

# Aplicação de redes neurais convolucionais na classificação de doenças de plantas

Vanessa Rezende  
Instituto de Tecnologia  
Universidade Federal do Pará  
Marabá, Brasil  
vanessarezende@unifesspa.edu.br

Adam Santos  
Instituto de Geociências e Engenharias  
Universidade Federal do Sul e Sudeste do Pará  
Marabá, Brazil  
adamdreyton@unifesspa.edu.br

Michel Costa  
Instituto de Geociências e Engenharias  
Universidade Federal do Sul e Sudeste do Pará  
Marabá, Brazil  
michelvictor@unifesspa.edu.br

Roberto C. L. de Oliveira  
Instituto de Tecnologia  
Universidade Federal do Pará  
Belém, Brasil  
limao@ufpa.br

**Resumo**—As redes neurais convolucionais (RNCs) são uma das técnicas mais promissoras da área de visão computacional, podendo gerar ganhos substanciais nos mais variados problemas de classificação, principalmente aqueles que envolvem imagens digitais. Nesse contexto, este trabalho tem como objetivo aplicar as arquiteturas de RNCs pré-treinadas VGG16, VGG19, DenseNet121 e DenseNet169 na tarefa de classificação de múltiplas patologias referentes a diversas espécies de plantas da base de dados composta de imagens de doenças em plantas XDB. Para este estudo, foram geradas cinco novas bases, a partir da XDB, com dimensões de  $32 \times 32$ ,  $64 \times 64$  e  $100 \times 100$ , variando entre 50 e 66 classes com maior representatividade. Um estudo comparativo foi conduzido com base em métricas de classificação amplamente utilizadas na área de aprendizagem profunda, como acurácia no teste, *f1-score* e área sob a curva. A fim de atestar a significância dos resultados obtidos, foi realizado o teste estatístico não-paramétrico de Friedman e dois procedimentos *post-hoc*, que demonstraram que as DenseNets obtiveram resultados superiores quando comparadas as VGGS; com maior destaque para a DenseNet169 e destaque negativo para a VGG19.

**Palavras-chave**—Rede neural convolucional, doenças em plantas, XDB, VGG, DenseNet.

## I. INTRODUÇÃO

Nas últimas décadas, os investimentos maciços na mecanização do campo, no melhoramento genético de sementes e nos avanços tecnológicos aplicados a fertilização do solo, ocasionaram a elevação da produtividade mundial do setor agrícola [1]. Seguindo essa tendência, a agricultura brasileira enfrentou uma série de mudanças significativas com relação a suas estruturas de produção, suprimento de insumos, comercialização e distribuição interna e externa, desempenhando um papel fundamental na economia do país [2].

Contudo, um dos maiores desafios desse setor está relacionado com a incidência de doenças em plantas que representam uma ameaça significativa à segurança alimentar global, comprometendo as lavouras nacionais e mundiais, além de ocasionarem grandes perdas de lucro para o setor. Os efeitos dessas desordens variam de sintomas leves a catástrofes nas

quais grandes áreas plantadas para a produção de alimentos são destruídas [3].

Doenças de plantas são anormalidades provocadas geralmente por microrganismos, como bactérias, fungos, nematóides e vírus, além da falta ou excesso de fatores essenciais para o crescimento das plantas, tais como nutrientes, água e luz, conhecidos como distúrbios fisiológicos [4].

Em termos biológicos, essas doenças são difíceis de controlar porque suas populações são variáveis no tempo, espaço e genótipo [3]. Normalmente, essas patologias danificam o aparelho fotossintético e afetam o crescimento da planta. Cerca de 85% das doenças de plantas são causadas por organismos semelhantes a fungos. Outras doenças graves das plantas são causadas por bactérias, vírus e viróides, e poucas doenças são causadas por certos nematóides [5], [6].

A identificação rápida e precisa do organismo causal, assim como estimativas precisas da gravidade da doença, seu efeito e a identificação de seus mecanismos de virulência, fazem com que seja possível tomar decisões que possam minimizar a doença a partir da redução do inóculo do agente patogênico, inibindo seus mecanismos de virulência, além de promover a diversidade genética na cultura. Atualmente, os métodos de diagnóstico precoce envolvem a combinação dos métodos humanos com os métodos tecnológicos ou somente a utilização do último. Essa mudança se deu pelos avanços tecnológicos e a redução de custos no campo da aquisição de imagens digitais, possibilitando a introdução de uma série de métodos de diagnóstico de doenças em plantas baseados na classificação de imagens [3].

Nesse cenário, sistemas de inteligência computacional podem fornecer ferramentas alternativas e confiáveis para identificação de padrões, reduzindo o número de identificações de rotina realizadas por especialistas. Uma técnica que tem se mostrado promissora na resolução desse tipo de problema é a aprendizagem profunda (AP) aplicada na classificação de imagens [7]. O objetivo da classificação de imagens através de

técnicas de AP é classificar uma imagem específica de acordo com um conjunto de categorias possíveis. Os classificadores de imagens geralmente resultam de abordagens baseadas em redes neurais pré-treinadas em domínios diversos.

Dentre as técnicas de AP que são amplamente aplicadas ao reconhecimento de imagens, as redes neurais convolucionais (RNCs) comprovadamente já possuem resultados confiáveis em diversos trabalhos da literatura que envolveram a classificação e detecção de imagens. A aplicação de tais modelos vêm sendo impulsionada pelos atuais avanços na capacidade de processamento dos computadores, principalmente com relação ao uso de unidades de processamento gráfico (GPU) [8].

As RNCs constituem uma das arquiteturas mais utilizadas na área de AP supervisionado, atuando de forma que as características dos dados possam ser exploradas de forma automática para obtenção dos melhores resultados. Esse tipo de rede pode ser resumido basicamente em um modelo que toma como base a integração da visão biológica com o sistema neural. Pesquisadores descrevem a RNC como uma arquitetura complexa que leva consideravelmente mais tempo na etapa de treinamento. Apesar disso, essas redes possuem um alto poder de precisão nas tarefas de classificação, além de uma alta taxa de reconhecimento de objetos, o que justifica seu uso em diversas áreas, como no reconhecimento de doenças em espécies de plantas [6].

O trabalho de Lu *et al.* [9] propôs um novo método de diagnóstico de doenças do arroz, baseado na utilização de RNCs que foram treinadas para identificar 10 doenças comuns relacionadas a essa planta em um conjunto de 500 imagens naturais referentes às folhas e hastes doentes e saudáveis. O modelo proposto atingiu uma acurácia de 95,48%.

A pesquisa de Liu *et al.* [10] apresentou uma abordagem de identificação de doenças da folha de maçã, utilizando uma RNC baseada no modelo de rede AlexNet em um conjunto de dados de 13.689 imagens. Os resultados apontam que a variante da AlexNet obteve uma acurácia geral de 97,62%. O estudo de Sladojevic *et al.* [11] desenvolveu uma RNC pré-treinada na base de dados ImageNet para o reconhecimento de doenças de plantas. As imagens foram agrupadas em 15 classes diferentes. Os resultados experimentais alcançados por esse modelo obtiveram precisão entre 91% e 98%.

O presente trabalho tem como foco a aplicação das arquiteturas de redes convolucionais VGG16, VGG19, DenseNet121 e DenseNet169 com o objetivo de classificar entre 50 e 66 doenças de múltiplas espécies de plantas presentes na base de dados de doenças de plantas XDB, disponibilizada pela Embrapa [12]. Foram criadas cinco bases de dados com diferentes números de classes e distintas resoluções de imagens. A avaliação dos resultados foi baseada primariamente em métricas amplamente difundidas, como acurácia na etapa de teste, *f1-score* e área sob a curva (AUC). Por completude, considerou-se um estudo estatístico não-paramétrico a fim de verificar a significância dos resultados obtidos, tendo como entrada a primeira métrica supracitada.

Este trabalho está organizado da seguinte forma. A seção

II aborda os principais conceitos acerca das arquiteturas de RNC que serão utilizadas nesse estudo. A seção III apresenta os detalhes relacionados com ao processo de classificação de imagens de doenças em plantas, fornecendo informações relacionadas à obtenção das imagens e o processamento nelas realizado, além de apresentar as configurações utilizadas nas arquiteturas de RNCs. A seção IV discute os resultados obtidos. As considerações finais e os trabalhos futuros são relatados na seção V.

## II. REDES NEURAIS CONVOLUCIONAIS

A RNC é uma arquitetura de AP inspirada em regiões específicas do mecanismo de percepção visual dos animais, denominadas campos receptores, os quais são responsáveis pela ativação de diferentes neurônios durante a captação visual de uma imagem em um determinado ambiente [13], [14]. Baseado nesse comportamento cerebral, diversos sistemas foram desenvolvidos para a extração de características específicas a partir de dados de entrada de um determinado problema.

Em geral, o funcionamento das RNCs envolve diversas etapas de processamento de informação, definidas como camadas convolucionais, que utilizam a operação de convolução e ativação (e.g., ReLu) para realizar a extração de características a partir dos dados da camada de entrada (pixels da imagem) através de uma janela de dados ou filtro convolutivo seguido por uma função de ativação não-linear. Esse filtro percorre a entrada da RNC e busca atuar de forma similar à sobreposição dos campos receptores biológicos. Durante o processo de treinamento da RNC, os filtros convolutivos têm seus valores ajustados para que sejam capazes de extrair diferentes características que, posteriormente, se convertem em entradas para outras camadas que realizam, por exemplo, a tarefa de classificação [7], [14].

Entre as camadas de convolução é comum a realização de diversas operações adicionais, como *pooling* (agrupamento), normalização e *padding*, que são responsáveis por reduzir a dimensionalidade espacial das representações, gerando mapas de características com diversos atributos específicos, tais como bordas, intensidade de cor, contornos e formas, que contém um conjunto de pesos compartilhados e podem reduzir relativamente à complexidade computacional da RNC. Ao final das camadas convolutivas, geralmente é utilizada uma sequência de camadas de neurônios totalmente conectados (FC) a todas as ativações das camadas anteriores de forma análoga às camadas das redes neurais tradicionais [14], [15].

### A. VGG

A arquitetura de RNC denominada VGG foi criada com o objetivo de investigar o efeito da adição de profundidade em uma rede convolucional, avaliando sua precisão quando aplicada ao reconhecimento de imagens em larga escala [16]. Especificamente, a arquitetura da VGG16 envolve cerca de 138 milhões de parâmetros, enquanto a VGG19 possui aproximadamente 144 milhões, distribuídos ao longo de suas camadas.

Originalmente, a imagem de entrada possui um tamanho pré-definido de  $224 \times 224$ , sendo em seguida passada através de uma pilha de camadas convolucionais com filtros cujo tamanho do campo receptivo corresponde a  $3 \times 3$ , todas seguidas de camadas convolucionais de tamanho  $2 \times 2$  que são responsáveis por realizar um agrupamento espacial, além de três camadas totalmente conectadas, que representam cerca de 89% de todos os parâmetros do modelo, além da camada de saída *softmax* [17]. A função de ativação ReLu é aplicada em todas as camadas ocultas e o modelo utiliza a regularização *dropout* nas camadas completamente conectadas.

A VGG foi a primeira rede da categoria a utilizar filtros pequenos ( $3 \times 3$ ) em cada camada convolucional. Isso era contrário aos princípios das suas antecessoras, LeNet e AlexNet, onde filtros grandes ( $9 \times 9$  e  $11 \times 11$ ) eram utilizados para capturar características similares na imagem. A grande contribuição da VGG foi a ideia de que múltiplas convoluções em sequência poderiam substituir os efeitos de filtros de máscaras maiores que resultavam em maior custo computacional [18]. Uma das principais vantagens desse modelo de RNC é o fato de ser uma arquitetura uniforme com seus pesos pré-treinados na base de dados ImageNet [19] disponíveis publicamente, tendo sido utilizada em muitas outras aplicações e desafios como um eficiente extrator de características. A Figura 1 apresenta o mecanismo de funcionamento dessa arquitetura na versão VGG16.

### B. DenseNet

Estudos recentes mostraram que as redes convolucionais podem ser substancialmente mais profundas, mais precisas e eficientes de treinar se possuírem conexões mais curtas entre as camadas próximas à entrada e as camadas próximas à saída. A DenseNet utiliza um padrão de conectividade simples, o qual conecta cada camada a todas as outras camadas da rede com o objetivo de garantir o máximo fluxo de informações entre as camadas. Logo, cada camada obtém entradas adicionais de todas as camadas anteriores e passa seus próprios mapas de características para todas as camadas subsequentes, preservando a natureza *feedforward* [20]. A arquitetura dessa rede é resumida através da Figura 2.

As principais vantagens percebidas na literatura acerca da utilização das DenseNets aplicadas na tarefas de reconhecimento de padrões incluem a redução do problema do gradiente de desaparecimento, o aumento na propagação e reutilização de características e a redução significativa da quantidade de parâmetros, pois não há necessidade de reaprender mapas de características redundantes, exigindo menos computação para obter um alto desempenho.

Além de melhorar a eficiência de parâmetros, uma grande vantagem desse modelo é o fluxo aprimorado de informações e gradientes em toda a rede, o que facilita o treinamento. Arquiteturas nas quais cada camada tem acesso direto aos gradientes da função de perda e do sinal de entrada original levam a uma supervisão profunda implícita [21]. Isso ajuda no treinamento de arquiteturas de rede mais profundas.

As camadas da DenseNet são muito estreitas (por exemplo, 12 filtros por camada), adicionando apenas um pequeno conjunto de mapas de características ao “conhecimento coletivo” da rede e mantendo os demais mapas de características inalterados. O classificador final toma uma decisão baseada em todas as características mapeadas pela rede [20].

As variações dessa arquitetura possuem versões de 121 e 169 camadas, nas quais cada uma utiliza aproximadamente 8 e 14 milhões de parâmetros, respectivamente [20]. Logo, apesar desses modelos serem consideravelmente mais profundos que algumas arquiteturas de RNC tradicionais, utilizam uma quantidade reduzida de parâmetros, fazendo com que as DenseNets se destaquem pela simplicidade arquitetural e pela redução de custo computacional.

## III. CLASSIFICAÇÃO DE DOENÇAS DE PLANTAS

### A. Base de dados de doenças de plantas XDB

Devido a importância da saúde alimentar brasileira, assim como a renda gerada pela exportação de alimentos e a agricultura familiar, surgiu a necessidade em diagnosticar doenças em plantas para reduzir perdas e manter os lucros do setor que é comprovadamente essencial para a economia. Uma das soluções desenvolvidas pela Embrapa para facilitar o diagnóstico precoce das patologias que acometem essas culturas foi a criação de uma base de dados de imagens digitais dessas doenças. A base de imagens, denominada de PDDB (Base de Imagens de Sintomas de Doenças de Plantas), foi construída ao longo de quatro anos. Como resultado, foram coletadas 2.326 imagens de 171 doenças e outras desordens acometendo 21 espécies de plantas [12].

Entretanto, a base PDDB não possui uma quantidade de imagens que viabilize a aplicação de técnicas de AP para a extração de características e classificação das doenças, uma vez que boa parte das classes não possui representatividade [12]. Baseado nesse problema, a fim de aumentar o tamanho da base de dados, as imagens foram divididas em imagens menores contendo lesões individuais ou regiões de sintomas, de acordo com alguns critérios, como a extração de folhas de plantas doentes. Como resultado, essa nova base de imagens expandida (XDB) atualmente contém 45.051 imagens e será a base de dados objeto de estudo deste trabalho.

A base de dados XDB original passou então por etapas de pré-processamento que buscavam gerar novas bases de dados, selecionando e redimensionando imagens de acordo com um fator de proximidade da dimensão alvo e um grau de representatividade de imagens pertencentes as classes. Dessa forma, especificamente para este trabalho foram criadas cinco bases de dados diferentes. A Tabela I apresenta as configurações das bases utilizadas. Devido o fato das imagens da base de dados XDB serem imagens menores que representam de forma mais eficaz a doença, as bases de dados com imagens de dimensão  $32 \times 32$  possuem um número maior de imagens, como esperado. Por outro lado, a base de dados com imagens de dimensão  $100 \times 100$  é aquela que agrupa o menor número de imagens.

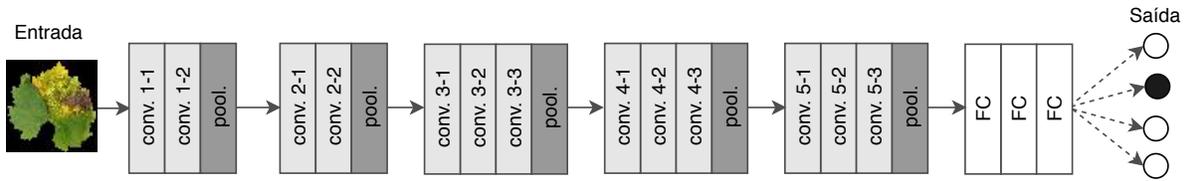


Fig. 1. Arquitetura VGG16 aplicada ao problema de classificação de doenças em plantas.

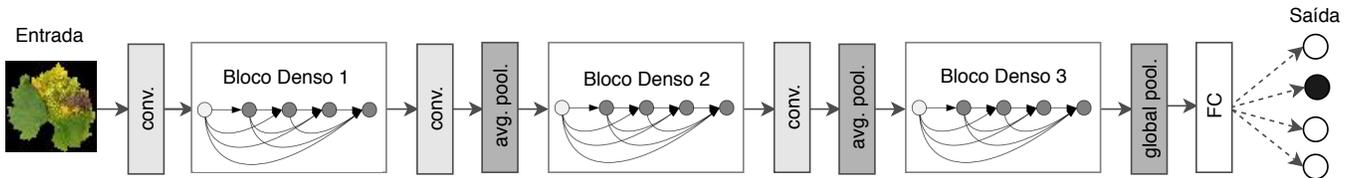


Fig. 2. Arquitetura DenseNet com três blocos densos aplicada ao problema de classificação de doenças em plantas.

TABELA I  
CONFIGURAÇÕES DAS BASES DE DADOS.

Denominação	Dim. alvo	Qtd. de Classes	Qtd. de Imagens
bd32final	32×32	50	35.430
bd32hard	32×32	66	36.558
bd64final	64×64	50	25.211
bd64hard	64×64	60	25.778
bd100final	100×100	50	14.882

### B. RNCs aplicadas e configurações

Neste estudo, as arquiteturas de RNCs VGG e DenseNet foram utilizadas para classificar variadas doenças em plantas. Especificamente, as redes VGG16, VGG19, DenseNet121, e DenseNet169 foram adaptadas e então aplicadas nas cinco bases geradas a partir da XDB. O modelo VGG16 foi selecionado devido ser uma RNC que é referência na área de visão computacional, sendo uma das primeiras a obter resultados satisfatórios em um grande volume de dados. Já a VGG19 foi escolhida a fim de avaliar o efeito da adição de profundidade no modelo VGG16. Já os modelos DenseNet foram selecionados por possuírem resultados superiores na literatura atual.

Em termos de adaptações nas arquiteturas originais, as VGGs foram configuradas com 200 neurônios nas duas primeiras camadas totalmente conectadas e as DenseNets tiveram a adição de uma camada totalmente conectada composta por 200 neurônios, seguida por uma regularização *dropout* de 0.5, antes da camada de saída. Todas as RNCs foram pré-treinadas na base de dados ImageNet [19] e suas implementações foram baseadas no framework Keras [22].

O treinamento das RNCs foi realizado usando o algoritmo *stochastic gradient descent* (SGD) com momento igual a 0,9 e taxa de aprendizado variável, iniciando em 1e-3 e decaindo ao

longo das épocas até no máximo 0,5e-6. Foram consideradas 200 épocas e *batch* de tamanho 64 para todas as RNCs, com redução na taxa de aprendizado pelos fatores 1e-1, 1e-2, 1e-3, e 0,5e-1 quando o número de épocas era maior que 80, 120, 160, e 180, respectivamente, considerando que o resultado nos dados de teste não melhorou após 5 épocas consecutivas. Adicionalmente, no treinamento foram geradas novas imagens a partir da aplicação de efeitos, como giro horizontal, deslocamento horizontal, deslocamento vertical, e ZCA *whitening*.

### C. Avaliação de desempenho dos resultados

Para avaliar de forma concisa os resultados, todas as RNCs foram executadas 10 vezes e então médias foram calculadas para obter valores das seguintes métricas: acurácia na etapa de teste, *f1-score*, e AUC. Tais métricas foram escolhidas por serem amplamente utilizadas na literatura de AP [23], [24].

Ressalta-se que todos os experimentos foram executados em uma GPU NVIDIA GeForce GTX 1050 TI com 4 GB de memória dedicada, juntamente com um processador Intel Core i7-8750H de 2,2 GHz (4,1 GHz) e 16 GB de memória principal. A média no tempo computacional das execuções mais custosas considerando a base de dados 32hard, que possui a maior quantidade de imagens, envolveu a DenseNet169 com aproximadamente 9h e 30min e a DenseNet121 com 8h e 15min. A CNN com menor tempo médio de execução foi a VGG16 que obteve 3h e 30min para essa mesma base, enquanto a VGG19 obteve cerca de 3h e 45min. Note que apesar das CNNs terem sido executadas em GPU, realizar mais que dez execuções para cada modelo mostrou-se custoso em termos temporais, mas as médias obtidas considerando todas as execuções apresentaram desvios-padrões relativamente baixos.

Note que apesar das RNCs terem sido executadas em GPU, realizar mais que 10 execuções para cada RNC mostrou-se

custoso em termos temporais, mas as médias obtidas considerando todas as execuções apresentaram desvios-padrões relativamente baixos.

Com o objetivo de atestar a significância dos resultados obtidos, a avaliação de desempenho das RNCs também considerou a aplicação do procedimento estatístico não-paramétrico denominado teste de Friedman, considerando os procedimentos *post-hoc* associados Shaffer e Bergmann [25]. As médias da métrica de acurácia na etapa de teste serviram como entrada para esse estudo estatístico.

#### IV. RESULTADOS

Considerando as cinco bases de dados geradas a partir da base de dados XDB, em todas foram selecionados aleatoriamente 70% e 30% dos dados para as etapas de treinamento e teste (que neste trabalho também pode ser chamada de etapa de validação), respectivamente.

A estratégia utilizada na aplicação dos modelos de RNC citados no problema de classificação de doenças de plantas envolveu iniciar o treinamento desses algoritmos a partir de um ponto intermediário, ao invés de realizar o treinamento do princípio, isso porque a base de dados ImageNet possui imagens de plantas [19], mesmo que saudáveis, logo o pré-treinamento nesse conjunto de dados contribuiu para uma maior eficácia no treinamento refinado realizado após essa etapa. Esse fato também justifica a utilização do algoritmo de treinamento SGD, uma vez que o objetivo é de otimização lenta de modelos pré-treinados, um refinamento, sem abruptas explorações do espaço de soluções/características.

A partir de 10 execuções de cada RNC, as médias de desempenho para as métricas de acurácia no teste, *f1-score* e AUC nos dados de teste das bases  $32 \times 32$ ,  $64 \times 64$  e  $100 \times 100$  são apresentadas através das Tabelas II, III e IV, respectivamente. Numa análise global, é possível verificar que na maioria dos casos, a RNC DenseNet169 manteve melhor desempenho, entre todas as RNCs consideradas, enquanto a VGG19 obteve os piores resultados. As métricas relacionadas com acurácia no teste e AUC ressaltaram melhor os desempenhos das RNCs, apresentando diferenças entre os resultados.

Analisando de forma mais específica os resultados das execuções, nas bases de dados com dimensão  $32 \times 32$ , é possível observar que os desempenhos das RNCs DenseNet121 e DenseNet169 se mantiveram similares, com pouca vantagem para a segunda, enquanto as VGGs apresentaram desempenho inferior, sendo a VGG19 o destaque negativo. É importante ressaltar que as bases de dados  $32 \times 32$  são aquelas que possuem mais imagens, o que conseqüentemente exigiu mais das RNCs, isto é, onde elas apresentaram seus menores desempenhos.

Tendência semelhante de resultados foram alcançados nas bases de dados  $64 \times 64$ . Na base de dados *64final*, para as métricas consideradas, a DenseNet169 obteve um desempenho levemente melhor que a DenseNet121, enquanto as VGGs, ainda num segundo patamar, obtiveram resultados com certa diferença, sendo a VGG16 o destaque positivo. Por outro lado, na base de dados *64hard*, essas diferenças se tornaram quase

imperceptíveis, possibilitando apenas a hipótese de mencionar que as DenseNets tiveram desempenho superior as VGGs em todas as métricas do estudo.

Na base de dados  $100 \times 100$ , as métricas relacionadas com a acurácia no teste e AUC apresentam mais claramente as diferenças de desempenho entre as RNCs, novamente atribuindo o melhor resultado a DenseNet169, seguida pela DenseNet121. Nesse caso, a VGG16 mostrou menor discrepância em relação aos resultados alcançados pela DenseNet121, enquanto a VGG19 permaneceu como a RNC com os piores resultados em todas as bases de dados.

Através dos resultados apresentados pelas métricas de acurácia no teste e AUC é verificado que os modelos de DenseNet obtiveram um melhor desempenho no que tange todas as bases de dados utilizadas nesse estudo. É possível inferir que um dos motivos que fizeram com que os modelos de DenseNet obtivessem melhor desempenho se deve principalmente às características intrínsecas de sua arquitetura, que promove um fluxo avançado de informações, sendo que todas as camadas do modelo conhecem as informações inerentes as camadas anteriores. Outro ponto a ressaltar, é o fato de que a DenseNet possui mecanismos de regularização que reduzem significativamente problemas como o gradiente de desaparecimento, que ainda se configura na arquitetura VGG.

Com a finalidade de verificar significância em todas as possíveis comparações de performance par-a-par entre as RNCs, o teste estatístico não-paramétrico de Friedman foi realizado, sendo complementado pelos procedimentos *post-hoc* Shaffer e Bergmann de forma a ajustar o valor  $p$  computado inicialmente. Esses experimentos assumiram as seguintes hipóteses nula e alternativa, respectivamente:  $H_0$  que considera que os resultados dos algoritmos comparados não apresentam diferenças; e  $H_1$  que considera que os resultados dos algoritmos comparados apresentam diferenças.

O teste de Friedman é um teste de hipótese estatístico não-paramétrico usado para detectar diferenças nos valores médios pelo menos entre duas populações. No contexto do desempenho das RNCs, o teste de Friedman detecta se há diferenças de significância entre duas ou mais RNCs. Se a significância estatística for válida para pelo menos duas RNCs consideradas, um procedimento *post-hoc* precisa ser executado para encontrar as RNCs específicas onde a significância existe. Para aplicar o teste de Friedman, os resultados originais, em termos de acurácia no teste, das Tabelas II, III e IV devem ser convertidos por meio de uma transformação baseada em ranking [25]. A Tabela V sintetiza o ranking computado a partir do teste de Friedman, ressaltando a DenseNet169 como melhor RNC e a VGG19 como pior RNC considerando todas as bases de dados deste estudo. Logo, é possível inferir que existe significância nas múltiplas comparações possíveis entre as RNCs, e que em alguns casos a hipótese  $H_0$  pode ser refutada e a hipótese  $H_1$  pode ser aceita.

Na Tabela VI são apresentadas todas as hipóteses de comparações e os valores  $p$  não-ajustado e ajustado pelos procedimentos *post-hoc* de Shaffer e Bergmann. É importante notar que quanto menor o valor de  $p$ , maior a significância

TABELA II  
MÉDIA DE 10 EXECUÇÕES NAS BASES DE DADOS 32×32: ACURÁCIA NO TESTE, *f1-score* E AUC.

RNC	bd32final			bd32hard		
	<i>Acurácia teste</i>	<i>f1-score</i>	AUC	<i>Acurácia teste</i>	<i>f1-score</i>	AUC
VGG16	0,936	0,917	0,957	0,929	0,881	0,937
VGG19	0,927	0,903	0,951	0,920	0,867	0,930
DenseNet121	0,941	0,923	0,961	0,939	0,909	0,949
DenseNet169	<b>0,942</b>	<b>0,926</b>	<b>0,962</b>	<b>0,940</b>	<b>0,910</b>	<b>0,951</b>

TABELA III  
MÉDIA DE 10 EXECUÇÕES NAS BASES DE DADOS 64×64: ACURÁCIA NO TESTE, *f1-score* E AUC.

RNC	bd64final			bd64hard		
	<i>Acurácia teste</i>	<i>f1-score</i>	AUC	<i>Acurácia teste</i>	<i>f1-score</i>	AUC
VGG16	0,963	0,944	0,969	0,960	0,929	0,963
VGG19	0,959	0,939	0,966	0,959	0,922	0,958
DenseNet121	0,969	0,952	0,973	<b>0,967</b>	0,941	0,968
DenseNet169	<b>0,971</b>	<b>0,956</b>	<b>0,975</b>	<b>0,967</b>	<b>0,946</b>	<b>0,970</b>

TABELA IV  
MÉDIA DE 10 EXECUÇÕES NA BASE DE DADOS 100×100: ACURÁCIA NO TESTE, *f1-score* E AUC.

RNC	bd100final		
	<i>Acurácia teste</i>	<i>f1-score</i>	AUC
VGG16	0,968	0,949	0,972
VGG19	0,965	0,940	0,968
DenseNet121	0,971	0,951	0,972
DenseNet169	<b>0,972</b>	<b>0,952</b>	<b>0,973</b>

TABELA V  
RANKING MÉDIO COMPUTADO PELO TESTE DE FRIEDMAN.

RNC	Ranking
DenseNet169	1,0
DenseNet121	2,0
VGG16	3,0
VGG19	4,0

entre os resultados das RNCs comparadas, culminando na rejeição da hipótese nula  $H_0$  e, conseqüente, aceitação da hipótese alternativa  $H_1$  com um grau de confiança elevado. Dessa forma, a DenseNet169 possui uma performance superior quando comparada com as duas versões da VGG, enquanto a DenseNet121 mostrou-se melhor na tarefa de classificação deste trabalho que a VGG19. Por outro lado, os comparativos entre os mesmos tipos de arquiteturas de RNCs (mudando apenas a profundidade), assim como o comparativo entre DenseNet121 e VGG16, não demonstraram significâncias adequadas para garantir inferências de performances melhores ou piores com grau de confiança satisfatório.

Em síntese, o teste estatístico não-paramétrico corroborou e ratificou as análises obtidas com as métricas utilizadas no estudo comparativo das RNCs no problema de classificação de doenças em plantas. O teste demonstrou primordialmente que há discrepância entre os resultados obtidos pelas DenseNets e pela VGG19. É importante destacar que maior significância entre os resultados das RNCs poderia ser alcançada com a criação de novas bases de dados e novos experimentos.

## V. CONCLUSÃO

Neste trabalho, as arquiteturas de redes convolucionais VGG16, VGG19, DenseNet121 e DenseNet169 foram utilizadas para classificar doenças em plantas da base de dados XDB que foi previamente tratada antes de ser submetida aos modelos. Posteriormente, foram criadas bases de dados com dimensões de 32×32 com 50 e 66 classes de doenças relativas a 11 e 14 culturas diferentes, respectivamente. Além disso, bases de dados de 64×64 com 50 e 60 classes referentes a 11 e 12 plantas diferentes, respectivamente. Finalmente, foi criada uma base de dados de 100×100 com 50 classes de patologias de 11 espécies vegetais distintas.

A partir dos resultados obtidos, o modelo DenseNet169 alcançou resultados de classificação promissores quando comparado as arquiteturas de VGG, tanto em termos de acurácia no teste e outras métricas relacionadas, quanto através do teste estatístico não-paramétrico de Friedman, embora não tenha apresentado resultados mais expressivos que sua versão com 121 camadas. Em contrapartida, os modelos de RNC do tipo VGG obtiveram resultados menos satisfatórios, com destaque

TABELA VI  
VALORES  $p$  NÃO-AJUSTADO (FRIEDMAN) E AJUSTADO (SHAFFER E BERGMANN) PARA MÚLTIPLAS COMPARAÇÕES.

Hipótese	$p$ não-ajustado	$p$ Shaffer	$p$ Bergmann
DenseNet169 vs. VGG19	2,38563454028667E-4	0,001431380724172002	0,001431380724172002
DenseNet121 vs. VGG19	0,014305878435429607	0,04291763530628882	0,04291763530628882
DenseNet169 vs. VGG16	0,014305878435429607	0,04291763530628882	0,04291763530628882
DenseNet121 vs. DenseNet169	0,22067136191984688	0,6620140857595407	0,44134272383969375
DenseNet121 vs. VGG16	0,22067136191984688	0,6620140857595407	0,44134272383969375
VGG16 vs. VGG19	0,22067136191984688	0,6620140857595407	0,44134272383969375

negativo para a versão de 19 camadas.

Além disso, constatou-se que, apesar das DenseNets possuírem uma quantidade relativamente superior de camadas, suas execuções tiveram menor custo computacional em comparação com os modelos de VGG. Esse fato pode ser justificado devido a quantidade superior de parâmetros das VGGs, uma vez que a arquitetura das DenseNets oferecem um balanceamento entre camadas convolucionais e camadas densas.

Como trabalhos futuros, os autores considerarão a aplicação e comparação de outros modelos de RNCs em uma maior quantidade de bases de dados com dimensões distintas para aumentar a quantidade de problemas que servirão de insumo para novos testes estatísticos não-paramétrico, cuja eficiência é comprovadamente melhor a medida que se tem uma quantidade de problemas (bases de dados) pelo menos duas vezes maior ou igual ao número de RNCs comparadas.

#### REFERÊNCIAS

- [1] V. Thorstensen, V. D. R. Gabriel, and A. L. Mesquita, "A participação de brasil e estados unidos na formulação das regras multilaterais do comércio agrícola," *Revista de Direito Internacional*, vol. 15, no. 3, Mar. 2019.
- [2] A. Messa, "Produtividade no brasil: desempenho e determinantes." Ipea, 2015.
- [3] R. N. Strange and P. R. Scott, "Plant disease: A threat to global food security," *Annual Review of Phytopathology*, vol. 43, no. 1, pp. 83–116, Sep. 2005.
- [4] C. A. Lopes, "Doenças," [https://sistemasdeproducao.cnptia.embrapa.br/FontesHTML/Pimenta/Pimenta\\_capsicum\\_spp/doencas.html](https://sistemasdeproducao.cnptia.embrapa.br/FontesHTML/Pimenta/Pimenta_capsicum_spp/doencas.html), 2007.
- [5] Y. Toda and F. Okura, "How convolutional neural networks diagnose plant disease," *Plant Phenomics*, vol. 2019, pp. 1–14, Mar. 2019.
- [6] K. Golhani, S. K. Balasundram, G. Vadmalai, and B. Pradhan, "A review of neural networks in plant disease detection using hyperspectral data," *Information Processing in Agriculture*, vol. 5, no. 3, pp. 354–371, Sep. 2018.
- [7] A. C. R. Marques, M. M. Raimundo, E. M. B. Cavalheiro, L. F. P. Salles, C. Lyra, and F. J. V. Zuben, "Ant genera identification using an ensemble of convolutional neural networks," *PLOS ONE*, vol. 13, no. 1, p. e0192011, Jan. 2018.
- [8] R. Oliveira, E. Alves, and C. Malqui, "Redes neurais convolucionais aplicadas à prensão robótica," in *Proceeding XIII Brazilian Congress on Computational Intelligence*. ABRICOM, Jan. 2018.
- [9] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, Dec. 2017.
- [10] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of apple leaf diseases based on deep convolutional neural networks," *Symmetry*, vol. 10, no. 1, p. 11, Dec. 2017.
- [11] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–11, 2016.
- [12] J. G. A. Barbedo, L. V. Koenigkan, B. A. Halfeld-Vieira, R. V. Costa, K. L. Nechet, C. V. Godoy, M. L. Junior, F. R. A. Patricio, V. Talamini, L. G. Chitarra, S. A. S. Oliveira, A. K. N. Ishida, J. M. C. Fernandes, T. T. Santos, F. R. Cavalcanti, D. Terao, and F. Angelotti, "Annotated plant pathology databases for image-based detection and recognition of diseases," *IEEE Latin America Transactions*, vol. 16, no. 6, pp. 1749–1757, Jun. 2018.
- [13] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of Physiology*, vol. 195, no. 1, pp. 215–243, Mar. 1968.
- [14] R. Oliveira, E. Alves, and C. Malqui, "Redes neurais convolucionais aplicadas à prensão robótica," in *Anais do 13 Congresso Brasileiro de Inteligência Computacional*, L. Martí and N. Sánchez Pi, Eds. Curitiba, PR: ABRICOM, 2017, pp. 1–11.
- [15] A. C. Gomes, A. Paes, and C. N. Vasconcelos, "Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres," in *2015 29rd Conference on Graphics, Patterns and Images (SIBGRAPI)*. SIBGRAPI, 2016.
- [16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition." ICLR, 2015.
- [17] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, Dec. 2017.
- [18] F. H. D. Araújo, A. C. Carneiro, R. R. V. Silva, F. N. S. Medeiros, and D. M. Ushizima, "Redes neurais convolucionais com tensorflow: Teoria e prática," in *2017 SBC III Escola Regional de Informática do Piauí*, vol. 1, no. 1. SBC, 2017, pp. 382–406.
- [19] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 248–255.
- [20] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jul. 2017.
- [21] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," *AISTATS*, vol. 5, no. 3, pp. 1–10, 2015.
- [22] "Keras: The python deep learning library," <https://keras.io/>, 2019.
- [23] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation," in *2006 AI Australasian Joint Conference on Artificial Intelligence (AI)*. Springer, Jul. 2006.
- [24] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, Jul. 1997.
- [25] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, Mar. 2011.