[1]

Algoritimo de Detecção de Retinopatia Diabética baseado em Aprendizado de Máquina

Leonardo Patrocínio dos Reis[a] e Celso Ap.de França[b]

Departamento de Engenharia Elétrica da Universidade Federal de São Carlos

***Resumo*—Neste estudo, foi explorada a aplicação de técnicas de redes neurais e aprendizado de máquina com o intuito de identificar a presença de lesões relacionadas à retinopatia diabética (RD) em imagens de fundo de olho. A RD é uma complicação frequente em indivíduos diabéticos, podendo levar à perda da visão caso não seja detectada e tratada em tempo hábil. A arquitetura do modelo de classificação proposta neste trabalho é composta por dois fluxos de decisão que são concatenados para gerar a classificação final. O primeiro fluxo utiliza uma rede U-Net para segmentar e extrair as veias e vasos sanguíneos da imagem original, seguido por um modelo *Inception* com mecanismo de atenção para a classificação. Já o segundo fluxo processa diretamente a imagem bruta por meio de um modelo *Inception* com mecanismo de atenção. O modelo proposto foi treinado e validado utilizando três conjuntos de dados públicos combinados (ARIA, RFMiD e STARE). As ferramentas empregadas no desenvolvimento incluíram Python, TensorFlow, Keras, OpenCV e outras bibliotecas complementares. O modelo final alcançou uma precisão de $95,4\%$ e sensibilidade de $94,87\%$ na classificação das lesões de retinopatia diabética, demonstrando seu potencial para contribuir na detecção precoce e no tratamento adequado desta complicação ocular.**

***Index Terms*—Aprendizado de máquina, *Inception*, Classificação de imagens médicas, Detecção precoce, Imagens de fundo de olho, Mecanismo de atenção, U-Net, Redes neurais, Retinopatia diabética, Segmentação de veias.**

I. Introdução

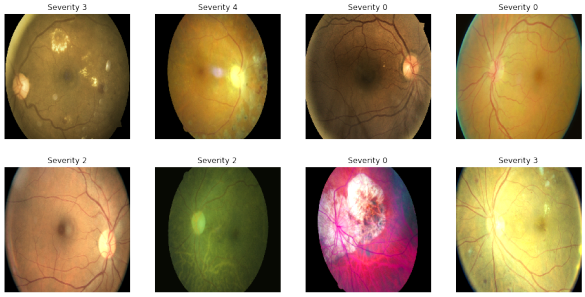
A retinopatia diabética (RD) é uma complicação ocular frequente em indivíduos diabéticos e é caracterizada pelo dano aos vasos sanguíneos da retina causado por níveis elevados de glicose no sangue. A RD, quando não diagnosticada precocemente e sem um tratamento adequado, pode levar a perda de visão e, eventualmente, à cegueira. Atualmente, a RD é uma das principais causas de perda de visão em adultos em todo o mundo, deixando claro a importância de sua detecção precoce [1].

A análise das imagens de fundo de olho é uma técnica comum para diagnosticar a RD, permitindo aos médicos identificar e avaliar a presença de lesões na retina. No entanto, a interpretação das imagens de fundo de olho requer profissionais especializados, como oftalmologistas, que nem sempre estão disponíveis em todas as áreas da saúde, especialmente em regiões remotas e subdesenvolvidas. Além disso, a crescente prevalência de diabetes exige soluções escaláveis para diagnosticar e tratar as complicações associadas de maneira eficiente [2].

A RD apresenta diferentes estágios de desenvolvimento, cada um com características e implicações específicas para a visão. No estágio 1, chamado de retinopatia diabética não proliferativa leve, ocorre o surgimento de microaneurismas, pequenos inchaços nos vasos sanguíneos da retina. Nesse estágio, embora possam causar vazamento de fluido e inchaço da mácula, geralmente não há sintomas claros. No estágio 2, a retinopatia diabética não proliferativa moderada, os vasos sanguíneos incham ainda mais, bloqueando o fluxo sanguíneo e a nutrição adequada da retina, podendo causar visão embaçada se houver acúmulo de líquidos na mácula. No estágio 3, a retinopatia diabética não proliferativa grave, um grande número de vasos sanguíneos fica bloqueado, diminuindo significativamente o fluxo sanguíneo e estimulando o crescimento de novos vasos na retina. Esses vasos são frágeis e podem causar inchaço retiniano, visão embaçada, pontos escuros e até perda de visão em áreas específicas. Caso vazem na mácula, pode ocorrer perda súbita e permanente da visão. No estágio 4, a retinopatia diabética proliferativa, os novos vasos sanguíneos continuam a crescer, formando tecido cicatricial que pode levar ao descolamento da retina. Isso resulta em visão embaçada, redução do campo visual e até cegueira permanente [3].

Portanto, fica evidente que os sintomas da RD podem ser encontrados tanto na retina quanto nas veias e vasos sanguíneos que compõem o nervo óptico. A Figura 1 exibe imagens de fundo de retina com diferentes estágios da doença.

Figura 1: Imagens de fundo de retina com diferentes graus de gravidade



Fonte: Autor

Neste contexto, o uso de técnicas de aprendizado de máquinas e redes neurais têm ganhado destaque na área da saúde, proporcionando soluções para automatizar e melhorar a precisão dos diagnósticos de várias condições médicas, incluindo a RD. A arquitetura *Inception* V2, especificamente, tem sido amplamente aplicada em tarefas de classificação e análise de imagens médicas, graças à sua capacidade de extrair características de diferentes escalas e hierarquias [4]. Já a arquitetura U-Net foi desenvolvida inicialmente para segmentação de imagens médicas, sendo muito utilizada para identificação de tumores cerebrais e lesões em órgãos [5].

[1]
[a] E-mail autor: leonardo.patrocinio@estudante.ufscar.br
[b] E-mail orientador: celsofr@ufscar.br

Além disso, os mecanismos de atenção têm sido incorporados às redes neurais para auxiliar na identificação e localização das áreas mais relevantes das imagens, contribuindo para uma melhor detecção e classificação das lesões [6].

O objetivo deste trabalho é explorar a aplicação de técnicas de redes neurais, aprendizado de máquina e processamento de imagem com o intuito de identificar a presença de lesões relacionadas a diferentes graus de retinopatia diabética em imagens de fundo de olho. A arquitetura do modelo de classificação proposto neste trabalho é composta inicialmente por um pré tratamento da imagem onde é aplicado uma máscara para realce de bordas e detalhes e em seguida por dois fluxos de decisão independentes que são concatenados no final para uma resposta mais assertiva.

O primeiro fluxo utiliza uma rede U-Net, como extratora de características, para segmentar e extrair as veias e os vasos sanguíneos da imagem pré processada, seguido por um modelo Inception com mecanismo de atenção para a classificação. Dessa forma, apenas são extraídas características importantes presentes apenas no nervo óptico para a tarefa de classificação.

O modelo U-Net foi treinado a partir do *dataset* STARE, conjunto de dados composto por cerca de 20 imagens de fundo de retina e imagens de rótulo, contendo apenas as veias e vasos sanguíneos, feitas por médicos especialistas, da imagem original.

Já o segundo fluxo recebe como entrada a imagem pré processada e por meio do modelo *Inception*, também com mecanismo de atenção, gera a classificação. Nesse fluxo é possível capturar, de forma conjunta, os sintomas da RD presentes na imagem bruta.

O modelo proposto foi treinado e validado a partir da junção de três conjuntos de dados públicos: ARIA [7], RFMiD [8] e STARE [9]. Esses conjuntos de dados incluem imagens de fundo de olho rotuladas com diferentes doenças de retina, incluindo os diferentes estágios da RD. Em uma primeira etapa de pré-processamento desses dados, além da união das bases, foi feito um filtro, selecionando apenas imagens de retinas saudáveis e retinas com algum grau de RD (sem distinção de estágio da doença).

II. TRABALHOS RELACIONADOS

Com a recente popularização de técnicas baseadas em *machine learning* e redes neurais profundas somada à crescente de casos de diabetes ao redor do mundo e, portanto, a necessidade de métodos escaláveis e eficientes para diagnosticar e tratar suas complicações, diversos estudos têm surgido no meio acadêmico. Nesta seção é apresentado uma revisão sobre trabalhos relacionados à detecção e classificação de retinopatia diabética.

A. Detecção Automática de Retinopatia Diabética

No artigo de Li et al [10] os autores apresentam um sistema para a detecção de retinopatia diabética usando fotografias de fundo de olho coloridas. O objetivo do estudo foi desenvolver e validar uma ferramenta eficiente e precisa para identificar pacientes que necessitam de acompanhamento.

Os autores propõem um sistema com diversas abordagens combinadas, como segmentação de lesões, extração de características e classificação por meio de algoritmos de aprendizado supervisionado. Os autores obtiveram resultados comparáveis aos dos especialistas humanos. O modelo proposto no trabalho foi treinado com um dataset de 106000 imagens e obtiveram as métricas AUC, sensibilidade e especificidade de 0.955, 92.5%, e 98.5% respectivamente.

Já no trabalho de Shah et al [11] os autores não especificam quais foram as escolhas de arquitetura realizadas, mas apresentam um sistema baseado em CNN que também obteve ótimos resultados na classificação de DR. Os autores atingiram 99.7% e 98.5% de sensibilidade e especificidade respectivamente.

Essas referências destacam que algoritmos baseados em CNN têm potencial para se tornarem uma alternativa promissora na triagem de DR, apresentando desempenho semelhante ao dos especialistas médicos. Isso sugere que tais abordagens podem oferecer soluções escaláveis e eficientes para a identificação e tratamento de pacientes em risco de perda de visão devido à retinopatia diabética.

B. U-Net Para Segmentação de Imagens Médicas

A U-Net é uma arquitetura de rede neural convolucional (CNN) desenvolvida especificamente para segmentação de imagens biomédicas e foi proposto por Ronneberger et al [5].

Após sua publicação, diversos estudos têm utilizado e adaptado a arquitetura U-Net para segmentação de imagens médicas em diferentes domínios. No trabalho de Rehman et al [12] os autores desenvolveram um U-Net 3D para segmentação de tumores cerebrais em imagens de ressonância magnética, alcançando resultados de segmentação de alta qualidade.

Já no trabalho de Xiao et al [13] os autores utilizaram uma versão modificada da U-Net para segmentação de vasos sanguíneos retinianos em imagens de fundo de olho, demonstrando a eficácia da arquitetura na detecção e segmentação de estruturas retinianas.

Em resumo, a arquitetura U-Net tem se demonstrado ser uma abordagem eficiente e versátil para a segmentação de imagens médicas, possibilitando a detecção e delimitação de estruturas em diferentes tipos de imagens.

C. Mecanismos de Atenção no âmbito de Imagens Médicas

Os mecanismos de atenção têm sido cada vez mais utilizados em conjunto com redes neurais convolucionais para melhorar a detecção e localização de lesões em imagens médicas. Estes mecanismos permitem que a rede neural se concentre nas áreas mais relevantes da imagem, melhorando a precisão e a interpretabilidade dos resultados.

No artigo de Schlemper et al [14] os autores apresentam uma abordagem baseada em mecanismos de atenção para melhorar a detecção e segmentação de estruturas em imagens médicas, incluindo a segmentação de vasos sanguíneos em imagens de fundo de olho.

A seção apresentou uma série de estudos que exploram diferentes abordagens do uso de aprendizado de máquina e inteligência artificial no campo de análise de imagens médicas.

No entanto, é importante notar que os algoritmos e abordagens discutidas neste capítulo apresentam apenas uma fração das pesquisas realizadas na área de detecção e segmentação de retinopatia diabética e outras condições médicas. Novas arquiteturas e técnicas continuam sendo desenvolvidas e aprimoradas, e os avanços futuros nesta área têm potencial para oferecer ainda mais melhorias na precisão, eficiência e escalabilidade das soluções de análise de imagens médicas.

III. FUNDAMENTAÇÃO TEÓRICA

A. Pré-Processamento

A etapa de pré-processamento dos dados é uma etapa crucial para projetos de aprendizado de máquina pois permite melhorar a qualidade dos dados, evidenciar as características mais importantes do conjunto e garantir a normalização das características dos dados sendo tratados de acordo com a necessidade do problema.

A primeira etapa de um pré-processamento é a coleta e organização do conjunto de dados e suas respectivas rotulações. Nessa etapa é importante se atentar à qualidade e confiabilidade da fonte do dado, garantindo a entrega de dados fiéis à realidade para o treinamento do modelo. Uma ferramenta bastante utilizada para essa coleta é a linguagem de programação *Python* dada a sua versatilidade e facilidade para tratamento de dados de diferentes formatos.

Em seguida é importante se atentar à finalidade de uso do dado, realizando análises no conjunto, para entender se existem características em comum das imagens que devem ou podem ser evidenciadas a fim de melhorar a capacidade de generalização do modelo [15].

Nessa etapa é comum a normalização dos dados, ajustando os valores numéricos para que todos estejam na mesma escala. Isso é importante, pois características com escalas diferentes podem afetar o desempenho dos algoritmos de aprendizado de máquina. A normalização pode ser realizada de várias maneiras, como *min-max scaling*, *z-score*, *standardization* ou escalonamento robusto.

No caso de normalização de imagens, o mais comumente usado é a normalização de intensidade de pixel, também conhecida como *min-max scaling*. Esta técnica ajusta os valores de todos os pixels da imagem de modo que eles estejam dentro de um intervalo específico, geralmente entre 0 e 1. O funcionamento dessa normalização pode ser expressado de forma geral pela Equação 1.

$$Pn = \frac{(P - V_{min})}{(V_{max} - V_{min})} \tag{1}$$

Onde:

- Pn é o pixel normalizado
- P é o valor original do pixel,
- Vmin é o valor mínimo que o pixel original pode assumir
- Vmax é o valor máximo que o pixel original pode assumir

Normalmente nas imagens RGB os valores de máximo e mínimo para um pixel é de 255 e 0 respectivamente. Logo, a Equação 2 expressa a fórmula utilizada para normalização das imagens.

$$Pn = \frac{P}{255} \tag{2}$$

Essa normalização é útil para algoritmos de *machine learning*, pois garante que todas as características estejam na mesma escala, evitando que características com valores maiores dominem as demais, melhorando a convergência e o desempenho do modelo durante o treinamento.

Outras técnicas bastante utilizadas são aquelas que têm a finalidade de evidenciar uma característica importante do dado como aumento de contraste, correção de iluminação e nitidez e filtragem de cores ou bordas.

Uma das técnicas utilizadas neste trabalho com intuito de melhorar o contraste da imagem é o "mascaramento de nitidez" [16] Esta abordagem envolve a combinação da imagem original com uma versão borrada da mesma, geralmente obtida por meio de um filtro Gaussiano. Ao subtrair a imagem borrada da imagem original, as bordas e detalhes de alto contraste são enfatizados, resultando em uma imagem com maior nitidez e clareza. Para evitar que os valores dos *pixels* se tornem negativos, um deslocamento é adicionado ao resultado final. A técnica é especialmente útil para melhorar a qualidade de imagens médicas, como imagens de fundo de olho, onde detalhes finos e contrastes são cruciais para identificação de patologias. [16]

As Equações 3 e 4 descrevem matematicamente o funcionamento do filtro descrito. Além disso, a Figura 2 mostra o resultado da aplicação do filtro em uma imagem de fundo de olho.

$$G(x, y) = (\frac{1}{(2 * \pi * \sigma^2)}) * exp(-\frac{(x^2 + y^2)}{(2 * \sigma^2)} \tag{3}$$

Onde:

- G(x, y) é o valor do filtro gaussiano no ponto (x, y)
- σ (sigma) é o desvio padrão da função gaussiana, que controla o grau de suavização

$$Ir = I + K * (I - Ib) \tag{4}$$

Onde:

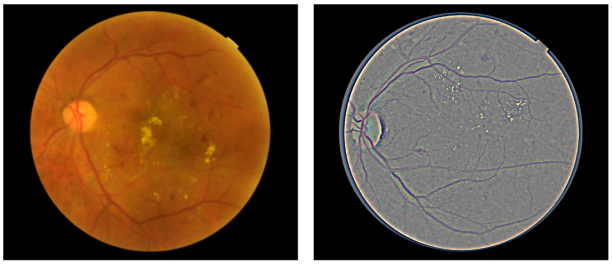
- Ir é a imagem realçada
- I é a imagem original
- K é o fator de realce, um valor escalar que determina o grau de realce aplicado à imagem
- Ib é a imagem borrada

B. Data Augmentation

A técnica de *data augmentation* é uma estratégia importante no campo de aprendizado de máquina, especialmente quando se trabalha com conjuntos de dados limitados ou desbalanceados.

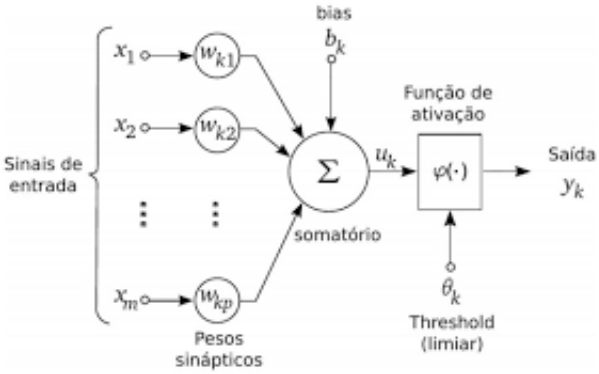
O objetivo principal dessa técnica é aumentar a quantidade de dados disponíveis para treinamento, gerando novas amostras a partir das existentes. Isso é feito por meio de diversas

Figura 2: Aplicação da técnica de mascaramento de nitidez para uma imagem de fundo de retina. À direita a imagem original, à esquerda a imagem com o filtro aplicado



Fonte: Autor

Figura 3: Representação gráfica de um neurônio artificial



Fonte: Juan G. [32].

transformações aplicadas aos dados originais, como rotações, translações, espelhamentos, mudanças de brilho e contraste, entre outras. Essas transformações ajudam a criar um conjunto de dados mais diversificado e robusto, contribuindo para o aumento da capacidade de generalização do modelo e a redução do *overfitting* [17].

Além disso, a *data augmentation* possibilita um melhor desempenho em cenários do mundo real, onde as condições podem variar consideravelmente em relação às imagens presentes no conjunto de treinamento.

C. Redes Neurais e Aprendizado de Máquina

Redes neurais artificiais (RNA) são sistemas computacionais inspirados na estrutura e no funcionamento do cérebro humano [18], que visam simular o processo de aprendizagem e reconhecimento de padrões. Esses sistemas são compostos por elementos chamados neurônios artificiais, interconectados através de sinapses, que transmitem informações entre si.

Um neurônio artificial é a unidade básica de processamento de qualquer arquitetura de rede neural. Ele recebe uma entrada e aplica uma função de ativação para gerar uma saída. A Figura 3 exemplifica esse processo. As funções de ativação são responsáveis por transformar a entrada em uma saída não linear, permitindo que a rede modele funções complexas e aprenda padrões não lineares. Algumas funções de ativação comumente usadas incluem a função sigmóide, a tangente hiperbólica e a função de unidade linear retificada (ReLU). Tais funções devem ser escolhidas de acordo com o problema que deseja ser resolvido [18].

Uma RNA é composta por várias camadas de neurônios de acordo com a complexidade do problema. Essas camadas podem ser classificadas como camada de entrada, camadas ocultas e camada de saída. A camada de entrada recebe os dados de entrada, as camadas ocultas realizam o processamento dos dados e a camada de saída gera os resultados.

O aprendizado de RNAs ocorre através de um processo de ajuste dos pesos das conexões entre os neurônios, de modo a minimizar a diferença entre as saídas geradas pela rede e os valores esperados. Esse processo é conhecido como treinamento, e envolve a apresentação de um conjunto de dados de treinamento à rede e a atualização dos pesos com base no erro calculado.

D. Redes Neurais Convolucionais

Em 1989 no artigo de LeCun et al [19] foi proposto a ideia de redes neurais convolucionais (CNNs). A ideia das CNN foi inspirada a partir de estudos de Hubel et al [20] sobre o córtex visual dos gatos, que mostrou que o cérebro processa a informação visual em hierarquias de características locais. As CNNs se mostraram eficazes na resolução de problemas de classificação já que são eficientes no reconhecimento de padrões e na extração de características hierárquicas a partir de dados de entrada.

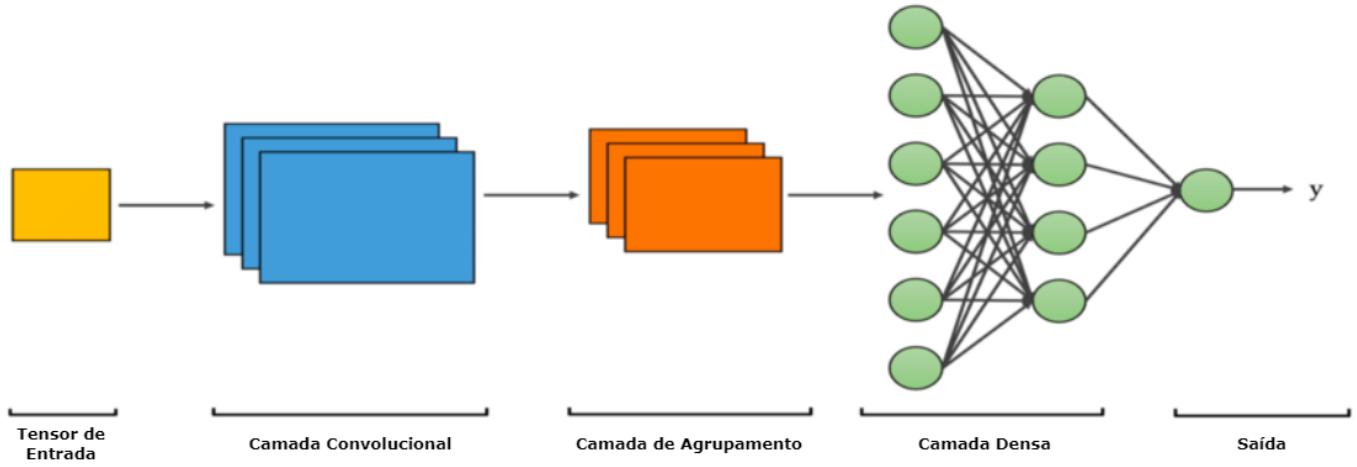
A arquitetura de uma CNN é normalmente composta por três tipos de camadas básicas chamadas de camada de convolução, agrupamento e rede totalmente conectada. A Figura 4 mostra uma arquitetura de CNN genérica.

Na camada convolucional são aplicados filtros, também conhecidos como núcleos ou *kernels*, que são pequenas matrizes de pesos aplicadas em toda a extensão da imagem. O resultado dessa operação é um mapa de características que representam a resposta da imagem ao filtro aplicado, permitindo que a rede neural aprenda a filtrar informações relevantes para a decisão do modelo como estruturas, cores, arestas e outras características.

As camadas de agrupamento reduzem a dimensionalidade dos mapas de características agrupando e resumindo as informações espaciais através de técnicas como o *max-pooling*, que seleciona o valor máximo de uma região ou o *average-pooling*, que calcula a média dos valores de uma vizinhança. Dessa forma tem-se uma diminuição da quantidade de parâmetros da rede e consequentemente o custo computacional do modelo [21]. A Figura 5 exemplifica o funcionamento da camada de agrupamento.

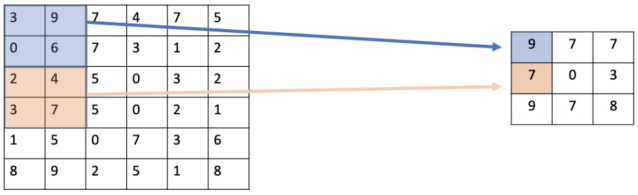
Por fim, as camadas totalmente conectadas, também conhecidas como camada densa, são usadas no final da arquitetura para transformar as informações extraídas pela rede anterior

Figura 4: Representação gráfica de uma CNN genérica.



Fonte: Shakudo [30].

Figura 5: Exemplo de uma camada de agrupamento utilizando a técnica de max-pooling



Fonte: Adaptado [31].

em um vetor de probabilidade usado para uma classificação final. Dessa forma a rede se torna capaz de aprender relações não-lineares e complexas entre as características extraídas [21].

Em uma arquitetura funcional é intercalado diversas camadas convolucionais com camadas de agrupamento a fim de conseguir generalizar toda a complexidade do dado de entrada e no final, uma rede densa com diversas camadas ocultas para conseguir realizar a classificação dado as inúmeras características extraídas [21].

E. TensorFlow e Keras

Segundo a documentação do TensorFlow [22]

> TensorFlow é uma interface para expressão de algoritmos de aprendizado de máquina e sua implementação para execução desses algoritmos. Uma computação expressada usando o TensorFlow pode ser executada com pouco ou nenhuma alteração em uma ampla variedade de sistemas heterogêneos, desde de dispositivos móveis como telefones e *tablets* até sistemas distribuídos de larga escala com centenas de máquinas e milhares de dispositivos computacionais como placas gráficas GPU. O sistema é flexível e pode ser usado para expressar uma ampla variedade de algoritmos, incluindo os de treinamento e inferência para modelos de redes neurais profundas, e tem sido utilizado para condução de pesquisa e para implementação de sistemas de aprendizado de máquina em produção em mais de uma dúzia de áreas das ciências da computação e outros campos, que incluem reconhecimento de fala, visão computacional, robótica, recuperação de informação, processamento de linguagem natural, extração de informação geográfica e descoberta computacional de drogas farmacológicas.

Já o Keras é uma API de alto nível para a construção e treinamento de modelos de aprendizado profundo, escrita em *Python* e executada em cima do TensorFlow. Keras foi desenvolvido com foco na simplicidade e facilidade de uso permitindo o desenvolvimento de protótipos e experimentos de forma ágil.

O TensorFlow e o Keras em conjunto são uma ótima combinação de ferramentas para desenvolvimento e teste de modelos de inteligência artificial e aprendizado de máquina.

F. U-Net

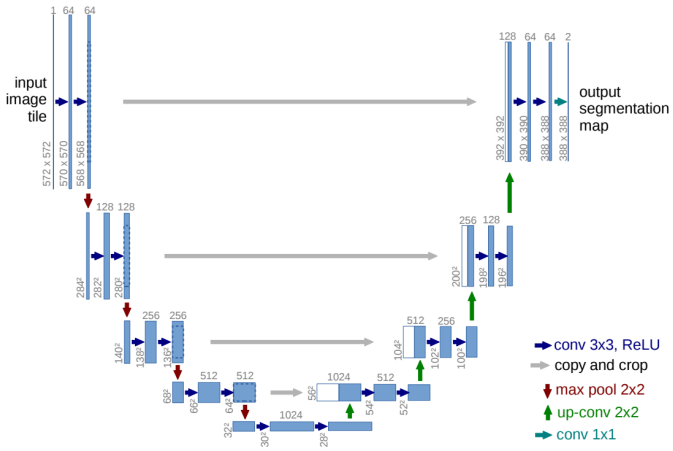
Usualmente CNNs são construídas para realizar classificações onde, a partir de uma imagem, é gerada uma classificação indicando se a imagem pertence a classe gato ou cachorro, por exemplo. Entretanto para algumas aplicações é necessário que a saída do modelo seja uma imagem com anotações de estruturas importantes para uma posterior análise. Para aplicações em imagens médicas esse cenário se repete com bastante frequência, principalmente ao analisar estruturas biológicas como células cancerígenas, tumores cerebrais, segmentação de veias e vasos sanguíneos a partir de imagens de exame médico.

Com essa problemática, alguns autores tentaram desenvolver arquiteturas de CNNs que fossem efetivas na resolução de tais problemas. Ciresan et al [23] por exemplo desenvolveu uma arquitetura que tenta realizar a classificação de cada pixel em uma imagem provendo como entrada para o modelo apenas a região de interesse, permitindo que a rede neural se concentre nas características locais da imagem e aprenda a classificar cada pixel individualmente.

A topologia proposta por Ciresan et al [23] funcionou bem e venceu um desafio de segmentação EM no ISBI 2012 com uma margem significativa. Entretanto essa abordagem tem duas desvantagens, segundo Ronnenberg et al [5] A primeira é que a rede acaba se tornando bastante lenta e onerosa, dado que a rede precisa rodar individualmente para cada região de uma mesma imagem, causando redundância devido à sobreposição das vizinhanças. Em segundo lugar existe uma perda de precisão da localização em função do contexto. Em outras palavras, ao treinar passando uma grande região da imagem são necessárias mais camadas de agrupamento, o que reduz a precisão da localização, enquanto pequenas regiões permitem que a rede aprenda com muito pouco contexto.[5]

Ronnenberg et al [5] propõe uma arquitetura um tanto quanto elegante que é capaz de realizar segmentações mais precisas com muito menos dados para treinamento, a chamada rede U-NET, nomeada assim por conta do seu formato em U, como mostra a Figura 6.

Figura 6: Representação gráfica de uma rede U-Net



Fonte: Ronnenberg et al

A rede U-NET é composta por uma etapa de contração e outra de expansão. A parte de contração contém camadas de convolução e agrupamento que reduzem as dimensões espaciais das imagens, enquanto aprendem representações hierárquicas e abstratas. A parte de expansão usa camadas de convolução transposta para reconstruir progressivamente as dimensões espaciais e gerar a máscara de segmentação final. Além disso, a rede inclui a concatenação entre as respectivas fases, ajudando a preservar as informações de localização espacial e melhorar a capacidade da rede de aprender a segmentar as estruturas [5].

Essa arquitetura permite que a rede alcance resultados de segmentação altamente precisos em diversos tipos de imagens, mesmo com poucos dados de treinamento.

G. *Inception*

No artigo de Szegedy et al [24] os autores propõem a primeira versão da arquitetura *Inception*, uma arquitetura cujo objetivo foi aumentar a profundidade e a largura das CNNs sem aumentar significativamente o número de parâmetros e a complexidade computacional do modelo.

A arquitetura surgiu no contexto do desafio ImageNet Large Scale Visual Recognition Challenge de 2014 e definiu um novo estado da arte, tendo sida campeã do desafio com 12 vezes menos parâmetros que a sucessora e um aumento significativo da acurácia nos problemas de classificação.[24]

O funcionamento da rede *Inception* baseia-se nos módulos *inception*, que são blocos de construção da arquitetura. Esses módulos combinam camadas de convolução de diferentes tamanhos e camadas de agrupamento em paralelo dentro de um único bloco. Isso permite que a rede extraia características em diferentes escalas da imagem.

Na Figura 7, por exemplo, são apresentadas três imagens de cães em diferentes escalas e posições. Na primeira imagem, o cão ocupa a maior parte do espaço, ocupando uma área significativa do quadro. Na segunda imagem, o cão está posicionado no centro e ocupa uma porção menor em relação à primeira. Por fim, na terceira imagem, o cão ocupa apenas uma pequena fração do quadro total. A arquitetura *Inception* é capaz de identificar o cão em todas essas variações de escala e posição, demonstrando sua versatilidade e capacidade de reconhecer objetos de interesse, independentemente de sua localização na imagem.

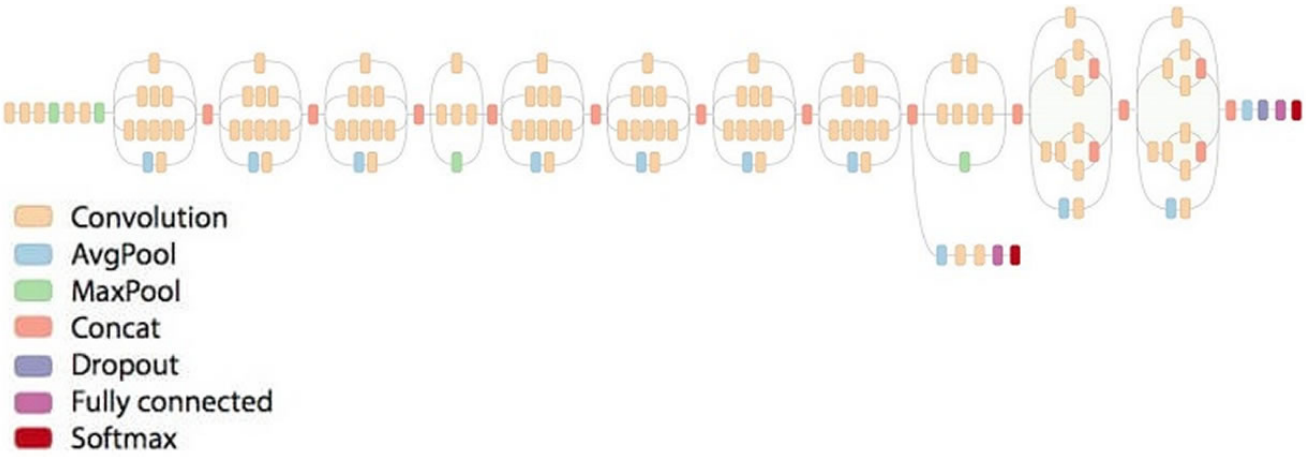
Figura 7: Cachorros em diferentes posições e escalas

Fonte: Bharath Raj [29].

Em resumo, a rede *Inception* é uma arquitetura de CNN que utiliza módulos *inception* para aprender recursos em várias escalas e abstrações. Essa abordagem tem sido eficaz em várias tarefas de visão computacional, incluindo classificação e detecção de objetos [24].

Ao longo dos anos, novas versões da arquitetura foram propostas pelos autores, sempre focadas em aumentar a profundidade da rede sem aumentar a quantidade de parâmetros da mesma. A Figura 8 mostra a arquitetura da rede *Inception*.

Figura 8: Representação gráfica da rede *Inception* V2



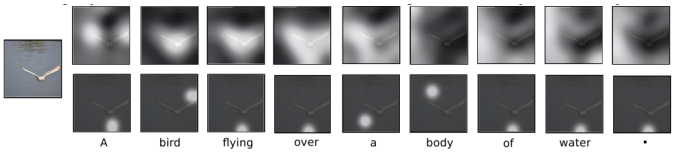
Fonte: Liu, Y., Jain, A., Vasconcelos, N. [28]

H. Mecanismo de Atenção

A primeira aparição do mecanismo de atenção aconteceu em um artigo de Vaswani et al [25] para resolver problemas relacionados à processamento de linguagem natural e traduções. A proposta se mostrou bastante eficiente, já que passou a permitir, ao modelo, olhar diferentes posições de entrada simultaneamente, possibilitando o aprendizado de representações contextuais mesmo entre sentenças grandes.[25]

Em seguida, Kelvin Xu et al [26] publicou um estudo onde mecanismos de atenção foram aplicados em imagens para gerar legendas das imagens, descrevendo-as. O modelo de atenção nesse contexto de uso, permite que o modelo se concentre seletivamente em diferentes partes da imagem ao gerar palavras para a legenda, tornando-o hábil a entender contextos. A Figura 9 exemplifica o comportamento do modelo de atenção para esse caso de uso.

Figura 9: Visualização da atenção gerada ao criar a legenda para uma imagem



Fonte: Kelvin Xu et al [26]

Por fim, em um trabalho publicado por Woo et al [27] foi proposto um modelo de mecanismo de atenção chamado *Convolutional Block Attention* Module (CBAM). Essa arquitetura recebe um mapa de características previamente geradas por outro modelo, como por exemplo o *Inception*, permitindo que o modelo aprenda a ponderar a importância espacial e de cada canal da imagem de entrada.

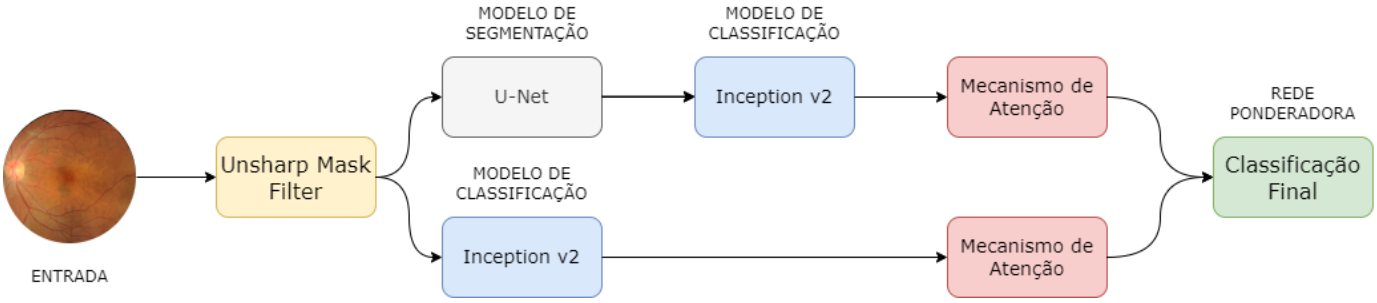
De forma geral, o modelo de atenção aplica uma série de camadas convolucionais com diferentes números de filtros que produz um mapa de atenção. Em seguida, o modelo de atenção é multiplicado ponto a ponto pelas características de entrada, gerando um mapa de características ponderadas. Por fim, é calculada uma média espacial do mapa de atenção que é utilizada em uma camada densa para gerar a classificação final [27].

IV. METODOLOGIA

Nesta seção será descrita a metodologia utilizada para desenvolvimento do algoritmo de classificação de retinopatia diabética e seus módulos. O diagrama na Figura 10 apresenta o fluxo final, resultado dos experimentos realizados para conclusão deste artigo.

Os dados utilizados para o treinamento do modelo são imagens de fundo de olho, também conhecidas como imagens

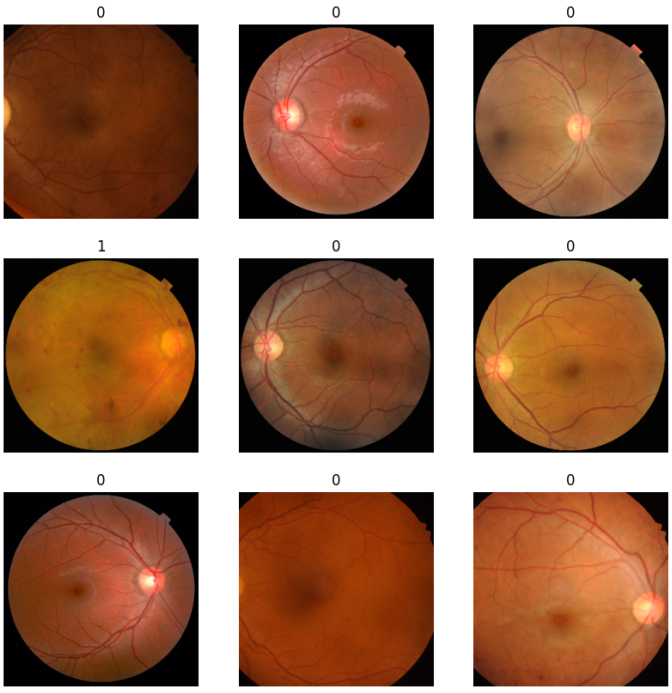
Figura 10: Representação gráfica do modelo proposto



Fonte: Autor

de fundoscopia ou retinografia. Essas imagens são utilizadas por oftalmologista e profissionais da saúde para diagnosticar e tratar várias doenças que afetam a retina ou se manifestam através dela. De maneira geral, essas imagens são fotografias da retina, a camada sensível à luz na parte posterior do olho. A Figura 11 exibe uma amostra do conjunto dos dados utilizados neste estudo.

Figura 11: Amostra do conjunto de dados utilizados para o treinamento dos modelos de classificação e seus respectivos rótulos, onde 1 é a presença da RD e 0 uma retina saudável



Fonte: Autor

Para o modelo de segmentação, o conjunto de dados contém, além da imagem de fundo de olho, imagens com as respectivas extrações de veias e artérias realizadas manualmente por especialistas que é usada como rótulo para treinamento. A Figura 12 exibe uma amostra desses dados.

Figura 12: Imagem de fundo de retina à esquerda e segmentação de veias e artérias realizadas por um especialista à direita

Fonte: Autor

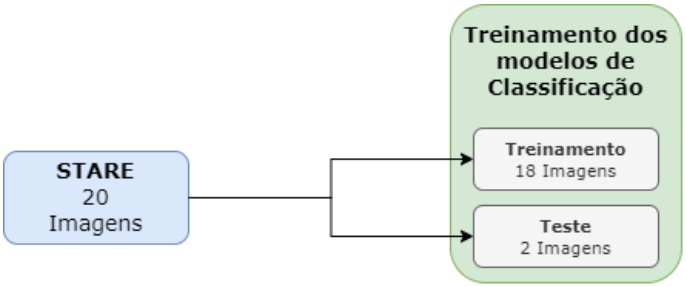
O conjunto de dados empregado no treinamento do modelo foi elaborado a partir da combinação de três conjuntos de dados públicos distintos. O primeiro conjunto, denominado *Automated Retinal Image Analysis* (ARIA) [7]. Esse conjunto consiste em 120 imagens de fundo de olho com resolução de 768 x 576 pixels no formato tiff, divididas em duas categorias: controle (imagens de fundo de retina sem indícios de doença) e retinopatia diabética.

O segundo conjunto de dados utilizado no experimento foi o *Retinal Fundus Multi-Disease Image Dataset* (RFMiD) [8] que é composto por 3200 imagens de fundo de olho e ao todo contém 46 classes distintas de patologias. As imagens são do formato PNG e têm resolução de 2144 x 1424 pixels.

Por fim, foi utilizado o *dataset Structured Analysis of the* Retina (STARE) [9] que é composto por 400 imagens de fundo de olho no formato ppm com resolução de 600 x 600 pixels. As imagens foram rotuladas em 39 classes distintas referentes a diferentes patologias e condições médicas. Além disso, o *dataset* contém 20 imagens de fundo de retina com a segmentação de veias e artérias feitas por profissionais experientes da área, a Figura 13 ilustra o conjunto de dados utilizado para treinamento da rede de segmentação e a Figura 14 o conjunto de dados utilizado para os modelos de classificação.

Para construção do conjunto de dados final foi desenvolvido um algoritmo para separação dos dados relevantes em suas

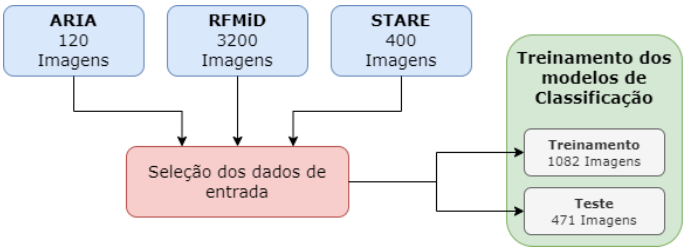
Figura 13: Diagrama lógico da seleção dos dados para o modelo de segmentação



Fonte: Autor

respectivas categorias. Como o presente trabalho tem como objetivo a classificação da RD, foram selecionadas apenas as imagens de classe normal (sem patologia) e RD.

Figura 14: Diagrama lógico da seleção dos dados para treinamento dos modelos de classificação



Fonte: Autor

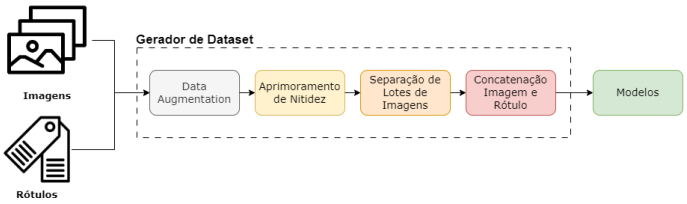
O conjunto de dados final é composto por 1533 arquivos de duas classes representando a ausência ou não da retinopatia diabética. Desse total 1082 imagens foram selecionadas aleatoriamente para compor o conjunto de treino e 471 para teste e validação do modelo.

Todo o projeto foi desenvolvido utilizando a linguagem de programação *Python*. Para desenvolvimento das redes neurais foram utilizadas as bibliotecas *TensorFlow* (TF) e Keras, essas bibliotecas fornecem uma ampla gama de funcionalidades e abstrações para facilitar a construção, treinamento e validação de modelos de aprendizado profundo. Além disso, outras bibliotecas adicionais foram utilizadas de forma auxiliar no projeto, como é o caso das bibliotecas Pandas e OpenCV.

Para treinar e validar o modelo, foi desenvolvido um algoritmo gerador de dados que se encarrega de fornecer ao modelo lotes de imagens e seus rótulos correspondentes de maneira aleatória. A Figura 15 mostra o fluxo lógico do gerador.

Durante a alimentação do modelo, as imagens são submetidas a um processo de aprimoramento de nitidez e aumento de dados (data augmentation). Nessa etapa inicial, são aplicadas transformações aleatórias nas imagens, como inversão vertical ou horizontal, rotações e recortes focados em regiões específicas, utilizando as camadas de ”preprocessing”da biblioteca TensorFlow. A Figura 16 representa um lote

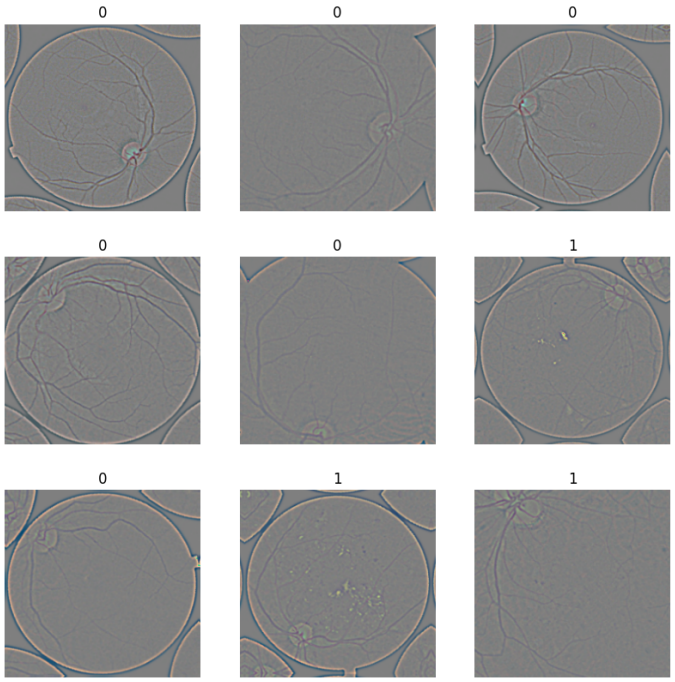
Figura 15: Fluxo lógico do gerador de *dataset*



Fonte: Autor

de imagens após a aplicação das técnicas de data augmentation e aprimoramento de nitidez.

Figura 16: Lote de imagens geradas pelo gerador de *dataset* após a técninca de *Data Augmentation*.



Fonte: Autor

No que diz respeito ao aprimoramento da nitidez, um filtro gaussiano é aplicado para suavizar a imagem, gerando uma versão desfocada. Posteriormente, a imagem desfocada é ponderada e adicionada à imagem original, realçando os detalhes e aumentando a nitidez. Essa técnica, conhecida como *unsharp masking*, é útil para melhorar a qualidade das imagens, o que pode contribuir para a eficácia do modelo no aprendizado de características relevantes a partir dos dados.

O modelo de segmentação U-Net foi construído utilizando as camadas convolucionais, de agrupamento e outras camadas disponíveis nas ferramentas TensorFlow e Keras.

Para as redes *Inception* utilizadas no modelo, foi empregada uma rede pré-treinada disponibilizada pela biblioteca TensorFlow, aproveitando a técnica conhecida como transferência

de aprendizado *(transfer learning)*. Nessa abordagem, uma rede neural previamente treinada em um conjunto de dados, geralmente em uma tarefa de classificação de imagens de grande escala, nesse caso a ImageNet, é adaptada para uma nova finalidade específica. A transferência de aprendizado permite aproveitar os recursos de alto nível aprendidos pela rede, o que geralmente resulta em um melhor desempenho e menor tempo de treinamento do modelo. Isso é especialmente útil quando se trabalha com conjuntos de dados menores ou quando se deseja reduzir o tempo e os recursos computacionais necessários para treinar um modelo do zero.

Para seleção do modelo *Inception* utilizado nos modelos de classificação, foi feito um teste prévio com a *Inception* V2 e a *Inception* V3, com e sem o mecanismo de atenção.

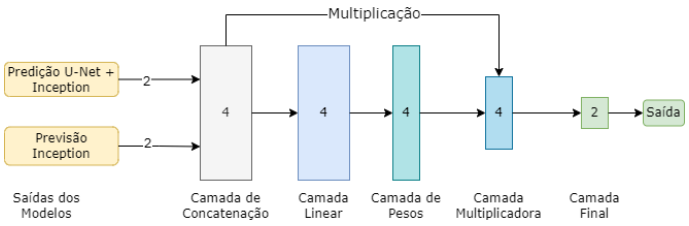
A técnica de *transfer learning* também foi utilizada para realizar a predição nas imagens segmentadas pela rede U-Net. O processo foi realizado em duas etapas. Na primeira etapa,a rede U-Net foi treinada para segmentação, aprendendo a extrair os caminhos das veias e artérias da imagem. Em seguida, os pesos dessa parte de segmentação foram fixados, impedindo que eles fossem alterados durante o treinamento subsequente.

Na segunda etapa, a parte segmentada da rede U-Net foi concatenada ao modelo *Inception* de classificação, criando um novo modelo que combina as capacidades de segmentação da U-Net com as capacidades de classificação da rede *Inception*. Ao utilizar *transfer learning* dessa maneira, o novo modelo se beneficia tanto do conhecimento prévio adquirido pela rede *Inception* quanto das características específicas aprendidas pela U-Net durante a segmentação.

Por fim, a saída dos dois modelos de classificação é concatenada, formando um vetor de tamanho 4, contendo a probabilidade inferida de cada classe (presença ou não da RD) por cada um dos fluxos. Após o treinamento de ambos os fluxos de predição, uma camada densa de classificação é então adicionada ao final da classificação dos dois modelos e através da técnica de *transfer learning*.

A camada final do modelo é composta por uma série de camadas projetadas para ponderar e combinar as inferências obtidas anteriormente. Primeiro, uma camada densa linear com quatro neurônios e função de ativação ReLU é aplicada. Em seguida, outra camada densa com quatro neurônios e função de ativação *softmax* é adicionada. Essas camadas são seguidas por uma camada de multiplicação que atribui pesos distintos a cada inferência. Por fim, uma camada densa adicional com ativação *softmax* é utilizada para gerar a previsão final do modelo. O treinamento desta última etapa tem como objetivo aprender a ponderar adequadamente as inferências anteriores, otimizando assim o desempenho geral do modelo. A Figura 17 exemplifica o funcionamento da última camada.

Figura 17: Representação gráfica da camada ponderadora.



Fonte: Autor

V. RESULTADOS E DISCUSSÕES

Nesta seção, é apresentado a análise dos resultados obtidos ao longo do desenvolvimento do modelo proposto. As análises foram divididas em seções que correspondem aos módulos utilizados no modelo final e as motivações por trás de cada escolha de projeto.

A. Redes de Classificação Inception

Como primeira análise, foi necessário entender quais arquiteturas e configurações de rede de classificação melhor se adéquem aos dados utilizados para a resolução do problema. Assim foi conduzido experimentos que combinaram as arquiteturas *Inception* V2 e *Inception* V3 com e sem os mecanismos de atenção integrados.

A Tabela I, a seguir, apresenta uma comparação das métricas-chave obtidas durante a validação de cada modelo usando o conjunto de dados de validação. As métricas consideradas na análise são acurácia, precisão, sensibilidade e F1 Score.

Tabela I: Métricas geradas a partir da validação dos modelos *Inception* com e sem mecanismo de atenção.

Modelo	Acurácia	Precisão	Sensibilidade	F1 Score
Inception V2	92,6%	98,8%	86,2%	92,1%
Inception V2 + Mecanismo de Atenção	94,4%	95,3%	**93,3%**	94,3%
Inception V3	94,1%	98,8%	89,2%	93,8%
Inception V3 + Mecanismo de Atenção	94,6%	97,8%	91,3%	94,4%

Fonte: Autor

Ao analisar os dados da Tabela 1 é possível notar que ambas as versões da arquitetura *Inception* demonstraram grande capacidade de lidar com o conjunto de dados e gerar classificações satisfatórias para o problema. Entretanto, alguns pontos se destacam.

Em primeiro lugar fica evidente a melhora na acurácia e na sensibilidade causada pela integração do mecanismo de atenção para ambas as versões da arquitetura. A diferença entre a Arquitetura V2 com e sem atenção é de quase 2 p.p para essas métricas. Porém, para métrica de precisão o cenário se inverte e demonstra que o uso do mecanismo de atenção pode prejudicar a precisão final do modelo.

Comparando as arquitetura *Inception* V2 e *Inception* V3 sem a presença do mecanismo de atenção, nota-se que a *Inception* V3 tem um desempenho ligeiramente maior do que a *Inception* V2 mesmo que a precisão se mantenha igual entre os dois modelos.

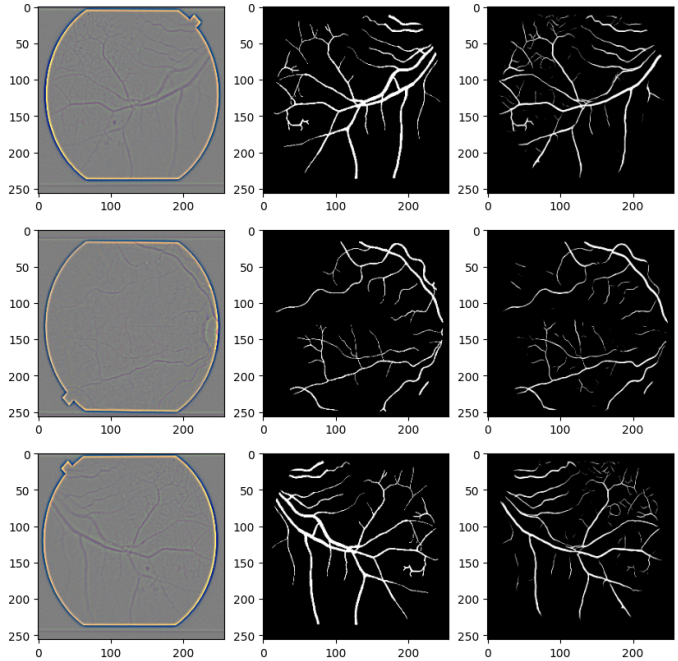
Por fim, nota-se que a sensibilidade para a rede *Inception* V2 com mecanismo de atenção é superior às demais redes. A sensibilidade é especialmente importante em situações em que é crucial minimizar a quantidade de falsos negativos. No caso de diagnósticos médicos, é preferível identificar a maioria dos casos positivos ao invés de classificar como negativo o diagnóstico de uma pessoa com a condição médica.

Logo, a partir das análises apresentadas nessa sub-seção, optou-se por escolher a rede *Inception* V2 com mecanismo de atenção para a rede final, por conta de seu desempenho e principalmente por sua alta sensibilidade e em relação aos outros modelos.

B. Rede de Segmentação U-Net

A CNN U-Net mostrou ter uma arquitetura extremamente poderosa e eficiente para segmentação de imagens. Com apenas 20 imagens rotuladas e uso da técnica de *data augmentation*, foi possível chegar a um ótimo resultado na segmentação de imagens. A Figura 18 a seguir, mostra uma comparação entre as imagem utilizada para alimentar o modelo, a imagem segmentada por especialistas da área e por fim a imagens geradas pelo modelo de segmentação.

Figura 18: Comparação entre entrada, segmentada manualmente e segmentada pelo modelo



Fonte: Autor

Analisando as imagens resultantes do modelo, é possível notar que o modelo conseguiu compreender bem a tarefa de segmentação e gerar caminhos parecidos com aqueles feitos manualmente por especialistas. Entretanto é possível notar alguns problemas na segmentação.

Observando atentamente a imagem gerada, é possível perceber a presença de ruídos inexistentes naquela gerada manualmente. Esses ruídos provavelmente foram causados por má interpretação do modelo em relação à presença de outros componentes da imagem.

Além disso é possível notar que alguns caminhos foram interrompidos na imagem gerada, causando descontinuação das veias e artérias e a espessura da segmentação gerada, por vezes, é menor do que a segmentada manualmente.

C. Rede de Classificação a partir das imagens segmentadas U-Net

Partindo da ideia de que a RD se manifesta na retina como um todo, parte da condição pode ser detectada também pela morfologia das veias e artérias. O modelo *inception* tratado nesta subseção, trata-se de um modelo alimentado a partir das imagens geradas pelo modelo U-Net.

A Tabela II a seguir, mostra a matriz de confusão gerada a partir das predições realizadas pelo modelo utilizando o conjunto de dados de validação.

Tabela II: Matriz de confusão para modelo de classificação a partir das imagens segmentadas

		Valor Real	
Valor Inferido	0	47.2%	16.1%
	1	3.1%	33.7%

Fonte: Autor

A matriz de confusão nos permite chegar a importantes métricas para análise do modelo, exibidas na Tabela **??** a seguir.

Tabela III: Métricas geradas a partir da validação do modelo U-Net + Inception.

Métrica	**Valor**
Acurácia	80,87%
Precisão	91,67%
Sensibilidade	67,69%
F1 Score	77,88%

Fonte: Autor

É possível notar que o desempenho do modelo é bastante baixo quando comparado com os modelos *Inception* alimentado com a imagem bruta. Tal comportamento era esperado, já que a imagem segmentada exclui da imagem características importantes para a detecção da RD como lesões e hemorragias na retina.

Entretanto, a partir da análise das métricas do modelo é possível concluir que a imagem segmentada traz informações que podem ser relevantes para a detecção de RD em um paciente.

Para justificar o uso da rede de classificação segmentada no modelo de predição de RD é necessário entender se existe alguma relação entre os acertos do modelo de classificação utilizando a U-Net e aquele com a imagem bruta. A Tabela IV a seguir mostra a comparação combinada das duas previsões isoladas.

Na Tabela IV em questão, é possível notar que quando combinamos os acertos das previsões feitas isoladamente pelos

Tabela IV: Comparação entre *Inception* e U-Net + *Inception*.

	Inception	
U-Net	**Certo**	**Errado**
Certo	76,98%	3,84%
Errado	17,39%	1,79%

Fonte: Autor

dois modelos, atingimos uma acurácia de 98,2%. Além disso, o modelo com U-Net foi capaz de acertar 68% dos erros causados pela *Inception* V2.

Essa análise mostra que apesar da rede U-Net + *Inception* V2 não ter um bom desempenho geral, ela consegue classificar a RD a partir de características que a *Inception* V2 não consegue, validando a hipótese de que ambas as redes combinadas podem ter um desempenho melhor do que apenas a *Inception* V2.

D. Modelo Final Ponderador

O modelo final, composto pelas duas redes mencionadas anteriormente e uma camada de ponderação adicional no final, apresentou resultados ligeiramente superiores aos da rede *Inception* V2 com mecanismo de atenção.

Inicialmente, desenvolveu-se uma camada ponderada simples, que alcançou resultados idênticos aos da rede *Inception* V2 com mecanismo de atenção. Este resultado foi intrigante, mas também esperado, uma vez que a camada ponderada simples aprendeu a considerar principalmente o modelo *Inception* V2, já que, na maioria das vezes, ele fazia inferências corretas sobre os dados.

Diante do desempenho insatisfatório do método anterior, elaborou-se uma nova rede de ponderação mais robusta, composta por uma camada linear e uma camada de pesos, permitindo que o modelo aprendesse de maneira mais eficiente a ponderar os resultados dos dois modelos.

A Tabela V a seguir apresenta a matriz de confusão do modelo final.

Tabela V: Comparação entre valores inferidos e reais do modelo final.

	Valor Inferido	
Valor Real	**1**	**0**
1	47,2%	2,55%
0	2,3%	48%

Fonte: Autor

A matriz de confusão nos permite chegar às métricas para análise do modelo, exibidas na Tabela VI a seguir.

É possível notar que o modelo final obteve métricas de desempenho superiores aos dos modelos apresentados anteriormente, apresentando um aumento de quase 2 p.p de sensibilidade em relação ao modelo *Inception* V2 com mecanismo de atenção. Além disso, também obteve acurácia e F1 Score superior aos outros modelos.

Com esse resultado, foi possível validar a hipótese de que as duas redes trabalhando em conjunto geram resultados melhores do que de forma isolada, já que cada fluxo observa e extrai características das imagens de forma distintas.

Tabela VI: Métricas geradas a partir da validação do modelo final.

Métrica	**Valor**
Acurácia	95.15%
Precisão	95.36%
Sensibilidade	94.87%
F1 Score	95.12%

Fonte: Autor

E. Comparação com Trabalhos Relacionados

Nos trabalhos utilizados e analisados como referência, apenas a precisão e sensibilidade são fornecidas pelos autores, além disso, não fica claro a arquitetura utilizada por eles nos trabalhos. A Tabela VII mostra uma comparação entre os trabalhos de Li et al [10] e Shah et al [11].

Tabela VII: Comparação de métricas entre o modelo proposto e trabalhos relacionados.

Modelo	**Precisão**	**Sensibilidade**
Modelo Proposto	95.36%	94.87%
Li et al [10]	91.4%	97%
Shah et al [11]	98.5%	99.7%

Fonte: Autor

Ao comparar os resultados obtidos com os de outros pesquisadores, fica evidente que o desempenho do modelo proposto se aproxima consideravelmente dos valores encontrados na literatura, validando portanto a eficácia do modelo proposto, mas também levanta questionamentos sobre como aprimorar ainda mais a arquitetura para obter um desempenho superior.

VI. Conclusões

A rede *Inception* se mostrou bastante eficiente nas tarefas de classificação de imagens, conseguindo realizar classificações assertivas de diferentes tipos e em diferentes contextos. Da mesma forma, o mecanismo de atenção se provou bastante eficiente como uma rede auxiliar no processo de classificação e detecção de padrões em imagens. Foi possível notar a partir dos resultados obtidos que as duas em conjunto formam uma arquitetura poderosa, capaz de realizar tarefas complexas de classificação.

Além disso, a rede U-Net se provou uma arquitetura extremamente eficiente na solução de problemas de segmentação de imagens, atingindo resultados satisfatórios com uma pequena massa de dados. A junção da rede U-Net com a *Inception* se mostrou possível, mas, pela natureza do problema, não obteve ótimos resultados.

Por fim, o modelo proposto se mostrou bastante eficiente na tarefa de classificação da RD, conseguindo atingir boas métricas de acurácia, sensibilidade e precisão. Entretanto, nota-se que existem bastante oportunidades de melhorias para a arquitetura proposta.

A primeira delas seria validar melhores técnicas e parâmetros para a etapa de pré processamento dos dados. Uma opção seria realizar um filtro de desfoque para eliminar as imagens com baixa qualidade do conjunto de dados e consequentemente aumentar a qualidade das imagens utilizadas para treinamento do modelo.

Além disso, pesquisar e descobrir melhores formas de explorar a técnica de data augmentation, através de diferentes tipos de tratamento nas imagens, assim como, a utilização de um conjunto de dados maior e mais robusto, a fim de aumentar o volume de dados utilizados para treinamento e consequentemente melhorar a capacidade de generalização do modelo.

VII. Referências

[1] EISMA, Jessica H; DULLE, Jennifer e; FOR, Patrice e. Current knowledge on diabetic retinopathy from human donor tissues. 2014. 10 f. Departments Of Ophthalmology And Visual Sciences, University Of Michigan, Michigan, 2014.

[2] Tabish SA. Is Diabetes Becoming the Biggest Epidemic of the Twenty-first Century? Int J Health Sci (Qassim). 2007 Jul;1(2):V-VIII. PMID: 21475425; PMCID: PMC3068646.

[3] Wilkinson CP, Ferris FL 3rd, Klein RE, Lee PP, Agardh CD, Davis M, Dills D, Kampik A, Pararajasegaram R, Verdaguer JT; Global Diabetic Retinopathy Project Group. Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales. Ophthalmology. 2003 Sep;110(9):1677-82. doi: 10.1016/S0161-6420(03)00475-5. PMID: 13129861.

[4] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).

[5] Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

[6] Woo, S., Park, J., Lee, J. Y., Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 3-19).

[7] St. Paul's Eye Unit (2016). Automated Retinal Image Analysis (ARIA) dataset. Liverpool, United Kingdom.

[8] Li, F., Chen, H., Liu, Z., Zhang, X., Wu, Z., Yang, Y. (2019). RFMiD: Retinal Fundus Multi-lesion Dataset and Its Application for Lesion Classification. arXiv preprint arXiv:1905.09334.

[9] Hoover, A., Kouznetsova, V., Goldbaum, M. (2000). Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. IEEE Transactions on Medical Imaging, 19(3), 203-210.

[10] Li, Z., Keel, S., Liu, C., He, Y., Meng, W., Scheetz, J., Lee, P. Y., Shaw, J., Ting, D., Wong, T. Y., Taylor, H. R. (2018). An automated grading system for detection of vision-threatening referable diabetic retinopathy on the basis of color fundus photographs. Diabetes Care, 41(12), 2509-2516.

[11] Shah P, Mishra DK, Shanmugam MP, Doshi B, Jayaraj H, Ramanjulu R. Validation of Deep Convolutional Neural Network-based algorithm for detection of diabetic retinopathy - Artificial intelligence versus clinician for screening. Indian J Ophthalmol. 2020 Feb;68(2):398-405. doi: 10.4103/ijo.IJO96619. PMID: 31957737; PMCID: PMC7003578.

[12] Rehman MU, Cho S, Kim JH, Chong KT. BU-Net: Brain Tumor Segmentation Using Modified U-Net Architecture. Electronics. 2020; 9(12):2203. https://doi.org/10.3390/electronics9122203.

[13] X. Xiao, S. Lian, Z. Luo and S. Li, "Weighted Res-UNet for High-Quality Retina Vessel Segmentation,"2018 9th International Conference on Information Technology in Medicine and Education (ITME), Hangzhou, China, 2018, pp. 327-331, doi: 10.1109/ITME.2018.00080.

[14] Schlemper, J., Oktay, O., Schaap, M., Heinrich, M., Kainz, B., Glocker, B., Rueckert, D. (2019). Attention gated networks: Learning to leverage salient regions in medical images. Medical Image Analysis, 53, 197-207. https://doi.org/10.1016/j.media.2019.01.012.

[15] García, S., Luengo, J., Herrera, F. (2015). Data preprocessing in data mining. Springer. https://doi.org/10.1007/978-3-319-10247-4

[16] Peli, E. (1990). Contrast in complex images. Journal of the Optical Society of America A, 7(10), 2032-2040. doi:10.1364/JOSAA.7.002032

[17] Perez, L., Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. arXiv preprint arXiv:1712.04621.

[18] AGARAP, Abien Fred. Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375, 2018.

[19] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition,"in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324,

Nov. 1998, doi: 10.1109/5.726791.

[20] HUBEL DH, WIESEL TN. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J Physiol. 1962 Jan;160(1):106-54. doi: 10.1113/jphysiol.1962.sp006837. PMID: 14449617; PMCID: PMC1359523.

[21] Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (pp. 1097-1105).

[22] Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (pp. 1097-1105). Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu e Xiaoqiang Zheng: TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. https://www.tensorflow.org/, Software available from tensorflow.org

[23] Ciresan, D.C., Gambardella, L.M., Giusti, A., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: NIPS. pp. 2852–2860 (2012)

[24] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2014). Going deeper with convolutions. arXiv preprint arXiv:1409.4842.

[25] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I. (2017). Attention Is All You Need. arXiv preprint arXiv:1706.03762.

[26] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R. amp; Bengio, Y.. (2015). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. ¡i¿Proceedings of the 32nd International Conference on Machine Learning¡/i¿, in ¡i¿Proceedings of Machine Learning Research¡/i¿ 37:2048-2057 Available from https://proceedings.mlr.press/v37/xuc15.html.

[27] Woo, S., Park, J., Lee, J.-Y., Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. arXiv preprint arXiv:1807.06521.

[28] Liu, Y., Jain, A., Vasconcelos, N. (2017). Arquitetura de convolução profunda Inception-v3 para classificar leucemia mieloide aguda/linfoblástica em imagens microscópicas de sangue. Intel. Recuperado de https://www.intel.cn/content/www/cn/zh/developer/articles/technical/inceptv3-deep-convolutional-architecture-for-classifying-acute-myeloidlymphoblastic.html

[29] Bharath Raj Recuperado de https://towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202

[30] Shakudo. (2023, March 16). The 7 Deep Learning Algorithms to Get Started with in 2023. Adaptado de https://www.shakudo.io/blog/the-7-deep-learning-algorithms-to-get-started-with-in-2023 //

[31] Retrieved from https://programmathically.com/what-is-pooling-in-aconvolutional-neural-network-cnn-pooling-layers-explained/ //

[32] Juan G. (2021). Aula 2 - Introdução às RNA - Perceptron, Adaline. Centro de Energias Alternativas e Renováveis (CEAR) - Universidade Federal da Paraíba. Retrieved from http://www.cear.ufpb.br/juan/wp-content/uploads/2021/08/Aula-2-Introdu%C3%A7%C3%A3o-as-RNA-Perceptron-Adaline-2021-2.pdf