

Deep Learning-Based Transfer Learning for Classification of Cassava Disease

Ademir G. Costa Junior¹, Fábio S. da Silva¹, Ricardo Rios¹

{agdcj.eng20, fssilva, rrios}@uea.edu.br

¹Escola Superior de Tecnologia – Universidade do Estado do Amazonas (UEA)
Manaus – AM – Brasil

Abstract. *This paper presents a performance comparison among four Convolutional Neural Network architectures (EfficientNet-B3, InceptionV3, ResNet50, and VGG16) for classifying cassava disease images. The images were sourced from an imbalanced dataset from a competition. Appropriate metrics were employed to address class imbalance. The results indicate that EfficientNet-B3 achieved on this task accuracy of 87.7%, precision of 87.8%, revocation of 87.8% and F1-Score of 87.7%. These findings suggest that EfficientNet-B3 could be a valuable tool to support Digital Agriculture.*

Resumo. *Esse artigo apresenta um comparativo de desempenho entre quatro arquiteturas de Redes Neurais Convolucionais (EfficientNet-B3, InceptionV3, ResNet50 e VGG16) na classificação de imagens de patologias da mandioca. As imagens foram obtidas de um conjunto de dados desbalanceado de uma competição. Foram utilizadas métricas adequadas para lidar com o desbalanceamento entre classes do conjunto. Os resultados indicam que a EfficientNet-B3 alcançou nessa tarefa acurácia de 87,7%, precisão de 87,8%, revocação de 87,8% e F1-Score de 87,7%. Isso sugere que o EfficientNet-B3 pode ser uma ferramenta valiosa de apoio para Agricultura Digital.*

1. Introdução

De acordo com a Empresa Brasileira de Pesquisa Agropecuária (EMPRABA), o investimento em ciência ao longo das décadas permitiu que o Brasil se posicionasse como uma importante potência mundial na produção de alimentos [Embrapa 2022]. Para a próxima década, as projeções apontam um significativo potencial de crescimento no setor. Comparado à produção de 2022-2023, espera-se um aumento de 24,1% até 2032-2033, equivalente a um crescimento anual de 2,4% [MAPA 2023]. As projeções para o próximo decênio indicam que o Brasil deve manter ou elevar o padrão que tem sido típico no século atual – aumentos da produção agrícola total superiores ao crescimento da área plantada, o que indica um crescimento da produção sem necessidade de expansão da área plantada.

A produção brasileira de mandioca (*Manihot esculenta Crantz*), por exemplo, em 2022 atingiu 18,2 milhões de toneladas em uma área de 1,27 milhões de hectares. Segundo levantamento do Instituto Brasileiro de Geografia e Estatística (IBGE), esse fato consolidou o Brasil como o quinto maior produtor mundial do tubérculo [Companhia Nacional de Abastecimento 2023]. A mandioca assume um papel fundamental na segurança alimentar das regiões tropicais. Sua rusticidade e adaptabilidade a so-

los ácidos, regimes pluviométricos variados e períodos de seca a tornam ideal para regiões com condições desafiadoras, em que outras culturas podem falhar [FARIAS et al. 2006].

A mandioca é a base alimentar de mais de 800 milhões de pessoas em países em desenvolvimento. Devido a fatores como doenças e pragas, enfrenta diversos desafios que ameaçam a produção e a segurança alimentar [Howeler et al. 2013]. Entre os principais, destacam-se as doenças fúngicas, virais e bacterianas que causam perdas significativas na colheita e na qualidade do tubérculo [Calvert and Thresh 2001].

O diagnóstico preciso e precoce de doenças da mandioca é crucial para o manejo eficaz e a mitigação dos impactos negativos na produção. No entanto, métodos tradicionais de diagnóstico, como a inspeção visual e testes laboratoriais, apresentam diversas limitações como resultados inconsistentes, demora para conclusão do processo e custos elevados [Santos et al. 2020].

Considerando esse cenário de ameaça à produção de mandioca, a Visão Computacional se apresenta como uma ferramenta promissora na detecção e identificação de pragas e doenças. Especificamente no que diz respeito à agricultura digital, ela se destaca na análise de manifestações perceptuais de doenças, como os sintomas visuais causados por esses patógenos. Através do uso de técnicas avançadas de Inteligência Artificial como Aprendizado Profundo (*Deep Learning* – DL), especificamente Redes Neurais Convolucionais (*Convolutional Neural Networks* – CNNs) [LeCun et al. 2015], é possível a identificação precoce e precisa de patologias na plantação de mandioca e o mapeamento da cobertura do solo [Kamilaris and Prenafeta-Boldú 2018]. Isso viabiliza a adoção de ações mais eficazes e direcionadas [Massruhá et al. 2020, Santos et al. 2020].

Nesse contexto, esse artigo apresenta um método de detecção automática de doenças da mandioca, utilizando CNNs para analisar imagens de folhas do tubérculo. São comparadas diferentes arquiteturas de CNNs empregadas na identificação de quatro doenças, a Bacteriose da Mandioca (*Cassava Bacterial Blight* – CBB), a Doença da Estria Marrom (*Cassava Brown Streak Disease* – CBSD), a Doença do Mosaico (*Cassava Mosaic Disease* – CMD) e o Vírus Mosqueado Verde (*Cassava Green Mottle* – CGM). Além disso, é analisado o desempenho da melhor Arquitetura para a detecção dessas doenças individualmente, pois é possível que uma CNN se comporte melhor na detecção da CBB e não da CBSD.

Esse estudo avalia uma metodologia para a classificação de doenças em mandioca, utilizando técnicas de Aprendizado Profundo e Transferência de Aprendizado. O objetivo principal da pesquisa é desenvolver um modelo robusto e eficiente para o diagnóstico precoce de doenças na cultura da mandioca. Para isso, são comparadas diferentes arquiteturas pré-treinadas de CNNs. O modelo desenvolvido será utilizado como suporte à tomada de decisão no manejo adequado da lavoura no que diz respeito à otimização da produtividade e minimização de perdas.

Esse artigo está organizado da seguinte forma. Na Seção 2, são apresentados os artigos relacionados à classificação de doenças em plantas utilizando Redes Neurais Artificiais (*Artificial Neural Networks* – ANNs). Na Seção 3, são apresentadas as configurações experimentais utilizadas, incluindo detalhes sobre o conjunto de dados, modelos, parametrização e métricas de desempenho utilizadas para avaliar os resultados. A Seção 4 aborda os resultados obtidos e a análise comparativa entre os diferentes mode-

los treinados. Por fim, na Seção 5, são apresentadas as considerações finais.

2. Trabalhos Relacionados

Na literatura, vários trabalhos demonstram a eficácia das ANNs na identificação de patologias em diversas espécies de plantas. Por exemplo, o trabalho [Mohanty et al. 2016] investigou o uso de duas arquiteturas de CNNs para classificar 26 doenças em 14 espécies agrícolas, resultando em uma acurácia de 99,35% no melhor modelo. Os dados utilizados foram obtidos do projeto *Plant Village* [Hughes et al. 2015], que disponibilizou um conjunto de 54.306 imagens.

O trabalho [Sangbamrung et al. 2020] propôs uma abordagem inovadora para a classificação automática de doenças da mandioca. Foram utilizados algoritmos de aprendizado profundo. A metodologia abordou o problema como um caso de classificação binária em duas etapas: (i) detecção de doença, diferenciando espécimes saudáveis de espécimes doentes; e (ii) classificação específica da doença, classificando a doença específica como *Doença da Raia Marrom da Mandioca* (CBSD) ou outras doenças.

O trabalho [Sambasivam and Opiyo 2021] introduziu a aplicação de uma CNN para desenvolver um método de aprendizado profundo eficiente e econômico, ou seja, exclui a coleta de amostras e análise laboratorial para detectar infecções nas folhas da mandioca. A abordagem proposta visava resolver um desafio em um conjunto de dados desbalanceado contendo 10.000 imagens classificadas em cinco classes distintas. Para isso, foi utilizada a Técnica de *Oversampling* de Minoria Sintética (*Synthetic Minority Over-sampling Technique* – SMOTE) para tratar o conjunto de dados desbalanceado e empregaram uma arquitetura CNN composta por três camadas convolucionais e quatro camadas totalmente conectadas no treinamento. Os resultados obtidos demonstraram um F1-Score de até 95% na classe com o melhor desempenho.

Embora esses trabalhos demonstrem um desempenho satisfatório na resolução do problema em foco, é perceptível a ausência de uma análise comparativa abrangente entre diferentes arquiteturas de CNNs para essa tarefa específica. Esse trabalho pretende preencher essa lacuna, apresentando uma comparação detalhada das métricas, utilizando as arquiteturas mais amplamente empregadas em publicações recentes.

3. Resultados Experimentais

3.1. Configuração dos Experimentos

Nesse trabalho, o problema de classificação de doenças da folha da mandioca é considerado como uma tarefa de classificação multiclasse. É utilizada a técnica de Aprendizado Supervisionado. Isso significa que um conjunto de dados preexistente, com imagens de folhas de mandioca rotuladas com suas respectivas doenças, foi utilizado para treinar os modelos de classificação.

Para o treinamento das arquiteturas de CNNs, foi utilizado um servidor equipado com um processador Intel(R) Core(TM) i7-8700 CPU @ 3,20 GHz, 32 GB de memória principal, 2,4 TB de memória secundária e duas GPUs NVIDIA GeForce GTX 1080 Ti com 11 GB de VRAM cada. O ambiente de programação consistiu na Linguagem de Programação Python 3.12.2 [Van Rossum and Drake 2009] e no framework PyTorch 2.3.0 [Paszke et al. 2019].

3.2. Conjunto de dados

O conjunto de dados utilizado nessa pesquisa compreende 21.367 imagens de folhas de mandioca, cuidadosamente coletadas durante um estudo de campo realizado em Uganda [Ernest Mwebaze 2020]. A maioria das imagens foi capturada por agricultores locais. As imagens foram posteriormente anotadas por especialistas do Instituto Nacional de Pesquisa e Recursos Culturais (NaCRRI), em colaboração com o laboratório de inteligência artificial da Universidade Makerere, em Kampala. As imagens foram cuidadosamente organizadas em cinco classes distintas, cada uma representando uma doença específica da mandioca: Bacteriose da Mandioca (*Cassava Bacterial Blight* – CBB); Doença da Estria Marrom (*Cassava Brown Streak Disease* – CBSD); Doença do Mosaico (*Cassava Mosaic Disease* – CMD); Vírus Mosqueado Verde (*Cassava Green Mottle* – CGM); e exemplares saudáveis.

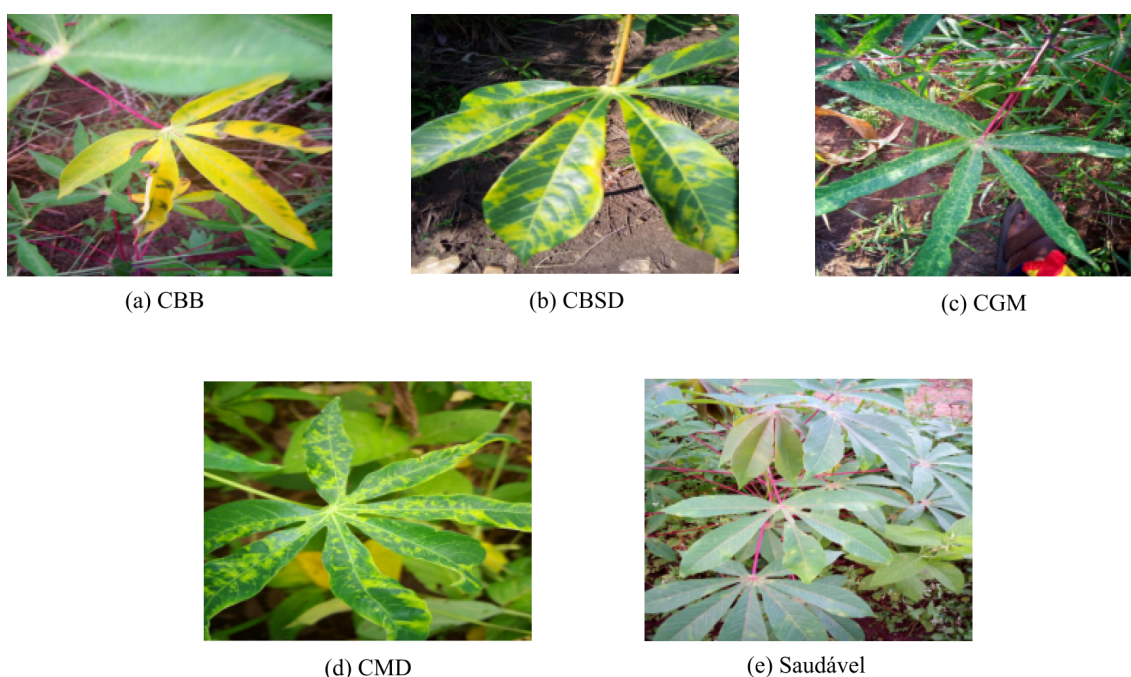


Figura 1. Exemplos aleatórios selecionados para cada classe.

A análise exploratória de dados (EDA) revelou um desbalanceamento significativo entre as classes do conjunto de dados. A classe CMD, por exemplo, apresentou um número de exemplos consideravelmente maior do que a soma de todas as outras classes juntas, sendo responsável por 61.5% do total de amostras. Classes como a CBB representam apenas 5%, conforme ilustrado na Figura 2. Para lidar com esse desafio, nesse trabalho, foram utilizadas métricas específicas para desbalanceamento de classes (Seção 3.7).

3.3. Modelos de CNNs

O método para solução do problema em questão fundamenta-se nos avanços recentes de (*Deep Learning*), nos quais imagens multidimensionais são fornecidas como entrada para CNNs. Essas redes realizam aprendizado automático, massivo e hierárquico dos parâmetros dos filtros extratores de características, capacitando-se, assim, para classificar padrões complexos nos dados de entrada [Goodfellow et al. 2016].

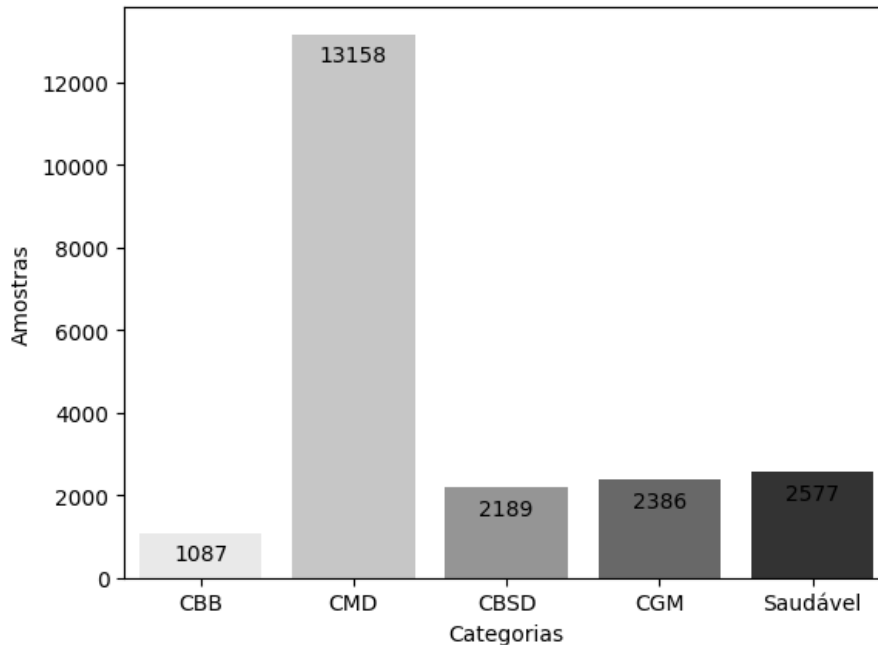


Figura 2. Distribuição de amostras na base de dados

Foram selecionados os seguintes modelos de CNNs para a solução proposta:

1. EfficientNet-B3: proposta por [Tan and Le 2019], é uma CNN que se destaca pela sua eficiência e desempenho. Essa rede utiliza um método de escalamento composto por coeficientes de largura, profundidade e resolução, para construir arquiteturas computacionalmente eficientes. A EfficientNet-B3 apresenta um equilíbrio entre complexidade e precisão, com 12 camadas principais e é particularmente eficaz em tarefas de visão computacional.
2. InceptionV3: proposta por [Szegedy et al. 2016], emprega uma arquitetura modularizada com blocos *Inception*, que consistem em convoluções de diferentes tamanhos de filtro, visando aumentar a eficiência computacional e melhorar a precisão. Com 48 camadas convolucionais, é considerada uma das CNNs mais eficientes para tarefas de classificação de imagens.
3. ResNet50: proposta por [He et al. 2016], é uma CNN profunda que utiliza o conceito de camadas residuais para mitigar o problema do desvanecimento do gradiente durante o treinamento. Composta por 50 camadas convolucionais, é amplamente empregada em tarefas de Visão Computacional.
4. VGG16: proposta por [Simonyan and Zisserman 2014], possui 16 camadas e utiliza filtros convolucionais de tamanho 3x3 para extração de características. Além disso, emprega camadas de *max-pooling* para redução de dimensionalidade, juntamente com camadas finais completamente conectadas para fins de classificação.

3.4. Pré Processamento das Imagens

Como mencionado na Seção 3.2, o conjunto de dados apresenta desbalanceamento entre as classes, com uma classe majoritária predominando significativamente sobre as demais. Para lidar com a situação de classe rara, ou seja, quando uma classe é minoritária, propomos a utilização da estratificação, conforme sugerido por [Amaral 2016]. Essa técnica

consiste em dividir o conjunto de imagens em subconjuntos, preservando a proporção de classes em cada um. Dessa forma, a divisão do conjunto de dados é apresentada da seguinte maneira:

- Treino (70%): utilizado para treinar o modelo.
- Validação (10%): empregada para avaliar o desempenho do modelo durante o treinamento e evitar o *overfitting*.
- Teste (20%): simula um ambiente real de aplicação, fornecendo uma avaliação imparcial do desempenho final do modelo.

Todas as imagens foram redimensionadas para se adequarem às entradas dos diferentes modelos utilizados. Para os modelos ResNet50 e VGG19, as imagens foram ajustadas para 224 x 224 pixels. Para o modelo InceptionV3, as imagens foram redimensionadas para 299 x 299 pixels. Para o modelo EfficientNet-B3, as imagens foram ajustadas para 300 x 300 pixels. Dessa forma, todas as imagens têm as dimensões apropriadas durante o treinamento de cada rede, garantindo a compatibilidade com os respectivos modelos apresentados.

Conjuntamente ao redimensionamento das imagens é aplicado o *Data Augmentation*, uma técnica utilizada em Aprendizado de Máquina e Visão Computacional para aumentar a diversidade dos dados de treinamento. Isso é feito aplicando diversas transformações nos dados originais, criando novas amostras sem a necessidade de coletar dados adicionais [Shorten and Khoshgoftaar 2019].

Para parametrizar o aumento de dados foram realizadas transformações geométricas, de cor e de ruído. Essas transformações incluem normalização dos valores dos pixels, rotação aleatória das imagens, deslocamento horizontal e vertical, transformação de cisalhamento, *zoom* aleatório, inversão horizontal e ajuste de brilho. É importante destacar que essas transformações são aplicadas apenas ao conjunto de treinamento. Nos conjuntos de teste e validação, apenas a normalização dos valores dos pixels é aplicada.

3.5. Transferência de Aprendizado

Treinar uma CNN para uma tarefa específica pode ser um processo computacionalmente caro. Essa tarefa exige grandes conjuntos de dados e poder de processamento significativo. A Transferência de Aprendizado [Pan and Yang 2009] oferece uma solução para contornar essa dificuldade. Essa técnica utiliza um modelo pré-treinado em um problema similar como ponto de partida para o treinamento da rede. Dessa forma, é possível otimizar o treinamento de redes convolucionais, especialmente quando recursos computacionais são limitados.

Para iniciação dos pesos dos modelos foi utilizado o pré-treinamento do ImageNet [Russakovsky et al. 2015], aproveitando os pesos das camadas genéricas para iniciar o treinamento para acelerar o processo de treinamento.

3.6. Construção do Modelo

Para todas as arquiteturas de CNNs selecionadas (Seção 3.3), os pesos foram iniciados conforme descrito na Seção 3.5. O treinamento foi conduzido com um limite máximo de 50 épocas. Adotou-se a técnica de *Early Stopping*, com uma paciência de 5 épocas,

interrompendo o treinamento caso a perda no conjunto de validação não melhore em 5 épocas consecutivas. Além disso, foi empregada a técnica de *Model Checkpoint* para monitorar a perda no conjunto de validação e persistir os pesos que proporcionassem melhor generalização.

O tamanho do lote (*batch size*) foi estabelecido em 32, o que significa que o algoritmo treinou os modelos em blocos de 32 amostras, até utilizar todo o conjunto de dados. Quando todos os blocos são passados pelo modelo, encerra-se uma época.

A taxa de aprendizado inicial foi definida em 10^{-3} com o uso do otimizador Adam [Kingma and Ba 2014], conhecido por sua capacidade de lidar com problemas de convergência em treinamentos de redes neurais profundas e por sua robustez em relação a variações na taxa de aprendizado.

A técnica de identificação de platô (*ReduceLROnPlateau*) foi implementada com o objetivo de otimizar a taxa de aprendizado do modelo. Essa estratégia assegura que, caso o modelo não apresente melhorias em suas métricas (a *loss* do conjunto de validação) por duas épocas consecutivas, a taxa de aprendizado seja automaticamente reduzida em um fator de 0,1. Isso evita a estagnação e promove um treinamento mais eficiente.

Para o cálculo da *loss*, optou-se pelo uso do critério Perda de Entropia Cruzada (*Cross-Entropy Loss*), também conhecido como Perda Logarítmica, disponível no Pytorch. Essa função de perda é usada para encontrar a solução ideal ajustando os pesos de um modelo de aprendizado de máquina durante o treinamento. O objetivo é minimizar o erro entre os resultados reais e previstos. Assim, uma medida mais próxima de 0 (zero) é sinal de um bom modelo, enquanto uma medida mais próxima de 1 (um) é sinal de um modelo de baixo desempenho.

A configuração das redes pré-treinadas é fundamental, especialmente considerando que foram originalmente treinadas no conjunto de dados *ImageNet*, que consiste em 1000 classes de saída possíveis. Para adaptar o modelo à tarefa de classificação de doenças da mandioca, que possui cinco classes de saída, é necessário realizar modificações na arquitetura. Inicialmente, as camadas finais do modelo são removidas para permitir a inclusão de camadas adequadas a essa classificação. Adiciona-se uma camada densa com cinco neurônios e ativação *softmax*, que converte as saídas da camada densa em probabilidades. Cada neurônio na camada de saída representa a probabilidade da imagem pertencer a uma das classes especificadas e considera-se que a classe com maior probabilidade é a predição do modelo.

3.7. Métricas de Avaliação

A avaliação da eficácia dos modelos selecionados requer métricas ponderadas adequadas ao desbalanceamento do conjunto de dados. Para isso, foram selecionadas métricas como acurácia, precisão, revocação e F1-Score, juntamente com a análise da matriz de confusão.

$$\text{Acurácia} = \frac{1}{|N|} \sum_{n \in N} \left(\frac{\text{TP}_n + \text{TN}_n}{\text{TP}_n + \text{TN}_n + \text{FP}_n + \text{FN}_n} \right) \quad (1)$$

$$\text{Precisão} = \frac{1}{|N|} \sum_{n \in N} \left(\frac{\text{TP}_n}{\text{TP}_n + \text{FP}_n} \right) \quad (2)$$

$$\text{Revocação} = \frac{1}{|N|} \sum_{n \in N} \left(\frac{\text{TP}_n}{\text{TP}_n + \text{FN}_n} \right) \quad (3)$$

$$\text{F1-Score} = \frac{2 \cdot \text{Precisão} \cdot \text{Revocação}}{(\text{Precisão} + \text{Revocação})} \quad (4)$$

Nas equações (1)-(3), em que N denota o conjunto de classes do problema, as abreviações representam quatro possíveis resultados em uma análise de classificação binária. Explicitamente, *True Positive* (TP) quantifica as instâncias corretamente identificadas como pertencentes à classe positiva; *True Negative* (TN) reflete as predições corretas para a classe negativa; *False Positive* (FP) representa os casos erroneamente classificados como positivos quando deveriam ser negativos; e *False Negative* (FN) representa os casos em que as predições indicam negatividade equivocadamente, quando deveriam ser positivas.

4. Resultados e Discussão

Os experimentos foram conduzidos de acordo com a metodologia proposta na Seção 3 e os resultados foram registrados na Tabela 1. Nessa seção, são apresentados os resultados e discute-se o uso das estratégias de treinamento empregadas, bem como avalia-se o desempenho da classificação de cada CNN escolhida, conforme as métricas descritas na Seção 3.7.

Para o comparativo principal entre os modelos, utilizou-se a métrica *F1-Score*, pois, além de equilibrar Precisão e Revocação, essa métrica é especialmente útil em cenários com conjuntos de dados desbalanceados, oferecendo uma avaliação mais precisa em situações em que algumas classes são subrepresentadas.

Tabela 1. Resultados obtidos para as Arquiteturas Selecionadas

Arquitetura	Parâmetros (M)	Acurácia	Precisão	Revocação	F1-Score
EfficientNet-B3	12	0.877	0.878	0.877	0.877
InceptionV3	23.8	0.866	0.867	0.866	0.866
ResNet50	25.6	0.863	0.862	0.863	0.862
VGG16	138	0.615	0.378	0.615	0.468

Com exceção da VGG16, todas as arquiteturas alcançaram resultados acima de 86% em todas as métricas calculadas sobre o conjunto de testes. Dentre elas, destaca-se positivamente a EfficientNet-B3, que obteve um *F1-Score* de 87,7%. A VGG16 apresentou um desempenho inferior devido à sua incapacidade de capturar padrões e características relevantes em um conjunto de dados tão desbalanceado. Como resultado, essa arquitetura apresentou um comportamento em que a predição era sempre a classe majoritária para todos os exemplos. Por isso sua acurácia é igual à proporção de casos da classe majoritária (CMD).

Entre os diversos fatores que justificam a ineficiência da VGG16 em comparação com outros modelos, destaca-se sua quantidade excessiva de parâmetros. A VGG16 possui até 10 vezes mais parâmetros do que as melhores arquiteturas, o que aumenta a propensão ao *overfitting*. Por outro lado, modelos como EfficientNet-B3, InceptionV3 e

ResNet50 não apenas possuem menos parâmetros, mas também são projetados para serem mais eficientes computacionalmente, permitindo alcançar melhores resultados com menos recursos. O EfficientNet-B3 incorpora técnicas avançadas de regularização, como o escalonamento composto, que envolve o aumento simultâneo de largura, profundidade e resolução da rede. Isso permite que a rede receba imagens de maior resolução, com mais camadas convolucionais e um número maior de canais em cada camada, capturando características mais complexas dos dados.

Devido aos resultados superiores da arquitetura EfficientNet-B3, decidiu-se explorar mais detalhadamente seus resultados. Observando a Tabela 2 e a Matriz de Confusão apresentada na Figura 3, nota-se que a arquitetura foi capaz de classificar todas as classes com mais de 70% de precisão, exceto a classe CBB. Essa classe tem a menor quantidade de amostras do conjunto de dados, representando cerca de 5% do total de amostras. A classe CBB teve o pior desempenho. Destaca-se a classificação da classe CMD, a classe majoritária do problema, com todas as métricas acima de 95%. Apesar de as classes CBSD, CGM e os exemplares saudáveis também serem classes raras no conjunto, os resultados para elas foram superiores, chegando a atingir até 84.3% de precisão. Isso demonstra que é possível atingir bons resultados apesar do conjunto desbalanceado.

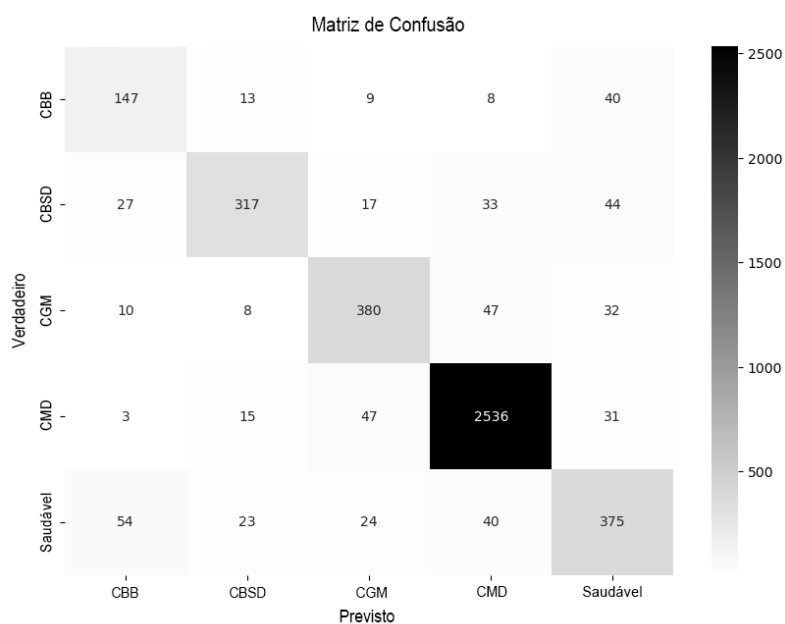


Figura 3. Matriz de Confusão da EfficientNet-B3

Tabela 2. Relatório de Classificação

Classe	Precisão	Revocação	F1-score	Suporte
CBB	0.610	0.677	0.642	217
CBSD	0.843	0.724	0.779	438
CGM	0.797	0.797	0.797	477
CMD	0.952	0.964	0.958	2632
Saudável	0.718	0.727	0.723	516

5. Conclusões

Esse trabalho teve como objetivo realizar um estudo comparativo do desempenho de diferentes arquiteturas de Redes Neurais Convolucionais utilizando a técnica de *Transfer Learning*. O objetivo foi encontrar um modelo eficaz na classificação de patologias da Mandioca, otimizando recursos computacionais e eliminando a necessidade de desenvolvimento de um modelo específico para essa tarefa.

Durante o desenvolvimento desse estudo, foi observado que as Redes Neurais Convolucionais podem ser ferramentas úteis na classificação dessas patologias. Destaca-se o EfficientNet-B3, que alcançou um *F1-Score* de 87,7% e até 95% de acurácia na classe com maior número de acertos, demonstrando sua capacidade de identificar corretamente a maioria das patologias apresentadas.

Para futuros trabalhos, planeja-se explorar o impacto de diferentes hiperparâmetros, como, por exemplo, o otimizador RMSProp [Tieleman and Hinton 2017] ou variantes do Adam. Planeja-se também explorar técnicas avançadas de otimização de hiperparâmetros, como, por exemplo, a busca em larga escala. Além disso, pretende-se avaliar o efeito do balanceamento do conjunto de dados através de técnicas, como replicação de imagens, ou geração de imagens sintéticas para classes minoritárias usando métodos, como SMOTE [Chawla et al. 2002] ou ADASYN [He et al. 2008].

6. Agradecimentos

Os autores agradecem pelo apoio fornecido pelo Laboratório de Sistemas Inteligentes (LSI) da Universidade do Estado do Amazonas (UEA), na pessoa da Professora do Núcleo de Computação (NuComp) da Escola Superior de Tecnologia (EST). Dra. Elloá B. Guedes.

Referências

- [Amaral 2016] Amaral, F. (2016). *Introdução à ciência de dados: mineração de dados e big data*. Alta Books Editora.
- [Calvert and Thresh 2001] Calvert, L. A. and Thresh, J. (2001). The viruses and virus diseases of cassava. In *Cassava: biology, production and utilization*, pages 237–260. CABI Wallingford UK.
- [Chawla et al. 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.
- [Companhia Nacional de Abastecimento 2023] Companhia Nacional de Abastecimento (2023). Análise mensal da produção de mandioca para fevereiro 2023. Technical report, Companhia Nacional de Abastecimento.
- [Embrapa 2022] Embrapa (2022). Ciência e tecnologia tornaram o brasil um dos maiores produtores mundiais de alimentos. <https://www.embrapa.br/busca-de-noticias/-/noticia/75085849/ciencia-e-tecnologia-tornaram-o-brasil-um-dos-maiores-produtores-mundiais-de-alimentos>. Acesso em: 28/02/2024.

- [Ernest Mwebaze 2020] Ernest Mwebaze, Jesse Mostipak, J. J. E. S. D. (2020). Cassava leaf disease classification.
- [FARIAS et al. 2006] FARIAS, A., SOUZA, L., MATTOS, P., et al. (2006). Aspectos socioeconômicos e agronômicos da mandioca. cruz das almas: Embrapa. *Mandioca e Fruticultura Tropical*.
- [Goodfellow et al. 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.
- [He et al. 2008] He, H., Bai, Y., Garcia, E. A., and Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pages 1322–1328. Ieee.
- [He et al. 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- [Howeler et al. 2013] Howeler, R., Lutaladio, E., and Nachury, Z. (2013). *Save and grow: Cassava: A guide to sustainable production intensification*. Food and Agriculture Organization of the United Nations (FAO).
- [Hughes et al. 2015] Hughes, D., Salathé, M., et al. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*.
- [Kamilaris and Prenafeta-Boldú 2018] Kamilaris, A. and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and electronics in agriculture*, 147:70–90.
- [Kingma and Ba 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- [LeCun et al. 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.
- [MAPA 2023] MAPA (2023). Projeções do agronegócio 2022-2023 a 2032-2033. <https://www.gov.br/agricultura/pt-br/assuntos/politica-agricola/todas-publicacoes-de-politica-agricola/projecoes-do-agronegocio/projecoes-do-agronegocio-2022-2023-a-2032-2033.pdf>. Acesso em: 28/02/2024.
- [Massruhá et al. 2020] Massruhá, S. M. F. S., LEITE, M. d. A., OLIVEIRA, S. d. M., Meira, C. A. A., Luchiari Junior, A., and Bolfe, E. (2020). Agricultura digital: pesquisa, desenvolvimento e inovação nas cadeias produtivas.
- [Mohanty et al. 2016] Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7:215232.
- [Pan and Yang 2009] Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.
- [Paszke et al. 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E.,

- DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.
- [Russakovsky et al. 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252.
- [Sambasivam and Opiyo 2021] Sambasivam, G. and Opiyo, G. D. (2021). A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egyptian informatics journal*, 22(1):27–34.
- [Sangbamrung et al. 2020] Sangbamrung, I., Praneetpholkrang, P., and Kanjanawattana, S. (2020). A novel automatic method for cassava disease classification using deep learning. *Journal of Advances in Information Technology Vol*, 11(4):241–248.
- [Santos et al. 2020] Santos, T. T., Barbedo, J. G. A., Ternes, S., Camargo Neto, J., Koenigkan, L. V., and de Souza, K. X. S. (2020). Visão computacional aplicada na agricultura.
- [Shorten and Khoshgoftaar 2019] Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.
- [Simonyan and Zisserman 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [Szegedy et al. 2016] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.
- [Tan and Le 2019] Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.
- [Tieleman and Hinton 2017] Tieleman, T. and Hinton, G. (2017). Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning. *Technical report*.
- [Van Rossum and Drake 2009] Van Rossum, G. and Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.