UNIVERSIDADE FEDERAL DO MARANHÃO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

RAPHAEL CAMARA SÁ

Classificação de fitopatologia fúngica baseada em *Deep Learning*

SÃO LUÍS – MA

2024

RAPHAEL CAMARA SÁ

Classificação de fitopatologia fúngica baseada em *Deep Learning*

Trabalho de Contextualização e Integração curricular, apresentado como requisito para Graduação em Bacharelado em Engenharia da
Computação pela Universidade Federal do Maranhão (UFMA).

Orientador: Prof. Dr. Haroldo Gomes Barroso Filho

SÃO LUÍS – MA

2024

RESUMO

A classificação de fitopatologia fúngica baseada em *Deep Learning* envolve a aplicação de técnicas avançadas de inteligência artificial para identificar e categorizar doenças causadas por fungos em plantas, utilizando imagens de tais plantas como dados de entrada. Esse processo é crucial na agricultura, pois doenças fúngicas podem devastar culturas e causar significativas perdas econômicas. O *Deep Learning*, especialmente por meio de redes neurais convolucionais (CNNs), permite a análise automática de imagens de plantas, detectando padrões específicos que indicam a presença de doenças. Essas redes aprendem a reconhecer diferentes doenças fúngicas através de um processo de treinamento que envolve grandes conjuntos de dados de imagens rotuladas. Além disso, técnicas como *Data Augmentation* são frequentemente utilizadas para aumentar a diversidade dos dados de treinamento, melhorando a robustez e a precisão dos modelos desenvolvidos. A aplicação desse método na agricultura, especialmente em culturas como o plantio de batata, pode transformar o manejo de doenças, permitindo intervenções mais rápidas e precisas, reduzindo a necessidade de insumos químicos e aumentando a produtividade agrícola. O desenvolvimento de sistemas automáticos de classificação fúngica baseados caracteriza-se como um processo de inovação promissor, que fomenta um potencial contribuição para uma agricultura mais sustentável e eficiente.

Palavras – chave: *Deep Learning*, doenças fúngicas, Redes neurais convolucionais (CNNs)

ABSTRACT

Fungal phytopathology classification based on Deep Learning involves the application of advanced artificial intelligence techniques to identify and categorize fungal diseases in plants using images as input data. This process is crucial in agriculture, as fungal diseases can devastate crops and cause significant economic losses. Deep Learning, particularly through Convolutional Neural Networks (CNNs), enables the automatic analysis of plant images by detecting specific patterns that indicate the presence of diseases. These networks learn to recognize different fungal diseases through a training process involving large datasets of labeled images. Additionally, techniques such as Data Augmentation are commonly used to enhance data diversity, improving the robustness and accuracy of the developed models. The application of this method in agriculture, especially in crops like potatoes, can revolutionize disease management by enabling faster and more precise interventions, reducing the need for chemical inputs, and increasing agricultural productivity. The development of automatic fungal classification systems represents a promising innovation, contributing to a more sustainable and efficient agricultural practice.

Keywords: *Deep Learning*, Fungal diseases, Convolutional Neural Networks (CNNs)

AGRADECIMENTOS

A conclusão deste Trabalho de Conclusão de Curso representa não apenas o encerramento de uma importante etapa acadêmica, mas também a realização de um grande sonho. Para chegar até aqui, contei com o apoio, incentivo e dedicação de muitas pessoas, às quais expresso minha mais profunda gratidão.

Primeiramente, agradeço a Deus, que me deu forças e sabedoria para enfrentar os desafios dessa jornada.

Ao meu orientador, Prof. Dr. Haroldo Gomes Barroso Filho, minha sincera gratidão por sua paciência, orientação e conhecimento compartilhado ao longo deste processo. Seu apoio e dedicação foram fundamentais para a construção deste trabalho e para o meu crescimento acadêmico e profissional.

Aos meus queridos pais, que sempre foram meu alicerce, oferecendo amor, apoio incondicional e incentivo em cada passo do caminho. Sem vocês, nada disso seria possível. Obrigado por acreditarem em mim e me motivarem a seguir em frente mesmo nos momentos mais difíceis.

Aos meus amigos, em especial Riordan dos Santos, por toda a parceria, apoio e palavras de incentivo ao longo dessa caminhada. Compartilhar essa trajetória com vocês tornou essa experiência mais leve e gratificante. Por fim, agradeço a todos que, direta ou indiretamente, contribuíram para a realização deste trabalho e para minha formação acadêmica. Cada gesto de apoio foi essencial para que este momento se tornasse realidade.

LISTA DE FIGURAS

SUMÁRIO

1. INTRODUÇÃO

A fitopatologia fúngica, ramo da ciência que estuda as doenças causadas por fungos em plantas, desempenha um papel fundamental na manutenção da produtividade agrícola. Essas fitopatologias podem comprometer significativamente a qualidade e o rendimento das colheitas, gerando prejuízos expressivos para os agricultores e impactando a economia em escala global. Entre as culturas mais vulneráveis a essas doenças, destaca-se a batata (*Solanum tuberosum*), um dos alimentos mais consumidos no mundo. Um exemplo grave é a requeima, causada pelo fungo *Phytophthora infestans*, capaz de dizimar plantações inteiras caso não seja identificada e controlada precocemente.

Antes do avanço do *Deep Learning*, a classificação de doenças em plantas dependia principalmente da análise visual feita por especialistas, exames laboratoriais e métodos tradicionais de aprendizado de máquina. A inspeção visual era amplamente utilizada, mas apresentava limitações, como subjetividade e dependência da experiência do avaliador, além de ser um processo demorado. Em casos mais complexos, eram realizados exames laboratoriais, que, embora mais precisos, demandam tempo, infraestrutura especializada e custos elevados. Nos últimos anos, as tecnologias baseadas em inteligência artificial, em especial o *Deep Learning* (DL), têm transformado diversas áreas, incluindo a agricultura. O uso dessas técnicas para a classificação de doenças fúngicas em plantas surge como uma alternativa inovadora e eficiente em relação aos métodos tradicionais de diagnóstico, que geralmente são mais lentos e exigem um alto nível de especialização. Segundo Camargo & Smith (2020), “o uso de inteligência artificial no manejo de doenças de plantas pode revolucionar a forma como monitoramos e controlamos enfermidades em culturas agrícolas, oferecendo um meio mais rápido e eficiente de detecção em comparação com os métodos convencionais.” A inteligência artificial possibilita a análise de grandes volumes de dados e a identificação de padrões complexos em imagens de plantas, viabilizando a detecção precoce de doenças e permitindo a adoção de medidas preventivas mais eficazes.

A pesquisa utilizará um conjunto de dados de imagens de plantas infectadas e realizará experimentos para comparar diferentes modelos de classificação, analisando sua precisão e eficiência. Além disso, serão aplicadas técnicas de *Data Augmentation* para aumentar a diversidade dos dados de treinamento e melhorar o desempenho dos modelos.

O estudo também abordará os conceitos fundamentais da fitopatologia fúngica e do DL, com ênfase nas doenças que impactam a cultura da batata e na eficácia das *CNN´s* na classificação de imagens. A metodologia inclui todas as etapas do processo, desde o pré-processamento dos dados até o treinamento e a avaliação dos modelos. Os resultados obtidos serão analisados com o objetivo de identificar as abordagens mais eficientes, sugerindo direções para pesquisas futuras e possíveis melhorias na aplicação da inteligência artificial no monitoramento e controle de doenças agrícolas.

1.1 Contextualização

A identificação e classificação de doenças fúngicas em plantas são fundamentais para a manutenção da saúde das culturas e para garantir a estabilidade da produção agrícola. Doenças causadas por fungos, como a requeima em batatas, podem devastar plantações, resultando em perdas significativas na colheita e impactando diretamente a economia agrícola. Conforme Almeida et al. (2019) destaca, “esse impacto agrava as questões humanitárias e econômicas relacionadas à produção agrícola e à segurança alimentar, sublinhando a importância da identificação precoce e das medidas de controle eficazes”.

A falta de detecção precoce dessas doenças acarreta no aumento do uso de pesticidas, elevando os custos de produção, trazendo impactos ambientais negativos e comprometendo a segurança alimentar. Nesse contexto, a demanda global por alimentos continua a crescer, tornando o manejo eficiente de doenças uma prioridade.

No entanto, os métodos tradicionais de diagnóstico, muitas vezes, são lentos e dependem de um nível de especialização que nem sempre está disponível, especialmente em regiões rurais ou em países em desenvolvimento. Segundo Almeida et al. (2021), “a aplicação de técnicas de DL na agricultura tem

o potencial de transformar o diagnóstico de doenças de plantas, oferecendo um método automatizado, rápido e preciso que pode mitigar perdas significativas de safra.”

A utilização de tecnologias de DL apresenta-se como uma alternativa promissora para aumentar a precisão do diagnóstico, permitindo respostas rápidas que podem salvar safras inteiras, reduzir perdas econômicas e promover uma produção agrícola mais sustentável. Com essas inovações, não apenas se melhora a eficiência do manejo de doenças, mas também se contribui para a segurança alimentar global e para a sustentabilidade do setor agrícola.

1.2 Justificativa

A escolha deste tema de pesquisa foi motivada por razões tanto pessoais quanto científicas. No âmbito pessoal, o interesse em integrar tecnologia e agricultura para enfrentar desafios práticos constitui uma das principais justificativas para a seleção deste estudo. A agricultura desempenha um papel essencial na sustentabilidade global, e a aplicação de inteligência artificial (IA), especialmente técnicas de DL, tem se destacado como uma área inovadora com grande potencial de impacto social e econômico.

Do ponto de vista científico, o estudo de técnicas de DL aplicadas à detecção de doenças fúngicas em plantas representa uma oportunidade para explorar e contribuir com um campo de pesquisa em expansão. Conforme destacado por Kamilaris e Prenafeta-Boldú (2018), Deep Learning é uma nova e avançada ferramenta para análise de dados e processamento de imagens. Ela tem imenso potencial, produz resultados promissores e tem sido usada com sucesso em diversas indústrias, incluindo a agricultura. Esse exponencial é evidenciado pelo uso crescente de modelos avançados de aprendizado de máquina para resolver problemas críticos no setor agrícola, especialmente no desenvolvimento de soluções aplicáveis em campo por meio da análise de dados reais.

Além disso, o impacto prático desta pesquisa reforça o compromisso com a inovação tecnológica voltada para o bem-estar global. A possibilidade de desenvolver ferramentas automatizadas e eficientes para o diagnóstico precoce

de doenças em culturas agrícolas contribui diretamente para o aumento da produtividade, a redução do uso de agroquímicos e a promoção de práticas mais sustentáveis. Dessa forma, este estudo não apenas avança no campo da inteligência artificial, mas também fortalece a relação entre tecnologia e agricultura, impulsionando soluções que beneficiam tanto produtores rurais quanto o meio ambiente.

1.3 Objetivos

Objetivo Geral:

Este trabalho tem como objetivo comparar o desempenho de cinco modelos de *Deep Learning* na classificação de doenças fúngicas em plantas de batata. Os modelos serão avaliados a partir de um conjunto de dados específico, visando identificar e categorizar com precisão diferentes tipos de doenças que afetam a cultura da batata. A comparação será baseada em métricas de avaliação como acurácia, precisão, *recall e F1-score*, permitindo verificar a eficácia de cada abordagem. O estudo busca contribuir para o aprimoramento da detecção de doenças agrícolas, auxiliando produtores no monitoramento e no manejo mais eficiente das plantações.

Específicos:

Implementar diferentes modelos de *Deep Learning*: Treinar diversos modelos de classificação de imagens, com o uso de Redes Neurais Convolucionais (CNNs), para identificar e diferenciar doenças fúngicas que afetam as plantas de batata.

Comparar o desempenho dos modelos: Avaliar e comparar os modelos quanto a métricas como precisão, recall, F1-score, entre outras, com o objetivo de identificar qual modelo apresenta o melhor desempenho na classificação das doenças.

Aplicar técnicas de *Data Augmentation*: Implementar técnicas para aumentar a variabilidade do conjunto de dados de treinamento, visando melhorar a robustez e a capacidade de generalização dos modelos.

Analisar a eficácia das técnicas utilizadas: Examinar como as diferentes abordagens de *Data Augmentation* impactam o desempenho dos modelos e identificar as melhores práticas para otimizar a classificação das doenças.

Utilizar um conjunto de dados específico: Trabalhar com um conjunto de dados que abranja diversas condições e variações de doenças fúngicas que afetam a batata, assegurando a relevância dos resultados obtidos.

Validar o sistema em dados não vistos: Testar os modelos em um conjunto de dados separado para garantir que o sistema seja eficaz na identificação de doenças fúngicas em situações reais e novos cenários, não incluídos no treinamento.

2. FUNDAMENTAÇÃO TEÓRICA

O avanço do *Deep Learning* tem transformado a forma como identificamos doenças fúngicas em plantas, especialmente na cultura da batata, que é altamente vulnerável a infecções como *pinta-preta* e *requeima*. Redes neurais convolucionais (*CNNs*) surgem como uma solução eficiente para essa tarefa, permitindo a análise automática de imagens e a classificação precisa das doenças.

Neste estudo, vamos explorar algumas das principais arquiteturas de *CNNs*, como *VGG*, *InceptionV3* e *ResNet-34*, destacando como cada uma contribui para a identificação fitopatológica. Além disso, discutiremos os desafios no treinamento desses modelos e as estratégias usadas para torná-los mais eficientes.

2.1 Fitopatologia fúngica

A fitopatologia fúngica é uma área essencial no estudo das doenças causadas por fungos em plantas, sendo fundamental para a manutenção da produtividade agrícola e da qualidade dos alimentos. Os fungos fitopatogênicos, organismos eucarióticos que desempenham funções importantes nos ecossistemas como decompositores, também atuam como agentes patogênicos, comprometendo a saúde e o rendimento de diversas culturas. A propagação desses fungos ocorre por meio de esporos transportados pelo vento, pela água ou por insetos, encontrando condições ideais de umidade e temperatura para a infecção das plantas, geralmente por meio de ferimentos ou aberturas naturais, como os estômatos.

Na cultura da cana-de-açúcar (*Saccharum spp.*), diversas doenças fúngicas se destacam pelo impacto negativo na produtividade e na qualidade da produção. Entre as principais enfermidades estão a ferrugem (*Puccinia melanocephala*), o carvão (*Ustilago scitaminea*), a mancha parda (*Cercospora longipes*), a podridão-abacaxi (*Ceratocystis paradoxa*), a podridão vermelha (*Colletotrichum falcatum*) e a fusariose (*Fusarium moniliforme*).

A ferrugem, uma das doenças mais disseminadas, pode causar perdas de até 50% em variedades suscetíveis. Seus sintomas iniciais manifestam-se como

pequenas pontuações cloróticas que evoluem para manchas amareladas e avermelhadas, com o desenvolvimento de pústulas que comprometem a área fotossintética da planta. Já o carvão, identificado no Brasil em 1946, caracteriza-se pela formação de estruturas alongadas e escuras conhecidas como “chicotes”, que emergem nos colmos da cana, resultando na redução drástica da produção e da qualidade do caldo.

Outra doença relevante é a podridão-abacaxi, causada por *Ceratocystis paradoxa*, que afeta principalmente o desenvolvimento inicial da planta. Os sintomas incluem baixa taxa de germinação, morte de brotos novos e uma coloração vermelha interna nos tecidos infectados, além de um odor característico semelhante ao de abacaxi.

O manejo dessas enfermidades envolve estratégias integradas, como a utilização de variedades geneticamente resistentes, a adoção de práticas culturais adequadas e, em alguns casos, a aplicação de fungicidas. O estudo aprofundado da fitopatologia fúngica é essencial para o desenvolvimento de soluções eficazes no controle dessas doenças, contribuindo para a sustentabilidade da produção e a segurança alimentar.

Quando não controladas de forma adequada, as doenças fúngicas podem ser devastadoras, resultando em perdas expressivas na produção agrícola. Essas enfermidades podem afetar diferentes partes das plantas, incluindo folhas, pecíolos, botões florais, frutos, caules e raízes, levando a sintomas como desfolha, redução do vigor, murchas, podridões e, em casos mais severos, a morte das plantas. Conforme destacado por Töfoli e Domingues (2018), “essas podem afetar folhas, pecíolos, botões florais, frutos, caules e sistema radicular, causando desfolha, queda de vigor, murchas, podridões e, em alguns casos, a morte de plantas.”

Portanto, a compreensão dos processos de infecção, disseminação e controle das doenças fúngicas na cana-de-açúcar é essencial para garantir a estabilidade da produção e minimizar os impactos econômicos e ambientais dessas enfermidades.

2.2 Deep Learning

O DL, uma subárea da Inteligência Artificial, tem se destacado significativamente nos últimos anos devido à sua capacidade de aprender padrões complexos a partir de grandes volumes de dados (LeCun, Bengio & Hinton, 2015). Diferentemente das técnicas tradicionais de aprendizado de máquina, o *Deep Learning* utiliza redes neurais artificiais compostas por múltiplas camadas, conhecidas como redes neurais profundas. Essas redes têm sido amplamente aplicadas em áreas como visão computacional, processamento de linguagem natural e reconhecimento de fala, transformando a forma como problemas complexos são abordados (Goodfellow, Bengio & Courville, 2016).

De acordo com LeCun et al. (2015), às Redes Neurais Profundas são compostas por múltiplas camadas de neurônios artificiais, permitindo a extração de características de alto nível a partir dos dados de entrada. Estas redes utilizam algoritmos como a retropropagação para o ajuste dos pesos sinápticos, aprimorando seu desempenho à medida que recebem mais dados.

O grande diferencial do DL reside na sua habilidade de extrair automaticamente características dos dados. Em abordagens tradicionais de aprendizado de máquina, é necessário que um especialista defina manualmente quais atributos são relevantes para a resolução de determinado problema. Entretanto, essa identificação é realizada pela própria rede neural, que aprende autonomamente quais padrões são cruciais para a tarefa em questão.

As redes neurais profundas são compostas por múltiplas camadas de neurônios, que simulam, de forma simplificada, o funcionamento do cérebro humano. Cada neurônio recebe um conjunto de entradas, processa essas informações e, por meio de uma função de ativação, decide se deve transmitir o sinal para a próxima camada. Durante o treinamento da rede, os pesos das conexões entre os neurônios são ajustados de maneira a minimizar os erros das previsões feitas pelo modelo. Esse processo de ajuste, denominado *backpropagation*, é um dos pilares do sucesso do *Deep Learning* (Werbos, 1974).

O crescimento exponencial do DL foi impulsionado, em grande parte, pelo aumento da capacidade computacional e pela disponibilidade de grandes

volumes de dados. Tecnologias como *GPUs* (*Graphics Processing Units*) e plataformas de computação em nuvem tornaram o treinamento dessas redes mais rápido e acessível. Além disso, *frameworks* como *TensorFlow* e *PyTorch* facilitaram o desenvolvimento de modelos complexos, permitindo a criação de soluções de alto desempenho em diversas áreas, como diagnósticos médicos, veículos autônomos e sistemas de recomendação.

Apesar do seu enorme sucesso, o DL ainda enfrenta desafios significativos. Um dos principais obstáculos é a necessidade de grandes volumes de dados rotulados para o treinamento dos modelos. Em áreas onde os dados são escassos ou difíceis de rotular, essa dependência pode limitar a aplicação da tecnologia. Outro desafio importante é a natureza de "caixa-preta" dos modelos, o que dificulta a interpretação e a explicação de como as decisões são tomadas. Além disso, o treinamento de redes neurais profundas pode ser extremamente dispendioso em termos de poder computacional e tempo de processamento.

Mesmo diante desses desafios, os avanços contínuos no DL têm expandido suas fronteiras, consolidando essa tecnologia como uma ferramenta essencial para a resolução de problemas complexos. Enquanto modelos tradicionais utilizam redes neurais com uma ou duas camadas computacionais, as abordagens baseadas em DL frequentemente possuem três ou mais camadas, podendo chegar a centenas ou milhares de camadas para alcançar níveis de desempenho extremamente altos (LeCun et al., 2015).

Dessa forma, o DL continua abrindo novas possibilidades e impactando diversas áreas da sociedade. Trata-se de um campo de estudo em rápida evolução, com promessas de inovações ainda mais significativas nos próximos anos, impulsionando avanços tecnológicos e científicos em múltiplos domínios.

2.3 CNN'S – Modelos 3 e 4

As redes neurais convolucionais (CNNs) revolucionaram o campo da visão computacional e do aprendizado de máquina, proporcionando avanços significativos na classificação de imagens e no reconhecimento de objetos. Antes de sua popularização, a identificação de padrões em imagens dependia de métodos manuais e demorados de extração de características. Hoje, as CNNs oferecem uma abordagem mais eficiente e escalável, baseada em princípios matemáticos, como a multiplicação de matrizes, para detectar padrões e estruturas em imagens (IBM, 2024).

No contexto da classificação de doenças em plantas de batata, a CNN personalizada utilizada nos Modelos 3 e 4 foi desenvolvida para identificar diferentes categorias com base em características visuais. Sua arquitetura é composta por cinco camadas convolucionais (*Conv2D*), responsáveis pela extração de padrões relevantes, intercaladas com camadas de pooling (*MaxPooling2D*), que reduzem a dimensionalidade e preservam as informações mais importantes. Essa estrutura hierárquica permite que, conforme os dados percorrem a rede, elementos simples como bordas e texturas sejam reconhecidos primeiro, evoluindo gradualmente até a identificação completa do objeto ou da doença.

Para melhorar a capacidade de generalização do modelo e evitar o *overfitting*, foram aplicadas técnicas de *Data Augmentation*, como rotações e zoom aleatórios, além do reescalonamento das imagens, normalizando os valores de pixel para um intervalo entre 0 e 1. Essas estratégias aumentam a diversidade do conjunto de treinamento e garantem que o modelo seja capaz de lidar com variações nas imagens, tornando-o mais robusto.

A parte final da rede consiste em camadas densas totalmente conectadas (*Dense*), com ativação *ReLU* para aprendizado não linear, garantindo que a rede possa aprender relações complexas nos dados. A última camada utiliza a função de ativação *softmax*, permitindo a classificação das imagens nas classes previamente definidas.

Apesar de sua eficiência, redes neurais convolucionais podem ser computacionalmente exigentes, especialmente durante o treinamento. Para lidar com esse desafio, são utilizadas unidades de processamento gráfico (GPUs), que possibilitam o processamento paralelo de grandes quantidades de dados, acelerando a aprendizagem do modelo.

Além das *CNNs,* existem outros tipos de redes neurais projetadas para diferentes aplicações. Redes neurais recorrentes (RNNs), por exemplo, são amplamente utilizadas para lidar com sequências temporais, como no processamento de linguagem natural e no reconhecimento de fala. No entanto, quando se trata de visão computacional e análise de imagens, as CNNs se destacam por seu desempenho superior (IBM, 2024).

A arquitetura desenvolvida para os Modelos 3 e 4 foi otimizada para garantir um bom equilíbrio entre desempenho e eficiência, tornando os modelos robustos e eficazes na tarefa de reconhecimento e classificação de doenças em plantas de batata. Ao unir técnicas avançadas de aprendizado profundo com estratégias de pré-processamento e otimização, essa abordagem representa um avanço significativo na detecção precoce de doenças agrícolas, contribuindo para práticas agrícolas mais eficientes e sustentáveis.

2.3.1 VGG

A arquitetura do modelo 1 combina influências do *LeNet-5*, de Yann LeCun (1998), e da VGG-16, de Karen Simonyan e Andrew Zisserman (2014). Assim como o *LeNet-5*, utiliza camadas convolucionais seguidas de funções de ativação *(ReLU)* e operações de *MaxPooling* para reduzir a dimensionalidade preservar as características mais relevantes da imagem. Da VGG-16, herda o uso de múltiplas camadas convolucionais com filtros pequenos (3×3) e o aumento progressivo do número de canais (32 → 64 → 128), permitindo um aprendizado mais rico e profundo das características visuais (Sarmah et al., 2022).

O modelo processa imagens RGB (3 canais) de 128×128 pixels através de três blocos principais de convolução. O primeiro bloco aplica 32 filtros 3×3, seguidos por *ReLU* e *MaxPooling* (2×2). No segundo bloco, a convolução

expande para 64 filtros, repetindo a ativação e o pooling. No terceiro bloco, o número de filtros aumenta para 128, mantendo a mesma estrutura. Após essa extração de características, a saída da última convolução é achatada e processada em camadas totalmente conectadas, começando com 512 neurônios e ativação *ReLU*, e finalizando com uma camada ajustada ao número de classes. Como utiliza a função de perda *CrossEntropyLoss*, a saída final é interpretada como probabilidades para cada classe (Sarmah et al., 2022).

A *VGG-16*, notável por sua precisão de 92,7% no *ImageNet* na ILSVRC-2014, recebe imagens de 224×224 pixels e opera em três canais (RGB). Sua estrutura inclui 13 camadas convolucionais com kernels 3×3, ativação *ReLU* e cinco camadas de *pooling* 2×2, possibilitando a extração eficiente de padrões complexos. Essa organização hierárquica favorece o aprendizado de representações abstratas e tem aplicações em diversas áreas de visão computacional, como reconhecimento facial e monitoramento do engajamento de alunos em aulas online.

2.3.2 *InceptionV3*

O modelo utilizado é baseado na arquitetura *InceptionV3*, uma rede neural convolucional desenvolvida pelo Google, reconhecida por sua eficiência na extração de características em imagens. Parte da família de modelos Inception, essa arquitetura otimiza convoluções e reduz a complexidade computacional ao substituir filtros 5×5 por duas convoluções 3×3 consecutivas, sem comprometer a capacidade de extração de padrões (Mbunge & Metfula, 2021).

No modelo 2, a opção *include_top=False* foi utilizada, removendo a parte densa original para personalizar a etapa de classificação. As camadas convolucionais, compostas por Inception Modules, aplicam múltiplos tamanhos de filtros convolucionais em paralelo, permitindo a captura de características em diferentes escalas, o que melhora a identificação de padrões associados a doenças em plantas de batata. A saída da camada mixed 7 foi escolhida como base para a nova cabeça de classificação, composta por uma camada Flatten() e uma camada *Dense()* com ativação *softmax*, refinando as características extraídas e aprimorando a precisão do modelo.

O modelo segue o conceito de *Transfer Learning*, utilizando pesos pré-treinados para extrair características relevantes e ajustando apenas a etapa final para adaptação ao novo conjunto de dados. Estratégias como reescalonamento de imagens, callbacks personalizados e métricas de avaliação como matriz de confusão e curva ROC foram aplicadas para garantir um bom desempenho.

Para aprimorar ainda mais o modelo, poderiam ser exploradas técnicas como *fine-tuning* de camadas finais, adição de camadas densas intermediárias e regularização com *Dropout* e *Batch Normalization*. Assim, a arquitetura *InceptionV3* se mostra uma solução eficiente e flexível para a classificação de imagens, combinando robustez na extração de características e adaptação a diferentes domínios (Mbunge & Metfula, 2021).

2.3.3 *ResNet-34*

O modelo 5 foi desenvolvido para a classificação de imagens de plantas de batata baseado na arquitetura *ResNet-34*, uma rede neural profunda reconhecida por sua eficiência em visão computacional. Apresentada por He et al. (2015), a arquitetura utiliza conexões residuais (*skip connections*), permitindo que a saída de uma camada seja adicionada diretamente à saída de uma camada posterior, facilitando o treinamento de redes profundas e evitando problemas como a degradação do gradiente.

Para o desenvolvimento do modelo, foi utilizada uma versão pré-treinada da *ResNet-34* no conjunto de dados *ImageNet*, ajustada para a classificação das imagens das plantas de batata. O aprendizado por transferência permitiu o aproveitamento dos pesos previamente aprendidos, sendo substituída apenas a última camada totalmente conectada por uma nova camada densa, adaptada às categorias específicas do banco de imagens.

A fim de melhorar a capacidade de generalização e precisão do modelo, foram aplicadas técnicas como *data augmentation* e redimensionamento de imagens. O *data augmentation* gerou variações artificiais nas imagens de treinamento, como rotações, espelhamentos e ajustes de brilho, aumentando a diversidade dos exemplos apresentados à rede neural. O redimensionamento

garantiu um formato padronizado para todas as imagens, otimizando o desempenho do modelo.

A introdução das redes residuais revolucionou a visão computacional, tornando viável o treinamento eficiente de redes profundas. Modelos como *ResNet-34*, *ResNet-50* e *ResNet-101* são amplamente utilizados em reconhecimento de imagens, diagnóstico médico e classificação de objetos. No contexto deste trabalho, a ResNet-34 foi utilizada com sucesso, consolidando-se como uma solução robusta para a identificação de diferentes condições nas plantas de batata, baseando-se exclusivamente em características visuais extraídas automaticamente pela rede neural.

3. METODOLOGIA

A metodologia de modelos de classificação em visão computacional utiliza redes neurais para identificar padrões e categorizar imagens com base em características extraídas automaticamente. No caso da classificação de doenças em plantas de batata, o *dataset* utilizado contém três classes principais: *Potato__healthy* (plantas saudáveis), *Potato__Early_blight* (folhas afetadas pela *pinta-preta*) e *Potato__late_blight* (plantas com *requeima*).

No entanto, desafios como o overfitting e *underfitting* podem comprometer o desempenho do modelo, afetando sua capacidade de generalização. Para mitigar esses problemas, aplicam-se técnicas como *data augmentation*, que aumenta a diversidade dos exemplos, regularização para evitar *overfitting* e *transfer learning* para aprimorar a extração de características. Essas estratégias garantem que o modelo possa diferenciar com precisão as classes, contribuindo para diagnósticos mais eficazes e um manejo agrícola mais sustentável**.**

3.1 VGG - Modelo 1 (Classificação de fitopatologia fúngica em folhas de batata)

O modelo foi implementado utilizando a biblioteca *PyTorch*, escolhida por sua eficiência e flexibilidade no treinamento de redes neurais profundas. Além disso, outras bibliotecas como *Torchvision*, *Matplotlib*, *Scikit-learn* (*sklearn*) e *Pillow (PIL)* foram empregadas para manipulação de imagens e análise dos resultados, garantindo um fluxo de aprendizado estruturado e eficiente.

A metodologia adotada abrange todas as etapas essenciais do desenvolvimento do modelo, desde o pré-processamento das imagens até a avaliação do desempenho, assegurando que o modelo seja capaz de generalizar bem para novos dados. O processo inicia-se com a criação do conjunto de dados, onde as imagens são organizadas em classes e passam por transformações adequadas para treinamento e validação. Em seguida, a arquitetura da CNN é definida, estruturando o modelo com múltiplas camadas convolucionais e totalmente conectadas para extração e aprendizado dos padrões visuais.

Durante o treinamento, os pesos da rede são ajustados com base na função de perda (*loss*), enquanto a acurácia é monitorada ao longo das épocas para garantir um aprendizado eficiente. Na fase de avaliação do desempenho, são utilizadas métricas como matriz de confusão e curva ROC, proporcionando uma análise detalhada da eficácia do modelo. Por fim, o modelo é testado em imagens inéditas, verificando sua capacidade de realizar predições corretas e assegurando sua aplicabilidade em cenários reais. Essa abordagem completa garante um modelo robusto e bem ajustado para a tarefa de classificação de imagens.

a) Divisão do Conjunto de Dados

Para garantir um treinamento eficiente e uma avaliação confiável, o conjunto de dados foi dividido da seguinte forma:

80% para treinamento – Utilizado para o ajuste dos pesos da rede neural.

10% para validação – Empregado para aferir o desempenho do modelo e reduzir o risco de *overfitting*.

10% para teste – Conjunto de imagens que não foi utilizado durante o treinamento.

Os dados normalizados entre as classes, assegurando representatividade e evitando vieses. Além disso, foi aplicado um processo randômico para impedir padrões indesejados durante o treinamento.

b) Pré-processamento das Imagens

Antes de serem alimentadas na rede neural, as imagens passaram por um rigoroso pré-processamento para garantir uniformidade e melhorar a capacidade de generalização do modelo. Inicialmente, todas as imagens foram redimensionadas para 128x128 pixels, assegurando um tamanho padronizado das amostras. Para aumentar a diversidade do conjunto de treinamento e tornar o modelo mais robusto a variações nas imagens, foram aplicadas técnicas de *Data Augmentation*, incluindo espelhamento horizontal, rotações aleatórias e ajustes de brilho, contraste e saturação. Essas transformações permitem que o modelo se adapte melhor a diferentes condições de iluminação e variações nos

dados de entrada. Além disso, as imagens foram convertidas para tensores *PyTorch*, garantindo um processamento eficiente pela rede neural. Esse conjunto de etapas fortalece a capacidade do modelo de aprender padrões relevantes e generalizar melhor para novas imagens.

c) Arquitetura da Rede Neural

A arquitetura da CNN foi cuidadosamente projetada para extrair padrões visuais relevantes das imagens e classificá-las com precisão. O modelo é composto por três camadas convolucionais, responsáveis por identificar características como bordas, texturas e formas dentro das imagens. Para otimizar o aprendizado e reduzir a dimensionalidade, são utilizadas camadas de agrupamento máximo (*MaxPooling2D*), que ajudam a destacar os padrões mais importantes enquanto diminuem a complexidade computacional. A função de ativação *ReLU* (*Rectified Linear Unit*) é empregada para introduzir não-linearidade ao modelo, permitindo que ele capture padrões mais complexos e evite problemas como o desaparecimento do gradiente.

Na etapa de classificação, a CNN conta com duas camadas densas totalmente conectadas, sendo que a última camada utiliza a função *Softmax* para transformar as saídas em probabilidades, determinando a categoria final de cada imagem. O treinamento do modelo é otimizado pelo algoritmo Adam, que ajusta os pesos de maneira eficiente, acelerando a convergência sem comprometer a estabilidade do aprendizado. Além disso, a função de perda *CrossEntropyLoss* é utilizada para minimizar o erro entre as previsões e as classes reais, garantindo uma melhor adaptação da rede aos dados. O modelo foi treinado por 10 épocas, permitindo que os pesos fossem ajustados progressivamente, resultando em uma melhoria contínua da acurácia na classificação das imagens.

d) Avaliação do Modelo

Após o treinamento, a avaliação do modelo foi realizada por meio de diversas métricas para medir sua capacidade de generalização e desempenho na classificação de novas imagens. Primeiramente, foi conduzida uma validação no conjunto de testes, permitindo verificar como o modelo se comporta diante de dados não vistos anteriormente. Além disso, foram gerados gráficos de acurácia

e perda ao longo das épocas, possibilitando uma análise visual do comportamento do treinamento e ajudando a identificar eventuais problemas como o *overfitting* e *underfitting*.

Para uma análise mais detalhada dos erros de classificação, foi construída uma matriz de confusão, permitindo identificar quais classes apresentaram maior dificuldade de distinção e onde o modelo teve mais acertos e falhas. Por fim, a curva ROC (*Receiver Operating Characteristic*) foi utilizada para avaliar a capacidade do modelo de diferenciar corretamente entre as classes, fornecendo uma visão mais aprofundada sobre sua precisão e confiabilidade. Esses métodos de avaliação garantem um entendimento completo do desempenho do modelo, possibilitando ajustes e otimizações para melhorar ainda mais sua eficácia.

e) Predição em Novas Imagens

Por fim, o modelo foi testado em imagens individuais, simulando um cenário real de classificação. Para cada imagem analisada:

A classe real e a classe prevista foram comparadas, permitindo verificar a precisão do modelo.

Os resultados foram avaliados em relação à matriz de confusão e à curva ROC, validando a eficácia do modelo em detectar doenças nas plantas de batata.

O modelo utilizado mostrou um bom desempenho na classificação de doenças em plantas de batata. A aplicação de técnicas de *Data Augmentation* contribuiu significativamente para a generalização do modelo, permitindo que ele identificasse padrões mesmo em imagens inéditas.

A análise das métricas de acurácia, matriz de confusão e curva ROC confirmou a eficácia da abordagem adotada. Além disso, a utilização do *PyTorch* proporcionou flexibilidade e eficiência no treinamento da rede neural.

3.2 *InceptionV3* - Modelo 2 (*Potato Plant Diseases Classification* - Pourea Ziasistani)

O modelo desenvolvido utiliza uma Rede Neural Convolucional (CNN) baseada na arquitetura *InceptionV3* pré-treinada para a classificação de doenças em plantas, especificamente na cultura da batata. A implementação foi projetada para garantir um fluxo de aprendizado eficiente, otimizando a divisão dos dados, o treinamento e a avaliação da rede neural.

O processo inicia-se com a importação das bibliotecas essenciais para manipulação de dados e *Deep Learning*, seguida de uma análise exploratória dos dados, permitindo visualizar a distribuição das classes e examinar amostras das imagens. No pré-processamento, as imagens são redimensionadas e normalizadas para garantir padronização e facilitar o aprendizado da rede. Em seguida, o conjunto de dados é dividido em treino, validação e teste, assegurando uma avaliação justa do modelo. Para um carregamento eficiente das imagens, são utilizados *DataLoaders*, que otimizam o processamento e aumentam a performance do treinamento.

A arquitetura da CNN baseia-se na *InceptionV3*, uma rede já treinada em grandes volumes de dados, com ajustes na camada final para adaptar-se ao número de classes do conjunto de dados. Durante o treinamento, técnicas como *early stopping* são aplicadas para evitar o *overfitting* e garantir um aprendizado eficiente. A avaliação do modelo é realizada por meio de métricas como matriz de confusão, curva ROC e relatório de classificação, fornecendo uma análise detalhada do desempenho. Por fim, o modelo é testado com imagens individuais, permitindo a predição de novas imagens e validando sua aplicabilidade em situações reais. Esse fluxo estruturado garante um desempenho robusto na classificação das doenças, tornando o modelo uma ferramenta eficaz para o monitoramento da saúde das plantas de batata.

a) Divisão do Conjunto de Dados

Para garantir um treinamento eficiente e uma avaliação precisa do modelo, os dados foram organizados em três subconjuntos principais. 80% do conjunto foi destinado ao treinamento, sendo utilizado para ajustar os pesos da rede neural

e permitir que o modelo aprenda padrões relevantes nas imagens. 10% foi reservado para validação, auxiliando na otimização dos hiperparâmetros e na prevenção do *overfitting*, garantindo que a rede generalize bem para novos dados. Os 10% restantes foram destinados ao teste, servindo para a avaliação final do modelo com imagens que não foram utilizadas no treinamento, permitindo uma análise imparcial de seu desempenho.

A separação dos dados foi realizada de forma equilibrada, assegurando que todas as classes estivessem representadas adequadamente em cada subconjunto. Além disso, foi aplicado um processo de embaralhamento, evitando padrões indesejados que poderiam influenciar o aprendizado do modelo. Essa abordagem estruturada melhora a capacidade de generalização da rede, tornando-a mais robusta e confiável para a classificação das doenças na cultura da batata.

b) Funcionamento do Modelo

O modelo segue um fluxo estruturado, abrangendo o pré-processamento das imagens, a configuração da rede neural e a validação dos resultados. Os principais passos incluem:

c) Pré-processamento das Imagens

No pré-processamento, as imagens são redimensionadas para 150x150 pixels, garantindo uniformidade no tamanho das amostras e facilitando a entrada na rede neural. Em seguida, são convertidas para tensores e normalizadas para um intervalo entre 0 e 1, permitindo que os valores de pixel sejam tratados de maneira eficiente pelos neurônios artificiais, acelerando o treinamento e melhorando a estabilidade do modelo.

Para aumentar a capacidade de generalização e evitar o *overfitting*, são aplicadas técnicas de *Data Augmentation*, introduzindo variações nas imagens sem alterar seu significado. Essas transformações incluem rotações aleatórias, espelhamentos horizontais e verticais, além de ajustes no brilho, simulando diferentes condições de iluminação e perspectivas. Esse conjunto de estratégias aprimora a robustez do modelo, garantindo que ele seja capaz de reconhecer padrões mesmo diante de variações nas imagens de entrada.

d) Definição e Treinamento da Rede Neural

A arquitetura do modelo utiliza a rede *InceptionV3* pré-treinada como base, aproveitando seu conhecimento adquirido em grandes bases de dados para extrair características relevantes das imagens. As camadas superiores da InceptionV3 são removidas, permitindo a personalização da rede para a tarefa específica de classificação de doenças na cultura da batata. Para adaptar o modelo ao problema, é adicionada uma nova camada densa totalmente conectada, responsável por processar as características extraídas e realizar a classificação final.

Para preservar o aprendizado do modelo pré-treinado, as camadas convolucionais da *InceptionV3* são congeladas, garantindo que apenas as novas camadas sejam treinadas. Isso reduz o tempo de treinamento e melhora a estabilidade do modelo. A última camada utiliza a função de ativação *Softmax,* ideal para problemas de múltiplas classes, convertendo as saídas em probabilidades para cada categoria.

O treinamento é otimizado com o otimizador *RMSprop*, que ajusta os pesos da rede de forma eficiente, acelerando a convergência sem comprometer a estabilidade do aprendizado. Além disso, um *callback* personalizado é implementado para interromper o treinamento caso a acurácia de validação atinja 99%, prevenindo o *overfitting* e garantindo um modelo bem ajustado aos dados sem memorizar excessivamente as amostras de treinamento. Essa abordagem permite um aprendizado eficiente, garantindo um desempenho robusto na classificação das imagens.

e) Avaliação do Modelo

Após o treinamento, o modelo é testado no conjunto de teste, utilizando imagens que não foram vistas durante o aprendizado para avaliar sua capacidade de generalização. Para analisar o desempenho, são calculadas métricas de avaliação e geradas visualizações que ajudam a compreender o comportamento da rede.

Entre essas visualizações, destacam-se as curvas de acurácia e perda ao longo do treinamento, que permitem identificar se o modelo convergiu adequadamente ou se houve problemas como *overfitting* ou *underfitting*. Além disso, uma matriz de confusão é construída para visualizar os erros de classificação e verificar quais classes apresentam maior dificuldade para a rede.

A avaliação também inclui a curva ROC (*Receiver Operating Characteristic*), que mede a capacidade do modelo de diferenciar corretamente entre as classes, sendo especialmente útil para entender a sensibilidade e especificidade da rede. Por fim, um relatório de classificação exibe métricas detalhadas como *precision*, *recall* e *F1-score*, proporcionando uma análise quantitativa do desempenho do modelo. Essa abordagem completa garante uma avaliação precisa da rede, facilitando ajustes e otimizações para melhorar ainda mais sua eficácia na classificação das doenças da cultura da batata.

3.3 CNN - Modelo 3 (*Potato Plant Diseases finding* - Rasam Anil)

O modelo desenvolvido utiliza uma Rede Neural Convolucional (CNN) para a classificação de imagens de plantas de batata, permitindo a identificação de possíveis doenças. O código segue um fluxo estruturado, garantindo desde o carregamento e processamento dos dados até a avaliação do desempenho e a predição de novas imagens.

Para a implementação, foram utilizadas bibliotecas especializadas, como *TensorFlow/Keras*, responsável pela construção, treinamento e avaliação da CNN, *NumPy*, para operações matemáticas e manipulação de matrizes, e *pandas*, caso necessário para organização e estruturação dos dados. Além disso, *Matplotlib* foi empregada para visualização dos resultados, enquanto PIL (*Pillow*) auxiliou no processamento das imagens.

O fluxo do código segue uma sequência lógica e otimizada. Inicialmente, são importadas as bibliotecas essenciais e carregado o conjunto de dados, que é devidamente organizado e dividido em treino, validação e teste. No pré-processamento, as imagens passam por normalização e técnicas de *Data Augmentation* para melhorar a capacidade de generalização da rede. Em seguida, a arquitetura da CNN é definida, composta por múltiplas camadas

convolucionais e totalmente conectadas, permitindo a extração eficiente de padrões das imagens.

Após a compilação e treinamento do modelo, a acurácia e a função de perda são monitorados ao longo das épocas, garantindo um ajuste adequado dos pesos da rede. A avaliação do desempenho é realizada com métricas detalhadas e análises visuais, incluindo gráficos de acurácia e perda, matriz de confusão e curva ROC. Por fim, o modelo é testado em imagens individuais, permitindo a predição de novas amostras e validando sua eficácia em situações reais. Essa abordagem estruturada garante um modelo robusto, preciso e aplicável ao diagnóstico automatizado de doenças em plantas de batata.

a) Divisão do Conjunto de Dados

Para garantir um treinamento eficaz e uma validação confiável do modelo, o conjunto de dados foi dividido estrategicamente em três partes. 80% das imagens foram destinadas ao treinamento, sendo utilizadas para ajustar os pesos da rede neural e permitir que o modelo aprenda os padrões visuais necessários para a classificação. 10% foram reservados para a validação, permitindo o monitoramento do desempenho do modelo ao longo do treinamento e ajudando a evitar problemas de *overfitting*, garantindo que a rede generalize bem para novos dados. Os 10% restantes foram utilizados para teste, servindo como um conjunto independente para avaliar a performance final da rede em imagens que não foram vistas anteriormente.

Para evitar viés no treinamento e garantir uma distribuição equilibrada entre as classes, os dados foram embaralhados antes da separação, assegurando que cada subconjunto contenha uma amostra representativa de todas as categorias. Essa abordagem melhora a robustez do modelo, permitindo um aprendizado mais eficiente e um desempenho mais confiável na classificação das doenças em plantas de batata.

b) Pré-processamento das Imagens

No pré-processamento das imagens, diversas etapas são aplicadas para garantir a padronização e melhorar a capacidade de generalização do modelo. Primeiramente, todas as imagens são redimensionadas para 256x256 pixels, assegurando um formato uniforme e facilitando sua entrada na rede neural. Em seguida, são normalizadas para o intervalo de 0 a 1, reduzindo discrepâncias nos valores de pixel e garantindo que o modelo aprenda de maneira mais eficiente, sem ser influenciado por variações extremas na iluminação ou contraste.

Além disso, para aumentar a diversidade do conjunto de treinamento e tornar o modelo mais robusto, são aplicadas técnicas de *Data Augmentation*, incluindo espelhamento horizontal e vertical, rotações aleatórias e ajustes de brilho e contraste. Essas transformações permitem que o modelo aprenda a reconhecer padrões visuais mesmo em condições variadas, reduzindo a dependência de características específicas da imagem original e ajudando a evitar o *overfitting*.

c) Definição e Treinamento da Rede Neural

A arquitetura da CNN foi projetada para extrair padrões visuais relevantes das imagens e classificá-las com alta precisão. O modelo conta com cinco camadas convolucionais, responsáveis por identificar características como bordas, texturas e formas, tornando a rede capaz de reconhecer detalhes essenciais para a diferenciação entre as classes. Cada uma dessas camadas é seguida por uma camada de agrupamento (*MaxPooling2D*), que reduz a dimensionalidade dos dados, otimizando o processamento e evitando a sobrecarga computacional.

As camadas convolucionais utilizam a função de ativação *ReLU*, introduzindo não-linearidade no modelo, permitindo que a rede capture padrões complexos e melhore sua capacidade de generalização. Após a extração de características, o modelo finaliza com camadas totalmente conectadas (*Dense*), sendo a última uma camada de saída com ativação *Softmax*, que converte os resultados em probabilidades e realiza a classificação final em uma das três categorias possíveis.

Para otimizar o aprendizado, a compilação do modelo é feita com o otimizador Adam, que ajusta os pesos de maneira eficiente, acelerando a convergência e garantindo estabilidade no treinamento. A função de perda *Sparse Categorical Crossentropy* é empregada para minimizar o erro entre as previsões do modelo e as classes reais, assegurando um aprendizado mais preciso. O treinamento é realizado por 30 épocas, permitindo que a rede aprenda padrões robustos nos dados e refine suas previsões ao longo do tempo.

d) Avaliação do Modelo

Após o treinamento, o modelo passa por uma fase de testes utilizando o conjunto de testes, que contém imagens nunca vistas pela rede durante o aprendizado. Essa etapa é essencial para verificar a capacidade de generalização do modelo e garantir que ele não apenas memorize os dados de treinamento, mas também consiga realizar previsões precisas em novas imagens. Para acompanhar o desempenho do treinamento e da validação ao longo das épocas, são gerados gráficos de acurácia e perda, permitindo uma análise visual do comportamento do modelo.

Além da avaliação quantitativa, o modelo é testado em imagens individuais, permitindo verificar sua precisão na classificação em um cenário mais realista. Para cada imagem analisada, são exibidos a classe real e a classe prevista, além da confiança da predição em percentual, indicando o grau de certeza do modelo na classificação. Essa abordagem garante uma avaliação completa do desempenho da CNN, validando sua aplicabilidade prática na identificação de doenças em plantas de batata e permitindo ajustes, caso necessário, para otimização do modelo.

3.4 CNN - Modelo 4

O modelo desenvolvido utiliza uma Rede Neural Convolucional (CNN) para a classificação de imagens, empregando bibliotecas especializadas para processamento, treinamento e avaliação. A estrutura do código garante eficiência e precisão em todas as etapas do desenvolvimento.

As principais bibliotecas utilizadas incluem *TensorFlow/Keras*, responsável pela construção e treinamento da rede neural, Matplotlib, usada para visualização dos dados e análise dos resultados, e *OS/ZipFile*, que facilitam o gerenciamento de arquivos e o armazenamento do modelo treinado.

O processo segue uma sequência estruturada para garantir um fluxo eficiente e otimizado. Inicialmente, as bibliotecas essenciais são importadas, e o conjunto de dados passa por um pré-processamento rigoroso, incluindo redimensionamento, normalização e técnicas de *Data Augmentation* para melhorar a generalização do modelo. Em seguida, a arquitetura da CNN é definida, combinando camadas convolucionais, de *pooling* e totalmente conectadas para extrair e aprender padrões relevantes das imagens.

Durante o treinamento e validação, os parâmetros da rede são ajustados, e métricas como acurácia e função de perda são monitoradas para evitar sobreajuste (*overfitting*). Após o término do treinamento, a rede é avaliada utilizando o conjunto de testes, garantindo que seu desempenho seja validado com imagens não vistas anteriormente.

Por fim, o modelo treinado é salvo e compactado, permitindo sua reutilização em aplicações futuras sem a necessidade de novo treinamento. Esse fluxo bem definido assegura um modelo robusto, escalável e pronto para ser implementado em diversas tarefas de classificação de imagens.

a) Divisão do Conjunto de Dados

A separação do conjunto de dados foi realizada de forma estratégica para garantir um treinamento eficiente e uma avaliação precisa do modelo. A distribuição dos dados seguiu o seguinte critério:

80% para treinamento – fase em que a rede ajusta seus pesos e aprende os padrões das imagens.
10% para validação – utilizado para ajustar hiperparâmetros e evitar *overfitting*, garantindo que o modelo não memorize os dados de treinamento.
10% para teste – empregado na avaliação final do modelo, permitindo medir sua capacidade de generalização com dados inéditos.

Para assegurar uma distribuição equilibrada entre os diferentes conjuntos, os dados foram embaralhados antes da separação. Esse procedimento evita que imagens semelhantes fiquem agrupadas, garantindo que a rede neural seja treinada com um conjunto de dados mais diversificado. Dessa forma, a metodologia adotada contribui para a construção de um modelo robusto e eficiente na classificação das imagens, assegurando um desempenho satisfatório na etapa de avaliação.

b) Pré-processamento das Imagens

Para garantir um treinamento eficiente e um modelo mais robusto, as imagens passam por um pré-processamento rigoroso antes de serem inseridas na rede neural. Primeiramente, são redimensionadas para um tamanho padronizado, garantindo uniformidade no treinamento. Em seguida, são normalizadas, ajustando os valores de pixel para um intervalo entre 0 e 1, o que facilita o processamento pelos neurônios artificiais e acelera a convergência do modelo. Além disso, são aplicadas técnicas de Data Augmentation para aumentar a diversidade do conjunto de dados e melhorar a capacidade de generalização da rede. Entre essas técnicas, incluem-se rotações aleatórias, espelhamentos horizontais e verticais e ajustes de contraste, tornando o modelo mais resistente a variações naturais das imagens. Esse processo reduz significativamente o risco de *overfitting*, garantindo um aprendizado mais eficiente.

c) Estrutura da Rede Neural Convolucional (CNN)

A arquitetura da CNN é projetada para processar imagens de maneira eficiente, extraindo padrões relevantes e permitindo uma classificação precisa. O modelo é composto por diferentes camadas especializadas que desempenham papéis essenciais no aprendizado. As camadas convolucionais são responsáveis por

identificar características como bordas, texturas e padrões complexos, permitindo que a rede aprenda representações ricas dos dados. Em seguida, as camadas de pooling reduzem a dimensionalidade das informações, otimizando a eficiência computacional e tornando a rede mais robusta a pequenas variações nas imagens. Por fim, as camadas totalmente conectadas (Dense) utilizam os padrões extraídos para realizar a classificação final, determinando a categoria da imagem com base nas informações processadas ao longo das camadas anteriores. Essa estrutura modular garante um modelo eficiente, preciso e capaz de lidar com diferentes variações nos dados de entrada, tornando a CNN uma abordagem poderosa para a classificação de imagens.

d) Treinamento e Otimização

Para garantir um aprendizado eficiente e otimizado, a CNN utiliza a função de ativação *ReLU*, que introduz não-linearidade ao modelo, permitindo a detecção de padrões mais complexos e melhorando a capacidade de generalização da rede. O otimizador Adam é empregado para ajustar automaticamente a taxa de aprendizado, acelerando a convergência e tornando o treinamento mais estável. A função de perda *Sparse Categorical Crossentropy* é utilizada para minimizar o erro entre as previsões do modelo e as classes reais, garantindo uma melhor adaptação da rede aos dados de entrada. O treinamento é realizado por um número fixo de épocas, assegurando que a rede tenha tempo suficiente para aprender os padrões dos dados sem super ajustar-se, resultando em um modelo equilibrado e eficaz na classificação das imagens.

e) Avaliação do Modelo

Após o treinamento, o modelo passa por uma fase de avaliação para medir sua capacidade de generalização e verificar se aprendeu de forma eficiente. Nessa etapa, métricas como acurácia e função de perda são analisadas para determinar a qualidade do aprendizado e identificar possíveis ajustes necessários. Além disso, gráficos de acurácia e perda ao longo das épocas são gerados, permitindo uma análise visual do desempenho do modelo e facilitando a detecção de problemas como *overfitting* ou *underfitting*. Para garantir sua reutilização em aplicações futuras, o modelo treinado é salvo e compactado, possibilitando que seja facilmente carregado para novas tarefas de classificação

sem a necessidade de repetir todo o processo de treinamento. Esse procedimento otimiza tempo e recursos computacionais, tornando o modelo mais prático e acessível para diferentes aplicações.

3.5 *ResNet 34* - Modelo 5

a) Estrutura Geral do Código

O modelo desenvolvido utiliza uma Rede Neural Convolucional (CNN) baseada na arquitetura ResNet-34, amplamente reconhecida por sua eficiência em tarefas de visão computacional. A implementação foi projetada para otimizar o treinamento e a avaliação da rede, garantindo um fluxo estruturado e eficiente na aprendizagem dos padrões das imagens.

As principais bibliotecas utilizadas no código são:

PyTorch – Construção, treinamento e avaliação da rede neural.
Torchvision – Manipulação do conjunto de imagens e uso de modelos pré-treinados.
Matplotlib – Visualização e análise dos dados.
OS – Manipulação de diretórios e arquivos.
TQDM – Monitoramento do progresso do treinamento.

O código segue uma abordagem estruturada e eficiente para a classificação de imagens, garantindo um processamento otimizado e de alto desempenho. Inicialmente, são importadas as bibliotecas essenciais para manipulação dos dados e construção da rede neural. Em seguida, define-se a estrutura do conjunto de dados, organizando os diretórios de treino, validação e teste para facilitar o carregamento e a separação adequada das imagens.

No pré-processamento, as imagens são redimensionadas e convertidas em tensores, preparando-as para a entrada na rede neural. Para garantir um fluxo eficiente de dados durante o treinamento, utilizam-se *DataLoaders,* que otimizam a leitura e o processamento das amostras. Além disso, o código identifica automaticamente o dispositivo de processamento disponível, utilizando a GPU sempre que possível para acelerar os cálculos.

A arquitetura escolhida para a CNN é a *ResNet-34,* um modelo pré-treinado que oferece excelente capacidade de extração de padrões visuais. A última camada é ajustada para a classificação específica do conjunto de dados, tornando o modelo adequado ao problema em questão. Durante o treinamento, métricas de desempenho são monitoradas a cada época, permitindo ajustes para otimizar os resultados. Ao final, o modelo treinado é salvo, possibilitando sua reutilização sem a necessidade de um novo treinamento.

Essa abordagem modular e escalável permite não apenas uma classificação eficiente das imagens, mas também um aproveitamento inteligente dos recursos computacionais, resultando em um modelo robusto e de alto desempenho.

b) Divisão do Conjunto de Dados

Para garantir um treinamento eficiente e uma validação confiável, o conjunto de dados foi cuidadosamente dividido em três partes: 80% para treinamento, 10% para validação e 10% para teste. Essa distribuição permite um ajuste adequado dos pesos da rede neural durante o aprendizado, enquanto a validação monitora o desempenho do modelo ao longo do treinamento, ajudando a evitar o sobreajuste (*overfitting*). Já o conjunto de teste, composto por imagens nunca vistas pelo modelo, serve para uma avaliação final e imparcial da sua capacidade de generalização.

A separação dos dados foi feita de forma equilibrada, garantindo que todas as classes estivessem bem representadas em cada subconjunto, evitando possíveis desbalanceamentos que poderiam comprometer a precisão da rede. Além disso, antes da distribuição, as imagens foram embaralhadas para reduzir viés e assegurar que o modelo não aprendesse padrões indesejados baseados na ordem dos dados. Esse processo estruturado proporciona um treinamento mais robusto e um desempenho confiável na classificação das imagens.

c) Funcionamento do Modelo

O modelo adota um fluxo estruturado para garantir eficiência tanto no treinamento quanto na classificação das imagens. No pré-processamento, as imagens são redimensionadas para 128x128 pixels, assegurando uma padronização que facilita a entrada na rede neural. Além disso, são convertidas para tensores, permitindo um processamento mais eficiente e compatível com os cálculos necessários para o aprendizado profundo. Essas etapas garantem que os dados estejam formatados de maneira adequada, otimizando o desempenho do modelo e contribuindo para uma classificação mais precisa e confiável.

d) Definição e Treinamento da Rede Neural

O modelo adotado tem como base a arquitetura *ResNet-34*, uma rede neural profunda e pré-treinada, amplamente utilizada para tarefas de visão computacional devido à sua eficiência em capturar padrões complexos em imagens.

Para garantir que a rede possa aprender representações mais ricas e complexas, é utilizada a função de ativação *ReLU (Rectified Linear Unit)*, que introduz não-linearidade ao modelo, ajudando a capturar padrões mais detalhados nos dados de entrada. Além disso, o treinamento é otimizado com o algoritmo *Adam (Adaptive Moment Estimation*), um dos otimizadores mais eficientes, que ajusta os pesos da rede de maneira adaptativa, acelerando a convergência e melhorando a estabilidade do aprendizado.

Durante o processo de treinamento, cada época é monitorada de perto para avaliar o desempenho da rede e identificar eventuais necessidades de ajustes. Isso permite que mudanças estratégicas, como a modificação da taxa de aprendizado, a aplicação de técnicas de regularização ou ajustes na arquitetura, sejam implementadas caso necessário. Dessa forma, o modelo busca equilibrar aprendizado e generalização, garantindo um desempenho sólido na tarefa de classificação.

e) Avaliação do Modelo

Durante o treinamento, a cada época, o modelo é validado utilizando o conjunto de validação para garantir um ajuste adequado dos pesos e identificar possíveis problemas como o *overfitting*. Métricas essenciais, como acurácia e função de perda, são monitoradas constantemente para avaliar o desempenho da rede e orientar ajustes necessários. Esse acompanhamento permite identificar se o modelo está generalizando bem ou apenas memorizando os dados de treinamento, possibilitando intervenções como ajustes na taxa de aprendizado, uso de técnicas de regularização ou aumento dos dados (*data Augmentation*).

4. *Dataset*

A agricultura enfrenta desafios cada vez maiores devido ao impacto das doenças nas plantações, comprometendo a produtividade e a segurança alimentar. Entre as culturas mais afetadas, a batata (*Solanum tuberosum*) destaca-se por sua vulnerabilidade a infecções fúngicas, como a *pinta-preta* (*Alternaria solani*) e a *requeima* (*Phytophthora infestans*). Essas doenças podem reduzir drasticamente a produção, causando prejuízos econômicos significativos. Diante desse cenário, a aplicação de técnicas de *deep learning* na identificação precoce de enfermidades surge como uma solução inovadora e eficaz, promovendo um manejo agrícola mais eficiente e sustentável.

A utilização de modelos de redes neurais convolucionais (*CNNs*) para a classificação de imagens permite a automatização do diagnóstico fitopatológico, superando as limitações dos métodos tradicionais. Normalmente, a detecção de doenças nas plantações depende da experiência visual de especialistas, um processo demorado, subjetivo e suscetível a erros. Com o avanço do aprendizado profundo, tornou-se possível desenvolver sistemas automatizados capazes de analisar imagens de alta resolução e identificar padrões característicos das doenças, proporcionando uma avaliação rápida e precisa do estado da lavoura.

O *dataset* analisado contém três classes que representam diferentes condições das plantas de batata: saudáveis e afetadas por doenças fúngicas. A primeira classe, *Potato__Early_blight*, contém imagens de plantas infectadas pela *pinta-preta*, caracterizada por manchas escuras e concêntricas nas folhas. Já a classe *Potato__healthy* agrupa imagens de plantas sem sinais aparentes de doenças, sendo essencial para a diferenciação entre um estado saudável e um comprometido. Por fim, a classe *Potato__Late_blight* representa plantas afetadas pela *requeima*, uma das doenças mais destrutivas da cultura da batata, caracterizada por lesões encharcadas que podem se espalhar rapidamente.

Esse conjunto de dados possibilita o treinamento de modelos que aprendem a diferenciar sintomas visíveis de infecções fúngicas, permitindo a criação de sistemas inteligentes de monitoramento agrícola. A detecção precoce dessas doenças minimiza o uso excessivo de defensivos agrícolas, reduzindo os

impactos ambientais e promovendo práticas mais sustentáveis. Além disso, ao automatizar esse processo, os agricultores ganham uma ferramenta poderosa para a tomada de decisões estratégicas, garantindo um controle preventivo mais eficiente.

No entanto, apesar dos benefícios, a implementação de sistemas baseados em inteligência artificial na agricultura ainda enfrenta desafios. Um dos principais entraves é a necessidade de grandes volumes de dados anotados para o treinamento eficiente dos modelos. Além disso, fatores como variações climáticas, iluminação e diferentes estágios de crescimento das plantas podem influenciar a precisão do diagnóstico. Dessa forma, é fundamental que os *datasets* sejam robustos e diversificados, refletindo a realidade das lavouras para garantir a confiabilidade dos modelos preditivos.

Figura 1: *Potato Plant Diseases* - Dataset, **Fonte:** Autor

5. RESULTADOS E DISCUSSÕES

Nesta seção, são apresentados os resultados dos cinco modelos de DL na classificação de doenças fúngicas em plantas de batata. A avaliação considera métricas como acurácia, F1-score, curva ROC e matriz de confusão.

A acurácia mede a proporção de classificações corretas, representado pela fórmula (*Acurracy* = (TP+TN) / (TP+TN+FP+FN)), enquanto o F1-score equilibra acurácia e recall, sendo calculado como (F1 = 2×(*Acurracy*×*Recall*) / (*Acurracy* + *Recall*)). Além disso, a curva ROC analisa o equilíbrio entre verdadeiros e falsos positivos, e a matriz de confusão detalha os erros e acertos do modelo. Essas análises permitem identificar o modelo mais eficiente para a detecção de doenças fúngicas na batata, contribuindo para o monitoramento e manejo mais preciso das plantações.

5.1 *VGG*

O treinamento do modelo 1 demonstrou uma evolução consistente, com melhora progressiva dos principais indicadores de desempenho. Desde as primeiras épocas, a função de perda (*loss*) diminuiu significativamente, sugerindo que a rede neural estava aprendendo e ajustando seus pesos de forma eficiente. No início, a perda no conjunto de treino era relativamente alta (0.8115 na Época 1), o que era esperado, pois o modelo ainda estava nos estágios iniciais de aprendizado. Com o avanço do treinamento, essa métrica caiu para 0.2250 na última época, evidenciando um ajuste eficaz.

No conjunto de validação, o comportamento foi semelhante: a loss reduziu de 0.6449 na Época 1 para 0.2498 na Época 9, o que indicava uma boa generalização do modelo. No entanto, na Época 10, a perda aumentou para 0.3200, possivelmente sinalizando o início de overfitting, ou seja, um ajuste excessivo aos dados de treino, prejudicando a capacidade de generalização. Para mitigar esse problema, estratégias como *early stopping* ou dropout poderiam ser aplicadas para interromper o treinamento no momento ideal.

A acurácia, que mede a proporção de previsões corretas, seguiu uma trajetória ascendente ao longo do treinamento. No início, a acurácia da validação era de 74,01% (Época 1), mas com o aprendizado da rede, atingiu um pico de 90,72% na Época 9. No entanto, na última época, houve uma leve queda para 87,94%, acompanhando o aumento da perda na validação, o que reforça a suspeita de overfitting ou variações naturais nos dados de validação. Apesar dessa leve oscilação, atingir 90,72% de acurácia mostra que o modelo é altamente confiável para a tarefa proposta.

A precisão (*precision*), que avalia quantas previsões positivas estavam corretas, também apresentou um crescimento relevante. Inicialmente, o modelo tinha uma precisão de 68,74% (Época 1), ou seja, cerca de 31% das previsões positivas estavam erradas. Com o treinamento, a precisão melhorou gradativamente, atingindo 91,32% na Época 9. Na Época 10, ocorreu uma leve queda para 87,63%, acompanhando a diminuição da acurácia, mas ainda dentro de um patamar bastante aceitável para aplicação prática.

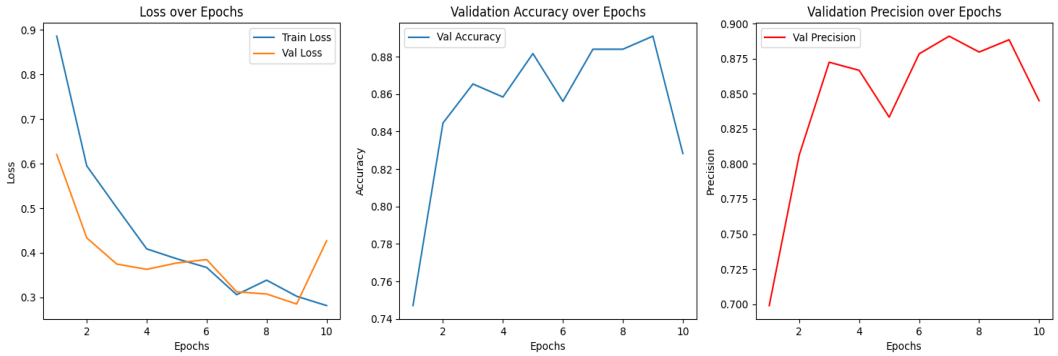

Figura 2: *Loss, accuracy and precision over epochs – Model 1*, **Fonte:** Autor

Apesar do desempenho positivo, um desafio identificado foi a dificuldade do modelo em classificar corretamente plantas saudáveis (*Healthy*). A matriz de confusão mostrou que apenas 4 amostras saudáveis foram corretamente identificadas, enquanto 31 foram erroneamente classificadas como *Late Blight*. Esse erro pode estar relacionado ou à similaridade visual entre folhas saudáveis e sintomas iniciais das doenças.

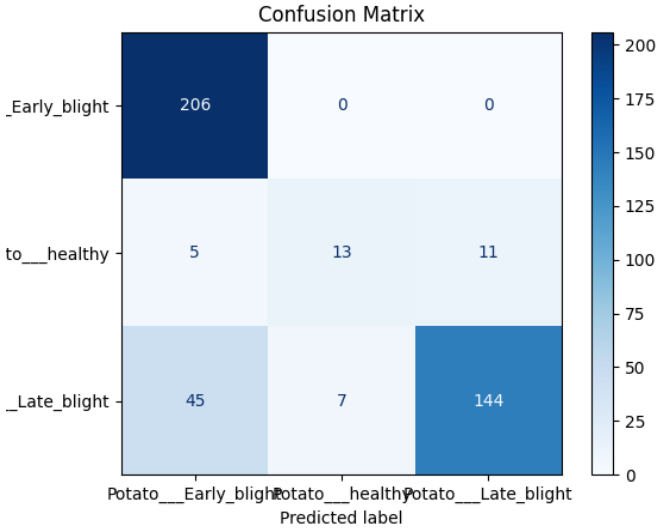

Figura 3: *Confusion Matrix Model - 1,* **Fonte:** Autor

A curva ROC confirmou a boa capacidade de discriminação do modelo. A Classe 0 apresentou um AUC de 0.99, indicando uma separação quase perfeita. A Classe 2 teve um AUC de 0.97, enquanto a Classe 1 ficou em 0.94, um pouco inferior, mas ainda satisfatório. Esse resultado sugere que, apesar da alta eficiência global, a Classe 1 poderia se beneficiar de ajustes, como balanceamento de dados ou otimização dos limites de decisão, para reduzir erros e melhorar sua separação em relação às outras categorias.

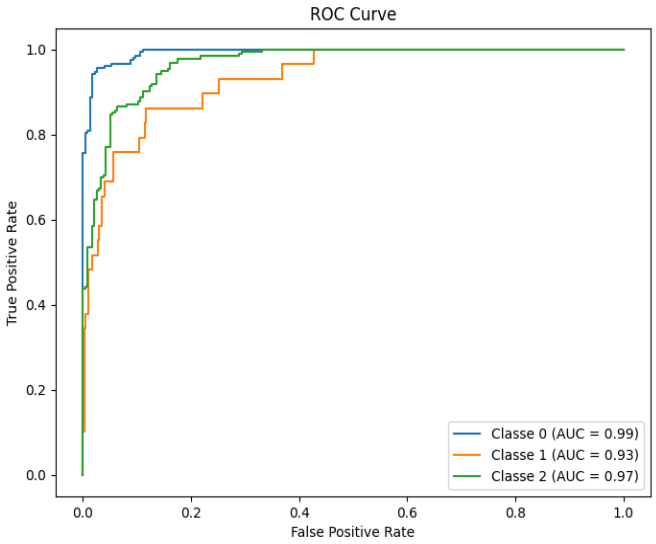

Figura 4: *ROC Curve - Model 1,* **Fonte:** Autor

5.2 *InceptionV3*

No início do treinamento, o modelo já conseguia captar alguns padrões básicos, atingindo 70,77% de acurácia no treino. No entanto, sua capacidade de generalização ainda era muito baixa, evidenciada pela acurácia de validação de apenas 51,39%, um valor próximo de uma escolha aleatória, além de uma *loss* relativamente alta (1.1720 no treino e 2.1334 na validação). Esses resultados sugeriam que o modelo estava aprendendo, mas sem eficiência suficiente para diferenciar corretamente as classes.

Com o avanço das épocas, o modelo apresentou um crescimento significativo nos seus índices de acurácia, alcançando 93,35% no treino e 95,98% na validação, acompanhado de uma queda expressiva na *loss* (0.1122 na validação). No entanto, um alerta surgiu: a precisão caiu para 47,43%, um indício de que o modelo estava enfrentando dificuldades para manter um equilíbrio entre as classes, possivelmente cometendo um número elevado de falsos positivos.

No final do treinamento, os números foram ainda mais impressionantes: 98,47% de acurácia no treino e 99,38% na validação, com uma *loss* extremamente baixa (0.0349). Apesar desses números aparentemente excelentes, a precisão caiu drasticamente para 40,76%, o que sugere que o modelo pode estar sofrendo de *overfitting* severo. Isso significa que ele pode estar memorizando os dados de treino ao invés de aprender padrões generalizáveis, tornando-se menos eficaz para novos exemplos.

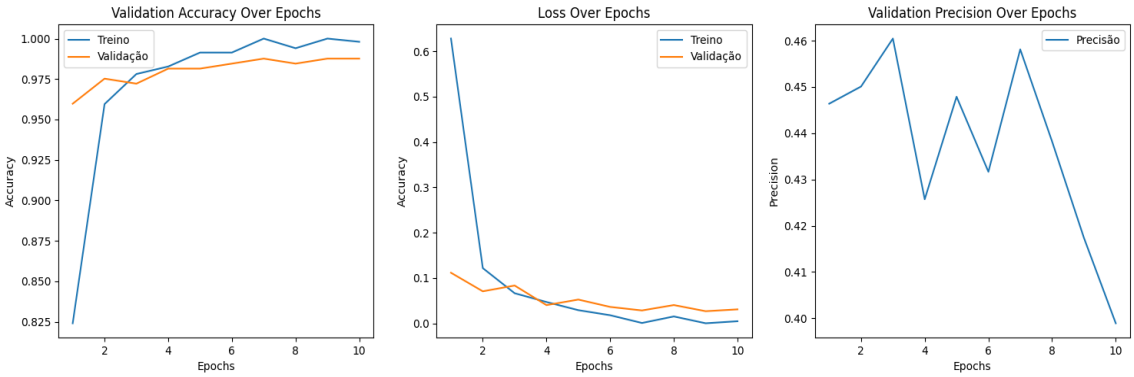

Figura 5: *Accuracy, loss and precision over epochs – Model 2*, **Fonte:** Autor

A matriz de confusão confirma as dificuldades do modelo em distinguir corretamente as classes. O maior problema está na confusão entre *Early Blight* e *Late Blight*, que são frequentemente classificados de forma errada entre si. Essa sobreposição pode indicar que os features extraídos pelo modelo não estão captando nuances suficientes para diferenciar essas doenças, o que compromete a acurácia real da classificação.

Outro problema grave foi a classificação da classe *Healthy* (saudável). O modelo teve extrema dificuldade em identificar corretamente plantas sem doenças, classificando a grande maioria como sendo afetadas por alguma praga. Apenas três amostras foram corretamente identificadas como saudáveis, enquanto a maioria foi erroneamente atribuída às classes de doenças. Isso pode indicar um desequilíbrio no conjunto de dados, onde a classe "*Healthy*" pode estar sub-representada, fazendo com que o modelo aprenda padrões mais fortes para as doenças, mas negligencie a classe saudável.

Além disso, esse padrão de erro reforça a hipótese de que o modelo não apenas sofre de *overfitting*, mas também pode estar desbalanceado, favorecendo categorias específicas em detrimento de outras.

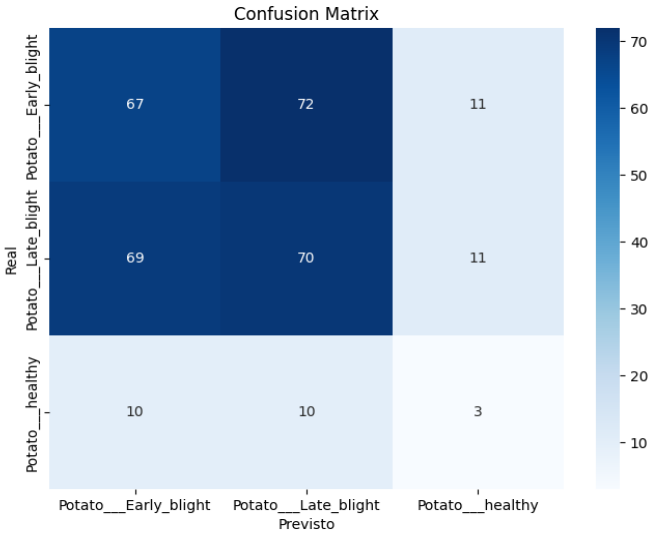

Figura 6: *Confusion Matrix – Model 2*, **Fonte:** Autor

A curva ROC reforça ainda mais as limitações do modelo. Os valores de AUC (Área Sob a Curva) ficaram próximos de 0.47 para Early Blight e Late Blight, e 0.50 para Healthy. Esses números indicam que o modelo não está

conseguindo distinguir bem entre as classes, pois um AUC de 0.50 equivale a um classificador aleatório.

O fato de a curva estar próxima da linha diagonal da aleatoriedade (linha pontilhada) demonstra que o modelo não está aprendendo padrões úteis o suficiente para fazer previsões confiáveis. Isso sugere um caso grave de *underfitting* – possivelmente devido a um modelo muito simples, um conjunto de dados insuficiente ou um problema na extração de características.

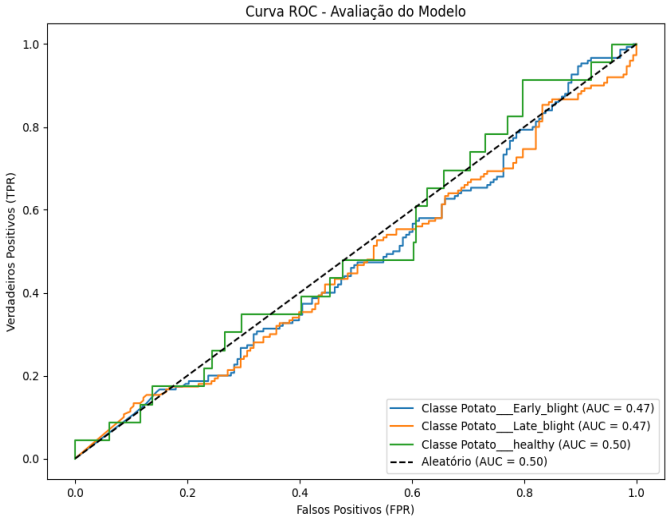

Figura 7: *ROC Curve* – Model 2, **Fonte:** Autor

5.3 CNN 1

O modelo de classificação de doenças na batata apresentou uma evolução consistente ao longo do treinamento, conforme observado nos gráficos de acurácia e perda por época. Inicialmente, a acurácia de treino era relativamente baixa (43,69%), enquanto a de validação já indicava um desempenho superior (73,96%). No entanto, já na segunda época, o modelo demonstrou um salto expressivo, atingindo 72,75% de acurácia de treino e 81,25% na validação, evidenciando um aprendizado acelerado. A partir da terceira época, a acurácia de treino ultrapassou 91%, estabilizando até atingir 96,75% na décima época, enquanto a validação se manteve próxima de 94,79%. Apesar dessa evolução positiva, na última época, a acurácia de validação apresentou uma leve queda, sugerindo um possível *overfitting*. Isso significa que o modelo pode estar memorizando padrões específicos dos dados de treino em vez de aprender padrões generalizáveis para novos dados. Esse comportamento fica ainda mais evidente ao analisarmos a *loss*: enquanto a perda de treinamento continuou diminuindo suavemente, a *loss* de validação oscilou e até aumentou ligeiramente no final, reforçando a suspeita de que o modelo está perdendo sua capacidade de generalizar.

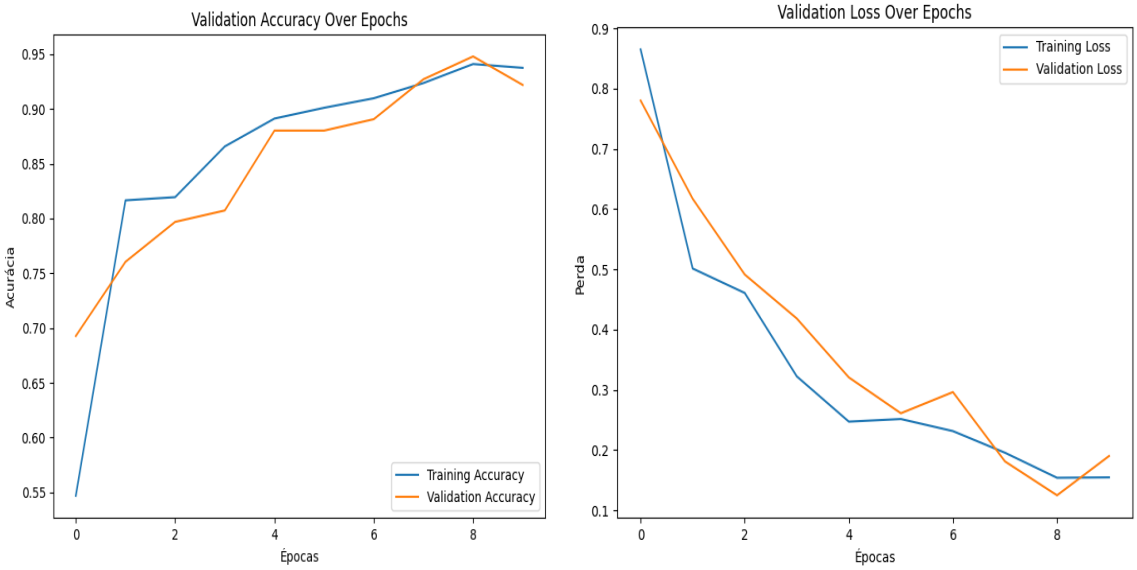

Figura 8: *Accuracy and Loss* – Model 3, **Fonte:** Autor

A matriz de confusão fornece mais detalhes sobre esse comportamento, mostrando onde o modelo acerta e onde ocorrem os principais erros. Para a classe *Early Blight*, o desempenho foi excelente, com 130 acertos e apenas 3 erros. No entanto, na classe *Late Blight*, algumas amostras foram erroneamente classificadas como *Healthy,* um problema que pode comprometer a confiabilidade do modelo no campo. Esse erro se agrava ao observar a classe *Healthy*, onde apenas 15 amostras foram corretamente classificadas, enquanto o restante foi confundido com plantas doentes. Essa dificuldade na distinção entre plantas saudáveis e doentes pode ter impacto significativo na tomada de decisões agrícolas, especialmente na detecção precoce de doenças para evitar perdas na colheita.

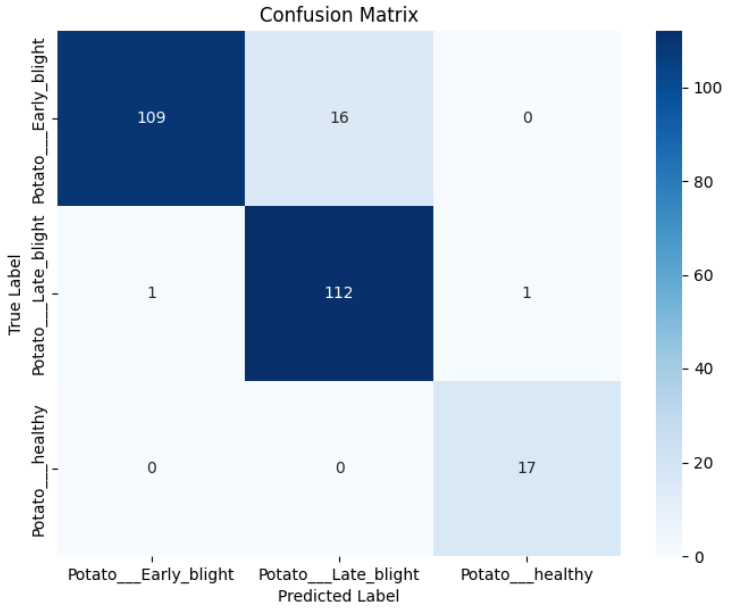

Figura 9: *Confusion Matrix* – Model 3, **Fonte:** Autor

A curva ROC reforça essa análise, evidenciando que o modelo ainda apresenta dificuldades na separação das classes. Os valores de AUC variam entre 0.60 e 0.69, indicando que ele não consegue distinguir perfeitamente entre plantas saudáveis e doentes. Embora a classe *Healthy* tenha tido o melhor desempenho, as classes *Early Blight* e *Late Blight* apresentaram AUCs mais baixos, demonstrando que o modelo frequentemente as confunde. Além disso, as curvas ROC estão relativamente próximas da diagonal de aleatoriedade (AUC = 0.5), indicando que há margem para melhorias.

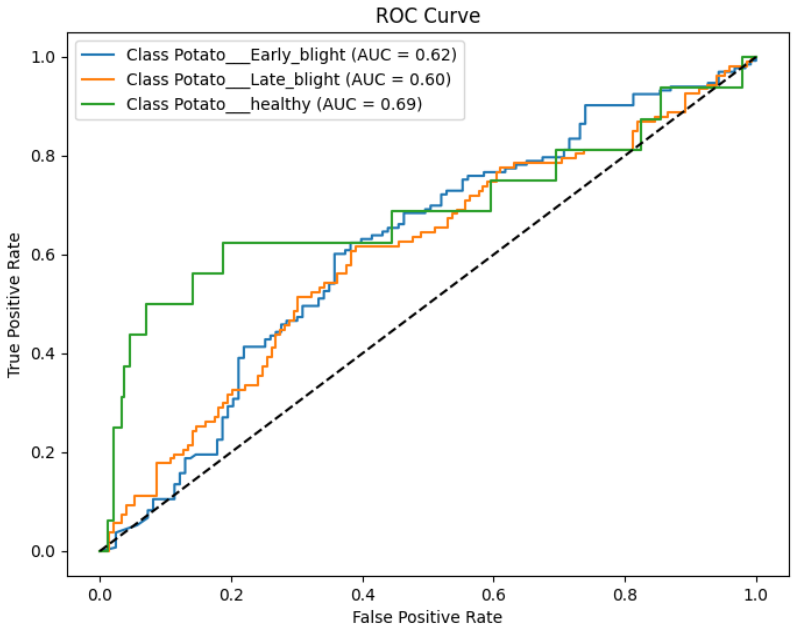

Figura 10: *ROC Curve* – Model 3, **Fonte:** Autor

5.4 CNN - Modelo 4

O treinamento do modelo de classificação de doenças na batata apresentou uma evolução consistente ao longo das épocas, com um aumento progressivo da acurácia e redução da perda tanto no conjunto de treino quanto no de validação. Inicialmente, o modelo começou com 52,81% de acurácia no treino e 75,63% na validação, demonstrando um aprendizado rápido dos padrões básicos dos dados. A partir da segunda época, houve um salto expressivo no desempenho, atingindo 82,07% de acurácia no treino e uma leve melhora na validação. Na quinta época, o modelo já alcançava 94,88% no treino e 92,19% na validação, consolidando um bom aprendizado e reduzindo significativamente a perda.

Entretanto, a partir da sétima época, surgiram indícios de *overfitting*, quando o modelo começa a memorizar padrões específicos dos dados de treinamento e perde a capacidade de generalização. Esse comportamento fica evidente ao observar que a acurácia de treino continuou subindo, chegando a 99,06% na nona época, enquanto a de validação começou a oscilar e caiu para 92,19% na última época. Paralelamente, a perda de validação, que havia diminuído até 0.1012, voltou a aumentar para 0.2255, reforçando que o modelo pode estar se ajustando demais aos dados de treino e tendo dificuldades para generalizar.

A análise dos gráficos e métricas complementa essa avaliação, revelando que o modelo teve um desempenho sólido na classificação das três categorias de batatas: *Early Blight*, *Late Blight e Healthy*. A curva de acurácia mostra que o modelo aprendeu bem os padrões dos dados, atingindo quase 100% de precisão no treino, enquanto a validação se estabilizou em torno de 90%. Esse comportamento confirma o leve *overfitting* detectado anteriormente. Além disso, a curva de perda (*loss*) reforça essa observação, pois, enquanto a perda de treino diminui continuamente, a de validação apresenta oscilações, um indicativo clássico de ajuste excessivo aos dados de treinamento.

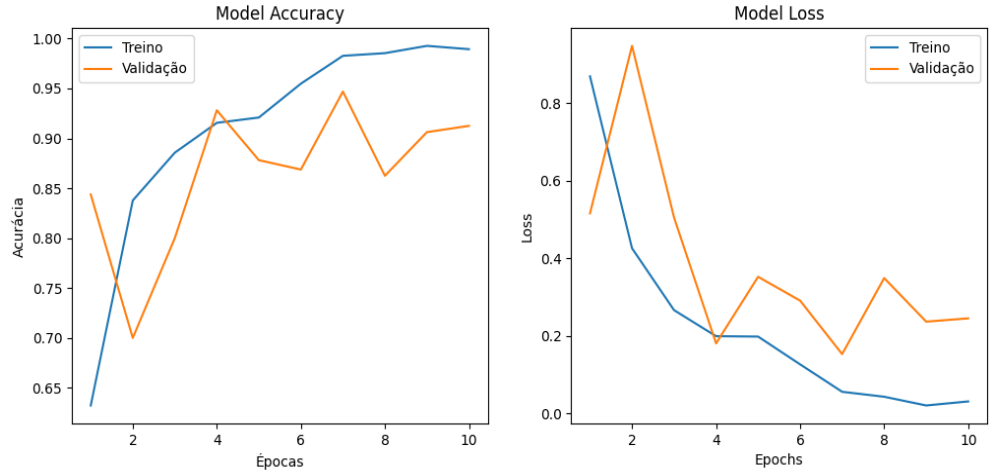

Figura 11: *Accuracy and loss over epochs* – Model 4, **Fonte:** Autor

A curva ROC reforça a boa capacidade do modelo em distinguir as classes, com áreas sob a curva (AUC) de 0.92 para *Early Blight* e *Late Blight* e 0.95 para *Healthy*. Isso indica que o modelo difere bem entre as classes, especialmente na identificação de batatas saudáveis. No entanto, as curvas das duas classes de doentes estão muito próximas, sugerindo que há alguma sobreposição entre os padrões dessas doenças, o que pode gerar erros na classificação.

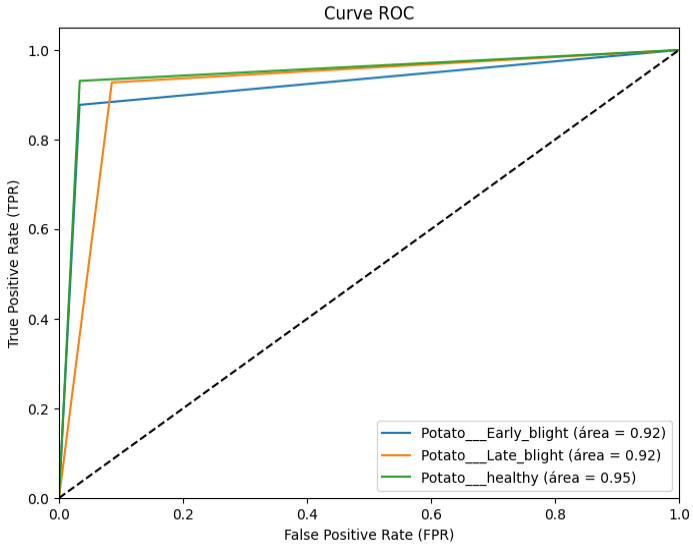

Figura 12: *ROC Curve* **–** Model 4**, Fonte:** Autor

Essa dificuldade na separação entre doenças fica ainda mais clara ao observar a matriz de confusão. O modelo classifica corretamente a maioria dos exemplos, mas comete erros principalmente entre *Early Blight* e *Late Blight*, com 14 amostras de *Early Blight* classificadas erroneamente como *Late Blight* e 5 casos do contrário. Isso faz sentido, pois ambas as doenças compartilham sintomas visuais semelhantes. Já a classe *Healthy* teve um desempenho quase impecável, com apenas dois erros, indicando que o modelo dificilmente confunde batatas saudáveis com doentes, um ótimo indicativo de confiabilidade nesse aspecto.

A avaliação final no conjunto de teste resultou em 92,84% de acurácia e 0.2147 de perda, confirmando um bom desempenho, mas evidenciando que a generalização poderia ser otimizada. Para reduzir o *overfitting* e melhorar a separação entre as doenças, algumas estratégias podem ser implementadas. O *Early Stopping* poderia interromper o treinamento no momento ideal, possivelmente na sétima época, evitando que o modelo se ajuste demais aos dados de treino. Além disso, o uso de regularização ajudaria a reduzir a complexidade do modelo, tornando-o mais robusto. Outra abordagem eficaz seria o aumento de dados (*Data Augmentation*), permitindo que o modelo aprenda a partir de um conjunto mais variado de exemplos e reduza a tendência de memorizar padrões fixos. O ajuste fino dos hiperparâmetros também pode

ajudar a melhorar o desempenho, principalmente na diferenciação entre *Early Blight* e *Late Blight.*

De modo geral, o modelo apresenta um desempenho bastante confiável na classificação de doenças da batata, especialmente na distinção entre plantas saudáveis e doentes.

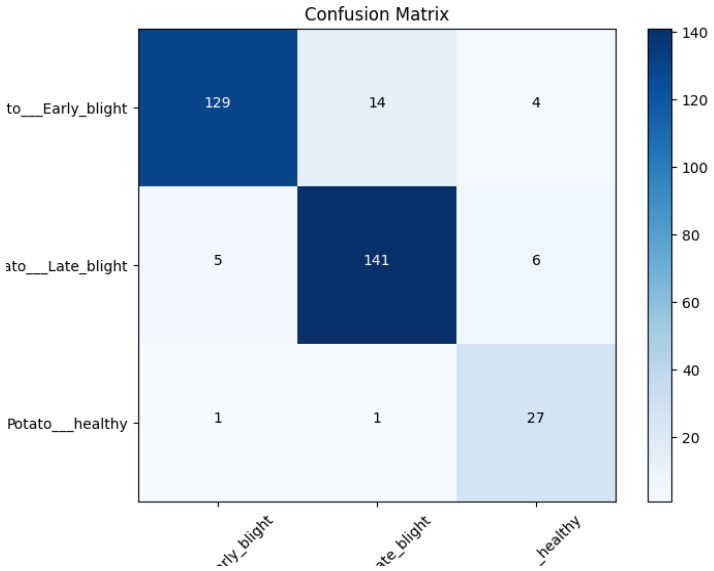

Figura 13: *Confusion Matrix* – Model 4, **Fonte:** Autor

5.5 *ResNet 34*

A análise dos resultados do modelo de classificação aplicado às doenças da batata revela um desempenho notável, evidenciado pela matriz de confusão, pela curva ROC e pela métrica AUC. Esses resultados oferecem uma visão clara sobre a eficácia do modelo na tarefa proposta. No entanto, alguns padrões observados sugerem a necessidade de ajustes para garantir que o modelo seja não apenas preciso, mas também generalizável e robusto a novos cenários.

A matriz de confusão destaca que o modelo apresenta um ótimo desempenho na classificação das três categorias: *Early blight*, *Potato Late blight* and *Potato Healthy*. Para a classe *Early Blight*, o modelo identificou corretamente todos os 152 casos sem cometer erros, evidenciando uma precisão perfeita para essa categoria. No entanto, na classe *Late Blight*, houve 138 acertos, mas com 8 amostras erroneamente classificadas como saudáveis e 2 como *Early Blight*. Já para a classe *Healthy*, todos os 24 exemplos foram corretamente classificados. Apesar desses pequenos desvios na classe *Late Blight*, os resultados gerais são sólidos e demonstram a eficácia do modelo.

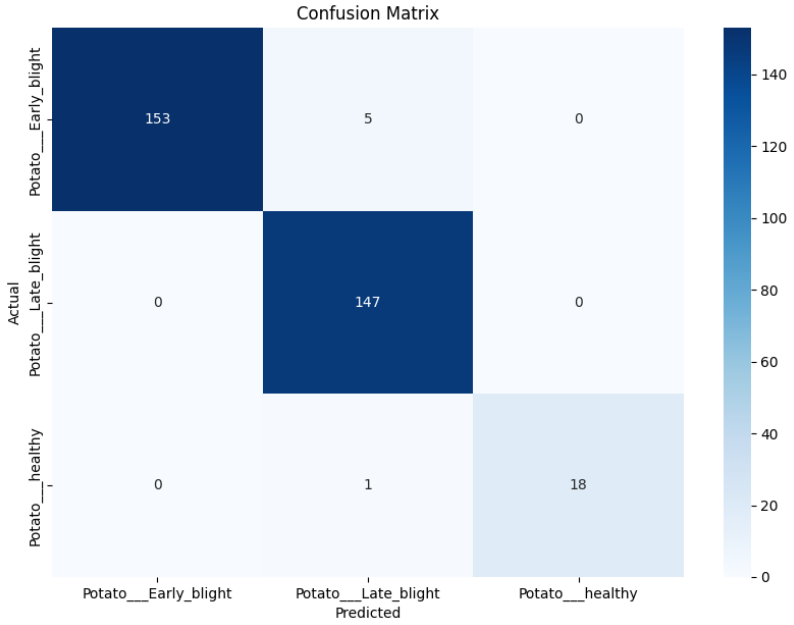

Figura 14: *Confusion Matrix* – Model 5, **Fonte:** Autor

Ao analisar a curva ROC, observa-se que as três curvas para as respectivas classes estão posicionadas no canto superior esquerdo do gráfico, um indicativo claro de que o modelo possui uma alta capacidade de separação entre as classes. A métrica AUC (Área Sob a Curva) complementa essa análise,

com valores de 1.00 para todas as classes, sugerindo um desempenho teoricamente perfeito na discriminação das categorias analisadas. No entanto, a presença de alguns erros na matriz de confusão levanta a questão de um possível *overfitting*, já que um AUC de 1.00 pode indicar que o modelo se ajustou excessivamente aos dados de treinamento.

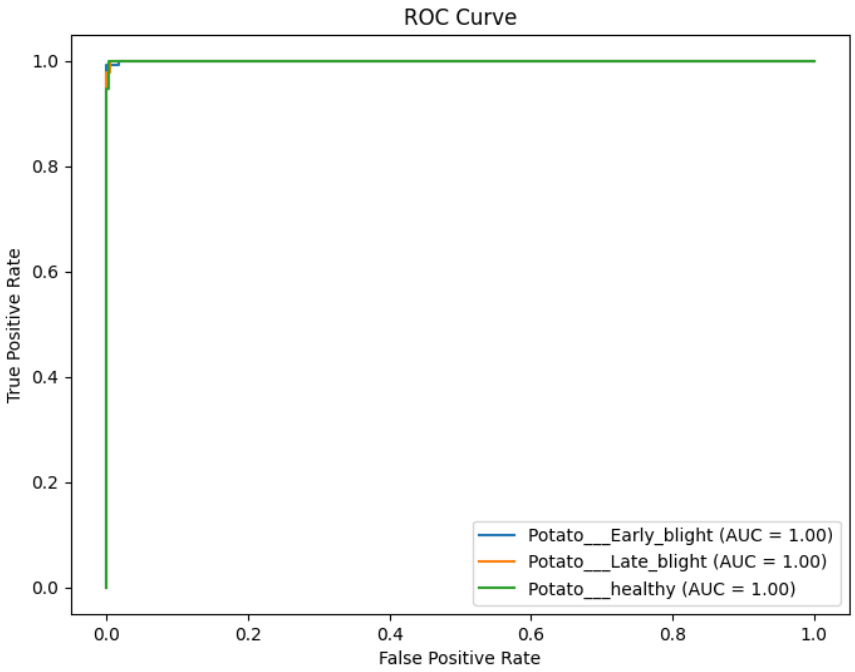

Figura 15: *ROC Curve* – Model 5, **Fonte:** Autor

Essa preocupação é reforçada ao analisarmos o treinamento do modelo ao longo das épocas. Nos primeiros ciclos, a perda de treinamento diminuiu rapidamente, com uma melhoria significativa na acurácia de validação. Contudo, em certos momentos, a validação oscilou, como observado na época 6, onde a perda de validação caiu para 0.0112, apenas para subir abruptamente para 0.3954 na época 8. Esse comportamento sugere que o modelo pode estar sofrendo com instabilidade na aprendizagem, possivelmente devido a uma taxa de aprendizado ainda um pouco alta ou à falta de técnicas mais robustas de regularização.

Além disso, a acurácia de validação, embora alta, apresenta uma leve queda nas últimas épocas. O modelo atinge 99.74% de acurácia na época 6 e 7, mas termina o treinamento com 97.40% na época 10. Isso sugere que continuar treinando após um certo ponto pode não estar trazendo benefícios reais e que o

modelo pode estar começando a memorizar os dados de treino em vez de aprender padrões generalizáveis.

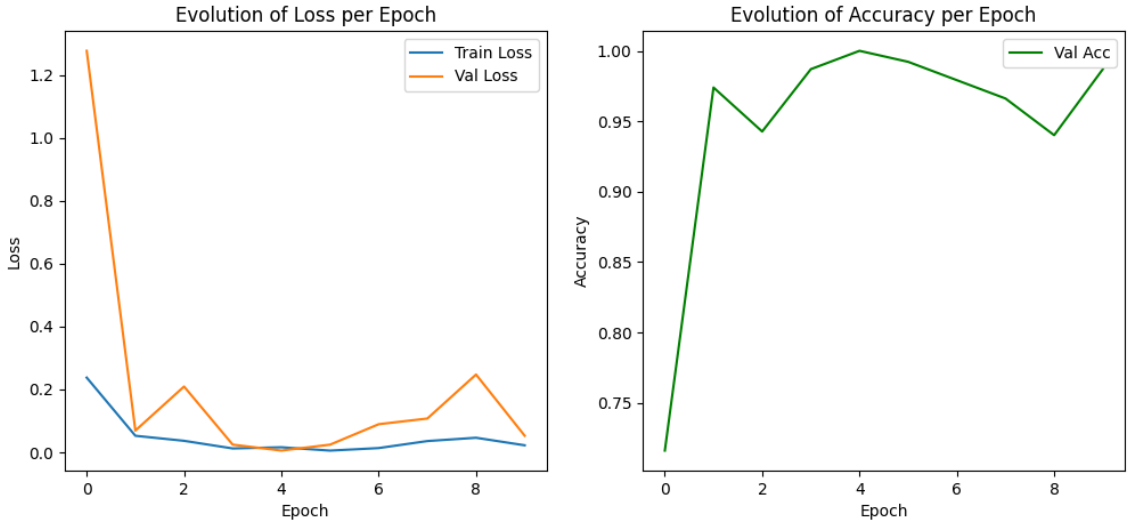

Figura 16: *Loss and Accuracy over epochs* – Model 5, **Fonte:** Autor

Diante disso, algumas estratégias de melhoria podem ser aplicadas para garantir que o modelo seja ainda mais eficiente e confiável. A implementação de *early stopping* pode ajudar a interromper o treinamento no momento ideal, evitando quedas no desempenho. Além disso, o uso de regularização, como dropout ou *L2 regularization*, pode reduzir o risco de *overfitting*. Outro ajuste interessante seria diminuir progressivamente a taxa de aprendizado, tornando a otimização mais estável. Por fim, recomenda-se realizar validação cruzada e testar o modelo com um conjunto de dados completamente novo para avaliar sua capacidade de generalização.

Em resumo, o modelo já demonstra um desempenho impressionante na classificação das doenças da batata, com métricas extremamente favoráveis. Entretanto, ajustes adicionais podem torná-lo ainda mais robusto e aplicável a cenários reais. A chave agora está em refinar o treinamento, validar os resultados de forma rigorosa e garantir que o modelo mantenha sua alta eficácia mesmo diante de dados desconhecidos.

6. CONCLUSÃO

A classificação de doenças na cultura da batata é uma tarefa desafiadora e essencial para a detecção precoce e manejo adequado dessas enfermidades. No presente trabalho, cinco modelos de aprendizado profundo foram desenvolvidos e avaliados, cada um com diferentes arquiteturas: VGG (Modelo 1), *InceptionV3* (Modelo 2), CNN personalizada (Modelos 3 e 4) e *ResNet-34* (Modelo 5). Os resultados indicaram que as performances variaram significativamente entre as abordagens, revelando pontos fortes e limitações que precisam ser considerados para aplicações práticas.

O Modelo 5 (*ResNet-34*) apresentou o melhor desempenho geral, com alta acurácia (97,40%) no conjunto de validação e AUC = 1,00 para todas as classes. A matriz de confusão mostrou uma classificação quase perfeita, destacando-se principalmente na identificação de plantas saudáveis e na diferenciação precisa entre *Early Blight* e *Late Blight*. Esse resultado demonstra a capacidade superior das redes residuais em capturar padrões complexos, oferecendo uma solução robusta e confiável para a tarefa proposta. Pequenos ajustes, como o uso de *early stopping* e técnicas de regularização, poderiam aumentar ainda mais sua capacidade de generalização.

O Modelo 4 (CNN personalizada) também obteve resultados expressivos, alcançando 94,88% de acurácia na validação e um AUC médio de 0,95. Apesar do bom desempenho, sinais de overfitting começaram a surgir após a sétima época, com oscilações na perda de validação. O modelo demonstrou dificuldades principalmente na diferenciação entre *Early Blight* e *Late Blight*, embora tenha classificado corretamente quase todas as amostras de plantas saudáveis. O ajuste fino dos hiperparâmetros e a introdução de data augmentation podem ser estratégias promissoras para otimizar esse modelo.

O Modelo 3 (CNN personalizada) apresentou uma evolução consistente durante o treinamento, mas seu desempenho foi inferior ao dos dois primeiros. Embora tenha atingido 94,79% de acurácia na validação, a análise da matriz de confusão revelou uma taxa mais elevada de erros ao classificar *Late Blight* como *Healthy*, o que pode ser crítico em aplicações práticas. A curva ROC evidenciou

uma menor capacidade de separação das classes, com AUC variando entre 0,60 e 0,69, indicando a necessidade de melhorias no processo de treinamento.

Já o Modelo 2 (*InceptionV3*) mostrou um desempenho promissor nas primeiras épocas, mas sofreu com uma queda drástica de precisão no final do treinamento, resultando em 40,76% de precisão final. A análise detalhada revelou uma alta taxa de falsos positivos e uma dificuldade acentuada na distinção entre *Early Blight* e *Late Blight*, possivelmente devido ao excesso de complexidade da arquitetura em relação ao tamanho do conjunto de dados disponível.

Por fim, o Modelo 1 (*VGG*) apresentou o pior desempenho, com 51,39% de acurácia na validação e um AUC próximo de 0,47, sugerindo um comportamento próximo ao de um classificador aleatório. O modelo não conseguiu capturar os padrões fundamentais das doenças, evidenciando sub ajuste (*underfitting*) e uma limitação severa na extração de características.

Em síntese, a comparação entre os modelos revelou que arquiteturas mais modernas e profundas, como ResNet-34, oferecem vantagens significativas para tarefas de classificação complexas, como a detecção de doenças em plantas. Para futuras melhorias, recomenda-se a utilização de técnicas de regularização, aumento de dados (*data augmentation*) e validação cruzada para reduzir o risco de overfitting e aprimorar a generalização. Assim, o Modelo 5 (*ResNet-34*) se destaca como a melhor alternativa para implementação em sistemas reais de monitoramento agrícola, proporcionando maior confiabilidade e precisão na classificação das doenças da batata.

7. REFERÊNCIAS BIBLIOGRÁFICAS

EMPRESA BRASILEIRA DE PESQUISA AGROPECUÁRIA – EMBRAPA. Doenças fúngicas em cana-de-açúcar. Disponível em: https://www.embrapa.br/agencia-de-informacao-tecnologica/cultivos/cana/producao/manejo/fitossanidade/doencas/doencas-fungicas. Acesso em: 25 ago. 2024.

INTERNATIONAL BUSINESS MACHINES – IBM. Deep learning: o que é e como funciona. Disponível em: https://www.ibm.com/br-pt/topics/deep-learning. Acesso em: 07 set. 2024.

HAFIZ, Nouman. Potato plant diseases data. Disponível em: https://www.kaggle.com/datasets/hafiznouman786/potato-plant-diseases-data. Acesso em: 01 set. 2024.

TÖFOLI, J.G.; DOMINGUES, R.J. Doenças fúngicas. In: BRANDÃO FILHO, J.U.T.; FREITAS, P.S.L.; BERIAN, L.O.S.; GOTO, R., comps. Hortaliças-fruto [online]. Maringá: EDUEM, 2018. p. 271-313. ISBN 978-65-86383-01-0. Disponível em: https://doi.org/10.7476/9786586383010.0010. Acesso em: 04 set. 2024.

Werbos, P. (1974). Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD thesis, Harvard University, Cambridge, MA.

Almeida, Fausto et al. "The Still Underestimated Problem of Fungal Diseases Worldwide." Frontiers in microbiology vol. 10 214. 12 Feb. 2019, doi:10.3389/fmicb.2019.00214.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, p. 436-444, 2015.

IBM. Redes Neurais Convolucionais (CNNs). Disponível em: https://www.ibm.com/br-pt/topics/convolutional-neural-networks. Acesso em: 26 ago. 2024.

SARMAH, Priyanshu; DAS, Rupam; DHAMIJA, Sachit; BILGAIYAN, Saurabh; MISHRA, Bhabani Shankar Prasad. Deep learning for computer vision. In: **BILGAIYAN, Saurabh (Org.).** *Deep learning and its applications*. 1. ed. [S.l.]: ScienceDirect, 2022.

MBUNGE, Elliot; METFULA, Andile S. Operações sustentáveis e computadores. 2021.