
RELATÓRIO PARCIAL DE INICIAÇÃO CIENTÍFICA

Período de setembro de 2023 à agosto de 2024

*Arquitetura de Redes Neurais Profundas - VGG - Aplicada
no Diagnóstico de Lesão de Pele*

Supervisor: Prof.º Dr. Ademar Takeo Akabane

Programa de Pós-Graduação em Gestão de Redes de Telecomunicações

<http://lattes.cnpq.br/6781874728187325>

ademar.akabane@puc-campinas.edu.br

Candidato: Amanda Soares da Silveira– **RA:** 21018595

Curso de Engenharia de Computação

Ano de início: 2021 – **Ano de conclusão:** 2025

<http://lattes.cnpq.br/7614054443983662>

amanda.ss5@puccampinas.edu.br

Grupo de Pesquisa: Sistemas Inteligentes

Linha de Pesquisa: Tecnologia da Informação Aplicada a Serviços em Redes de Telecomunicações

Área de Tecnologias Prioritárias do MCTIC: Tecnologias para o Desenvolvimento Sustentável

Modalidade: FAPIC/Reitoria

*Esse Plano de Trabalho de Iniciação Científica está vinculado ao Plano de Trabalho de Pesquisa do docente pesquisador intitulado de **Sistemas de Auxílio ao Diagnóstico Médico Aplicando Redes Neurais Profundas para Classificação de Lesões de Pele***

Campinas

Agosto de 2024

Sumário

1	Introdução	1
2	Objetivos	2
3	Metodologia Específica	2
3.1	Atividades Previstas no Cronograma Original	3
3.2	Cronograma do Trabalho	4
3.3	Atividades Realizadas	4
4	Resultado Obtidos	5
4.1	Preparação do Ambiente de Produção	5
4.2	Organização e Divisão dos Dados	6
4.2.1	Treinamento e Teste	6
4.2.2	Divisão Interna para Validação	7
4.2.3	Divisão Aleatória dos Dados	7
4.2.4	Normalização dos Dados	7
4.3	Balanceamento do Banco de Dados por Sobreamostragem	8
4.4	Treinamento e Validação do Modelo VGG	9
4.5	Resultados e Discussão sobre o Desempenho	10
5	Discussão	11
6	Conclusão	12
	Referências	15

1 Introdução

De acordo com o Instituto Nacional de Câncer (INCA) são esperados 704 mil casos novos de câncer no Brasil para cada ano do triênio 2023-2025 [1], com destaque para as regiões Sul e Sudeste, que concentram cerca de 70% da incidência [2]. Neste mesmo estudo, destaca-se que o câncer de pele é o tipo mais comum no Brasil, representando cerca de 30% de todos os casos de câncer no país. Vale destacar que há vários tipos de câncer de pele, tais como, melanoma, carcinoma basocelular e carcinoma espinocelular, entre outros. Dos citados, o melanoma é considerado o mais letal, pois apesar de representar menos de 5% dos tumores de pele, é o responsável por mais de 90% das mortes [3, 4]. Apesar do melanoma ter altas taxas de letalidade, quando descoberto em fase inicial pode ser facilmente curado.

Normalmente, quando há suspeita de câncer de pele, a primeira etapa do diagnóstico é realizado por meio de inspeção visual de um especialista. Um ponto importante a ser destacado nessa etapa é a dificuldade de distinguir entre melanoma e nevos (pintas) melanocíticos, especialmente nos estágios iniciais da doença. Então, parte-se para a segunda etapa, que é uma análise dermatoscópica. Nessa etapa, realiza-se análises de imagens dermatoscópicas com intuito de aumentar a chance de descoberta de melanomas malignos de forma precoce. Para se ter uma ideia de números, a acurácia de diagnóstico por inspeção visual **não auxiliada** é de aproximadamente 60% [5] e aumentando para 75% com o auxílio de dermatoscopias [5, 6].

Por um lado, até mesmo por especialistas, realizar a classificação manualmente por meio de análise dermatoscópica das lesões de pele é muito custoso e difícil de ser realizado, devido a grande variedade de manchas que atualmente existem. Por outro lado, a rápida detecção é essencial para a cura, uma vez que a taxa de sobrevivência de mais de 95% nos estágios iniciais da doença cai para menos de 15% nos estágios mais avançados.

A *aprendizagem* é uma das funções mais importantes para os seres humanos, animais e sistemas inteligentes. Os humanos e os animais aprendem com a experiência adquirida, nesse sentido, a aprendizagem é um processo pelo qual o conhecimento é adquirido por meio de experiências passadas. Sistemas inteligentes aplicam técnicas de aprendizado de máquina para simular o aprendizado humano, permitindo que os computadores adquiram conhecimento por meio de algoritmos e dados do mundo real.

Aprendizado de máquina é uma área da Inteligência Artificial que tem como objetivo o desenvolvimento de técnicas computacionais que permitem a construção de sistemas com capacidade de adquirir conhecimento de forma automática a partir de dados de entrada não-aleatórios (ou exemplos). Vale destacar que essas técnicas são possíveis porque tais dados possuem um padrão e assim permitindo a extração de generalizações. Então com esses dados, é possível obter importantes aspectos sobre o conjunto de dados no qual o modelo foi treinado, veja a Figura 1.1.

Uma subárea de ML é *Deep Learning* - DL (aprendizado profundo) que está sendo aplicada

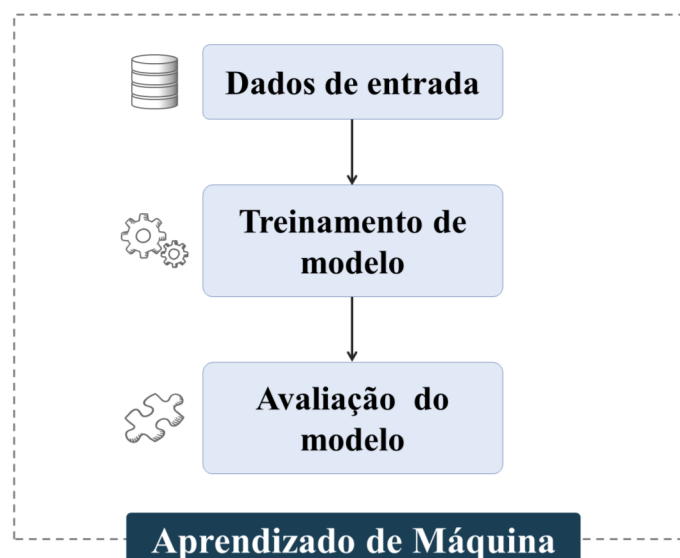


Figura 1.1: Arquitetura geral de aprendizado de máquina [7]

em diversas áreas do conhecimento, desde a área médica [8] até a área da engenharia civil [9]. Em DL os modelos organizados em camadas, onde cada camada utiliza os dados de saída da camada anterior, como dados de entrada. Dessa forma, uma camada posterior na rede é resultado de transformações lineares e não lineares das camadas anteriores a ela [10]. Para a aplicações de visão computacional e de processamento de imagem, as arquiteturas de redes neurais profundas de maior uso são as *Convolutional Neural Networks* – CNN (redes neurais convolucionais), propostas por Yann LeCun e demais autores [11].

Os recentes avanços das CNN possibilitaram o uso dessas redes na automação de diagnósticos médicos por meio de imagens. Vale destacar que, que algumas aplicação de CNN em diagnósticos médicos chegou a ultrapassar o desempenho de médicos especialistas, por exemplo, diagnóstico de retinopatia diabética em imagens de retina [12, 13].

2 Objetivos

O presente Plano de Trabalho tem como objetivo geral realizar experimentos relacionados ao reconhecimento de padrões em imagens dermatoscópicas para a detecção de melanoma utilizando a arquitetura de aprendizado profundo VGG. Além disso, pretende-se aplicar a técnica de *transfer learning* com intuito de promover um aumento no desempenho do modelo utilizado.

3 Metodologia Específica

Esta seção foi dividida em três seções: Atividades Previstas no Cronograma Original (Seção 3.1), Cronograma do Trabalho (Seção 3.2) e Atividades Realizadas (Seção 3.3).

3.1 Atividades Previstas no Cronograma Original

Afim de atingir o objetivo proposto para esta pesquisa, foram propostas as seguintes atividades:

- **A1 - Verificação do estado da arte:** Estudo dos trabalhos existentes na literatura, por meio de *surveys*, *journals* e artigos. Pode-se destacar dois objetivos principais da revisão da literatura: (i) aprofundar os conhecimentos de técnicas e conceitos em arquiteturas de redes neurais profundas VGG e (ii) identificar as lacunas em aberto. Além do mais, realizar reuniões periódicas com o supervisor;
- **A2 - Participação no Encontro de Iniciação Científica e Encontro de Iniciação em Desenvolvimento Tecnológico e Inovação:** A participação consolida a missão da Universidade de qualificar a graduação e proporcionar ao estudante seu engajamento na pesquisa científica;
- **A3 - Obtenção do conjunto de dados e estudo da arquitetura VGG:** Nesta etapa, tem como objetivo obter um conjunto de dados de imagens de lesão de pele, mas especificamente o conjunto de dados ISIC-2017¹ (International Skin Imaging Collaboration) e também o estudo desse conjunto de dados. Além disso, familiarizar-se com classificador de imagens utilizando as redes neurais convolucionais;
- **A4 - Entrega do Relatório Parcial:** Escrita do Relatório Parcial detalhando as atividades realizadas até o momento da escrita;
- **A5 - Implementação do classificador:** Nesta etapa, tem como objetivo implementar um sistema que permita a classificação automatizada de câncer de pele usando imagens digitais com o apoio da arquitetura VGG;
- **A6 - Avaliação quantitativa e qualitativa:** Avaliação quantitativa e qualitativa serão realizadas com o objetivo de avaliar a solução proposta deste plano de pesquisa. Dessa forma, a escolha das características a serem avaliadas serão definidas e, também examinadas quais delas podem influenciar no desempenho da solução;
- **A7 - Escrita de artigo científico:** Os resultados obtidos durante o período da Iniciação Científica serão divulgados em eventos científicos. As melhorias das soluções ocorrerão a partir de *feedbacks* das reuniões periódicas, comentários dos revisores e da própria soluções do estado-da-arte;
- **A8 - Entrega do Relatório Final:** Escrita do Relatório Final detalhando todas as atividades desenvolvidas no período de Iniciação Científica.

¹<https://challenge.isic-archive.com/data/#2017>

3.2 Cronograma do Trabalho

Na Tabela 3.1 é apresentado o cronograma das atividades original do Plano de Pesquisa e que foram seguidas durante o decorrer do trabalho. Os círculos preenchidos se referem às atividades realizadas.

Tabela 3.1: Cronograma de trabalho da Iniciação Científica.

		2023		2024			
		Set.~Out.	Nov.~Dez.	Jan.~Fev.	Mar.~Abr.	Mai.~Jun.	Jul~Ag.
Atividades	A1	●	●	●	●	●	●
	A2	●					
	A3	●	●	●			
	A4				●		
	A5			●	●	●	
	A6				●	●	●
	A7					●	●
	A8						●

3.3 Atividades Realizadas

A Atividade 1 (A1) foi concluída com êxito, tendo como objetivos principais aprofundar o conhecimento em técnicas e conceitos de arquiteturas de redes neurais profundas, com ênfase na VGG, e identificar lacunas de pesquisa. Além disso, foram realizadas reuniões periódicas com o supervisor para discussão e orientação.

Durante o intervalo entre a Atividade 1 (A1) e a Atividade 3 (A3), foi executada a Atividade 2 (A2), que compreendeu a apresentação na sessão de pôster da PUC Campinas no XXXVIII Encontro de Iniciação Científica, XIII Encontro de Iniciação e Desenvolvimento Tecnológico e Inovação de 2023, nos dias 24 e 25 de outubro de 2023.

A Atividade 3 (A3) foi concluída conforme planejado, envolvendo a obtenção do conjunto de dados HAM-10000 de imagens de lesões de pele. Além disso, foi realizado um estudo detalhado deste conjunto de dados, juntamente com a familiarização com o classificador de imagens VGG, conforme estipulado pelo cronograma original. Esta etapa contribuiu significativamente para a compreensão dos dados e técnicas necessárias para os próximos passos da pesquisa. Seguida pela Atividade 4 (A4), que diz respeito ao desenvolvimento deste relatório parcial detalhando as atividades realizadas até o momento.

Após a entrega do Relatório Parcial (A4), foram realizadas as etapas da Atividade 5 (A5), que é crucial para o avanço do projeto. Nesta fase, foi implementado um sistema de classificação automatizada de câncer de pele utilizando imagens digitais e a arquitetura VGG. Esta implementação envolveu o treinamento do modelo VGG, ajustes na arquitetura para otimizar

a eficiência computacional, e a aplicação de técnicas de aumento de dados para lidar com o desequilíbrio das classes.

Seguindo a implementação, passamos para a Atividade 6 (A6), onde realizamos uma avaliação quantitativa e qualitativa do sistema. A avaliação envolveu a análise dos resultados obtidos, a precisão do modelo, e a capacidade de generalização para novos dados.

Posteriormente, a Atividade 7 (A7) envolveu a escrita de um artigo científico para divulgar os resultados obtidos até o momento. Os resultados foram apresentados em eventos científicos, e as melhorias das soluções foram baseadas no feedback recebido das reuniões periódicas, comentários dos revisores, e na comparação com o estado-da-arte da área.

Finalmente, a Atividade 8 (A8) a elaboração deste Relatório Final, que detalhará todas as atividades desenvolvidas durante o período de Iniciação Científica

4 Resultado Obtidos

Esta seção está estruturada em dois principais tópicos para proporcionar uma melhor compreensão do trabalho realizado e dos passos necessários para a produção de um modelo de classificação de lesões de pele. Os tópicos abordados são: Preparação do Ambiente de Produção (Seção 4.1), Organização e Divisão dos Dados (Seção 4.2), Balanceamento do Banco de Dados por Sobreamostragem (Seção 4.3), Treinamento e Validação do Modelo VGG (Seção 4.4) e Resultados e Discussão sobre o Desempenho (Seção 4.5)

4.1 Preparação do Ambiente de Produção

O desenvolvimento de um modelo de deep learning, como o proposto neste projeto, é um processo computacionalmente custoso, tanto em termos de espaço de memória quanto de desempenho de hardware. Para superar esses desafios, foi utilizado o Google Colab, uma plataforma gratuita baseada na nuvem oferecida pelo Google. O Google Colab fornece um ambiente de notebook interativo e colaborativo, permitindo a criação e execução de código diretamente no navegador, sem a necessidade de configurar ou instalar qualquer software no computador, garantindo assim um ambiente estável de produção.

O banco de dados HAM10000, essencial para este projeto, foi obtido através de uma API do site Kaggle. Após o download do arquivo, foi necessário organizar os diretórios para garantir que os dados fossem acessados sem problemas de caminho entre arquivos. O arquivo principal contém todo o conteúdo necessário para o treinamento do modelo, incluindo as imagens e um arquivo CSV (*HAM10000_metadata.csv*), que reúne informações relevantes sobre as imagens, como *lesion_id*, *image_id*, *dx*, *dx_type*, *age*, *sex*, e *localization*.

As variáveis presentes no arquivo CSV são relevantes para o treinamento do modelo, pois as imagens não são suficientes para associar uma classificação precisa de lesões de pele cancerí-

genas. Para facilitar o processo de treinamento, foram feitas algumas modificações na estrutura dos dados. As informações foram organizadas em diferentes diretórios, melhorando o fluxo de trabalho.

Além disso, uma nova coluna foi adicionada ao conjunto de dados, mapeando os códigos de diagnóstico originais para suas respectivas descrições textuais, como “Melanoma” e “Nevo melanocítico”. Essa coluna adicional facilita a análise e a interpretação dos dados. Para que o modelo de rede neural convolucional (CNN) pudesse processar os rótulos das classes de maneira eficiente, os tipos de lesões, originalmente representados em texto, foram convertidos para valores numéricos. Este processo de codificação de rótulos transforma cada categoria de lesão em um número inteiro único.

A codificação numérica dos rótulos é essencial por várias razões:

- **Compatibilidade com o Modelo:** Modelos de machine learning, como as CNNs, exigem rótulos numéricos para calcular funções de perda e executar o processo de treinamento.
- **Eficiência:** A utilização de números inteiros é mais eficiente em termos de processamento e memória, melhorando a performance do modelo durante o treinamento.
- **Facilidade de Implementação:** Ferramentas e frameworks utilizados no desenvolvimento de modelos de deep learning, como TensorFlow e PyTorch, requerem rótulos numéricos para o treinamento e avaliação corretos do modelo.

Com essas preparações, os dados estavam prontos para serem utilizados no treinamento da rede neural convolucional.

Além disso, bibliotecas como Matplotlib e TensorFlow foram utilizadas na produção do modelo. O Matplotlib permitiu a análise visual dos dados e a criação de gráficos para avaliar os resultados, enquanto o TensorFlow foi essencial para a importação da arquitetura VGG.

4.2 Organização e Divisão dos Dados

Ao trabalhar com machine learning, é essencial dividir o dataset em três conjuntos distintos: conjunto de treinamento, conjunto de validação e conjunto de teste. O conjunto de treinamento e o conjunto de validação são utilizados durante o processo de treinamento, que é dividido em múltiplas etapas, chamadas de épocas (*epochs*). O conjunto de teste, por outro lado, é reservado para a avaliação final do modelo, após o término do treinamento.

4.2.1 Treinamento e Teste

Inicialmente, o dataset completo foi dividido de forma que 70% das imagens fossem utilizadas para o treinamento e 30% para o teste. O conjunto de treinamento é empregado para que o modelo aprenda a classificar corretamente as lesões de pele a partir das imagens disponíveis. Já o

conjunto de teste é utilizado para avaliar a qualidade do modelo após o treinamento, verificando sua capacidade de generalizar para dados novos, não vistos anteriormente.

Esta distinção entre dados de treinamento, validação e teste é fundamental para o desenvolvimento e avaliação correta de modelos de machine learning, assegurando que o modelo treinado seja robusto e capaz de generalizar bem para novos dados.

4.2.2 Divisão Interna para Validação

Dentro do conjunto de treinamento, foi realizada uma nova divisão para criar um conjunto de validação. Este conjunto de validação é essencial para ajustar o modelo durante o treinamento. Ao longo de cada epoch, o desempenho do modelo é avaliado utilizando os dados de validação. Esse processo permite monitorar a acurácia (*accuracy*) e o erro (*loss*) ao longo do treinamento.

Acurácia (*Accuracy*): Refere-se à porcentagem de classificações corretas realizadas pelo modelo. No contexto deste estudo, estamos classificando lesões de pele, mas a acurácia é uma métrica geral em machine learning, utilizada para entender o desempenho do modelo durante o treinamento. Uma acurácia alta indica que o modelo está realizando previsões corretas com maior frequência.

Erro (*Loss*): Refere-se ao somatório dos erros cometidos pelo modelo durante o treinamento ou validação. Quanto menor o erro, melhor o desempenho do modelo. Monitorar essa métrica é crucial para evitar o *overfitting*, onde o modelo começa a "memorizar" os dados de treinamento em vez de aprender a generalizar para novos dados.

4.2.3 Divisão Aleatória dos Dados

Utilizando Python e TensorFlow para desenvolver o modelo e organizar as imagens, foram empregadas funções prontas que facilitam o processo de separação dos dados. Essas funções incluem uma taxa de randomização que assegura que a divisão entre os conjuntos de treinamento, validação e teste seja realizada de forma aleatória e consistente. Isso evita viés na seleção das amostras e garante que a divisão seja representativa e justa.

4.2.4 Normalização dos Dados

A normalização dos dados é uma etapa crucial no pré-processamento, garantindo que todas as características do dataset estejam na mesma escala. Isso não só facilita a interpretação dos dados, mas também torna mais fácil identificar e visualizar as relações entre diferentes características, permitindo comparações significativas. Para normalizar os dados, as seguintes etapas foram realizadas:

O processo de normalização envolve subtrair a média dos dados e dividi-los pelo desvio padrão. Isso assegura que os dados tenham uma média de 0 e um desvio padrão de 1, resultando

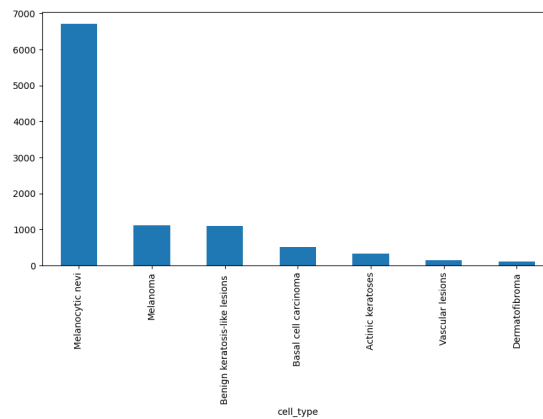


Figura 4.1: Distribuição das classes de lesões no dataset, destacando o desbalanceamento entre lesões benignas e malignas.

em uma distribuição mais uniforme e estável para o treinamento do modelo.

4.3 Balanceamento do Banco de Dados por Sobreamostragem

Durante a análise do dataset, observou-se um desequilíbrio significativo entre as classes de lesões. Por exemplo, há uma quantidade substancialmente maior de imagens de lesões benignas do tipo *melanocytic nevi* em comparação com lesões cancerígenas, como o melanoma. Esse desequilíbrio pode resultar em um modelo que tenha maior precisão em classificar lesões benignas, mas falhe em identificar corretamente as lesões malignas, que são o foco principal da classificação.

Para corrigir esse desequilíbrio sem descartar dados valiosos, optou-se por uma técnica de aumento de dados (*data augmentation*) em vez de simplesmente remover parte dos dados da classe majoritária. O aumento de dados envolve a criação de novas imagens a partir das existentes, aplicando transformações como rotações, zoom e deslocamentos. Embora essas imagens transformadas sejam semelhantes às originais, elas são consideradas novas pelo modelo de rede neural convolucional (CNN). Isso enriquece o conjunto de dados de treinamento e melhora a capacidade do modelo de generalizar.

A generalização é fundamental para garantir que o modelo seja capaz de classificar corretamente lesões de pele que podem aparecer em diferentes ângulos ou sob diversas condições. Por exemplo, se usarmos imagens com a mesma angulação repetidamente, corremos o risco de criar um modelo que só consiga identificar lesões quando estas aparecem exatamente sob essa mesma perspectiva. Modelos de *deep learning*, baseados em estatística e matemática, são particularmente sensíveis aos dados de treinamento. Enquanto um ser humano pode diferenciar uma lesão mesmo que vista sob um ângulo diferente, um modelo treinado de forma inadequada pode falhar nesse aspecto.

Portanto, o aumento de dados se torna um passo de extrema importância, pois ajuda a criar

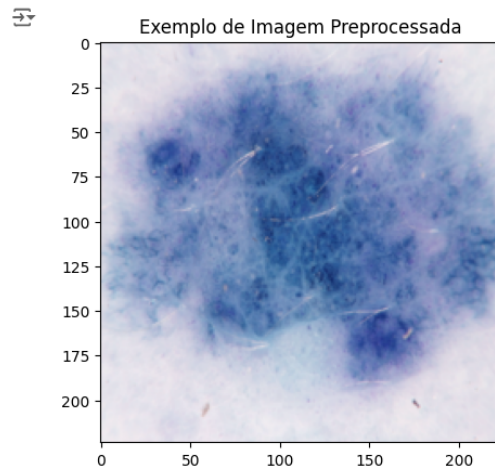


Figura 4.2: Exemplo de imagem de lesão modificada por aumento de dados.

um modelo mais robusto e versátil, capaz de lidar com a variabilidade natural das imagens. Para implementar essa técnica, utilizamos o *ImageDataGenerator* do TensorFlow, configurado com parâmetros específicos para garantir a variabilidade desejada nas imagens:

- **Rotação (*rotation_range=10*):** As imagens podem ser rotacionadas em até 10 graus, aumentando a variabilidade no ângulo das lesões.
- **Zoom (*zoom_range=0.1*):** A aplicação de zoom adiciona variações no tamanho da lesão dentro da imagem.
- **Deslocamento Horizontal e Vertical (*width_shift_range=0.1, height_shift_range=0.1*):** Pequenos deslocamentos das imagens aumentam a variabilidade de posicionamento das lesões.
- **Sem Flip (*horizontal_flip=False, vertical_flip=False*):** Optamos por não aplicar flip horizontal ou vertical, pois poderia distorcer a orientação natural das lesões.

Com essas modificações, o aspecto das imagens fica próximo deste exemplo:

4.4 Treinamento e Validação do Modelo VGG

Durante a fase inicial do nosso projeto de iniciação científica, enfrentamos desafios significativos ao tentar treinar um modelo com o tamanho e complexidade da VGG16. Embora as imagens tenham o tamanho adequado de 224x224 pixels, compatível com a entrada do VGG16, a alta demanda computacional rapidamente preencheu a memória RAM disponível, evidenciando o custo elevado associado a essa arquitetura.

O modelo genérico, devido ao seu grande número de parâmetros treináveis (aproximadamente 138 milhões da VGG16) e à necessidade de grandes camadas fully-connected, rapida-

mente consumiu toda a memória disponível durante o treinamento. Este alto custo computacional tornou evidente a dificuldade de trabalhar com a arquitetura completa do VGG16 em um ambiente com recursos limitados, como o Google Colab.

Para contornar esses problemas, decidimos modificar a arquitetura e aplicar o transfer learning, aproveitando pesos pré-treinados em grandes bases de dados, que ajuda o modelo a assimilar a tarefa de classificação de imagens. Removemos a última camada fully-connected da VGG16, o que nos permitiu reduzir o tamanho das imagens e, consequentemente, o custo computacional. Implementamos então uma camada fully-connected simplificada.

Nossa implementação inclui uma camada de `GlobalAveragePooling2D`, que reduz a dimensionalidade das características extraídas ao gerar uma média global para cada mapa de característica. Isso substitui a necessidade das grandes camadas fully-connected da VGG16 original, reduzindo drasticamente o número de parâmetros e aliviando a carga computacional. Em seguida, adicionamos uma camada fully-connected com 128 unidades e ativação `ReLU`, complementada por uma camada de `Dropout` para reduzir o overfitting. A camada de saída, com ativação `softmax`, mantém a funcionalidade necessária para a classificação, ajustando-se ao número de classes do nosso problema.

Essa abordagem não só simplifica o modelo, mas também permite que ele seja treinado de forma mais eficiente, aproveitando melhor os recursos disponíveis. Comparado à arquitetura original da VGG16, que possui aproximadamente 138 milhões de parâmetros treináveis, nossa solução reduz significativamente o número de parâmetros e melhora a eficiência do treinamento.

Em resumo, a adaptação da VGG16 para um formato mais leve e eficiente foi crucial para superar as limitações de recursos e manter a qualidade do modelo, permitindo-nos avançar de forma prática e eficaz em nossa pesquisa.

4.5 Resultados e Discussão sobre o Desempenho

Após o treinamento do modelo, diversos dados são obtidos para avaliar seu desempenho e chegar a conclusões sobre sua eficácia. Entre as principais ferramentas de análise, destacam-se os gráficos de acurácia e perda (loss), que fornecem uma visão clara sobre como o modelo se comportou ao longo do treinamento:

Esses gráficos mostram como a acurácia do modelo evoluiu e como a perda foi reduzida durante o treinamento, ajudando a avaliar se o modelo está aprendendo de forma adequada e se está convergindo para um bom desempenho.

Outra ferramenta visual importante utilizada para entender o comportamento do modelo é a matriz de confusão. Esta matriz permite uma análise detalhada dos resultados do modelo, comparando os falsos positivos, falsos negativos, verdadeiros positivos e verdadeiros negativos:

Após o uso dos dados de teste para avaliar o desempenho do modelo, obtivemos os seguintes valores de acurácia e erro. Abaixo está o gráfico que ilustra a evolução da acurácia e da perda

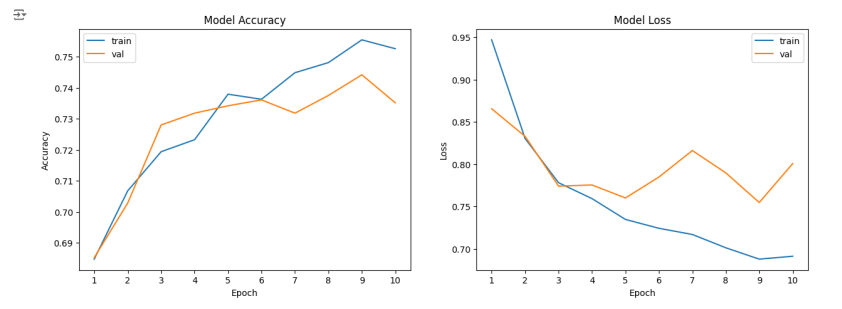


Figura 4.3: Gráfico de Acurácia do Modelo

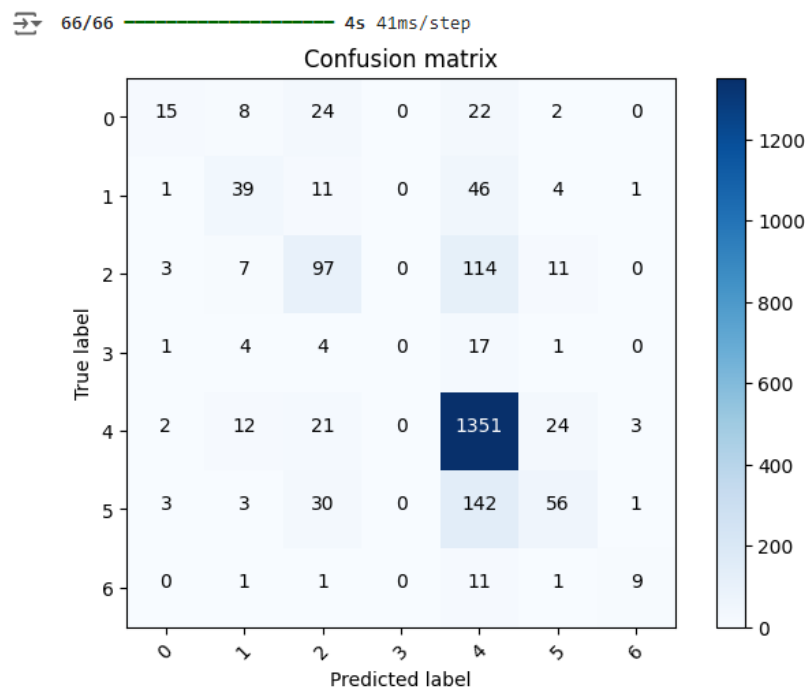


Figura 4.4: Matriz de Confusão

(loss) durante o treinamento e teste do modelo:

Este gráfico mostra como a acurácia e a perda variaram ao longo do treinamento e dos dados de teste. A análise desses valores ajuda a entender a eficácia do modelo em classificar corretamente os casos e a identificar áreas para possíveis melhorias.

5 Discussão

Os resultados obtidos durante o treinamento do modelo indicam que ele apresenta um desempenho medíocre. A acurácia e o erro para os dados de treinamento se aproximam de 0.75 e 0.5, respectivamente, sugerindo que o modelo não está generalizando bem. A análise dos gráficos gerados, especialmente o que compara a acurácia entre os dados de treinamento e validação, mostra que as curvas se encontram e, em seguida, a acurácia do treinamento continua a aumen-

```
Validation: accuracy = 0.747979 ; loss_v = 0.765100
Test: accuracy = 0.752745 ; loss = 0.755994
```

Figura 4.5: Gráfico de Acurácia e Perda (Loss) do Modelo nos Dados de Teste

tar enquanto a de validação se estabiliza ou diminui. Este é um forte indicativo de *overfitting*, que ocorre quando o modelo se ajusta excessivamente aos dados de treinamento, perdendo a capacidade de generalizar para novos dados.

Esse comportamento também é observado no gráfico de erro, onde a diferença significativa entre o erro nos dados de treinamento e nos dados de validação a partir da 5ª ou 6ª época reforça a hipótese de *overfitting*.

Para tentar mitigar esse problema, o número de épocas foi reduzido em um novo teste. Contudo, mesmo com essa alteração, o modelo apresentou desempenho semelhante ao treinamento anterior, com 10 épocas, mostrando que a redução do número de épocas não foi suficiente para melhorar significativamente a generalização do modelo:

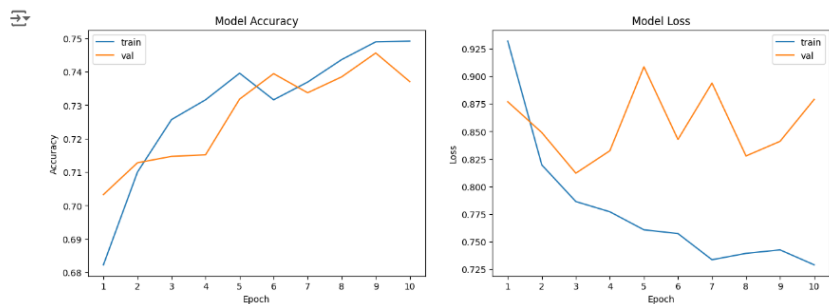


Figura 5.1: Gráfico de acurácia após redução do número de épocas.

Embora o modelo seja capaz de realizar algum nível de classificação, ele ainda está longe de ser adequado para uso em um contexto real. Para ser viável em aplicações médicas, como a classificação de lesões de pele, seria necessário atingir uma acurácia próxima de 0.9 ou superior, acompanhada de um erro significativamente menor. Isso, aliado a uma validação externa rigorosa, seria crucial para garantir a eficácia do modelo em situações práticas.

Em resumo, enquanto a simplificação do VGG permitiu algum progresso no treinamento, os resultados indicam que o modelo, em sua forma atual, não atende aos requisitos necessários para ser útil em um ambiente clínico ou profissional.

6 Conclusão

Com base nos resultados obtidos e nas análises realizadas durante o desenvolvimento e treinamento do modelo VGG simplificado para a classificação de lesões de pele, algumas conclusões

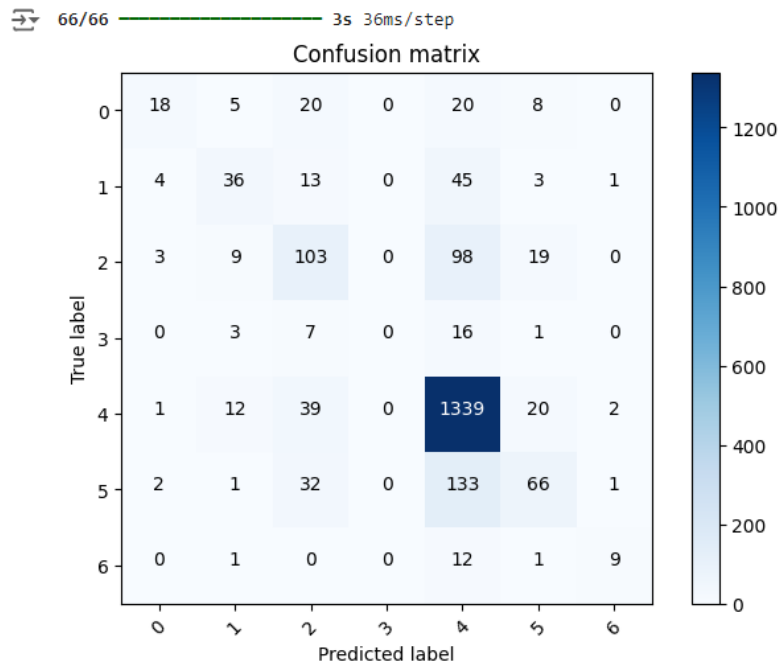


Figura 5.2: Matriz de confusão para os dados de teste.

importantes podem ser feitas. A simplificação da arquitetura do VGG16, embora tenha permitido que o modelo fosse treinado em um ambiente com recursos limitados, como o Google Colab, resultou em um desempenho aquém do necessário para aplicações práticas em contextos médicos.

Um dos fatores que pode ter contribuído para os resultados insatisfatórios foi a remoção da camada totalmente conectada original da VGG16. Essa camada desempenha um papel crucial na capacidade de classificação do modelo, fornecendo uma alta capacidade de representação que pode ser essencial para capturar nuances nos dados de imagens médicas. Ao substituí-la por uma camada totalmente conectada simplificada, é possível que o modelo tenha perdido parte dessa capacidade, resultando em uma acurácia e um desempenho geral inferiores.

Além disso, o modelo apresentou sinais claros de overfitting a partir da 5ª ou 6ª época, com a acurácia nos dados de treinamento se distanciando da acurácia de validação e o erro de treinamento mostrando uma discrepância significativa em relação ao erro de validação. A tentativa de mitigar o overfitting ao reduzir o número de épocas não resultou em melhorias significativas no desempenho geral do modelo, sugerindo que outras abordagens, como o ajuste da arquitetura ou a utilização de técnicas mais avançadas de regularização, podem ser necessárias.

É importante ressaltar, no entanto, que a remoção da camada totalmente conectada original foi uma escolha necessária para lidar com as limitações de recursos computacionais disponíveis. O custo computacional deve sempre ser considerado ao elaborar modelos complexos, especialmente em ambientes com restrições como o Google Colab. Portanto, enquanto a simplificação do modelo permitiu um treinamento viável, os resultados indicam que um equilíbrio cuidadoso

deve ser mantido entre a complexidade do modelo e os recursos computacionais, para evitar comprometer a eficácia do modelo em aplicações práticas.

Em suma, a retirada da camada totalmente conectada original pode ter contribuído para os resultados medíocres, mas a escolha foi necessária devido às restrições computacionais. Futuros trabalhos devem explorar maneiras de otimizar essa troca, talvez através de técnicas de compressão de modelos ou do uso de arquiteturas alternativas que ofereçam um bom equilíbrio entre desempenho e eficiência computacional.

Referências

- [1] “Inca estima 704 mil casos de câncer por ano no brasil até 2025.” <https://www.gov.br/inca/pt-br/assuntos/noticias/2022/inca-estima-704-mil-casos-de-cancer-por-ano-no-brasil-ate-2025>. Acessado: 25/01/2023.
- [2] “Estimativa 2023 – incidência de câncer no brasil.” <https://www.gov.br/inca/pt-br/assuntos/cancer/numeros/estimativa>. Acessado: 25/01/2023.
- [3] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, “Dermatologist-level classification of skin cancer with deep neural networks,” *nature*, vol. 542, no. 7639, pp. 115–118, 2017.
- [4] S. Jain, N. Pise, *et al.*, “Computer aided melanoma skin cancer detection using image processing,” *Procedia Computer Science*, vol. 48, pp. 735–740, 2015.
- [5] H. Kittler, H. Pehamberger, K. Wolff, and M. Binder, “Diagnostic accuracy of dermoscopy,” *The lancet oncology*, vol. 3, no. 3, pp. 159–165, 2002.
- [6] M. Vestergaard, P. Macaskill, P. Holt, and S. Menzies, “Dermoscopy compared with naked eye examination for the diagnosis of primary melanoma: a meta-analysis of studies performed in a clinical setting,” *British Journal of Dermatology*, vol. 159, no. 3, pp. 669–676, 2008.
- [7] E. Y. A. Rodríguez, “Técnicas de aprendizado de máquina para predição do custo da logística de transporte: uma aplicação em empresa do segmento de autopeças,” 2020.
- [8] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, “A survey on deep learning in medical image analysis,” *Medical image analysis*, vol. 42, pp. 60–88, 2017.
- [9] A. Zhang, K. C. Wang, Y. Fei, Y. Liu, S. Tao, C. Chen, J. Q. Li, and B. Li, “Deep learning-based fully automated pavement crack detection on 3d asphalt surfaces with an improved cracknet,” *Journal of Computing in Civil Engineering*, vol. 32, no. 5, p. 04018041, 2018.
- [10] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [11] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Back-propagation applied to handwritten zip code recognition,” *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.
- [12] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, *et al.*, “Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs,” *Jama*, vol. 316, no. 22, pp. 2402–2410, 2016.
- [13] J. Son, J. Y. Shin, H. D. Kim, K.-H. Jung, K. H. Park, and S. J. Park, “Development and validation of deep learning models for screening multiple abnormal findings in retinal fundus images,” *Ophthalmology*, vol. 127, no. 1, pp. 85–94, 2020.