

INSTITUTO FEDERAL DO ESPÍRITO SANTO
MESTRADO EM TECNOLOGIAS SUSTENTÁVEIS

HUMBERTO DA SILVA NETO

**SEGMENTAÇÃO DE PRAGAS E DOENÇAS EM FOLHAS DE CAFÉ UTILIZANDO
REDES CONVOLUCIONAIS**

Vitória
2023

HUMBERTO DA SILVA NETO

**SEGMENTAÇÃO DE PRAGAS E DOENÇAS EM FOLHAS DE CAFÉ UTILIZANDO
REDES CONVOLUCIONAIS**

Trabalho de Dissertação apresentado à Coordenadoria do Curso de Tecnologias Sustentáveis do Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo, como requisito parcial para a obtenção do título de Mestre.

Orientadora: Profa. Dra. Shirley Peroni Neves Cani

Coorientadora: Profa. Dra. Mariana Rampinelli Fernandes

Vitória

2023

Dados Internacionais de Catalogação na Publicação (CIP)
(Biblioteca Nilo Peçanha do Instituto Federal do Espírito Santo)

S586s Silva Neto, Humberto da.

Segmentação de pragas e doenças em folhas de café utilizando
redes convolucionais / Humberto da Silva Neto – 2023.

71 f. : il. ; 30 cm.

Orientadora: Shirley Peroni Neves Cani.

Coorientadora: Mariana Rampinelli Fernandes.

Dissertação (Mestrado) – Instituto Federal do Espírito Santo,
Programa de Pós-graduação em Tecnologias Sustentáveis, Vitória, 2023.

1. Redes neurais (Computação). 2. Sistemas de reconhecimento de
padrões. 3. Processamento de imagens – Técnicas digitais. 4. Café –
Doenças e pragas. 5. Aprendizado do computador. 6. Visão por
computador. I. Cani, Shirley Peroni Neves. II. Fernandes, Mariana
Rampinelli. III. Instituto Federal do Espírito Santo. IV. Título.

CDD 21 – 006.32

Elaborada por Ronald Aguiar Nascimento – CRB-6/MG – 3.116

HUMBERTO DA SILVA NETO

**SEGMENTAÇÃO DE PRAGAS E DOENÇAS EM FOLHAS DE CAFÉ
UTILIZANDO REDES CONVOLUCIONAIS**

Dissertação apresentada ao Mestrado Profissional em Tecnologias Sustentáveis, vinculado ao Programa de Pós-Graduação em Tecnologias Sustentáveis do Instituto Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Tecnologias Sustentáveis.

Aprovada em 14 de junho de 2023

COMISSÃO EXAMINADORA

Documento assinado digitalmente



SHIRLEY PERONI NEVES CANI
Data: 15/06/2023 11:19:11-0300
Verifique em <https://validar.iti.gov.br>

Doutora Shirley Peroni Neves Cani
Instituto Federal do Espírito Santo - Ifes
Orientadora

(Telepresença: Portaria Nº 783 de 19/11/2021 - Campus Vitória)

Documento assinado digitalmente



MARIANA RAMPINELLI FERNANDES
Data: 15/06/2023 14:29:00-0300
Verifique em <https://validar.iti.gov.br>

Doutora Mariana Rampinelli Fernandes
Instituto Federal do Espírito Santo - Ifes
Coorientadora

(Telepresença: Portaria Nº 783 de 19/11/2021 - Campus Vitória)

Documento assinado digitalmente



REGINALDO BARBOSA NUNES
Data: 16/06/2023 16:07:56-0300
Verifique em <https://validar.iti.gov.br>

Doutor Reginaldo Barbosa Nunes
Instituto Federal do Espírito Santo - Ifes
Membro Interno

(Telepresença: Portaria Nº 783 de 19/11/2021 - Campus Vitória)

Documento assinado digitalmente



CLEBESON CANUTO DOS SANTOS
Data: 18/06/2023 16:13:00-0300
Verifique em <https://validar.iti.gov.br>

Doutor Clebeson Canuto dos Santos
Dell Technologies
Membro Externo

(Telepresença: Portaria Nº 783 de 19/11/2021 - Campus Vitória)

AGRADECIMENTOS

Gostaria de expressar minha gratidão a todos que me apoiaram durante esta jornada acadêmica. Primeiramente, gostaria de agradecer à minha família e amigos por estarem constantemente presentes em minha vida e me apoiando em todas as minhas decisões, incluindo a escolha de seguir este caminho.

Gostaria de agradecer também às minhas orientadoras, cuja paciência, apoio e orientação foram essenciais para a realização deste trabalho. Sem o seu apoio e dedicação, este trabalho não teria sido possível.

Além disso, quero agradecer aos meus colegas e professores, cujas ideias e conselhos enriqueceram o meu trabalho. A todos aqueles que dedicaram o seu tempo para discutir ideias, trocar impressões e fornecer *feedback* construtivo.

Viu a necessidade, atenda! (GRANDE SOLDADOR, 2005)

RESUMO

Nos últimos anos, métodos de aprendizado profundo foram introduzidos para segmentação e classificação de lesões em folhas causadas por pragas e outras doenças. Dentre as abordagens comumente utilizadas, as redes neurais convolucionais têm fornecido resultados com alta precisão. Por essa razão, o objetivo deste estudo é apresentar um sistema prático e eficaz que usa redes neurais convolucionais para segmentar e classificar diferentes tipos de lesões em folhas de café e estimar a gravidade do dano causado por agentes bióticos. Neste trabalho é proposto dois *frameworks* diferentes. O primeiro usa segmentação semântica para identificar cada sintoma e calcular sua gravidade, enquanto o segundo usa duas etapas: segmentação semântica para calcular a gravidade e classificação dos sintomas. O primeiro sistema é cerca de 1,80 vezes mais rápido que o segundo e obteve um índice *mIoU* de 84,27%. Já o segundo atingiu 95,04% de *mIoU* e uma acurácia de 97,31% na classificação dos sintomas. Além disso, de acordo com o nosso conhecimento atual, este é o primeiro estudo que sugere a identificação e a avaliação da gravidade de doenças em folhas de café utilizando exclusivamente um modelo de segmentação. Por fim, os resultados são promissores e conseguem segmentar regiões lesionadas com alta precisão.

Palavras-chave: Folhas de Café. Pragas e Doenças. Visão Computacional. Segmentação Semântica. Aprendizado Profundo.

ABSTRACT

In recent years, deep learning methods have been introduced for segmentation and classification of leaf lesions caused by pests and other diseases. Among the commonly used approaches, convolutional neural networks have provided results with high precision. For this reason, the aim of this study is to present a practical and effective system that uses convolutional neural networks to segment and classify different types of lesions on coffee leaves and estimate the severity of the damage caused by biotic agents. In this work two different frameworks are proposed. The first uses semantic segmentation to identify each symptom and calculate its severity, while the second uses two steps: semantic segmentation to calculate severity and classification of symptoms. The first system is about 1.80 times faster than the second and obtained a *mIoU* index of 84.27%. The second reached 95.04% of *mIoU* and an accuracy of 97.31% in the classification of symptoms. Furthermore, according to our current knowledge, this is the first study that suggests the identification and assessment of disease severity in coffee leaves using exclusively a segmentation model. Finally, the results are promising and manage to segment injured regions with high precision.

Keywords: Coffee Leafs. Pests and Diseases. Computer Vision. Semantic Segmentation. Deep Learning.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de folha com bicho-mineiro.	20
Figura 2 – Exemplo de folha com ferrugem.	20
Figura 3 – Exemplo de folha com cercospora.	21
Figura 4 – Exemplo de folha com phoma.	21
Figura 5 – Um exemplo dos diferentes tipos de segmentação. (a) Imagem de entrada. (b) resultado da segmentação semântica. (c) resultado da segmentação de instâncias.	23
Figura 6 – FCNs podem aprender com eficiência a fazer previsões densas para tarefas por <i>pixel</i> , como segmentação semântica.	24
Figura 7 – Resultados da DeepLab. Para cada linha, têm-se a imagem de entrada, o resultado do modelo sem CRF e o resultado da segmentação com CRF.	25
Figura 8 – Segmentação semântica deconvolucional.	25
Figura 9 – Arquitetura U-net.	26
Figura 10 – Arquitetura da DeepLab V3+.	27
Figura 11 – Representação da equação de IoU.	30
Figura 12 – Fluxograma do sistema proposto por Ramos <i>et al.</i> (2017).	32
Figura 13 – Exemplos de imagens usadas nos trabalhos de (a) Ramos <i>et al.</i> (2017) e (b) Rodríguez <i>et al.</i> (2020).	32
Figura 14 – <i>Pipeline</i> proposto em (TASSIS; SOUZA; KROHLING, 2021).	34
Figura 15 – Exemplos dos resultados da segmentação semântica da U=Net e PSPNet.	35
Figura 16 – <i>Pipeline</i> proposto em (ESGARIO <i>et al.</i> , 2022).	36
Figura 17 – Capturas de tela feitas durante o uso do aplicativo: câmera (A); verificação e <i>upload</i> (B); e resultados (C).	36
Figura 18 – <i>Pipelines</i> dos <i>frameworks</i> propostos.	37
Figura 19 – Classes do modelo de segmentação da PROPOSTA A	38
Figura 20 – Classes do modelo de segmentação da PROPOSTA B	39
Figura 21 – Exemplos de imagens da base Bracol. A primeira coluna exhibe as imagens, a central apresenta a máscara original e, por último, as máscaras adaptadas para segmentar cada sintoma individualmente.	41

Figura 22 – Exemplos da variação da base Bracol para classificação.	42
Figura 23 – Aumento de dados aplicados na base Bracol.	45
Figura 24 – Exemplos do aumento de dados realizados por recorte de sintomas.	45
Figura 25 – Exemplos do aumento de dados realizados na base de classificação.	47
Figura 26 – Exemplos de previsões de modelos de segmentação de sintomas treinados com diferentes funções de perda	51
Figura 27 – Exemplos da técnica de aumento de dados para inserir sombras aleatórias.	52
Figura 28 – Curva da perda, acurácia média, coeficiente <i>Dice</i> e <i>mIou</i> durante o treinamento da U-Net++.	56
Figura 29 – Curva da perda, acurácia média, coeficiente <i>Dice</i> e <i>mIou</i> durante o treinamento do DeepLab V3+.	57
Figura 30 – Erros mais grosseiros do segmentador de sintomas no conjunto de validação. A primeira coluna é a imagem de entrada, seguida pela máscara real e a predita. A última coluna representa um recorte da imagem de entrada onde o modelo errou (destacado por uma circunferência vermelha nas máscaras preditas).	58
Figura 31 – Comparação entre as máscaras preditas pelo modelo de segmenta- ção de sintomas e as máscaras verdadeiras.	59
Figura 32 – Lote de treinamento do modelo de classificação contendo 32 imagens com aumento de dados	60
Figura 33 – Curva da perda e acurácia durante o treinamento do modelo de classificação.	60
Figura 34 – Gráfico de dispersão da severidade predita e real dos modelos de segmentação de sintomas (a) DeepLab V3+. (b) U-Net++.	62
Figura 35 – Gráfico de dispersão da severidade predita e real dos modelos de segmentação de lesões (a) DeepLab V3+. (b) U-Net++.	63
Figura 36 – Exemplo de aplicativo para dispositivo móvel baseado nos modelos desenvolvidos nesse projeto.	67

LISTA DE TABELAS

Tabela 1 – Divisão da base de classificação.	41
Tabela 2 – Performance obtida para cada função de perda utilizada no treinamento de uma U-Net.	51
Tabela 3 – Comparativo de performance dos modelos treinados para a PRO-POSTA A	55
Tabela 4 – Valores de acurácia e <i>IoU</i> por cada classe do segmentador.	55
Tabela 5 – Comparativo dos modelos treinados com as do estado da arte.	61
Tabela 6 – Comparativo dos resultados obtidos nos modelos de segmentação, com a U-Net++, nas duas propostas desenvolvidas.	62
Tabela 7 – Comparativo de número de parâmetros e tamanho estimado dos modelos desenvolvidos com a DeepLab V3+.	63

LISTA DE QUADROS

Quadro 1 – Principais diferenças entre as propostas A e B	40
Quadro 2 – Hiperparâmetros usados para definir a melhor função de perda para o modelo de segmentação de sintomas.	43
Quadro 3 – Hiperparâmetros usados para definir o melhor modelo de segmentação de sintomas.	44
Quadro 4 – Faixa de Hiperparâmetros testados no modelo de segmentação de sintomas.	46
Quadro 5 – Hiperparâmetros usados para avaliar a contribuição das técnicas de aumento de dados no modelo de classificação.	48
Quadro 6 – Faixa de Hiperparâmetros testados no classificador.	49
Quadro 7 – Resultados obtidos no conjunto de validação combinando a U-Net++ com diversos codificadores.	51
Quadro 8 – Resultados obtidos no conjunto de validação combinando a DeepLab V3+ com diversos codificadores.	52
Quadro 9 – Hiperparâmetros selecionados para o treinamento da rede de segmentação de sintomas.	53
Quadro 10 – Hiperparâmetros do treinamento da rede de classificação.	54

LISTA DE SIGLAS

ASPP	<i>Atrous Spatial Pyramid Pooling</i>
ConvNet	Rede Neural Convolucional
CRF	<i>Conditional Random Fields</i>
Faster RCNN	<i>Faster Region-based Convolutional Neural Network</i>
FCN	<i>Fully Convolutional Network</i>
FPN	<i>Feature Pyramid Network</i>
kNN	<i>k-Nearest Neighbors</i>
MRF	<i>Markov Random Field</i>
SVM	Máquinas de Vetores de Suporte

LISTA DE SÍMBOLOS

<i>Dice</i>	<i>Dice Coefficient</i>
<i>IoU</i>	<i>Intersection over Union</i>
<i>mIoU</i>	<i>Mean-IoU</i>
<i>mPA</i>	<i>Mean Pixel Accuracy</i>
<i>PA</i>	<i>Pixel Accuracy</i>

SUMÁRIO

1	INTRODUÇÃO	15
1.1	OBJETIVOS	16
1.1.1	Objetivos Específicos	16
1.2	ESTRUTURA DO TEXTO	16
2	REFERENCIAL TEÓRICO	18
2.1	CAFÉ	18
2.1.1	Contexto Capixaba	18
2.1.2	Pragas e Doenças	19
2.2	APRENDIZADO PROFUNDO	22
2.2.1	Segmentação Semântica	23
2.2.2	Funções de Perda	27
2.2.2.1	Perda baseada em distribuição	28
2.2.2.2	Perda baseada na região	28
2.2.3	Crterios de Avaliao	29
2.3	TRABALHOS RELACIONADOS	30
3	MÉTODO	37
3.1	PROPOSTA	37
3.1.1	PROPOSTA A	38
3.1.2	PROPOSTA B	39
3.2	BASE DE DADOS	39
3.3	EXPERIMENTOS	42
3.3.1	PROPOSTA A – Modelo de segmentao de sintomas	42
3.3.1.1	Definiao da funao de perda	43
3.3.1.2	Definiao do modelo	43
3.3.1.3	Tcnicas de aumento de dados	44
3.3.1.4	Definiao dos hiperparmetros	44
3.3.1.5	Pr-treinamento do codificador	46
3.3.2	PROPOSTA B – Modelo de segmentao de lesoes	46
3.3.3	PROPOSTA B – Modelo de classificao	46
3.3.3.1	Tcnicas de aumento de dados	47
3.3.3.2	Ajuste fino	47

3.3.3.3	Definição dos hiperparâmetros	48
4	RESULTADOS	50
4.1	EXPERIMENTOS	50
4.1.1	PROPOSTA A – Modelo de segmentação de sintomas	50
4.1.1.1	Definição da função de perda	50
4.1.1.2	Definição do modelo	50
4.1.1.3	Técnicas de aumento de dados	52
4.1.1.4	Definição de hiperparâmetros	53
4.1.1.5	Pré-treinamento do codificador	53
4.1.2	PROPOSTA B – Modelo de classificação	53
4.2	RESULTADOS	54
4.2.1	PROPOSTA A	54
4.2.2	PROPOSTA B	58
4.2.3	Comparando as propostas	61
5	CONCLUSÕES E TRABALHOS FUTUROS	65
5.1	CONCLUSÕES GERAIS	65
5.2	TRABALHOS FUTUROS	66
	REFERÊNCIAS	68

1 INTRODUÇÃO

O café é uma das produções agrícolas mais importantes do mundo, com um valor líquido de US\$ 200 bilhões por ano e produção em cerca de 80 países. O Brasil é o maior produtor de café do mundo, sendo responsável por aproximadamente um terço da produção global. Nesse cenário, o estado do Espírito Santo é o segundo maior produtor de café do país. Sua produção expressiva composta por variedades de café arábica e conilon o torna responsável por mais de 30,00% da produção brasileira (INCAPER, 2019).

A cafeicultura está em todas as regiões do estado de maneira bem diversificada, com tamanhos de propriedades diversos, diferentes espécies e altitudes, além do nível tecnológico implantado. Apesar da diversificação, existem uma série de fatores que podem afetar a produção de café em todo o estado, como pragas e doenças que acometem as folhas da planta, reduzindo a capacidade fotossintética e a qualidade final do produto. Por esse motivo, o controle desses fatores é essencial para garantir a produção e a qualidade do café, bem como para evitar danos econômicos e sociais para os produtores.

A detecção precoce dessas ameaças é essencial para controlar sua propagação (ESGARIO; KROHLING; VENTURA, 2020). Atualmente, a maior parte da detecção nas propriedades rurais é feita visualmente pelos próprios produtores e especialistas, sendo que a inspeção visual é um processo demorado e propenso a erros, visto que um único erro na identificação pode ter um impacto significativo na colheita do café. Por conta disso, nos últimos anos, técnicas de visão computacional e aprendizado de máquinas têm sido amplamente utilizadas para detectar e classificar doenças e pragas de plantas cultivadas, incluindo o café.

Técnicas de aprendizado de máquina conseguem detectar pragas em folhas de café com precisão (ESGARIO; KROHLING; VENTURA, 2020). Alguns estudos vêm sendo realizados com o objetivo de desenvolver modelos de detecção automatizada de pragas em folhas de café utilizando sistemas de aprendizado de máquina. Um exemplo disso é o estudo realizado por Tassis, Souza e Krohling (2021), que usou aprendizado profundo para detectar sinais de infestação em folhas de café através de análise de imagens.

Para isso, os pesquisadores compararam várias arquiteturas de rede e conjuntos de dados para determinar qual delas produzia os melhores resultados.

De maneira geral, a aplicação de técnicas de *machine learning* mostra-se extremamente útil na detecção de pragas e doenças nas folhas de café, e vem sendo cada vez mais utilizada por produtores e pesquisadores. Essa tecnologia não só ajuda a diminuir o tempo de detecção dessas ameaças, mas também auxilia na identificação correta e no controle adequado de infestações, contribuindo para a garantia da produção sustentável de café a nível global.

1.1 OBJETIVOS

Este trabalho tem como objetivo desenvolver um sistema capaz de identificar e quantificar a severidade de quatro enfermidades distintas que afetam as folhas de café.

1.1.1 Objetivos Específicos

- Definir a base de dados que será utilizada no trabalho.
- Desenvolver um pipeline com um único modelo de segmentação semântica para segmentar os sintomas em folhas de café.
- Desenvolver um pipeline com um modelo de segmentação semântica e outro de classificação para segmentar os sintomas em folhas de café.

1.2 ESTRUTURA DO TEXTO

A parte escrita deste trabalho de dissertação está dividida em cinco capítulos, descritos a seguir:

- **Introdução:** capítulo inicial que tem como objetivo contextualizar o trabalho desenvolvido. Além disso, apresenta brevemente o objetivo do projeto.
- **Referencial Teórico:** neste capítulo será apresentado toda a fundamentação teórica necessário para o desenvolvimento e compreensão dos resultados obtidos.

- **Metodologia:** neste capítulo será detalhada a implementação realizada, compreendendo todos os níveis de implementação e experimentos realizados.
- **Resultados:** após a apresentação da proposta desenvolvida, serão apresentados os resultados dos experimentos realizados para validar o que foi proposto.
- **Conclusões e Trabalhos Futuros:** esse capítulo apresenta uma breve discussão geral do trabalho desenvolvido, bem como uma lista de trabalhos futuros.

2 REFERENCIAL TEÓRICO

2.1 CAFÉ

O café destaca-se como um dos produtos agrícolas mais importantes nos aspectos sociais e econômicos para mais de 60 países produtores, ocupando a segunda colocação na geração de riqueza após o petróleo (MISHRA; SLATER, 2012). No Brasil, desde a sua introdução em 1727, o café tornou-se um dos produtos mais relevantes do país na geração de empregos e produção de riquezas (FERRÃO *et al.*, 2016).

Das 124 espécies de café catalogadas, o *Coffea arabica*, conhecido como arábica, e o *Coffea canephora*, cujo as principais variedades são o conilon e a robusta, compõem quase todo o café consumido no mundo (DAVIS *et al.*, 2011). As duas espécies distinguem-se em aspectos agrônômicos, bioquímicos, sensoriais, de mercado e utilização; contudo, independente desse tópico, verifica-se uma demanda cada vez maior por cafés de qualidade superior tanto em países produtores quanto nos consumidores (FERRÃO *et al.*, 2016).

O *Coffea arabica* possui uma bebida mais suave com aroma e sabor mais acentuado. É comercializado puro ou em mistura com o conilon ou robusta. Em contrapartida, o *Coffea canephora* caracteriza-se por uma bebida mais neutra, amargor e maior teor de cafeína, sendo comumente utilizados em misturas e cafés solúveis, sendo o tipo robusta predominantemente cultivado na África e Ásia, enquanto que no Brasil, predomina-se o conilon (FERRÃO *et al.*, 2016).

2.1.1 Contexto Capixaba

O Espírito Santo destaca-se como o maior produtor de café conilon do país, respondendo por cerca de 70,00% da produção nacional e 20,00% da produção mundial de café robusta. Tal cultura representa a principal fonte de renda em 80,00% das propriedades rurais localizadas em regiões quentes do estado, sendo responsável por contribuir com 37,00% do PIB Agrícola. O café conilon ocupa 283 mil hectares, distribuídos em 40 mil propriedades rurais em 63 municípios capixabas, sustentando aproximadamente 78 mil famílias produtoras e gerando mais de 250 mil empregos

diretos e indiretos (INCAPER, 2020).

A cafeicultura do conilon é um importante setor para o estado, com destaque em âmbito nacional e mundial por ter atingido uma produtividade média de 45,94 sacas por hectare (sc/ha), sendo que muitos produtores tecnificados conseguem colher mais de 180 sc/ha (INCAPER, 2020).

Além disso, nas últimas décadas, a cafeicultura do conilon passou por um grande processo de transformação que, aliado a um perfil empreendedor dos cafeicultores, permitiu que desenvolvesse uma cafeicultura altamente competitiva, especialmente na região norte do estado (FERRÃO *et al.*, 2016).

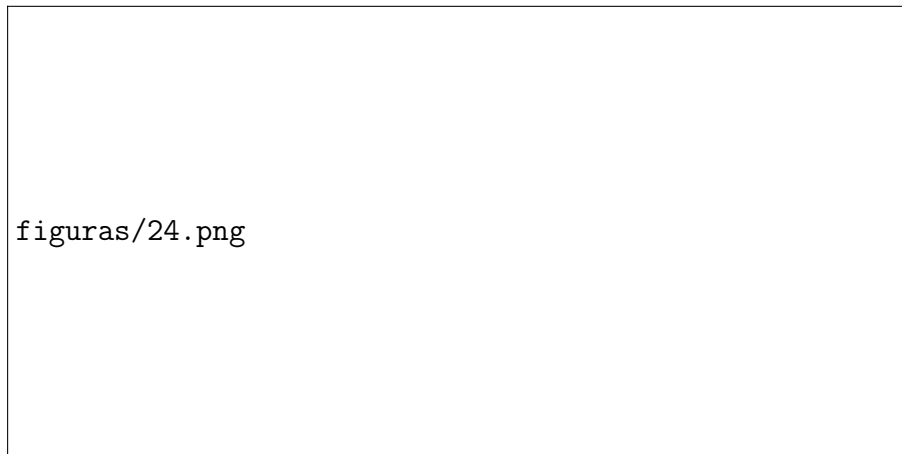
2.1.2 Pragas e Doenças

O café é uma das principais *commodities* do mundo e a sua produção pode ser afetada por diversas doenças e pragas. A identificação rápida de uma doença ou praga em uma plantação de café permite que medidas de controle sejam tomadas de maneira imediata e eficiente, minimizando os prejuízos na colheita e na rentabilidade da lavoura, sendo assim fundamental para garantir a produtividade e a qualidade dos produtos agrícolas. Entre as possíveis enfermidades que prejudicam o café capixaba, destacam-se o bicho-mineiro, ferrugem, cercospora e a phoma.

O bicho-mineiro (*Leucoptera coffeella*) é uma das principais pragas do café, contribuindo para a redução da produtividade e qualidade dos frutos. De acordo com Ferrão *et al.* (2012), essa praga alimenta-se do parênquima das folhas, causando a formação de galerias que comprometem a capacidade fotossintética da planta. Para o controle dessa praga, faz-se necessário que reconheça a infestação e recomenda-se a mínima utilização de inseticidas de forma localizada. A Figura 1 expõe a aparência de uma folha com lesões causadas por essa praga.

A ferrugem do café (*Hemileia vastatrix*) é uma das principais doenças que afetam os cafezais, podendo ocorrer tanto em viveiros como em lavouras e podem causar perdas de até 50% da produção, afetando também a qualidade do café, e gerando custos e impactos ambientais com a utilização de fungicidas. Os sintomas dessa doença

Figura 1 – Exemplo de folha com bicho-mineiro.



Fonte: Elaborado pelo autor (2023).

apresentam-se como manchas amarelas, conforme ilustrado na Figura 2 (FERRÃO *et al.*, 2012).

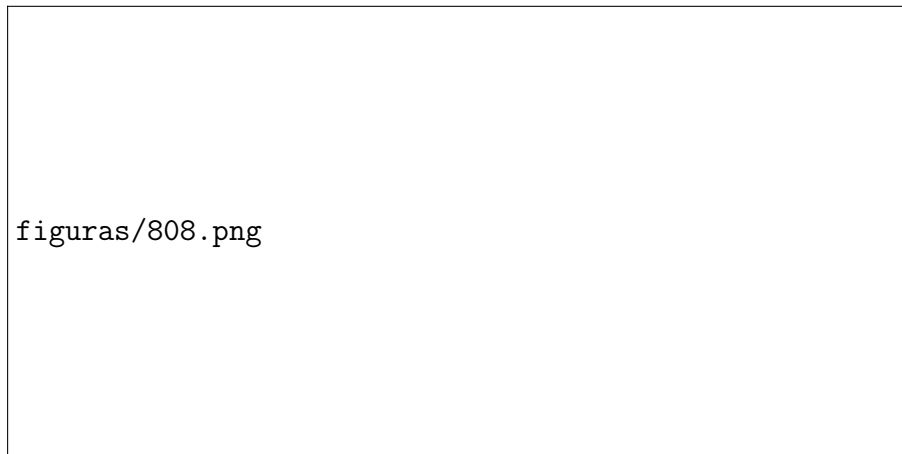
Figura 2 – Exemplo de folha com ferrugem.



Fonte: Elaborado pelo autor (2023).

A mancha de olho pardo ou cercospora (*Cercospora coffeicola*) é uma doença causada por fungos que afetam as folhas do café, podendo reduzir significativamente a produtividade da lavoura. Segundo Ferrão *et al.* (2012), essa doença manifesta-se por meio de manchas circulares, escuras e de formas irregulares nas folhas, que posteriormente evoluem para lesões necróticas, exibidas na Figura 3. O controle da cercospora pode ser feito por meio da aplicação de fungicidas e medidas culturais, como a poda de ramos infestados.

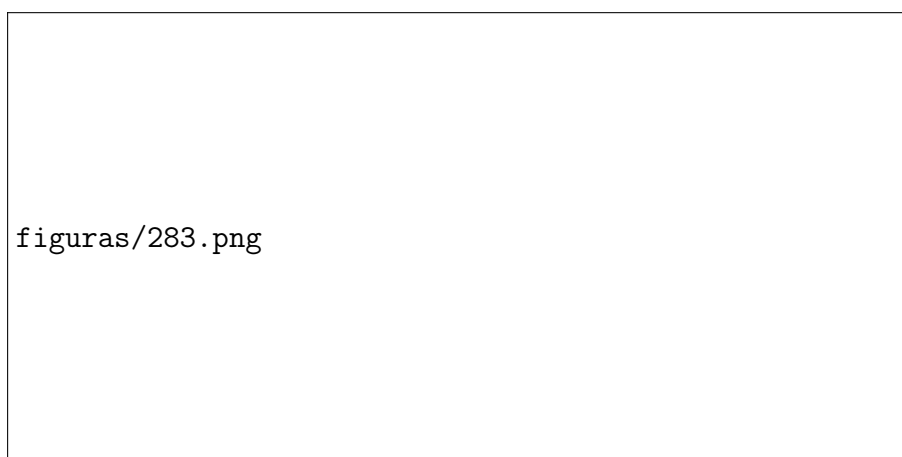
Figura 3 – Exemplo de folha com cercospora.



Fonte: Elaborado pelo autor (2023).

A phoma (*Phoma tarda*), Figura 4, é uma das doenças fúngicas mais comuns em cafezais, podendo causar desde a queda prematura de folhas até a morte da planta. Essa doença manifesta-se por meio de lesões necróticas circulares nas folhas, que posteriormente evoluem para manchas marrons escuras e estendem-se para o caule e ramos. O controle da phoma pode ser feito evitando instalar as lavouras em áreas onde ocorrem ventos fortes e frios. O uso de proteção química, isoladamente, geralmente não apresenta resultados satisfatórios.

Figura 4 – Exemplo de folha com phoma.



Fonte: Elaborado pelo autor (2023).

2.2 APRENDIZADO PROFUNDO

De acordo com LeCun, Bengio e Hinton (2015), os sistemas de aprendizado de máquina (*Machine Learning*) são amplamente utilizados em aplicações como reconhecimento de objetos em imagens, tradução de textos, reconhecimento de voz e fala, determinação de notícias, propaganda ou produtos de interesse do usuário, entre outras. No entanto, as técnicas tradicionais são limitadas na capacidade de processamento de dados brutos, exigindo engenharia de pré-processamento complexa para representação adequada dos dados para os modelos clássicos.

Por sua vez, o aprendizado de representação engloba métodos que permitem à máquina criar representações necessárias para detectar ou classificar tarefas com base em dados brutos. Entre as técnicas de aprendizado de representação, o aprendizado profundo se destaca pelo uso de múltiplos níveis de representação que transformam dados brutos em complexidades abstratas (LECUN; BENGIO; HINTON, 2015).

Em 1998, Lecun *et al.* (1998) apresentaram a LeNet-5, a primeira Rede Neural Convolutiva (ConvNet), para o reconhecimento de dígitos. Apesar da capacidade de aprender com os dados bruto, as ConvNets foram utilizadas somente 14 anos depois para resolver problemas da complexidade do desafio ImageNet (DENG *et al.*, 2009) com a rede AlexNet, desenvolvida por Krizhevsky, Sutskever e Hinton (2012). Desde então, abordagens em aprendizado profundo alcançaram excelentes resultados em tarefas de reconhecimento de imagens, de fala, de compreensão da linguagem natural e até mesmo em jogos.

Segundo Mitchell (1997, p. 2), em geral, os modelos de aprendizado de máquina são algoritmos que aprendem através de dados. "um programa de computador aprende a partir de uma experiência (E) com relação a alguma classe de tarefa (T) e a medida de desempenho (P) se seu desempenho na tarefa em T , medido por P , melhora com a experiência E ". Por exemplo, na tarefa de filtrar SPAMs, a tarefa T seria categorizar mensagens de e-mail como SPAM ou não, a medida de desempenho P seria a porcentagem de mensagens corretamente classificadas e a experiência E seria o conjunto de e-mails rotulados por seres humanos.

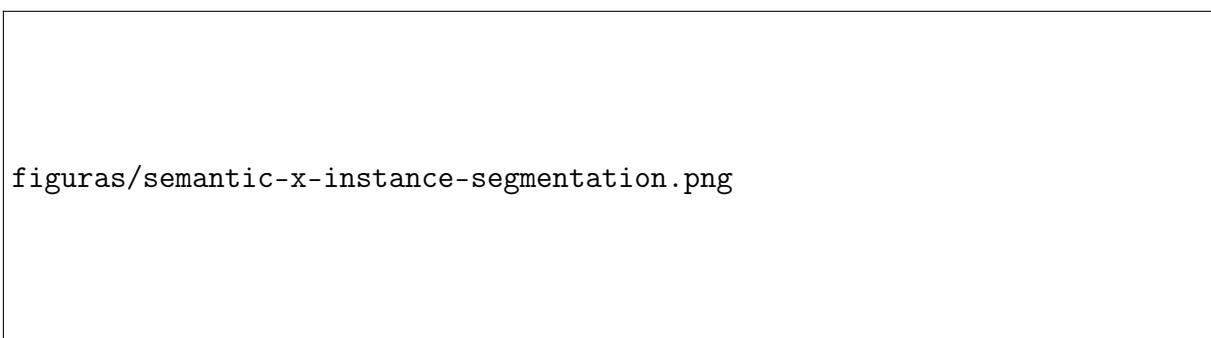
A tarefa T é geralmente descrita em termos de como o sistema deve processar os dados. Algumas das tarefas mais comuns de ML são classificação, regressão e agrupamento. A medida de desempenho P é uma medida quantitativa para avaliar como o sistema realiza a tarefa T . A experiência E é categorizada como aprendizado supervisionado ou não supervisionado.

Com base nos métodos de aprendizagem e resultados obtidos por abordagens profundas, as próximas subseções detalham os modelos de segmentação semântica e suas respectivas métricas de performance.

2.2.1 Segmentação Semântica

A segmentação de imagens pode ser formulada como um problema de classificação para cada *pixel* com rótulos semânticos (segmentação semântica) ou como parte de objetos individuais (segmentação de instâncias). A segmentação semântica realiza a classificação em nível de *pixel* com um conjunto de categorias de objetos (por exemplo, humano, carro, árvore, céu, etc.) para todos os *pixels* da imagem, diferente de uma tarefa de classificação que retorna apenas um rótulo para toda a imagem. A segmentação de instância estende ainda mais o escopo, detectando e delineando cada objeto de interesse na imagem (por exemplo, identificando cada pessoa individualmente) (MINAEE *et al.*, 2020). A Figura 5 ilustra a diferença desses dois métodos de segmentação.

Figura 5 – Um exemplo dos diferentes tipos de segmentação. (a) Imagem de entrada. (b) resultado da segmentação semântica. (c) resultado da segmentação de instâncias.

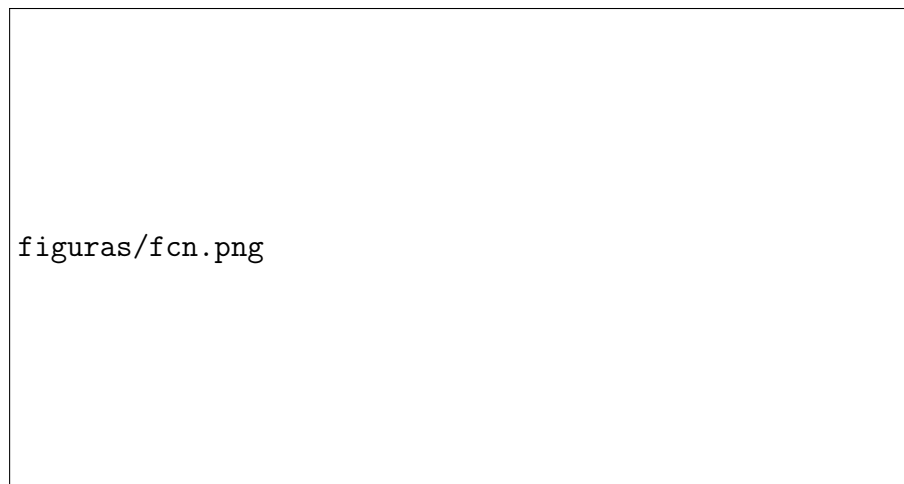


Fonte: Adaptado de Sultana, Sufian e Dutta (2020).

Após o sucesso da AlexNet em 2012, diversos modelos de segmentação semântica

baseados em redes convolucionais foram propostos. Long, Shelhamer e Darrell (2015) propuseram um dos primeiros trabalhos que aplicaram aprendizado profundo para a tarefa de segmentação, removendo as camadas totalmente conectadas e criando uma rede somente com camadas convolucionais, chamada de *Fully Convolutional Network* (FCN), mostrada na Figura 6. Essa modificação permite que a rede receba uma imagem de tamanho arbitrário e produza um mapa de segmentação do mesmo tamanho. Apesar de sua popularidade, a FCN possui algumas limitações quanto ao tempo de inferência e o fato que ela não leva em consideração o contexto global das informações (MINAEE *et al.*, 2020).

Figura 6 – FCNs podem aprender com eficiência a fazer previsões densas para tarefas por *pixel*, como segmentação semântica.

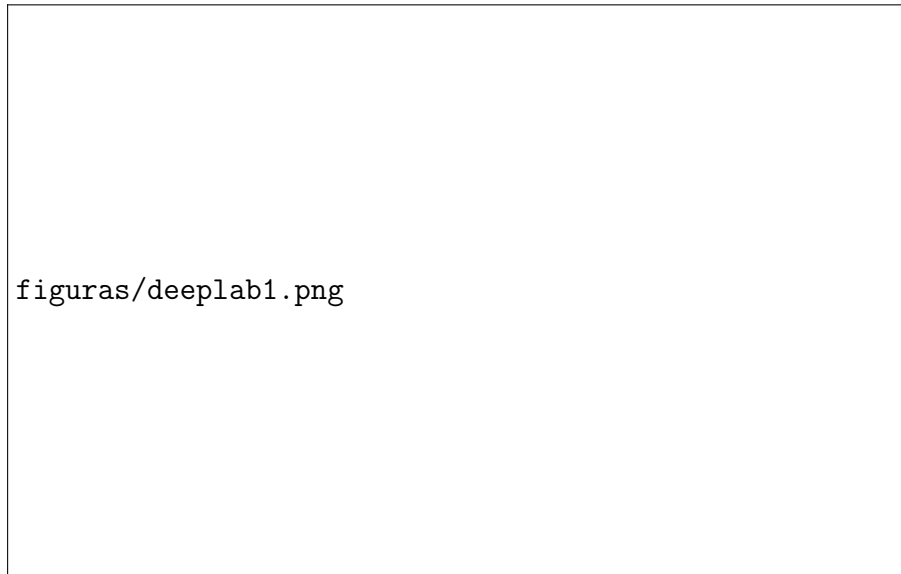


Fonte: Retirado de Long, Shelhamer e Darrell (2015).

Para aprimorar a FCN, diversas abordagens foram propostas com a finalidade de integrar mais contexto ao modelo, tais como incorporar *Conditional Random Fields* (CRFs) e *Markov Random Field* (MRFs) nessas arquiteturas. Chen *et al.* (2016) propuseram uma combinação de redes convolucionais e CRFs, mostrando que as respostas da última camada das ConvNets não eram suficientes para localizar os objetos de segmentação com precisão e, que, ao combinar as respostas dessa camada com CRF, o modelo é capaz de detectar limites dos objetos com melhor precisão. Esse comportamento é melhor demonstrado na Figura 7.

Outra família de modelos para segmentação baseia-se em uma estrutura de Codificador-Decodificador. Noh, Hong e Han (2015) propuseram uma arquitetura com deconvo-

Figura 7 – Resultados da DeepLab. Para cada linha, têm-se a imagem de entrada, o resultado do modelo sem CRF e o resultado da segmentação com CRF.



Fonte: Adaptado de Chen *et al.* (2016).

luções (*i.e.* convoluções transpostas), consistindo em duas partes: um codificador baseado na VGG16 (SIMONYAN; ZISSERMAN, 2014) e uma rede deconvolucional que recebe o mapa de características do codificador e gera uma mapa de probabilidades de classes por *pixel*. A rede deconvolucional é composta por camadas de *unpooling* e deconvoluções, conforme representado na Figura 8.

Figura 8 – Segmentação semântica deconvolucional.

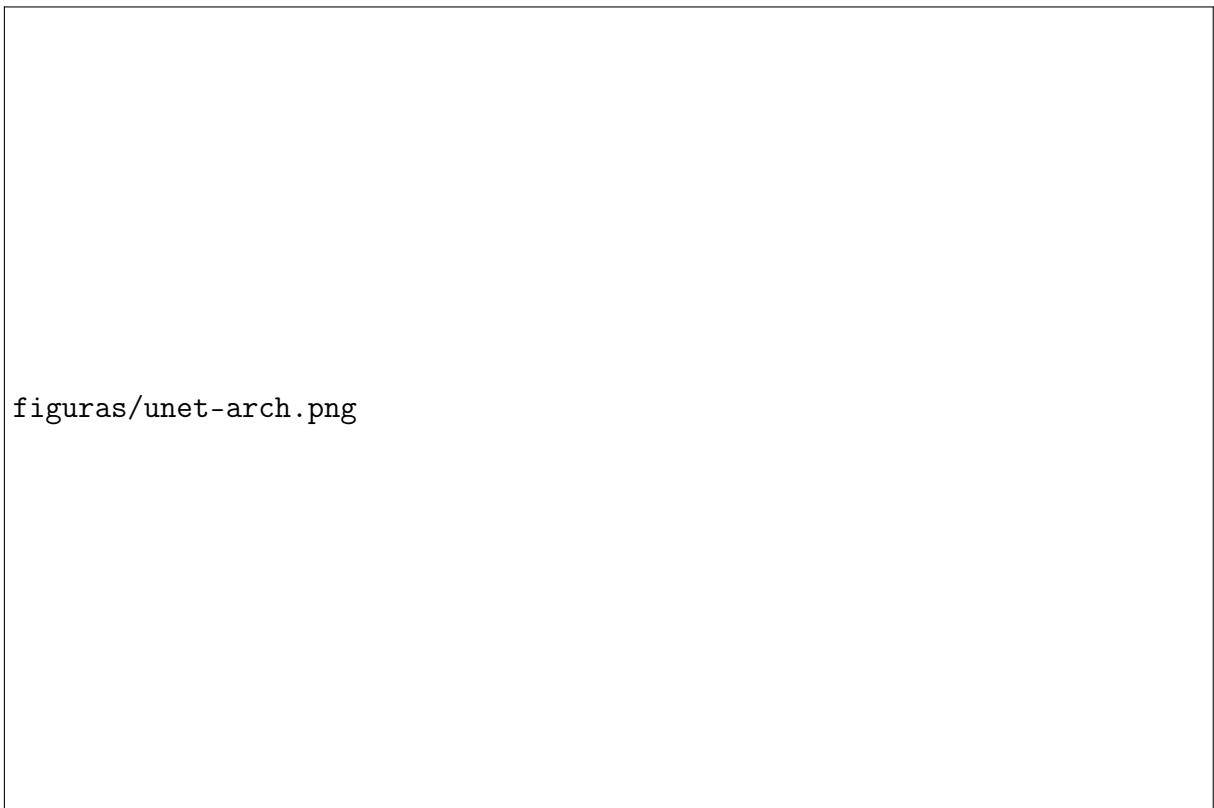


Fonte: Retirado de Noh, Hong e Han (2015).

Ronneberger, Fischer e Brox (2015) propuseram a U-Net para a segmentação de imagens biomédicas. Ele também possui uma arquitetura Codificador-Decodificador que incorpora camadas de *max-pooling* para criar uma representação compacta da imagem. A rede é então expandida em uma arquitetura de "U" para recuperar a resolução da

imagem original, utilizando técnicas de concatenação dos mapas de características. O modelo também utiliza uma função de perda específica para segmentação, o Coeficiente *Dice*, a fim de melhorar a precisão da segmentação. Sua arquitetura em "U" é representada na Figura 9.

Figura 9 – Arquitetura U-net.



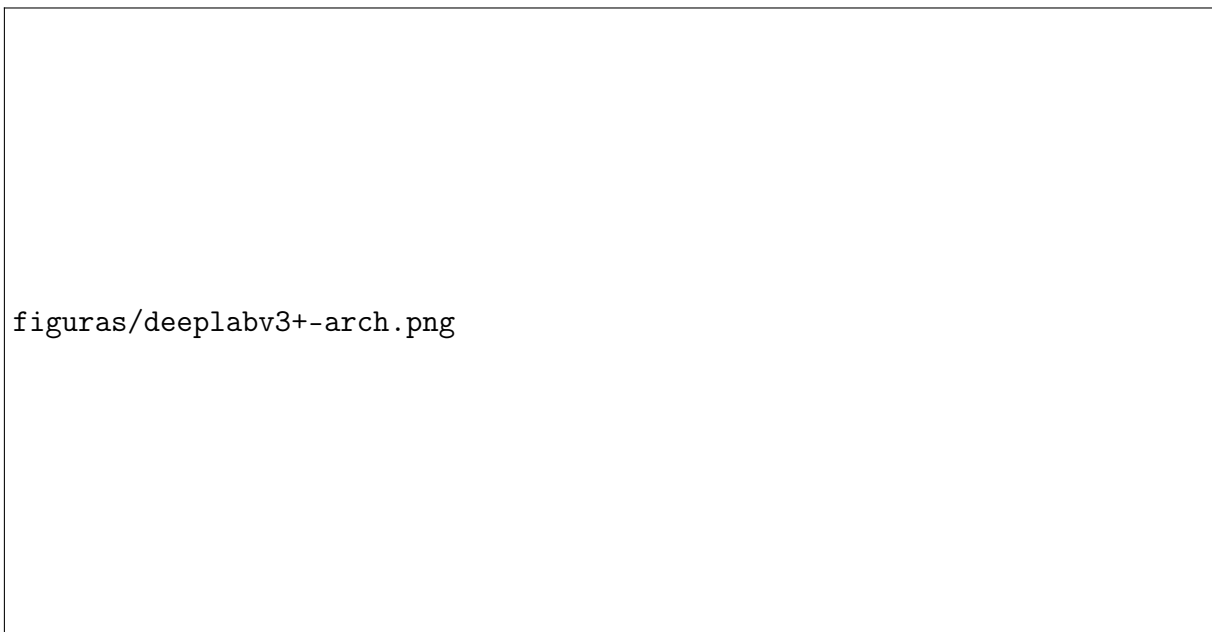
Fonte: Retirado de Ronneberger, Fischer e Brox (2015).

Zhou *et al.* (2018) propuseram a U-Net++, inspirados na ideia de blocos densos da DenseNet (HUANG *et al.*, 2016). Essa abordagem empregou saltos entre camadas convolucionais que ajudaram no fluxo do gradiente e preencheram a lacuna semântica entre os mapas de características do codificador. Comparado com a U-Net original, a U-Net++ obteve, em diferentes bases de dados, um aumento entre 1,86 (*cell nuclei dataset*) e 4,97 (*lung nodule dataset*) pontos de *IoU*.

O modelo DeepLab (CHEN *et al.*, 2016) é uma arquitetura de rede neural convolucional usada para segmentação semântica de imagens. A versão inicial do DeepLab utilizava uma estrutura de múltiplas escalas para capturar informações contextuais e reduzir problemas de sobreposição na segmentação. A seguir, o DeepLab V2 (CHEN *et al.*,

2017a) adicionou derivações de convoluções dilatadas para melhorar a resolução espacial e uma fusão de canal para incorporar informações de recurso de diferentes caminhos. O DeepLab V3 (CHEN *et al.*, 2017b) implementou o modelo de *Atrous Spatial Pyramid Pooling* (ASPP) para capturar informações em múltiplas escalas e utilizou *batch normalization* em toda a rede. A última versão, a DeepLab V3+ (CHEN *et al.*, 2018), estende o DeepLab V3 empregando uma estrutura de Codificador-Decodificador, exposta na Figura 10. O módulo codificador codifica informações contextuais em multiescalas, enquanto o módulo decodificador é mais simples, mas eficaz, refina os resultados da segmentação ao longo das bordas do objeto.

Figura 10 – Arquitetura da DeepLab V3+.



Fonte: Retirado de Chen *et al.* (2018).

2.2.2 Funções de Perda

Os algoritmos de Aprendizado Profundo usam abordagens com gradiente descendente para otimizar e aprender uma determinada tarefa. Para um processo de aprendizagem eficaz, faz-se necessário uma representação matemática do objetivo que visam medir a dissimilaridade entre a verdade básica e a prevista, conhecido como funções de perda ou custo, pois elas instigam o processo de aprendizagem do modelo (JADON, 2020). De acordo com Ma *et al.* (2021), podemos classificar essas funções nos seguintes conjuntos:

2.2.2.1 Perda baseada em distribuição

As funções de perda baseadas em distribuição visa minimizar a dissimilaridade entre duas distribuições. A função mais fundamental nesta categoria é a *Cross-Entropy*, do português entropia cruzada, sendo todas as outras derivadas dessa.

Cross-Entropy Loss – mede a dissimilaridade entre duas distribuições de probabilidade. Em termos simples, ela mede o quanto as previsões do modelo se desviam da distribuição real dos dados.

Weighted Cross-Entropy Loss – é uma extensão do método acima, em que pesos são atribuídos a cada classe. Muitas vezes usado para lidar com dados desbalanceados. Geralmente, o peso de cada classe é inversamente proporcional à frequência, penalizando os objetos mais frequentes.

2.2.2.2 Perda baseada na região

As funções de perda baseadas em regiões visam minimizar a incompatibilidade ou maximizar as regiões de sobreposição entre a verdade básica e a segmentação predita.

Dice Loss – é uma função de perda que calcula a sobreposição entre a máscara prevista e a máscara de referência e, em seguida, retorna o valor 1 menos o coeficiente *Dice*. Essa função de perda é calculada como a diferença entre a máscara de saída e a máscara de referência, dividida pela soma das duas máscaras.

Focal Loss – foi desenvolvida para lidar com o problema de desequilíbrio de classes na segmentação de imagens médicas. Essa função de perda atribui pesos elevados para amostras difíceis (ou seja, amostras que são frequentemente rotuladas erroneamente) e pesos baixos (ou nenhum peso) para amostras fáceis. A *Focal Loss* é definida como uma ponderação escalonada do erro logarítmico da probabilidade de previsão.

Jaccard Loss – é semelhante à *Dice Loss* em que calcula a sobreposição entre a máscara de referência e a máscara prevista e, em seguida, retorna o valor 1 menos o coeficiente Jaccard. Entretanto, a *Jaccard Loss* difere-se da *Dice Loss* por levar em conta o espaço entre objetos segmentados, o que ajuda a evitar o erro de segmentar

objetos que estão contidos em outro objeto segmentado.

2.2.3 Critérios de Avaliação

Idealmente, um modelo deveria ser avaliado em diversos aspectos, tais como precisão, tempo de inferência, memória requerida e outros. Medições como velocidade podem ser complicadas, pois dependem do dispositivos que estão executando o algoritmo, mas deve ser levado em consideração quando pretende-se embarcar a aplicação em dispositivos móveis ou similares. Além disso, a maioria dos trabalhos científicos usam métricas quantitativas para avaliar a precisão do modelo desenvolvido (MINAEE *et al.*, 2020). Entre as principais métricas para modelos de segmentação semântica, destacam-se:

Pixel Accuracy (PA) – no português, acurácia dos *pixels*, é simplesmente a proporção de *pixels* classificados corretamente, dividida pelo número total de *pixels*. Para $K + 1$ classes (K categorias que se deseja segmentar, mais o fundo), a Acurácia dos *Pixels* é definida pela equação 2.1.

$$PA = \frac{\sum_{i=0}^K p_{ii}}{\sum_{i=0}^K \sum_{j=0}^K p_{ij}} \quad (2.1)$$

onde p_{ij} é o número de *pixels* da classe i predita como se pertencesse a classe j .

Mean Pixel Accuracy (mPA) – é a versão estendida da métrica acima, em que a relação de *pixels* classificados corretamente é computado por classe e então é feito uma média sobre o número total de classes, conforme equação abaixo:

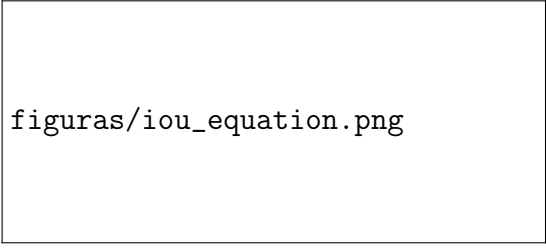
$$mPA = \frac{1}{K + 1} \sum_{i=0}^K \frac{p_{ii}}{\sum_{j=0}^K p_{ij}} \quad (2.2)$$

Intersection over Union (IoU) – também chamado de *Jaccard Index*, é um das métricas mais utilizadas para modelos de segmentação semântica. Ela é definida como a área de interseção entre o mapa de segmentação previsto e o real (*ground truth*), dividido pela área de união entre o mapa de segmentação previsto e o real.

A equação 2.3 indica como calcular essa métrica, sendo que A e B denotam os mapas previstos e reais, respectivamente. A Figura 11 ajuda a ilustrar essa equação.

$$IoU = J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (2.3)$$

Figura 11 – Representação da equação de IoU.



figuras/iou_equation.png

Fonte: Adaptado de de Rosebrock (2016).

Mean-IoU ($mIoU$) – é computada realizando a média da IoU entre todas as classes e é amplamente utilizado para relatar o desempenho de algoritmos de segmentação modernos.

Dice Coefficient ($Dice$) – pode ser definido como o dobro da sobreposição de área entre os mapas preditos e reais, dividida pelo número total de *pixels* em ambas as imagens. É descrita pela equação 2.4.

$$Dice = \frac{2|A \cap B|}{|A| + |B|} \quad (2.4)$$

2.3 TRABALHOS RELACIONADOS

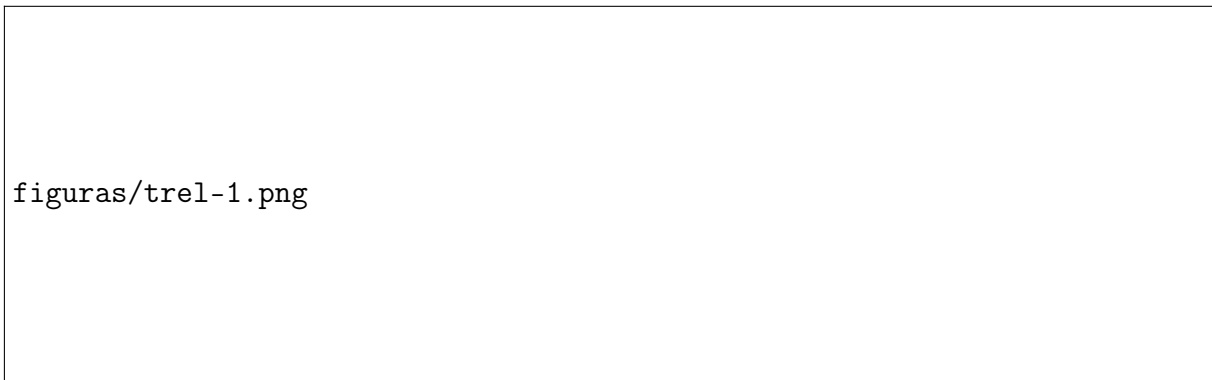
Atualmente, diversos trabalhos buscam incorporar novas formas de inteligência na busca contínua de aprimorar os processos agrícolas. No contexto do café brasileiro, muitos desses trabalhos investigam métodos para auxiliar o produtor rural a aumentar a produtividade e qualidade de seus produtos. Em particular, o café é um dos poucos produtos agrícolas, cujo valor comercial é vinculado à qualidade (SILVA; MORELI; FILHO, 2015), sendo essa qualidade dependente da espécie, composição química dos grãos crus, da cultivar, clima, altitude, forma da colheita, processamento pós-colheita e

das técnicas utilizadas no processamento industrial (CORRÊA; OLIVEIRA; BOTELHO, 2015).

Por essa razão, Ramos *et al.* (2017) construíram um Sistema de Visão de Máquina capaz de contar e identificar frutos entre colhíveis ou não colhíveis em imagens de ramos de café. Este sistema utiliza *smartphones* como meio de aquisição de imagens e processamento do algoritmo, sem exigir controle das condições de ambiente, como iluminação. Dessa forma, a proposta do trabalho é de desenvolver um método de baixo custo, eficiente e não-destrutivo que pudesse contabilizar os frutos e estimar seus respectivos percentuais de maturação e peso. O sistema de visão de máquina desenvolvida possui cinco estágios principais: (1) aquisição e ajuste das imagens, (2) segmentação de regiões homogêneas, (3) detector de contornos, para identificar os arcos e analisar se eles pertencem a frutos de café; (4) ajuste e seletor de elipses e (5) detecção, classificação e contagem dos frutos. A Figura 12 ilustra os estágios descritos. Para o último estágio, foram treinados três classificadores: *k-Nearest Neighbors* (KNN) (ou método dos vizinhos mais próximos), *Naive Bayes* e Máquinas de Vetores de Suporte (*Support Vector Machine*, SVM), do qual o SVM foi selecionado por oferecer melhor generalização e resultados específicos de cada classe. Uma das grandes vantagens do método proposto é a detecção de frutos oclusos contudo, o tempo de processamento de uma imagem (um único ramo de café) é de cerca de 1 minuto. Entre outros problemas, encontrou-se um erro variando de 18% a 40% na contagem de frutos. Após análise, identificou-se que o sistema falhava mais quando havia um contraste inadequado entre o ramo e o fundo, frutos pequenos, ramos não centralizados ou poucos frutos na imagem.

Rodríguez *et al.* (2020) expandiram o objeto de estudo ao tentarem estimar a produção de café utilizando imagens de árvores e não apenas os ramos de café, conforme apresentado na Figura 13. Inicialmente os autores propuseram uma solução baseada em visão computacional para extrair as características do café através de técnicas de segmentação e transformações morfológicas. Posteriormente, optou-se por treinar uma *Faster Region-based Convolutional Neural Network* (Faster RCNN) em função dos resultados obtidos por modelos de aprendizagem profundo, aumentando o desempenho geral da solução. Esse trabalho diferencia-se por apresentar uma abordagem para

Figura 12 – Fluxograma do sistema proposto por Ramos *et al.* (2017).



Fonte: Adaptado de Ramos *et al.* (2017).

extrair os grãos de café na própria árvore, sem focar na qualidade, cor e maturação do produto.

Figura 13 – Exemplos de imagens usadas nos trabalhos de (a) Ramos *et al.* (2017) e (b) Rodríguez *et al.* (2020).



Fonte: Adaptado de Ramos *et al.* (2017) e Rodríguez *et al.* (2020).

Além de identificar o grau de maturidade dos frutos de café, outros estudos exploram

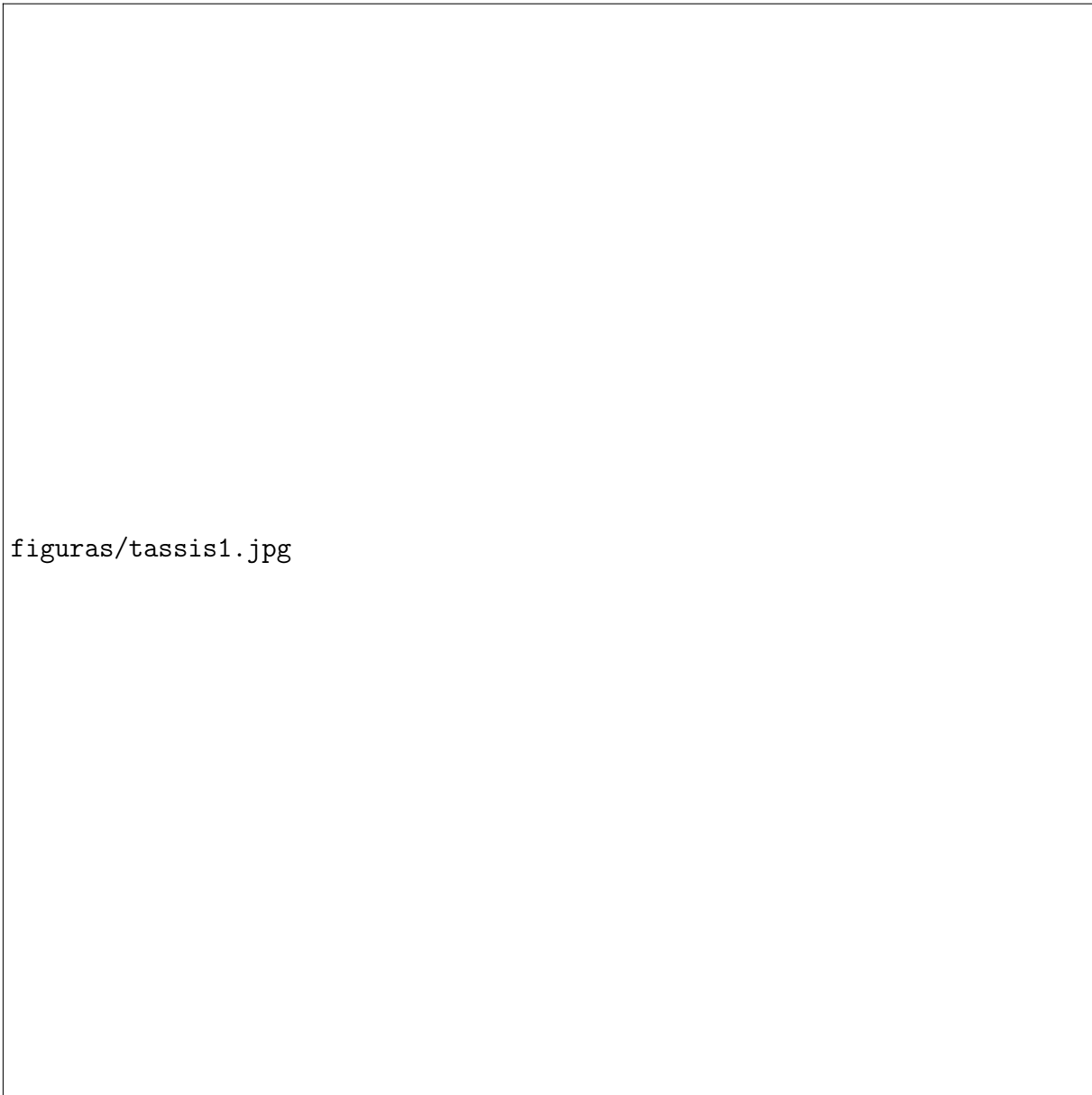
suas folhas, permitindo identificar pragas e doenças que atacam a plantação e também quantificar o dano causado por essas. Essa abordagem auxilia o agricultor a tomar decisões de controle de pesticidas e outros produtos químicos de forma assertiva, tornando todo o processo mais barato e eficaz.

Esgario, Krohling e Ventura (2020) apresentam um estudo que tem como objetivo desenvolver um método que possa classificar e estimar a severidade do estresse biótico das folhas de café usando técnicas de aprendizagem profunda, explorando o uso de técnicas de aumento de dados para tornar o sistema mais robusto e preciso. As principais contribuições desse estudo foram a criação de uma base de dados, denominada Bracol, com imagens de folhas de café saudáveis e doentes e um sistema para classificar e estimar a severidade de um estresse biótico que esteja afetando as folhas. Os autores selecionaram cinco arquiteturas (AlexNet, GoogleLeNet, VGG16, ResNet50 e MobileNetV2) pré-treinadas e aplicaram *transfer learning* para ajustar os pesos da rede neural para a nova tarefa. Dentre as opções selecionadas, a ResNet50 obteve os melhores resultados atingindo uma acurácia de 95,24% para a classificação dos sintomas e 86,51% para estimar a severidade da doença. Outra arquitetura que se destacou foi a MobileNetV2 com 94,84% e 84,52% de acurácia para sintomas e severidade, respectivamente. Salienta-se que a classe com pior acurácia foi a cercospora, que por sua vez é a que possui menos amostras na base de dados. Outra observação importante é que essa abordagem não consegue tratar folhas com mais de um sintoma.

Diferente do trabalho acima, Tassis, Souza e Krohling (2021) faz uso de modelos de segmentação para estimar a severidade das doenças com maior precisão. A proposta feita emprega a integração de um *framework* de três estágios, cada um com objetivo específico. O primeiro utiliza a Mask R-CNN (HE *et al.*, 2018) para uma tarefa de segmentação de instâncias, com o objetivo de isolar folhas individualmente em imagens na natureza. A segunda etapa usa uma U-Net ou PSPNet para segmentação semântica, cujo propósito é segmentar as regiões lesionadas e computar a porcentagem de ocupação delas na folha encontrada pela Mask R-CNN. Por fim, o último estágio classifica as regiões segmentadas entre as possíveis doenças com uma ResNet. As redes do primeiro e segundo estágio foram treinadas com as bases Bracot, desenvolvida

nesse trabalho, e a Bracol (ESGARIO; KROHLING; VENTURA, 2020), respectivamente. Para o segundo estágio, a U-Net e a PSPNet obtiveram 94,25% e 93,54% de *mIoU*, respectivamente. A Figura 14 apresenta o *pipeline* proposto, enquanto que a Figura 15 exhibe os resultados de cada arquitetura de segmentação semântica. Uma das grandes vantagens desse trabalho a possibilidade de identificar diversos sintomas em uma única folha devido ao modelo de segmentação, por exceção de sintomas sobrepostos. Isso ocorre porque o segmentador recortaria uma região contendo dois ou mais estresses bióticos e o classificador atribuiria a essa região apenas um rótulo.

Figura 14 – *Pipeline* proposto em (TASSIS; SOUZA; KROHLING, 2021).



figuras/tassis1.jpg

Fonte: Adaptado de Tassis, Souza e Krohling (2021).

Figura 15 – Exemplos dos resultados da segmentação semântica da U-Net e PSPNet.



Fonte: Adaptado de Tassis, Souza e Krohling (2021).

Em (ESGARIO *et al.*, 2022), um aplicativo *android* foi desenvolvido para identificar as doenças em imagens de folhas de café. Para tal, os autores desenvolveram um *framework* capaz de automaticamente integrar os processos de segmentação, recorte e classificação do sintomas. Por limitações práticas de memória e recursos computacionais em dispositivos móveis, os autores optaram por implementar o modelo desenvolvido em um servidor. Em contrapartida, o ponto negativo da solução é que o aplicativo deve estar conectado a internet. Nesse trabalho, os autores conseguiram aumentar a acurácia do modelo de classificação para 97,07% em relação à rede desenvolvida em (ESGARIO; KROHLING; VENTURA, 2020), utilizando uma ResNet50 pré-treinada. Para o estágio de segmentação, exploraram o uso da U-Net e PSPNet novamente, sendo que a U-Net apresentou a melhor performance obtendo um valor de *mIou* de 94,85% na base de testes da Bracol. O *pipeline* proposto é exibido na Figura 16, enquanto que a Figura 17 exhibe capturas de telas do aplicativo desenvolvido. Por fim, ressalta-se que essa abordagem possui a mesma carência que a de (TASSIS; SOUZA; KROHLING, 2021), isto é, ser incapaz de lidar com sintomas sobrepostos. Por essa razão, propõe-se a construção de um pipeline baseado em único modelo de

segmentação que seja apto a tratar esse cenário.

Figura 16 – *Pipeline* proposto em (ESGARIO *et al.*, 2022).

figuras/esgario1.jpg

Fonte: Adaptado de Esgario *et al.* (2022).

Figura 17 – Capturas de tela feitas durante o uso do aplicativo: câmera (A); verificação e *upload* (B); e resultados (C).

figuras/esgario3.jpg

Fonte: Retirado de Esgario *et al.* (2022).

3 MÉTODO

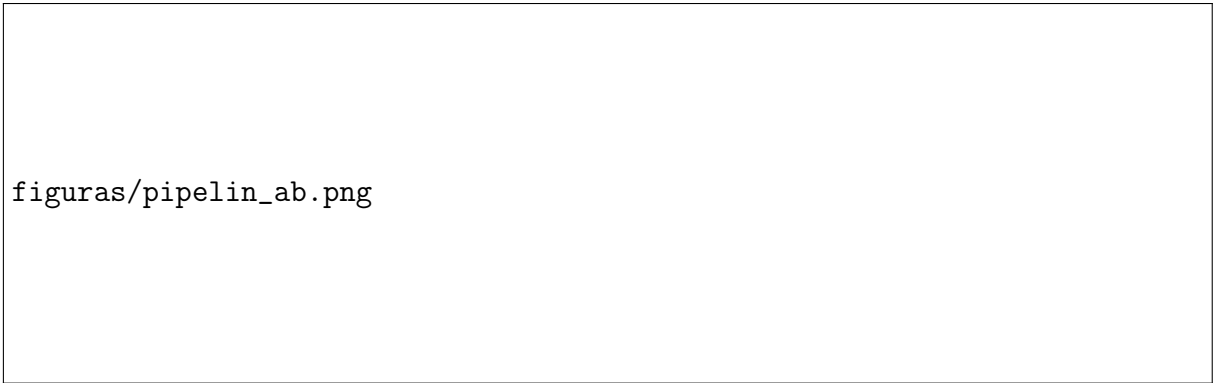
Neste capítulo, está descrita a proposta deste trabalho, seguido da base de dados utilizada e os experimentos realizados.

3.1 PROPOSTA

O objetivo deste trabalho é desenvolver uma arquitetura capaz de identificar estresses bióticos que afetam as folhas de café e determinar o grau de severidade desses, segundo as proporções de regiões saudáveis e lesionadas da folha. A partir das bases de dados disponíveis, as pragas e doenças a serem identificadas são: ferrugem, bicho-mineiro, phoma e cercospora.

Fundamentado por trabalhos semelhantes, tais como (ESGARIO *et al.*, 2022) e (TASSIS; SOUZA; KROHLING, 2021), percebeu-se que a utilização de modelos de segmentação semântica permite isolar com precisão as regiões lesionadas. Essa premissa viabiliza calcular a severidade de uma determinada doença a partir da sua porcentagem de ocupação da folha, isto é, o número total de *pixels* da classe doente sobre o número total de *pixels* que correspondem a folha (regiões saudáveis e doentes). Em vista disso, duas propostas foram concebidas: a primeira orientada para um sistema mais leve e rápido e a segunda para um sistema mais robusto. A Figura 18 exemplifica o *pipeline* de ambas as propostas.

Figura 18 – *Pipelines* dos *frameworks* propostos.



figuras/pipelin_ab.png

Fonte: Elaborado pelo autor (2023).

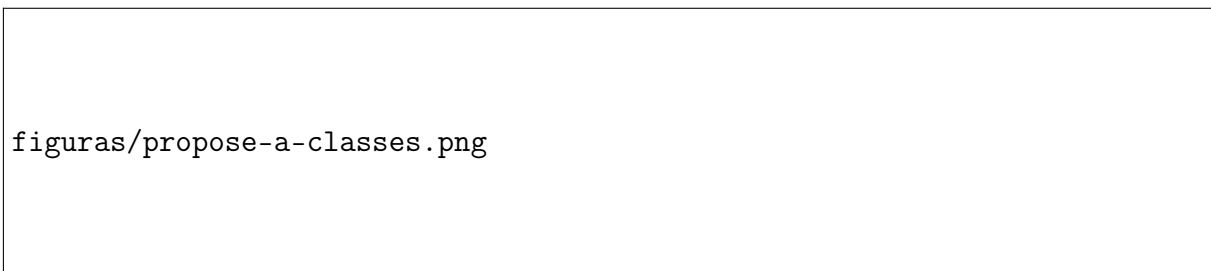
3.1.1 PROPOSTA A

A primeira proposta consiste em um *framework* baseado em uma única rede neural capaz de segmentar os estresses bióticos na folha, classificá-los e estimar suas respectivas severidades. Para tal, esse modelo foi treinado em uma base com seis classes - fundo, folha (saudável) e cada um dos sintomas supracitados, além do fundo da imagem, conforme exposto na Figura 19. Em termos gerais, esta abordagem enfatiza uma solução que requeira menos recursos computacionais, distinguindo-se dos demais trabalhos encontrados na literatura para esse problema.

Dessa maneira, em concordância ao *pipeline* da Figura 18, ao receber uma determinada imagem de entrada contendo uma folha, o modelo irá classificar todos os *pixels* da imagem como pertencentes a uma das classes treinadas, de forma que as lesões serão segmentadas por seus respectivos sintomas e, portanto, permitirá estimar a porcentagem de ocupação (severidade) de cada praga ou doença presente na folha. Destaca-se que essa abordagem permite isolar sintomas sobrepostos, diferente dos demais trabalhos realizados até então na área.

No exemplo contido na Figura 18, a imagem de entrada apresenta lesões causadas pela doença ferrugem e outras pela praga do bicho-mineiro. O segundo estágio representa a saída do modelo de segmentação em que a máscara predita possui regiões em vermelho, representando as lesões causadas pela ferrugem, e outras em amarelo, caracterizando o bicho-mineiro. Por último, é calculada a severidade de cada sintoma presente a partir da máscara predita no estágio anterior.

Figura 19 – Classes do modelo de segmentação da **PROPOSTA A**.



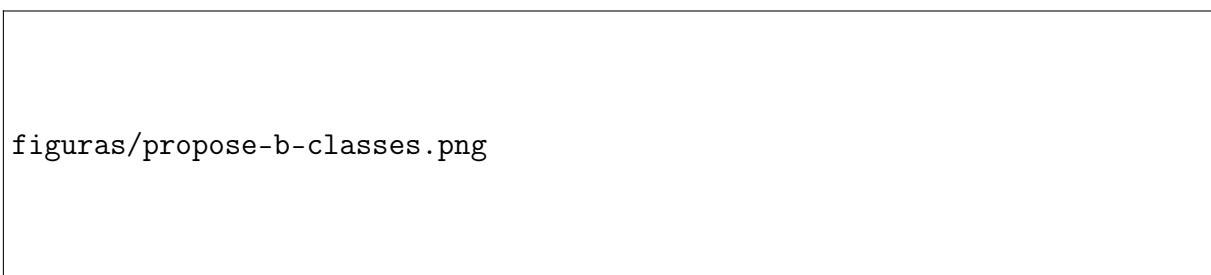
Fonte: Elaborado pelo autor (2023).

3.1.2 PROPOSTA B

Tal como (TASSIS; SOUZA; KROHLING, 2021) e (ESGARIO *et al.*, 2022), esta proposta emprega duas redes neurais: um modelo de segmentação semântica, responsável por isolar as regiões lesionadas, independente do estresse biótico, e outro modelo de classificação encarregado de identificar as pragas e doenças que afetaram as regiões segmentadas. Diferente da **PROPOSTA A**, esta abordagem é focada em obter resultados melhores nas métricas de avaliação descritas da Seção 2.2.3, independente dos recursos computacionais exigidos pelo sistema.

Para tanto, o modelo de segmentação semântica desta proposta trabalha somente com três classes: fundo, folha (saudável) e lesões, assim como exibido na Figura 20. O modelo de classificação foi treinado com as cinco classes empregadas na **PROPOSTA A**, por exceção do fundo, como pode ser visto na Figura 19. Dessa forma, ao receber uma imagem de entrada, o modelo de segmentação isola todas as regiões lesionadas, demarcadas pela cor laranja na Figura 18. Em seguida, essas regiões lesionadas são recortadas e enviadas para o classificador, que identifica seus respectivos sintomas. Baseado nas predições dos dois modelos, o sistema é capaz de quantificar a severidade por cada sintoma encontrado na folha. Dito isso, as principais diferenças entre as abordagens propostas são destacadas no Quadro 1.

Figura 20 – Classes do modelo de segmentação da **PROPOSTA B**.



Fonte: Elaborado pelo autor (2023).

3.2 BASE DE DADOS

A base de dados utilizada neste projeto foi a Bracol (ESGARIO; KROHLING; VENTURA, 2020), que contém 500 imagens de folhas capturadas em um ambiente controlado. As

Quadro 1 – Principais diferenças entre as propostas **A** e **B**.

-	PROPOSTA A	PROPOSTA B
Prioridade	Velocidade	Precisão
Segmentador	6 classes	3 classes
Classificador	N/A	5 classes

Fonte: Elaborado pelo autor (2023).

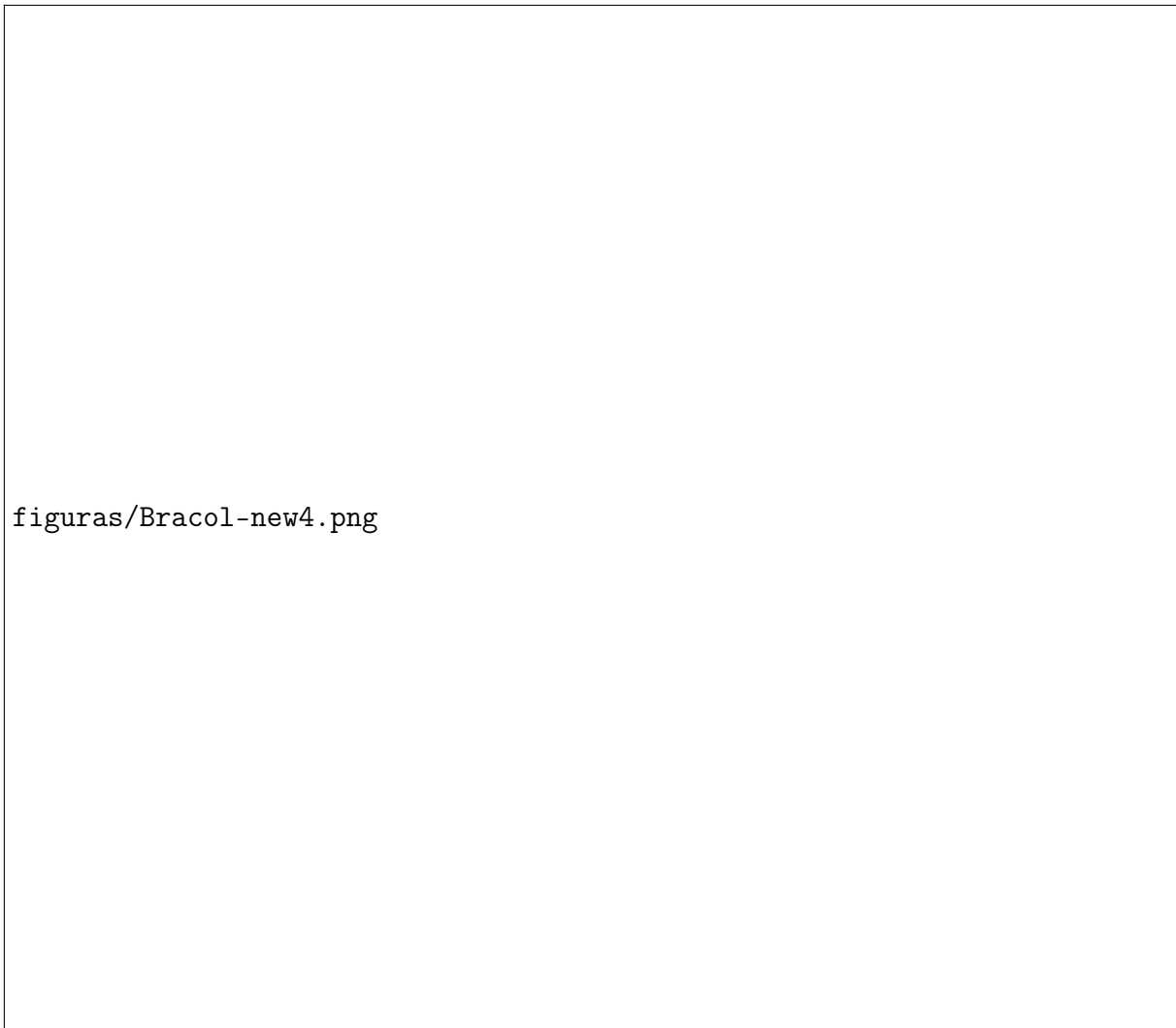
imagens foram coletadas por uma variedade de *smartphones* e contém folhas saudáveis e outras com lesões causadas pelo bicho-mineiro, ferrugem, phoma e cercospora.

A Bracol fornece para cada imagem em sua base uma máscara contendo três classes distintas: (0) Fundo, (1) Folha e (2) Lesão, sendo esta um indicativo que a região lesionada possui qualquer um dos quatro sintomas supracitados. Em razão da **PROPOSTA A**, adaptou-se as máscaras disponibilizadas para que cada rótulo indique o seu respectivo sintoma. Dessa forma, a nova base possui seis classes: (0) Fundo, (1) Folha, (2) Ferrugem, (3) Bicho-Mineiro, (4) Phoma e (5) Cercospora. A Figura 21 ilustra as máscaras da base original, a ser usada no treinamento do modelo de segmentação da **PROPOSTA B**, e as máscaras adaptadas. Os autores dessa base a dividem nos seguintes conjuntos de dados:

- 80% para o conjunto de treinamento (400 imagens).
- 10% para o conjunto de validação (50 imagens).
- 10% para o conjunto de teste (50 imagens).

Além disso, uma variação da Bracol é elaborada em (ESGARIO; KROHLING; VENTURA, 2020) a partir de recortes de regiões de folhas saudáveis e lesionadas, visando construir uma base para o treinamento de um modelo de classificação. Essa variação é composta por 2209 recortes, cuja distribuição de dados é apresentada na Tabela 1, conforme fornecida pelo autores. Essa divisão confere ao conjunto de treinamento cerca de 70,00% das imagens, 15,00% para os conjuntos de validação e 15,00% para o de teste. Por fim, a Figura 22 exhibe exemplos dessa base.

Figura 21 – Exemplos de imagens da base Bracol. A primeira coluna exibe as imagens, a central apresenta a máscara original e, por último, as máscaras adaptadas para segmentar cada sintoma individualmente.



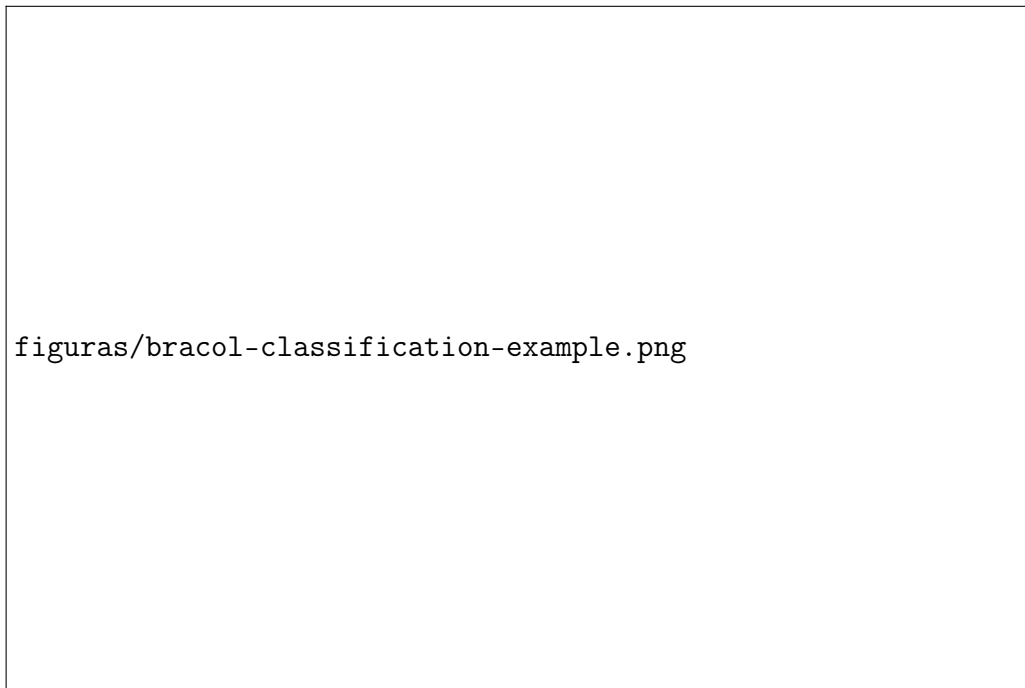
Fonte: Elaborado pelo autor (2023).

Tabela 1 – Divisão da base de classificação.

-	Saudável	Ferrugem	Bicho-Mineiro	Phoma	Cercospora	Total
Treinamento	182	434	374	325	224	1539
Validação	40	93	83	70	49	335
Teste	39	94	83	69	50	335
Total	11,82%	28,11%	24,45%	21,00%	14,62%	100,00%

Fonte: Elaborado pelo autor (2023).

Figura 22 – Exemplos da variação da base Bracol para classificação.



Fonte: Retirado de Esgario, Krohling e Ventura (2020).

Por fim, destaca-se que a adaptação da base de segmentação da Bracol, realizada para a **PROPOSTA A**, baseou-se nas informações dispostas nas bases de segmentação e a de recortes. Além disso, contou-se com o auxílio do INCAPER para rotular algumas imagens cujo sintomas não eram facilmente identificáveis ou não pertenciam à base de classificação.

3.3 EXPERIMENTOS

Para fins de denominação, o modelo de segmentação com seis classes definido na **PROPOSTA A** será chamado de **modelo de segmentação de sintomas**, ao mesmo tempo que o modelo da **PROPOSTA B** será intitulado de **modelo de segmentação de lesões**. Dito isso, esta seção irá explanar os experimentos realizados para cada rede neural proposta.

3.3.1 PROPOSTA A – Modelo de segmentação de sintomas

Com a finalidade de treinar uma rede para segmentar estresses bióticos em folhas de café, os seguintes experimentos foram realizados:

3.3.1.1 Definição da função de perda

A função de perda instiga o processo de aprendizagem e contribui diretamente no treinamento da rede e, conseqüentemente, no desempenho geral do modelo, principalmente quando a base de dados é demasiadamente desbalanceada. Por essa razão, o primeiro experimento procurou verificar o comportamento das diferentes funções apresentadas na Seção 2.2.2: *Cross-Entropy Loss*, *Focal Loss*, *Dice Loss* e *Jaccard Loss*. Para este experimento, uma U-Net foi treinada com os hiperparâmetros indicados no Quadro 2, alterando-se somente a função de perda. Neste caso, a U-Net foi selecionada, pois ela foi a rede que obteve os melhores resultados em (TASSIS; SOUZA; KROHLING, 2021) e (ESGARIO *et al.*, 2022).

Quadro 2 – Hiperparâmetros usados para definir a melhor função de perda para o modelo de segmentação de sintomas.

Hiperparâmetro	Valor
Tamanho do Lote	8
Tamaho da Imagem	128 x 256
Nº de Épocas	50
Taxa de Aprendizagem	0,001

Fonte: Elaborado pelo autor (2023).

3.3.1.2 Definição do modelo

O segundo experimento teve como objetivo definir o melhor modelo de segmentação semântica para as tarefas propostas. Para esta parte, diversas combinações de modelos de segmentação e codificadores foram treinadas por 50 épocas. O critério de decisão foram os valores de perda e *mIoU*, sendo esta a métrica mais comum para avaliar a performance de redes de segmentação. Os demais hiperparâmetros utilizados neste experimento são apresentados no Quadro 3.

Foram testados os seguintes modelos de segmentação: U-Net (RONNEBERGER; FISCHER; BROX, 2015), U-Net++ (ZHOU *et al.*, 2018), DeepLab V3+ (CHEN *et al.*, 2018) e PAN (*Pyramid Attention Network* (LI *et al.*, 2018)), combinados com diversos codificadores disponibilizados em (IAKUBOVSKII, 2019), tais como ResNet (HE *et al.*,

Quadro 3 – Hiperparâmetros usados para definir o melhor modelo de segmentação de sintomas.

Hiperparâmetro	Valor
Tamanho do Lote	4
Tamaho da Imagem	256 x 512
Nº de Épocas	50
Taxa de Aprendizagem	0,001
Função de Perda	<i>Jaccard Loss</i>

Fonte: Elaborado pelo autor (2023).

2016), Xception (CHOLLET, 2017), DenseNet161 (HUANG *et al.*, 2016), EfficientNet (TAN; LE, 2019), entre outros.

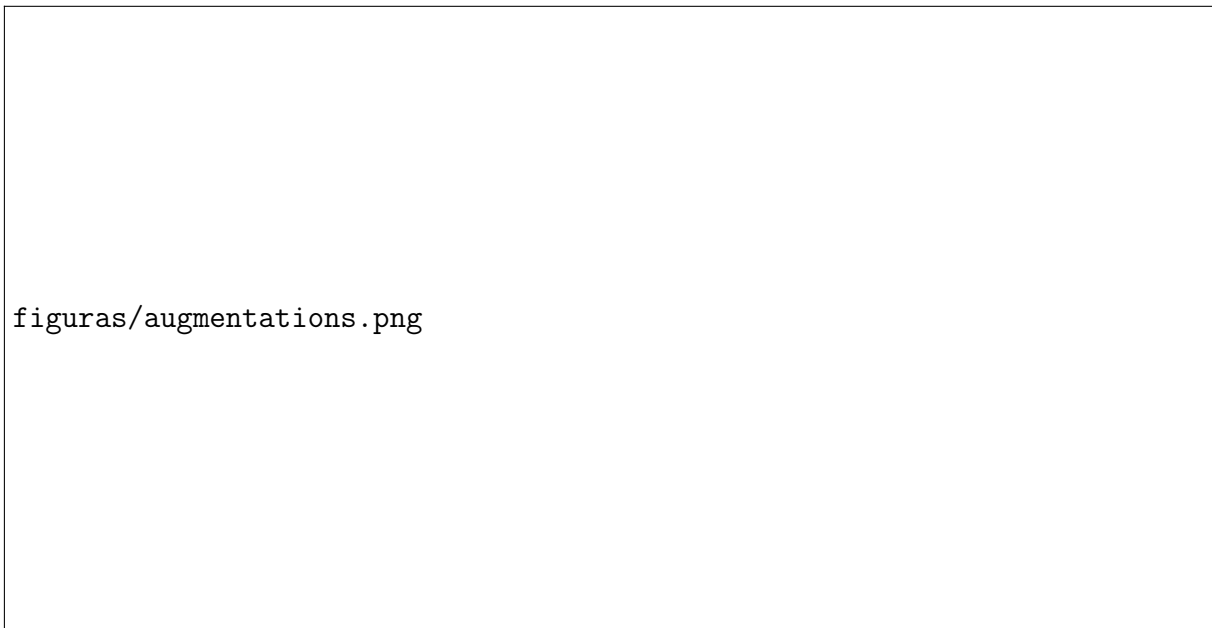
3.3.1.3 Técnicas de aumento de dados

Em razão da limitação da biblioteca da *Torchvision* em relação às técnicas de aumento de dados para bases de segmentação, aplicou-se os métodos disponibilizados *Albu-mentations* (BUSLAEV *et al.*, 2020), exibidas na Figura 23. Além disso, foi explorada a opção de aumentar o conjunto de treinamento com recortes de sintomas. Para criar esse novo conjunto de dados, todas as regiões lesionadas na base de treinamento da Bracol foram recortadas, a partir das máscaras reais. Em um segundo momento, após analisar os dados gerados, percebeu-se que a maioria dos recortes eram de ferrugem, por ser uma lesão que provoca diversas manchas isoladas. Para balancear a distribuição dos dados gerados, optou-se por manter os 55 maiores recortes de cada classe, totalizando 220 novas imagens, cujo exemplos são exibidos na Figura 24. Desta forma, o novo conjunto de treinamento inclui 620 imagens, sendo que 220 delas contém apenas recortes de folhas e as demais são as imagens originais do conjunto de treinamento.

3.3.1.4 Definição dos hiperparâmetros

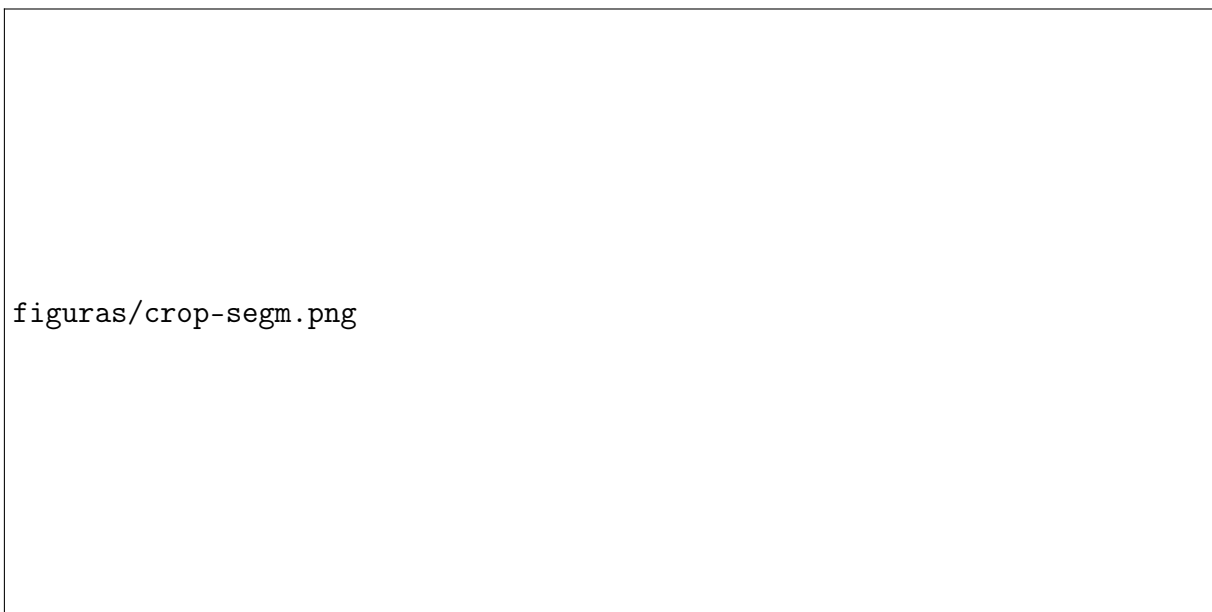
Tendo definido o modelo de segmentação, a função de perda e as técnicas de aumento de dados, treinou-se 30 modelos variando os hiperparâmetros dispostos no Quadro 4,

Figura 23 – Aumento de dados aplicados na base Bracol.



Fonte: Elaborado pelo autor (2023).

Figura 24 – Exemplos do aumento de dados realizados por recorte de sintomas.



Fonte: Elaborado pelo autor (2023).

fazendo as seguintes considerações:

- Para imagens de tamanho 256 x 512, o maior tamanho de lote possível era de 4 em razão dos recursos computacionais disponíveis.

- A taxa de aprendizagem utilizava um valor aleatório dentro da faixa apresentada e era reduzida por um fator de 0,50 a cada 10 épocas sem redução da perda no conjunto de validação.

Quadro 4 – Faixa de Hiperparâmetros testados no modelo de segmentação de sintomas.

Hiperparâmetro	Valores
Otimizador	[Adam, RMSProp]
Tamanho do Lote	[1, 4, 8]
Tamanho da Imagem	[128 x 256, 256 x 512]
Nº de Épocas	50
Taxa de Aprendizagem	[$10^{-2} \sim 10^{-5}$]

Fonte: Elaborado pelo autor (2023).

3.3.1.5 Pré-treinamento do codificador

Por último, foi investigada a opção de treinar uma rede cuja tarefa era classificar os sintomas disponibilizados na Bracol, similar ao modelo de classificação da **PROPOSTA B**, retratado no pipeline da Figura 18, para então usá-lo como base do codificador no modelo de segmentação. A hipótese era que esse modelo teria aprendido representações melhores para encontrar regiões lesionadas por pragas e doenças do que as redes treinadas na ImageNet.

3.3.2 PROPOSTA B – Modelo de segmentação de lesões

Replicou-se os parâmetros definidos para o modelo de segmentação de sintomas, por exceção da extensão do conjunto de treinamento descrita na Seção 3.3.1.3 - Técnicas de aumento de dados, que por sua vez não impactou no desempenho da rede treinada.

3.3.3 PROPOSTA B – Modelo de classificação

Com a finalidade de treinar uma rede para classificar os estresses bióticos segmentados pelo modelo de segmentação de lesões, optou-se por treinar uma ResNet50 em função

dos resultados que essa rede obteve em (ESGARIO *et al.*, 2022). As subseções seguintes explanam os experimentos realizados para esse modelo.

3.3.3.1 Técnicas de aumento de dados

Com o propósito de aumentar a robustez do classificador, diversas técnicas de aumento de dados foram utilizadas durante o treinamento da rede. A Figura 25 exemplifica as técnicas utilizadas, também disponíveis na biblioteca da *Torchvision*. O Quadro 5 apresenta os hiperparâmetros usados para esse experimento.

Figura 25 – Exemplos do aumento de dados realizados na base de classificação.



Fonte: Elaborado pelo autor (2023).

3.3.3.2 Ajuste fino

O ajuste fino, também conhecido como *fine-tuning*, é uma técnica utilizada no treinamento de redes neurais que consiste em ajustar os parâmetros da rede já treinada em uma tarefa específica para que ela possa ser utilizada em outras tarefas relacionadas.

Quadro 5 – Hiperparâmetros usados para avaliar a contribuição das técnicas de aumento de dados no modelo de classificação.

Hiperparâmetro	Valor
Tamanho do Lote	32
Tamaho da Imagem	224 x 224
Nº de Épocas	50
Taxa de Aprendizagem	0,01

Fonte: Elaborado pelo autor (2023).

Durante o ajuste fino, a rede pré-treinada é adaptada para a nova tarefa, atualizando os pesos das camadas finais da rede para obter uma melhor performance.

Em vista disso, o treinamento desse modelo se deu em duas etapas. A primeira consiste em utilizar um modelo pré-treinado na ImageNet e reajustar o peso de toda a rede para a nova tarefa. Na segunda fase, os parâmetros do modelo são congelados por exceção das camadas finais, que por sua vez serão reajustadas novamente. Dessa forma, este experimento resume-se em definir quais parâmetros serão reajustados na segunda fase. Os hiperparâmetros para esse teste foram os mesmos do experimento anterior, apresentado no Quadro 5.

3.3.3.3 Definição dos hiperparâmetros

O primeiro conjunto de treinamento testou combinações dos hiperparâmetros dispostos no Quadro 6 para determinar o melhor modelo de classificação da primeira fase de treinamento. Além disso, também foi testado o uso da função de perda *Cross Entropy Loss* com e sem pesos devido ao desbalanceamento da base. Após essa definição, uma segunda rodada de experimentos foi realizada para encontrar a taxa de aprendizagem a ser utilizada no ajuste fino.

Quadro 6 – Faixa de Hiperparâmetros testados no classificador.

Hiperparâmetro	Valores
Tamanho do Lote	[1, 32, 64]
Tamanho da Imagem	224x224
Nº de Épocas	50
Taxa de Aprendizagem	$[1 \times 10^{-5}, 2.5 \times 10^{-5}, 5 \times 10^{-5}, 7.5 \times 10^{-5}, 1 \times 10^{-4}, 2.5 \times 10^{-4}, 5 \times 10^{-4}, 7.5 \times 10^{-4}, 1 \times 10^{-3}, 2.5 \times 10^{-3}, 5 \times 10^{-3}, 7.5 \times 10^{-3}, 1 \times 10^{-2}]$

Fonte: Elaborado pelo autor (2023).

4 RESULTADOS

Neste capítulo, serão apresentados os resultados dos experimentos descritos no capítulo anterior, assim como a comparação entre as propostas desenvolvidas e o estado da arte na tarefa de segmentação e classificação de sintomas em folhas de café.

4.1 EXPERIMENTOS

4.1.1 PROPOSTA A – Modelo de segmentação de sintomas

4.1.1.1 Definição da função de perda

Os resultados obtidos nesse experimento são dispostos na Tabela 2, na qual é possível observar que a rede treinada com a *Jaccard Loss* obteve resultados melhores em todas as métricas medidas (*mPA*, *Dice* e *mIoU*). Além disso, um comparativo visual é exibido na Figura 26, em que a imagem apresenta a seguinte estrutura: cada linha foi o resultado preliminar de uma rede treinada com uma determinada função de custo. A primeira coluna denota a imagem original, a segunda é a máscara real, a terceira representa a máscara predita pela rede e, por último, um mapa binário indicando os erros de predição, representado pelos *pixels* mais escuros. Por fim, esse experimento também revelou que, no geral, os modelos estavam com dificuldades em predizer corretamente regiões de sombra da folha. Essa dificuldade motivou o uso de um novo método de aumento de dados para adicionar sombras aleatórias nas imagens, o que pode ser observado na Figura 27.

4.1.1.2 Definição do modelo

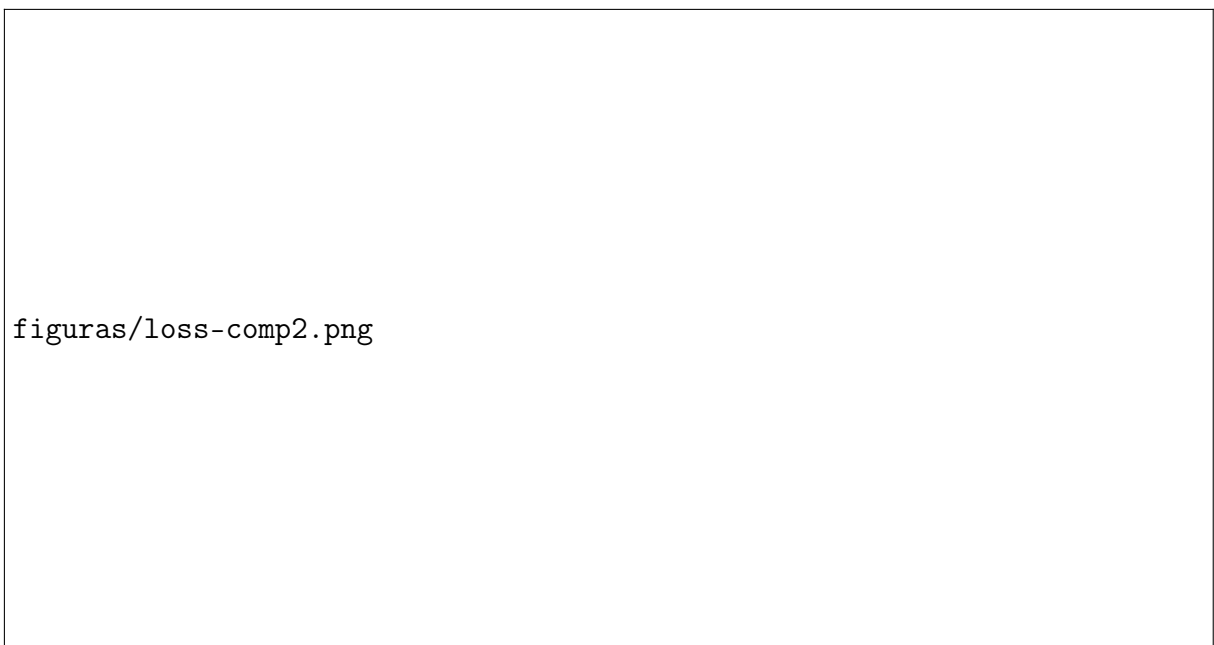
Entre os modelos de segmentação e codificadores citados na Seção 3.3.1.2, os modelos que obtiveram os melhores resultados foram a U-Net++ e DeepLab V3+. Os Quadros 7 e 8 representam o desempenho de cada uma das combinações realizadas com esses dois modelos, em que as linhas destacadas indicam os codificadores que obtiveram as menores perdas. Comparando os quadros, foi possível notar que ambas as redes obtiveram melhores resultados quando utilizando a EfficientNet como base do

Tabela 2 – Performance obtida para cada função de perda utilizada no treinamento de uma U-Net.

Função de Perda	<i>mAP</i>	<i>Dice</i>	<i>mIoU</i>
<i>CrossEntropyLoss</i>	56,05%	58,33%	50,65%
DiceLoss	60,10%	58,91%	49,79%
FocalLoss	48,80%	51,51%	45,72%
Jaccard Loss	60,30%	64,15%	54,58%

Fonte: Elaborado pelo autor (2023).

Figura 26 – Exemplos de predições de modelos de segmentação de sintomas treinados com diferentes funções de perda



Fonte: Elaborado pelo autor (2023).

codificador.

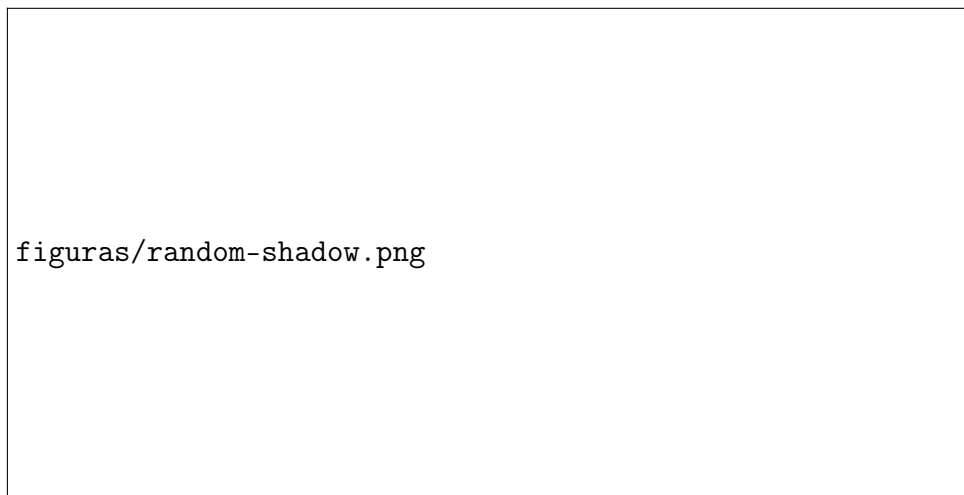
Quadro 7 – Resultados obtidos no conjunto de validação combinando a U-Net++ com diversos codificadores.

Codificador	Pesos	Perda	<i>mPA</i>	<i>C</i>	Comparativo de performance dos modelos treinados para
--------------------	--------------	--------------	------------	----------	---

Fonte: Elaborado pelo autor (2023).

A partir dos resultados apurados, optou-se por continuar trabalhando com os modelos

Figura 27 – Exemplos da técnica de aumento de dados para inserir sombras aleatórias.



Fonte: Elaborado pelo autor (2023).

Quadro 8 – Resultados obtidos no conjunto de validação combinando a DeepLab V3+ com diversos codificadores.

Codificador	Pesos	Perda	<i>mPA</i>	<i>Dice</i>	<i>mIoU</i>
ResNet34	<i>ImageNet</i>	0,25	69,75	81,25	71,35
ResNet50	<i>ImageNet</i>	0,23	68,34	79,36	70,16
ResNet152	<i>ImageNet</i>	0,26	67,18	80,56	68,68
EfficientNet-B5	<i>ImageNet</i>	0,16	72,58	87,75	76,97

Fonte: Elaborado pelo autor (2023).

U-Net++ e DeepLab V3+, empregando um codificador baseado na EfficientNet-B5 pré-treinada na ImageNet. A EfficientNet-B7 foi desconsiderada pois ela possui 67 milhões de parâmetros, mais que o dobro da B5 e, portanto, fere a **PROPOSTA A** que busca criar um *framework* mais leve e otimizado.

4.1.1.3 Técnicas de aumento de dados

O emprego das técnicas de aumento de dados acrescentou em média um valor de 3,70 pontos percentuais de *mIoU* nos modelos treinados. Combinados com a extensão da base de dados, formada por recortes de sintomas (ver Figura 24), esse acréscimo atingiu quase sete pontos percentuais.

4.1.1.4 Definição de hiperparâmetros

Por último, após a definição da função de perda, modelos e técnicas de aumento de dados, a rede que obteve os melhores resultados foi treinada com os hiperparâmetros estabelecidos no Quadro 9, alcançando um valor de *mIoU* de 84,27% com a U-Net++.

Quadro 9 – Hiperparâmetros selecionados para o treinamento da rede de segmentação de sintomas.

Hiperparâmetro	Valor
Otimizador	RMSProp
Função de Perda	<i>Jaccard Loss</i>
Tamanho do Lote	4
Tamaho da Imagem	256 x 512
Nº de Épocas	100
Taxa de Aprendizagem	0,00075*

* A taxa de aprendizado era reduzida por um fator de 0,50 a cada 10 épocas sem redução da perda no conjunto de validação.

Fonte: Elaborado pelo autor (2023).

4.1.1.5 Pré-treinamento do codificador

Para esse experimento, utilizou-se o classificador treinado na **PROPOSTA B** como base da rede de codificação dos modelos de segmentação semântica, tanto para a U-Net++ quanto para a DeepLabV3+, a fim de validar a premissa que um codificador treinado para resolver o problema de doenças em folhas de café teria aprendido a identificar padrões mais relevantes que outro treinado na ImageNet. Apesar disso, esse experimento não apresentou resultados expressivos, isto é, o emprego de um codificador treinado na ImageNet gerou um modelo extremamente similar ao treinado na Bracol, quando comparado as métricas de desempenho obtidas.

4.1.2 PROPOSTA B – Modelo de classificação

Os experimentos realizados para o modelo de classificação, podem ser resumidos da seguinte forma:

- O emprego de técnicas de aumento de dados proporcionou um aumento médio de 3,30% na acurácia dos modelos treinados.
- O ajuste fino na segunda fase de treinamento obteve um acréscimo de 2,69% quando reajustando apenas os últimos 20 parâmetros.

Por fim, o Quadro 10 expõe o conjunto de hiperparâmetros que, combinados às técnicas de aumento de dados e ajuste fino, obtiveram a melhor acurácia no conjunto de validação.

Quadro 10 – Hiperparâmetros do treinamento da rede de classificação.

Hiperparâmetro	Valor (etapa #1)	Valor (etapa #2)
Otimizador	Adam	Adam
Função de Perda	<i>CrossEntropy</i>	<i>CrossEntropy</i>
Tamanho do Lote	32	32
Tamaho da Imagem	224x224	224x224
Nº de Épocas	50	50
Taxa de Aprendizagem	0,01	0,003

Fonte: Elaborado pelo autor (2023).

4.2 RESULTADOS

4.2.1 PROPOSTA A

Conforme especificado na Seção 3.1, essa proposta consiste em desenvolver uma arquitetura capaz de identificar o agente causador de cada lesão na folha de café, assim como expressar as suas respectivas severidades, com uma única ConvNet.

Por essa razão, modelos de segmentação semântica foram explorados para essa tarefa e, a partir dos experimentos descritos, foram desenvolvidos dois modelos: uma U-Net++ e outra DeepLab V3+, com um codificador baseado na EfficientNet-B5. Com os hiperparâmetros descritos no Quadro 9, a DeepLab V3+ e a U-Net++ obtiveram no conjunto de testes um valor de *mIoU* de 79,83% e 84,27%, respectivamente. A Tabela 3 apresenta as métricas obtidas por cada modelo. As Figuras 28 e 29 apresentam as

curvas de treinamento de cada um desses modelos, exibindo a evolução da perda, acurácia, coeficiente *Dice* e *mIoU* ao longo das épocas.

Tabela 3 – Comparativo de performance dos modelos treinados para a **PROPOSTA A**.

Modelo	<i>mAP</i>	<i>Dice</i>	<i>mIoU</i>
<i>DeepLab V3+</i>	87,84%	86,95%	79,83%
U-Net++	90,98%	90,50%	84,27%

Fonte: Elaborado pelo autor (2023).

A Tabela 4 expande os resultados de acurácia e *IoU* para cada classe do segmentador. Analisando essas informações, nota-se que a classe cercospora apresenta os piores resultados, seguida da ferrugem. Ao observar esses dados acompanhado da Figura 30, percebe-se que:

- Regiões circulares e amarronzadas, cercadas por regiões amareladas foram classificadas como cercospora, embora a máscara real indicasse que o sintoma fosse apenas ferrugem (linhas 1 e 2 da Figura 30).
- Regiões amareladas foram classificadas como ferrugem, embora a máscara real indicasse que todo o sintoma fosse cercospora (linha 3 da Figura 30) ou bicho-mineiro (linha 4 da Figura 30).

Tabela 4 – Valores de acurácia e *IoU* por cada classe do segmentador.

Proposta	<i>AP</i>	<i>IoU</i>
0 - Fundo	99,77%	99,38%
1 - Folha	98,70%	97,95%
2 - Ferrugem	84,25%	71,84%
3 - Bicho-mineiro	96,56%	82,12%
4 - Phoma	97,01%	90,41%
5 - Cercospora	44,16%	37,26%

Fonte: Elaborado pelo autor (2023).

Figura 28 – Curva da perda, acurácia média, coeficiente *Dice* e *mIoU* durante o treinamento da U-Net++.



Fonte: Elaborado pelo autor (2023).

Ao olho humano comum, isto é, de um não especialista no assunto, os erros cometidos são compreensíveis, considerando que a cercospora é caracterizada por essas regiões marrons e arredondadas e a ferrugem pelas suas manchas amareladas. Isso denota algumas possibilidades, incluindo que a base de dados utilizada não rotula corretamente regiões em que existam duas ou mais lesões sobrepostas e, conseqüentemente, o

Figura 29 – Curva da perda, acurácia média, coeficiente *Dice* e *mIoU* durante o treinamento do DeepLab V3+.



Fonte: Elaborado pelo autor (2023).

modelo aprendido com esses dados também apresenta essa dificuldade. Portanto, estima-se que um novo processo de rotulação deveria ser realizado, acompanhado por um especialista capaz de decernir corretamente os limites entre cada sintoma.

Além disso, é pertinente mencionar que a *cercospora* é o sintoma com menos exemplos

Figura 30 – Erros mais grosseiros do segmentador de sintomas no conjunto de validação. A primeira coluna é a imagem de entrada, seguida pela máscara real e a predita. A última coluna representa um recorte da imagem de entrada onde o modelo errou (destacado por uma circunferência vermelha nas máscaras preditas).



Fonte: Elaborado pelo autor (2023).

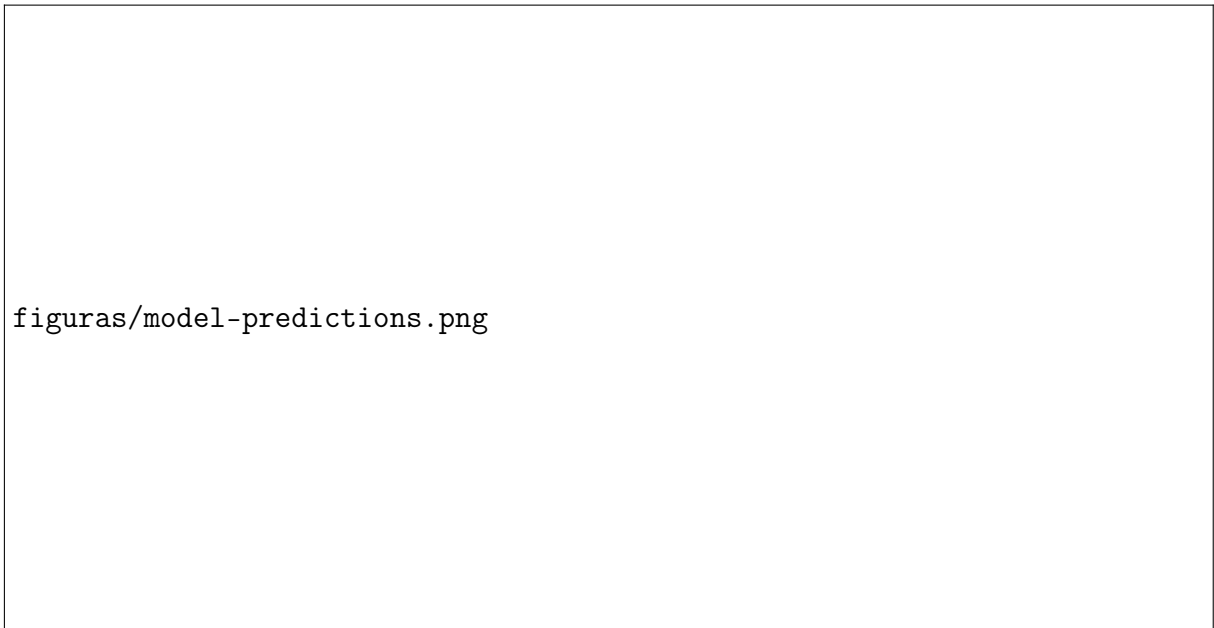
na base de dados, de modo que, ao comparar a quantidade de *pixels* por sintoma, ela representa apenas 4,20% de toda a base disponível para treinamento. Estima-se que esse desbalanceamento é o principal motivo para os baixos valores de acurácia a *mIoU* na classe *cercospora*.

A Figura 31 exibe mais exemplos de predições realizadas pelo modelo no primeiro lote de imagens do conjunto de teste. A última coluna da imagem representa a dissimilaridade entre as máscaras preditas e as reais que, por exceção das bordas, são praticamente nulas. Esse comportamento se estende para quase todas as imagens do conjunto de testes, excetuando exemplos de sintomas sobrepostos e os da Figura 30.

4.2.2 PROPOSTA B

Nesta proposta, duas redes neurais são desenvolvidas. A primeira tem como objetivo segmentar as regiões lesionadas para que estas possam ser recortadas e enviadas

Figura 31 – Comparação entre as máscaras preditas pelo modelo de segmentação de sintomas e as máscaras verdadeiras.



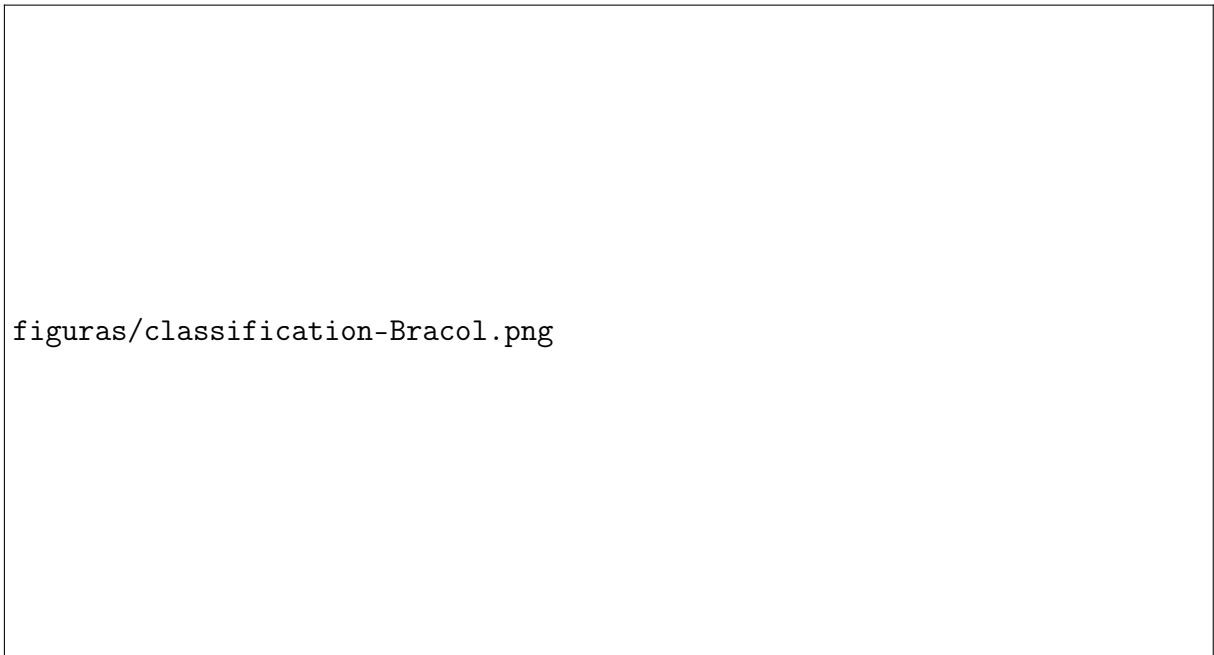
Fonte: Elaborado pelo autor (2023).

para um segundo estágio, onde outra rede será responsável por classificar os sintomas, similar aos projetos de Tassis, Souza e Krohling (2021) e Esgario *et al.* (2022).

A rede neural utilizada para o estágio de classificação foi a ResNet50 (em virtude dos trabalhos relacionados), treinada anteriormente na base de dados da ImageNet. Para adaptar o conhecimento dessa rede para as tarefas de pragas e doenças de café, a rede foi retreinada com recortes das imagens da Bracol em duas etapas, que passaram por aumento de dados online (ou seja, novas transformações eram aplicadas a cada lote de imagens), conforme mostrado na Figura 32. A primeira etapa reajustou os pesos de todos os parâmetros da rede por 50 épocas, enquanto que no segundo momento, parte dos pesos foram congelados e a rede foi treinada novamente por mais 50 épocas com uma taxa de aprendizado reduzida. O histórico de treinamento é exibido na Figura 33.

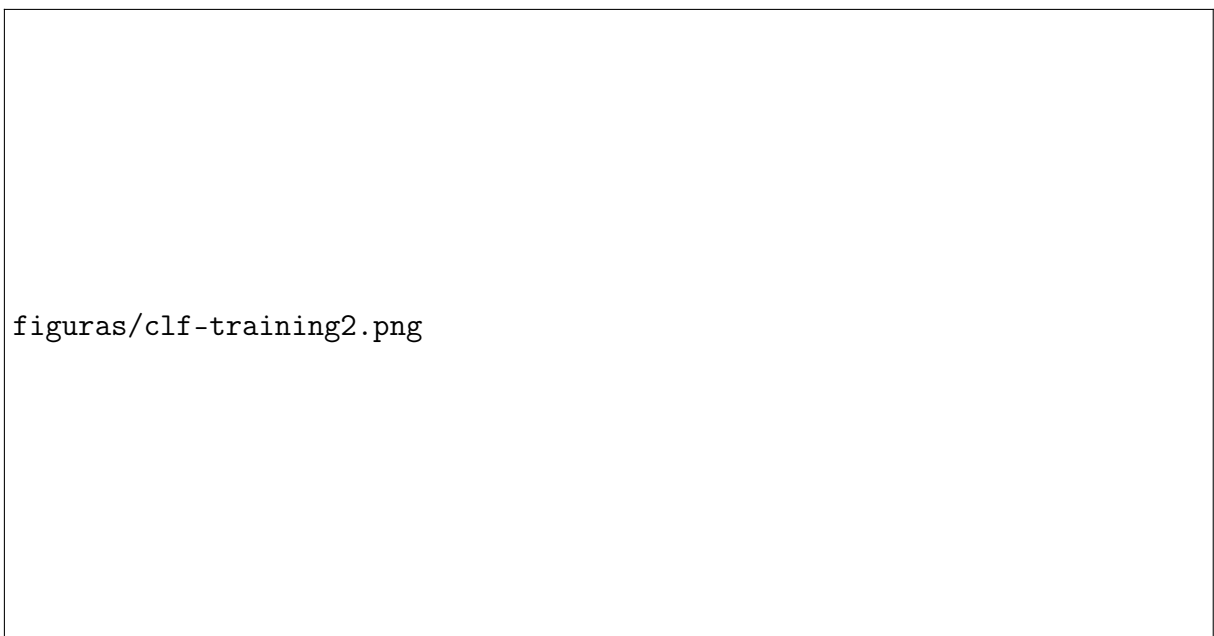
Durante o treinamento, sempre que a perda do conjunto de validação diminuísse, os parâmetros da rede eram salvos para guardar a melhor rede treinada. Nesses experimentos, esse resultado foi alcançado na época #29 da segunda etapa de treinamento, alcançando uma acurácia de 98,51% no conjunto de validação e 97,31% na de testes.

Figura 32 – Lote de treinamento do modelo de classificação contendo 32 imagens com aumento de dados



Fonte: Elaborado pelo autor (2023).

Figura 33 – Curva da perda e acurácia durante o treinamento do modelo de classificação.



Fonte: Elaborado pelo autor (2023).

Em comparação, o trabalho de (ESGARIO *et al.*, 2022) obteve uma acurácia de 97,07% com a mesma ResNet50.

Além do classificador, fez-se necessário treinar um novo modelo de segmentação semântica com a base original da Bracol para encontrar as regiões lesionadas. As arquiteturas selecionadas foram a DeepLab V3+ e a U-Net++, empregando o codificador baseado na EfficientNet-B5, em razão dos resultados obtidos nos experimentos da **PROPOSTA A**. Essas configurações retornaram um valor de *mIoU* de 95,01% e 95,04% para a DeepLab V3+ e U-Net++, respectivamente. Esse resultado é equivalente aos resultados encontrados em Esgario *et al.* (2022) e Tassis, Souza e Krohling (2021), que obtiveram um valor de *mIoU* igual a 94,85% e o 94,25%, respectivamente. A Tabela 5 exibe os resultados obtidos em comparação com a literatura.

Tabela 5 – Comparativo dos modelos treinados com as do estado da arte.

Modelo	<i>mAP</i>	<i>Dice</i>	<i>mIoU</i>
<i>DeepLab V3+</i>	97,24%	97,35%	95,01%
<i>U-Net++</i>	97,18%	97,35%	95,04%
<i>U-Net (TASSIS; SOUZA; KROHLING, 2021)</i>	99,54%	-	94,25%
<i>U-Net (ESGARIO et al., 2022)</i>	99,53	-	94,85%

Fonte: Elaborado pelo autor (2023).

4.2.3 Comparando as propostas

Comparando os dois *pipelines* desenvolvidos e seus respectivos resultados, torna-se evidente que a **PROPOSTA B** ostenta métricas melhores que a **PROPOSTA A**, conforme indicado na Tabela 6. Seu desempenho deve-se ao fato de ela dividir de forma eficiente o problema de classificar e quantificar a severidade de quatro diferentes tipos de pragas e doenças em dois problemas mais simples. Entretanto, como é esperada, essa solução possui um tempo de inferência maior, ou seja, é mais lenta e também exige mais recursos computacionais em termos de memória. A análise do custo computacional será apresentada mais a frente.

Considerando que as máscaras de segmentação ofereceram um cálculo preciso da gravidade dos sintomas, a severidade de cada amostra foi determinada a partir das máscaras verdadeiras e das preditas pelos modelos de segmentação, considerando apenas se a região segmentada representava um estresse biótico qualquer. A partir

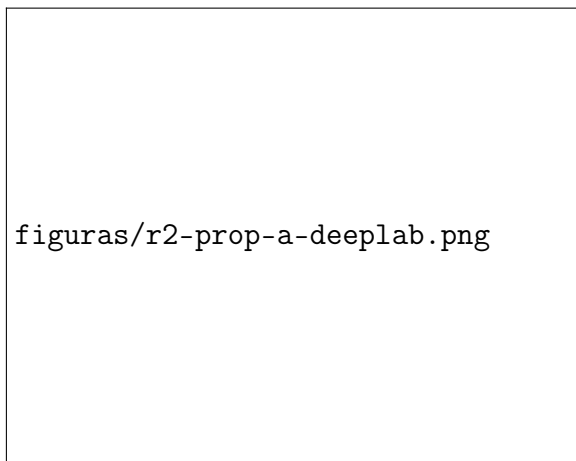
Tabela 6 – Comparativo dos resultados obtidos nos modelos de segmentação, com a U-Net++, nas duas propostas desenvolvidas.

Proposta	<i>mAP</i>	<i>Dice</i>	<i>mIoU</i>
A	90,98%	90,50%	84,27%
B	97,18%	97,35%	95,04%

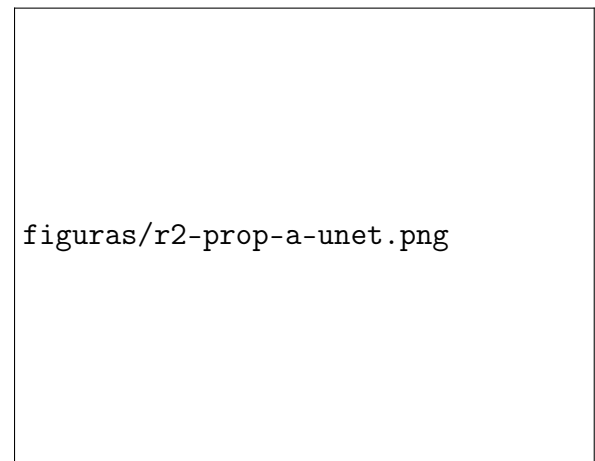
Fonte: Elaborado pelo autor (2023).

desses dados, foram elaborados os gráficos de dispersão e obtidos os coeficientes de determinação R^2 a fim de averiguar a coerência das informações com a linha de regressão. Os gráficos dos modelos da **PROPOSTA A** e **B** são exibidos nas Figuras 34 e 35, respectivamente.

Figura 34 – Gráfico de dispersão da severidade predita e real dos modelos de segmentação de sintomas (a) DeepLab V3+. (b) U-Net++.



(a)

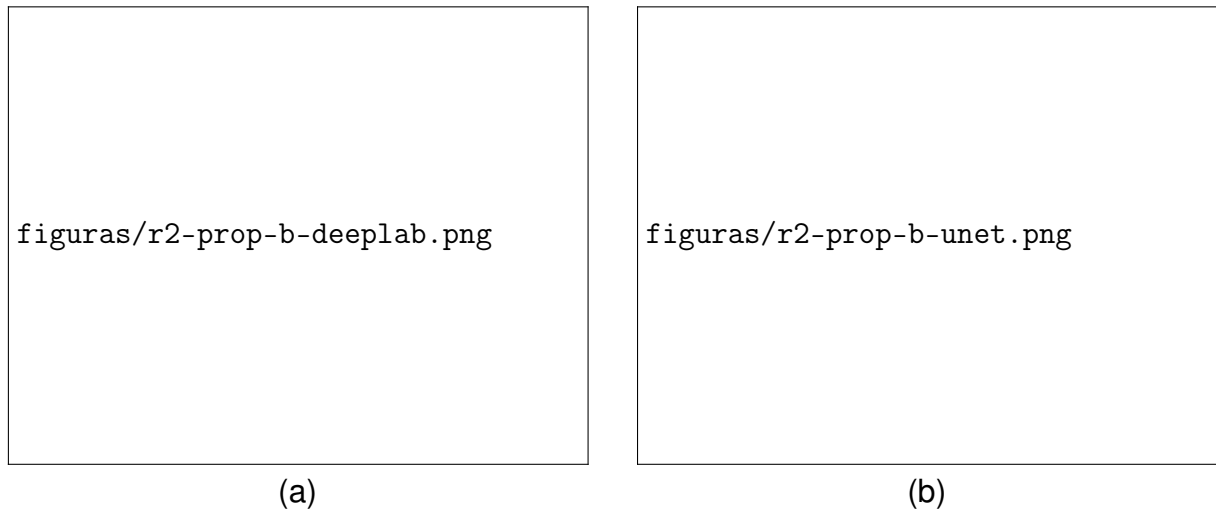


(b)

Fonte: Elaborado pelo autor (2023).

Com base nisso, nota-se que ambas as arquiteturas apresentaram alta correlação entre a severidade prevista e a verdadeira, em especial as que utilizaram o modelo U-Net++ que obtiveram um valor de R^2 de 0,9868 e 0,9932 para o modelo de segmentação de sintomas e lesões, respectivamente. Vale a pena mencionar que o melhor modelo desenvolvido em (ESGARIO *et al.*, 2022) obteve um valor de R^2 igual a 0,9855. Isso indica que, apesar do modelo da **PROPOSTA A** apresentar um valor de *mIoU* menor, a segmentação das regiões lesionadas por ela são mais precisas que a literatura. Em retrospectiva, essa análise também revela que o maior erro dessa arquitetura não

Figura 35 – Gráfico de dispersão da severidade predita e real dos modelos de segmentação de lesões (a) DeepLab V3+. (b) U-Net++.



Fonte: Elaborado pelo autor (2023).

está em segmentar as regiões lesionadas, mas sim em classificá-las corretamente, conforme foi ilustrado na Figura 30.

A Tabela 7 expõe uma análise rápida sobre o número de parâmetros e memória estimada para cada modelo desenvolvido. Nela é possível observar que a **PROPOSTA A** necessita de apenas 61,31% da memória total ocupada pela **PROPOSTA B**, evidenciando que ela é significativamente mais leve.

Tabela 7 – Comparativo de número de parâmetros e tamanho estimado dos modelos desenvolvidos com a DeepLab V3+.

Proposta	Modelo	Nº de parâmetros	Memória (Mb)
A	Segmentação de Sintomas	28.441.030	2736,98
B	Segmentação de Lesões	28.440.259	2727,79
B	Classificação de Sintomas	28.351.029	1736,00

Fonte: Elaborado pelo autor (2023).

Outro fator relevante é a velocidade do modelo. Para um lote de 50 imagens, a **PROPOSTA B** necessita de 2,89 segundos usando GPU e 33,37 segundos usando somente a CPU. Em contrapartida, a arquitetura de um único estágio leva apenas 1,60 e 18,44 segundos com e sem GPU, respectivamente. Em outras palavras, a

PROPOSTA A consegue ser cerca de 1,80 vezes mais rápida. Esse teste foi realizado em um computador particular, processador Intel® Core™ i5-8400 CPU @ 2.80GHz × 6, com 16 Gb de memória RAM e placa de vídeo dedicada GeForce RTX 2060.

Por fim, considerando o objetivo deste trabalho de elaborar um sistema capaz de identificar e quantificar a severidade de diferentes estresses bióticos que afetam as folhas de café, os modelos desenvolvidos podem ser usados em aplicações que permitam o produtor rural a reconhecer as pragas e doenças que prejudicam a sua produção. Em vista disso, é interessante que a aplicação final consiga, a partir de uma foto de uma folha de café, identificar os sintomas e informar o grau de severidade em níveis discretos, para facilitar o entendimento. Também é interessante que o modelo indique o grau de confiança na classificação ou que notifique o usuário final em casos de alta incerteza do resultado.

5 CONCLUSÕES E TRABALHOS FUTUROS

5.1 CONCLUSÕES GERAIS

A metodologia apresentada consistiu na avaliação de diferentes abordagens baseadas em aprendizado profundo para o problema de segmentação, classificação e quantificação do estresse biótico das folhas de café. Para tal finalidade, duas propostas foram feitas para abordar esse problema.

A **PROPOSTA A** desvencilha-se das abordagens encontradas na literatura ao conceber um único modelo de segmentação semântica apto a segmentar e classificar os estresses bióticos. Dos modelos desenvolvidos, a U-Net++ apresentou o melhor resultado com um *mIoU* de 84,24%. Apesar da performance quantitativa inferior, em comparação com a **PROPOSTA B**, essa abordagem é compensada por usar menos memória e ser quase duas vezes mais rápida e, portanto, torna a solução mais viável de ser embarcada em um dispositivo móvel. Ademais, conforme o nosso entendimento atual, este é o primeiro trabalho que propôs identificar e calcular a severidade de doenças em folhas de café com um único modelo de segmentação.

A **PROPOSTA B** adota as abordagens empregadas na literatura e cria um *framework* composto por duas redes neurais para segmentar e classificar as lesões em folhas. Os resultados da segmentação semântica chegaram a um valor de 95,04% de *mIoU* utilizando uma U-Net++, a medida que o estado da arte até o momento alcançou 94,85%. Para o classificador, ao passo que outros trabalhos obtiveram uma acurácia de até 97,07%, o modelo desenvolvido atingiu 97,31%. Esses resultados demonstram que o sistema produzido apresenta uma performance consistente com a literatura, com um incremento superficial de forma geral.

Uma limitação que ambas as propostas apresentaram está relacionado à falta de capacidade de classificar áreas onde duas ou mais lesões se sobrepõem. Apesar dessa limitação, as duas propostas apresentaram resultados promissores no que tange à identificação e estimação da severidade de doenças em folhas de café. Por fim, vale mencionar que embora o estudo tenha sido focado em folhas de café, a metodologia é escalável para culturas diferentes.

5.2 TRABALHOS FUTUROS

Fora as observações citadas na Seção 5.1, compreende-se que este trabalho tem espaço para melhoria. Vale destacar as seguintes propostas de trabalhos futuros:

- Reavaliar a rotulação da base Bracol, com auxílio de um especialista, para isolar corretamente estresses bióticos sobrepostos.
- Acrescentar mais exemplos de folhas com cercospora por meio de novas imagens ou técnicas de aumento de dados a fim de balancear a base trabalhada.
- Explorar novas bases de dados que tratam outros sintomas que afetam as folhas de café, como o ácaro-vermelho existente na base RoCoLe (PARRAGA-ALAVA *et al.*, 2019).
- Explorar modelos mais leves, como o MobileNetV2, para embarcar em um dispositivo móvel.
- Desenvolver um aplicativo de *smartphone* que identifique as pragas e doenças em folhas de café, embarcando o modelo de segmentação semântica concebido pela **PROPOSTA A**.

Baseado nos modelos propostos, um aplicativo pode ser desenvolvido de modo que o usuário final carregue uma imagem de uma folha de café isolada e, partir dessa imagem, o algoritmo irá segmentar as regiões doentes, indicando seus respectivos estresses bióticos, conforme exemplificado na Figura 36. Por fim, é interessante que o aplicativo apresente um resumo das condições encontradas na folha, classificando a severidade de cada sintoma em níveis de gravidade, como por exemplo Gravidade 1 (ocupação da doença entre de 0 a 2%), Gravidade 2 (entre 2 e 5%) e assim por diante. Para tal, recomenda-se o acompanhamento de um especialista para determinar essas novas categorias. Dessa maneira, o usuário final conseguirá ter uma indicação mais intuitiva sobre a gravidade dos sintomas existentes na folha e pode acompanhar na imagem segmentada a aparência de cada um dos sintomas identificados.

Figura 36 – Exemplo de aplicativo para dispositivo móvel baseado nos modelos desenvolvidos nesse projeto.



Fonte: Elaborado pelo autor (2023).

REFERÊNCIAS

- BUSLAEV, A. *et al.* Alumentations: Fast and flexible image augmentations. **Information**, v. 11, n. 2, 2020. ISSN 2078-2489. Disponível em: <<https://www.mdpi.com/2078-2489/11/2/125>>. Acesso em: 08 abr. 2022.
- CHEN, L.-C. *et al.* **Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs**. 2016. Disponível em: <<https://arxiv.org/abs/1412.7062>>. Acesso em: 12 jan. 2023.
- CHEN, L.-C. *et al.* **DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs**. 2017. Disponível em: <<https://arxiv.org/abs/1606.00915>>. Acesso em: 18 abr. 2023.
- CHEN, L.-C. *et al.* **Rethinking Atrous Convolution for Semantic Image Segmentation**. 2017. Disponível em: <<https://arxiv.org/abs/1706.05587>>. Acesso em: 18 abr. 2023.
- CHEN, L.-C. *et al.* **Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation**. 2018. Disponível em: <<https://arxiv.org/abs/1802.02611>>. Acesso em: 18 abr. 2023.
- CHOLLET, F. **Xception: Deep Learning with Depthwise Separable Convolutions**. 2017. Disponível em: <<https://arxiv.org/abs/1610.02357>>. Acesso em: 14 abr. 2022.
- CORRÊA, P. C.; OLIVEIRA, G. H. H.; BOTELHO, F. M. Armazenamento. In: FONSECA, A.; SAKIYMA, N.; BORÉM, A. (Ed.). **Café Conilon: do Plantio à Colheita**. Ufv. Viçosa, MG: [s.n.], 2015. cap. 11, p. 231–257. ISBN 978-85-7269-528-2.
- DAVIS, A. P. *et al.* Growing coffee: *Psilanthus* (Rubiaceae) subsumed on the basis of molecular and morphological data; implications for the size, morphology, distribution and evolutionary history of *Coffea*. **Botanical Journal of the Linnean Society**, v. 167, n. 4, p. 357–377, dec 2011. ISSN 00244074. Disponível em: <<https://academic.oup.com/botlinnean/article-lookup/doi/10.1111/j.1095-8339.2011.01177.x>>. Acesso em: 8 jun. 2021.
- DENG, J. *et al.* Imagenet: A large-scale hierarchical image database. In: **2009 IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2009. p. 248–255. ISSN 1063-6919. Acesso em: 13 mar. 2019.
- ESGARIO, J. G. *et al.* An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning. **Information Processing in Agriculture**, v. 9, p. 38–47, 2022. ISSN 22143173. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2214317321000044>>. Acesso em: 22 set. 2022.
- ESGARIO, J. G.; KROHLING, R. A.; VENTURA, J. A. Deep learning for classification and severity estimation of coffee leaf biotic stress. **Computers and Electronics in Agriculture**, v. 169, 2020. ISSN 01681699. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S0168169919313225>>. Acesso em: 23 set. 2022.

FERRÃO, R. G. *et al.* **Café Conilon: Técnicas de Produção com Variedades Melhoradas**. 4. ed. Vitória, ES: [s.n.], 2012. 75 p. ISSN 1519-0528. Disponível em: <http://www.sapc.embrapa.br/arquivos/consorcio/publicacoes_tecnicas/circular tecnica-cafeconilon.pdf>. Acesso em: 10 nov. 2020.

FERRÃO, R. G. *et al.* **Café Conilon**. 2. ed. Vitória, ES: Incaper, 2017., 2016. ISBN 978-85-89274-26-5. Disponível em: <<http://biblioteca.incaper.es.gov.br/digital/handle/123456789/3114>>. Acesso em: 10 nov. 2020.

GRANDE SOLDADOR. **Robôs**. [S.l.]: 20th Century Fox, 2005.

HE, K. *et al.* **Mask R-CNN**. 2018. Disponível em: <<https://arxiv.org/abs/1703.06870>>. Acesso em: 23 set. 2022.

HE, K. *et al.* 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. ISSN 1063-6919. Acesso em: 24 mar. 2019.

HUANG, G. *et al.* Densely Connected Convolutional Networks. **Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017**, v. 2017-Janua, p. 2261–2269, aug 2016. Disponível em: <<http://arxiv.org/abs/1608.06993>>. Acesso em: 10 out. 2020.

IAKUBOVSKII, P. **Segmentation Models Pytorch**. GitHub, 2019. <https://github.com/qubvel/segmentation_models.pytorch>. Disponível em: <https://github.com/qubvel/segmentation_models.pytorch>. Acesso em: 23 jan. 2023.

INCAPER. **Cafeicultura**. 2019. Disponível em: <<https://incaper.es.gov.br/cafeicultura>>. Acesso em: 10 abr. 2023.

INCAPER. **Cafeicultura - Café Conilon**. 2020. Disponível em: <<https://incaper.es.gov.br/cafeicultura-conilon>>. Acesso em: 23 mai. 2023.

JADON, S. A survey of loss functions for semantic segmentation. In: . [S.l.]: IEEE, 2020. p. 1–7. ISBN 978-1-7281-9468-4. Acesso em: 18 jan. 2023.

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>>. Acesso em: 23 mar. 2019.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, v. 521, n. 7553, p. 436–444, may 2015. ISSN 0028-0836. Disponível em: <<http://www.nature.com/articles/nature14539>>. Acesso em: 12 jul. 2019.

Lecun, Y. *et al.* Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, Nov 1998. Acesso em: 08 mai. 2019.

LI, H. *et al.* Pyramid attention network for semantic segmentation. **CoRR**, abs/1805.10180, 2018. Disponível em: <<http://arxiv.org/abs/1805.10180>>. Acesso em: 09 mar. 2023.

- LONG, J.; SHELHAMER, E.; DARRELL, T. **Fully Convolutional Networks for Semantic Segmentation**. 2015. Disponível em: <<https://arxiv.org/abs/1411.4038>>. Acesso em: 12 jan. 2023.
- MA, J. *et al.* Loss odyssey in medical image segmentation. **Medical Image Analysis**, Elsevier B.V., v. 71, p. 102035, 7 2021. ISSN 13618415. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S1361841521000815>>. Acesso em: 18 jan. 2023.
- MINAEE, S. *et al.* Image segmentation using deep learning: A survey. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 44, p. 3523–3542, 1 2020. ISSN 19393539. Disponível em: <<http://arxiv.org/abs/2001.05566>>. Acesso em: 18 nov. 2022.
- MISHRA, M. K.; SLATER, A. Recent Advances in the Genetic Transformation of Coffee. **Biotechnology Research International**, Hindawi Publishing Corporation, v. 2012, p. 580857, 2012. ISSN 2090-3138. Disponível em: <<https://doi.org/10.1155/2012/580857>>. Acesso em: 8 jun. 2021.
- MITCHELL, T. M. **Machine Learning**. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072.
- NOH, H.; HONG, S.; HAN, B. **Learning Deconvolution Network for Semantic Segmentation**. 2015. Disponível em: <<https://arxiv.org/abs/1505.04366>>. Acesso em: 18 abr. 2023.
- PARRAGA-ALAVA, J. *et al.* Rocola: A robusta coffee leaf images dataset for evaluation of machine learning based methods in plant diseases recognition. **Data in Brief**, v. 25, p. 104414, 8 2019. ISSN 23523409. RoCole Classes:
1. Rust (Lv. 1, 2 e 3)
2. Red Spider Mite (Ácaro vermelho)
. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S2352340919307693>>. Acesso em: 03 out. 2021.
- RAMOS, P. *et al.* Automatic fruit count on coffee branches using computer vision. **Computers and Electronics in Agriculture**, Elsevier B.V., v. 137, p. 9–22, may 2017. ISSN 01681699. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S016816991630922X>>. Acesso em: 5 nov. 2020.
- RODRÍGUEZ, J. P. *et al.* A computer vision system for automatic cherry beans detection on coffee trees. **Pattern Recognition Letters**, Elsevier B.V., v. 136, p. 142–153, aug 2020. ISSN 01678655. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0167865520302117>>. Acesso em: 5 nov. 2020.
- RONNEBERGER, O.; FISCHER, P.; BROX, T. **U-Net: Convolutional Networks for Biomedical Image Segmentation**. 2015. Disponível em: <<https://arxiv.org/abs/1505.04597>>. Acesso em: 18 abr. 2023.
- ROSEBROCK, A. **Intersection over Union (IoU) for object detection**. 2016. Disponível em: <<https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>>. Acesso em: 20 jun. 2021.

SILVA, J. d. S. e.; MORELI, A. P.; FILHO, A. C. V. Tecnologias pós-colheita para conilon de qualidade. In: FONSECA, A.; SAKIYMA, N.; BORÉM, A. (Ed.). **Café Conilon: do Plantio à Colheita**. Ufv. Viçosa, MG: [s.n.], 2015. cap. 10, p. 204–230. ISBN 978-85-7269-528-2.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014. Disponível em: <<https://arxiv.org/abs/1409.1556>>. Acesso em: 24 mar. 2019.

SULTANA, F.; SUFIAN, A.; DUTTA, P. Evolution of image segmentation using deep convolutional neural network: A survey. **Knowledge-Based Systems**, v. 201-202, p. 1–38, 1 2020. ISSN 09507051. Disponível em: <<http://arxiv.org/abs/2001.04074><http://dx.doi.org/10.1016/j.knosys.2020.106062>>. Acesso em: 18 jan. 2023.

TAN, M.; LE, Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks. **CoRR**, abs/1905.11946, 2019. Disponível em: <<http://arxiv.org/abs/1905.11946>>. Acesso em: 14 fev. 2023.

TASSIS, L. M.; SOUZA, J. E. T. de; KROHLING, R. A. A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images. **Computers and Electronics in Agriculture**, v. 186, 2021. ISSN 01681699. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S0168169921002088>>. Acesso em: 23 set. 2022.

ZHOU, Z. *et al.* Unet++: A nested u-net architecture for medical image segmentation. **CoRR**, abs/1807.10165, 2018. Disponível em: <<http://arxiv.org/abs/1807.10165>>. Acesso em: 09 mar. 2023.