC podem ser comparadas para vários modelos, e o modelo que possui uma AUC alta é considerado como tendo um bom desempenho. Na Figura 3.7 é demonstrando como a Curva ROC se comporta ao longo de diferentes pontos de corte (*thresholds*) na probabilidade.

3.4 Fases da Pesquisa

Estabelecido o roteiro metodológico e a estrutura da abordagem com suas devidas contribuições, elaborou-se um fluxograma que descreve em detalhes o processo replicável de desenvolvimento deste trabalho. As etapas do processo de pesquisa foram divididas em 7 Fases com suas respectivas atividades, tendo como pilar a fundamentação teórica ao longo de toda sua execução.

Na Fase 1, descrita como ‘Refinamento dos Requisitos e Experimento de Soluções’ é elaborada uma análise comparativa das soluções existentes quanto à identificação de doenças de plantas usando CNNs. Como parte desta etapa, as arquiteturas CNNs tradicionais foram implementadas no processo de identificação de doenças de plantas. Adicionalmente, foram avaliadas as principais características que influenciam na concepção de soluções usando CNNs para a

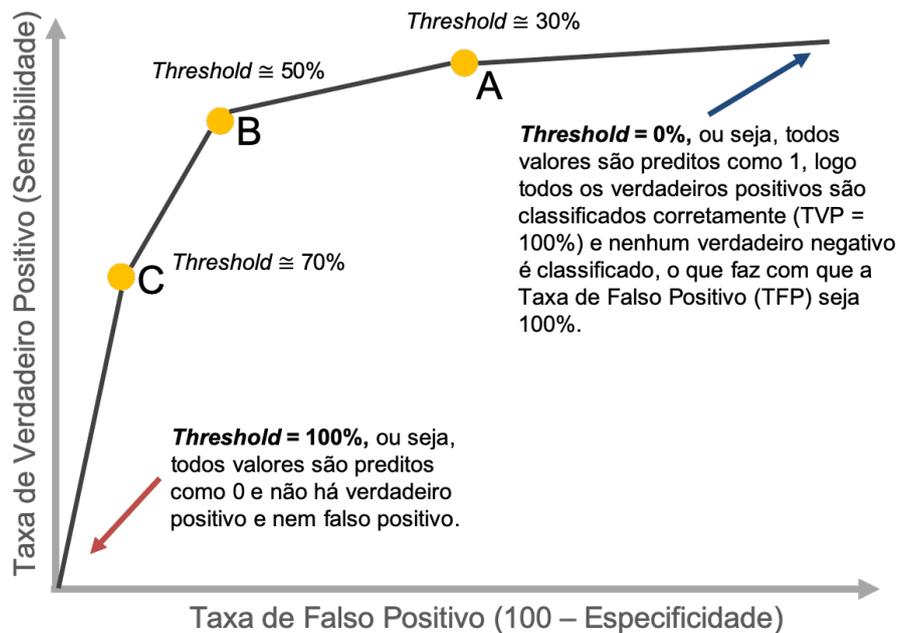


Figura 3.7: Ilustração da Curva ROC-AUC - Adaptada de Prates (2020).

identificação de doenças de plantas.

A Fase 2 é pautada pelas conclusões sobre as características que influenciam na concepção de soluções CNNs no processo de identificação de doenças de plantas. Deste modo, esta fase é descrita como ‘Revisão Sistemática da Literatura e Lacunas do Conhecimento’, pois tem por objetivo caracterizar o estado da arte sobre o reconhecimento de doenças de plantas com imagens usando redes neurais convolucionais. Consequentemente, elencar as lacunas do conhecimento e permitir a formulação de algumas das hipóteses desta pesquisa estão entre as atribuições desta fase.

Conjuntos de dados de imagens com objetos de natureza específica constituem uma lacuna de consenso entre os investigadores dessa área. Deste modo, a Fase 3 é dedicada à criação de um conjunto de dados com imagens microscópicas devidamente rotuladas para identificação de fitonematoides. As etapas para obtenção das amostras dos patógenos e todo seu preparo laboratorial estão detalhadas na Seção 6.1.1, sendo aqui sintetizadas em 3 atividades. Cronologicamente, podem ser descritas como a captura das imagens microscópicas, construção de um banco de dados de imagens de acordo com a taxonomia de classificação de fitonematoides e, por fim, a indexação e rotulagem dos tipos de fitonematoides presentes nas imagens.

Na Fase 4, descrita como ‘Aplicação e Concepção de Modelos e Arquiteturas CNNs’ foram aplicadas e avaliadas as principais arquiteturas CNNs utilizando o conjunto de dados criado na Fase 3. Posteriormente, empregaram-se técnicas de otimização de hiperparâmetros e ajuste fino, visando o incremento da acurácia desses modelos.

Partindo da sumarização dos resultados obtidos na Fase 4, juntamente com as peculiaridades do conjunto de dados criado na Fase 3 e o propósito de atender cada objetivo específico para subsidiar afirmativamente as hipóteses levantadas, a Fase 5 foi organizada em 3 etapas e descrita

como Proposta, Teste e Validação. A Proposta descreve uma implementação de uma arquitetura customizada com as características do contexto de aplicação. Na etapa de Teste são avaliadas as funcionalidades, necessidades de adequações das camadas e blocos que compõe a arquitetura. Por fim, na etapa de Validação, o modelo é executado com simulações em pequenos lotes de dados para observar sua capacidade de adequação, convergência, eficiência, tamanho e número de parâmetros.

Já na Fase 6, descrita como ‘Implantação e Validação das Hipóteses’, a arquitetura proposta é implantada como estudo experimental, utilizando o conjunto de dados criado na Fase 3. Foram realizadas avaliações de desempenho entre a abordagem proposta e as CNNs tradicionais, cujos resultados já haviam sido aferidos na Fase 4. Procedendo às análises comparativas, foi possível explicar alguns fenômenos envolvidos em detrimento das casualidades e singularidades do problema investigado.

Por fim, na Fase 7 são formalizados os relatórios das atividades desenvolvidas nesta pesquisa e, conseqüentemente, os resultados alcançados. Os Resultados significativos desta pesquisa deram origem a manuscritos científicos e foram publicados em Congressos e Periódicos de alto fator de impacto. Adicionalmente, a redação da tese e a transferência das contribuições científicas, tecnológicas ou de inovação alcançada ocorrem também nesta última fase da pesquisa.

O fluxograma apresentado por meio da Figura 3.8 foi organizado e estruturado em fases com suas respectivas atividades, delimitando o escopo para cada etapa do processo de desenvolvimento. A base de habilidades e competências foi fomentada pela fundamentação teórica que permeia cada fase de execução em um processo de consolidação dos saberes.

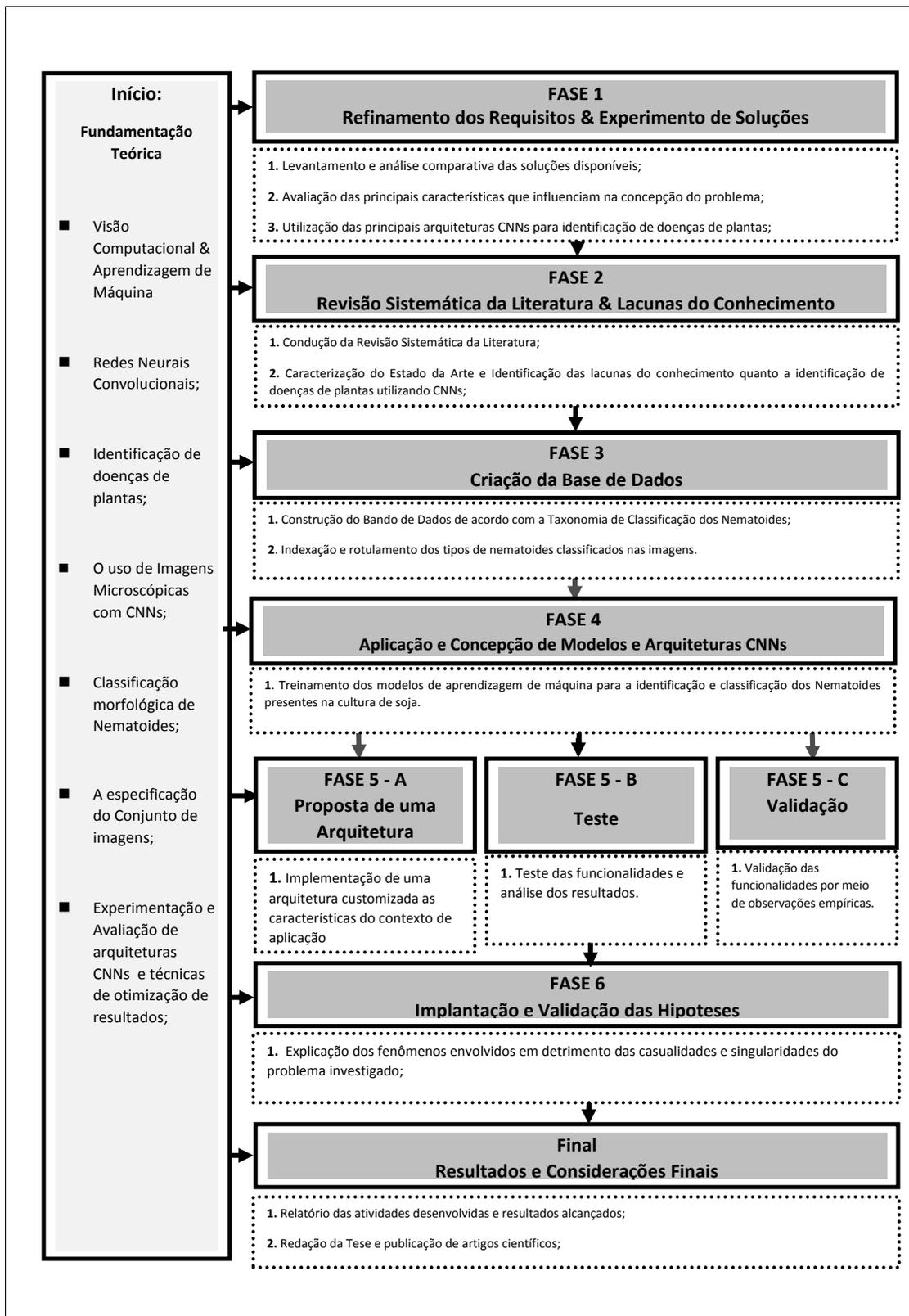


Figura 3.8: Descrição do processo replicável de Pesquisa.

3.5 Considerações Finais

Neste capítulo, a proposta metodológica descreve os métodos e processos aplicáveis à pesquisa em suas diversas etapas. Para que o resultado desta pesquisa científica possa ser considerado válido e confiável é necessário garantir que os resultados alcançados possam ser replicados ou reproduzidos.

Deste modo, na Seção 3.2 a estrutura da metodologia proposta é apresentada e os detalhes que seguem o fluxo da execução da pesquisa são evidenciados.

Na Seção 3.4 as etapas do processo de pesquisa foram divididas em 7 Fases com suas respectivas atividades, tendo como pilar a fundamentação teórica ao longo de toda sua execução. Cada Fase é composta por um conjunto de atividades comuns a um dos objetivos da pesquisa.

Nos próximos três capítulos serão apresentadas os resultados alcançados no desenvolvimento deste trabalho. O Capítulo 4 apresenta uma revisão sistemática da literatura sobre o reconhecimento de doenças de plantas com imagens usando CNNs. Já no Capítulo 5 aborda o reconhecimento de doenças de plantas a partir de imagens digitais usando CNNs. Por fim, o Capítulo 6 apresenta um modelo de rede neural convolucional para identificação de fitonematoides em imagens microscópicas.

Capítulo 4

Revisão Sistemática sobre o Reconhecimento de Doenças de Plantas usando Redes Neurais Convolucionais

4.1 Introdução

A identificação e reconhecimento de doenças de plantas, por meio de ferramentas computacionais, é uma área que demanda atenção dos pesquisadores. A agricultura, em qualquer parte do mundo, depende da qualidade e quantidade dos produtos agrícolas, especialmente as culturas com propósito alimentício. A necessidade de otimização das técnicas de detecção de doenças de plantas impulsionou os pesquisadores a empregar primordialmente técnicas tradicionais de visão computacional e aprendizagem de máquina para facilitar essa difícil tarefa. As Redes Convolucionais (CNN), um subconjunto das abordagens de aprendizagem de máquina, surgiram como uma ferramenta versátil para assimilar grandes quantidades de dados heterogêneos e fornecer previsões confiáveis de fenômenos complexos e incertos. Entretanto, inúmeros desafios ainda dificultam a correta classificação das fitopatologias, tais como a diversidade genética e fenotípica das culturas, a grande variedade de pragas e doenças, bem como as características dos conjuntos de dados, os tipos e peculiaridades das arquiteturas e modelos de redes neurais convolucionais e a complexidade envolvida nas técnicas de otimização de resultados.

A Revisão Sistemática da Literatura (RSL) é uma técnica científica objetiva, eficiente e replicável, que permite extrapolar achados de estudos independentes, avaliar a consistência de cada um deles e explicar as possíveis inconsistências e conflitos (KITCHENHAM, 2012). Além disso, é uma técnica que possibilita aumentar a acurácia dos resultados, melhorando a precisão das estimativas de efeito de uma determinada solução.

Deste modo, com intuito de caracterizar o estado da arte, elaborou-se uma revisão sistemática da literatura, a qual busca identificar os aportes significativos da construção da teoria às aplicações práticas. Buscou-se apontar as restrições sobre o campo em que se move as pes-

quisas e mapeando as lacunas do conhecimento, a fim de identificar experiências inovadoras investigadas que apontem alternativas de solução para os problemas da prática.

Esta Seção apresenta o planejamento do estudo e resume os principais procedimentos para realizar a RSL. De modo geral, os estudos primários que compõe uma RSL visam propor novas soluções e tecnologias e/ou caracterizar uma determinada solução ou tecnologia em uso, dentro de um contexto específico. De acordo com Brereton *et al.* (2007), a sumarização dos resultados dos estudos primários, por meio de estudos secundários, é um mecanismo de pesquisa valioso por prover conhecimento de um dado tópico e apoiar na identificação de temas para pesquisas futuras.

Uma RSL é conduzida por meio de um processo que envolve primordialmente três fases (BIOLCHINI *et al.*, 2005; BRERETON *et al.*, 2007; KITCHENHAM, 2004; MAFRA; TRAVASSOS, 2006): Planejamento, Condução e Publicação dos resultados. Essas fases, bem como as respectivas atividades, são apresentadas por meio da Figura 4.1, e podem ser conduzidas de modo iterativo.

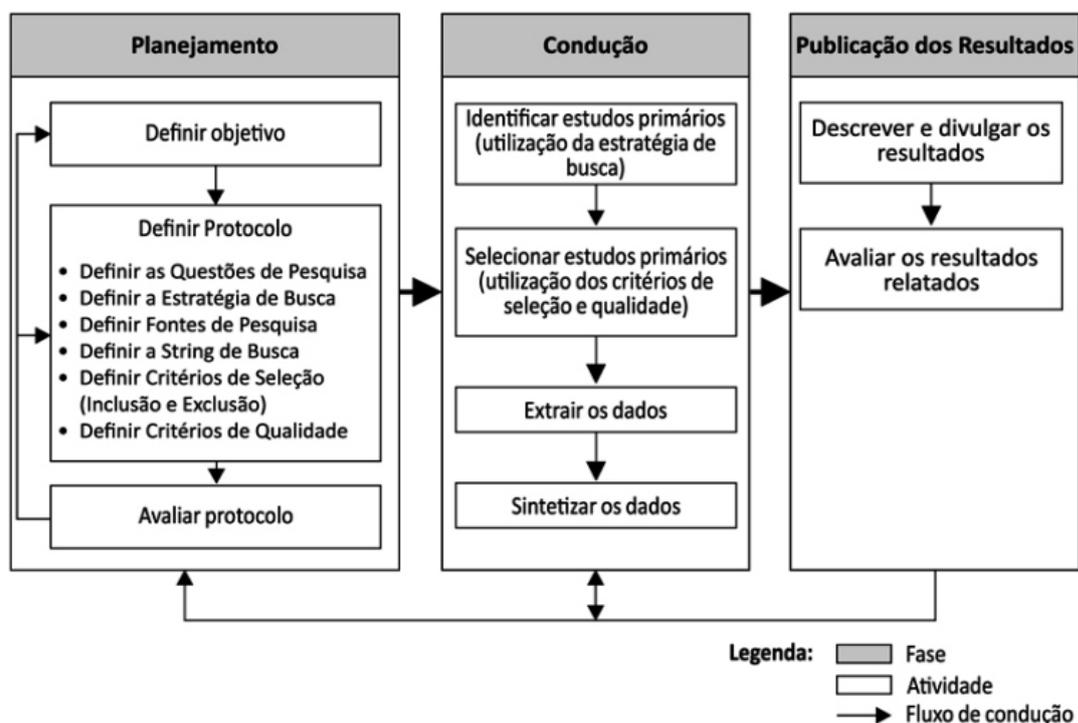


Figura 4.1: Fases e Atividades do Processo de Revisão Sistemática da Literatura. Fonte: Nakagawa *et al.* (2017)

4.2 Planejamento da Revisão

Na fase de planejamento é preciso identificar a real necessidade, ou seja, a motivação para execução da RSL. Assim, definir os objetivos e elaborar um protocolo para RSL são elementos essenciais para uma execução exitosa da técnica. Cabe destacar que a qualidade do protocolo impacta diretamente nos resultados da RSL. O protocolo minimiza vieses que possam vir a ser cometidos pelos pesquisadores (BRERETON *et al.*, 2007), especificando as questões de pesquisa, a estratégia que será utilizada, as fontes consideradas para busca, os critérios para a seleção e definição de qualidade dos estudos, como os dados serão extraídos dos estudos e, em seguida, como serão sintetizados.

O planejamento desta RSL seguiu os processos definidos por Fabbri *et al.* (2013). Para prover transparência e permitir a replicação e auditoria de todo processo (requisitos essenciais em revisões sistemáticas), utilizou-se a StArt Tool (State of the Art Through Systematic Review) (HERNANDES *et al.*, 2012), que apoia todas as fases de uma RSL.

- **Questões da Pesquisa**

É sabido que as CNNs têm demonstrado melhores resultados que as técnicas tradicionais de Visão Computacional e demais técnicas usando Aprendizagem de Máquina. Deste modo, esta RSL destina-se a identificar as características que afetam a eficácia das abordagens de identificação de doenças de plantas utilizando redes neurais convolucionais. E a pergunta de pesquisa principal da SRL que se pretende responder, é:

"Como as redes neurais convolucionais inovam e superam os desafios do processo de identificação de doenças de plantas?"

Adicionalmente, questões secundárias foram formuladas para auxiliar na identificação de características determinantes à resposta almejada, sendo elas: *Quais abordagens fazem uso de novas arquiteturas ou modelos?; Quais as características dos conjuntos de dados predominantemente utilizados?; Quais os tipos de culturas mais investigadas com as abordagens utilizando CNNs?; Quais tipos de abordagens e frameworks comumente utilizados? Qual algoritmo CNN predominante nas abordagens atuais?; Quais tipos de doenças de plantas são mais investigados com abordagens usando CNNs?*

- **Estratégia da Pesquisa**

Esta RSL concentra sua pesquisa em bases de dados científicas digitais, deste modo não foram inclusos no processo de investigação qualquer tipo de literatura cinzenta. O estudo realizado parte do pressuposto de que a maioria das pesquisas com resultados relevantes que possam constar na literatura cinzenta são tipicamente descritos ou referenciados em trabalhos científicos já publicados.

A estratégia de busca e seleção dos estudos primários concentrou-se em 5 bases de dados eletrônicas, sendo elas:

- ACM Library (<https://dl.acm.org>)
- IEEE Xplore Digital Libray (<https://ieeexplore.ieee.org/Xplore>)
- Elsevier Scopus (<https://www.scopus.com>)
- Springer Link (<https://link.springer.com>)
- Google Academic (<https://www.scholar.google.com>)

As bases de dados foram selecionadas por oferecerem o maior volume de anais de conferências e periódicos científicos de alto impacto, cobrindo amplamente o campo das investigações sobre doenças de plantas e aprendizagem de máquina por meio de CNNs.

De acordo com Petersen *et al.* (2008) e Fabbri *et al.* (2013), o processo de coleta do conjunto de estudos começa por meio da definição da *String* de Busca. Deste modo, a *String* de Busca deve ser formulada com base na experiência dos pesquisadores e revisores envolvidos no processo da RSL. Apesar de cada base de dados indexada utilizar uma sintaxe particular para implementar a *String* de Busca, será apresentada neste estudo uma versão genérica que, posteriormente, foi adaptada para cada repositório.

As palavras-chave designadas para a RSL são utilizadas para compor a *String* de Busca e localizar os estudos primários relevantes observando título, palavras-chave e resumo dos artigos indexados.

("plant disease"OR "plant pathology"OR "crop disease")

AND

("machine learning"OR "convolutional neural network"OR "deep learning"OR "DNN"OR "CNN")

• Critérios de Seleção

Uma vez identificados, os estudos precisam ser selecionados por meio da aplicação de critérios de seleção (que são os critérios de inclusão e de exclusão) e podem ser avaliados pelos critérios de qualidade. Os critérios de seleção devem especificar as principais características e/ou conteúdos que os estudos devem ter para serem incluídos ou excluídos. Já os critérios de qualidade têm como objetivo avaliar aspectos metodológicos dos estudos, ou seja, podem ser considerados aspectos como a relevância do tema de pesquisa e o uso de métodos que conduzam aos objetivos propostos no estudo.

- Critérios de Inclusão e Exclusão

Na Figura 4.2 são apresentados os critérios de inclusão e exclusão que foram utilizados para seleção dos estudos primários.

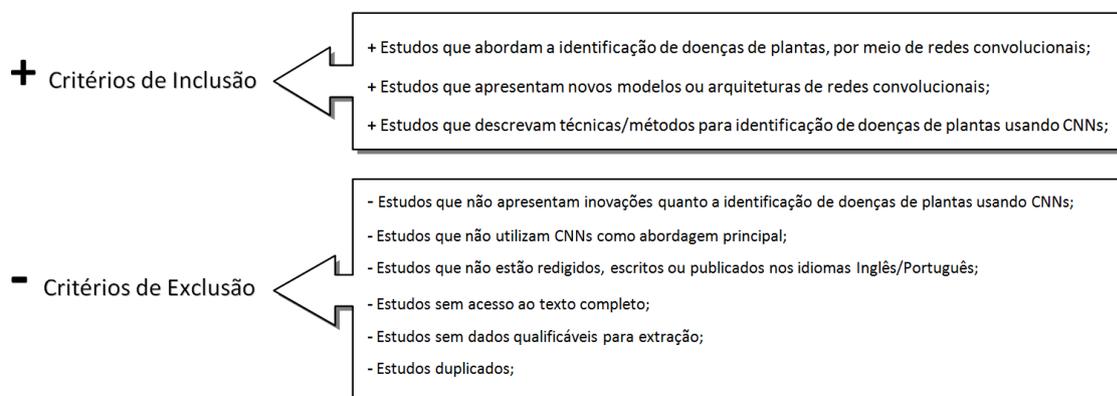


Figura 4.2: Critérios de Inclusão e Exclusão da RSL.

- Critérios de Qualidade

A avaliação da qualidade é realizada por meio de formulários desenvolvidos pelos próprios pesquisadores. Esses são compostos pelos critérios de qualidade, ou seja, perguntas sobre aspectos metodológicos de cada estudo (KITCHENHAM, 2004; STAPLES; NIAZI, 2007).

Tabela 4.1: Questões de qualidade para o computo do Índice de Qualidade dos Estudos.

ID	Questões	Indicator			Referência
		S	P	N	
Q1	Os estudos são citados por outros trabalhos ou investigações científicas?	1,0	0,5	0,0	Sim ⇒ Estudos publicados com significativa quantidade de citações. Parcialmente ⇒ Estudos publicados com baixo número de citações. Não ⇒ Estudos sem citações.
Q2	Os estudos descrevem as abordagens com detalhes e fornecem subsídios para replicação?	1,0	0,5	0,0	Sim ⇒ Estudos que descrevem explicitamente as abordagens. Parcialmente ⇒ Estudos que descrevem as abordagens, entretanto, não fornecem subsídios para replicação e verificação dos resultados. Não ⇒ Estudos que não fornecem características suficientes para descrever as abordagens, nem permitem a replicação e verificação dos resultados.
Q3	As principais constatações são claras, com relação a credibilidade, validade e confiabilidade?	1,0	0,5	0,0	Sim ⇒ Os resultados são claramente relatados. Parcialmente ⇒ As principais conclusões são parcialmente confiáveis e credíveis. Não ⇒ Não há credibilidade nas conclusões apresentadas.
Q4	As limitações dos estudos estão claramente especificadas e documentadas?	1,0	0,5	0,0	Sim ⇒ Especifica e documenta as limitações do estudo. Parcialmente ⇒ Menciona as limitações, porém não especifica tampouco documenta seus determinantes. Não ⇒ Não menciona as limitações e ameaças à validade dos estudos.
Q5	As conclusões são credíveis e consistentes com os resultados apresentados?	1,0	0,5	0,0	Sim ⇒ Estudos metodologicamente elaborados com coerência dos fatos e confiança nas descobertas apresentadas. Parcialmente ⇒ Estudos metodologicamente elaborados, porém com detalhes insuficientes para certificação dos resultados e conclusões. Not ⇒ Estudos metodologicamente mal elaborados.

A qualidade de uma RS depende da qualidade dos estudos primários incluídos. A qualidade de cada estudo está diretamente associada à sua qualidade metodológica, que pode ser medida por aspectos como: validade interna (medir efetivamente o que se deseja mensurar), validade externa (poder de generalizar os resultados), relevância do tema de pesquisa (questões de pesquisa e objetivos bem definidos e embasados na literatura), adoção de métodos que conduzam aos objetivos propostos, entre outros.

Foram definidos 5 critérios com base na influência da qualidade do estudo. A estes critérios foram atribuídos pontos, na seguinte escala de relevância: Sim (1,0), Parcialmente (0,5) e Não (0). Deste modo, os estudos primários que obtiveram um índice de qualidade (IQ) inferior a 3 pontos foram excluídos desta RSL, a fim de garantir a alta qualidade do conjunto final de estudos selecionados.

4.2.1 Condução da Revisão

Para eleger os estudos primários percorreram-se as etapas do PRISMA (*Preferred Reporting Items for Systematic and Meta-Analyses*) e foi apresentada por meio de um diagrama de fluxo, conforme as etapas: 1) identificação; 2) triagem; 3) extração e 4) seleção final (MOHER *et al.*, 2009).

Na identificação, a amostra de estudos foi extraída, armazenada e quantificada eliminando-se os estudos duplicados. Na triagem, os estudos foram pré-selecionados a partir da leitura de título, resumo e palavras-chave por dois revisores independentes, obtendo um índice Kappa >80% ($p < 0,05$) de concordância interobservador. As discrepâncias foram dirimidas por um terceiro revisor (BUSCEMI *et al.*, 2006).

No estágio da extração, realizou-se a leitura integral dos estudos pré-selecionados, excluindo os estudos que não atenderam aos critérios de inclusão e qualidade, definindo-se a amostra final.

- **Extração de Dados e Seleção Final**

A extração de dados é uma atividade em que são obtidos dados a partir da leitura completa dos estudos primários incluídos na fase de seleção. O objetivo dessa atividade é registrar, de maneira precisa, os dados necessários para responder às questões de pesquisa estabelecidas para a RSL (KITCHENHAM; CHARTERS, 2007).

Os primeiros dados a serem coletados de cada estudo primário referem-se aos metadados que serão utilizados para identificar cada estudo individualmente e também gerenciar a atividade de extração. O formulário de extração elaborado contempla cada questão de pesquisa e permite registrar as características determinantes para categorização de cada estudo primário.

A partir da extração de dados foi elaborado um quadro com as informações extraídas de cada estudo e em seguida foi desenvolvido um diagrama visual para descrever e catalogar as informações referentes às abordagens que utilizam redes neurais convolucionais para identificação de doenças de plantas. Para visualizar a localização geográfica dos estudos foi elaborado um mapa temático, com auxílio do software ArcGis 10.6.

4.2.2 Resultados e Evidências da RSL

Foram recuperados um total de 2.221 estudos, dos quais 1.433 foram identificados na base de dados *Google Academic*, 322 na *IEEE Library*, 233 na *ACM Library Database*, 195 na

Elsevier Scopus e 38 estudos na *Springer Link*.

Durante a fase de identificação foram excluídos 91 estudos duplicados, e durante a seleção inicial foram eliminados 682 por não atenderem aos critérios de inclusão estabelecidos. Para fase de extração foram selecionados 1.488 estudos e, assim, após a leitura completa do texto, foram sumarizados 121 estudos. Na Figura 4.3 estão ilustradas as etapas seguidas para a condução da RSL com a sumarização dos resultados.

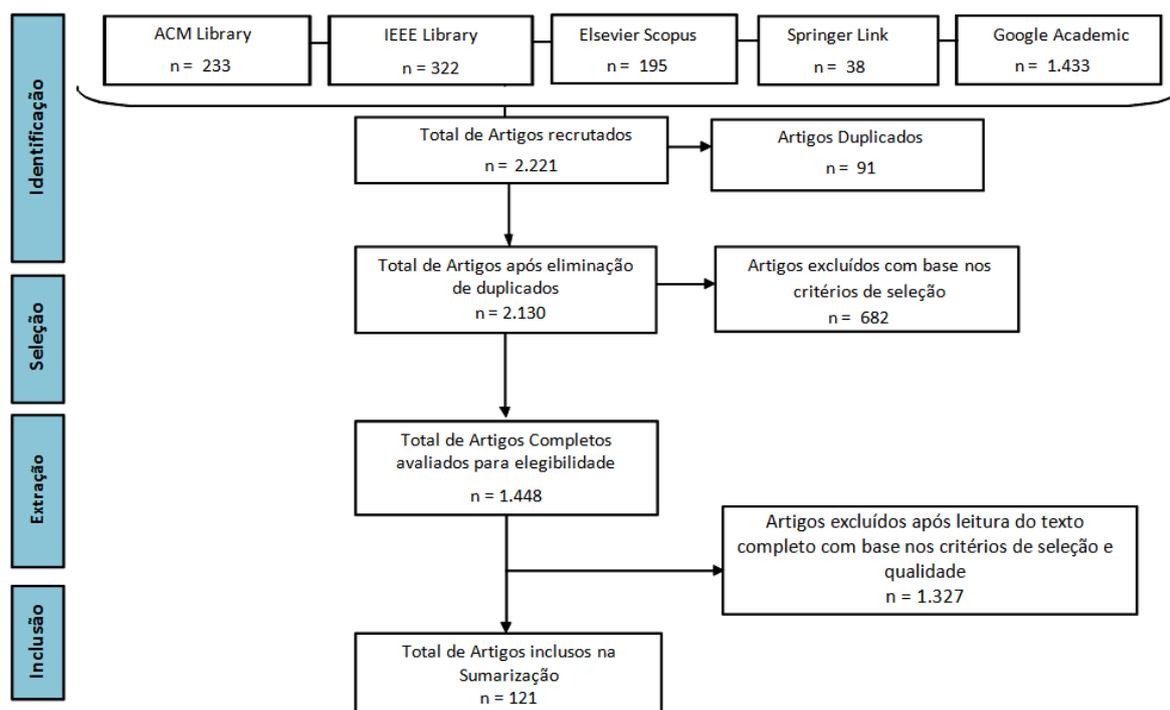


Figura 4.3: Fluxograma das fases e seleções dos estudos da RSL.

A atividade de síntese é essencial em uma RSL e tem como objetivo principal combinar os dados extraídos de cada um dos estudos primários considerados. Deste modo, foram elaborados gráficos que combinaram os atributos ano de publicação, quantidade de estudos publicados e base de dados de origem para demonstrar a evolução progressiva de estudos que abordam a área investigada.

Na Figura 4.4 é apresentada a evolução dos estudos publicados que foram selecionados. Ao longo dos 10 anos de estudos investigados para a elaboração da RSL, nota-se que o ano de 2012 é um marco não somente para os estudos que abordam as redes convolucionais, como também da aplicação desta abordagem à identificação de doenças de plantas.

Na Figura 4.5 é possível observar a distribuição dos estudos ao longo dos anos agrupados pelas Base de Dados investigadas. Nota-se que *Google Academic* concentra o maior número

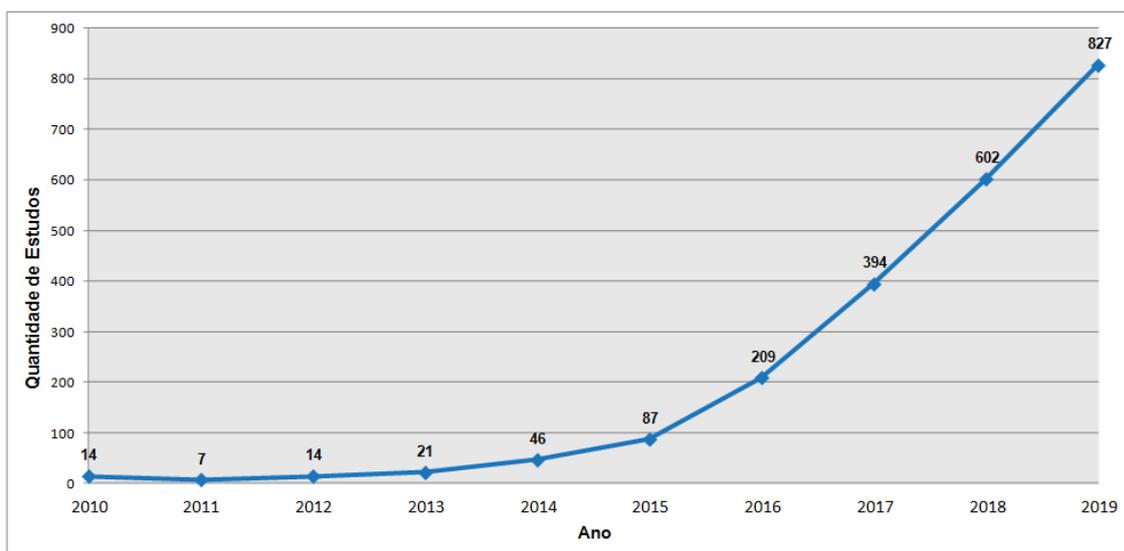


Figura 4.4: Gráfico com a evolução temporal da quantidade de estudos publicados por ano

de estudos selecionados, por ser considerada uma base de dados híbrida e por indexar maior número de estudos de congressos e periódicos asiáticos.

Os estudos selecionados foram distribuídos geograficamente conforme ilustrado, por meio da Figura 4.6. A Figura (A) apresenta a distribuição geográfica de 1.448 estudos elegíveis para extração e síntese de dados. Na Figura (B) é apresentada a distribuição dos 121 estudos incluídos na síntese dos resultados desta RSL.

Notou-se que as investigações e, conseqüentemente, o interesse científico sobre a identificação de doenças de plantas utilizando técnicas de aprendizagem de máquina são coincidentes

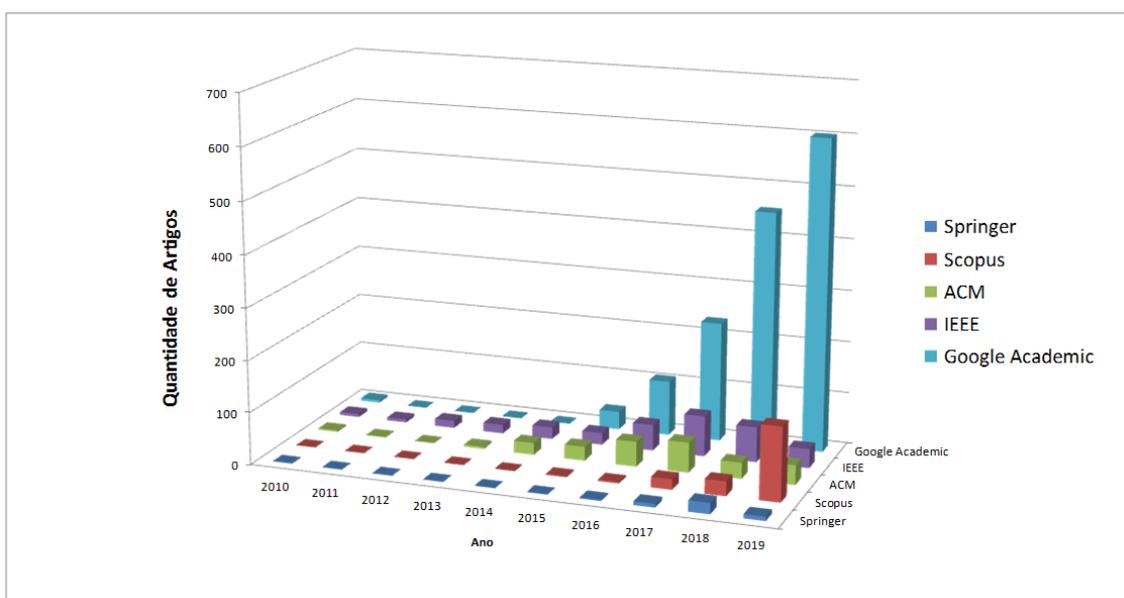


Figura 4.5: Distribuição temporal dos estudos por Base de Dados.

com os países com maior população do mundo, entre eles: China, Índia, Estados Unidos, Indonésia, Paquistão e Brasil (DIVISION, 2019).

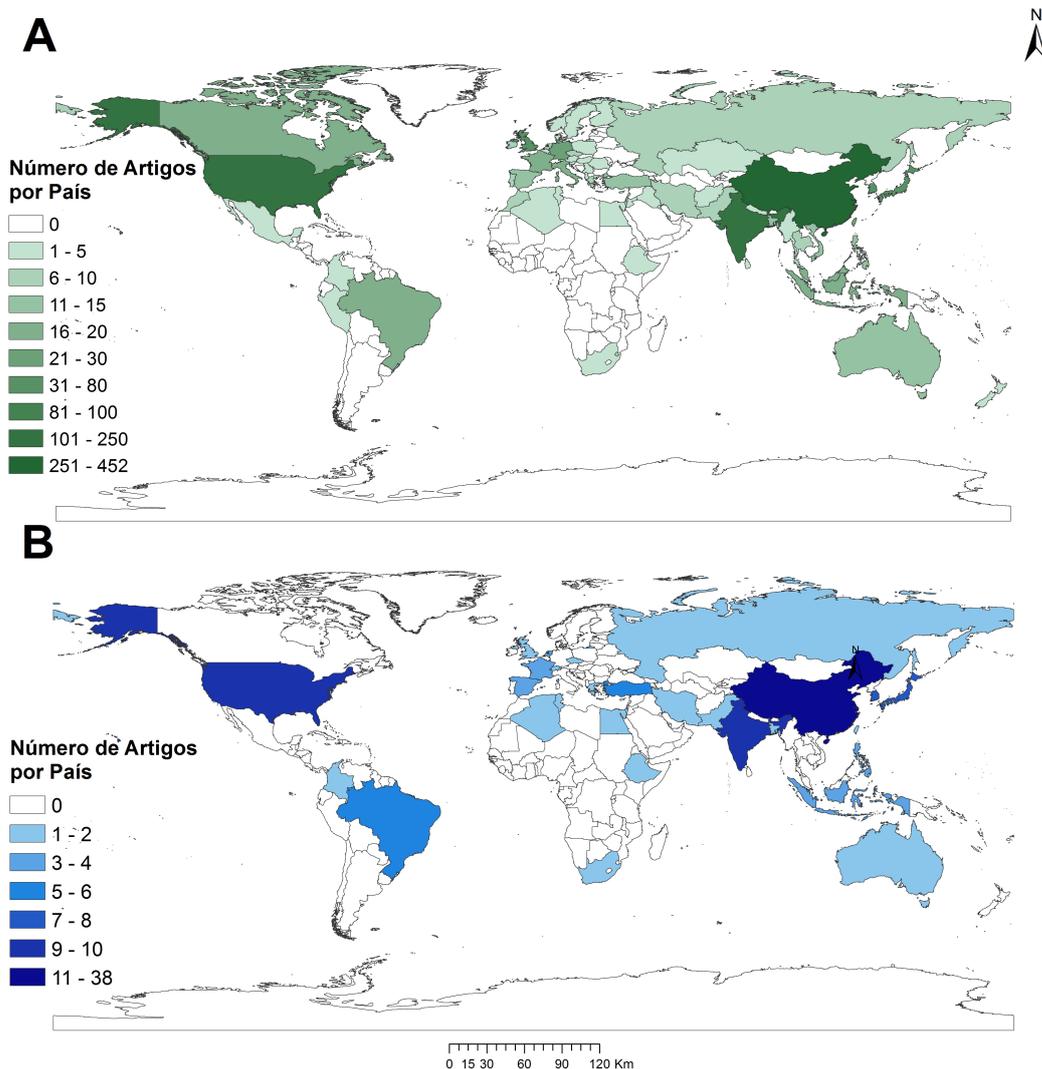


Figura 4.6: (A) Distribuição geográfica dos estudos avaliados para elegibilidade e (B) Distribuição geográfica dos estudos inclusos na seleção final.

Na Tabela A.1 do apêndice A as informações sobre cada tipo de doença ou injúria causada por patógenos foram detalhadas e agrupadas de acordo com o conjunto de dados utilizado em cada estudo. Cada coluna da Tabela resume um recurso relevante do processo de investigação que está diretamente relacionado às perguntas formuladas pela RSL.

Adicionalmente, na Tabela A.2 do apêndice A são listados, de acordo com ano de publicação e o índice de qualidade, os 121 estudos primários selecionados para a fase de seleção final. Cada coluna da tabela sintetiza uma característica relevante do processo de investigação que está diretamente relacionada com as questões formuladas para a RSL. Destaca-se que para cada estudo fora designado um identificador (ID), que será usado daqui para frente para facilitar a identificação e referência de cada trabalho.

As informações organizadas nas Tabelas A.1 e A.2 proporcionam uma visão analítica so-

bre cada investigação selecionada para o processo de extração. Entretanto, diante do grande número de estudos que compõe a seleção final, optou-se por sumarizar graficamente as características mais relevantes que ajudam a responder quantitativamente as questões secundárias determinantes ao objetivo desta RSL.

Quais abordagens fazem uso de novas arquiteturas ou modelos?

Com intuito de responder esta questão, a extração de dados dos 121 estudos sumarizados possibilitou agrupar as investigações por abordagens predominantes. Foram categorizados setes grupos de abordagens, devidamente apresentados por meio da Figura 4.7.

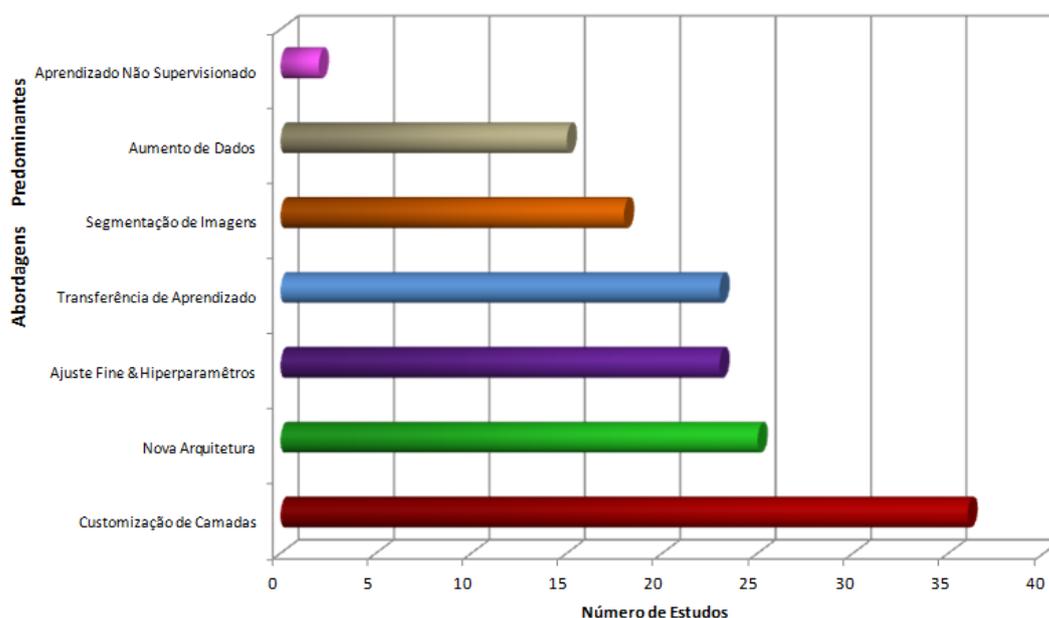


Figura 4.7: Número de estudos agrupados pela abordagem predominante.

Entre os 121 estudos avaliados é possível perceber que apenas 25 investigações (S1, S2, S6, S7, S11, S12, S13, S15, S22, S35, S43, S54, S61, S65, S68, S70, S75, S84, S86, S91, S100, S113, S114, S116, S120) propõem uma nova arquitetura.

Quais as características dos conjuntos de dados predominantemente utilizados?

Na Figura 4.8 é possível observar a distribuição dos estudos de acordo com os Conjunto de Dados (*DataSet*) de maior recorrência no processo de treinamento dos modelos para reconhecimento de doenças de plantas. Cada conjunto de dados foi categorizado em dois tipos, sendo: Imagens adquiridas em ambiente controlado com iluminação e background uniformes e Imagens adquiridas em ambiente real de cultivo sem qualquer tipo de pré processamento ou

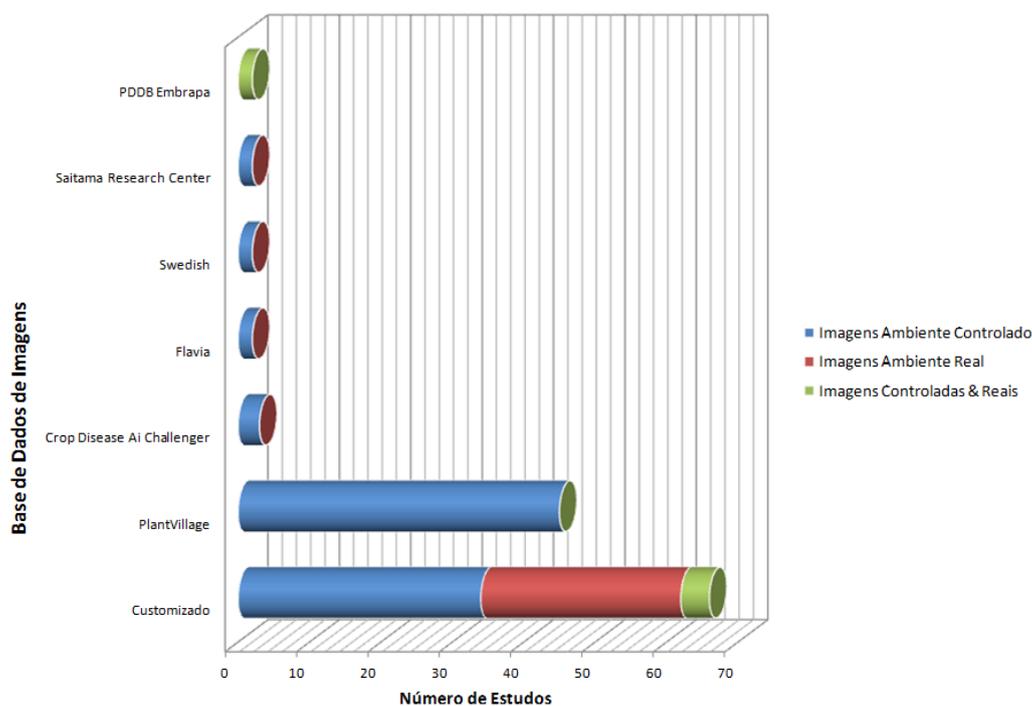


Figura 4.8: Número de estudos agrupados por Conjunto de Dados

controle de captação.

Entre os conjuntos de dados mais utilizadas nos estudos investigados, destacam-se as iniciativas customizadas elaboradas particularmente para uma necessidade de investigação e o PlantVillage (HUGHES; SALATHÉ, 2016) com satisfatório volume de imagens, tipos de culturas e fitopatologias diversificadas.

Os Conjuntos de Dados Customizadas estão presentes em 66 estudos (S1, S2, S3, S6, S7, S8, S9, S12, S15, S17, S18, S20, S22, S24, S25, S28, S32, S33, S34, S36, S37, S38, S40, S43, S44, S45, S46, S47, S49, S50, S52, S58, S64, S68, S69, S70, S71, S73, S74, S75, S77, S79, S80, S82, S84, S87, S88, S89, S90, S91, S92, S95, S96, S97, S98, S100, S102, S103, S104, S107, S112, S113, S115, S116, S119, S121) e o PlantVillage é utilizado em sua versão completa ou parcial em 45 estudos investigados (S3, S4, S9, S11, S13, S14, S18, S21, S23, S26, S27, S28, S29, S35, S41, S48, S51, S53, S55, S56, S57, S60, S61, S62, S63, S65, S66, S67, S68, S69, S72, S73, S76, S78, S81, S85, S86, S93, S99, S101, S105, S108, S110, S114, S117).

A diversidade de sensores e as diferentes perspectivas de captura de imagens também são fatores decisivos para a caracterização do conjunto de dados. Na sumarização dos estudos, quatro tipos de categorias de imagem foram identificados: RGB, Multiespectral, Hiperespectral e Microscópica, incluindo três perspectivas para aquisição dessas imagens, sendo elas: Vista Superior, Vista Frontal e Vista Aérea. As informações extraídas entre os 121 estudos da seleção final são apresentadas na Figura 4.9.

A grande maioria das investigações utilizou um conjunto de dados com imagens do tipo

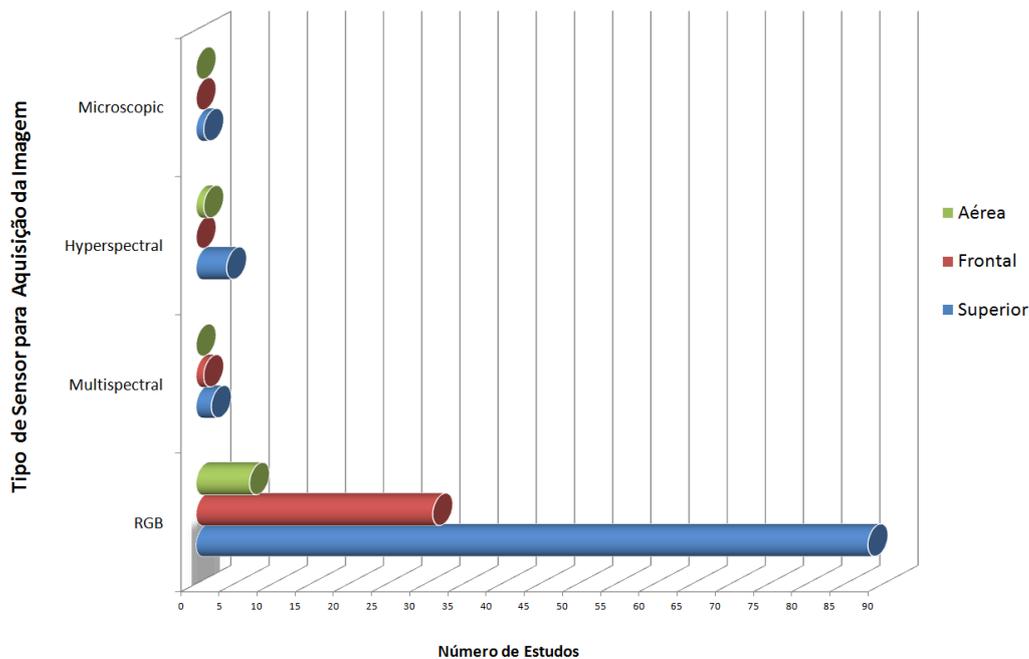


Figura 4.9: Número de estudos agrupados pelo tipo de sensor e perspectiva de aquisição das imagens.

RGB com uma perspectiva de visão superior do objeto, totalizando 88 estudos (S1, S2, S3, S4, S5, S9, S11, S13, S14, S16, S17, S18, S19, S21, S23, S26, S27, S28, S29, S31, S32, S34, S35, S36, S38, S39, S40, S41,42, S45, S46, S48, S51, S53, S55, S56, S57, S58, S59, S60, S61, S62, S63, S66, S67, S68, S69, S72, S73, S76, S78, S79, S81, S84, S86, S89, S93, S94, S96, S99, S101, S105, S107, S108, S109, S110, S111, S113, S114, S116, S118, S119, S120, S121). Investigações que utilizaram imagens RGB com vista frontal totalizaram 31 estudos (S6, S7, S10, S12, S15, S20, S24, S30, S43, S44, S49, S50, S52, S64, S65, S71, S75, S77, S83, S87, S90, S91, S92, S98, S100, S102, S103, S104, S106, S115, S117) e imagens RGB com vista aérea totalizaram apenas 7 estudos (S8, S47, S54, S82, S95, S97, S112).

Entre os 121 estudos sumarizados, três utilizaram imagens multiespectrais, sendo os estudos S25 e S33 utilizando imagens na perspectiva de visão superior e no S85 sob a perspectiva de visão frontal dos objetos. Cinco estudos usaram imagens hiperespectrais, sendo os estudos S37, S70, S74, S80 sob a perspectiva de visão superior e o estudo S22 com uma perspectiva de vista aérea. Apenas o estudo S88 usa imagens microscópicas em seu conjunto de dados.

Quais os tipos de culturas mais investigadas com as abordagens utilizando CNNs?

Os tipos de culturas presentes nas abordagens que fazem uso de CNNs para a identificação de doenças são sumarizados graficamente e apresentados por meio da Figura 4.10. É possível observar que o "Tipo" denominado "Diversas" categoriza todos os estudos que utilizam mais de um tipo de cultura em suas abordagens e, conseqüentemente, faz uso de base de dados com fenótipos diversificados de plantas. Particularmente, esta é uma característica predominante na Base de Dados PlantVillage.

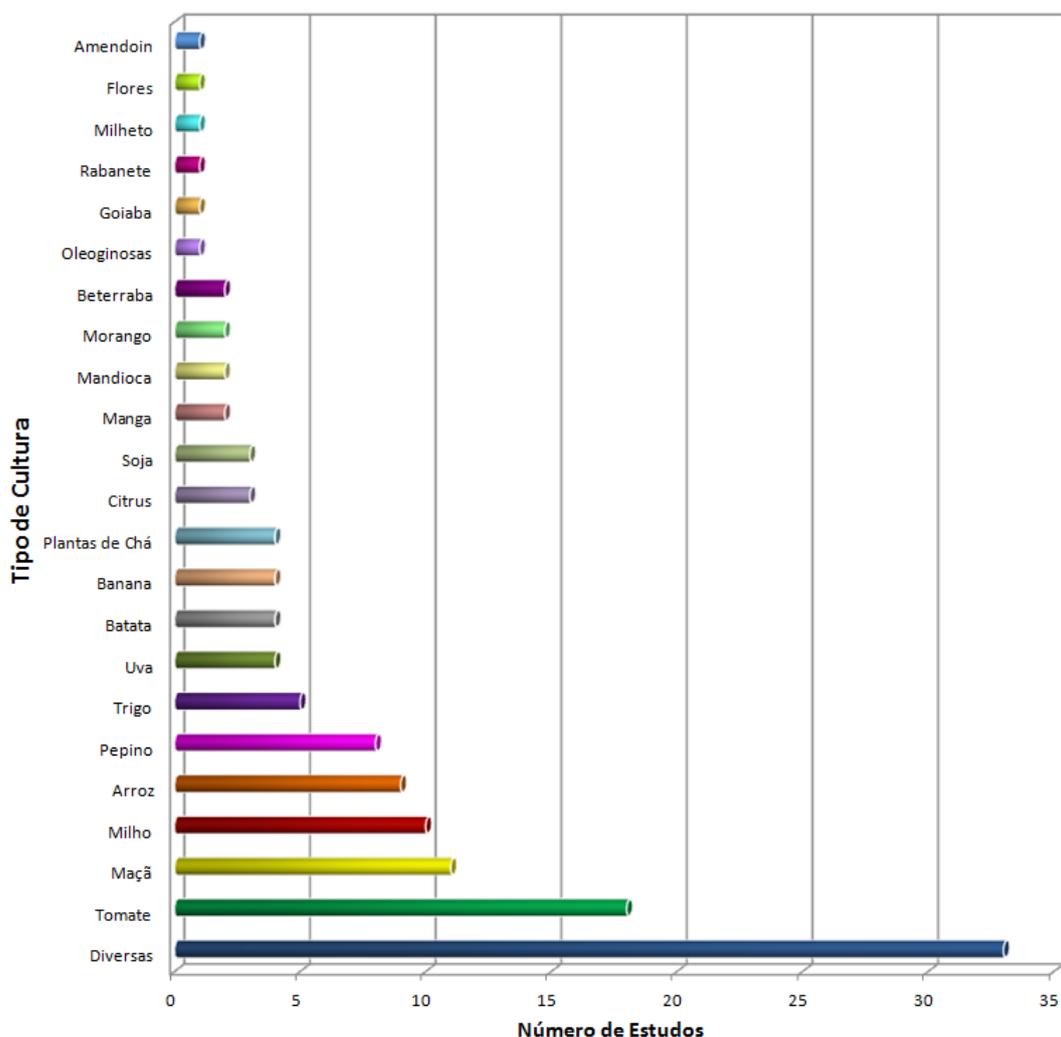


Figura 4.10: Número de estudos agrupados pelo tipo de cultura.

Entre os estudos que investigam apenas um tipo de cultura em particular, destacam-se as investigações que abordam as fitopatologias da cultura do Tomate com 18 estudos (S3, S9, S27, S44, S48, S55, S58, S61, S67, S70, S75, S78, S86, S93, S99, S102, S105, S108); Maçã com 11 estudos (S15, S25, S32, S39, S72, S85, S101, S107, S113, S116, S118); Milho com 10 estudos (S8, S27, S43, S47, S66, S71, S73, S80, S96, S103); Arroz com 9 estudos (S6, S10, S68, S86, S87, S90, S91, S95, S100); Pepinos com 8 estudos (S1, S9, S33, S65, S89, S98, S119, S121) e Trigo com 5 estudos (S2, S6, S12, S22, S77).

Quais tipos de abordagens e frameworks comumente utilizados?

A escolha de um Framework para construção e treinamento de CNNs deve levar em consideração alguns fatores como, por exemplo, o tipo de rede convolucional que será implementada à relação desempenho e escalabilidade. No processo de sumarização constatou-se que os estudos investigados fazem uso dos 6 principais Frameworks utilizados para a implementação de

soluções envolvendo aprendizagem de máquina, bem como análise preditiva.

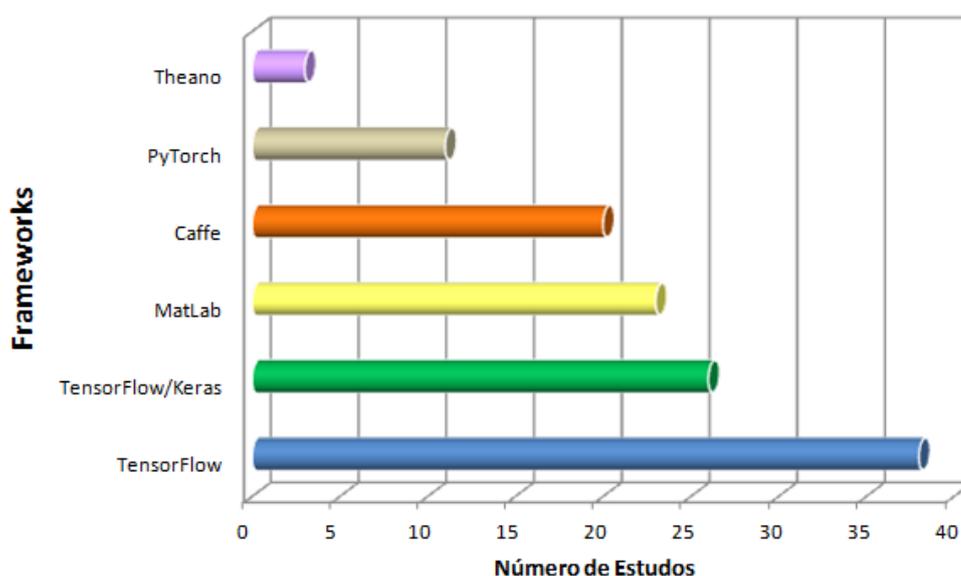


Figura 4.11: Número de estudos agrupados por Frameworks.

Na Figura 4.11 são apresentadas as distribuições dos estudos agrupados pelo Framework utilizado no processo de desenvolvimento da abordagem.

O TensorFlow é o Framework mais utilizado nas abordagens que investigam doenças de plantas por meio de redes neurais convolucionais. Entre os 121 estudos sumarizados, 38 estudos (S1, S2, S3, S5, S8, S11, S12, S14, S15, S23, S25, S27, S28, S38, S39, S40, S42, S44, S50, S53, S55, S56, S61, S62, S63, S64, S65, S66, S68, S69, S70, S76, S77, S81, S86, S104, S106, S110) utilizam o TensorFlow, seguidos por 26 estudos (S6, S7, S13, S18, S20, S21, S24, S29, S31, S33, S34, S35, S36, S41, S49, S51, S59, S72, S74, S78, S83, S88, S91, S102, S111, S118) que fazem uso do TensorFlow/Keras e 23 estudos (S9, S16, S17, S26, S30, S43, S45, S46, S52, S58, S67, S73, S89, S90, S92, S93, S95, S96, S97, S108, S114, S116, S120) que utilizam o Framework MatLab.

Qual algoritmo CNN predominante nas abordagens atuais?

O aprendizado profundo não é uma abordagem única, mas uma classe de algoritmos e topologias que pode ser aplicada a uma ampla variedade de problemas. Na Figura 4.12 é possível observar a distribuição desses algoritmos agrupados de acordo com os 121 estudos sumarizados.

Foram contabilizados 30 algoritmos usados nas abordagens, sendo predominantes aqueles já descritos na literatura como arquiteturas clássicas, com destaque para AlexNet e VGG. Foi possível observar que 19 estudos (S1, S6, S7, S11, S12, S13, S22, S35, S43, S61, S68, S70, S75, S84, S86, S91, S100, S116, S120) propõem novas arquiteturas e 15 estudos (S9, S18, S19, S23, S32, S34, S39, S41, S42, S49, S50, S51, S53, S65, S103) customizam algoritmos clássicos para resolver algum tipo de problema relacionado à identificação de doenças de plantas usando

Algoritmos

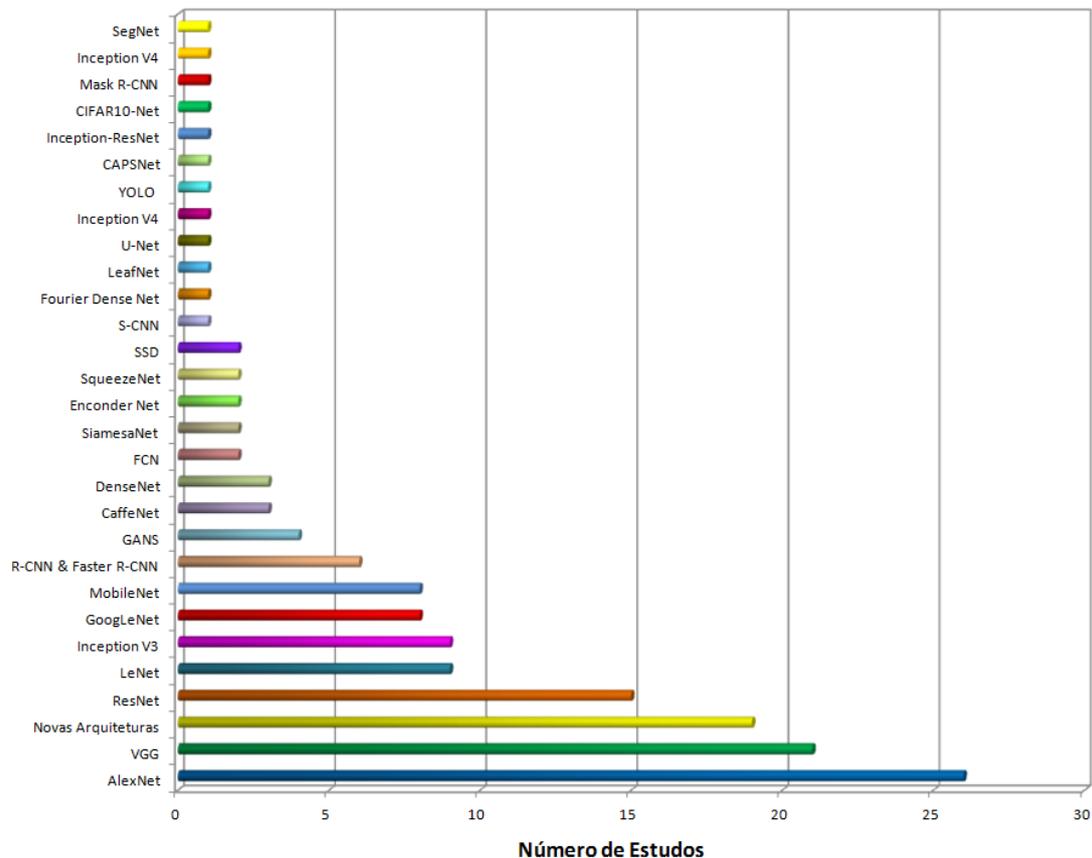


Figura 4.12: Número de estudos agrupados por Algoritmos.

CNNs.

Adicionalmente, os algoritmos foram sumarizados e categorizados de acordo com seu objetivo principal, sendo estes: Classificação Multiclasse; Classificação Binária; Detecção de Objetos; Segmentação Multiclasse e Segmentação Binária. Na Figura 4.13 é possível observar a categorização desses estudos de acordo com cada tipo de tarefa realizado pela CNN.

Os algoritmos CNN que realizam uma tarefa de classificação multiclasse (vários tipos ou estágios de doenças em plantas) são predominantes em 89 estudos (S1, S4, S6, S9, S11, S12, S13, S14, S16, S18, S19, S21, S23, S26, S27, S28, S29, S30, S31, S32, S34, S38, S39, S41, S42, S43, S46, S48, S50, S51, S52, S53, S54, S55, S56, S57, S58, S59, S60, S61, S62, S63, S64, S65, S67, S68, S69, S71, S72, S73, S74, S76, S77, S78, S79, S81, S82, S83, S84, S85, S86, S87, S89, S90, S91, S92, S93, S97, S98, S99, S100, S101, S104, S105, S106, S108, S109, S110, S111, S112, S113, S114, S115, S116, S117, S118, S119, S120, S121). No entanto, a tarefa de Classificação Binária (doente ou saudável) surge em apenas 11 estudos (S7, S10, S22, S36, S37, S47, S49, S66, S88, S95, S103).

Para a tarefa de Detecção de Objetos focada em identificar a posição da planta doente (ou a parte afetada pela doença ou por um agente causador da doença da planta) apenas 11 estu-

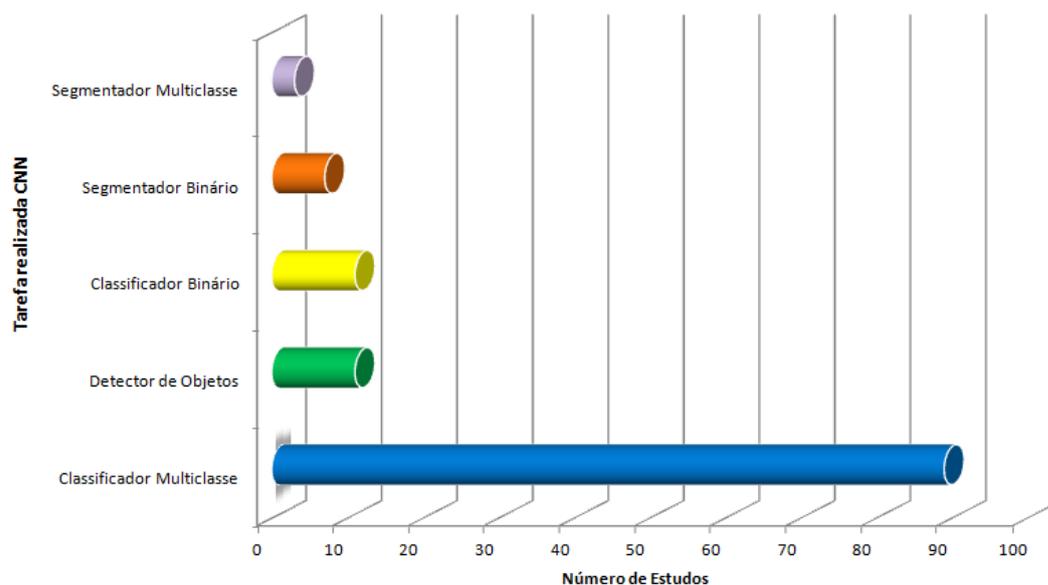


Figura 4.13: Número de estudos agrupados por tarefa principal realizada pela CNN.

dos foram identificados (S2, S15, S20, S24, S35, S40, S44, S45, S75, S94, S102). Quanto à Segmentação Binária, 7 estudos (S8, S17, S25, S33, S70, S80, S107) implementaram a técnica dividindo a imagem em regiões homogêneas de acordo com um critério definido, gerando imagens binárias das doenças de plantas. Por fim, foram sumarizados três estudos (S3, S5, S96) que implementam a tarefa de Segmentação Multiclasse, cujo objetivo é fornecer um rótulo semântico para cada pixel de uma imagem usando um conjunto predefinido de rótulos.

Quais tipos de doenças de plantas são mais investigados com abordagens usando CNNs?

Utilizando os dados sumarizados e apresentados por meio da Tabela A.1, foram analisados 54 conjuntos de dados com 142 tipos de culturas, associando cada patógeno causador de doença ou fenômeno abiótico causador em 7 grupos, sendo eles: vírus, bactérias, fungos, algas, pragas, nematoides e abióticos. Na Figura 4.14 são apresentadas as relações entre o tipo de culturas com grupos de patógenos, quantificando o número de ocorrências de doenças investigadas por seus agentes causadores.

4.3 Análise e Discussão

A análise e a discussão dos resultados obtidos por meio desta RSL é organizada de acordo com as Abordagens Predominantes nos estudos investigados. Basicamente, as propriedades determinantes à compreensão das possibilidades de inovação no processo de identificação de doenças de plantas usando as CNNs perpassam por cada grupo de abordagens de modo particular.

Customização de Camadas

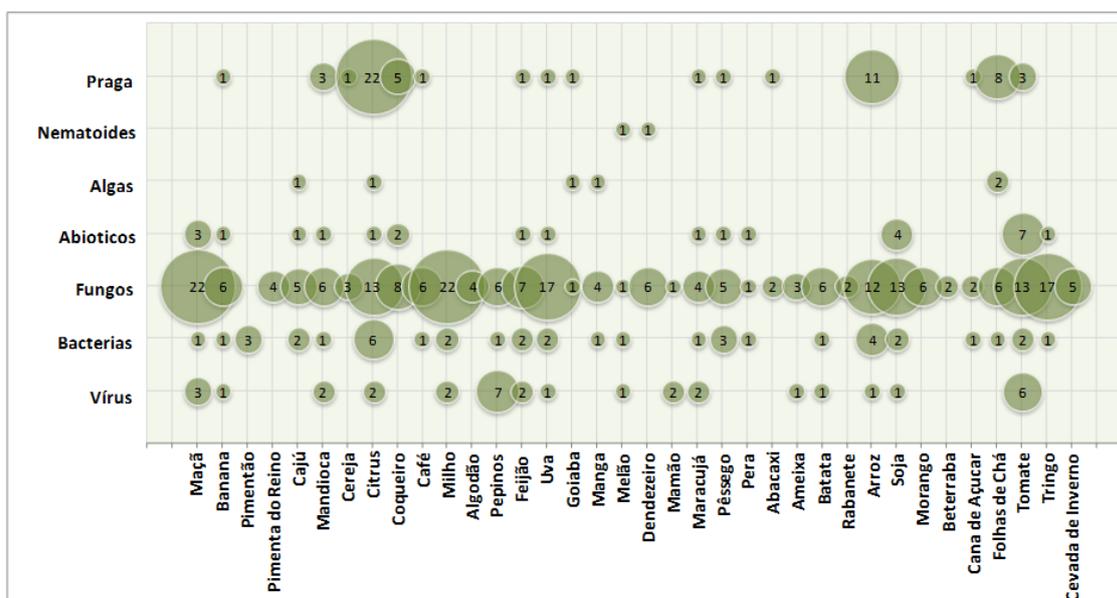


Figura 4.14: Relação do tipo de cultura X grupos de patógenos X ocorrências de doenças investigadas por agentes causadores.

As Arquiteturas de CNNs tradicionais basicamente constroem seus modelos usando 3 tipos de camadas, sendo elas Entrada, Intermediária e Saída. Geralmente, os Frameworks possuem uma lista de camadas padronizadas conforme os modelos de maior relevância já existentes. Entretanto, quando as peculiaridades do problema demandam uma necessidade de operar em um nível de abstração mais baixo surge a necessidade de implementar uma camada customizada ou personalizada que atenda à demanda específica. Por exemplo, ao utilizar a ResNet para um determinado propósito surge a necessidade de adicionar uma camada extra ao final do modelo que normalize a saída do classificador.

Deste modo, Camadas Customizadas são definidas como as que não estão incluídas em uma lista de camadas conhecidas e implementadas nativamente pelos Frameworks por meio dos modelos de maior relevância. É preciso destacar que a implementação de camadas customizadas pautadas em uma arquitetura ou modelo já existente não constitui um nova arquitetura ou modelo de CNN.

Foram sumarizados 36 estudos, tendo a customização de camadas como uma das abordagens predominantes. Destes, 7 estudos customizaram o modelo LeNet (LECUN *et al.*, 1998), que é considerado uma arquitetura clássica das redes CNN.

No estudo S10 são customizados duas estruturas LeNet para o reconhecimento de doença do arroz. A investigação compara o desempenho de três métodos de extração de recursos usando CNN, LBPH e Haar-WT com o mesmo conjunto de dados. A primeira estrutura CNN possui customizações quanto à implementação do classificador softmax, enquanto a segunda estrutura proposta emprega de forma combinada um classificador SVM customizado na camada 8 do modelo. Os resultados sugerem que os recursos extraídos usando CNN são mais discriminativos do que aqueles feitos usando LBPH e Haar-WT. Para os mesmos recursos da CNN, o SVM e o

Softmax obtêm maior precisão (SVM: 95,82%, Softmax: 95,83%).

Já os estudos S34, S78, S81, e S104 são customizações do modelo LeNet para diferentes culturas que implementam a adição de bloco de camadas convolucionais, de ativação e de agrupamento. As customizações realizadas por cada abordagem variam de acordo com o tipo de problema decorrente da cultura investigada e a qualificação do conjunto de dados.

No estudo S48, a abordagem customiza duas estruturas derivadas da LeNet adequando os modelos para imagens com 3 canais de cores e 1 canal de cor (escala de cinza). A abordagem explora a premissa que cada modelo aprende padrões diferentes conforme a extração de recursos face ao número de canais das imagens de entrada. Uma das customizações significativas que diferem os dois modelos está localizada após a camada achatada (*flatten layer*). Ambas as abordagens possuem duas camadas densas; entretanto, o modelo que recebe imagens com 3 canais de cores (RGB) possui 256 unidades ocultas na primeira camada densa. Já o modelo que recebe imagens em escala de cinza é estruturado com apenas 128 unidades ocultas, diminuindo assim o número de parâmetros treináveis do segundo modelo. O modelo com entrada de imagens coloridas alcançou uma precisão de 99,84%, enquanto o modelo com imagens em escala de cinza alcançou uma precisão de 95,54% na identificação de doenças do tomate.

Sun *et al.* (2017) (S111) propõe uma abordagem que customiza um modelo LeNet para aceitar 3 entradas de imagens distintas simultaneamente. Este tipo de configuração de entrada para as CNNs recebe o nome de ‘*Multi-Channel*’ ou ‘*Multi-Path*’ ou ‘*Multi-Input*’. É geralmente utilizada quando o processamento paralelo dos dados de entrada é desejado (KARPATHY *et al.*, 2014). Esses fluxos podem eventualmente se fundir em um nas últimas camadas da rede; assim, é comum que o ponto de concatenação esteja presente antes da primeira camada totalmente conectada, ou seja, o processamento paralelo é concentrado entre as camadas de convolução. Comparado com uma CNN de entrada simples, a acurácia no processo de classificação das flores Gloxinia é melhorada em 5% com o uso das redes Multi-Input.

Entre os estudos investigados, 7 abordagens customizaram o modelo proposto por Krizhevsky *et al.* (2012) denominado AlexNet. O estudo S58 propôs uma abordagem para identificar 4 tipos de doenças recorrentes no Tomate. Foi utilizado um conjunto de dados com 4.293 imagens e o treinamento para reconhecimento das doenças foi implementado utilizando transferência de aprendizagem. As customizações relatadas ocorreram na camada softmax do modelo. A precisão da rede para o reconhecimento das doenças ficou em 95,75%.

As investigações propostas por Pawara. *et al.* (2017) (S26) e Fujita *et al.* (2016) (S119), aplicadas nas culturas de uva e pepino também customizam o modelo AlexNet com ajustes pontuais nas camadas de classificação de acordo com o número de classes e predição de resultados. Os estudos S23 de Alruwaili *et al.* (2019) e S41 de G. e J. (2019) utilizam o mesmo conjunto de dados customizando o modelo AlexNet.

Uma abordagem proposta pelo estudo S18 apresenta uma solução para reconhecimento de uma doença fúngica denominada antracnose, que ataca as folhas e frutos da mangueira. A investigação implementa uma CNN denominada MCNN, validada em um conjunto de dados com

1.070 imagens das folhas da mangueira. A customização implementada na abordagem proposta traz modificações significativas nas camadas convolucionais, camadas de ativação (ReLU) e camadas de pooling. As camadas densas e camada softmax sofrem aprimoramentos para ajustar o desempenho e simplificar o manuseio dos dados. O melhor desempenho da proposta é confirmado com uma acurácia de 97,13%.

A VGG Net é uma arquitetura exitosa proposta por Simonyan e Zisserman (2014) que usa convoluções 3x3 e um maior número de filtros em relação à AlexNet. O desempenho desta arquitetura popularizou seu emprego em soluções para extração de características de imagens, apesar dos seus 138 milhões de parâmetros. No estudo S74 a VGG Net é customizada para identificar doenças incidentes nas frutas e folhas de morango com uma precisão média de 92%.

Uma investigação proposta no estudo S83 customiza um modelo denominado VGG-A com 11 camadas para reconhecer pragas (insetos) presentes na agricultura e silvicultura. A arquitetura VGG Net é simplificada por parâmetros otimizados, como o número de convoluções e a largura das camadas totalmente conectadas. Os resultados experimentais demonstram que a precisão do modelo customizado atinge 99,02% em comparação com outros resultados empregados no mesmo conjunto de dados.

O estudo S112 implementa uma solução para identificação de doenças do rabanete. Um modelo VGG Net é customizado por meio de um classificador softmax juntamente com K-Means. O conjunto de imagens é obtido por meio de VANTs e alcança uma precisão superior a 97,4%. Na abordagem S32 de Jiang *et al.* (2019) é proposta a customização de CNNs para detecção de doenças das folhas da maçã em tempo real.

A GoogLeNet (SZEGEDY *et al.*, 2015a) é o primeiro modelo a utilizar a arquitetura Inception que possibilita aumentar a profundidade e a largura da rede, mantendo o custo computacional constante. Nesta perspectiva, o estudo S71 proposto por Zhang *et al.* (2018) customiza a GoogLeNet aumentando a diversidade de operações de *pooling* e adicionando camadas de ativação (ReLU) juntamente com algumas operações de desistência (*DropOut*). A abordagem propõe o reconhecimento de oito tipos de doenças das folhas de milho, obtendo um modelo significativamente melhor, com uma precisão média de 98,8%.

A investigação proposta por Toda e Okura (2019) no estudo S29 efetua uma customização no modelo Inception V3 (SZEGEDY *et al.*, 2015b) para identificar doenças de plantas no conjunto de dados PlantVillage com 54.306 imagens, 26 doenças e 14 espécies de plantas. A abordagem interpreta os mapas de atenção gerados por meio dos treinamentos, identificando várias camadas que não contribuem para inferência dos resultados. Assim, as camadas consideradas inertes foram removidas do modelo, diminuindo o número de parâmetros em 75% sem afetar a precisão da classificação.

O modelo Inception V4 (SZEGEDY *et al.*, 2017b) foi customizado no estudo S109 de Sulc e Matas (2017) com a adição de uma camada totalmente conectada com 4.096 unidades e uma camada de desistência antes das camadas totalmente conectadas. A abordagem utilizou um conjunto com 9.745 imagens no reconhecimento de 153 espécies das florestas europeias. Os

experimentos demonstraram que a mescla das arquiteturas Inception e ResNet elevaram os resultados comparados a outras CNNs de ponta.

A arquitetura ResNet ou Rede Residual (HE *et al.*, 2016) propõe resolver o problema da profundidade e da degradação da precisão das CNNs utilizando resíduos que são transportados a camadas superiores por meio de conexões de atalho ou salto. Esta implementação tenta garantir que a camada superior tenha um desempenho tão bom quanto a camada inferior, amenizando o clássico problema do perda do gradiente.

O Estudo S105 propõe uma customização de uma CNN usando a arquitetura residual de aprendizagem na identificação de doenças do tomate com o conjunto de dados do PlantVillage. Com a customização de uma CNN residual, a abordagem alcançou uma acurácia na classificação de doenças da folha do tomate de 97.53%. Uma abordagem proposta no estudo S77 explora a customização de uma CNN residual para lidar com a identificação precoce de três doenças endêmicas europeias relevantes do trigo. Os resultados revelam uma melhoria geral da precisão em até 0,96 pontos percentuais na acurácia em comparação com outros estudos.

A proposta do estudo S79 implementa uma função de ativação denominada PENLU em uma ResNet para o reconhecimento de lesões na laranja. O método alcança maior precisão em comparação com o reconhecimento tradicional de lesões. A investigação proposta pelo estudo S30 customiza algumas tipos de CNNs, com destaque para os modelos ResNet50 e ResNet101 na extração profunda de características do conjunto de dados. Posteriormente, são utilizados classificadores tradicionais como SVM, ELM e KNN. Os resultados demonstram que há menor complexidade computacional e ganho na acurácia tendo em vista outras abordagens comparadas.

As redes siamesas são compostas por duas sub-redes com pesos compartilhados que aprendem com as características extraídas de cada sub-rede, comparando o aprendizado dos pares de recursos por meio da distância euclidiana (BROMLEY *et al.*, 1993). No estudo 28 a abordagem customiza uma rede Siamese Net com mudanças significativas nas camadas de ativação e a inclusão de camadas de regularização após cada convolução realizada. Os resultados demonstram um aumento de acurácia na classificação das doenças da uva utilizando o conjunto de dados PlantVillage.

Transferência de Aprendizagem

Conforme fora demonstrado por meio da Figura 4.8, a grande maioria dos conjuntos de dados não possui uma quantidade de imagens que propicie às redes convolucionais um desempenho adequado. Assim, a classificação apropriada de doenças de plantas com pequenos conjuntos de dados usando CNNs ainda é um desafio científico. Foram sumarizados 23 estudos que fazem uso da técnica de transferência de aprendizagem como abordagem principal no processo de investigação.

Os estudos S52, S76 e S90 utilizam a transferência de aprendizagem com a arquitetura AlexNet. Todas as abordagens reportam um aumento significativo da acurácia em comparação com iniciativas que foram treinadas puramente com o conjunto de dados utilizado na investiga-

ção. Destaca-se também que todas as redes foram pré-treinadas utilizando a ImageNet, que se notabilizou como um conjunto de dados com mais de 14 milhões de imagens organizadas em 21.841 subcategorias (DENG *et al.*, 2009).

A arquitetura VGG é pré-treinada com o ImageNet nos estudos S36, S55 e S98. Essas abordagens investigavam a melhoria da acurácia por meio da transferência de aprendizagem com uma arquitetura mais profunda e com tamanho de filtros que possibilitassem a redução do custo computacional. Os estudos S53 e S93 demonstram que a transferência de aprendizado pode fornecer benefícios importantes para automação da identificação e classificação de doenças de plantas, melhorando a acurácia dos modelos. As abordagens apresentam resultados para 4 conjuntos de dados distintos utilizando as arquiteturas AlexNet e VGG pré-treinadas.

Os estudos S59 e S73 utilizam a GoogLeNet para demonstrar fatores que influenciam o uso da transferência de aprendizagem com os modelos pré-treinados para a classificação de doenças de plantas. As investigações salientam fatores como o tamanho e variedade do conjunto de dados, o domínio do problema de aprendizagem, a qualidade e variedade das imagens e a origem destas. Sabidamente, conjuntos de dados formados por imagens capturadas em ambientes controlados ou pré-processadas obtém melhor desempenho de classificação, em detrimento a imagens capturadas em ambiente real. Assim, modelos que fazem uso puramente de conjuntos de dados com imagens obtidas em ambientes controlados degradam a acurácia quando submetidos à classificação de doenças de plantas no ambiente de cultivo.

O estudo S16 emprega um modelo pré-treinado da GoogLeNet demonstrando avanços e estabilidade dos resultados, mesclando o conjunto de dados com imagens controladas e imagens obtidas no ambiente de cultivo. Adicionalmente, propõe técnicas para subdivisão das amostras em áreas de interesse com delimitação das regiões nas imagens. A abordagem proposta no estudo S117 faz uma comparação utilizando as arquiteturas AlexNet e GoogLeNet com modelos pré-treinados e treinamento inicializado do zero para três variações do conjunto de dados PlantVillage. Os resultados demonstraram que, com a técnica de transferência de aprendizagem, a acurácia foi superior em todos os experimentos com diferentes configurações. O estudo S60 propõe uma abordagem comparativa utilizando as arquiteturas AlexNet, VGG, Inception, ResNet, SqueezeNet e DenseNet utilizando três diferentes tipos de treinamento. Em todas as arquiteturas experimentadas o melhor resultado de classificação fora obtido por meio da transferência de aprendizagem. O modelo Inception V3 obteve uma acurácia de 99.76%, tendo assim superados todas as demais iniciativas.

A proposta apresentada no estudo S106 faz uso do modelo Inception V3 pré-treinado com o ImageNet para identificação de doenças da mandioca. Os autores reportam, por meio dos resultados obtidos, que o treinamento usando a transferência de aprendizado alavancou a acurácia da identificação de doenças e diminuiu a complexidade com relação aos métodos tradicionais de extração de características de imagens. A transferência de aprendizado é utilizada na abordagem do estudo S14, juntamente com a arquitetura ResNet. Os autores destacam que alcançaram uma acurácia de 99.37% utilizando o PlantVillage como conjunto de dados e salientam que o ajuste fino dos parâmetros é uma tarefa que demandou atenção. Outra ressalva importante

destacada pelos autores é sobre as características do conjunto de dados com imagens obtidas em ambientes controlados ou pré-processadas, o que fez com que o estudo não pudesse validar imagens capturadas em ambiente real com plantas doentes.

Os estudos S24 e S39 fazem uso da arquitetura MobileNet proposta por Howard *et al.* (2017). A MobileNet pode ser facilmente embarcada em dispositivos móveis por gerar um modelo de tamanho otimizado e propício a sistemas embarcados. As abordagens fazem uso da técnica de transferência de aprendizagem com esta arquitetura, na expectativa de aumentar a acurácia da identificação de doenças de plantas em tempo real diretamente no ambiente da cultura. Entretanto, é preciso destacar que uma das desvantagens das transferências de aprendizagem é o aumento exacerbado do modelo face ao grande volume de dados do ImageNet. Destacando ainda que nem sempre as características incorporadas ao modelo, por meio dos dados da ImageNet, são representativas ao domínio do problema investigado.

Ajustes Fino & Hiperparâmetros

Foram sumarizados 23 estudos que fazem uso da técnica de Ajuste Fino e Hiperparâmetros para a classificação de doenças de plantas. Muitas vezes essa tarefa é subestimada, assim é preciso destacar que encontrar a melhor configuração de ajuste para um determinado domínio de problema e seu conjunto de dado demanda um empirismo face à arquitetura utilizada e as peculiaridades do problema.

A LeafNet é uma abordagem proposta pelo estudo S46 com a finalidade de desenvolver uma arquitetura capaz de identificar os tipos de doenças que atingem as folhas das plantas de chá. A arquitetura foi projetada com filtros extratores de diferentes tamanhos para extração dos recursos. Os resultados demonstram que a LeafNet foi superior no reconhecimento de doenças das folhas de chá em comparação com classificadores MLP e algoritmos SVM, alcançando uma acurácia de 90,16%.

Os estudos S108 e S118 propõem ajuste fino das configurações e hiperparâmetros usando a AlexNet. Os resultados demonstram aumentos significativos na acurácia de classificação e destacam algumas estratégias complementares que podem ser utilizadas no processo de treinamento dos modelos. A abordagem proposta no estudo S101 utiliza os mapas de ativação das camadas convolucionais para implementar uma estratégia de ajuste fino utilizando a VGG. O estudo S92 faz uso da arquitetura VGG para a classificação de doenças da batata aplicando o ajuste fino e definição dos hiperparâmetros.

O estudo S82 propõe um método automático de detecção e classificação de doenças em campos de rabanete usando imagens obtidas por meio de um VANT e pré processadas pelo algoritmo extrator K-Means. Uma arquitetura GoogLeNet tem seus parâmetros e hiperparâmetros ajustados a fim de obter o melhor resultado de classificação. A investigação S67 faz um comparativo entre as arquiteturas AlexNet, GoogLeNet e ResNet. A arquitetura ResNet é eleita e ajuste fino de parâmetros e configurações dos hiperparâmetros são realizados para o reconhecimento e classificação de doenças das folhas de tomate.

No estudo S115 os autores implementam solução para classificação de plantas saudáveis e

doentes utilizando a CaffeNet (JIA *et al.*, 2014). Após o ajuste fino dos parâmetros da rede, a acurácia foi incrementada em 0,5%, chegando a 96,3%. A arquitetura MobilNet foi implementada no estudo S88 para detectar o *Fusarium* em imagens baseadas em microscopia. Após o ajuste fino e configuração dos hiperparâmetros as taxas de detecção das regiões infectadas tiveram melhor desempenho.

O estudo S93 utiliza diferentes arquiteturas aplicando a técnica de ajuste fino e configuração de hiperparâmetros para classificação e identificação de doenças presentes nas folhas do tomate. Em todos os experimentos realizados são reportados significativos avanços na acurácia. Os autores do estudo S51 utilizaram as arquiteturas Multichannel AlexNet e GoogLeNet aplicando técnicas de ajuste fino e hiperparâmetros com pesquisa em grade. A abordagem reporta que houve ganhos significativos de acurácia e diminuição dos modelos treinados, visto que os modelos foram treinados do zero sem transferência de aprendizado.

O estudo S57 faz um comparativo entre as arquiteturas VGG, Inception V4, ResNet e DenseNet, aplicando as técnicas de ajuste fino e configuração de hiperparâmetros. A DenseNet alcança uma acurácia de 99,75%, superando as demais arquiteturas. Os resultados demonstram que a DenseNet tende a melhorar a precisão com número crescente de épocas, sem sinais de sobre ajuste (*overfitting*) e deterioração do desempenho.

O estudo S109 propõe uma investigação para identificação de espécies vegetais usando o reconhecimento de textura com método Ffirst e CNNs de última geração. Os experimentos realizados com dados de treinamento suficientes alcançaram uma acurácia entre 99,0% a 100% no conjunto de dados MEW com 153 espécies de plantas. Os resultados sugerem que, com uma quantidade suficiente de dados, o reconhecimento de folhas segmentadas é praticamente um problema resolvido e o grande desafio é a classificação e reconhecimento de doenças e espécies de plantas com pequeno conjunto de imagens obtidas em ambientes reais que variam significativamente e sofrem com oclusões e plano de fundo caóticos.

Nova Arquitetura

Durante a execução da RSL constatou-se que 25 estudos são sumarizados como proposta de uma nova arquitetura e pautados nas arquiteturas de maior destaque no cenário de aprendizado de máquina. Quanto às novas arquiteturas, foram notadas 3 tendências para a formulação de abordagens. Uma delas é a reformulação estrutural de arquiteturas tradicionais, como a LeNet presentes nos estudos S61, S65, S68, S84, S100, S114 e S116, AlexNet nos estudos S113 e S120, VGGNet no estudo S91 e ResNet nos estudos S6,S43,S75 e S86.

A abordagem S84 proposta por Ghosal *et al.* (2018) utiliza a LeNet como base para uma nova arquitetura explicável, que pode identificar e quantificar as tensões foliares de maneira consistente, rápida e precisa. A investigação busca uma solução para plataformas móveis para detecção em tempo real com precisão de 95,04%, visando uma observação rápida e em larga escala de culturas em ambientes reais de cultivo. O estudo S116 propõe uma investigação com uma nova arquitetura voltada a plataformas móveis. A arquitetura baseada na LeNet conta com 5 camadas convolucionais e funções de ativação customizadas. A investigação faz a captura de

imagens das lesões da maçã e efetua ajustes nos parâmetros da arquitetura proposta, alcançando uma precisão satisfatória de 98,5%.

A abordagem S100 propõe uma arquitetura baseada na LeNet para identificação e diagnóstico automáticos de doenças do arroz. Por meio de 500 imagens naturais de folhas e caules de arroz, uma arquitetura é proposta para identificar 10 doenças comuns da cultura desta planta. A investigação faz inúmeros experimentos com o tamanho dos filtros das camadas convolucionais e algoritmos das camadas de pooling. Os resultados alcançam uma precisão de 95,48%, maior que os modelos convencionais já apresentados. No estudo S61 é apresentada uma arquitetura CNN juntamente com método baseado no algoritmo de quantificação de vetores de aprendizagem (LVQ - Learning Vector Quantization) para detecção e classificação de doenças das folhas do tomateiro. Os resultados avaliam que a abordagem proposta é efetiva quanto ao reconhecimento de quatro diferentes tipos de doenças das folhas do tomate.

O estudo S113 propõe uma abordagem para classificação e identificação precisa de doenças de plantas baseadas na AlexNet. A implementação desenvolvida alcança a redução significativa do número de parâmetros, juntamente com avanços na precisão. Na abordagem S120 é proposta uma nova arquitetura para classificação foliar. A investigação faz uso da Unidade Linear Exponencial (ELU) em substituição à função (ReLU) tradicionalmente utilizada nas arquiteturas AlexNet. Os resultados demonstram que a proposta é eficaz na classificação foliar e concorre com os modelos que refletem o estado da arte.

O estudo S91 faz uma investigação com várias arquiteturas CNNs de ponta para detecção de doenças e pragas em plantas de arroz usando imagens capturadas em ambiente real da cultura. Entre os modelos experimentados, a VGG16 obteve uma precisão de 99,53% no conjunto de dados utilizado. Embora os resultados sejam impressionantes, constatou-se que este modelo não é adequado para dispositivos móveis devido ao seu grande tamanho e número de parâmetros. Uma nova arquitetura é proposta denominada Stacked CNN, que realiza o treinamento em duas etapas para reduzir significativamente o tamanho do modelo, mantendo a precisão. Os resultados alcançam uma acurácia de 95% e reduzem o tamanho do modelo em 98% em comparação a VGG16.

Uma arquitetura denominada HOREsNet é proposta no estudo S86, prospectando um robusto reconhecimento de doenças de plantas. A investigação explora o problema da baixa precisão quanto ao reconhecimento e classificação de doenças de plantas com imagens adquiridas em ambientes reais da cultura e melhora a capacidade de anti-interferência explorando imagens com tamanhos, ângulos de disparo, poses, plano de fundo e iluminações em diferentes aspectos. Os resultados demonstram que a abordagem alcança alta precisão no conjunto de dados com 9.214 imagens com seis tipos de doenças das culturas de arroz e pepino.

O problema dos falsos positivos e o desequilíbrio entre classes é persistente nas investigações usando CNNs, mesmo utilizando as técnicas de aumento de dados e transferência de aprendizagem. O estudo S75 propõe uma abordagem que implementa uma nova arquitetura denominada CNN Filter Bank, dentro de uma solução que prospecta minimizar o problema

dos falsos positivos. Os resultados demonstram que a acurácia chega a 96% no processo de reconhecimento de doenças do tomate, o que representa uma melhoria de 13% em relação ao trabalho anterior desenvolvido pelo autor.

Outra tendência é a mesclagem de blocos entre as arquiteturas, como apresentado no estudo S15 (Yolo-DenseNet) e o estudo S22 (Inception-ResNet). Na proposta de Tian *et al.* (2019) do estudo S15, um método baseado em CNN é desenvolvido para detectar lesões de antracnose implementando técnicas de aumento de dados por meio de uma Rede Adversária Consistente em Ciclos (CycleGan) e desenvolvendo um modelo usando como base a DenseNet para otimizar as camadas de recursos de baixa resolução da Yolo-V3, atingindo uma precisão de 95,57%. Já o estudo S22 de Zhang *et al.* (2019e) propõe uma nova abordagem baseada em CNNs para detecção de doenças na cultura de tomates usando imagens hiperespectrais de alta resolução capturadas com VANTs. O modelo proposto apresenta várias camadas Inception-ResNet para extração de recursos e foi otimizado para estabelecer a profundidade e largura mais adequada da rede. A abordagem trata de dados tridimensionais usando informações espaciais e espectrais para detectar ferrugem amarela nas culturas de trigo com uma precisão de 85%.

A última tendência inclui propostas que implementam arquiteturas projetadas especificamente para a identificação e classificação de doenças de plantas, com ênfase em modelos que usam imagens hiperespectrais, como: GPDCNN (S1), 3D-CNN (S7), PDDN (S11), M-bCNN-CKM (S12), Teacher/Student Architecture (S13), PlantDiseaseNet (S35), Fourier Dense Network (S54) e a OR-AC-GAN (S70).

No estudo S70, proposto por Wang *et al.* (2018), é descrita uma abordagem chamada OR-CA-GAN, desenvolvida para detecção precoce do vírus da murcha do tomate usando imagens hiperespectrais e um classificador auxiliar de remoção externo usando redes adversárias. A proposta integra as tarefas de segmentação de plantas, classificação de espectro e classificação de imagens. Os resultados demonstram que a precisão atinge 96,25% antes que os sintomas visíveis apareçam. Já a abordagem proposta por Nagasubramanian *et al.* (2019) no estudo S7 projeta uma arquitetura CNN chamada modelo supervisionado 3D-CNN para aprender informações espectrais e espaciais de imagens hiperespectrais para classificação de amostras de plantas de soja infectadas pela podridão do carvão. Um método de visualização baseado em mapas de saliência é utilizado para identificar os comprimentos de onda hiperespectrais, que podem contribuir significativamente para a precisão da classificação. O modelo proposto alcançou uma precisão de 95,73%.

Segmentação de Imagens

O estudo S33 propõe um modelo para segmentação semântica baseado em CNN, com objetivo de segmentar a doença fúngica Oídio, que ataca as folhas do pepino. A abordagem alcança uma precisão média de 96,08%, sendo capaz de segmentar a doença nas folhas de pepino no nível de pixel.

A abordagem implementada no estudo S77 para identificação de três tipos de doenças presentes no trigo faz uso da segmentação por super pixel, dividindo a imagem em centenas de su-

per pixels não sobrepostos. A proposta tenta evitar a redução significativa de partes da imagem que podem conter características determinantes à identificação de uma doença em potencial. As imagens alimentam uma ResNet adaptada ao número de classes do conjunto de dados investigado, obtendo uma melhoria equilibrada em comparação a trabalhos anteriores, com 96,0% de acurácia. No estudo S94 mais de 1000 imagens de folhas de chá são usadas para treinar uma rede neural convolucional mais rápida baseada em região (Faster R-CNN). A solução classifica três tipos de doenças do chá com uma precisão média satisfatória e configura-se como uma solução conveniente para identificar essas doenças em condições reais do campo.

Uma investigação que explora o espaço colorimétrico e índices de vegetação para detecção de doenças da videira usando imagens de veículos aéreos é proposta no estudo S97. A abordagem compara o desempenho das CNNs usando diferentes espaços de cores e índices de vegetação a fim de segmentar as doenças nas imagens. Os resultados demonstram uma precisão de 95,8%. O estudo S95 também utiliza imagens aéreas para monitorar a cultura do arroz. A abordagem utiliza a estrutura CapsNet (SABOUR *et al.*, 2017) e todas as imagens de entrada são segmentadas usando o algoritmo super pixel.

O estudo S89 propõe uma abordagem para reconhecer doenças nas folhas do pepino. As imagens dos sintomas foram segmentadas usando o método de abrangente de recursos de cores (CCF - comprehensive color feature). Com o conjunto de dados aumentado, a CNN alcançou uma acurácia de 93,4% no processo de identificação das doenças do pepino. A abordagem proposta no estudo S80 faz uso de imagens obtidas por meio de sensores LIDAR (Light Detection and Ranging) para obter características fenotípicas tridimensionais de plantas de milho. O estudo implementa um método para combinar o uso de R-CNN e algoritmos de crescimento por região para segmentar a planta do milho individualmente a partir das imagens. Os resultados da investigação demonstram que o método é promissor para segmentação individual da planta.

Uma investigação é apresentada no estudo S107 para detecção da região da doença do pingo de mel (watercore) nos frutos da maçã. A abordagem propõe segmentar a região da doença extraíndo representações abrangentes da imagem usando uma FCN para efetuar uma segmentação semântica dos dados. O estudo S103 propõe uma abordagem para classificar a queima das folhas do milho usando várias CNNs treinadas, formando um pipeline computacional de redes neurais. As imagens são segmentadas e extraídos os mapas de calor para o treinamento, classificando plantas doentes e não doentes. Os resultados alcançados demonstram uma acurácia de 96,7%.

Aumento de Dados

Um problema recorrente no campo de aprendizado de máquina é a falta de quantidade significativa de dados de treinamento ou o desequilíbrio das amostra de cada classe nos conjuntos de dados. O Aumento de Dados engloba um conjunto de técnicas que aprimoram o tamanho e a qualidade dos conjuntos de dados de treinamento, de modo que as arquiteturas de aprendizagem profunda possam explorar todas suas potencialidades (SHORTEN; KHOSHGOFTAAR, 2019).

Esses aprimoramentos aumentam artificialmente o tamanho do conjunto de dados de treina-

mento, distorcendo ou super amostrando os dados. Os aprimoramentos de distorção de dados transformam as imagens existentes de modo que seu rótulo seja preservado, como, por exemplo, transformações geométricas e de cores, apagamento aleatório, treinamento antagônico e transferência de estilo neural. Já os aprimoramentos de super amostragem criam instâncias sintéticas e as adicionam ao conjunto de treinamento. Isso inclui misturar imagens, aumentar o espaço de recursos e redes adversárias generativas (GANs) (MIKOAJCZYK; GROCHOWSKI, 2018).

A abordagem proposta no estudo S89, com intuito de minimizar as chances de sobre adaptação (*overfitting*), implementa métodos de aumento de dados para ampliar o conjunto de dados formado com imagens da doença do pepino. Os resultados alcançaram uma eficiência de 93,4%. O estudo S119 também propõe uma abordagem para identificar e classificar doenças do pepino usando dois conjuntos de dados e duas arquiteturas de CNNs (AlexNet e VGGNet). A investigação utiliza três tipos de aumento de dados com pré-processamento das imagens. Os resultados são comparados entre cada modelo de CNN treinada, usando as técnicas de aumento de dados e treinamentos sem usar o aumento de dados.

No estudo S87 os autores propõem uma solução para identificar e classificar doenças que ameaçam a produção de arroz. Um conjunto de dados é composto por 4.511 imagens, 13 categorias de doenças e pragas que são encontradas tipicamente nas plantações de arroz da Indonésia. O método de aumento de dados foi empregado, rotacionando as imagens em 90 e 270 graus, girando as imagens horizontalmente e verticalmente e efetuando um corte aleatório de 50% da imagem original. Após o pré-processamento os dados alimentaram o modelo da CaffeNet, alcançando uma precisão de 87%.

O estudo S72 propõe uma avaliação dos métodos de aumento de dados utilizados na identificação e classificação de doenças de plantas usando CNNs. A investigação tenta esclarecer a correlação entre a distância inicial e a precisão da acurácia por meio da distribuição de Fréchet (distribuição de valores extremos - EVD) como indicadores para determinar se é adequada ou não a utilização da distância entre os dados de treinamento e dados de validação para aplicação dos métodos de aumento de dados.

Uma investigação utilizando a arquitetura MobileNet, juntamente com um conjunto de dados de imagens microscópicas do fungo *Fusarium Oxysporum*, a principal causa da doença de murcha que assola as plantações de banana, implementou várias técnicas de aumento de dados a fim de discriminar o *Fusarium* de outros artefatos presentes nas imagens, como solo e partículas. No estudo S67 os autores exploram as arquiteturas AlexNet, GoogLeNet e ResNet para identificar as doenças das folhas do tomate. A abordagem concilia os métodos de transferência de aprendizagem e aumento de dados e alcança uma acurácia de 97,28%.

Com objetivo de determinar a arquitetura de aprendizado profundo mais adequada para detectar doenças e pragas no tomateiro, a abordagem proposta no estudo S102 investiga as combinações das redes Faster R-CNN, R-FCN e detector SSD (Single Shot Multibox Detector) com os extratores das redes VGG e ResNet. Várias técnicas de aumento de dados foram implementadas para aumentar a precisão e reduzir o número de falsos positivos durante o treinamento.

Os resultados demonstraram que a abordagem é efetiva em reconhecer nove tipos diferentes de doenças e pragas do tomateiro e lidar com cenários complexos que compõem o background das plantas.

Face aos muitos desafios envolvidos na construção de um conjunto de dados com imagens adequadas ao contexto de investigação e ao tamanho do conjunto de dados em relação à acurácia alcançada pelas arquiteturas CNNs, o estudo S59 investiga como o tamanho e a variedade do conjunto de dados impactam na eficácia das técnicas de aprendizado profundo aplicados à patologia das plantas.

Aprendizagem Não Supervisionada e GANs

A aprendizagem não supervisionada é o treinamento de um algoritmo de inteligência artificial usando informações que não são classificadas nem rotuladas, fazendo com que o algoritmo atue nesse conjunto de dados sem orientação. Ao contrário do aprendizado supervisionado, nem sempre é fácil criar métricas sobre o desempenho de um algoritmo de aprendizado não supervisionado. O "desempenho" geralmente é subjetivo e específico do domínio.

As redes adversárias generativas (GANs) são uma classe de modelos de aprendizado não supervisionado, nos quais os dados de treinamento são fornecidos e novas amostras são geradas a partir da mesma distribuição. Esses modelos devem descobrir e aprender com eficiência a essência dos dados fornecidos para tentar gerar dados semelhantes. O benefício a longo prazo desse tipo de modelo é sua capacidade de aprender automaticamente os recursos dos dados fornecidos.

No estudo S63 de Li *et al.* (2018a), uma abordagem usando um conjunto de dados públicos de 54.306 imagens de folhas de plantas doentes e saudáveis coletadas sob condições controladas e uma rede neural convolucional profunda juntamente com um método não supervisionado são utilizados para identificar 14 espécies de culturas e 26 doenças. O modelo treinado atinge uma precisão de 89,83% em um conjunto de testes realizado, demonstrando a viabilidade dessa abordagem.

A abordagem proposta por Pardede *et al.* (2018) apresenta um algoritmo de aprendizagem não supervisionada usando redes convolucionais auto codificadoras, destacando duas vantagens essenciais. A primeira vantagem versa sobre a capacidade de produzir características discriminativas automaticamente, eliminando a necessidade da extração de características de modo artesanal. Na segunda vantagem, o procedimento é realizado de maneira não supervisionada e, portanto, nenhuma rotulagem dos dados é requerida.

Tipos de Culturas e Agentes Causais de Doenças

Os resultados mostraram que, além das abordagens que investigam um conjunto diversificado de culturas, o tomate, juntamente com maçãs, milho, arroz, pepino, trigo, uvas, batata e banana, são os que concentram o maior número de investigações usando CNNs. Tomando como conceito que grãos e cereais são as principais fontes de energia ingeridas pelos seres humanos, ficou evidente a falta de estudos com abordagens que contemplassem principalmente

essas culturas.

Ao analisar os resultados que lidam com os tipos de doenças de plantas, a RSL estabeleceu uma relação entre as doenças investigadas nas abordagens com seus agentes causais. Foi possível observar que a grande maioria das investigações identifica e classifica doenças causadas por fungos. Esse fato pode ser explicado porque se estima que 70% das principais doenças de plantas são causadas por esse grupo de patógenos (AGRIOS, 2005; GODFRAY *et al.*, 2016).

As doenças causadas por vírus e bactérias, embora não tenham abordagens quantitativas relevantes, estão representadas em uma diversidade de culturas encontradas nos conjuntos de dados. As doenças causadas pelas algas são representadas pela cultura de caju, citros, goiaba, morangos e folhas de chá. Também identificou-se a classificação de doenças causadas por fatores abióticos em investigações usando CNNs.

O número inexpressivo de abordagens que classificam doenças causadas por nematoides chamou nossa atenção. Esses patógenos são considerados um dos maiores riscos fitossanitários nas principais culturas de grãos e cereais do mundo. Certamente, a ausência de investigações usando CNNs para identificar sintomas causados por nematoides pode ser associada à similaridade sintomática das características apresentadas pela planta com outras doenças ou particularidades que não são expressas visivelmente nas folhas e caules.

4.4 Considerações Finais

Neste capítulo buscou-se caracterizar o estado da arte sobre os sistemas de identificação e classificação de doenças de plantas usando CNNs. Notoriamente, os desafios que permeiam essa temática despertam na comunidade de investigadores relevante interesse.

A diversidade de problemas e as especificidades dos cenários do mundo real dificultam catalogar semanticamente os dados em conjuntos representativos com um número suficiente de amostras rotuladas. Esse problema se torna um gargalo relevante e desafiador para tornar os métodos de aprendizado de máquina mais aplicáveis na prática quanto à identificação de doenças de plantas nos ambientes de cultivo.

Os resultados demonstram avanços significativos quanto ao uso das CNNs em processos preditivos de doenças de plantas. Foi possível observar que as arquiteturas tradicionais, aliadas aos métodos de otimização, apesar da complexidade de conjunto de dados compostos por imagens capturadas em ambientes reais de cultivo, apresentam acurácias relevantes. Observou-se também uma tendência crescente de abordagens que propõem novas arquiteturas CNNs pautadas no processo de identificação de doenças de plantas. Entretanto, notou-se uma especialização dessas arquiteturas em detrimento ao tipo de cultura investigada.

Adicionalmente, é possível destacar algumas situações que são desafiadoras, carentes de propostas ou soluções, sendo elas:

- Embora a diversidade sintomática causada por um mesmo patógeno seja um fator complicador para a inferência da detecção e reconhecimento de doenças de plantas, notou-se uma carência de estudos que proponham arquiteturas CNNs que tenham como objeto a doença, independentemente do tipo da cultura;
- 95% dos estudos fazem uso de imagens no padrão RGB com espectro visível. Apenas 5% utilizam imagens multiespectrais, hiperespectrais ou microscópicas;
- Ainda que a identificação de doenças pelos sintomas foliares seja relevante e predominante nas abordagens, a carência de estudos que fazem uso de imagens microscópicas para identificação dos patógenos e consequente diagnose das doenças de plantas é notória;
- Os estudos reportam apenas investigações que abordam doenças bióticas causadas por fungos, vírus e bactérias. As buscas realizadas durante a execução da RSL não retornaram qualquer estudo para identificação de doenças causadas por nematoides, seja por sintomas foliares ou métodos microscópicos que utilizam a morfologia do patógenos;
- Há uma diversidade de estudos que customiza camadas de arquiteturas tradicionais. Entretanto, constatou-se uma ausência de abordagens que customizassem topologias mesclando arranjos de blocos e camadas convolucionais.

Capítulo 5

Reconhecimento de Doenças de Plantas a partir de Imagens Digitais usando Redes Neurais Convolucionais Multicanais

Dentro da estrutura metodológica proposta, conforme detalhado nas Seções 3.3.1 e 3.4, este Capítulo foi um experimento inicial da abordagem proposta.

O processo de investigação deste estudo foi norteado pelo trabalhos de Mohanty *et al.* (2016) e Ferentinos (2018). O estado da arte demonstrado pelos autores possibilitou a exploração de algumas lacunas do conhecimento, nas quais os esforços desta investigação foram centrados não apenas na melhoria da precisão alcançada pelos métodos propostos anteriormente, mas também no desenvolvimento e implementação de uma abordagem capaz de alcançar resultados mais consistentes.

Os estudos de Mohanty *et al.* (2016) e Ferentinos (2018) fazem uso de técnicas de aprendizagem profunda na agricultura, em particular no diagnóstico de doenças de plantas com sintomas foliares. As abordagens propostas utilizaram duas arquiteturas populares, a AlexNet (KRIZHEVSKY *et al.*, 2012) e a GoogLeNet (Inception V1) (SZEGEDY *et al.*, 2015a), ambas projetadas no contexto do desafio de reconhecimento visual em escala (ILSVRC - Scale Visual Recognition Challenge) (RUSSAKOVSKY *et al.*, 2014) para o conjunto de dados da ImageNet (DENG *et al.*, 2009).

Basicamente, as arquiteturas mencionadas nos trabalhos de seus respectivos autores implementam suas abordagens pautadas em um versão do conjunto de dados PlantVillage (HUGHES; SALATHÉ, 2016) com imagens coloridas fazendo uso da transferência de aprendizagem como alavancador de seus resultados. Mohanty *et al.* (2016) afirma por meio de seus experimentos que outras versões com imagens em escala de cinza ou segmentadas não contribuem para melhoria na precisão dos modelos.

5.1 Método Proposto

A abordagem proposta neste estudo combina uma rede neural convolucional multicanal (M-CNN) implementando as mesmas arquiteturas utilizadas nos estudos de Mohanty *et al.* (2016) e Ferentinos (2018), condicionando uma justa comparação entre as abordagens utilizadas como referência e a proposta nesta investigação.

Adicionalmente, o trabalho desenvolvido explora outras versões do conjunto de dados PlantVillage, com imagens em escala de cinza e segmentadas, otimizando o acurácia sem implementar as técnicas de transferência de aprendizagem, o que reduz significativamente o tamanho dos modelos treinados. Os experimentos foram conduzidos com um rateio de 80% das imagens utilizadas para treinamento e 20% para testes; essa relação produziu os melhores resultados nos experimentos dos trabalhos de referência.

Conjunto de Dados

A abordagem proposta utiliza o conjunto de dados PlantVillage, fornecido por Hughes e Salathé (2016), contendo 54.306 imagens de folhas de plantas e 38 classes diferentes, cada uma correspondendo a uma doença de diferentes culturas. Cada classe possui três versões diferentes: imagens coloridas originais, imagens em escala de cinza e imagens segmentadas. A Figura 5.1 apresenta um exemplo por meio de suas sub-figuras de (A) Imagens coloridas; (B) Imagens em escala de cinza e (C) Imagens segmentadas das três diferentes versões do conjunto de dados utilizado.

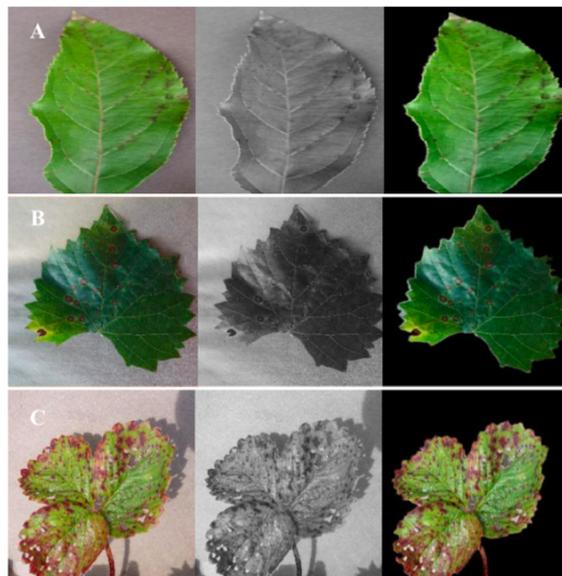


Figura 5.1: Exemplos do Conjunto de Dados PlantVillage com imagens de Doenças de Plantas em três versões (Colorida, Escala de Cinza e Segmentada): (A) Sarna da macieira. (B) Uma amostra da podridão negra nas folhas da uva. (C) Queima das folhas do morango

CNN Multicanais

As redes neurais convolucionais multicanais (M-CNNs) geralmente são utilizadas quando há uma possibilidade de processamento paralelo dos dados (KARPATHY *et al.*, 2014). Esses fluxos podem eventualmente se fundir em uma das últimas camadas da rede. As relevantes contribuições de Baccouche *et al.* (2011) e Ji *et al.* (2013) são como o ponto de concatenação entre os canais estarem presente antes da primeira camada totalmente conectada da rede, ou seja, o processamento paralelo é concentrado entre as camadas de convolução. Outra vantagem do uso das M-CNNs, destacada por Karpathy *et al.* (2014), consiste na redução da dimensionalidade da entrada da rede, o que ajuda a diminuir o tempo de processamento.

Na Figura 5.2 é apresentado um layout da arquitetura geral de uma rede convolucional com dois canais de entrada (M-CNN). Cada canal recebe uma versão diferente do conjunto de dados, gerando 3 conjuntos com diferentes combinações de entrada: Versão 1: Colorida + Escala de Cinza; Versão 2: Colorida + Segmentada e Versão 3: Escala de Cinza + Segmentada.

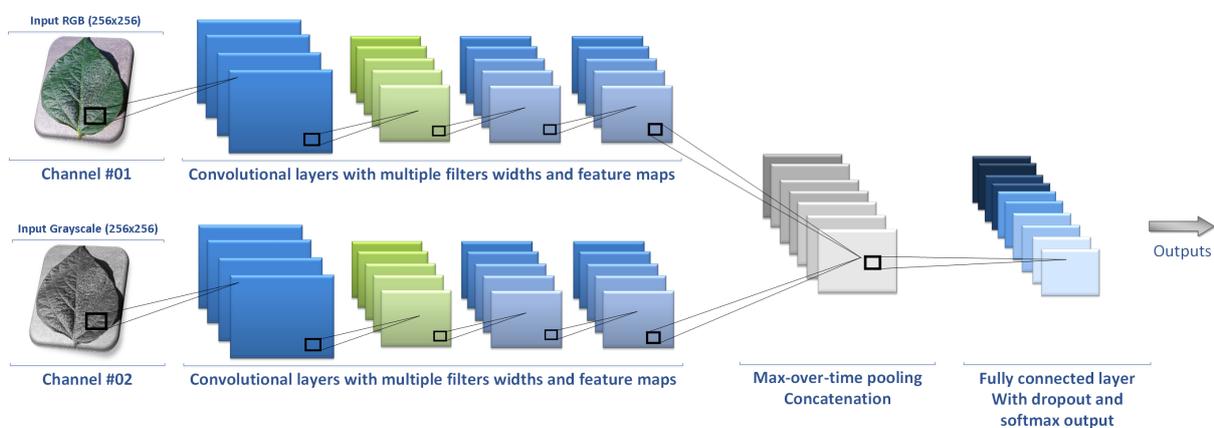


Figura 5.2: Uma ilustração da arquitetura genérica de uma rede convolucional multicanal. O modelo generaliza uma estrutura com dois canais de entrada e identifica os segmentos relevantes da arquitetura.

O objetivo por trás do uso de redes M-CNNs é observar se a rede neural era capaz de produzir melhores resultados com a adição de informações com novas características. A proposta deste estudo implementa a técnica de fusão tardia (*laterfusion technique*) (KARPATHY *et al.*, 2014), na qual as redes de canais separados com parâmetros compartilhados são mescladas na primeira camada totalmente conectada, computando recursos globais comparando as saídas de ambos os fluxos.

Durante a realização dos experimentos, adotou-se a estratégia de otimização dos hiperparâmetros implementando uma busca em grade para ajustar os valores de acordo com o modelo de aprendizagem utilizado. Os valores de referência que foram tomados como ponto de partida para exploração da faixa de valores são apresentados nos trabalhos de (MOHANTY *et al.*, 2016). Todos os modelos CNN e M-CNN iniciaram os treinamentos utilizando os hiperparâmetros apresentados por meio da Tabela 5.1.

Na abordagem proposta, as camadas de abandono (*dropout layers*) são adicionadas antes e depois da fusão que ocorre na arquitetura M-CNN. Foi introduzida uma camada de abandono

Tabela 5.1: Hiperparâmetros de Treinamento

Modelo	Hiperparâmetros	Valores
AlexNet	Learning rate	0.01
	Momentum	0.9
	Weight decay	1e-6
	Batch size	128
GoogleNet	Learning rate	0.0205
	Momentum	0.9
	Weight decay	0.0005
	Batch size	16

entre as camadas de *pooling*, que precede a fusão das redes e, após a fusão, para as primeiras duas camadas totalmente conectadas. Adicionalmente, com base nas observações sobre os valores de perda (loss), ajustou-se a fração de entradas para 0 a cada atualização durante o tempo de treinamento, a fim de evitar o *overfitting*.

De acordo com as recomendações de Pan e Yang (2010), e tendo em vista que o conjunto de dados utilizado possui peculiaridades únicas ao seu conjunto de objetos e as imagens de entrada possuem diferentes dimensões do modelo pré-treinado, o uso da técnica de transferência de aprendizado fora repensado. Deste modo, um dos desafios da abordagem era superar a acurácia do estado da arte demonstrados por Mohanty *et al.* (2016) e Ferentinos (2018), sem implementar a técnica de transferência de aprendizagem.

5.2 Resultados

Esta Seção apresenta os resultados obtidos durante a realização do estudo e elabora uma comparação com os resultados dos trabalhos de Mohanty *et al.* (2016). Para uma melhor visualização dos resultados, as arquiteturas AlexNet e GoogLeNet com arquitetura multicanal são nomeadas como M-AlexNet e M-GoogLeNet. Cabe destacar que todos os experimentos realizados com as arquiteturas propostas iniciaram o processo de treinamento do zero (*from scratch*).

Na Tabela 5.2 são apresentados os resultados de acordo com metodologia proposta para arquiteturas com canal único (*single channel*) e multicanal (*multichannel*) após o cálculo da média *F1 Score* de cada rede, com o melhor resultado em destaque.

5.3 Discussão

No estudo de Mohanty *et al.* (2016), seus resultados permitem concluir que as versões do conjunto de dados em escala de cinza e segmentadas não colaboram para melhorar a precisão

Tabela 5.2: Comparação das Pontuações $F_1 score$ das arquiteturas investigadas.

	Modelos	Tipo Dataset	F_1 Score
Mohanty et al. (2016)	AlexNet (transfer learning)	Color	0.9927
	GoogleNet (transfer learning)	Color	0.9934
M-CNN	AlexNet (from scratch)	Color	0.9873
	GoogleNet (from scratch)	Color	0.9940
	M-AlexNet (from scratch)	Versão 1	0.9959
		Versão 2	0.9920
		Versão 3	0.9923
	M-GoogleNet (from scratch)	Versão 1	0.9955
		Versão 2	0.9938
		Versão 3	0.9941

quando comparadas à versão colorida. Nos experimentos realizados foi possível observar que para modelos com arquitetura de canal único essa suposição é consistente, mesmo com modelos treinados a partir do zero e com otimização dos hiperparâmetros.

Entretanto, quando foram utilizadas as redes M-CNN, mesclando as três versões do conjunto de dados em uma arquitetura com 2 canais, explorou-se a extração exclusiva de características de cada versão do conjunto de dados. A abordagem proposta demonstrou que cada versão do conjunto de dados enriquece o aprendizado dos modelos.

Os resultados dos experimentos usando as redes M-CNN demonstram que a abordagem proposta é equivalente às arquiteturas tradicionais avaliadas, conforme a métrica $F_1 Score$. Considerando o uso das três versões do conjunto de dados PlantVillage (Coloridas; Escala de Cinza e Segmentadas), o melhor resultado dos experimentos realizados usando as M-CNN supera em 0,25% o valores de referência estabelecidos como estado da arte e a própria melhoria alcançada na abordagem proposta com redes de arquitetura de canal único.

Os melhores resultados com as M-CNN foram obtidos com a versão 1 (Coloridas + Escala de Cinza) do conjunto de dados. Os resultados gerais das M-CNNs foram consistentes com qualquer um dos diferentes modelos utilizados.

É preciso destacar que o conjunto de dados PlantVillage utilizado nos experimentos, possui uma hegemonia das imagens e um balanceamento de classes que justifica a alta pontuação nas métricas de avaliação. A boa formação desse conjunto de dados, pautada basicamente em imagens coletadas em ambientes controlados, de plantas saudáveis ou doentes foi um limitador significativo tanto aos experimentos realizados quanto as possibilidades de evolução da abordagem.

O modelo AlexNet tradicional, possui 58.361.738 parâmetros, enquanto o modelo customizado denominado M-AlexNet, totaliza 116.723.476 parâmetros.

5.4 Considerações Finais

Neste capítulo, exploraram-se as potencialidades das redes neurais convolucionais já evidenciadas pela literatura para identificar doenças de plantas por meio de amostras de plantas saudáveis e doentes. Primordialmente, exploraram-se as lacunas do conhecimento destacadas por Mohanty *et al.* (2016), comparando seus resultados por meio de uma abordagem que faz uso de redes neurais convolucionais com multicanais de entrada. Os experimentos realizados neste estudo adotaram a mesma estratégia de Mohanty *et al.* (2016) ao executar o treinamento com três versões pré-processadas do conjunto de dados PlantVillage.

Na primeira etapa da abordagem proposta foram alcançados incrementos de acurácia nas redes de arquitetura com canal único, apenas otimizando os hiperparâmetros e ajustando as camadas de abandono de acordo com as características do conjunto de dados para minimizar o *overfitting*. Destaca-se que, mesmo com as evidências citadas pela literatura de ganhos utilizando as técnicas de transferência de aprendizado, a abordagem proposta optou por treinar os modelos a partir do zero e demonstrar a possibilidade de customização dos ganhos no processo de aprendizagem.

A abordagem utilizando as redes M-CNNs mostrou-se promissora, utilizando uma arquitetura customizada com dois canais de entrada e processamento paralelo, aumentando assim o mapa de características. É preciso destacar que o conjunto de dados PlantVillage disponível não possuía imagens de plantas em ambientes de cultivo, portanto, os resultados da abordagem contemplam apenas os testes realizados especificamente sob as imagens pré-processadas e adquiridas em ambientes controlados.

Capítulo 6

***NEMANet*: Um Modelo de Rede Neural Convolucional para Identificação de Fitonematoides em Imagens Microscópicas**

A expansão das áreas cultivadas no país, sobretudo com práticas de rotação de culturas inadequadas ou inexistentes, tem proporcionado aumento na incidência de fitonematoides (FERRAZ *et al.*, 2010). Estas pragas são consideradas inimigos ocultos dos produtores porque nem sempre é possível visualizá-las ou identificá-las no campo (FERRAZ *et al.*, 2010; FERRAZ; BROWN, 2016). Na cultura da soja, os principais nematoides são o Nematóide de Galha (*Meloydogine incognita* (KOFOID & WHITE, 1919) CHITWOOD, 1949; *Meloydogine javanica* (TREUB, 1885) CHITWOOD, 1949), o Nematóide das Lesões Radiculares (*Pratylenchus brachyurus* (GODFREY, 1929) FILIPJEV & S. STEKHOVEN, 1941), o Nematóide de Cisto da Soja (*Heterodera glycines* (ICHINOHE, 1952)) e o nematóide reniforme (*Rotylenchulus reniformis* (LINFORD & OLIVEIRA, 1940)). No entanto, existem outros nematoides associados à cultura da soja, como *Mesocriconema* sp., *Scutellonema brachyurus* (STEINER, 1938) ANDRASSY, 1958, *Trichodorus* sp., *Helicotylenchus* sp., *Aphelenchoides besseyi* (CHRISTIE, 1942), *Tubixaba* sp. e outros (TIHOHOD, 1997; FERRAZ; BROWN, 2016)

O manejo dos fitonematoides é mais difícil quando comparado a outros fitopatógenos, visto que a maioria deles habita inicialmente o solo e ataca as raízes das plantas (STIRLING; ATKINSON, 1992). Os métodos utilizados no controle são variados e envolvem o princípio da exclusão, controle químico, manejo do solo, variedade de plantas resistentes, solarização do solo e controle biológico, com graus variados de sucesso na proteção das plantas. No entanto, independentemente do método escolhido para o manejo, é de suma importância a identificação da espécie de nematóide que está causando o problema na área de cultivo (FERRAZ *et al.*, 2010).

Os fitonematoides podem causar diversos sintomas na soja, podendo infectar as raízes e a parte aérea das plantas. No entanto, muitas vezes esses sintomas são confundidos por técnicos ou agricultores, pois apresentam muita similaridade com aqueles oriundos de fatores bióticos e

abióticos, a exemplo de vários sintomas causados por fungos, bactérias e vírus fitopatogênicos, deficiência nutricional, déficit hídrico e fitotoxidez oriundas de alguns defensivos agrícolas. Além desses fatores, o tamanho microscópico desses organismos dificulta a identificação no campo, sendo necessários procedimentos laboratoriais para a diagnose correta (COYNE *et al.*, 2007; FREITAS *et al.*, 2014).

Os sintomas na parte aérea das plantas, na maioria das vezes, são facilmente confundidos com outras causas, entre elas, deficiência de nutrientes, ataque de pragas e patógenos, estiagem e compactação do solo (COYNE *et al.*, 2007). De acordo com a Sociedade Brasileira de Nematologia (SBN, 2019), as perdas variam em média entre 5 e 35%, dependendo do tipo de cultivo. Em casos mais severos, as perdas podem ser ainda maiores.

É preciso destacar que, na grande maioria os métodos utilizados para identificar e classificar nematoides é feita por mostra visual, com base nos caracteres morfológicos do nematoide, e conta exclusivamente com *know-how* do profissional especialista para o reconhecimento adequado (TIHOHOD, 1997; COYNE *et al.*, 2007).

Nos laboratórios de análise nematológicas, a identificação dos gêneros e espécies dos fitonematoides é feita com base em caracteres morfológicos e morfométricos dos órgãos e de regiões do corpo das fêmeas e dos machos. Após a visualização do espécime, para a classificação são utilizadas chaves dicotômicas com centenas de caracteres morfológicos, conforme proposta por Mai *et al.* (1996), Tihohod (1997) e Gonzaga (2006). Este é um processo bastante trabalhoso e conta muito com o *know-how* do profissional especialista. Além disso, a quantificação dos fitonematoides nas amostras, aliada à identificação, torna o processo bastante demorado e passível de erros.

Nesse contexto, o desenvolvimento de métodos automáticos capazes de identificar esses fitonematoides de forma rápida e confiável é convincente. Soluções automatizadas para identificar doenças de plantas usando imagens e aprendizado de máquina, especialmente redes neurais convolucionais (CNNs), forneceram avanços significativos para maximizar a precisão do diagnóstico correto (KAUR *et al.*, 2019; THYAGHARAJAN; RAJI, 2019; ABADE *et al.*, 2021a).

As Redes neurais convolucionais são uma classe de modelos de aprendizado de máquina atualmente no estado da arte em muitas tarefas de visão computacional, incluindo classificação e detecção de objetos. Parte desse sucesso reside na capacidade de uma CNN de realizar extração automatizada de recursos, em oposição aos métodos clássicos que podem exigir recursos artesanais (KRIZHEVSKY *et al.*, 2012; KHAN *et al.*, 2020).

Diante do exposto, fatores como o labor estressante do processo manual de identificação e classificação dos nematoides, a dependência de mão de obra extremamente especializada, o excessivo tempo gasto e a alta taxa de erros durante o processo de análise podem ser amenizados com a utilização das redes de aprendizagem profundas na identificação e classificação desses fitonematoides.

6.1 Método Proposto

O fluxo de trabalho para o método proposto é descrito por meio da Figura 6.1, e as subseções a seguir apresentam cada detalhe envolvido no processo de construção da abordagem.

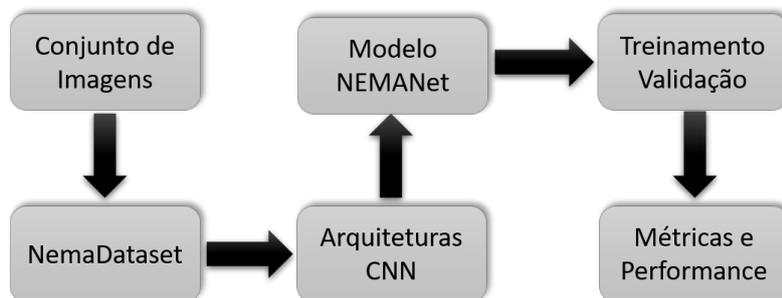


Figura 6.1: O fluxo de trabalho do método para a identificação de nematoides da cultura da soja usando a Rede Neural Convolutiva NEMANet.

6.1.1 Conjunto de Imagens

Foi criado um conjunto de dados denominado NemaDataset, contendo 3.063 imagens microscópicas das cinco espécies de fitonematoides com maior relevância de dano para cultura de soja. O NemaDataset foi construído em três etapas distintas. A primeira etapa detalha todo o protocolo de extração dos fitonematoides presentes no solo e amostras de raízes enviadas ao laboratório. Na segunda etapa é detalhada a captura de imagens microscópicas. Por fim, na última etapa, um especialista em nematologia classifica os espécimes em cada imagem, gerando seus respectivos rótulos.

6.1.1.1 Etapa de Extração dos Nematoides

O processo de extração dos patógenos nas amostras de solo e raízes seguiu os protocolos propostos por Coolen and D'herde (1972) e Jenkins (1964). Este método permite a extração de espécimes móveis, imóveis ou mortos.

Na Figura 6.2 é possível seguir cada detalhe do protocolo de extração desses fitonematoides das amostras de solo e raiz. O protocolo de extração faz uso da diferença de densidade para garantir a separação dos nematoides dos demais objetos presentes nas amostras. Entretanto, a câmara de contagem é populada por uma diversidade de espécimes que devem ser identificadas conforme o interesse da investigação.

6.1.1.2 Etapa de Captura das Imagens

Após a etapa de extração, o processo de captura da imagem utiliza o microscópio e a câmera, com as especificações apresentadas na Tabela 6.1.

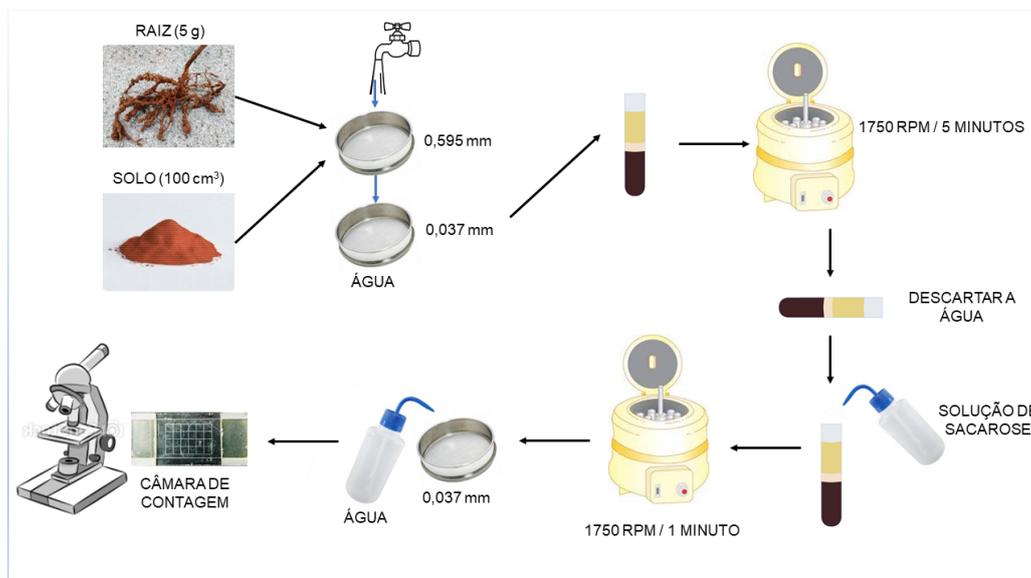


Figura 6.2: Ilustração do protocolo utilizado no método de extração de nematoides.

Tabela 6.1: Especificações técnicas do microscópio e da câmera

Microscópio		Câmera	
<i>Modelo</i>	Binocular	<i>Digital camera resolution</i>	16 MegaPixels
<i>Head</i>	Observation head, inclined 30° and rotatable 360°	<i>Sensor Size</i>	2.33 inches
<i>Eyepieces</i>	WF 10x/18mm, secured by screw	<i>Sensor technology</i>	CMOS Panasonic
<i>Nosepiece</i>	Fixed quadruple nosepiece	<i>Pixel size</i>	1.335 x 1.335 micro
<i>Objectives</i>	N-PLAN 4x/0.10, 10x/0.25, 40x/0.65, 100x/1.25	<i>Image size</i>	5120 x 3840
<i>Stage</i>	Double layer with mechanical sliding stage	<i>Magnification</i>	21x

As amostras extraídas são dosadas em uma câmara de contagem de *Peters*. O volume de preenchimento da câmara é de 2 ml, e o volume da área de visualização é de 1 ml, ou seja, o retângulo reticulado (Quadrado) mede 18,0 mm x 28,0 mm vezes a altura da câmara (2,0 mm), portanto temos 1,008 mm ao cubo, ou aproximadamente 1 ml. A área de visualização foi percorrida mecanicamente pelo operador do microscópio, que fotografou cada espécime de nematoide encontrado durante o processo de captura das imagens. Nesse primeiro momento, foram captadas 8.987 imagens não classificadas.

6.1.1.3 Etapa de Classificação e Rotulagem

Análises fitonematológicas foram realizadas para identificar gêneros e espécies com base nas características morfológicas e morfométricas dos órgãos femininos, masculinos e regiões do corpo. Todas as imagens foram separadas e classificadas de acordo com cada espécie em uma classe específica, organizada em uma estrutura de diretórios, na qual cada imagem capturada tinha dimensões espaciais de 5120 × 3840 pixels com 72 dpi. Cabe ressaltar que as configurações de captura das imagens são idênticas ao protocolo de rotina de um laboratório para identificação e classificação por análise nematológica. Assim, o conjunto de imagens captadas replica as mesmas condições visuais obtidas por um especialista em um laboratório de rotina. A Tabela 6.2 apresenta a distribuição das imagens por espécies.

Tabela 6.2: Composição do conjunto de dados de imagens com as espécies de fitonematoides.

Classes (especies)	Nº de Images	Resolução pixel / dpi	Ampliação
<i>Helicotylenchus dihystra</i>	556	5120×3840 72 dpi	105×
<i>Heterodera glycines</i> (J2)	605		
<i>Meloydogine incognita</i> (J2)	635		
<i>Pratylenchus brachyurus</i>	635		
<i>Rotylenchulus reniformis</i>	632		

Geralmente, as medidas de controle desses patógenos baseiam-se na quantificação das populações por espécie. Devido às características parasitárias dos fitonematoides investigados, as imagens dessas espécies foram capturadas apenas na forma vermiforme. Espécimes em sua forma globosa são geralmente destruídas no processo de extração (MACHADO *et al.*, 2019). Na Figura 6.3 são apresentadas cinco imagens capturadas por microscopia digital, representando as espécies investigadas.

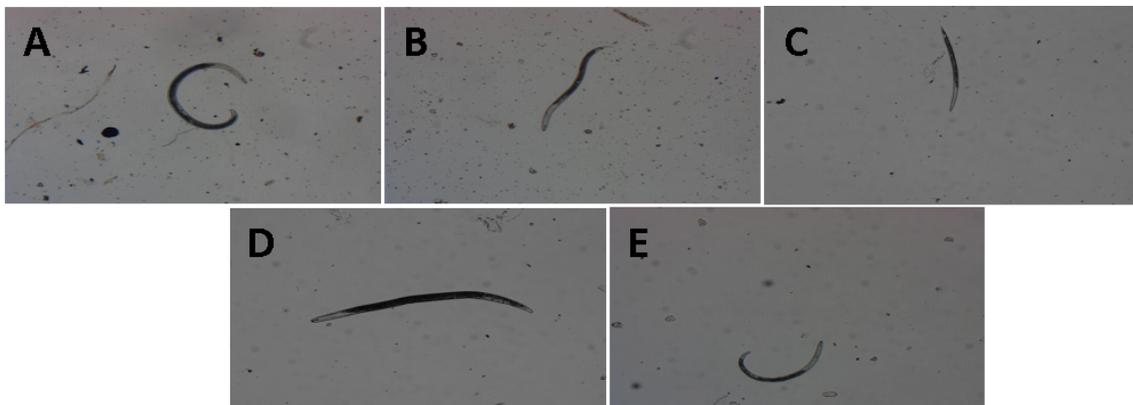


Figura 6.3: Exemplos de imagens no estado original de fitonematoides no NemaDataset: (A) *Helicotylenchus dihystra* (B) *Heterodera glycines* (J2). (C) *Meloydogine incognita* (J2). (D) *Pratylenchus brachyurus*. (E) *Rotylenchulus reniformis*.

6.1.2 Pre-processamento do NemaDataset

A imagem óptica é uma distribuição contínua da intensidade da luz em uma superfície bidimensional. Esta projeção bidimensional é limitada em resolução e está sujeita a distorção e ruído introduzidos pelo processo de captação da imagem. Os principais fatores que podem degradar uma imagem no processo de digitalização são os seguintes: perda de detalhes, ruído, *aliasing*, sombreamento, não linearidade fotométrica e distorção geométrica (WU *et al.*, 2008; ROEDER, 2019). Se cada uma delas for mantida suficientemente baixa, as imagens digitais obtidas por meio da microscopia podem ser utilizadas para o processo de treinamento das CNNs, garantindo qualidade no nível de inferência.

Deste modo, nesta etapa de pré-processamento, os esforços foram focados em preservar um nível adequado de detalhes e relação sinal-ruído, evitando *aliasing* e fazendo isso com níveis aceitáveis de sombreamento, não linearidade fotométrica e distorção geométrica. Além disso,

foi definida uma região de interesse nas imagens, tornando o patógeno centralizado nas imagens de entrada para o treinamento das CNNs.

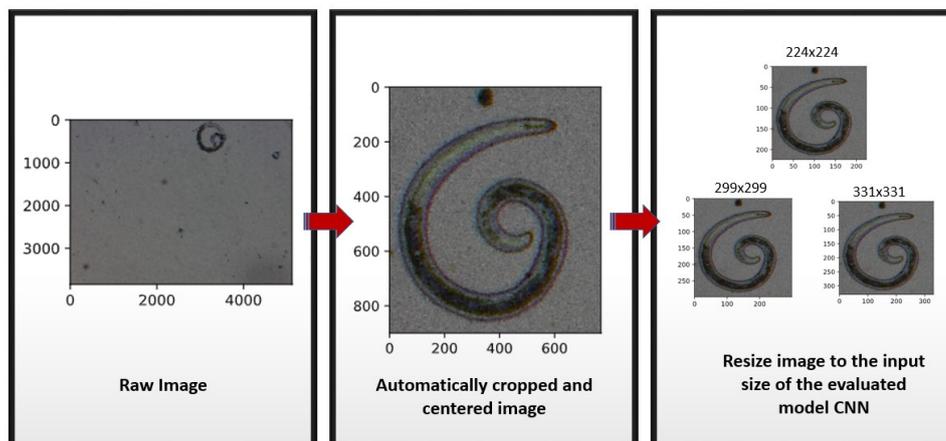


Figura 6.4: Visão geral das etapas de pré-processamento - Nemadataset.

Na Figura 6.4 é apresentada a evolução das etapas de pré-processamento das imagens brutas capturadas pela câmera adaptada ao microscópio. A partir da imagem bruta, um processo de recorte e centralização do objeto de interesse foi aplicado para reduzir significativamente características que não são determinantes à identificação do patógeno. Um algoritmo proposto por Rojas-Albarracín *et al.* (2010) foi adaptado para determinar os pontos iniciais da cabeça e cauda do patógeno, delimitando assim a área de recorte.

6.1.3 Arquiteturas CNNs Avaliadas

As CNNs têm sido aplicadas com grande sucesso para tarefas de visão computacional de alto nível, como classificação e reconhecimento de objetos. Neste estudo, foi avaliado o desempenho das 13 arquiteturas populares disponibilizadas pela API Keras. Na Tabela 6.3, os modelos são apresentados com suas principais características.

Tabela 6.3: Arquiteturas Populares Avaliadas

Modelos	Tamanho	Parâmetros	Profundidade	Imagem Tam (H×W)	Hiperparâmetros				
					Algoritmo Otimização	Batch size	Momentum	Weight decay	Learning Rate
Xception	88 MB	22,910,480	126	229 × 229		32			
VGG16	526 MB	138,357,544	23	224 × 224		32			
InceptionV3	92 MB	23,851,784	159	299 × 299		32			
ResNet50	98 MB	25,636,712	-	224 × 224		100			Base lr = 0.001
ResNet101	171 MB	44,707,176	-	224 × 224		64			
ResNet152	232 MB	60,419,944	-	224 × 224		64		1e-5	Max lr = 0.00006
InceptionResNetV2	215 MB	55,873,736	572	299 × 299	SGD	32	0.9	~	
DenseNet121	33 MB	8,062,504	121	224 × 224		32		1e-6	Step size = 100
DenseNet169	57 MB	14,307,880	169	224 × 224		32			
DenseNet201	80 MB	20,242,984	201	224 × 224		32			Mode = triangular
EfficientNetB0	29 MB	5,330,571	-	224 × 224		100			
EfficientNetB3	48 MB	12,320,535	-	320 × 320		32			
NASNetLarge	343 MB	88,949,818	-	331 × 331		16			

6.1.4 Modelo Proposto *NEMANet*

A extração e classificação das características foram realizadas nas imagens do microscópio extraídas na etapa de pré-processamento dos dados (6.1.2) com um novo modelo denominado *NemaNet* (ABADE *et al.*, 2021b). A *NemaNet* é baseada na topologia CNN e inclui uma estrutura DenseNet (HUANG *et al.*, 2017) original com vários blocos Inception combinando e otimizando duas arquiteturas conhecidas: InceptionV3 e DenseNet121 para extração profunda de recursos. Os blocos Inception são utilizados para controlar a profundidade, o tamanho e o número de parâmetros do modelo. Dessa forma, esses blocos permitem o uso de vários tamanhos de filtro, em vez de ficarem restritos a um único tamanho de filtro em um único bloco de imagem, sendo concatenados e passados para a próxima camada.

A Figura 6.5 apresenta a arquitetura da DenseNet121, sendo composta por três tipos de blocos em sua implementação. O primeiro é o bloco de convolução, que é um bloco básico de um bloco denso. O bloco de convolução é semelhante ao bloco de identidade uma arquitetura ResNet (HE *et al.*, 2016). O segundo é o bloco denso, no qual os blocos de convolução são concatenados e densamente conectados. O bloco denso é o principal componente da DenseNet. A última é a camada de transição, que conecta dois blocos densos contíguos. Como os tamanhos dos mapas de características são iguais no bloco denso, a camada de transição reduz as dimensões do mapa de características.

Os blocos Inception usados, por outro lado, implementam uma abordagem multi-escalar. Cada bloco tem vários ramos com diferentes tamanhos de núcleos ($[1 \times 1]$, $[3 \times 3]$, $[5 \times 5]$ e $[7 \times 7]$). Esses filtros extraem e concatenam mapas de características de diferentes escalas e enviam a combinação para o próximo estágio. A convolução 1×1 em cada módulo Inception é usada para redução de dimensionalidade antes de aplicar as convoluções computacionalmente caras $[3 \times 3]$ e $[5 \times 5]$. A fatoração das convoluções $[5 \times 5]$ e $[7 \times 7]$ em convoluções menores $[3 \times 3]$ ou convoluções assimétricas ($[1 \times 7]$, $[7 \times 1]$) reduz o número de parâmetros CNN (SZEGEDY *et al.*, 2015b). Na Figura 6.6 são apresentados os blocos definidos como Inception A, B e C que foram implementados no modelo *NemaNet*.

Após a extração profunda de características que ocorre em paralelo através dos blocos Inception e da estrutura da DenseNet, o estágio de Concatenação é implementado, permitindo que cada camada compartilhe o sinal de todas as camadas anteriores. Na Figura 6.7 é apresentada a proposta de uma arquitetura customizada, denominada *NemaNet*.

A *NemaNet* tem um total de 17.918.565 parâmetros, sendo 17.817.925 treináveis e 100.640 não treináveis.

6.1.5 Treinamento e Validação

Neste trabalho, cada arquitetura foi avaliada utilizando duas abordagens de treinamento: Partindo do Zero (*From Scratch*) e Transferência de Aprendizagem (*Transfer Learning*). A

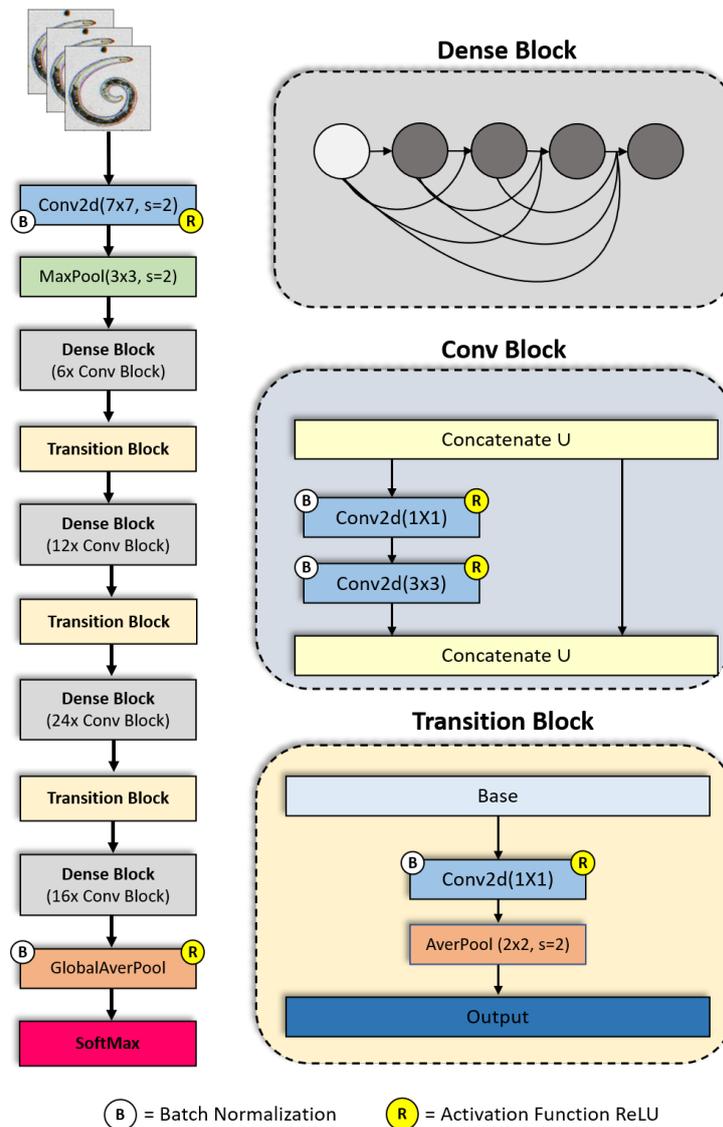


Figura 6.5: **(Esquerda)** Arquitetura DenseNet121. **(Direita)** Dense block, Conv block and Transition block. Adaptado (HUANG *et al.*, 2017)

partir deste momento, a abordagem denominada Partindo do Zero será descrita como **FS** e a abordagem denominada Transferência de Aprendizagem como **TL**.

Treinamento Partindo do Zero (FS) é quando os pesos da rede não são herdados de um modelo anterior, mas são inicializados aleatoriamente. Requer um conjunto de treinamento maior, e o risco de *overfitting* (HAWKINS, 2004) é maior, pois a rede não tem experiência de sessões de treinamento anteriores, assim deve contar com os dados de entrada para definir todos os seus pesos. No entanto, essa abordagem permite definir uma topologia de rede específica, visando o aumento de desempenho na solução de problemas de classificação e reconhecimento de objetos.

Transferência de Aprendizagem (TL) é um método que reutiliza modelos aplicados a tarefas específicas como ponto de partida para um modelo relacionado a um novo domínio de interesse. Assim, o objetivo é emprestar dados rotulados ou extrair conhecimento de campos relacionados

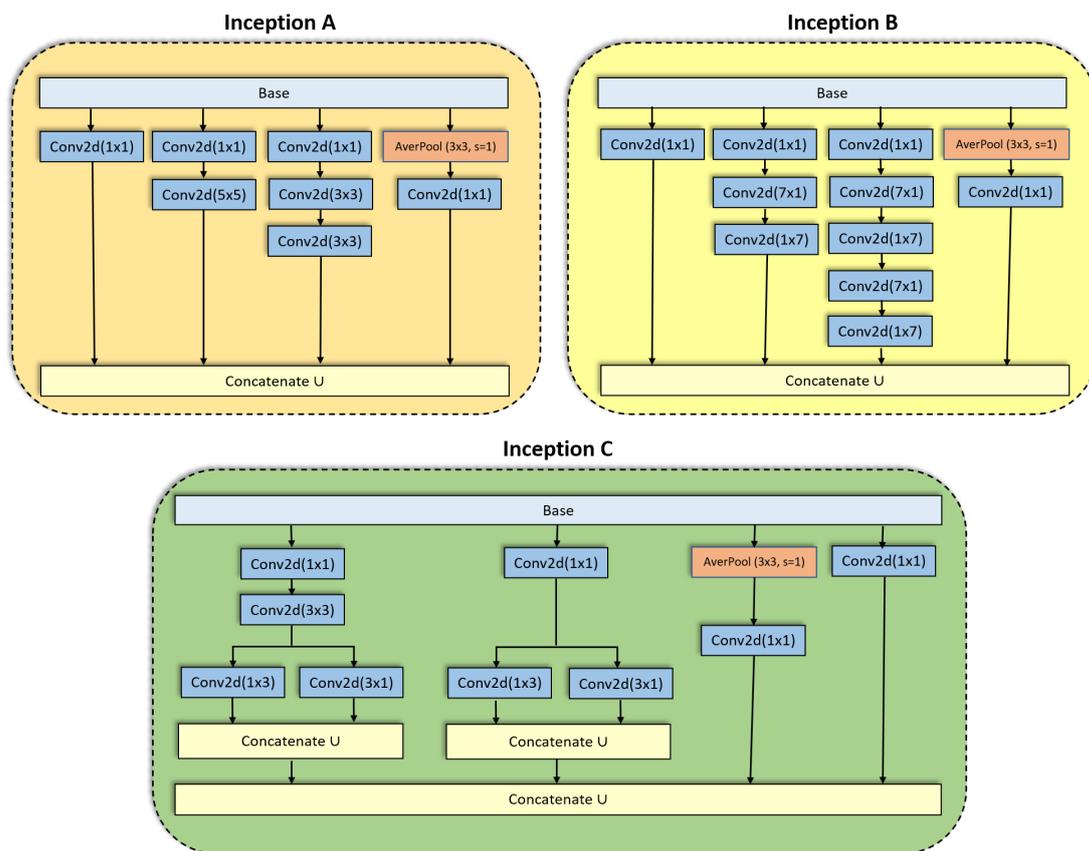


Figura 6.6: Blocos Inception implementados na arquitetura InceptionV3.

para obter o melhor desempenho possível na área de interesse (PAN; YANG, 2010; SHAO *et al.*, 2015).

De acordo com as práticas padrão, existem duas maneiras de abordar TL (KHANDELWAL; RAMAN, 2019). Na primeira, utiliza-se uma rede neural de base como um extrator de características fixo, neste caso as imagens do conjunto de dados alvo são alimentadas para a rede neural profunda e as características que são geradas como entrada para a camada classificadora final são extraídas. Por meio dessas características, um novo classificador é construído e o modelo é criado. Na segunda abordagem ocorre o ajuste fino da rede base, no qual a camada final do classificador é substituída, assim como na abordagem anterior, o ajuste fino é adicionado e os pesos das camadas anteriores também são modificados.

Neste estudo, utilizou-se a segunda abordagem para TL com modelos pré-treinados no ImageNet (DENG *et al.*, 2009) e implementando o ajuste fino adequado às peculiaridades do NemaDataset.

Para evitar o *overfitting* e melhorar a generalização dos modelos avaliados, devido à restrição quantitativa de imagens no conjunto de dados, foram usadas camadas de pré-processamento Keras para criar um pipeline de camadas de aumento de dados de imagem, que podem ser usadas como código de pré-processamento independentes (CHOLLET *et al.*, 2021). Essas camadas aplicam transformações de aumento aleatórias a um lote de imagens e só ficam ativas durante o treinamento (CHOLLET *et al.*, 2021). A Tabela 6.4 apresenta cada camada com seus respecti-

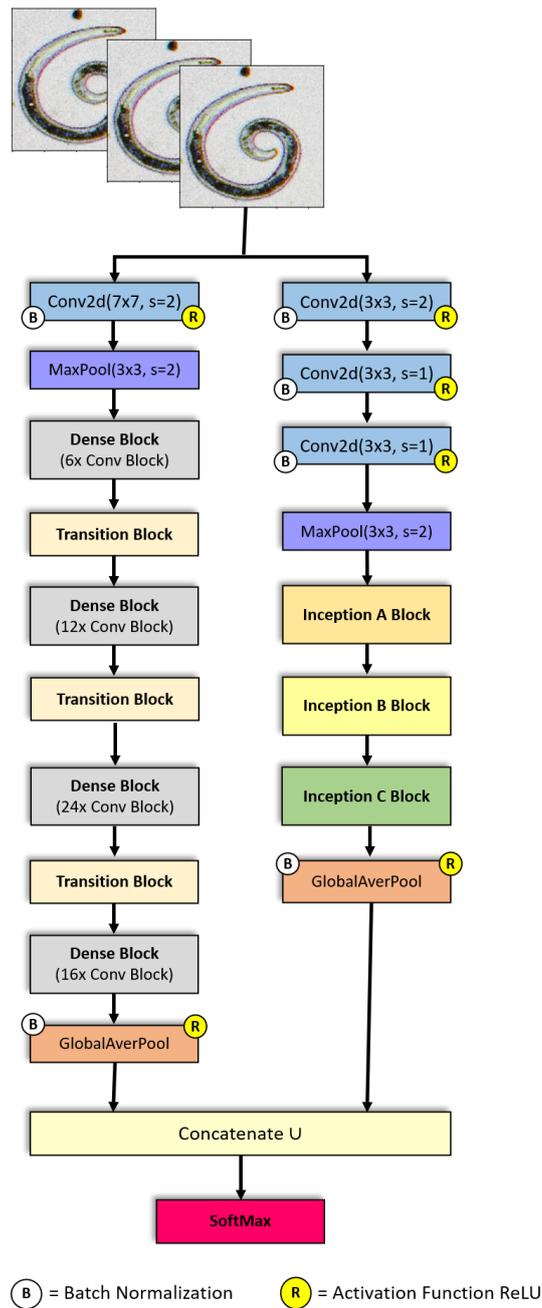


Figura 6.7: Arquitetura NemaNet customizada com a estrutura de blocos DenseNet121 e Inception

vos parâmetros implementados.

Tabela 6.4: Camadas e parâmetros de aumento de dados de imagem.

Camadas	Parâmetros
RandomTranslation	height_factor=0.1, width_factor=0.1, fill_mode='reflect'
RandomFlip	mode='horizontal_and_vertical'
RandomRotation	factor=0.1, fill_mode='reflect', interpolation='bilinear'
RandomContrast	factor=0.1

Um algoritmo de otimização estocástica (SGD) foi utilizado para a otimização no treinamento da rede proposta. Inicialmente, definiu-se uma taxa de aprendizagem básica de 1×10^{-3} .

A taxa de aprendizagem básica foi reduzida para 6×10^{-6} com o aumento das iterações.

No processo de validação usou-se o método de validação cruzada *k-fold* (HASTIE *et al.*, 2009a; KOHAVI *et al.*, 1995). O conjunto de dados foi dividido em 5 (*k*) subconjuntos mutuamente exclusivos do mesmo tamanho. Essa estratégia faz com que um subconjunto seja usado para os testes e o restante $k - 1$ seja usado para estimar os parâmetros, calculando assim a precisão do modelo.

6.2 Resultados

Esta seção apresenta os resultados obtidos com base nos modelos CNN detalhados na Seção 6.1.3. Todos os diferentes modelos foram treinados usando os parâmetros apresentados na Tabela 6.3. Cada modelo foi avaliado usando as abordagens de treinamento FS e TL, implementando validação cruzada com cinco dobras ($k = 5$). Tentou-se padronizar o tamanho do lote de acordo com as referências dos modelos pré-treinados usando ImageNet. No entanto, alguns modelos excederam nossa capacidade computacional e foram ajustados corretamente.

Na Tabela A.3 do apêndice A são apresentadas as precisões gerais obtidas para cada iteração da validação cruzada usando NemaDataset. Ao final de cada tipo de treinamento realizado foi apresentada uma média dos valores de cada métrica utilizada.

Os resultados demonstraram que os experimentos realizados com treinamento FS e aumento de dados não produziram necessariamente ganhos significativos. Entretanto, os resultados são condizentes com a literatura (HAN *et al.*, 2018; SHORTEN; KHOSHGOFTAAR, 2019; ZOPH *et al.*, 2020), demonstrando que a utilização de TL com aumento de dados, além de atenuar os problemas de *overfitting* e *underfitting*, permite aumentos significativos na precisão da acurácia dos modelos. Todos os modelos avaliados, incluindo a *NemaNet*, obtiveram melhor desempenho usando TL com aumento de dados.

Entre os treze modelos tradicionais e a proposta do modelo customizado denominado *NemaNet* que foram avaliados, foram eleitos os cinco melhores modelos para cada tipo de treinamento com maior acurácia calculada para a média geral. Na Tabela 6.5 é apresentada uma sumarização com os cinco melhores modelos para cada tipo de treinamento.

Para os modelos com treinamento FS, destacaram-se, por ordem de relevância: *NemaNet* (96,76%); InceptionV3 (95,82%); Xception (95,23%); InceptionResNetV2 (94,84%) e DenseNet169 (93,20%).

Quando foram implementados utilizando o treinamento com TL, os cinco modelos que se destacaram com aumento de dados foram: *NemaNet* (98,82%), DenseNet201 (97,97%), DenseNet169 (97,94%), DenseNet121 (97,61%) e InceptionV3 (96,60%).

Os experimentos realizados também demonstraram que a arquitetura customizada proposta denominada *NemaNet* obteve o melhor desempenho entre as duas estratégias de treinamento utilizadas. É preciso destacar também que os resultados utilizando TL envolvem o carregamento

Tabela 6.5: Sumarização dos Resultados das Métricas por Modelo. Os cinco principais modelos foram destacados de acordo com o tipo de treinamento (FS/TL), com ou sem aumento de dados.

Tipo	Treinamento	CNNs	Métricas										
			Sem Aumento de Dados						Com Aumento de Dados				
			Loss	Accuracy	F1-Score	Precision	Recall	Specificity	Loss	Accuracy	F1-Score	Precision	Recall
FS	NemaNet	0,2017	0,9676	0,9668	0,9674	0,9663	0,9903	0,2720	0,9612	0,9605	0,9632	0,9580	0,9861
	InceptionV3	0,1898	0,9582	0,9549	0,9566	0,9534	0,9856	0,1726	0,9559	0,9562	0,9606	0,9521	0,9808
	Xception	0,2022	0,9523	0,9506	0,9528	0,9486	0,9843	0,2289	0,9471	0,9475	0,9525	0,9429	0,9797
	InceptionResNetV2	0,1956	0,9484	0,9477	0,9600	0,9382	0,9734	0,2239	0,9350	0,9359	0,9445	0,9281	0,9748
	DenseNet169	0,3891	0,9320	0,9323	0,9377	0,9274	0,9765	0,2885	0,9239	0,9207	0,9310	0,9120	0,9716
	NasNetLarge	0,3909	0,9281	0,9280	0,9292	0,9268	0,9802	0,3069	0,9316	0,9328	0,9347	0,9311	0,9803
	DenseNet201	0,6224	0,9180	0,9164	0,9250	0,9092	0,9739	0,2773	0,9177	0,9146	0,9273	0,9031	0,9681
	EfficientNetB3	1,7515	0,9180	0,9210	0,9248	0,9174	0,9756	0,2580	0,9167	0,9124	0,9259	0,9015	0,9676
	ResNet50	0,3779	0,9144	0,9128	0,9197	0,9065	0,9731	0,2937	0,9131	0,9169	0,9259	0,9085	0,9677
	DenseNet121	0,5342	0,9089	0,9050	0,9118	0,8990	0,9714	0,3631	0,9144	0,9181	0,9254	0,9115	0,9717
	ResNet101	0,4389	0,8984	0,8990	0,9068	0,8916	0,9672	0,3667	0,9206	0,9215	0,9288	0,9145	0,9717
	ResNet152	0,4216	0,8925	0,8942	0,9012	0,8876	0,9648	0,2757	0,9141	0,9151	0,9256	0,9052	0,9660
	EfficientNetB0	1,2912	0,8556	0,8582	0,8672	0,8500	0,9581	1,1196	0,8534	0,8421	0,8668	0,8199	0,9433
	VGG16	1,2081	0,7917	0,7949	0,7987	0,7914	0,9444	0,4219	0,8517	0,8520	0,8821	0,8268	0,9431
TL	NemaNet	0,1049	0,9817	0,9821	0,9826	0,9817	0,9942	0,0909	0,9882	0,9881	0,9888	0,9872	0,9955
	DenseNet201	0,0884	0,9777	0,9760	0,9774	0,9746	0,9915	0,0887	0,9797	0,9802	0,9815	0,9790	0,9930
	DenseNet169	0,0088	0,9768	0,9777	0,9790	0,9765	0,9918	0,0870	0,9794	0,9801	0,9813	0,9791	0,9925
	DenseNet121	0,1006	0,9715	0,9725	0,9739	0,9712	0,9902	0,0826	0,9761	0,9760	0,9788	0,9734	0,9893
	InceptionV3	0,1678	0,9523	0,9549	0,9569	0,9531	0,9856	0,1242	0,9660	0,9678	0,9718	0,9637	0,9857
	ResNet101	0,1774	0,9467	0,9470	0,9522	0,9420	0,9803	0,1119	0,9657	0,9648	0,9697	0,9602	0,9847
	Xception	0,1570	0,9588	0,9577	0,9629	0,9528	0,9788	0,1357	0,9621	0,9629	0,9671	0,9590	0,9818
	NasNetLarge	0,1756	0,9451	0,9456	0,9499	0,9419	0,9788	0,3087	0,9618	0,9625	0,9625	0,9625	0,9898
	ResNet152	0,1516	0,9536	0,9529	0,9586	0,9474	0,9813	0,1648	0,9546	0,9537	0,9588	0,9489	0,9796
	InceptionResNetV2	0,2626	0,9311	0,9332	0,9389	0,9280	0,9787	0,1537	0,9536	0,9538	0,9590	0,9490	0,9813
	EfficientNetB3	2,0474	0,9190	0,9154	0,9316	0,9016	0,9647	0,7790	0,9474	0,9478	0,9565	0,9399	0,9790
	ResNet50	0,3377	0,9023	0,9094	0,9203	0,9001	0,9661	0,1859	0,9467	0,9512	0,9572	0,9455	0,9796
	EfficientNetB0	0,2537	0,9112	0,9152	0,9283	0,9030	0,9682	0,2433	0,9392	0,9377	0,9394	0,9361	0,9821
	VGG16	0,2647	0,9209	0,9228	0,9278	0,9181	0,9728	0,2553	0,9213	0,9231	0,9322	0,9150	0,9697

de pesos pré-treinados no ImageNet para a arquitetura DenseNet121, aproveitando a similaridade do modelo customizado com os blocos densos. A parte formada por blocos Inception não foi inicializada com pesos pré-treinados.

Foi possível perceber que a *NemaNet*, tanto para FS quanto para TL, passou a convergir progressivamente a partir da vigésima época, apresentando um comportamento melhor que os demais modelos avaliados. As Figuras 6.8, 6.9, 6.10 e 6.11 apresentam a evolução das etapas de treinamento e validação desses resultados.

Nas Figuras 6.12 (FS) e 6.13 (TL) são apresentadas as matrizes de confusão normalizadas para os cinco melhores modelos de acordo com as estratégias de treinamento utilizadas. Adicionalmente, as Figuras 6.14 (FS) e 6.15 (TL) apresentam a Curva ROC Multiclass calculada para cada modelo CNN.

6.3 Discussão

A abordagem avaliou treze arquiteturas populares de redes neurais convolucionais e propôs uma nova arquitetura customizada chamada *NemaNet* para identificar os principais nematoides que causam danos à soja. Os resultados mostraram que a proposta obteve o melhor desempenho entre as demais arquiteturas avaliadas, tanto para treinamento de FS quanto para TL.

Entre os modelos de CNNs treinados e avaliados pela técnica FS, as arquiteturas compostas por blocos Inception, nomeadamente InceptionV3, Xception e InceptionResNetV2, ocupam posições de destaque entre as cinco primeiras posições em termos de precisão média. Os mode-

Inicializando Pesos sem Aumento de Dados

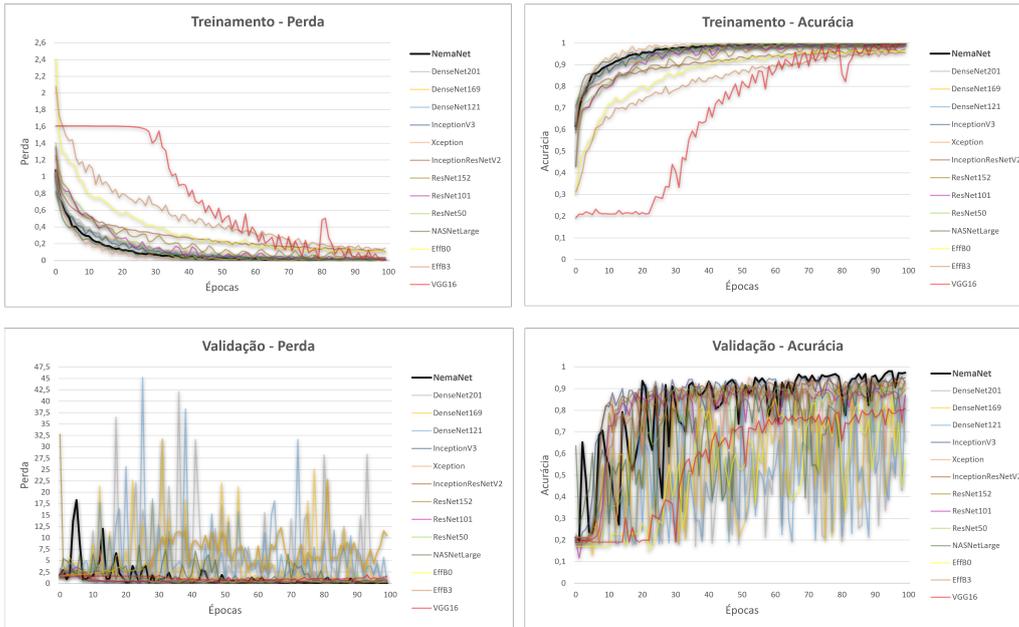


Figura 6.8: Curvas de perda e precisão do processo de treinamento FS sem aumento de dados.

Inicializando Pesos com Aumento de Dados

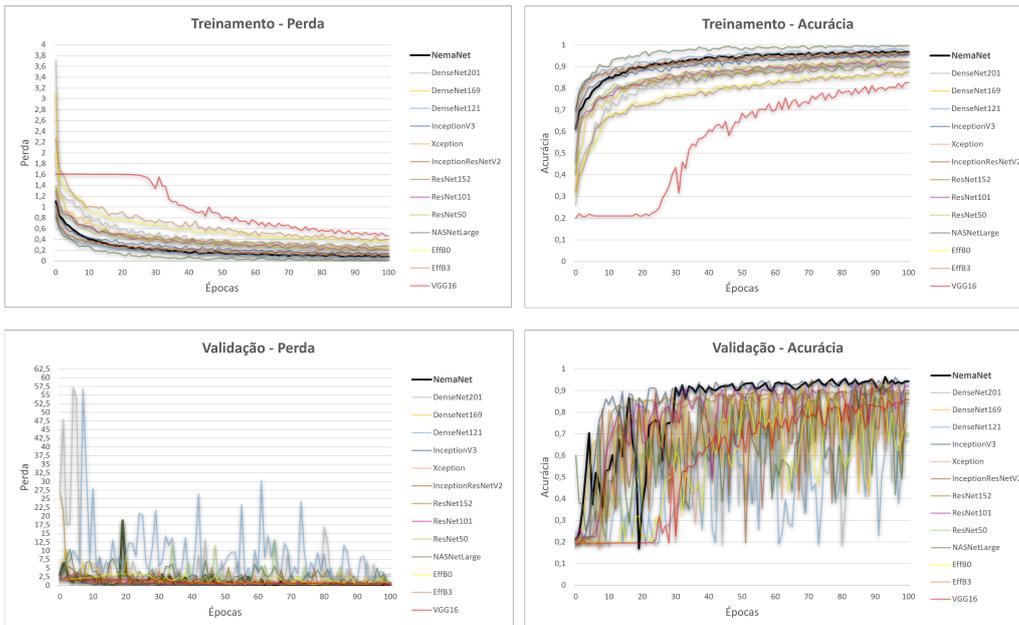


Figura 6.9: Curvas de perda e precisão do processo de treinamento FS com aumento de dados

los Xception e InceptionResNetV2, que usam conexões residuais combinadas com a arquitetura Inception, tiveram um desempenho pior do que a rede InceptionV3. De acordo com Szegedy et al. (2017) é necessário customizar essas arquiteturas para mostrar os ganhos proporcionados pelas conexões residuais (SZEGEDY *et al.*, 2017a).

O bloco de construção fundamental dos modelos de estilo Inception é o módulo Inception, do qual existem várias versões diferentes. O bloco Inception é equivalente a uma sub-rede com quatro caminhos. Ele extrai informações em paralelo por meio de camadas convolucionais de

Transferência de Aprendizagem sem Aumento de Dados

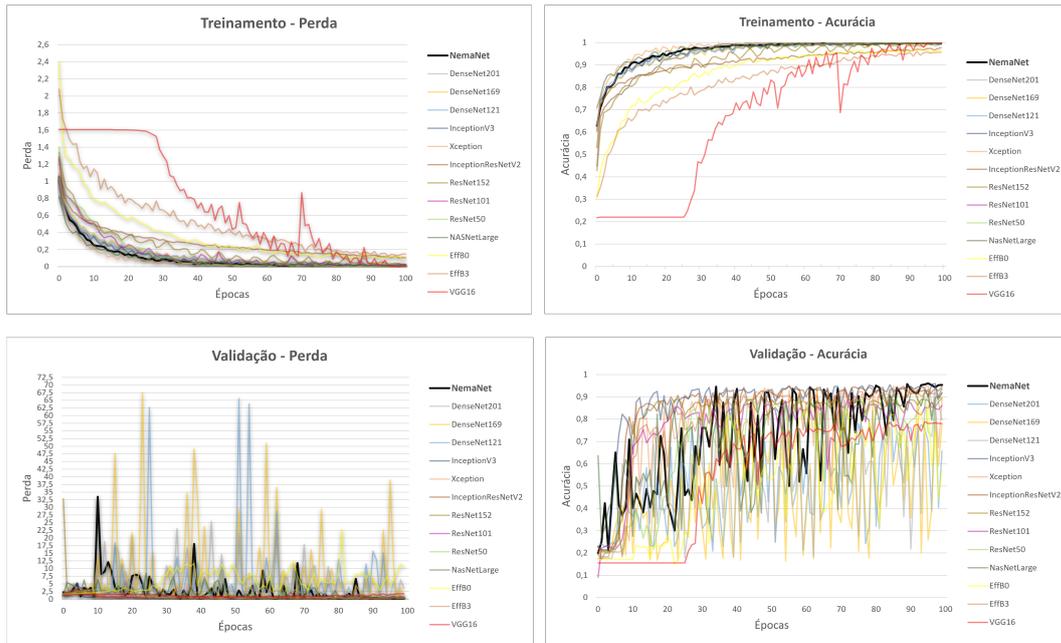


Figura 6.10: Curvas de perda e precisão do processo de treinamento TL sem aumento de dados.

Transferência de Aprendizagem com Aumento de Dados

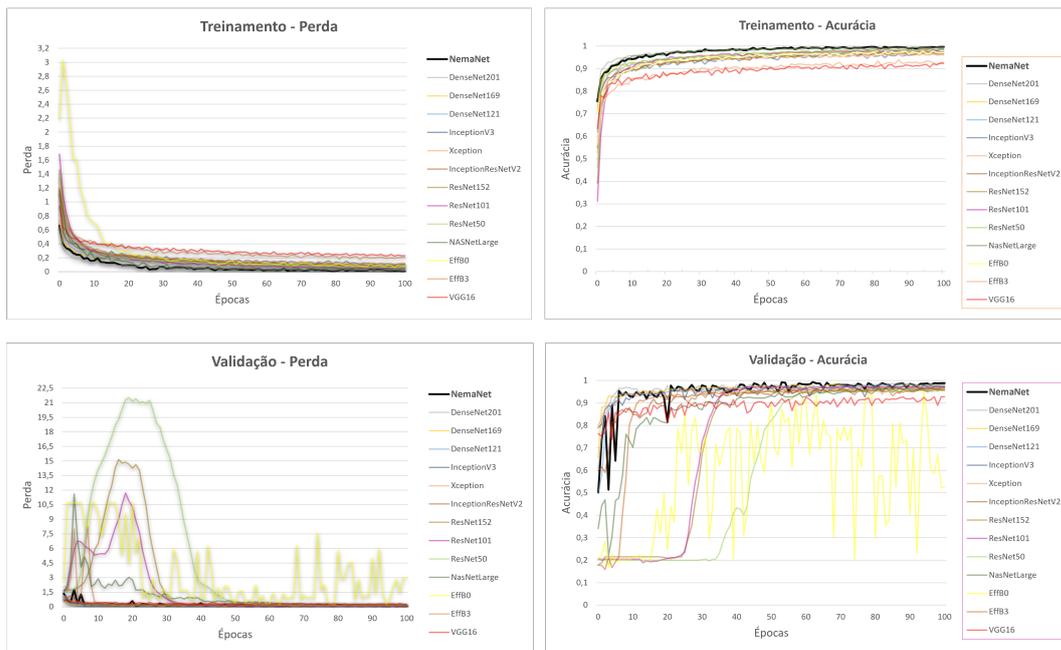


Figura 6.11: Curvas de perda e precisão do processo de treinamento TL com aumento de dados.

formatos de janela variados e camadas de agrupamento máximo (SZEGEDY *et al.*, 2015b). Embora os módulos Inception sejam conceitualmente semelhantes às convoluções (eles são extratores de recursos convolucionais), eles empiricamente parecem conseguir aprender representações mais ricas com menos parâmetros (CHOLLET, 2017).

Quando os modelos treinados usando TL foram avaliados, as arquiteturas compostas por blocos densos denominadas DenseNets (DenseNet201, DenseNet169 e DenseNet121) ocupam

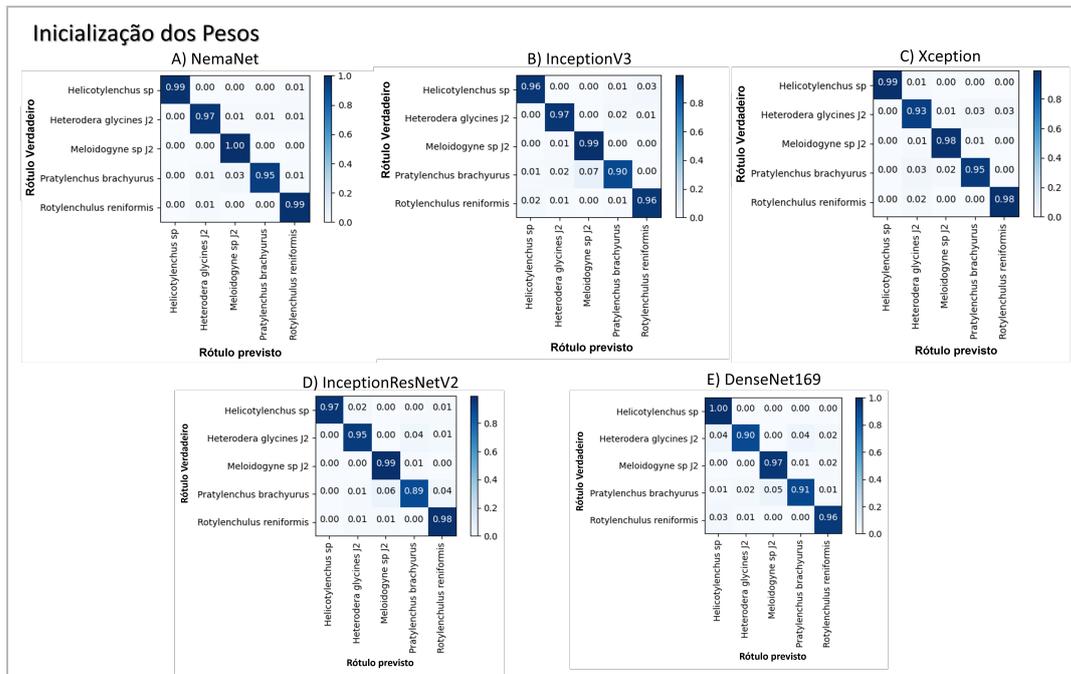


Figura 6.12: Matrix de Confusão Normalizada - Inicialização de Pesos: (A) NemaNet; (B) InceptionV3; (C) Xception; (D) InceptionResNetV2 e (E) DenseNet169

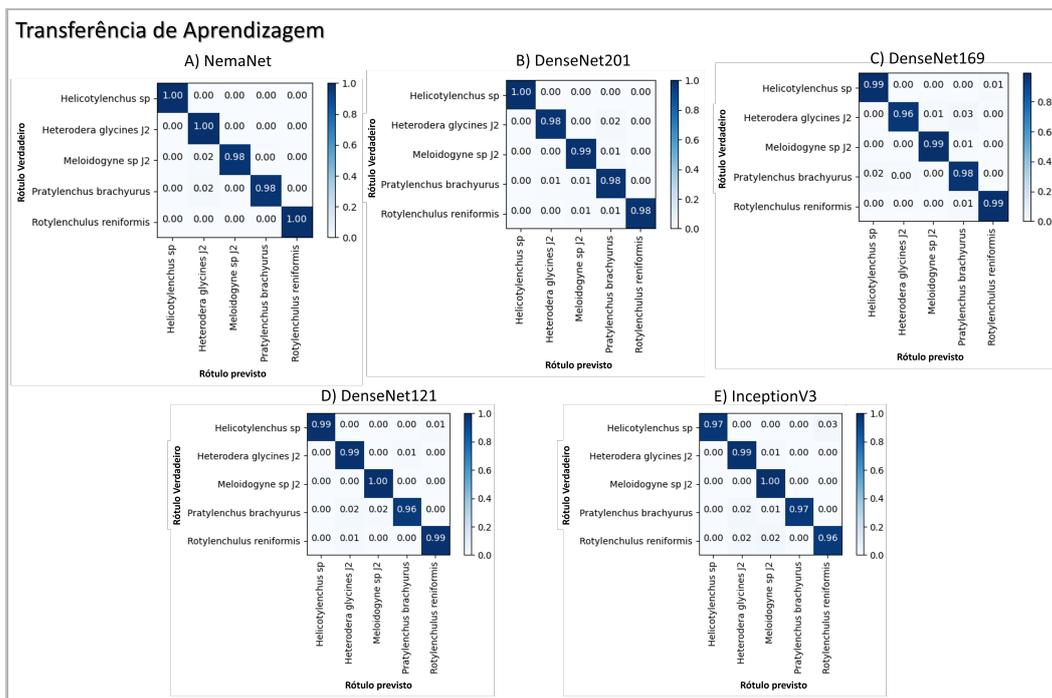


Figura 6.13: Matrix de Confusão Normalizada - Transferência de Aprendizagem: (A) NemaNet; (B) DenseNet201; (C) DenseNet169; (D) DenseNet121 e (E) InceptionV3

as cinco primeiras posições em termos de precisão média. A arquitetura DenseNet tem várias vantagens atraentes: alivia o problema do gradiente descendente, fortalece a propagação de

Inicialização dos Pesos

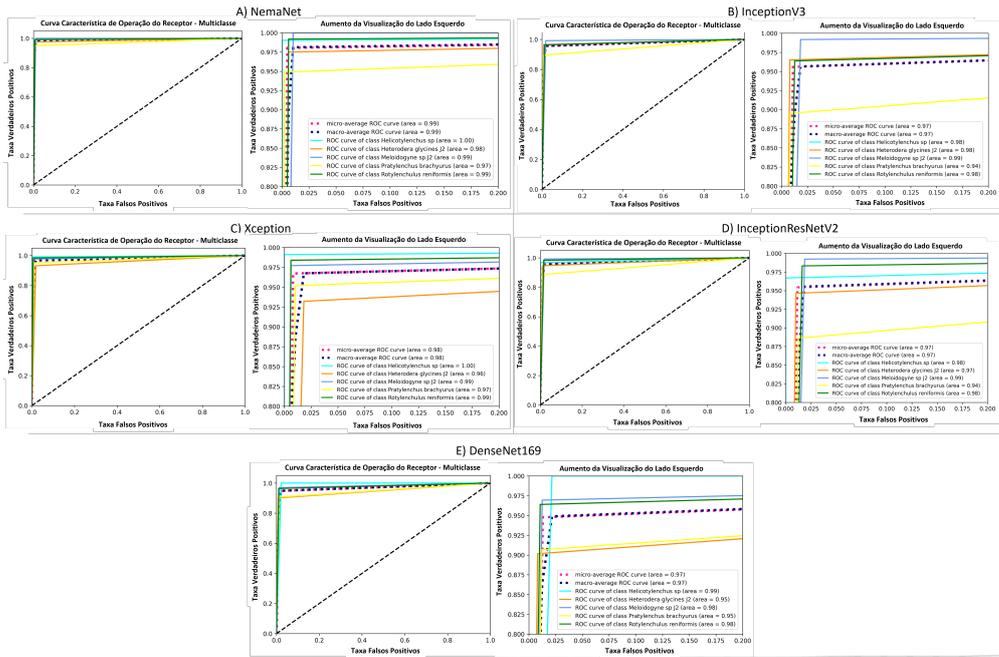


Figura 6.14: Curva Característica de Operação do Receptor(ROC) Multiclasse com Inicialização dos Pesos: (A) NemaNet; (B) DenseNet201; (C) DenseNet169; (D) DenseNet121 e (E) InceptionResNetV2

Transferência de Aprendizagem

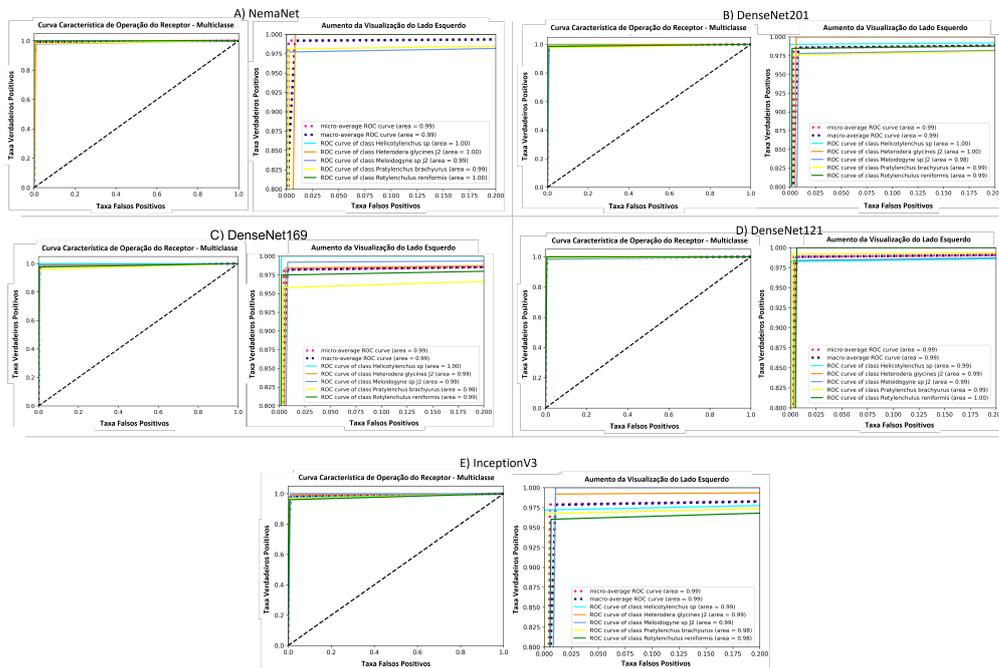


Figura 6.15: Curva Característica de Operação do Receptor(ROC) Multiclasse com Transferência de Aprendizagem: (A) NemaNet; (B) DenseNet201; (C) DenseNet169; (D) DenseNet121 e (E) InceptionV3

características, incentiva a reutilização de características e reduz substancialmente o número de parâmetros (HUANG *et al.*, 2017).

Na DenseNet, o classificador usa recursos de todos os níveis de complexidade, e isso tende a dar limites de decisão mais suaves. Esse fator também explica por que a DenseNet tem um bom desempenho quando os dados de treinamento são insuficientes. Cada camada da DenseNet recebe todas as camadas anteriores como entrada, resultando em recursos mais diversificados e padrões mais ricos. Com treinamento usando um modelo pré-treinado no conjunto de dados ImageNet, este modelo aproveita recursos extraídos por camadas muito iniciais que são usadas diretamente por camadas mais profundas em todo o mesmo bloco denso. Esta funcionalidade, combinada com o ajuste fino e a otimização de hiperparâmetros, aumenta a capacidade preditiva desta arquitetura.

A *NemaNet* explora os diferentes comportamentos das duas arquiteturas, InceptionV3 e DenseNet121, personalizando um modelo capaz de convergir com o menor número de épocas, com um tamanho de modelo e número de parâmetros pouco superior à DenseNet121 e apresentando maior precisão. O arranjo da topologia customizada da *NemaNet* foi elaborado para garantir que o modelo tivesse um bom desempenho em imagens de alta e baixa resolução, habilitadas pelos blocos Inception, e analisasse as representações da imagem em diferentes escalas.

Um fator crítico que pode ser mencionado é a ocorrência de *overfitting* nas arquiteturas DenseNet originais com treinamento FS. O modelo proposto *NemaNet*, por meio da adição de blocos Inception à estrutura DenseNet121 garante a estabilidade da rede, mesmo diante de um conjunto de dados considerado pequeno, como o NemaDataSet que foi usado nos experimentos deste estudo.

O otimizador de gradiente descendente estocástico com *momentum* foi usado para treinar todos os modelos avaliados. A taxa de aprendizagem é um dos hiperparâmetros mais relevantes para um processo de treinamento da CNN e possivelmente a chave para um treinamento prático e mais rápido da rede. Nos experimentos realizados, optou-se por usar um método de taxa de aprendizado cíclico (CLR) que praticamente elimina a necessidade de ajustar a taxa de aprendizado, alcançando uma precisão de classificação quase ideal. Ao contrário das taxas de aprendizagem adaptativa, o método CLR essencialmente não requer nenhum cálculo adicional (SMITH, 2017).

Entretanto, foi possível observar que, durante o processo de treinamento e validação da *NemaNet*, ocorreu uma variação brusca nos valores de perda e, conseqüentemente, menor acurácia nas primeiras vinte épocas de treinamento. Associamos esse fato à estrutura híbrida da rede, composta por blocos densos e blocos Inception. Essa peculiaridade faz com que o método CLR de ajuste da taxa de aprendizado cause variações mais significativas, pois suas mudanças ocorrem ciclicamente a cada lote, ao invés de uma taxa de aprendizado não cíclica que é constante ou muda a cada época.

Notadamente, quando se utilizou o TL no processo de treinamento e validação é possível perceber que a maioria dos comportamentos dos modelos avaliados tende a ter um ajuste adequado. Nesse sentido, ao avaliar a *NemaNet* e sua topologia híbrida proposta foram usados pesos pré-treinados no ImageNet apenas para a parte padrão da estrutura DenseNet121. Tendo

em vista que os blocos Inception são customizados e adaptados para funcionar como uma estrutura paralela e auxiliar, não há possibilidade de reaproveitar o conhecimento para esses blocos, portanto considerou-se que a *NemaNet* utiliza parcialmente os benefícios da TL.

A estratégia usando um TL parcial é bem-sucedida quando comparada aos processos de treinamento e validação. Percebeu-se que a capacidade de convergência da abordagem é superior aos demais modelos avaliados.

Outro fator importante é que todos os modelos treinados e avaliados que utilizaram TL tiveram um desempenho melhor do que os treinados com FS. Assim, mesmo diante das peculiaridades das imagens microscópicas e das morfologias semelhantes das espécies de fitonematoides, o comportamento discriminatório dos modelos CNN se mostrou uma abordagem eficiente para a identificação desses patógenos.

6.4 Considerações Finais

Neste capítulo, foi apresentado um conjunto de dados denominado NemaDataset, que inclui as cinco principais espécies de fitonematoides que causam danos à cultura de soja. Além disso, foram treinados e avaliados treze arquiteturas CNN usando o NemaDataset, representando o que há de mais moderno em classificação e reconhecimento de objetos na área de pesquisa em visão computacional. Todas as arquiteturas avaliadas foram comparadas com o modelo proposto neste estudo, denominado *NEMANet*.

Por meio das métricas utilizadas, foram selecionados os cinco melhores modelos destacados na Tabela 6.5, que demonstraram uma variação média na precisão para treinamento de FS de 79,17% a 96,76% e para TL de 92,13% a 98,82%. O modelo *NemaNet* obteve os melhores resultados para as duas estratégias de treinamento e validação utilizadas.

Na média geral calculada para cada modelo, para o treinamento de FS, a *NemaNet* atingiu 96,76% de acurácia, enquanto a melhor dobra de avaliação atingiu 98,04%. Quanto ao treinamento TL, a acurácia média atingiu 98,82%, enquanto a melhor dobra de avaliação atingiu 99,35%.

Capítulo 7

Conclusões e Trabalhos Futuros

Este trabalho explorou as potencialidades das CNNs com arquiteturas customizadas para classificação de doenças de plantas e a identificação de patógenos, conciliando o know-how de especialistas em fitopatologia e a capacidade de extração de características sintomáticas por meio das redes neurais convolucionais.

Para analisar e avaliar as diferentes técnicas e métodos de identificação automatizados de doenças de plantas usando CNNs, o Capítulo 4 apresenta a caracterização do estado da arte por meio de uma revisão sistemática da literatura. Os resultados demonstraram avanços significativos no uso de CNNs em processos de previsão de doenças de plantas.

Foi possível observar uma tendência crescente de abordagens que propõe novas arquiteturas ou modelos CNNs baseados no processo de identificação de doenças de plantas. Entretanto, constatou-se uma especialização dessas arquiteturas em detrimento do tipo de cultura investigada.

Embora a diversidade sintomática causada por um mesmo patógeno seja um fator complicador para a inferência para detecção e reconhecimento de doenças de plantas, constatou-se uma carência de estudos que proponhem arquiteturas ou modelos que visem especificamente a doença, independentemente do tipo de cultivo. Além disso, ficou evidente o número inexpressivo de abordagens que identificam patógenos causadores de doenças em plantas, em especial os fitonematoides.

No Capítulo 5 é apresentada uma abordagem que propõe modelos M-CNN com arquitetura customizada e com otimização de hiperparâmetros nas camadas de regularização para classificação de doenças de plantas usando imagens com sintomas foliares presentes no conjunto de dados PlantVillage. Os resultados alcançados demonstraram a viabilidade da abordagem e a sua capacidade de estruturar novas soluções sem especializar um modelo para um determinado tipo de cultura.

Os resultados apresentados no Capítulo 6 propõem uma solução inovadora que cria um conjunto de dados com imagens de fitonematoides com significância ao cultivo de soja, denominado NemaDataset. Adicionalmente, é proposto um modelo denominado *NEMANet* com

arquitetura híbrida, mesclando os blocos de processamento da DenseNet e a InceptionV3. Essa abordagem inova no processo de customização da arquitetura, explorando as potencialidades de duas arquiteturas já reconhecidas pela literatura. Inúmeras adequações e customizações de blocos foram implementadas para contabilizar as peculiaridades estruturais de cada arquitetura de modo a produzir o melhor resultado.

Os experimentos realizados com a proposta denominada *NEMANet*, em comparação com 13 arquiteturas tradicionais da literatura que representam o estado da arte no processo de classificação de objetos, demonstraram que a abordagem apresentada supera as demais arquiteturas em desempenho e precisão.

Devido ao pequeno quantitativo de abordagens que identificam fitonematoides e aos promissores resultados apresentados pela *NEMANet*, esta solução pode subsidiar o aumento do controle e agilidade dos processos de tomada de decisão quanto ao manejo da cultura em detrimento à presença dos tipos qualificados de patógenos usando CNNs.

A análise dos resultados obtidos permite concluir que este estudo avançou no estado da arte e todos seus objetivos foram atingidos.

7.1 Validação das Hipóteses

Hipótese 1 (H1): Para validar a concepção desta proposta, foram analisados e avaliados os estudos de Mohanty *et al.* (2016) e Ferentinos (2018). O Capítulo 5 descreve a abordagem que cria um modelo CNN descrito como (M-CNN), que explora o potencial das arquiteturas paralelas de múltiplos caminhos. Houve uma preocupação metodológica para condicionar uma justa comparação entre as abordagens utilizadas como referência e a proposta como solução. A abordagem utilizando as (M-CNN) foi capaz de melhorar os valores médios F_1score de modo regular e robusto, independente do modelo escolhido. O processo de avaliação permitiu concluir também que um modelo M-CNN treinado a partir do zero tem melhor desempenho que um modelo com arquitetura tradicional com transferência de aprendizado em dois aspectos: incremento significativo na precisão geral e maior acurácia em um número menor de épocas. Além disso, os modelos treinados a partir do zero são significativamente menores que modelos que recorrem às técnicas de transferência de aprendizado.

Nesta perspectiva, é possível validar a primeira hipótese quanto a viabilidade de utilização de novos arranjos arquitetônicos, que conciliem os conceitos de exploração espacial, profundidade, multiconexão com base em largura e múltiplos canais de entrada no processo de classificação de doenças de plantas. Os resultados demonstraram sistematicamente ganhos evidentes que justificam sua utilização. Entretanto, não foi possível analisar e validar essa abordagem com conjuntos de dados com imagens de ambientes reais de cultivo.

Hipótese 2 (H2): Não foi possível encontrar, nas buscas realizadas na literatura, abordagens que identificam fitonematoides usando CNNs. Deste modo, para validar essa hipótese não

houve possibilidade de comparação com outros trabalhos. Entretanto, treze arquiteturas tradicionais foram avaliadas e um conjunto de dados denominado NemaDataset foi criado exclusivamente para este propósito, sendo apresentados no Capítulo 6. Diferente da primeira abordagem (M-CNN), na qual as redes paralelas replicavam a mesma estrutura sofrendo ajustes apenas nas camadas de regularização, a abordagem denominada *NemaNet* customiza blocos de duas arquiteturas distintas de modo a otimizar o desempenho do modelo para o contexto das imagens microscópicas presentes no conjunto de dados avaliados. Os experimentos demonstraram que o modelo *NemaNet* obteve os melhores resultados para as duas estratégias de treinamento e validação utilizadas.

Os resultados alcançados permitem validar a segunda hipótese, evidenciando características já apresentadas na validação da primeira hipótese. A *NemaNet*, além de atingir melhores resultados, também possui maior capacidade de convergência e, conseqüentemente, pode ter um tempo de treinamento reduzido. O modelo apresentou um comportamento adequado ao tipo de imagem avaliada e a customização dos blocos Inception foram complementares à estrutura da DenseNet, configurando um modelo promissor.

Hipótese 3 (H3): Para validar essa hipótese, foram analisados os resultados dos 2 experimentos apresentados no Capítulo 5 e 6. O primeiro resultado demonstra que o modelo M-CNN customizado treinado do zero supera as abordagens utilizando a técnica de transferência de aprendizagem. Entretanto, diferente da validação da primeira hipótese, a estratégia de treinamento do zero não foi suficiente para superar 3 arquiteturas tradicionais do total de 13 avaliadas nos resultados do Capítulo 6. Essa controversa não garante a validação desta hipótese.

A diversidade de problemas e as especificidades dos cenários do mundo real aumentam a dificuldade de catalogar semanticamente conjunto de dados representativos com um número suficiente de amostras rotuladas. Essa questão foi um aspecto desfavorável presente durante todo o desenvolvimento desta pesquisa. Encontrar conjuntos de imagens de doenças de plantas em ambientes reais de cultivo, devidamente rotuladas e com um quantitativo balanceado de amostras disponível para estudo e análise não é uma tarefa fácil.

Sobre a identificação de fitonematoides, deve-se notar que com a *NemaNet* não houve a oportunidade de validar seu desempenho em um conjunto de dados com um número maior de classes. Não foi possível encontrar outros conjuntos de dados de fitonematoides que permitissem uma comparação e avaliação de desempenho. Outros fatores limitantes podem ameaçar a validade de nossos resultados, como, por exemplo, a diversidade de métodos de extração de fitonematoides do solo e das raízes e o tipo de tecnologia microscópica utilizada para adquirir as imagens.

7.2 Trabalhos Futuros

Futuramente, pretende-se aumentar o número de espécies (classes) de fitonematoides no conjunto de dados NemaDataset e, conseqüentemente, aumentar o número de amostras para

cada categoria, ampliando sua capacidade de representação. Além disso, imagens descritivas com chaves taxonômicas serão adicionadas para melhorar a capacidade de extrair características para fenotipagem de espécies.

Outras otimizações no modelo de topologia proposto serão implementadas, desta vez usando técnicas de *Neural Architecture Research (NAS)* (ELSKEN *et al.*, 2019). Existem muitas abordagens relacionadas a espaços de busca arquitetônicos, algoritmos de otimização e métodos para avaliar arquiteturas candidatas. Nesse sentido, ao definir um espaço de busca com os blocos Inception e Densos, prospecta-se um arranjo maior de possibilidades para compor topologias ainda mais eficientes.

Quanto à detecção de objetos, pretende-se utilizar a RetinaNet, customizando o backbone para *NemaNet*. Estas adaptações demandaram ajustes na estrutura de regressão e classificação da RetinaNet. Essa abordagem será testada com o conjunto de dados NemaDataset

Avaliar o modelo *NemaNet* com outros conjuntos de imagens microscópicas para identificação de diferentes patógenos causadores de doenças em plantas.

Dado a relevância das imagens multiespectrais e hiperespectrais no processo de diagnóstico de doenças de plantas, implementar a abordagem desenvolvida utilizando conjuntos de dados com estas características.

Referências Bibliográficas

ABADE, A.; ALMEIDA., A. P. G. S. de; VIDAL, F. B. Plant diseases recognition from digital images using multichannel convolutional neural networks. In: INSTICC. *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP*. Prague, Czech Republic: SciTePress, 2019. p. 450–458. ISBN 978-989-758-354-4.

ABADE, A.; FERREIRA, P. A.; VIDAL, F. de B. Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture*, v. 185, p. 106–125, 2021. ISSN 0168-1699. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0168169921001435>>.

ABADE, A.; PORTO, L. F.; FERREIRA, P. A.; VIDAL, F. de B. Nemanet: A convolutional neural network model for identification of nematodes soybean crop in brazil. *Biosystems Engineering*, abs/2103.03717, 2021. ISSN 1537-5110.

ABDALLA, A.; CEN, H.; WAN, L.; RASHID, R.; WENG, H.; ZHOU, W.; HE, Y. Fine-tuning convolutional neural network with transfer learning for semantic segmentation of ground-level oilseed rape images in a field with high weed pressure. *Computers and Electronics in Agriculture*, v. 167, p. 105091, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169919317107>>.

AGGARWAL, C. C. *Data Classification: Algorithms and Applications*. 1st. ed. [S.l.]: Chapman & Hall/CRC, 2014. ISBN 1466586745, 9781466586741.

AGRIOS, G. *Plant pathology: Fifth edition*. [S.l.]: Academic Press, 2005. v. 9780080473789. 1–922 p. ISBN 9780120445653.

ALFARISY, A. A.; CHEN, Q.; GUO, M. Deep learning based classification for paddy pests & diseases recognition. In: *Proceedings of 2018 International Conference on Mathematics and Artificial Intelligence*. New York, NY, USA: ACM, 2018. (ICMAI '18), p. 21–25. ISBN 978-1-4503-6420-1. Disponível em: <<http://doi.acm.org/10.1145/3208788.3208795>>.

ALPAYDIN, E. *Introduction to Machine Learning*. [S.l.]: The MIT Press, 2014. ISBN 0262028182, 9780262028189.

ALRUWAILI, M.; ALANAZI, S.; EL-GHANY, S. A.; SHEHAB, A. An efficient deep learning model for olive diseases detection. *International Journal of Advanced Computer Science and Applications*, The Science and Information Organization, v. 10, n. 8, 2019. Disponível em: <<http://dx.doi.org/10.14569/IJACSA.2019.0100863>>.

ALZUBAIDI, L.; ZHANG, J.; HUMAIDI, A. J.; AL-DUJAILI, A.; DUAN, Y.; AL-SHAMMA, O.; SANTAMARÍA, J.; FADHEL, M. A.; AL-AMIDIE, M.; FARHAN, L. Review of deep

learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of Big Data*, v. 8, n. 1, p. 53, Mar 2021. ISSN 2196-1115.

AMARA, J.; BOUAZIZ, B.; ALGERGAWY, A. A deep learning-based approach for banana leaf diseases classification. In: *BTW*. [S.l.: s.n.], 2017.

AMORIM, L.; REZENDE, J. A. M.; FILHO, A. B. *Manual de Fitopatologia: Conceitos e princípios*. Brasil: Agronomica Ceres, 2011. v. 4. ISBN 9788531800528.

ARAVIND, K. R.; RAJA, P.; ANIIRUDH, R.; MUKESH, K. V.; ASHIWIN, R.; VIKAS, G. Grape crop disease classification using transfer learning approach. In: PANDIAN, D.; FERNANDO, X.; BAIG, Z.; SHI, F. (Ed.). *Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018 (ISMAC-CVB)*. Cham: Springer International Publishing, 2019. p. 1623–1633. ISBN 978-3-030-00665-5.

ARSENOVIC, M.; KARANOVIC, M.; SLADOJEVIC, S.; ANDERLA, A.; STEFANOVIC, D. Solving current limitations of deep learning based approaches for plant disease detection. *MDPI Open Access Journal - Symmetry*, MDPI, v. 11(7), n. 939, 2019.

ASHQAR, B. A. M.; ABU-NASER, S. S. Image-based tomato leaves diseases detection using deep learning. *International Journal of Academic Engineering Research (IJAER)*, v. 2, n. 1T2, p. 10–16, 2019.

ATABAY, H. A. A convolutional neural network with a new architecture applied on leaf classification. *IIOAB J*, v. 7, n. 5, p. 226–331, 2016.

ATABAY, H. A. Deep residual learning for tomato plant leaf disease identification. *Journal of Theoretical and Applied Information Technology*, v. 95, p. 866, 2017. ISSN 1992-8645. Disponível em: <<http://www.jatit.org/volumes/Vol95No24/10Vol95No24.pdf>>.

ATOLE, R. R.; PARK, D. A multiclass deep convolutional neural network classifier for detection of common rice plant anomalies. *International Journal of Advanced Computer Science and Applications*, The Science and Information Organization, v. 9, n. 1, 2018. Disponível em: <<http://dx.doi.org/10.14569/IJACSA.2018.090109>>.

AUER, P.; HERBSTER, M.; WARMUTH, M. K. Exponentially many local minima for single neurons. In: *Proceedings of the 8th International Conference on Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 1995. (NIPS'95), p. 316–322. Disponível em: <<http://dl.acm.org/citation.cfm?id=2998828.2998873>>.

BACCOUCHE, M.; MAMALET, F.; WOLF, C.; GARCIA, C.; BASKURT, A. Sequential deep learning for human action recognition. In: SPRINGER. *International Workshop on Human Behavior Understanding*. [S.l.], 2011. p. 29–39.

BARBEDO, J. G. Factors influencing the use of deep learning for plant disease recognition. *Biosystems Engineering*, v. 172, p. 84 – 91, 2018. ISSN 1537-5110. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1537511018303027>>.

BARBEDO, J. G. A. A review on the main challenges in automatic plant disease identification based on visible range images. *Biosystems Engineering*, v. 144, p. 52 – 60, 2016. ISSN 1537-5110. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1537511015302476>>.

- BARBEDO, J. G. A. Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Computers and Electronics in Agriculture*, v. 153, p. 46 – 53, 2018. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918304617>>.
- BARBEDO, J. G. A. Plant disease identification from individual lesions and spots using deep learning. *Biosystems Engineering*, v. 180, p. 96 – 107, 2019. ISSN 1537-5110. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1537511018307797>>.
- BENGIO, Y. Learning deep architectures for ai. *Found. Trends Mach. Learn.*, Now Publishers Inc., Hanover, MA, USA, v. 2, n. 1, p. 1–127, jan. 2009. ISSN 1935-8237. Disponível em: <<http://dx.doi.org/10.1561/22000000006>>.
- BENGIO, Y.; SIMARD, P.; FRASCONI, P. Learning long-term dependencies with gradient descent is difficult. *Trans. Neur. Netw.*, IEEE Press, Piscataway, NJ, USA, v. 5, n. 2, p. 157–166, mar. 1994. ISSN 1045-9227. Disponível em: <<https://doi.org/10.1109/72.279181>>.
- BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, JMLR.org, v. 13, n. 1, p. 281–305, fev. 2012. ISSN 1532-4435. Disponível em: <<http://dl.acm.org/citation.cfm?id=2503308.2188395>>.
- BERNAL, J.; KUSHIBAR, K.; ASFAW, D. S.; VALVERDE, S.; OLIVER, A.; MARTÍ, R.; LLADÓ, X. Deep convolutional neural networks for brain image analysis on magnetic resonance imaging: a review. *Artificial Intelligence in Medicine*, v. 95, p. 64–81, 2019. ISSN 0933-3657. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0933365716305206>>.
- BIOLCHINI, J.; MIAN, P. G.; NATALI, A. C. C.; TRAVASSOS, G. H. Systematic review in software engineering. *System Engineering and Computer Science Department COPPE/UFRJ, Technical Report ES*, v. 679, n. 05, p. 45, 2005.
- BISHOP, C. M. *Neural Networks for Pattern Recognition*. New York, NY, USA: Oxford University Press, Inc., 1995. ISBN 0198538642.
- BOS, L.; PARLEVLIET, J. E. Concepts and terminology on plant/pest relationships: Toward consensus in plant pathology and crop protection. *Annual Review of Phytopathology*, v. 33, n. 1, p. 69–102, 1995. PMID: 18288897. Disponível em: <<https://doi.org/10.1146/annurev.py.33.090195.000441>>.
- BRAHIMI, M.; ARSENOVIC, M.; LARABA, S.; SLADOJEVIC, S.; BOUKHALFA, K.; MOUSSAOUI, A. Deep learning for plant diseases: Detection and saliency map visualisation. In: *Human and Machine Learning*. [S.l.: s.n.], 2018.
- BRAHIMI, M.; MAHMOUDI, S.; BOUKHALFA, K.; MOUSSAOUI, A. Deep interpretable architecture for plant diseases classification. In: *2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*. [S.l.: s.n.], 2019. v. 12, n. 8, p. 111–116.
- BRERETON, P.; KITCHENHAM, B. A.; BUDGEN, D.; TURNER, M.; KHALIL, M. Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software*, v. 80, n. 4, p. 571 – 583, 2007. ISSN 0164-1212. Software Performance. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S016412120600197X>>.

- BROMLEY, J.; GUYON, I.; LECUN, Y.; SÄCKINGER, E.; SHAH, R. Signature verification using a "siamese" time delay neural network. In: *Proceedings of the 6th International Conference on Neural Information Processing Systems*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993. (NIPS'93), p. 737–744. Disponível em: <<http://dl.acm.org/citation.cfm?id=2987189.2987282>>.
- BUSCEMI, N.; HARTLING, L.; VANDERMEER, B.; TJOSVOLD, L.; KLASSEN, T. P. Single data extraction generated more errors than double data extraction in systematic reviews. *Journal of Clinical Epidemiology*, v. 59, n. 7, p. 697–703, jul 2006. ISSN 08954356. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/16765272https://linkinghub.elsevier.com/retrieve/pii/S0895435605004117>>.
- CAI, Z.; VASCONCELOS, N. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 43, n. 5, p. 1483–1498, 2021.
- CAP, H. Q.; SUWA, K.; FUJITA, E.; KAGIWADA, S.; UGA, H.; IYATOMI, H. A deep learning approach for on-site plant leaf detection. In: *2018 IEEE 14th International Colloquium on Signal Processing Its Applications (CSPA)*. [S.l.: s.n.], 2018. p. 118–122.
- CARES, J. E.; HUANG, S. P. Taxonomia atual de fitonematoides: chave sistemática simplificada para gêneros. parte i. *Revisão Anual de Patologia de Plantas - RAPP*, p. 185–223, 2000.
- CETINIC, E.; LIPIC, T.; GRGIC, S. Fine-tuning convolutional neural networks for fine art classification. *Expert Systems with Applications*, v. 114, p. 107 – 118, 2018. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417418304421>>.
- CHAPELLE, O.; SCHLKOPF, B.; ZIEN, A. *Semi-Supervised Learning*. 1st. ed. [S.l.]: The MIT Press, 2010. ISBN 0262514125, 9780262514125.
- CHAUVIN, Y.; RUMELHART, D. E. (Ed.). *Backpropagation: Theory, Architectures, and Applications*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1995. ISBN 0-8058-1259-8.
- CHEN, H.; LI, Y.; SU, D. Multi-modal fusion network with multi-scale multi-path and cross-modal interactions for rgb-d salient object detection. *Pattern Recognition*, v. 86, p. 376–385, 2019. ISSN 0031-3203. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0031320318303054>>.
- CHEN, J.; LIU, Q.; GAO, L. Visual tea leaf disease recognition using a convolutional neural network model. *Symmetry*, v. 11, n. 3, 2019. ISSN 2073-8994. Disponível em: <<https://www.mdpi.com/2073-8994/11/3/343>>.
- CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017. p. 1800–1807.
- CHOLLET, F. *et al. Keras API References - Preprocessing Layers*. [S.l.]: Keras.io, 2021. <https://keras.io/api/layers/preprocessinglayers/>.
- CONAB, C. N. d. A. *Conab - Boletim da Safra de Grãos*. 2021. Disponível em: <<https://www.conab.gov.br/info-agro/safra-graos/boletim-da-safra-de-graos>>.

- COOMANS, A.; HEYNS, J. *Oncholaimus sahariensis* sp.n. (nematoda) from the algerian sahara. *Hydrobiologia*, v. 107, n. 3, p. 193–201, Nov 1983. ISSN 1573-5117. Disponível em: <<https://doi.org/10.1007/BF00036688>>.
- CORTES, E. *Plant Disease Classification Using Convolutional Networks and Generative Adversarial Networks*. [S.l.], 2017. Disponível em: <<http://cs231n.stanford.edu/reports/2017/pdfs/325.pdf>>.
- COULIBALY, S.; KAMSU-FOGUEM, B.; KAMISSOKO, D.; TRAORE, D. Deep neural networks with transfer learning in millet crop images. *Computers in Industry*, v. 108, p. 115 – 120, 2019. ISSN 0166-3615. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0166361518305888>>.
- COYNE, D.; NICOL, J.; CLAUDIUS-COLE, B. *Nematologia prática: um guia de campo e de laboratório*. West Africa: International Institute of Tropical Agriculture (IITA), 2007. ISBN 978-131-337-4.
- CYBENKO, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, v. 2, n. 4, p. 303–314, 1989. ISSN 1435-568X. Disponível em: <<https://doi.org/10.1007/BF02551274>>.
- DANG, L.; HASSAN, S. I.; SUHYEON, I.; SANGAIAH, A. K.; MEHMOOD, I.; RHO, S.; SEO, S.; MOON, H. Uav based wilt detection system via convolutional neural networks. In: . [S.l.: s.n.], 2018.
- DAWEI, W.; LIMIAO, D.; JIANGONG, N.; JIYUE, G.; HONGFEI, Z.; ZHONGZHI, H. Recognition pest by image-based transfer learning. *Journal of the Science of Food and Agriculture*, v. 99, n. 10, p. 4524–4531, 2019. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/jsfa.9689>>.
- DECHANT, C.; WIESNER-HANKS, T.; CHEN, S.; STEWART, E. L.; YOSINSKI, J.; GORE, M. A.; NELSON, R. J.; LIPSON, H. Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. *Phytopathology*, v. 107, n. 11, p. 1426–1432, 2017. PMID: 28653579. Disponível em: <<https://doi.org/10.1094/PHYTO-11-16-0417-R>>.
- DENG, J.; DONG, W.; SOCHER, R.; LI, L.-J.; LI, K.; FEI-FEI, L. ImageNet: A Large-Scale Hierarchical Image Database. In: *CVPR09*. [S.l.: s.n.], 2009.
- DIAZ, G.; FOKOUE-NKOUTCHE, A.; NANNICINI, G.; SAMULOWITZ, H. An effective algorithm for hyperparameter optimization of neural networks. *IBM Journal of Research and Development*, v. 61, n. 4, 2017.
- DIVISION, U. N. P. *World Population Prospects 2019*. 2019. 1 p. Disponível em: <<https://population.un.org/wpp/>>.
- DONG, C.; LOY, C. C.; HE, K.; TANG, X. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 38, n. 2, p. 295–307, 2016.
- DONG, N.; ZHAO, L.; WU, C.; CHANG, J. Inception v3 based cervical cell classification combined with artificially extracted features. *Applied Soft Computing*, v. 93, p. 106311, 2020. ISSN 1568-4946. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1568494620302519>>.

- DOUARRE, C.; CRISPIM-JUNIOR, C. F.; GELIBERT, A.; TOUGNE, L.; ROUSSEAU, D. Novel data augmentation strategies to boost supervised segmentation of plant disease. *Computers and Electronics in Agriculture*, v. 165, p. 104967, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169919304879>>.
- DROPKIN, V. H. *Introduction to plant nematology*. New York, USA: John Wiley and Sons Inc., 1989.
- DUDA, R. O.; HART, P. E. *Pattern classification and scene analysis*. Wiley, 1974. (A Wiley-Interscience publication). ISBN 0471223611. Disponível em: <<http://www.worldcat.org/oclc/00388788>>.
- DURMUŞ, H.; GÜNEŞ, E. O.; KIRCI, M. Disease detection on the leaves of the tomato plants by using deep learning. In: *2017 6th International Conference on Agro-Geoinformatics*. [S.l.: s.n.], 2017. p. 1–5.
- ELNEMR, H. A.; ZAYED, N. M.; FAKHRELDEIN, M. A. Handbook of research on emerging perspectives in intelligent pattern recognition, analysis, and image processing. In: _____. Hershey, PA, USA: IGI Global, 2016. cap. Feature Extraction Techniques: Fundamental Concepts and Survey, p. 264–294. ISBN 9781466686540. Disponível em: <<http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-4666-8654-0.ch013>>.
- ELSKEN, T.; METZEN, J. H.; HUTTER, F. Neural architecture search: A survey. *Journal of Machine Learning Research*, v. 20, n. 55, p. 1–21, 2019. Disponível em: <<http://jmlr.org/papers/v20/18-598.html>>.
- EMBRAPA. *Agricultura 4.0: a agricultura conectada*. 2021. Disponível em: <<https://embrapa.br/busca-de-noticias/-noticia/15894563/agricultura-40-a-agricultura-conectada>>.
- ENGELEN, J. E. van; HOOS, H. H. A survey on semi-supervised learning. *Machine Learning*, 2019. ISSN 1573-0565. Disponível em: <<https://doi.org/10.1007/s10994-019-05855-6>>.
- FABBRI, S. C. P. F.; FELIZARDO, K. R.; FERRARI, F. C.; HERNANDES, E. C. M.; OCTAVIANO, F. R.; NAKAGAWA, E. Y.; MALDONADO, J. C. Externalising tacit knowledge of the systematic review process. *IET Software*, The Institution of Engineering and Technology, v. 7, n. 6, p. 298–307, 2013. ISSN 1751-8806.
- FAO, F. *The future of food and agriculture - Trends and Challenges*. 2021. Disponível em: <<https://www.fao.org/global-perspectives-studies/>>.
- FERENTINOS, K. P. Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, v. 145, p. 311 – 318, 2018. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169917311742>>.
- FERRAZ, L.; BROWN, D. *Nematologia de Plantas: fundamentos e importância*. Viçosa - MG: Sociedade Brasileira de Nematologia, 2016. ISBN 978-85-99031-26-1.
- FERRAZ, S.; FREITAS, L. G.; LOPES, E. A.; DIAS-ARIEIRA, C. R. *Manejo Sustentável de Fitonematoides*. 1st. ed. [S.l.]: Editora UFV, 2010. ISBN 978-85-7269-395-0.
- FREITAS, L. G.; O., R. D. L.; FERRAZ, S. *Introdução à Nematologia*. 1. ed. Viçosa - MG: Editora UFV, 2014. v. 1. ISBN 8572690840.

- FUENTES, A.; YOON, S.; KIM, S. C.; PARK, D. S. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, v. 17, n. 9, 2017. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/17/9/2022>>.
- FUENTES, A. F.; YOON, S.; LEE, J.; PARK, D. S. High-performance deep neural network-based tomato plant diseases and pests diagnosis system with refinement filter bank. *Frontiers in Plant Science*, v. 9, p. 1162, 2018. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2018.01162>>.
- FUJITA, E.; KAWASAKI, Y.; UGA, H.; KAGIWADA, S.; IYATOMI, H. Basic investigation on a robust and practical plant diagnostic system. In: *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. [S.l.: s.n.], 2016. p. 989–992.
- FUJITA, E.; UGA, H.; KAGIWADA, S.; IYATOMI, H. A practical plant diagnosis system for field leaf images and feature visualization. In: . [S.l.: s.n.], 2018.
- FUKUMIZU, K.; AMARI, S. Local minima and plateaus in hierarchical structures of multilayer perceptions. *Neural Netw.*, Elsevier Science Ltd., Oxford, UK, UK, v. 13, n. 3, p. 317–327, abr. 2000. ISSN 0893-6080. Disponível em: <[http://dx.doi.org/10.1016/S0893-6080\(00\)00009-5](http://dx.doi.org/10.1016/S0893-6080(00)00009-5)>.
- G., G.; J., A. P. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering*, v. 76, p. 323 – 338, 2019. ISSN 0045-7906. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0045790619300023>>.
- GANDHI, R.; NIMBALKAR, S.; YELAMANCHILI, N.; PONKSHE, S. Plant disease detection using cnns and gans as an augmentative approach. In: *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*. [S.l.: s.n.], 2018. p. 1–5.
- GHOSAL, S.; BLYSTONE, D.; SINGH, A. K.; GANAPATHYSUBRAMANIAN, B.; SINGH, A.; SARKAR, S. An explainable deep machine vision framework for plant stress phenotyping. *Proceedings of the National Academy of Sciences*, National Academy of Sciences, v. 115, n. 18, p. 4613–4618, 2018. ISSN 0027-8424. Disponível em: <<https://www.pnas.org/content/115/18/4613>>.
- GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. In: GORDON, G.; DUNSON, D.; DUDÍK, M. (Ed.). *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Fort Lauderdale, FL, USA: PMLR, 2011. (Proceedings of Machine Learning Research, v. 15), p. 315–323. Disponível em: <<http://proceedings.mlr.press/v15/glorot11a.html>>.
- GODFRAY, H. C. J.; MASON-D’CROZ, D.; ROBINSON, S. Food system consequences of a fungal disease epidemic in a major crop. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, The Royal Society, v. 371, n. 1709, p. 20150467, Dec 2016. ISSN 1471-2970. Disponível em: <<https://pubmed.ncbi.nlm.nih.gov/28080990>>.
- GOLATKAR, A.; ANAND, D.; SETHI, A. Classification of breast cancer histology using deep learning. In: CAMPILHO, A.; KARRAY, F.; ROMENY, B. ter H. (Ed.). *Image Analysis and Recognition*. Cham: Springer International Publishing, 2018. p. 837–844. ISBN 978-3-319-93000-8.
- GONCHAROV, P.; OSOSKOV, G.; NECHAEVSKIY, A.; UZHINSKIY, A.; NESTSIARENIA, I. Disease detection on the plant leaves by deep learning. In: KRYZHANOVSKY, B.; DUNIN-BARKOWSKI, W.; REDKO, V.; TIUMENTSEV, Y. (Ed.). *Advances in Neural*

Computation, Machine Learning, and Cognitive Research II. Cham: Springer International Publishing, 2019. p. 151–159. ISBN 978-3-030-01328-8.

GONZAGA, V. *CARACTERIZAÇÃO MORFOLÓGICA, MORFOMÉTRICA E MULTIPLICAÇÃO IN VITRO DAS SEIS ESPÉCIES MAIS COMUNS DE Pratylenchus Filipjev, 1936 QUE OCORREM NO BRASIL*. Tese (Doutorado) — UNESP, 2006. Disponível em: <<https://repositorio.unesp.br/handle/11449/105288>>.

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

GU, J.; WANG, Z.; KUEN, J.; MA, L.; SHAHROUDY, A.; SHUAI, B.; LIU, T.; WANG, X.; WANG, G.; CAI, J.; CHEN, T. Recent advances in convolutional neural networks. *Pattern Recognition*, v. 77, p. 354–377, 2018. ISSN 0031-3203. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0031320317304120>>.

HA, J. G.; MOON, H.; KWAK, J. T.; HASSAN, S. I.; DANG, M.; LEE, O. N.; PARK, H. Y. Deep convolutional neural network for classifying fusarium wilt of radish from unmanned aerial vehicles. *Journal of Applied Remote Sensing*, v. 11, n. 4, p. 1 – 14 – 14, 2017. Disponível em: <<https://doi.org/10.1117/1.JRS.11.042621>>.

HAN, D.; KIM, J.; KIM, J. Deep pyramidal residual networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017. p. 6307–6315.

HAN, D.; LIU, Q.; FAN, W. A new image classification method using cnn transfer learning and web data augmentation. *Expert Systems with Applications*, v. 95, p. 43–56, 2018. ISSN 0957-4174. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0957417417307844>>.

HANG, J.; ZHANG, D.; CHEN, P.; ZHANG, J.; WANG, B. Classification of plant leaf diseases based on improved convolutional neural network. *Sensors (Basel, Switzerland)*, MDPI, v. 19, n. 19, p. 4161, Sep 2019. ISSN 1424-8220. Disponível em: <<https://pubmed.ncbi.nlm.nih.gov/31557958>>.

HARI, S. S.; SIVAKUMAR, M.; RENUGA, P.; KARTHIKEYAN, S.; SURIYA, S. Detection of plant disease by leaf image using convolutional neural network. In: *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*. [S.l.: s.n.], 2019. v. 8, n. 14, p. 1–5.

HASAN, M. M.; CHOPIN, J. P.; LAGA, H.; MIKLAVCIC, S. J. Detection and analysis of wheat spikes using convolutional neural networks. *Plant Methods*, v. 15, Mar 2019. ISSN 1746-4811. Disponível em: <<https://doi.org/10.1186/s13007-019-0405-0>>.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The elements of statistical learning: data mining, inference and prediction*. 2. ed. Springer, 2009. Disponível em: <<http://www-stat.stanford.edu/~tibs/ElemStatLearn/>>.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer New York, 2009. (Springer Series in Statistics). ISBN 9780387848587. Disponível em: <<https://books.google.com.br/books?id=tVIjmNS3Ob8C>>.

HAWKINS, D. M. The problem of overfitting. *Journal of Chemical Information and Computer Sciences*, v. 44, n. 1, p. 1–12, 2004. Disponível em: <<https://doi.org/10.1021/ci0342472>>.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778.

HERNANDES, E. M.; ZAMBONI, A.; FABBRI, S.; THOMMAZO, A. D. Using GQM and TAM to evaluate StArt - a tool that supports systematic review. *CLEI Electronic Journal*, v. 15, n. 1, 2012.

HINTON, G. E.; SRIVASTAVA, N.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. R. *Improving neural networks by preventing co-adaptation of feature detectors*. 2012.

HOCHREITER, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, World Scientific Publishing Co., Inc., River Edge, NJ, USA, v. 6, n. 2, p. 107–116, abr. 1998. ISSN 0218-4885. Disponível em: <<http://dx.doi.org/10.1142/S0218488598000094>>.

HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, v. 2, n. 5, p. 359 – 366, 1989. ISSN 0893-6080. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0893608089900208>>.

HOWARD, A. G.; ZHU, M.; CHEN, B.; KALENICHENKO, D.; WANG, W.; WEYAND, T.; ANDREETTO, M.; ADAM, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv*, abs/1704.04861, 2017.

HOWLADER, M. R.; HABIBA, U.; FAISAL, R. H.; RAHMAN, M. M. Automatic recognition of guava leaf diseases using deep convolution neural network. In: *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. [S.l.: s.n.], 2019. v. 5, n. 2, p. 1–5.

HU, G.; YANG, X.; ZHANG, Y.; WAN, M. Identification of tea leaf diseases by using an improved deep convolutional neural network. *Sustainable Computing: Informatics and Systems*, v. 24, p. 100353, 2019. ISSN 2210-5379. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S221053791930201X>>.

HUANG, G.; LIU, Z.; MAATEN, L. v. d.; WEINBERGER, K. Q. Densely connected convolutional networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017. p. 2261–2269.

HUANG, G.; SUN, Y.; LIU, Z.; SEDRA, D.; WEINBERGER, K. Q. Deep networks with stochastic depth. In: LEIBE, B.; MATAS, J.; SEBE, N.; WELLING, M. (Ed.). *Computer Vision – ECCV 2016*. Cham: Springer International Publishing, 2016. p. 646–661. ISBN 978-3-319-46493-0.

HUANG, S.; LIU, W.; QI, F.; YANG, K. Development and validation of a deep learning algorithm for the recognition of plant disease. In: *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. [S.l.: s.n.], 2019. v. 5, n. 25, p. 1951–1957.

HUGHES, D. P.; SALATHÉ, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. *CoRR*, abs/1511.08060, 2016. Disponível em: <<http://arxiv.org/abs/1511.08060>>.

- HUSIN, Z. B.; SHAKAFF, A. Y. B. M.; AZIZ, A. H. B. A.; FAROOK, R. B. S. M. Feasibility study on plant chili disease detection using image processing techniques. In: *2012 Third International Conference on Intelligent Systems Modelling and Simulation*. [S.l.: s.n.], 2012. p. 291–296.
- IANDOLA, F. N.; HAN, S.; MOSKEWICZ, M. W.; ASHRAF, K.; DALLY, W. J.; KEUTZER, K. *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size*. 2016.
- JI, S.; XU, W.; YANG, M.; YU, K. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 35, n. 1, p. 221–231, 2013.
- JIA, Y.; SHELHAMER, E.; DONAHUE, J.; KARAYEV, S.; LONG, J.; GIRSHICK, R.; GUADARRAMA, S.; DARRELL, T. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.
- JIANG, P.; CHEN, Y.; LIU, B.; HE, D.; LIANG, C. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access*, v. 7, p. 59069–59080, 2019.
- JIN, S.; SU, Y.; GAO, S.; WU, F.; HU, T.; LIU, J.; LI, W.; WANG, D.; CHEN, S.; JIANG, Y.; PANG, S.; GUO, Q. Deep learning: Individual maize segmentation from terrestrial lidar data using faster r-cnn and regional growth algorithms. *Frontiers in Plant Science*, v. 9, p. 866, 2018. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2018.00866>>.
- KARIM, R. *Illustrated: 10 CNN Architectures - A compiled visualisation of the common convolutional neural networks*. 2019. Disponível em: <<https://towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d#bca5>>.
- KARPATHY, A.; TODERICI, G.; SHETTY, S.; LEUNG, T.; SUKTHANKAR, R.; FEI-FEI, L. Large-scale video classification with convolutional neural networks. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2014. p. 1725–1732.
- KAUR, S.; PANDEY, S.; GOEL, S. Plants disease identification and classification through leaf images: A survey. *Archives of Computational Methods in Engineering*, v. 26, n. 2, p. 507–530, Apr 2019. ISSN 1886-1784. Disponível em: <<https://doi.org/10.1007/s11831-018-9255-6>>.
- KAWASAKI, Y.; UGA, H.; KAGIWADA, S.; IYATOMI, H. Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In: BEBIS, G.; BOYLE, R.; PARVIN, B.; KORACIN, D.; PAVLIDIS, I.; FERIS, R.; MCGRAW, T.; ELENDDT, M.; KOPPER, R.; RAGAN, E.; YE, Z.; WEBER, G. (Ed.). *Advances in Visual Computing*. Cham: Springer International Publishing, 2015. p. 638–645. ISBN 978-3-319-27863-6.
- KAYA, A.; KECELI, A. S.; CATAL, C.; YALIC, H. Y.; TEMUCIN, H.; TEKINERDOGAN, B. Analysis of transfer learning for deep neural network based plant classification models. *Computers and Electronics in Agriculture*, v. 158, p. 20 – 29, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918315308>>.
- KC, K.; YIN, Z.; WU, M.; WU, Z. Depthwise separable convolution architectures for plant disease classification. *Computers and Electronics in Agriculture*, v. 165, p. 104948, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918318696>>.

KENSERT, A.; HARRISON, P. J.; SPJUTH, O. Transfer learning with deep convolutional neural networks for classifying cellular morphological changes. *SLAS DISCOVERY: Advancing the Science of Drug Discovery*, v. 24, n. 4, p. 466–475, 2019.

KERKECH, M.; HAFIANE, A.; CANALS, R. Deep learning approach with colorimetric spaces and vegetation indices for vine diseases detection in uav images. *Computers and Electronics in Agriculture*, v. 155, p. 237 – 243, 2018. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918310044>>.

KHAMPARIA, A.; SAINI, G.; GUPTA, D.; KHANNA, A.; TIWARI, S.; ALBUQUERQUE, V. H. C. de. Seasonal crops disease prediction and classification using deep convolutional encoder network. *Circuits, Systems, and Signal Processing*, Jan 2019. ISSN 1531-5878. Disponível em: <<https://doi.org/10.1007/s00034-019-01041-0>>.

KHAN, A.; SOHAIL, A.; ZAHOORA, U.; QURESHI, A. S. A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, v. 53, n. 8, p. 5455–5516, Dec 2020. ISSN 1573-7462.

KHAN, M. A.; AKRAM, T.; SHARIF, M.; AWAIS, M.; JAVED, K.; ALI, H.; SABA, T. Ccdf: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep cnn features. *Computers and Electronics in Agriculture*, v. 155, p. 220 – 236, 2018. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918303120>>.

KHANDELWAL, I.; RAMAN, S. Analysis of transfer and residual learning for detecting plant diseases using images of leaves. In: VERMA, N. K.; GHOSH, A. K. (Ed.). *Computational Intelligence: Theories, Applications and Future Directions - Volume II*. Singapore: Springer Singapore, 2019. p. 295–306. ISBN 978-981-13-1135-2.

KHOSRAVI, P.; KAZEMI, E.; IMIELINSKI, M.; ELEMENTO, O.; HAJIRASOULIHA, I. Deep convolutional neural networks enable discrimination of heterogeneous digital pathology images. *EBioMedicine*, v. 27, p. 317–328, 2018. ISSN 2352-3964. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2352396417305078>>.

KISWANI, A.; VERNER, U. *Performance comparison of basic image processing algorithms on different architectures*. 2014. 1 p. Disponível em: <<https://sipl.eelabs.technion.ac.il/wp-content/uploads/sites/6/2016/10/poster-1599-2-13.png>>.

KITCHENHAM, B. *Procedures for Performing Systematic Reviews*. Department of Computer Science, Keele University, UK, 2004.

KITCHENHAM, B.; CHARTERS, S. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. [S.l.], 2007. Disponível em: <<http://www.dur.ac.uk/ebse/resources/Systematic-reviews-5-8.pdf>>.

KITCHENHAM, B. A. Systematic review in software engineering: Where we are and where we should be going. In: *Proceedings of the 2Nd International Workshop on Evidential Assessment of Software Technologies*. New York, NY, USA: ACM, 2012. (EAST '12), p. 1–2. ISBN 978-1-4503-1509-8. Disponível em: <<http://doi.acm.org/10.1145/2372233.2372235>>.

KOBAYASHI, K.; TSUJI, J.; NOTO, M. Evaluation of data augmentation for image-based plant-disease detection. In: *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.: s.n.], 2018. p. 2206–2211. ISSN 2577-1655.

- KOHAVI, R. *et al.* A study of cross-validation and bootstrap for accuracy estimation and model selection. In: STANFORD, CA. *Ijcai*. [S.l.], 1995. v. 14, p. 1137–1145.
- KOLEN, J. F.; KREMER, S. C. Gradient flow in recurrent nets: The difficulty of learning longterm dependencies. In: _____. *A Field Guide to Dynamical Recurrent Networks*. IEEE, 2001. p. 237–243. ISBN null. Disponível em: <<https://ieeexplore.ieee.org/document/5264952>>.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*. USA: Curran Associates Inc., 2012. (NIPS'12), p. 1097–1105. Disponível em: <<http://dl.acm.org/citation.cfm?id=2999134.2999257>>.
- KUEN, J.; KONG, X.; WANG, G. Delugenets: Deep networks with massive and flexible cross-layer information inflows. *CoRR*, abs/1611.05552, 2018. Disponível em: <<http://arxiv.org/abs/1611.05552>>.
- KUEN, J.; KONG, X.; WANG, G.; TAN, Y.-P. Delugenets: Deep networks with efficient and flexible cross-layer information inflows. In: *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. [S.l.: s.n.], 2017. p. 958–966.
- LABAÑA, F. M.; RUIZ, A.; GARCÍA-SÁNCHEZ, F. Pestdetect: Pest recognition using convolutional neural network. In: VALENCIA-GARCÍA, R.; ALCARAZ-MÁRMOL, G.; CIOPPO-MORSTADT, J. d.; VERA-LUCIO, N.; BUCARAM-LEVERONE, M. (Ed.). *ICT for Agriculture and Environment*. Cham: Springer International Publishing, 2019. p. 99–108. ISBN 978-3-030-10728-4.
- LAKATOS, E. M.; MARCONI, M. de A. *et al.* *Metodologia científica*. [S.l.]: Atlas, 2007.
- LAMBERT, K.; BEKAL, S. Introduction to plant-parasitic nematodes. *The Plant Health Instructor*, 2002.
- LARSSON, G.; MAIRE, M.; SHAKHNAROVICH, G. Fractalnet: Ultra-deep neural networks without residuals. *CoRR*, abs/1605.07648, 2016.
- LECUN, Y.; BOSER, B.; DENKER, J. S.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W.; JACKEL, L. D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, v. 1, n. 4, p. 541–551, 12 1989. ISSN 0899-7667.
- LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, Nov 1998.
- LEE, S.-H.; WU, C.-C.; CHEN, S.-F. Development of Image Recognition and Classification Algorithm for Tea Leaf Diseases Using Convolutional Neural Network. In: *2018 Detroit, Michigan July 29 - August 1, 2018*. St. Joseph, MI: American Society of Agricultural and Biological Engineers, 2018. p. 1.
- LI, J.; JIA, J.; XU, D. Unsupervised representation learning of image-based plant disease with deep convolutional generative adversarial networks. In: *2018 37th Chinese Control Conference (CCC)*. [S.l.: s.n.], 2018. v. 2, n. 10, p. 9159–9163. ISSN 1934-1768.
- LI, Y.; QIAN, M.; LIU, P.; CAI, Q.; LI, X.; GUO, J.; YAN, H.; YU, F.; YUAN, K.; YU, J.; QIN, L.; LIU, H.; WU, W.; XIAO, P.; ZHOU, Z. The recognition of rice images by uav based on capsule network. *Cluster Computing*, Mar 2018. ISSN 1573-7543. Disponível em: <<https://doi.org/10.1007/s10586-018-2482-7>>.

- LIANG, Q.; XIANG, S.; HU, Y.; COPPOLA, G.; ZHANG, D.; SUN, W. Pd2se-net: Computer-assisted plant disease diagnosis and severity estimation network. *Computers and Electronics in Agriculture*, v. 157, p. 518 – 529, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918318313>>.
- LIANG, W.-j.; ZHANG, H.; ZHANG, G.-f.; CAO, H.-x. Rice Blast Disease Recognition Using a Deep Convolutional Neural Network. *Scientific Reports*, Nature Publishing Group, v. 9, n. 1, p. 2869, dec 2019. ISSN 2045-2322. Disponível em: <<http://www.nature.com/articles/s41598-019-38966-0>>.
- LIASHCHYNSKYI, P.; LIASHCHYNSKYI, P. Grid search, random search, genetic algorithm: a big comparison for nas. *arXiv preprint arXiv:1912.06059*, 2019.
- LIM, H. P. M.; ESTUAR, M. R. J. E. Microscopic fusarium detection and verification with convolutional neural networks. In: *Proceedings of the 2018 International Conference on Machine Learning Technologies*. New York, NY, USA: ACM, 2018. (ICMLT '18), p. 48–52. ISBN 978-1-4503-6432-4. Disponível em: <<http://doi.acm.org/10.1145/3231884.3231892>>.
- LIN, C.; DING, Q.; TU, W.; HUANG, J.; LIU, J. Fourier dense network to conduct plant classification using uav-based optical images. *IEEE Access*, v. 7, p. 17736–17749, 2019. ISSN 2169-3536.
- LIN, K.; GONG, L.; HUANG, Y.; LIU, C.; PAN, J. Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network. *Frontiers in Plant Science*, v. 10, p. 155, 2019. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2019.00155>>.
- LIN, T.; DOLLAR, P.; GIRSHICK, R.; HE, K.; HARIHARAN, B.; BELONGIE, S. Feature pyramid networks for object detection. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2017. p. 936–944. ISSN 1063-6919. Disponível em: <<https://doi.ieeecomputersociety.org/10.1109/CVPR.2017.106>>.
- LIN, Z.; HUANG, J.; CHEN, Y.; ZHANG, X.; ZHAO, W.; LI, Y.; LU, L.; ZHAN, M.; JIANG, X.; LIANG, X. A high resolution representation network with multi-path scale for retinal vessel segmentation. *Computer Methods and Programs in Biomedicine*, v. 208, p. 106206, 2021. ISSN 0169-2607. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0169260721002807>>.
- LIN, Z.; MU, S.; HUANG, F.; MATEEN, K. A.; WANG, M.; GAO, W.; JIA, J. A unified matrix-based convolutional neural network for fine-grained image classification of wheat leaf diseases. *IEEE Access*, v. 7, n. 25, p. 11570–11590, 2019.
- LIU, B.; ZHANG, Y.; HE, D.; LI, Y. Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry*, v. 10, n. 1, 2017. ISSN 2073-8994. Disponível em: <<https://www.mdpi.com/2073-8994/10/1/11>>.
- LONG, Y.; LIU, C. Research on deep learning method of crop disease identification. In: *Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing*. New York, NY, USA: Association for Computing Machinery, 2019. (AIIPCC '19). ISBN 9781450376334. Disponível em: <<https://doi.org/10.1145/3371425.3371497>>.

- LU, Y.; YI, S.; ZENG, N.; LIU, Y.; ZHANG, Y. Identification of rice diseases using deep convolutional neural networks. *Neurocomputing*, v. 267, p. 378 – 384, 2017. ISSN 0925-2312. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0925231217311384>>.
- 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: *TENCON 2018 - 2018 IEEE Region 10 Conference*. [S.l.: s.n.], 2018. p. 1414–1419. ISSN 2159-3450.
- LUXBURG, U. von; SCHÖLKOPF, B. Statistical learning theory: Models, concepts, and results. In: GABBAY, D. M.; HARTMANN, S.; WOODS, J. (Ed.). *Inductive Logic*. North-Holland, 2011, (Handbook of the History of Logic, v. 10). p. 651 – 706. Disponível em: <<http://www.sciencedirect.com/science/article/pii/B9780444529367500161>>.
- LV, E.; WANG, X.; CHENG, Y.; YU, Q. Deep ensemble network based on multi-path fusion. *Artificial Intelligence Review*, v. 52, n. 1, p. 151–168, Jun 2019. ISSN 1573-7462.
- MA, J.; DU, K.; ZHENG, F.; ZHANG, L.; GONG, Z.; SUN, Z. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Computers and Electronics in Agriculture*, v. 154, p. 18 – 24, 2018. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918309360>>.
- MACHADO, A. C. Z.; SILVA, S. A. da; FERRAZ, L. C. C. B. *Métodos em Nematologia Agrícola*. 1st. ed. Piracicaba - SP: Sociedade Brasileira de Nematologia, 2019. ISBN 012372578X.
- MAFRA, S. N.; TRAVASSOS, G. H. *Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software*. [S.l.], 2006.
- MAI, W. F.; MULLIN, P. G.; LYON, H. H.; LOEFFLER, K. *Plant-Parasitic Nematodes: A Pictorial Key to Genera*. 5. ed. Cornell University Press, 1996. ISBN 9780801431166. Disponível em: <<http://www.jstor.org/stable/10.7591/j.ctv5rdz0t>>.
- MIKOAJCZYK, A.; GROCHOWSKI, M. Data augmentation for improving deep learning in image classification problem. In: *2018 International Interdisciplinary PhD Workshop (IIPhDW)*. [S.l.: s.n.], 2018. p. 117–122.
- MITCHELL, T. M. *Machine Learning*. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072.
- MODARRES, M. H.; AVERSA, R.; COZZINI, S.; CIANCIO, R.; LETO, A.; BRANDINO, G. P. Neural network for nanoscience scanning electron microscope image recognition. *Scientific Reports*, v. 7, n. 1, p. 13282, Oct 2017. ISSN 2045-2322.
- MOHANTY, S. P.; HUGHES, D. P.; SALATHÉ, M. Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, v. 7, p. 1419, 2016. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2016.01419>>.
- MOHER, D.; LIBERATI, A.; ALTMAN, D. G.; TETZLAFF, J.; MULROW, C.; GØTZSCHE, P. C.; IOANNIDIS, J. P. A.; CLARKE, M.; DEVEREAUX, P. J.; KLEIJNEN, J. The prisma statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: Explanation and elaboration. *PLOS Medicine*, Public Library of Science, v. 6, n. 7, p. 1–28, 07 2009. Disponível em: <<https://doi.org/10.1371/journal.pmed.1000100>>.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations of Machine Learning*. 2nd. ed. [S.l.]: The MIT Press, 2018. ISBN 0262039400, 9780262039406.

MURPHY, K. P. *Machine Learning: A Probabilistic Perspective*. [S.l.]: The MIT Press, 2012. ISBN 0262018020, 9780262018029.

NACHTIGALL, L. G.; ARAUJO, R. M.; NACHTIGALL, G. R. Classification of apple tree disorders using convolutional neural networks. In: *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*. [S.l.: s.n.], 2016. p. 472–476. ISSN 2375-0197.

NAGASUBRAMANIAN, K.; JONES, S.; SINGH, A. K.; SARKAR, S.; SINGH, A.; GANAPATHYSUBRAMANIAN, B. Plant disease identification using explainable 3d deep learning on hyperspectral images. *Plant Methods*, v. 15, n. 1, p. 98, 2019. ISSN 1746-4811. Disponível em: <<https://doi.org/10.1186/s13007-019-0479-8>>.

NAGI, J.; DUCATELLE, F.; CARO, G. A. D.; CIREŞAN, D.; MEIER, U.; GIUSTI, A.; NAGI, F.; SCHMIDHUBER, J.; GAMBARDELLA, L. M. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In: *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*. [S.l.: s.n.], 2011. p. 342–347. ISSN null.

NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. USA: Omnipress, 2010. (ICML'10), p. 807–814. ISBN 978-1-60558-907-7. Disponível em: <<http://dl.acm.org/citation.cfm?id=3104322.3104425>>.

NAKAGAWA, E. Y.; SCANNAVINO, K. R. F.; FABBRI, S. C. P. F.; FERRARI, F. C. *Revisão sistemática da literatura em Engenharia de Software: teoria e prática*. [S.l.]: Elsevier Brasil, 2017.

OCAMPO, A. L. P. d.; DADIOS, E. P. Mobile platform implementation of lightweight neural network model for plant disease detection and recognition. In: *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*. [S.l.: s.n.], 2018. p. 1–4.

OLIVEIRA, C. M. G. de; INAMOTO, M. M.; BESSI, R.; TOMAZINI, M. D.; BLOK, V. C. Técnicas moleculares e taxonomia clássica na diagnose de nematodes parasitos de plantas. *Revisão Anual de Patologia de Plantas - RAPP*, p. 309–336, 2011.

OPPENHEIM, D.; SHANI, G.; ERLICH, O.; TSROR, L. Using deep learning for image-based potato tuber disease detection. *Phytopathology*, v. 109, n. 6, p. 1083–1087, 2019. PMID: 30543489. Disponível em: <<https://doi.org/10.1094/PHYTO-08-18-0288-R>>.

OZGUVEN, M. M.; ADEM, K. Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms. *Physica A: Statistical Mechanics and its Applications*, v. 535, p. 122537, 2019. ISSN 0378-4371. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0378437119314529>>.

PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 22, n. 10, p. 1345–1359, Oct 2010. ISSN 1041-4347.

PANG, J.; BAI, Z.; LAI, J. chen; LI, S. Automatic segmentation of crop leaf spot disease images by integrating local threshold and seeded region growing. In: *2011 International Conference on Image Analysis and Signal Processing*. [S.l.: s.n.], 2011. p. 590–594.

- PANG, J.; CHEN, K.; SHI, J.; FENG, H.; OUYANG, W.; LIN, D. Libra r-cnn: Towards balanced learning for object detection. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2019. p. 821–830. Disponível em: <<https://doi.ieeecomputersociety.org/10.1109/CVPR.2019.00091>>.
- PARDEDE, H. F.; SURYAWATI, E.; SUSTIKA, R.; ZILVAN, V. Unsupervised convolutional autoencoder-based feature learning for automatic detection of plant diseases. In: *2018 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*. [S.l.: s.n.], 2018. p. 158–162.
- PARK, H.; EUN, J.; KIM, S. Image-based disease diagnosing and predicting of the crops through the deep learning mechanism. In: *2017 International Conference on Information and Communication Technology Convergence (ICTC)*. [S.l.: s.n.], 2017. p. 129–131.
- PARK, K.; HONG, Y. ki; KIM, G. hwan; LEE, J. Classification of apple leaf conditions in hyper-spectral images for diagnosis of marssonina blotch using mrmr and deep neural network. *Computers and Electronics in Agriculture*, v. 148, p. 179 – 187, 2018. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169917308050>>.
- PAWARA., P.; OKAFOR., E.; SURINTA., O.; SCHOMAKER., L.; WIERING., M. Comparing local descriptors and bags of visual words to deep convolutional neural networks for plant recognition. In: INSTICC. *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM.* [S.l.]: SciTePress, 2017. p. 479–486. ISBN 978-989-758-222-6.
- PENG, Z.; CAI, C. An effective segmentation algorithm of apple watercore disease region using fully convolutional neural networks. In: *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. [S.l.: s.n.], 2017. v. 51, n. 12, p. 1292–1299.
- PERRY, R. N.; MOENS, M. *Introduction to Plant-Parasitic Nematodes; Modes of Parasitism*. Dordrecht: Springer Netherlands, 2011. 3–20 p. ISBN 978-94-007-0434-3. Disponível em: <<https://doi.org/10.1007/978-94-007-0434-31>>.
- PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. Systematic mapping studies in software engineering. In: *12th International Conference on Evaluation and Assessment in Software Engineering (EASE)*. Swinton, UK, UK: British Computer Society, 2008. p. 68–77.
- PICON, A.; ALVAREZ-GILA, A.; SEITZ, M.; ORTIZ-BARREDO, A.; ECHAZARRA, J.; JOHANNES, A. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Computers and Electronics in Agriculture*, v. 161, p. 280 – 290, 2018. ISSN 0168-1699. BigData and DSS in Agriculture. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169917312619>>.
- PICON, A.; SEITZ, M.; ALVAREZ-GILA, A.; MOHNKE, P.; ORTIZ-BARREDO, A.; ECHAZARRA, J. Crop conditional convolutional neural networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field conditions. *Computers and Electronics in Agriculture*, v. 167, p. 105093, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169919309329>>.
- POLDER, G.; BLOK, P. M.; VILLIERS, H. A. C. de; WOLF, J. M. van der; KAMP, J. Potato virus y detection in seed potatoes using deep learning on hyperspectral images.

- Frontiers in Plant Science*, v. 10, p. 209, 2019. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2019.00209>>.
- POWERS, D. M. W. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, v. 2, n. 1, p. 37–63, 2011.
- RAHMAN, C. R.; ARKO, P. S.; ALI, M. E.; KHAN, M. A. I.; WASIF, A.; JANI, M. R.; KABIR, M. S. Identification and recognition of rice diseases and pests using deep convolutional neural networks. *CoRR*, abs/1812.01043, 2018. Disponível em: <<http://arxiv.org/abs/1812.01043>>.
- RAMCHARAN, A.; BARANOWSKI, K.; MCCLOSKEY, P.; AHMED, B.; LEGG, J.; HUGHES, D. P. Deep learning for image-based cassava disease detection. *Frontiers in Plant Science*, v. 8, p. 1852, 2017. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2017.01852>>.
- RAMCHARAN, A.; MCCLOSKEY, P.; BARANOWSKI, K.; MBILINYI, N.; MRISHO, L.; NDALAHWA, M.; LEGG, J.; HUGHES, D. P. A mobile-based deep learning model for cassava disease diagnosis. *Frontiers in Plant Science*, v. 10, p. 272, 2019. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2019.00272>>.
- RANCON, F.; BOMBRUN, L.; KERESZTES, B.; GERMAIN, C. Comparison of sift encoded and deep learning features for the classification and detection of esca disease in bordeaux vineyards. *Remote Sensing*, v. 11, n. 1, 2018. ISSN 2072-4292. Disponível em: <<https://www.mdpi.com/2072-4292/11/1/1>>.
- RANGARAJAN, A. K.; PURUSHOTHAMAN, R.; RAMESH, A. Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Computer Science*, v. 133, p. 1040 – 1047, 2018. ISSN 1877-0509. International Conference on Robotics and Smart Manufacturing (RoSMa2018). Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1877050918310159>>.
- REN, L.; HU, M.; FANG, Y.; DU, X.; FENG, H. Recognition of common pests in agriculture and forestry based on convolutional neural networks. In: *2018 Chinese Automation Congress (CAC)*. [S.l.: s.n.], 2018. v. 08, n. 11, p. 2985–2989.
- ROEDER, A. Computational image analysis for microscopy. *The Plant Cell*, American Society of Plant Biologists, v. 31, n. 10, 2019. ISSN 1040-4651. Disponível em: <<http://www.plantcell.org/content/31/10/tpc.119.tt0819>>.
- ROJAS-ALBARRACÍN, G.; CARBAJAL, C. A.; FERNÁNDEZ-CABALLERO, A.; LÓPEZ, M. T. Skeleton simplification by key points identification. In: *Proceedings of the 2nd Mexican Conference on Pattern Recognition: Advances in Pattern Recognition*. Berlin, Heidelberg: Springer-Verlag, 2010. (MCPR'10), p. 30–39. ISBN 3642159915.
- ROSENBLATT, F. F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, v. 65, p. 386–408, 1958.
- RUMPF, T.; MAHLEIN, A.-K.; STEINER, U.; OERKE, E.-C.; DEHNE, H.-W.; PLÜMER, L. Early detection and classification of plant diseases with support vector machines based on hyperspectral reflectance. *Computers and Electronics in Agriculture*, v. 74, n. 1, p. 91 – 99, 2010. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169910001262>>.

- RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATY, A.; KHOSLA, A.; BERNSTEIN, M. S.; BERG, A. C.; LI, F. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014. Disponível em: <<http://arxiv.org/abs/1409.0575>>.
- SABOUR, S.; FROSST, N.; HINTON, G. E. Dynamic routing between capsules. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. USA: Curran Associates Inc., 2017. (NIPS'17), p. 3859–3869. ISBN 978-1-5108-6096-4. Disponível em: <<http://dl.acm.org/citation.cfm?id=3294996.3295142>>.
- SAINATH, T. N.; MOHAMED, A. r.; KINGSBURY, B.; RAMABHADRAN, B. Deep convolutional neural networks for lvcsr. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. [S.l.: s.n.], 2013. p. 8614–8618. ISSN 1520-6149.
- SARDOGAN, M.; TUNCER, A.; OZEN, Y. Plant leaf disease detection and classification based on cnn with lvq algorithm. In: *2018 3rd International Conference on Computer Science and Engineering (UBMK)*. [S.l.: s.n.], 2018. p. 382–385.
- SBN, S. B. d. N. *Perdas causadas por Nematoides*. 2019. Disponível em: <<https://nematologia.com.br/files/livros/1.pdf>>.
- SELVARAJ, M. G.; VERGARA, A.; RUIZ, H.; SAFARI, N.; ELAYABALAN, S.; OCIMATI, W.; BLOMME, G. Ai-powered banana diseases and pest detection. *Plant Methods*, v. 15, n. 1, p. 92, 2019. ISSN 1746-4811. Disponível em: <<https://doi.org/10.1186/s13007-019-0475-z>>.
- SETTLES, B. *Active learning literature survey*. [S.l.], 2009.
- SHAKEEL, M. F.; BAJWA, N. A.; ANWAAR, A. M.; SOHAIL, A.; KHAN, A.; RASHID, H. ur. Detecting driver drowsiness in real time through deep learning based object detection. In: ROJAS, I.; JOYA, G.; CATALA, A. (Ed.). *Advances in Computational Intelligence*. Cham: Springer International Publishing, 2019. p. 283–296. ISBN 978-3-030-20521-8.
- SHAO, L.; ZHU, F.; LI, X. Transfer learning for visual categorization: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, v. 26, n. 5, p. 1019–1034, May 2015. ISSN 2162-2388.
- SHARMA, P.; BERWAL, Y. P. S.; GHAI, W. Performance analysis of deep learning cnn models for disease detection in plants using image segmentation. *Information Processing in Agriculture*, 2019. ISSN 2214-3173. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2214317319301957>>.
- SHORTEN, C.; KHOSHGOFTAAR, T. M. A survey on image data augmentation for deep learning. *Journal of Big Data*, v. 6, n. 1, p. 60, Jul 2019. ISSN 2196-1115. Disponível em: <<https://doi.org/10.1186/s40537-019-0197-0>>.
- SIBIYA, M.; SUMBWANYAMBE, M. A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks. *AgriEngineering*, v. 1, n. 1, p. 119–131, 2019. ISSN 2624-7402. Disponível em: <<https://www.mdpi.com/2624-7402/1/1/9>>.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

- SINGH, A. K.; GANAPATHYSUBRAMANIAN, B.; SARKAR, S.; SINGH, A. Deep learning for plant stress phenotyping: Trends and future perspectives. *Trends in Plant Science*, v. 23, n. 10, p. 883 – 898, 2018. ISSN 1360-1385. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1360138518301572>>.
- SINGH, K. K. An artificial intelligence and cloud based collaborative platform for plant disease identification, tracking and forecasting for farmers. In: *2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*. [S.l.: s.n.], 2018. p. 49–56.
- SINGH, U. P.; CHOUHAN, S. S.; JAIN, S.; JAIN, S. Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. *IEEE Access*, v. 7, n. 3, p. 43721–43729, Jul 2019. ISSN 2169-3536.
- SLADOJEVIC, S.; ARSENOVIC, M.; ANDERLA, A.; CULIBRK, D.; STEFANOVIC, D. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, Hindawi, v. 2016, p. 1–11, jun 2016. ISSN 1687-5265. Disponível em: <<http://www.hindawi.com/journals/cin/2016/3289801/>>.
- SMITH, L. N. Cyclical learning rates for training neural networks. In: *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. [S.l.: s.n.], 2017. p. 464–472.
- SNOEK, J.; LAROCHELLE, H.; ADAMS, R. P. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, v. 25, 2012.
- SNOEK, J.; LAROCHELLE, H.; ADAMS, R. P. Practical bayesian optimization of machine learning algorithms. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*. USA: Curran Associates Inc., 2012. (NIPS' 12), p. 2951–2959. Disponível em: <<http://dl.acm.org/citation.cfm?id=2999325.2999464>>.
- SRIVASTAVA, R. K.; GREFF, K.; SCHMIDHUBER, J. Highway networks. *CoRR*, abs/1505.00387, 2015. Disponível em: <<http://arxiv.org/abs/1505.00387>>.
- STAPLES, M.; NIAZI, M. Experiences using systematic review guidelines. *J. Syst. Softw.*, Elsevier Science Inc., New York, NY, USA, v. 80, n. 9, p. 1425–1437, set. 2007. ISSN 0164-1212. Disponível em: <<http://dx.doi.org/10.1016/j.jss.2006.09.046>>.
- STIRLING, G. R.; ATKINSON, H. J. Biological control of plant parasitic nematodes: Progress, problems and prospects. *The Journal of Agricultural Science*, Cambridge University Press, v. 118, n. 2, p. 263–263, 1992.
- SULC, M.; MATAS, J. Fine-grained recognition of plants from images. *Plant Methods*, BioMed Central, v. 13, n. 1, p. 115, dec 2017. ISSN 1746-4811. Disponível em: <<https://plantmethods.biomedcentral.com/articles/10.1186/s13007-017-0265-4>>.
- SUN, Y.; XUE, B.; ZHANG, M.; YEN, G. G. Evolving deep convolutional neural networks for image classification. *IEEE Transactions on Evolutionary Computation*, v. 24, n. 2, p. 394–407, 2020.
- SUN, Y.; ZHU, L.; WANG, G.; ZHAO, F. Multi-Input Convolutional Neural Network for Flower Grading. *Journal of Electrical and Computer Engineering*, Hindawi, v. 2017, p. 1–8, aug 2017. ISSN 2090-0147. Disponível em: <<https://www.hindawi.com/journals/jece/2017/9240407/>>.

SURYAWATI, E.; SUSTIKA, R.; YUWANA, R. S.; SUBEKTI, A.; PARDEDE, H. F. Deep structured convolutional neural network for tomato diseases detection. In: *2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. [S.l.: s.n.], 2018. p. 385–390. ISSN 2330-4588.

SUTHAHARAN, S. *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*. 1st. ed. [S.l.]: Springer Publishing Company, Incorporated, 2015. ISBN 148997640X, 9781489976406.

SZEGEDY, C.; IOFFE, S.; VANHOUCKE, V.; ALEMI, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. [S.l.]: AAAI Press, 2017. (AAAI'17), p. 4278–4284.

SZEGEDY, C.; IOFFE, S.; VANHOUCKE, V.; ALEMI, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. AAAI Press, 2017. (AAAI'17), p. 4278–4284. Disponível em: <<http://dl.acm.org/citation.cfm?id=3298023.3298188>>.

SZEGEDY, C.; LIU, W.; JIA, Y.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCKE, V.; RABINOVICH, A. Going deeper with convolutions. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. v. 07-12-June-2015, p. 1–9.

SZEGEDY, C.; VANHOUCKE, V.; IOFFE, S.; SHLENS, J.; WOJNA, Z. Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 2818–2826, 2015.

TAIGMAN, Y.; YANG, M.; RANZATO, M.; WOLF, L. Deepface: Closing the gap to human-level performance in face verification. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2014. p. 1701–1708.

TAN, W.; ZHAO, C.; WU, H. Intelligent alerting for fruit-melon lesion image based on momentum deep learning. *Multimedia Tools and Applications*, v. 75, n. 24, p. 16741–16761, Dec 2016. ISSN 1573-7721. Disponível em: <<https://doi.org/10.1007/s11042-015-2940-7>>.

TARG, S.; ALMEIDA, D.; LYMAN, K. Resnet in resnet: Generalizing residual architectures. *CoRR*, abs/1603.08029, 2017. Disponível em: <<http://arxiv.org/abs/1603.08029>>.

THOMAS, A.; HARIKRISHNAN, P.; RAMACHANDRAN, R.; RAMACHANDRAN, S.; MANOJ, R.; PALANISAMY, P.; GOPI, V. P. A novel multiscale and multipath convolutional neural network based age-related macular degeneration detection using oct images. *Computer Methods and Programs in Biomedicine*, v. 209, p. 106294, 2021. ISSN 0169-2607. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0169260721003680>>.

THYAGHARAJAN, K. K.; RAJI, I. K. A review of visual descriptors and classification techniques used in leaf species identification. *Archives of Computational Methods in Engineering*, v. 26, n. 4, p. 933–960, Sep 2019. ISSN 1886-1784. Disponível em: <<https://doi.org/10.1007/s11831-018-9266-3>>.

TIAN, Y.; YANG, G.; WANG, Z.; LI, E.; LIANG, Z. Detection of apple lesions in orchards based on deep learning methods of cyclegan and yolov3-dense. *Journal of Sensors*, v. 2019, p. 1 – 13, 2019. ISSN 1687-725X. Hindawi. Disponível em: <<https://www.hindawi.com/journals/js/2019/7630926/>>.

- TIHOHOD, D. *Guia Prático para a identificação de fitonematóides*. São Paulo: FCAV, FAPESP, 1997.
- TM, P.; PRANATHI, A.; SAIASHRITHA, K.; CHITTARAGI, N. B.; KOOLAGUDI, S. G. Tomato leaf disease detection using convolutional neural networks. In: *2018 Eleventh International Conference on Contemporary Computing (IC3)*. [S.l.: s.n.], 2018. p. 1–5. ISSN 2572-6129.
- TODA, Y.; OKURA, F. How Convolutional Neural Networks Diagnose Plant Disease. *Plant Phenomics*, AAAS, v. 2019, p. 1–14, mar 2019. ISSN 2643-6515. Disponível em: <<https://spj.sciencemag.org/plantphenomics/2019/9237136/>>.
- TONG, S.; CHANG, E. Support vector machine active learning for image retrieval. In: . [S.l.: s.n.], 2001. p. 107–118.
- TONG, T.; LI, G.; LIU, X.; GAO, Q. Image super-resolution using dense skip connections. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2017. p. 4809–4817.
- TOO, E. C.; YUJIAN, L.; NJUKI, S.; YINGCHUN, L. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, v. 161, p. 272 – 279, 2018. ISSN 0168-1699. BigData and DSS in Agriculture. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169917313303>>.
- TUIA, D.; VOLPI, M.; COPA, L.; KANEVSKI, M.; MUNOZ-MARI, J. A survey of active learning algorithms for supervised remote sensing image classification. *IEEE Journal of Selected Topics in Signal Processing*, v. 5, n. 3, p. 606–617, June 2011. ISSN 1941-0484.
- TÜRKOGLU, M.; HANBAY, D. Plant disease and pest detection using deep learning-based features. *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, v. 27, n. 3, p. 1636–1651, may 2019. ISSN 13036203. Disponível em: <<http://online.journals.tubitak.gov.tr/openDoiPdf.htm?mKodu=elk-1809-181>>.
- U.R., A.; PAUL, S. Feature selection and extraction in data mining. In: *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*. [S.l.: s.n.], 2016. p. 1–3.
- USDA, U. S. D. o. A. *Soybeans | USDA Foreign Agricultural Service*. 2021. Disponível em: <<https://www.fas.usda.gov/commodities/soybeans>>.
- VISA, S.; RAMSAY, B.; RALESCU, A.; KNAAP, E. Confusion matrix-based feature selection. In: *Proceedings of The 22nd Midwest Artificial Intelligence and Cognitive Science Conference*. Massachusetts - USA: College of Woosters, 2011. p. 120–127.
- WALLELIGN, S.; POLCEANU, M.; BUCHE, C. Soybean Plant Disease Identification Using Convolutional Neural Network. In: *FLAIRS-31*. Melbourne, United States: [s.n.], 2018. p. 146–151. Disponível em: <<https://hal.archives-ouvertes.fr/hal-01807760>>.
- WAN, L.; ZEILER, M.; ZHANG, S.; LECUN, Y.; FERGUS, R. Regularization of neural networks using dropconnect. In: *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*. [S.l.]: JMLR.org, 2013. (ICML'13), p. III–1058–III–1066.

WANG, D.; VINSON, R.; HOLMES, M.; SEIBEL, G.; BECHAR, A.; NOF, S.; TAO, Y. Early Detection of Tomato Spotted Wilt Virus by Hyperspectral Imaging and Outlier Removal Auxiliary Classifier Generative Adversarial Nets (OR-AC-GAN). *Scientific Reports*, Nature Publishing Group, v. 9, n. 1, p. 4377, dec 2018. ISSN 2045-2322. Disponível em: <<http://www.nature.com/articles/s41598-019-40066-y>>.

WANG, F.; JIANG, M.; QIAN, C.; YANG, S.; LI, C.; ZHANG, H.; WANG, X.; TANG, X. Residual attention network for image classification. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017. p. 6450–6458.

WANG, G.; SUN, Y.; WANG, J. Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning. *Computational Intelligence and Neuroscience*, Hindawi, v. 2017, p. 1–8, jul 2017. ISSN 1687-5265. Disponível em: <<https://www.hindawi.com/journals/cin/2017/2917536/>>.

WANG, J.; CHEN, L.; ZHANG, J.; YUAN, Y.; LI, M.; ZENG, W. Cnn transfer learning for automatic image-based classification of crop disease. In: *IGTA*. [S.l.: s.n.], 2018.

WANG, L.; OUYANG, W.; WANG, X.; LU, H. STCT: sequentially training convolutional networks for visual tracking. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. [s.n.], 2016. p. 1373–1381. Disponível em: <<http://dx.doi.org/10.1109/CVPR.2016.153>>.

WANG, Q.; QI, F.; SUN, M.; QU, J.; XUE, J. Identification of tomato disease types and detection of infected areas based on deep convolutional neural networks and object detection techniques. *Computational Intelligence and Neuroscience*, Hindawi, v. 2019, n. 9142753, p. 15, 2019. ISSN 1687-5265. Disponível em: <<https://doi.org/10.1155/2019/9142753>>.

WANG, X.; BAO, A.; LV, E.; CHENG, Y. Multiscale multipath ensemble convolutional neural network. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, v. 51, n. 9, p. 5918–5928, 2021.

WANG, Z.; ZHANG, S. Segmentation of corn leaf disease based on fully convolution neural network. In: . [S.l.: s.n.], 2018.

WHEELER, H. Concepts and definitions. In: _____. *Plant Pathogenesis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1975. p. 1–4. ISBN 978-3-642-66184-6. Disponível em: <https://doi.org/10.1007/978-3-642-66184-6_1>.

7 the convolution transform. In: WIDDER, D. (Ed.). *An Introduction to Transform Theory*. Elsevier, 1971, (Pure and Applied Mathematics, v. 42). p. 169 – 191. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0079816908608016>>.

WIESNER-HANKS, T.; WU, H.; STEWART, E.; DECHANT, C.; KACZMAR, N.; LIPSON, H.; GORE, M. A.; NELSON, R. J. Millimeter-level plant disease detection from aerial photographs via deep learning and crowdsourced data. *Frontiers in Plant Science*, v. 10, p. 1550, 2019. ISSN 1664-462X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fpls.2019.01550>>.

WIJNHOFEN, R.; WITH, P. D. Fast training of object detection using stochastic gradient descent. In: . [S.l.: s.n.], 2010. p. 424–427.

WOO, S.; PARK, J.; LEE, J.-Y.; KWEON, I. S. Cbam: Convolutional block attention module. In: FERRARI, V.; HEBERT, M.; SMINCHISESCU, C.; WEISS, Y. (Ed.). *Computer Vision – ECCV 2018*. Cham: Springer International Publishing, 2018. p. 3–19. ISBN 978-3-030-01234-2.

WU, H.; WIESNER-HANKS, T.; STEWART, E. L.; DECHANT, C.; KACZMAR, N.; GORE, M. A.; NELSON, R. J.; LIPSON, H. Autonomous detection of plant disease symptoms directly from aerial imagery. *The Plant Phenome Journal*, v. 2, n. 1, p. 190006, 2019. Disponível em: <<https://access.onlinelibrary.wiley.com/doi/abs/10.2135/tppj2019.03.0006>>.

WU, Q.; MERCHANT, F.; CASTLEMAN, K. *Microscope Image Processing*. 1st. ed. USA: Academic Press, Inc., 2008. ISBN 012372578X.

XING, S.; LEE, M.; LEE, K.-K. Citrus pests and diseases recognition model using weakly dense connected convolution network. *Sensors (Basel, Switzerland)*, MDPI, v. 19, n. 14, p. 3195, Jul 2019. ISSN 1424-8220. Disponível em: <<https://pubmed.ncbi.nlm.nih.gov/31331122>>.

YAMAMOTO, K.; TOGAMI, T.; YAMAGUCHI, N. Super-resolution of plant disease images for the acceleration of image-based phenotyping and vigor diagnosis in agriculture. *Sensors*, v. 17, n. 11, 2017. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/17/11/2557>>.

YANG, G.; XU, N.; HONG, Z. Identification of navel orange lesions by nonlinear deep learning algorithm. *Engenharia Agrícola*, scielo, v. 38, p. 783 – 796, 09 2018. ISSN 0100-6916.

YU, H.-J.; SON, C.-H. Apple leaf disease identification through region-of-interest-aware deep convolutional neural network. *ArXiv*, abs/1903.10356, 2019.

YUAN, Y.; FANG, S.; CHEN, L. Crop disease image classification based on transfer learning with dcns. In: LAI, J.-H.; LIU, C.-L.; CHEN, X.; ZHOU, J.; TAN, T.; ZHENG, N.; ZHA, H. (Ed.). *Pattern Recognition and Computer Vision*. Cham: Springer International Publishing, 2018. p. 457–468. ISBN 978-3-030-03335-4.

ZAGORUYKO, S.; KOMODAKIS, N. Wide residual networks. In: WILSON, E. R. H. R. C.; SMITH, W. A. P. (Ed.). *Proceedings of the British Machine Vision Conference (BMVC)*. BMVA Press, 2016. p. 87.1–87.12. ISBN 1-901725-59-6. Disponível em: <<https://dx.doi.org/10.5244/C.30.87>>.

ZAJAČKO, I.; GÁL, T.; SAGOVA, Z.; MATEICHYK, V.; WIĘCEK, D. Application of artificial intelligence principles in mechanical engineering. *MATEC Web of Conferences*, v. 244, p. 01027, 01 2018.

ZEILER, M. D.; FERGUS, R. Visualizing and understanding convolutional networks. In: FLEET, D.; PAJDLA, T.; SCHIELE, B.; TUYTELAARS, T. (Ed.). *Computer Vision ECCV 2014*. Cham: Springer International Publishing, 2014. p. 818–833. ISBN 978-3-319-10590-1.

ZENG, W.; LI, M.; ZHANG, J.; CHEN, L.; FANG, S.; WANG, J. High-order residual convolutional neural network for robust crop disease recognition. In: *Proceedings of the 2Nd International Conference on Computer Science and Application Engineering*. New York, NY, USA: ACM, 2018. (CSAE '18), p. 101:1–101:5. ISBN 978-1-4503-6512-3. Disponível em: <<http://doi.acm.org/10.1145/3207677.3277952>>.

ZHANG, K.; WU, Q.; LIU, A.; MENG, X. Can deep learning identify tomato leaf disease? *Adv. in MM*, v. 2018, p. 6710865:1–6710865:10, 2018.

ZHANG, M.; LIU, S.; YANG, F.; LIU, J. Classification of canker on small datasets using improved deep convolutional generative adversarial networks. *IEEE Access*, v. 7, n. 9, p. 49680–49690, Feb 2019. ISSN 2169-3536.

ZHANG, M.; LIU, S.; YANG, F.; LIU, J. Classification of canker on small datasets using improved deep convolutional generative adversarial networks. *IEEE Access*, v. 7, p. 49680–49690, 2019.

ZHANG, S.; HUANG, W.; ZHANG, C. Three-channel convolutional neural networks for vegetable leaf disease recognition. *Cognitive Systems Research*, v. 53, p. 31 – 41, 2019. ISSN 1389-0417. Advanced Intelligent Computing. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1389041717303236>>.

ZHANG, S.; ZHANG, S.; ZHANG, C.; WANG, X.; SHI, Y. Cucumber leaf disease identification with global pooling dilated convolutional neural network. *Computers and Electronics in Agriculture*, v. 162, p. 422 – 430, 2019. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169918317976>>.

ZHANG, X.; HAN, L.; DONG, Y.; SHI, Y.; HUANG, W.; HAN, L.; GONZÁLEZ-MORENO, P.; MA, H.; YE, H.; SOBEIH, T. A deep learning-based approach for automated yellow rust disease detection from high-resolution hyperspectral uav images. *MDPI Open Access Journal - Remote Sensing in Agriculture and Vegetation*, v. 11, p. 1554, 2019. ISSN 2072-4292.

ZHANG, X.; QIAO, Y.; MENG, F.; FAN, C.; ZHANG, M. Identification of maize leaf diseases using improved deep convolutional neural networks. *IEEE Access*, v. 6, p. 30370–30377, 2018. ISSN 2169-3536.

ZHOU, Y. .; CHELLAPPA, R.; VAID, A.; JENKINS, B. K. Image restoration using a neural network. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. 36, n. 7, p. 1141–1151, July 1988. ISSN 0096-3518.

ZHU, X. J. *Semi-supervised learning literature survey*. [S.l.], 2005.

ZOPH, B.; CUBUK, E. D.; GHIASI, G.; LIN, T.-Y.; SHLENS, J.; LE, Q. V. Learning data augmentation strategies for object detection. In: VEDALDI, A.; BISCHOF, H.; BROX, T.; FRAHM, J.-M. (Ed.). *Computer Vision – ECCV 2020*. Cham: Springer International Publishing, 2020. p. 566–583.

Apêndice A

Tabelas de Sumarização de Resultados

A.1 Capítulo 04 - Revisão Sistemática

Tabela A.1: Características dos Estudos Primários da Seleção Final.

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
S3, S4, S9, S11, S13, S14, S18, S21, S23, S26, S27, S28, S29, S35, S41, S48, S51, S53, S55, S56, S57, S60, S61, S62, S63, S65, S66, S67, S68, S69, S72, S73, S76, S78, S81, S85, S86, S93, S99, S101, S105, S108, S110, S114, S117	PlantVillage	Maçã	Apple scab	Fungos
			Black rot	Fungos
			Cedar apple rust	Fungos
			Leaf Black Rot	Fungos
		Cereja	Powdery mildew	Fungos
			Cercospora leaf spot	Fungos
		Milho	Gray leaf spot	Fungos
			Common rust	Fungos
			Northern leaf blight	Fungos
		Uva	Black rot	Fungos
			Esca (Black measles)	Fungos
			Bacterial leaf spot	Bactéria
			Isariopsis leaf spot	Fungos
		Laranja	Haunglongbing (Citrus greening)	Bactéria
		Pêssego	Bacterial spot	Bactéria
		Pimentão	Bacterial spot	Bactéria
		Batata	Early blight	Fungos
			Late blight	Fungos
		Abóbora	Powdery mildew	Fungos
		Morango	Leaf scorch	Fungos
			Bacterial spot	Bactéria
		Tomate	Early blight	Fungos
			Late blight	Fungos
			Leaf mold	Fungos
			Septoria leaf spot	Fungos
			Spider mites Two spotted	Peste
			Target spot	Fungos
		Manga	Tomato mosaic virus	Vírus
			Tomato yellow leaf curl virus	Vírus
			Anthracoze	Fungos
Banana Sigatoka	Fungos			
Banana	Banana Cordana musae	Fungos		
	Deightoniella Leaf Spot	Fungos		
	Banana Diamond Leaf Spot	Fungos		
S1	Custom	Pepino	Downy mildew	Fungos
			Anthracoze	Fungos
			Gray mold	Fungos
			Angular leaf spot	Bactéria
			Black spot	Fungos
S2	Custom Spike	Trigo	Powdery mildew	Fungos
			Nutritional Deficiency	Abiótico
S3	Custom	Tomate	Bacterial spot	Bactéria
			Early blight	Fungos

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno		
S5, S19, S42	Crop Disease AI Challenger 2018		Late blight	Fungos		
			Leaf mold	Fungos		
			Spider mites Two spotted	Peste		
			Target spot	Fungos		
			Tomato yellow leaf curl virus	Virus		
		Maçã	Apple scab	Fungos		
			Apple Frogeye Spot	Fungos		
			Cedar Apple Rust	Fungos		
		Cereja	Powdery mildew	Fungos		
			Cercospora zeaemaydis Tehon and Daniels	Fungos		
		Milho	Puccinia polysora	Fungos		
			Corn Curvularia leaf spot	Fungos		
			Dwarf mosaic	Virus		
			Black Rot	Fungos		
		Uva	Black Measles	Fungos		
			Leaf Blight	Fungos		
			Greening June	Bactéria		
		Pêssego	Bacterial spot	Bactéria		
		Pimenta	Pepper Scab	Bactéria		
			Early blight	Fungos		
			Batata	Late blight	Fungos	
				Morango	Strawberry Scorch	Fungos
			Tomate	Powdery mildew	Fungos	
				Bacterial spot	Bactéria	
				Early blight	Fungos	
			Tomate	Late blight	Fungos	
				Leaf mold	Fungos	
				Target spot	Fungos	
				Septoria leaf spot	Fungos	
			Tomate	Spider mites Two spotted	Peste	
				YLCV Virus	Virus	
				Tomato Tomv	Virus	
				Septoria Tritici	Fungos	
		S6	Custom	Trigo de Inverno	Puccinia Striformis	Fungos
					Puccinia Recondita	Fungos
					Septoria Nodorum	Fungos
					Drechslera tritici-repentis	Fungos
					Oculimacula Yallundae	Fungos
					Gibberella zeae	Fungos
					Blumeria graminis	Fungos
				Milho	Helminthosporium turcicum	Fungos
					Semente de Colza	Phoma lingam
Cevada de Inverno	Pyrenophora teres			Fungos		
	Ramularia collo-cygni			Fungos		
	Rhynchosporium secalis			Fungos		
	Puccinia hordei			Fungos		
Arroz Comum	Thanatephorus cucumeris			Fungos		
	Pyricularia oryzae			Fungos		
S7	Custom	Soja	charcoal rot	Fungos		
S8	Custom	Milho	Northern leaf blight	Fungos		
S9	Custom	Pepino	Scab Angular	Fungos		
			Powdery mildew	Fungos		
			Downy mildew	Fungos		
			Anthracnose and Scab	Fungos		
S10	Custom	Arroz Comum	Scab	Fungos		
			Magnaporthe grisea	Fungos		
			Powdery mildew	Fungos		
S12	Custom	Trigo de Inverno Trigo	Cochliobolus Heterostrophus	Fungos		
			Bacterial Leaf Streak	Bactéria		
			Leaf Blight	Fungos		
			Stripe Rust	Fungos		
			Leaf Rust	Fungos		
S15	Custom	Maçã	Anthracnose	Fungos		
S16,S59	PDDB Embrapa	Abacaxi	Pineapple Fruit Borer	Peste		
			Fusariose	Fungos		
			Black Rot	Fungos		
Algodão			Fusarium Wilt	Fungos		
			Myrothecium Leaf Spot	Fungos		
			Soreshin	Fungos		

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno	
			Areolate Mildew	Fungos	
		Arroz	Rice Blast	Fungos	
			Leaf Scald	Fungos	
			Leaf Miner	Peste	
		Café	Cercospora leaf spot	Fungos	
			Rust	Fungos	
			Bacterial Blight	Bacteria	
			Blister Spot	Fungos	
			Brown Leaf Spot	Fungos	
			Pellicularia Koleroga	Fungos	
			Damping Off	Fungos	
		Cajueiro	Algae	Algae	
			Anthracnose	Fungos	
			Worm	Abiotic	
			Angular leaf spot	Bacteria	
			Black Mould	Fungos	
			Powdery mildew	Fungos	
			Gummosis	Bacteria	
			Seedling Blight	Fungos	
			Phoma costaricensis	Fungos	
			Cana de Açúcar	Rust	Fungos
		Ring Spot		Fungos	
		White Fly		Peste	
		Red Stripe		Bacteria	
		Citrus	Algae	Algae	
			Alternaria Brown Spot	Fungos	
			Citrus Canker	Bacteria	
			Citrus Variedgated Chlorosis	Bacteria	
			Nutritional Deficiency	Abiotic	
			Felt	Fungos	
			Sooty Mold	Fungos	
			Leprosis	Virus	
			Halo Blight	Bacteria	
			Citrus Greasy Spot	Fungos	
		Citrus	Black Spot	Fungos	
			Mosaic	Virus	
			Penicillium	Fungos	
			Postbloom Fruit Drop of Citrus	Fungos	
			White-Thread Blight	Fungos	
			Scab	Fungos	
			Coqueiro	Steneotarsonemus furcatus	Peste
				Aceria guerreronis	Peste
				Caterpillar damage	Peste
				Coconut Scale	Peste
		Phytotoxicity		Abiotic	
		Sooty Mold		Fungos	
		Dreschlera incurvata		Fungos	
		Sphaerodothis acrocomiae		Fungos	
		Phyllachora torrendiella		Fungos	
		Cylindrocladium leaf spot		Fungos	
		White Fly		Peste	
		Coconut Rot		Fungos	
		Premature Fruit Drop		Abiotic	
		Coconut Leaf Blight		Fungos	
		Stem Bleeding	Fungos		
		Repolho	Alternaria Leaf Spot	Fungos	
			Powdery mildew	Fungos	
		Cupuaçu	Lasiodiplodia Rot	Fungos	
			Witches Broom	Fungos	
		Feijão Seco	Anthracnose	Fungos	
			Cercospora leaf spot	Fungos	
			Cowpea Mild Mottle Virus	Virus	
			Common Bacterial Blight	Bacteria	
			Rust	Fungos	
			Phytotoxicity	Abiotic	
			Hedylepta Indicata	Peste	
			Target Leaf Spot	Fungos	

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
			Bacterial Spot	Bacteria
			Web Blight	Fungos
			Powdery mildew	Fungos
			Charcoal Rot	Fungos
			Bean Golden Mosaic	Virus
		Papaya	Anthracnose	Fungos
			Mosaic Virus	Virus
			Smallpox	Virus
			Green Mite	Peste
			Anthracnose	Fungos
			Bacterial Blight	Bacteria
			White Leaf Spot	Fungos
		Mandioca	Brown Leaf Spot	Fungos
			Cassava Common Mosaic	Virus
			Leaf Galls	Peste
			Powdery mildew	Fungos
			Root Rot	Fungos
			Blight Leaf Spot	Fungos
			Cercospora leaf spot	Fungos
			Scab	Fungos
			Fusariosis	Fungos
			Dione junco junco	Peste
		Maracujá	Bacterial Spot	Bacteria
			Mosaic Virus	Virus
			Base Rot	Fungos
			Senescence	Abiotic
			Septoria leaf spot	Fungos
			Woodiness virus	Virus
			Bacterial Fruit Blotch	Bacteria
		Meião	Mosaic	Virus
			Nematode	Nematode
			Powdery mildew	Fungos
			Anthracnose	Fungos
			Bushy Stunt	Bacteria
			Tropical Rust	Fungos
			Southern Corn Rust	Fungos
		Milho	Scab	Fungos
			Southern Corn Leaf Blight	Fungos
			Phaeosphaeria Leaf Spot	Fungos
			Diplodia Leaf Streak	Fungos
			Physoderma Brown Spot	Fungos
			Northern leaf blight	Fungos
			Red Ring	Nematode
			Anthracnose	Fungos
		Dendezeiros	Fusarium Wilt	Fungos
			Necrotic Spotting	Fungos
			Curvularia Leaf Spot	Fungos
			Black Sooty Mold	Fungos
			Thielaviopsis rot	Fungos
			Anthracnose	Fungos
		Pimenta do Reino	Yellow Wilt	Fungos
			Phytophthora Foot Rot	Fungos
			White-Thread Blight	Fungos
			Bacterial Blight	Bacteria
			Cercospora Blight	Fungos
			Charcoal Rot	Fungos
			Rust	Fungos
			Rust and Target Spot	Fungos
			Copper Phytotoxicity	Abiotic
			Soybean Mosaic Virus	Virus
		Soja	Southern Blight	Fungos
			Myrothecium Leaf Spot	Fungos
			Rhizoctonia aerial blight	Fungos
			Downy mildew	Fungos
			Southern Blight	Fungos
			Powdery mildew	Fungos
			Powdery mildew and Rust	Fungos
			Phytophthora Rot	Fungos
			Brown Leaf Spot	Fungos
			Wheat Blast	Fungos

Trigo

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno	
		Uva	Rust	Fungos	
			Tan Spot	Fungos	
			Powdery mildew	Fungos	
			Anthracnose	Fungos	
			Bacterial Canker	Bacteria	
			Rust	Fungos	
			Isariopsis Leaf Spot	Fungos	
			Downy mildew	Fungos	
			Powdery mildew	Fungos	
			Grapevine fanleaf virus	Virus	
S18	Custom	Manga	Anthracnose	Fungos	
S20	Custom	Banana	Xanthomonas wilt	Bacteria	
			Bunchy top disease	Virus	
			Black sigatoka	Fungos	
			Yellow sigatoka	Fungos	
			Fusarium wilt	Fungos	
			Corm weevil	Peste	
			Dried/old leaves	Abiotic	
S22	Custom	Trigo	Yellow Rust	Fungos	
S24	Custom	Uva	Esca (Black measles)	Fungos	
S25	Custom	Maçã	Apple Scab	Fungos	
S28	Custom	Uva	Black Rot	Fungos	
			Chlorosis	Abiotic	
			Esca (Black measles)	Fungos	
S30	Custom Turkey	Pêssego	Coryneum beijerinckii	Fungos	
			Peach sphaerolectanium prunastri	Peste	
			Xanthomonas arboricola	Bacteria	
		Damasco	Erwinia amylovora	Bacteria	
			Monilinia laxa	Fungos	
			Apricot monilia laxa	Fungos	
		Noz	Walnut leaf Gall Mite	Peste	
		Cereja	Cherry myzus cerasi	Peste	
S31	BU-Guava-Leaf	Goiaba	Algal leaf spot	Alga	
			Rust	Fungos	
			White Fly	Peste	
S32	ALDD Apple	Maçã	Alternaria Leaf Spot	Fungos	
			Brown Leaf Spot	Fungos	
			Mosaic	Virus	
			Grey Spot	Fungos	
S33	Custom	Pepino	Rust	Fungos	
S34	Custom	Citrus	Powdery mildew	Fungos	
			Pezothrips	Peste	
			Parlatoria pergandii	Peste	
			Aonidiella aurantii	Peste	
S35	PlantDisease	Maçã	Scirtothrips	Peste	
			Venturia inaequalis	Fungos	
			Gymnosporangium juniperi-virginianae	Fungos	
		Pimentão	Podosphaera leucotricha	Fungos	
			Botryosphaeria obtuse	Fungos	
			Alternaria pomi	Fungos	
			Xanthomonas euvesicatoria	Bacteria	
			Xanthomonas camPestris	Bacteria	
		Cereja	Blumeriella jaapii	Fungos	
			Podosphaera spp.	Fungos	
		Uva	Uncinula necator	Fungos	
			Plasmopara viticola	Fungos	
Botrytis cinereal	Fungos				
Botryosphaeria obtuse	Fungos				
Pseudocercospor a vitis	Fungos				
Guignardia bidwellii	Fungos				
Phaeomoniella spp.	Fungos				
Cebola	Peronospora destructor	Fungos			
Pêssego	Clasterosporium carpophilum	Fungos			
Batata	Alternaria solani	Fungos			
	Polystigma rubrum	Fungos			
Ameixa	Plum Plox	Virus			
	Tranzschelia pruni-spinosae	Fungos			
	Stigmia carpofilia	Fungos			

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
		Morango	Mycosphaerella fragariae	Fungos
		Beterraba	Cercospora beticola	Fungos
		Tomate	Phytophthora infestans	Fungos
			Septoria lycopersici	Fungos
		Trigo	Erysiphe graminis	Fungos
			Puccinia spp.	Fungos
			Septoria spp.	Fungos
S36	Custom	Milheto	Mildew	Fungos
S37	Custom	Batata	Potato PVY	Virus
			Tea bud blight	Fungos
S38	Custom	Folhas de Chá	Tea leaf blight	Fungos
			Tea red Scab	Fungos
			Marssonia Blotch	Fungos
S39	Custom	Maçã	Alternaria Leaf Spot	Fungos
			Brown streak	Virus
			Mosaic	Virus
			Green Mite	Peste
S40	IITA Cassava	Mandioca	Red Mite	Peste
			Brown Leaf Spot	Fungos
			Nutritional Deficiency	Abiotic
			Northern leaf blight	Fungos
S43	Custom	Milho	Puccinia Sorghi	Fungos
			Gray Leaf Spot	Fungos
			Malformed fruit	Abiotic
			Blotchy ripening	Abiotic
			Puffy fruit	Abiotic
			Dehiscent fruit	Abiotic
S44	Custom	Tomate	Blossom-end rot	Abiotic
			Sunscald	Abiotic
			Tomato Virus	Virus
			Gray mold	Fungos
			Tomato ulcer	Fungos
			Anthracoze	Fungos
S45	Custom	Beterraba	Cercospora Leaf Spot	Fungos
			White Leaf Spot	Fungos
			Birds eye spot	Fungos
			Red leaf spot	Alga
S46	Custom	Folhas de Chá	Gray blight	Fungos
			Anthracoze	Fungos
			Brown Blight	Bacteria
			Algal leaf spot	Alga
S47	Custom	Milho	Northern leaf blight	Fungos
S49	Custom	Citrus	Citrus Canker	Bacteria
			Ceratitidis capitata	Peste
S50	Custom	Citrus	Diaphorina citri Kuwayama	Peste
			Anoplophora chinensis	Peste
			Halyomorpha halys	Peste
			Nezara viridula	Pest
			Othreis fullonica	Peste
			Papilio demodocus	Peste
			Metcalfa pruinosa (Say)	Peste
			Planococcus citri	Peste
			Toxoptera citricida	Peste
			Hemiptera: Coccidae	Peste
			Thaumatotibia leucotreta	Peste
			Root Weevil	Peste
			Scudderia furcata	Peste
			Cicadoidea	Peste
			Cornu aspersum	Peste
			Homalodisca vitripennis	Peste
			Colletotrichum gloeosporioides	Fungos
			Xanthomonas axonopodis	Bacteria
			Diaporthe citri	Fungos
			Elsinoë fawcettii	Fungos
			Liriomyza brassicae	Peste
			Capnodium spp	Fungos
			Locusta migratoria	Peste
			Parasa lepida	Peste
			Gypsy moth larva	Peste
			Empoasca flavescens	Peste
S52	Custom	Folhas de Chá		Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
			Spodoptera exigua	Peste
			Chrysochus chinensis	Peste
			Laspeyresia pomonella larva	Peste
			Spodoptera exigua larva	Peste
			Atractomorpha sinensis	Peste
			Laspeyresia pomonella	Peste
		Phyllostachys edulis (Carr.) Houz.	N/A	N/A
		Aesculus chinensis	N/A	N/A
		Berberis anthei Ahrendt	N/A	N/A
		Cercis chinensis	N/A	N/A
		Indigofera tinctoria L.	N/A	N/A
		Acer Palmatum	N/A	N/A
		Phoebe nanmu	N/A	N/A
		Kalopanax septemlobus	N/A	N/A
		Cinnamomum japonicum Sieb.	N/A	N/A
		Koelreuteria paniculata Laxm.	N/A	N/A
		Ilex macrocarpa Oliv.	N/A	N/A
		Pittosporum tobira (Thunb.) Ait. f.	N/A	N/A
		Chimonanthus praecox L.	N/A	N/A
		Cinnamomum camphora	N/A	N/A
		Viburnum awabuki K.Koch	N/A	N/A
		Osmanthus fragrans Lour.	N/A	N/A
S53,S109,S120	Flavia Dataset	Cedrus deodara (Roxb.) G. Don	N/A	N/A
		Ginkgo biloba L.	N/A	N/A
		Lagerstroemia indica (L.) Pers.	N/A	N/A
		Nerium oleander L.	N/A	N/A
		Podocarpus macrophyllus	N/A	N/A
		Prunus serrulata Lindl.	N/A	N/A
		Ligustrum lucidum Ait. f.	N/A	N/A
		Tonna sinensis M. Roem.	N/A	N/A
		Prunus persica (L.) Batsch	N/A	N/A
		Manglietia fordiana Oliv.	N/A	N/A
		Acer buergerianum Miq.	N/A	N/A
		Mahonia bealei (Fortune) Carr.	N/A	N/A
		Magnolia grandiflora L.	N/A	N/A
		Populus canadensis Moench	N/A	N/A
		Liriodendron chinense	N/A	N/A
		Citrus reticulata Blanco	N/A	N/A
		Quercus suber	N/A	N/A
		Salix atrocinerea	N/A	N/A
		Populus nigra	N/A	N/A
		Alnus sp	N/A	N/A
		Quercus robur	N/A	N/A
		Crataegus monogyna	N/A	N/A
		Ilex aquifolium	N/A	N/A
		Nerium oleander L.	N/A	N/A
		Betula pubescens	N/A	N/A
		Tilia tomentosa	N/A	N/A
		Acer palmaturu	N/A	N/A
		Celtis sp	N/A	N/A
		Corylus avellana	N/A	N/A
		Castanea sativa	N/A	N/A
		Populus alba	N/A	N/A
		Acer negundo	N/A	N/A
		Taxus bacatta	N/A	N/A
		Polypodium vulgare	N/A	N/A
S53,S109,S120	UCI Leaf Dataset	Pinus sp	N/A	N/A
		Fraxinus sp	N/A	N/A
		Primula vulgaris	N/A	N/A
		Erodium sp	N/A	N/A
		Bougainvillea sp	N/A	N/A
		Arisarum vulgare	N/A	N/A
		Euonymus japonicus	N/A	N/A
		Ilex perado ssp azorica	N/A	N/A
		Magnolia soulangeana	N/A	N/A
		Buxus sempervirens	N/A	N/A
		Urtica dioica	N/A	N/A
		Podocarpus sp	N/A	N/A
		Acca sellowiana	N/A	N/A
		Hydrangea sp	N/A	N/A
		Pseudosasa japonica	N/A	N/A

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
	Swedish Leaf Dataset	Magnolia grandiflora L.	N/A	N/A
		Geranium sp	N/A	N/A
		Aesculus californica	N/A	N/A
		Chelidonium majus	N/A	N/A
		Ulmus carpinifolia	N/A	N/A
		Acer	N/A	N/A
		Salix aurita	N/A	N/A
		Quercus	N/A	N/A
		Alnus incana	N/A	N/A
		Betula pubescens	N/A	N/A
		Salix alba 'Sericea'	N/A	N/A
		Populus tremula	N/A	N/A
		Ulmus glabra	N/A	N/A
		Sorbus aucuparia	N/A	N/A
		Salix sinerea	N/A	N/A
		Populus	N/A	N/A
		Tilia	N/A	N/A
		Sorbus intermedia	N/A	N/A
		Fagus sylvatica	N/A	N/A
		S54	VOAI UAV Dataset	Adenanthera pavonina Linn
Blechnum orientale Linn	N/A			N/A
Psychotria serpens Linn	N/A			N/A
Carmona microphylla (Lam)	N/A			N/A
Dicranopteris dichotoma (Thunb)	N/A			N/A
Eucalyptus citriodora Hook	N/A			N/A
Photinia serrulata Lindl	N/A			N/A
Pinus massoniana Lamb	N/A			N/A
Withered dicranopteris dichotoma	N/A			N/A
Withered Mosla chinensis Maxim	N/A			N/A
Miscanthus sinensis Anderss	N/A			N/A
Acacia confusa Merr	N/A			N/A
S58	Custom			Tomate
		Phoma Rot	Fungos	
		Target Leaf Spot	Fungos	
S64	Custom PDDS	Maçã	Scab	Fungos
			Black Rot	Fungos
			Ceder Rust	Fungos
		Milho	Cercospora leaf spot	Fungos
			Common rust	Fungos
			Northern leaf blight	Fungos
		Tomate	Early Blight	Fungos
			Leaf Mold	Fungos
			Septoria leaf spot	Fungos
Target Leaf Spot	Fungos			
S65	Saiana Research Center	Pepino	N/A	N/A
S69	Custom	Manga	Bacterial Canker	Bacteria
			Mildew Mango	Fungos
			Phoma Blight	Fungos
		Groundnut	Red Rust	Alga
			Leaf Spot	Fungos
S70	Custom	Pimenta	Bud Necrosis	Virus
			Tomato Spotted Wilt Virus	Virus
S71	Custom	Milho	Curvularia Leaf Spot	Fungos
			Dwarf Mosaic	Virus
			Gray Leaf Spot	Fungos
			Northern leaf blight	Fungos
			Brown Spot	Fungos
			Round Spot	Bacteria
			Rust	Fungos
			Southern Blight	Fungos
S74	Custom	Maçã	Apple Marssonina blotch (AMB)	Fungos
			Young Leaf	Fungos
			Nutritional Deficiency	Abiotic
S75	Custom	Tomate	Leaf Mold	Fungos
			Gray Mold	Fungos
			Canker	Bacteria
			Plague	N/A
			Miner	Peste
			Powdery mildew	Fungos
			White Fly	Peste

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
S77	Custom	Trigo	Yellow leaf curl	Virus
			Nutritional excess	Abiotic
			Rust	Fungos
			Septoria leaf spot	Fungos
			Tan Spot	Fungos
S79	Custom	Citrus	Yellowing	N/A
S80	Custom	Milho	N/A	N/A
S82	Custom	Rabanete	Fusarium wilt	Fungos
			Anomala corpulenta	Peste
			Motsehulsiy	Peste
			Papilio xuthus Linnaeus	Peste
			Gryllotalpa	Peste
			Erthesina fullo	Peste
			Apriona germari	Peste
			Lebeda nobilis Walker	Peste
			Papilio bianor	Peste
			Ctenuchidae	Peste
			Gongylolus adyposus	Peste
			Platylomia piel Kato	Peste
			Stichophthalma louisa	Peste
			Anoplophora chinensis	Peste
			Eurydema dominulus	Peste
			Pieris rapae crucivora	Peste
			Nezara viridula	Peste
			Linnaeus	Peste
			Parnara guttata	Peste
			Bremer et Grey	Peste
			Dichocrocis punctiferalis	Peste
			Guenée	Peste
			Ostrinia nubilalis	Peste
			Gryllulus	Peste
			Grasshopper	Peste
			Pergesa elpenorlewisi	Peste
			Henosepilachna	Peste
Vigintioctopunctata	Peste			
Prosopocoilas gracilis	Peste			
S83	Pest Image Dataset	N/A	Coccinella septempunctata	Peste
			Paratenosera seu Hierodula	Peste
			Dragonfly	Peste
			Tettigoniidae	Peste
			Bacterial Blight	Bacteria
			Septoria Brown Spot	Fungos
			Frogeye Leaf Spot	Fungos
			Herbicide Injury	Abiotic
			Iron Deficiency Chlorosis	Abiotic
			Potassium Deficiency	Abiotic
			Bacterial Pustule	Bacteria
			Sudden Death Syndrome	Fungos
S84	Custom	Soja	Apple Rot	Fungos
			Apple Scab	Fungos
			Sheath blight	Fungos
			Rice Blast	Fungos
			Rice flax spot	Fungos
S85	CASC-IFW	Maçã	Powdery mildew	Fungos
			Downy mildew	Fungos
			Target Leaf Spot	Fungos
			Leptocorisa acuta	Peste
			Locusta migratoria	Peste
S86	AES-CD9214 Dataset	Arroz	Nephotettix virescens	Peste
			Nilaparvata lugens	Peste
			Pomacea canaliculata egg	Peste
			Pomacea canaliculata adult	Peste
			Pycularia oryzae leaf	Fungos
			Pycularia oryzae neck panicle	Fungos
		Pepino	Sogatella furcifera	Peste
			Stemborer adult	Peste
			Stemborer larva	Peste
			Tungro leaf	Virus
			Xanthomonas oryzae	Bacteria
S87	Custom	Arroz	Fusarium oxysporum f. sp.	Fungos
			Anthracnose	Fungos
S88	Custom	Banana		
S89	Custom	Pepino		Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
			Downy mildew	Fungos
			Powdery mildew	Fungos
			Target Leaf Spot	Fungos
S90	Custom	Arroz	Golden Apple Snail	Peste
			Tungro leaf	Virus
			Black Bug	Peste
S91	Custom	Arroz	Bacterial Leaf Blight	Bactéria
			Brown Spot	Fungos
			Brown Plant Hopper	Peste
			False smut	Fungos
			Stemborer adult	Peste
			Hispa	Peste
			Neck Blast	Fungos
S92	Custom	Batata	Sheath Blight	Fungos
			Sheath Rot	Fungos
			Black Scurf	Fungos
			Silver Scurf	Fungos
S94	TRES Taiwan Tea Research	Folhas de Chá	Common Scab	Bactéria
			Black Dot	Fungos
			Brown Blight	Bactéria
S95	Custom UAV	Arroz	Blister Blight	Fungos
S96	Custom	Milho	Algal leaf spot	Alga
			N/A	N/A
			Corn Leaf spot	Fungos
			Small Spot	Fungos
S97	Custom	Uva	Leaf Spot	Fungos
			Brown Spot	Fungos
			Streak and round spot	Virus
S98	Custom	Pepino	Esca (Black measles)	Fungos
			Melon Yellow Spot Virus	Virus
			Zucchini Yellow Mosaic Virus	Virus
			Cucurbit Chlorotic Yellows Virus	Virus
			Cucumber Mosaic Virus	Virus
			Papaya ring Spot Virus	Virus
			Watermelon Mosaic Virus	Virus
S100	Custom	Arroz	Green Mottle mosaic Virus	Virus
			Downy mildew	Fungos
			bacterial wilt	Bactéria
			seeding blight	Fungos
			bacterial sheath rot	Bactéria
			Bacterial Leaf Blight	Bactéria
			Sheath Rot	Fungos
			Sheath Blight	Fungos
			bakanae disease	Fungos
			Brown Spot	Fungos
S102	Custom	Tomate	False smut	Fungos
			Rice Blast	Fungos
			Nutritional excess	Abiotic
			Whitefly	Peste
			Powdery mildew	Fungos
			Leaf Miner	Peste
			Canker	Bactéria
S103	Custom	Milho	Gray mold	Fungos
			Leaf mold	Fungos
S104	Custom	Morango	Northern leaf blight (NLB)	Fungos
			Gray mold rot	Fungos
			Powdery mildew	Fungos
S106	Custom	Mandioca	Fusarium wilt	Fungos
			Anthracnos	Fungos
			Cassava mosaic disease	Virus
			Cassava brown streak disease	Virus
S107	Custom	Maçã	Green mite damage	Peste
			Red mite damage	Peste
			Brown leaf spot	Fungos
S109	Middle European Wood	Diversos	Abiotic disturb	Abiotic
			Austrian Federal Forest (AFF)	Diversos
			Foliage Leaf Dataset	Diversos
			N/A	N/A
			N/A	N/A
			N/A	N/A

Continua na próxima página...

Tabela A.1 – continuação da página anterior

ID Estudo	Conjunto de Dados	Tipo de Cultura	Nome da Doença	Patógeno
	Leafsnap Dataset	Diversos	N/A	N/A
S111	BifuGloxinia	Flores	N/A	N/A
S112	Custom	Rabanete	Fusarium wilt of radish	Fungos
			Mosaic	Virus
S113	Custom	Maçã	Brown spot	Fungos
			Rust	Fungos
			Alternaria leaf spot	Fungos
			Porosity	NA
		Pera	Erwinia amylovora	Bacteria
			Venturia	Fungos
		Cereja	Porosity	NA
			Porosity	NA
		Pêssego	Powdery mildew	Fungos
			Taphrina deformans	Fungos
S115	Custom	Maçã	Erwinia amylovora	Bacteria
			Venturia	Fungos
			Powdery mildew	Fungos
			Rust	Fungos
		Pair	Gymnosporangium sabiniae	Fungos
			Gray leaf spot	Fungos
			Wilt	Fungos
		Uva	Mites	Peste
			Powdery mildew	Fungos
			Downy mildew	Fungos
S116	Custom	Maçã	Scab skin	Fungos
			Black rot	Fungos
			Scar skin	Virus
			Ring spot	Fungos
S118	Custom	Maçã	Potassium deficiency	Abiotic
			Magnesium deficiency	NA
			Scab damage	Fungos
			Glomerella stain	Fungos
			Herbicide damage	NA
S119	Custom	Pepino	Melon yellow spot virus: MYSV	Virus
			Zucchini yellow mosaic virus: ZYMV	Virus
			Cucurbit chlorotic yellows virus: CCYV	Virus
			Cucumber mosaic virus: CMV	Virus
			Papaya ring spot virus: PRSV	Virus
			Watermelon mosaic virus: WMV	Virus
			green mottle mosaic virus	Virus
S121	Custom	Pepino	Melon yellow spot virus: MYSV	Virus
			Zucchini yellow mosaic virus: ZYMV	Virus

Tabela A.2: Características dos Estudos Primários da Seleção Final.

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S1	(ZHANG <i>et al.</i> , 2019d)	2019	5	GPDCNN	Multiclass Classifier	94,65%	Customized 700 Images / 7 Class Real Environment	Pepino	RGB	Top View	TensorFlow	New Architecture
S2	(HASAN <i>et al.</i> , 2019)	2019	4,5	R-CNN & Faster R-CNN	Object Detection	93,4%	Customized 335 Images / 3 Class Real Environment	Trigo	RGB	Top View	TensorFlow	Transfer Learning
S3	(SHARMA <i>et al.</i> , 2019)	2019	4,5	S-CNN	Multiclass Segmentation	98,6%	PlantVillage Subset 15.817 Images / 8 Class Controlled Environment	Tomate	RGB	Top View	TensorFlow	Image Segmentation
S4	(LIANG <i>et al.</i> , 2019a)	2019	4,5	ResNet	Multiclass Classifier	98%	Customized 17.000 Images / 10 Class Real Environment	Diversos	RGB	Top View	PyTorch	Fine Tune & Hyperparameters
S5	(HUANG <i>et al.</i> , 2019)	2019	4,5	UNet + InceptionV3	Multiclass Segmentation	87,45%	PlantVillage 32.933 Images / 45 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Image Segmentation
S6	(PICON <i>et al.</i> , 2019)	2019	4	RESNET-MC1	Multiclass Classifier	98% BAC	Crop Disease AI Challenger 2018 40.000 Images / 19 Class Controlled Environment	Diversos	RGB	Front View	TensorFlow/Keras	New Architecture
S7	(NAGASUBRAMANIAN <i>et al.</i> , 2019)	2019	4	3D-CNN	Binary Classifier	95,73%	Customized 121.955 Images / 21 Class Controlled Environment e Real	Soja	Hyperspectral	Front View	TensorFlow/Keras	New Architecture
S8	(WIESNER-HANKS <i>et al.</i> , 2019)	2019	4	ResNet - Crowdsourced	Binary Segmentation	99,79%	Customized 15.240 Images / 2 Class Real Environment	Milho	RGB	Aerial View	TensorFlow	Image Segmentation
S9	(ZHANG <i>et al.</i> , 2019c)	2019	4	Three Channel CNN AlexNet	Multiclass Classifier	91,15% 94,27%	PlantVillage Subset 15.817 Images / 8 Class Controlled Environment	Tomate	RGB	Top View	MatLab	Layers Custom
S10	(LIANG <i>et al.</i> , 2019b)	2019	4	LeNet	Binary Classifier	95,83%	Customized 500 Images / 5 Class Real Environment	Pepino				
S11	(HARI <i>et al.</i> , 2019)	2019	4	PDDNN	Binary Classifier	95,83%	Rice Blast Disease 5.808 Images / 2 Class Controlled Environment	Arroz	RGB	Patches Front View	PyTorch	Layers Custom
S12	(LIN <i>et al.</i> , 2019)	2019	4	M-bCNN-CKM AlexNet VGG	Multiclass Classifier	96,5% 83,1% 88,5%	PlantVillage 14.810 Images / 10 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	New Architecture
S13	(BRAHIMI <i>et al.</i> , 2019)	2019	4	Teacher/Student Architecture	Multiclass Classifier	90,7% AOPC	Customized 16.652 Images / 8 Class Real Environment	Trigo	RGB	Front View	TensorFlow	New Architecture
S13	(BRAHIMI <i>et al.</i> , 2019)	2019	4	Teacher/Student Architecture	Multiclass Classifier	90,7% AOPC	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow/Keras	New Architecture

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S14	(KHANDELWAL; RAMAN, 2019)	2019	4	ResNet	Multiclass Classifier	99,37%	PlantVillage 86.198 Images / 57 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Transfer Learning
S15	(TIAN <i>et al.</i> , 2019)	2019	4	YOLOV3 - DenseNet	Object Detection	95,75%	Customized 640 Images / 2 Class Real Environment	Maçã	RGB	Front View	TensorFlow	New Architecture
S16	(BARBEDO, 2019)	2019	4	GoogLeNet	Multiclass Classifier	94%	PDDB Embrapa 46.409 Images / 79 Class Real Environment	Diversos	RGB	Top View	MatLab	Transfer Learning
S17	(ABDALLA <i>et al.</i> , 2019)	2019	4	VGG16 Encoder	Binary Segmentation	96%	Customized 400 Images / 2 Class Controlled Environment	Oilseeds	RGB	Top View	MatLab	Fine Tune & Image Segmentation
S18	(SINGH <i>et al.</i> , 2019)	2019	4	MCNN - AlexNet	Multiclass Classifier	97,13%	PlantVillage Subset (1130) Custom Mango Leaf (1070) 2.200 Images / 2 Class Controlled Environment e Real	Manga	RGB	Top View	TensorFlow/Keras	Layers Custom
S19	(LONG; LIU, 2019)	2019	4	ResNet-B VGG16 InceptionV3 MobileNet	Multiclass Classifier	99,7% 94,7% 99,5% 99,6%	AI Challenger 40.772 Images / 23 Class Controlled Environment	Diversos	RGB	Top View	PyTorch	Layers Custom
S20	(SELVARAJ <i>et al.</i> , 2019)	2019	4	Faster R-CNN InceptionV2 Faster R-CNN ResNet50 SSD MobileNetV1	Object Detection	99,0% 99,0% 98,2%	Customized 18.000 Images / 2 Class Real Environment	Banana	RGB	Front View	TensorFlow/Keras	Layers Custom
S21	(KC <i>et al.</i> , 2019)	2019	3,5	MobileNet Reduce MobileNet	Multiclass Classifier	97,65% 98,34%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow/Keras	Layers Custom
S22	(ZHANG <i>et al.</i> , 2019e)	2019	3,5	DCNN Inception-ResNet Block	Binary Classifier	85,0%	Customized 15.000 Images / 2 Class Controlled Environment	Trigo	Hyperspectral	Aerial View	PyTorch	New Architecture
S23	(ALRUWAILI <i>et al.</i> , 2019)	2019	3,5	DCNN based AlexNet	Multiclass Classifier	99,11%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Layers Custom
S24	(RANCON <i>et al.</i> , 2018)	2019	3,5	MobileNet	Object Detection	91,0%	Customized 6.000 Images / 2 Class Real Environment	Uva	RGB	Front View	TensorFlow/Keras	Transfer Learning
S25	(DOUARRE <i>et al.</i> , 2019)	2019	3,5	SegNet e GANs	Binary Segmentation	≈ 64,3%	Customized 50 Images / 2 Class Real Environment	Maçã	Multispectral	Top View	TensorFlow	Data Augmentation
S26	(ARAVIND <i>et al.</i> , 2019)	2019	3,5	AlexNet	Multiclass Classifier	99,16%	PlantVillage Subset 4063 Images / 4 Class Controlled Environment	Uva	RGB	Top View	MatLab	Layers Custom
S27	(KHAMPARIA <i>et al.</i> , 2019)	2019	3,5	Convolutional Encoder Network	Multiclass Classifier	99,9%	PlantVillage Subset 900 Images / 6 Class Controlled Environment	Milho Tomate Batata	RGB	Top View	Python	Layers Custom & Hyperparameters

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
				ResNet SiamesaNet		99,4% 94,3%	PlantVillage Subset 4063 Images / 4 Class Controlled Environment					
S28	(GONCHAROV <i>et al.</i> , 2019)	2019	3,5		Multiclass Classifier			Uva	RGB	Top View	TensorFlow	Layers Custom
				ResNet SiamesaNet		98,0% 92,0%	Customized Internet 170 Images / 4 Class Real Environment					
S29	(TODA; OKURA, 2019)	2019	3,5	InceptionV3	Multiclass Classifier	97,14%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow/Keras	Layers Custom 75% reduction in parameters
S30	(TÜRKOGLU; HANBAY, 2019)	2019	3,5	ResNet	Multiclass Classifier	97,86%	Turquia Customized 1.965 Images / 8 Class Controlled Environment	Diversos	RGB	Front View	MatLab	Layers Custom
S31	(HOWLADER <i>et al.</i> , 2019)	2019	3,5	AlexNet	Multiclass Classifier	98,74%	BU-Guava-Leaf 2.075 Images / 4 Class Controlled Environment	Goiaba	RGB	Top View	TensorFlow/Keras	Fine Tune & Hyperparameters
S32	(JIANG <i>et al.</i> , 2019)	2019	3,5	VGG-INCEP	Multiclass Classifier	97,14%	ALDD 26.377 Images / 6 Class Real Environment	Maçã	RGB	Top View	Caffe	Layers Custom
S33	(LIN <i>et al.</i> , 2019b)	2019	3,5	U-Net Segmentation	Binary Segmentation	96,08%	Customized 50 Images / 2 Class Controlled Environment	Pepino	Multispectral	Top View	TensorFlow/Keras	Data Augmentation & Image Segmentation
S34	(LABAÑA <i>et al.</i> , 2019)	2019	3,5	Baseado LeNet	Multiclass Classifier	91,0%	Customized 2000 Images / 4 Class Controlled Environment	Citrus	RGB	Top View	TensorFlow/Keras	Layers Custom & Data Augmentation
S35	(ARSENOVIC <i>et al.</i> , 2019)	2019	3,5	PlantDiseaseNet Architecture	Object Detection	93,67%	PlantVillage (54.306) PlantDataSet (24.959) 79.265 Images / 42 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow/Keras	New Architecture
S36	(COULIBALY <i>et al.</i> , 2019)	2019	3,5	VGG16	Binary Classifier	95,0%	Customized 124 Images / 2 Class Controlled Environment	Millet	RGB	Top View	TensorFlow/Keras	Transfer Learning
S37	(POLDER <i>et al.</i> , 2019)	2019	3,5	FCN	Binary Classifier	88,0%	Customized 22.000 Images / 2 Class Controlled Environment	Batata	Hyperspectral	Top View	PyTorch	Fine Tune & Hyperparameters & Image Segmentation
S38	(HU <i>et al.</i> , 2019)	2019	3,5	CIFAR-10Net	Multiclass Classifier	95,5%	Customized 144 Images / 4 Class Real Environment	Tear Leaf	RGB	Top View	TensorFlow	Layers Custom
S39	(YU; SON, 2019)	2019	3,5	ROI-aware DCNN Based VGG	Multiclass Classifier	84,3%	Apple Research Soul Korea 865 Images / 4 Class Real Environment	Maçã	RGB	Top View	TensorFlow	Transfer Learning & Image Segmentation
S40	(RAMCHARAN <i>et al.</i> , 2019)	2019	3,5	MobileNet & SSD	Object Detection	94,0%	IITA Cassava 2.017 Images / 7 Class Controlled Environment	Mandioca	RGB	Top View	TensorFlow	Transfer Learning

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
						75,0%	Customized 1.375 Images / 7 Class Real Environment					
S41	(G.; J., 2019)	2019	3,5	Custom CNN Based AlexNet	Multiclass Classifier	96,46%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow/Keras	Layers Custom
S42	(HANG <i>et al.</i> , 2019)	2019	3	Custom Inception	Multiclass Classifier	91,7%	IA Challenger Plant Leaf Disease 4.523 Images / 10 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Layers Custom
S43	(SIBIYA; SUMBWANYAMBE, 2019)	2019	3	CNN 50 hidden layers	Multiclass Classifier	92,85%	Customized 100 Images / 3 Class Real Environment	Milho	RGB	Front View	MatLab	New Architecture
S44	(WANG <i>et al.</i> , 2019)	2019	3	Mask R-CNN e ResNet101	Object Detection	99,64%	Customized 1.430 Images / 11 Class Real Environment	Tomate	RGB	Front View	TensorFlow	Image Segmentation
S45	(OZGUVEN; ADEM, 2019)	2019	3	Faster R-CNN	Object Detection	95,48%	Customized 155 Images / 4 Class Real Environment	Sugar-beet	RGB	Top View	MatLab	Layers Custom
S46	(CHEN <i>et al.</i> , 2019b)	2019	3	LeafNet	Multiclass Classifier	99,84%	Customized 7.905 Images / 7 Class Controlled Environment	Tear Leaf	RGB	Top View	MatLab	Fine Tune & Hyperparameters
S47	(WU <i>et al.</i> , 2019)	2019	3	ResNet-34	Binary Classifier	95,1%	Customized 6.267 Images / 2 classes Real Environment	Milho	RGB	Aerial View	PyTorch	Fine Tune & Hyperparameters
S48	(ASHQAR; ABU-NASER, 2019)	2019	3	LeNet	Multiclass Classifier	99,84%	PlantVillage Subset 9.000 Images / 6 Class Controlled Environment	Tomate	RGB	Top View	Caffe	Layers Custom
S49	(ZHANG <i>et al.</i> , 2019b)	2019	3	DCGAN + Custom AlexNet	Binary Classifier	90,0%	Customized 800 Images / 2 Class Real Environment	Citrus	RGB	Front View	TensorFlow/Keras	Layers Custom e Data Augmentation
S50	(XING <i>et al.</i> , 2019)	2019	3	WeaklyDenseNet-16	Multiclass Classifier	99,83%	Customized 3.510 Images / 17 Class Controlled Environment	Citrus	RGB	Front View	TensorFlow	Layers Custom
S51	(ABADE <i>et al.</i> , 2019)	2019	3	M-AlexNet M-GoogLeNet	Multiclass Classifier	99,59% 99,55%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow/Keras	Fine Tune & Hyperparameters
S52	(DAWEI <i>et al.</i> , 2019)	2019	3	AlexNet	Multiclass Classifier	98,92%	Customized 484 Images / 10 Class Real Environment	Tear Leaf	RGB	Front View	MatLab	Transfer Learning
S53	(KAYA <i>et al.</i> , 2019)	2019	3	VGG16 VGG16/LDA	Multiclass Classifier	99,8% 99,0%	PlantVillage 86.198 Images / 57 Class Controlled Environment Flavia 1.600 Images / 32 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Transfer Learning & Fine Tune

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
				AlexNet/LDA		96,20%	UCI Leaf 443 Images / 40 Class Real Environment					
				CNN Custom		98,8%	Swedish 1.125 Images / 15 Class Controlled Environment					
S54	(LIN <i>et al.</i> , 2019a)	2019	3	Fourier Dense Network (FDN)	Multiclass Classifier	92,85%	VOAI UAV 2.293 Images / 12 Class Real Environment	Diversos	RGB	Aerial View	Caffe	New Architecture
S55	(SURYAWATI <i>et al.</i> , 2018)	2018	4,5	VGG	Multiclass Classifier	95,24%	PlantVillage Subset 18.160 Images / 10 Class Controlled Environment	Tomate	RGB	Top View	TensorFlow	Transfer Learning
S56	(GANDHI <i>et al.</i> , 2018)	2018	4,5	InceptionV3	Multiclass Classifier	88,6%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Data Augmentation
				MobileNet		92,0%						
S57	(TOO <i>et al.</i> , 2018)	2018	4,5	VGG InceptionV4 ResNet-50 ResNet-101 ResNet-121 ResNet-152	Multiclass Classifier	98,08% 81,33% 99,59% 99,66% 99,59% 99,75%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	Theano	Fine Tune & Hyperparameters
S58	(LUNA <i>et al.</i> , 2018)	2018	4,5	AlexNet	Multiclass Classifier	95,75%	Customized 4.923 Images / 4 Class Controlled Environment	Tomate	RGB	Top View	MatLab	Layers Custom
S59	(BARBEDO, 2018)	2018	4	GoogLeNet	Multiclass Classifier	87,0%	PDDB Embrapa 1.383 Images / 56 Class Controlled Environment & Real	Diversos	RGB	Top View	TensorFlow/Keras	Transfer Learning & Data Augmentation
S60	(BRAHIMI <i>et al.</i> , 2018)	2018	4	AleNet DenseNet InceptionV3 ResNet SqueezeNet VGG	Multiclass Classifier	99,24% 99,72% 99,76% 99,67% 99,2% 99,49%	PlantVillage 55.038 Images / 39 Class Controlled Environment	Diversos	RGB	Top View	PyTorch	Transfer Learning & Visualization Maps
S61	(SARDOGAN <i>et al.</i> , 2018)	2018	4	Based LeNet + LVQ	Multiclass Classifier	86,0%	PlantVillage Subset 500 Images / 5 Class Controlled Environment	Tomate	RGB	Top View	TensorFlow	New Architecture
S62	(FERENTINOS, 2018)	2018	4	AlexNet GoogLeNet VGG	Multiclass Classifier	99,44% 97,7% 99,48%	PlantVillage 87.848 Images / 58 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Transfer Learning
S63	(LI <i>et al.</i> , 2018a)	2018	4	CNN Generative Adversario Networks DCGNAS	Binary Classifier Generator/Discriminator	89,93%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Unsupervised Learning
S64	(OCAMPO; DADIOS, 2018)	2018	4	MobileNet	Multiclass Classifier	89,0%	Customized & PDDB 6.970 Images / 6 Class Controlled Environment	Diversos	RGB	Front View	TensorFlow	Fine Tune & Hyperparameters & Data Augmentation

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S65	(CAP <i>et al.</i> , 2018)	2018	4	Baseado LeNet	Multiclass Classifier	78,0%	Saitama Research Center 960.000 Images / 2 Class Controlled Environment & Real	Pepino	RGB	Front View	TensorFlow	New Architecture
S66	(PARDEDE <i>et al.</i> , 2018)	2018	3,5	AutoEncoder CNN	Binary Classifier Generator/Discriminator	87,01% 80,42	PlantVillage Subset 6.004 Images / 7 Class Controlled Environment	Batata Milho	RGB	Top View	TensorFlow	Unsupervised Learning
S67	(ZHANG <i>et al.</i> , 2018)	2018	3,5	ResNet (SGD)	Multiclass Classifier	96,51%	PlantVillage Subset 5.550 Images / 8 Class Controlled Environment	Tomate	RGB	Top View	MatLab	Fine Tune & Hyperparameters & Data Augmentation
S68	(WANG <i>et al.</i> , 2018)	2018	3,5	Based LeNet	Multiclass Classifier	90,84%	PlantVillage Subset (4.123) Customized (2.430) 6.553 Images / 8 Class Controlled Environment & Real	Arroz Pepino	RGB	Top View	TensorFlow	New Architecture
S69	(SINGH, 2018)	2018	3,5	InceptionV3	Multiclass Classifier	92,7% 82,7% 92,8%	PantVillage 54.306 Images / 38 Class Controlled Environment Customized 196 Images / 4 Class Real Environment Customized 811 Images / 3 Class Real Environment	Diversos Manga Peanut	RGB	Top View	TensorFlow	Transfer Learning
S70	(WANG <i>et al.</i> , 2018)	2018	3,5	OR-AC-GAN	Binary Segmentation	98,03%	Customized 60 Images / 2 Class Controlled Environment	Pimenta	Hyperspectral	Top View	TensorFlow	New Architecture
S71	(ZHANG <i>et al.</i> , 2018)	2018	3,5	GoogLeNet	Multiclass Classifier	98,9%	Customized 500 Images / 9 Class Controlled Environment	Milho	RGB	Front View	Caffe	Layers Custom
S72	(KOBAYASHI <i>et al.</i> , 2018)	2018	3,5	InceptionV3	Multiclass Classifier	93,0%	PlantVillage Subset 9.568 Images / 12 Class Controlled Environment	Maçã	RGB	Top View	TensorFlow/Keras	Transfer Learning & Data Augmentation
S73	(BARBEDO, 2018)	2018	3,5	GoogLeNet	Multiclass Classifier	87,0%	PlantVillage Subset & Customized 10.441 Images / 10 Class Controlled Environment & Real	Milho	RGB	Top View	MatLab	Transfer Learning
S74	(PARK <i>et al.</i> , 2018)	2018	3,5	VGG	Multiclass Classifier	96,0%	Customized 2.639 Images / 4 Class Controlled Environment	Maçã	Hyperspectral	Top View	TensorFlow/Keras	Layers Custom
S75	(FUENTES <i>et al.</i> , 2018)	2018	3,5	Refinement Filter Bank Based ResNet	Object Detection	96,0%	Customized 8.927 Images / 10 Class Controlled Environment & Real	Tomate	RGB	Front View	Caffe	New Architecture
S76	(YUAN <i>et al.</i> , 2018)	2018	3	AlexNet	Multiclass Classifier	95,93%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Transfer Learning
S77	(PICON <i>et al.</i> , 2018)	2018	3	ResNet	Multiclass Classifier	96,0%	Customized 8.178 Images / 4 Class Real Environment	Trigo	RGB	Front View	TensorFlow	Layers Custom & Image Segmentation

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S78	(TM <i>et al.</i> , 2018)	2018	3	LeNet	Multiclass Classifier	94,85%	PlantVillage Subset 18.160 Images / 10 Class Controlled Environment	Tomate	RGB	Top View	TensorFlow/Keras	Layers Custom
S79	(YANG <i>et al.</i> , 2018)	2018	3	ResNet (PENLU)	Multiclass Classifier	98,3%	Customized 960 Images / 3 Class Real Environment	Citrus	RGB	Top View	Caffe	Layers Custom
S80	(JIN <i>et al.</i> , 2018)	2018	3	R-CNN	Binary Segmentation	95,0%	Customized 10.784 Images Controlled Environment	Milho	Hyperspectral	Top View	Caffe	Image Segmentation
S81	(WALLELIGN <i>et al.</i> , 2018)	2018	3	LeNet	Multiclass Classifier	99,32%	PlantVillage Subset 12.673 Images / 4 Class Controlled Environment	Soja	RGB	Top View	TensorFlow	Layers Custom
S82	(DANG <i>et al.</i> , 2018)	2018	3	GoogLeNet	Multiclass Classifier	90,0%	Customized 40 Images 1500 ROI / 3 Class Controlled Environment	Rabanete	RGB	Aerial View	PyTorch	Fine Tune & Hyperparameters & Image Segmentation
S83	(REN <i>et al.</i> , 2018)	2018	3	VGG	Multiclass Classifier	99,04%	Pest Image DataSet 32.000 Images / 32 Class Controlled Environment	Diversos	RGB	Front View	TensorFlow/Keras	Layers Custom
S84	(GHOSAL <i>et al.</i> , 2018)	2018	3	CNN xPLNet Based LeNet	Multiclass Classifier	95,04%	Customized 25.000 Images / 7 Class Controlled Environment	Soja	RGB	Top View	PyTorch	New Architecture
S85	(KHAN <i>et al.</i> , 2018)	2018	3	VGG	Multiclass Classifier	98,6%	PlantVillage Subset CASC-IFW 6.309 Images / 6 Class Controlled Environment	Banana & Apple	Multispectral	Front View	Caffe	Transfer Learning & Fine Tune
S86	(ZENG <i>et al.</i> , 2018)	2018	3	HoResNet Based ResNet	Multiclass Classifier	91,79%	PlantVillage Subset 10.478 Images / 6 Class Controlled Environment	Tomate	RGB	Top View	TensorFlow	New Architecture
						90,14%	AES-CD9214 9.214 Images / 6 Class Controlled Environment		Arroz			
S87	(ALFARISY <i>et al.</i> , 2018)	2018	3	CaffeNet	Multiclass Classifier	87,0%	Customized 4.511 Images / 13 Class Real Environment	Arroz	RGB	Front View	Caffe	Data Augmentation
S88	(LIM; ESTUAR, 2018)	2018	3	MobileNet	Binary Classifier	65,93%	Customized Fungos 122 Images / 2 Class Controlled Environment	Banana	Microscopic	Top View	TensorFlow/Keras	Fine Tune & Hyperparameters & Data Augmentation
S89	(MA <i>et al.</i> , 2018)	2018	3	AlexNet	Multiclass Classifier	93,4%	Customized 1.184 Images / 5 Class Controlled Environment & Real	Pepino	RGB	Top View	MatLab	Image Segmentation & Data Augmentation
S90	(ATOLE; PARK, 2018)	2018	3	AlexNet	Multiclass Classifier	91,23%	Customized 600 Images / 3 Class Controlled Environment	Arroz	RGB	Front View	MatLab	Transfer Learning
S91	(RAHMAN <i>et al.</i> , 2018)	2018	3	Based VGG16 Stacked Mobile	Multiclass Classifier	95,0%	Customized 1.426 Images / 9 Class Real Environment	Arroz	RGB	Front View	TensorFlow/Keras	New Architecture

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S92	(OPPENHEIM <i>et al.</i> , 2019)	2018	3	VGG	Multiclass Classifier	96,0%	Customized 493 Images / 5 Class Controlled Environment	Batata	RGB	Front View	MatLab	Fine Tune & Hyperparameters
S93	(RANGARAJAN <i>et al.</i> , 2018)	2018	3	AlexNet VGG16	Multiclass Classifier	97,49% 97,23%	PlantVillage Subset 13.262 Images / 6 Class Controlled Environment	Tomate	RGB	Top View	MatLab	Transfer Learning & Fine Tune
S94	(LEE <i>et al.</i> , 2018)	2018	3	VGG & F-RCNN	Object Detection	69,79%	TRES Taiwan Tea Research 1.269 Images / 3 Class Controlled Environment	Tear Leaf	RGB	Top View	Caffe	Image Segmentation
S95	(LI <i>et al.</i> , 2018b)	2018	3	CaspNet	Binary Classifier	N/A	Customized UAV 500 Images / 2 Class Controlled Environment	Arroz	RGB	Aerial View	MatLab	Image Segmentation
S96	(WANG; ZHANG, 2018)	2018	3	FCN AlexNet	Multiclass Segmentation	96,0%	Customized 750 Images / 6 Class Controlled Environment	Milho	RGB	Top View	MatLab	Fine Tune & Image Segmentation
S97	(KERKECH <i>et al.</i> , 2018)	2018	3	LeNet-5	Multiclass Classifier	95,8%	Customized UAV 70.560 Images / 4 Class Controlled Environment	Uva	RGB	Aerial View	MatLab	Image Segmentation
S98	(FUJITA <i>et al.</i> , 2018)	2018	3	VGG	Multiclass Classifier	93,6%	Customized 9.000 Images / 9 Class Controlled Environment	Pepino	RGB	Front View	Caffe	Transfer Learning
S99	(DURMUŞ <i>et al.</i> , 2017)	2017	4,5	AlexNet SqueezeNet	Multiclass Classifier	95,65% 94,3%	PlantVillage 54.306 Images / 38 Class Controlled Environment	Tomate	RGB	Top View	Caffe	Transfer Learning
S100	(LU <i>et al.</i> , 2017)	2017	4	Based LeNet	Multiclass Classifier	95,48%	Customized 500 Images / 10 Class Controlled Environment	Arroz	RGB	Front View	PyTorch	New Architecture
S101	(WANG <i>et al.</i> , 2017b)	2017	4	VGG	Multiclass Classifier	90,4%	PlantVillage Subset 2.086 Images / 4 Class Controlled Environment	Maçã	RGB	Top View	Theano	Fine Tune & Hyperparameters
S102	(FUENTES <i>et al.</i> , 2017)	2017	4	R-FCN & ResNet	Object Detection	85,98%	Customized 5.000 Images / 9 Class Controlled Environment	Tomate	RGB	Front View	TensorFlow/Keras	Data Augmentation
S103	(DECHANT <i>et al.</i> , 2017)	2017	4	Based AlexNet	Binary Classifier	97,8%	Customized 1.796 Images / 2 Class Controlled Environment	Milho	RGB	Front View	Theano	Image Segmentation
S104	(PARK <i>et al.</i> , 2017)	2017	3,5	LeNet	Multiclass Classifier	89,7%	Customized 1500 Images / 4 Class Controlled Environment	Morango	RGB	Front View	TensorFlow	Layers Custom
S105	(ATABAY, 2017)	2017	3,5	ResNet	Multiclass Classifier	97,57%	PlantVillage Subset 19.742 Images / 10 Class Controlled Environment	Tomate	RGB	Top View	Caffe	Layers Custom
S106	(RAMCHARAN <i>et al.</i> , 2017)	2017	3,5	Inception V3	Multiclass Classifier	98,0%	Original Cassava (2.756) Leaf Cassava (15.000) Controlled Environment & Real	Mandioca	RGB	Front View	TensorFlow	Transfer Learning
S107	(PENG; CAI, 2017)	2017	3,5	FCN	Binary Segmentation	87,5%	Customized 546 Images / 2 Class Controlled Environment	Maçã	RGB	Top View	Caffe	Image Segmentation

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S108	(YAMAMOTO <i>et al.</i> , 2017)	2017	3	AlexNet	Multiclass Classifier	78%	PlantVillage Subset 18.149 Images / 9 Class Controlled Environment	Tomate	RGB	Top View	MatLab	Fine Tune & Hyperparameters
S109	(SULC; MATAS, 2017)	2017	3	Inception-ResNet V2	Multiclass Classifier	98,8%	Middle European Wood 9.745 Images / 153 Class Controlled Environment	Diversos	RGB	Top View	PyTorch	Fine Tune & Layers Custom
						97,1%	Austrian Federal Forest (AFF) 134 Images / 5 Class Controlled Environment					
						99,4%	Flavia leaf Dataset 1907 Images / 32 Class Controlled Environment					
						99,2%	Foliage Leaf Dataset 6.000 Images / 60 Class Controlled Environment					
						99,7%	Swedish Leaf Dataset 1.125 Images / 15 Class Controlled Environment					
						81,2%	Leafsnap Dataset 23.147 Images / 185 Class Controlled Environment					
S110	(CORTES, 2017)	2017	3	LeNet & GANs	Multiclass Classifier	78%	PlantVillage Segmentado 86.147 Images / 57 Class Controlled Environment	Diversos	RGB	Top View	TensorFlow	Layers Custom
S111	(SUN <i>et al.</i> , 2017)	2017	3	LeNet	Multiclass Classifier	89,6%	BjfuGloxinia 321 Images / 3 Class Controlled Environment	Flores	RGB	Top View	TensorFlow/Keras	Layers Custom
S112	(HA <i>et al.</i> , 2017)	2017	3	VGG	Multiclass Classifier	97,4%	Customized 1.734 Images / 4 Class Real Environment	Rabanete	RGB	Aerial View	Caffe	Layers Custom
S113	(LIU <i>et al.</i> , 2017)	2017	3	Based AlexNet	Multiclass Classifier	97,62%	Customized 13.689 Images / 4 Class Controlled Environment	Maçã	RGB	Top View	Caffe	New Architecture
S114	(AMARA <i>et al.</i> , 2017)	2017	3	Based LeNet	Multiclass Classifier	92,88%	PlantVillage Subset 3.700 Images / 3 Class Controlled Environment	Banana	RGB	Top View	MatLab	New Architecture
S115	(SLADOJEVIC <i>et al.</i> , 2016)	2016	4,5	CaffeNet	Multiclass Classifier	96,3%	Customized 4.483 Images / 15 Class Controlled Environment	Diversos	RGB	Front View	Caffe	Fine Tune & Hyperparameters
S116	(TAN <i>et al.</i> , 2016)	2016	4	Based LeNet	Multiclass Classifier	95,5%	Customized 400 Images Controlled Environment	Maçã	RGB	Top View	MatLab	New Architecture

continua na próxima página...

Tabela A.2 – continua na próxima página...

ID	Ref	Ano	IQ	Algoritmo	Tarefa do Modelo	Melhor Acurácia	Característica Dataset	Tipo de Cultura	Tipo de Imagens	Perspectiva Visão Imagem	Linguagem & Framework	Abordagem Predominante
S117	(MOHANTY <i>et al.</i> , 2016)	2016	3,5	AlexNet	Multiclass Classifier	99,27%	PlantVillage	Diversos	RGB	Front View	Caffe	Transfer Learning
				GoogLeNet		97,24%	54.306 Images / 38 Class Controlled Environment					
S118	(NACHTIGALL <i>et al.</i> , 2016)	2016	3	AlexNet	Multiclass Classifier	97,3%	TreeApple Embrapa 1.450 Images / 3 Class Controlled Environment	Maçã	RGB	Top View	TensorFlow/Keras	Fine Tune & Hyperparameters
S119	(FUJITA <i>et al.</i> , 2016)	2016	3	VGG	Multiclass Classifier	82,9%	Saitama Research Center 7,320 Images / 4 Class Controlled Environment	Pepino	RGB	Top View	Caffe	Layers Custom & Data Augmentation
				AlexNet		75,05%	Customized 7.520 Images / 4 Class Controlled Environment & Real					
S120	(ATABAY, 2016)	2016	3	Based AlexNet	Multiclass Classifier	97,24%	Flavia 1.907 Images / 32 Class Controlled Environment	Diversos	RGB	Top View	MatLab	New Architecture
						99,11%	Swedish 75 Images / 15 Class Controlled Environment					
S121	(KAWASAKI <i>et al.</i> , 2015)	2015	4,5	CaffeNet	Multiclass Classifier	94,9%	Customized 800 Images / 3 Class Controlled Environment	Pepino	RGB	Top View	Caffe	Data Augmentation

A.2 Capítulo 06 - NEMANet

Tabela A.3: Métricas para avaliação dos diferentes modelos implementados neste estudo.

CNNs	Tipo Treinamento	K-Fold	Métricas											
			Sem Aumento de Dados						Com Aumento de Dados					
			Loss	Accuracy	F1-Score	Precision	Recall	Specificity	Loss	Accuracy	F1-Score	Precision	Recall	Specificity
VGG16 100 Epochs Batch Size = 32	FS	Fold 1	1,2761	0,7895	0,7981	0,8011	0,7953	0,9432	0,3713	0,8711	0,8692	0,8989	0,8437	0,9462
		Fold 2	1,1609	0,7928	0,7918	0,7932	0,7906	0,9444	0,4404	0,8336	0,8358	0,8757	0,8031	0,9382
		Fold 3	1,4579	0,8091	0,8101	0,8114	0,8088	0,9487	0,4041	0,8580	0,8525	0,8855	0,8265	0,9463
		Fold 4	1,2584	0,7810	0,7888	0,7955	0,7828	0,9436	0,4026	0,8545	0,8518	0,8869	0,8218	0,9420
		Fold 5	0,8869	0,7859	0,7857	0,7924	0,7796	0,9421	0,4911	0,8415	0,8506	0,8633	0,8390	0,9429
	Média	1,2081	0,7917	0,7949	0,7987	0,7914	0,9444	0,4219	0,8517	0,8520	0,8821	0,8268	0,9431	
	TL	Fold 1	0,2378	0,9282	0,9316	0,9354	0,9281	0,9767	0,2398	0,9282	0,9309	0,9443	0,9187	0,9726
		Fold 2	0,2285	0,9265	0,9302	0,9359	0,9250	0,9723	0,2584	0,9233	0,9256	0,9317	0,9203	0,9693
		Fold 3	0,3069	0,9184	0,9220	0,9273	0,9171	0,9711	0,3155	0,9053	0,9038	0,9153	0,8937	0,9659
		Fold 4	0,2495	0,9215	0,9185	0,9251	0,9125	0,9735	0,2214	0,9281	0,9308	0,9372	0,9250	0,9703
		Fold 5	0,3010	0,9101	0,9114	0,9153	0,9078	0,9706	0,2415	0,9215	0,9245	0,9327	0,9171	0,9704
	Média	0,2647	0,9209	0,9228	0,9278	0,9181	0,9728	0,2553	0,9213	0,9231	0,9322	0,9150	0,9697	
Xception 100 Epochs Batch Size = 32	FS	Fold 1	0,1883	0,9477	0,9796	0,9543	0,94536	0,9834	0,1360	0,9543	0,9591	0,9621	0,9562	0,9836
		Fold 2	0,1237	0,9575	0,9533	0,9541	0,9526	0,9843	0,4501	0,9543	0,9494	0,9494	0,9494	0,9814
		Fold 3	0,4136	0,9363	0,9405	0,9420	0,9390	0,9825	0,1578	0,9543	0,9461	0,9580	0,9354	0,9775
		Fold 4	0,1855	0,9526	0,9426	0,9447	0,9406	0,9828	0,2521	0,9133	0,9211	0,9289	0,9140	0,9733
		Fold 5	0,1000	0,9673	0,9671	0,9687	0,9656	0,9883	0,1487	0,9591	0,9616	0,9640	0,9593	0,9828
	Média	0,2022	0,9523	0,9506	0,9528	0,9466	0,9843	0,2289	0,9471	0,9475	0,9525	0,9429	0,9797	
	TL	Fold 1	0,1718	0,9543	0,9542	0,9588	0,9499	0,9783	0,1134	0,9690	0,9699	0,9746	0,9656	0,9827
		Fold 2	0,1758	0,9510	0,9511	0,9558	0,9468	0,9778	0,1351	0,9624	0,9630	0,9652	0,9609	0,9830
		Fold 3	0,1584	0,9641	0,9619	0,9680	0,9562	0,9768	0,1565	0,9526	0,9521	0,9561	0,9484	0,9806
		Fold 4	0,1572	0,9542	0,9528	0,9575	0,9484	0,9797	0,1210	0,9673	0,9685	0,9717	0,9656	0,9841
		Fold 5	0,1217	0,9705	0,9683	0,9746	0,9624	0,9812	0,1527	0,9591	0,9611	0,9681	0,9546	0,9785
	Média	0,1570	0,9588	0,9577	0,9629	0,9528	0,9788	0,1357	0,9621	0,9629	0,9671	0,9590	0,9818	
InceptionV3 100 Epochs Batch Size = 32	FS	Fold 1	0,1284	0,9608	0,9507	0,9514	0,9499	0,9857	0,1568	0,9624	0,9607	0,9639	0,9578	0,9829
		Fold 2	0,1629	0,9592	0,9613	0,9635	0,9593	0,9851	0,1890	0,9641	0,9638	0,9668	0,9609	0,9835
		Fold 3	0,2748	0,9592	0,9483	0,9498	0,9468	0,9847	0,1870	0,9461	0,9448	0,9514	0,9390	0,9790
		Fold 4	0,1900	0,9575	0,9593	0,9593	0,9593	0,9866	0,1639	0,9591	0,9605	0,9652	0,9562	0,9814
		Fold 5	0,1931	0,9542	0,9550	0,9588	0,9515	0,9860	0,1666	0,9477	0,9511	0,9556	0,9468	0,9774
	Média	0,1898	0,9582	0,9549	0,9566	0,9534	0,9856	0,1726	0,9559	0,9562	0,9606	0,9521	0,9808	
	TL	Fold 1	0,1618	0,9510	0,9527	0,9557	0,9499	0,9855	0,1218	0,9722	0,9725	0,9750	0,97031	0,9871
		Fold 2	0,1747	0,9543	0,9561	0,9577	0,9546	0,9856	0,0897	0,9787	0,9803	0,9828	0,9781	0,9891
		Fold 3	0,1629	0,9559	0,9585	0,9592	0,9578	0,9861	0,1216	0,9592	0,9629	0,9685	0,9578	0,9852
		Fold 4	0,1578	0,9509	0,9538	0,9546	0,9531	0,9859	0,1441	0,9624	0,9629	0,9666	0,9593	0,9837
		Fold 5	0,1817	0,9493	0,9535	0,9572	0,9499	0,9849	0,1438	0,9575	0,9594	0,9663	0,9531	0,9838
	Média	0,1678	0,9523	0,9549	0,9569	0,9531	0,9856	0,1242	0,9660	0,9678	0,9718	0,9637	0,9857	
ResNet50 100 Epochs Batch Size = 100	FS	Fold 1	0,2630	0,9184	0,9100	0,9191	0,9019	0,9740	0,4112	0,9168	0,9044	0,9146	0,8947	0,9644
		Fold 2	0,3806	0,9037	0,9134	0,9155	0,9114	0,9741	0,2887	0,9070	0,9214	0,9287	0,9142	0,9703
		Fold 3	0,2278	0,9151	0,9263	0,9329	0,9200	0,9763	0,2208	0,9298	0,9307	0,9346	0,9269	0,9694
		Fold 4	0,2900	0,9248	0,9225	0,9369	0,9095	0,9734	0,2733	0,9101	0,9190	0,9255	0,9128	0,9673
		Fold 5	0,7280	0,9101	0,8920	0,8941	0,8899	0,9674	0,2743	0,9019	0,9092	0,9262	0,8938	0,9670
	Média	0,3779	0,9144	0,9128	0,9197	0,9065	0,9731	0,2937	0,9131	0,9169	0,9259	0,9085	0,9677	
	TL	Fold 1	0,3062	0,9037	0,9141	0,9246	0,9042	0,9663	0,1593	0,9559	0,9609	0,9677	0,9542	0,9797
		Fold 2	0,2933	0,9151	0,9257	0,9302	0,9214	0,9703	0,1570	0,9494	0,9552	0,9605	0,9499	0,9815
		Fold 3	0,3569	0,8890	0,8879	0,9060	0,8747	0,9623	0,1258	0,9494	0,9541	0,9636	0,9455	0,9822
		Fold 4	0,4458	0,8921	0,9092	0,9142	0,9042	0,9660	0,2942	0,9460	0,9533	0,9568	0,9499	0,9784
		Fold 5	0,2865	0,9117	0,9101	0,9263	0,8961	0,9657	0,1934	0,9330	0,9327	0,9374	0,9280	0,9762
	Média	0,3377	0,9023	0,9094	0,9203	0,9001	0,9661	0,1859	0,9467	0,9512	0,9572	0,9455	0,9796	
ResNet101 100 Epochs Batch Size = 64	FS	Fold 1	0,3711	0,9004	0,8962	0,9032	0,8896	0,9684	0,2555	0,9021	0,9062	0,9164	0,8965	0,9678
		Fold 2	0,3871	0,8939	0,8946	0,9007	0,8889	0,9662	0,2004	0,9314	0,9301	0,9359	0,9244	0,9751
		Fold 3	0,3790	0,8923	0,8965	0,9028	0,8906	0,9692	0,8689	0,9314	0,9332	0,9402	0,9266	0,9745
		Fold 4	0,3312	0,9052	0,9055	0,9095	0,9017	0,9699	0,2595	0,9232	0,9246	0,9317	0,9178	0,9711
		Fold 5	0,7260	0,9003	0,9001	0,9178	0,8873	0,9623	0,2490	0,9150	0,9134	0,9198	0,9072	0,9701
	Média	0,4389	0,8984	0,8990	0,9068	0,8916	0,9672	0,3667	0,9206	0,9215	0,9288	0,9145	0,9717	
	TL	Fold 1	0,1528	0,9429	0,9456	0,9487	0,9426	0,9820	0,1408	0,9624	0,9631	0,9654	0,9609	0,9845
		Fold 2	0,1365	0,9592	0,9594	0,9640	0,9550	0,9826	0,0947	0,9706	0,9684	0,9762	0,9609	0,9863
		Fold 3	0,1616	0,9461	0,9484	0,9527	0,9442	0,9810	0,1035	0,9771	0,9749	0,9794	0,9706	0,9837
		Fold 4	0,2359	0,9477	0,9470	0,9551	0,9394	0,9807	0,1024	0,9705	0,9696	0,9734	0,9659	0,9875
		Fold 5	0,2003	0,9379	0,9345	0,9404	0,9288	0,9750	0,1536	0,9477	0,9481	0,9539	0,9425	0,9818
	Média	0,1774	0,9467	0,9470	0,9522	0,9420	0,9803	0,1119	0,9657	0,9648	0,9697	0,9602	0,9847	

Tabela A.3: Métricas para avaliação dos diferentes modelos implementados neste estudo.

CNNs	Tipo Treinamento	K-Fold	Métricas											
			Sem Aumento de Dados						Com Aumento de Dados					
			Loss	Accuracy	F1-Score	Precision	Recall	Specificity	Loss	Accuracy	F1-Score	Precision	Recall	Specificity
ResNet152 100 Epochs Batch Size = 64	FS	Fold 1	0,3265	0,9021	0,9045	0,9184	0,8913	0,9621	0,2488	0,9151	0,9138	0,9255	0,9028	0,9674
		Fold 2	0,7708	0,8858	0,8897	0,8903	0,8890	0,9667	0,2722	0,9053	0,9068	0,9147	0,8992	0,9637
		Fold 3	0,3490	0,8890	0,8881	0,8959	0,8806	0,9651	0,2529	0,9200	0,9195	0,9355	0,9044	0,9659
		Fold 4	0,3123	0,9003	0,9056	0,9120	0,8994	0,9670	0,2664	0,9150	0,9147	0,9232	0,9065	0,9660
		Fold 5	0,3497	0,8856	0,8832	0,8893	0,8776	0,9631	0,3383	0,9150	0,9208	0,9289	0,9131	0,9671
	Média	0,4216	0,8925	0,8942	0,9012	0,8876	0,9648	0,2757	0,9141	0,9151	0,9256	0,9052	0,9660	
	TL	Fold 1	0,1692	0,9575	0,9563	0,9608	0,9520	0,9815	0,1585	0,9543	0,9575	0,9621	0,9531	0,9794
		Fold 2	0,1033	0,9575	0,9589	0,9633	0,9546	0,9841	0,1695	0,9445	0,9446	0,9492	0,9402	0,9781
		Fold 3	0,1390	0,9608	0,9612	0,9681	0,9546	0,9817	0,1502	0,9494	0,9486	0,9532	0,9442	0,9786
		Fold 4	0,1650	0,9444	0,9412	0,9480	0,9347	0,9798	0,1355	0,9575	0,9519	0,9581	0,9460	0,9797
Fold 5		0,1813	0,9477	0,9470	0,9530	0,9413	0,9796	0,2103	0,9673	0,9659	0,9712	0,9609	0,9821	
Média	0,1516	0,9536	0,9529	0,9586	0,9474	0,9813	0,1648	0,9546	0,9537	0,9588	0,9489	0,9796		
InceptionResNetV2 100 Epochs Batch Size = 32	FS	Fold 1	0,1658	0,9543	0,9452	0,9609	0,9380	0,9758	0,3494	0,9282	0,9298	0,9402	0,9203	0,9701
		Fold 2	0,2884	0,9494	0,9413	0,9598	0,9250	0,9676	0,2163	0,9298	0,9314	0,9420	0,9218	0,9774
		Fold 3	0,1749	0,9396	0,9427	0,9542	0,9328	0,9738	0,1919	0,9396	0,9415	0,9491	0,9343	0,9762
		Fold 4	0,1780	0,9509	0,9567	0,9607	0,9531	0,9763	0,2181	0,9281	0,9276	0,9356	0,9203	0,9727
		Fold 5	0,1711	0,9477	0,9529	0,9645	0,9421	0,9733	0,1439	0,9493	0,9494	0,9557	0,9437	0,9775
	Média	0,1956	0,9484	0,9477	0,9600	0,9382	0,9734	0,2239	0,9350	0,9359	0,9445	0,9281	0,9748	
	TL	Fold 1	0,3027	0,9119	0,9099	0,9182	0,9026	0,9726	0,1337	0,9592	0,9615	0,9638	0,9593	0,9833
		Fold 2	0,2000	0,9461	0,9503	0,9556	0,9453	0,9825	0,1134	0,9657	0,9661	0,9684	0,9640	0,9844
		Fold 3	0,1558	0,9477	0,9502	0,9538	0,9468	0,9830	0,1160	0,9608	0,9622	0,9654	0,9593	0,9853
		Fold 4	0,4213	0,9150	0,9190	0,9265	0,9125	0,9747	0,1912	0,9575	0,9590	0,9620	0,9562	0,9835
Fold 5		0,2331	0,9346	0,9365	0,9405	0,9328	0,9805	0,2144	0,9248	0,9202	0,9356	0,9062	0,9702	
Média	0,2626	0,9311	0,9332	0,9389	0,9280	0,9787	0,1537	0,9536	0,9538	0,9590	0,9490	0,9813		
EfficientNetB0 100 Epochs Batch Size = 100	FS	Fold 1	4,4441	0,8662	0,8849	0,8930	0,8771	0,9663	0,5606	0,8123	0,7765	0,8055	0,7508	0,9257
		Fold 2	0,4529	0,8629	0,8821	0,8859	0,8785	0,9645	0,3413	0,8760	0,8352	0,8505	0,8217	0,9486
		Fold 3	0,5380	0,8548	0,8633	0,8702	0,8566	0,9591	0,4056	0,8646	0,8676	0,9089	0,8309	0,9438
		Fold 4	0,4516	0,8480	0,8148	0,8350	0,7976	0,9464	0,4080	0,8545	0,8699	0,8880	0,8528	0,9501
		Fold 5	0,5692	0,8464	0,8461	0,8521	0,8404	0,9543	3,8824	0,8594	0,8612	0,8810	0,8433	0,9486
	Média	1,2912	0,8556	0,8582	0,8672	0,8500	0,9581	1,1196	0,8534	0,8421	0,8668	0,8199	0,9433	
	TL	Fold 1	0,2579	0,9168	0,9179	0,9329	0,9038	0,9688	0,2610	0,9282	0,9241	0,9255	0,9226	0,9792
		Fold 2	0,2800	0,9004	0,9002	0,9145	0,8866	0,9656	0,2894	0,9331	0,9314	0,9333	0,9296	0,9799
		Fold 3	0,2764	0,8923	0,8948	0,9034	0,8866	0,9651	0,2258	0,9396	0,9424	0,9443	0,9406	0,9827
		Fold 4	0,2362	0,9281	0,9307	0,9509	0,9123	0,9697	0,1592	0,9509	0,9486	0,9494	0,9479	0,9861
Fold 5		0,2179	0,9183	0,9325	0,9397	0,9257	0,9717	0,2813	0,9444	0,9423	0,9446	0,9401	0,9827	
Média	0,2537	0,9112	0,9152	0,9283	0,9030	0,9682	0,2433	0,9392	0,9377	0,9394	0,9361	0,9821		
EfficientNetB3 100 Epochs Batch Size = 32	FS	Fold 1	0,4078	0,8890	0,8939	0,8973	0,8906	0,9693	0,2255	0,9233	0,9162	0,9290	0,9046	0,9703
		Fold 2	7,5936	0,9053	0,9096	0,9116	0,9078	0,9727	0,2532	0,9265	0,9246	0,9363	0,9140	0,9679
		Fold 3	0,2524	0,9265	0,9305	0,9350	0,9265	0,9788	0,2995	0,9021	0,9026	0,9106	0,8953	0,9672
		Fold 4	0,2787	0,9330	0,9362	0,9398	0,9328	0,9794	0,2741	0,9117	0,9036	0,9255	0,8874	0,9632
		Fold 5	0,2251	0,9362	0,9347	0,9402	0,9296	0,9778	0,2375	0,9199	0,9149	0,9281	0,9062	0,9692
	Média	1,7515	0,9180	0,9210	0,9248	0,9174	0,9756	0,2580	0,9167	0,9124	0,9259	0,9015	0,9676	
	TL	Fold 1	9,2479	0,9282	0,9316	0,9441	0,9203	0,9679	0,1357	0,9575	0,9538	0,9630	0,9453	0,9813
		Fold 2	0,2514	0,9119	0,9037	0,9194	0,8901	0,9632	3,2199	0,9526	0,9531	0,9618	0,9453	0,9798
		Fold 3	0,2269	0,9184	0,9116	0,9254	0,8994	0,9645	0,2031	0,9347	0,9410	0,9514	0,9312	0,9762
		Fold 4	0,2441	0,9330	0,9313	0,9449	0,9187	0,9663	0,2081	0,9297	0,9284	0,9377	0,9203	0,9756
Fold 5		0,2664	0,9035	0,8987	0,9240	0,8796	0,9614	0,1283	0,9624	0,9628	0,9684	0,9578	0,9820	
Média	2,0474	0,9190	0,9154	0,9316	0,9016	0,9647	0,7790	0,9474	0,9478	0,9565	0,9399	0,9790		
NasNetLarge 100 Epochs Batch Size = 16	FS	Fold 1	0,2830	0,9347	0,9338	0,9346	0,9331	0,9815	0,2691	0,9441	0,9475	0,9489	0,9461	0,9850
		Fold 2	0,4308	0,9363	0,9363	0,9363	0,9363	0,9826	0,4202	0,9115	0,9126	0,9135	0,9119	0,9779
		Fold 3	0,3026	0,9380	0,9369	0,9393	0,9347	0,9819	0,3226	0,9282	0,9282	0,9282	0,9282	0,9792
		Fold 4	0,3659	0,9183	0,9172	0,9178	0,9166	0,9775	0,2316	0,9375	0,9386	0,9395	0,9379	0,9812
		Fold 5	0,5723	0,9133	0,9156	0,9180	0,9133	0,9773	0,2908	0,9369	0,9371	0,9432	0,9313	0,9782
	Média	0,3909	0,9281	0,9280	0,9292	0,9268	0,9802	0,3069	0,9316	0,9328	0,9347	0,9311	0,9803	
	TL	Fold 1	0,1815	0,9429	0,9422	0,9475	0,9375	0,9753	0,2440	0,9673	0,9679	0,9679	0,9679	0,9910
		Fold 2	0,1679	0,9543	0,9523	0,9546	0,9503	0,9828	0,3481	0,9575	0,9583	0,9583	0,9583	0,9886
		Fold 3	0,1532	0,9461	0,9494	0,9560	0,9439	0,9781	0,4380	0,9624	0,9631	0,9631	0,9631	0,9901
		Fold 4	0,1528	0,9591	0,9605	0,9631	0,9583	0,9819	0,1879	0,9689	0,9695	0,9695	0,9695	0,9912
Fold 5		0,2227	0,9232	0,9236	0,9282	0,9198	0,9759	0,3255	0,9526	0,9535	0,9535	0,9535	0,9881	
Média	0,1756	0,9451	0,9456	0,9499	0,9419	0,9788	0,3087	0,9618	0,9625	0,9625	0,9625	0,9898		

Tabela A.3: Métricas para avaliação dos diferentes modelos implementados neste estudo.

CNNs	Tipo Treinamento	K-Fold	Métricas											
			Sem Aumento de Dados						Com Aumento de Dados					
			Loss	Accuracy	F1-Score	Precision	Recall	Specificity	Loss	Accuracy	F1-Score	Precision	Recall	Specificity
DenseNet121 100 Epochs	FS	Fold 1	0,2156	0,9445	0,9399	0,9479	0,9338	0,9811	0,7888	0,9086	0,9103	0,9201	0,9015	0,9709
		Fold 2	0,3908	0,8988	0,9007	0,9091	0,8932	0,9699	0,2166	0,9200	0,9259	0,9320	0,9203	0,9710
		Fold 3	0,2446	0,9216	0,9158	0,9202	0,9119	0,9731	0,3035	0,9102	0,9162	0,9219	0,9109	0,9693
		Fold 4	1,3428	0,8790	0,8638	0,8721	0,8562	0,9608	0,2317	0,9248	0,9284	0,9320	0,9250	0,9768
		Fold 5	0,4774	0,9003	0,9047	0,9100	0,8999	0,9721	0,2748	0,9084	0,9098	0,9208	0,8999	0,9707
	Média	0,5342	0,9089	0,9050	0,9118	0,8990	0,9714	0,3631	0,9144	0,9181	0,9254	0,9115	0,9717	
	TL	Fold 1	0,0928	0,9722	0,9741	0,9748	0,9734	0,9907	0,0792	0,9738	0,9740	0,9747	0,9734	0,9895
		Fold 2	0,1061	0,9787	0,9788	0,9796	0,9781	0,9916	0,0877	0,9722	0,9731	0,9762	0,9703	0,9882
		Fold 3	0,1260	0,9641	0,9655	0,9670	0,9640	0,9885	0,1066	0,9706	0,9715	0,9746	0,9687	0,9886
		Fold 4	0,1055	0,9640	0,9652	0,9682	0,9624	0,9888	0,0637	0,9869	0,9849	0,9890	0,9812	0,9910
Fold 5		0,0723	0,9787	0,9788	0,9896	0,9781	0,9916	0,0760	0,9771	0,9764	0,9796	0,9734	0,9890	
Média	0,1006	0,9715	0,9725	0,9739	0,9712	0,9902	0,0826	0,9761	0,9760	0,9788	0,9734	0,9893		
DenseNet169 100 Epochs	FS	Fold 1	0,1637	0,9477	0,9475	0,9584	0,9375	0,9774	0,2594	0,9249	0,9202	0,9242	0,9166	0,9715
		Fold 2	0,2216	0,9249	0,9299	0,9372	0,9234	0,9774	0,1924	0,9363	0,9369	0,9430	0,9312	0,9762
		Fold 3	1,1203	0,9298	0,9312	0,9363	0,9265	0,9754	0,2292	0,9102	0,9016	0,9141	0,8906	0,9682
		Fold 4	0,2160	0,9362	0,9388	0,9402	0,9375	0,9781	0,5446	0,9166	0,9132	0,9313	0,8999	0,9703
		Fold 5	0,2239	0,9215	0,9144	0,9164	0,9125	0,9742	0,2171	0,9313	0,9316	0,9425	0,9218	0,9719
	Média	0,3891	0,9320	0,9323	0,9377	0,9274	0,9765	0,2885	0,9239	0,9207	0,9310	0,9120	0,9716	
	TL	Fold 1	0,0782	0,9836	0,9842	0,9875	0,9812	0,9930	0,0860	0,9804	0,9806	0,9823	0,9791	0,9930
		Fold 2	0,0711	0,9836	0,9843	0,9859	0,9828	0,9936	0,0998	0,9820	0,9823	0,9823	0,9823	0,9935
		Fold 3	0,0978	0,9722	0,9734	0,9734	0,9734	0,9900	0,1076	0,9738	0,9742	0,9759	0,9727	0,9912
		Fold 4	0,0835	0,9738	0,9740	0,9747	0,9734	0,9913	0,0732	0,9836	0,9846	0,9853	0,9839	0,9926
Fold 5		0,1117	0,9705	0,9725	0,9733	0,9718	0,9911	0,0683	0,9771	0,9789	0,9805	0,9775	0,9924	
Média	0,0088	0,9768	0,9777	0,9790	0,9765	0,9918	0,0870	0,9794	0,9801	0,9813	0,9791	0,9925		
DenseNet201 100 Epochs	FS	Fold 1	0,2176	0,9298	0,9253	0,9380	0,9171	0,9756	0,3347	0,9037	0,9097	0,9220	0,8984	0,9625
		Fold 2	0,1837	0,9347	0,9358	0,9424	0,9296	0,9803	0,2443	0,9102	0,9039	0,9204	0,8890	0,9690
		Fold 3	2,1346	0,9151	0,9137	0,9223	0,9057	0,9719	0,2959	0,9168	0,9178	0,9270	0,9093	0,9715
		Fold 4	0,3258	0,8905	0,8960	0,9020	0,8906	0,9683	0,2260	0,9281	0,9255	0,9387	0,9140	0,9690
		Fold 5	0,2503	0,9199	0,9112	0,9204	0,9031	0,9732	0,2855	0,9297	0,9161	0,9286	0,9046	0,9687
	Média	0,6224	0,9180	0,9164	0,9250	0,9092	0,9739	0,2773	0,9177	0,9146	0,9273	0,9031	0,9681	
	TL	Fold 1	0,0766	0,9755	0,9763	0,9778	0,9750	0,9915	0,1045	0,9706	0,9726	0,9734	0,9718	0,9918
		Fold 2	0,1063	0,9787	0,9796	0,9796	0,9796	0,9924	0,1153	0,9804	0,9812	0,9812	0,9812	0,9937
		Fold 3	0,0762	0,9820	0,9795	0,9828	0,9765	0,9924	0,1052	0,9738	0,9733	0,9750	0,9718	0,9904
		Fold 4	0,1096	0,9656	0,9678	0,9686	0,9671	0,9896	0,0679	0,9852	0,9873	0,9889	0,9859	0,9945
Fold 5		0,0731	0,9869	0,9765	0,9781	0,9750	0,9917	0,0505	0,9885	0,9866	0,9890	0,9843	0,9946	
Média	0,0884	0,9777	0,9760	0,9774	0,9746	0,9915	0,0887	0,9797	0,9802	0,9815	0,9790	0,9930		
NemaNet 100 Epochs	FS	Fold 1	0,0920	0,9804	0,9807	0,9807	0,9807	0,9929	0,1425	0,9592	0,9613	0,9647	0,9583	0,9846
		Fold 2	0,0950	0,9755	0,9751	0,9759	0,9743	0,9925	0,7319	0,9575	0,9585	0,9606	0,9567	0,9868
		Fold 3	0,4750	0,9526	0,9487	0,9487	0,9487	0,9862	0,1107	0,9657	0,9608	0,9656	0,9567	0,9859
		Fold 4	0,1909	0,9607	0,9612	0,9626	0,9599	0,9891	0,1716	0,9612	0,9618	0,9629	0,9601	0,9871
		Fold 5	0,1556	0,9689	0,9685	0,9693	0,9679	0,9910	0,2033	0,9624	0,9602	0,9624	0,9583	0,9864
	Média	0,2017	0,9676	0,9668	0,9674	0,9663	0,9903	0,2720	0,9612	0,9605	0,9632	0,9580	0,9861	
	TL	Fold 1	0,1370	0,9771	0,9775	0,9775	0,9775	0,9940	0,0649	0,9902	0,9895	0,9903	0,9887	0,9958
		Fold 2	0,1372	0,9820	0,9820	0,9835	0,9807	0,9945	0,1797	0,9902	0,9904	0,9904	0,9904	0,9964
		Fold 3	0,0649	0,9885	0,9887	0,9887	0,9887	0,9963	0,0464	0,9935	0,9918	0,9935	0,9888	0,9964
		Fold 4	0,0495	0,9885	0,9895	0,9903	0,9887	0,9954	0,0822	0,9852	0,9863	0,9871	0,9855	0,9952
Fold 5		0,1359	0,9722	0,9727	0,9727	0,9727	0,9910	0,0815	0,9820	0,9823	0,9823	0,9823	0,9938	
Média	0,1049	0,9817	0,9821	0,9826	0,9817	0,9942	0,0909	0,9882	0,9881	0,9888	0,9872	0,9955		

Apêndice B

Artigos Publicados

Neste Apêndice são apresentados os Artigos publicados ou aceitos para publicação decorrentes dos resultados deste trabalho. Cada Artigo foi listado conforme as normas de citação, sendo adicionadas informações quanto à qualidade e relevância do periódico/conferência ao qual foi submetido para avaliação.

B.1 Artigos Publicados em Periódicos Internacionais

- ABADE, A.; FERREIRA, P. A.; DE BARROS VIDAL, F. Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture*, v. 185, p. 106–125, 2021. ISSN 0168-1699. (Journal Qualis A1, Impact Factor = 5.565, CiteScore = 8.6, Highest percentile = 99%)
- ABADE, A.; PORTO, L. F.; FERREIRA, P. A.; VIDAL, F. de B. *Nemanet: A Convolutional Neural Network Model for Identification of Soybean Nematodes*. *Biosystems Engineering*, abs/2103.03717, 2021. ISSN 1537-5110. (Journal Qualis A1, Impact Factor = 4.123, CiteScore = 7.2, Highest percentile = 93%)

B.2 Artigos Publicados em Conferências Internacionais

- ABADE, A.; ALMEIDA., A. P. G. S. de; VIDAL, F. B. Plant diseases recognition from digital images using multichannel convolutional neural networks. In: *INTICC.Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP*, Prague, Czech Republic: SciTePress, 2019. p. 450–458. ISBN 978-989-758-354-4. (Conference Qualis CC - A3, Índice H5 = 21)

B.3 Primeira Página dos Artigos em Ordem de Publicação

Plant Diseases Recognition from Digital Images using Multichannel Convolutional Neural Networks

Andre da Silva Abade¹, Ana Paula G. S. de Almeida² and Flavio de Barros Vidal³

¹Federal Institute of Education, Science and Technology of Mato Grosso, Brazil

²Department of Mechanical Engineering, University of Brasilia, Distrito Federal, Brazil

³Department of Computer Science, University of Brasilia, Distrito Federal, Brazil

Keywords: Convolutional Neural Networks, Multichannel Convolutional Neural Networks, Plant Disease, Crop Disease Recognition, Computer Vision.

Abstract: Plant diseases are considered one of the main factors influencing food production and to minimize losses in production, it is essential that crop diseases have a fast detection and recognition. Nowadays, recent studies use deep learning techniques to diagnose plant diseases in an attempt to solve the main problem: a fast, low-cost and efficient methodology to diagnose plant diseases. In this work, we propose the use of classical convolutional neural network (CNN) models trained from scratch and a Multichannel CNN (M-CNN) approach to train and evaluate the PlantVillage dataset, containing several plant diseases and more than 54,000 images (divided into 38 diseases classes with 14 plant species). In both proposed approaches, our results achieved better accuracies than the *state-of-the-art*, with faster convergence and without the use of transfer learning techniques. Our multichannel approach also demonstrates that the three versions of the dataset (colored, grayscale and segmented) can contribute to improve accuracy, adding relevant information to the proposed artificial neural network.

1 INTRODUCTION

Plant diseases are considered one of the main factors influencing food production, being responsible for the significant reduction of the physical or economic productivity of the crops and, in certain cases, may be an impediment to this activity. According to Altieri (2018), in order to minimize production losses and maintain crop sustainability, it is essential that disease management and control measures be carried out in an appropriate manner, highlighting the constant monitoring of the crop, combined with the rapid and accurate diagnosis of the diseases. These practices are the most recommended by phytopathologists.

The major challenge for agriculture is the correct identification of the symptoms of major diseases that affect crops (Anderson et al., 2004). Manual and mechanized practices in traditional planting processes are not able to cover large areas of plantation and provide essential early information to decision-making processes (Miller et al., 2009). Thus, it is necessary to develop automated solutions, practical, reliable and economical able to monitor the health of plants providing meaningful information to the decision-making

process (e.g. correct dosage of pesticides (Mahlein, 2016)).

Computer Vision along with Artificial Intelligence (AI) has been developing techniques and methods for recognizing and classifying objects with significant advances (Arnal Barbedo, 2013). These systems use Convolutional Neural Networks (CNNs) (Lecun et al., 1998) and their results in some experiments are already superior to humans in large-scale reconnaissance tasks. The studies presented in Mohanty et al. (2016) and Ferentinos (2018) make use of deep learning techniques in agriculture, in particular in the diagnosis of plant diseases. These approaches have used two popular architectures, namely AlexNet (Krizhevsky et al., 2012) and GoogLeNet (Inception v1) (Szegedy et al., 2014), which were designed in the context of the Scale Visual Recognition Challenge (ILSVRC) (Russakovsky et al., 2014) for the ImageNet dataset (Deng et al., 2009).

With the aforementioned architectures, Mohanty et al. (2016) show that only the colored dataset is sufficient to recognize plant diseases. However, more information about a subject can contribute to improve the network accuracy and to confirm this assump-

450

Abade, A., S. de Almeida, A. and Vidal, F.

Plant Diseases Recognition from Digital Images using Multichannel Convolutional Neural Networks.

DOI: 10.5220/0007383904500458

In Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2019), pages 450-458

ISBN: 978-989-758-354-4

Copyright © 2019 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved



Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Review

Plant diseases recognition on images using convolutional neural networks: A systematic review

 André Abade^{a,*}, Paulo Afonso Ferreira^b, Flavio de Barros Vidal^c
^a Federal Institute of Education, Science and Technology of Mato Grosso, Brazil

^b Department of Agronomy, Federal University of Mato Grosso, Brazil

^c Department of Computer Science, University of Brasilia – Brasilia, Brazil


ARTICLE INFO

Keywords:

 Plant diseases
 Convolutional neural networks
 Crop disease recognition
 Plant pathogen
 SLR

ABSTRACT

Plant diseases are considered one of the main factors influencing food production and minimize losses in production, and it is essential that crop diseases have fast detection and recognition. The recent expansion of deep learning methods has found its application in plant disease detection, offering a robust tool with highly accurate results. In this context, this work presents a systematic review of the literature that aims to identify the state of the art of the use of convolutional neural networks (CNN) in the process of identification and classification of plant diseases, delimiting trends, and indicating gaps. In this sense, we review 121 papers selected in the last ten years with different approaches to treat aspects related to disease detection, characteristics of the dataset, the crops and pathogens investigated. From the results of the systematic review, it is possible to understand the innovative trends regarding the use of CNN's in the identification of plant diseases and to identify the gaps that need the attention of the research community.

1. Introduction

Plant diseases are considered one of the main factors influencing food production, being responsible for the significant reduction of the physical or economic productivity of the crops and, in some instances, maybe an impediment to this activity. According to Altieri (2018), to minimize production losses and maintain crop sustainability, it is essential that disease management and control measures be carried out appropriately, highlighting the constant monitoring of the crop, combined with the rapid and accurate diagnosis of the diseases. These practices are the most recommended by phytopathologists (Altieri, 2018).

The major challenge for agriculture is the correct identification of the symptoms of major diseases that affect crops (Mohanty et al., 2016). Manual and mechanized practices in traditional planting processes cannot cover large areas of plantations and provide essential early information to decision-making processes, according to Miller et al. (2009). Thus, it is necessary to develop automated solutions, practical, reliable, and economically able to monitor the health of plants providing meaningful information to the decision-making process, for example, the application and correct dosage of pesticides in specific treatment certain diseases (Mahlein, 2016).

Computer Vision, along with Artificial Intelligence (AI), has been developing techniques and methods for recognizing and classifying objects with significant advances (Arnal Barbedo, 2013). According to LeCun et al. (2015) and, deep learning allows computational models to learn representations of data with multiple levels of abstraction, improving the state-of-the-art in many domains, such as speech recognition, object recognition, object detection.

The simplest Deep Learning models are called Deep Feedforward, in which information is only propagated in one direction through neurons. Other examples of algorithms are: Back-Propagation, Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), Auto-Encoder (AE), Deep Belief Network (DBN), Generative Adversarial Network (GAN), and Deep Reinforcement Learning (DRL) (Goodfellow et al., 2016). One particular type of deep, feedforward network that was much easier to train and generalized much better than networks with full connectivity was the convolutional neural networks (CNN's) (Salas et al., 2019).

CNN's constitute one of the most powerful techniques for modeling complex processes and performing pattern recognition in applications with a large amount of data, as the pattern recognition in images (LeCun et al., 2015). This is a connectionist approach, stands out as one of the

* Corresponding author.

E-mail address: andre.abade@bag.ifmt.edu.br (A. Abade).<https://doi.org/10.1016/j.compag.2021.106125>

Received 10 July 2020; Received in revised form 2 November 2020; Accepted 21 March 2021

Available online 30 April 2021

0168-1699/© 2021 Elsevier B.V. All rights reserved.

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/issn/15375110

Research Paper

NemaNet: A convolutional neural network model for identification of soybean nematodes

06

05

01

André Abade^{a,*}, Lucas Faria Porto^c, Paulo Afonso Ferreira^b,
Flávio de Barros Vidal^d

^a Federal Institute of Education, Science and Technology of Mato Grosso, Brazil

^b Department of Agronomy, Federal University of Mato Grosso, Brazil

^c Department of Mechanical Engineering, University of Brasília, Distrito Federal, Brazil

^d Department of Computer Science, University of Brasília, Distrito Federal, Brazil

ARTICLE INFO

Article history:

Received 4 March 2021

Received in revised form

27 October 2021

Accepted 17 November 2021

Published online xxx

Keywords:

Plant disease

Phytonematodes

Plant pathogen

Plant-parasitic nematode

CNN

Deep learning

Phytoparasitic nematodes (or phytonematodes) are causing severe damage to crops and generating large-scale economic losses worldwide. In soybean crops, annual losses are estimated at 10.6% of the world production. Besides, the identification of these species through microscopic analysis by an expert with taxonomic knowledge is often laborious, time-consuming, and susceptible to failure. From this perspective, robust and automatic approaches are necessary for identifying phytonematodes that are capable of providing correct diagnoses for the classification of species and subsidizing of all control and prevention measures. This work presents a new public data set called NemaDataset containing 3063 microscopic images from five nematode species with the most significant damage relevance for the soybean crop. Additionally, we propose a new Convolutional Neural Network (CNN) model defined as NemaNet and present a comparative assessment with thirteen popular models of CNNs, all of them representing state-of-the-art classification and recognition. The general average was calculated for each model, on a from-scratch training; the NemaNet model reached 96.76% accuracy, while the best evaluation fold reached 98.04%. When training with transfer learning was performed, the average accuracy reached 98.82%. The best evaluation fold reached 99.35%, and overall accuracy improvements of over 6.83% and 4.1%, for from-scratch and transfer learning training, respectively, compared to other popular models were achieved.

© 2021 IAGrE. Published by Elsevier Ltd. All rights reserved.

1. Introduction

The identification of plant diseases is one of the most basic and important agricultural activities. In most cases, identification is performed manually, visually or by microscopy

(Agrios, 2005; Bos & Parlevliet, 1995). The main problem of the visual assessment for identifying diseases is that the expert takes on a subjective task, prone to psychological and cognitive phenomena that can lead to prejudice, optical illusions, and, finally, error (Barbedo, 2016). Furthermore, laboratory

* Corresponding author.

E-mail address: andre.abade@bag.ifmt.edu.br (A. Abade).

<https://doi.org/10.1016/j.biosystemseng.2021.11.016>

1537-5110/© 2021 IAGrE. Published by Elsevier Ltd. All rights reserved.

Please cite this article as: Abade, A et al., NemaNet: A convolutional neural network model for identification of soybean nematodes, *Biosystems Engineering*, <https://doi.org/10.1016/j.biosystemseng.2021.11.016>