

Identificação e classificação de doenças da folha do milho através do uso de Rede Neural Convolucional

Lucas Paschoalick
Engenharia de Computação
Universidade Tecnológica Federal do Paraná
Cornélio Procópio, Paraná, Brasil
paschoalick@alunos.utfpr.edu.br

Luiz Felipe Alves Ferreira
Engenharia de Computação
Universidade Tecnológica Federal do Paraná
Cornélio Procópio, Paraná, Brasil
lferreira.2000@alunos.utfpr.edu.br

Vítor Ângelo Misciato Teixeira
Engenharia de Computação
Universidade Tecnológica Federal do Paraná
Cornélio Procópio, Paraná, Brasil
vitor.2018@alunos.utfpr.edu.br

Resumo—O presente trabalho é resultado da tentativa de classificar possíveis doenças presentes nas folhas do milho através do uso de Rede Neural Convolucional (CNN). Para a tarefa, foi utilizado um conjunto de imagens dividido em quatro classes, obtido na plataforma online de ciência de dados *Kaggle*. Na construção dos modelos, foi utilizada a técnica de *transfer learning*, testando-se três arquiteturas disponíveis, variando também nas construções o valor de *dropout*. No melhor caso, a rede criada com uso da *ResNet50* e 0.8 de *dropout* atingiu uma acurácia de 91.96% nos testes.

Palavras-chave—Milho, Rede Neural Convolucional, *Transfer Learning*.

I. INTRODUÇÃO E MOTIVAÇÃO

O milho é um dos principais alimentos cultivados ao redor do mundo, servindo como base para produção de alimentos tanto para animais, quanto para nós seres humanos. Segundo o relatório Produção Agrícola Mundial, do Departamento de Agricultura dos Estados Unidos (USDA), a expectativa de produção anual mundial para 2021/2022 é de mais de 1 bilhão de toneladas [1]. Apesar da elevada produção, um dos principais fatores que impacta negativamente e impede que ela seja ainda maior é ocorrência de doenças que afetam as plantações [2], como é o caso das doenças que afetam as folhas das plantas.

Identificar se uma planta possui algum tipo de doença e classificá-la é um trabalho muito custoso em termos de esforço e tempo quando feito manualmente, sendo inviável analisar um grande volume de folhas uma por uma, além de exigir que o trabalho seja feito por um profissional capacitado, o que demanda mais recursos financeiros. Pensando nisso, com os avanços no campo da inteligência artificial e com o aumento do poder de processamento dos computadores, somados ao grande volume de dados disponível, uma série de trabalhos foram e continuam sendo desenvolvidos no campo de detecção de doenças em plantas através do uso de processamento de imagens. Tais trabalhos utilizam técnicas e algoritmos como *k-Nearest Neighbor* (KNN) [3], *Support Vector Machine* (SVM)

[4], Árvore de decisão [5] e redes neurais, especificamente *Convolutional Neural Network* (CNN) [6].

Com base nessas informações e levando em consideração que o Brasil é um dos principais produtores mundiais, a ideia neste trabalho é construir e treinar um modelo baseado em CNN para automatizar e facilitar o processo de identificação de possíveis doenças presentes nas folhas da cultura do milho, explorando também o impacto das técnicas de pré-processamento de imagens e parâmetros do modelo no desempenho do classificador.

II. OBJETIVOS

O objetivo principal deste trabalho é construir um modelo de classificação de doenças que afetam as folhas do milho baseado em CNN, treinado a partir de um conjunto de imagens previamente rotulado. Além disso, também identificar através de testes comparativos o impacto da aplicação de técnicas de pré-processamento e valores dos parâmetros no desempenho do modelo e da previsão.

III. TRABALHOS CORRELATOS

Nesta seção, o objetivo é apresentar alguns dos inúmeros trabalhos desenvolvidos por outros pesquisadores na classificação e identificação de doenças nas folhas de plantas, utilizando diferentes abordagens e algoritmos.

Em [6], os autores propuseram um modelo de CNN profunda para classificar doenças de folhas de diferentes tipos de plantas. A rede criada possui 9 camadas, sendo que os testes foram realizados nos casos em que as imagens passaram pelo processo de *image augmentation* e em seu estado original. Modificando os hiperparâmetros do modelo e comparando com outros tipos de algoritmos, atingiram uma acurácia média de 96.46% no melhor caso.

Explorando o uso do algoritmo KNN, em [3], os autores buscaram classificar as doenças presentes nas folhas da planta do amendoim. Para isso, no modelo que propuseram, a imagem era pré-processada e segmentada, extraindo como principal

característica os padrões de cores RGB, posteriormente convertidos para HSV, que serviam de entrada para classificação da imagem levando em conta o valor K de vizinhos.

Em [4], buscou-se classificar as doenças presentes nas folhas da planta do tomate através do uso da técnica de SVM. Após o pré-processamento das imagens, como redução de escala e *masking*, foram extraídas 60 características que serviram de entrada para o modelo, que atingiu uma acurácia de 87.60%.

[5] exploraram o uso de árvores de decisão em cinco tipos diferentes de plantas. A proposta deles foi de primeiramente classificar uma folha como saudável ou não, e, caso não fosse, identificar qual a doença e apresentar alguma sugestão de tratamento. No treinamento do modelo, foi gerada uma árvore com 1000 folhas, atingindo uma acurácia média de 95.25% nos testes realizados.

IV. METODOLOGIA EXPERIMENTAL

A. Aquisição dos dados

Para realização deste trabalho, como conjunto de dados optou-se por utilizar um conjunto de imagens de folhas do milho disponível no *Kaggle* ([link](#) para o conjunto), que é uma plataforma *online* de ciência de dados. Tal conjunto foi criado a partir de outros dois *datasets* já disponíveis, o *PlantVillage* [7] e o *PlantDoc* [8], totalizando 4188 imagens divididas em quatro categorias de estado das folhas:

- Ferrugem comum – 1306 imagens;
- Manchas cinzas – 574 imagens;
- Praga – 1146 imagens;
- Saudáveis – 1162 imagens.

B. Pré-processamento

Com o conjunto de dados definido, a primeira etapa consistiu em separar as imagens para treino e para teste. O principal objetivo de dividir o conjunto é evitar uma situação conhecida como *overfitting*, que é quando o modelo aprende tanto sobre os dados, que acaba na verdade memorizando os padrões. Outra justificativa é para poder avaliar o modelo com valores que ele não viu durante o treinamento. Assim, foram separadas 30% das imagens para testes e 70% para treino, dividindo equivalentemente as quatro classes.

Além dessa divisão, o conjunto de treinamento também foi dividido antes da criação e treinamento das redes, sendo que 30% também dessas imagens foram dedicadas a validação, que é uma maneira do modelo avaliar seus resultados enquanto treina.

No que diz respeito especificamente à pré-processamento das imagens, todas foram redimensionadas para 224x224 (resolução da imagem). Para as imagens que de fato servem para treinamento, foram aplicados procedimentos de *data augmentation*, produzindo imagens artificiais com base nas imagens reais, sendo utilizadas as técnicas de *random flip*, *random rotation* e *random zoom*.

C. Estrutura das redes

Para construção dos modelos, foi utilizada a técnica de *transfer learning*, que consiste em inserir no modelo alguma

arquitetura já conhecida disponível que seja genérica, treinada geralmente em algum conjunto de dados muito grande e com diversas classes. Para este trabalho, as três arquiteturas escolhidas foram a *EfficientNetB3*, a *ResNet50* e a *MobileNetV2*.

Assim, cada uma das arquiteturas ficou responsável por extrair as principais características (*features*) das imagens, ou seja, elas não são efetivamente treinadas a cada cada passagem. Além da arquitetura, também foram adicionadas à cada modelo uma camada de *Polling* e uma camada densa, que são as que de fato fazem a classificação.

Além desses aspectos, todos os modelos foram compilados utilizando como otimizador o algoritmo Adam e a função *Cross Entropy* como *loss function*.

D. Cenários de treinamento e métricas de avaliação

Para cada uma das arquiteturas, foram treinados três modelos diferentes durante 10 épocas, com base na aplicação do valor de *dropout*. Essa técnica de regularização desativa, aleatoriamente, uma quantidade pré-definida (porcentagem representada por um valor de 0 a 1) de nós (neurônios) da rede, atualizando, assim, apenas uma parte dos pesos a cada passagem de acordo com o erro obtido propagado. Os valores de *dropout* escolhidos foram 0, 0.2 e 0.8.

Como métricas de avaliação nos testes, foram escolhidas as medidas estatísticas de:

- **Acurácia:** mede o total de imagens classificadas corretamente;
- **Precisão:** das imagens que receberam uma classe específica, indica quantas delas realmente estão corretas;
- **Recall:** das imagens que compõem uma classe específica, mostra quantas foram classificadas corretamente;
- **F1-Score:** uma média harmônica de precisão e *recall* para obter um único resultado, ponderando para o pior valor.

V. RESULTADOS E DISCUSSÕES

Nas figuras 1, 2 e 3, apresentadas a seguir, estão os gráficos que apresentam o comportamento das acurácias de treinamento e validação dos modelos ao longo das épocas. Em cada um deles, as curvas 1, 2 e 3 representam os casos com *dropout* de 0, 0.2 e 0.8, respectivamente.

Como pode ser notado, no uso das três arquiteturas, no caso em que o valor de *dropout* é igual a 0 o treinamento teve maior assertividade ao classificar as imagens. Ademais, também percebe-se que conforme o valor aumenta, a acurácia diminui, sendo mais nítido o impacto na acurácia de treino.

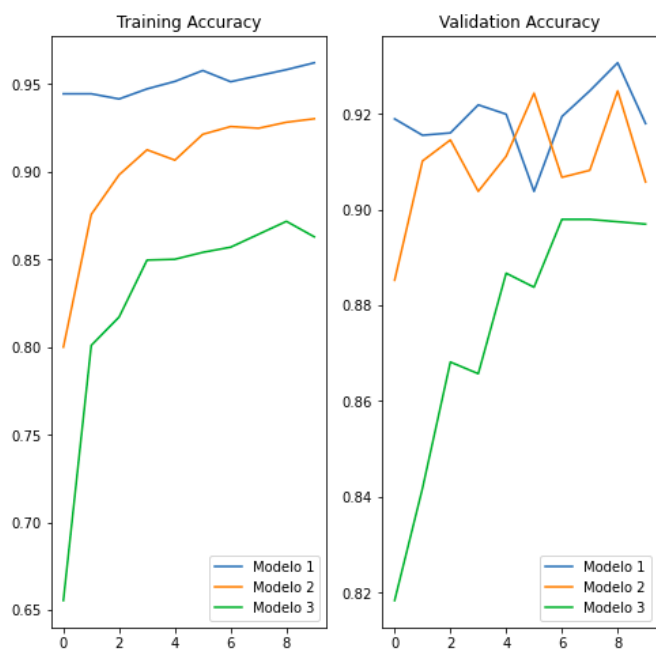


Figura 1. Acurácias de treinamento - *EfficientNetB3*.

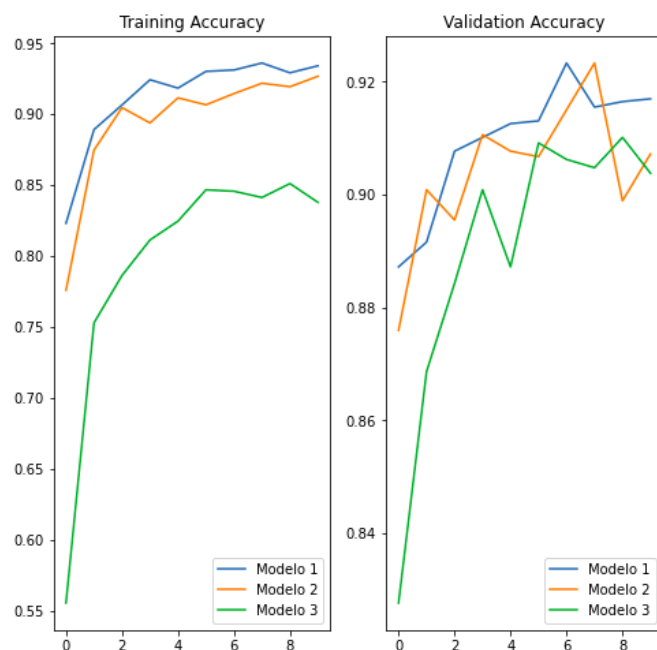


Figura 3. Acurácias de treinamento - *MobileNetV2*.

Na tabela I estão os valores das métricas estatísticas obtidos a partir da aplicação dos modelos na classificação das imagens do conjunto de testes. As medidas de precisão, *recall* e *f1-score* apresentadas são as médias entre os valores atingidos para cada classe.

Assim como nos resultados do treinamento, as acurácias diminuíram conforme foi aumentado o valor de *dropout*, exceto para o caso em que o valor de 0.8 para a *ResNet50* foi maior, sendo inclusive a maior taxa de assertividade entre os modelos.

Outro aspecto importante está nos valores de *recall*, que se mantêm em média próximo aos 80%, enquanto os valores de precisão se aproximam de 90%. O fator agravante para que isso acontecesse foi o desbalanceamento do conjunto, causado pela baixa quantidade de imagens do grupo das Manchas Cinzas. Sendo assim, os modelos, por terem uma quantidade de amostras menor, tiveram mais dificuldade em classificar essa doença, classificando-a com frequência como outros tipos.

Tabela I
MÉTRICAS OBTIDAS A PARTIR DA CLASSIFICAÇÃO DO CONJUNTO DE TESTES.

Arquitetura	Dropout	Métrica (%)			
		Acurácia	Precisão	Recall	F1-Score
<i>EfficientNetB3</i>	0	90.21	90.62	86.47	87.68
	0.2	88.14	90.69	81.97	83.15
	0.8	86.95	88.05	81.61	82.84
<i>ResNet50</i>	0	91.80	93.06	86.65	88.01
	0.2	87.27	90.85	79.30	79.31
	0.8	91.96	92.62	87.55	88.87
<i>MobileNetV2</i>	0	89.73	90.87	83.87	85.03
	0.2	89.33	90.60	83.44	84.60
	0.8	89.26	90.15	82.77	84.80



Figura 2. Acurácias de treinamento - *ResNet50*.

VI. CONCLUSÕES E TRABALHOS FUTUROS

De acordo com o que foi proposto inicialmente como objetivos do trabalho, pode-se dizer que foram atingidos parcialmente. Apesar dos testes serem bem sucedidos, não foi possível testar mais casos, como variações de parâmetros como *batch size* e otimizador, muito em função de limitações de *hardware*, visto que modelos de redes neurais, principalmente na classificação de imagens, utilizam muitos recursos de processamento.

Porém, com os casos trabalhados foi possível atingir uma boa taxa de assertividade, sendo o melhor caso com uso da *ResNet50*, com 91.96% de acurácia. Quando comparado com o valor obtido por [6], apesar de ainda ser mais baixo, percebe-se que trabalhando as possibilidades pode ser atingida uma qualidade equivalente.

Como propostas de trabalhos futuros, podem ser citadas como extensões trabalhar no balanceamento das classes, melhorar o procedimento de *data augmentation* e explorar mais técnicas de redes, como o uso de *fine tuning*.

REFERÊNCIAS

- [1] United States Department of Agriculture (USDA). (2021). World Agricultural Production. Disponível em: <<https://apps.fas.usda.gov/psdonline/circulars/production.pdf>>.
- [2] Mueller, D. S., Wise, K. A., Sisson, A. J., Allen, T. W., Bergstrom, G. C., Bissonnette, K. M., ... Wiebold, W. J. (2020). Corn Yield Loss Estimates Due to Diseases in the United States and Ontario, Canada, from 2016 to 2019. *Plant Health Progress*, 238–247. doi:10.1094/php-05-20-0038-rs.
- [3] Vaishnav, M. P., Devi, K. S., Srinivasan, P., ArutPerumJothi, G. (2019). Detection and Classification of Groundnut Leaf Diseases using KNN classifier. 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN). doi:10.1109/icscan.2019.8878733.
- [4] Das, D., Singh, M., Mohanty, S. S., Chakravarty, S. (2020). Leaf Disease Detection using Support Vector Machine. 2020 International Conference on Communication and Signal Processing (ICCSP). doi:10.1109/ICCSP48568.2020.9182128.
- [5] B. Rajesh, M. V. Sai Vardhan and L. Sujihelen, "Leaf Disease Detection and Classification by Decision Tree" 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 2020, pp. 705-708, doi: 10.1109/ICOEI48184.2020.9142988.
- [6] G., G., J., A. P. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering*, 76, 323–338. doi:10.1016/j.compeleceng.2019.04.011.
- [7] HUGHES, D. P.; SALATHE, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. [S.l.: s.n.], 2016. arXiv:1511.08060 [cs.CY].
- [8] SINGH, D. et al. PlantDoc: A Dataset for Visual Plant Disease Detection. In: PROCEEDINGS of the 7th ACM IKDD CoDS and 25th COMAD. Hyderabad, India: Association for Computing Machinery, 2020. (CoDS COMAD 2020), p. 249–253. ISBN 9781450377386. DOI:10.1145/3371158.3371196. Disponível em: <<https://doi.org/10.1145/3371158.3371196>>.